


中国研究生创新实践系列大赛

“华为杯”第十八届中国研究生

数学建模竞赛

学　　校	北京邮电大学
参赛队号	21100130067
队员姓名	1. 曹世翔
	2. 刘佳恒
	3. 潘航

中国研究生创新实践系列大赛

“华为杯”第十八届中国研究生

数学建模竞赛

题 目 空气质量二次预报模型的设计与优化

摘 要：

近年来，大气污染问题日趋严重，对人民的生命健康和地球的生态环境造成了严重危害。建立空气质量预报模型是提高环境空气质量的有效手段，但由于气象场数据和污染排放数据具有不确定性，与此同时科研人员也不能完全明晰各污染物的生成机理，现行常用的 WRF-CMAQ 预报模型的预报结果准确度有限。因此，如何在 WRF-CMAQ 模型一次预报结果的基础上，综合更多数据源进行二次建模，以实现预报准确度的提高，成为当下空气污染物防治领域内的研究重点之一。

本文针对空气质量二次预报模型的设计与优化进行了研究，以污染物浓度、AQI 等作为指标，通过相关性分析和聚类处理得到影响指标的关键属性，并分析了不同气象条件特征下的污染物浓度变化，为数据降维创造条件，进而构建天气质量二次预报模型，优化了原一次预报相对误差较大的问题。之后在上述独立预报模型的基础上引入监测点间直线距离、风力影响距离等变量，充分利用邻近地区的位置与天气条件的相关性，构建二次预报的协同预测模型，进一步增强了模型预测准确度，提升了天气质量预报模型的价值。

对于问题一，依据附录中 AQI（即空气质量指数）计算与评价的方法，使用监测点 A 长期空气质量预报基础数据中的污染物浓度每日实测数据，计算自 2020 年 8 月 25 日到 8 月 28 日期间，监测点 A 每日实测的 AQI 和首要污染物，并得出这四天的空气质量等级。

对于问题二，通过拉格朗日插值、箱线图等方法处理原数据集中的缺失值与异常值，构建能够描述监测点 A 一个时间周期内气象条件变化量与各污染物浓度变化量的新数据集，并对气象条件与各污染物浓度做相关性分析。使用基于 EM 算法的高斯混合模型对新数据集进行聚类处理，并使用 t-SNE 降维可视化技术展示聚类结果。结果共分为 6 类，分别对应 6 种气象条件对污染物的扩散或沉降造成的影响。结合相关性分析结果与气象相关文献，对聚类结果进行解读，并阐述各类气象条件的特征。

对于问题三，横向合并监测点 A、B、C 数据集，使用问题二中的方法处理缺失值与异常值。将预处理后的数据集拆分为三个数据集，据此建立三个基于 XGBoost 算法的二次预报预测模型，分别用于预测未来第一天数据、未来第二天数据、未来第三天数据，每个模型由六个子模型组成，对应每种污染物浓度值的预测。通过数据降维、参数调优等方法提高模型精度。通过与一次预报模型的对比与任意选取数据测试，证明二次预报模型的优越性，在对臭氧浓度、AQI 以及首要污染物的预测上体现的尤为明显。最后，使用二次预报模型预测了 A、B、C 三点在 2021 年 7 月 13 日至 7 月 15 日 6 种常规污染物的单日浓度值，计算出相应的 AQI 和首要污染物。

对于问题四，纵向合并监测点 A、A1、A2、A3 的数据集，使用问题二、三中的数据预处

理方法对数据初步处理。根据平面上四监测点的相对位置关系，计算它们之间的直线距离，将风向变量转化为更容易度量的风力影响距离变量——每个监测点与其余三个监测点的风向的垂直距离，使得监测点附近地区的风力、温度、湿度等天气影响因子能更准确地作用于协同预报模型之中；并在问题 3 中的 XGBoost 独立二次预报预测模型的基础上进行优化，构建四个协同预报子模型，分别用来预测监测点 A、A1、A2、A3 的污染物浓度值以及相应的 AQI 和首要污染物。经过数据测试，本文证实了所建立的协同预报模型相比于独立模型具有更高的准确性，同时说明了协同预报模型能够提升针对监测点 A 的污染物浓度预报准确度。
关键词：空气质量二次预报；EM 算法；GMM 模型；t-SNE；XGBoost 算法；协同预报

目 录

1. 问题重述

1.1 课题背景

近年来，随着工业生产的发展、能源消耗的提高以及城市人口的迅速增长，大气污染问题日趋严重，这给人们的生命健康和生产生活带来了严重威胁，也对地球的生态环境造成了严重危害。因此在大力发展生产力的同时，我们迫切需要保护和改善大气环境。依据有关部门的污染防治实践，建立空气质量预报模型是提高环境空气质量的有效手段，对水资源、农业、交通运输业等多方面都具有重要的影响。

目前针对空气质量预报这一亟待解决的问题，相关领域内的研究人员已经提出了一些方法，例如现行常用的 WRF-CMAQ 预报模型。WRF-CMAQ 预报模型主要包括 WRF 系统和 CMAQ 系统，依据 WRF 系统提供的气象场数据，以及场域内的污染排放数据，CMAQ 系统模拟多种污染物的物理和化学反应过程，预报某时间段内或某个具体时间点的空气质量情况。但由于气象场数据和污染排放数据具有不确定性，与此同时科研人员也不能完全明晰各污染物（例如臭氧）的生成机理，WRF-CMAQ 模型的预报结果无法满足人们的实际需求。因此，我们考虑在 WRF-CMAQ 模型一次预报结果的基础上，综合更多数据源进行二次建模，实现预报准确度的提高。

《环境空气质量标准》（GB3095-2012）指出，我国六种主要常规大气污染物分别为：二氧化硫（SO_2）、二氧化氮（NO_2）、粒径小于 10 μm 的颗粒物（PM_{10}）、粒径小于 2.5 μm 的颗粒物（$PM_{2.5}$）、臭氧（O_3）和一氧化碳（CO）。根据学术研究及工业实践情况，上述六种污染物浓度实测数据在一段时间内的变化情况，会对未来空气质量预报提供一定的参考价值。此外，空气质量还受到区域内实际气象条件，例如当地温度湿度、是否刮风、是否下雨等条件的影响。

而在所有大气污染问题中，由于燃料（例如石油和煤炭等）消耗量的剧增，臭氧前体物的排放量不断增加，导致臭氧污染问题显得尤为突出。作为上述六种污染物中唯一的二次污染物和危害最严重的二次污染物之一，臭氧除了影响各种农作物及植被的正常生长发育，还会造成云雨水的酸化，进而导致严重的酸雨现象，危害人们的生命健康安全。与此同时，臭氧并非污染源直接排放的原始污染物质（即一次污染物），而是经过一系列复杂的化学或光化学反应生成的，使用 WRF-CMAQ 模型一次预报臭氧浓度变化的精确度不高。

综合来看，空气质量的预报受到众多因素的影响。在认真分析 WRF-CMAQ 模型一次预报的结果上，我们考虑依据空气质量监测点获得的气象与污染物实测数据，使用聚类、相关性分析等方法，尽可能准确地筛选出具备较大参考价值的影响因素；并综合一次预报数据和实测数据的分析结果，使用 XGBoost 等方法，合理建立二次预报模型，优化原有的一次预报模型。与此同时，鉴于臭氧污染防治在大气污染物预警与防治中的重要地位，且预报难度较高，我们还需要重点考虑提高臭氧预报准确度的问题。优化后的二次预报模型与 WRF-CMAQ 模型关系如图 1-1 所示：

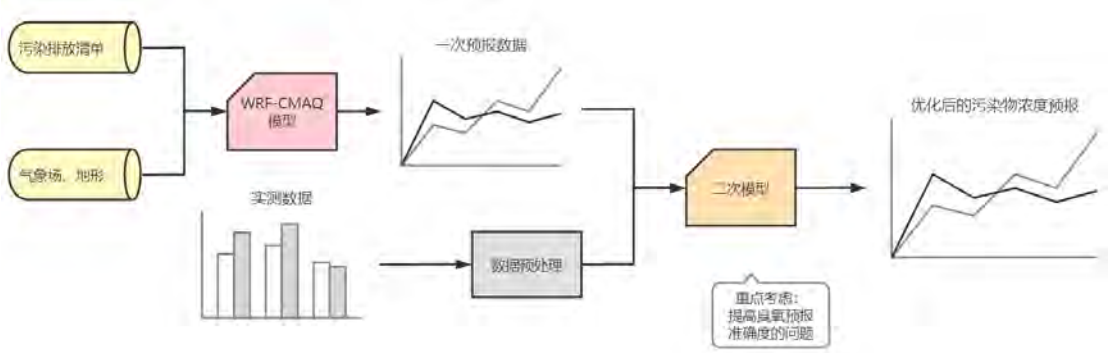


图 1-1 二次预报模型与 WRF-CMAQ 模型关系图

1.2 问题提出

基于上述研究背景，本文需研究和解决如下问题：

问题 1：数据计算

依据附录中的方法，使用附件 1 提供的数据，计算自 2020 年 8 月 25 日到 8 月 28 日期间，监测点 A 每日实测的 AQI 和首要污染物。并按照附录中“AQI 计算结果表”的格式，将计算结果放在正文中。

问题 2：气象条件分类

在某区域内污染物排放情况一定的前提条件下，当气象条件有利于污染物扩散时，该地区的 AQI 会下降；而当气象条件有利于污染物沉降时，该地区的 AQI 会上升。根据不同影响因素对污染物浓度影响程度的不同，需对附件 1 中的数据进行一定的处理，以实现对气象条件的合理分类，并结合实际情况阐述各类气象条件的特征。原始数据中包括一定程度的脏数据，需根据相应的数据预处理方法来整理数据，为本题的解决与后续的建模奠定基础。

问题 3：建立二次预报数学模型

A、B、C 三个监测点两两之间的直线距离大于 100km，在忽略它们之间相互影响的基础上，使用附件 1 中监测点 A 和附件 2 中监测点 B、C 的空气质量一次预报基础数据，建立一个二次预报数学模型，同时适用于上述三个监测点。该二次预报模型可以预测未来三天 6 种常规污染物的单日浓度值，其预测结果中 AQI 预报值的最大相对误差要尽可能小，并且首要污染物的预测准确度要尽可能高。

使用该二次预报模型预测 A、B、C 三个监测点在 2021 年 7 月 13 日到 7 月 15 日这段时间内 6 种常规污染物的单日浓度值，并计算相应的 AQI 和首要污染物，最后按照附录中“污染物浓度及 AQI 预测结果表”的格式，将结果放在正文中。

问题 4：建立区域协同预报模型

实际情况中，相邻区域内的污染物浓度在很大程度上具有一定的相关性，在问题 3 的基础上使用区域协同预报模型，可能会在一定程度上提升空气质量预报的准确度。以题中所给的监测点 A 及其临近区域内的监测点 A1、A2、A3 为例，在二次预报模型的基础上建立包含 A、A1、A2、A3 四个监测点在内的协同预报模型。该协同预报模型预测结果中 AQI 预报值的最大相对误差要尽可能小，并且首要污染物的预测准确度要尽可能高。

使用该协同预报模型预测 A、A1、A2、A3 四个监测点在 2021 年 7 月 13 日到 7 月 15 日这段时间内 6 种常规污染物的单日浓度值，并计算相应的 AQI 和首要污染物，最后按照附录中“污染物浓度及 AQI 预测结果表”的格式，将结果放在正文中。

将问题 4 建立的区域协同预报模型与问题 3 建立的二次预报模型进行对比，观察针对监测点 A 的污染物浓度预报准确度，区域协同预报模型的结果是否优于二次预报模型结果，并思考相应的原因。

2. 模型假设及关键性符号说明

2.1 模型假设

(1) 假设污染物浓度与气象一次预报数据中天气条件对污染物浓度及AQI的影响程度与实测数据相似；
(2) 假设题目中A、B、C三个监测点之间的相互影响可忽略不计；
(3) 假设一次预报对邻近日期的准确度较高，且二次预报对邻近日期的准确度也较高；
(4) 假设题目中监测点A、A1、A2、A3位置关系可以在二维平面上近似表示。

2.2 关键性符号说明

关键性符号说明见表 2-1：

表 2-1 关键性符号说明

符号	含义
AQI	空气质量指数（Air Quality Index）
$IAQI_P$	污染物P的空气质量分指数
T	一个时间周期，三日
d_Ai	该点与 Ai 点的直线绝对距离
w_Ai	该点受 Ai 点的风力影响距离（km）
$stepN$	处理某问题的第 N 个步骤

3. 问题 1 分析与求解

3.1 思路分析

本题数据来源于监测点 A 长期空气质量一次预报基础数据中的**污染物浓度每日实测数据**，数据时间范围为 **2020 年 8 月 25 日到 8 月 28 日**。需要按照附录中的方法，使用这些数据计算时间范围内每天实测的 AQI 和首要污染物。

AQI，即空气质量指数（Air Quality Index），是定量描述空气质量状况的无量纲指数。根据《环境空气质量指数（AQI）技术规定（试行）》（HJ633-2012），AQI 可用于判别空气质量等级。

AQI 计算与评价的过程大致可分为以下三个步骤：

***step1.* 计算空气质量分指数（IAQI）**

对照各项污染物的分级浓度限值，以一氧化碳（CO）、二氧化硫（SO_2）、二氧化氮（NO_2）、臭氧（O_3）、粒径小于等于 10μm颗粒物（PM_{10}）和粒径小于等于 2.5μm颗粒物（$PM_{2.5}$）等各项污染物的实测浓度值（其中，O_3 为最大 8 小时滑动平均浓度，CO、SO_2、NO_2、$PM_{2.5}$ 和 PM_{10} 为 24 小时平均浓度）分别计算得出空气质量分指数（Individual Air Quality Index，简称 IAQI）。

