

绘制散点图

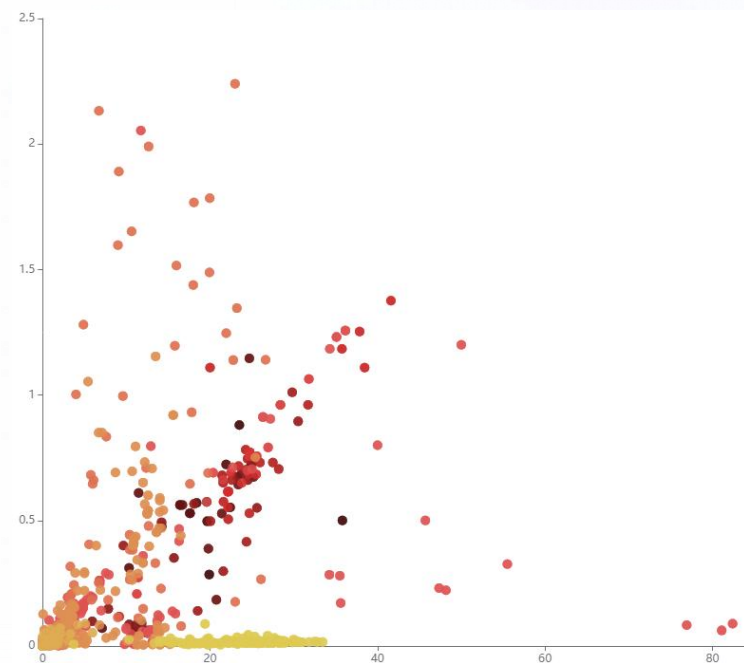
主讲：卢春光





绘制散点图

散点图（Scatter Plot）是一种用于展示两个变量之间关系的图表类型。在散点图中，每个数据点代表一对变量的观测值，其中一个变量的值在水平轴（X轴）上表示，另一个变量的值在垂直轴（Y轴）上表示。通过观察数据点在图表中的分布，可以分析两个变量之间是否存在某种关联、趋势或模式。





绘制散点图

>>> 基础语法

```
scatter = (Scatter(init_opts=opts.InitOpts()))  
    .add_xaxis()  
    .add_yaxis()  
    .set_global_opts()  
    .set_series_opts()
```

参数	说明
<code>opts.InitOpts()</code>	设置初始化配置项
<code>add_xaxis()</code>	设置X轴数据
<code>add_yaxis()</code>	设置Y轴数据
<code>set_global_opts()</code>	设置全局配置项
<code>set_series_opts()</code>	设置系列配置项



>>> 基础语法

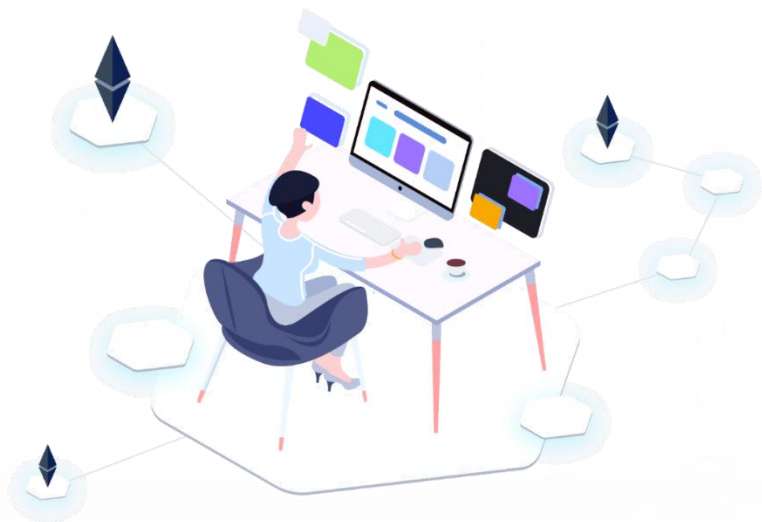
add_yaxis()常用参数

参数	说明
series_name	系列名称，用于tooltip的显示，legend的图例筛选
y_axis	系列数据
is_selected	是否选中图例
xaxis_index	X轴的索引，在单个图表实例中存在多个X轴的时候有用
yaxis_index	Y轴的索引，在单个图表实例中存在多个Y轴的时候有用
color	系列 label 颜色
symbol	标记的图形
symbol_size	标记的大小
label_opts	标签配置项
markpoint_opts	标记点配置项
markline_opts	标记线配置项
tooltip_opts	提示框组件配置项
itemstyle_opts	图元样式配置项



