



# 《Python数据分析案例教程》

## 第十二章 案例报告

# 杭州地铁乘客流量数据分析与可视化研究

Visualization Research and Data Analysis on Passenger Flows of Hangzhou Metro

## 摘要



本章以一个综合案例的数据分析报告作为整个数据分析过程的总结，相关源代码参见本书电子资源网站。



**01** 摘要

**02** 研究背景

**03** 数据获取

**04** 数据预处理

**05** 数据探查

**06** 数据处理

**07** 可视化数据分许

**08** 可解释性数据分析及验证

**09** 数据分析结果

**10** 可能的解决方法

**11** 总结与展望

**12** 参考文献

**目录** content

The background is a dark blue gradient. On the left, there is a stylized globe with glowing blue lines representing a network or data connections. The right side of the image features a network of glowing green and blue dots connected by thin lines, creating a web-like structure.

# PART ONE

摘要

Abstract

# 杭州地铁乘客流量数据分析与可视化研究

Visualization Research and Data Analysis on Passenger Flows of Hangzhou Metro

## ● 摘要

**摘要** 地铁交通具有快速、准时、运输量大、事故率低、相对环保等优势，已经成为城市居民的重要出行方式，也是缓解城市交通压力的主要手段。本文依据杭州地铁交通刷卡数据，地铁路网地图数据，融合地铁沿线房价分布数据以及杭州卫星图等多源数据，运用时间取样法、刷卡计数法等对杭州地铁乘客流量进行统计分析，挖掘杭州地铁乘客出行规律，进而对代表性地铁站点、重要时间段、城市不同功能区的乘客流量规律进行可视化研究。最后，结合杭州实际情况对研究结果进行验证，提出可能的解决方案。

**关键字** 地铁刷卡数据；乘客流量；Python；数据分析；可视化分析

**中图分类号** U293.13; U239.5



# 杭州地铁乘客流量数据分析与可视化研究

Visualization Research and Data Analysis on Passenger Flows of Hangzhou Metro

## ● 摘要

### Visualization Research and Data Analysis on Passenger Flows of Hangzhou Metro

**ABSTRACT** With the characteristics of fast, punctual, large transport volume, low accident rate, green and other features, taking metro has become a customary trip mode of urban residents, which is also the main means to relieve the urban traffic pressure. In this report, the passenger flows of Hangzhou Metro are analyzed using the methods of time sampling and swiped counting, which is based on multi-source data fusion of swiping record, subway network map, the data of housing price distribution along the subway and Hangzhou satellite map. Furthermore, the travel patterns of Hangzhou subway passengers are mined, and the rules of passenger flow are visualized, especially in representative subway stations, important time periods and different functional areas of the city. Finally, the research results are verified and possible solutions are put forward according to the actual situation of Hangzhou.

**Key words** Swiping record; Passenger flows; Python; Data analysis; Visual analysis

# 杭州地铁乘客流量数据分析与可视化研究

Visualization Research and Data Analysis on Passenger Flows of Hangzhou Metro

## ● 摘要

近年来，地铁交通线网规模的不断扩大，地铁交通网络化客流特征日趋明显，地铁交通线网客流分布特征呈现新的特点。分析、研究杭州地铁交通网络化客流特征和规律，可以帮助杭州市民选择更加合理的出行路线，规避交通堵塞；有助于杭州交通管理部门合理分配人力和设备，提前部署地铁站点安保措施；有利于实现大数据助力城市居民快速出行的目标。



# PART TWO

研究背景

Research background



# 研究背景

Research background



杭州地铁 ( Hangzhou Metro ) 是指服务于杭州市及杭州都市圈各地区的城市轨道交通，其首条线路杭州地铁1号线于2012年11月24日正式开通<sup>[1]</sup>。截至2019年5月，杭州地铁运营线路共3条，分别为杭州地铁1号线、杭州地铁2号线、杭州地铁4号线，总营运里程约135.36千米，共设站点81个（包括5个换乘站），日均客运量达到145万左右。其中1号线由杭港地铁负责运营；2号线、4号线由杭州地铁运营分公司负责运营。

截至2018年9月，杭州市城市轨道交通线网规划总里程539公里，其中地铁三期建设规划总里程为387.8公里<sup>[2]</sup>。2022年杭州亚运会前，杭州将形成“10条轨道普线+1条轨道快线+2条市域线”共计13条线路，总长度达516公里的城市轨道交通骨干网络，实现杭州十城区轨道交通线网全覆盖<sup>[3]</sup>。

# 研究背景

Research background

年份	客运总量（亿乘次）	年增长率（%）	总运营里程（km）
2018年	5.3	55.9	117.72
2017年	3.4	26.5	107.02
2016年	2.69	20.3	81.52
2015年	2.23	53.9	81.52
2014年	1.45	57.1	66.27
2013年	1.17		47.97

近年来，地铁在杭州城市生活中扮演着越来越重要的角色。它具有速度快、运量大、污染小、效率高、安全性好等优点，能有效缓解地面交通压力，缓解城市交通的供需矛盾，有效降低整个城市的交通成本，满足城市化日益增长的交通需求。由于杭州地铁的便捷性，它已经成为杭州城市居民的重要出行方式。在杭州，越来越多的出行者选择地铁出行，轨道交通的优势和重要性逐步显现。

# 研究背景

Research background

目前，杭州地铁每个站点均有闸机，乘客可以选择包括杭州通通用卡、交通卡、学生卡、长者卡以及优待卡等刷卡入闸乘车，也可以直接扫支付宝“杭州地铁乘车码”实现扫码进出站点。在刷卡或扫码乘车过程中会产生大量的刷卡数据，包括进站、出站、进站时间和出站时间等。从这些地铁乘客流量相关数据中分析整个地铁系统的交通流量变化，探索代表性地铁站点、重要时间段、城市不同功能区在杭州地铁系统中的乘客流量变化规律，对改善杭州地铁交通站点周边的交通状况，降低沿线及周边居民的出行时间成本和经济成本，提高城市居民的生活水平，具有明显的实用价值。作为杭州轨道交通项目进一步建设的前提，客流统计分析和可视化研究为下一步建设规模的确定、车辆选型及编组方案、设备配置、运输组织、经济效益评价以及工程投资等提供依据，同时也对杭州地铁管理和运营部门进行地铁安全预警和控制具有重要意义<sup>[4]</sup>。





# PART THREE

数据获取

Data acquisition

# 数据获取

Data acquisition

01

“阿里天池全球城市计算AI挑战赛”开放了20190101至20190125共25天杭州地铁刷卡数据记录<sup>[5]</sup>，共涉及3条线路81个地铁站约7000万条数据作为训练数据（Metro\_train.zip）。同时提供了路网地图，即各地铁站之间的连接关系表，存储在文件Metro\_roadMap.csv文件中。

# 数据获取

Data acquisition

02

将训练数据压缩包Metro\_train.zip解压后得到25个csv文件，每天的刷卡数据单独存在一个csv文件中，以record为前缀。如2019年1月1日所有线路所有站点的刷卡数据记录存储在record\_2019-01-01.csv文件中，以此类推。在record\_2019-01-xx.csv文件中，除第一行外，其余每行包含一条乘客刷卡记录。对于userID属性，在payType属性为3时无法唯一标识用户身份。即此userID可能为多人使用，但在一次进出站期间可以视为同一用户。对于其他取值的payType，对应的userID可以唯一标识一个用户。

列名	类型	说明	示例
time	String	刷卡发生时间	2019/2/1 0:30
lineID	String	地铁路线ID	C
stationID	int	地铁站ID	15
deviceID	int	刷卡设备编号ID	2992
status	int	进出站状态，0为出站，1为进站	1
userID	String	用户身份ID	Ad53ce59370e8b141dbc99c03d2158fe4
payType	int	用户刷卡类型	0



# 数据获取

Data acquisition

03

路网地图文件Metro\_roadMap.csv提供了各地铁站之间的连接关系表，相应的邻接矩阵存储在roadMap.csv中，其中包含一个 $81 \times 81$ 的二维矩阵。文件中首行和首列表示地铁站ID（stationID），均为 $[0, 80]$ 区间的整数。其中 $\text{roadMap}[i][j] = 1$ 表示stationID为 $i$ （ $0 \leq i \leq 80$ ）的地铁站和stationID为 $j$ （ $0 \leq j \leq 80$ ）的地铁站直接相连； $\text{roadMap}[i][j] = 0$ 表示stationID为 $i$ 的地铁站和stationID为 $j$ 的地铁站不相连。此外，测试数据包括2019年1月26日或28日的刷卡数据，大赛要求对2019年1月27日或29日全天各地铁站以10分钟为单位的乘客流量进行预测。这些数据来自杭州地铁公司和杭州市公安机关，比较可靠。



# PART FOUR

## 数据预处理

Data preprocessing

# 数据预处理

Data preprocessing

## ● 数据清洗



“

杭州地铁刷卡数据预处理包括数据缺失值与异常值的探索分析，数据的属性规约、清洗、和变换等。

杭州地铁刷卡数据清洗是指对数据进行重新审查和校验，目的在于删除重复信息、纠正存在的错误，并提供数据一致性。

”



# 数据预处理

Data preprocessing

## 数据清洗

01

### 空缺和缺失值的处理

本文使用Python扩展库pandas中的isnull()方法和notnull()方法判断数据集中是否存在空值和缺失值。这里以杭州地铁站1月16日三条线路所有站点的刷卡数据记录为例。isnull()方法返回值全为“False”，说明没有一个空值或缺失值；notnull()方法返回值全为“True”，没有一个空值或缺失值，说明提供的数据很干净。由于stationID为54的站点数据缺失，在后续的数据处理中使用 fillna()方法将54站点的数据用零填充；27日数据没有预测，也用零填充。

```
In [3]: test_28 = pd.read_csv(open(path + '/Metro_train/record_2019-01-16.csv', encoding='utf8'))
In [4]: test_28.isnull()
Out[4]:
```

	time	lineID	stationID	deviceID	status	userID	payType
0	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...
2417054	False	False	False	False	False	False	False
2417055	False	False	False	False	False	False	False
2417056	False	False	False	False	False	False	False
2417057	False	False	False	False	False	False	False
2417058	False	False	False	False	False	False	False

2417059 rows × 7 columns

```
In [5]: test_28.notnull()
Out[5]:
```

	time	lineID	stationID	deviceID	status	userID	payType
0	True	True	True	True	True	True	True
1	True	True	True	True	True	True	True
2	True	True	True	True	True	True	True
3	True	True	True	True	True	True	True
4	True	True	True	True	True	True	True
...	...	...	...	...	...	...	...
2417054	True	True	True	True	True	True	True
2417055	True	True	True	True	True	True	True
2417056	True	True	True	True	True	True	True
2417057	True	True	True	True	True	True	True
2417058	True	True	True	True	True	True	True

2417059 rows × 7 columns

# 数据预处理

Data preprocessing

## 数据清洗

02

### 重复值的处理

乘客地铁刷卡记录中的时间点是唯一的，不可能同一个乘客在相同的时间点（如2019/01/01 08：11）有两条甚至是三条相同的记录出现。因为一位乘客搭乘地铁一次是不可能出现多条记录的，除非系统记录出现漏洞，所以这次数据清洗需要检查并处理重复值。本文使用 duplicated() 方法检测刷卡数据是否有重复值，所有的标记都显示为“False”。说明杭州地铁站1月16日三条线路所有站点的刷卡数据记录没有重复值。

```
In [6]: test_28.duplicated()

Out[6]: 0          False
        1          False
        2          False
        3          False
        4          False
        ...
        2417054     False
        2417055     False
        2417056     False
        2417057     False
        2417058     False
        Length: 2417059, dtype: bool
```