各项污染物项目浓度限值及对应的空气质量分指数级别见表 3-1。

表 3-1 空气质量分指数（IAQI）及对应的污染物项目浓度限值

序号	指数或污染物项目	空气质量分指数及对应污染物浓度限值								单位
0	**空气质量分指数（IAQI）**	0	50	100	150	200	300	400	500	-
1	**一氧化碳（CO）24 小时平均**	0	2	4	14	24	36	48	60	mg / m³
2	**二氧化硫（SO_2）24 小时平均**	0	50	150	475	800	1600	2100	2620	μg / m³
3	**二氧化氮（NO_2）24 小时平均**	0	40	80	180	280	565	750	940	
4	**臭氧（O_3）最大 8 小时滑动平均**	0	100	160	215	265	800	-	-	
5	**粒径小于等于 10μm颗粒物（PM_{10}）24 小时平均**	0	50	150	250	350	420	500	600	
6	**粒径小于等于 2.5μm颗粒物（$PM_{2.5}$）24 小时平均**	0	35	75	115	150	250	350	500	

空气质量分指数（IAQI）的计算公式如下：

$$IAQI_P = \frac{IAQI_{Hi} - IAQI_{Lo}}{BP_{Hi} - BP_{Lo}} \cdot (C_P - BP_{Lo}) + IAQI_{Lo} \tag{3-1}$$

上述式子中各符号含义如下：

$IAQI_P$　　污染物P的空气质量分指数，**结果进位取整数**；

C_P　　污染物P的质量浓度值；

BP_{Hi}，BP_{Lo}　　与C_P相近的污染物浓度限值的高位值与低位值；

$IAQI_{Hi}, IAQI_{Lo}$　　与BP_{Hi},BP_{Lo}对应的空气质量分指数。

臭氧（O3）最大 8 小时滑动平均指一个自然日内 8 时至 24 时的所有 8 小时滑动平均浓度中的最大值，其中 8 小时滑动平均值指连续 8 小时平均浓度的算术平均值。其计算公式如下：

$$C_{O_3} = \max_{t=8,9,\ldots,24} \{\frac{1}{8}\sum_{i=t-7}^{t} c_i\} \tag{3-2}$$

其中c_i为臭氧在某日$i-1$时至i时的平均污染物浓度。

***step2.* 确定 AQI 及首要污染物**

从各项污染物的 IAQI 中选择最大值确定为 AQI，即

$$AQI = max\{IAQI_1, IAQI_2, IAQI_3, \dots, IAQI_n\} \tag{3-3}$$

上述式子中，$\mathrm{IAQI}_1, \mathrm{IAQI}_2, \mathrm{IAQI}_3, \dots, \mathrm{IAQI}_n$为各污染物项目的分指数。在本题中，对于 AQI 的计算仅涉及提供的六种污染物，因此计算公式如下：

$$AQI = max\{IAQI_{SO_2}, IAQI_{NO_2}, IAQI_{PM_{10}}, IAQI_{PM_{2.5}}, IAQI_{O_3}, IAQI_{CO}\} \tag{3-4}$$

当 AQI 小于或等于 50 时，认为当天无首要污染物；

当 AQI 大于 50 时，IAQI 最大的污染物为首要污染物。若 IAQI 最大的污染物为两项或两项以上时，并列为首要污染物。

IAQI 大于 100 的污染物为超标污染物。

***step3.* 确定空气质量等级**

对照 AQI 分级标准，确定空气质量等级，从而提出一些建设性的措施。空气质量等级范围根据 AQI 数值划分，等级对应的 AQI 范围见表 3-2。

表 3-2 空气质量等级及对应空气质量指数（AQI）范围

空气质量等级	优	良	轻度污染	中度污染	重度污染	严重污染
空气质量指数（AQI）范围	[0,50]	[51,100]	[101,150]	[151,200]	[201,300]	[301,+∞)

在实际计算中，当臭氧（O_3）最大 8 小时滑动平均浓度值高于 800 $\mu g / m^3$或其余污染物浓度高于 IAQI=500 对应限值时，无需再进行其空气质量分指数计算。对于$\mathrm{BP_{Hi}}$、$\mathrm{BP_{Lo}}$、$\mathrm{IAQI_{Hi}}$、$\mathrm{IAQI_{Lo}}$这四个参数我们直接使用附录中给出的数据，并参考Hi和Lo这两个元素的参数值计算即可。而$\mathrm{C_P}$是本题的变量，指的是污染物 P 的质量浓度值，在计算过程中需要注意单位。AQI 的最终值取所有污染物 IAQI 的最大值。需要注意的是，首要污染物是否被认定存在，要视 AQI 的取值来决定。若当日空气质量为“优”（AQI 小于或等于 50）时，则忽略首要污染物。并且，首要污染物可能不止一项。

综上所述，本题按照给定的方法进行计算即可。

3.2 算法及处理过程

本题求解的算法流程如图 3-1 所示：

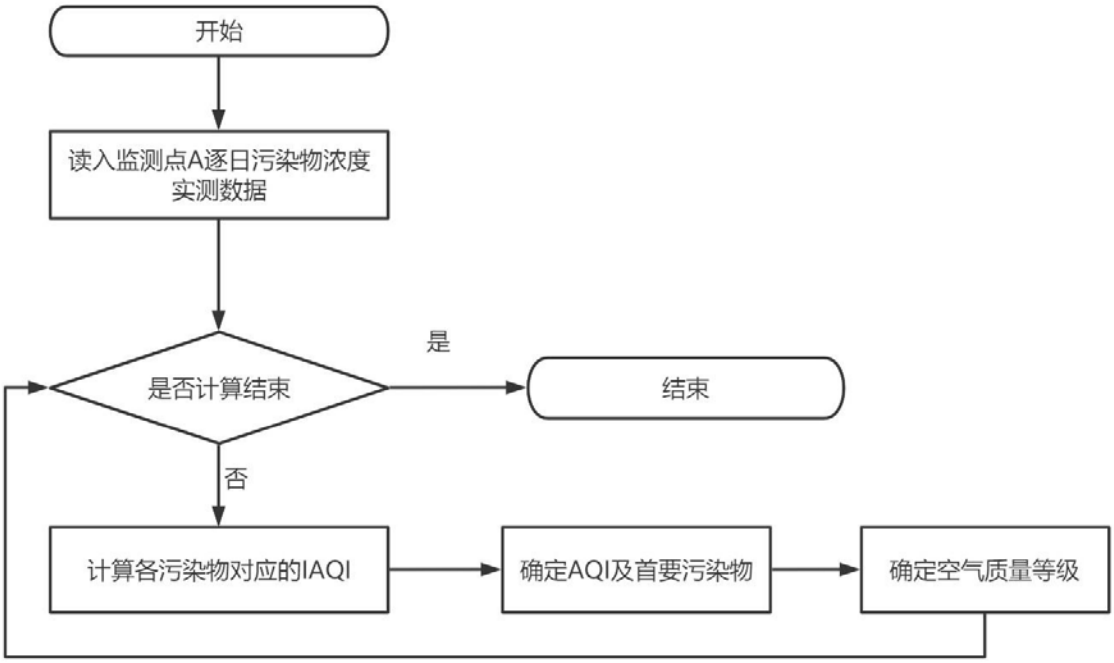


图 3-1 问题 1 算法流程图

设置判断条件的目的是逐一计算每日数据，以得到本题期望的结果。

3.3 计算结果

计算结果如表 3-3 所示：

表 3-3 AQI 计算结果表

监测日期	地点	AQI 计算	
		AQI	首要污染物
2020/8/25	**监测点 A**	60	O_3
2020/8/26	**监测点 A**	46	无首要污染物
2020/8/27	**监测点 A**	109	O_3
2020/8/28	**监测点 A**	138	O_3

根据上述 AQI 计算结果及表 3-2 中对应的等级标准，可以得到 2020/8/25~2020/8/28 这四天的空气质量等级分别为：良、优、轻度污染、轻度污染。

4. 问题 2 分析与求解

4.1 思路分析

在本题中，要求使用附件 1 中的数据，根据对污染物浓度的影响程度，对气象条件进行合理分类，并阐述各类气象条件的特征。

首先观察附件 1 中的数据，包括监测点 A 逐小时污染物浓度与气象一次预报数据、监测点 A 逐小时污染物浓度与气象实测数据以及监测点 A 逐日污染物浓度实测数据。结合附录-数据异常情形所述，主要需要对监测点 A 逐小时污染物浓度实测数据做数据预处理，对其中的缺失值、异常值进行合理处理。监测点 A 逐小时污染物浓度实测数据采集时间从 2019/4/16 0:00 至 2021/7/13 7:00，采集特征包括六种污染物在该小时的平均监测浓度，以及五种天气条件（温度、湿度、气压、风速、风向）。针对原始数据可能存在的异常情况作如下：

(1) 变量值缺失

由于监测站点设备调试、维护等原因，各个监测点的污染物浓度与气象实测数据在连续时间内存在部分或全部缺失的情况。若直接删除这些缺失数据行将影响数据集的完整性或连续性。因此，对于连续时间内部分缺失的数据，可以使用一定的方法进行插值。在比较过常用的插值方法后，我们选取拉格朗日插值法来对表中缺失数据进行插值。而对于连续时间内全部缺失的数据，其无法用插值法补充，只能剔除。

(2) 变量值超出合理范围

在实际统计过程中，可能由于设备或操作人员的原因导致某特征值超出该特征的合理范围，对这种情况则考虑将该特征值对应的时间节点数据信息记录剔除出去。如六种污染物浓度在实际意义的情况下应该始终保持大于等于零，因此需要将小于零的浓度项对应的时间节点数据信息记录剔除。

(3) 变量值超出箱线图区间范围

因受监测站点及其附近某些偶然因素的影响，实测数据存在于某个小时（某天）的数值偏离数据正常分布的可能。由于这部分偏离数据项有可能对模型产生波动影响，因此需要将这部分数据剔除。依据箱线图的原理，做出六种污染物浓度以及五种天气条件数据分布的箱线图，计算 11 个特征箱线图的合理区间，将超出范围区间的值对应的时间节点数据信息项剔除。

解读问题 2 题干可知，要求获取根据对污染物浓度影响程度的气象条件分类，因此应重点关注因气象条件变化而导致的污染物浓度变化。为了将气象条件变化、污染物浓度变化使用准确的数据表示出来，本文对上述预处理后得到的数据集进行了重新构建。以监测点 A 逐日污染物浓度实测数据为基础，每三天为一个时间周期，观察三天时间气象条件的变化情况以及污染物浓度的变化情况。具体操作为，从附件一监测点 A 逐小时污染物浓度实测数据中第一条数据开始，与其三天后对应的时间节点的各天气情况、污染物浓度情况建立联系，计算三天后各天气条件、污染物浓度变化量（若对应的时间节点没有数据信息，则向上查找与对应时间节点最邻近的时间节点项）。因此最终得到的新数据集特征如下，每条记录命名为一个为期三天的时间周期，如 2020/7/13 0:00~2020/7/16 0:00，每条记录包括 22 个特征（起始时间的六种污染物浓度监测值、起始时间的五种天气条件、三天后六种污染物浓度各自的变化量、三天后五种天气条件各自的变化量），共计 18732 条数据记录。该数据集描述了每个时间周期下，污染物浓度、天气条件的初始值以及三天后的变化量。

构建新数据集后，为了根据对污染物浓度的影响程度对气象条件进行分类，实为使用非监督学习算法对其进行聚类。本文采用基于 EM 算法的高斯混合模型（GMM）聚类方法[5] 对 18724 个时间周期信息进行聚类，并使用 t-SNE 降维可视化[6] 方法观察聚类的效果，最终得到 6 个簇，即将气象条件分为 6 类。

同时，为了保证实验的严谨性，本文对附件 1 中监测点 A 逐小时污染物浓度与气象一次预报数据也做了相应的处理。主要操作是将一次预报数据中 15 个天气条件、构建好的新数据集中 10 个天气特征变量分别与 6 种污染物浓度做相关性分析，使用斯皮尔曼（Spearman）相关系数计算出变量相关系数矩阵，生成可视化图像，并分析每种天气条件对每种污染物浓度的相关程度，阐述每种天气条件的特征。最后结合实测数据与参考文献，总结 6 类气象条件对污染物浓度的影响程度，并阐述每类气象条件的特征。解决问题 2 的技术路线如下图 4-1 所示：

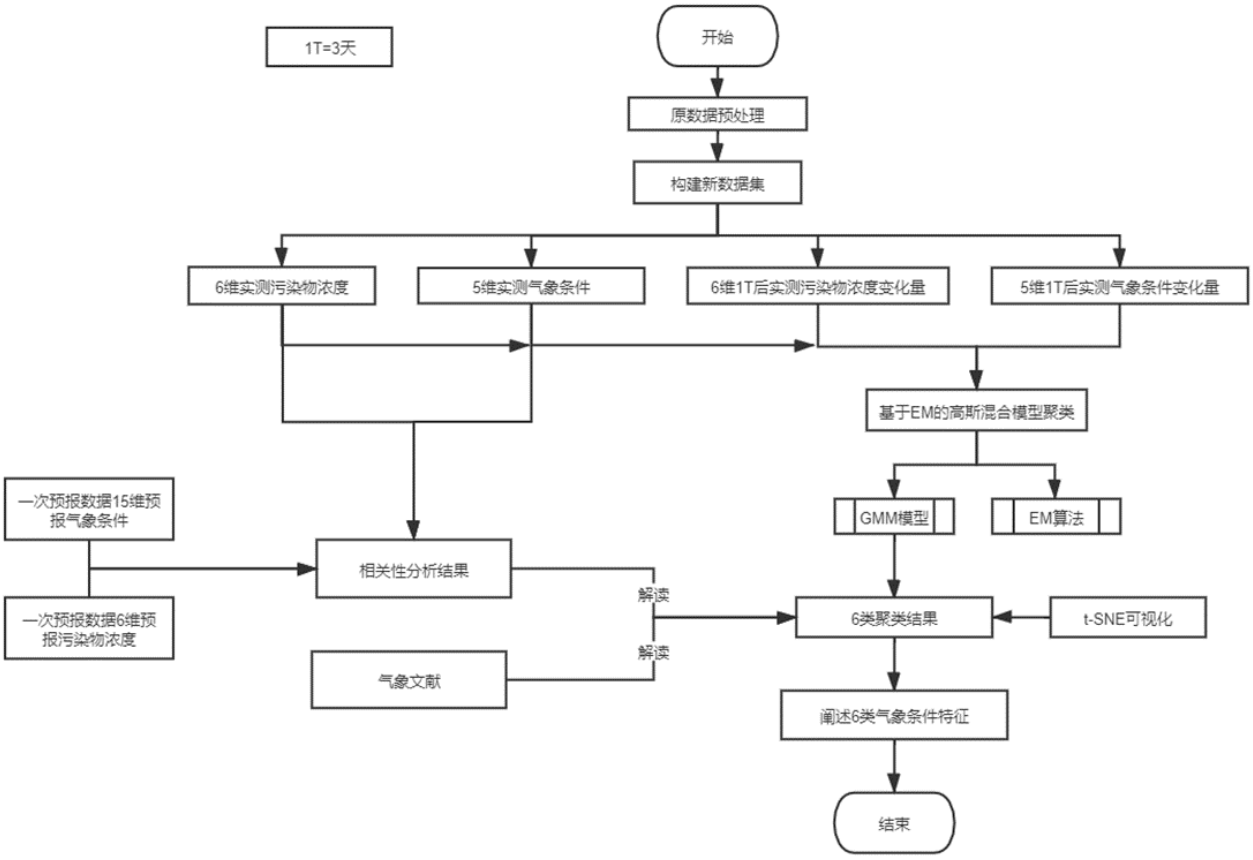


图 4-1 问题 2 技术路线图

4.2 数据预处理

4.2.1 缺失值、异常值处理

(1) 缺失值处理

拉格朗日插值法主要思想是将某列数据的分布看成二维平面上的若干点，并利用数据的数值连续性，拟合出一条尽量能穿过缺失数据处周围点的曲线，缺失数据在曲线上的对应位置即为其所预测的缺失值的大小。

