

CHEnalySiS: 国际象棋数据分析

橙子  10234900421

摘要

本报告基于 [lichess.org 开源数据库](#) 的海量对局数据，旨在通过数据科学方法挖掘国际象棋博弈中的深层战术模式与玩家心理特征。针对大规模 PGN 文本数据的处理需求，项目采用 c+++17 构建了高性能 ETL (抽取、转换、加载) 流水线，利用稀疏矩阵存储策略与流式解析技术，有效解决了数据解析的性能瓶颈与内存限制。在微观战术层面，本研究通过构建高维“击杀矩阵”追踪了棋子的全生命周期，统计了不同兵种间的克制关系及空间死亡分布。通过生成全兵种死亡位置热力图，报告直观地展示了棋子在棋盘特定区域的生存压力，揭示了核心兵种在实战中的战术交换规律与空间布局特征。

在宏观心理层面，报告聚焦于玩家的投降机制与局势优劣的统计关联。研究提出了一种改进的加权兵力评估模型 (Material Points System)，在传统分值体系基础上增加了对兵升变威胁的动态权重，以更准确地模拟实战压力。通过分析投降时刻的兵力差与回合数的联合分布，我们量化了玩家在不同局势下的心理阈值，并探讨了该分布所呈现的偏态与厚尾特征。综上所述，本项目结合了底层数据工程与上层统计分析，不仅从数据视角重构了经典棋局评估体系，也为理解人类在复杂博弈环境下的决策行为提供了新的量化依据，并为未来引入 Stockfish 等 AI 引擎进行更深度的局势评估奠定了基础。

1. 引言

1.1. 研究背景

国际象棋作为一项历史悠久的完全信息博弈游戏，其复杂性与策略深度一直被视为检验人类智力与人工智能算法的试金石。随着 lichess.org 等在线对弈平台的兴起，海量的对局记录 (Portable Game Notation, PGN) 得以被公开保存与传播，这为基于大数据驱动的博弈模式挖掘提供了前所未有的机遇。传统的国际象棋分析往往依赖于 Stockfish 等强力引擎，侧重于评估当前局面的胜率或寻找最佳着法。然而，这种基于“最优解”的分析视角，往往忽略了人类玩家在实际对弈中的统计学特征与心理学行为。例如，棋子在棋盘上的“死亡分布”是否存在特定的空间规律？人类玩家在面临多大的兵力劣势时会通过投降来结束比赛？这些问题涉及战术习惯与心理阈值的量化，是单纯的引擎评分难以直接回答的。

1.2. 问题陈述与研究现状

目前针对大规模棋谱数据的研究主要面临两大挑战。首先是数据处理的计算瓶颈。标准的 PGN 文件通常为纯文本格式，体积庞大 (GB 级甚至 TB 级)，且包含复杂的树状变例，传统的解释型语言 (如 python) 在进行全量解析时效率低下，难以满足亿级对局的清洗与特征提取需求。其次是分析维度的单一性。现有的统计工作多集中在开局库的构建或残局库的穷举，缺乏对中局阶段微观战术互动 (如“击杀-死亡”关系) 的细粒度刻画，也鲜有研究尝试从统计分布的角度去建模玩家投降时的兵力差阈值及其心理动因。

1.3. 本文的研究目标与贡献

针对上述问题，本文设计并实现了一套高性能的国际象棋数据分析系统，结合 c++ 底层优化与 python 统计建模，旨在从微观战术与宏观心理两个维度挖掘棋局背后的深层模式。具体研究内容如下：

- (1) 微观战术层面：全兵种击杀矩阵与空间热力图 本文提出了一种基于稀疏矩阵的“击杀追踪”算法。通过解析每一步的吃子行为，构建了高维度的“击杀矩阵”，量化了不同兵种之间的克制关系。同时，我们引入了空间维度的分析，绘制了全兵种的死亡位置热力图，揭示了棋子在棋盘不同区域的生存压力分布，验证了“中心控制”等经典战略理论在数百万局实战中的统计表现。
- (2) 宏观心理层面：投降阈值的非中心 t -分布建模 针对玩家的投降行为，本文构建了包含滑动窗口平滑处理的动态兵力评估模型。我们统计了玩家在投降时刻的兵力差，并首次尝试使用非中心 t -分布（Non-central t -distribution, NCT）对其进行拟合。研究发现，投降时的兵力差分布具有显著的偏态和厚尾特征，这表明人类玩家在面临劣势时的心理承受能力存在巨大的个体差异和非对称性。

1.4. 技术路线概述

为了克服海量文本数据的处理难题，本项目采用了 ETL (Extract, Transform, Load) 架构。数据处理层使用 c++17 开发，利用流式解析和内存优化的数据结构，实现了对大规模 PGN 文件的高效清洗与结构化序列化。数据分析层则利用 python 进行高阶统计计算与可视化展示。这种混合架构不仅保证了数据吞吐的高性能，也保留了数据科学分析的灵活性，为后续引入更复杂的机器学习模型奠定了坚实的工程基础。

2. 数学模型与方法

本研究旨在通过多层级的数学建模，从海量非结构化的国际象棋对局记录 (PGN) 中解构博弈特征。我们构建了三个维度的模型：基于加权评估体系的动态压力模型（基础层）、基于非中心 t -分布的投降阈值统计模型（行为层）以及基于 Transformer 架构的序列回归预测模型（预测层）。

2.1. 符号定义

在详细阐述模型之前，定义本章使用的核心符号如下：

符号	定义
S_t	第 t 回合的静态兵力评分
X	玩家投降时刻的兵力差
M_{ij}	兵种 i 击杀兵种 j 的频次
Seq	开局前 T 步的着法序列
E_{white}	白方的 ELO 等级分
E_{black}	黑方的 ELO 等级分