# 数据预处理

Data preprocessing

## 数据清洗

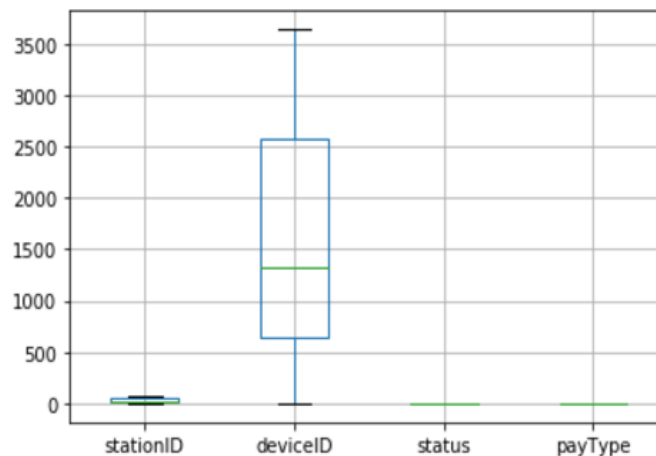
03

### 异常值的处理

异常值是指样本中的个别值，其数值明显偏离所属样本集的其余观测值，这些数值是不合理的或错误的，需要进行检查和处理。检查一组数据是否包含异常值，常用箱型图进行可视化查看。在箱型图的上下界之外的离散点表示异常值。从箱型图可以看出杭州地铁站1月16日三条线路所有站点的刷卡数据记录没有出现离散点，说明提供的数据规范干净，没有出现异常值。

```
In [9]: test_28.boxplot()
```

```
Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0xb46d688>
```





# 数据预处理

Data preprocessing

## 数据合并



杭州地铁刷卡数据集中每天的刷卡数据均单独存放在一个csv文件中，这些刷卡数据文件格式相同。为了方便数据处理，本文将需要分析的25个数据文件合并为一个完整的总数据文件。数据合并操作使用pandas.concat()方法，将数据按照纵轴进行简单的数据融合。以图5为例，分别读取1月1号到1月3号这三天的刷卡数据，将相同字段的数据内容首尾相接，合并这三张表。在后面的数据处理中，本文依据此方法将1月1日到1月25日的刷卡数据首尾相连，合并为一个完整的数据文件。

```
test_1 = pd.read_csv(open(path + '/Metro_train/record_2019-01-01.csv', encoding='utf8'))
test_2 = pd.read_csv(open(path + '/Metro_train/record_2019-01-02.csv', encoding='utf8'))
test_3 = pd.read_csv(open(path + '/Metro_train/record_2019-01-03.csv', encoding='utf8'))
```

```
In [3]: file_data=[test_1, test_2, test_3]
```

```
In [4]: result=pd.concat(file_data)
```

```
In [5]: result
```

```
Out[5]:
```

	time	lineID	stationID	deviceID	status	userID	payType
0	2019-01-01 02:00:05	B	27	1354	0	D13f76f42c9a677c4add94d9e480fb5c5	3
1	2019-01-01 02:01:40	B	5	200	1	D9a337d37d9512184b8e3fd477934b293	3
2	2019-01-01 02:01:53	B	5	247	0	Dc9e179298617f40b782490c1f3e2346c	3
3	2019-01-01 02:02:38	B	5	235	0	D9a337d37d9512184b8e3fd477934b293	3
4	2019-01-01 02:03:42	B	23	1198	0	Dd1cde61886c23fdb7ef1fdb76c9b1234	3
...	...	...	...	...	...	...	...
2293466	2019-01-03 23:59:33	C	64	2979	0	Af01db6ea87fd8e7df7f09087abf3ac07	0
2293467	2019-01-03 23:59:36	C	35	1671	0	B706e4e60de5b413520b7506b092069a2	1
2293468	2019-01-03 23:59:39	C	35	1674	0	C409614c3ad090164ca7fa29e5b7b49c4	2
2293469	2019-01-03 23:59:57	C	64	2981	0	Adee7375f5ef8999b94894e8f1b3b0471	0
2293470	2019-01-03 23:59:59	C	64	2980	0	B4cb86f8efaff4ea0103fa66f7a10ae51	1

7209525 rows × 7 columns



# PART FIVE

## 数据探查

Data exploration



# 数据探查

Data exploration

## 数据读取

读取杭州地铁三条线路所有站点的刷卡数据记录，以1月16日的数据为例。可以看出，刷卡数据记录包括刷卡时间、乘客所在线路、搭车站点、刷卡设备号等，其中刷卡状态为0或1，表示“有”或“没有”刷卡，客户ID是一个字符串，支付类型有三种。



读取csv数据文件

```
file_data = pd.read_csv(file_path)
file_data
```

Out[1]:

	time	lineID	stationID	deviceID	status	userID	payType
0	2019-01-16 00:00:05	C	64	2980	0	Bee069dae5399509d4427e1bda7a344ff	1
1	2019-01-16 00:00:12	C	64	2980	0	D755fb649e121396c2cbd1d077979f1d7	3
2	2019-01-16 00:00:28	B	31	1523	1	D46300a0b9cfbc742a02315a5e3a40483	3
3	2019-01-16 00:02:20	C	65	3036	0	C656e7789dba16442c2330d4015ff35a2	2
4	2019-01-16 00:02:22	C	65	3019	0	B4be1b508ef536ec3ede05386ca967926	1
...	...	...	...	...	...	...	...
2417054	2019-01-16 23:59:32	C	64	2993	0	B771656a569a81981f268fbc53cd47a81	1
2417055	2019-01-16 23:59:34	C	64	2980	0	Aeb94121ffb9ccc49ac39b76879b4d761	0
2417056	2019-01-16 23:59:35	C	64	2994	0	Bdc60ad97bef1044bcf35ad305358ebbe	1
2417057	2019-01-16 23:59:54	C	64	2994	0	B78adaa0231664c6e6d91758e25f7094c	1
2417058	2019-01-16 23:59:55	C	35	1687	0	Ae99a6b1614b206e1aa783d39d25de358	0

2417059 rows × 7 columns

# 数据探查

Data exploration

## 数据分组

为了针对杭州地铁刷卡数据进行统计研究，本文首先将原始数据按照特征划分成不同的组别，得到分组数据。这里进行数据分组的目的是观察数据的分布特征，为接下来的数据聚合做准备。

```
In [2]: #创建一个DataFrame对象, 该对象只有一列数据:lineID[线路ID]
new_df = pd.DataFrame({'lineID':file_data['lineID'].unique()})
new_df
```

```
Out[2]:
```

	lineID
0	C
1	B
2	A

按lineID列数据分组

```
In [4]: groupy_area = file_data.groupby(by='lineID').count()
groupy_area
#按“userID”一列从大到小排列
groupy_area.sort_values(by=['userID'], ascending=False)
```

```
Out[4]:
```

	time	stationID	deviceID	status	userID	payType
lineID						
B	1347738	1347738	1347738	1347738	1347738	1347738
C	792347	792347	792347	792347	792347	792347
A	276974	276974	276974	276974	276974	276974

按userID列排序

```
In [6]: #按“userID”一列从大到小排列
groupy_area.sort_values(by=['userID'], ascending=False)
```

```
Out[6]:
```

	time	lineID	deviceID	status	userID	payType
stationID						
15	173048	173048	173048	173048	173048	173048
9	91220	91220	91220	91220	91220	91220
4	75352	75352	75352	75352	75352	75352
7	67875	67875	67875	67875	67875	67875
10	55754	55754	55754	55754	55754	55754
...	...	...	...	...	...	...
28	9303	9303	9303	9303	9303	9303
35	9229	9229	9229	9229	9229	9229
31	8433	8433	8433	8433	8433	8433
72	8301	8301	8301	8301	8301	8301
74	6652	6652	6652	6652	6652	6652

80 rows × 6 columns

按stationID列数据分组

```
In [8]: #按“userID”一列从大到小排列
groupy_area.sort_values(by=['userID'], ascending=False)
```

```
Out[8]:
```

	time	lineID	stationID	status	userID	payType
deviceID						
474	8634	8634	8634	8634	8634	8634
473	8396	8396	8396	8396	8396	8396
475	7936	7936	7936	7936	7936	7936
1149	7661	7661	7661	7661	7661	7661
156	7273	7273	7273	7273	7273	7273
...	...	...	...	...	...	...
3379	1	1	1	1	1	1
3378	1	1	1	1	1	1
3304	1	1	1	1	1	1
3303	1	1	1	1	1	1
3279	1	1	1	1	1	1

1700 rows × 6 columns

按deviceID数据分组



# 数据探查

Data exploration

## 数据分组

首先读取lineID这一列，只取唯一值。通过读取lineID特征的唯一值可以看出，给定的所有数据文件中只涉及三条地铁线路的刷卡数据。接着按照lineID特征将数据分组，统计每个分组的数量。然后按照“userID”一列从大到小排列得到统计结果：B线路的人流量最大，A线路人流量最小，所以B线路更容易发生人流拥堵问题。再按stationID列进行数据分组，统计每个分组的数量。

```
In [2]: #创建一个DataFrame对象, 该对象只有一列数据:lineID[线路ID]
new_df = pd.DataFrame({'lineID':file_data['lineID'].unique()})
new_df
```

Out[2]:

	lineID
0	C
1	B
2	A

按lineID列数据分组

```
In [4]: groupy_area = file_data.groupby(by='lineID').count()
groupy_area
#按“userID”一列从大到小排列
groupy_area.sort_values(by=['userID'], ascending=False)
```

Out[4]:

	time	stationID	deviceID	status	userID	payType
lineID						
B	1347738	1347738	1347738	1347738	1347738	1347738
C	792347	792347	792347	792347	792347	792347
A	276974	276974	276974	276974	276974	276974

按userID列排序

# 数据探查

Data exploration

## 数据分组

```
In [6]: #按 "userID" 一列从大到小排列
groupby_area.sort_values(by=['userID'], ascending=False)
```

```
Out[6]:
```

	time	lineID	deviceID	status	userID	payType
stationID						
15	173048	173048	173048	173048	173048	173048
9	91220	91220	91220	91220	91220	91220
4	75352	75352	75352	75352	75352	75352
7	67875	67875	67875	67875	67875	67875
10	55754	55754	55754	55754	55754	55754
...	...	...	...	...	...	...
28	9303	9303	9303	9303	9303	9303
35	9229	9229	9229	9229	9229	9229
31	8433	8433	8433	8433	8433	8433
72	8301	8301	8301	8301	8301	8301
74	6652	6652	6652	6652	6652	6652

80 rows x 6 columns

仍然按照“userID”一列从大到小排列。可以看出，15号站点人流量最大，容易发生人流拥堵问题，74号站点人流量最少。

还可以按照deviceID列进行数据分组，统计每个分组的数量。按“userID”一列从大到小排列后可以看出，474号设备的刷卡数最多，3130号设备的刷卡数最少。

```
In [8]: #按 "userID" 一列从大到小排列
groupby_area.sort_values(by=['userID'], ascending=False)
```

```
Out[8]:
```

	time	lineID	stationID	status	userID	payType
deviceID						
474	8634	8634	8634	8634	8634	8634
473	8396	8396	8396	8396	8396	8396
475	7936	7936	7936	7936	7936	7936
1149	7661	7661	7661	7661	7661	7661
156	7273	7273	7273	7273	7273	7273
...	...	...	...	...	...	...
3379	1	1	1	1	1	1
3378	1	1	1	1	1	1
3304	1	1	1	1	1	1
3303	1	1	1	1	1	1
3279	1	1	1	1	1	1

1700 rows x 6 columns

# 数据探查

Data exploration

## ● 数据聚合

按照特定条件将数据划分为不同的分组后，通过数据聚合对每个分组中的数据执行操作，将计算结果整合。本文分别使用lineID、stationID和deviceID特征对杭州地铁打卡数据分组，求得每个分组的最大值，最小值，中位数，平均值等，得知人流量为中位数的站点是stationID为39和63的站点。