而求得这条过 n 个点的平面曲线即转化为求一个 n-1 次多项式的表达式：

$$y = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1} \tag{4-1}$$

将已知的 n 个点$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$带入多项式函数后可得如下等式：

$$y_1 = a_0 + a_1 x_1 + a_2 {x_1}^2 + a_n {x_1}^{n-1} \tag{4-2}$$

$$y_2 = a_0 + a_1 x_2 + a_2 {x_2}^2 + a_n {x_2}^{n-1} \tag{4-3}$$

……

$$y_n = a_0 + a_1 x_n + a_2 {x_n}^2 + a_n {x_n}^{n-1} \tag{4-4}$$

解出拉格朗日插值多项式为：

$$L(x) = \sum_{i=0}^{n} y_i \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} \tag{4-5}$$

将缺失值对应的点 x 代入已求得的多项式中，得到的函数值即为缺失值的近似值。

由此可看出，由于本题数据具有时间序列连续性的特点，拉格朗日插值法比传统的均值/中位数/众数插值法更适用于本题，同时也比线性插值能够适应更多类型的数据变化趋势，插值效果更精准。

本题中我们选取 n＝4，即以缺失值位置的前 2 个数据点以及后 2 个数据点为已知点，并求解相应的 3 次多项式，从而预测缺失值的大小。

另外，表中一些连续时间内完全缺失的数据无法通过插值方法处理，所以需要剔除这些数据行。

(2) 超出合理范围的异常值处理

该步骤主要目标为剔除一些数值所在范围不符合常理的异常值，根据附录中表 1 给出的空气质量分指数（IAQI）及对应的污染物项目浓度限值，剔除原数据集中各污染物浓度小于零的数据项。

(3) 超出箱线图区间范围的异常值处理

箱线图是一种用作显示一种数据分散情况的统计图，主要用于反映原始数据分布的特征，还可以进行多组数据分布特征的比较。箱线图主要包含六个数据节点，将一组数据从大到小排列，分别计算出上边缘，上四分位数Q_3、中位数、下四分位数Q_1、下边缘，并显示异常值。如图 4-2 箱线图所示，标示了每条线的表示含义。数据x的合理范围为：

$$Q_3 - 1.5(Q_3 - Q_1) \leq x \leq Q_3 + 1.5(Q_3 - Q_1) \tag{4-6}$$

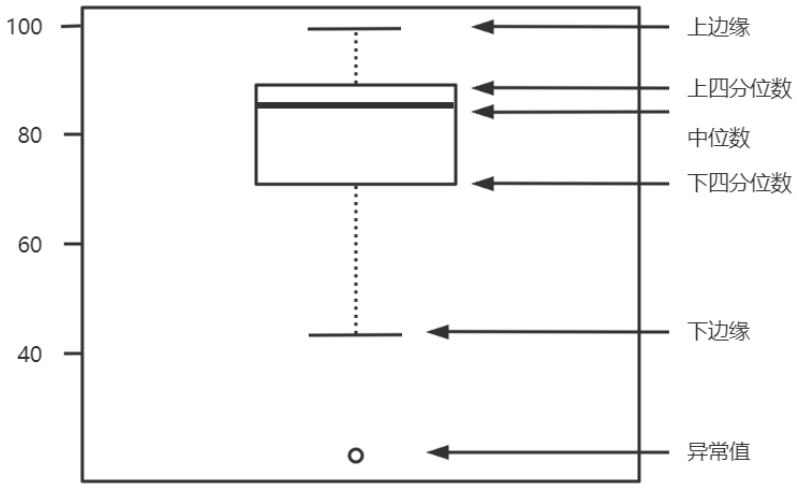


图 4-2 箱线图

在常见的数据预处理方法中，一般使用基于正态分布的3σ准则，该准则默认数据服从正态分布，并且以数据集的均值与标准差为基础标准判断一个数据是否为异常值。但实际数据很少严格服从正态分布，同时均值与标准差的耐抗性较小，异常值往往会对其产生较大的波动影响，

这样计算得到的异常值个数不会多于总数的 0.7%。观察监测点 A 逐小时污染物浓度实测数据的部分特征数据分布情况如下图。

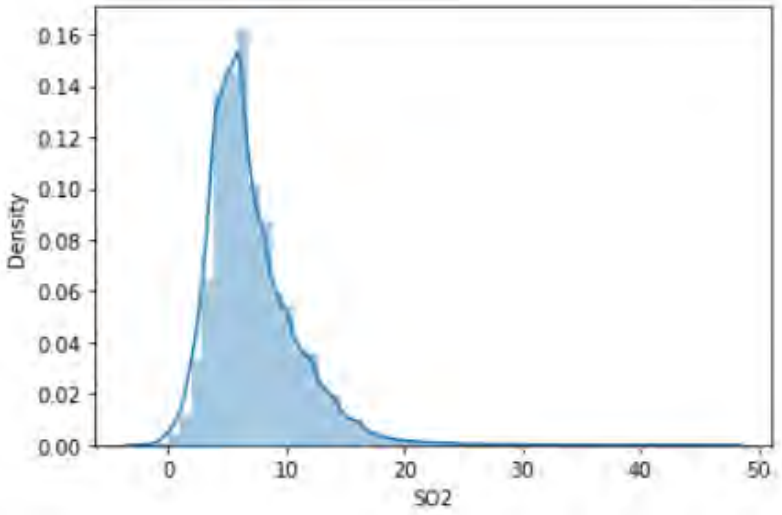


图 4-3 SO_2 实测数据分布图

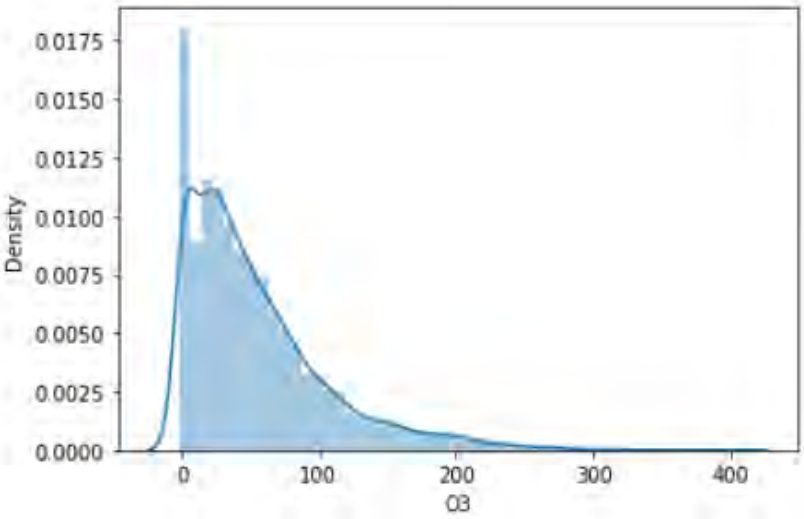


图 4-4 O_3 实测数据分布图

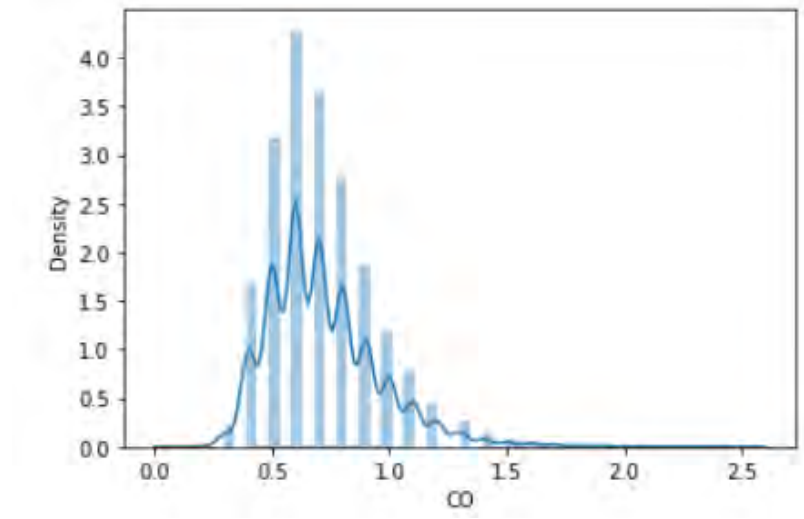


图 4-5 CO 实测数据分布图

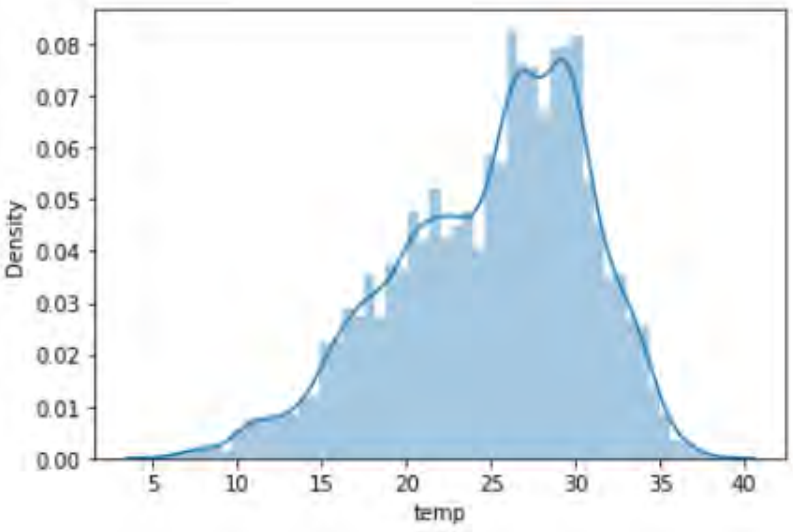


图 4-6 temp 实测数据分布图

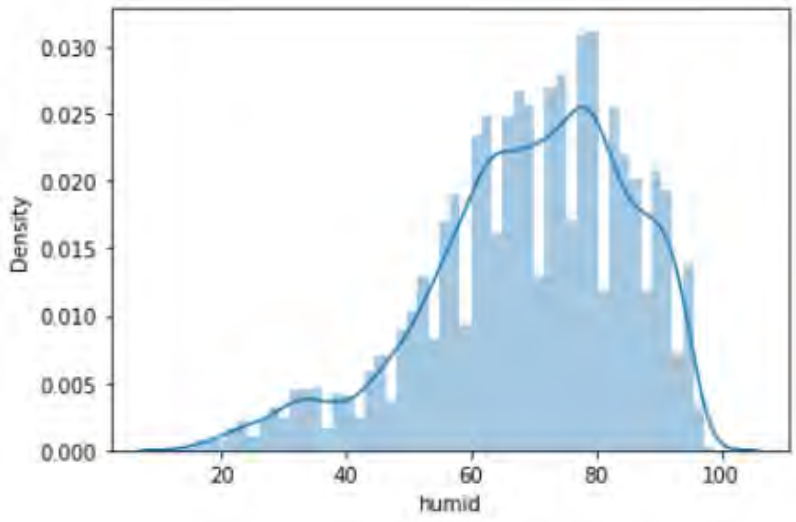


图 4-7 humid 实测数据分布图

可以看到，数据分布是非严格正态分布的。箱线图依靠实际数据，不需要以数据服从正态分布为前提，没有任何限制性要求，真实直观的表现数据形状；此外，箱线图判断异常值的标准以四分位数和四分位距为基础，四分位数具有一定的耐抗性，多达 25%的数据可以变得任意远而不会很大地扰动四分位数，所以异常值不能对这个标准施加影响，箱形图识别异常值的结果比较客观。

