实践报告：GitHub 用户数据洞察分析

1. 实验背景

本次实验的目标是通过对 GitHub 上 500 名具有协作行为日志数据的用户进行分析，洞察用户的分布特征和协作行为模式。实验内容包括人口统计分析（国家和地区分布、城市级别分布、时区分布）和协作行为分析（提交频率、协作时间模式、事件类型分布等）。通过本次实验，我们旨在提升数据处理与分析能力，掌握 GPT 工具的应用，并理解数据隐私与伦理的重要性。

---

2. 数据来源与预处理

数据来源于 GitHub 上的用户协作行为日志数据，包含以下字段：

- `user_id`：用户 ID
- `name`：用户姓名
- `location`：用户地理位置
- `country`：用户所在国家
- `event_type`：事件类型（如提交、评论等）
- `event_action`：事件动作，这表示在事件类型下执行的具体动作。
- `event_time`：事件发生时间
- `hour_of_day`：一天中的小时，这是事件发生时的具体时间（小时）
- `timezone_offset`：用户时区偏移，这表示事件发生的时间相对于UTC（协调世界时）的偏移量
- `local_time`：本地时间，这是事件发生时的本地时间，包括日期、时间和时区信息。
- `local_hour`：本地小时，这是事件发生时的本地时间的小时部分。
- `total_influence`：用户总影响力

数据预处理包括：

- 数据加载与清洗：处理缺失值和异常值。
- 时区转换：将 `event_time` 转换为用户的本地时间。
- 数据分组与统计：按国家、城市、时区等维度进行分组统计。

