海藻数据的分析

1. 问题描述

某些高浓度的有害藻类对河流生态环境的破坏是一个严重的问题。它们不仅破坏河流的生物，也破坏水质。能够监测并在早期对海藻的繁殖进行预测对提高河流质量是很有必要的。

针对这一问题的预测目标，在大约一年的时间内，在不同时间内收集了欧洲多条河流的水样。对于每个水样，测定了它们的不同化学性质以及 7 种有害藻类的存在频率。在水样收集过程中，也记录了一些其他特性，如收集的季节、河流大小和水流速度。

2. 数据说明

数据：Analysis.txt

有 200 个水样，每条记录是同一条河流在该年的同一个季节的三个月内收集的水样的平均值。

每条记录由 11 个变量构成，3 个是标称变量，分别描述水样收集的季节，河流大小和河水速度，剩下的 8 个变量是水样的化学参数：

- 最大 pH 值(mxPH)
- 最小含氧量(mnO2)
- 平均氯化物含量(Cl)
- 平均硝酸盐含量(NO3)
- 平均氨含量(NH4)
- 平均正磷酸盐含量(oPO4)
- 平均磷酸盐含量(PO4)
- 平均叶绿素含量(Chla)

a1-a7 为 7 种不同有害藻类在相应水样中的频率数目。

3. 数据分析要求

3.1 数据可视化和摘要

数据摘要

- 对标称属性，给出每个可能取值的频数，
- 数值属性，给出最大、最小、均值、中位数、四分位数及缺失值的个数。

首先将原数据文件转化为.csv 格式，利用 python 对其进行数据读取和统计如上信息，结果如下：

```
season 的频数为：
winter    62
spring    53
summer    45
autumn    40
```

```
river_size 的频数为：
medium    84
small     71
large     45
```

```
river_speed 的频数为：
high      84
medium    83
low       33
```

	max	min	mean	median	quartile	missing
mxPH	9.70000	5.600	8.011734	8.0600	7.70000	1.0
mnO2	13.40000	1.500	9.117778	9.8000	7.72500	2.0
Cl	391.50000	0.222	43.636279	32.7300	10.98125	10.0
NO3	45.65000	0.050	3.282389	2.6750	1.29600	2.0
NH4	24064.00000	5.000	501.295828	103.1665	38.33325	2.0
oPO4	564.59998	1.000	73.590596	40.1500	15.70000	2.0
PO4	771.59998	1.000	137.882101	103.2855	41.37525	2.0
Chla	110.45600	0.200	13.971197	5.4750	2.00000	12.0
a1	89.80000	0.000	16.923500	6.9500	1.50000	0.0
a2	72.60000	0.000	7.458500	3.0000	0.00000	0.0
a3	42.80000	0.000	4.309500	1.5500	0.00000	0.0
a4	44.60000	0.000	1.992500	0.0000	0.00000	0.0
a5	44.40000	0.000	5.064500	1.9000	0.00000	0.0
a6	77.60000	0.000	5.964000	0.0000	0.00000	0.0
a7	31.60000	0.000	2.495500	1.0000	0.00000	0.0