4.2.2 算法及处理过程

数据预处理流程如图 4-8 所示：

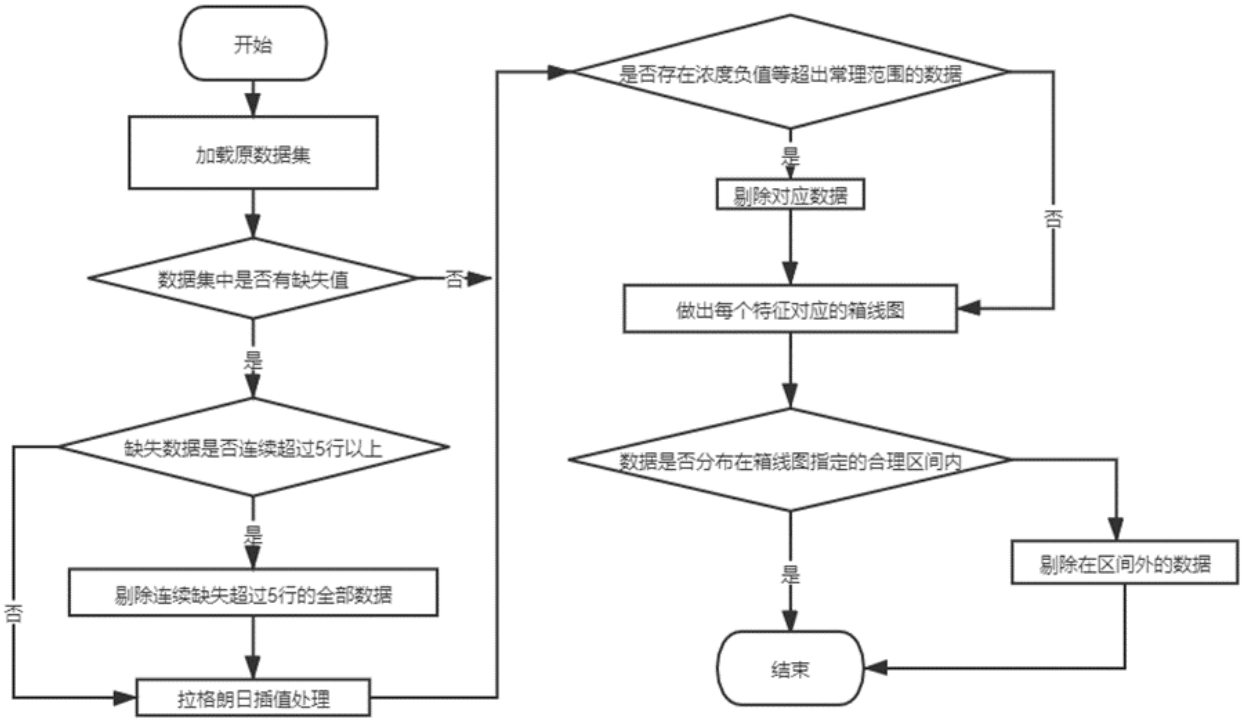


图 4-8 数据预处理流程图

4.2.3 数据预处理结果

对附件一监测点 A 逐小时污染物浓度实测数据中六种污染物浓度监测值与五种天气条件的原始数据信息进行数据预处理。本文采用 python 语言编程实现对原始数据信息的预处理，具体的操作步骤与结果展示如下：

(1) 原始数据中所有存在的缺失值进行检查，发现存在部分值缺失，使用拉格朗日插值法进行填补。

183	2019-04-23 13:00:00	监测点A	5.666667	14	28	10
184	2019-04-23 14:00:00	监测点A	5	14	26	13
185	2019-04-23 15:00:00	监测点A	5	14	24	12
186	2019-04-23 16:00:00	监测点A	4	16.33333	26.33333	10
187	2019-04-23 17:00:00	监测点A	4	20	36.88889	16
188	2019-04-23 18:00:00	监测点A	5	24	44	14
189	2019-04-23 19:00:00	监测点A	4	26	36	16

图 4-9 原始数据

183	2019/4/23 13:00	监测点A	nan	14	28	10
184	2019/4/23 14:00	监测点A	5	14	26	13
185	2019/4/23 15:00	监测点A	5	14	nan	12
186	2019/4/23 16:00	监测点A	4	nan	nan	10
187	2019/4/23 17:00	监测点A	4	20	nan	16
188	2019/4/23 18:00	监测点A	5	24	44	14
189	2019/4/23 19:00	监测点A	4	26	36	16

图 4-10 拉格朗日插值法处理后的数据

如图 4-9 和图 4-10 所示，部分缺失的数据已根据拉格朗日插值法完成了补充。连续三次缺失的数据也可以做到较为合理的插值。另外，由于所插的值已经是预测值，故保留小数，不进行取整，否则将会影响插值的准确度。

502	2019/5/6 21:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
503	2019/5/6 22:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
504	2019/5/6 23:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
505	2019/5/7 0:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
506	2019/5/7 1:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
507	2019/5/7 2:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
508	2019/5/7 3:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
509	2019/5/7 4:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
510	2019/5/7 5:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
511	2019/5/7 6:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
512	2019/5/7 7:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
513	2019/5/7 8:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan
514	2019/5/7 9:00	监测点A	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan

图 4-11 剔除处理后的数据

如图 4-11 所示，连续时间内完全缺失的数据已无法使用拉格朗日插值法进行填补，将被剔除。经过剔除与插值处理后的“监测点 A 逐小时污染物浓度与气象实测数据”表共计 19238 行（不计表头）。

(2) 检查原始数据中监测点 A 所有时间节点采集到的各污染物浓度监测值、各天气条件值是否符合附录中规定的范围要求。经检查发现，污染物浓度监测值存在部分负值的情况，如下二表所示部分数据，将这些负值对应的时间节点数据项进行剔除。

表 4-1 SO_2 监测浓度负值

监测时间	SO_2 监测浓度(μg/m³)
2019/4/25 13:00	-1
2019/5/17 11:00	-2
2020/4/23 4:00	-1
2020/5/10 15:00	-1
2020/5/11 2:00	-1
2020/5/11 4:00	-1

表 4-2 PM_{10} 监测浓度负值

监测时间	PM_{10} 监测浓度(μg/m³)
2019/6/11 17:00	-1
2019/6/11 22:00	-1
2020/3/30 4:00	-2
2020/8/3 5:00	-3
2021/2/10 12:00	-2
2021/6/2 20:00	-2

(3) 在(2)的基础上，继续对数据进行箱线图法检查筛选，将其中可能偏离正常范围的值筛选出去。这里给出六种污染物浓度监测值与五种天气条件值对应的箱线图，以便直观地检查异常值的分布。得到各特征下对应的箱线图区间后，剔除部分在箱线图区间外的数据项。6 种污染物实测浓度箱线图如图 4-12 所示，5 个气象条件实测数据箱线图如图 4-13 所示。

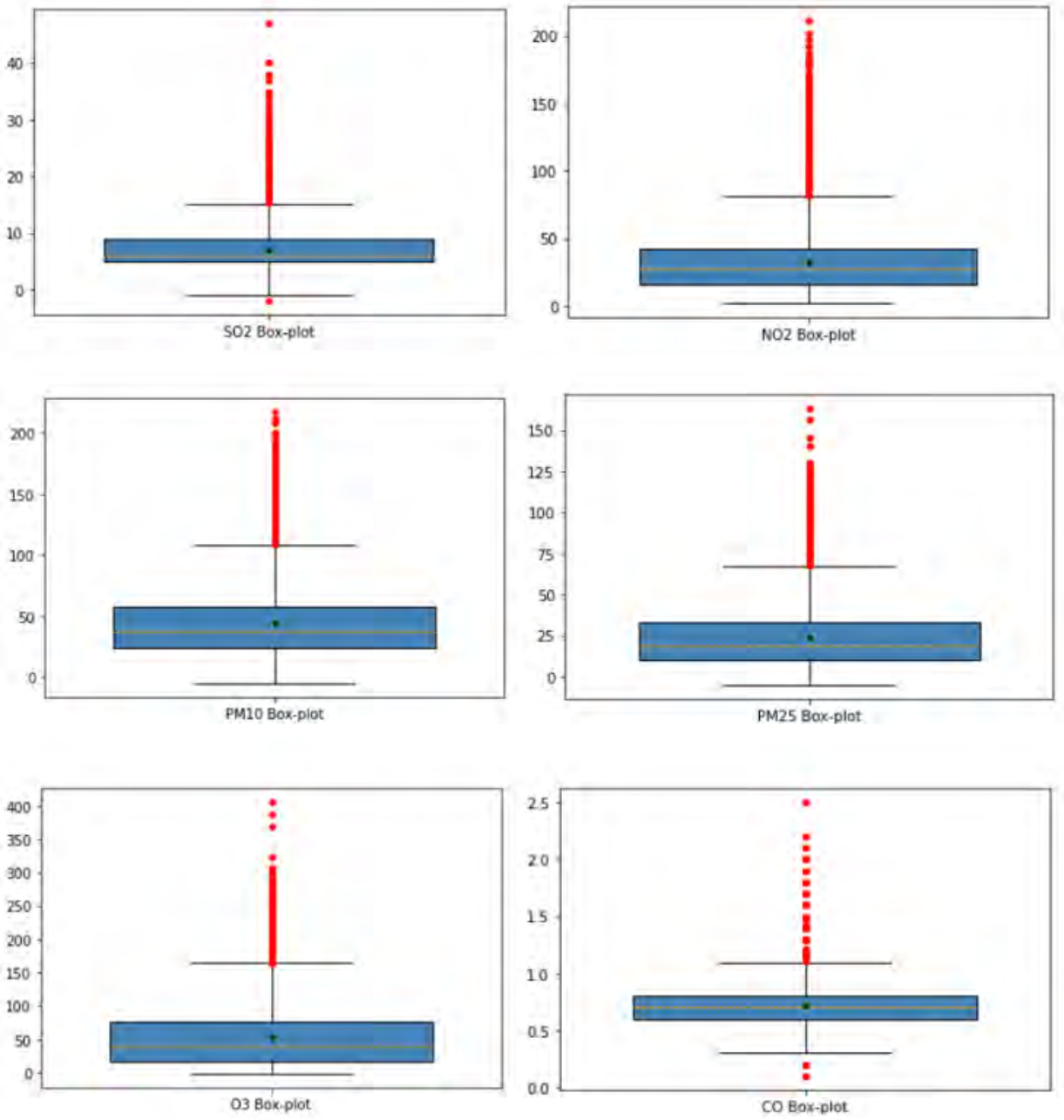


图 4-12 6 种污染物实测浓度箱线图

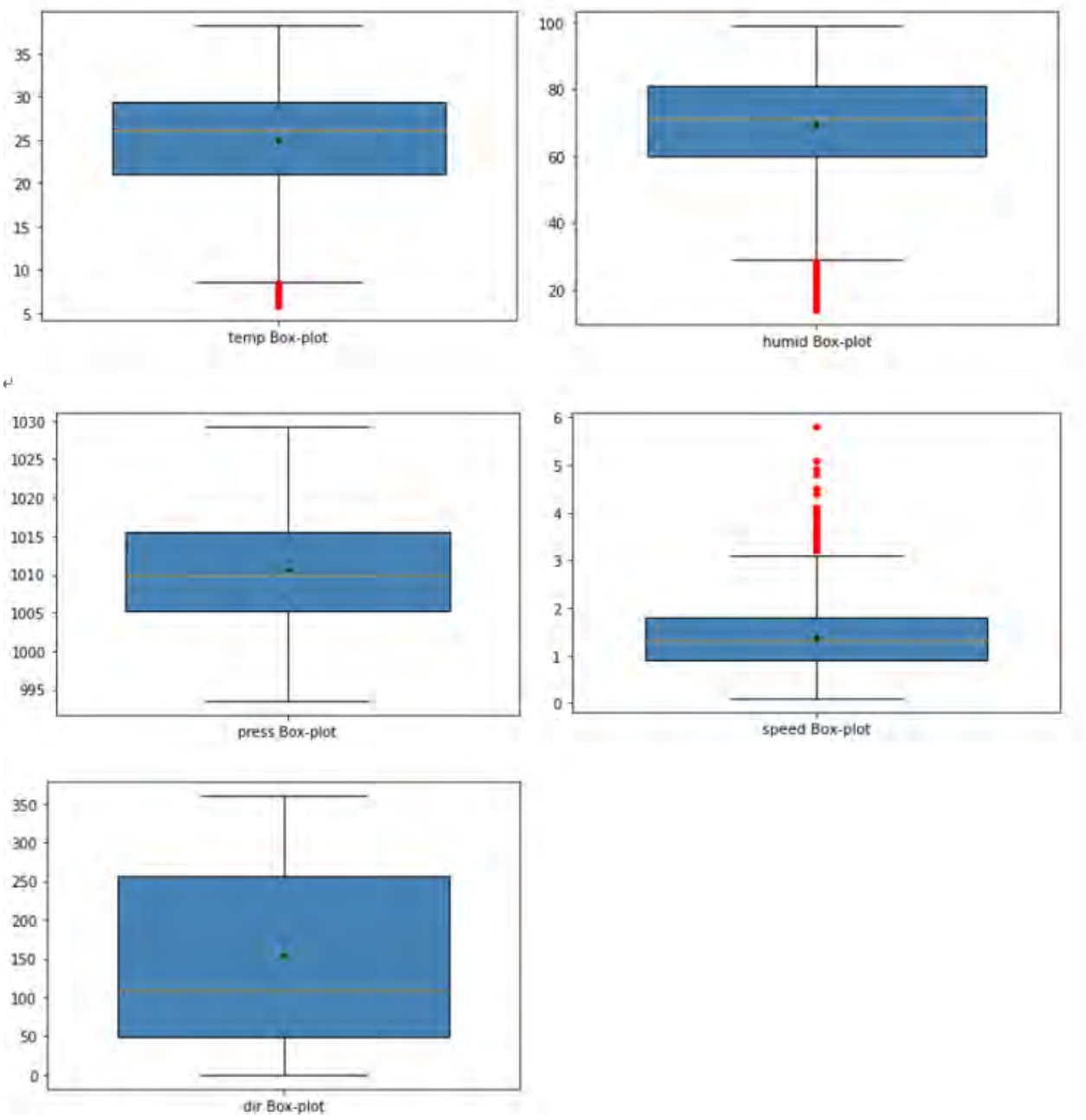


图 4-13 5 个气象条件实测数据箱线图

最后，经过插值处理与异常值处理后的“监测点 A 逐小时污染物浓度与气象实测数据”表共计 18795 行（不计表头）。

4.3 构建新数据集

4.3.1 生成气象条件及各种污染物浓度的周期变化信息

为了探索气象条件的变化对各种污染物浓度变化的影响，需要构建一个新的数据集，从而清晰地表示出每个时间周期内，相应的气象条件变化对应的各种污染物浓度变化。本文规定以三天为一个时间周期，构建记录 2019/4/16 0:00 至 2021/7/10 7:00 时间内相应的气象条件变化

对应的各种污染物浓度变化的新数据集。每条数据记录一个时间周期内的初始各污染物浓度、初始各天气条件、各污染物浓度的变化量、各天气条件的变化量，共 22 个特征，如表 4-3 所示(该表省略了监测时间与地点变量)：