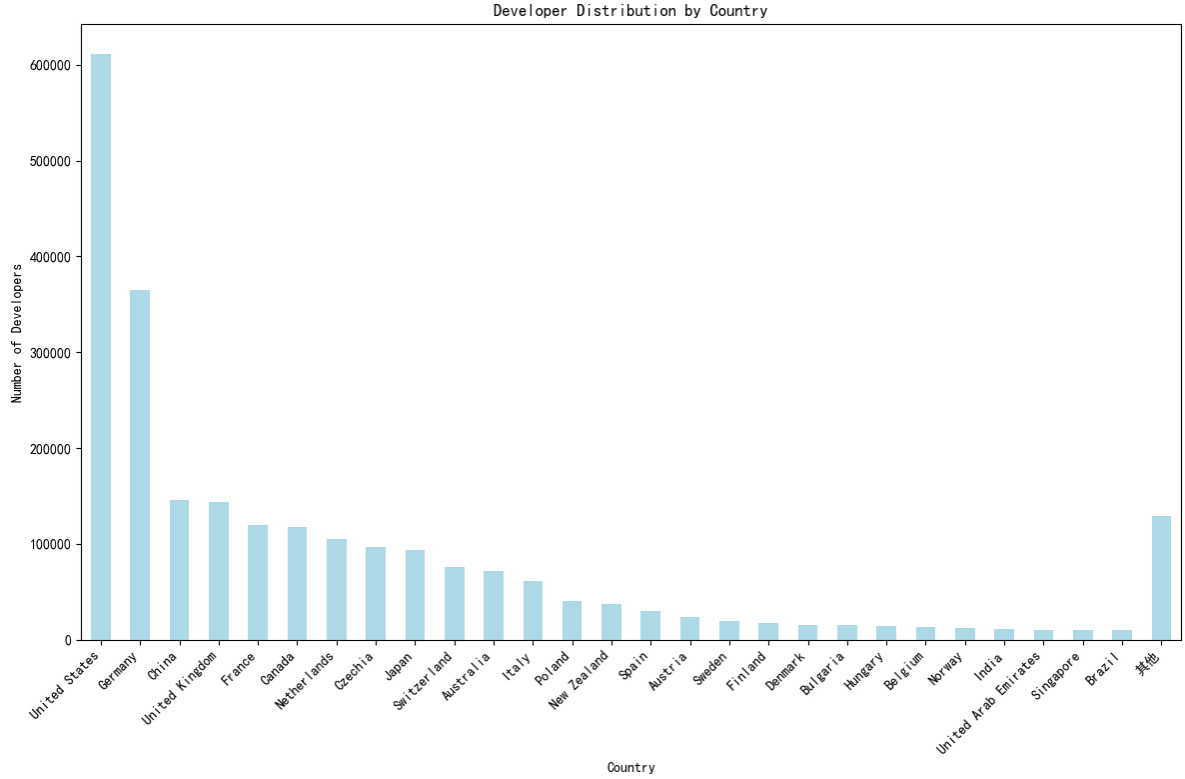
---

3. 人口统计分析

(1) 国家和地区分布

使用`value_counts()`方法统计`country`列中每个国家/地区出现的次数，结果存储在`country_counts`变量中。设定一个阈值（该实践用的是10000），将开发者数量小于这个值的国家/地区归为“其他”类别，确保可视化效果清晰。使用`plot`方法绘制主要国家/地区的开发者数量柱状图，如果数据量级差异太大（最大值与最小值之比大于100），则对y轴使用对数刻度。

通过统计用户所在国家和地区的分布，我们发现主要开发者集中在美国、中国、德国等国家。

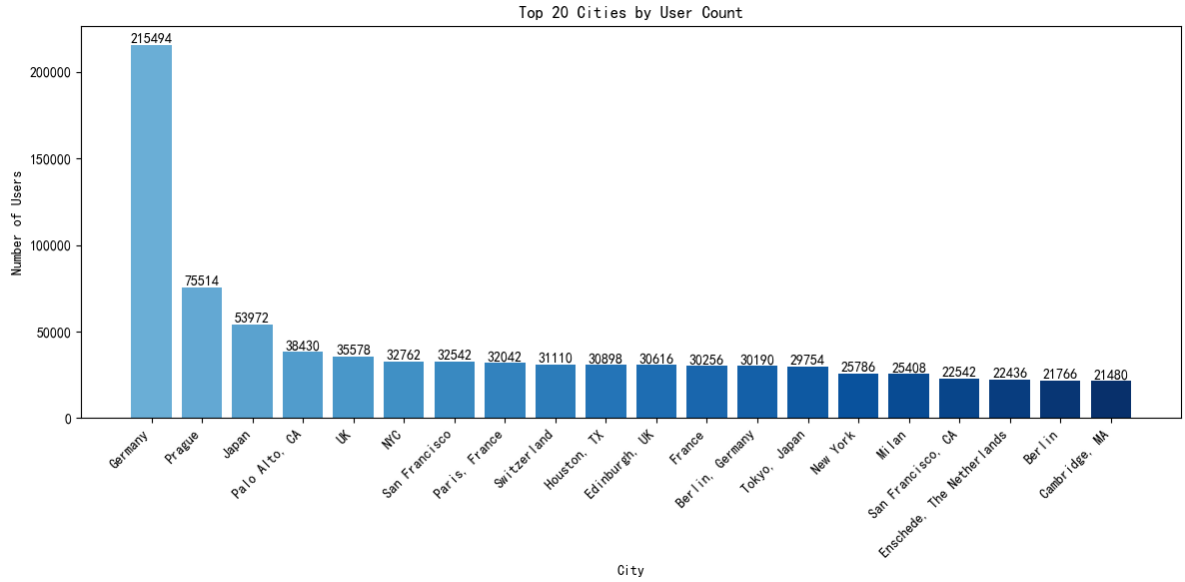


结论:

- 开发者主要集中在科技发达的国家和地区。
- 美国是全球开发者的主要集中地，其次是亚洲和欧洲国家。

(2) 城市级别分布

后统计并展示了用户数量最多的前20个城市的分布情况。通过使用matplotlib库，创建一个柱状图，展示这些城市的用户数量，城市名称被旋转45度并右对齐显示，以便更好地在图表中展示。