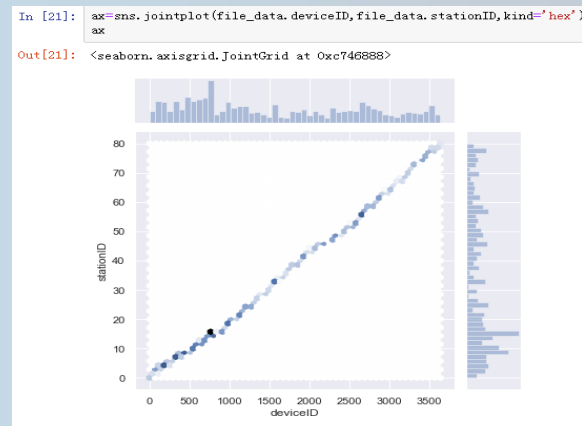
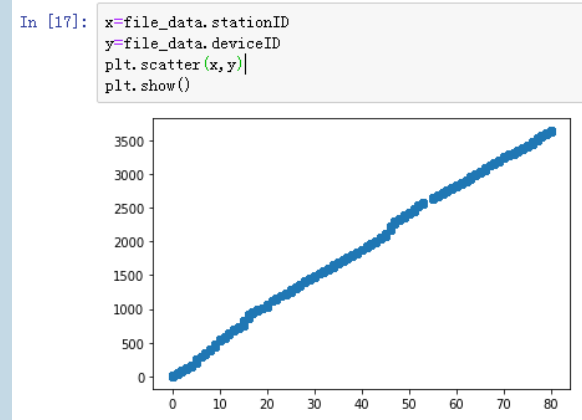
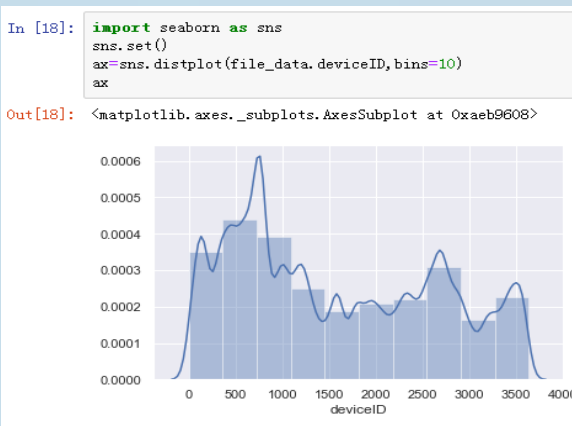
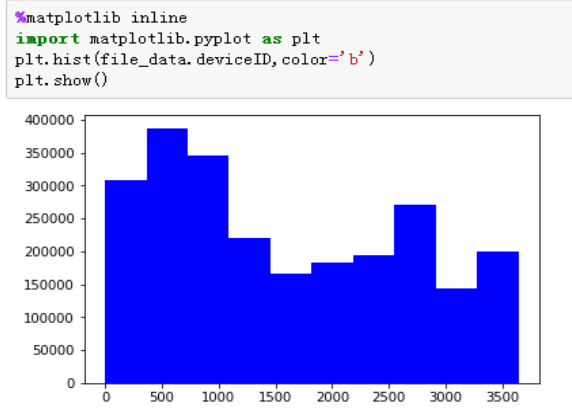


# 数据探查

Data exploration

## 数据可视化

使用图形和图表可以将数据特征和变量清晰有效地展示出来。通过不同维度探查数据，可以更深入观察和分析数据。本文分别使用matplotlib库和seaborn库中的图形绘制函数，从不同角度绘制了1月16日杭州地铁打卡数据的直方图和散点图。

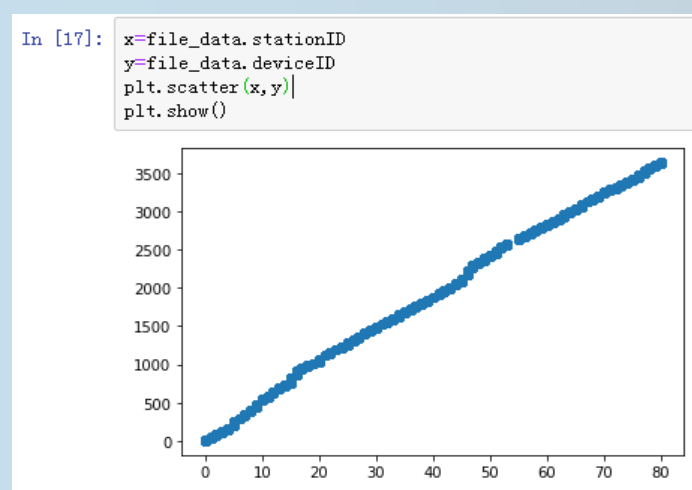
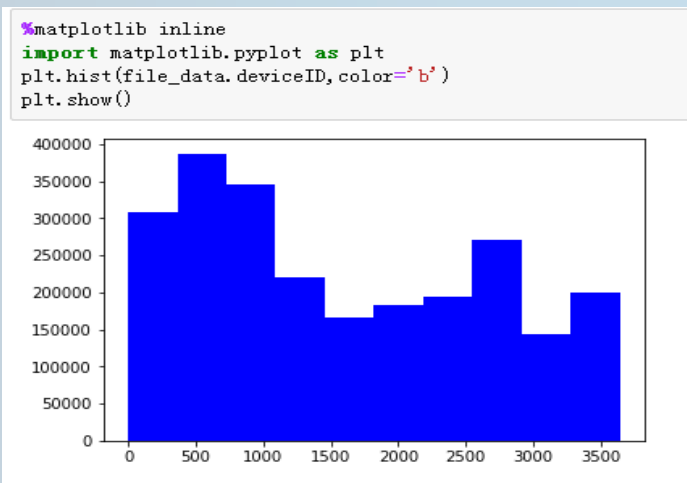




# 数据探查

Data exploration

## 数据可视化



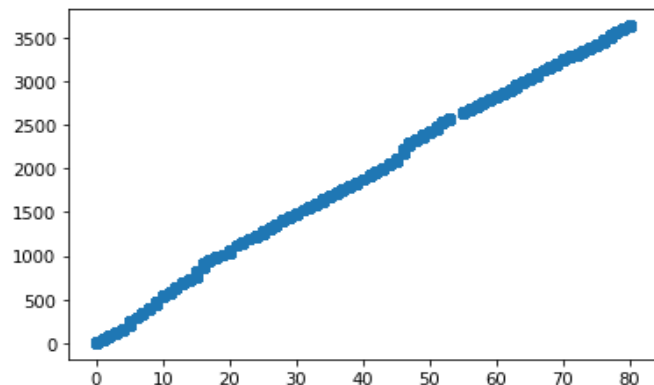
通过对比可以看出，直方图适合表示数量的多少，而散点图适合描述若干数据系列中各数值之间的关系。

# 数据探查

Data exploration

## 数据可视化

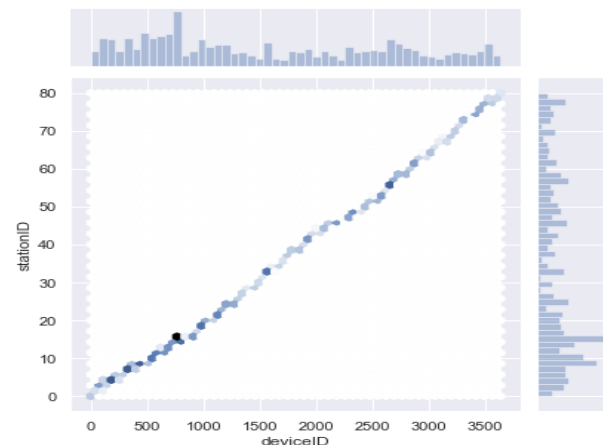
```
In [17]: x=file_data.stationID  
y=file_data.deviceID  
plt.scatter(x,y)  
plt.show()
```



```
In [21]: ax=sns.jointplot(file_data.deviceID, file_data.stationID, kind='hex')
```

```
ax
```

```
Out[21]: <seaborn.axisgrid.JointGrid at 0xc746888>
```



两个散点图均明确显示出stationID为54的站点数据缺失。

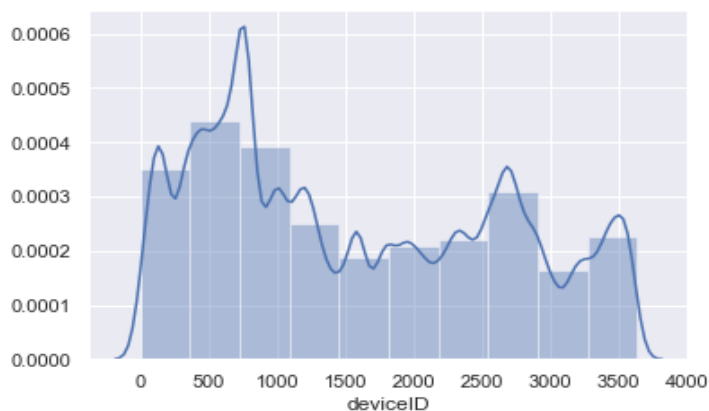
# 数据探查

Data exploration

## 数据可视化

```
In [18]: import seaborn as sns  
sns.set()  
ax=sns.distplot(file_data.deviceID,bins=10)  
ax
```

```
Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0xae9608>
```



明确显示出deviceID不同的设备承担刷卡任务呈现不均衡性。



# PART **SIX**

数据处理

Data processing






# 数据处理

Data processing

杭州地铁打卡数据处理主要由两部分组成：构建完整数据集及特征并进行数据处理；融合多源数据构建空间地铁线路以及连接关系表。



构建完整数据集及特征并  
进行数据处理

融合多源数据构建空间地  
铁线路以及连接关系表

# 数据处理

Data processing

## ● 构建完整数据集

### Step 01 构造训练数据集的基本特征

杭州地铁刷卡数据文件主要包括刷卡时间time、线路lineID、站点stationID、闸机deviceID、乘客userID等特征。为了构建用于杭州地铁打卡数据处理的完整数据集，首先将源数据文件的时间特征细粒度化，拆分成三个时间特征：minute、hour和day，其中minute特征以十分钟为计数单位；增加了week和weekend特征，分别表示“星期几（整型数据）”，“是否为周末（标称数据）”。然后以10分钟为计数单位统计“刷卡次数”和“累计刷卡总数”。接着以不同列作为分组依据统计“进站人数”和“出站人数”，得到杭州地铁刷卡数据分析需要的两个重要特征：inNums和outNums特征。下面加载Metro\_train文件夹下保存1月1日至1月25日杭州地铁刷卡数据的25个CSV文件，调用自定义函数get\_base\_features(df\_)分别生成数据集的基本特征，将25个刷卡数据文件生成的基本特征统计合并至一个总的文件。

# 数据处理

Data processing

## 构建完整数据集

```
def get_base_features(df_):
    df = df_.copy()

    # base time
    df['day'] = df['time'].apply(lambda x: int(x[8:10]))
    df['week'] = pd.to_datetime(df['time']).dt.dayofweek + 1
    df['weekend'] = (pd.to_datetime(df['time']).dt.weekday >= 5).astype(int)
    df['hour'] = df['time'].apply(lambda x: int(x[11:13]))
    df['minute'] = df['time'].apply(lambda x: int(x[14:15] + '0'))

    # count, sum
    result = df.groupby(['stationID', 'week', 'weekend', 'day', 'hour', 'minute']).status.agg(
        ['count', 'sum']).reset_index()

    # nunique deviceID 闸机编号
    tmp = df.groupby(['stationID'])['deviceID'].nunique().reset_index(name='nuni_deviceID_of_stationID')
    result = result.merge(tmp, on=['stationID'], how='left')
    tmp = df.groupby(['stationID', 'hour'])['deviceID'].nunique().reset_index(name='nuni_deviceID_of_stationID_hour')
    result = result.merge(tmp, on=['stationID', 'hour'], how='left')
    tmp = df.groupby(['stationID', 'hour', 'minute'])['deviceID'].nunique(). \
        reset_index(name='nuni_deviceID_of_stationID_hour_minute')
    result = result.merge(tmp, on=['stationID', 'hour', 'minute'], how='left')

    # in, out
    result['inNums'] = result['sum']
    result['outNums'] = result['count'] - result['sum']

    #
    result['day_since_first'] = result['day'] - 1
    result.fillna(0, inplace=True)
    del result['sum'], result['count']

    return result
```

