

1. 人口统计分析

国家和地区分布：统计用户所在国家和地区的分布，识别主要的开发者集中地。

国家和地区分布：统计用户所在国家和地区的分布，识别主要的开发者集中地。给出完整的python代码并绘制图像，文件路径为data/user_data/users_combined_info_500.csv

XLS

1
CSV, 928 Bytes



为了统计用户所在国家和地区的分布并绘制图像，我们可以使用Python中的pandas库来处理数据，以及matplotlib库来绘制图表。以下是一个完整的Python代码示例，用于实现这一任务：

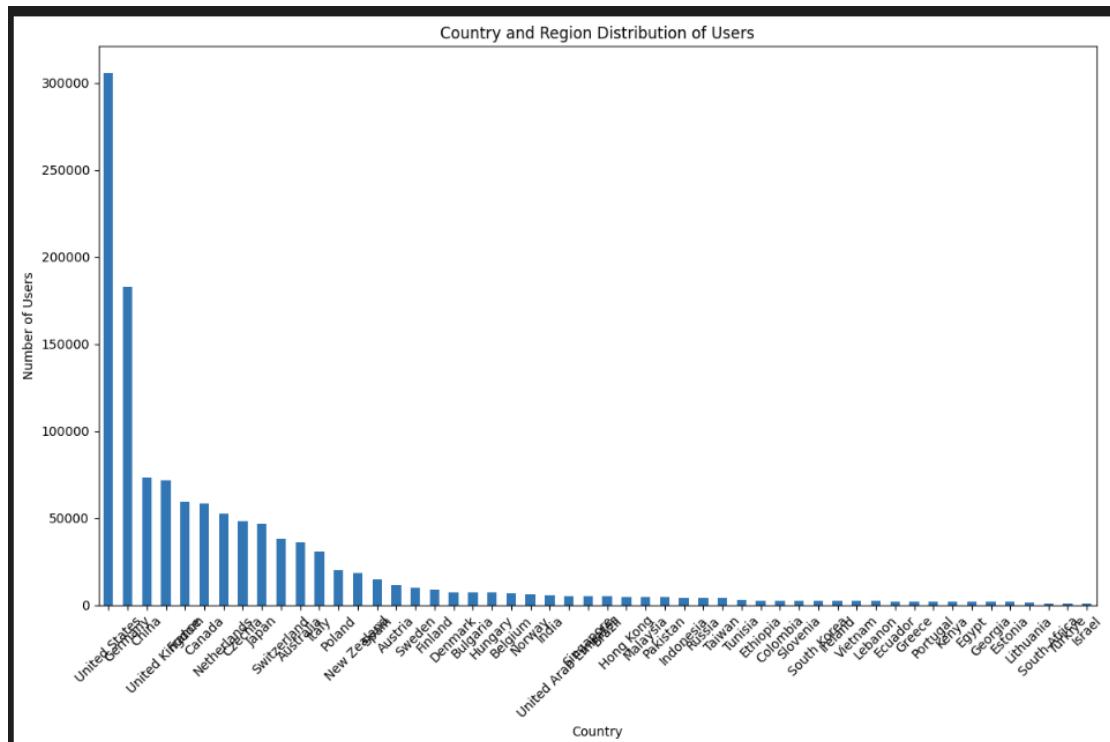
```
python
import pandas as pd
import matplotlib.pyplot as plt

# 加载数据
file_path = 'data/user_data/users_combined_info_500.csv'
data = pd.read_csv(file_path)

# 统计国家和地区分布
country_distribution = data['country'].value counts()
```

