山东大学计算机科学与技术学院

大数据分析实践课程实验报告

学号：202300130030	姓名：赵汉哲	班级：数据 23
实验题目：数据质量实践		
实验学时：4	实验日期：2025.10.10	

实验目标：

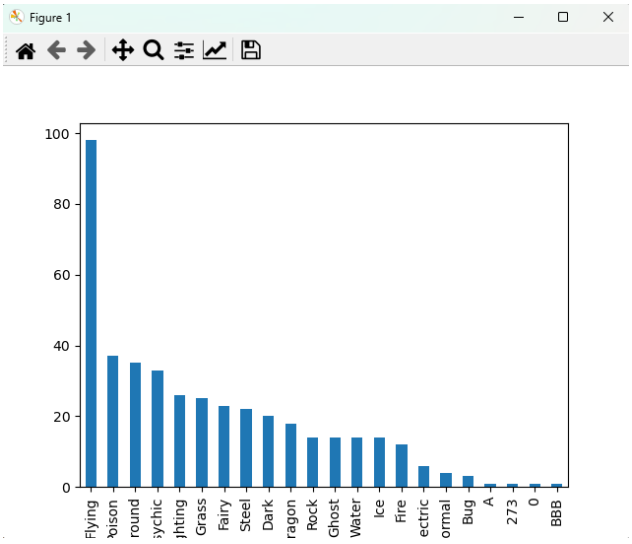
本次实验主要围绕宝可梦数据集进行分析，考察在拿到数据后如何对现有的数据进行预处理清洗操作，建立起对于脏数据、缺失数据等异常情况的一套完整流程的认识。

实验描述：

1. 数据集最后两行数据无意义，可直接删去

```
data = pd.read_csv( filepath_or_buffer: "Pokemon.csv", encoding='latin-1')
data = data.loc[data['#']!='undefined']
data = data.dropna(how='all')
data = data.drop([808,809])
print(data)
```

2. type2 存在异常的数值取值，可清空



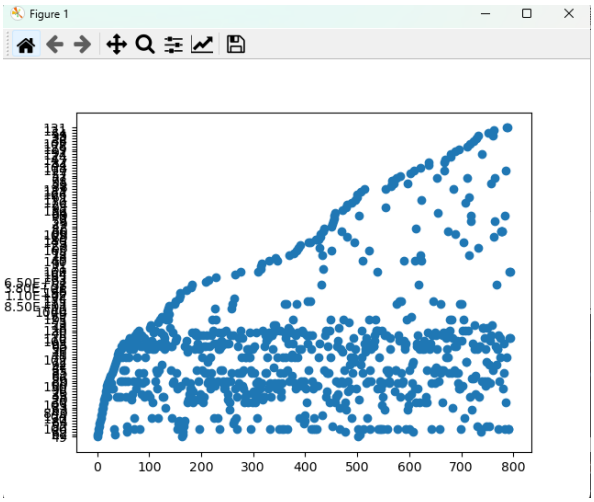

```
data['Type 2'].value_counts().plot(kind='bar')
plt.show()
data = data.loc[data['Type 2']!='A']
data = data.loc[data['Type 2']!='273']
data = data.loc[data['Type 2']!='0']
data = data.loc[data['Type 2']!='BBB']
data['Type 2'].value_counts().plot(kind='bar')
plt.show()
```

3. 数据集中存在重复值
   查询重复值并删除重复的行

```
[805 rows x 13 columns]
       #       Name  Type 1  Type 2  ... Sp. Def Speed Generation Legendary
15    11    Metapod     Bug     NaN  ...      25    30          1     FALSE
23    17  Pidgeotto  Normal  Flying  ...      50    71          1     FALSE
185  168    Ariados     Bug  Poison  ...      60    40          2     FALSE
186  168    Ariados     Bug  Poison  ...      60    40          2     FALSE
187  168    Ariados     Bug  Poison  ...      60    40          2     FALSE
```

```
print(data[data.duplicated()])
data = data.drop([15,23,185,186,187])
print(data[data.duplicated()])
```