构造数据集的基本特征

# 数据处理

Data processing

## 构建完整数据集

### Step 02 构造测试结果文件的基本特征

构造测试结果文件（如：1月27日）所需的特征，主要为时间特征，包括是否周末（weekend列，标称数据），打卡前一天是几号（day\_since\_first，整型数据）等，删除startTime、endTime列。代码如图所示。其中54站点数据缺失，用零填充。

```
In [10]: test = get_test_features(test)
data = pd.concat([data, test], axis=0, ignore_index=True)
#####
#构造全部枚举值
temp_df = pd.DataFrame({"minute": [], "hour": [], "day": [], "stationID": []})
for station in range(81):
    print(station)
    for day in range(1, 29):
        for hour in range(24):
            temp = pd.DataFrame({"minute": [0, 10, 20, 30, 40, 50]})
            temp["hour"] = int(hour)
            temp["day"] = int(day)
            temp["stationID"] = int(station)
            temp_df = pd.concat([temp_df, temp], axis=0)
temp_df = temp_df.reset_index(drop=True)
data_min_all = temp_df.merge(data, on=["stationID", "day", "hour", "minute"], how="left")
data_min_all = data_min_all.fillna(0)
```

# 数据处理

Data processing

## ● 构建完整数据集

### Step 03 构建一个存放28天打卡数据的完整数据集文件

经过上面两个步骤，已经把1-28日的刷卡数据合并至data\_all\_b.csv文件。该数据集文件包含1月1-28日的杭州地铁打卡数据，囊括81个车站，每个小时的特征数据。因为54站点数据缺失，所以用零填充；由于27日的数据没有预测，也用零填充。该数据集包含11个特征：Minute, hour, day、day\_since\_first、inNums、nuni\_deviceID\_of\_stationID、nuni\_deviceID\_of\_stationID\_hour、nuni\_deviceID\_of\_stationID\_hour\_minute、outNums, week、weekend和stationID。

data\_all\_b.csv文件局部数据

minute	hour	day	stationID	day_since	inNums	nuni_dev	nuni_dev	nuni_devi	outNums	week	weekend
0	0	28	80	0	0	0	0	0	0	1	0
10	0	28	80	0	0	0	0	0	0	1	0
20	0	28	80	0	0	0	0	0	0	1	0
30	0	28	80	27	1	14	3	2	1	1	0
40	0	28	80	0	0	0	0	0	0	1	0
50	0	28	80	27	1	14	3	2	1	1	0
0	1	28	80	0	0	0	0	0	0	1	0



# 数据处理

Data processing

## ● 构建地铁线路以及连接关系表



杭州地铁营运关系图

截止2019年2月，杭州地铁共开通三条线路。依据数据预处理阶段的统计结果，本文数据集只涉及A、B、C三条地铁线路。因此，对给定源文件Metro\_roadMap.csv进行数据处理，融合杭州地铁营运关系图和路网地图，得到81个地铁站点对应地理位置及站点名称，保存在Station\_ID.xlsx文件中。

# 数据处理

Data processing

## ● 构建地铁线路以及连接关系表

stationID	lineID	线路	其他	相隔站	站名	特殊站	
0	B	一号线	终点		湘湖		
1	B	一号线			滨康路		
2	B	一号线			西兴		
3	B	一号线			滨和路		
4	B	一号线			江陵路		
5	B	一号线	4号换乘	2	近江		
6	B	一号线			婺江路	汽车南站	
7	B	一号线			城站	火车站	站前客运站
8	B	一号线			定安路		
9	B	一号线	2号换乘	1	龙翔桥		
10	B	一号线	2号换乘	2	凤起路		
11	B	一号线	2号换乘	1	武林广场		
12	B	一号线			西湖文化广场		
13	B	一号线			打铁关		
14	B	一号线			闸弄口		
15	B	一号线	4号换乘	1	火车东站	火车东站	
16	B	一号线			彭埠		
17	B	一号线			七堡		
18	B	一号线			九和路		
19	B	一号线			九堡		
20	B	一号线	交叉点	1	客运中心	客运中心	

Station\_ID.xlsx文件

如图所示，源数据文件中B路线包含地铁站点34个，为杭州地铁1号线；C路线包含地铁站点33个，为杭州地铁2号线；A路线包含地铁站点14个，为杭州地铁4号线。



# PART SEVEN

## 可视化数据分析

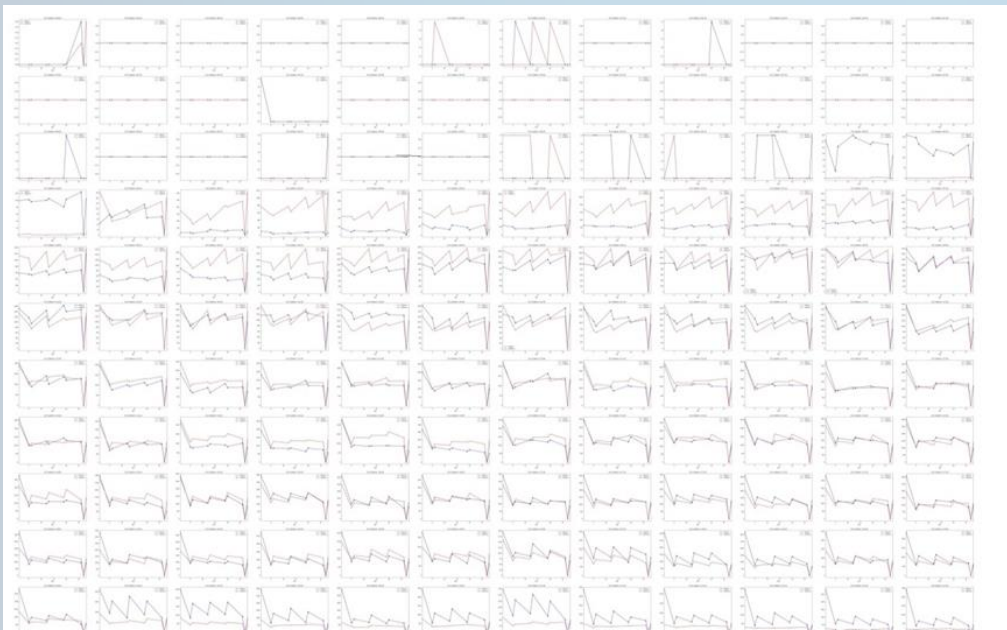
Visual data analysis

# 可视化数据分析

Visual data analysis

## ● 确定计数单位，生成可视化图形

首先重构时间数据，使得图形化界面上时间标签的显示形式为“小时:分钟”，小时数值和分钟数值各占两位。然后对本文数据文件中出现的所有81个站点，分别以10分钟为计数单位，构建包括周末及节假日在内每天出入站人流量的折线图。其中inNums为入站人流量，显示为蓝色；outNums为出站人流量，显示为红色。这样，每天每个站点生成144张折线图并保存至文件夹fig\_holiday\_min内，文件名为“站点编号”。



以10分钟为计数单位的人流量折线图

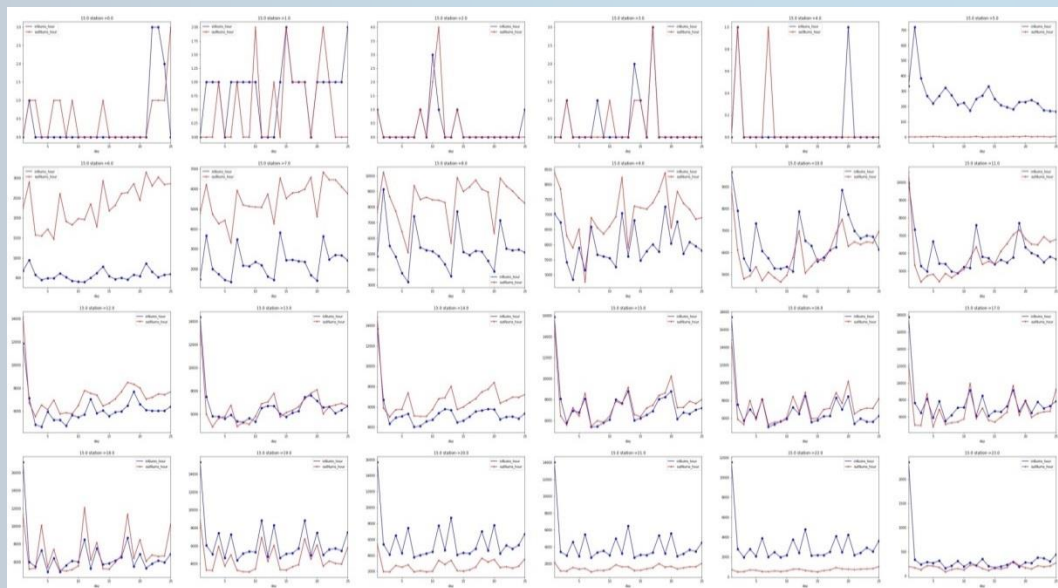
图显示的是人流量最大的15站点一天的刷卡数据。可以看出，以10分钟为计数单位的人流量折线图中，前后时间段的图形相似度较大，更适合细粒度数据分析。如果仅根据图形可视化结果进行粗粒度数据分析，这些图形之间存在一定的数据相似性，可视化数据分析效果较差。

