本科生毕业论文（设计）

题目：大语言模型示例泛化的自旋模型

姓　　名　李宇豪

学　　号　21305412

院　　系　物理学院

专　　业　光电信息科学与工程

指导教师　黄海平 教授

2025 年 5 月 13 日

大语言模型示例泛化的自旋模型

Spin Model of In-Context Generalization in Large Language Models

姓　　名　李宇豪

学　　号　21305412

院　　系　物理学院

专　　业　光电信息科学与工程

指导教师　黄海平 教授

2025 年 5 月 13 日

学术诚信声明

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文（设计）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本论文（设计）的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文（设计）的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日　　期：　　年　　月　　日

摘 要

大语言模型展现出令人惊讶的上下文学习能力，能够仅依靠输入提示与少量示例，在无需额外训练的情况下完成从未见过的复杂任务，这与传统的监督学习形成鲜明对比。如何为这一经验现象提供理论解释，并将其与理论物理建立联系，是一个具有挑战性且尚未解决的问题。

本文研究了一种简化的 Transformer 结构，并将其映射为一个自旋玻璃模型。在该映射中，取值为实数的自旋表示 Transformer 的权重参数，相互作用和外场刻画了任务结构和数据内在的无序性。我们分析了任务形式和数据分布对相互作用和外场的统计性质的影响，使用平均场理论求解了该模型，并且通过近似消息传递方程快速寻找模型的基态。

我们的自旋模型揭示了在预训练过程中 Transformer 的权重参数之间如何相互作用。我们的理论表明，对于具有示例的训练方案，随着任务多样性的增加，会自然地催生出模型的上下文学习能力。这是因为任务多样性会改变自旋模型的能量景观，使其出现一个全局最小值。在玻尔兹曼分布下，模型很容易收敛到这个基态，从而使得预训练 Transformer 在新颖的提示设定下具备预测能力。

我们提出的模型首次将 Transformer 与统计物理模型结合起来，并且可以在理论上进行求解，为理解大型语言模型诸多令人关注的特性提供了一个有前景的研究方向。

关键词： 大语言模型，上下文学习，统计物理，自旋玻璃模型，平均场理论

ABSTRACT

Large language models show a surprising ability to learn in context, being able to complete complex unseen tasks with only input prompts and a few examples, yet without additional training, in stark contrast to traditional supervised learning. Providing a mechanistic interpretation and linking the empirical phenomenon to physics are thus challenging and remain unsolved.

In this thesis, we study a simplified yet expressive transformer structure with linear attention and map it to a spin glass model. In this mapping, real-valued spins represent the weight parameters of the transformer, while the couplings and fields characterize the task structure and intrinsic disorder in data. We analyze how the task form and data distribution affect the statistical properties of couplings and fields, solve the model using mean-field theory, and quickly find the ground state of the model through approximate message-passing equations.

Our spin model reveals how the weight parameters of the transformer interact with each other during pre-training. Our theory shows that for training schemes with instance, increasing task diversity naturally leads to the emergence of in-context learning. This is because task diversity changes the energy landscape of the spin model, leading to a global minimum. Under the Boltzmann distribution, the model easily converges to this ground state, thus enabling the pre-trained transformer to exhibit predictive power in novel prompt settings.

Our proposed model is the first to combine transformers with statistical physics models and can be solved theoretically, providing a promising avenue for understanding many intriguing properties of large language models.

Keywords: Large language models, in-context learning, statistical physics, spin glass model, mean-field theory

目　录

1　绪论　1

1.1　选题背景　1

1.2　相关工作　3

1.3　本文结构　4

2　大语言模型的基本原理　5

2.1　Transformer 模型的结构　5

2.2　大语言模型的构建流程　9

2.3　提示学习方法　13

3　自旋玻璃模型的平均场理论　15

3.1　自旋玻璃模型简介　15

3.2　复本方法　17

3.3　空腔方法　20

3.4　复本对称性破缺　23

4　线性注意力机制的自旋模型　29

4.1　线性注意力模型　29

4.2　自旋玻璃模型映射　30

4.3　相互作用和外场的统计性质　31

4.4　基于平均场理论的解决方案　33

5　结果与讨论　39

5.1　线性回归任务的最优解　39

5.2　模型的凸性与能量景观　41

5.3　近似消息传递算法的结果　43

5.4　在更复杂的任务和模型上的结果　43

5.5　讨论　46

6　总结与展望　49

参考文献　51

本科期间取得的研究成果　61

致谢　63

1 绪论

1.1 选题背景

人工智能是当今科技领域的热门话题，其应用已经渗透到我们生活的方方面面。自然语言处理（Natural Language Processing，NLP）是人工智能领域的一个重要分支，它致力于让计算机能够理解、处理和生成自然语言。然而，人类语言复杂多变，语法和语义极为丰富，并且常常具有模糊性和多义性，这使得自然语言建模成为一项极具挑战性的任务。

早期的语言模型主要是基于统计学习方法，包括隐马尔可夫模型（HMM）[1]、n-gram 模型[2]、最大熵模型[3]和条件随机场（CRF）[4]等。这些方法通过对大量文本数据进行统计分析，实现对文本的建模和生成。虽然这些方法在一定程度上能够捕捉到语言的结构和规律，但由于它们通常依赖于手工设计的特征和规则，泛化能力有限。在深度学习兴起以后，研究者们尝试使用神经网络来建模文本序列的生成，包括词向量（word2vec）模型[5-6]、基于循环神经网络（RNN）的模型[7]和序列到序列（seq2seq）模型[8]等，这些方法能够捕捉到更丰富的语义信息，但仍然难以有效地捕捉长距离的上下文信息，导致模型的表达能力受限。

2017 年，A. Vaswani 等人提出 Transformer 模型[9]，引入自注意力机制，替代了传统循环结构（比如 RNN）的顺序处理方式，能够更高效地捕捉序列中的长距离关联和上下文依赖，并且还能够实现并行计算，显著提升模型的训练和推理效率。此后，基于 Transformer 的各种预训练模型，如 BERT[10]、GPT[11-12]、T5[13]等，相继问世，形成了“预训练—微调”范式的核心框架：模型通过在大规模语料上进行预训练获得通用语言知识，然后通过少量任务特定数据进行微调，以适应各种下游任务[14]。这一范式在文本分类、问答、翻译、生成等任务上显著提高了模型的性能和迁移能力。

随着计算资源和数据规模的提升，研究者发现模型性能与参数规模、数据量、计算量之间存在一个缩放定律（Scaling Law）[15]。这一规律为语言模型设计提供了明确的工程指导：只要具备足够的训练数据和计算资源，就有可能通过简单的扩展规模获得更强的语言能力。这种方法论的转变，促使了大语言模型（Large Language Model，LLMs）的迅速崛起。以 OpenAI 的 GPT 系列为代表，参数规模从数亿扩展到千亿乃至万亿级别，大幅提升了模型的语言理解与生成能力，在多种任务上取得了接近甚至超越人类水平的表现[16]。

在提出 GPT-3 的同时，研究者们还发现了大语言模型的一个重要特性：上下文学习（In-context Learning，ICL）[17]。与传统的微调训练不同，上下文学习不需要对模型权重进行更新，而是通过设计提示词，在推理阶段加入少量示例，使模型能够直接根据上下文来完成未经过专门训练的复杂任务。这种交互方式具有高度的灵活性，为大语言模型的应用开辟了新的途径：用户不再需要为每个任务训练一个新的模型，仅仅通过设计不同的提示词就可以完成多样化的任务。同时，这也启发了诸如思维链（Chain of Thought，CoT）[18]等新的研究方向。

尽管大语言模型在实践中取得了巨大的成功，但是其工作原理和背后的机制仍然不够清晰。

首先，大语言模型的预训练目标相对简单，通常采用基于自回归语言建模和似然估计，并没有显式引入语义理解、逻辑推理或任务泛化的监督信号。然而，训练后的模型却在多种复杂的下游任务中实现良好的泛化。这种从简单目标出发到涌现出复杂能力的转变机制，目前尚缺乏明确解释。

另外，在大语言模型的训练与推理阶段，模型参数的变化如何影响知识的存储、更新和表达，同样是一个亟待研究的重要问题。传统机器学习的主流观念认为，模型的参数是知识的主要存储形式，模型的训练过程相当于将知识以某种方式编码到参数空间中。然而，上下文学习的机制更多依赖于输入空间的信息，而非参数空间中的梯度优化，这颠覆了传统机器学习关于“学习依赖于参数更新”的基本思想，并引发了对知识存储和表达机制的重新思考。

更为重要的是，尽管已有部分研究尝试从优化理论、算法模拟等角度来研究上下文学习的机制，但从统计物理视角出发对上下文学习的研究尚属空白领域，尤其是缺乏上下文学习与物理模型之间的联系。自旋模型是统计物理中一个重要的研究对象，具有丰富的相变现象和复杂的能量景观，常用于刻画具有高度复杂结构的系统，这为我们从统计物理视角理解大语言模型的机制提供了可能。自旋玻璃的平均场理论为我们提供了一种有效的理论工具来研究复杂系统的行为，从中发展出的诸多方法在理论物理和统计学习中得到了广泛应用。

本文尝试将 Transformer 与自旋模型结合，借助平均场理论深入探讨大语言模型的泛化能力和上下文学习机制。我们希望通过这种方法填补当前的研究空白，揭示大语言模型在上下文中学习的潜在机制，为理解其泛化能力提供新的视角。

1.2 相关工作

最近的许多研究表明大语言模型能够通过上下文学习执行一系列复杂任务，比如代码生成、数学推理等，这些强大的能力引起了学术界广泛的研究兴趣[19-20]。尽管目前有一些工作探索 Transformer 以外的架构是否存在上下文学习能力[21-24]，但主流的研究方向仍然聚焦在基于自注意力机制的架构上。

在实验方面，研究者们发现大语言模型的某些能力只有在模型的规模达到一定的阈值后才会出现，即所谓的涌现[25]，而微调策略、训练数据的分布特性等因素会对上下文学习能力的涌现产生关键影响[26-27]。令人惊奇的是，有研究表明上下文学习的性能更依赖于示例中“问题—答案”的形式，而非示例答案的正确性[28]。为此研究者们尝试将上下文学习分为不同的阶段，比如任务识别和任务学习[29]，进行研究。此外还有一些研究通过设定不同的任务来评估大语言模型的学习范围[30]。

在理论方面，一类观点认为，尽管此时不需要训练，但模型在推理过程中隐式地执行了梯度下降算法[31-34]。当然，也有一些研究者认为这是由于设定过于简化导致的，真实任务中上下文学习与梯度下降仍然存在差别[35-36]。另一类观点则认为上下文学习是一种隐式的贝叶斯推断过程，模型通过上下文利用先验知识进行预测[37-40]。此外，还有很多研究尝试从信息论、学习算法、梯度流动力学等方面来解释上下文学习现象，为理解其机制提供了多样化视角[41-44]。

尽管自然语言任务中的上下文学习令人印象深刻并且具备实用价值，但是由于实际任务的数据中存在高度复杂的结构与分布偏差，无法进行精确的变量控制。因此，最近的许多研究尝试探索简单的小模型和参数可控的合成任务中的上下文学习，特别是线性模型[45-47]、分类任务[48-49]和回归任务[50-51]。这类工作中有一大部分都在尝试利用数学和物理工具完全从理论上解析大语言模型的动力学和泛化性能[47,49,51-53]。与我们的工作密切相关的是聚焦于预训练任务多样性[42,50,52,54]和单层线性注意力机制[31,45-46,52]的研究，但是目前仍然缺乏直接将上下文学习与统计物理模型结合在一起的工作，这就是本文的主要贡献所在。

1.3 本文结构

本文剩余部分的安排如下：

- 第二章介绍 Transformer 模型的结构和工作原理，然后介绍完整的大语言模型的构建流程，以及大语言模型的提示学习方法，特别是上下文学习；
- 第三章介绍自旋玻璃的平均场理论，以球面 p 自旋模型为例，详细介绍如何使用复本方法和空腔方法求解自旋玻璃模型，然后进一步介绍一阶复本对称性破缺及其对应的空腔方程；
- 第四章介绍我们主要研究的线性注意力机制，说明如何将 Transformer 映射为一个自旋玻璃模型，分析上下文学习任务和预训练数据集如何影响相互作用和外场的统计性质，然后利用平均场理论求解这个自旋模型的基态；
- 第五章展示了我们的结果，对理论和实验结果进行了比较，对模型的最优解、能量景观等平衡态性质进行了分析，探究我们的理论在更复杂的任务和模型上的推广，并且对一些关键问题进行了讨论；
- 第六章是总结，回顾了本文的主要工作和贡献，并展望了未来的研究方向。

2 大语言模型的基本原理

2.1 Transformer 模型的结构

正如在绪论中所言，自从 GPT 等模型取得成功之后，Transformer 替代了循环神经网络和卷积神经网络，成为了当前大语言模型的标配。本节我们首先介绍一下 Transformer 模型的基本结构。

如图 1 所示，一个完整的 Transformer 可以大概分为输入模块、编码器、解码器以及输出模块四部分。输入模块负责将输入文本及其上下文位置信息转换为一个向量表示；编码器负责为每个输入向量构造对应的语义表示，即负责理解输入文本；解码器需要根据输入信息，结合编码器输出的语义表示，生成相应的输出序列；输出模块负责将解码器的输出转换为最终的预测结果。

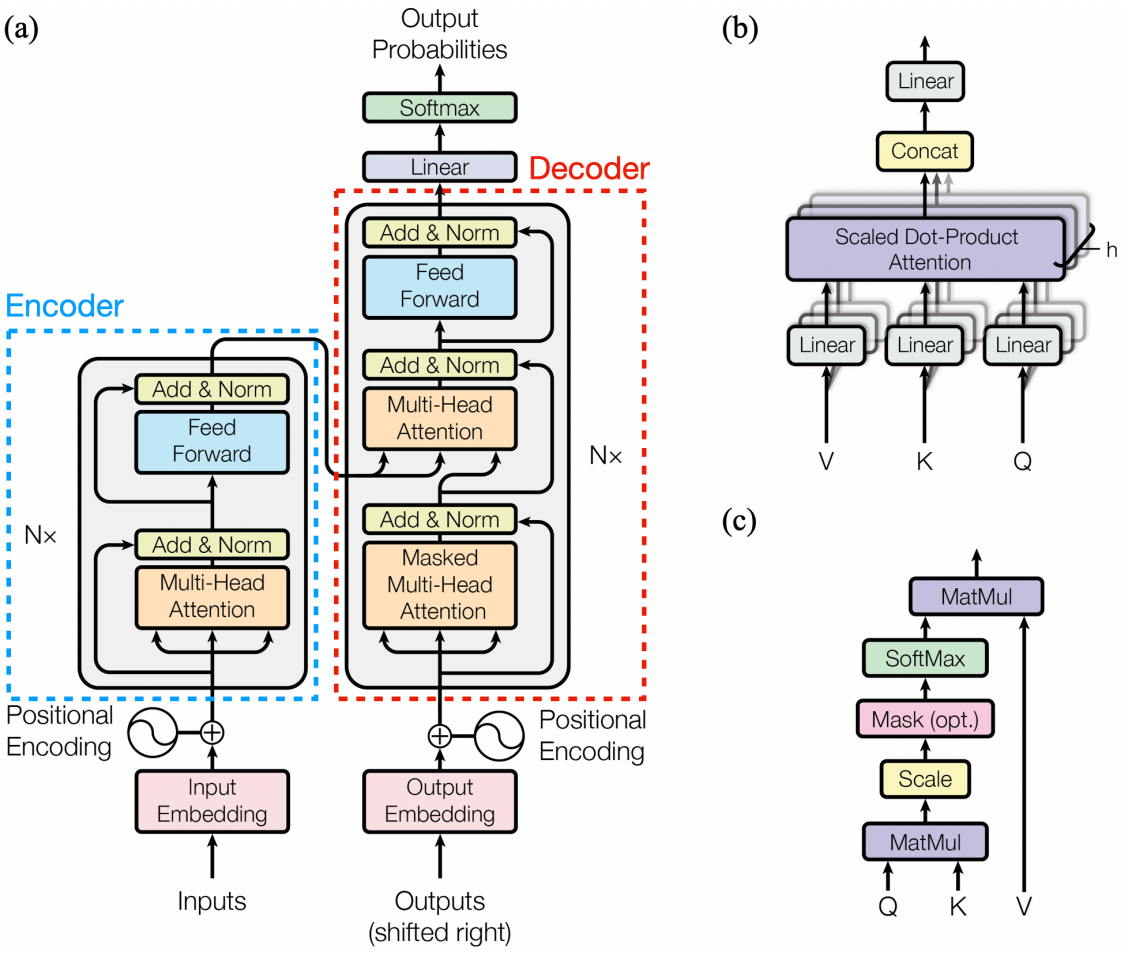

图 1　Transformer 模型架构[9]。(a) 整体结构；(b) 多头注意力模块；(c) 缩放点积注意力机制。

2.1.1 输入模块

输入模块由词嵌入和位置编码两部分组成。一段文本可以可以看作是由一个个词组成的序列 $\boldsymbol{u} = [u_1, \cdots, u_N]$，由于模型不能直接处理文字信息，首先需要将每个词 u_i 映射为一个 D 维的向量 $\boldsymbol{v}_i \in \mathbb{R}^D$，同时建立一个词典 $\mathcal{V}$ 来存储 u_i 和 $\boldsymbol{v}_i$ 之间的映射关系。常见的词嵌入方法有 Word2Vec[5]、GloVe[6]等。

另外，为了捕获词语在上下文中的位置，模型还需要编码每个词的位置信息。原版的 Transformer 使用了绝对位置编码，通过正弦函数将每个词在文本中的位置编码为一个 D 维的向量 $\boldsymbol{p}_i \in \mathbb{R}^D$，然后将词向量和位置编码相加，得到一个词最终的向量表示 $\boldsymbol{x}_i = \boldsymbol{v}_i + \boldsymbol{p}_i$。现在的大语言模型一般采用更加高效的旋转位置编码（Rotary Position Embedding，RoPE）方案[55]。

这些经过位置编码的词向量 $\boldsymbol{x}_i$ 被称为词元（token），是文本数据的最小分割单元，也就是不可再拆分的最小语义单位，将它们排列在一起就形成了一个完整的输入序列 $\mathbf{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]$。其中 $\boldsymbol{x}$ 的维度 D 一般被称为嵌入维度，词元的数量 N 也被称为输入序列的长度，$\mathbf{X} \in \mathbb{R}^{D\times N}$。

2.1.2 编码器

编码器由多个相同的层堆叠而成，其中每一层都包含一个多头自注意力模块和一个前馈网络模块。在每个注意力模块和前馈网络后，使用残差连接[56]和层归一化[57]来增强模型的效果，并且提升训练的稳定性。

从形式上来说，假设编码器有 L 层，第 ℓ 层的输入为 $\mathbf{E}_{\ell-1} \in \mathbb{R}^{D\times N}$，$\mathbf{E}_0 = \mathbf{X}$，第 ℓ 层的输出 $\mathbf{E}_\ell$ 为

$$\mathbf{E}_\ell = \mathsf{LayerNorm}\big(\mathsf{FNN}\left(\mathbf{Z}_\ell\right) + \mathbf{Z}_\ell\big), \tag{1}$$

$$\mathbf{Z}_\ell = \mathsf{LayerNorm}\big(\mathsf{MHA}\left(\mathbf{E}_{\ell-1}\right) + \mathbf{E}_{\ell-1}\big). \tag{2}$$

MHA 表示多头自注意力机制，如图 1 (b) 所示，它使用 H 个结构相同但权重参数不同的注意力模块并行计算，通过不同的注意力头来高效地捕捉输入序列中不同的语义信息，然后将不同的头的输出拼接在一起，通过一个权重矩阵 $\mathbf{W}_O \in \mathbb{R}^{H\times H}$ 进行线性变换，得到最终的输出

$$\mathsf{MHA}\left(\mathbf{X}\right) = \mathsf{Concat}\left(\mathsf{Attn}_1, \cdots, \mathsf{Attn}_H\right)\mathbf{W}_O \tag{3}$$

其中 $\mathsf{Concat}(\cdot)$ 表示拼接操作。

每个注意力头 Attn 都采用缩放点积注意力机制，如图 1 (c) 所示，首先将输入矩阵 $\mathbf{X}$ 通过三个线性变换分别映射到查询 $\mathbf{Q} = \mathbf{X}\mathbf{W}_{\mathrm{Q}}$、键 $\mathbf{K} = \mathbf{X}\mathbf{W}_{\mathrm{K}}$ 和值 $\mathbf{V} = \mathbf{X}\mathbf{W}_{\mathrm{V}}$，其中 $\mathbf{W}_{\mathrm{Q}} \in \mathbb{R}^{D\times d_k}$、$\mathbf{W}_{\mathrm{K}} \in \mathbb{R}^{D\times d_k}$ 和 $\mathbf{W}_{\mathrm{V}} \in \mathbb{R}^{D\times d_v}$ 分别是查询、键和值的权重矩阵，d_k 和 d_v 分别是键和值的维度；然后计算注意力

$$\mathsf{Attn}\left(\mathbf{Q},\mathbf{K},\mathbf{V}\right) = \mathsf{softmax}\left(\frac{1}{\sqrt{d_k}}\mathbf{Q}\mathbf{K}^{\top}\right)\mathbf{V}, \tag{4}$$

其中非线性函数 softmax 函数的形式为

$$\mathsf{softmax}\left(\boldsymbol{x}\right)_i = \frac{\exp\left(\boldsymbol{x}\right)}{\sum_i \exp\left(\boldsymbol{x}_i\right)}. \tag{5}$$

FNN 表示前馈神经网络

$$\mathsf{FNN}\left(\mathbf{X}\right) = \sigma\left(\mathbf{X}\mathbf{W}_1 + \mathbf{b}_1\right)\mathbf{W}_2 + \mathbf{b}_2, \tag{6}$$

其中，$\mathbf{W}_1 \in \mathbb{R}^{D\times d_f}$ 和 $\mathbf{W}_2 \in \mathbb{R}^{d_f\times D}$ 是权重矩阵，$\mathbf{b}_1 \in \mathbb{R}^{d_f}$ 和 $\mathbf{b}_2 \in \mathbb{R}^{D}$ 是偏置项，d_f 是前馈神经网络的隐藏层维度，$\sigma(\cdot)$ 是非线性激活函数。

LayerNorm 表示层归一化操作，它会计算每一层中所有激活值的均值 $\boldsymbol{\mu}$ 和标准差 $\boldsymbol{\sigma}$，然后重新调整激活值的中心和缩放比例

$$\mathsf{LayerNorm}\left(\boldsymbol{x}\right) = \frac{\boldsymbol{x}-\boldsymbol{\mu}}{\boldsymbol{\sigma}}\cdot\boldsymbol{\gamma} + \boldsymbol{\beta}, \tag{7}$$

其中 $\boldsymbol{\gamma}$ 和 $\boldsymbol{\beta}$ 是可以训练的缩放和偏移参数。研究表明，层归一化取得成功的关键在于缩放不变性，因此现在一般采用更加高效的 RMSNorm[58]

$$\mathsf{RMSNorm}\left(\boldsymbol{x}\right) = \frac{\boldsymbol{x}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i^2}}, \tag{8}$$

来代替层归一化。

2.1.3 解码器

解码器的结构与编码器类似，但它每一层都有两个注意力模块：一个称为掩码注意力，用于处理输入序列，即上一层解码器的输出 $\mathbf{D}_{\ell-1}$；另一个称为交叉注意力，用来同时关注编码器的输出 $\mathbf{E}_{\ell}$ 和掩码注意力模块的输出 $\mathsf{MHA}^{\ell}_{\mathrm{mask}}\left(\mathbf{D}_{\ell-1}\right)$。为了确保解码器在生成目标序列时不依赖于未来的信息，掩码注意力模块通过插