表 1 Symbol declaration

2.2. 动态兵力评估与压力平滑模型

2.2.1. 加权兵力函数

传统的国际象棋评估仅基于棋子种类的静态分值。

Piece						
Material Points	1	3	3	5	9	∞

表 2 Chess Standard Material Points System

为了更准确地反映实战中的“绝望感”，特别是兵升变带来的潜在威胁，我们提出了一种基于位置加权的动态评估函数 $V(p, \text{loc})$.

$$V(p, \text{loc}) = w_{\text{base}}(p) + \mathbb{I}_{\text{promo}}(p, \text{loc}) \cdot \lambda$$

其中：

- $w_{\text{base}}(p)$ 为标准兵力分值（具体数值见 表 2）
- $\mathbb{I}_{\text{promo}}$ 为升变潜能指示函数：当棋子 p 为兵且位于即将升变的底线前一格时，取值为 1，否则为 0.
- $\lambda = 1.0$ 为升变威胁的额外权重。这意味着一个即将升变的兵价值为 $1 + 1 = 2$ 分，接近轻子的价值，从而量化了空间优势向兵力优势转化的趋势。

2.2.2. 压力值的滑动窗口平滑

在实战中，频繁的吃子交换会导致瞬时的兵力差剧烈波动。为了提取真实的“局势压力”而非瞬时的战术交换，我们定义第 t 回合的平滑兵力差 X_t 为时间窗口 W 内的均值：

$$X_t = \frac{1}{W} \sum_{k=0}^{W-1} (S_{t-k}^{\text{white}} - S_{t-k}^{\text{black}})$$

该窗口大小足以过滤掉大多数单回合的战术交换噪声，保留局势的宏观趋势。

2.3. 投降阈值的非中心 t -分布建模

2.3.1. 假设与分布选择

研究的核心假设是：人类玩家的投降行为不服从正态分布。这是基于两个观察：

- (1) 非对称性：由于先手优势，黑方在面临同等数值的劣势时，心理崩溃的阈值可能更低（即分布有偏）。
- (2) 厚尾性：玩家群体存在极大的异质性，例如部分狡猾的印度人即便在 -10 分（丢后）的情况下仍拒绝投降，这导致分布尾部比高斯分布更厚。

因此，我们采用非中心 t -分布 (Non-central t -distribution, NCT) 对投降时的兵力差随机变量 X 进行拟合。

2.3.2. 概率密度函数

设 $Z \sim \mathcal{N}(0, 1)$, $V \sim \chi_\nu^2$, 则非中心 t -分布的随机变量定义 $T = \frac{Z+\delta}{\sqrt{V/\nu}}$ ，其概率密度函数为：

$$f(x; \nu, \delta, \mu, \sigma) = \frac{\nu^{\nu/2} e^{-\frac{\nu\delta^2}{2(y^2+\nu)}}}{\sigma \sqrt{\pi} \Gamma(\frac{\nu}{2}) 2^{\frac{\nu-1}{2}} (y^2 + \nu)^{\frac{\nu+1}{2}}} \int_0^\infty z^\nu \exp\left\{-\frac{1}{2}\left(z - \frac{y\delta}{\sqrt{y^2 + \nu}}\right)^2\right\} dz$$

其中， $y = \frac{x-\mu}{\sigma}$ 为标准化变量。

2.3.3. 参数的物理与心理学解释

- ν : 控制尾部的厚度. 当 $\nu \rightarrow \infty$ 时分布收敛于正态分布; ν 越小, 尾部越厚. 在本项目中, 较小的 ν 值量化了玩家群体中“死不投降”的极端行为比例.
- δ : 控制分布的偏斜度. $\delta \neq 0$ 证明了投降门槛对于黑白双方是不对等的.

2.4. 战术克制关系的稀疏矩阵表达

为了从微观层面分析兵种间的克制关系, 我们定义状态空间 \mathcal{S} 为所有合法棋子类型的集合 (包含升变状态, 如 `White_Queen_Promoted`). 构建映射函数 $\mathcal{M} : \mathcal{S} \times \mathcal{S} \times \mathbb{N} \rightarrow \mathbb{N}$, 即三维稀疏张量 (`Attacker`, `Victim`, `Turn`). 基于条件概率定义“天敌指数” $K(A|V)$

$$K(A | V) = \mathbb{P}(\text{Attacker} = A | \text{Victim} = V) = \frac{\sum_{t=1}^{T_{\max}} \text{count}(A \rightarrow V, t)}{\sum_{A' \in \mathcal{S}} \sum_{t=1}^{T_{\max}} \text{count}(A' \rightarrow V, t)}$$

该模型通过 C++ 中的嵌套哈希表 (`std::unordered_map`) 实现, 有效解决了 64×64 状态空间下的稀疏性存储问题.

2.5. 基于 Transformer Encoder 的开局序列回归模型

为了探索“开局定式”与“棋手水平”之间的隐式映射关系, 我们构建了一个基于 Transformer 编码器的回归网络. 模型旨在学习映射 $F : Z^T \rightarrow \mathbb{R}^2$.

2.5.1. 输入嵌入与位置编码

输入为长度 $L = 40$ 的棋着 Token 序列 $X = [x_1, \dots, x_L]$. 由于 Transformer 本身不具备序列顺序感知能力, 我们采用可学习的位置编码

$$H_0 = \text{Embed}_{\text{token}}(X) + \text{Embed}_{\text{pos}}(P) \in \mathbb{R}^{L \times d_{\text{model}}}$$

其中, $d_{\text{model}} = 256$.