# 可视化数据分析

Visual data analysis

## ● 确定计数单位，生成可视化图形

因此，本文采用60分钟为计数单位，分别对杭州地铁每个站点的出入人流量进行可视化图形输出。以站点编号为文件名保存，每个jpg文件存储24幅不同时段的数据折线图。



以60分钟为计数单位的人流量折线图

从图可以明显看出，在一个周期（7天）内，杭州地铁站点工作日和周末的出入人流量呈现出显著差异。因此，本文将工作日和周末的出入站点人流量分别进行可视化分析。

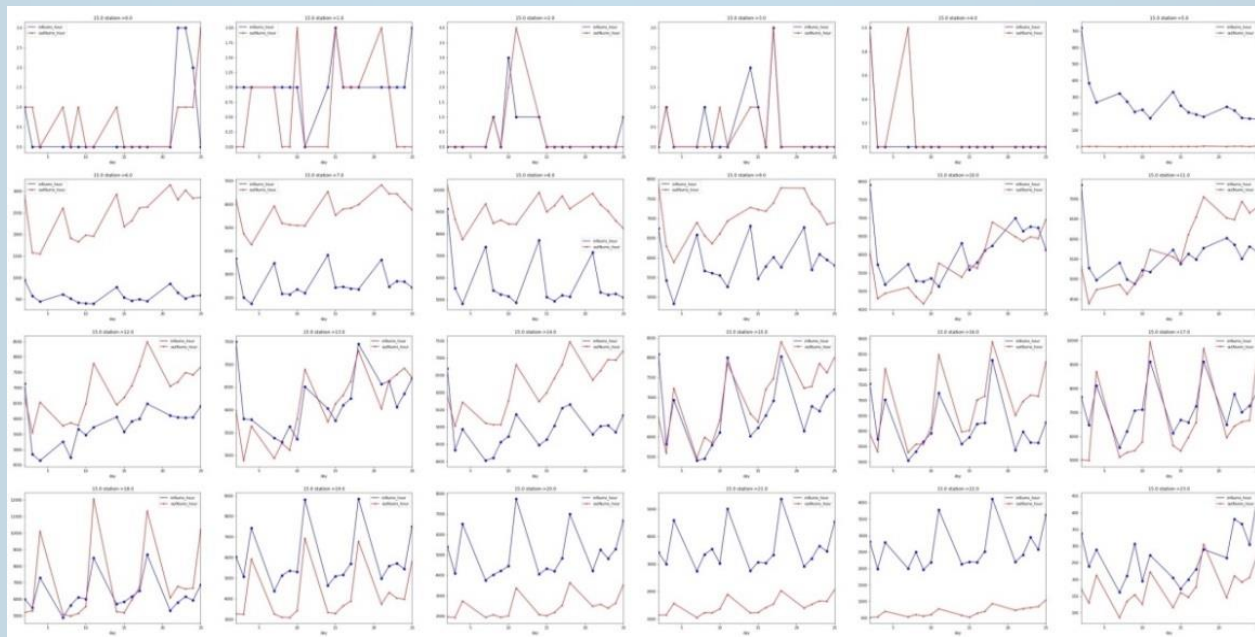


# 可视化数据分析

Visual data analysis

## ● 确定计数单位，生成可视化图形

以人流量最大的15号站点为例，图为工作日出入地铁站点人流量情况。出站入站人流量均比较大，形成许多人流量小高峰；几个相邻时间段出入站人流量呈现相似性。

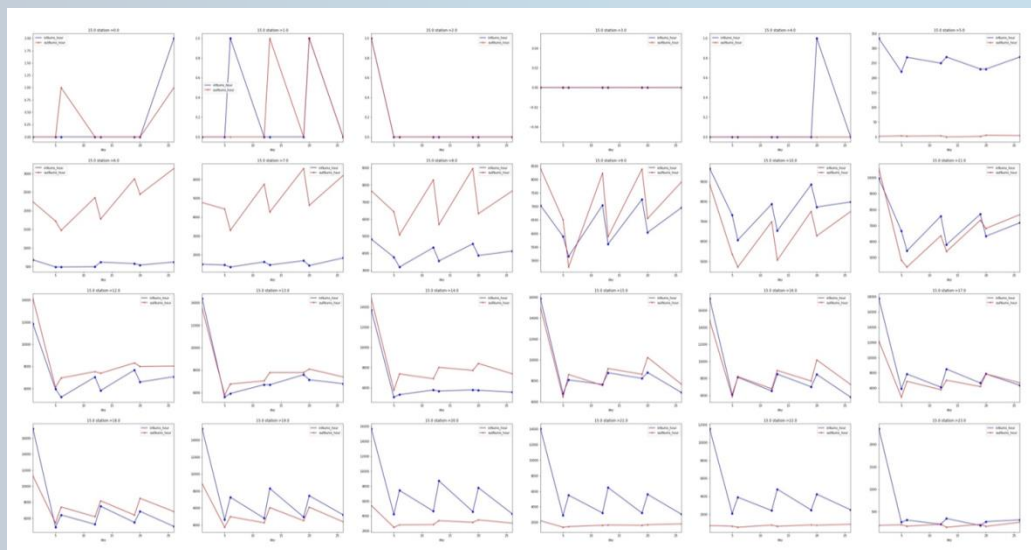


工作日出入15号地铁站点人流量情况

# 可视化数据分析

Visual data analysis

## ● 确定计数单位，生成可视化图形



周末出入15号地铁站点人流量情况

图中可见，周末出站入站人流量均呈锯齿状，形成周末人流量小高峰。数据显示，早5-6点入站人数更多，早6-9点出站人数更多，晚8-11点入站更多。1月5日前出入站人数急剧上升，晚7-12点形成入站高峰，20号后早4-6点入站人数剧增。

# 可视化数据分析

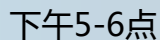
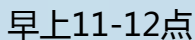
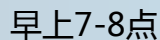
Visual data analysis

## ● 融合多源数据的可视化分析

根据本文4.2节数据探查结果得知，15号站点是全网人流量最大的站点；查询给定源文件Metro\_roadMap.csv提供的路网地图得知，15号站点位于最繁忙的1号线，是1号线和4号线的换乘站；融合杭州地铁营运关系图构建地铁站点对应地理位置及站点名称文件Station\_ID.xlsx，可以确定15号地铁站点为杭州市火车东站。杭州东站火车站既有高铁，也有普速火车，客流量巨大。特别是周末和节假日，早上5-6赶火车出杭的人们，选择地铁作为主要交通工具，形成进站高峰；早上6-9点乘坐火车到杭的人们，或旅游或探亲，地铁作为首选交通工具，形成出站高峰。同样道理，完成白天的行程，晚上8-11点乘坐火车离开杭州，地铁也是重要的交通工具。更重要的是，从6.1节图中可以明显地看到，除了元旦期间显著的人流量高峰之外，随着日期的推移，出入站人流量均呈现缓慢上升的趋势。本文推测，随着春节临近，杭州地铁15号站点出入站人流量仍然有继续上升的趋势。

## Visual data analysis

本文继续选取地理位置和功能定位与15号站点有一定相似性的7号站点进行可视化分析。同在地铁1号线上的7号站点为城战，位于杭州火车站。杭州火车站列车数量少且大部分为普速列车。虽然城战客流量远小于杭州东站，但其客流量仍为全网第四。如图所示，7号站点工作日早上7-8点进出站均有明显的早高峰；周末返杭人数明显增多，并且随着春节来近，返杭人数急速增长，甚至有翻倍的趋势；下午5-6点入站人流量呈现明显的晚高峰，并且此时段出站客流量随日期的推移仍呈现明显的增长趋势。这在一定程度上印证了上述推测的有效性。



# 可视化数据分析

Visual data analysis

## ● 商业区可视化数据分析



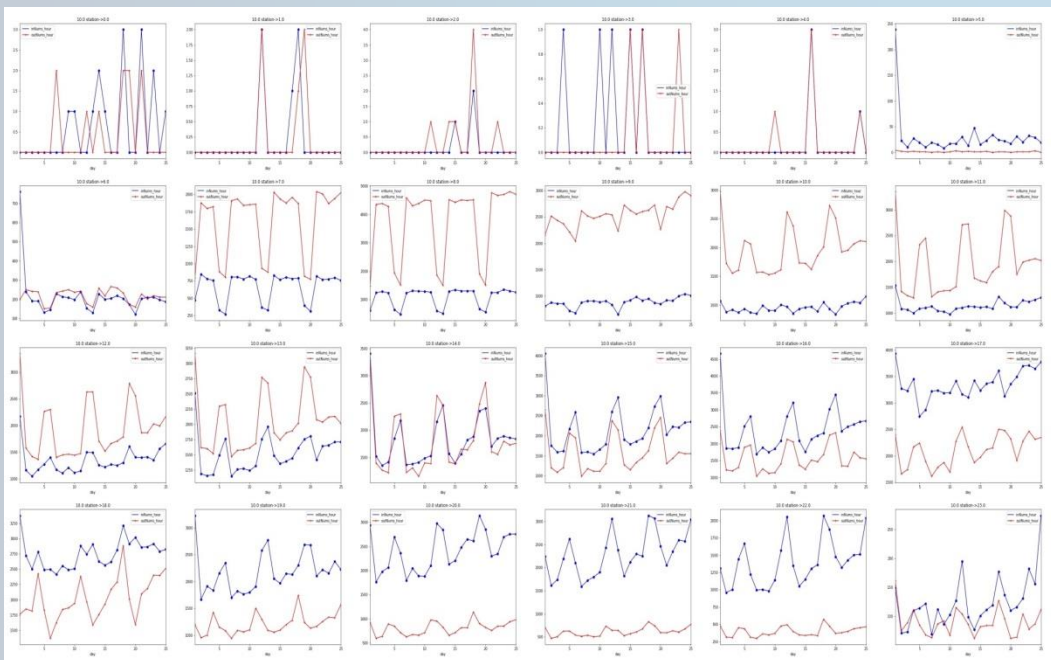
凤起路位于杭州市中心附近，是重要的商业区。本文选取凤起路所在的10号站点和51号站点进行可视化分析。其中10号站点位于最繁忙的1号线，出入站人流量居全网第五；51号站点位于2号线，出入站人流量居于全网的中位数。这样算来，作为源数据中唯一拥有两个ID号的凤起路，其出入站人流量总和稳居全网首位。



# 可视化数据分析

Visual data analysis

## 商业区可视化数据分析



凤起路10号站点26天出入站人流量

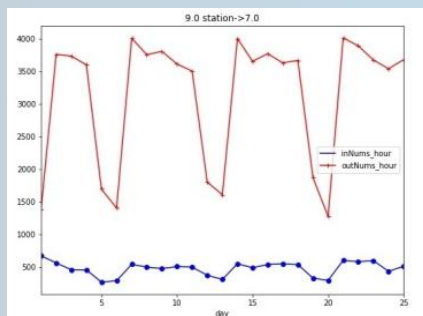
这里以凤起路10号站点26天的出入站人流量为例。从图可以看到，入站和出站人流量均呈现明显周期性，并有锯齿状高峰；入站人流量随日期推移（春节临近）呈锯齿状上升趋势；上午出站人数更多，下午入站人数更多，晚上7点左右有入站小高峰，体现了商业区的特点，并且在购物高峰时段，几个相邻时间段出入站人流量规律相似。同为凤起路的51号站点出入站人流量比10号站点少，但在大多数时间段呈现出与10号站点相似的规律性。

# 可视化数据分析

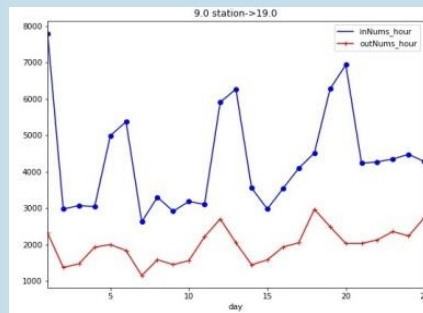
Visual data analysis

## 商业区可视化数据分析

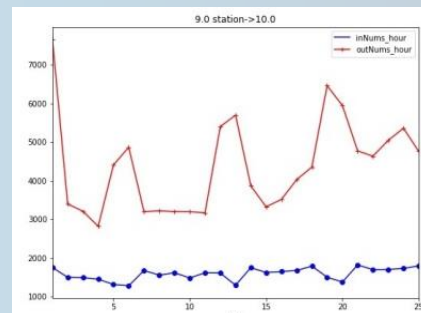
接着，本文选取地理位置和功能定位与凤起路站点较相似的9号站点进行可视化分析。同在地铁1号线上，与凤起路站相邻的9号站点龙翔桥是全网客流量第二大站，杭州市中心的商业区，周围有湖滨银泰等多个大型商场，毗邻西湖景区，相当于上海的南京路和外滩。本文选取几个关键时段进行可视化分析。如图所示，早上7-8点，龙翔桥站点出站人数远多于进站人数，明显的早高峰，尤其在工作日期间；早上9点半左右，正是大型商场开门的时间，周末的龙翔桥站点出站客流量显著增加，应该是居民来此购物；晚上7-8点，该站点出站人数远多于进站人数，出现了比较明显的晚高峰，无论是工作日还是周末，此时段的客流量高峰都与商业区的购物高峰时段呈现出明显的关联性。



早上7-8点



早上10-11点



晚上7-8点

# 可视化数据分析

Visual data analysis

## ● 商业区可视化数据分析

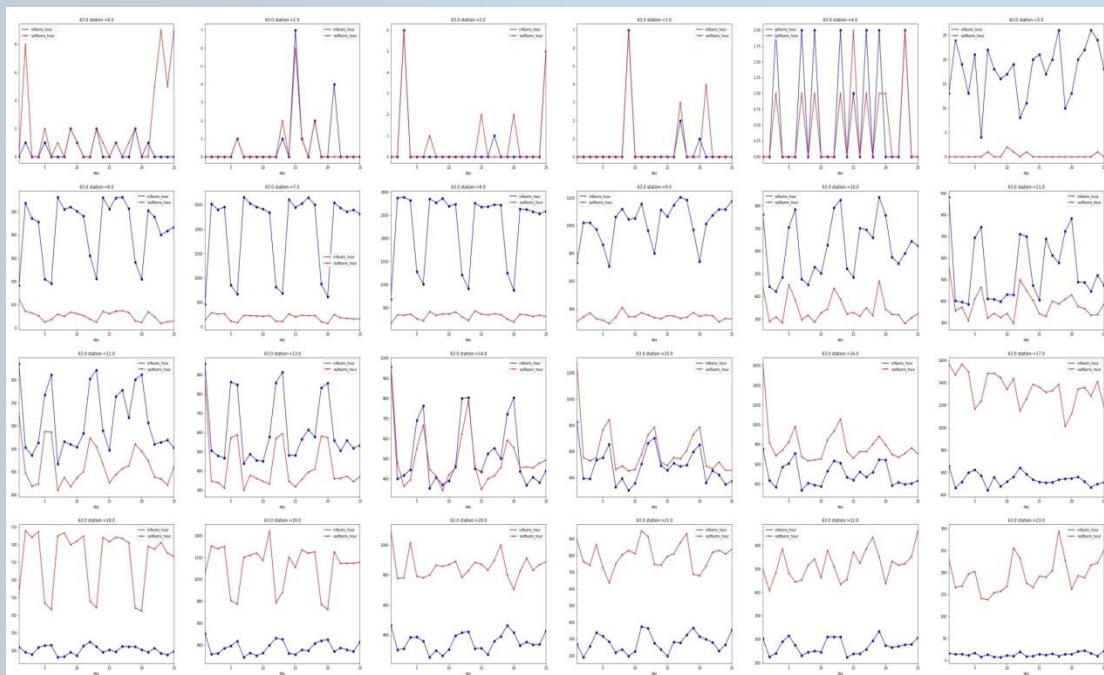
由此可见，临近商业区的地铁站点在刷卡数据上呈现出如下特点：

- 1) 工作日早上7-8点来此站点的人数较多，可能是来商业区工作的人群；
- 2) 晚上7-8点到达此站点的人数较多，可能是休闲购物的人们；
- 3) 周末客流量急剧增加，可能与周末休闲购物等生活需求有关；
- 4) 随着日期的推移（春节临近），出入站客流量呈现逐步上升的趋势，可能与临近春节人们购买年货有关。

