

第 7 周 数据可视化与交互

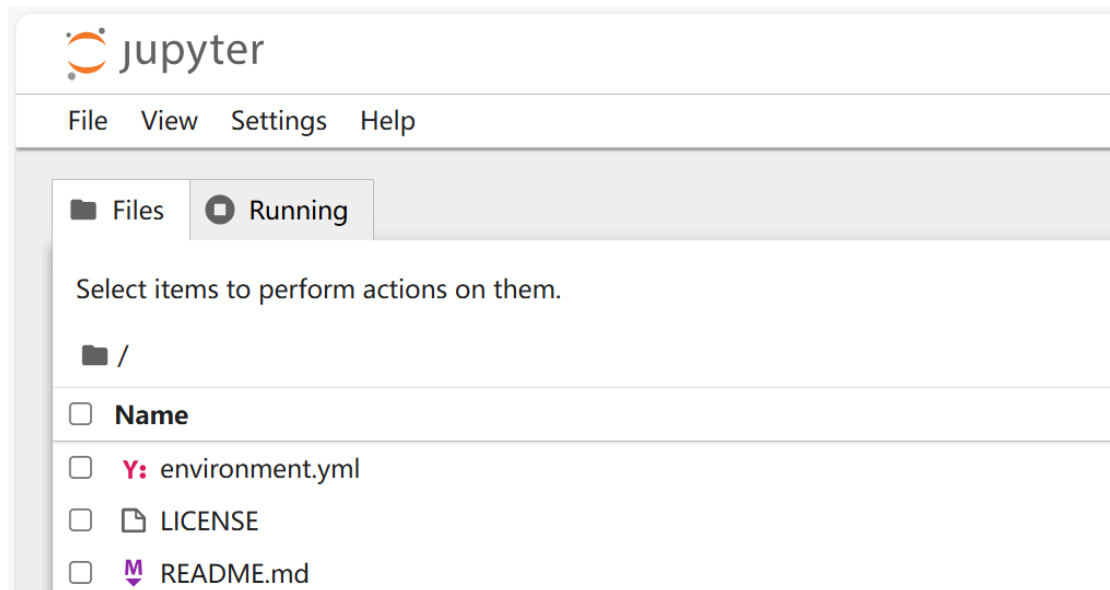
1. 创建 conda 环境

```
! environment.yml
1  name: week07
2  channels:
3    - conda-forge
4  dependencies:
5    - python=3.12
6    - wat-inspector
7    - jupyterlab
```

2. Jupyter notebook

```
[C 2025-04-23 15:51:36.329 ServerApp]
To access the server, open this file in a browser:
file:///C:/Users/cherry/AppData/Roaming/jupyter/runtime/jpserver-19584-open.html
Or copy and paste one of these URLs:
http://localhost:8888/tree?token=5d0dc0d7c88efa2df05dbdde41ec1512ee99fece1aa9
http://127.0.0.1:8888/tree?token=5d0dc0d7c88efa2df05dbdde41ec1512ee99fece1aa9
[I 2025-04-23 15:51:36.530 ServerApp] Skipped non-installed server(s): bash-language-
nodejs, javascript-typescript-langserver, jedi-language-server, julia-language-server,
r-languageserver, sql-language-server, texlab, typescript-language-server, unified-la
server-bin, vscode-html-languageserver-bin, vscode-json-languageserver-bin, yaml-langu
0.01s - Debugger warning: It seems that frozen modules are being used, which may
0.00s - make the debugger miss breakpoints. Please pass -Xfrozen_modules=off
```

● 后端

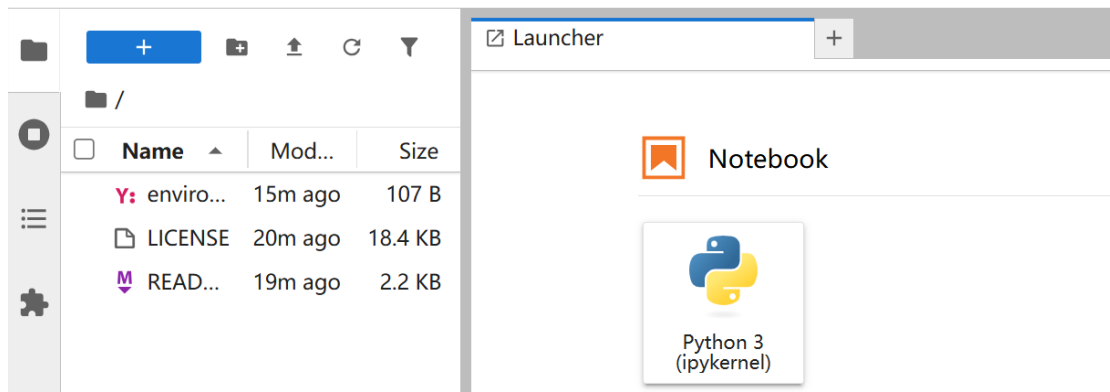


● 前端

3. Jupyter lab

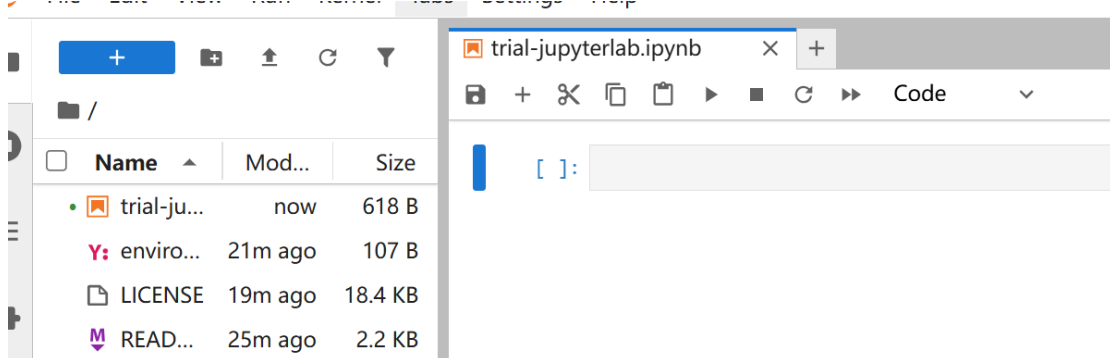
```
To access the server, open this file in a browser:
file:///C:/Users/cherry/AppData/Roaming/jupyter/runtime/jpserver-8796-open.html
Or copy and paste one of these URLs:
http://localhost:8888/lab?token=af8122da4650dc1a6bf46406bfabfc9b1a888cfb244f61a7
http://127.0.0.1:8888/lab?token=af8122da4650dc1a6bf46406bfabfc9b1a888cfb244f61a7
```

● 后端

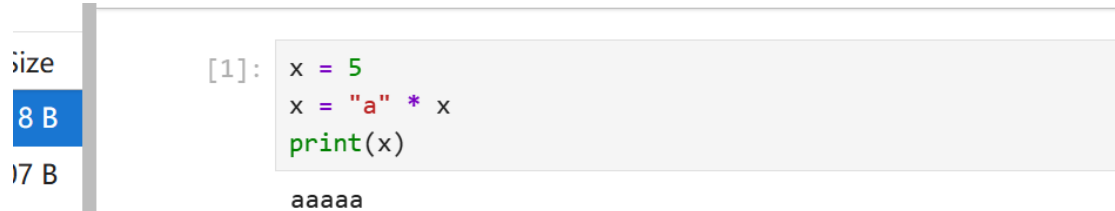


● 前端

4. 新建一个 Notebook，命名为 trial-jupyterlab.ipynb，并掌握相应功能



1. 在单元格中写命令，按 shift+enter 运行命令，只按 enter 是换行



2. 单元格最后一句是否为表达式



```
[5]: x = 5
      x = "a" * x
      print(x)
      x = 9
      x * 3
```

是表达式

aaaaa

```
[5]: 27
```

有输出

```
[7]: a = [4, 2, 9]
      a
```

```
[7]: [4, 2, 9]
```

```
[8]: a.pop()
```

```
[8]: 9
```

- 表达式会有输出

```
[9]: a
```

```
[9]: [4, 2]
```

```
[10]: print(a)
```

```
[4, 2]
```

- Print () 是表达式，但是 print () 的返回值是 none，所以才没有编号

```
[13]: _8
```

```
[13]: 9
```

```
[14]: _8 * 9
```

```
[14]: 81
```