表 4-3 新数据集特征表

属性名称	变量名	属性名称	变量名
SO_2 监测浓度(μ g/m³)	SO2	SO_2 三日监测浓度变化	delta_SO2
NO_2 监测浓度(μ g/m³)	NO2	NO_2 三日监测浓度变化	delta_NO2
PM_{10} 监测浓度(μ g/m³)	PM10	PM_{10} 三日监测浓度变化	delta_PM10
$PM_{2.5}$ 监测浓度(μ g/m³)	PM25	$PM_{2.5}$ 三日监测浓度变化	delta_PM25
O_3 监测浓度(μ g/m³)	O3	O_3 三日监测浓度变化	delta_O3
CO 监测浓度(mg/m³)	CO	CO 三日监测浓度变化	delta_CO
温度(℃)	temp	温度三日监测变化	delta_temp
湿度(%)	humid	湿度三日监测变化	delta_humid
气压(MBar)	press	气压三日监测变化	delta_press
近地风速(m/s)	speed	近地风速三日监测变化	delta_speed
风向(°)	dir	风向三日监测变化	delta_dir

4.3.2 缺失值处理

由于在 4.2 数据预处理中剔除了一些具有完全缺失值或异常值的数据行，构建新数据时会存在某个时刻对应第三天（时间属性+72 小时）的数据已被剔除的情况，无法为该时刻补充属性变化信息。针对此类情况，我们选取其对应第三天的时刻前后最近的某时刻记录，以此代替原第三天的时刻记录进行计算。例如：在补充 2019/4/16 0:00 时刻的变化信息时，若发现 2019/4/19 0:00 时刻的数据已被剔除，而存在 2019/4/19 2:00 时刻与 2019/4/18 23:00 时刻的数据。那么根据就近原则，将选取 2019/4/18 23:00 时刻的数据来代替 2019/4/19 0:00 时刻的数据。

经过上述处理所构建的新数据集中，共有 18723 条数据，该数据将最终作为解决问题 2 的输入数据。

4.4 一次预报数据与实测数据的相关性分析

4.4.1 斯皮尔曼相关系数

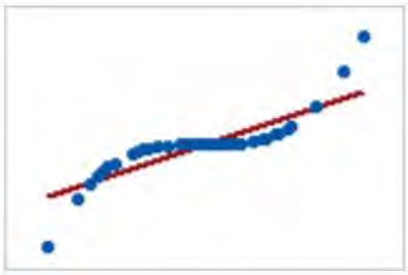
斯皮尔曼是用来衡量两个变量的依赖性的非参数指标，其公式如下：

$$r_8 = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{4-7}$$

由于斯皮尔曼更关注数据集的排序（及单调性）的差异情况，通常也叫斯皮尔曼秩相关系数。

统计学中常用的相关系数还包括皮尔森相关系数和肯德尔相关系数，它们均可反应两个变量之间变化趋势的方向以及程度。但是皮尔森相关系数更要求变量间呈严格的线性关系，而本题数据中的天气变量与污染物浓度并非一定呈线性关系，只需要有单调增减关系即可，因此不适合使用皮尔森相关系数；而肯德尔相关系数常用于分类变量（优、中、差等），本题数据更具有连续性，故也不适合使用肯德尔相关系数。

如下图 4-14 所示，同为递增关系的蓝色变量曲线与红色变量曲线，使用斯皮尔曼相关系数能得出+1 的最高相关系数；而由于线性关系稍弱，使用皮尔森相关系数只能得到+0.851 的相关系数。



Pearson = +0.851, Spearman = +1

图 4-14 剔除处理后的数据

同时，斯皮尔曼相关系数对数据错误和极端值的反应也不敏感。因此，对于本题目的来说，使用肯德尔相关系数更为合适。

4.4.2 处理结果与分析

如图 4-15 所示为一次预报数据的斯皮尔曼相关系数矩阵。根据该矩阵，可以得出以下结论：

(1) 近地 2 米温度、地面温度这两个天气条件对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有较强的负影响，即温度升高时，这些空气污染物可以在较大程度上被扩散或沉降，这使得它们的浓度下降；但上述天气条件并不能降低臭氧（O_3）的浓度甚至还可能会使臭氧的浓度有提升趋势。

(2) 湿度对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、O_3 这五种污染物均具有一定的负影响，即湿度增加时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 CO 几乎没有影响。而比湿对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有较强的负影响，即空气中的水汽质量在混合空气中的质量占比增加时，这些空气污染物可以在较大程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 几乎没有影响。

(3) 近地 10 米风速对 NO_2、PM_{10}、$PM_{2.5}$、CO 这四种污染物均具有一定的负影响，即近地 10 米风速增加时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 SO_2、O_3 这两种污染物几乎没有影响。而近地 10 米风向对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有一定的负影响，即近地 10 米风向增大时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 几乎没有影响。

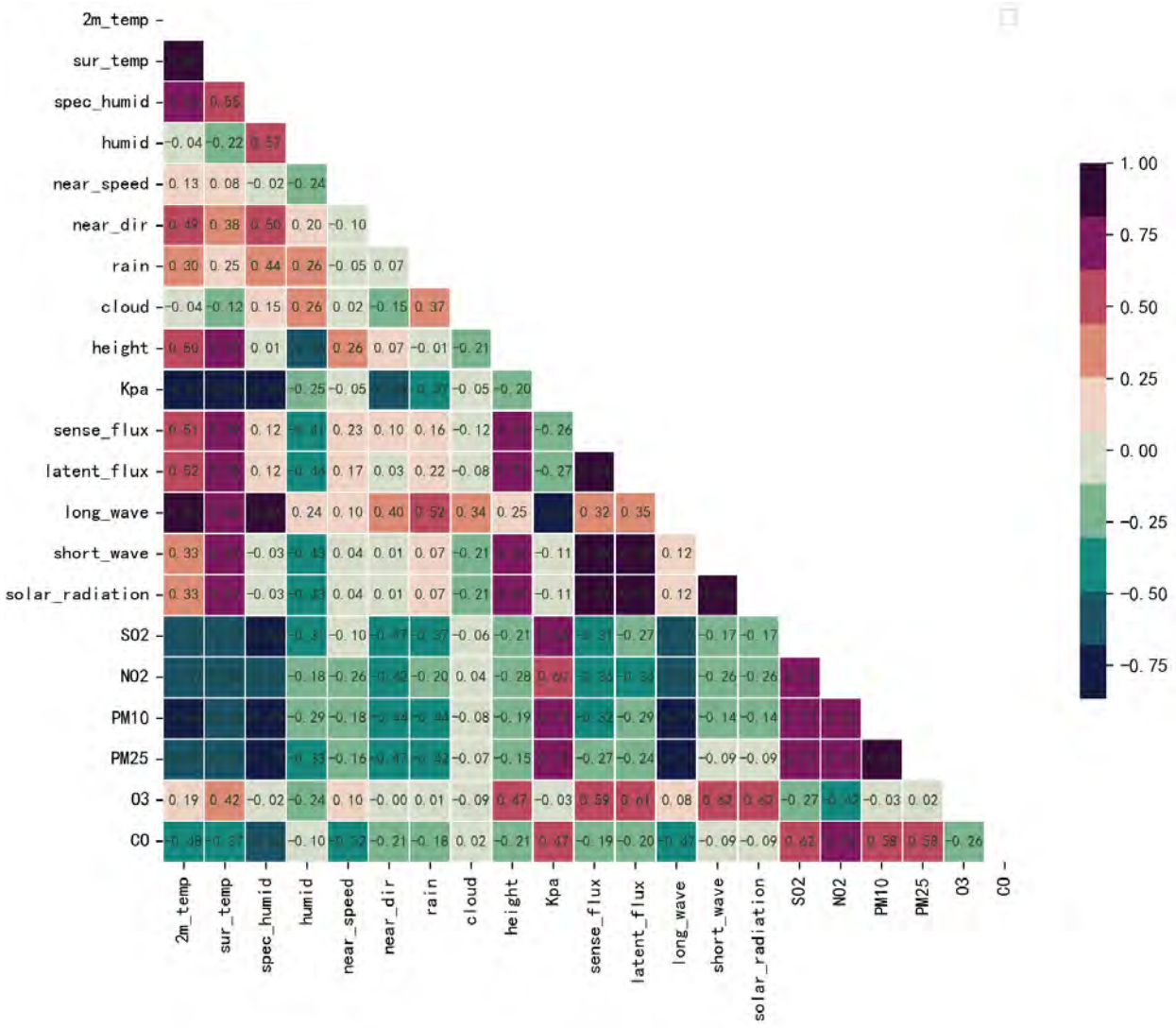


图 4-15 一次预报数据的斯皮尔曼相关系数矩阵

(4) 雨量对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有一定的负影响，即近地 10 米风向增大时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 几乎没有影响。而云量对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、O_3、CO 这六种污染物均无明显影响。

(5) 边界层高度对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有较弱的负影响，即边界层高度增加时，这些空气污染物可以在较小程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 具有较强的正影响，即边界层高度减少时，O_3 可以在较大程度上被扩散或沉降，这使得它的浓度下降。而大气压对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有较强的正影响，即大气压减少时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 几乎没有影响。

(6) 感热通量、潜热通量这两个天气条件对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有一定的负影响，即感热通量或潜热通量增加时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 具有较强的正影响，即感热通量或潜热通量减少时，O_3 可以在较大程度上被扩散或沉降，这使得它的浓度下降。

(7) 长波辐射对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有较强的负影响，即长波辐

射增加时，这些空气污染物可以在较大程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 几乎没有影响。而短波辐射、地面太阳能辐射这两个天气条件对 SO_2、NO_2、PM_{10} 这三种污染物均具有较弱的负影响，即短波辐射或地面太阳能辐射增加时，这些空气污染物可以在较小程度上被扩散或沉降，这使得它们的浓度下降；对 O_3 具有较强的正影响，即短波辐射或地面太阳能辐射减少时，O_3 可以在较大程度上被扩散或沉降，这使得它的浓度下降；但对 $PM_{2.5}$、CO 这两种污染物几乎没有影响。

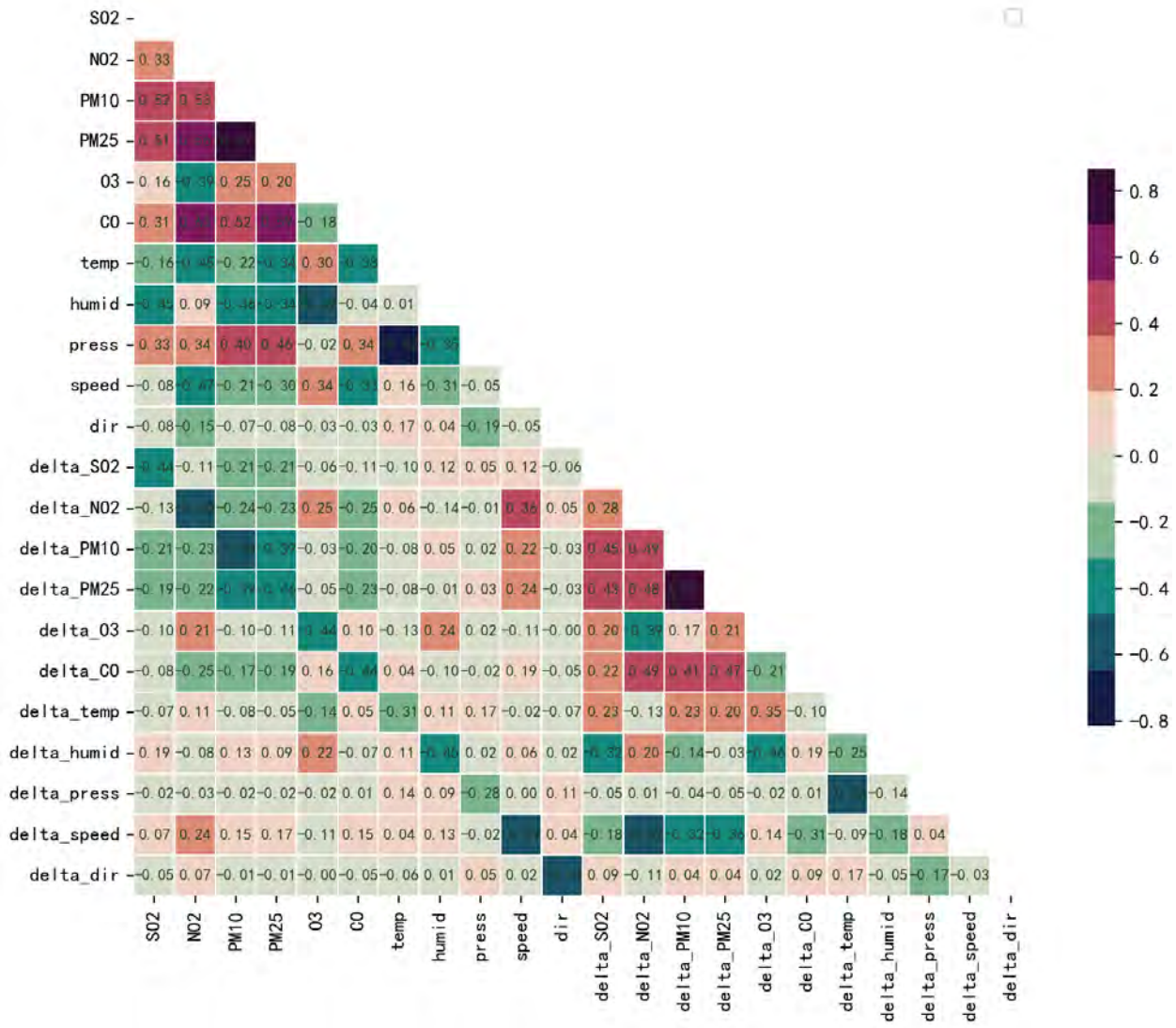


图 4-16 实测数据的斯皮尔曼相关系数矩阵

如图 4-16 所示为实测数据的斯皮尔曼相关系数矩阵。根据该矩阵，可以得出以下结论：

(1) 温度对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有一定的负影响，即温度升高时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 具有一定的正影响，即温度降低时，O_3 可以在一定程度上被扩散或沉降，这使得它的浓度下降。

(2) 湿度对 SO_2、PM_{10}、$PM_{2.5}$、O_3 这四种污染物均具有较强的负影响，即湿度增加时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 NO_2、CO 这两种污染物几乎没有影响。