入一个掩码矩阵

$$\mathbf{M}_{ij} = \begin{cases} 0 & i \leq j \\ -\infty & i > j \end{cases}, \tag{9}$$

将当前要预测的词之后的部分置为负无穷大，然后再计算多头注意力

$$\mathsf{MHA}^{\ell}_{\text{mask}}\left(\mathbf{D}_{\ell-1}\right) = \mathsf{Concat}\left(\mathsf{Attn}^{\ell}_{1}, \cdots, \mathsf{Attn}^{\ell}_{H}\right)\mathbf{W}^{\ell}_{O}, \tag{10a}$$

$$\mathsf{Attn}^{\ell}_{h} = \mathsf{softmax}\left(\frac{1}{\sqrt{d_k}}\left(\mathbf{Q}^{\ell}_{h}\mathbf{K}^{\ell\top}_{h} + \mathbf{M}\right)\right)\mathbf{V}^{\ell}_{h}, \tag{10b}$$

$$\mathbf{Q}^{\ell}_{h} = \mathbf{D}_{\ell-1}\mathbf{W}^{h}_{\text{Q}}, \quad \mathbf{K}^{\ell}_{h} = \mathbf{D}_{\ell-1}\mathbf{W}^{h}_{\text{K}}, \quad \mathbf{V}^{\ell}_{h} = \mathbf{D}_{\ell-1}\mathbf{W}^{h}_{\text{V}}. \tag{10c}$$

掩码注意力模块的输出经过残差连接和层归一化后，得到

$$\mathbf{A}_{\ell} = \mathsf{LayerNorm}\big(\mathsf{MHA}^{\ell}_{\text{mask}}\left(\mathbf{D}_{\ell-1}\right) + \mathbf{D}_{\ell-1}\big). \tag{11}$$

交叉注意力模块将解码器中的序列 $\mathbf{A}_{\ell}$ 作为查询 $\mathbf{Q}$，将编码器的输出序列 $\mathbf{E}_{\ell}$ 作为键 $\mathbf{K}$ 和值 $\mathbf{V}$，计算多头注意力

$$\mathsf{MHA}^{\ell}_{\text{cross}}\left(\mathbf{E}_{\ell}, \mathbf{A}_{\ell}\right) = \mathsf{Concat}\left(\mathsf{Attn}^{\ell}_{1}, \cdots, \mathsf{Attn}^{\ell}_{H}\right)\mathbf{W}^{\ell}_{O}, \tag{12a}$$

$$\mathsf{Attn}^{\ell}_{h} = \mathsf{softmax}\left(\frac{1}{\sqrt{d_k}}\mathbf{Q}^{\ell}_{h}\mathbf{K}^{\ell\top}_{h}\right)\mathbf{V}^{\ell}_{h}, \tag{12b}$$

$$\mathbf{Q}^{\ell}_{h} = \mathbf{A}_{\ell}\mathbf{W}^{h}_{\text{Q}}, \quad \mathbf{K}^{\ell}_{h} = \mathbf{E}_{\ell}\mathbf{W}^{h}_{\text{K}}, \quad \mathbf{V}^{\ell}_{h} = \mathbf{E}_{\ell}\mathbf{W}^{h}_{\text{V}}, \tag{12c}$$

然后再将结果与上一个模块的输出进行残差连接和层归一化，得到

$$\mathbf{D}_{\ell} = \mathsf{LayerNorm}\big(\mathsf{MHA}^{\ell}_{\text{cross}}\left(\mathbf{E}_{\ell}, \mathbf{A}_{\ell}\right) + \mathbf{A}_{\ell}\big), \tag{13}$$

这就是解码器的输出。

2.1.4 输出模块

输出模块是一个非线性读出头，负责将解码器的输出转换为最终的预测结果

$$\mathbf{Y} = \mathsf{softmax}\left(\mathbf{D}_{L}\mathbf{W}_{\text{O}} + \mathbf{b}_{\text{O}}\right), \tag{14}$$

其中 $\mathbf{W}_{\text{O}} \in \mathbb{R}^{D\times V}$，$\mathbf{b}_{\text{O}} \in \mathbb{R}^{V}$，$V = |\mathcal{V}|$ 是词典的大小。读出头使用 softmax 函数将输出转换为一个概率分布，然后选取概率最大的词作为模型的预测结果。

2.2 大语言模型的构建流程

与 Transformer 相比，大语言模型的特点是使用了更高的向量维度、更深的网络层数，包含了更大规模的模型参数，但是对 Transformer 本身结构的改变并不大。图 2 展示了各种大语言模型的演化树，根据采用的 Transformer 类型不同，可以将大语言模型分为三类：纯编码器架构（Encoder-only）、纯解码器架构（Decoder-only）和编码器—解码器架构（Encoder-Decoder）。

纯编码器架构主要用于文本分类等任务，只在早期的少数模型中使用，如 BERT[10]、RoBERTa[59] 等。纯解码器架构主要用于文本生成等任务，是目前生成式大语言模型应用最广泛的架构，如 GPT 系列[11-12,17]、LLaMA 系列[60-62]、DeepSeek 系列[63-65]等。编码器—解码器架构结合了两种架构的优点，最初用于机器翻译任务，后来也被广泛应用到各种模型中，如 T5[13]、BART[66]、GLM 系列[67-68]等，但是由于其训练和推理速度较慢，逐渐被纯解码器架构所取代。

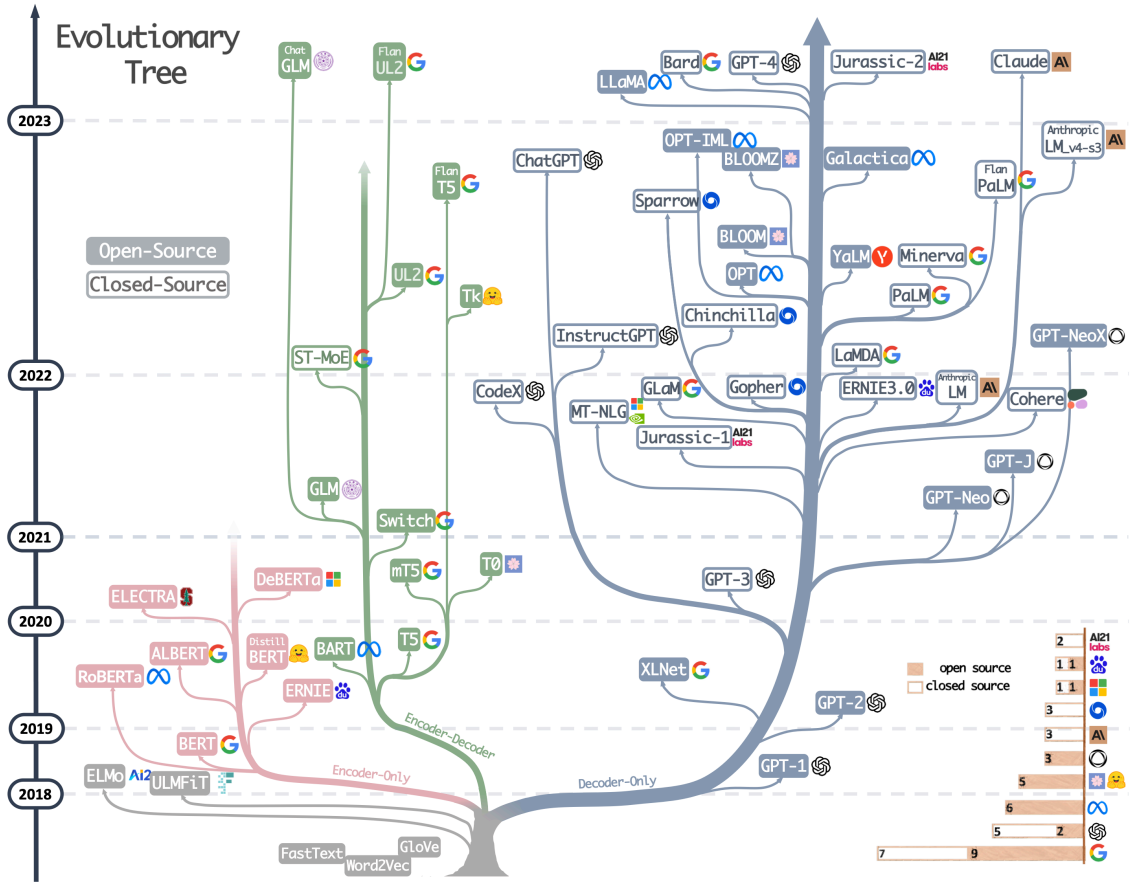

图 2　大语言模型的演化树[69]。粉色的分支表示采用了纯编码器架构的模型，蓝色的分支表示采用了纯解码器架构的模型，绿色的分支表示采用了编码器-解码器架构的模型。

由于大语言模型的参数规模十分庞大，训练过程需要大量的数据和计算资源，并且训练的目标更加泛化，因此大语言模型的构建过程需要更为复杂的训练方法。图 3 展示了大语言模型的完整构建流程，主要分为数据预处理、预训练、微调和对齐、评估、部署五个阶段。

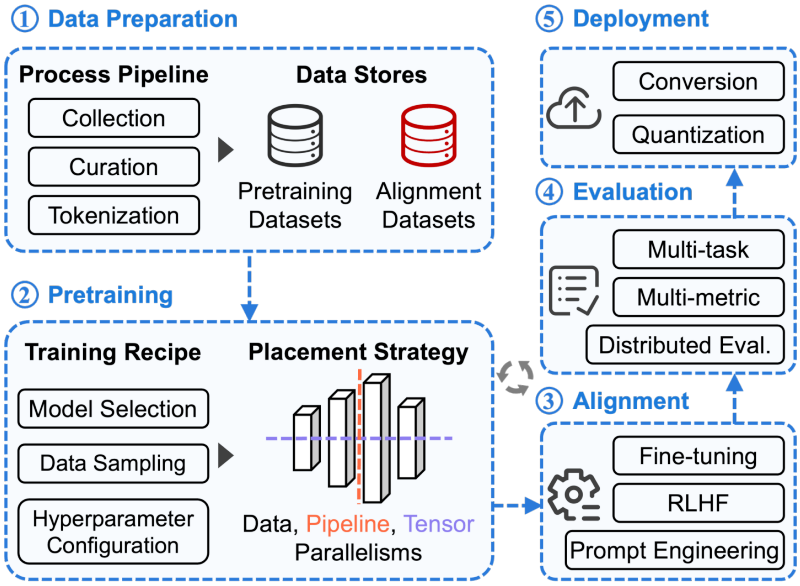

图 3　大语言模型的完整构建流程[70]。

2.2.1　数据预处理

大语言模型的训练离不开海量且高质量的数据。数据预处理是模型训练流程的起点，包括收集数据、清洗和过滤、分词等步骤。要构建一个丰富的语料库，首先就需要从多种来源收集数据，包括网页、书籍、代码、文献等，并且对这些数据进行清洗和过滤，以去除噪声（格式标签、重复内容）和敏感信息（有害内容、隐私数据），确保数据的质量和安全性。当然，也可以使用一些开源的语料库，比如 Common Crawl、Wikipedia、BooksCorpus[71]等，来构建数据集。

完成数据清洗后，就需要构建分词器（Tokenizer），将原始文本分割成模型可识别的词元（token）序列。目前的语言模型大多采用子词分词器（Subword Tokenizer），常见的子词切分算法包括 BPE 分词[72]、WordPiece 分词[73]和 Unigram 分词[74]等。另一种更加有效的方法是针对特定的语料库设计专门的分词器，比如 SentencePiece[75] 可以根据语料库的分布特性自动学习分词规则。

用于训练大模型的数据集通常分为预训练数据集和微调数据集两部分。预训练数据集采用的数据就是经过清洗和分词后的大规模无标记的数据，通常达到 TB 级别，涵盖了广泛的领域和多样化的内容。常用的数据集有 The Pile[76]、ROOTS[77]、Dolma[78]等。微调数据集则是使用经过精心挑选和人工标注的数据，规模相对较

小，通常在 GB 级别，主要用于模型的微调和对齐，使其更好地适应特定任务和人类期望。常用的数据集有 P3[79]、HH-RLHF[80]、Self-Instruct[81] 等。

2.2.2 预训练

预训练（Pretraining）是大语言模型获得通用的基础语言能力的核心环节，其核心思想是通过最大化训练数据的似然函数，进行自监督学习，使得模型能够学习到语言的基本规律、语法和语义知识、上下文关系等内容。目前最常用的预训练任务是语言建模（Language Modeling）。根据利用上下文的方式不同，语言建模任务又分为掩码语言建模（Masked Language Modeling，MLM）和因果语言建模（Causal Language Modeling，CLM）。

掩码语言建模任务主要用于非生成式（编码器或编码器—解码器架构）的大语言模型，要求模型根据上下文预测输入序列中被随机掩盖掉的词。例如，在 BERT 模型中，输入序列中的 15% 的词会被随机进行掩码处理：其中 80% 的概率用特殊标记 `[MASK]` 替换，10% 的概率替换为语料库中的一个随机词，剩余 10% 的概率保持不变。训练的目标是最大化似然函数

$$\mathcal{L}_{\mathrm{MLM}} = \sum_{i \in \mathcal{M}} \log P\left(\boldsymbol{x}_i \mid \mathbf{X}_{(i)}\right), \tag{15}$$

其中 $\mathcal{M}$ 为被掩码词的索引集合，$\mathbf{X}_{(i)}$ 表示词 $\boldsymbol{x}_i$ 的上下文信息。

因果语言建模常用于生成式模型（解码器架构），要求模型根据前面的词预测序列中的下一个词，因此也被称为下一词预测（Next Token Prediction）。在训练时，模型会接收一个完整的输入序列 $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N\}$，通过掩码机制来确保每个词只能看到它之前的词，而不能看到未来的信息。训练的目标是最大化似然函数

$$\mathcal{L}_{\mathrm{CLM}} = \frac{1}{N} \sum_{i=1}^{N} \log P\left(\boldsymbol{x}_i \mid \boldsymbol{x}_1, \cdots, \boldsymbol{x}_{i-1}\right). \tag{16}$$

此外，去噪自编码（Denoising Autoencoding）也是一种常用的预训练任务，将输入序列经过随机替换、删除等操作后，要求模型通过去除噪声来恢复原始序列。后来还出现了将掩码语言建模和去噪自编码结合起来的混合去噪器，但是这些任务需要设定额外的去噪和优化策略，较为复杂，实际应用较为有限。

预训练需要巨大的计算资源，百亿参数的模型一般需要百卡规模的算力集群训练数周的时间，千亿参数的模型则需要千卡集群训练数月的时间。当然也有一些方法可以加速预训练过程，如 3D 并行训练[82-85]、混合精度训练[86]等。

2.2.3 微调和对齐

微调（Fine-tuning）是将预训练模型调整为适应特定任务的模型，一般采用监督学习的方式进行，因而也被称为有监督微调（Supervised Fine-tuning，SFT）。早期的微调方法是全量微调（Full Fine-tuning）[14]，使用特定任务数据集更新基础模型的所有参数。这种方法能够提升模型在目标任务上的性能，但其计算成本高昂，并且容易导致模型丢失在预训练阶段学到的通用知识，即所谓的灾难性遗忘。

为了解决全量微调存在的问题，参数高效微调（Parameter-Efficient Fine-tuning，PEFT）方法应运而生[87]。其核心思想是在微调过程中冻结大部分参数，仅更新少量参数，这大大降低了计算和存储开销，并有助于缓解灾难性遗忘。常见的 PEFT 技术包括：适配器微调、前缀微调、提示微调和低秩适应（Low-Rank Adaptation，LoRA）[88]。LoRA 在预训练模型的参数矩阵上添加了一个可以低秩分解的矩阵，微调时只更新这个矩阵的参数，是目前广泛应用的一种高效微调技术。

对齐（Alignment）是保证大语言模型的行为与人类的价值观和伦理道德一致，以提升模型的安全性。目前广泛认可的对齐标准包括有用性、诚实性和无害性[89-90]，这些标准难以通过形式化的目标进行建模，因此主要通过强化学习来实现。目前最常用的是基于人类反馈的强化学习（Reinforcement Learning from Human Feedback，RLHF）[89,91]，包括三个阶段：首先进行有监督微调；然后收集人类对于模型不同输出的偏好数据，训练一个奖励模型（Reward Model）来模拟人类的偏好判断；最后将这个奖励模型作为环境的奖励函数，使用强化学习算法（比如 PPO 算法[92]）优化语言模型的策略，使其生成的内容更符合人类偏好。

2.2.4 评估

对大语言模型进行全面而准确的评估是了解其能力边界、比较不同模型的性能、指导未来研究方向的关键环节。然而，由于大语言模型能力的多样性和生成内容的开放性，传统自然语言处理任务的指标（如 BLEU[93]、ROUGE[94] 等）虽然在某些特定任务上有效，但往往不足以衡量模型的综合表现。

目前对大语言模型的主流评估方法是选择一些不同的能力维度（比如语言理解、文本生成、知识问答、逻辑推理、代码生成等），构建对应的数据集和评测任务，然后对模型的性能进行测试，这种方式也被称为基准评测。常用的评测数据集有 GLUE[95]、SuperGLUE[96]、MMLU[97]、MATH[98]等。此外，也有一些基于人类和模型的评测方法，比如 Chatbot Arena 项目、MT-Bench 项目等[99]。

2.2.5 部署

由于大语言模型在推理过程中需要消耗大量的计算资源和内存，这使得它们在实际部署中面临着巨大的挑战。为了在资源有限的环境中高效地运行大语言模型，研究者们提出了多种模型压缩技术。

最直接的压缩思路就是降低模型的参数量，一种方案是在训练好的模型上进行剪枝（Pruning），去除冗余的参数和神经元[100]。另一种方案是通过知识蒸馏（Knowledge Distillation），将大模型的知识迁移到小模型中，从而实现压缩[101]。

另一种压缩思路是将模型的权重和激活值从高精度的浮点数转换为低精度的整数，从而减少存储和计算开销，这种方案被称为量化（Quantization）[102]。研究表明，在相同显存开销的情况下，参数规模大的模型通常比表示精度高的模型更具优势[103]。虽然量化可能会导致模型的性能下降，但是可以再通过一些轻量化微调方法来恢复模型的性能[104]。

2.3 提示学习方法

由于大语言模型微调的成本十分昂贵，并且微调后的模型难以迁移到新的任务上，因此基于自然语言的提示方法就成为了使用大语言模型解决下游任务的主要方式。常见的提示方法有提示工程、上下文学习和思维链等。

提示工程就是指针对特定任务设计合适的提示词，以引导大语言模型生成所需的输出[105]。除了人工设计提示词外，研究者们还提出了一些自动优化提示的方法，比如基于梯度的提示搜索方法[106]、基于大语言模型的提示生成方法[107]。但是由于大语言模型的参数量巨大，工作机制复杂，这类提示优化方法往往需要大量的时间和算力，又难以保证生成的提示词有效，因此在实际应用中并不常用。

目前最流行的提示学习方法是上下文学习。上下文学习的提示由任务描述和示例组成，形式上可以写为

$$\mathsf{LLM}\big(I, f(x_1, y_1), \cdots, f(x_N, y_N), f(\tilde{x}, __)\big) \to \tilde{y}, \tag{17}$$

其中 I 是任务描述，$\{(x_1, y_1), \cdots, (x_N, y_N)\}$ 是 N 个样本组成的示例集合，$f(\cdot)$ 是将样本转为自然语言提示的函数，$\tilde{x}$ 是待预测的输入，$\tilde{y}$ 是模型的输出。上下文学习的关键在于选择合适的示例，并且将它们以合适的顺序排列在提示中。

思维链是一种高级提示策略，是对上下文学习的一种拓展，在示例中不仅要包含输入和输出，还要包含中间的推理过程，从而引导大语言模型在输出正确答案之前进行更加详细的推理。图 4 通过一个例子展示了上下文学习和思维链的区

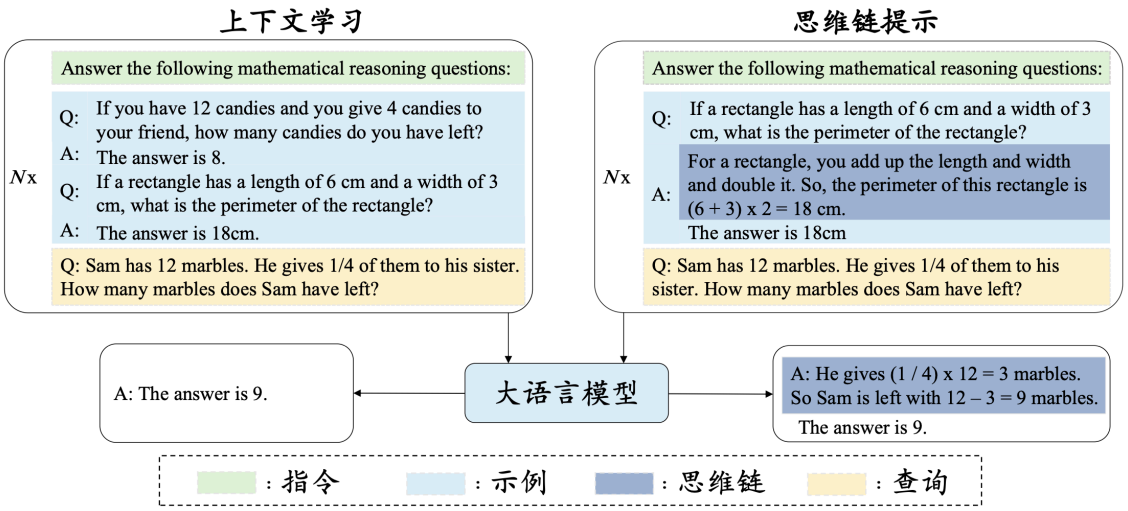

图 4　上下文学习和思维链提示的比较说明[16]

别。在实际应用中，相比于上下文学习的“输入—输出”对，思维链的获取更加复杂，通常需要借助人类的知识和经验进行设计。

上下文学习和思维链都是具有代表性的“涌现能力”，能够显著提升大语言模型的性能。但是大语言模型如何在预训练过程中获得这些能力，又如何在推理阶段执行上下文学习、提高推理能力，都是目前尚未完全理解的问题，也是目前的研究热点之一。

3 自旋玻璃模型的平均场理论

3.1 自旋玻璃模型简介

自旋玻璃理论是统计力学在神经网络研究中的一个重要工具。与一般的铁磁材料中原子能够形成长程有序的排列不同，自旋玻璃是一种典型的无序系统，其中原子之间的相互作用是随机的，会产生阻抗效应，在温度降低时使得各个磁矩被随机地冻结在某个方向，最终呈现出一种无序的、具有玻璃态特征的结构。

1970 年，P. Anderson 首次提出“自旋玻璃”的概念[108]。此后，一系列描述自旋玻璃的模型，如 Edwards–Anderson 模型[109]、Sherrington–Kirkpatrick 模型[110]等，以及相关的平均场理论，如复本方法[111]、高温展开[112-113]、空腔方法[114]等，相继被提出并得到了广泛的研究。其中最为著名的是 G. Parisi 在 1979 年提出的全阶复本对称性破缺方案[115-117]，该理论首次得到了平均场自旋玻璃模型的严格解。

使用自旋玻璃理论研究神经网络，始于上世纪 80 年代，并很快在 90 年代取得了一系列重要进展。1982 年，J. Hopfield 使用自旋玻璃模型来研究神经网络的记忆存储和模式识别，提出了 Hopfield 模型[118]，这是统计物理在神经网络研究中的第一次成功应用。随后，1985 年，D. Amit 等人使用复本方法研究了 Hopfield 模型的存储容量[119]。1987 年，E. Gardner 等人开创性地定义了感知机的存储容量，并提出其对应的平均场理论[120]。1988 年，H. Sompolinsky 等人则首次提出了循环神经网络（RNN）的动力学平均场理论（DMFT），成为后来研究神经网络的混沌现象的基础[121]。1992 年，H. Sompolinsky 等人使用复本方法详细研究了感知机的学习问题[122]。2001 年，J. Yedidia 等人将推理问题与物理中的自由能最小化联系起来，证明了信念传播算法（BP）的不动点是 Bethe 自由能的极小值[123]。2002 年，M. Mezard 等人使用空腔方法分析了约束满足问题，提出调查传播算法[124-125]。