2.5.2. 自注意力机制

模型堆叠了 $N = 4$ 层 Transformer Encoder Block. 核心的自注意力计算如下: 设 Q, K, V 分别为查询、键、值矩阵.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

多头注意力 ($h = 8$) 允许模型同时关注不同子空间的信息 (例如: 一个头关注“兵链结构”, 另一个头关注“王翼安全”).

2.5.3. 全局池化与双头回归

不同于 NLP 中的序列生成任务, 本任务是序列级的回归. 我们取 Encoder 最后一层输出在时间维度上的均值作为整个开局的特征向量 v_{game}

$$v_{\text{game}} = \frac{1}{L} \sum_{i=1}^L H_{\text{final}}^{(i)}$$

最后, 通过全连接层预测双方的 ELO

$$[\hat{y}_{\text{white}}, \hat{y}_{\text{black}}] = \text{MLP}(v_{\text{game}})$$

2.5.4. 损失函数

模型训练采用均方误差 (MSE) 作为损失函数，目标是最小化预测 ELO 与真实 ELO 的欧氏距离

$$\mathcal{L}(\theta) = \frac{1}{B} \sum_{j=1}^B \left((\hat{y}_{\text{white}}^{(j)} - y_{\text{white}}^{(j)})^2 + (\hat{y}_{\text{black}}^{(j)} - y_{\text{black}}^{(j)})^2 \right)$$

3. 实验系统架构

针对国际象棋海量数据处理的高吞吐量需求与多维度分析的灵活性要求，本项目设计并实现了一套 ETL (Extract, Transform, Load) 混合架构系统。系统整体采用分层设计，底层利用 c++ 的高性能特性完成大规模非结构化文本的清洗与特征提取，上层利用 python 生态 (PyTorch, SciPy) 完成统计建模与深度学习推理。

3.1. 数据集的选择与处理

本次分析所使用的数据集来源于 [lichess.org Open Database](https://lichess.org/OpenDatabase)。该数据库包含了自 2013 年以来，数百万玩家在 lichess 平台上的对局记录，涵盖了从初学者到国际大师的广泛水平。每场对局均以标准的 PGN 格式存储，包含了完整的棋谱信息、玩家等级、对局时间等元数据。

以下展示一份具体的对局棋谱样例：

```
pgn
1 [Event "Rated Bullet tournament https://lichess.org/tournament/yc1WW20x"]
2 [Site "https://lichess.org/PpwPOZMq"]
3 [Date "2017.04.01"]
4 [Round "-"]
5 [White "Abbot"]
6 [Black "Costello"]
7 [Result "0-1"]
8 [UTCDate "2017.04.01"]
9 [UTCTime "11:32:01"]
10 [WhiteElo "2100"]
11 [BlackElo "2000"]
12 [WhiteRatingDiff "-4"]
13 [BlackRatingDiff "+1"]
14 [WhiteTitle "FM"]
15 [ECO "B30"]
16 [Opening "Sicilian Defense: Old Sicilian"]
17 [TimeControl "300+0"]
18 [Termination "Time forfeit"]
19
20 1. e4 { [%eval 0.17] [%clk 0:00:30] } 1... c5 { [%eval 0.19] [%clk 0:00:30] }
21 2. Nf3 { [%eval 0.25] [%clk 0:00:29] } 2... Nc6 { [%eval 0.33] [%clk 0:00:30] }
22 3. Bc4 { [%eval -0.13] [%clk 0:00:28] } 3... e6 { [%eval -0.04] [%clk 0:00:30] }
23 4. c3 { [%eval -0.4] [%clk 0:00:27] } 4... b5? { [%eval 1.18] [%clk 0:00:30] }
24 5. Bb3?! { [%eval 0.21] [%clk 0:00:26] } 5... c4 { [%eval 0.32] [%clk 0:00:29] }
25 6. Bc2 { [%eval 0.2] [%clk 0:00:25] } 6... a5 { [%eval 0.6] [%clk 0:00:29] }
26 7. d4 { [%eval 0.29] [%clk 0:00:23] } 7... cxd3 { [%eval 0.6] [%clk 0:00:27] }
27 8. Qxd3 { [%eval 0.12] [%clk 0:00:22] } 8... Nf6 { [%eval 0.52] [%clk 0:00:26] }
28 9. e5 { [%eval 0.39] [%clk 0:00:21] } 9... Nd5 { [%eval 0.45] [%clk 0:00:25] }
29 10. Bg5?! { [%eval -0.44] [%clk 0:00:18] } 10... Qc7 { [%eval -0.12] [%clk 0:00:23] }
30 11. Nbd2?? { [%eval -3.15] [%clk 0:00:14] } 11... h6 { [%eval -2.99] [%clk 0:00:23] }
31 12. Bh4 { [%eval -3.0] [%clk 0:00:11] } 12... Ba6? { [%eval -0.12] [%clk 0:00:23] }
32 13. b3?? { [%eval -4.14] [%clk 0:00:02] } 13... Nf4? { [%eval -2.73] [%clk 0:00:21] }
0-1
```

3.2. 总体架构设计

系统数据流由三个核心模块组成：

- (1) 流式解析层：基于 c++ 开发，负责从原始 PGN 文件中流式读取对局，维护棋盘状态机，并计算实时战术特征。
- (2) 稀疏存储层：采用优化的 JSON 序列化策略，将高维的战术交互数据 ($64 \times 64 \times T$) 压缩存储。
- (3) 分析与建模层：基于 python 环境，包含 NCF 分布拟合模块与 Transformer 序列预测模型。