复制

得到代码运行结果



由此可知美国、中国等地区是开发者主要集中地

城市级别分布：分析主要城市的开发者密度，发现技术热点区域。

城市级别分布：分析主要城市的开发者密度，发现技术热点区域。同上要求



为了分析主要城市的开发者密度并发现技术热点区域，我们可以使用Python中的pandas库来处理数据，以及matplotlib库来绘制图表。以下是一个完整的Python代码示例，用于实现这一任务：

```
python

import pandas as pd
import matplotlib.pyplot as plt

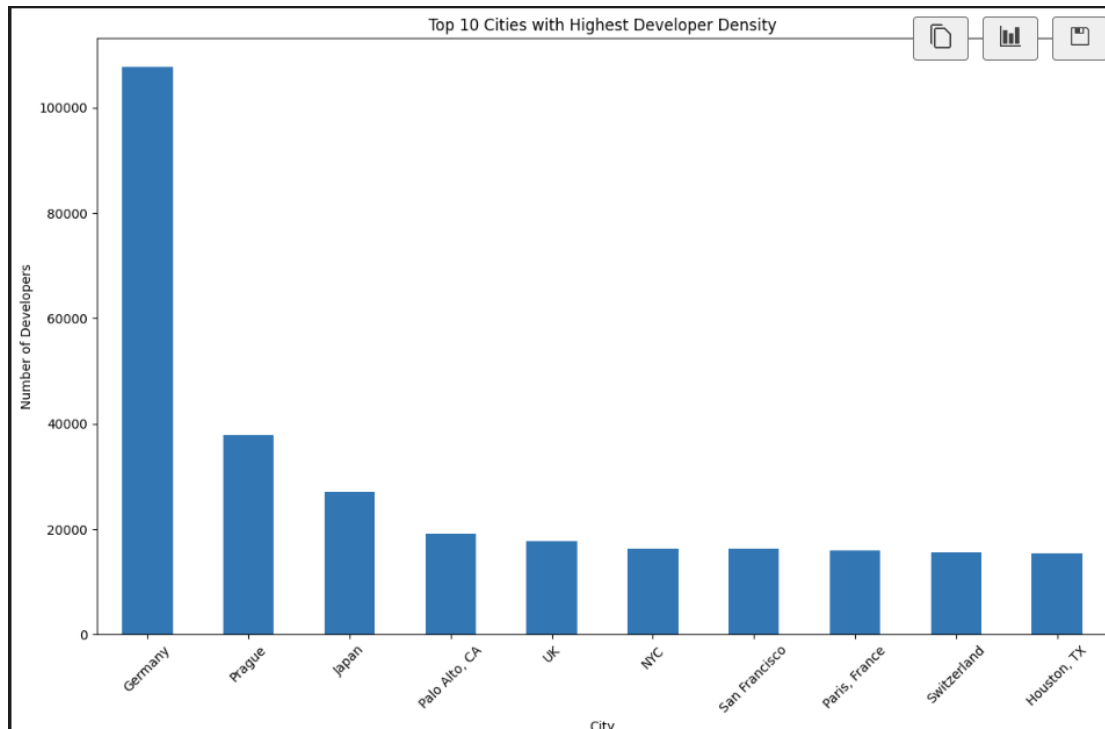
# 加载数据
file_path = 'data/user_data/users_combined_info_500.csv'
data = pd.read_csv(file_path)

# 统计城市的分布
city_distribution = data['location'].value_counts()

# 取前10个城市进行分析
```

复制

得到代码运行结果：



由此可知，德国是开发者密度较高的地区，也是技术热点区域

时区分布:了解用户的时区分布，分析不同地区用户的协作时间模式。

时区分布:了解用户的时区分布，分析不同地区用户的协作时间模式。同上

为了了解用户的时区分布并分析不同地区用户的协作时间模式，我们可以使用Python中的pandas库来处理数据，以及matplotlib库来绘制图表。以下是一个完整的Python代码示例，用于实现这一任务：