在深度学习兴起后，统计物理，特别是自旋玻璃理论，成为了研究和解释神经网络的重要工具。最近一些比较重要的工作有：S. Ganguli 等人受非平衡态统计物理的启发提出了生成扩散模型[126]；L. Zdeborová 等人使用复本方法研究了各类神经网络的泛化误差和相变[127]；A. Montanari 等人使用随机矩阵理论研究了过参数化模型的双下降现象[128]；C. Pehlevan 等人开发出一套新的基于核回归的框架研究无限宽神经网络的性质[129]；M. Helias 等人详细阐述了如何从统计场论视角来研究神经网络的动力学和混沌现象[130]；G. Biroli 等人最近则使用统计物理方法在理论上研究了生成扩散模型的不同动力学区域以及每个区域的性质[131]。

球面 p 自旋模型是自旋玻璃中一种十分重要的可解模型。1992 年，A. Crisanti 等人利用复本方法研究了该模型的平衡态性质[132]。2020 年，F. Ricci-Tersenghi 等人详细讨论了该模型的空腔方程及其一阶复本对称性破缺对应的版本[133]。本章中我们以球面 p 自旋模型为例，分别介绍复本方法和空腔方法。

我们考虑 N 个自旋 $\sigma_i \in \mathbb{R}$（$i = 1, \cdots, N$）组成的系统，满足球面约束

$$\mathcal{N}(\boldsymbol{\sigma}) := \sum_{i=1}^{N} \sigma_i^2 = N, \tag{18}$$

并且具有 p 体相互作用。系统的哈密顿量写为

$$\mathcal{H}(\boldsymbol{\sigma}) = -\sum_{\alpha}^{M} J_\alpha \prod_{i \in \partial\alpha} \sigma_i, \tag{19}$$

其中 α 是相互作用的索引，J_α 表示相互作用强度，$\partial\alpha$ 是参与相互作用 J_α 的所有自旋的索引的集合，固定 $|\partial\alpha| = p$ 表示系统具有 p 体相互作用。我们假设 J_α 是独立同分布的随机变量，服从高斯分布，即

$$\langle J_\alpha \rangle = 0, \quad \langle J_\alpha^2 \rangle = \frac{p!}{2N^{p-1}}, \tag{20}$$

这样的方差设定可以保证哈密顿量 (19) 是一个广延量。

要求系统满足式 (18) (19)，在数学上是一个约束问题，在物理上是一个微正则系综，很容易写出系统的微观状态数为

$$\Omega(N, E) = \int \prod_{i=1}^{N} \mathrm{d}\sigma_i \, \delta\big(N - \mathcal{A}(\boldsymbol{\sigma})\big) \, \delta\big(E - \mathcal{H}(\boldsymbol{\sigma})\big). \tag{21}$$

对该微正则系综做拉普拉斯变换

$$Z(N, E) = \iint \mathrm{d}N \, \mathrm{d}E \, \Omega(N, E) \, e^{-\beta E - \lambda N}, \tag{22}$$

在数学上相当于使用拉格朗日乘数法转换约束问题，可以得到正则系综配分函数

$$Z(\mathbf{J}) = \int \prod_{i} \mathrm{d}\sigma_i \, \exp\left(-\lambda \sum_{i} \sigma_i^2 + \beta \sum_{\alpha} J_\alpha \prod_{i \in \partial\alpha} \sigma_i \right), \tag{23}$$

其中 λ 和 β 是拉格朗日乘子，在给定约束条件 N、E 时，如果拉格朗日乘子存在实值，那么满足 $N = -\partial_\lambda \log Z$，$E = -\partial_\beta \log Z$。

3.2 复本方法

式 (23) 所描述的系统，其性质取决于相互作用矩阵 $\mathbf{J}$，而 $\mathbf{J}$ 是一个不随时间变化的随机变量，我们称这样的系统具有淬火无序。我们真正关心的自由能是对 $\mathbf{J}$ 做无序平均后的结果，即

$$f = \frac{1}{N}\mathbb{E}_{\mathbf{J}} F = -\frac{1}{\beta N}\mathbb{E}_{\mathbf{J}} \ln Z. \tag{24}$$

但是 $\mathbb{E}_{\mathbf{J}} \ln Z$ 这一项很难直接计算。一种方法是利用如下恒等式

$$\mathbb{E} \ln Z = \lim_{n\to 0} \frac{1}{n} \left(\mathbb{E} Z^n - 1\right), \tag{25}$$

将求解 $\mathbb{E} \ln Z$ 转化为求解 $\mathbb{E} Z^n$，这在物理上等同于对系统的 n 个副本进行平均，因而被称为复本方法。根据式 (20) 写出 $P(J)$ 的显式

$$P(J) = \sqrt{\frac{N^{p-1}}{\pi p!}} \exp\left(-\frac{N^{p-1} J^2}{p!}\right), \tag{26}$$

因此可以计算

$$\mathbb{E}_{\mathbf{J}} Z^n = \int \prod_{\alpha} \mathrm{d}J_\alpha \; P(J_\alpha) \left[\int \prod_i \mathrm{d}\sigma_i \; \exp\left(-\lambda \sum_i \sigma_i^2 + \beta \sum_\alpha J_\alpha \prod_{i\in\partial\alpha} \sigma_i\right)\right]^n \tag{27a}$$

$$\simeq \int \prod_{i,a} \mathrm{d}\sigma_i^a \; e^{-\lambda \sum_{i,a} (\sigma_i^a)^2} \prod_\alpha \int \mathrm{d}J_\alpha \; \exp\left(-\frac{N^{p-1}}{p!} J_\alpha^2 + \beta J_\alpha \sum_a \prod_{i\in\partial\alpha} \sigma_i^a\right) \tag{27b}$$

$$\simeq \int \prod_{i,a} \mathrm{d}\sigma_i^a \; \exp\left(-\lambda \sum_{i,a} (\sigma_i^a)^2 + \frac{\beta^2 p!}{4N^{p-1}} \sum_{a,b} \sum_\alpha \prod_{i\in\partial\alpha} \sigma_i^a \sigma_i^b\right). \tag{27c}$$

其中关于相互作用索引 α 的求和可以重写为 $\sum_\alpha = \sum_{1\le i_1<\cdots<i_p\le N}$，考虑到在大 N 极限下，相互作用的数量 M 可以近似为

$$M = \mathrm{C}_N^p = \frac{N(N-1)\cdots(N-p+1)}{p!} \approx \frac{N^p}{p!} \tag{28}$$

因此

$$\sum_{1\le i_1<\cdots<i_p\le N} \approx \frac{1}{p!} \sum_{i_1,\cdots,i_p=1}^{N}. \tag{29}$$

式 (27c) 中的最后一项可以写为

$$\frac{\beta^2 p!}{4N^{p-1}} \sum_{a,b,\alpha} \prod_{i\in\partial\alpha} \sigma_i^a \sigma_i^b \approx \frac{\beta^2}{4N^{p-1}} \sum_{a,b} \sum_{i_1,\cdots,i_p} \sigma_{i_1}^a \sigma_{i_1}^b \cdots \sigma_{i_p}^a \sigma_{i_p}^b \tag{30a}$$

$$= \frac{\beta^2}{4N^{p-1}} \sum_{a,b} \left(\sum_i \sigma_i^a \sigma_i^b \right)^p \tag{30b}$$

$$= \frac{\beta^2 N}{4} \sum_{a,b} q_{ab}^p. \tag{30c}$$

式 (30c) 中定义了复本之间的重叠参量

$$q_{ab} = \frac{1}{N} \sum_i \sigma_i^a \sigma_i^b. \tag{31}$$

利用狄拉克 δ 函数及其傅里叶变换

$$1 = \prod_{a<b} \int \mathrm{d}q_{ab}\, \delta \left(N q_{ab} - \sum_i \sigma_i^a \sigma_i^b \right) \tag{32a}$$

$$= \prod_{a<b} \int \int_{i\mathbb{R}} \frac{\mathrm{d}q_{ab} \mathrm{d}\hat{q}_{ab}}{2\pi i/N} \exp \left(N q_{ab} \hat{q}_{ab} - \hat{q}_{ab} \sum_i \sigma_i^a \sigma_i^b \right), \tag{32b}$$

将 q_{ab} 的定义引入到配分函数 (27c) 中，得到

$$\mathbb{E}_{\mathbf{J}} Z^n = \int \left(\prod_{a<b} \frac{\mathrm{d}q_{ab} \mathrm{d}\hat{q}_{ab}}{2\pi i/N} \right) \exp \left(N \sum_{a<b} q_{ab} \hat{q}_{ab} - \frac{\beta^2 N}{4} \sum_{a,b} q_{ab}^p \right)$$
$$\times \int \prod_{i,a} \mathrm{d}\sigma_i^a \ \exp \left(-\lambda \sum_{i,a} (\sigma_i^a)^2 - \sum_{a<b} \hat{q}_{ab} \sum_i \sigma_i^a \sigma_i^b \right). \tag{33}$$

最后一项积分可以计算为

$$\mathcal{I} = \prod_i \int \prod_a \mathrm{d}\sigma_i^a \ \exp \left(-\lambda \sum_a (\sigma_i^a)^2 - \frac{1}{2} \sum_{a\neq b} \hat{q}_{ab}\, \sigma_i^a \sigma_i^b \right) = \left(\sqrt{\frac{(2\pi)^n}{\det \mathbf{Q}}} \right)^N, \tag{34}$$

其中 $\mathbf{Q}$ 是一个 $n \times n$ 的矩阵，

$$Q_{ab} = 2\lambda \delta_{ab} + \hat{q}_{ab}(1 - \delta_{ab}). \tag{35}$$

因此，配分函数 (33) 可以写为

$$\mathbb{E}_{\mathbf{J}} Z^n = \int \left(\prod_{a<b} \frac{\mathrm{d}q_{ab}\mathrm{d}\hat{q}_{ab}}{2\pi i/N} \right) \exp\Big(N G[Q] \Big), \tag{36}$$

其中

$$G[Q] = \sum_{a<b} q_{ab}\hat{q}_{ab} - \frac{1}{4}\beta^2 \sum_{a,b} q_{ab}^p - \frac{1}{2}\ln\det \mathbf{Q} + \frac{n}{2}\ln(2\pi). \tag{37}$$

式 (36) 中的积分可以利用大 N 极限下的鞍点近似来评估，即

$$\int \mathrm{d}x\, e^{Nf(x)} \approx e^{Nf(x^*)}, \tag{38}$$

其中 $f(x^*) = \max f(x)$。因此可以将自由能 (24) 写为

$$-\beta f = \lim_{n\to 0} \frac{1}{n} \left(\mathbb{E}_{\mathbf{J}} Z^n - 1 \right) = \lim_{n\to 0} \frac{1}{n} G[Q^\star] \equiv \mathcal{G}[Q^\star]. \tag{39}$$

要进一步计算 $\mathcal{G}[Q^\star]$，需要考虑序参量对于不同复本指标 a、b 的依赖关系。最朴素的假设就是各个复本都是等价的，即 $q_{ab} = q$，$\hat{q}_{ab} = \hat{q}$，$\forall a \neq b$，这就是所谓的复本对称性假设。需要注意，由于球面约束 (18)，$q_{aa} = 1$。

在复本对称假设下，式 (37) 可以计算为

$$\begin{aligned} G[Q] =& \frac{n(n-1)}{2} q\hat{q} - \frac{n(n-1)}{4}\beta^2 q^p - \frac{n}{4}\beta^2 \\ & - \frac{1}{2}\ln\left[(2\lambda + \hat{q}(n-1)) \, (2\lambda - \hat{q})^{n-1} \right] + \frac{n}{2}\ln(2\pi). \end{aligned} \tag{40}$$

因此可以求得

$$2\mathcal{G}[q,\hat{q}] = -q\hat{q} + \frac{1}{2}\beta^2(q^p - 1) - \frac{\hat{q}}{2\lambda - \hat{q}} - \ln(2\lambda - \hat{q}) + \ln(2\pi). \tag{41}$$

通过对 $\mathcal{G}[q,\hat{q}]$ 求导可以得到相应的极值点。令

$$\frac{\partial \mathcal{G}}{\partial q} = -\hat{q} + \frac{1}{2}\beta^2 p q^{p-1} \equiv 0, \tag{42a}$$

$$\frac{\partial \mathcal{G}}{\partial \hat{q}} = -q - \frac{\hat{q}}{(2\lambda - \hat{q})^2} \equiv 0, \tag{42b}$$

可以得到

$$\hat{q} = \frac{1}{2}\beta^2 p q^{p-1}, \qquad q = -\frac{\hat{q}}{(2\lambda - \hat{q})^2}. \tag{43}$$

另外利用拉格朗日因子 λ 施加的约束条件 $N=-\partial\log Z/\partial\lambda$，即 $\partial\mathcal{G}/\partial\lambda=-1$，可以得到

$$1=\frac{1}{2\lambda-\hat{q}}-\frac{\hat{q}}{(2\lambda-\hat{q})^2}. \tag{44}$$

联立式 (43) 和式 (44)，可以得到关于 q 的迭代方程

$$\beta^2pq^{p-2}=\frac{2}{(1-q)^2}. \tag{45}$$

将式 (43) (44) (41) 代回式 (39) 中，可以得到取无序平均后的自由能密度

$$-2\beta f=\frac{1}{2}\beta^2\left(1-q^p\right)+\ln(1-q)+\frac{q}{1-q}. \tag{46}$$

这就是通过复本方法求解球面 p 自旋模型的过程。

3.3 空腔方法

空腔方法是自旋玻璃理论中另一种重要的平均场理论方法。该方法将自旋系统看作是一个由很多节点组成的图，与自旋有关的节点 $\{i\}$ 称为变量节点，与相互作用有关的节点 $\{a\}$ 称为功能节点（请注意，在上一小节，我们使用 α 作为相互作用的索引，但是本小节我们改为使用 a）。如果相互作用是全连接的，那么系统可以看作是一个完全图，反之，则称为稀疏图。

空腔方法引入了两个新的变量：空腔概率 $\eta_{i\to a}(\sigma_i)$ 表示在移除变量节点 i 与相互作用节点 a 之间的连接时，节点上的自旋取值 σ_i 的概率；共轭空腔概率 $\hat{\eta}_{a\to i}(\sigma_i)$ 表示只有节点 a 与节点 i 之间的连接时，节点上的自旋取值 σ_i 的概率。

对于 p 自旋模型，一共有 $M=\mathrm{C}_N^p$ 个功能节点，每个功能节点 a 连接 p 个变量节点 $\{i_1,\cdots,i_p\}$，每个变量节点 i 连接 $K=\mathrm{C}_{N-1}^{p-1}$ 个功能节点。我们将配分函数 (23) 改写为因子联乘的形式

$$Z=\int\prod_i\mathrm{d}\sigma_i\prod_i e^{-\lambda\sigma_i^2}\prod_a e^{\beta J_a\sum_{i\in\partial a}\sigma_i}. \tag{47}$$

然后据此写出标准的空腔迭代方程[134]

$$\eta_{i\to a}(\sigma_i)=\frac{1}{Z_{i\to a}}\prod_{b\in\partial i\backslash a}\hat{\eta}_{b\to i}(\sigma_i), \tag{48}$$

$$\hat{\eta}_{a\to i}(\sigma_i)=\frac{1}{Z_{a\to i}}e^{-\lambda\sigma_i^2/2K}\int\prod_{j\in\partial a\backslash i}\mathrm{d}\sigma_j\,\eta_{j\to a}(\sigma_j)\,\exp\left(\beta J_a\sigma_i\prod_{j\in\partial a\backslash i}\sigma_j\right). \tag{49}$$

迭代以上两组方程到达不动点以后，就可以计算每个自旋的边际概率分布

$$\eta_i(\sigma_i) = \frac{1}{Z_i}\prod_{a\in\partial i}\eta_{a\to i}(\sigma_i), \tag{50}$$

其中 $Z_{i\to a}$、$Z_{a\to i}$ 和 Z_i 是归一化常数。

对于自旋数量比较大的系统，迭代方程 (48) (49) (50) 需要巨大的计算资源和时间。为了降低计算复杂度，一种近似方法是假设空腔概率 $\eta_{i\to j}(\sigma_i)$ 是一个高斯分布，即 $\eta_{i\to j}(\sigma_i)\sim\mathcal{N}(m_{i\to j}, v_{i\to j})$，然后推导出一阶矩 $m_{i\to j}$ 和二阶矩 $v_{i\to j}$ 的迭代方程这种迭代方程被称为松弛信念传播（relaxed belief propagation, r-BP）方程。

我们假设在 $N\to\infty$ 的极限下，空腔概率 $\eta_{i\to a}(\sigma_i)$ 服从高斯分布

$$\eta_{i\to a}(\sigma_i) = \frac{1}{\sqrt{2\pi v_{i\to a}}}\exp\left[-\frac{(\sigma_i - m_{i\to a})^2}{2v_{i\to a}}\right]. \tag{51}$$

将 J 展开到二阶，代入式 (48) (49) 中，得到

$$\begin{aligned}
&\hat{\eta}_{a\to i}(\sigma_i)\\
&= \frac{1}{\hat{Z}_{a\to i}}e^{-\frac{\lambda\sigma_i^2}{2K}}\int\prod_{j\in\partial a\setminus i}d\sigma_j\,\eta_{j\to a}(\sigma_j)\,\exp\left\{\beta J_a\sigma_i\prod_{j\in\partial a\setminus i}\sigma_j\right\} && \text{(52a)}\\
&\simeq \frac{1}{\hat{Z}_{a\to i}}e^{-\frac{\lambda\sigma_i^2}{2K}}\left[1+\beta J_a\sigma_i\prod_{j\in\partial a\setminus i}m_{j\to a}+\frac{\beta^2J_a^2}{2}\sigma_i^2\prod_{j\in\partial a\setminus i}\left(m_{j\to a}^2+v_{j\to a}\right)\right] && \text{(52b)}\\
&\simeq \frac{1}{\hat{Z}_{a\to i}}e^{-\frac{\lambda\sigma_i^2}{2K}}\exp\left\{\beta J_a\sigma_i\prod_{j\in\partial a\setminus i}m_{j\to a}\right.\\
&\qquad\left.+\frac{\beta^2J_a^2}{2}\sigma_i^2\left(\prod_{j\in\partial a\setminus i}\left(m_{j\to a}^2+v_{j\to a}\right)-\prod_{j\in\partial a\setminus i}m_{j\to a}^2\right)\right\}, && \text{(52c)}
\end{aligned}$$

以及

$$\begin{aligned}
\eta_{i\to a}(\sigma_i) &= \frac{1}{Z_{i\to a}}\prod_{b\in\partial i\setminus a}\hat{\eta}_{b\to i}(\sigma_i) && \text{(53a)}\\
&= \frac{1}{Z_{i\to a}}e^{-\frac{\lambda}{2}\sigma_i^2}\exp\left\{\beta\sigma_i\sum_{b\in\partial i\setminus a}J_b\prod_{j\in\partial b\setminus i}m_{j\to b}\right.\\
&\quad\left.+\frac{\beta^2}{2}\sigma_i^2\sum_{b\in\partial i\setminus a}J_b^2\left(\prod_{j\in\partial b\setminus i}\left(m_{j\to b}^2+v_{j\to b}\right)-\prod_{j\in\partial b\setminus i}m_{j\to b}^2\right)\right\}. && \text{(53b)}
\end{aligned}$$

比较式 (51) 和式 (53)，可以得到高斯分布的均值和方差满足的自洽方程

$$\frac{m_{i\to a}}{v_{i\to a}} = \beta \sum_{b\in\partial i\backslash a} J_b \prod_{j\in\partial b\backslash i} m_{j\to b}, \tag{54}$$

$$\frac{1}{v_{i\to a}} = \lambda - \beta^2 \sum_{b\in\partial i\backslash a} J_b^2 \left(\prod_{j\in\partial b\backslash i} \left(m_{j\to b}^2 + v_{j\to b}\right) - \prod_{j\in\partial b\backslash i} m_{j\to b}^2 \right). \tag{55}$$

在大 N 极限下，每个节点连接的功能节点数 K 也趋于无穷大，式 (54) (55) 中的求和可以用大数定律和中心极限定理来简化。注意到在大 K 极限下，相互作用 $\langle J\rangle \sim 1/K$、$\langle J^2\rangle \sim 1/K$，因此 $v_{i\to a}$ 集中在其均值 $v = \mathbb{E}(v_{i\to a})$ 附近，而 $m_{i\to a}$ 是一个具有一阶矩 $m = \mathbb{E}(m_{i\to a})$ 和二阶矩 $v = \mathbb{E}(m_{i\to a}^2)$ 的高斯随机变量，满足

$$\frac{m}{v} = \beta\langle J\rangle K\, m^{p-1}, \tag{56}$$

$$\frac{q}{v^2} = \beta^2\langle J^2\rangle K\, q^{p-1} + \beta^2\langle J\rangle^2 K^2 m^{2(p-1)}, \tag{57}$$

$$\frac{1}{v} = \lambda - \beta^2\langle J^2\rangle K \left((q+v)^{p-1} - q^{p-1}\right). \tag{58}$$

考虑到球面约束 $\sum_i \langle\sigma_i^2\rangle = N$，可以得到恒等关系 $q + v = 1$，从而进一步简化方程

$$\lambda = \frac{1}{1-q} + \beta^2\langle J^2\rangle K(1 - q^{p-1}), \tag{59}$$

$$m = \beta\langle J\rangle K\, m^{p-1}(1-q), \tag{60}$$

$$q = \left[\beta^2\langle J^2\rangle K\, q^{p-1} + \beta^2\langle J\rangle^2 K^2 m^{2(p-1)}\right](1-q)^2. \tag{61}$$

接下来，我们计算空腔方法得到的自由能，也称为 Bethe 自由能，定义为[135]

$$-\beta F = \beta\left(\sum_a \mathbb{F}_a + \sum_i \mathbb{F}_i - \sum_{(ai)} \mathbb{F}_{ai}\right) = \sum_a \log Z_a + \sum_i \log Z_i - \sum_{(ai)} \log Z_{(ai)}, \tag{62}$$

其中

$$Z_a = \int \prod_{i\in\partial a} d\sigma_i\, \eta_{i\to a}(\sigma_i)\, e^{\beta J_a \prod_{i\in\partial a}\sigma_i}, \tag{63}$$

$$Z_i = \int d\sigma_i \prod_{a\in\partial i} \hat{\eta}_{a\to i}(\sigma_i), \tag{64}$$

$$Z_{(ai)} = \int d\sigma_i\, \hat{\eta}_{a\to i}(\sigma_i)\, \eta_{i\to a}(\sigma_i). \tag{65}$$

分别计算这三项对自由能的贡献，可以得到[133]

$$Z_a = \exp\left\{\beta J_a \prod_{i\in\partial a} m_{i\to a} + \frac{\beta^2}{2}J_a^2\left(\prod_{i\in\partial a}\left(v_{i\to a}+m_{i\to a}^2\right) - \prod_{i\in\partial a} m_{i\to a}^2\right)\right\}, \tag{66}$$

$$Z_i = \left[\prod_{a\in\partial i}\hat{Z}_{a\to i}^{-1}\right]\sqrt{2\pi}\,(1-q)^{1/2}\exp\left\{\frac{1}{2}\frac{q}{(1-q)}\right\}, \tag{67}$$

$$Z_{(ai)} = \frac{1}{\hat{Z}_{a\to i}} \tag{68}$$

因此

$$\sum_a \log Z_a = \beta m^p M\langle J\rangle + \frac{\beta^2}{2}(1-q^p)M\langle J^2\rangle = N\frac{\beta^2}{4}(1-q^p), \tag{69}$$

$$\sum_i \log Z_i = \frac{N}{2}\left[\log(1-q)+\frac{q}{(1-q)}\right] - \sum_i\sum_{a\in\partial i}\log\hat{Z}_{a\to i}. \tag{70}$$