- 引用序号为 8 的输出，_8 相当于变量

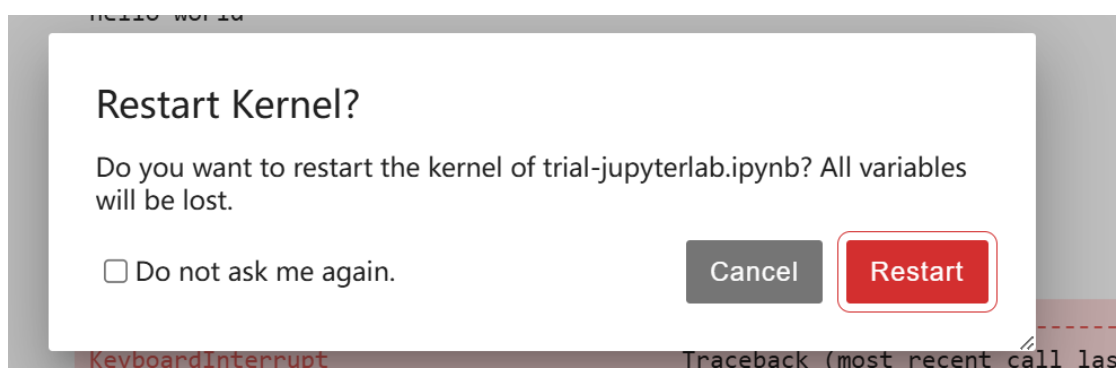
[*] `import time`
`while True:` 无限循环
`print('hello world')`
`time.sleep(5)` 每隔 5 秒钟 print 一次

hello world
 hello world
 hello world

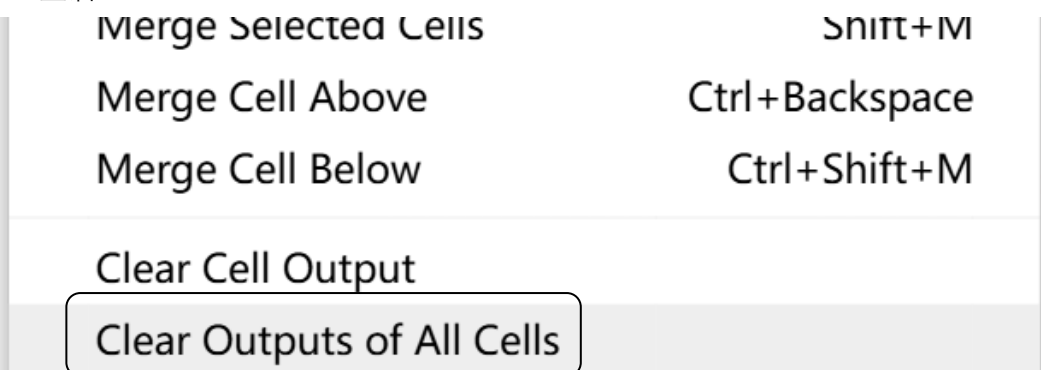
```

KeyboardInterrupt                                Traceback (most recent call last)
Cell In[15], line 4
      2 while True:
      3     print('hello world')
----> 4     time.sleep(5)
KeyboardInterrupt:
  
```

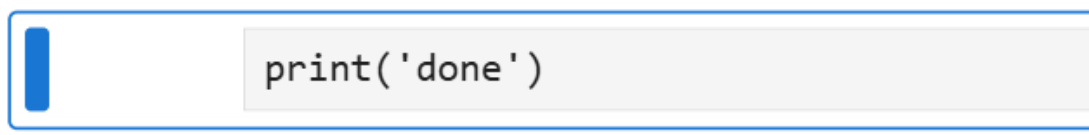
- 按 ii 打断进程，否则之后写的代码也都无法运行



- 00 重启



- 先清空所有输出，再 shift+enter 重新运行



- (m) Markdown 模式，前面没有方括号

```
[ ]: x= 3  
print(x ** 2)
```

- (y)Python 模式，前面有方括号

这是一个 Markdown 示例文档

一、标题分级（共六级）

一级标题

二级标题

三级标题

四级标题

五级标题

六级标题

- Markdown 语法

这是一个 Markdown 示例文档

一、标题分级（共六级）

一级标题

二级标题

三级标题

四级标题

五级标题

六级标题

- 运行结果

二、列表语法

1. 无序列表（使用 ` - ` 或 ` * `）

- 列表项1
- 列表项2
 - 子列表项1
 - 子列表项2
- * 列表项3

2. 有序列表（使用数字加点）

1. 第一项
2. 第二项
3. 第三项

1. 无序列表（使用 - 或 *）

- 列表项1
- 列表项2
 - 子列表项1
 - 子列表项2
- 列表项3

2. 有序列表（使用数字加点）

1. 第一项
2. 第二项
3. 第三项

2. 代码块（支持语法高亮，需指定语言）

```
python
# Python 示例代码
def calculate_sum(a, b):
    """计算两数之和"""
    return a + b

result = calculate_sum(10, 20)
print(f"结果: {result}")
```

Jupyter的发展历程如下：

- ****IPython的诞生（2001年）****：Fernando Pérez在2001年发起了IPython项目，旨在创建一个强大的交互式计算环境，最初主要是增强Python的交互式解释器功能，方便开发者通过命令行交互方式、自动补全等功能，让科学家们能更高效地进行实验。
- ****IPython功能扩展与Notebook出现（2007 - 2011年）****：2007年，IPython系统，用于将文本、计算和可视化相结合。2011年，第一个版本的IPython Notebook可以在单个文档中编写代码、文本和多媒体内容，这一功能极大地便利了科学家展示结果。同时，IPython也逐渐开始支持其他编程语言，如R、Julia等，其功能平台扩展。

- 大模型的输出结果是 markdown 格式

示例网页



欢迎来到示例网页

- [主页](#)
- [关于](#)
- [联系我们](#)

主要内容

这是一个示例网页，展示了基本的 HTML 结构和 Tailwind CSS 的使用。你可以根据需要修改和扩展这个页面。

[了解更多](#)

图片展示



- HTML 代码

复杂表格示例

搜索...

重置搜索

编号	姓名	年龄	邮箱	操作
1	张三	25	zhangsan@example.com	<div>编辑</div> <div>删除</div>
2	李四	30	lisi@example.com	<div>编辑</div> <div>删除</div>
3	王五	22	wangwu@example.com	<div>编辑</div> <div>删除</div>

显示 1 - 3 条记录，共 3 条

上一页

下一页

● 用 HTML 生成表格

在多元微积分中，三重积分的定义如下：

$$\iiint\limits_V f(x,y,z) \, dV = \int_a^b \int_{c(x)}^{d(x)} \int_{e(x,y)}^{f(x,y)} f(x,y,z) \, dz \, dy \, dx$$

其中 (V) 是三维空间中的一个区域， $(f(x,y,z))$ 是定义在 (V) 上的函数。

一个公式

● Latex 语法

在多元微积分中，三重积分的定义如下：

$$\iiint\limits_V f(x,y,z) \, dV = \int_a^b \int_{c(x)}^{d(x)} \int_{e(x,y)}^{f(x,y)} f(x,y,z) \, dz \, dy \, dx$$

其中 (V) 是三维空间中的一个区域， $(f(x,y,z))$ 是定义在 (V) 上的函数。

傅里叶级数

周期为 $(2L)$ 的函数 $(f(x))$ 的傅里叶级数展开式为：

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right) \right]$$

其中系数 (a_n) 和 (b_n) 由以下公式计算：

$$a_n = \frac{1}{L} \int_{-L}^L f(x) \cos\left(\frac{n\pi x}{L}\right) dx, \quad n = 0, 1, 2, \cdots$$
$$b_n = \frac{1}{L} \int_{-L}^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx, \quad n = 1, 2, 3, \cdots$$

Latex 语法，markdown 无法正确显示

复杂数学公式示例

多元积分

在多元微积分中，三重积分的定义如下：

$$\iiint\limits_V f(x,y,z) \, dV = \int_a^b \int_{c(x)}^{d(x)} \int_{e(x,y)}^{f(x,y)} f(x,y,z) \, dz \, dy \, dx$$

其中 (V) 是三维空间中的一个区域， $(f(x,y,z))$ 是定义在 (V) 上的函数。

● 修改后的 markdown 代码

多元积分

在多元微积分中，三重积分的定义如下：

$$\iiint_V f(x, y, z) dV = \int_a^b \int_{c(x)}^{d(x)} \int_{e(x,y)}^{f(x,y)} f(x, y, z) dz dy dx$$