(3) 气压对 SO_2、NO_2、PM_{10}、$PM_{2.5}$、CO 这五种污染物均具有一定的正影响，即气压降低时，

这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；但对 O_3 几乎没有影响。

(4) 近地风速对 NO_2、PM_{10}、$PM_{2.5}$、CO 这四种污染物均具有一定的负影响，即近地风速增大时，这些空气污染物可以在一定程度上被扩散或沉降，这使得它们的浓度下降；对 O_3 具有一定的正影响，即近地风速减小时，O_3 可以在一定程度上被扩散或沉降，这使得它的浓度下降；但对 SO_2 几乎没有影响。

(5) 风向对 NO_2 具有较弱的负影响，即风向增大时，NO_2 可以在较小程度上被扩散或沉降，这使得 NO_2 的浓度下降；但对 SO_2、PM_{10}、$PM_{2.5}$、O_3、CO 这五种污染物几乎没有影响。

4.5 基于 EM 的 GMM 聚类算法分析

构建新数据集后，新数据集描述了每个时间周期下，污染物浓度、天气条件的初始值以及三天后的变化量，因此对该 22 维数据集进行聚类操作，即可分出不同类别的簇，每个簇具有相似的特征，即可找到气象条件的变化对个污染物浓度变化的影响程度。本文选用基于 EM 的 GMM 聚类算法，使用高斯混合模型拟合新数据集中的数据分布。此外，使用 t-SNE 降维可视化技术展示聚类结果的有效性。

4.5.1 基于 EM 的 GMM 聚类

高斯混合模型（Gaussian mixed model, GMM）[5] 是多个高斯分布函数的线性组合，采用概率模型刻画每个样本的簇类，理论上可以拟合任意形状的数据分布，能够解决同一集合下的数据包含多个不同的分布的情况。对于 4.3 中构建的 22 维数据集，样本数量为d，另随机变量$X=(x^1,x^2,\dots,x^d)$，包含了K个单一高斯模型分量的高斯混合模型可以用下式表示：

$$p(x)=\sum_{k=1}^{K}\pi_k\,\mathcal{N}(x|\mu_k,\boldsymbol{\Sigma}_k) \tag{4-8}$$

其中$\mathcal{N}(x|\mu_k,\boldsymbol{\Sigma}_k)$称为高斯混合模型中的第$k$个单一高斯模型分量，$\mu_k$为第$k$个高斯模型的均值，$\boldsymbol{\Sigma}_k$为第$k$个高斯模型的方差，该高级模型表示为：

$$\mathcal{N}(x|\mu_k,\boldsymbol{\Sigma}_k)=\frac{1}{(2\pi)^{\frac{2}{d}}\boldsymbol{\Sigma}_k^{\frac{1}{2}}}e^{-\frac{1}{2}(x-\mu_k)^T\boldsymbol{\Sigma}_k^{-1}(x-\mu_k)} \tag{4-9}$$

π_k为每个分量$\mathcal{N}(x|\mu_k,\boldsymbol{\Sigma}_k)$所占的权重，且满足：

$$\sum_{k=1}^{K}\pi_k=1\quad(0\le\pi_k\le1) \tag{4-10}$$

构建高斯混合模型重点在于参数的求解，而计算高斯混合模型的参数无法像计算单一高斯模型参数那样使用最大似然法求解。因为对于每个观测样本来说，并不知道其属于哪个分量，需要使用迭代算法求解。

EM（Expectation-Maximum）算法[5] 是一种迭代算法，用于含有隐变量的概率模型参数的最大似然估计。EM 算法经常用于迭代求解高斯混合模型的参数，其每次迭代包括两个步骤，第一步求期望，第二步求极大并作为新一轮迭代的模型参数，最后重复计算第一步与第二步直至收敛。

step1. 根据初始化参数，计算每个数据x^j来自分量模型k的可能性

$$\gamma_{jk}=\frac{\pi_k\mathcal{N}(x^j|\mu_k,\boldsymbol{\Sigma}_k)}{\sum_{k=1}^{K}\pi_k\mathcal{N}(x^j|\mu_k,\boldsymbol{\Sigma}_k)},j=1,2,\dots,d;k=1,2,\dots,K \tag{4-11}$$

step2. 计算新一轮迭代的模型参数

$$\mu_k=\frac{\sum_j^N(\gamma_{jk}x^j)}{\sum_j^N\gamma_{jk}},k=1,2,\dots,K \tag{4-12}$$

$$\Sigma_k = \frac{\sum_j^N \gamma_{jk}(x^j-\mu_k)(x^j-\mu_k)^T}{\sum_j^N \gamma_{jk}}, k = 1,2,\dots,K \tag{4-13}$$

$$\pi_k = \frac{\sum_{j=1}^N \gamma_{jk}}{d}, k = 1,2,\dots,K \tag{4-14}$$

与传统算法 K-means 聚类相比，K-means 的本质在于以每个簇的中心为圆心，簇中其他点到中心的欧式距离最大值为半径做圆，因此最终 K-means 聚类得出的簇形状是一个圆形，并且可能多个簇混在一起，而实际数据的分布几乎很少为圆形，可能为椭圆形或其他形状，使得该聚类算法不够灵活，精度较低，且样本只能存在或不存在与一个簇中。而基于 EM 算法的高斯混合模型，以概率值的方式向每一个簇中分配样本，并且理论上可以拟合任意形状的数据分布，模型鲁棒性更强，聚类结果更加精确。二者聚类效果对比图如下图所示：

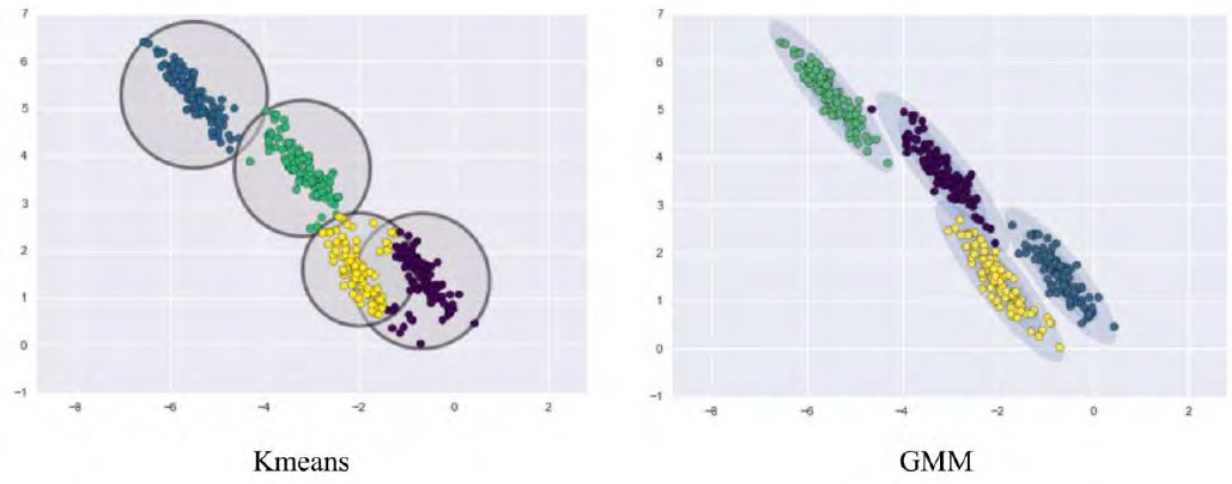


图 4-17 EM 算法与 K-means 算法聚类效果对比图

4.5.2 聚类结果及可视化

对 4.3 中构建的新数据集构建基于 EM 算法的高斯混合模型，将新数据集中的 22 维特征、18723 条数据进行聚类分析，最终得到了 6 个类别。为了检验聚类结果是否合理，本文引入数据降维与可视化技术 t-SNE[6] 。t-SNE 是当下来说效果较优的数据降维与可视化技术，其将数据点之间的相似程度转换为概率计算。当面对一个高维数据时，事先并不了解该高维数据集是否具有较好的可聚性，即同一类别具有较短“距离”，不同类别之间具有较长“距离”，这时可以通过 t-SNE 技术将该高维数据集投影到 2 维或 3 维空间中。如果在 t-SNE 投影的低维空间中具有较明显的可聚性，则该高维数据是可以做聚类分析的；若在低维空间不具备可聚性，则说明该数据集无法做效果较好的聚类分析（也存在高维数据无法投影至低维空间的可能）。对 4.4.1 中得到的聚类结果使用 t-SNE 数据降维与可视化技术，分别投影至 2 维空间与 3 维空间，观察投影后的 2 维数据、3 维数据在空间中的分布情况。数据分布如下图所示：

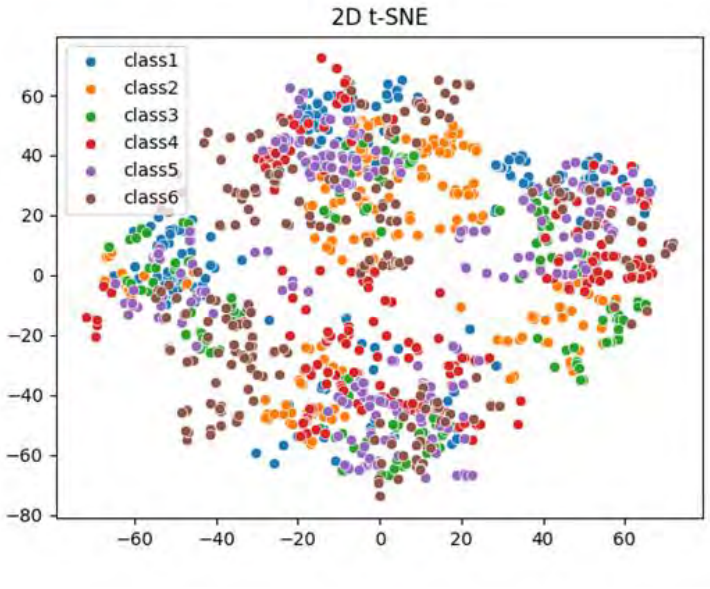


图 4-18 二维数据分布图

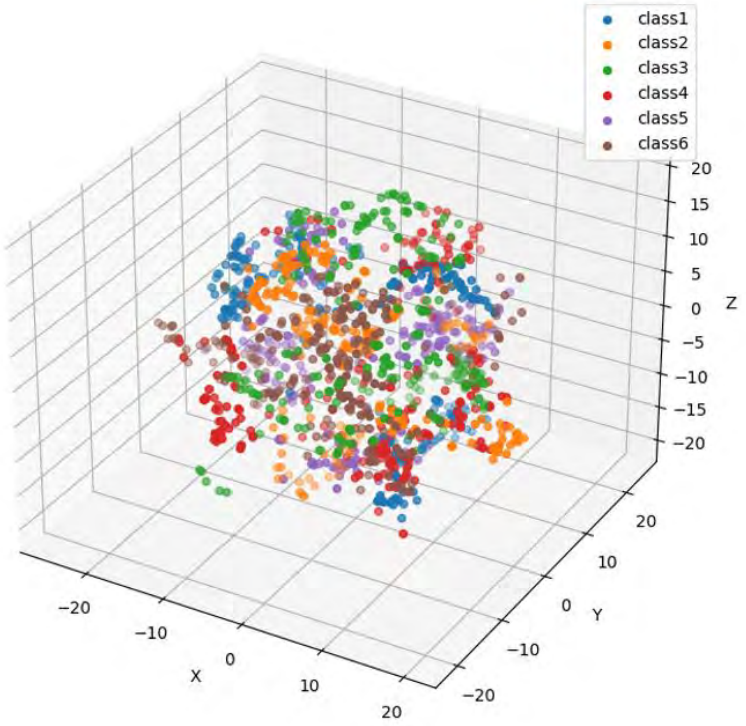


图 4-19 三维数据分布图

4.5.3 聚类结果分析

观察图 4-18 与图 4-19，可以看出 6 类 22 维数据在投影至低维空间后，显示出了较好的可聚性，因此得到的 6 类聚类结果是有效的。基于 EM 的 GMM 模型对新数据集聚类后，得到 6 类聚类结果，每个类别都对应这一个高斯模型，每个类别都有自己独特的数据分布。

本文通过调研我国部分地区的气象状况与空气质量，了解了气象因子对各种污染物的影响

方式与程度[2]，尤其是臭氧对气象因子的敏感度[3]。根据每类模型的特征值以及每类数据的特征，将 6 类聚类结果进行总结。根据是否有利于污染物的扩散或沉降，以及对污染物的扩散或沉降的影响程度排序，对 6 类气象条件特征做如下阐述：

Ⅰ类：当温度约在 18~23 摄氏度范围内，湿度约在 60~65 范围内，气压较高（约在 1017~1020MBar），风速较强，且风向有较大角度改变，此时气象条件为晴天，较为温暖，湿度适宜，有明显风感。**该气象条件对污染物的扩散或沉降具有明显的积极作用**，此时各污染物浓度都有明显程度的下降。

Ⅱ类：当温度约在 20~22 摄氏度范围内，湿度约在 75~80%范围内，气压适中（约在 1005~1015MBar），风速稍强，无明显风向改变，此时气象条件为晴天，较为温暖，空气较为潮湿，有 1 级风。**该气象条件对污染物的扩散或沉降有较为明显的积极作用**，除一氧化碳浓度有轻微升高，其余污染物浓度均有下降趋势，臭氧浓度下降较为明显。

Ⅲ类：当温度范围约在 30~33 摄氏度范围内，湿度约在 65~70%范围内，气压较低（约在 1004~1008MBar），风速稍强，风向角度有较大改变时，此时气象条件为阴天，较为炎热，湿度适宜，有轻微风感。**该气象条件对污染物的扩散或沉降有轻微的积极作用**，除二氧化氮、一氧化碳有轻微升高，其他污染物浓度轻微下降，臭氧浓度下降较其他明显。

Ⅳ类：当温度约在 25~28 度范围内，湿度在 60~63%范围内，气压、风速范围适中时，无明显风向改变时，此时气象条件为晴天，温暖，湿度适宜，伴有微风，气象无明细波动，比较平稳。**该气象条件对污染物的扩散或沉降无明显作用**，因此对除臭氧外的污染物浓度的影响并不显著，但此气象条件有利于部分以臭氧为生成物的化学反应的发生，使得臭氧的浓度有轻微上升。

Ⅴ类：当温度约在 30 摄氏度，湿度约在 75~80%范围内，气压较低（约在 1000~1005MBar），风速较强时，此时气象条件为阴天，可能伴随降水，空气较为潮湿，有 1~2 级大风。**该气象条件对污染物的扩散或沉降具有轻微的负影响**，此时臭氧、二氧化硫浓度有小幅度升高趋势，对其他污染物浓度影响不显著。

