Python 数据分析与应用

实验五：Pandas 统计分析及数据预处理

丁烨
dingye@dgut.edu.cn
计算机科学与技术学院

2023-12-06

東莞理工學院
DONGGUAN UNIVERSITY OF TECHNOLOGY

数据简介

- 精灵宝可梦（ポケットモンスター、Pokémon）
- https://www.pokemon.co.jp/
- 一个跨媒体制作的作品系列，包括游戏、动画、漫画、卡片游戏及相关产品
- 游戏允许玩家捕获，收集，培育数百只生物，也就是通常所说的宝可梦
- 借由与其他宝可梦对战，宝可梦能够提升等级甚至进化，成为更强大的宝可梦

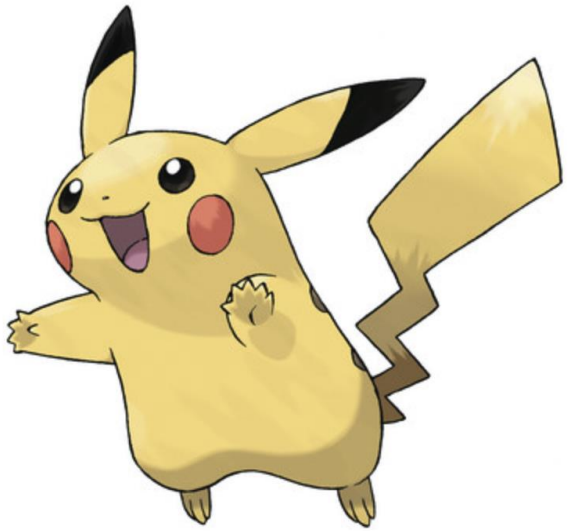
数据简介

- Pokémon Database
- https://pokemondb.net/

#006 Charizard
#025 Pikachu
#094 Gengar
#130 Gyarados
#133 Eevee
#149 Dragonite
#248 Tyranitar
#445 Garchomp
#448 Lucario
#823 Corviknight
#849 Toxtricity
#887 Dragapult

NORMAL FIRE WATER ELECTRIC GRASS
ICE FIGHTING POISON GROUND FLYING
PSYCHIC BUG ROCK GHOST DRAGON
DARK STEEL FAIRY

Pikachu | Partner Pikachu



Pokédex data

National №	025
Type	ELECTRIC
Species	Mouse Pokémon
Height	0.4 m (1′04″)
Weight	6.0 kg (13.2 lbs)
Abilities	1. Static Lightning Rod (hidden ability)
	025 (Yellow/Red/Blue) 022 (Gold/Silver/Crystal) 156 (Ruby/Sapphire/Emerald)

Base stats

			Min	Max
HP	35		180	274
Attack	55		103	229
Defense	40		76	196
Sp. Atk	50		94	218
Sp. Def	50		94	218
Speed	90		166	306
Total	320			

The ranges shown on the right are for a level 100 Pokémon. Maximum values are based on a beneficial nature, 252 EVs, 31 IVs; minimum values are based on a hindering nature, 0 EVs, 0 IVs.

使用 Pandas 与数据库交互

获取并读取数据

```
df = pd.read_csv('https://unicorn.org.cn/valency/src/pokemon-v0.5.27.csv')
df.head()
```

	#	Name	Type 1	Type 2	Total	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legendary
0	1	Bulbasaur	Grass	Poison	318	45	49	49	65	65	45	1	False
1	2	Ivysaur	Grass	Poison	405	60	62	63	80	80	60	1	False
2	3	Venusaur	Grass	Poison	525	80	82	83	100	100	80	1	False
3	3	VenusaurMega Venusaur	Grass	Poison	625	80	100	123	122	120	80	1	False
4	4	Charmander	Fire	NaN	309	39	52	43	60	50	65	1	False