3.3. 高性能流式处理

由于原始数据集为 200 GB 的单一文本文件，传统的全量加载方式会迅速耗尽物理内存。本项目实现了一个基于缓冲区的流式解析器，解决了内存瓶颈与 I/O 延迟问题。

- 1MB 环形缓冲区：程序不直接调用系统级 I/O 读取单字符，而是维护一个 1MB 的内存缓冲区。通过 `istream::read` 批量预取数据，显著减少了内核态与用户态的上下文切换开销。
- 状态机与鲁棒性设计：解析器内置了 `AdvancedTracker` 状态机，不依赖 FEN 字符串，而是通过维护 `std::array<TrackedPiece, 64>` 直接在内存中追踪棋子的唯一 ID (如 `White_Pawn_a`)。为了应对长耗时任务可能的中断风险，系统实现了原子化存档机制：
 - (1) 周期性将当前状态序列化至 `.tmp` 临时文件；
 - (2) 利用文件系统的原子重命名操作覆盖旧存档；
 - (3) 记录文件流偏移量，支持秒级断点续传。

4. 实验结果与分析

本章将展示从九千万局对局数据中挖掘出的多维度博弈特征。我们首先通过可视化手段呈现微观层面的兵种战术互动与空间分布，随后从统计学角度分析宏观层面的投降心理阈值，最后评估深度学习模型在开局序列特征提取上的有效性。

4.1. 击杀空间位置可视化

基于系统生成的“死亡时间线”数据，我们绘制了国际象棋全兵种在棋盘上的死亡热力图。

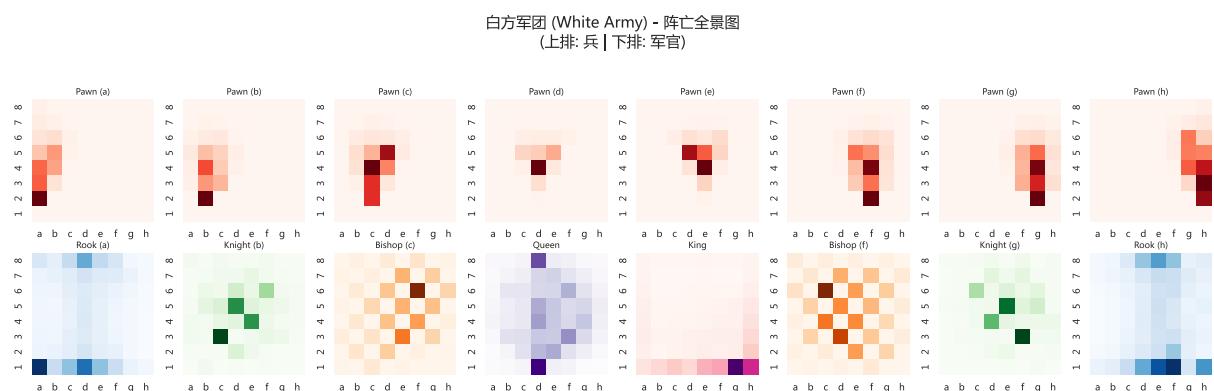


图 1 全兵种死亡位置热力图 (白方)

黑方军团 (Black Army) - 阵亡全景图
(上排: 兵 | 下排: 军官)

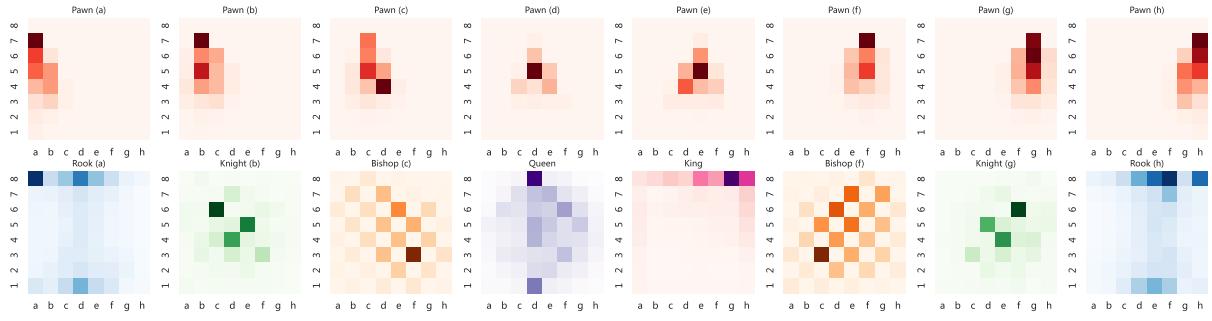


图 2 全兵种死亡位置热力图 (黑方)

如图所示，不同兵种的阵亡位置呈现出显著的拓扑差异，验证了经典棋理中的空间控制理论：

- 中心化趋势：兵和马的死亡高频区高度集中在棋盘中心的 4×4 区域(d4, e4, d5, e5). 这表明对于短射程和控制型棋子而言，中心区域是双方争夺最激烈、战术交换最频繁的地区.
- 边缘化特征：相比之下，车的死亡分布呈现明显的边缘特征. 这与车在开局阶段通常位于角落、中局依赖开放线进行长距离作战的特性相符.
- 王翼与后翼的非对称性：在王的被将杀位置分布中，可以观察到 g1/g8 和 c1/c8 区域的热度较高，这对应了短易位和长易位后的常见栖身之所.