Ⅵ类：当温度约在 8~12 摄氏度范围内，湿度约在 50~60%范围内，气压适中（约在 1010~1025MBar），风速适中，风向稍有改变时，此时气象条件为晴天，温度较低，微冷，较为干燥，伴有微风。**该气象条件明显不利于污染物的扩散或沉降**，此时除臭氧外的其他各污染物浓度都有很大程度的升高，但由于气温较低，不利于臭氧的生成，因此臭氧浓度有小幅度下降。

从新数据集中的 6 个类别分别随机挑选 3 条数据，在此省略了每个数据的部分特征，保留了温度、湿度、气压、风速、风向、SO_2 浓度变化量、NO_2 浓度变化量、PM_{10} 浓度变化量、$PM_{2.5}$ 浓度变化量、O_3 浓度变化量、CO 浓度变化量、温度变化量、湿度变化量、气压变化量、风速变化量、风向变化量，共 16 个特征，其中变化量均代表一个时间周期内数值的变化量。从表 4-4、4-5、4-6、4-7、4-8、4-9 对应的 6 个类别共 18 条数据中，可以简单反映 6 个类别所代表的气象条件对各污染物浓度的影响。

Ⅰ类：

表 4-4 Ⅰ类数据特征表

属性名称	数据 1	数据 2	数据 3
温度(℃)	20.9	19.8	23
湿度(%)	60	62	61
气压(MBar)	1019.2	1019	1019.4
风速(m/s)	1.8	2.2	1

风向(°)	9.6	21.5	2
SO_2 浓度变化量(μg/m³)	-1	-1	-9
NO_2 浓度变化量(μg/m³)	-4	-3	-46
PM_{10} 浓度变化量(μg/m³)	-56	-46	-36
$PM_{2.5}$ 浓度变化量(μg/m³)	-39	-35	-21
O_3 浓度变化量(μg/m³)	-13	-11	-32
CO 浓度变化量(mg/m³)	-0.3	-0.3	-0.7
温度变化量(℃)	-5.9	-5.2	-5.2
湿度变化量(%)	-4	-4	15
气压变化量(MBar)	3	3	6.9
风速变化量(m/s)	0.5	0.1	0.8
风向变化量(°)	13.7	-11.7	69.5

Ⅱ类：

表 4-5 Ⅱ类数据特征表

属性名称	数据 1	数据 2	数据 3
温度(℃)	22	21	20.9
湿度(%)	79	80	76
气压(MBar)	1011.7	1014.2	1010.3
风速(m/s)	0.5	1.6	1.4
风向(°)	43.4	57	106.1
SO_2 浓度变化量(μg/m³)	-1	-1	-1
NO_2 浓度变化量(μg/m³)	-3	-7	1
PM_{10} 浓度变化量(μg/m³)	-46	-3	-1
$PM_{2.5}$ 浓度变化量(μg/m³)	-27	3	-4
O_3 浓度变化量(μg/m³)	-10	-24	-31
CO 浓度变化量(mg/m³)	0	0	0.3
温度变化量(℃)	-1.4	2.1	-5.1
湿度变化量(%)	15	9	7
气压变化量(MBar)	-3.9	-2.2	4.5
风速变化量(m/s)	1	-1	-0.7
风向变化量(°)	5.2	4.5	-45.1

Ⅲ类：

表 4-6 Ⅲ类数据特征表

属性名称	数据 1	数据 2	数据 3
温度(℃)	32.4	32.9	31.4
湿度(%)	65	65	69
气压(MBar)	1004	1004.2	1004.1
风速(m/s)	2.2	1.7	1.7
风向(°)	114.3	252.4	237
SO_2 浓度变化量(μg/m³)	-1	0	-6
NO_2 浓度变化量(μg/m³)	8	1	16
PM_{10} 浓度变化量(μg/m³)	-27	-15	-3
$PM_{2.5}$ 浓度变化量(μg/m³)	-19	-8	3
O_3 浓度变化量(μg/m³)	-48	-52	-20
CO 浓度变化量(mg/m³)	0.1	0	0.1
温度变化量(℃)	-5.8	-1.8	-5.1
湿度变化量(%)	23	6	5
气压变化量(MBar)	-2	-3.6	5.7
风速变化量(m/s)	-0.5	0.2	-0.2
风向变化量(°)	-21	45.3	-161.8

Ⅳ类：

表 4-7 Ⅳ类数据特征表

属性名称	数据 1	数据 2	数据 3
温度(℃)	28	27	26.8
湿度(%)	60	62	62
气压(MBar)	1011.5	1019.4	1010.6
风速(m/s)	1.1	1.6	1.4
风向(°)	18.3	49.6	111.2
SO_2 浓度变化量(μg/m³)	4	1	0
NO_2 浓度变化量(μg/m³)	-6	3	2
PM_{10} 浓度变化量(μg/m³)	-9	-1	5
$PM_{2.5}$ 浓度变化量(μg/m³)	-8	-5	15
O_3 浓度变化量(μg/m³)	21	21	43
CO 浓度变化量(mg/m³)	-0.2	-0.4	0
温度变化量(℃)	-1.8	-1.2	-0.5

湿度变化量(%)	-2.2	-1.2	-9
气压变化量(MBar)	0.7	-1.5	3.9
风速变化量(m/s)	0.7	-0.5	-0.6
风向变化量(°)	-4.9	-35.4	-28.4

Ⅴ类：

表 4-8 Ⅴ类数据特征表

属性名称	数据 1	数据 2	数据 3
温度(℃)	30.2	29.7	30.2
湿度(%)	77	78	78
气压(MBar)	1004.5	1004.5	1002.6
风速(m/s)	1.9	2	2.1
风向(°)	251.6	240	254.1
SO_2 浓度变化量(μg/m³)	3	0	2
NO_2 浓度变化量(μg/m³)	5	7	-4
PM_{10} 浓度变化量(μg/m³)	-1	-2	6
$PM_{2.5}$ 浓度变化量(μg/m³)	-1	-2	-2
O_3 浓度变化量(μg/m³)	1	6	2
CO 浓度变化量(mg/m³)	0.1	0	0.1
温度变化量(℃)	0	-0.4	-0.1
湿度变化量(%)	-7	-2	1
气压变化量(MBar)	0.9	0.6	-0.1
风速变化量(m/s)	0.8	-1.1	0.1
风向变化量(°)	-10.8	-117.3	-2.5

Ⅵ类：

表 4-9 Ⅵ类数据特征表

属性名称	数据 1	数据 2	数据 3
温度(℃)	10.8	11.5	9.5
湿度(%)	51	55	60
气压(MBar)	1020	1020.5	1017.6
风速(m/s)	1.2	2.5	2.2
风向(°)	20.3	25.1	3.3
SO_2 浓度变化量(μg/m³)	2	2	3
NO_2 浓度变化量(μg/m³)	17	13	19

PM_{10}浓度变化量(μg/m³)	11	28	28
$PM_{2.5}$浓度变化量(μg/m³)	5	20	24
O_3浓度变化量(μg/m³)	-14	-5	-2
CO 浓度变化量(mg/m³)	-0.1	-0.1	-0.1
温度变化量(℃)	4.2	0.6	2.2
湿度变化量(%)	15	1	-13
气压变化量(MBar)	-0.6	0	2.1
风速变化量(m/s)	-0.5	-2.1	-1.4
风向变化量(°)	25.7	32.9	52

4.6 总结

本节根据对污染物浓度的影响程度，对气象条件进行了分类建模。首先通过拉格朗日插值、箱线图等方法对原数据中的缺失值、异常值进行数据预处理；然后为了体现气象条件的变化对各污染物浓度变化产生的影响，本文构建了一个新的数据集，以三天为一个时间周期，描述这三天内的气象变化情况与各污染物浓度变化情况。

在进行聚类分析之前，分别将一次预报数据中的 15 个气象预报条件与 6 个污染物预报浓度做相关性分析，将实测数据中的 5 个气象实测条件与 6 个污染物实测浓度做相关性分析。简单分析单个气象条件与某个污染物浓度之间的相关性，结合一些气象知识，作为解析聚类结果的一个参考。如通过相关性分析，并查阅相关资料知晓在气压较高时，且风速达到一定水平时，有利于污染物的扩散，使得污染物浓度下降；当湿度较高时，若此时气压较低，则可能会伴有降雨，此时对污染物的沉降具有明显的左右，因此污染物浓度下降；而当湿度较低，环境干燥寒冷时，污染物几乎不会做扩散或沉降运动，所以污染物的浓度会明显升高。将本文得到的相关性分析结果与气象知识作为先验条件，会使得聚类结果的特征更容易被察觉。

使用基于 EM 的 GMM 算法对构建的新数据集做聚类处理，最终根据气象条件对污染物浓度的影响分为 6 类，气象条件分别对应：对污染物的扩散或沉降有明显的积极作用、对污染物的扩散或沉降有较为明显的积极作用、对污染物的扩散或沉降有轻微的积极作用、对污染物的扩散或沉降无明显作用、对污染物的扩散或沉降有轻微的负作用、对污染物的扩散或沉降具有明显的负作用。根据上述已知的先验条件，总结阐述 6 类气象条件的特征。

5. 问题 3 分析与求解

5.1 思路分析

在本题中，要求使用附件 1、2 中的数据，根据 A、B、C 三个监测点的一次预报数据与实测数据，建立能够同时适用于三个监测点的二次预报数学模型，用来预测 6 种常规污染物的单日浓度值，目的是让 AQI 与首要污染物的预报值的预测准确度尽量高。并最终使用该模型预报三个监测点在 2021 年 7 月 13 日至 7 月 15 日 6 种常规污染物的单日浓度值、相应的 AQI 和首要污染物。

观察附件 2 可以看出监测点 B、C 的数据格式与附件 1 中监测点 A 的数据格式相同，均包含逐小时污染物浓度与气象一次预报数据、逐小时污染物浓度与气象实测数据以及逐日污染物浓度实测数据。题目要求预测的 6 种常规污染物的单日浓度值是由逐小时测得的浓度值转换而来的，而其作为模型的输出变量，需要先得到相应的数据用于建模。

已知由于一次预报对邻近日期的准确度较高，二次预报对邻近日期的准确度也较高。并且，通过比较一次预报数据与实测数据可以发现，大部分日期所预测的未来三天的逐小时污染物浓度准确度确实是逐日递减的。由此可以推得：一次预报数据对未来三天每天的污染物浓度的影响程度是不同的。因此本文对预测未来三天的逐小时污染物浓度分别建立一个预测模型。即：该三个预测模型分别用来预测未来第一天、未来第二天和未来第三天的逐小时污染物浓度，以此提升模型的准确度。同时，为了配合三个模型的建立，每个监测点的一次预报数据也应当被拆分为三张数据表——某监测点对未来第一天的一次预报数据、某监测点对未来第二天的一次预报数据、某监测点对未来第三天的一次预报数据。

因此监测点 A、B、C 的污染物浓度与气象一次预报数据与实测数据要通过如下数据预处理步骤来获得能够作为模型的训练数据：

step1. 对逐小时实测数据的缺失值、异常值进行处理；

step2. 通过逐小时污染物浓度实测/一次预报数据计算出逐日实测/一次预报数据的单日浓度值、AQI 值与首要污染物，进而求得 AQI 与首要污染物；

step3. 将逐小时一次预报数据拆分为分别预测未来第一天、第二天和第三天的三张数据表；

step4. 将上一步生成的三张一次预报数据表分别与逐小时实测数据纵向合并，生成三张一次预报数据与实测数据并存表；

step5. 将上一步生成的 A、B、C 三监测点的数据集横向合并，用以最终输入给预测模型。

数据构建好后，本文建立基于 XGBoost 算法建立二次预报回归预测模型。XGBoost（eXtreme Gradient Boosting）[7] 是一个开源的机器学习项目，高效地实现了 GBDT(Gradient Boosting Decision Tree)算法并进行了算法和工程上的许多改进。由于 XGBoost 对量级在十万及以下的数据集进行数据挖掘具有高精度、灵活性强、正则化等优点，本文决定采用 XGBoost 算法建立二次预报回归预测模型，其适用于 A、B、C 三个监测点的二次预报，用来预测未来三天 6 种污染物实测单日浓度值。如附件 1 中所示，每天的一次预报数据包括未来三天的 6 种浓度值情况、气象情况，如一次预报模型于 2020/7/1 对 2020/7/1、2020/7/2、2020/7/3 三天每小时的 6 种污染物浓度值、15 种气象条件情况做预报。本文根据当日预测未来第一天、预测未来第二天、预测未来第三天分类，使用 XGBoost 算法构建了三个对应的二次预报回归预测模型，每个模型种包含 6 个子模型，分别对应对 6 种污染物浓度的预测。模型的分类情况如下图 5-1 所示。