使用 Pandas 与数据库交互

清理数据

```
df.rename(columns={'#': 'id'}, inplace=True)
df.columns = df.columns.str.lower()
df[df.duplicated('id', keep=False)].head()
```

	id	name	type 1	type 2	total	hp	attack	defense	sp. atk	sp. def	speed	generation	legendary
2	3	Venusaur	Grass	Poison	525	80	82	83	100	100	80	1	False
3	3	VenusaurMega Venusaur	Grass	Poison	625	80	100	123	122	120	80	1	False
6	6	Charizard	Fire	Flying	534	78	84	78	109	85	100	1	False
7	6	CharizardMega Charizard X	Fire	Dragon	634	78	130	111	130	85	100	1	False
8	6	CharizardMega Charizard Y	Fire	Flying	634	78	104	78	159	115	100	1	False

使用 Pandas 与数据库交互

清理数据、拆分表格

```
df.drop_duplicates('id', keep='first', inplace=True)
df['type 2'].fillna(value='None', inplace=True)
```

```
pokedex = df[['id', 'name', 'type 1', 'type 2', 'generation', 'legendary']]


statistics = pd.merge(
    df,
    pokedex,
    on='id'
).loc[:, ['id', 'hp', 'attack', 'defense', 'sp. atk', 'sp. def', 'speed', 'total']]
```

使用 Pandas 与数据库交互

拆分表格

```
pokedex.head()
```

	id	name	type 1	type 2	generation	legendary
0	1	Bulbasaur	Grass	Poison	1	False
1	2	Ivysaur	Grass	Poison	1	False
2	3	Venusaur	Grass	Poison	1	False
4	4	Charmander	Fire	None	1	False
5	5	Charmeleon	Fire	None	1	False

使用 Pandas 与数据库交互

拆分表格

```
statistics.head()
```

	id	hp	attack	defense	sp. atk	sp. def	speed	total
0	1	45	49	49	65	65	45	318
1	2	60	62	63	80	80	60	405
2	3	80	82	83	100	100	80	525
3	4	39	52	43	60	50	65	309
4	5	58	64	58	80	65	80	405

统计数据

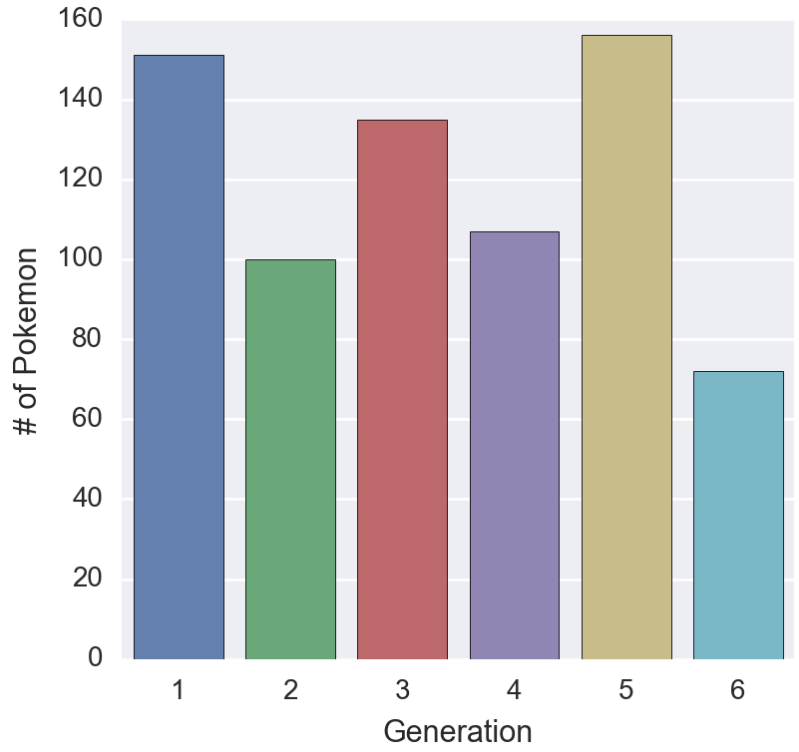
- Seaborn
- https://seaborn.pydata.org/
- Seaborn 是一个基于 Matplotlib 开发，更简单易用的可视化代码库