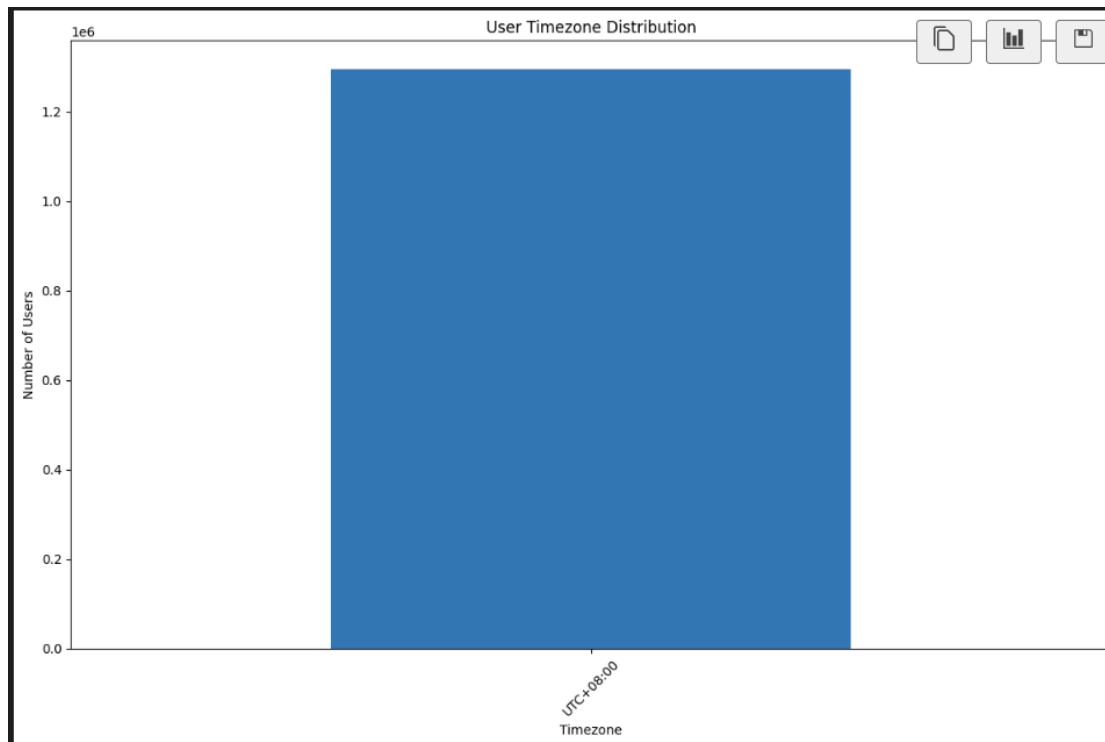
```
python
import pandas as pd
import matplotlib.pyplot as plt
from datetime import datetime

# 加载数据
file_path = 'data/user_data/users_combined_info_500.csv'
data = pd.read_csv(file_path)

# 将事件时间转换为日期时间格式
data['event_time'] = pd.to_datetime(data['event_time'])

# 提取时区信息
data['timezone'] = data['event_time'].dt.tz
```

得到代码运行结果：



由此可知，不同地区用户的协作时间模式大致相同

协作行为分析

提交频率：统计每个用户的提交次数，识别高活跃用户和低活跃用户。

提交频率：统计每个用户的提交次数，识别高活跃用户和低活跃用户。同上



为了统计每个用户的提交次数并识别高活跃用户和低活跃用户，我们可以使用Python中的pandas库来处理数据，并使用matplotlib库来绘制图表。以下是一个完整的Python代码示例，用于实现这一任务：

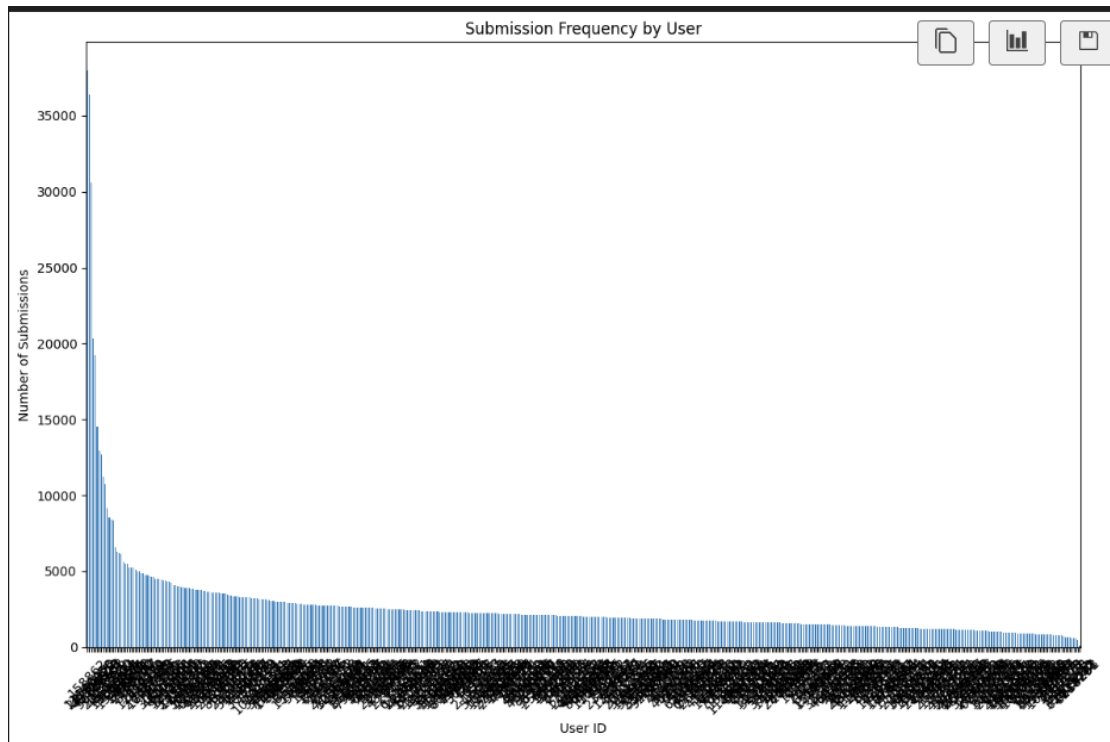
```
python
import pandas as pd
import matplotlib.pyplot as plt

# 加载数据
file_path = 'data/user_data/users_combined_info_500.csv'
data = pd.read_csv(file_path)

# 统计每个用户的提交次数
submission_counts = data['user_id'].value_counts()
```

