实验一 数据预处理

10215501412 彭一珅

数据质量问题

1.缺失值

较差的数据质量可能会对数据挖掘产生不利影响。常见的数据质量问题包括噪声、异常值、缺失值和重复数据。

数据：breast-cancer-wisconsin.data

```
数据集实例个数：699
属性个数：11
```

- 缺失值在数据集中编码为“?”，将缺失值转换为NaN，并计算每列数据中缺失值的数量
  data.replace('?',np.NaN)

```
每列中缺失值的数量：
Sample code                    0
Clump Thickness                0
Uniformity of Cell Size        0
Uniformity of Cell Shape       0
Marginal Adhesion              0
Single Epithelial Cell Size    0
Bare Nuclei                   16
Bland Chromatin                0
Normal Nucleoli                0
Mitoses                        0
Class                          0
dtype: int64
```

- 将缺失值替换为该列的中值：fillna()

```
# 将缺失值替换为该列的中值
data2=data1.fillna(data1.median())
```

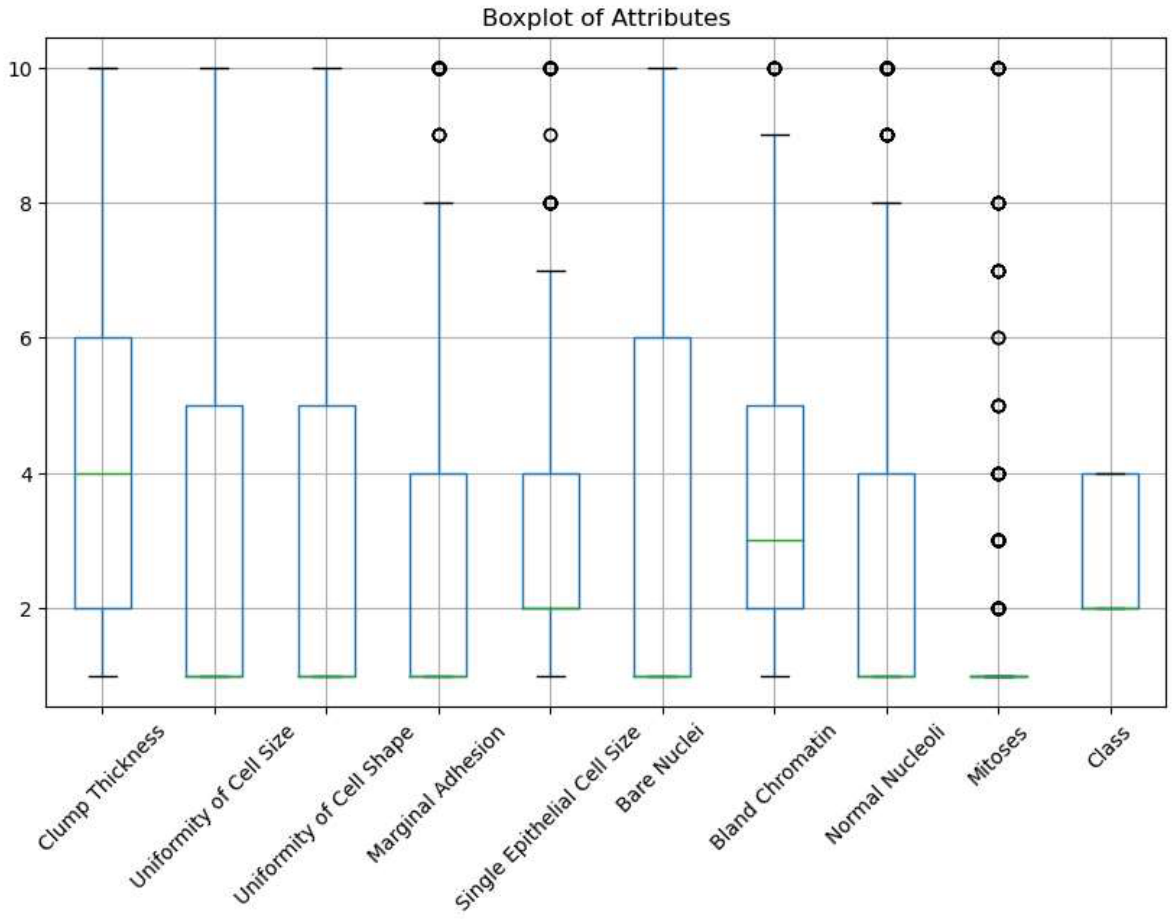
- 丢弃包含缺失值的数据点：dropna()

```
数据集实例个数：683
```