# 可视化数据分析

Visual data analysis

## ● 居住区可视化数据分析



金家渡站26天24小时出入站人流量情况

全网中位数的63号站点在客流量较少的2号线上，居于2号线的一端，在地理位置上称金家渡站。从图可以看出，该站点的出入站人流量呈现明显特点。首先，该站点人流量以7天为单位呈现明显的周期性，工作日的出入站人流量处于高峰期，周末的人流量为低谷期。在工作日，上午5-10点入站人流量显著增大，说明人们出行的首选交通工具为地铁；下午5-12点出站人流量明显更大，说明结束了一天的辛劳，人们乘坐地铁回到这里。这些特征使得金家渡站呈现出距离市区较远的居住区特点。此外，周末入站人流量呈现锯齿状，形成周日入站小低谷，说明是居住区，也有周六加班，但周日上班的人数较少。晚上入站人数稀少，说明鲜有夜晚出门的人流，凌晨之后偶有出站人流，可能是加班回家的人们。

# 可视化数据分析

Visual data analysis

## ● 居住区可视化数据分析

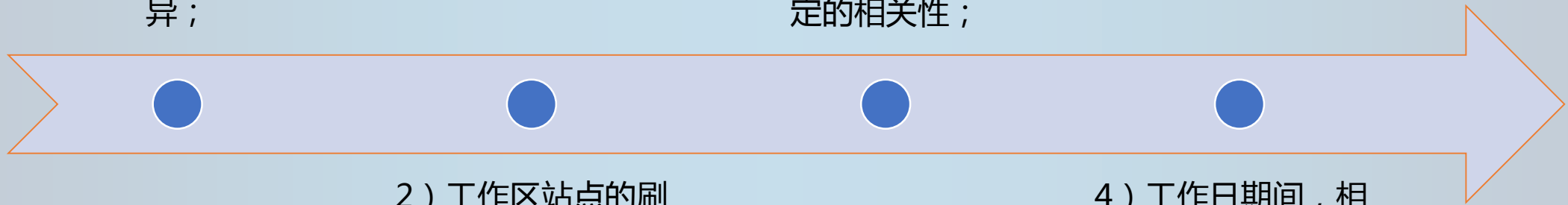
由此可见，居民区的地铁站点在刷卡数据上呈现出如下特点：

1) 工作日和周末的  
出站和进站人流量情  
况差异明显，规律迥  
异；

3) 工作日期间，8小  
时之内的相邻时间段  
和8小时之外的相邻  
时间段各自呈现出一  
定的相关性；

2) 工作区站点的刷  
卡数据呈现出明显的  
“早出晚归”特点；

4) 工作日期间，相  
邻日期的出入站人流  
量呈现出明显相似性。







# PART EIGHT

## 可解释性数据分析及验证

Analysis and verification of interpretable data

# 可解释性数据分析及验证

Analysis and verification of interpretable data



对于地理位置有代表性或区域功能性比较单一的站点，本文进行了可视化分析，取得了有效的分析结果。然而，对于地理位置或功能性比较特殊和复杂的站点，仅仅依靠简单的折线图无法达到预定的可视化分析效果。因此，本文融合“杭州地铁网”、“百度”、“知乎”等多种来源的数据，对生成的站点人流量进行综合分析，剖析数据背后的“秘密”。

# 可解释性数据分析及验证

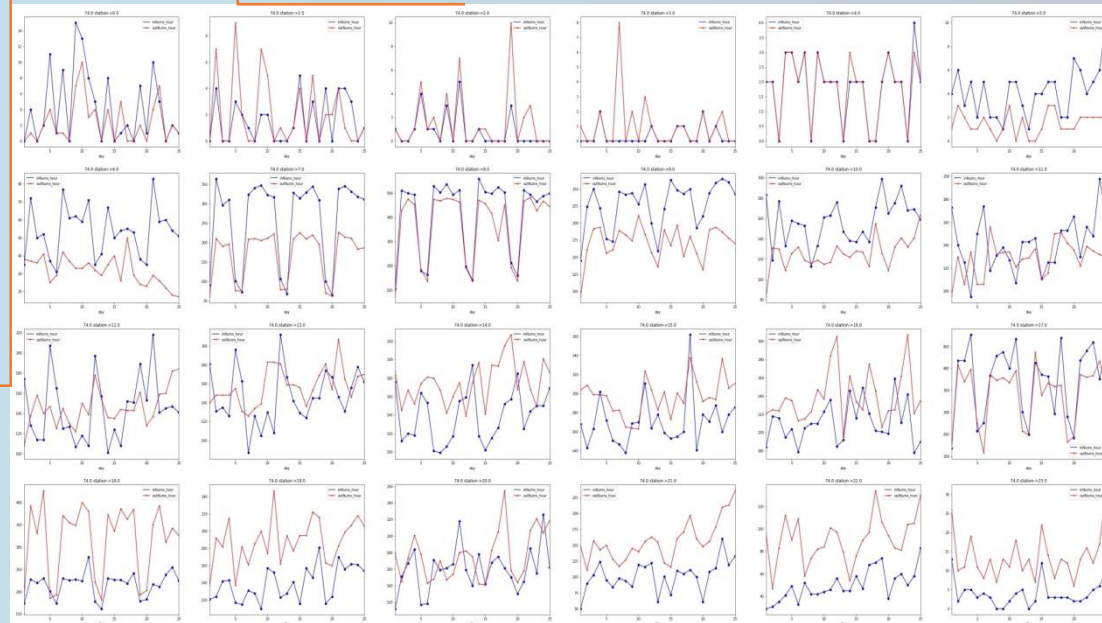
Analysis and verification of interpretable data

## ● 甬江路站人流量数据分析

人流量最小的74号站点为甬江路站。它位于人流量最少的4号线，2018年1月通车。此站点离市中心并不远，只有5站路，位置也并不偏僻，但是客流量却是全网最少的。从图可以看出，大多数时段出站入站人流量差距不大，上午时段出入站人流量呈现一定周期性；上午入站略高，晚上出站略高，似乎呈现出居民区的迹象。周末入站人流量呈现锯齿状，周日入站进入低谷，可能是住宅区。总的来说，甬江路站早上进站人数多于出站人数，晚上出站人数多于进站人数，属于居民区。本文依据多源数据，希望探索甬江路站人流量最少的可能原因。



甬江路站26天24小时出入站人流量情况





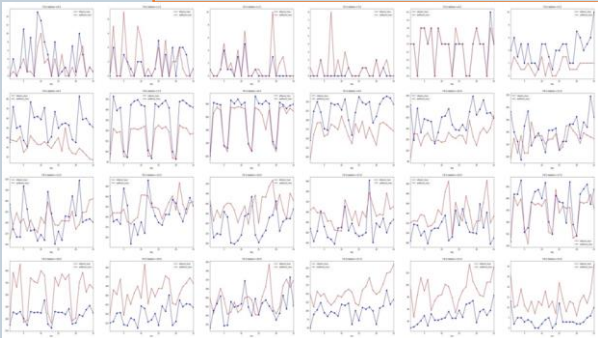
# 可解释性数据分析及验证

Analysis and verification of interpretable data

## ● 甬江路站人流量数据分析



地铁为该地区的居民提供了更为便捷的出行方式。位于市中心，交通便利，基础设施齐全的地区应该吸引更多的居民前来购房。总的来说，越靠近市中心的地铁房，价格就越贵，虽然这并不绝对，但是甬江路站的确是杭州市房价最贵的地铁站<sup>[6]</sup>。甬江路站位于钱江新城豪宅区，方圆500米内，一大波江景豪宅，如信达滨江壹品、候潮府、望江府和蓝色钱江等，房价高居杭州市地铁房榜首。看来，甬江路站点人流量小的原因可能是精英人士出入主要交通工具不以地铁为主，早上6-8点入站人员可能是服务人员和社区周边工作人员。