绘制散点图

>>> 全国居民人均收入数据



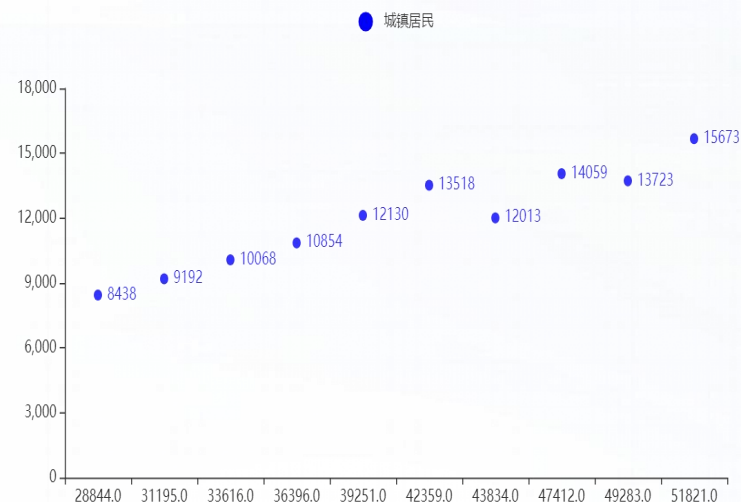
时间：最近10年							
指标	2023年	2022年	2021年	2020年	2019年	2018年	2017年
居民人均可支配收入(元)	39218	36883	35128	32189	30733	28228	25974
居民人均可支配收入 比上年增长(%)	6.1	2.9	8.1	2.1	5.8	6.5	7.3
居民人均可支配收入中位数(元)	33036	31370	29975	27540	26523	24336	22408
居民人均可支配收入中位数 比上年增长(%)	5.3	4.7	8.8	3.8	9.0	8.6	7.3
居民人均可支配工资性收入(元)	22053	20590	19629	17917	17186	15829	14620
居民人均可支配工资性收入 比上年增长(%)	7.1	4.9	9.6	4.3	8.6	8.3	8.7
居民人均可支配经营净收入(元)	6542	6175	5893	5307	5247	4852	4502
居民人均可支配经营净收入 比上年增长(%)	6.0	4.8	11.0	1.1	8.1	7.8	6.7
居民人均可支配财产净收入(元)	3362	3227	3076	2791	2619	2379	2107
居民人均可支配财产净收入 比上年增长(%)	4.2	4.9	10.2	6.6	10.1	12.9	11.6
居民人均可支配转移净收入(元)	7261	6892	6531	6173	5680	5168	4744
居民人均可支配转移净收入 比上年增长(%)	5.4	5.5	5.8	8.7	9.9	8.9	11.4