2.异常值

- 通过绘制boxplot来识别数据中包含异常值的列：boxplot()

  由于“Bare Nuclei”列中的值存储为字符串对象，应该先将该列转换为数值：pandas.to_numeric



- 计算每个属性的Z分数，并删除那些包含Z分数异常高或异常低的属性的实例 (例如，if Z>3 or Z<=-3)

  Z = (data-data.mean())/data.std()

  Z 中筛选出每行恰好有9个（属性个数）元素大于 -3 且恰好有9个元素小于等于 3 的行

```
selected_rows = Z_scores[(Z_scores > -3).sum(axis=1) == 9]  # 大于 -3 的元
selected_rows2 = Z_scores[(Z_scores <= 3).sum(axis=1) == 9]  # 小于等于 3
intersection_index=selected_rows.index.intersection(selected_rows2.index)
```

3.重复数据

- 检查数据中的重复样本：duplicated()

```
重复样本个数：236
```

- 删除重复行：drop_duplicates()

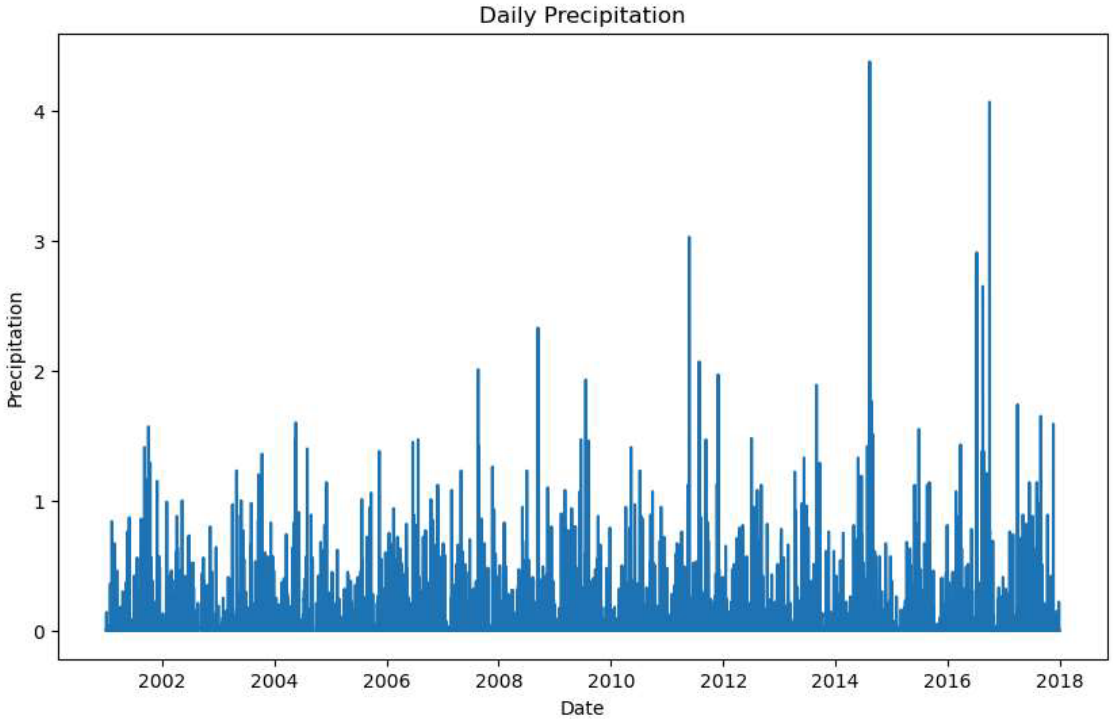
```
样本量：699 463
```

数据聚合

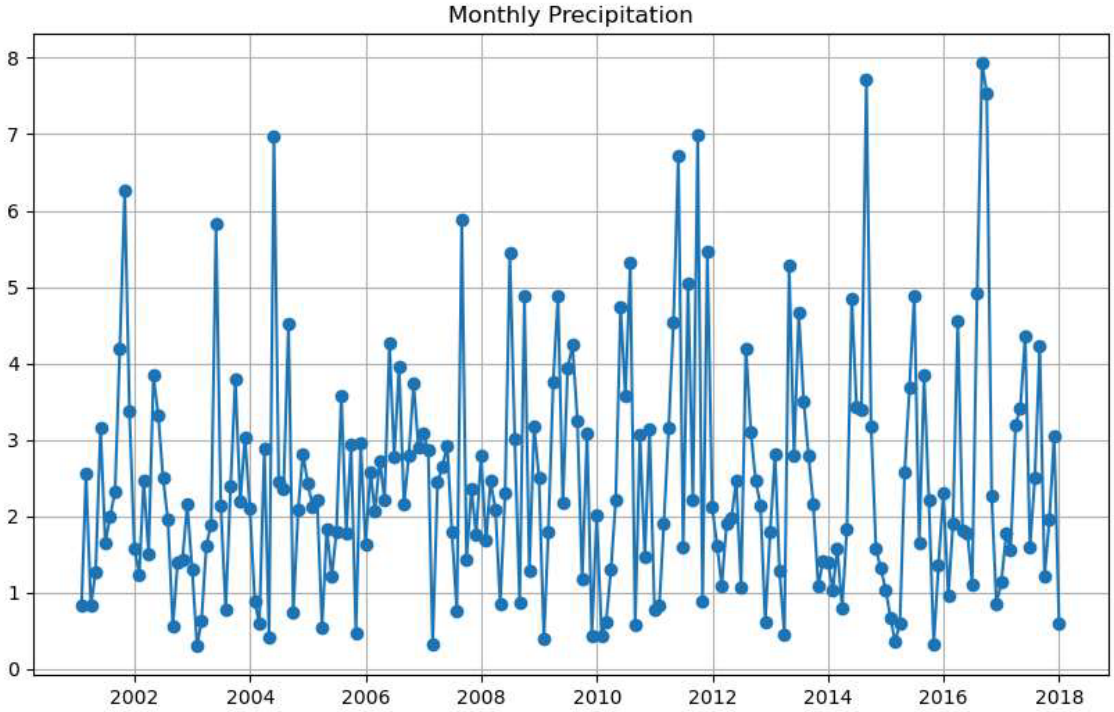
目的：

- 减小要处理的数据的大小；
- 改变分析的粒度（从细粒度到粗粒度）；
- 提高数据的稳定性

数据的方差：0.05304985960911706

- 绘制其每日时间序列的折线图
- 

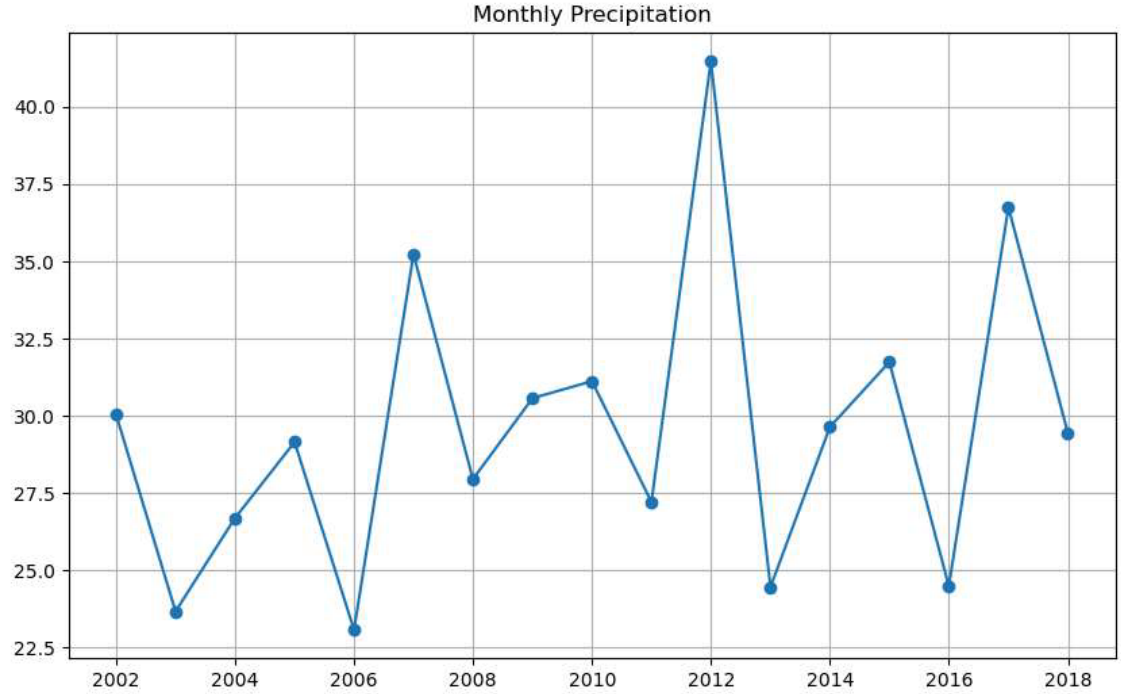
- 绘制其每月时间序列的折线图



pandas.Grouper

DataFrame.groupby()

- 绘制其每年时间序列的折线图



采样

- 从原始数据中随机选择（不替换）大小为3的样本

	Sample code	Clump Thickness	Uniformity of Cell Size	\
407	1234554	1	1	
215	1222936	8	7	
457	1259008	8	8	

	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size	\
407	1	1	2	
215	8	7	5	
457	9	6	6	

	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	Class
407	1	2	1	1	2
215	5	5	10	2	4
457	3	10	10	1	4

- 随机选择1%的数据（不替换）并显示所选样本

	Sample code	Clump Thickness	Uniformity of Cell Size	\
137	1182410	3	1	
463	1280258	4	1	
201	1216694	10	8	
589	1272166	5	1	
486	1070522	3	1	
114	1173235	3	3	
469	1181685	1	1	

	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size	\
137	1	1	2	
463	1	1	2	
201	8	4	10	
589	1	1	2	
486	1	1	1	
114	2	1	2	
469	2	1	2	

	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	Class
137	1	1	1	1	2
463	1	1	2	1	2
201	10	8	1	1	4
589	1	1	1	1	2
486	1	2	1	1	2
114	3	3	1	1	2
469	1	2	1	1	2