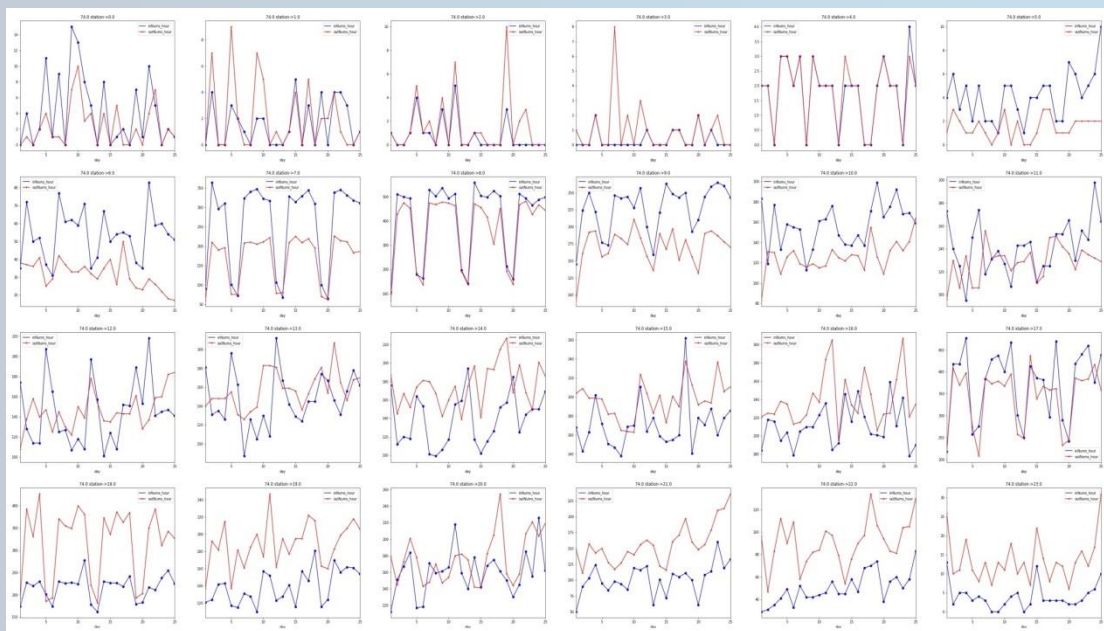
# 可解释性数据分析及验证

Analysis and verification of interpretable data

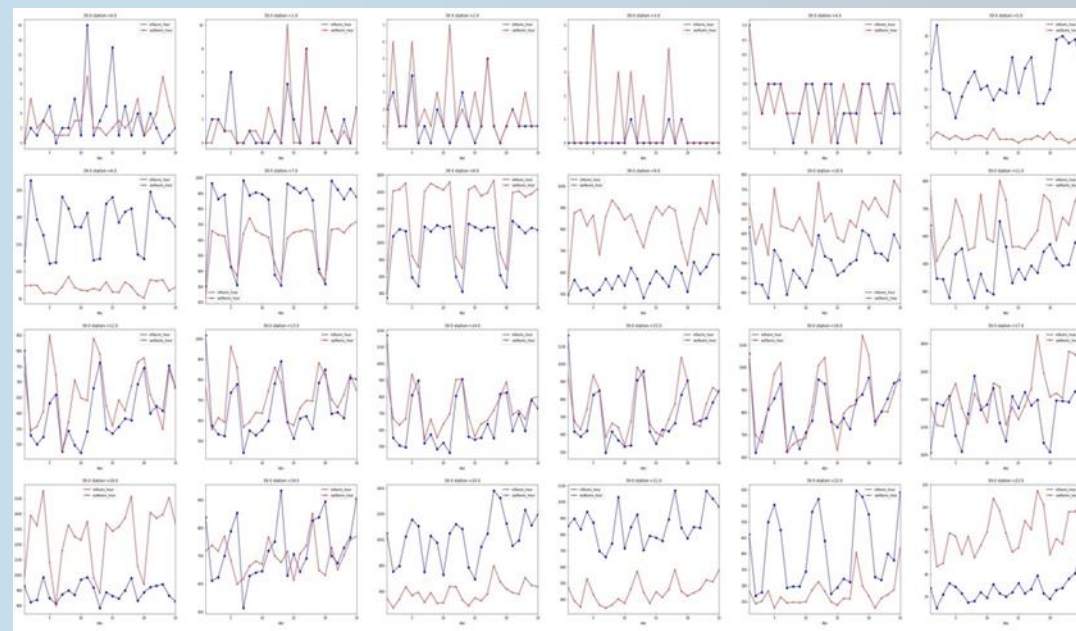
## ● 人民广场站人流量数据分析

出入站人流量位于全网中位数的39号站点和63号站点，同属于人流量较少的2号线，分别居于2号线的两端。

对比图34和图32的出入站人流量情况，这两个站点呈现出较大的差异性。



甬江路站26天24小时出入站人流量情况



人民广场站26天24小时出入站人流量情况

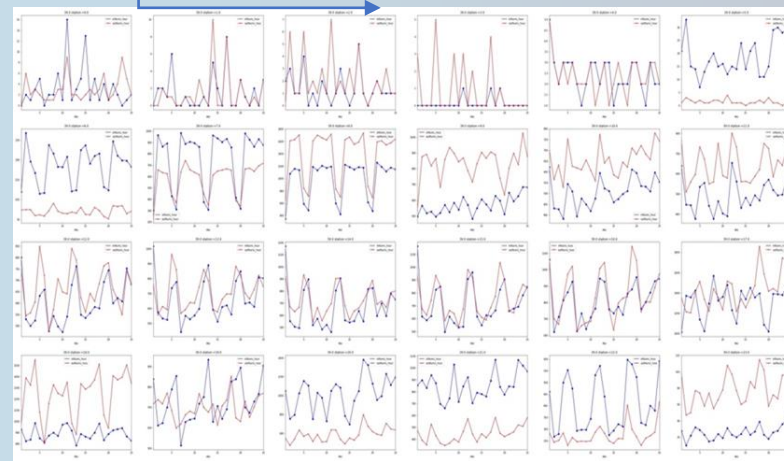


# 可解释性数据分析及验证

Analysis and verification of interpretable data

## ● 人民广场站人流量数据分析

39号站点为人民广场站，位于钱塘江南岸的萧山区中心位置。如图所示，人民广场站工作日早高峰明显，上午时段出入站人流量呈现一定周期性，并有锯齿状人流量小高峰。锯齿状的形成应该与工作日、周末人流量差距较大有关。早上5-8点入站人流量更大，早上8-12点出站人流量更大，出入站分界线明显，并有锯齿状人流量小高峰。晚高峰与早高峰出现相似的情况。周末早高峰不明显，周六周日客流差别大。在工作日，晚上8点以后入站人数明显增大，可以看出本站是某小型区域的中心站点，以生活区为主。该站点出入站情况在1-5号，20号以后无异常变化，可能居住的是杭州老住户。早上5-8时进站人流量大，8-12时出站人流量大，本站周围具有商业区的部分功能。因此，人民广场站应该是兼有生活区和商业区部分特点的区域性中心站点，属于居民区。



# 可解释性数据分析及验证

Analysis and verification of interpretable data

## ● 人民广场站人流量数据分析

综上所述，杭州地铁站点在刷卡数据上呈现如下特点：

出站数据和入站数据均呈现周期性。以7天为一周期，工作日和周末的人流量差别很大，分别呈现不同的周期特点

工作日出站入站人流量大小区别明显。其中住宅区和生活区附近的地铁站人流量呈现出“早入晚出”的特点，工作区附近的地铁站人流量呈现出“早出晚归”的特点，而商业区附近的地铁站临近春节的人流量有上升趋势

周末数据与工作日数据应该分别进行数据分析，源数据中只有1月1日元旦这一天为假期，且呈现明显不同的模式，在进一步数据预测时可以尝试不考虑这一天的数据

工作日的上午、晚上几个相邻时间段出入站人流量呈现一定的相似性，工作日期间相邻日期的出入站人流量具有一定的关联性

# 可解释性数据分析及验证

Analysis and verification of interpretable data

## ● 地铁线路数据分析

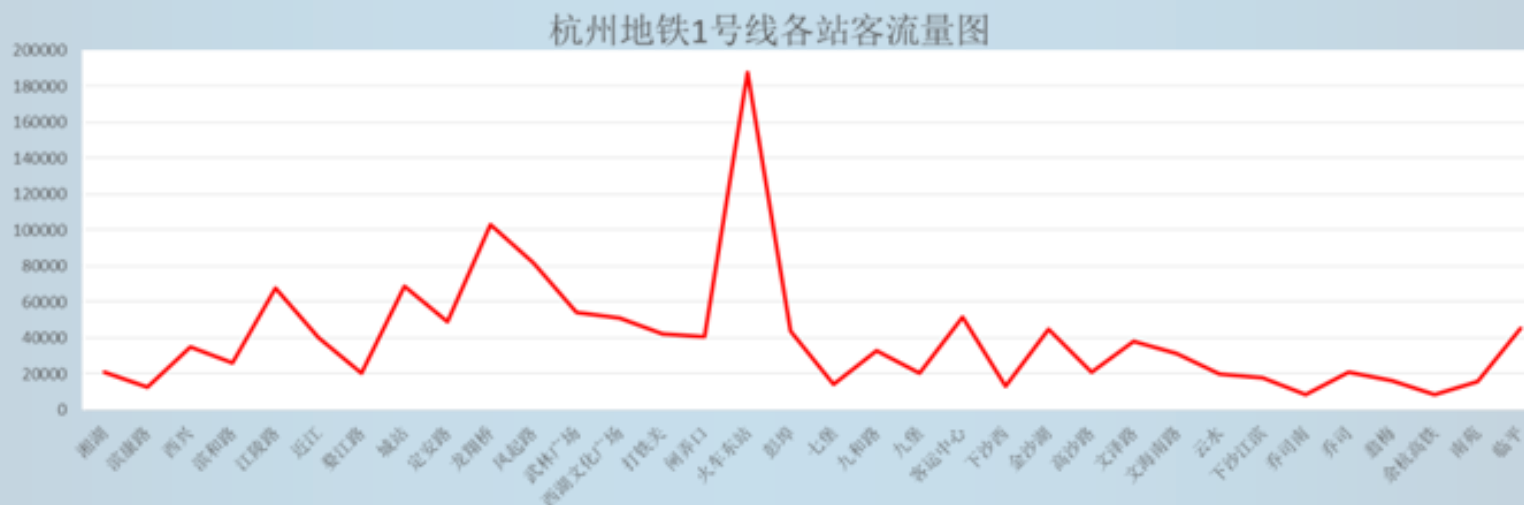
本节累计杭州每条地铁线路各站点客流量数据，结合杭州地理位置和行政区划情况，对每天地铁线路情况综合分析。



# 可解释性数据分析及验证

Analysis and verification of interpretable data

## ● 地铁线路数据分析



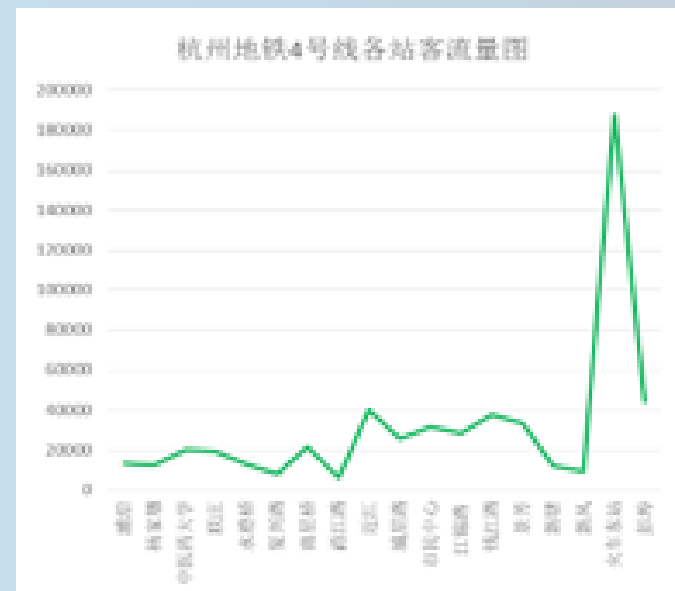
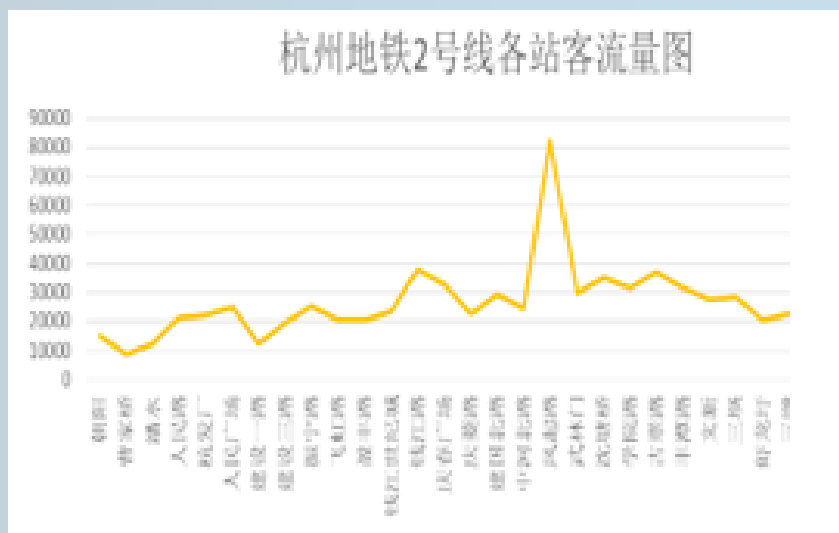
杭州位于平原和丘陵的交界处，钱塘江、西湖景区、西溪湿地将城市分割开来，城市交通由此受阻。城市分为许多组团，每个组团都有自己的居民区和商业区。

地铁1号线串联了城市的多个组团，如主城区，滨江，下沙，临平等。从图可以看出，客流量曲线有许多极大值点，说明杭州的区域性中心有多个，分别对应每个组团或卫星城的中心，但是它们的发展状况仍远不及主城区。

## 可解释性数据分析及验证

## Analysis and verification of interpretable data

## ● 地铁线路数据分析



2号线和4号线作为客流量较少的两条地铁线路，相比附近的非换乘站，换乘站（凤起路，近江，火车东站等）拥有较大的客流量，明显高于非换乘站。可以说，杭州地铁换乘站的设置比较合理。





# PART **NINE**

## 数据分析结果

Data analysis results



# 数据分析结果

Data analysis results



地铁交通系统是一个复杂的系统，涉及人、车辆、道路和环境的相互作用。其中车辆和道路基础设施状况是影响交通流量的重要因素，很大程度上决定了地铁客流量具有非线性和不确定性。然而，人们乘坐地铁出行的行为总体上具有规律性，在一定程度上决定了地铁客流量具有时空相似性，主要表现在以下几个方面。

# 数据分析结果

Data analysis results

## ● 动态性

同一路段在不同时刻的交通流量分布不同<sup>[7]</sup>，其原因主要有：

经济社会的快速发展促使人们的需求和出行结构向多元化发展，如本文甬江路站附近豪宅林立，许多居民出行的首选交通工具并不一定是地铁；

恶劣天气情况的频发，使得人们的出行方式和出行频率也随之变化，例如雨天的道路行驶状况和晴天可能有区别。本文也收集整理了2019年1月份每天的天气情况，粗略探索杭州天气情况与地铁刷卡数据的潜在关系。由于训练样本有限，天气变化不够显著，目前未形成具有显著意义的结论，故没有展示评价结果；

