实验 3-2 处理空值的属性数据

建议课时：10 分钟

一、 实验目的

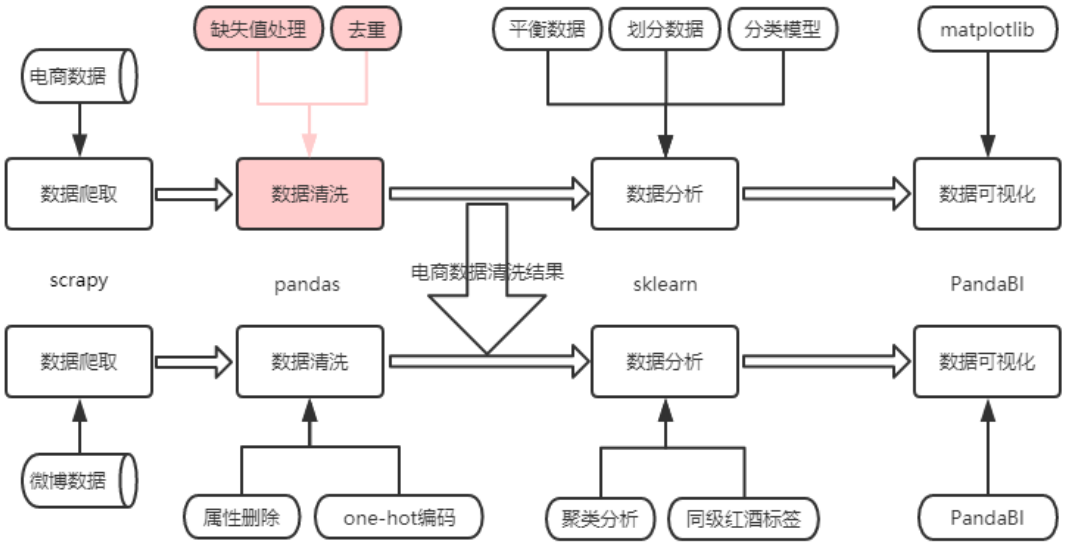
- 了解 pandas 的常用函数
- 了解数据清洗的基础方法

二、 实验环境

Python3 开发环境，第三方包有 pandas

三、 实验步骤

本节处理的内容有：



经过上一小节处理之后，当前留下的属性特征有：

```
df.columns.values
```

```
array(['keyword', 'price', '产区', '保质期', '原产地', '口感', '国产/进口', '存储方法',
       '年份', '特性', '甜度', '类别', '葡萄品种', '酒精度', '颜色'], dtype=object)
```

查看有哪些属性是存在缺失的

```
df.isnull().sum()
```

```
keyword        0
name           0
price          0
产区             0
产品重量（kg）        0
保质期            0
包装             0
原产地            0
口感             0
国产/进口          0
存储方法           0
容量             0
年份             0
特性           117
甜度             0
类别            41
葡萄品种           0
酒精度            0
颜色             0
dtype: int64
```

可见，本节我们需要处理的是特性和类别两个字段，由下图中的聚合语句结果中可发现两个字段都是有限种取值，即离散值，且考虑到同一品牌的红酒的特性、类别极大可能是相同的，因而采用众数替代的方法来处理缺失值，即拿同一品牌的红酒的特性、类别出现最多的值最为替代值。

```
df.groupby("特性").size().sort_values(ascending=False)
```

```
特性
普通餐酒               6123
精品葡萄酒              1563
列级庄                1455
法定产区酒（AOC/AOP等）      958
名庄葡萄酒               586
中级庄                578
酒杯/酒具              167
有机葡萄酒               76
AOC/AOP             25
dtype: int64
```

众数替代代码如下：

```
# 众数替代缺失值

#上述统计结果可以看出“特性”和“类别”出现缺失值
def process_nan(col,gp_col,df):
    #计算该列分组众数，可能出现某个品牌的众数为nan，以“no match”代替
    df_mode = df.groupby(gp_col)[col].agg(lambda x: next(iter(x.value_counts().index), 'no match'))
    df[col] = df[col].fillna(df[gp_col].map(df_mode))
    df = df[~df[col].isin(["no match"])] #删除“no match”
    return df

df = process_nan("特性", "keyword", df)
df = process_nan("类别", "keyword", df)

# 保存数据
df.to_csv("pre-processed.csv",encoding="utf-8_sig",index = False)
```

如上述代码所示，我们保存下来 pre-processed.csv 文件，下节讲述利用 pandaBI 观察数据再对属性作调整。