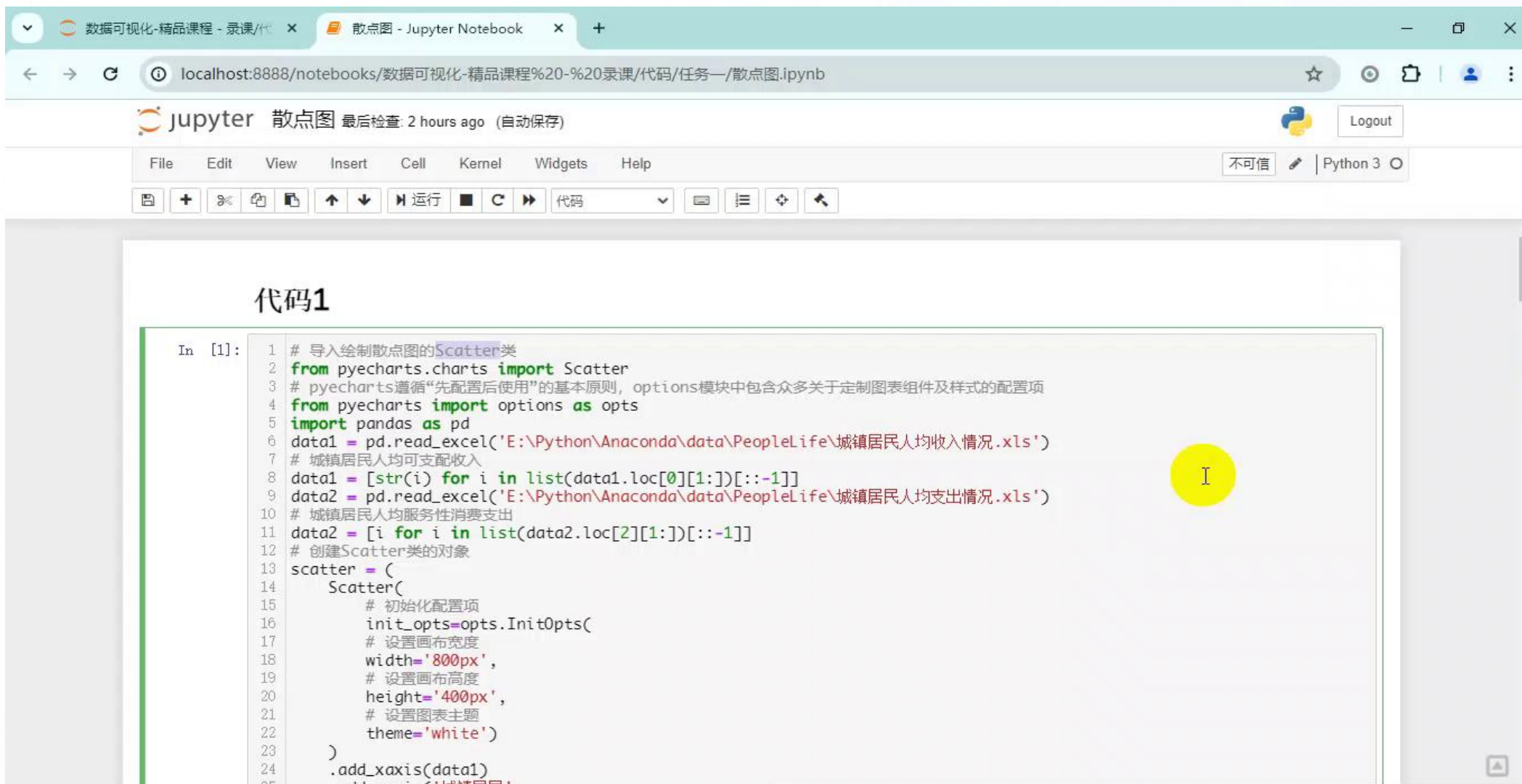


绘制散点图

绘制散点图展示近10年城镇居民人均可支配收入与服务性消费支出关系

```
data1 = pd.read_excel('E:\Python\Anaconda\data\PeopleLife\
城镇居民人均收入情况.xls')
data1 = [str(i) for i in list(data1.loc[0][1:])[::-1]]
data2 = pd.read_excel('E:\Python\Anaconda\data\PeopleLife\
城镇居民人均支出情况.xls')
data2 = [i for i in list(data2.loc[2][1:])[::-1]]
scatter = (Scatter(init_opts=opts.InitOpts(width='800px',
height='400px', theme='white'))
    .add_xaxis(data1)
    .add_yaxis('城镇居民', data2,
    itemstyle_opts=opts.ItemStyleOpts(color='blue'),
    symbol='circle', symbol_size=8))
scatter.render_notebook()
```