注意到

$$\begin{aligned}
&\sum_i \log(Z_i) - \sum_{(ai)\in E}\log(Z_{(ai)}) \\
&= \frac{N}{2}\left[\log(1-q)+\frac{q}{(1-q)}\right] - \sum_i\sum_{a\in\partial i}\log(\hat{Z}_{a\to i}) + \sum_i\sum_{a\in\partial i}\log(\hat{Z}_{a\to i}) &\text{(71a)}\\
&= \frac{N}{2}\left[\log(1-q)+\frac{q}{(1-q)}\right]. &\text{(71b)}
\end{aligned}$$

代入式 (62) 中，可以计算出自由能为

$$-\beta F = \frac{N}{2}\left[\frac{\beta^2}{2}(1-q^p)+\log(1-q)+\frac{q}{(1-q)}\right] \tag{72}$$

这与我们之前使用复本方法得到的自由能 (46) 是一致的。

3.4 复本对称性破缺

前两节中我们介绍的都是自旋玻璃在复本对称假设下的解。空腔方法中的不断迭代的空腔概率（或称为“消息”），在热力学极限下可以稳定收敛到某个固定的自洽分布，并且不依赖于初始条件。这种形式与复本对称假设是一致的：系统在达到某个平衡态时，不同复本之间的重叠度 q_{ab} 是相同的，即随机进行多次不同的初始化，系统总是会达到同一个平衡态。这实际上是假设系统具有单一纯态，然而这通常只在高温情况下是成立的。随着温度的降低，除了顺磁态以外，系统还会出现很多被称为玻璃态的亚稳态，并且其数量呈指数级增长。

类似于在经典热力学中使用熵 $S=\frac{1}{N}\log\sum_i\delta(E-E_i)$ 来描述系统的能量状态数，我们可以定义一个复杂度 $\Sigma(f)$ 来描述系统的亚稳态数目，其表达式为

$$\Sigma(f)=\frac{1}{N}\log\sum_{\eta}\delta(f-f_\eta), \tag{73}$$

其中 f_η 是玻璃态 η 的自由能。通常来说，$\Sigma(f)$ 是 f 的单调递增函数，即随着自由能 f 的增加，系统中具有相同自由能的亚稳纯态数量也在增加。

一般地，对于一个自旋玻璃系统，分别存在动力学临界温度 T_d 和热力学临界温度 T_K[136]。在 T_d 以下，系统的遍历性（各态历经）假设失效，模型的动力学主要受到自由能的较高的态控制；而在 T_K 以下，系统会发生相变，进入复本对称破缺（RSB）相，此时玻尔兹曼分布会分散到多个不同的状态中，两种不同的构型可以处于相同或不同的亚稳态中。此时系统的总自由能为

$$\mathcal{F}=-\frac{1}{\beta N}\log\int\mathrm{d}f\,\mathrm{e}^{-N(\beta f-\Sigma(f))}. \tag{74}$$

可以看到，多个亚稳纯态的存在会为自由能带来了额外的贡献。式 (74) 在定义上是十分清楚的，但是我们无法表征不同的亚稳态并且对它们进行计数来获得 $\Sigma(f)$。1995 年，R. Monasson 提出了耦合复本的方法来解决这个问题[137]。具体来说，我们引入 x 个耦合在一起的复本，并确保他们在低温时都处于相同的纯态中，此时 x 个复本的自由能可以写为

$$\Phi(x)=-\frac{1}{\beta}\max_f\left(\beta x f-\Sigma(f)\right), \tag{75}$$

这样，复杂度 Σ 就可以通过复本自由能的勒让德变换求出。一阶 RSB 相的总自由能与复本自由能之间的关系为 $\mathcal{F}=\min_x\left[\Phi(x)/x\right]$。

与复本对称相中空腔的高斯假设 (51) 类似，在一阶复本对称破缺相中，我们引入 x 个复本耦合的联合高斯分布[138]

$$\begin{aligned}\eta_{i\to a}\left(\boldsymbol{\sigma}_i\right)=&\int\mathrm{d}m_{i\to a}\frac{1}{\sqrt{2\pi\Delta_{i\to a}^{(0)}}}\exp\left(-\frac{\left(m_{i\to a}-h_{i\to a}\right)^2}{2\Delta_{i\to a}^{(0)}}\right)\\&\times\frac{1}{\left[\sqrt{2\pi\Delta_{i\to a}^{(1)}}\right]^x}\exp\left(-\sum_{\alpha=1}^{x}\frac{\left(\sigma_i^\alpha-m_{i\to a}\right)^2}{2\Delta_{i\to a}^{(1)}}\right).\end{aligned} \tag{76}$$

其中 $\boldsymbol{\sigma}_i=\{\sigma_i^\alpha\}$，$\alpha=1,\ldots,x$。方差 $\Delta_{i\to a}^{(1)}$ 刻画了在各个纯态内部 σ_i 的涨落，而方

差 $\Delta_{i\to a}^{(0)}$ 则刻画了各个纯态的涨落。标记积分测度

$$\int \mathcal{D}\boldsymbol{\sigma}_i = \int \prod_{\alpha=1}^{x} \mathrm{d}\sigma_i^{\alpha}, \tag{77}$$

并定义分布

$$Q_{i\to a}\left(m_{i\to a}\right) \equiv \frac{1}{\sqrt{2\pi\Delta_{i\to a}^{(0)}}} \exp\left(-\frac{\left(m_{i\to a}-h_{i\to a}\right)^2}{2\Delta_{i\to a}^{(0)}}\right), \tag{78}$$

可以写出每个自旋在空腔边际概率中的一阶和二阶矩

$$\left\langle \sigma_i^{\alpha} \right\rangle = \int \mathcal{D}\boldsymbol{\sigma}_i \sigma_i^{\alpha} \eta_{i\to a}\left(\boldsymbol{\sigma}_i\right) = \int \mathrm{d}m_{i\to a} m_{i\to a} Q_{i\to a}\left(m_{i\to a}\right) = h_{i\to a}, \tag{79}$$

$$\begin{aligned}\left\langle (\sigma_i^{\alpha})^2 \right\rangle &= \int \mathcal{D}\boldsymbol{\sigma}_i \left(\sigma_i^{\alpha}\right)^2 \eta_{i\to a}\left(\boldsymbol{\sigma}_i\right) = \int \mathrm{d}m_{i\to a} \left(\Delta_{i\to a}^{(1)} + m_{i\to a}^2\right) Q_{i\to a}\left(m_{i\to a}\right) \\ &= \Delta_{i\to a}^{(1)} + \Delta_{i\to a}^{(0)} + h_{i\to a}^2,\end{aligned} \tag{80}$$

$$\left\langle \sigma_i^{\alpha}\sigma_i^{\beta} \right\rangle = \int \mathcal{D}\boldsymbol{\sigma}_i \sigma_i^{\alpha}\sigma_i^{\beta} \eta_{i\to a}\left(\boldsymbol{\sigma}_i\right) = \int \mathrm{d}m_{i\to a} m_{i\to a}^2 Q_{i\to a}\left(m_{i\to a}\right) = \Delta_{i\to a}^{(0)} + h_{i\to a}^2, \tag{81}$$

即

$$h_{i\to a} = \left\langle \sigma_i^{\alpha} \right\rangle, \tag{82}$$

$$\Delta_{i\to a}^{(1)} = \left\langle (\sigma_i^{\alpha})^2 \right\rangle - \left\langle \sigma_i^{\alpha}\sigma_i^{\beta} \right\rangle = 1 - q_1^{i\to a}, \tag{83}$$

$$\Delta_{i\to a}^{(0)} = \left\langle \sigma_i^{\alpha}\sigma_i^{\beta} \right\rangle - \left\langle \sigma_i^{\alpha} \right\rangle^2 = q_1^{i\to a} - q_0^{i\to a}, \tag{84}$$

其中 $q_1^{i\to a}$ 和 $q_0^{i\to a}$ 分别是态之间和态内的局域重叠度。类似地，写出空腔方程

$$\eta_{i\to a}\left(\boldsymbol{\sigma}_i\right) \propto \prod_{b\in\partial i\backslash a} \hat{\eta}_{b\to i}\left(\boldsymbol{\sigma}_i\right), \tag{85}$$

$$\hat{\eta}_{a\to i}\left(\boldsymbol{\sigma}_i\right) \propto e^{-\frac{\lambda}{2K}\sum_{\alpha=1}^{x}\left(\sigma_i^{\alpha}\right)^2} \int \prod_{k\in\partial a\backslash i} \mathcal{D}\boldsymbol{\sigma}_k \, \eta_{k\to a}\left(\boldsymbol{\sigma}_k\right) \exp\left\{\beta J_a \sum_{\alpha=1}^{x} \sigma_i^{\alpha} \prod_{k\in\partial a\backslash i} \sigma_k^{\alpha}\right\}. \tag{86}$$

与上一小节类似，将 $\exp\left\{\beta J_a \sum_{\alpha=1}^{x} \sigma_i^{\alpha} \prod_{k\in\partial a\backslash i} \sigma_k^{\alpha}\right\}$ 展开到二阶，可以得到

$$\begin{aligned}&\int \prod_{k\in\partial a\backslash i} \mathcal{D}\boldsymbol{\sigma}_k \eta_{k\to a}\left(\boldsymbol{\sigma}_k\right) \exp\left\{\beta J_a \sum_{\alpha=1}^{x} \sigma_i^{\alpha} \prod_{k\in\partial a\backslash i} \sigma_k^{\alpha}\right\} \\ &\simeq \exp\left\{\hat{A}_{a\to i} \sum_{\alpha=1}^{x} \sigma_i^{\alpha} - \frac{1}{2}\hat{B}_{a\to i}^{(\mathrm{d})} \sum_{\alpha=1}^{x} \left(\sigma_i^{\alpha}\right)^2 + \frac{1}{2}\hat{B}_{a\to i}^{(\mathrm{nd})} \sum_{\alpha\neq\beta}^{x} \sigma_i^{\alpha}\sigma_i^{\beta}\right\},\end{aligned} \tag{87}$$

其中

$$\hat{A}_{a\to i} = \beta J_a \prod_{k\in\partial a\backslash i} h_{k\to a}, \tag{88}$$

$$\hat{B}^{(\mathrm{d})}_{a\to i} = \beta^2 J_a^2 \left[\prod_{k\in\partial a\backslash i} h^2_{k\to a} - \prod_{k\in\partial a\backslash i} \left(\Delta^{(1)}_{k\to a} + \Delta^{(0)}_{k\to a} + h^2_{k\to a} \right) \right], \tag{89}$$

$$\hat{B}^{(\mathrm{nd})}_{a\to i} = \beta^2 J_a^2 \left[\prod_{k\in\partial a\backslash i} \left(\Delta^{(0)}_{k\to a} + h^2_{k\to a} \right) - \prod_{k\in\partial a\backslash i} h^2_{k\to a} \right]. \tag{90}$$

功能节点到变量节点的消息可以计算为

$$\hat{\eta}_{a\to i}\left(\boldsymbol{\sigma}_i\right) \propto \exp\left\{ \hat{A}_{a\to i} \sum_{\alpha=1}^{x} \sigma_i^\alpha - \frac{1}{2}\left(\hat{B}^{(\mathrm{d})}_{a\to i} + \frac{\lambda}{K} \right) \sum_{\alpha=1}^{x} (\sigma_i^\alpha)^2 + \frac{1}{2} \hat{B}^{(\mathrm{nd})}_{a\to i} \sum_{\alpha\neq\beta}^{x} \sigma_i^\alpha \sigma_i^\beta \right\}, \tag{91}$$

变量节点到功能节点的消息则为

$$\begin{aligned} \eta_{i\to a}\left(\boldsymbol{\sigma}_i\right) \propto \exp\Bigg\{ & \left(\sum_{b\in\partial i\backslash a} \hat{A}_{b\to i} \right) \sum_{\alpha=1}^{x} \sigma_i^\alpha - \frac{1}{2}\left(\sum_{b\in\partial i\backslash a} \hat{B}^{(\mathrm{d})}_{b\to i} + \lambda \right) \sum_{\alpha=1}^{x} (\sigma_i^\alpha)^2 \\ & + \frac{1}{2}\left(\sum_{b\in\partial i\backslash a} \hat{B}^{(\mathrm{nd})}_{b\to i} \right) \sum_{\alpha\neq\beta}^{x} \sigma_i^\alpha \sigma_i^\beta \Bigg\}. \end{aligned} \tag{92}$$

为了简化记号，我们不妨定义

$$A_{i\to a} \equiv \sum_{b\in\partial i\backslash a} \hat{A}_{b\to i} = \sum_{b\in\partial i\backslash a} \beta J_b \prod_{k\in\partial b\backslash i} h_{k\to b} \tag{93}$$

$$B^{(\mathrm{d})}_{i\to a} \equiv \lambda + \sum_{b\in\partial i\backslash a} \hat{B}^{(\mathrm{d})}_{a\to i} = \lambda - \sum_{b\in\partial i\backslash a} \beta^2 J_b^2 \left[\prod_{k\in\partial b\backslash i} \left(\Delta^{(1)}_{k\to b} + \Delta^{(0)}_{k\to b} + h^2_{k\to b} \right) - \prod_{k\in\partial bi} h^2_{k\to b} \right] \tag{94}$$

$$B^{(\mathrm{nd})}_{i\to a} \equiv \sum_{b\in\partial i\backslash a} \hat{B}^{(\mathrm{nd})}_{a\to i} = \sum_{b\in\partial i\backslash a} \beta^2 J_b^2 \left[\prod_{k\in\partial b\backslash i} \left(\Delta^{(0)}_{k\to b} + h^2_{k\to b} \right) - \prod_{k\in\partial b\backslash i} h^2_{k\to b} \right], \tag{95}$$

那么空腔概率可以写为

$$\eta_{i\to a}\left(\boldsymbol{\sigma}_i\right) \propto \exp\left\{ A_{i\to a} \sum_{\alpha=1}^{x} \sigma_i^\alpha - \frac{1}{2} B^{(\mathrm{d})}_{i\to a} \sum_{\alpha=1}^{x} (\sigma_i^\alpha)^2 + \frac{1}{2} B^{(\mathrm{nd})}_{i\to a} \sum_{\alpha\neq\beta}^{x} \sigma_i^\alpha \sigma_i^\beta \right\}. \tag{96}$$

定义

$$u_\alpha = \frac{A_{i\to a}}{B^{(\mathrm{d})}_{i\to a} - (x-1) B^{(\mathrm{nd})}_{i\to a}}, \qquad \mathcal{M}_{\alpha\beta} = \delta_{\alpha\beta} B^{(\mathrm{d})}_{i\to a} + (1-\delta_{\alpha\beta}) \left(-B^{(\mathrm{nd})}_{i\to a} \right), \tag{97}$$

就可以把空腔概率再次写成高斯分布的形式

$$\eta_{i\to a}\left(\boldsymbol{\sigma}_i\right)=\sqrt{\frac{\det\mathcal{M}}{(2\pi)^x}}\exp\left\{-\frac{1}{2}\left(\boldsymbol{\sigma}_i-u\right)^T\mathcal{M}\left(\boldsymbol{\sigma}_i-u\right)\right\}. \tag{98}$$

此时，式 (82) (83) (84) 可以表示为

$$h_{i\to a}=\langle\sigma_i^\alpha\rangle=u_\alpha, \tag{99}$$

$$\Delta^{(0)}_{i\to a}=\left\langle\sigma_i^\alpha\sigma_i^\beta\right\rangle-\left[\langle\sigma_i^\alpha\rangle\right]^2=\mathcal{M}^{-1}_{\alpha\beta}, \tag{100}$$

$$\Delta^{(1)}_{i\to a}=\left\langle\left[\sigma_i^\alpha\right]^2\right\rangle-\left\langle\sigma_i^\alpha\sigma_i^\beta\right\rangle=\mathcal{M}^{-1}_{\alpha\alpha}-\mathcal{M}^{-1}_{\alpha\beta}, \tag{101}$$

其中

$$\mathcal{M}^{-1}_{\alpha\beta}=\frac{1}{\left(B^{(\mathrm{d})}_{i\to a}+B^{(\mathrm{nd})}_{i\to a}\right)}\delta_{\alpha\beta}+\frac{B^{(\mathrm{nd})}_{i\to a}}{\left(B^{(\mathrm{d})}_{i\to a}+B^{(\mathrm{nd})}_{i\to a}\right)\left(B^{(\mathrm{d})}_{i\to a}+(1-x)B^{(\mathrm{nd})}_{i\to a}\right)}. \tag{102}$$

接下来我们再次应用中心极限定理，简化上述各条鞍点方程。首先注意到 $h_{i\to a}$ 可以看作一个高斯随机变量，其一阶矩和二阶矩分别为

$$m\equiv\langle h_{i\to a}\rangle=\frac{\beta\langle J\rangle Km^{p-1}}{\mathcal{D}^{(1)}-x\mathcal{D}^{(0)}} \tag{103}$$

$$q_0\equiv\left\langle h^2_{i\to a}\right\rangle=\frac{\beta^2\left[\langle J^2\rangle Kq_0^{p-1}+\langle J\rangle^2K^2m^{2(p-1)}\right]}{\left(\mathcal{D}^{(1)}-x\mathcal{D}^{(0)}\right)^2} \tag{104}$$

其中

$$\mathcal{D}^{(1)}\equiv\lambda-\beta^2\left\langle J^2\right\rangle K\left[\left(q_0+\Delta^{(0)}+\Delta^{(1)}\right)^{p-1}-\left(q_0+\Delta^{(0)}\right)^{p-1}\right] \tag{105}$$

$$\mathcal{D}^{(0)}\equiv\beta^2\left\langle J^2\right\rangle K\left[\left(q_0+\Delta^{(0)}\right)^{p-1}-q_0^{p-1}\right] \tag{106}$$

而方差 $\Delta^{(0)}_{i\to a}$ 和 $\Delta^{(1)}_{i\to a}$ 集中在其均值附近，其一阶矩分别为

$$\Delta^{(0)}=\frac{\mathcal{D}^{(0)}}{\mathcal{D}^{(1)}\left(\mathcal{D}^{(1)}-x\mathcal{D}^{(0)}\right)} \tag{107}$$

$$\Delta^{(1)}=\frac{1}{\mathcal{D}^{(1)}} \tag{108}$$

接下来，我们根据 J 的高斯分布性质 (20)，热力学极限下

$$K\sim\frac{N^{p-1}}{(p-1)!}\quad\Longrightarrow\quad\langle J^2\rangle K\sim\frac{p}{2}, \tag{109}$$

的性质，记 $\Delta^{(0)}=q_1-q_0$，$\Delta^{(1)}=1-q_1$，$\langle J^2\rangle K\sim\frac{p}{2}$，将鞍点方程写为闭合方程

$$\beta^2\frac{p}{2}q_0^{p-2}=\left[\lambda-p\frac{\beta^2}{2}\left(1-xq_0^{p-1}+(x-1)q_1^{p-1}\right)\right]^2 \tag{110}$$

$$q_1-q_0=\frac{p\frac{\beta^2}{2}\left(q_1^{p-1}-q_0^{p-1}\right)}{\left[\lambda-p\frac{\beta^2}{2}\left(1-q_1^{p-1}\right)\right]\left[\lambda-p\frac{\beta^2}{2}\left(1-xq_0^{p-1}+(x-1)q_1^{p-1}\right)\right]} \tag{111}$$

$$1-q_1=\frac{1}{\lambda-p\frac{\beta^2}{2}\left(1-q_1^{p-1}\right)} \tag{112}$$

类似复本对称相中 Bethe 自由能的定义，一阶复本对称破缺自由能定义为[139]

$$\Phi(x)=-\left(\sum_a\mathbb{F}_a^{\mathrm{RSB}}(x)+\sum_i\mathbb{F}_i^{\mathrm{RSB}}(x)-\sum_{(ai)\in E}\mathbb{F}_{ai}^{\mathrm{RSB}}(x)\right) \tag{113}$$

其中

$$\beta\mathbb{F}_a^{\mathrm{RSB}}(x)=\log\left\{\int\prod_{i\in\partial a}\left[\,\mathrm{d}m_{i\to a}Q_{i\to a}\left(m_{i\to a}\right)\right]\mathrm{e}^{x\beta\mathbb{F}_a(\{m_{i\to a}\})}\right\}, \tag{114}$$

$$\beta\mathbb{F}_i^{\mathrm{RSB}}(x)=\log\left\{\int\prod_{a\in\partial i}\left[\,\mathrm{d}\hat{m}_{a\to i}\hat{Q}_{a\to i}\left(\hat{m}_{a\to i}\right)\right]\mathrm{e}^{x\beta\mathbb{F}_i(\{\hat{m}_{a\to i}\})}\right\}, \tag{115}$$

$$\beta\mathbb{F}_{ai}^{\mathrm{RSB}}(x)=\log\left\{\int\mathrm{d}\hat{m}_{a\to i}\,\mathrm{d}m_{i\to a}\hat{Q}_{a\to i}\left(\hat{m}_{a\to i}\right)Q_{i\to a}\left(m_{i\to a}\right)\mathrm{e}^{x\beta\mathbb{F}_{ai}(m_{i\to a},\hat{m}_{a\to i})}\right\}, \tag{116}$$

而 $\mathbb{F}_a$、$\mathbb{F}_i$ 和 $\mathbb{F}_{ai}$ 分别是复本对称情况下的自由能。最终可以求得一阶复本对称破缺自由能为[133]

$$-x\beta\mathcal{F}(x)=\sum_{a=1}^{M}\beta\mathbb{F}_a^{\mathrm{RSB}}+\sum_{i=1}^{N}\beta\mathbb{F}_i^{\mathrm{RSB}}-\sum_{(ai)\in E}\beta\mathbb{F}_{ai}^{\mathrm{RSB}}, \tag{117}$$

$$\begin{aligned}=&\frac{xN}{2}\left\{\frac{\beta^2}{2}\left[1-(1-x)q_1^p-xq_0^p\right]+\frac{q_0}{\left[1-xq_0-(1-x)q_1\right]}\right.\\&\left.+\frac{x-1}{x}\log\left(1-q_1\right)+\frac{1}{x}\log\left[1-xq_0-(1-x)q_1\right]\right\}.\end{aligned} \tag{118}$$

4 线性注意力机制的自旋模型

4.1 线性注意力模型

回顾 Transformer 中的自注意力机制

$$\mathbf{Y}\left(\mathbf{X}\right)=\mathbf{W}_{\mathrm{V}}\mathbf{X}\cdot\mathsf{softmax}\left(\frac{\left(\mathbf{W}_{\mathrm{Q}}\mathbf{X}\right)^{\top}\mathbf{W}_{\mathrm{K}}\mathbf{X}}{\sqrt{d_k}}\right),\tag{119}$$

其中的非线性函数 $\mathsf{softmax}(\cdot)$ 增强了模型的表达能力，但是也极大地增加了训练过程中的计算成本。一种简化的方案是去掉 $\mathsf{softmax}(\cdot)$ 函数，使用线性注意力机制

$$\mathbf{Y}\left(\mathbf{X}\right)=\frac{1}{N}\mathbf{W}_{\mathrm{V}}\mathbf{X}\mathbf{X}^{\top}\mathbf{W}_{\mathrm{Q}}^{\top}\mathbf{W}_{\mathrm{K}}\mathbf{X}.\tag{120}$$