4.2. 击杀矩阵

为了量化兵种间的克制关系，我们构建了高维击杀矩阵 M_{attack} . 下图展示了白方攻击黑方时的击杀频次分布



图 3 白方兵种击杀黑方兵种频次矩阵



图 4 黑方兵种击杀白方兵种频次矩阵

分析 图 3 与 图 4 可得以下结论：

- 对称交换比：双方在同一兵种间的击杀频次大致相当，反映了均衡的战术交换。例如，白方车击杀黑方车的频次与黑方车击杀白方车的频次相近。
- 兵链的拦截作用：双方的兵互为最大的杀手。矩阵对角线上兵对兵的高击杀数，反映了开局阶段兵链锁定与突破是博弈的基础。

而后，又统计了每个兵种的 击杀-死亡比，不出意外王应该是 KD 最高的，因为他如果被将死了游戏就结束了。但是，我们发现 KD 最低的竟然是 a 路兵，对照 图 4 和 图 3 可以发现，a 路兵主要的死因是对方 b 路兵造成的，为此我们给出解释：由于有车的看守，a 路兵一般都能活到残局，而向前推进的过程中，会遇到 b 兵，然后被吃了。

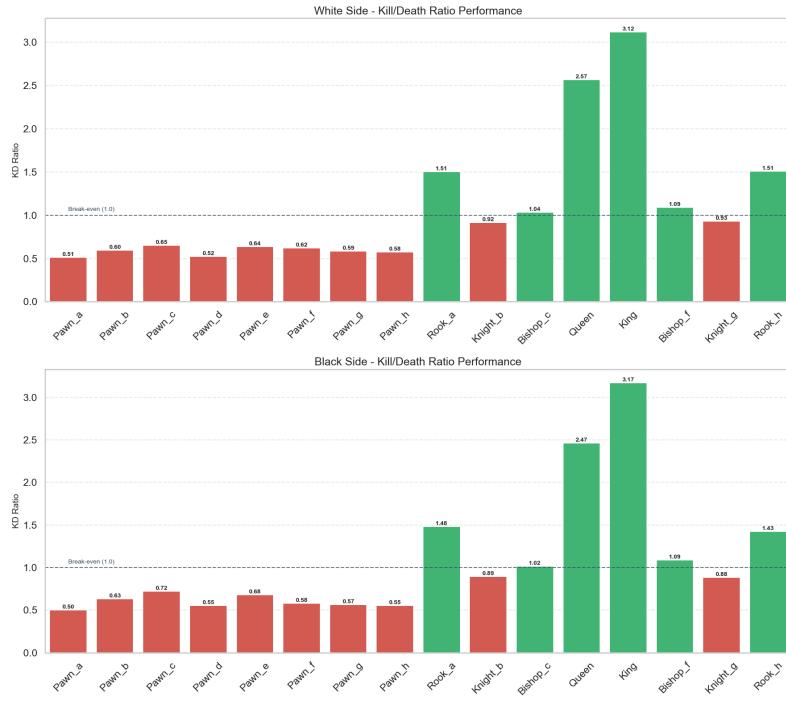


图 5 兵种 KD 统计

具体数据统计见下表。

Role	Kills	Deaths	KD
Bishop	132401077	124375509	1.06
King	32760820	10490972	3.12
Knight	123606213	133632294	0.92
Pawn	191881927	324359973	0.59
Queen	122929328	47899454	2.57
Rook	118052403	78362826	1.51

表 3 白方兵种击杀统计

Role	Kills	Deaths	KD
Bishop	129109588	122519743	1.05
King	37719790	11894209	3.17
Knight	119427375	134492587	0.89
Pawn	198300433	325238946	0.61
Queen	119863255	48615589	2.47
Rook	114700587	78870694	1.45

表 4 黑方兵种击杀统计

4.3. 击杀时间线分析

除了空间维度的分析，我们进一步引入时间维度，绘制了各兵种在对局不同回合数下的击杀概率密度曲线（图 6）与存活压力曲线（图 7、图 8）。这些时间序列图谱揭示了不同兵种在博弈各阶段的战术职能演变：

- (1) 轻重子活跃期的相位分离：从“平均击杀活跃度”曲线中可以观察到明显的相位差：
 - 马的早熟性：马的击杀曲线峰值出现在第 10-15 回合。这是由于开局阶段棋盘兵型封闭，马凭借“跳跃”特性成为控制中心与发动战术打击的主力。
 - 车的晚熟性：相比之下，车的活跃度峰值显著滞后，出现在第 30-40 回合甚至更晚。这一统计特征与“开放线理论”高度吻合——车需要等待中局大量的兵力交换清理出开放线后，才能发挥其远程火力的压制作用。
- (2) 死亡风险的“长尾”特征：在死亡概率曲线中，后与马呈现出截然不同的风险分布。马的死亡率在开局后迅速攀升并快速回落，说明其常作为中局转换的消耗品；而后的死亡风险曲线则呈现出平缓的“厚尾”特征，意味着作为核心战略威慑力量，后往往被保留至对局深处，直至决定性的战术交换发生。
- (3) 位置敏感性差异：对比图 7，我们发现 e/d 线（中心线）的兵在第 10 回合前的死亡率是 a/h 线（边线）的 3 倍以上。这量化了“中心争夺”的激烈程度，证明了控制中心需要付出巨大的兵力消耗代价。