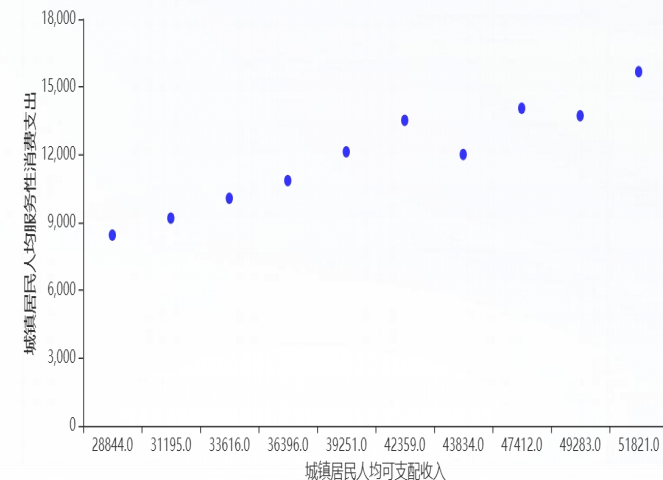
绘制散点图

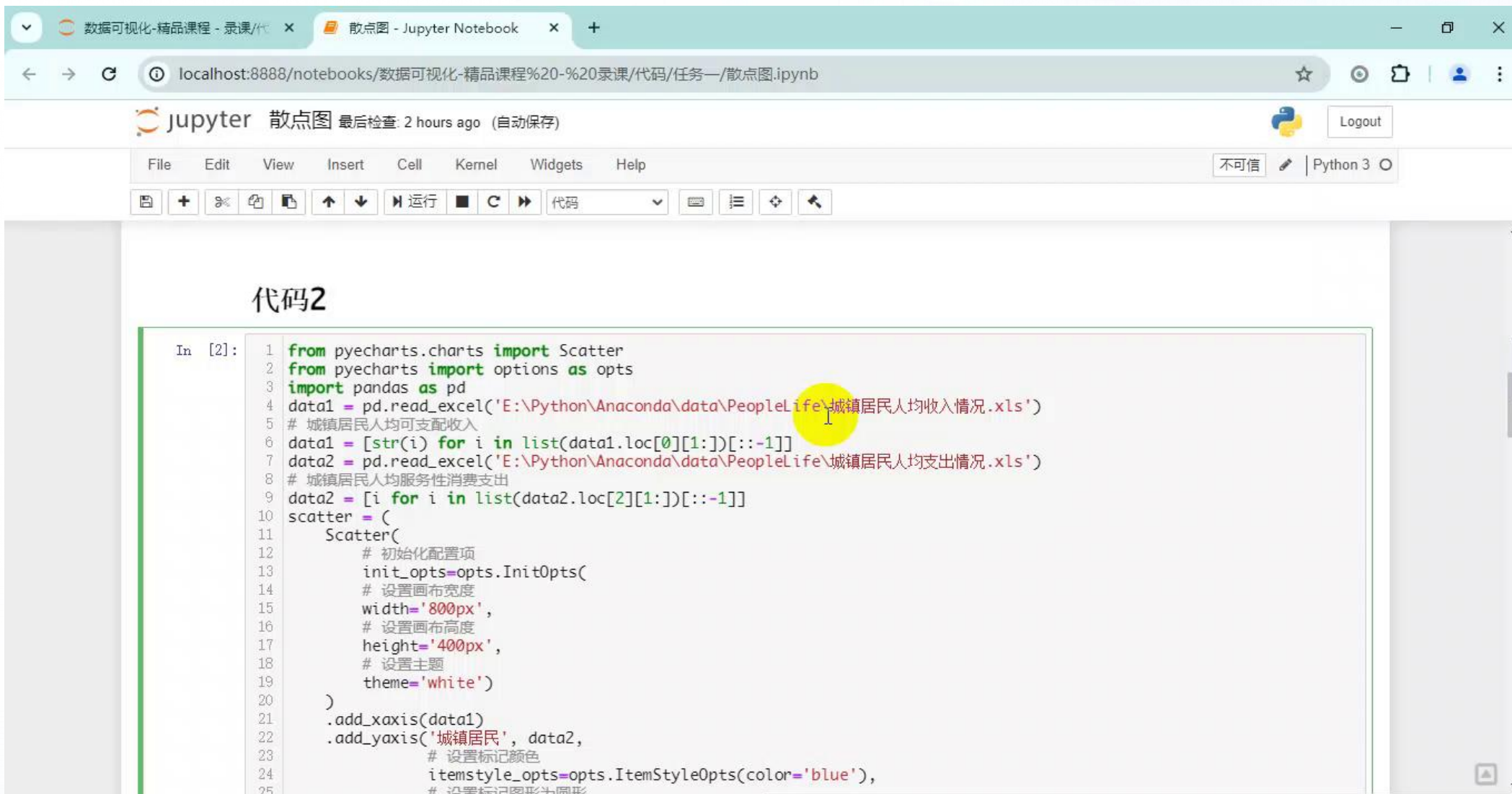
设置全局配置项和系列配置项

```
.set_global_opts(  
    xaxis_opts=opts.AxisOpts(name='城镇居民人均可支配收入',  
        name_textstyle_opts=opts.LabelOpts(font_size=14),  
        name_location='middle',name_gap=25,axislabel_opts=opts.LabelOpts(font_size=12)),  
    yaxis_opts=opts.AxisOpts(name='城镇居民人均服务性消费支出',  
        name_textstyle_opts=opts.LabelOpts(font_size=14),  
        name_location='middle',name_gap=50,axislabel_opts=opts.LabelOpts(font_size=12)),  
    title_opts=opts.TitleOpts(title='2014-2023年城镇居民人均可支配收入与服务性消费支出',  
        subtitle='单位: 元',  
        pos_left='left',pos_top='top',  
        title_textstyle_opts=opts.TextStyleOpts(color='red',font_size=14),  
        subtitle_textstyle_opts=opts.TextStyleOpts(color='red',font_size=14)),  
    legend_opts=opts.LegendOpts(is_show=False))  
.set_series_opts(  
    label_opts=opts.LabelOpts(is_show=False))
```