为了进一步简化模型，可以将权重矩阵 $\mathbf{W}_{\mathrm{Q}}$ 和 $\mathbf{W}_{\mathrm{K}}$ 合并为 $\mathbf{W}\equiv\mathbf{W}_{\mathrm{Q}}^{\top}\mathbf{W}_{\mathrm{K}}$，并且取 $\mathbf{W}_{\mathrm{V}}=\mathbf{I}$，得到一个简化的线性注意力机制

$$\mathbf{Y}\left(\mathbf{X}\right)=\frac{1}{N}\mathbf{X}\mathbf{X}^{\top}\mathbf{W}\mathbf{X}.\tag{121}$$

这就是我们在下文中主要讨论的模型。

接下来我们构建 ICL 版本的线性回归任务。对于 D 维线性回归任务，每个样本由数据 $\boldsymbol{x}\sim\mathcal{N}(\mathbf{0},\mathbf{I}_D)$ 和对应的标签 $y=\boldsymbol{w}^{\top}\boldsymbol{x}/\sqrt{D}$ 组成，其中任务向量 $\boldsymbol{w}\sim\mathcal{N}(\mathbf{0},\mathbf{I}_D)$。在 ICL 情景中，对于第 μ 个输入矩阵，我们使用由同一个任务向量 $\boldsymbol{w}^{\mu}$ 生成的 N 个样本作为提示，使用一个额外的样本 $(\tilde{\boldsymbol{x}},\tilde{y})$ 作为查询，将输入矩阵 $\mathbf{X}^{\mu}$ 表示为

$$\mathbf{X}^{\mu}=\begin{bmatrix}\boldsymbol{x}_1^{\mu} & \cdots & \boldsymbol{x}_N^{\mu} & \tilde{\boldsymbol{x}}^{\mu}\\ y_1^{\mu} & \cdots & y_N^{\mu} & 0\end{bmatrix}\in\mathbb{R}^{(D+1)\times(N+1)},\tag{122}$$

其中查询标签 $\tilde{y}$ 对应的位置被设置为 0。因此 ICL 任务的样本为 $(\mathbf{X}^{\mu},\tilde{y}^{\mu})$。我们将模型输出 $\mathbf{Y}$ 中最后一个元素作为模型对查询样本 $\tilde{\boldsymbol{x}}$ 的预测值，即 $\hat{y}=\mathbf{Y}_{D+1,N+1}$。

ICL 的目标是利用提示对查询形成预测，由于每个 $\mathbf{X}^{\mu}$ 都是由独立采样的 $\boldsymbol{w}^{\mu}$ 生成的，测试集中样本和数据之间的线性关系是模型在训练时没有见过的。我们使用 P 个样本，也就是 P 个任务向量 $\boldsymbol{w}$ 进行训练，因此 P 也被称为任务多样性。

预训练过程通过最小化均方误差损失函数来更新权重矩阵 $\mathbf{W}$，即

$$\mathcal{L}=\frac{1}{2P}\sum_{\mu=1}^{P}(\hat{y}^{\mu}-\tilde{y}^{\mu})^{2}+\frac{\lambda}{2}\|\mathbf{W}\|^{2}, \tag{123}$$

其中 λ 是正则化强度或者权重衰减因子。模型的泛化误差定义为

$$\varepsilon_{\mathrm{g}}=\mathbb{E}_{\boldsymbol{x},\tilde{\boldsymbol{x}},\boldsymbol{w}}\left[(\hat{y}-y)^{2}\right]. \tag{124}$$

4.2 自旋玻璃模型映射

对于类似线性回归这样的任务，如果我们只考虑输出矩阵的最后一个元素（而不是最后一列）作为预测值，即 $\hat{y}=\mathbf{Y}_{D+1,N+1}$，结合式 (121)，我们可以将模型对于第 μ 个输入矩阵 $\mathbf{X}$ 的预测值改写为

$$\hat{y}^{\mu}=\mathbf{Y}_{D+1,N+1}=\frac{1}{N}\sum_{m,n,i}\mathbf{X}^{\mu}_{D,i}\mathbf{X}^{\mu}_{m,i}\mathbf{W}_{m,n}\mathbf{X}^{\mu}_{n,N+1}. \tag{125}$$

通过行主序向量化操作 $\mathrm{vec}(\cdot):\mathbb{R}^{d_1\times d_2}\to\mathbb{R}^{d_1 d_2}$，我们可以将权重矩阵 $\mathbf{W}$ 映射为向量 $\boldsymbol{\sigma}=\mathrm{vec}(\mathbf{W})$，将其他与数据有关的结构映射为向量 $\boldsymbol{s}^{\mu}=\mathrm{vec}\,(\mathbf{S}^{\mu})$，其中

$$\mathbf{S}^{\mu}_{mn}=\frac{1}{N}\sum_{i}\mathbf{X}^{\mu}_{D+1,i}\mathbf{X}^{\mu}_{m,i}\mathbf{X}^{\mu}_{n,N+1}. \tag{126}$$

这样就可以将式 (125) 改写为 $\hat{y}^{\mu}=\sum_i\sigma_i s_i^{\mu}$，因此，损失函数 (123) 可以写为

$$\mathcal{L}=\frac{1}{2P}\sum_{\mu=1}^{P}\left(\tilde{y}^{\mu}-\sum_{i}\sigma_i s_i^{\mu}\right)^{2}+\frac{\lambda}{2}\|\mathbf{W}\|^{2} \tag{127a}$$

$$=\frac{1}{2P}\sum_{\mu,i,j}s_i^{\mu}s_j^{\mu}\sigma_i\sigma_j-\frac{1}{P}\sum_{\mu,i}\tilde{y}^{\mu}s_i^{\mu}\sigma_i+\frac{1}{2}\lambda\sum_{i}\sigma_i^{2} \tag{127b}$$

这个形式与 Hopfield 模型[118]十分相似，这启发我们定义相互作用 $J_{ij}\equiv-\frac{1}{P}\sum_{\mu}s_i^{\mu}s_j^{\mu}$、外场 $h_i\equiv\frac{1}{P}\sum_{\mu}\tilde{y}^{\mu}s_i^{\mu}$ 以及正则化因子 $\lambda_i\equiv\lambda-\frac{1}{P}\sum_{\mu}(s_i^{\mu})^2$，并将损失函数改写为一个哈密顿量的形式

$$\mathcal{H}(\boldsymbol{\sigma})=-\sum_{i<j}J_{ij}\sigma_i\sigma_j-\sum_{i}h_i\sigma_i+\frac{1}{2}\sum_{i}\lambda_i\sigma_i^{2}. \tag{128}$$

显然，这是一个全连接的自旋玻璃模型。

我们将预训练数据集中的无序性，包括潜在任务向量 $\boldsymbol{w}$ 的多样性，编码为自旋之间的相互作用，以及自旋感受到的外场。将注意力机制中要训练的参数编码为自旋。基于给定的数据集，一旦我们求解了哈密顿量 (128) 的基态，就可以得到注意力机制中的最优参数 $\boldsymbol{\sigma}^*$，从而得到最优的预测值 $\hat{y}^*$。

4.3 相互作用和外场的统计性质

自旋玻璃模型的性质主要取决于相互作用和外场的分布。与传统的自旋玻璃模型直接使用高斯分布不同[132]，我们的模型中的相互作用 $\mathbf{J}$ 和外场 $\boldsymbol{h}$ 是由数据集构造而来，其统计分布更加复杂，并且很难得到解析的概率密度函数[140]。在本节中，我们通过数值方法对 $\mathbf{J}$ 和 $\boldsymbol{h}$ 的分布做一些简单的分析。

我们将输入矩阵重新表示为如下形式

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_0 & \tilde{\boldsymbol{x}} \\ \boldsymbol{y}_0^\top & 0 \end{bmatrix}, \tag{129}$$

其中 $\mathbf{X}_0 = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N] \in \mathbb{R}^{D\times N}$ 表示用作提示的数据，$\boldsymbol{y}_0 = \mathbf{X}_0^\top \boldsymbol{w}/\sqrt{D}$ 是对应的标签，$\tilde{\boldsymbol{x}}$ 是用于预测的数据。因此我们可以计算

$$\mathbf{X}\mathbf{X}^\top = \begin{bmatrix} \mathbf{X}_0 & \tilde{\boldsymbol{x}} \\ \boldsymbol{y}_0^\top & 0 \end{bmatrix} \begin{bmatrix} \mathbf{X}_0^\top & \boldsymbol{y}_0 \\ \tilde{\boldsymbol{x}}^\top & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{X}_0\mathbf{X}_0^\top + \tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\top & \mathbf{X}_0\boldsymbol{y}_0 \\ \boldsymbol{y}_0^\top\mathbf{X}_0^\top & \boldsymbol{y}_0^\top\boldsymbol{y}_0 \end{bmatrix}, \tag{130}$$

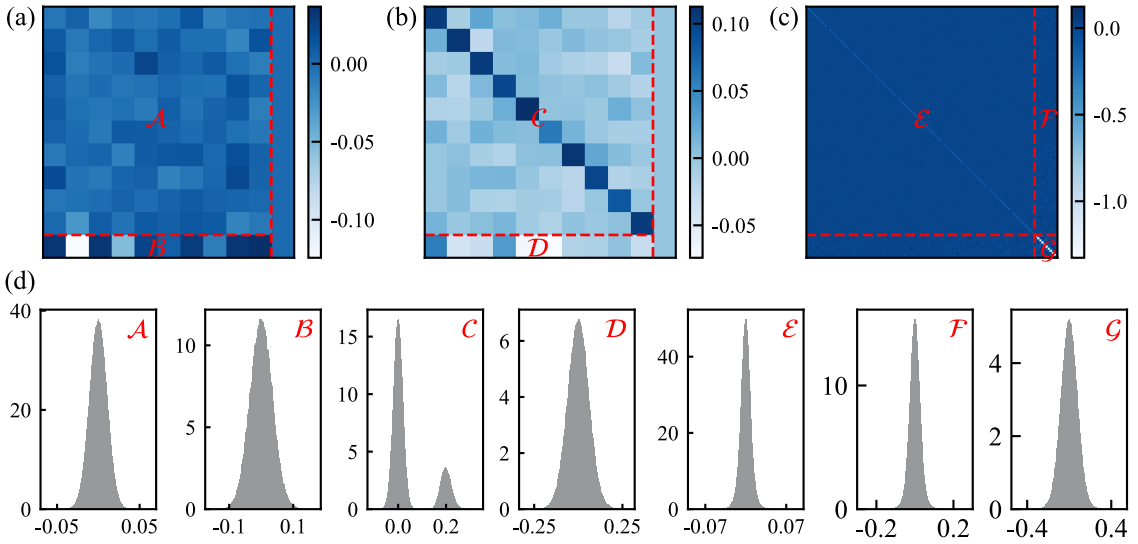

图 5　矩阵 S、H 和 J 的分块结构以及各个分块的统计分布。(a) 矩阵 $\mathbf{S}$ 的分块：$\mathcal{A} \in \mathbb{R}^{D\times D}$、$\mathcal{B} \in \mathbb{R}^{D}$ 和 $\mathbf{0} \in \mathbb{R}^{D+1}$。(b) 矩阵 $\mathbf{H}$ 的分块：$\mathcal{C} \in \mathbb{R}^{D\times D}$、$\mathcal{D} \in \mathbb{R}^{D}$ 和 $\mathbf{0} \in \mathbb{R}^{D+1}$。(c) 对称矩阵 $\mathbf{J}$ 的分块：$\mathcal{C} \in \mathbb{R}^{D(D+1)\times D(D+1)}$、$\mathcal{D} \in \mathbb{R}^{D(D+1)\times(D+1)}$ 和 $\mathcal{E} \in \mathbb{R}^{(D+1)\times(D+1)}$。(d) 不同分块的统计分布，来自 100 000 个 $(P, D, N) = (1\,000, 10, 100)$ 的系综统计结果。

并且将 $\mathbf{S}$ 重新写为

$$\mathbf{S} = \frac{1}{N}\begin{bmatrix} \mathbf{X}_0\boldsymbol{y}_0 \\ \boldsymbol{y}_0^\top\boldsymbol{y}_0 \end{bmatrix} \otimes \begin{bmatrix} \tilde{\boldsymbol{x}} \\ 0 \end{bmatrix} = \frac{1}{N}\begin{bmatrix} \mathbf{X}_0\boldsymbol{y}_0\tilde{\boldsymbol{x}}^\top & 0 \\ \boldsymbol{y}_0^\top\boldsymbol{y}_0\,\tilde{\boldsymbol{x}}^\top & 0 \end{bmatrix}. \tag{131}$$

显然，$\mathbf{S}$ 矩阵可以分为三个不同的分块，如图 5 (a) 所示，最后一列为全零向量，而其他两个分块分别标记为 $\mathcal{A}$（$m < D+1, n \neq D+1$）和 $\mathcal{B}$（$m = D+1, n \neq D+1$。结合 $\mathbf{J}$ 和 $\boldsymbol{h}$ 的定义，$\mathbf{H} = \mathrm{vec}^{-1}(\boldsymbol{h})$ 也有两个主要的分块，分别标记为 $\mathcal{C}$ 和 $\mathcal{D}$，如图 5 (b) 所示；$\mathbf{J}$ 有三个主要分块，分别标记为 $\mathcal{E}$、$\mathcal{F}$ 和 $\mathcal{G}$，如图 5 (c) 所示。

事实上，我们可以写出 $\mathbf{J}$ 和 $\mathbf{H}$ 矩阵中每个分块的元素的具体表达式。用 C、D 分别表示 $\mathcal{C}$、$\mathcal{D}$ 分块中的元素，那么

$$C_{ij} = \frac{1}{PDN}\sum_{\mu,k} w_k\tilde{x}_k^\mu \sum_m x_{im}^\mu \sum_n w_n x_{nm}^\mu \tilde{x}_j^\mu, \tag{132a}$$

$$D_i = \frac{1}{PDN}\sum_{\mu,k} w_k\tilde{x}_k^\mu \sum_m \Big(\sum_n w_n x_{nm}^\mu\Big)^2 \tilde{x}_i^\mu, \tag{132b}$$

其中所有的变量 $w, x, \tilde{x}$ 都是独立同分布的标准高斯随机变量。我们用 $P(C)$ 和 $P(D)$ 分别表示分块 $\mathcal{C}$ 和 $\mathcal{D}$ 中元素的分布，它们都是许多高斯随机变量的和与积的复合形式。这样，我们也可以将其他分块的元素写为：$E = C_1C_2$、$F = C_1D_1$ 和 $G = D_1D_2$，其中 $C_1, C_2 \sim P(C)$，$D_1, D_2 \sim P(D)$。

通过上述表达式直接求解这些分布的概率密度函数是非常困难的，这里我们只给出一些数值结果，如图 5 (d) 所示。我们还绘制了各个分块的元素分布随着 N 变化的 KL 散度，如图 6 所示。可以看到，$\mathcal{C}$、$\mathcal{E}$、$\mathcal{F}$ 分块的元素分布对示例长度 N 比较敏感，而 $\mathcal{D}$、$\mathcal{G}$ 分块的元素分布对 N 的依赖性较弱。

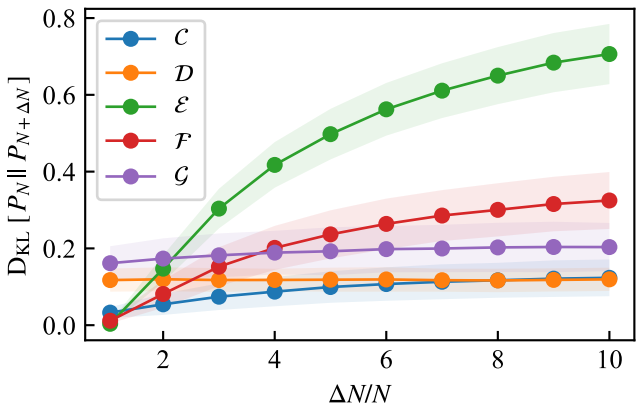

图 6　不同分块的元素分布随着 N 变化的 KL 散度 $\mathrm{D}_{\mathrm{KL}}[P_N \| P_{N+\Delta N}]$。图中的结果对 1 000 个 $P = 1\,000$、$D = 20$、$N = 20$ 的样本取了平均。

4.4 基于平均场理论的解决方案

4.4.1 空腔方法

回顾在 4.2 小节中定义的哈密顿量 (128)，在系统处于平衡态时，自旋构型的玻尔兹曼分布可以写为

$$P(\boldsymbol{\sigma}) = \frac{1}{Z} e^{-\beta H(\boldsymbol{\sigma})} = \frac{1}{Z} \prod_i e^{\beta h_i \sigma_i - \frac{\beta\lambda}{2}\sigma_i^2} \prod_{i<j} e^{\beta J_{ij}\sigma_i\sigma_j}, \tag{133}$$

其中 Z 是配分函数，β 是逆温度。

这种系统可以使用第 3.3 小节中介绍的空腔方法来求解。同样地，定义空腔概率 $\eta_{i\to ij}(\sigma_i)$ 表示当不考虑自旋 i 与自旋 j 的相互作用时自旋 i 的概率分布，并定义其共轭概率 $\eta_{ij\to i}(\sigma_i)$ 表示只考虑自旋 i 与自旋 j 的相互作用时自旋 i 的概率分布。根据式 (133) 的形式，我们可以写出标准的空腔迭代方程（也被称为信念传播方程，参考教材[134]）

$$\eta_{i\to ij}(\sigma_i) = \frac{1}{z_{i\to ij}} \exp\left(\beta h_i\sigma_i - \frac{\beta\lambda}{2}\sigma_i^2\right) \prod_{ik\in\partial i\backslash ij} \eta_{ik\to i}(\sigma_b), \tag{134}$$

$$\eta_{ij\to i}(\sigma_i) = \frac{1}{z_{ij\to i}} \int \prod_{j\in\partial ij\backslash i} \mathrm{d}\sigma_j\ \eta_{j\to ij}(\sigma_j)\ \exp\left(\beta J_{ij}\sigma_i\sigma_j\right). \tag{135}$$

注意到我们的模型只考虑二体相互作用，这意味着 $\partial ij\backslash i$ 只包含一个元素 j，因此可以省略乘积符号 $\prod_{j\in\partial ij\backslash i}$，并且可以将 $i\to ij$ 简化为 $i\to j$，将 $ij\to i$ 简化为 $j\to i$。因此可以将式 (134) 和式 (135) 合并为

$$\eta_{i\to j}(\sigma_i) = \frac{1}{z_{i\to j}} \exp\left(\beta h_i\sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2\right) \prod_{k\neq i,j} \left[\int \mathrm{d}\sigma_k\ \eta_{k\to i}(\sigma_k)\ \exp\left(\beta J_{ik}\sigma_i\sigma_k\right)\right]. \tag{136}$$

其中 $\eta_{i\to j}(\sigma_i)$ 表示在系统中移除掉自旋 j 后自旋 i 的概率分布。

不断迭代式 (136) 得到各个自旋节点的空腔概率后，就可以计算每个自旋的边际概率分布为

$$\eta_i(\sigma_i) = \frac{1}{z_i} \exp\left(\beta h_i\sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2\right) \prod_{j\neq i} \int \mathrm{d}\sigma_j\ \eta_{j\to i}(\sigma_j)\ \exp\left(\beta J_{ij}\sigma_i\sigma_j\right). \tag{137}$$

取 $\beta\to\infty$，就可以获得系统处于基态时每个自旋取值的概率分布。

4.4.2 松弛信念传播方程

在第 3.3 小节中提到过，直接迭代式 (136) 的复杂度较高，我们可以使用高斯近似推导出 r-BP 方程，简化这个迭代过程。为了验证这个直觉，本节中我们不直接假设 $\eta_{i\to j}(\sigma_i)$ 是高斯分布 (51)，而是从式 (136) 出发，通过傅里叶变换和泰勒展开，得到空腔概率的近似形式。

定义 $\xi_{i\to j}=\sum_{k\neq i,j}\beta J_{ik}\sigma_k$ 和 $G(\xi_{i\to j})=\mathrm{e}^{\sigma_i\xi_{i\to j}}$，那么可以将式 (136) 写为

$$\eta_{i\to j}(\sigma_i)=\frac{1}{z_{i\to j}}\mathrm{e}^{\beta h_i\sigma_i-\frac{1}{2}\beta\lambda_i\sigma_i^2}\int\prod_{k\neq i,j}\left[\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\right]G(\xi_{i\to j}) \tag{138a}$$

$$=\frac{1}{z_{i\to j}}\mathrm{e}^{\beta h_i\sigma_i-\frac{1}{2}\beta\lambda_i\sigma_i^2}\int\prod_{k\neq i,j}\left[\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\right]\int\mathrm{d}\hat{\xi}_{i\to j}\,\hat{G}(\hat{\xi}_{i\to j})\,e^{i\xi_{i\to j}\hat{\xi}_{i\to j}} \tag{138b}$$

$$=\frac{1}{z_{i\to j}}\mathrm{e}^{\beta h_i\sigma_i-\frac{1}{2}\beta\lambda_i\sigma_i^2}\int\mathrm{d}\hat{\xi}_{i\to j}\hat{G}(\hat{\xi}_{i\to j})\prod_{k\neq i,j}\mathcal{I}_{k\to i}. \tag{138c}$$

其中，在式 (138b) 中我们引入了 $G(\xi)$ 的傅里叶变换 $\hat{G}(\hat{\xi}_{i\to j})$，并且将一些常数项移到归一化因子 $z_{i\to j}$ 中。式 (138c) 中的 $\mathcal{I}_{k\to i}$ 定义为

$$\mathcal{I}_{k\to i}\equiv\int\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\,\exp\left[i\hat{\xi}\left(\beta J_{ik}\sigma_k\right)\right]. \tag{139}$$

这个积分可以详细计算为

$$\mathcal{I}_{k\to i}\approx\int\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\left[1+i\hat{\xi}\left(\beta J_{ik}\sigma_k\right)-\frac{1}{2}\hat{\xi}^2\left(\beta J_{ik}\sigma_k\right)^2\right] \tag{140a}$$

$$=1+i\hat{\xi}\beta J_{ik}\int\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\sigma_k-\frac{1}{2}\hat{\xi}^2\beta^2J_{ik}^2\int\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\sigma_k^2 \tag{140b}$$

$$=1+i\hat{\xi}\beta J_{ik}m_{k\to i}-\frac{1}{2}\hat{\xi}^2\beta^2J_{ik}^2\left(v_{k\to i}+m_{k\to i}^2\right) \tag{140c}$$

$$\simeq\exp\left[i\hat{\xi}\beta J_{ik}m_{k\to i}-\frac{1}{2}\hat{\xi}^2\beta^2J_{ik}^2v_{k\to i}\right]. \tag{140d}$$

其中，在式 (140a) 中，我们使用了泰勒展开，忽略了高阶项；在式 (140b) 中，我们定义了消息（即空腔边际概率）的均值 $m_{k\to i}$ 和方差 $v_{k\to i}$ 分别为

$$m_{k\to i}\equiv\int\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\,\sigma_k, \tag{141}$$

$$v_{k\to i}\equiv\int\mathrm{d}\sigma_k\,\eta_{k\to i}(\sigma_k)\,\sigma_k^2-m_{k\to i}^2; \tag{142}$$

在式 (140d) 中，我们忽略了 J_{ij} 的高阶项，并将表达式写回指数形式。然后，将式

(140d) 代回式 (138c)，可以得到

$$\eta_{i\to j}(\sigma_i) = \frac{1}{z_{i\to j}} \exp\left(\beta h_i \sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2\right) \times \int \mathrm{d}\hat{\xi}_{i\to j}\hat{G}(\hat{\xi}_{i\to j}) \prod_{k\neq i,j} \exp\left[i\hat{\xi}_{i\to j}\beta J_{ik} m_{k\to i} - \frac{1}{2}\hat{\xi}_{i\to j}^2\beta^2 J_{ik}^2 v_{k\to i}\right]. \tag{143}$$