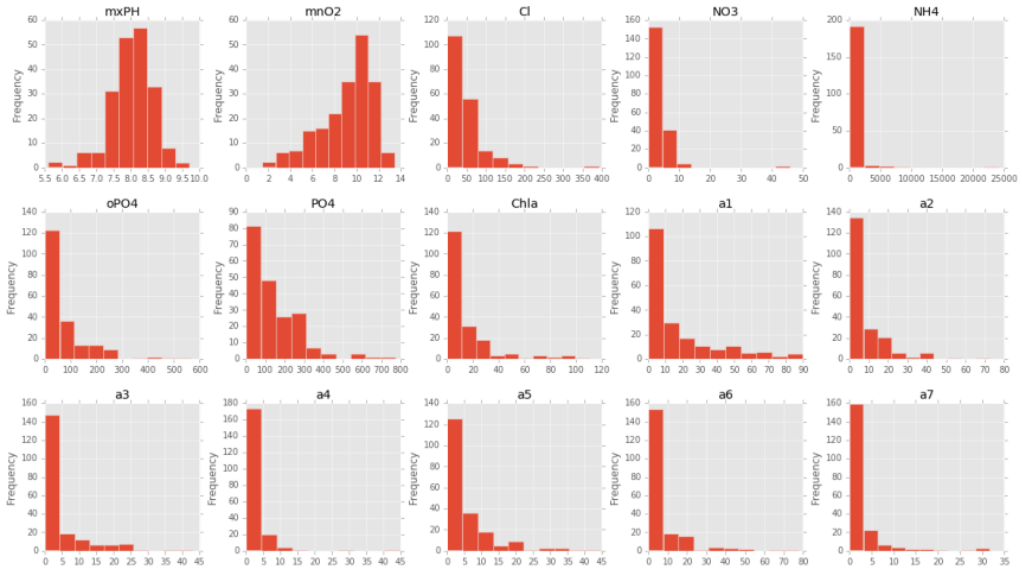
数据的可视化

针对数值属性，

- 绘制直方图，如 mxPH，用 qq 图检验其分布是否为正态分布。
- 绘制盒图，对离群值进行识别

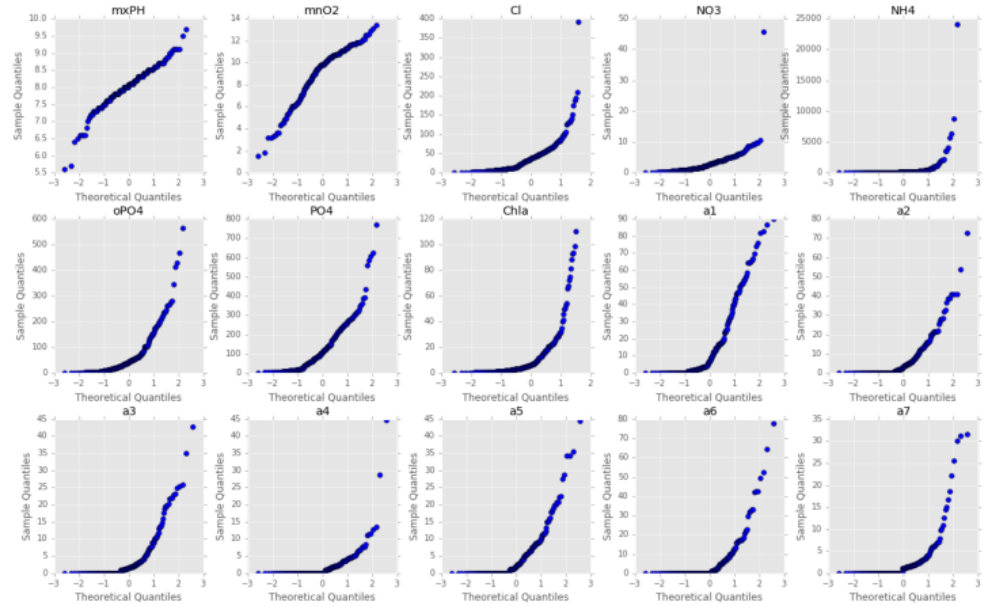
对 7 种海藻，分别绘制其数量与标称变量，如 size 的条件盒图。

直方图绘制结果如下：



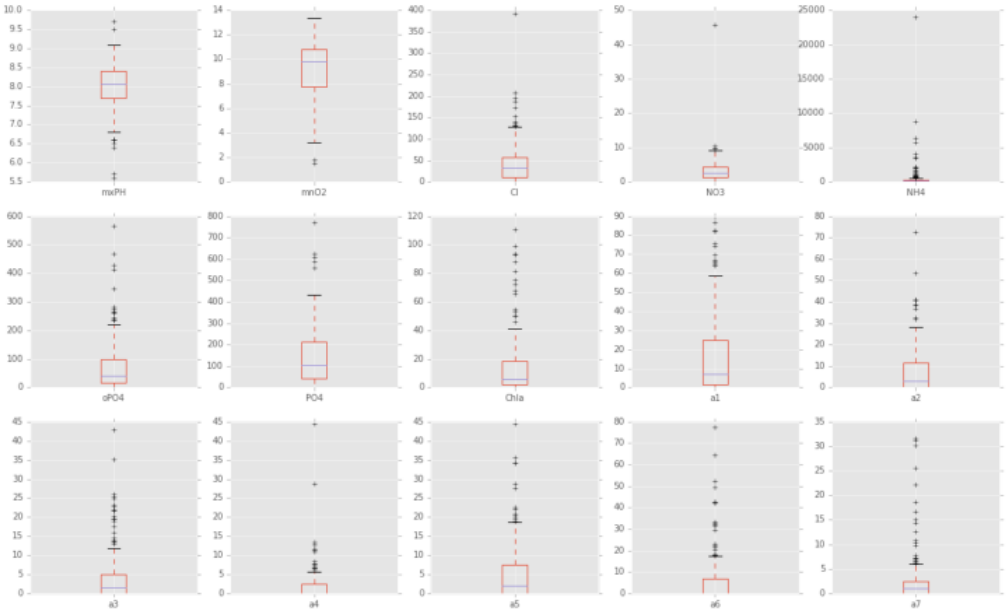