其中 (V) 是三维空间中的一个区域，(f(x,y,z)) 是定义在 (V) 上的函数。

傅里叶级数

周期为 (2L) 的函数 (f(x)) 的傅里叶级数展开式为：

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right) \right)$$

- 修改后的结果

```
f-5b52d019146e
[! 2025-04-25 15:49:35.641 ServerApp] Interrupted...
[IPKernelApp] WARNING | Parent appears to have exited, shutting down.
[IPKernelApp] WARNING | Parent appears to have exited, shutting down.
```

- 关掉前端后，后端还没有关掉，ctrl+c 退掉
3. 使用 tushare 获取一些数据

```
! environment.yml
1  name: week07
2  channels:
3    - conda-forge
4  dependencies:
5    - python=3.12
6    - wat-inspector
7    - jupyterlab
8    - pip
9    - pip:
10   - tushare
```

- 修改 conda 环境，安装 tushare

```
>>> import tushare as ts
>>> ts.set_token("82fe0893301b834c5")
```

```
4月 25 16:31 tk.csv
```

- Set_token 函数会把我的 token 字符串保存在 tk.csv 文件中

```
In [5]: pro
Out[5]: <tushare.pro.client.DataApi at 0x2493d283b60>

In [6]: pro.new_share()
Out[6]:
```

	ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	pe	limit_a
0	603014.SH	732014	威高血净	20250508	None	4114.0		0.0	0.00	0.00
1	301595.SZ	301595	太力科技	20250508	None	2707.0		0.0	0.00	0.00

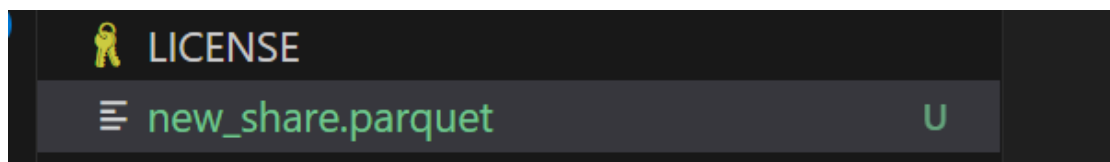
- 调用 IPO 新股上市数据，不设置日期的情况下单次 2000 条

```
In [7]: df = pro.new_share()

In [8]: type(df)
Out[8]: pandas.core.frame.DataFrame
```

- 数据类型

```
In [5]: df.to_parquet("new_share.parquet")
```



- 将下载的数据按照 parquet 格式保存至本地

```
In [9]: df.columns
Out[9]:
Index(['ts_code', 'symbol', 'name', 'area', 'industry', 'cnsPELL', 'market',
      'list_date', 'act_name', 'act_ent_type'],
      dtype='object')
```

- 输出数据的列

```
In [12]: df = pro.stock_basic(fields="ts_code,symbol,name,area,industry,fullname,ename,cnsPELL,market,exchange,curr_ty
      : pe,list_status,list_date,delist_date,is_hs,act_name,act_ent_type")

In [13]: df.shape
Out[13]: (5415, 17)
```

```
In [15]: df.to_parquet("stock_basic.parquet")
```

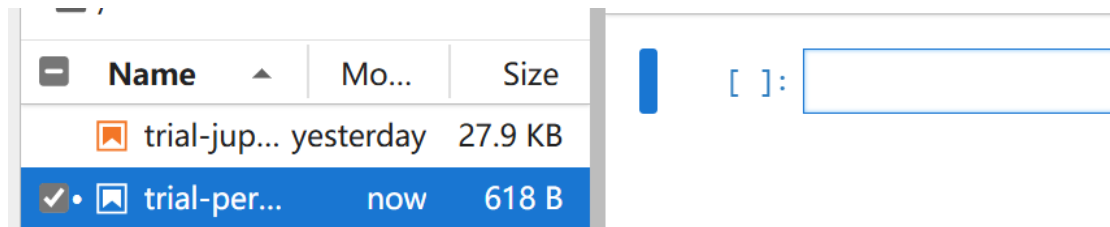
- 下载股票列表数据
4. 通过 perspective-python 软件包查看 polars.DataFrame 数据，实践交互式可视化

```

! environment.yml
1  name: week07
2  channels:
3    - conda-forge
4  dependencies:
5    - python=3.12
6    - wat-inspector
7    - jupyterlab
8    - pyarrow
9    - perspective-python
10   - polars
11   - pip
12   - pip:
13     - tushare

```

- 修改环境文件，安装 perspective-python 和依赖项 polars



- 打开 jupyter lab 并新建一个文件

```

• [2]: d1 = pl.read_parquet("new_share.parquet")
      d1

```

[2]: shape: (2_000, 12)

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	p
str	str	str	str	str	f64	f64	f64	f64
"603014.SH"	"732014"	"威高血净"	"20250508"	null	4114.0	0.0	0.0	0.0

- 读取数据文件，最后一行为表达式 (d1/d2) 就会有输出，即展示数据

```

[6]: d1.with_columns(
      ipo_date=pl.col.ipo_date.str.to_date("%Y%m%d"),
      issue_date=pl.col.issue_date.str.to_date("%Y%m%d"),
    )

```

[6]: shape: (2_000, 12)

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	pe	limit
str	str	str	date	date	f64	f64	f64	f64	f64
"603014.SH"	"732014"	"威高血净"	2025-05-08	null	4114.0	0.0	0.0	0.0	0.0
"301595.SZ"	"301595"	"太力科技"	2025-05-08	null	2707.0	0.0	0.0	0.0	0.0

- 对数据进行列变换，将字符串类型的日期转换为 polars.date()

```
[4]: d1 = d1.with_columns(
    ipo_date=pl.col.ipo_date.str.to_date("%Y%m%d"),
    issue_date=pl.col.issue_date.str.to_date("%Y%m%d"),
)
```

```
[5]: d2 = d2.with_columns(
    list_date=pl.col.list_date.str.to_date("%Y%m%d"),
)
```

- 准备好 d1、d2 两个数据

PerspectiveWidget(d1)

untitled 2,000 x 12

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	pe	limit_amount	funds
Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit
603014.SH	732014	威高血净	2025/5/7	-	4,114.00	0.00	0.00	0.00	1.10	0.00
301595.SZ	301595	太力科技	2025/5/7	-	2,707.00	0.00	0.00	0.00	0.65	0.00
688755.SH	787755	汉邦科技	2025/5/6	-	2,200.00	0.00	0.00	0.00	0.50	0.00
301636.SZ	301636	泽润新能	2025/4/27	-	1,597.00	455.00	33.06	17.57	0.45	5.28
920068.BJ	920068	天工股份	2025/4/27	-	6,000.00	4,200.00	3.94	14.98	255.00	2.36
001400.SZ	001400	江顺科技	2025/4/14	2025/4/23	1,500.00	1,500.00	37.36	15.32	1.50	5.60
301560.SZ	301560	众捷汽车	2025/4/14	2025/4/24	3,040.00	1,216.00	16.50	21.30	0.70	5.02
603202.SH	732202	天有为	2025/4/13	2025/4/23	4,000.00	2,611.00	93.50	13.50	1.25	37.40
301662.SZ	301662	宏工科技	2025/4/7	2025/4/16	2,000.00	813.00	26.60	7.05	0.45	5.32

Group By

Split By

Order By

Where

Columns

ts_code

- 初始化数据

↑ ↓ ↕ ↶ ↷

🔍 🔒 ✂️ ↺ ↻ 📄 >>

Datagrid

- Datagrid 是默认视图

s_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price
Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit
03014.SH	732014	威高血净	2025/5/7	-	4,114.00	0.00	0.00
01595.SZ	301595	太力科技	2025/5/7	-	2,707.00	0.00	0.00
88755.SH	787755	汉邦科技	2025/5/6	-	2,200.00	0.00	0.00
01636.SZ	301636	泽润新能	2025/4/27	-	1,597.00	455.00	33.06
20068.BJ	920068	天工股份	2025/4/27	-	6,000.00	4,200.00	3.94
01400.SZ	001400	江顺科技	2025/4/14	2025/4/23	1,500.00	1,500.00	37.36

Style

Format

italics

none

link

bold

italics

- 对于字符串可以进行一些设置（链接/加粗/斜体）

symbol	name	area	industry	fullname	enname	Style
Edit	Edit	↑ Edit	Edit	Edit	Edit	
000001	平安银行	深圳	银行	平安银行股份有限公司	Ping An Bank Co	Format none
000002	万科A	深圳	全国地产	万科企业股份有限公司	China Vanke Co.	Color series
000004	国华网安	深圳	软件服务	深圳国华网安科技股份有限公司	Shenzhen Guohua	Series
000006	深振业A	深圳	区域地产	深圳市振业(集团)股份有限公司	Shenzhen Zhenye	
000007	全新好	深圳	其他商业	深圳市全新好股份有限公司	Shenzhen Quanyi	
000008	神州高铁	北京	运输设备	神州高铁技术股份有限公司	China High Speed	
000009	中国宝安	深圳	电气设备	中国宝安集团股份有限公司	China Baoan Gro	