对上式做逆傅里叶变换，将 $\hat{G}(\hat{\xi}_{i\to j})$ 重表示为 $G(\xi_{i\to j})$，可以得到

$$\eta_{i\to j}(\sigma_i) = \frac{1}{z_{i\to j}} \exp\left(\beta h_i \sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2\right) \int \mathrm{d}\hat{\xi}_{i\to j} \int \mathrm{d}\xi_{i\to j}\, G(\xi_{i\to j}) \times \exp\left(-i\xi_{i\to j}\hat{\xi}_{i\to j}\right) \prod_{k\neq i,j} \exp\left[i\hat{\xi}_{i\to j}\beta J_{ik} m_{k\to i} - \frac{1}{2}\hat{\xi}_{i\to j}^2\beta^2 J_{ik}^2 v_{k\to i}\right] \tag{144a}$$

$$= \frac{1}{z_{i\to j}} \exp\left(\beta h_i \sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2\right) \int \mathrm{d}\xi_{i\to j}\, G(\xi_{i\to j}) \times \int \mathrm{d}\hat{\xi}_{i\to j} \exp\left[-\frac{1}{2}\Big(\beta^2 \sum_{k\neq i,j} J_{ik}^2 v_{k\to i}\Big)\hat{\xi}_{i\to j}^2 + i\Big(\beta \sum_{k\neq i,j} J_{ik} m_{k\to i} - \xi_{i\to j}\Big)\hat{\xi}_{i\to j}\right] \tag{144b}$$

$$= \frac{1}{z_{i\to j}} \mathrm{e}^{\beta h_i\sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2} \int \mathrm{d}\xi_{i\to j}\, G(\xi_{i\to j}) \exp\left\{-\frac{1}{2}\frac{\left(\xi_{i\to j} - \beta\sum_{k\neq i,j} J_{ik} m_{k\to i}\right)^2}{\beta^2 \sum_{k\neq i,j} J_{ik}^2 v_{k\to i}}\right\}. \tag{144c}$$

因此我们可以定义 $\xi_{i\to j}$ 的均值和方差

$$M_{i\to j} = \beta \sum_{k\neq i,j} J_{ik} m_{k\to i}, \qquad V_{i\to j} = \beta^2 \sum_{k\neq i,j} J_{ik}^2 v_{k\to i}, \tag{145}$$

并进一步计算空腔概率 $\eta_{i\to j}(\sigma_i)$ 为

$$\eta_{i\to j}(\sigma_i) = \frac{1}{z_{i\to j}} \mathrm{e}^{\beta h_i\sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2} \int \mathrm{d}\xi_{i\to j}\, G(\xi_{i\to j}) \exp\left[-\frac{(\xi_{i\to j} - M_{i\to j})^2}{2V_{i\to j}}\right] \tag{146a}$$

$$= \frac{1}{z_{i\to j}} \mathrm{e}^{\beta h_i\sigma_i - \frac{1}{2}\beta\lambda_i\sigma_i^2} \times \int \mathrm{d}\xi_{i\to j} \exp\left[-\frac{1}{2}\frac{1}{V_{i\to j}}\xi_{i\to j}^2 + \left(\frac{M_{i\to j}}{V_{i\to j}} + \sigma_i\right)\xi_{i\to j} - \frac{M_{i\to j}^2}{2V_{i\to j}}\right] \tag{146b}$$

$$= \frac{1}{z_{i\to j}} \mathrm{e}^{\beta h_i\sigma_i - \frac{\beta\lambda}{2}\sigma_i^2} \exp\left(M_{i\to j}\sigma_i + \frac{1}{2}V_{i\to j}\sigma_i^2\right) \tag{146c}$$

$$= \frac{1}{z_{i\to j}} \exp\left[-\frac{1}{2}\left(\beta\lambda_i - V_{i\to j}\right)\sigma_i^2 + \left(\beta h_i + M_{i\to j}\right)\sigma_i\right]. \tag{146d}$$

正如预期的那样，式 (146d) 表明 $\eta_{i\to j}(\sigma_i)$ 服从高斯分布 $\mathcal{N}\left(m_{i\to j}, v_{i\to j}\right)$，其均值和方差分别为

$$m_{i\to j}=\frac{\beta h_i+M_{i\to j}}{\beta\lambda_i-V_{i\to j}},\qquad v_{i\to j}=\frac{1}{\beta\lambda_i-V_{i\to j}}. \tag{147}$$

迭代式 (145) (147) 达到不动点后，就计算边际概率分布

$$\begin{aligned}\eta_i(\sigma_i)=&\frac{1}{z_i}\exp\left(\beta h_i\sigma_i-\frac{1}{2}\beta\lambda_i\sigma_i^2\right)\\&\times\prod_{j\neq i}\left\{\int \mathrm{d}\sigma_j\ \exp\left[-\frac{1}{2}\left(\beta\lambda_j-V_{j\to i}\right)\sigma_j^2+\left(\beta h_j+\beta J_{ij}\sigma_i+M_{j\to i}\right)\sigma_j\right]\right\}\end{aligned} \tag{148a}$$

$$=\frac{1}{z_i}\exp\left(\beta h_i\sigma_i-\frac{1}{2}\beta\lambda_i\sigma_i^2\right)\prod_{j\neq i}\exp\left[\frac{\left(\beta h_j+\beta J_{ij}\sigma_i+M_{j\to i}\right)^2}{2\left(\beta\lambda_j-V_{j\to i}\right)}\right] \tag{148b}$$

$$=\frac{1}{z_i}\exp\left[-\frac{1}{2}\Big(\beta\lambda_i-\sum_{j\neq i}\beta^2J_{ij}^2v_{j\to i}\Big)\sigma_i^2+\Big(\beta h_i+\sum_{j\neq i}\beta J_{ij}m_{j\to i}\Big)\sigma_i\right]. \tag{148c}$$

式 (148c) 表明每个节点的边际概率分布 $\eta_i(\sigma_i)$ 也服从高斯分布 $P_i(\sigma_i)\sim\mathcal{N}(m_i,v_i)$，其均值和方差分别为

$$m_i=\frac{\beta h_i+\beta\sum_{j\neq i}J_{ij}m_{j\to i}}{\beta\lambda_i-\beta^2\sum_{j\neq i}J_{ij}^2v_{j\to i}},\qquad v_i=\frac{1}{\beta\lambda_i-\beta^2\sum_{j\neq i}J_{ij}^2v_{j\to i}}. \tag{149}$$

现在，我们可以通过迭代式 (145) (147)，得到每个节点的空腔概率的均值和方差，然后代入式 (149)，得到每个自旋的均值和方差。

4.4.3 近似消息传递方程

上一小节得到的 r-BP 迭代方程的空间复杂度仍为 $\mathcal{O}(D^4)$，我们尝试通过移除形如 $i\to j$ 这样的箭头来降低复杂度。定义

$$M_i=M_{i\to j}+\beta J_{ij}m_{j\to i},\qquad V_i=V_{i\to j}+\beta^2J_{ij}^2v_{j\to i}, \tag{150}$$

我们发现

$$m_i-m_{i\to j}=\frac{\beta h_i+M_i}{\beta\lambda-V_i}-\frac{\beta h_i+M_{i\to j}}{\beta\lambda-V_{i\to j}} \tag{151a}$$

$$=\frac{\beta J_{ij}m_{j\to i}(\beta\lambda_i-V_i)+\beta^2J_{ij}^2v_{j\to i}(\beta h_i+M_i)}{\beta^2\lambda_i^2-2\beta\lambda_iV_i+V_i^2}\sim\mathcal{O}\left(\frac{1}{D^2}\right), \tag{151b}$$

以及

$$v_i - v_{i\to j} = \frac{1}{\beta\lambda - V_i} - \frac{1}{\beta\lambda - V_{i\to j}} = \frac{\beta^2 J_{ij}^2 v_{j\to i}}{\beta^2\lambda_i^2 - 2\beta\lambda_i V_i + V_i^2} \sim \mathcal{O}\left(\frac{1}{D^4}\right). \tag{152}$$

在大 D 极限下（实际上，只要 D 不太小），我们可以将 r-BP 方程简化为近似消息传递（approximate message passing, AMP）方程

$$m_i = \frac{\beta h_i + \beta\sum_{j\neq i} J_{ij} m_j}{\beta\lambda_i - \beta^2\sum_{j\neq i} J_{ij}^2 v_j}, \qquad v_i = \frac{1}{\beta\lambda_i - \beta^2\sum_{j\neq i} J_{ij}^2 v_j}. \tag{153}$$

在这种情况下，我们只需要迭代以上关于 (m_i, v_i) 的方程达到不动点，就可以得到每个自旋的均值和方差。AMP 方程的空间复杂度为 $\mathcal{O}(D^2)$，计算开销远远小于 r-BP 方程。

通过式 (153) 得到每个自旋的均值 $\{m_i\}$ 后，通过逆向量化操作 $\mathbf{W} = \mathrm{vec}^{-1}(\boldsymbol{\sigma})$，就可以得到神经网络模型的权重参数。

5 结果与讨论

5.1 线性回归任务的最优解

对于线性回归任务来说，我们可以从理论上推导出模型 (121) 的最优解。将权重矩阵也按照 $\mathbf{X}$ 的形式分为四个分块，即

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix}, \tag{154}$$

其中 $\mathbf{W}_{11} \in \mathbb{R}^{D\times D}$，$\mathbf{W}_{12} \in \mathbb{R}^{D\times 1}$，$\mathbf{W}_{21} \in \mathbb{R}^{1\times D}$，$\mathbf{W}_{22} \in \mathbb{R}$。利用分块矩阵乘法，可以计算

$$\mathbf{W}\mathbf{X} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{X}_0 & \tilde{\boldsymbol{x}} \\ \boldsymbol{y}_0^\top & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{11}\mathbf{X}_0 + \mathbf{W}_{12}\boldsymbol{y}_0^\top & \mathbf{W}_{11}\tilde{\boldsymbol{x}} \\ \mathbf{W}_{21}\mathbf{X}_0 + \mathbf{W}_{22}\boldsymbol{y}_0^\top & \mathbf{W}_{21}\tilde{\boldsymbol{x}} \end{bmatrix}. \tag{155}$$

那么，结合式 (130) 和模型的预测值 $\hat{y} = \mathbf{Y}_{D+1,N+1}$，可以得到

$$\begin{aligned}
\hat{y} &= \frac{1}{N}\left[\mathbf{X}\mathbf{X}^\top\mathbf{W}\mathbf{X}\right]_{D+1,N+1} && \text{(156a)}\\
&= \frac{1}{N}\left[\boldsymbol{y}_0^\top\mathbf{X}_0^\top\mathbf{W}_{11}\tilde{\boldsymbol{x}} + \boldsymbol{y}_0^\top\boldsymbol{y}_0\mathbf{W}_{21}\tilde{\boldsymbol{x}}\right] && \text{(156b)}\\
&= \frac{1}{N}\left[\left(\frac{1}{\sqrt{D}}\mathbf{X}_0^\top\boldsymbol{w}_\star\right)^\top\mathbf{X}_0^\top\mathbf{W}_{11}\tilde{\boldsymbol{x}} + \left(\frac{1}{\sqrt{D}}\mathbf{X}_0^\top\boldsymbol{w}_\star\right)^\top\left(\frac{1}{\sqrt{D}}\mathbf{X}_0^\top\boldsymbol{w}_\star\right)\mathbf{W}_{21}\tilde{\boldsymbol{x}}\right] && \text{(156c)}\\
&= \frac{1}{N}\left[\frac{1}{\sqrt{D}}\boldsymbol{w}_\star^\top\mathbf{X}_0\mathbf{X}_0^\top\mathbf{W}_{11}\tilde{\boldsymbol{x}} + \frac{1}{D}\boldsymbol{w}_\star^\top\mathbf{X}_0\mathbf{X}_0^\top\boldsymbol{w}_\star\mathbf{W}_{21}\tilde{\boldsymbol{x}}\right] && \text{(156d)}\\
&= \frac{1}{\sqrt{D}}\boldsymbol{w}_\star^\top\mathbf{C}_0\left(\mathbf{W}_{11} + \frac{1}{\sqrt{D}}\boldsymbol{w}_\star\mathbf{W}_{21}\right)\tilde{\boldsymbol{x}}, && \text{(156e)}
\end{aligned}$$

在式 (156c) 中，使用了 $\boldsymbol{y}_0 = \mathbf{X}_0^\top\boldsymbol{w}_\star/\sqrt{D}$；在式 (156d) 中，我们定义了 $\mathbf{C}_0 = \frac{1}{N}\mathbf{X}_0\mathbf{X}_0^\top$。结合实际标签的定义 $\tilde{\boldsymbol{y}} = \boldsymbol{w}^\top\tilde{\boldsymbol{x}}/\sqrt{D}$，一个训练良好的权重具有应该满足如下条件：

$$\mathbf{C}_0\left(\mathbf{W}_{11} + \frac{1}{\sqrt{D}}\boldsymbol{w}_\star\mathbf{W}_{21}\right) = \mathbf{I}_D. \tag{157}$$

这样，模型的预测值 $\hat{y}$ 就可以收敛到实际标签 $\tilde{y}$。

如果矩阵 $\mathbf{X}_0$ 每个元素都是独立同分布的高斯随机变量，即 $\mathbf{X}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$，在比例极限下，即 $D, N \to \infty$ 但是 $N/D \sim \mathcal{O}(1)$，矩阵 $\mathbf{C}_0$ 是一个归一化的 Wishart

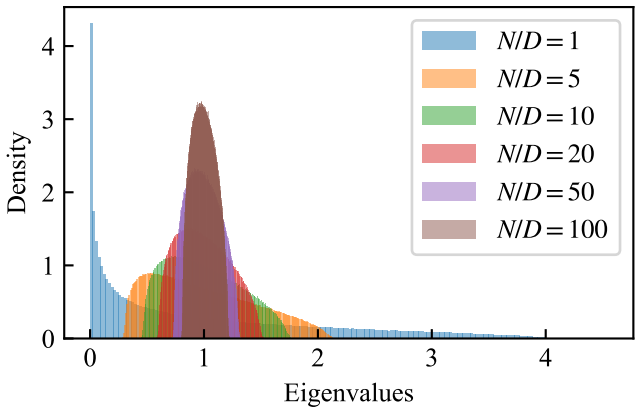

图 7　Wishart 随机矩阵的谱密度。$D = 100$，结果来自对 10000 个矩阵的统计。

随机矩阵。我们在图 7 中展示了 Wishart 随机矩阵的谱密度，可以看到，随着 N 的增大，矩阵的本征值逐渐集中在 1 附近。

实际上，对于 N 足够大的情况，我们可以考虑将 Wishart 矩阵看作一个单位矩阵和一个噪声矩阵的和。定义噪声矩阵 $\boldsymbol{\epsilon} = \mathbf{C}_0 - \mathbf{I}_D$，其对角项为 $\boldsymbol{\epsilon}_{ii} = \frac{1}{N}\sum_k \mathbf{X}_{0,ik}^2 - 1$，其中 $\sum_k \mathbf{X}_{0,ik}^2 \sim \chi^2(N)$ 服从卡方分布；非对角项为 $\boldsymbol{\epsilon}_{ij} = \frac{1}{N}\sum_k \mathbf{X}_{0,ik}\mathbf{X}_{0,jk}$，其中 $\mathbf{X}_{0,ik}$、$\mathbf{X}_{0,jk} \sim \mathcal{N}(0,1)$ 服从高斯分布。根据中心极限定理，$\frac{1}{N}\chi^2(N) \approx \mathcal{N}(1, 2/N)$，我们可以看出对角项的方差为 $2/N$，而非对角项的方差为 $1/N$。这种小方差的涨落允许我们在实践中取 $\mathbf{C}_0 \approx \mathbf{I}_D$，从而将式 (157) 简化为

$$\mathbf{W}_{11} + \frac{1}{\sqrt{D}}\boldsymbol{w}_\star \mathbf{W}_{21} = \mathbf{I}_D. \tag{158}$$

式 (158) 可以写成一个具有 $D^2 + D$ 个变量的线性方程组。给定一个输入矩阵 $\mathbf{X}$，就是指定了一个任务向量 $\boldsymbol{w}_\star$，也就写出了关于式 (158) 的 D^2 个方程。因此，线性方程组的解理论告诉我们：当输入矩阵 $P = 1$ 时，权重矩阵 $\mathbf{W}$ 有无穷多解；而当 $P > 1$ 时，方程组有唯一解，即 $\mathbf{W}_{11}^\star = \mathbf{I}_D$，$\mathbf{W}_{21}^\star = \mathbf{0}$。

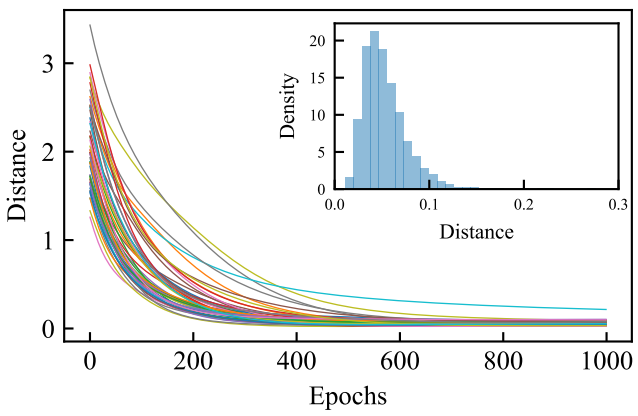

图 8　在 50 次的不同初始化下，方程 (157) 的解与方程 (158) 的最优解的欧氏距离在 SGD 训练过程中的变化。右上角的插图：5000 次的不同初始化下，求解出的欧氏距离的统计分布。

为了进一步验证，我们使用 SGD 算法直接求解方程 (157)，然后计算这个解与方程 (158) 的最优解的欧几里得距离，定义为 $\text{Distance} = \|\mathbf{W}_{11} - \mathbf{I}_D\| + \|\mathbf{W}_{21} - \mathbf{0}\|$。我们取多次不同的初始化，即给定不同的 $\mathbf{C}_0$ 和 $\boldsymbol{w}_\star$，结果如图 8 所示，只有极少数初始化产生了较大的距离，这表明我们的近似是合理的。

接下来，我们选取了 $P = 10$ 和 $P = 1000$ 两种不同的数据量进行 SGD 训练，结果如图 9 (a) 所示。对于 $P = 10$ 的情况，模型的测试误差非常大，训练结束后的权重矩阵如图 9 (b) 所示，并没有达到我们期望的最优解 $\mathbf{W}_{11} = \mathbf{I}_D$ 和 $\mathbf{W}_{21} = \mathbf{0}$。而对于 $P = 1000$ 的情况，模型的测试误差已经收敛到一个很小的值，训练结束后的权重矩阵如图 9 (c) 所示，十分接近理论上的最优解。同时，我们也使用 AMP 算法进行迭代，将其结果还原成矩阵形式，如图 9 (d) 和 (e) 所示。可以看到，AMP 算法得到的均值与理论上的最优解一致。由于权重矩阵的最后一列不参与训练过程，并且初始化为 $\mathcal{N}(0, 1)$，因此 AMP 的结果中对应的方差为 1。

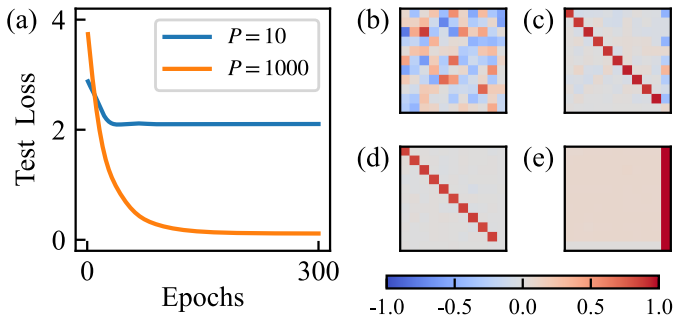

图 9　ICL 任务的测试误差和权重矩阵。图 (a)：$P = 10$ 和 $P = 1000$ 对应的 SGD 测试误差。图 (b) 和 (c)：图 (a) 中两条曲线末端对应的权重矩阵。图 (d) 和 (e)：AMP 算法得到的均值和方差。图 (a) 到 (e) 共享右下角的颜色条。

5.2　模型的凸性与能量景观

在机器学习中通常使用海森矩阵来描述损失函数的曲率，从而判断模型的凸性和优化难度。如果海森矩阵是正定的，即所有特征值都是正的，那么损失函数是严格凸的，只有一个全局最小值，模型的优化会非常简单；如果海森矩阵是半正定的，即所有特征值都是非负的，那么损失函数是凸的，但是可能存在局部平坦的区域；如果海森矩阵有负特征值，那么损失函数是非凸的，可能存在鞍点或者局部极小值，此时模型的优化难度会非常大。

对于能量景观 (128)，我们可以计算其海森矩阵为[141]

$$\mathbf{H}_{ij} = \frac{\partial^2 \mathcal{H}}{\partial \sigma_i \partial \sigma_j} = \frac{1}{P} \sum_{\mu=1}^{P} s_i^\mu s_j^\mu. \tag{159}$$

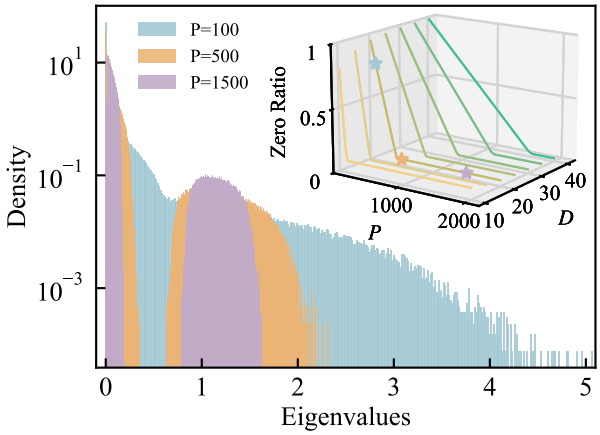

图 10　不同数据量 P 对应的海森矩阵的谱密度，$D = 20$，$N = 100$，结果来自对 5 000 个矩阵的统计。右上角的插图：不同 P 和 D 的情况下零本征值的比例。

即 $\mathbf{H} = \langle \boldsymbol{s}\boldsymbol{s}^{\top} \rangle$。对于任意一个向量 $\boldsymbol{v} \in \mathbb{R}^{D}$，我们可以计算

$$\boldsymbol{v}^{\top}\mathbf{H}\boldsymbol{v} = \boldsymbol{v}^{\top} \langle \boldsymbol{s}\boldsymbol{s}^{\top} \rangle \boldsymbol{v} = \left\langle \left(\boldsymbol{v}^{\top}\boldsymbol{s}\right)^{2} \right\rangle \geq 0, \tag{160}$$

这表明海森矩阵 $\mathbf{H}$ 是半正定的，因此我们的模型是一个凸优化问题。

在图 10 中，我们展示了海森矩阵的谱密度，可以看到，$\mathbf{H}$ 的所有本征值都是非负的，但是在 P 比较小的时候，会有大量的零本征值。图 10 右上角的插图展示了不同 P 和 D 的情况下零本征值的比例。在 D 比较大而 P 比较小（即训练数据不足）的情况下，零本征值的比例甚至达到了 98% 以上。