如上图所示，横轴是分布区间，纵轴是频数；

qq 图绘制结果如下：

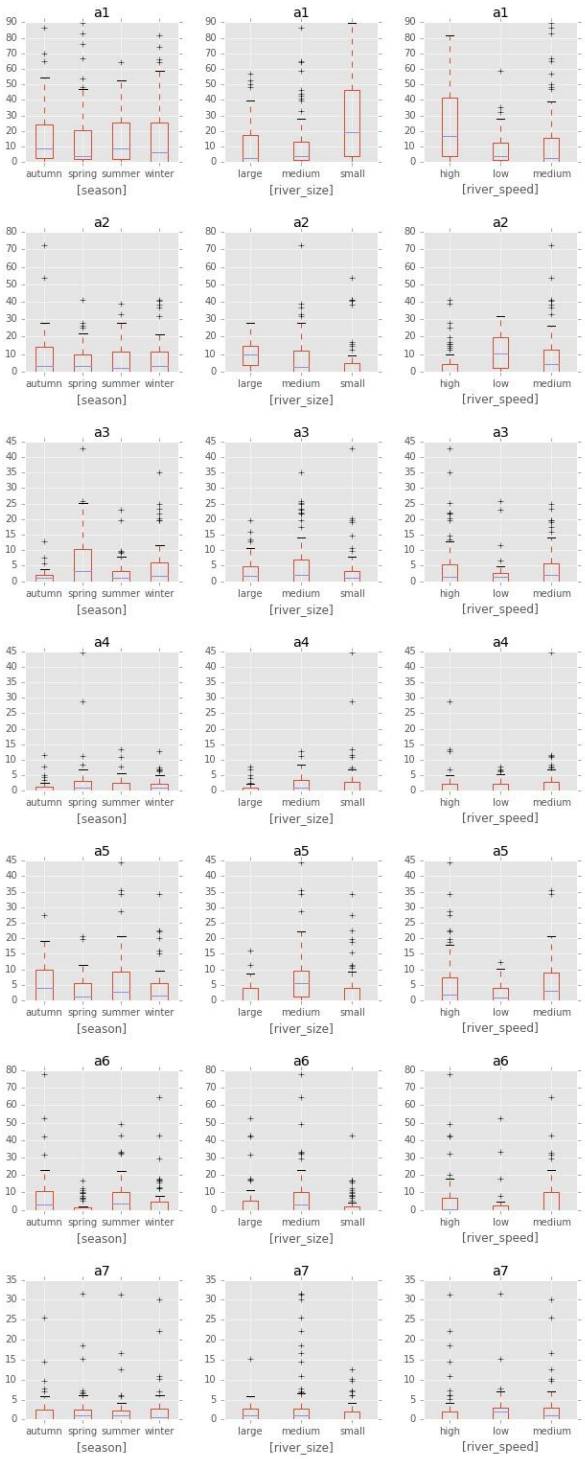


qq 图中的数据分布显示，mxPH 和 mnO2 的分布接近对角线，因此其分布接近于正态分布，而其它则相差较远。

数值属性的盒图绘制结果如下：



对 7 种海藻分别绘制其数量与 size 的条件盒图，结果如下：

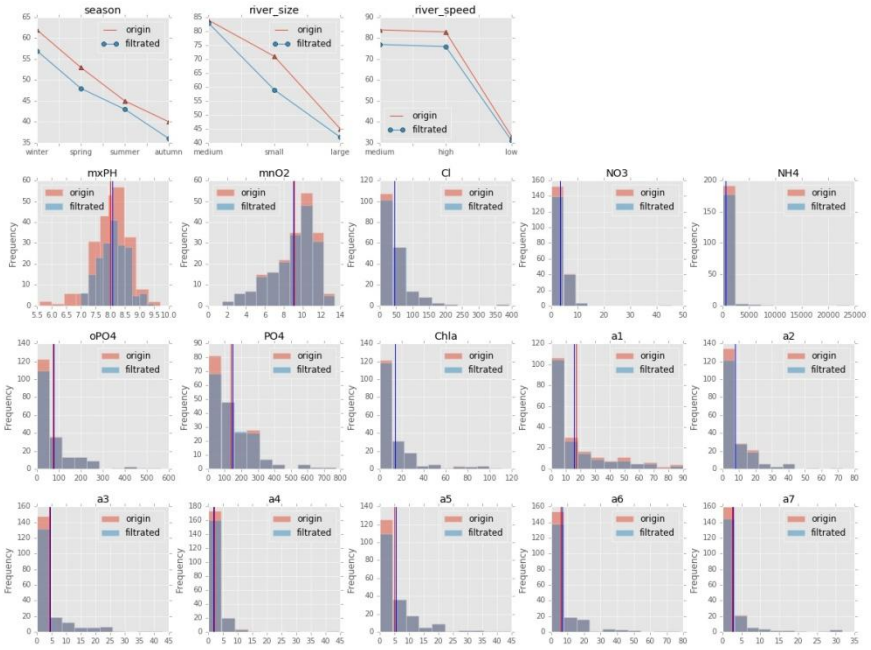