- 对字符串的颜色进行设置

	↑ Edit	Edit	Edit	Edit
	4,114	0.00	0.00	0.00
	2,707	0.00	0.00	0.00
	2,200	0.00	0.00	0.00
	1,597	455.00	33.06	17.57
	6,000	4,200.00	3.94	14.98
	1,500	1,500.00	37.36	15.32
	3,040	1,216.00	16.50	21.30
	4,000	2,611.00	93.50	13.50
	2,000	813.00	26.60	7.05

- 对数值的显示进行设置（调整小数点的保留）

issue_date	amount	market_amount	price
Edit	↑ Edit	Edit	Edit
2021/8/19		380,402.00	4.53
2021/6/9		625,451.00	2.65
2020/1/15		234,379.00	4.88
2019/12/9		181,026.00	5.50
2019/8/25		184,252.00	2.49

- 选择 bar 并进行排序（柱状图）

market_amount	price	
Edit	↑ Edit	
312.00	148.90	6
735.00	34.75	1
640.00	5.77	1
640.00	11.00	1
736.00	10.88	2
315.00	45.72	5
933.00	58.31	3
659.00	18.68	1

ts_code	symbol	name	area	industry	fullname	enname	cnsPELL	market	exchange	curr_type	list_status
Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit
TOTAL	5,415	5,415	5,415	5,415	5,415	5,415	5,415	5,415	5,415	5,415	5,415
主板	3,183	3,183	3,183	3,183	3,183	3,183	3,183	3,183	3,183	3,183	3,183
创业板	1,381	1,381	1,381	1,381	1,381	1,381	1,381	1,381	1,381	1,381	1,381
北交所	265	265	265	265	265	265	265	265	265	265	265
科创板	586	586	586	586	586	586	586	586	586	586	586

- 将 d2 按照市场类型进行分类

industry	fullname	enname	cnsPELL	market	exchange	curr_type	list_status
Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit
111	5,415	5,415	5,415	5,415	5,415	5,415	5,415
110	3,183	3,183	3,183	3,183	3,183	3,183	3,183
77	1,381	1,381	1,381	1,381	1,381	1,381	1,381
51	265	265	265	265	265	265	265
33	586	586	586	586	586	586	586

- 对行业进行不重复的计数

	ts_code	industry	industry2
	Edit	Edit	Edit
- TOTAL	5,415	111	电气设备
主板	3,183	110	电气设备
创业板	1,381	77	软件服务
北交所	265	51	汽车配件
科创板	586	33	半导体

- Dominate: 查看数据中占比最高的一类

	ts_code	industry	industry2
	Edit	Edit	Edit
TOTAL	5,415	111	电气设备
- BSE	265	51	汽车配件
北交所	265	51	汽车配件
- SSE	2,282	107	半导体
主板	1,696	107	汽车配件
科创板	586	33	半导体
- SZSE	2,868	110	元器件
主板	1,487	107	元器件
创业板	1,381	77	软件服务

- 进行多层分组

	null	IT设备	专用机...	中成药	乳制品	互联网	仓储物...	供气供...	保险	元器件	全国地...	公共交...
	ts_code	ts_code	ts_code	ts_code	ts_code	ts_code	ts_code	ts_code	ts_code	ts_code	ts_code	ts_cod
	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit
- TOTAL	17	75	274	71	21	82	46	44	6	295	26	
- BSE	17	5	17	2	1	-	1	1	-	14	-	
北交所	17	5	17	2	1	-	1	1	-	14	-	
- SSE	-	12	113	31	9	20	25	27	5	81	12	
主板	-	7	54	31	9	19	25	27	5	42	12	
科创板	-	5	59	-	-	1	-	-	-	39	-	
- SZSE	-	58	144	38	11	62	20	16	1	200	14	
主板	-	23	45	27	7	26	16	13	1	89	14	
创业板	-	35	99	11	4	36	4	3	-	111	-	

- 使用 split by 作为拆分依据

	ts_code
	Edit
- TOTAL	5,415
- BSE	265
北交所	265
- SSE	2,282
科创板	586
主板	1,696
- SZSE	2,868
创业板	1,381
主板	1,487

- 在每一个分组中进行从小到大的排序

ts_code	list_date
Edit	Edit
000001.SZ	1991/4/2
000002.SZ	1991/1/28
000004.SZ	1991/1/13
000006.SZ	1992/4/26
000007.SZ	1992/4/12
000008.SZ	1992/5/6
000009.SZ	1991/6/24
000010.SZ	1995/10/26
000011.SZ	1992/3/29

Group By
Split By
Order By
ts_code
Where
list_date > 1997/12/01

● 根据上市日期进行筛选

12)

ipo_date	issue_date	amount	market_amount	上网发行比例
Edit	Edit	Edit	Edit	Edit
2025/5/7	-	4,114	0	0.00%
2025/5/7	-	2,707	0	0.00%
2025/5/6	-	2,200	0	0.00%
2025/4/27	-	1,597	455	28.49%
2025/4/27	-	6,000	4,200	70.00%
2025/4/14	2025/4/23	1,500	1,500	100.00%
2025/4/14	2025/4/24	3,040	1,216	40.00%

New Column 1
Attributes
Expression
1 X
Empty expression!

Datagrid
Columns
ts_code
sub_code
name
ipo_date
issue_date
amount

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	上网发行比例
Edit	Edit	Edit	Edit	Edit	Edit	Edit	Edit
603014.SH	732014	威高血净	2025/5/7	-	4,114	0	0.00%
301595.SZ	301595	太力科技	2025/5/7	-	2,707	0	0.00%
688755.SH	787755	汉邦科技	2025/5/6	-	2,200	0	0.00%
301636.SZ	301636	泽润新能	2025/4/27	-	1,597	455	28.49%
920068.BJ	920068	天工股份	2025/4/27	-	6,000	4,200	70.00%
001400.SZ	001400	江顺科技	2025/4/14	2025/4/23	1,500	1,500	100.00%
301560.SZ	301560	众捷汽车	2025/4/14	2025/4/24	3,040	1,216	40.00%
603202.SH	732202	天有为	2025/4/13	2025/4/23	4,000	2,611	65.28%
301667.SZ	301667	安丁科技	2025/4/7	2025/4/16	2,000	813	40.65%

Style
Attributes
Foreground
Color
Color Range
Background
Gradient
Color Range

● 创造新的一列

Datagrid
Export
Save as
untitled
Current View
untitled.csv
untitled.json
untitled.ndjson
untitled.arrow
untitled.html

price_cent	pe	limit_amount	fu
Edit	Edit	Edit	Edit
¥0	0.00	1.10	0
¥0	0.00	0.65	0
¥0	0.00	0.50	0
¥3,306	17.57	0.45	5
¥394	14.98	255.00	2
¥3,736	15.32	1.50	5

● 导出

¥1,027	13.40	0.70	4	
¥860	18.60	2.70	7	Config
¥2,052	18.17	0.95	6	untitled.config.json

- 导出经过图形界面设置的纯文本文件

```
C: > Users > cherry > Downloads > {} untitled.config.json > ...
1 [{"version": "3.6.0", "plugin": "Datagrid", "plugin_config": {"columns": {}, "edit_mode": "READ_ONLY", "sc
```