- `pip3 install --user -U seaborn`
- `import seaborn as sns`

使用 Pandas 与数据库交互

统计数据：不同世代的宝可梦数量

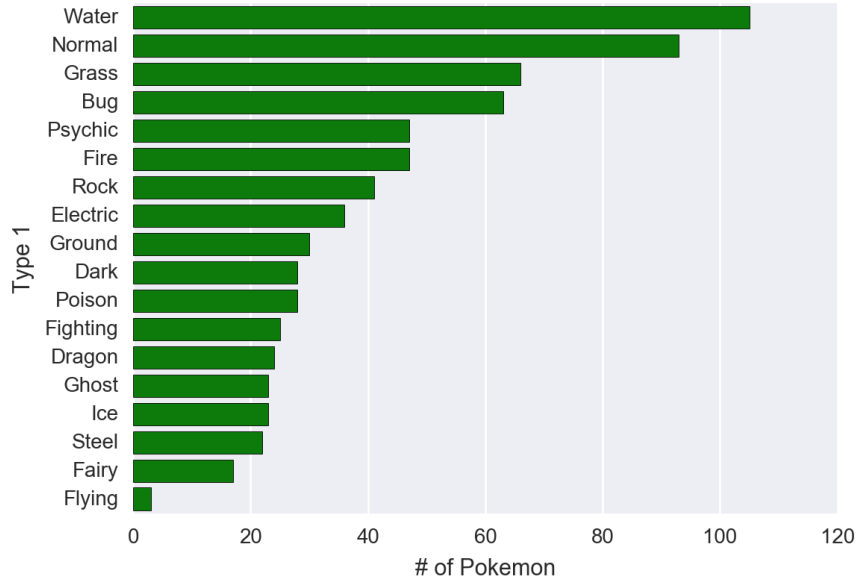
```
sns.factorplot(
    x='generation',
    data=pokedex,
    kind='count'
).set_axis_labels('Generation', '# of Pokemon' );
```



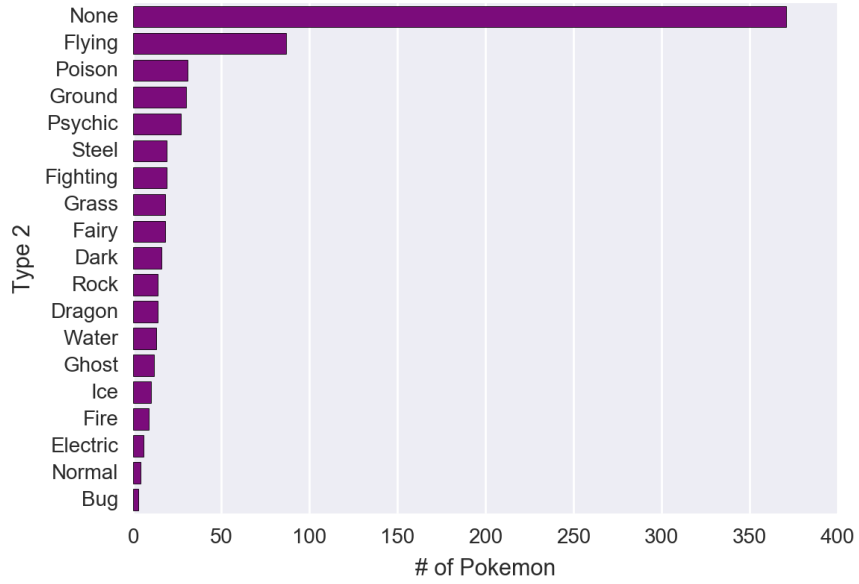
使用 Pandas 与数据库交互

统计数据：最常见的宝可梦属性

```
sns.factorplot(
    y='type 1',
    data=pokedex,
    kind='count',
    order=pokedex['type 1'].value_counts().index,
    aspect=1.5,
    color='green'
).set_axis_labels('# of Pokemon', 'Type 1')
```




```
sns.factorplot(
    y='type 2',
    data=pokedex,
    kind='count',
    order=pokedex['type 2'].value_counts().index,
    aspect=1.5,
    color='purple'
).set_axis_labels('# of Pokemon', 'Type 2');
```