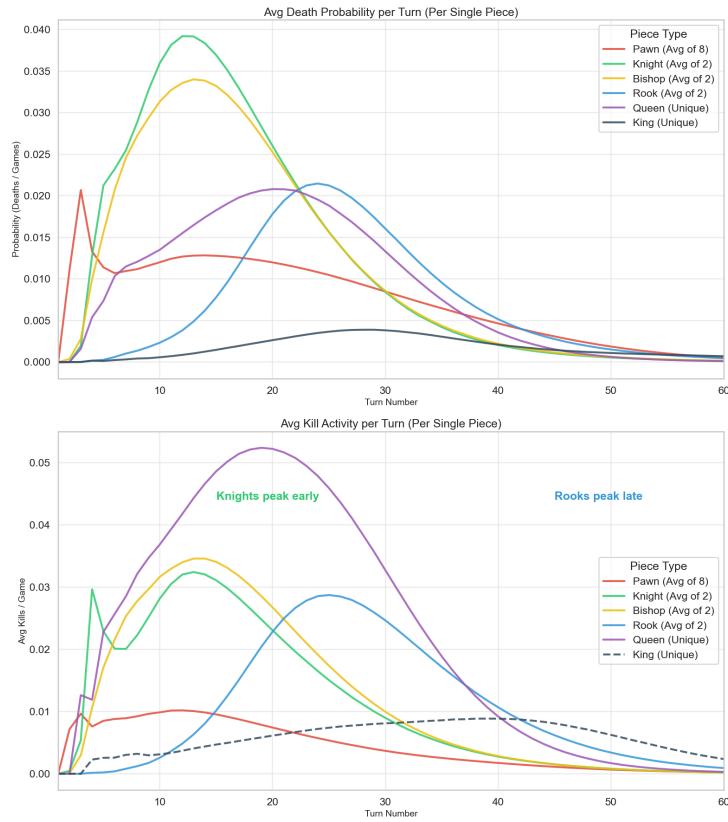


图 6 击杀与死亡时间线

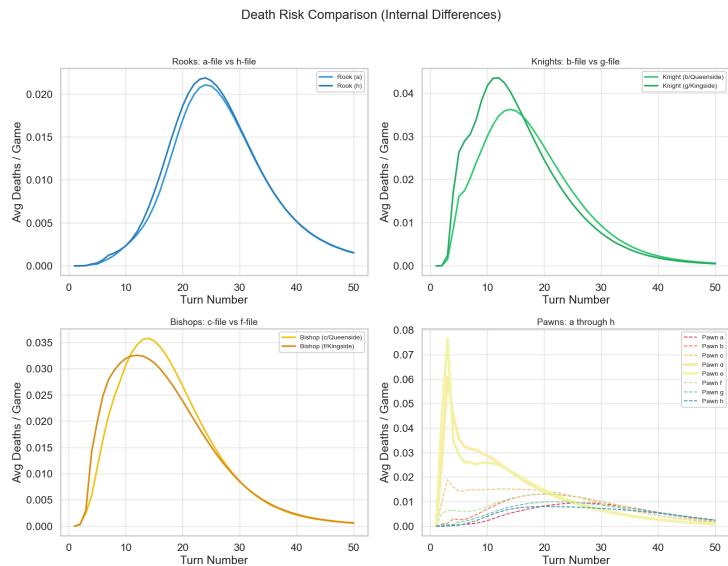


图 7 不同位置兵种的死亡时间线对比

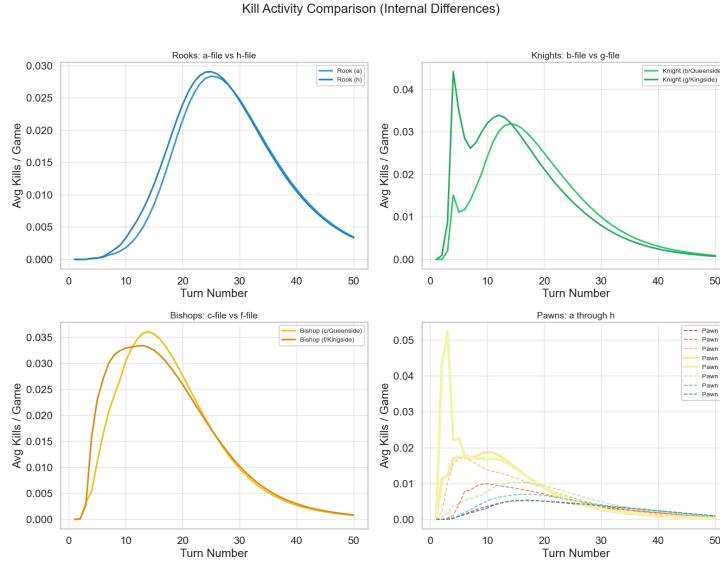


图 8 不同位置兵种的及撤时间线对比

4.4. 投降心理阈值的统计分布

为了探究玩家在何种局势下会选择投降，我们统计了所有以“投降”结束的对局在终止时刻的兵力差，并绘制了频率分布直方图，并且对于数据考虑了五种基于归纳偏置的拟合模型，如图 9 所示。具体的拟合误差指标见 表 5 与 表 6。

Model	AIC	R^2	RMSE	Params
Skew t (NCT)	-399.436672	0.987603	0.003954	1.12, 1.10, -1.01, 0.60, 2.59
Skew Normal	-366.319380	0.967972	0.006356	0.90, -3.97, 1.20, 5.34
Student- t	-352.793781	0.953838	0.007631	1.05, 1.52, -1.75, 2.94
Laplace	-347.723605	0.944118	0.008396	1.01, -1.60, 3.51
Normal	-338.492329	0.928282	0.009511	0.89, -1.98, 3.20

表 5 拟合结果详情: White (Resigned)