- 导出的纯文本文件

0.70	5.02			unt
1.25	37.40			Config
0.45	5.32			clipboard.config.json

- 把图形界面的设置复制到剪贴板

```
: config = {"version": "3.6.0", "plugin": "Datagrid", "plugin_config": {"columns": {}, "edit_mode": "READ_ONLY", "scroll_lock": false}, "columns_config": {"amount":
: print(config)

{"version": "3.6.0", "plugin": "Datagrid", "plugin_config": {"columns": {}, "edit_mode": "READ_ONLY", "scroll_lock": false}, "columns_config": {"amount":
{"number_format": {"minimumFractionDigits": 0.0, "maximumFractionDigits": 2.0}, "price": {"number_bg_mode": "gradient", "bg_gradient": 557.8}, "price_cent": {"number_format": {"style": "currency", "currency": "CNY", "minimumFractionDigits": 0.0, "maximumFractionDigits": 2.0}, "上网发行比例": {"number_bg_mode": "gradient", "bg_gradient": 1.0, "number_format": {"style": "percent", "minimumFractionDigits": 2.0, "maximumFractionDigits": 2.0}, "name": {"format": "bold", "string_color_mode": "foreground"}, "market_amount": {"number_format": {"minimumFractionDigits": 0.0, "maximumFractionDigits": 2.0}}, "settings": true, "theme": "Pro Light", "title": null, "group_by": [], "split_by": [], "sort": [], "filter": [], "expressions": {"price_cent": "price * 100", "上网发行比例": "market_amount / amount"}, "columns": ["ts_code", "sub_code", "name", "ipo_date", "issue_date", "amount", "market_amount", "上网发行比例", "price", "price_cent", "pe", "limit_amount", "funds", "ballot"], "aggregates": {}}

[17]: config2 = Path("C:/Users/cherry/Downloads/untitled.config.json").read_text(encoding="utf8")
[18]: print(config2)

{"version": "3.6.0", "plugin": "Datagrid", "plugin_config": {"columns": {}, "edit_mode": "READ_ONLY", "scroll_lock": false}, "columns_config": {"amount":
{"number_format": {"minimumFractionDigits": 0.0, "maximumFractionDigits": 2.0}, "price": {"number_bg_mode": "gradient", "bg_gradient": 557.8}, "price_cent": {"number_format": {"style": "currency", "currency": "CNY", "minimumFractionDigits": 0.0, "maximumFractionDigits": 2.0}, "上网发行比例": {"number_bg_mode": "gradient", "bg_gradient": 1.0, "number_format": {"style": "percent", "minimumFractionDigits": 2.0, "maximumFractionDigits": 2.0}, "name": {"format": "bold", "string_color_mode": "foreground"}, "market_amount": {"number_format": {"minimumFractionDigits": 0.0, "maximumFractionDigits": 2.0}}, "settings": true, "theme": "Pro Light", "title": null, "group_by": [], "split_by": [], "sort": [], "filter": [], "expressions": {"price_cent": "price * 100", "上网发行比例": "market_amount / amount"}, "columns": ["ts_code", "sub_code", "name", "ipo_date", "issue_date", "amount", "market_amount", "上网发行比例", "price", "price_cent", "pe", "limit_amount", "funds", "ballot"], "aggregates": {}}
```

- 无论是复制到剪切板还是导出文件都可以将图形界面的设置导出来

```
[19]: import json
[21]: d = json.loads(config2)
[22]: d
[22]: {'version': '3.6.0',
      'plugin': 'Datagrid',
      'plugin_config': {'columns': {}},
      'edit_mode': 'READ_ONLY',
      'scroll_lock': False,
      'columns_config': {'amount': {'number_format': {'minimumFractionDigits': 0.0,
      'maximumFractionDigits': 2.0}},
      'price': {'number_bg_mode': 'gradient', 'bg_gradient': 557.8},
```

```
23]: type(d)
```

```
23]: dict
```

```
25]: d["columns_config"]
```

```
25]: {'amount': {'number_format': {'minimumFractionDigits': 0.0,
    'maximumFractionDigits': 2.0}},
    'price': {'number_bg_mode': 'gradient', 'bg_gradient': 557.8},
    'price_cent': {'number_format': {'style': 'currency',
    'currency': 'CNY',
    'minimumFractionDigits': 0.0,
    'maximumFractionDigits': 2.0}},
    '上网发行比例': {'number_bg_mode': 'gradient',
    'bg_gradient': 1.0,
    'number_format': {'style': 'percent',
    'minimumFractionDigits': 2.0,
    'maximumFractionDigits': 2.0}},
    'name': {'format': 'bold', 'string_color_mode': 'foreground'},
    'market_amount': {'number_format': {'minimumFractionDigits': 0.0,
    'maximumFractionDigits': 2.0}}}
```

- 字典中有键和值，值中又有字典

```
[29]: PerspectiveWidget(
    d1,
    expressions={
        'price_cent': '"price" * 100',
        '上网发行比例': '"market_amount"/"amount"'
    }
)
```

- 在初始化时利用代码进行设置

The screenshot displays a Jupyter Notebook interface. On the left, a code cell contains a JSON configuration for a Datagrid plugin. The configuration includes settings for columns, number formatting, and expressions. On the right, the 'Tree Viewer' panel shows the hierarchical structure of the configuration, with nodes expanded to show their contents. The configuration is as follows:

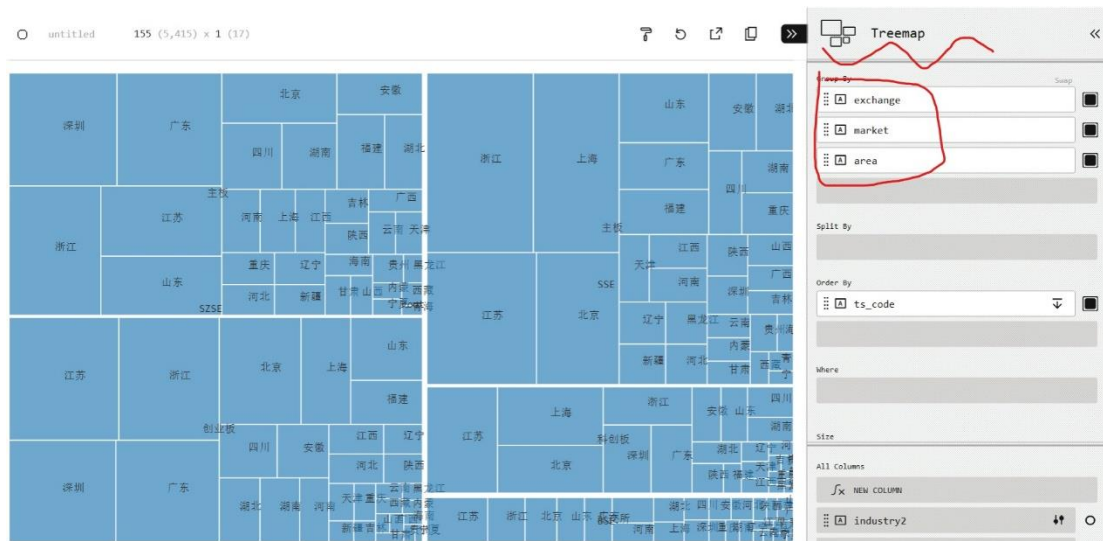
```
{
  "version": "3.6.0",
  "plugin": "Datagrid",
  "plugin_config": {
    "columns": {},
    "edit_mode": "READ_ONLY",
    "scroll_lock": false,
    "columns_config": {
      "amount": {
        "number_format": {
          "minimumFractionDigits": 0.0,
          "maximumFractionDigits": 2.0
        },
        "price": {
          "number_bg_mode": "gradient",
          "bg_gradient": 557.8,
          "price_cent": {
            "number_format": {
              "style": "currency",
              "currency": "CNY",
              "minimumFractionDigits": 0.0,
              "maximumFractionDigits": 2.0
            },
            "上网发行比例": {
              "number_bg_mode": "gradient",
              "bg_gradient": 1.0,
              "number_format": {
                "style": "percent",
                "minimumFractionDigits": 2.0,
                "maximumFractionDigits": 2.0
              },
              "name": {
                "format": "bold",
                "string_color_mode": "foreground"
              },
              "market_amount": {
                "number_format": {
                  "minimumFractionDigits": 0.0,
                  "maximumFractionDigits": 2.0
                }
              },
              "settings": true,
              "theme": "Pro Light",
              "title": null,
              "group_by": [],
              "split_by": [],
              "sort": [],
              "filter": [],
              "expressions": {
                "price_cent": "\"price\" * 100",
                "上网发行比例": "\"market_amount\"/\"amount\""
              }
            },
            "columns": {
              "ts_code": "ts_code",
              "sub_code": "sub_code",
              "name": "name",
              "ipo_date": "ipo_date",
              "market_amount": "market_amount",
              "上网发行比例": "上网发行比例"
            }
          }
        }
      }
    }
  }
}
```

The 'Tree Viewer' panel on the right shows the following structure:

- object {14}
 - version: 3.6.0
 - plugin: Datagrid
 - plugin_config {3}
 - columns {0}
 - (清空 object)
 - edit_mode: READ_ONLY
 - scroll_lock: false
 - columns_config {6}
 - amount {1}
 - number_format {2}
 - minimumFractionDigits: 0
 - maximumFractionDigits: 2