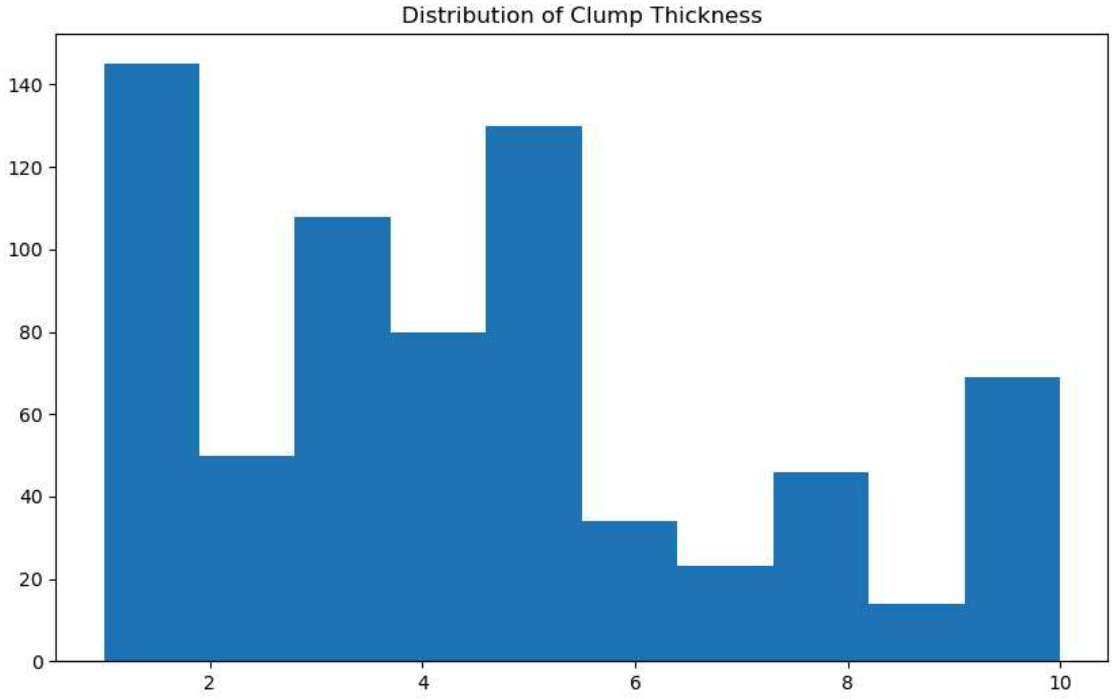
- （替换）采样1%的数据

	Sample code	Clump Thickness	Uniformity of Cell Size	\
62	1116116	9	10	
66	1117152	4	1	
594	1315506	4	8	
655	1326892	3	1	
630	1225382	6	2	
455	1246562	10	2	
52	1110102	10	3	

	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size	\
62	10	1	10	
66	1	1	2	
594	6	3	4	
655	1	1	2	
630	3	1	2	
455	2	1	2	
52	6	2	3	

	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	Class
62	8	3	3	1	4
66	1	3	1	1	2
594	10	7	1	1	4
655	1	2	1	1	2
630	1	1	1	1	2
455	6	1	1	2	4
52	5	4	10	2	4

离散化

- 使用Counter()对Clump Thickness属性的取值进行计数
-
```
Counter({5: 130,
         3: 108,
         6: 34,
         4: 80,
         8: 46,
         1: 145,
         2: 50,
         7: 23,
         10: 69,
         9: 14})
```
- 绘制直方图，显示属性值的分布
- 

- equal width：应用pandas.cut将属性离散为4个间隔宽度相似的bin，value_counts()用于确定每个bin中的实例数
-
```
Clump Thickness
(0.99, 3.25]     303
(3.25, 5.5]      210
(7.75, 10.0]     129
(5.5, 7.75]       57
Name: count, dtype: int64
```

为什么第一个bin区间的左端点是0.991？如果想要将左端点变成0.99，怎么改?

因为pd.cut的作用是获取左开右闭区间，而最小值是否包含在区间内由选项include_lowest决定。

```
bins = pd.cut(data['Clump Thickness'], bins=4) #加入include_lowest=True可以得到0.99
```

- equal frequency：qcut()函数可用于将值划分为4个bin，以便每个bin具有几乎相同数量的实例

```
Clump Thickness
(0.999, 2.0]     195
(2.0, 4.0]       188
(4.0, 6.0]       164
(6.0, 10.0]      152
Name: count, dtype: int64
```