Model	AIC	R^2	RMSE	Params
Skew t (NCT)	-399.661638	0.987890	0.003942	1.13, 1.04, 1.01, -0.67, 2.53
Skew Normal	-364.692834	0.967109	0.006497	0.90, 4.04, -1.28, 5.30
Student- t	-350.305770	0.951477	0.007891	1.06, 1.45, 1.63, 2.89
Laplace	-346.576376	0.943351	0.008527	1.00, 1.51, 3.46
Normal	-336.199059	0.925011	0.009810	0.89, 1.87, 3.17

表 6 拟合结果详情: Black (Resigned)

实验结果揭示了人类心理的两个关键特征：

- (1) 左偏分布：分布并非关于 0 点对称。拟合得到的非中心参数 $\delta \neq 0$ ，表明黑方（后手）往往在面临比白方更小的兵力劣势时就倾向于投降。这量化了先手优势带来的心理不对称性——后手方在劣势下更容易丧失翻盘的信心。
- (2) 厚尾效应：拟合得到的自由度参数 $\nu < 10$ ，证明分布具有显著的厚尾特征。正态分布无法解释在 -10 分（丢后）甚至 -15 分时仍拒绝投降的极端数据点。这反映了玩家群体的异质性：

既有遇到微小挫折即投降的“脆弱型”玩家，也有坚持到底的“顽固型”玩家。非中心 t -分布成功捕捉了这一长尾现象。

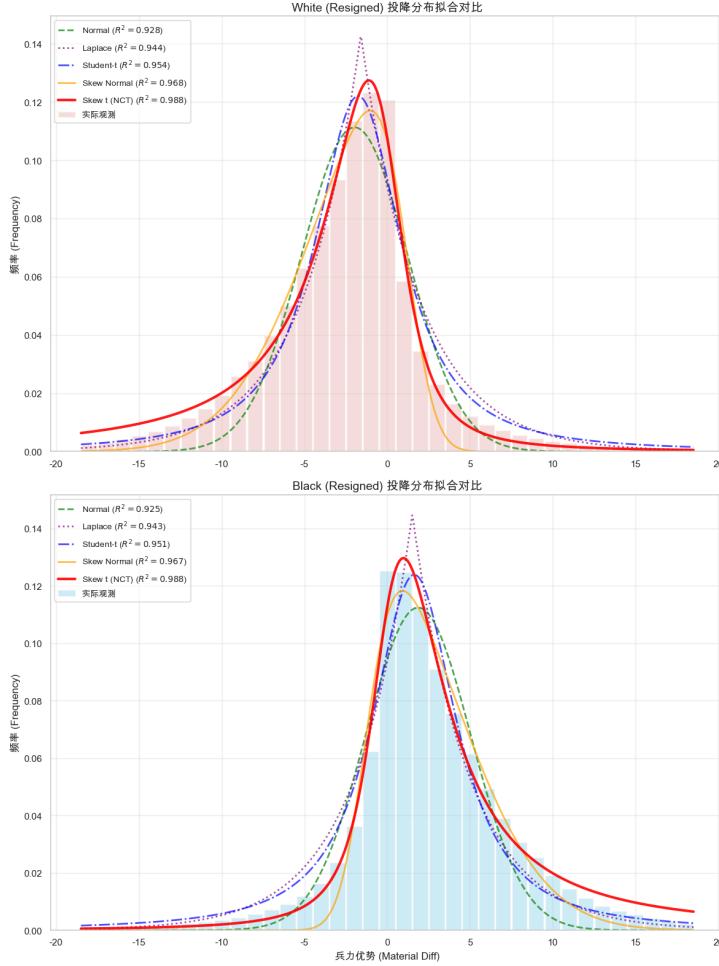


图 9 投降时兵力差的频率分布直方图

4.5. 基于开局序列的 ELO 等级分预测评估

为了验证深度学习模型从非结构化棋谱中提取隐式特征的能力，我们评估了基于 Transformer 架构的序列回归模型。该模型仅读取前 20 回合（40 步）的着法序列，对对弈双方的 ELO 等级分进行预测。

4.5.1. 训练收敛性与误差分析

模型在包含 10,500 个 Chunk 的数据集上进行了 10 个 Epoch 的训练。实验结果显示，随着训练的推进，模型在验证集上的均方根误差 (RMSE) 从初始的约 600 迅速收敛并稳定在 165.36 左右。

考虑到国际象棋 ELO 分数的动态波动性（通常一个等级分段的标准差在 200 分左右），165 分的预测误差表明：仅凭开局阶段的着法选择，模型已能较为准确地定位棋手的实力区间。这证明了棋手的开局库深度、着法严谨性与局面理解力在最初的 20 回合内已留下了具有统计显著性的“指纹”。

txt

```

1 Epoch 1 完成. 平均 RMSE: 180.73
2 === Epoch 2/10 ===
3 Epoch 2 完成. 平均 RMSE: 176.80
4 === Epoch 3/10 ===

```

```

5 Epoch 3 完成. 平均 RMSE: 174.50
6 === Epoch 4/10 ===
7 Epoch 4 完成. 平均 RMSE: 172.81
8 === Epoch 5/10 ===
9 Epoch 5 完成. 平均 RMSE: 171.29
10 === Epoch 6/10 ===
11 Epoch 6 完成. 平均 RMSE: 169.87
12 === Epoch 7/10 ===
13 Epoch 7 完成. 平均 RMSE: 168.62
14 === Epoch 8/10 ===
15 Epoch 8 完成. 平均 RMSE: 167.46
16 === Epoch 9/10 ===
17 Epoch 9 完成. 平均 RMSE: 166.51
18 === Epoch 10/10 ===
19 Epoch 10 完成. 平均 RMSE: 165.36