- Json 美化

5. 把 PerspectiveWidget 切换为 Treemap 视图，并尝试设置各种选项

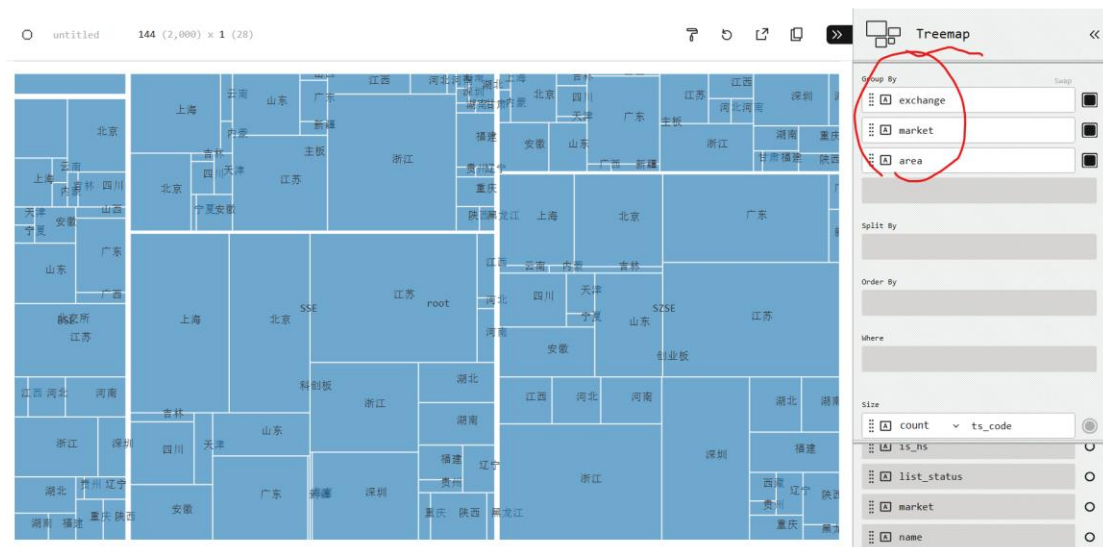


```
[32]: df = d1.join(d2, on="ts_code", how="left")
      df
```

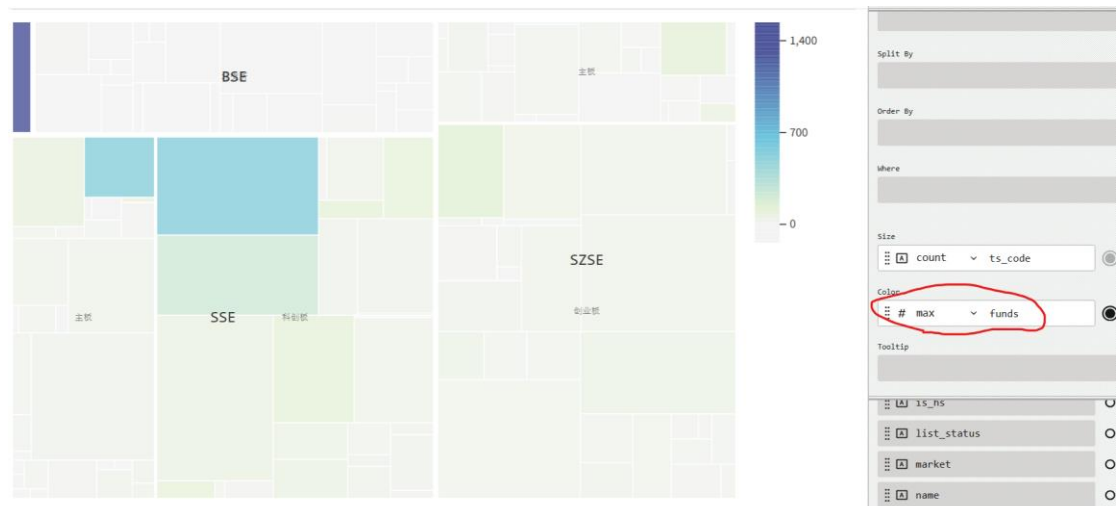
[32]: shape: (2_000, 28)

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price
str	str	str	date	date	f64	f64	f64
"603014.SS"	"732014"	"威高	2025-	null	4114.0	0.0	0.0

- 进行数据的合并，按照 d1 数据的 ts_code 进行合并



- 对合并后的数据进行可视化处理



- 根据筹集资金进行上色，可以选择均值、总值、最大/最小值等

untitled 93 (2,000) × 28

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	pe	limit_amount	funds
688981.SH	787981	中芯国际	2020/7/6	2020/7/15	193,846.00	25,284.00	27.46	0.00	42.10	532.30
688347.SH	787347	华虹公司	2023/7/24	2023/8/6	40,775.00	6,116.00	52.00	34.71	4.05	212.03
688271.SH	787271	联影医疗	2022/8/9	2022/8/21	10,000.00	1,500.00	109.88	77.69	1.00	109.88
688538.SH	787538	和辉光电	2021/5/17	2021/5/27	308,366.00	56,310.00	2.65	0.00	77.75	81.72
688220.SH	787220	翱捷科技	2022/1/3	2022/1/13	4,183.00	1,025.00	164.54	0.00	0.65	68.83
688065.SH	787065	凯赛生物	2020/8/2	2020/8/11	4,167.00	1,110.00	133.45	120.70	0.70	55.61
688608.SH	787608	恒玄科技	2020/12/2	2020/12/15	3,000.00	960.00	162.07	355.03	0.70	48.62
688180.SH	787180	君实生物	2020/7/1	2020/7/14	8,713.00	2,120.00	55.50	0.00	1.35	48.36
688082.SH	787082	盛美上海	2021/11/7	2021/11/17	4,336.00	1,050.00	85.00	398.67	0.65	36.85
688728.SH	787728	格科微	2021/8/8	2021/8/17	24,989.00	5,248.00	14.38	46.92	3.45	35.93