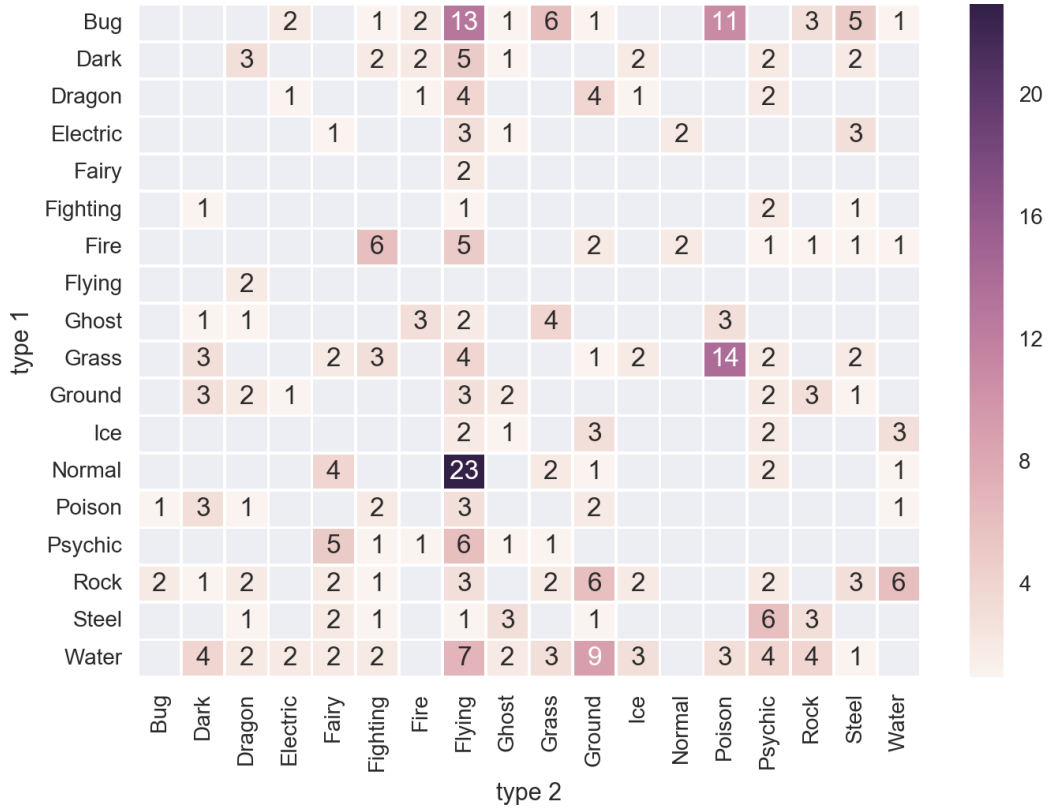


使用 Pandas 与数据库交互

统计数据：具备双重属性的宝可梦分布

```
dual_types = pokedex[pokedex['type 2'] != 'None']


sns.heatmap(
    dual_types.groupby(['type 1', 'type 2']).size().unstack(),
    linewidths=1,
    annot=True
);
```