2014-2023年城镇居民人均可支配收入与服务性消费支出

单位: 元







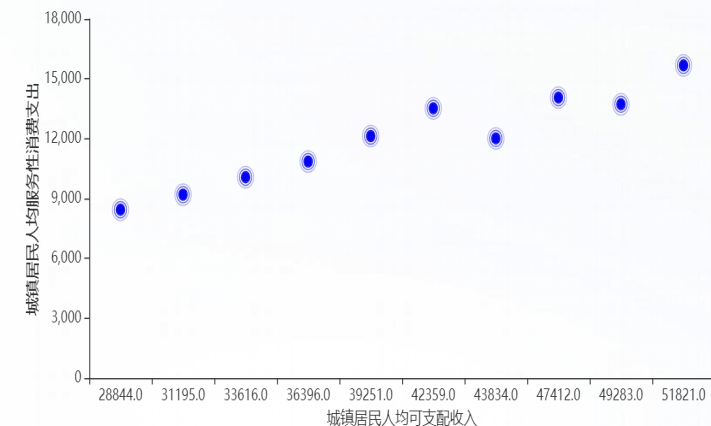
绘制散点图

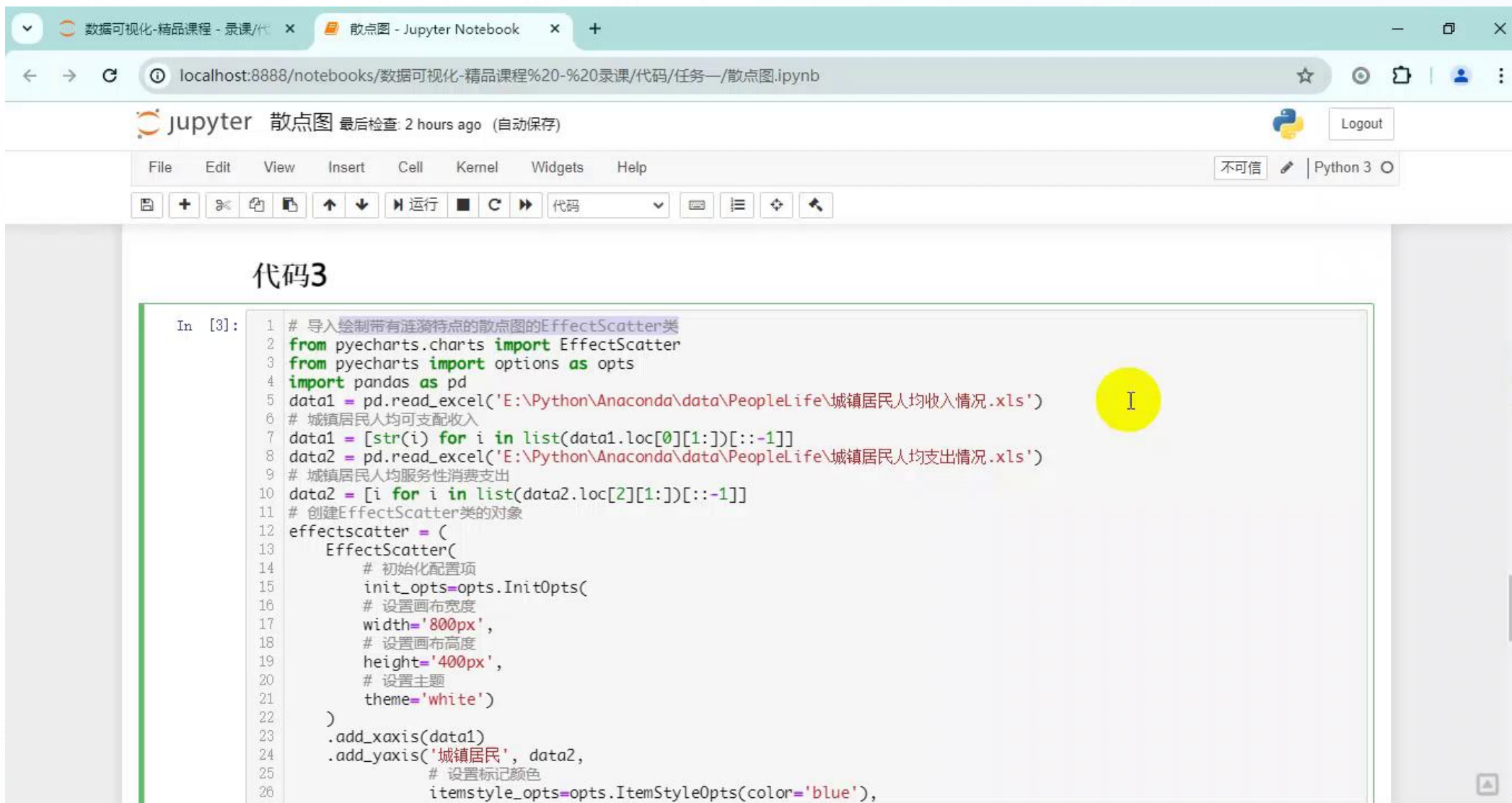
绘制带有涟漪特效的散点图

```
from pyecharts.charts import EffectScatter
from pyecharts import options as opts
import pandas as pd
data1 =
pd.read_excel('E:\Python\Anaconda\data\PeopleLife\城镇居
民人均收入情况.xls')
data1 = [str(i) for i in list(data1.loc[0][1:])[::-1]]
data2 =
pd.read_excel('E:\Python\Anaconda\data\PeopleLife\城镇居
民人均支出情况.xls')
data2 = [i for i in list(data2.loc[2][1:])[::-1]]
effectscatter = (
    EffectScatter(
        init_opts=opts.InitOpts(
            width='800px',
            height='400px',
            theme='white')))
```