Boxplot grouped by river_speed
a1
a2
a3
a4
a5
a6
a7
autumn spring summer winter
[season]
large medium small
[river_size]
high low medium
[river_speed]

3.2 数据缺失的处理

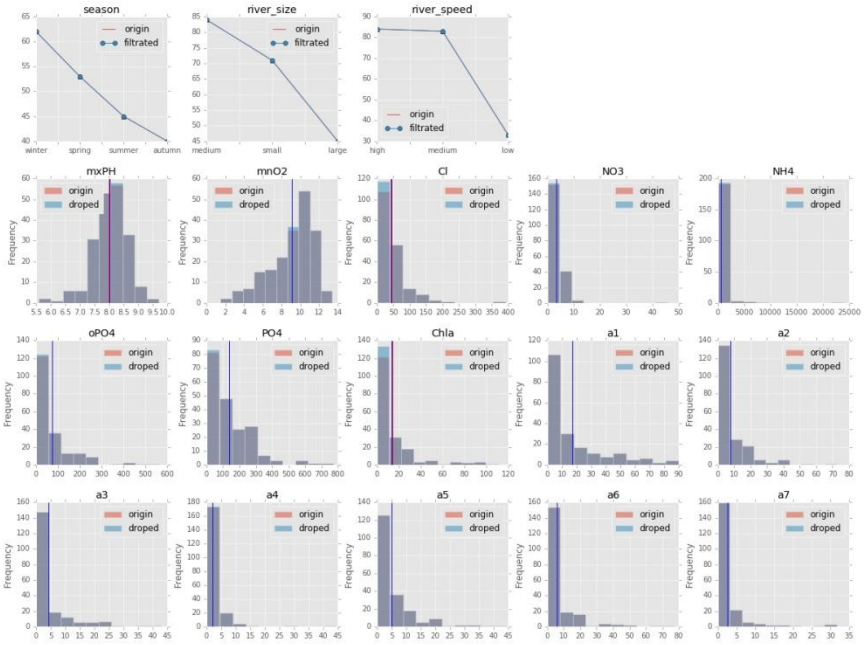
分别使用下列四种策略对缺失值进行处理:

处理后，可视化地对比新旧数据集。

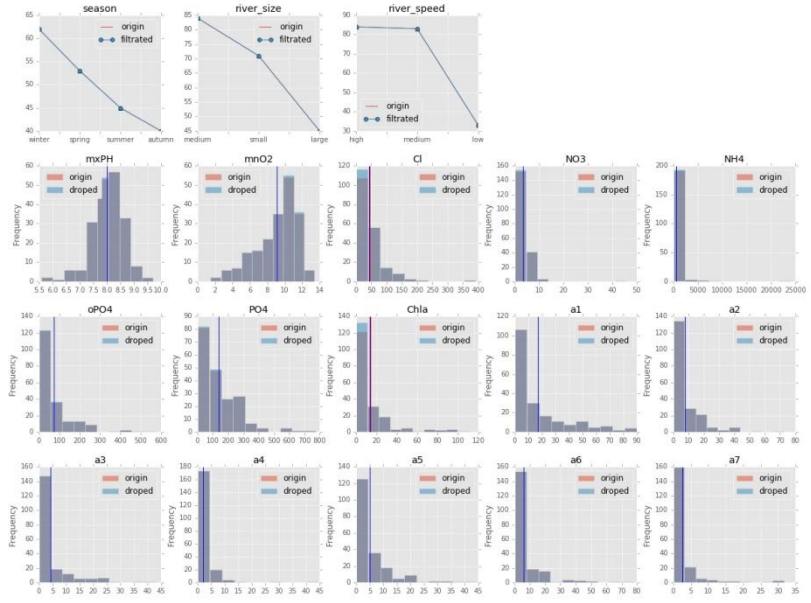
- 首先使用 dropna()函数将缺失数据剔除：



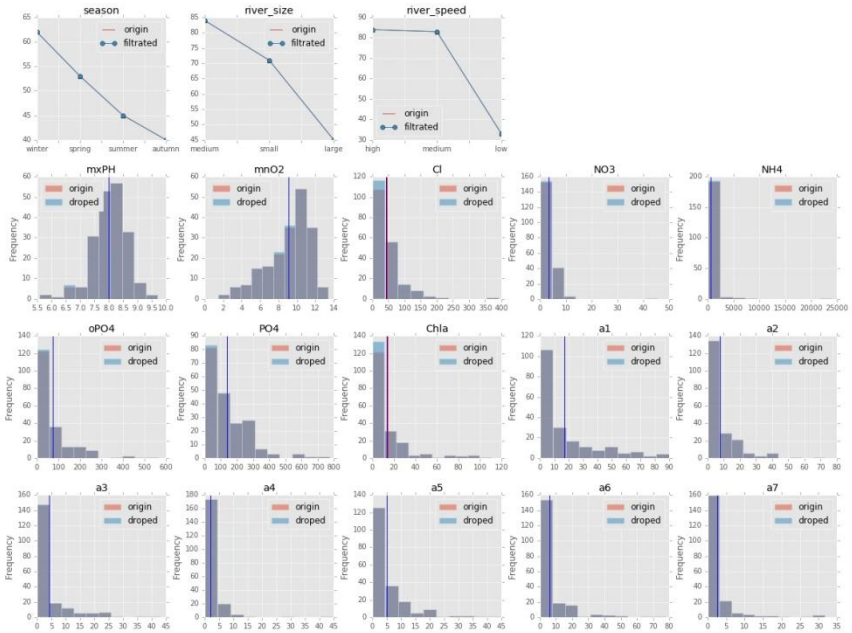
- 用最高频率值来填补缺失值：



- 通过属性的相关关系来填补缺失值：



- 通过数据对象之间的相似性来填补缺失值：


season
river_size
river_speed
origin
filtrated
winter
spring
summer
autumn
medium
small
large
high
low
mxPH
mnO2
Cl
NO3
NH4
oPO4
PO4
Chla
a1
a2
a3
a4
a5
a6
a7
origin
droped
Frequency