Datagrid

Group By

Split By

Order By

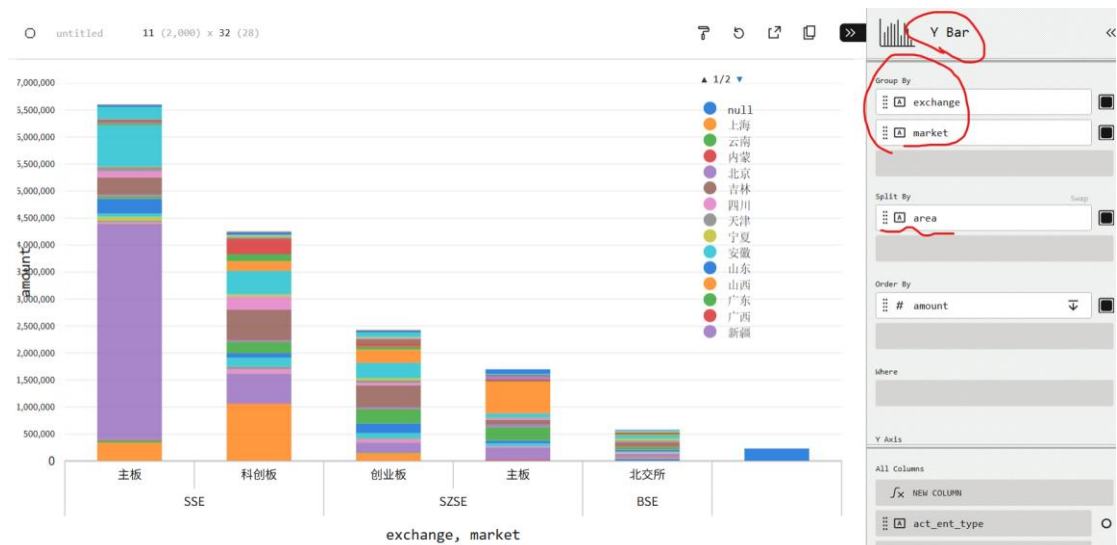
Where

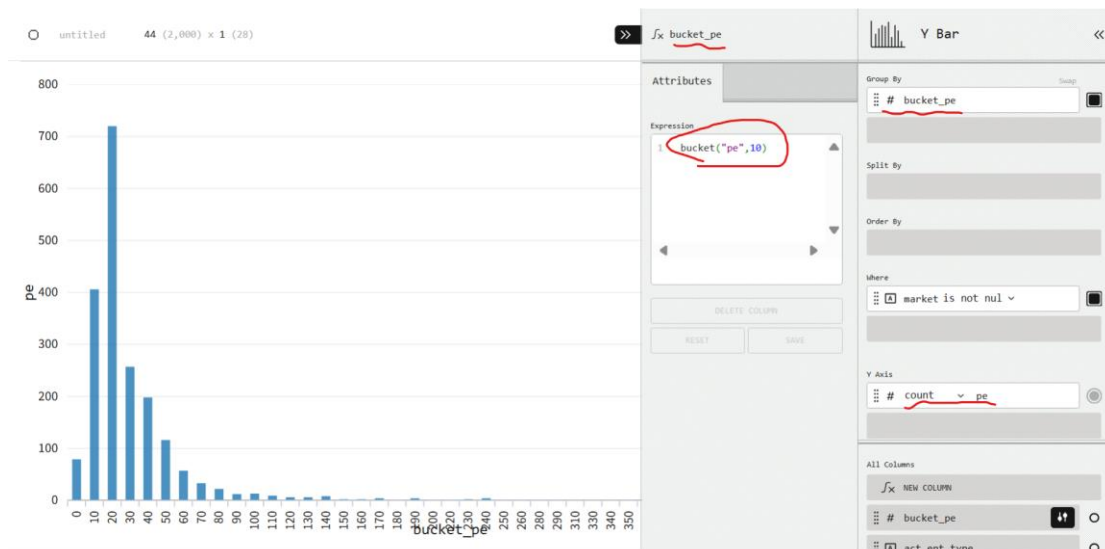
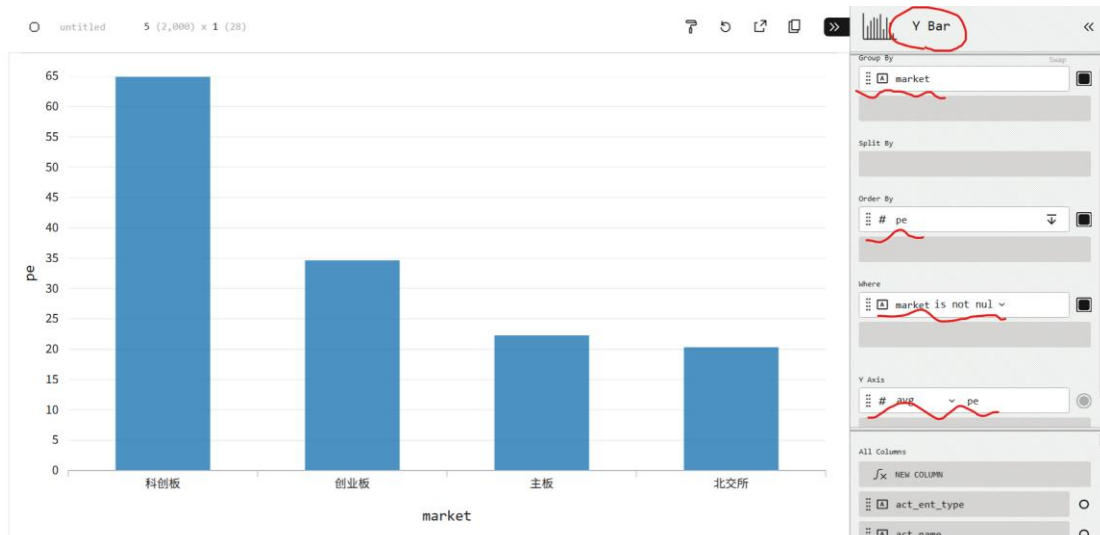
exchange == SSE

market == 科创板

area == 上海

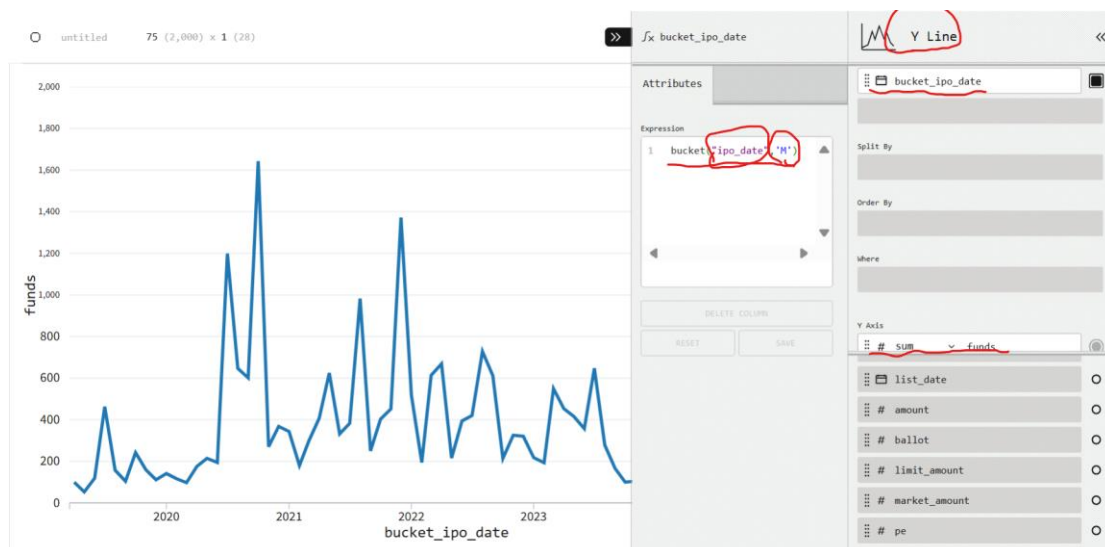
6. 把 PerspectiveWidget 切换为 Y Bar 视图，并尝试设置各种选项

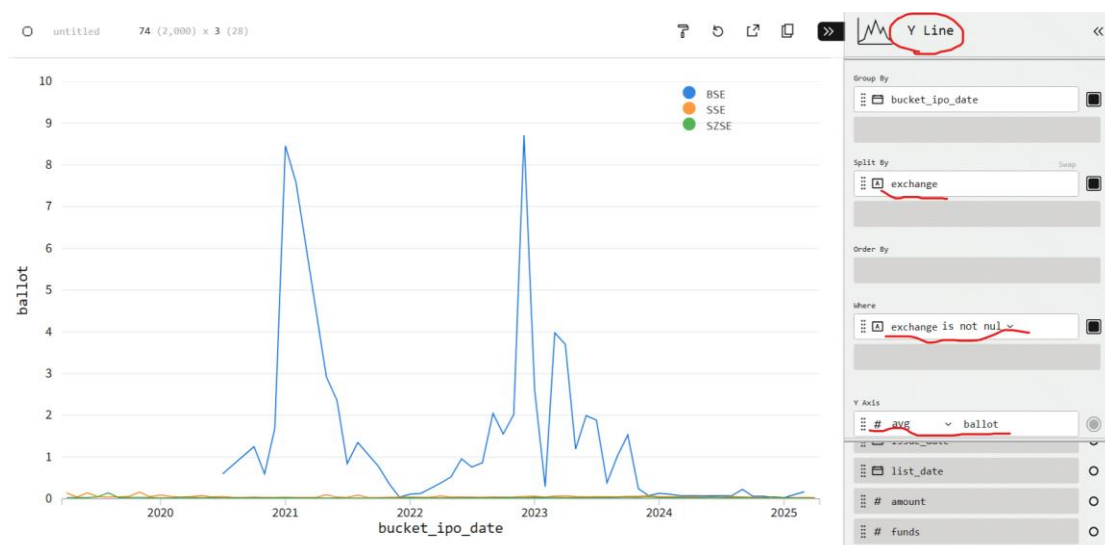
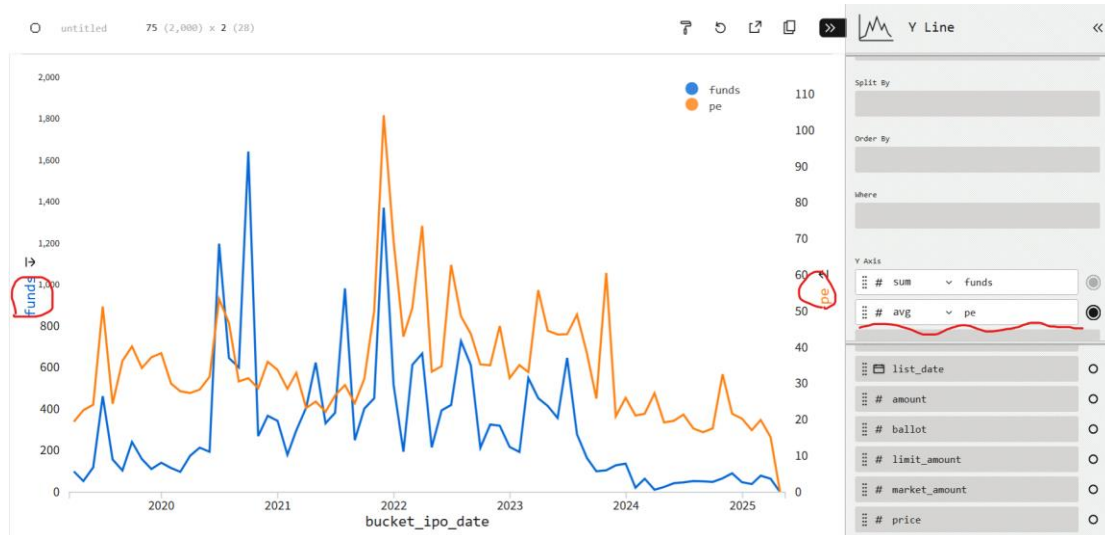