结论:

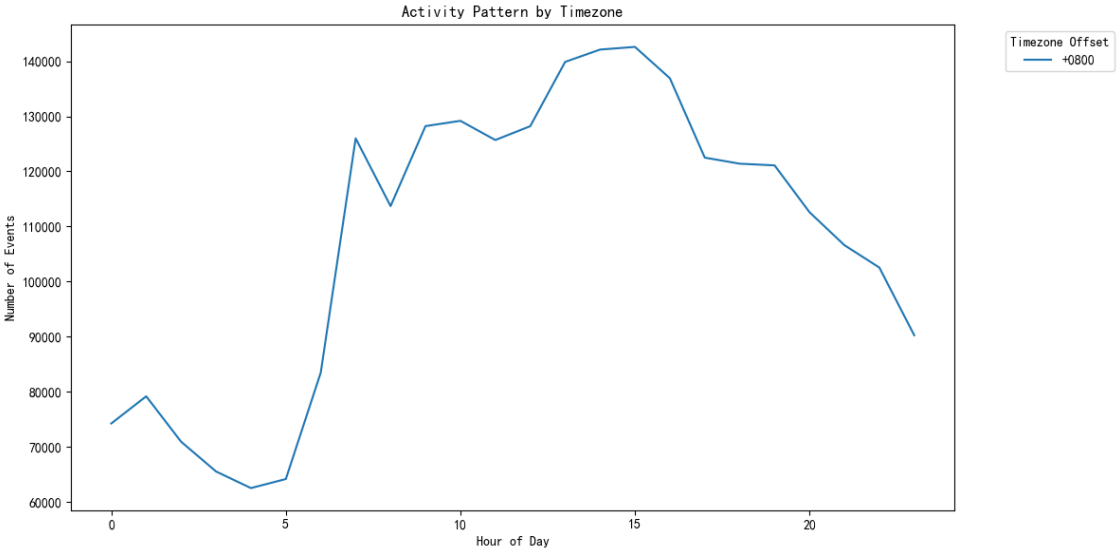
- Germany用户数量最多，其次是Prague和Japan
- 开发者主要集中在经济发达或科技产业集中的城市
- 少数城市集中了大量开发者，而其他城市的开发者数量相对较少，呈现出明显的长尾分布特征