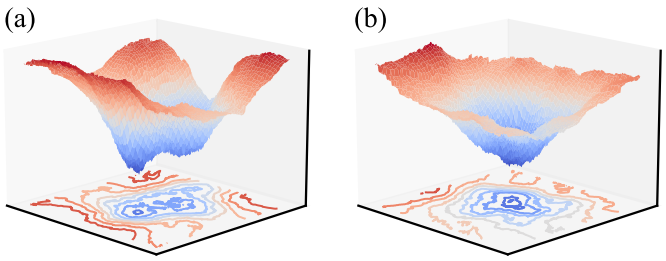

图 11　自旋玻璃模型的能量景观，$D = 10$，$N = 50$，$\lambda = 0.01$，图 (a) 对应 $P = 10$，图 (b) 对应 $P = 1000$。为了绘制这个景观，我们首先从整个权重空间中随机采样 1 000 个点通过式 (128) 计算每个点的能量值。然后，我们通过 T 分布随机近邻嵌入（t-SNE）算法[142]将 $(D+1)^2$ 维的权重空间映射到二维平面上，通过径向基函数（RBF）插值算法绘制能量曲面。红色表示能量高的区域，蓝色表示能量低的区域。

在图 11 中，我们展示了自旋玻璃模型 (128) 的能量景观。我们可以看到，在 P 比较小的时候，由于海森矩阵存在大量的零本征值，能量景观表现出局部平坦的特征，此时的训练动力学容易被困在这些平坦谷地中，导致模型的训练效果不好，这与玻尔兹曼机器的学习特征很相似[143]。当 P 比较大的时候，能量景观表现出只有一个全局最小值，此时模型很容易到达全局最优解。

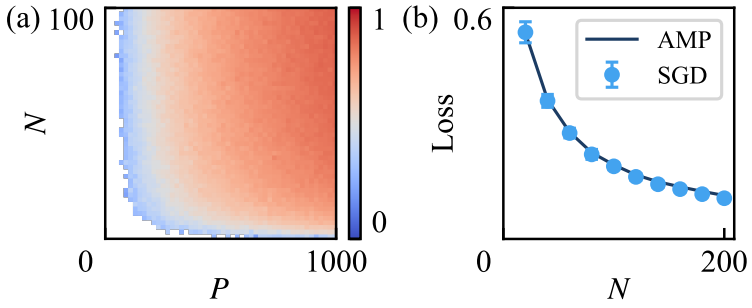

图 12　AMP 算法得到的结果及其与 SGD 结果的比较。图 (a)：对比度 $\mathscr{C}$ 的热力图，$D = 20$，$\lambda = 1$，$\beta = 100$，左下角的空白区域代表 AMP 算法没有收敛。图 (b)：AMP 与 SGD 得到的泛化误差的比较。$P = 10\,000$，$D = 20$，$\lambda = 0,01$，$\beta = 100$。所有的结果对 50 次实验取平均值。

5.3　近似消息传递算法的结果

为了衡量我们的模型是否捕捉到了理论最优解的特征，我们定义 AMP 算法得到的均值 $\mathbf{m}$ 和单位矩阵 $\mathbf{I}_D$ 之间的余弦相似度为对比度

$$\mathscr{C} = \left[\frac{\langle \mathbf{m}, \mathbf{I}_D \rangle}{\|\mathbf{m}\|^2 \|\mathbf{I}_D\|^2}\right] \tag{161}$$

其中 $\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{ij} \mathbf{A}_{ij}\mathbf{B}_{ij}$ 表示矩阵内积，方括号 $[\cdots]$ 表示对数据集的平均。对比度 $\mathscr{C}$ 的值越接近 1，说明 AMP 算法预测的模型效果越好。

我们绘制了对比度 $\mathscr{C}$ 的热力图，如图 12 (a) 所示，展示了对比度 $\mathscr{C}$ 随着任务多样性 P 和提示长度 N 的变化情况。可以看到，当任务多样性增加时，模型的训练效果逐渐变好，最终收敛到一个全局最优解；而当任务多样性保持较大值时，增加提示长度会进一步降低模型的泛化误差，这与最近的实证研究[50] 和基于随机矩阵的理论研究[52] 一致（尽管设置略有不同）。此外，AMP 算法得到的结果与 SGD 算法得到的结果完全一致，如图 12 (b) 所示，这再次验证了我们的自旋玻璃模型能够预测在提示中未见过的嵌入函数的能力，该函数由 $\mathcal{H}(\boldsymbol{\sigma})$ 的基态决定。

5.4　在更复杂的任务和模型上的结果

5.4.1　在非独立同分布任务上的结果

在先前的实验中，我们只考虑了数据都是独立同分布的任务。为了验证我们的理论能否胜任更复杂的情况，我们模仿语言任务中的上下文关联，为矩阵 $\mathbf{X}_0$ 中的元素引入一些关联性。具体来说，我们考虑 $\mathbf{X}_0$ 中的元素具有如下协方差[144]

$$\langle x_{ij} x_{kl} \rangle = \delta_{ik} \left[c + (1-c)\delta_{jl}\right]. \tag{162}$$

此时图 5 (c) 中的相关分布改为如图 13 所示。

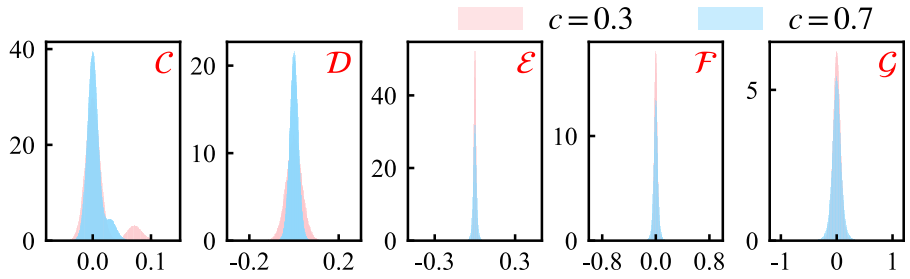

图 13　具有不同协方差 c 的数据产生的不同分块的元素分布。

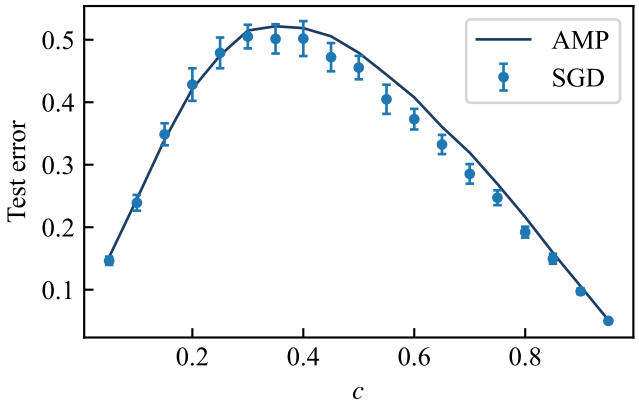

图 14　AMP 和 SGD 的泛化误差随着协方差 c 的变化。$P = 10\,000$，$D = 20$，$N = 200$，$\lambda = 0.01$，$\beta = 100$。所有结果都对 20 次实验取平均值。

保持其他设置不变，我们绘制了 SGD 和 AMP 在这个上下文相关的数据集上的表现，如图 14 所示。尽管泛化误差与 c 的关系并不是单调的，但 AMP 的结果仍然能够与 SGD 匹配。

5.4.2　在更复杂的模型上的结果

为了验证我们的理论在更复杂的注意力结构下的鲁棒性，我们在不同的模型上进行了实验。我们考虑以下几种模型：

1. 简化的线性注意力（Simplified Linear Attention，SLA）模型，形式为

$$\mathbf{Y}_{\mathrm{SLA}} = \frac{1}{N}\mathbf{X}\mathbf{X}^{\top}\mathbf{W}\mathbf{X}. \tag{163}$$

其中 $\mathbf{W} \in \mathbb{R}^{(D+1)\times(D+1)}$。

2. 全线性注意力（Full Linear Attention，FLA）模型，形式为

$$\mathbf{Y}_{\mathrm{FLA}} = \frac{1}{N}\mathbf{W}_{\mathrm{V}}\mathbf{X}\mathbf{X}^{\top}\mathbf{W}\mathbf{X}. \tag{164}$$

其中 $\mathbf{W}, \mathbf{W}_{\mathrm{V}} \in \mathbb{R}^{(D+1)\times(D+1)}$。

3. 缩放点积注意力，即 Softmax 注意力（Softmax Attention，SA）模型，形式为

$$\mathbf{Y}_{\mathrm{SA}} = \mathbf{W}_{\mathrm{V}}\mathbf{X} \cdot \mathrm{Softmax}\left(\frac{(\mathbf{W}_{\mathrm{Q}}\mathbf{X})^{\top}\mathbf{W}_{\mathrm{K}}\mathbf{X}}{\sqrt{D}}\right). \tag{165}$$

其中 $\mathbf{W}_{\mathrm{Q}}, \mathbf{W}_{\mathrm{K}} \in \mathbb{R}^{H\times(D+1)}$、$\mathbf{W}_{\mathrm{V}} \in \mathbb{R}^{(D+1)\times(D+1)}$。

4. 小型完整 Transformer 编码器（TF）模型，形式为

$$\mathbf{Y}_{\mathrm{TF}} = \mathbf{z}_1 + \mathbf{z}_3, \tag{166}$$

$$\mathbf{z}_1 = \mathbf{Y}_{\mathrm{SA}}(\mathbf{X}) + \mathbf{X}, \tag{167}$$

$$\mathbf{z}_2 = \mathrm{ReLU}\left(\mathbf{W}_1 \cdot \mathbf{z}_1 + b_1\right), \tag{168}$$

$$\mathbf{z}_3 = \mathbf{W}_2 \cdot \mathbf{z}_2 + b_2, \tag{169}$$

其中 $\mathbf{Y}_{\mathrm{SA}}(\mathbf{X})$ 是标准的自注意力机制 (165)，$\mathbf{W}_1$、$\mathbf{W}_2$ 和 b_1、b_2 是前馈网络的权重和偏置。

5. 多头线性注意力（Multi-head Linear Attention，MHLA）模型，形式为

$$\mathbf{Y}_{\mathrm{MHLA}} = \mathbf{W}_O\left[\mathbf{Y}^1_{\mathrm{FLA}}, \mathbf{Y}^2_{\mathrm{FLA}}, \cdots, \mathbf{Y}^M_{\mathrm{FLA}}\right], \tag{170}$$

其中 $\mathbf{Y}^i_{\mathrm{FLA}}$ 是第 i 个头的输出，$\mathbf{W}_O \in \mathbb{R}^{D\times(D+1)M}$ 是可学习的读出矩阵，M 是注意力头的数量。

我们使用以上模型对第 4.1 小节中介绍的线性回归任务进行训练，结果如图 15 所示。可以看出，SLA、FLA 和 MHLA 三种线性模型的泛化误差比较接近，也与我们的理论结果一致。SA 和 TF 模型的泛化误差较高，这是因为线性模型在线性任务中更具计算优势，而非线性机制在此任务中较难获得简洁的解析结果。

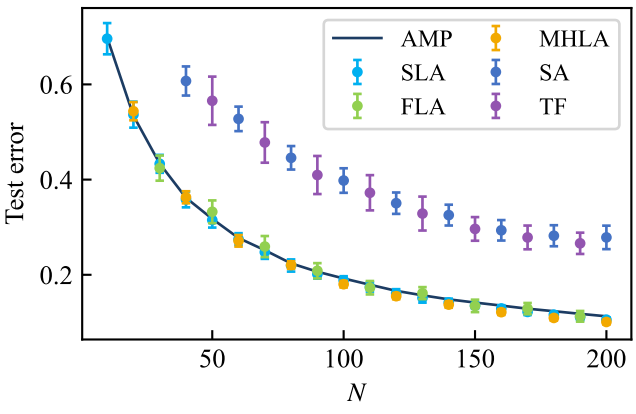

图 15　不同 Transformer 设置下的泛化误差的比较。$\lambda = 0.01$，$P = 10\,000$，$D = 20$，AMP 的 $\beta = 100$，SA 和 TF 的 $H = 32$，MHLA 的 $M = 10$。所有的结果都对 100 次独立实验取平均。

5.5 讨论

5.5.1 合并权重矩阵与分离权重矩阵的模型在动力学上的区别

在本节中，我们讨论前文中使用的合并的权重矩阵 $\mathbf{W}$ 的模型和使用分离的权重矩阵 $\mathbf{W}_{\mathrm{Q}}$ 和 $\mathbf{W}_{\mathrm{K}}$ 的标准线性注意力模型之间的区别。具体来说，我们考虑如下两种线性注意力机制：

1. 合并权重的线性注意力：

$$\mathbf{Y}_{\mathrm{merged}} = \frac{1}{N}\mathbf{X}\mathbf{X}^{\top}\mathbf{W}\mathbf{X}, \tag{171}$$

2. 分离权重的线性注意力：

$$\mathbf{Y}_{\mathrm{separate}} = \frac{1}{N}\mathbf{X}\mathbf{X}^{\top}\mathbf{W}_{\mathrm{Q}}^{\top}\mathbf{W}_{\mathrm{K}}\mathbf{X}, \tag{172}$$

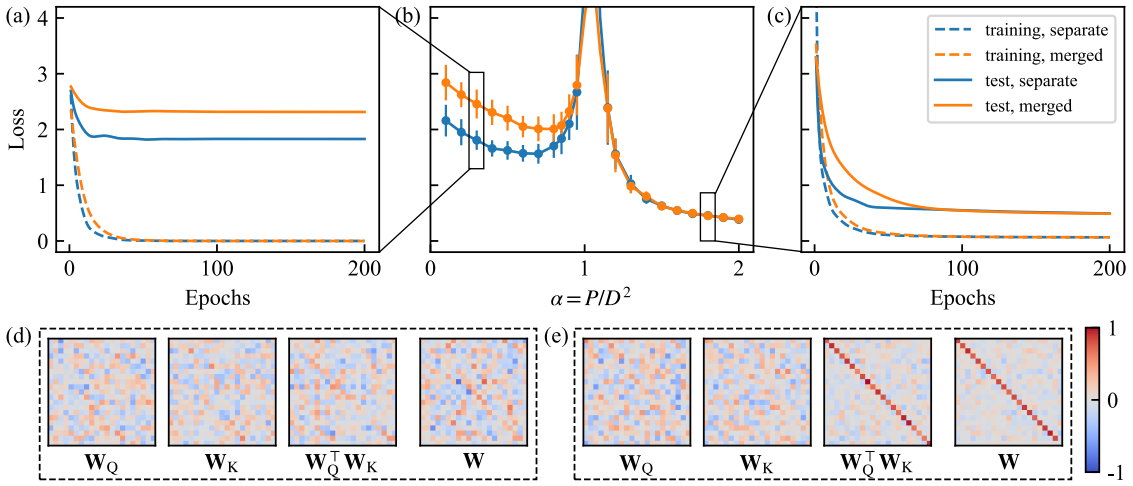

图 16　使用 SGD 训练的合并权重和分离权重模型的损失和权重矩阵。(a) 数据欠缺情况下（$\alpha = 0.3$）的训练和测试损失。(b) 两种模型的测试误差随着数据密度变化表现出双下降现象。(c) 数据丰富情况下（$\alpha = 1.8$）的训练和测试损失。(d) 两种模型在 (a) 图动力学结束时的权重矩阵。(e) 两种模型在 (c) 图动力学结束时的权重矩阵。参数：$D = 20$，$N = 100$，$\lambda = 0$。图 (b) 的结果取 50 次实验的平均值。

两种方案的泛化误差在无岭极限 $\lambda \to 0$ 下均表现出预期的双下降现象，如图 16 (b) 所示。在数据欠缺（$\alpha \equiv P/D^2 < 1$）的情况下，两种方案的训练过程动力学相似，分离权重模型 $\mathbf{Y}_{\mathrm{separate}}$ 的泛化误差相对较低，如图 16 (a) 所示，但是两种模型都未能捕捉到正确的权重结构，如图 16 (d) 所示。在数据丰富（$\alpha \equiv P/D^2 > 1$）的情况下，分离权重模型在测试数据集上的收敛速度更快，但两种方案的最终泛化性能非常相近，如图 16 (c) 所示，并且两种模型的权重矩阵都接近于对角结构，与理论结果一致。这些结果为我们在前文中使用的合并权重模型提供了合理性。

5.5.2 我们的模型能否看作一个纯自旋玻璃模型?

在前文中，我们使用实际任务的数据构造了相互作用 $\mathbf{J}$ 和外场 $\boldsymbol{h}$，并求解了式 (128) 所描述的自旋模型。一个自然的问题是，我们的模型是否可以看作一个纯自旋玻璃问题，即 $\mathbf{J}$ 和 $\boldsymbol{h}$ 是直接从某种分布中采样的，而不是由数据构造的。

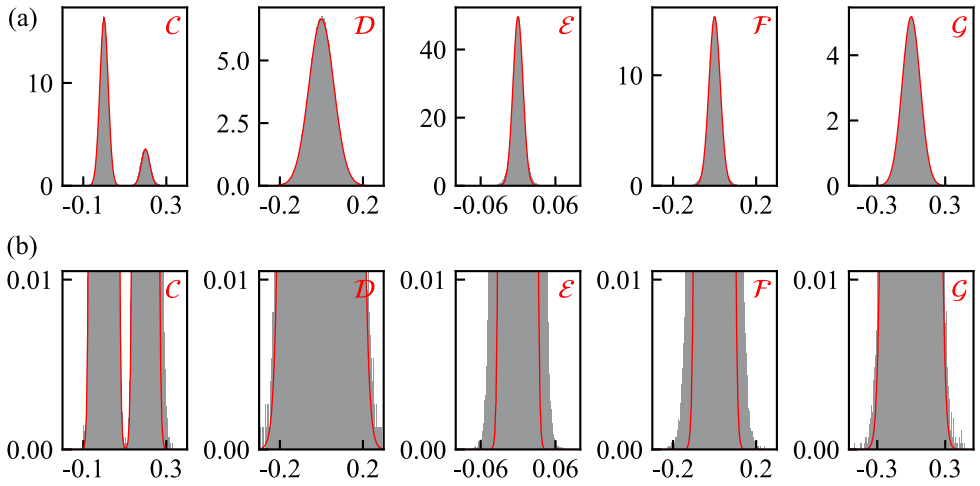

图 17　使用高斯分布和混合高斯分布拟合的各个分块的元素分布。图 (a) 展示了完整的概率密度函数，图 (b) 展示了概率密度函数底部拟合效果不好的区域。

在数学上很难通过式 (132) 得到 $\mathcal{C}$、$\mathcal{D}$ 分块的分布对应的概率密度函数，对于分块 $\mathcal{E}$、$\mathcal{F}$ 和 $\mathcal{G}$ 来说就更加困难了。但我们可以通过数值方法，用简单的高斯分布或高斯混合分布对其进行近似拟合。如图 17 (a) 所示，我们在 $P = 1\,000$、$D = 10$、$N = 100$ 的情况下对所有分块的分布进行了拟合，但是拟合得到的概率密度函数丢失了一些底部的细节，如图 17 (b) 所示。

然后，我们从数值拟合得到的概率密度函数中随机重新采样生成 $\mathbf{J}$ 和 $\boldsymbol{h}$，如图 18 (a) (b) 所示。我们使用相同的参数运行 AMP 方程，结果如图 18 (c) (d) 所示，表明 AMP 仍然能够捕捉到最优解的对角结构。

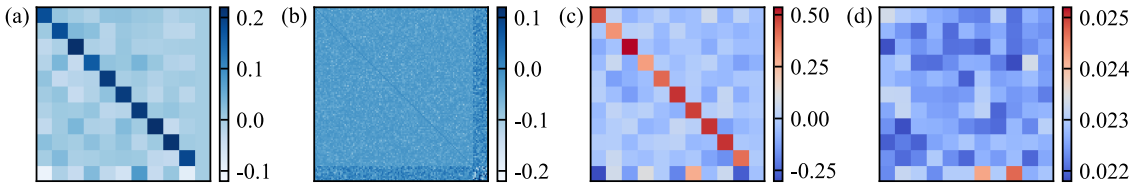

图 18　(a) 从拟合的概率密度函数中重新采样生成的 $\mathbf{H}$ 矩阵。(b) 从拟合的概率密度函数中重新采样生成的 $\mathbf{J}$ 矩阵。(c) 通过 AMP 方程求解得到的权重矩阵的均值 $\mathbf{m}$。(d) 通过 AMP 方程求解得到的权重矩阵的方差 $\mathbf{v}$。参数：$P = 1\,000$，$D = 10$，$N = 100$，$\lambda = 1$。

这些结果表明，我们的模型可以看作一个纯自旋玻璃模型，尽管在实际任务中 $\mathbf{J}$ 和 $\boldsymbol{h}$ 是由数据构造的，但是只要能够得到其概率密度函数，就可以直接采样得到 $\mathbf{J}$ 和 $\boldsymbol{h}$，并且使用 AMP 方程进行求解。

6 总结与展望

本文的工作旨在探究大语言模型的上下文学习（ICL）能力背后的机制，并尝试将其与统计物理理论建立联系。尽管大语言模型在各种任务中展现出强大的能力，尤其是仅通过提示和少量示例就能完成未见过的新任务，但其内在的工作原理，特别是 ICL 的涌现机制，仍然缺乏清晰的理论解释。

针对这一挑战，我们首先回顾了 Transformer 模型的结构、大语言模型的构建流程及其提示学习方法，特别是上下文学习。随后，我们详细介绍了统计物理中处理无序系统的重要工具——自旋玻璃模型的平均场理论，包括复本方法和空腔方法，为后续的理论分析奠定了基础。

本文的核心工作集中在第四章和第五章。我们研究了一种简化的、使用线性注意力机制的 Transformer 结构。通过巧妙的映射，我们将该模型的预训练过程与一个全连接的自旋玻璃模型联系起来。在这个映射中，模型的权重参数对应于实数值的自旋，而预训练数据的结构和任务的多样性则体现在自旋间的相互作用 J_{ij} 和外场 h_i 的统计性质中。我们分析了这些相互作用和外场的统计分布特性，发现它们具有复杂的分块结构，并且其分布受到输入数据（如提示长度 N）的影响。

基于建立的自旋玻璃模型，我们运用了平均场理论进行求解。我们推导了该模型的信念传播（BP）方程，并进一步简化为计算效率更高的松弛信念传播（r-BP）方程和近似消息传递（AMP）方程。通过求解这些方程，我们能够有效地找到模型的基态，这对应于预训练后 Transformer 的最优权重参数。

我们的理论分析和数值模拟结果揭示了以下关键点：

1. 预训练任务多样性 P 对模型的学习能力起着至关重要的作用。任务多样性较低时，自旋模型的能量景观存在许多局部极小值或平坦区域，模型难以学习到最优解。随着任务多样性增加，能量景观转变为具有唯一全局最小值的形态，模型在预训练过程中更容易收敛到对应于良好 ICL 能力的基态。

2. 我们通过 AMP 算法找到自旋玻璃模型的基态，得到模型权重的理论最优解，与通过随机梯度下降（SGD）训练得到的实际权重高度吻合，验证了我们理论框架的有效性。

3. 我们的理论框架在更复杂的模型和任务上也表现出良好的鲁棒性。我们对不同的注意力机制进行了实验，同时也考虑了数据的非独立同分布情况，结果表明我们的理论能够很好地捕捉到模型的学习特性。

我们的工作的主要贡献在于：

1. 首次将 Transformer（简化形式）与可解析求解的统计物理模型（自旋玻璃模型）建立了直接联系。

2. 开发了基于平均场理论（特别是 AMP 算法）的方法来预测和分析 Transformer 模型的学习结果。

3. 提供了一个基于能量景观变化的理论视角来解释任务多样性如何驱动 LLMs 中 ICL 能力的涌现。

尽管我们的工作为理解 ICL 提供了一个新的理论视角，但仍存在一些局限性，并为未来的研究指明了方向：

1. 更复杂的模型：我们主要分析了单层线性注意力模型，未来的工作可以将此框架扩展到更复杂的模型，例如包含 Softmax 非线性注意力、多层结构、前馈网络（FFN）、层归一化（LayerNorm）等组件的完整 Transformer 模型。研究这些组件如何在自旋玻璃图像中体现，以及它们对能量景观和学习动力学的影响，将是重要的方向。

2. 更复杂的任务：我们的研究集中在线性回归任务，将模型扩展到更复杂的任务，如分类任务、序列生成任务，甚至模拟自然语言任务的某些特征，将有助于检验理论的普适性。

3. 动力学过程：我们主要关注模型的平衡态性质，即预训练完成后的最优权重。使用动力学平均场理论（DMFT）分析和研究权重在训练过程中的演化动力学，可能会揭示更多关于学习效率和泛化能力形成的细节。

4. 与其他理论的联系：探索我们的自旋玻璃模型与其他解释 ICL 的理论（如隐式梯度下降、贝叶斯推断）之间的联系，可能会形成更统一的理解。

我们希望本文的工作能够为理解大语言模型中的 ICL 现象提供新的视角，并激励更多的研究者在这一领域进行深入探索。通过将统计物理与机器学习结合，我们不仅可以更好地理解现有模型的行为，还可以为未来的模型设计和优化提供理论指导。

参考文献

[1] Rabiner L. A tutorial on hidden Markov models and selected applications in speech recognition [J]. Proceedings of the IEEE, 1989, 77(2): 257-286.