4. Attack 属性存在过高的异常值（未发现）



5. 有两条数据的 generation 与 Legendary 属性被置换，并且有几行 Legendary 标注错误
将 generation 与 Legendary 属性置换的互换回去，把 Legendary 标注错误的行删除

```
       #                 Name    Type 1  ... Speed Generation Legendary
11     9            Blastoise     Water  ...    78      FALSE         1
32    25              Pikachu  Electric  ...    90      FALSE         0
45    38            Ninetales      Fire  ...   100          1         0
78    70           Weepinbell     Grass  ...    55          1    Poison
115  105              Marowak    Ground  ...    45          1    Ground
130  119              Seaking     Water  ...    68          1         0
533  475  GalladeMega Gallade   Psychic  ...   110          4       NaN
```

```
print(data.loc[(data['Legendary'] != 'TRUE') & (data['Legendary'] != 'FALSE')])
data = data.drop([78,115,533])
data.loc[11, 'Legendary'] = 'FALSE'
data.loc[11, 'Generation'] = '1'
data.loc[32, 'Legendary'] = 'FALSE'
data.loc[32, 'Generation'] = '0'
data.loc[45, 'Legendary'] = 'FALSE'
data.loc[130, 'Legendary'] = 'FALSE'
print(data.loc[(data['Legendary'] != 'TRUE') & (data['Legendary'] != 'FALSE')])
```

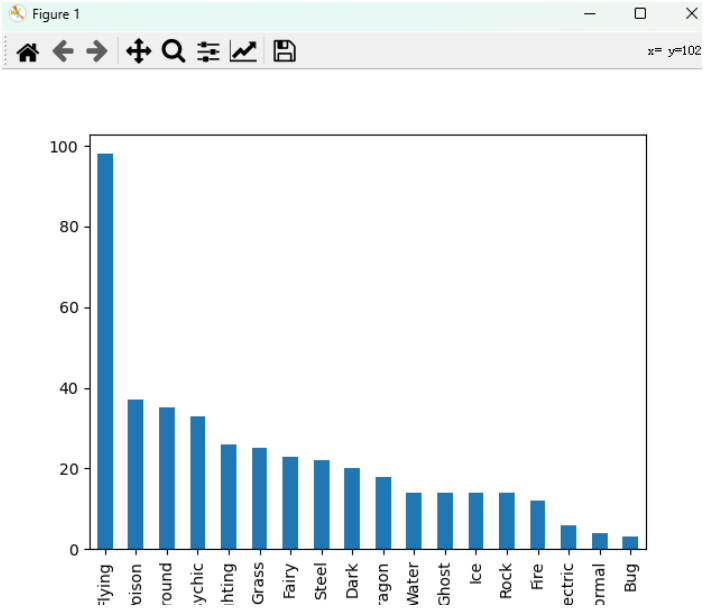
结果图片：

1. 处理后结果：

```
     #                  Name  Type 1  ... Speed Generation Legendary
0    1             Bulbasaur   Grass  ...    45          1     FALSE
1    2               Ivysaur   Grass  ...    60          1     FALSE
2    3              Venusaur   Grass  ...    80          1     FALSE
3    3  VenusaurMega Venusaur   Grass  ...    80          1     FALSE
4    4            Charmander    Fire  ...    65          1     FALSE
..  ...                   ...     ...  ...   ...        ...       ...
801  719               Diancie    Rock  ...    50          6      TRUE
802  719   DiancieMega Diancie    Rock  ...   110          6      TRUE
803  720   HoopaHoopa Confined  Psychic  ...    70          6      TRUE
804  720    HoopaHoopa Unbound  Psychic  ...    80          6      TRUE
805  721             Volcanion    Fire  ...    70          6      TRUE

[805 rows x 13 columns]
```

2. 处理后的 type2 值统计：



3. 处理后查询重复值：

```
Empty DataFrame
Columns: [#, Name, Type 1, Type 2, Total, HP, Attack, Defense, Sp. Atk, Sp. Def, Speed, Generation, Le
Index: []
```

5. 处理后查询 Legendary 的异常值：

```
Empty DataFrame
Columns: [#, Name, Type 1, Type 2, Total, HP, Attack, Defense, Sp. Atk, Sp. Def, Speed, Generation, Le
Index: []
```