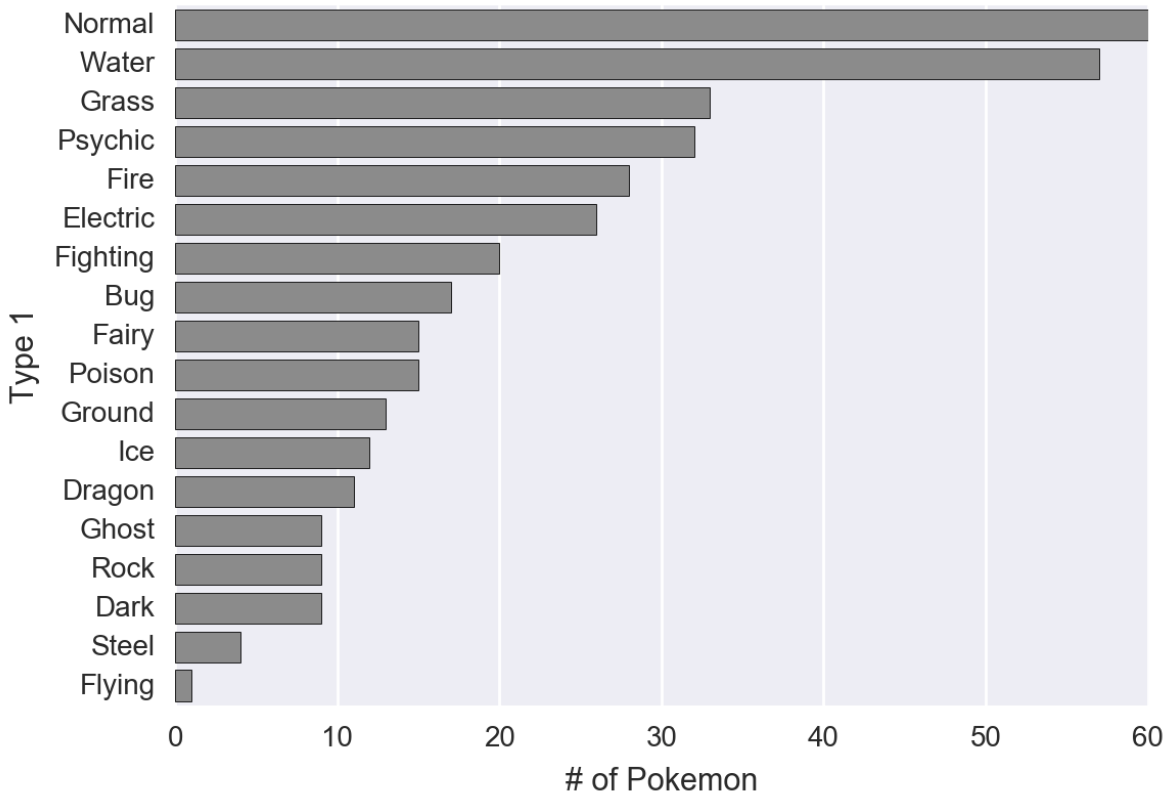


使用 Pandas 与数据库交互

统计数据：没有第二属性的宝可梦的第一属性分布

```
single_types = pokedex[pokedex['type 2'] == 'None']


sns.factorplot(
    y='type 1',
    data=single_types,
    kind='count',
    order=single_types['type 1'].value_counts().index,
    aspect=1.5,
    color='grey'
).set_axis_labels('# of Pokemon', 'Type 1' );
```



作业

❖ 使用 Pandas 从 pokedex 和 statistics 两张表中统计得出前十最强力的宝可梦：

	id	name	type 1	type 2	generation	legendary	hp	attack	defense	sp. atk	sp. def	speed	total
492	493	Arceus	Normal	None	4	True	120	120	120	120	120	120	720
643	644	Zekrom	Dragon	Electric	5	True	100	150	120	120	100	90	680
486	487	GiratinaAltered Forme	Ghost	Dragon	4	True	150	100	120	100	120	90	680
249	250	Ho-oh	Fire	Flying	2	True	106	130	90	110	154	90	680
248	249	Lugia	Psychic	Flying	2	True	106	90	130	90	154	110	680
483	484	Palkia	Water	Dragon	4	True	90	120	100	150	120	100	680
642	643	Reshiram	Dragon	Fire	5	True	100	120	100	150	120	90	680
482	483	Dialga	Steel	Dragon	4	True	100	120	120	150	100	90	680
716	717	Yveltal	Dark	Flying	6	True	126	131	95	131	98	99	680
149	150	Mewtwo	Psychic	None	1	True	106	110	90	154	90	130	680

作业

- 在作业系统中下载并完成本实验课对应实验报告
- https://hw.dgut.edu.cn/
- 注意：所有标识为 * 的地方都需要填写
- 截止日期：2023-12-13 23:59:59

课程名称：Python 数据分析与应用　　学期：20

实验名称	Python 语言回顾			实验序号	
姓　名	***	学　号	***	班　级	
实验地点	***	实验日期	***	指导老师	
教师评语	-			实验成绩	
				百分制	
同组同学	无				

四、　实验作业及分析

4.1　实验过程

1) *** 请将详细实验过程填写在此处 ***

4.2　实验结果

*** 请将实验结果截图填写在此处 ***

五、　实验总结

*** 请撰写一段 200 字左右的实验总结 ***

GOOD LUCK!