[2] Brown P F, deSouza P V, Mercer R L, et al. Class-based n-gram models of natural language[J]. Computational Linguistics, 1992, 18(4): 467-479.

[3] Berger A L, Pietra V J D, Pietra S A D. A maximum entropy approach to natural language processing[J]. Computational Linguistics, 1996, 22(1): 39-71.

[4] Lafferty J, McCallum A, Pereira F, et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data[C]//Proceedings of the Eighteenth International Conference on Machine Learning. 2001: 282-289.

[5] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[A]. 2013. arXiv: 1301.3781.

[6] Pennington J, Socher R, Manning C D. GloVe: Global Vectors for Word Representation[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1532-1543.

[7] Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model[C]// Interspeech. 2010: 1045-1048.

[8] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]// Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2. 2014: 3104-3112.

[9] Vaswani A, Shazeer N, Parmar N, et al. Attention is All you Need[C]//Advances in Neural Information Processing Systems: Vol. 30. 2017.

[10] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019: 4171-4186.

[11] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[J]. OpenAI blog, 2018.

[12] Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners[J]. OpenAI blog, 2019.

[13] Raffel C, Shazeer N, Roberts A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. The Journal of Machine Learning Research, 2020, 21(1).

[14] Howard J, Ruder S. Universal Language Model Fine-tuning for Text Classification[C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018: 328-339.

[15] Kaplan J, McCandlish S, Henighan T, et al. Scaling Laws for Neural Language Models[A]. 2020. arXiv: 2001.08361.

[16] Zhao W X, Zhou K, Li J, et al. A Survey of Large Language Models[A]. 2025. arXiv: 2303.18223.

[17] Brown T, Mann B, Ryder N, et al. Language Models are Few-Shot Learners[C]//Advances in Neural Information Processing Systems: Vol. 33. 2020: 1877-1901.

[18] Wei J, Wang X, Schuurmans D, et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models[C]//Advances in Neural Information Processing Systems: Vol. 35. 2022: 16385-16396.

[19] Dong Q, Li L, Dai D, et al. A Survey on In-context Learning[C]//Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. 2024: 1107-1128.

[20] Lu S, Bigoulaeva I, Sachdeva R S, et al. Are Emergent Abilities in Large Language Models just In-Context Learning?[C]//Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics. 2024: 5098-5139.

[21] Ahuja K, Lopez-Paz D. A Closer Look at In-Context Learning under Distribution Shifts[C]// Workshop on Efficient Systems for Foundation Models @ ICML2023. 2023.

[22] Lee I, Jiang N, Berg-Kirkpatrick T. Is attention required for ICL? Exploring the Relationship Between Model Architecture and In-Context Learning Ability[A]. 2024. arXiv: 2310.08049.

[23] Park J, Park J, Xiong Z, et al. Can Mamba Learn How to Learn? A Comparative Study on In-Context Learning Tasks[C]//Proceedings of the 41st International Conference on Machine Learning. 2024: 39793-39812.

[24] Tong W L, Pehlevan C. MLPs Learn In-Context on Regression and Classification Tasks[A]. 2024. arXiv: 2405.15618.

[25] Wei J, Tay Y, Bommasani R, et al. Emergent Abilities of Large Language Models[J]. Transactions on Machine Learning Research, 2022.

[26] Wei J, Bosma M, Zhao V, et al. Finetuned Language Models are Zero-Shot Learners[C]// International Conference on Learning Representations. 2022.

[27] Chan S, Santoro A, Lampinen A, et al. Data distributional properties drive emergent in-context learning in transformers[C]//Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022: 18878-18891.

[28] Min S, Lyu X, Holtzman A, et al. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?[A]. 2022. arXiv: 2202.12837.

[29] Pan J, Gao T, Chen H, et al. What In-Context Learning "Learns" In-Context: Disentangling Task Recognition and Task Learning[C]//Findings of the Association for Computational Linguistics: ACL 2023. 2023: 8298-8319.

[30] Garg S, Tsipras D, Liang P S, et al. What Can Transformers Learn In-Context? A Case Study of Simple Function Classes[C]//Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022: 30583-30598.

[31] Von Oswald J, Niklasson E, Randazzo E, et al. Transformers Learn In-Context by Gradient Descent[C]. International Conference on Machine Learning, ICML, 2023.

[32] Dai D, Sun Y, Dong L, et al. Why Can GPT Learn In-Context? Language Models Implicitly Perform Gradient Descent as Meta-Optimizers[C]//ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models. 2023.

[33] Ahn K, Cheng X, Daneshmand H, et al. Transformers learn to implement preconditioned gradient descent for in-context learning[C]//Proceedings of the 37th International Conference on Neural Information Processing Systems. 2023: 45614 - 45650.

[34] Cheng X, Chen Y, Sra S. Transformers Implement Functional Gradient Descent to Learn Non-Linear Functions In Context[C]//Proceedings of the 41st International Conference on Machine Learning: Vol. 235. 2024: 8002-8037.

[35] Shen L, Mishra A, Khashabi D. Do pretrained Transformers Learn In-Context by Gradient Descent?[A]. 2023. arXiv: 2310.08540.

[36] Deutch G, Magar N, Natan T, et al. In-context Learning and Gradient Descent Revisited[C]. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), 2024.

[37] Xie S M, Raghunathan A, Liang P, et al. An Explanation of In-context Learning as Implicit Bayesian Inference[C]//Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers). 2021: 1017-1028.

[38] Panwar M, Ahuja K, Goyal N. In-Context Learning through the Bayesian Prism[C]//The Twelfth International Conference on Learning Representations. 2024.

[39] Wang X, Zhu W, Saxon M, et al. Large Language Models Are Latent Variable Models: Explaining and Finding Good Demonstrations for In-Context Learning[C]//Proceedings of the 37th International Conference on Neural Information Processing Systems. 2023: 15614 - 15638.

[40] Bai Y, Chen F, Wang H, et al. Transformers as Statisticians: Provable In-Context Learning with In-Context Algorithm Selection[C]//Proceedings of the 37th International Conference on Neural Information Processing Systems. 2023: 57125 - 57211.

[41] Hahn M, Goyal N. A Theory of Emergent In-Context Learning as Implicit Structure Induction [A]. 2023. arXiv: 2303.07971.

[42] Li Y, Ildiz M E, Papailiopoulos D, et al. Transformers as Algorithms: Generalization and Stability in In-context Learning[C]//Proceedings of the 40th International Conference on Machine Learning. 2023: 19565 - 19594.

[43] Han C, Wang Z, Zhao H, et al. Explaining Emergent In-Context Learning as Kernel Regression [A]. 2023. arXiv: 2305.12766.

[44] Chen S, Sheen H, Wang T, et al. Unveiling Induction Heads: Provable Training Dynamics and Feature Learning in Transformers[C]//The Thirty-eighth Annual Conference on Neural Information Processing Systems. 2024.

[45] Akyürek E, Schuurmans D, Andreas J, et al. What learning algorithm is in-context learning? Investigations with linear models[C]//The Eleventh International Conference on Learning Representations. 2023.

[46] Zhang R, Frei S, Bartlett P L. Trained transformers learn linear models in-context[J]. Journal of Machine Learning Research, 2024, 25(49): 1-55.

[47] Kim J, Suzuki T. Transformers Learn Nonlinear Features In Context: Nonconvex Mean-field Dynamics on the Attention Landscape[C]//Proceedings of the 41st International Conference on Machine Learning. 2024: 24527-24561.

[48] Reddy G. The mechanistic basis of data dependence and abrupt learning in an in-context classification task[C]//12th International Conference on Learning Representations. 2023.

[49] Li H, Wang M, Lu S, et al. How Do Nonlinear Transformers Learn and Generalize in In-Context Learning?[C]//Proceedings of the 41st International Conference on Machine Learning. 2024: 28734 - 28783.

[50] Raventós A, Paul M, Chen F, et al. Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression[C]//Proceedings of the 37th International Conference on Neural Information Processing Systems. 2023: 14228-14246.

[51] Huang Y, Cheng Y, Liang Y. In-Context Convergence of Transformers[C]. 2023: 19660-19722.

[52] Lu Y M, Letey M I, Zavatone-Veth J A, et al. Asymptotic theory of in-context learning by linear attention[A]. 2024. arXiv: 2405.11751.

[53] Chen S, Sheen H, Wang T, et al. Training Dynamics of Multi-Head Softmax Attention for In-Context Learning: Emergence, Convergence, and Optimality[A]. 2024. arXiv: 2402.19442.

[54] Wu J, Zou D, Chen Z, et al. How Many Pretraining Tasks Are Needed for In-Context Learning of Linear Regression?[C]//Advances in Neural Information Processing Systems: Vol. 36. 2024: 14228-14246.

[55] Su J, Ahmed M, Lu Y, et al. Roformer: Enhanced transformer with rotary position embedding [J]. Neurocomputing, 2024, 568: 127063.

[56] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 770-778.

[57] Ba J L, Kiros J R, Hinton G E. Layer Normalization[A]. 2016. arXiv: 1607.06450.

[58] Zhang B, Sennrich R. Root Mean Square Layer Normalization[C]//Proceedings of the 33rd International Conference on Neural Information Processing Systems. 2019: 12381 - 12392.

[59] Liu Y, Ott M, Goyal N, et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach [C]//Proceedings of the 20th Chinese National Conference on Computational Linguistics. 2019: 1218-1227.

[60] Touvron H, Lavril T, Izacard G, et al. LLaMA: Open and Efficient Foundation Language Models [A]. 2023. arXiv: 2302.13971.

[61] Touvron H, Martin L, Stone K, et al. Llama 2: Open Foundation and Fine-Tuned Chat Models [A]. 2023. arXiv: 2307.09288.

[62] AI at Meta. Introducing Meta Llama 3: The most capable openly available LLM to date[J]. Meta AI Blog Post, 2024.

[63] DeepSeek-AI, :, Bi X, et al. DeepSeek LLM: Scaling Open-Source Language Models with Longtermism[A]. 2024. arXiv: 2401.02954.

[64] DeepSeek-AI, Liu A, Feng B, et al. DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model[A]. 2024. arXiv: 2405.04434.

[65] DeepSeek-AI, Liu A, Feng B, et al. DeepSeek-V3 Technical Report[A]. 2025. arXiv: 2412.19437.

[66] Lewis M, Liu Y, Goyal N, et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2019: 7871-7880.

[67] Du Z, Qian Y, Liu X, et al. GLM: General Language Model Pretraining with Autoregressive Blank Infilling[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2022: 320-335.

[68] GLM T, Zeng A, Xu B, et al. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools[A]. 2024. arXiv: 2406.12793.

[69] Yang J, Jin H, Tang R, et al. Harnessing the power of llms in practice: A survey on chatgpt and beyond[J]. ACM Transactions on Knowledge Discovery from Data, 2024, 18(6): 1-32.

[70] Hu Q, Ye Z, Wang Z, et al. Characterization of large language model development in the datacenter[C]//Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation. 2024: 709 - 729.

[71] Zhu Y, Kiros R, Zemel R, et al. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books[C]//Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV). 2015: 19-27.

[72] Sennrich R, Haddow B, Birch A. Neural Machine Translation of Rare Words with Subword Units[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 1715-1725.

[73] Wu Y, Schuster M, Chen Z, et al. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation[A]. 2016. arXiv: 1609.08144.

[74] Kudo T. Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates[C]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018: 66-75.

[75] Kudo T, Richardson J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2018: 66-71.

[76] Gao L, Biderman S, Black S, et al. The Pile: An 800GB Dataset of Diverse Text for Language Modeling[A]. 2020. arXiv: 2101.00027.

[77] Laurençon H, Saulnier L, Wang T, et al. The BigScience ROOTS Corpus: A 1.6TB Composite

Multilingual Dataset[C]//Advances in Neural Information Processing Systems: Vol. 35. Curran Associates, Inc., 2022: 31809-31826.

[78] Soldaini L, Kinney R, Bhagia A, et al. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research[C]//Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2024: 15725-15788.

[79] Sanh V, Webson A, Raffel C, et al. Multitask Prompted Training Enables Zero-Shot Task Generalization[C]//International Conference on Learning Representations. 2022.

[80] Bai Y, Jones A, Ndousse K, et al. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback[A]. 2022. arXiv: 2204.05862.

[81] Wang Y, Kordi Y, Mishra S, et al. Self-Instruct: Aligning Language Models with Self-Generated Instructions[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2023.

[82] Shoeybi M, Patwary M, Puri R, et al. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism[A]. 2020. arXiv: 1909.08053.

[83] Rajbhandari S, Rasley J, Ruwase O, et al. ZeRO: Memory Optimizations Toward Training Trillion Parameter Models[C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2020: 1-16.

[84] Harlap A, Narayanan D, Phanishayee A, et al. PipeDream: Fast and Efficient Pipeline Parallel DNN Training[A]. 2018. arXiv: 1806.03377.

[85] Rasley J, Rajbhandari S, Ruwase O, et al. DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020: 3505–3506.

[86] Micikevicius P, Narang S, Alben J, et al. Mixed Precision Training[C]//International Conference on Learning Representations. 2018.

[87] Lialin V, Deshpande V, Yao X, et al. Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning[A]. 2024. arXiv: 2303.15647.

[88] Hu E J, yelong shen, Wallis P, et al. LoRA: Low-Rank Adaptation of Large Language Models [C]//International Conference on Learning Representations. 2022.

[89] Ouyang L, Wu J, Jiang X, et al. Training language models to follow instructions with human feedback[C]//Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022: 27730 - 27744.

[90] Glaese A, McAleese N, Trębacz M, et al. Improving alignment of dialogue agents via targeted human judgements[A]. 2022. arXiv: 2209.14375.

[91] Christiano P F, Leike J, Brown T B, et al. Deep reinforcement learning from human preferences[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017: 4302 - 4310.

[92] Schulman J, Wolski F, Dhariwal P, et al. Proximal Policy Optimization Algorithms[A]. 2017.

arXiv: 1707.06347.

[93] Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation[C]//Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. 2002: 311-318.

[94] Lin C Y. ROUGE: A Package for Automatic Evaluation of Summaries[C]//Text Summarization Branches Out. Barcelona, Spain: Association for Computational Linguistics, 2004: 74-81.

[95] Wang A, Singh A, Michael J, et al. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding[C]//Linzen T, Chrupała G, Alishahi A. Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Brussels, Belgium: Association for Computational Linguistics, 2018: 353-355.

[96] Wang A, Pruksachatkun Y, Nangia N, et al. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems[C]//Proceedings of the 33rd International Conference on Neural Information Processing Systems. 2019: 3266 - 3280.

[97] Hendrycks D, Burns C, Basart S, et al. Measuring Massive Multitask Language Understanding [C]//International Conference on Learning Representations. 2021.

[98] Hendrycks D, Burns C, Kadavath S, et al. Measuring Mathematical Problem Solving With the MATH Dataset[C]//Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2). 2021.

[99] Zheng L, Chiang W L, Sheng Y, et al. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena[C]//Proceedings of the 37th International Conference on Neural Information Processing Systems. 2023: 46595 - 46623.

[100] Han S, Pool J, Tran J, et al. Learning both weights and connections for efficient neural networks[C]//Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1. 2015: 1135-1143.

[101] Hinton G, Vinyals O, Dean J. Distilling the Knowledge in a Neural Network[A]. 2015. arXiv: 1503.02531.

[102] Gholami A, Kim S, Dong Z, et al. A survey of quantization methods for efficient neural network inference[M]//Low-power computer vision. 2022: 291-326.

[103] Dettmers T, Zettlemoyer L. The case for 4-bit precision: k-bit Inference Scaling Laws[C]// Proceedings of the 40th International Conference on Machine Learning. 2023: 7750-7774.

[104] Liu P, Liu Z, Gao Z F, et al. Do Emergent Abilities Exist in Quantized Large Language Models: An Empirical Study[C]//Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). 2024: 5174-5190.

[105] Meskó B. Prompt engineering as an important emerging skill for medical professionals: tutorial [J]. Journal of medical Internet research, 2023, 25: e50638.

[106] Shin T, Razeghi Y, IV R L L, et al. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts[C]//Proceedings of the 2020 Conference on Empirical

Methods in Natural Language Processing (EMNLP). 2020: 4222-4235.

[107] Zhou Y, Muresanu A I, Han Z, et al. Large Language Models are Human-Level Prompt Engineers[C]//The Eleventh International Conference on Learning Representations. 2023.

[108] Anderson P W. Localisation theory and the Cu-Mn problem: Spin glasses[J]. Materials Research Bulletin, 1970, 5(8): 549-554.

[109] Edwards S, Anderson P W. Theory of spin glasses[J]. Journal of Physics F: Metal Physics, 1975, 5(5): 965.

[110] Sherrington D, Kirkpatrick S. Solvable Model of a Spin-Glass[J]. Physical Review Letters, 1975, 35: 1792-1796.

[111] Mezard M, Parisi G, Virasoro M A. Spin Glass Theory and Beyond[M]. World Scientific, 1986.

[112] Thouless D J, Anderson P W, Palmer R G. Solution of 'Solvable model of a spin glass'[J]. The Philosophical Magazine: A Journal of Theoretical Experimental and Applied Physics, 1977, 35 (3): 593-601.

[113] Plefka T. Convergence condition of the TAP equation for the infinite-ranged Ising spin glass model[J]. Journal of Physics A: Mathematical and General, 1982, 15(6): 1971.

[114] Mezard M, Parisi G, Virasoro M A. SK Model: The Replica Solution without Replicas[J]. Europhysics Letters, 1986, 1(2): 77.

[115] Parisi G. Infinite number of order parameters for spin-glasses[J]. Physical Review Letters, 1979, 43(23): 1754.

[116] Parisi G. A sequence of approximated solutions to the S-K model for spin glasses[J]. Journal of Physics A: Mathematical and General, 1980, 13(4): L115.

[117] Parisi G. The order parameter for spin glasses: a function on the interval 0-1[J]. Journal of Physics A: Mathematical and General, 1980, 13(3): 1101.

[118] Hopfield J J. Neural networks and physical systems with emergent collective computational abilities.[J]. Proceedings of the national academy of sciences, 1982, 79(8): 2554-2558.

[119] Amit D J, Gutfreund H, Sompolinsky H. Storing Infinite Numbers of Patterns in a Spin-Glass Model of Neural Networks[J]. Phys. Rev. Lett., 1985, 55: 1530-1533.

[120] Gardner E, Derrida B. Optimal storage properties of neural network models[J]. Journal of Physics A: Mathematical and general, 1988, 21(1): 271.

[121] Sompolinsky H, Crisanti A, Sommers H J. Chaos in Random Neural Networks[J]. Phys. Rev. Lett., 1988, 61: 259-262.

[122] Seung H S, Sompolinsky H, Tishby N. Statistical mechanics of learning from examples[J]. Physical Review A, 1992, 45: 6056-6091.

[123] Yedidia J S, Freeman W T, Weiss Y. Bethe free energy, Kikuchi approximations, and belief propagation algorithms[C]. Advances in neural information processing systems, 2001.

[124] Mezard M, Parisi G, Zecchina R. Analytic and Algorithmic Solution of Random Satisfiability Problems[J]. Science, 2002, 297(5582): 812-815.

[125] Mezard M, Zecchina R. Random K-satisfiability problem: From an analytic solution to an

efficient algorithm[J]. Physical Review E, 2002, 66: 056126.

[126] Sohl-Dickstein J, Weiss E, Maheswaranathan N, et al. Deep Unsupervised Learning using Nonequilibrium Thermodynamics[C]//Bach F, Blei D. Proceedings of Machine Learning Research: Vol. 37 Proceedings of the 32nd International Conference on Machine Learning. Lille, France: PMLR, 2015: 2256-2265.

[127] Gerace F, Loureiro B, Krzakala F, et al. Generalisation error in learning with random features and the hidden manifold model[C]//III H D, Singh A. Proceedings of Machine Learning Research: Vol. 119 Proceedings of the 37th International Conference on Machine Learning. PMLR, 2020: 3452-3462.

[128] Mei S, Montanari A. The generalization error of random features regression: Precise asymptotics and the double descent curve[J]. Communications on Pure and Applied Mathematics, 2022, 75(4): 667-766.

[129] Canatar A, Bordelon B, Pehlevan C. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks[J]. Nature communications, 2021, 12(1): 2914.

[130] Helias M, Dahmen D. Statistical field theory for neural networks: Vol. 970[M]. Springer, 2020.

[131] Biroli G, Bonnaire T, De Bortoli V, et al. Dynamical regimes of diffusion models[J]. Nature Communications, 2024, 15(1): 9957.

[132] Crisanti A, Sommers H J. The spherical p-spin interaction spin glass model: the statics[J]. Zeitschrift für Physik B Condensed Matter, 1992, 87(3): 341-354.

[133] Gradenigo G, Angelini M C, Leuzzi L, et al. Solving the spherical p-spin model with the cavity method: equivalence with the replica results[J]. Journal of Statistical Mechanics: Theory and Experiment, 2020, 2020(11): 113302.

[134] Huang H. Statistical mechanics of neural networks[M]. Springer, 2021.

[135] Mezard M, Montanari A. Information, physics, and computation[M]. Oxford University Press, 2009.

[136] Zamponi F. Mean field theory of spin glasses[A]. 2014. arXiv: 1008.4844.

[137] Monasson R. Structural Glass Transition and the Entropy of the Metastable States[J]. Phys. Rev. Lett., 1995, 75: 2847-2850.

[138] Antenucci F, Krzakala F, Urbani P, et al. Approximate survey propagation for statistical inference[J]. Journal of Statistical Mechanics: Theory and Experiment, 2019, 2019(2): 023401.

[139] Mézard M. How to compute the thermodynamics of a glass using a cloned liquid[J]. Physica A: Statistical Mechanics and its Applications, 1999, 265(3): 352-369.

[140] Springer M D, Thompson W E. The Distribution of Products of Beta, Gamma and Gaussian Random Variables[J]. SIAM Journal on Applied Mathematics, 1970, 18(4): 721-737.

[141] Krause A, Hübotter J. Probabilistic Artificial Intelligence[A]. 2025. arXiv: 2502.05244.

[142] van der Maaten L, Hinton G. Visualizing Data using t-SNE[J]. Journal of Machine Learning Research, 2008, 9(86): 2579-2605.

[143] Cocco S, Monasson R. Adaptive cluster expansion for the inverse Ising problem: convergence, algorithm and tests[J]. Journal of Statistical Physics, 2012, 147: 252-314.

[144] Fontanari J F, Meir R. Mapping correlated Gaussian patterns in a perceptron[J]. Journal of Physics A: Mathematical and General, 1989, 22(16): L803.

本科期间取得的研究成果

[1] **Yuhao Li**, Ruoran Bai, and Haiping Huang. Spin glass model of in-context learning. arXiv:2408.02288, 2024