- 将市盈率可视化直方图——首先要利用 bucket 函数生成新的变量（分桶，桶的宽度为 10）

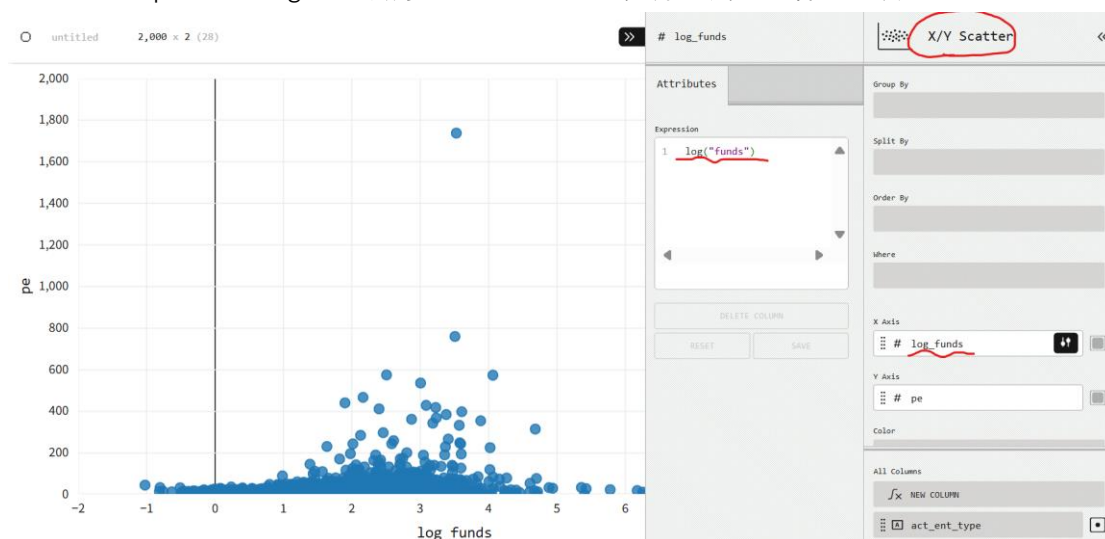
7. 把 PerspectiveWidget 切换为 Y Line 视图，并尝试设置各种选项



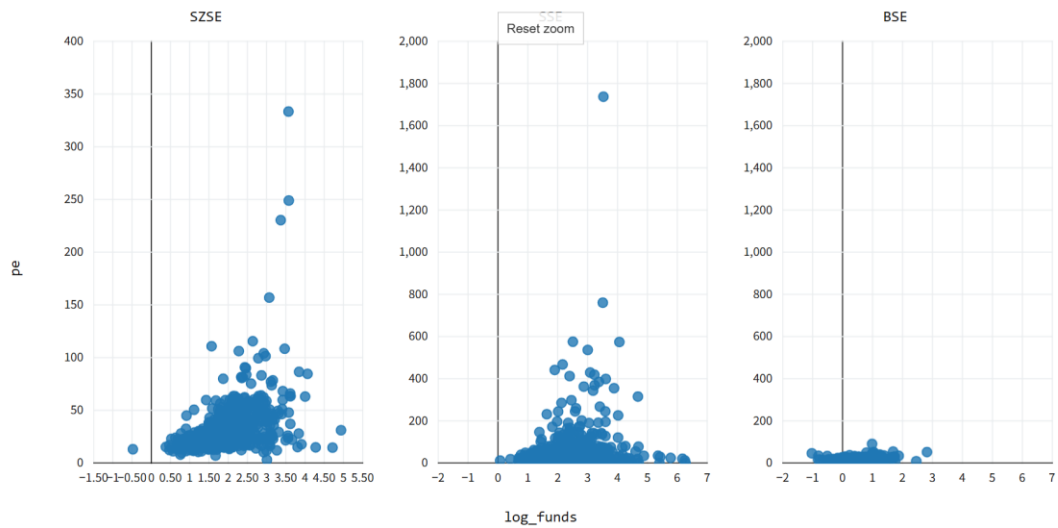


- 不同交易所的新股中签率

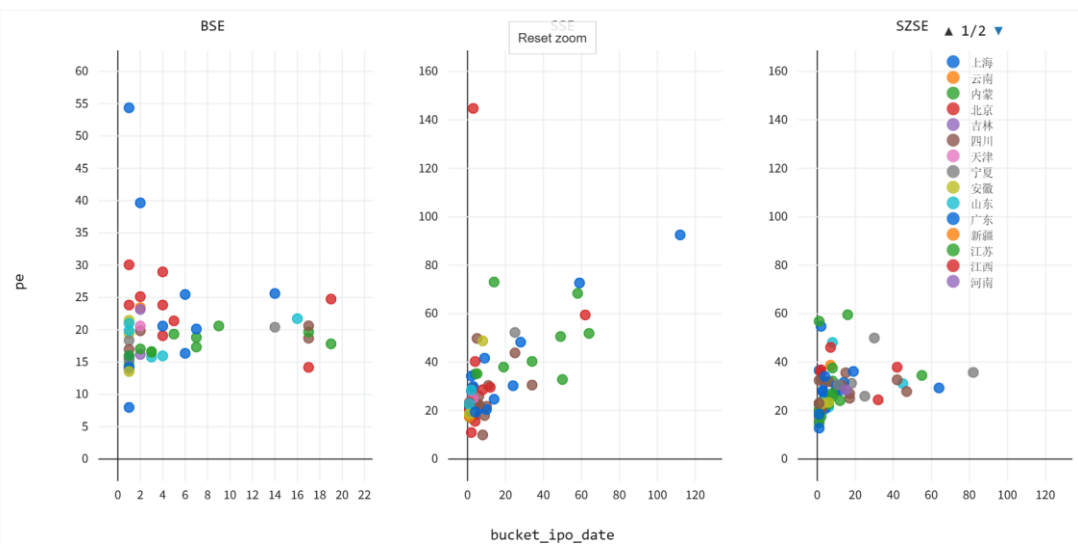
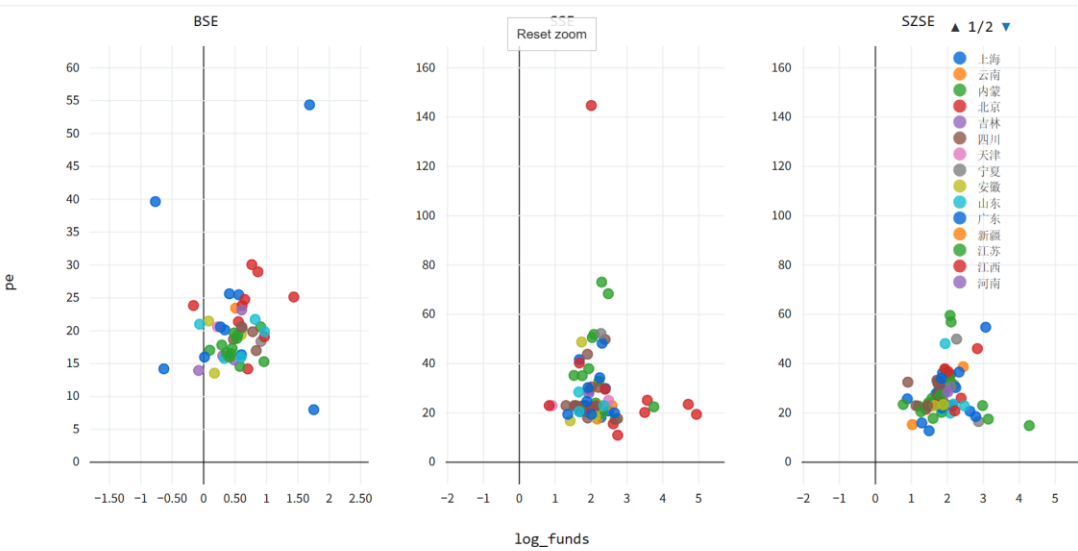
8. 把 PerspectiveWidget 切换为 X/Y Scatter 视图, 并尝试设置各种选项



- 对 x 轴取对数



● Split by (exchange)



● 按照 ipo_date 分桶