主成分分析（PCA）

PCA是一种通过将数据从其原始高维空间投影到低维空间来减少数据中属性数量的经典方法。

PCA创建的新属性具有以下特点：

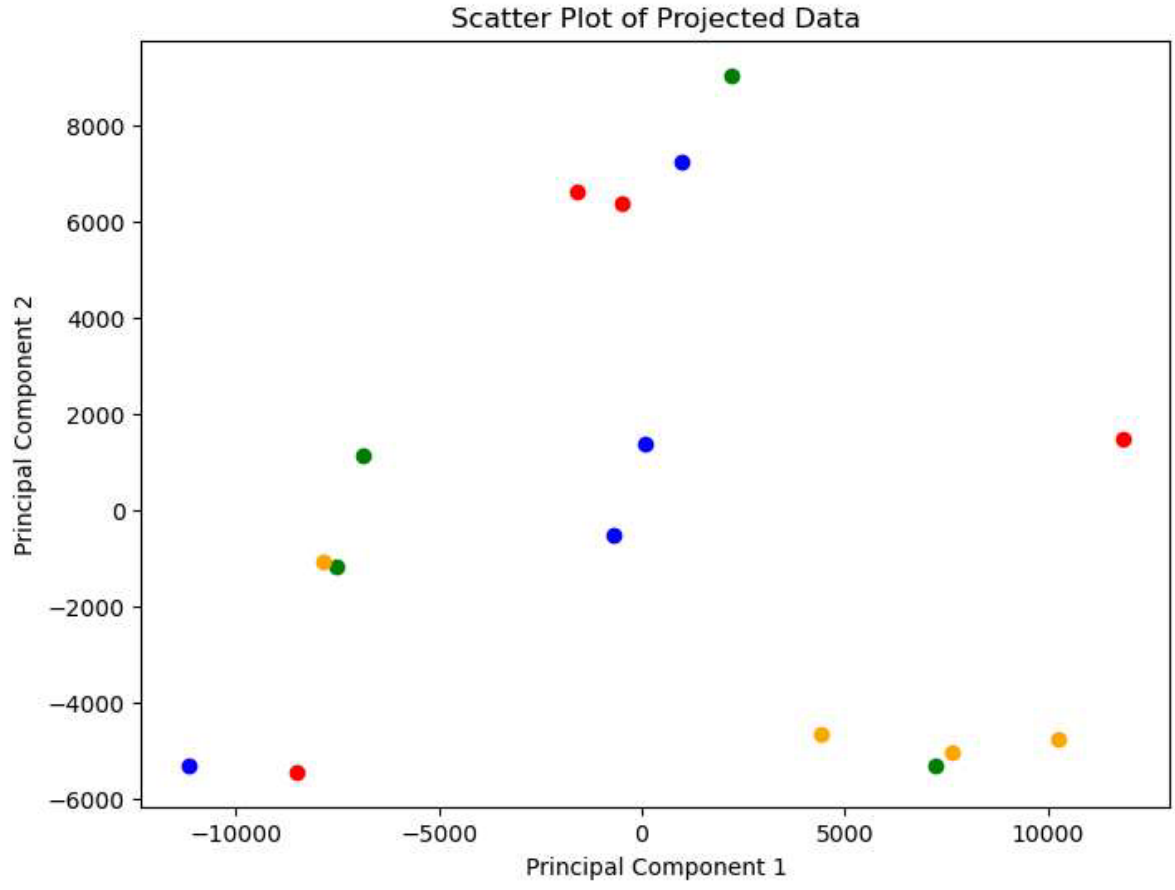
- 它们是原始属性的线性组合；
- 它们彼此正交（垂直）；
- 它们捕获数据中的最大变化量。

数据：pics文件夹下包含16个RGB图像文件，每个文件的大小为111×111像素

- 读取图像数据，将RGB图像转换为111×111×3=36963个特征值，最终得到一个16×36963的矩阵

$$(16,\ 36963)$$

- 使用主成分分析，将数据矩阵投影到其前两个主成分。无需编写PCA代码，直接导入sklearn.decomposition中的PCA类
- 绘制散点图来显示投影值



其中红、蓝、绿、黄这四种颜色分别对应汉堡、可乐、面、鸡腿的图片。