复制

得到代码运行结果：



High Active Users:

user_id

158862 37960

11146458 36400

28706372 30616

1580956 20300

40306929 19215

...

459648 2614

4616906 2613

259697 2609

1162160 2608

3807253 2606

Name: count, Length: 142, dtype: int64

```
Low Active Users:
user_id
7548295      2590
6281704      2589
161631       2578
4179064      2568
9713184      2554
...
62625502     621
1609022      599
1133777      582
3238291      485
814283       75
Name: count, Length: 355, dtype: int64
```

由此可知高活跃用户和低活跃用户

其他维度有趣洞察

事件类型分布

你能写出其他维度的有趣观察吗，同上的



当然可以！以下是一些其他维度的有趣观察，以及如何使用Python代码来实现这些分析：

1. 事件类型分布

分析不同事件类型的分布，了解用户在平台上的主要活动类型。

python

复制

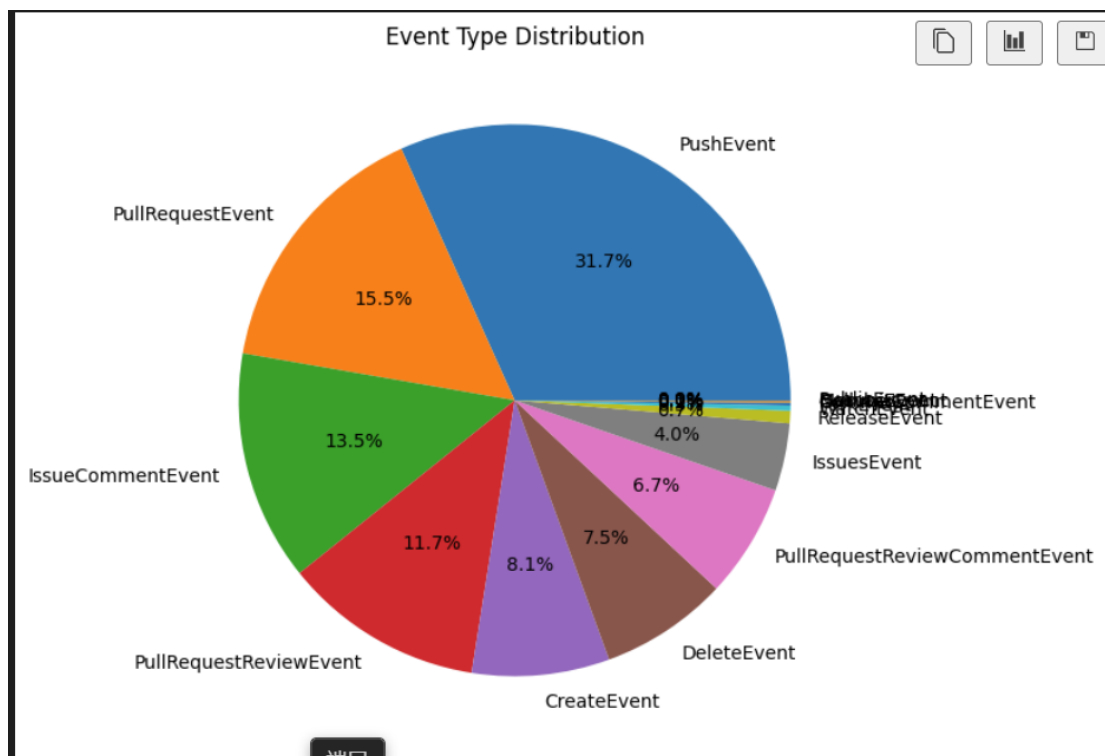
```
import pandas as pd
import matplotlib.pyplot as plt

# 加载数据
file_path = 'data/user_data/users_combined_info_500.csv'
data = pd.read_csv(file_path)

# 统计事件类型的分布
event_type_distribution = data['event_type'].value_counts()

# 绘制饼图
```