(3) 时区分布

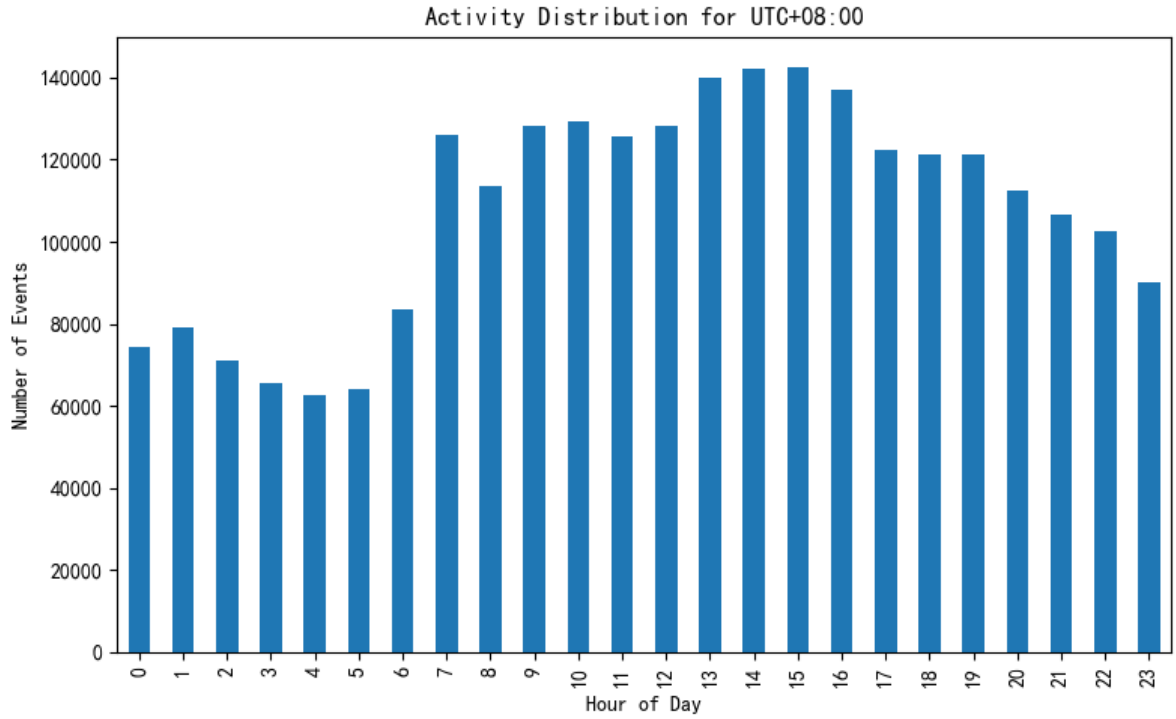
1. **时区转换与活动时间分析**：
   - `data['event_time']`：将`event_time`列转换为`datetime`格式。
   - `data['timezone_offset']`：提取时区偏移量（如`+0800`）。
   - `data['local_hour']`：提取事件发生的小时部分。
2. **时区分布统计**：
   - `timezone_counts`：统计每个时区的用户数量，并使用柱状图展示时区分布。
3. **活动时间模式分析**：
   - `timezone_hourly_activity`：按时区和小时统计事件的数量，并使用折线图展示不同时区的活动时间模式。
   - 可以进一步分析每个时区（如`+0800`）的活动分布，并使用条形图展示每小时的事件数量。

输出的结果：

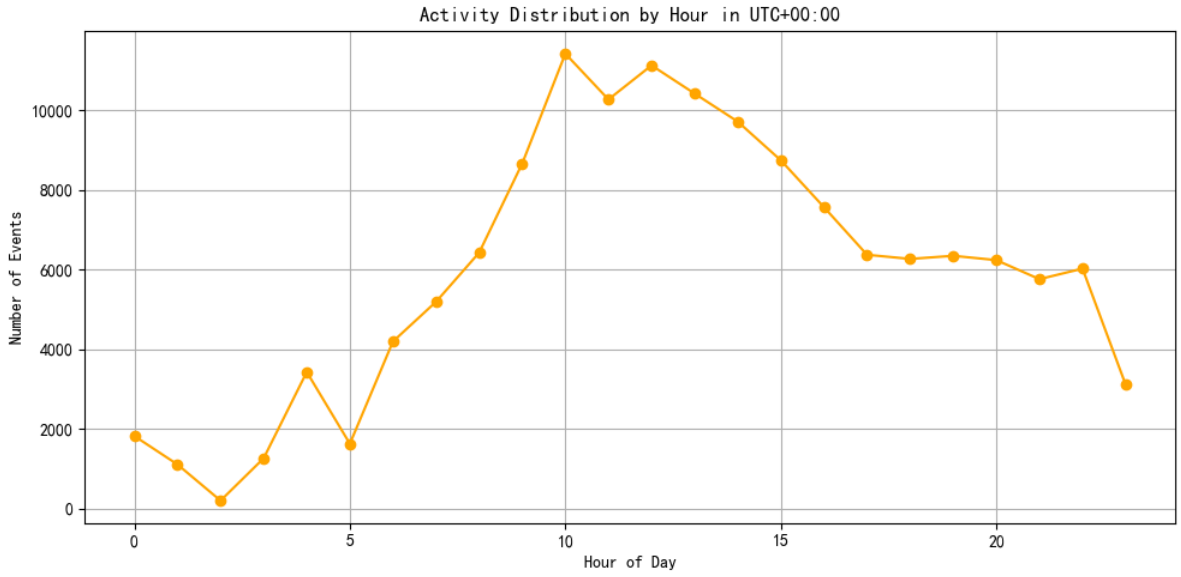
- **时区分布图**：展示各个时区的用户数量，帮助了解开发者的地理分布。但发现所有数据均为+0800.
- **活动模式图**：分析不同时区用户的活跃时间段，显示他们的协作时间模式。