季节的更替变化，冬季白昼短，夏季白昼长，很多单位会相应调整工作时间；

节假日的影响，例如元旦、春节等，本文提供的杭州地铁打卡数据中，节假日和工作日的交通流量分布规律有着显著差别；

交通事故或临时性的交通管制等。

实际上，在相同时间的相邻路段，交通流量也不尽相同。总之，多种影响因子的变化会造成地铁客流量的不断变化。

# 数据分析结果

Data analysis results

## ● 时间相似性

通过对比分析2019年1月份26天杭州81个地铁站点的出入站人流量数据，可以发现人们出行的规律性呈现出以周为单位的周期性，进而地铁客流量也具有周相似性。通过数据对比发现，五个工作日的交通流量异于周末和节假日。为了验证交通流量的时间相似性，本文整理了近一个月的地铁人流量数据，数据采集的时间间隔为60分钟，经过数据预处理绘制了2019年1月1日至28日共81个站点的出入站人流量折线图。从上述可视化数据分析结果看出：交通流量具有时间相似性。在工作日通勤高峰期，交通流量达到峰值，部分客流量较大的站点可能会出现交通拥堵现象。而周六周日的交通流量呈现出先上升后下降的趋势，明显不同于工作日状况。

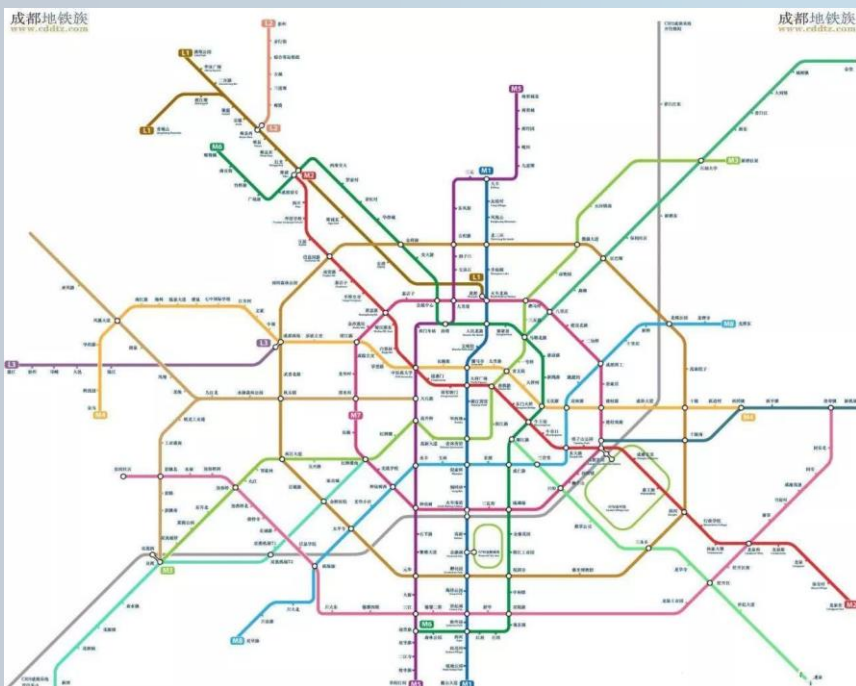
日	一	二	三	四	五	六
27 十三	28 十四	29 十五	30 二九	31 二九 年终盛典	1 元旦 休	2 十九 休
3 二十 休	4 廿一	5 小寒	6 廿三	7 廿四	8 三九	9 廿六
10 廿七	11 廿八	12 廿九	13 腊月	14 初二	15 初三	16 初四
17 四九	18 初六	19 初七	20 大寒	21 初九	22 初十	23 十一
24 十二	25 十三	26 五九	27 十五	28 十六	29 十七	30 十八
31 十九	1 二十	2 廿一	3 立春	4 雨水	5 惊蛰	6 廿五



# 数据分析结果

Data analysis results

## ● 空间相似性



杭州地铁系统是一个庞大并交互的繁杂系统，通常显示出非线性、非稳定性等特点<sup>[8]</sup>。由于城市道路网具有连通性，不同路段之间存在拓扑关系，这些路段的交通量也会彼此影响，表现为交通流量的空间相关性。且距离愈近，彼此作用愈强；距离愈远，彼此作用愈弱。换乘站点也会影响路段间的相互作用，故路网的连通性决定了交通流量具有空间相关性<sup>[9]</sup>。若想获取较高的预测精度，需要充分考虑路网的拓扑关系，服务于杭州地铁交通控制和疏导。



# 数据分析结果

Data analysis results

## ● 地铁房价空间增值性

地铁交通给沿线居民带来出行便利的同时，也带动了沿线经济的发展，尤其突出的是对沿线房地产价格的增值效应。总的来说，越靠近市中心的地铁房，价格就越贵，但这也并不绝对。具体到每条线路，它们并非平滑地向两端延伸递减<sup>[10]</sup>。



图为北京13号线地铁各站点房价

# 数据分析结果

Data analysis results

## ● 地铁房价空间增值性



以地铁1号线下沙为例，房价最高的区域为龙翔桥至西湖文化广场一带，价格在49486-51469元/平方米；随后滨江的江陵路（41399元/平方米）、江干的彭埠站（40492元/平方米）也出现了一个高点；全线最便宜的房价出现在下沙西，为20711元/平方米，随后再往东，房价又出现了一个小高峰。房屋均价高于50000的有7个站点，除了最贵的甬江路站（62500元/平方米）位于钱江新城，其余都散布在武林门周围区域，如武林门（59610元/平方米）、龙翔桥（50346元/平方米）等。

杭州地铁1、2、4号线中，房价最贵的线路是地铁4号线，均价42177元/平方米，主要因为4号线沿线经过一大批钱江新城豪宅；平均房价最低的是地铁1号线，为32353元/平方米，主要受临平、下沙一带房价“拖后腿”。

# 数据分析结果

Data analysis results

## ● 地铁房价空间增值性



由此可以看出，房价高和客流量大的成因可能是相同的，即位于市中心，交通便利，基础设施齐全。对于房价较高的几个站点均是如此。但对于房价最高的甬江路来说，客流量反而最低。客流量小的原因可能是精英人士出行的交通工具不以地铁为主。

对于杭州来说，地铁线路客流量和地铁周边房价恰好呈现反相关关系。可能和杭州地铁的线路设置有关。综上所述，地铁客流量和地铁周边房价的关系很复杂，需要进一步的分析和学习。



# PART TEN

可能的解决方案

Possible solutions









# 可能的解决方案

Possible solutions

01

加强数据收集



加强杭州地铁大数据的收集、处理和利用工作。本文数据分析发现，杭州地铁客流量高峰期为每天上午的6点-9点、下午的5点-9点，结合实际情况便可明白引发这种现象的主要原因是“上下班高峰期”。为了解决这一问题，可以采用大数据技术对杭州地铁以前的数据进行分析整理，制定出了一套科学的A方案。每当高峰期临近时，杭州地铁管理部门可以增加繁忙线路上车辆数量，适当调整地铁班次，启动进站口刷卡处的报警装置。当站点可承受客流量临近上限，报警装置示警，启动紧急处理方案A。这样，车辆班次的增加、等待时间的缩短对降低地铁运营负荷具有积极作用。



# 可能的解决方案

Possible solutions

02

提高运营效率



利用大数据相关技术完善地铁交通管理，改善地铁交通运行的不确定性与不平衡性，提高地铁运营效率。目前地铁交通客流量会随着时间、季节、事件的变化而变化，其中不确定性问题主要表现在节假日或大型活动举办期间。这时地铁客流量会集中在某一区域的某一时间点，致使该时间段的客流量大幅增加，给地铁运营带来较大压力；不平衡性问题主要表现为商业区、换乘车站等人流量较大的区域也是地铁客流量较大的站点，而郊区、“城中村”等人流量较小的区域也是地铁客流量较少的地区。可以通过提前部署，合理调配等方式尽量缓解暂时拥堵现象。



# 可能的解决方案

Possible solutions

03

提高服务质量



利用智能设备规避地铁风险，提高地铁交通客运服务质量。随着地铁客流量增加，客运服务工作量愈发庞大，服务人员的工作压力骤然上升。为了保证乘客人身安全，应该安排工作人员站在屏蔽门前维护秩序，避免乘客因为拥挤、踩踏发生意外，尤其是杭州西湖等著名景点附近的地铁站。这种情况下，可以利用大数据技术开发智能检测系统<sup>[12]</sup>，将系统安装于屏蔽门前的黄线内，当红外线感应器感应到人体红外线时会发出警告，提醒乘客后退，从而达到节约人力资源、减少工作量，提高地铁客运服务质量的目。





# PART ELEVEN

## 总结与展望

Summary and prospect



# 总结与展望

Summary and prospect

本文整理了2019年1月份的杭州地铁人流量数据，完成数据预处理并绘制了2019年1月1日至28日共81个站点的出入站人流量数据折线图，对代表性地铁站点、重要时间段、城市不同功能区的乘客流量规律进行可视化研究，挖掘杭州地铁乘客出行规律，验证了杭州地铁客流量具有时空相似性。这对规避交通堵塞，部署站点安保，实现智慧出行具有显著意义和实用价值，对研究地铁交通与沿线房地产价格的关系具一定借鉴意义。

杭州地铁交通系统是动态的庞大系统，影响杭州地铁刷卡数量的因素繁杂而多样。本文仅仅完成了粗粒度的出入站人流量可视化分析。为了更详细、更全面地探究杭州地铁人流量规律，本文下一步的工作将研究XGBoost<sup>[13]</sup>、lightGBW<sup>[14]</sup>算法在地铁人流量预测模型中的应用，对本文的源数据进行细粒度的数据分析、挖掘和预测，为大数据助力智慧出行提供更详实的解决方案。







# PART **TWELVE**

参考文献

Reference

# 参考文献

Reference

- [1] 杭州地铁, 百度百科. <https://baike.baidu.com/item/杭州地铁/9670206?fr=aladdin> [DB/OL], 2020.01.05
- [2] 杭州地铁. <http://www.hzmetro.com/> [DB/OL], 2020.01.04
- [3] 杭州 : 2022 年亚运会前将新建 400 公里轨道交通 [EB/OL], 中国政府网 [http://www.gov.cn/xinwen/2018-12/22/content\\_5351166.htm?\\_zbs\\_baidu\\_bk](http://www.gov.cn/xinwen/2018-12/22/content_5351166.htm?_zbs_baidu_bk), 2018.12.22.
- [4] 陈白磊. 杭州市轨道交通客流预测中的一些问题及对策, 城市轨道交通研究[J], 2003 : p81-84
- [5] 天池大数据众智平台-阿里云天池, <https://tianchi.aliyun.com/home/> [DB/OL], 2019.05.20
- [6] 杭州地铁房价 [EB/OL] <https://hz.focus.cn/zixun/ec5aaa1fa47b6407.html>
- [7] 沈丽萍, 马莹, 高世廉. 城市轨道交通客流分析, 2007.05, 第5卷3期: p14-19.
- [8] 蔡后琮. 无锡、上海、杭州轨道交通考察印象浅析, 企业管理, DOI : 10.16661/j.cnki.1672-3791.2018.23.136, 2018 NO.23:136-137.



# 参考文献

Reference

[9]刘剑锋,罗铭,马毅林,王静,孙福亮,陈锋.北京轨道交通网络化客流特征分析与启示. 都市快轨交通, 2012 年10 月第25 卷,第5 期: p27-32.

[10]张术.城市轨道交通对房地产价格的空间效应研究-以杭州市地铁1号线为例,浙江大学硕士论文, 2014.04.

[11]朱玮.轨道交通发展对杭州商业空间形态的影响,浙江大学硕士论文, 2016.03.

[12]金昱.基于上海轨道交通刷卡数据的乘客出行模式研究, 都市快轨交通, 2019.06, 第32卷 第3 期:p91-96

[13]范淼,李超.python机器学习及实践, 清华大学出版社, 2016,12.

[14] LightGBW基本原理介绍. [https://blog.csdn.net/qq\\_24519677/article/details/82811215](https://blog.csdn.net/qq_24519677/article/details/82811215), 2020.01.05