得到代码运行结果：



由此可知用户的主要活动类型为 PushEvent

用户活跃时间段

2. 用户活跃时间段

分析用户在一天中哪个时间段最活跃，以便了解最佳的协作时间。

```
python

import pandas as pd
import matplotlib.pyplot as plt

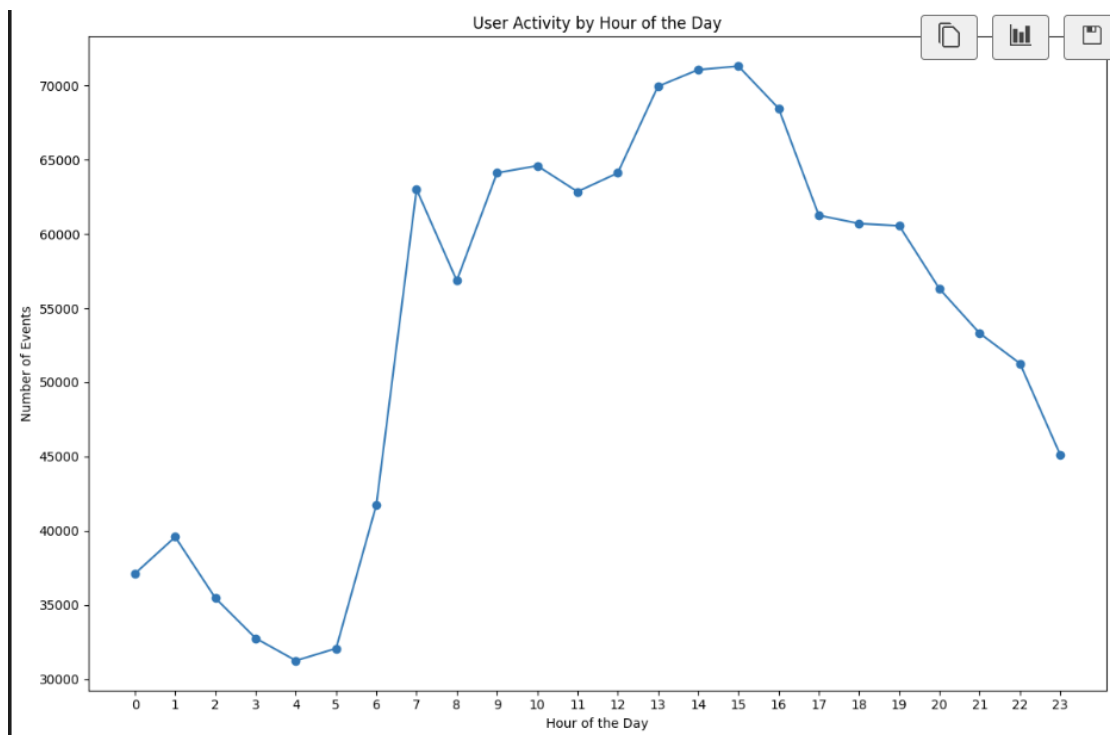
# 加载数据
file_path = 'data/user_data/users_combined_info_500.csv'
data = pd.read_csv(file_path)

# 将事件时间转换为日期时间格式
data['event_time'] = pd.to_datetime(data['event_time'])

# 提取小时信息
data['hour'] = data['event_time'].dt.hour

# 统计每个小时的事件数量
hourly_activity = data['hour'].value_counts().sort_index()
```

得到代码运行结果：



由此可知用户的主要活动时间段为 13~15 点