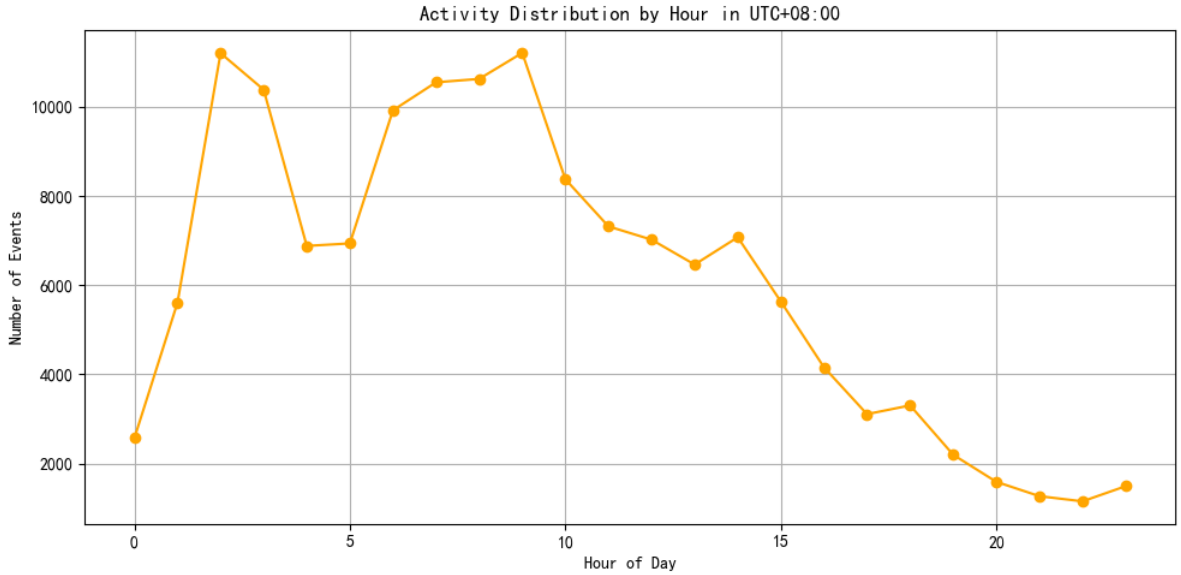


- **特定时区活动图**：分析某个时区（例如`+0800`）的活动情况，得到该时区每小时的活动数量。发现13-16时间段活动数量最多，0-6时间段活动数量最少。