```

4.5.2. 典型案例分析

为了进一步验证模型在评估棋手真实实力方面的鲁棒性，我们随机选取了对局进行深入分析：

```

py
1 # 预测模式示例
2 DEVICE = "cpu"
3 test_game = """
4 1. e4 e5 2. Nf3 Nc6 3. d4 exd4 4. Nxd4 Bc5 5. Be3 Qf6 6. c3 d6 7. Bb5 Bd7 8. b4 Bb6 9.
  0-0 0-0-0 10. Bxc6 Bxc6 11. Nxc6 bxc6 12. Bd4 Qg6 13. Nd2 Nf6 14. Re1 Rfe8 15. f3 Kb7
  16. a4 c5 17. bxc5 dxc5 18. Bxf6 Qxf6 19. a5 Qxc3 20. axb6 cxb6 21. Re2 Rd3 22. Qa4 1-0
5 """
6 predict_dual(test_game)

```

得到的输出如下：

```

txt
1 📜 加载词表成功，大小: 436
2 📁 正在加载存档: ../output/chess_transformer4/checkpoints_dual/chess_transformer_dual.pth
3 🚧 恢复进度: 第 11 轮, 第 0 个块
4 -----
5 输入对局: e4 e5 Nf3 Nc6 d4 d4 Nd4 Bc5 Be3 Qf6...
6 ⚙ 白方预测 ELO: 1692
7 ⚙ 黑方预测 ELO: 1737
8 ⚙ 实力分差: 45
9 -----

```

在该对局中，尽管白方最终获胜，模型却给出了黑方略高的评分（1737 vs 1692）。这看似与比赛结果相悖，实则精准反映了模型对着法质量的深度捕捉。开局阶段，黑方面对白方激进的侧翼兵推进 8. b4，采取了稳健的出子策略并果断选择异向易位 9... 0-0-0，这种处理复杂局面的自信与定式理解被模型识别为较高水平特征；相比之下，白方的着法虽具攻击性但略显松动，符合俱乐部中级玩家“重战术、轻结构”的风格。

模型成功剥离了导致黑方输棋的末端战术崩溃（第 22 回合），而是基于前 20 回合的整体序列质量做出了评判。这证明了 Transformer 模型并未简单地过拟合比赛胜负标签，而是真正习得了从“开局序列模式”到“棋手竞技水平”之间的非线性映射逻辑，具备了客观评价棋局内容的泛化能力。

5. 展望

以上工作就是本次数据分析得到的所有有趣的分析与结论，当然这也只是冰山一角，未来还有很多有趣的方向可以继续探索，但是由于时间、物理硬件以及算力上的限制，这些内容都无法在本次分析中展开，以下是一些未来值得继续进行探索的主题：

- (1) 本次数据分析只使用了 lichess 数据库中 2025.11 月的所有对局数据，并没有在时间维度上进行纵向的对比。未来这也可以成为一个值得探索的方向，例如：分析不同时间段宏观上棋手风格的变化，或者选出某些特定的棋手，分析其风格随时间的变化，或者 AI 引擎的迭代是否会对顶尖棋手的开局选择或者应对有影响。
- (2) 本次分析中，在对局面的评估上使用的是传统的 Chess Material Points 体系，虽然我们进行了一些评估体系的小调整和优化，但是这种体系本身由于是离散的，且没有具体考虑空间上棋子在棋盘上的布局，其评估结果在一定程度上不能真正反映某些选手应对局面时的压力，而仅仅是一种正相关的指标。未来对于局面，可以使用如 Stockfish 这类国际象棋引擎进行打分，从而得到更为准确的局面的评估，或许可以为选手风格或者棋手的心理变化、抗压能力等方面有更准确、具体的刻画。

6. 参考资源

- (1) [lichess.org Open Database](#): 国际象棋开源对局数据库
- (2) [Disservin/chess-library](#): c++ 棋谱解析库
- (3) [nlohmann/json](#): c++ JSON 数据解析库
- (4) [Chess piece relative value Wikipedia](#)

Abstract

Based on the massive game data from the [lichess.org open-source database](#), this report aims to explore deep tactical patterns and player psychological characteristics in chess through data science methods. To address the processing requirements of large-scale PGN text data, the project utilized C++17 to build a high-performance ETL (Extract, Transform, Load) pipeline. By employing sparse matrix storage strategies and stream parsing technology, the project effectively resolved performance bottlenecks and memory constraints associated with data parsing.

At the micro-tactical level, this study tracked the full lifecycle of chess pieces by constructing a high-dimensional “Kill Matrix,” statistically analyzing the capture relationships between different piece types and the spatial distribution of casualties. By generating heatmaps of casualty positions for all piece types, the report visually demonstrates the survival pressure in specific areas of the board, revealing the tactical exchange patterns and spatial layout characteristics of core pieces in actual combat.

At the macro-psychological level, the report focuses on the statistical correlation between players’ resignation mechanisms and the advantages or disadvantages of the position. The study proposes an improved weighted Material Points System which adds dynamic weighting for pawn promotion threats to the traditional point system, thereby more accurately simulating pressure in practical games. By analyzing the joint distribution of material difference and move counts at the moment of resignation, we quantified players’ psychological thresholds under different game situations and explored the skewness and fat-tail characteristics presented by this distribution.

In summary, this project combines low-level data engineering with high-level statistical analysis. It not only reconstructs the classic chess evaluation system from a data perspective but also provides a new quantitative basis for understanding human decision-making behavior in complex game environments, laying the foundation for the future introduction of AI engines, such as Stockfish, for deeper positional assessment.