2014-2023年城镇居民人均可支配收入与服务性消费支出

单位: 元







绘制散点图

>>> 总结

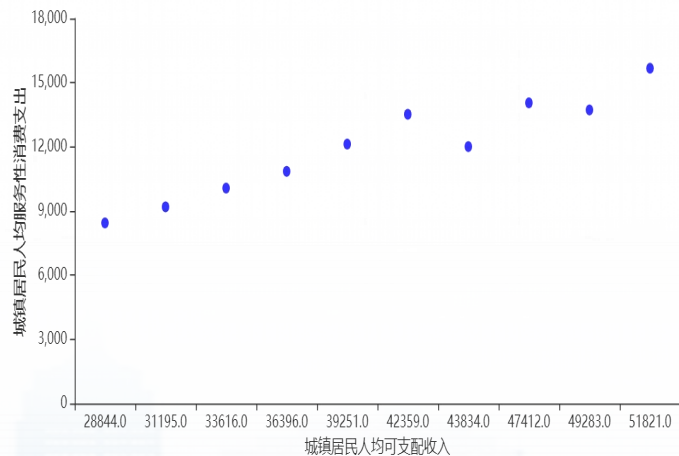
`pyecharts.charts`

`scatter = Scatter()`

`effectscatter = EffectScatter()`

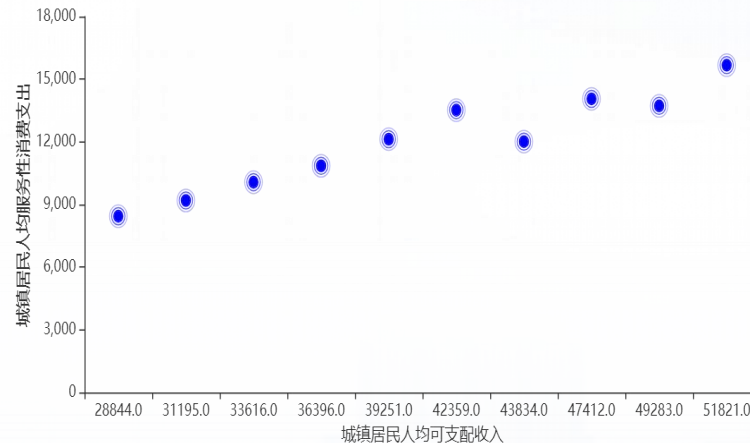
2014-2023年城镇居民人均可支配收入与服务性消费支出

单位：元



2014-2023年城镇居民人均可支配收入与服务性消费支出

单位：元



绘制散点图

主讲：卢春光