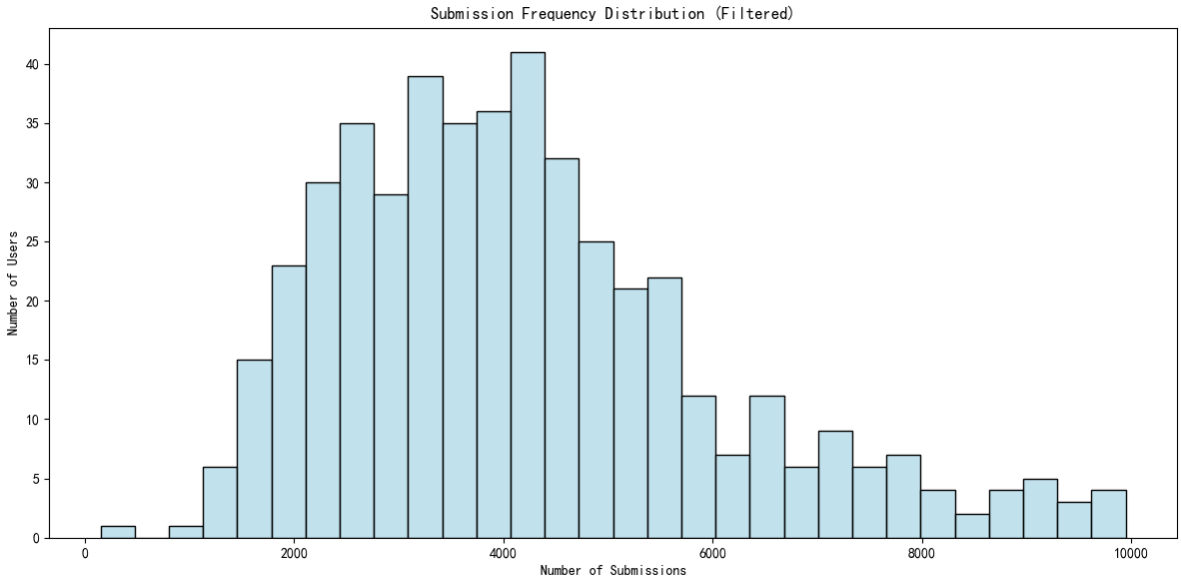
创建`country_timezone_dict`字典，将国家（`country`）映射到对应的时区（`timezone`）。它的作用是帮助在数据集中为每个用户标记时区。接着，将`event_time`列的数据类型转换为字符串类型，筛选出长度大于10的`event_time`值。将`event_time`列中的字符串转换为`datetime`类型，以便日期和时间值可以进行更方便的时间运算，比如提取小时、分钟等。通过`map()`方法，将`country`列中的国家名称映射到`country_timezone_dict`字典中的时区信息，并将结果添加到`timezone`列。循环遍历每个时区， 对于每个时区，筛选出当前时区的数据，统计每个小时的事件数量，最后绘制了每个时区的事件分布图。





4. 协作行为分析

首先使用pandas库对数据集`data`按`user_id`进行分组，并计算每个用户的`event_action`事件数量（即用户活动次数），然后对结果进行降序排序。接着，它将无穷大值替换为pandas的NA值，并删除这些NA值。之后，使用matplotlib和seaborn库绘制了一个直方图，该直方图展示了用户活动次数小于95%分位数的用户分布（即过滤掉了一部分极端活跃的用户），直方图的x轴表示提交次数（即用户活动次数），y轴表示用户数量，颜色为浅蓝色，并且没有绘制核密度估计（KDE）曲线。最后，设置了图表的标题、x轴和y轴的标签，并通过`plt.tight_layout()`调整了布局以避免标签重叠，最后展示这个图表。



结论:

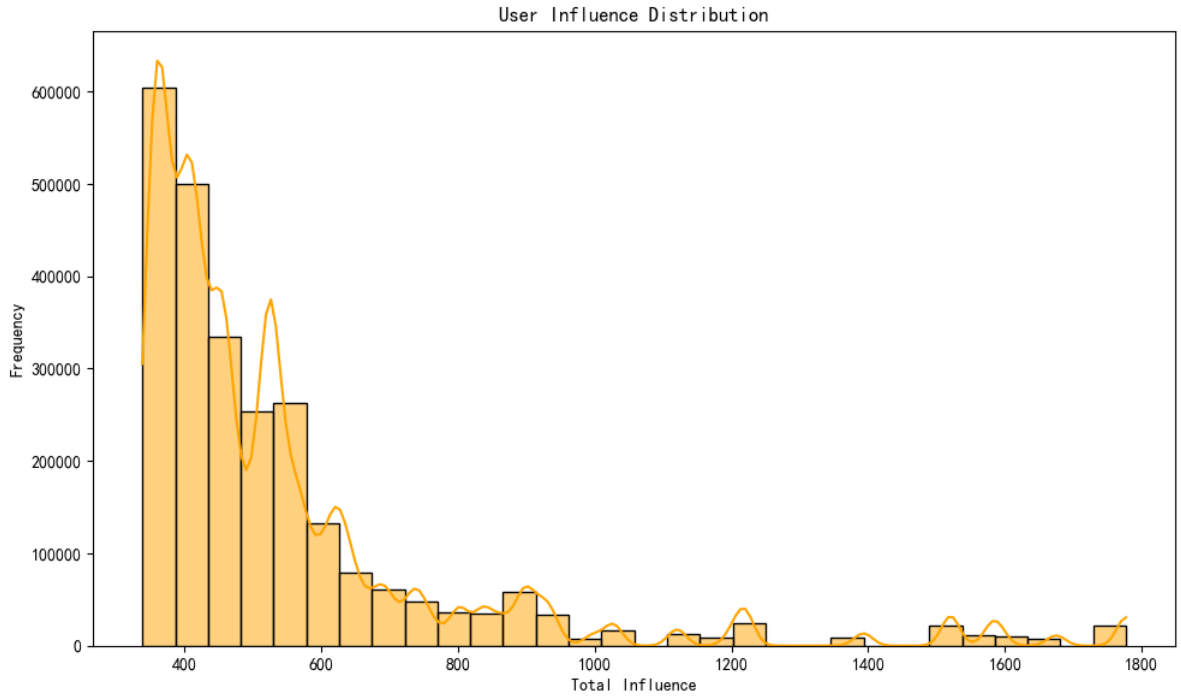
- 开发者群体中存在明显的活跃度差异。
- 高活跃用户可能是核心贡献者，值得重点关注。

5. 其他数据洞察

(1) 用户影响力分布

通过分析用户的总影响力（即`total_influence`字段），我们发现：

- 大多数用户的影响力较低，集中在 0-600 之间。
- 少数用户的影响力较高，超过 1000。



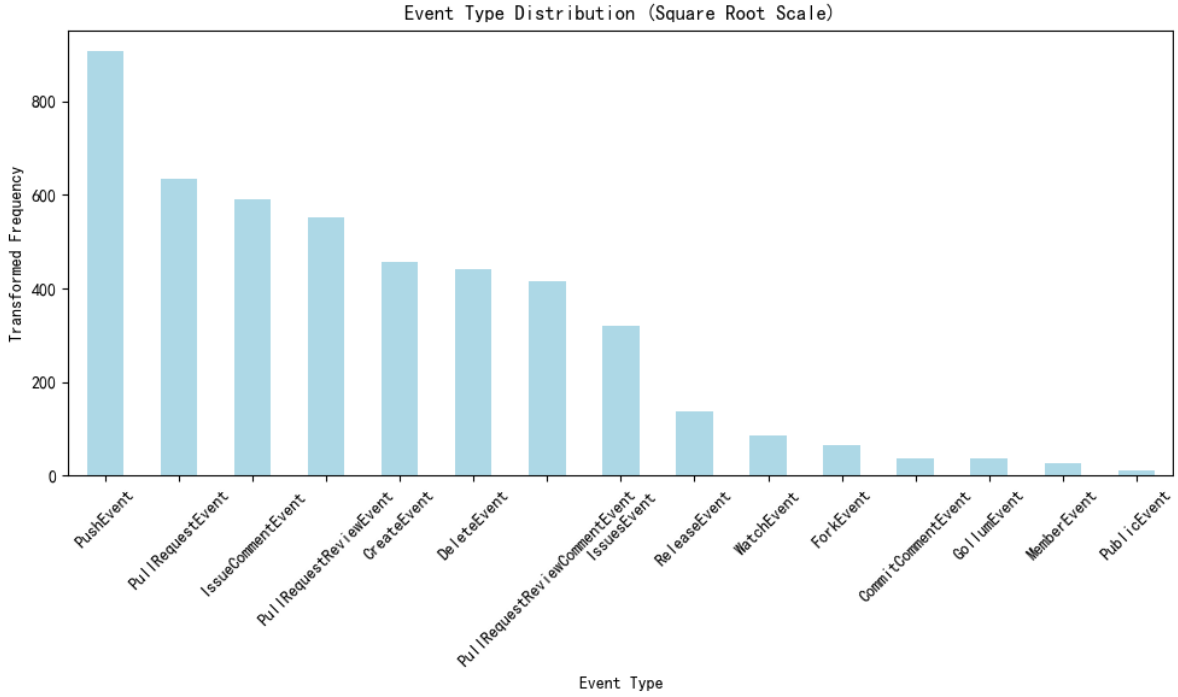
结论：

- 开发者群体的影响力分布呈现长尾效应。
- 高影响力用户可能是社区中的关键人物。

(2) 事件类型分布（平滑处理）

通过对事件类型分布进行平滑处理（取平方根），我们发现：

- Push Event的数量远高于其他事件类型。
- Public Event和Member Event数量相对较少。



结论:

- 提交是开发者最主要的协作行为。
- 评论和合并请求事件的数量较少，可能反映了代码审查的活跃度。

6、总结

本次实验通过对GitHub用户的协作行为数据进行分析，获得了多个关于用户分布和协作模式的洞察。以下是分析结果的总结：

1. **国家与地区分布**:
    - 开发者主要集中在科技发达的国家和地区，尤其是美国、中国和德国等国家。
    - 通过将用户数量较少的国家归为“其他”类别，呈现出全球开发者集中在少数几个国家的分布特征。
    - 这些结果表明，开发者社区的活跃度受各国经济发展水平、技术创新能力以及教育体系等因素的影响。
2. **城市级别分布**:
    - 数据显示，大城市和科技产业集中的城市是开发者集中的主要区域，德国、捷克和日本等城市的开发者数量尤其突出。
    - 开发者分布呈现出明显的“长尾效应”，少数城市聚集了大量的开发者，而大部分城市的开发者数量相对较少。
3. **时区与活动时间分布**:
    - 用户活动在不同的小时之间存在显著差异，比如'France': 'UTC+01:00'用户活跃度较高集中在9-15时间段，'United States': 'UTC-05:00'集中在15-20时间段，体现了用户的本地时间和工作习惯的差异。
    - 各国用户的活跃时间模式也有所不同，可能反映了不同时区的工作节奏和社区参与度。
4. **协作行为分析**:
    - 活跃度差异显著，核心开发者的贡献远远高于一般用户。通过对用户活动频率的分析，我们能够识别出贡献较大的开发者，他们在推动项目进展、提供重要代码或参与讨论方面发挥着关键作用。
    - 此外，用户的活跃程度与其对开源项目的影响力有很强的关联，高影响力用户通常是社区中的核心人物，他们的贡献对项目发展具有重要意义。
5. **其他数据洞察**:

- **用户影响力**：大部分用户的影响力较低，但少数用户的影响力较高，表现出明显的长尾分布特征。
- **事件类型分布**：提交（Push Event）是最常见的事件类型，而公共事件（Public Event）和成员事件（Member Event）则相对较少。这表明大多数开发者的主要贡献体现在代码的提交上，而评论和合并请求的活跃度相对较低。

本次实验揭示了开发者在全球范围内的分布模式，以及他们在不同国家、城市、时区的活跃情况。活跃的开发者和高影响力用户对项目的贡献至关重要，而协作行为和事件类型的分布则反映了开源社区中的常见行为模式。实验结果为进一步研究开源社区的协作机制和开发者行为提供了有价值的数据支持，能够帮助开发者、项目管理者和社区维护者更好地理解开发者的活跃度、贡献模式和区域分布，从而优化协作方式和项目管理。通过本次分析，我们不仅能够获得对GitHub开发者活动的深刻理解，还能够为今后的研究和实践提供理论依据。