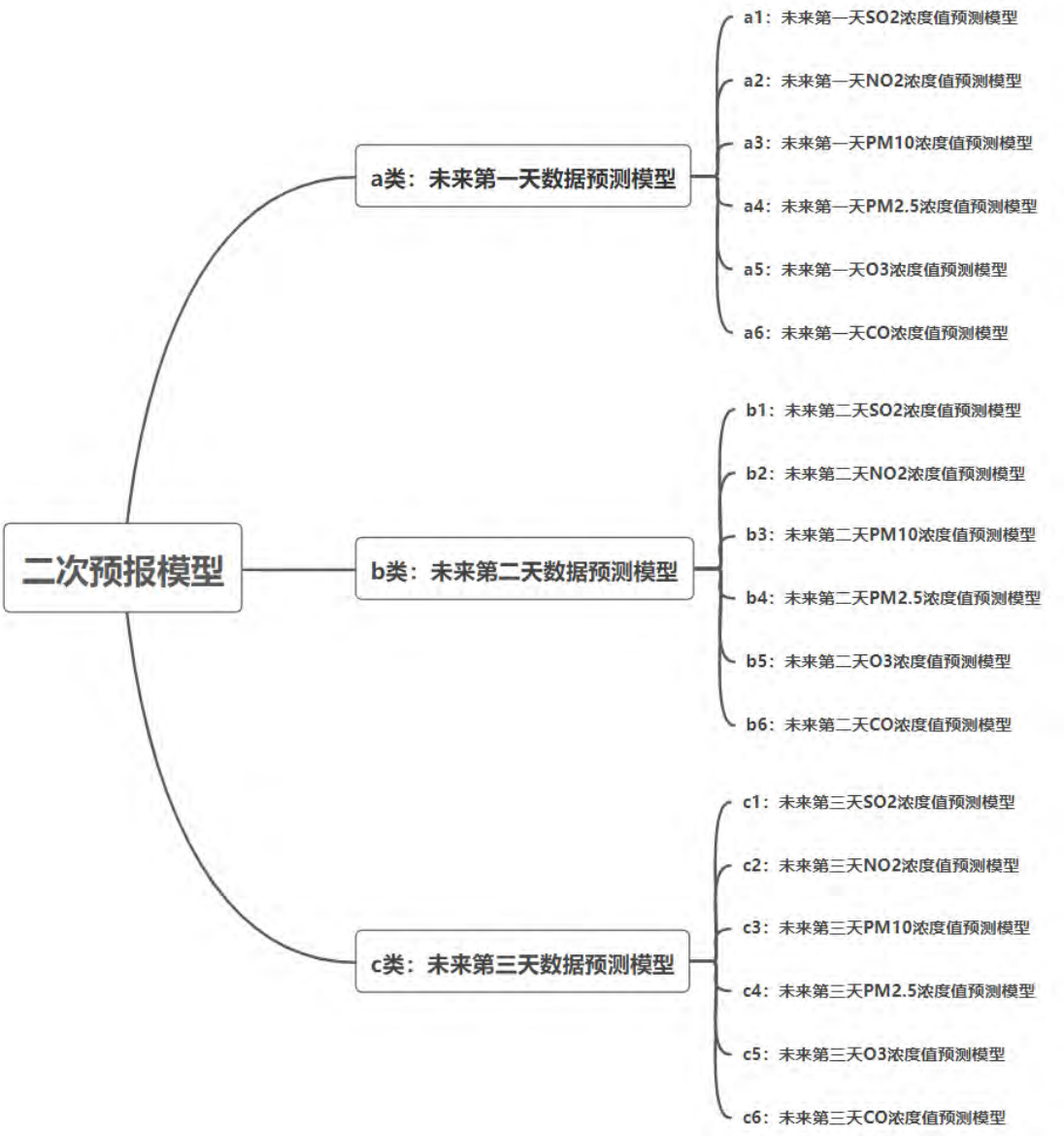


图 5-1 回归预测模型分类图

a 类模型：用于在当日预测未来第一天数据的模型，如在 2020/1/1 日预测 2020/1/1 日的数据。模型的输入为所有一次预报数据中模型运行日期与模型预测日期同一天（即未来第一天）的 6 种污染物浓度预测以及 15 种气象条件预测。根据模型的输出，分为 a1、a2、a3、a4、a5、a6 共六个子模型，其中 a1 子模型输出值为未来第一天各小时实测 SO_2 浓度值，a2 子模型输出值为未来第一天各小时实测 NO_2 浓度值，a3 子模型输出值为未来第一天各小时实测 PM_{10} 浓度值，a4 子模型输出值为未来第一天各小时实测 $PM_{2.5}$ 浓度值，a5 子模型输出值为未来第一天各小时实测 O_3 浓度值，a6 子模型输出为未来第一天各小时实测 CO 浓度值。最终，使用该模型以 2021/7/13 日一次预报数据为输入，输出 2021/7/13 日各小时实测 6 种污染物浓度值，据此计算得出 2021/7/13 日单日 6 种实测污染物浓度值。

b 类模型：用于在当日预测未来第二天数据的模型，如在 2020/1/1 日预测 2020/1/2 日的数据。模型的输入为所有一次预报数据中模型运行日期在模型预测日期前一天（即未来第二天）的 6 种污染物浓度预测以及 15 种气象条件预测。根据模型的输出，分为 b1、b2、b3、b4、b5、b6 共六个子模型，其中 b1 子模型输出值为未来第二天各小时实测 SO_2 浓度值，b2 子模型输出

值为未来第二天各小时实测 NO_2 浓度值，b3 子模型输出值为未来第一天各小时实测 PM_{10} 浓度值，b4 子模型输出值为未来第二天各小时实测 $PM_{2.5}$ 浓度值，b5 子模型输出值为未来第二天各小时实测 O_3 浓度值，b6 子模型输出为未来第二天各小时实测 CO 浓度值。最终，使用该模型以 2021/7/13 日一次预报数据为输入，输出 2021/7/14 日各小时实测 6 种污染物浓度值，据此计算得出 2021/7/14 日单日 6 种实测污染物浓度值。

c 类模型：用于在当日预测未来第三天数据的模型，如在 2020/1/1 日预测 2020/1/3 日的数据。模型的输入为所有一次预报数据中模型运行日期在模型预测日期前两天（即未来第三天）的 6 种污染物浓度预测以及 15 种气象条件预测。根据模型的输出，分为 c1、c2、c3、c4、c5、c6 共六个子模型，其中 c1 子模型输出值为未来第三天各小时实测 SO_2 浓度值，c2 子模型输出值为未来第三天各小时实测 NO_2 浓度值，c3 子模型输出值为未来第三天各小时实测 PM_{10} 浓度值，c4 子模型输出值为未来第三天各小时实测 $PM_{2.5}$ 浓度值，c5 子模型输出值为未来第三天各小时实测 O_3 浓度值，c6 子模型输出为未来第三天各小时实测 CO 浓度值。最终，使用该模型以 2021/7/13 日一次预报数据为输入，输出 2021/7/15 日各小时实测 6 种污染物浓度值，据此计算得出 2021/7/15 日单日 6 种实测污染物浓度值。

综上所述，以 b2 子模型为例，即使用二次预报模型预测未来第二天 NO_2 浓度值，解决问题 3 的技术路线如下图 5-2 所示：

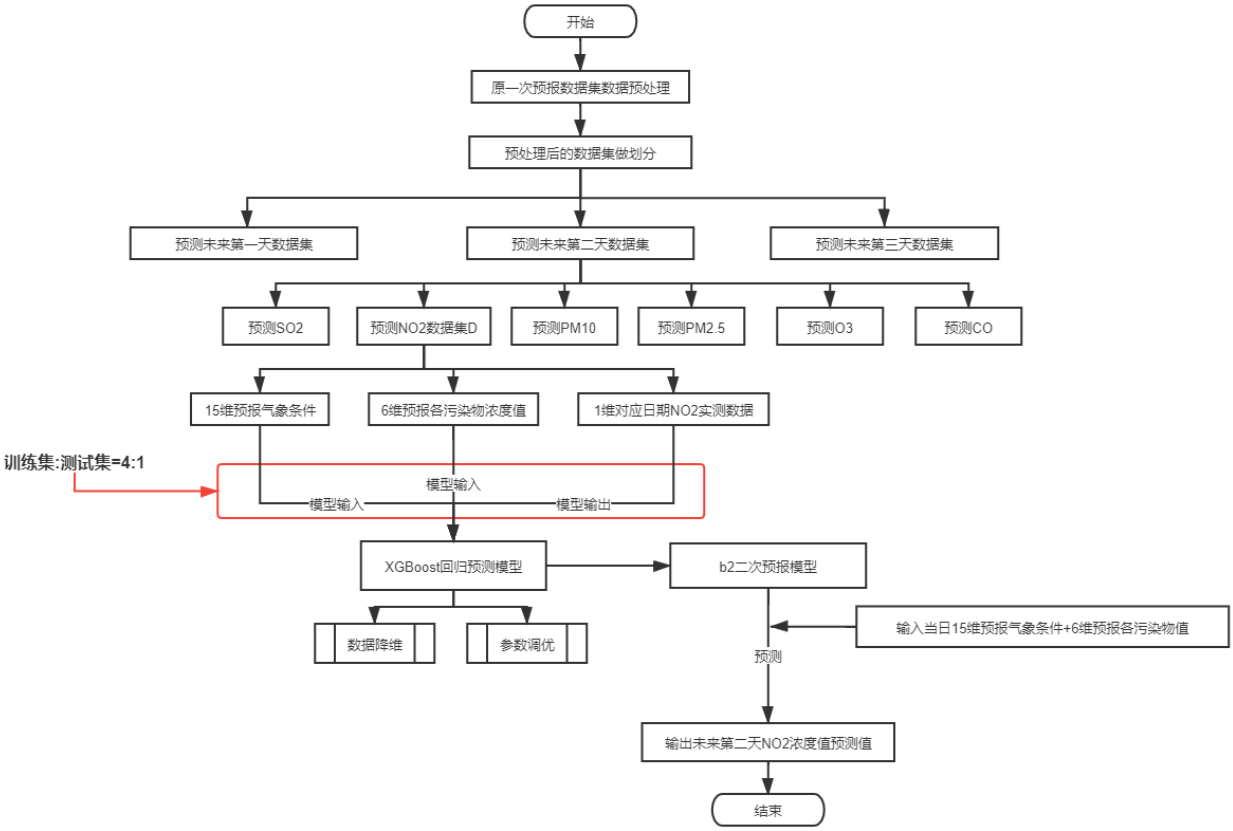


图 5-2 问题 3 技术路线图

5.2 数据预处理

监测点 B、C 的逐小时污染物浓度与气象实测数据同样存在缺失值与异常值，因此需要先执行 4.2 中叙述的数据预处理方案，由于过程相同，不在此赘述，本节将介绍后续的数据预处理步骤。

5.2.1 填补单日浓度值、AQI 值与首要污染物

根据监测点 A、B、C 的逐小时污染物实测/一次预报浓度值计算出对应日期的单日浓度值，其中 CO、SO_2、NO_2、PM_{10}、$PM_{2.5}$ 的单日浓度值可通过 24 小时平均计算，臭氧的单日浓度值需要根据 3-2 式最大 8 小时滑动平均求得。

已完成填补的数据样例见下表 5-1：

表 5-1 数据样例表

实测日期	地点	SO_2	NO_2	PM_{10}	$PM_{2.5}$	O_3	CO	AQI	首要污染物
2019/11/26	**监测点 A**	12	53	84	48	112	1.1	67	['NO_2', 'PM_{10}', '$PM_{2.5}$']
2019/11/27	**监测点 A**	17	60	94	49	138	1	82	['O_3']
2019/11/28	**监测点 A**	9	30	38	12	68	0.9	38	无首要污染物
2019/11/29	**监测点 A**	9	41	44	18	74	0.8	52	['NO_2']
2019/11/30	**监测点 A**	13	57	68	35	150	1	92	['O_3']
2019/12/1	**监测点 A**	15	56	77	37	97	1	70	['NO_2']

5.2.2 数据拆分与重新构建

将监测点 A、B、C 每地的逐小时污染物浓度与气象一次预报数据拆分为分别预测未来第一天、第二天和第三天的三张数据表。

如：在 2019/11/26 当天预测得到的未来三天的逐小时一次预报数据中，将 2019/11/26 日的数据放入第一张表，将 2019/11/27 日的数据放入第二张表、2019/11/28 日的数据放入第三张表。

将上一步生成的三张一次预报数据表分别与逐小时实测数据纵向合并，构建成一次预报数据与实测数据并存表（监测点 A、B、C 均做相同的处理）。

在合并时其中要注意数据对齐：逐小时实测数据中经预处理剔除的数据行在一次预报数据表中也相应删除。

最后，为了提升模型的普适性，A、B、C 三地的数据集进行横向合并，总共条 74,281 记录。

5.3 预测模型建模

5.3.1 XGBoost 回归预测模型原理

XGBoost9[7] 是在 GBDT 的基础上对 boosting 算法进行的改进，其内部决策树使用的是回归树。XGBoost 是由 k 个基模型组成的一个加法模型，假设第 t 次迭代要训练的树模型是$f_t(x_i)$，则有：

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{5-1}$$

其中，$\hat{y}_i^{(t)}$表示第t次迭代后样本i的预测结果，$\hat{y}_i^{(t-1)}$表示前$t-1$棵树的预测结果，$f_t(x_i)$表示第 t 棵树的模型，所有的树模型求和即为预测结果值。

XGBoost 的损失函数由预测值$\hat{y}_i$与真实值y_i进行表示，n为样本数量：

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i) \tag{5-2}$$

其目标函数由两项构成，分别为损失函数与正则化项$\sum_{i=1}^{t} \Omega(f_i)$，该项将全部树的复杂度

进行求和，用于抑制模型的复杂度，从而避免过拟合发生。公式表示为：

$$Obj = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i\right) + \sum_{i=1}^{t} \Omega\left(f_i\right) \tag{5-3}$$

假定g_i表示损失函数的一阶导数项，h_i表示损失函数的二阶导数项，Δx对应正在训练的第t棵树$f_t(x_i)$，则此时损失函数可以表示为：

$$l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) = l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \tag{5-4}$$

将公式 5-x 展示的二阶展开式代入 XGBoost 目标函数中，可得：

$$Obj^{(t)} \simeq \sum_{i=1}^{n} \left[l(y_i, \hat{y}_i^{t-1}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) + C \tag{5-5}$$

由于$\hat{y}_i^{(t-1)}$已知，故$l(y_i, \hat{y}_i^{t-1})$为常数。因此，去掉目标函数中的所有常数项可得：

$$Obj^{(t)} \simeq \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \tag{5-6}$$

通过上述 XGBoost 目标函数构建过程，可以看到 XGBoost 使用二阶展开、引入正则化项等方式提高预测精度，因此该模型具有较好的回归预测功能。XGBoost 算法的核心过程总结如下：

step1. 不断地添加树，不断地进行特征分裂来生成一棵新树，每次添加一个树，其实是以上次的预测数据为基础学习了一个新的函数$f_t(x_i)$，去拟合上次预测学习的残差。

step2. 模型训练完成后，XGBoost 模型由 k 个树模型组成，要预测一个样本的特征，即根据该样本的输入特征，在每棵树中找到一个对应的叶子节点，每个叶子节点对应一个值。

step3. 将 k 棵树对应的叶子节点的值加起来就得到了模型的输出值，即样本某个特征的预测值。

5.3.2 模型调优

5.3.2.1 数据降维

本文构建的基于 XGBoost 的回归预测模型，输入数据的维度为 21，最终得到未来三天内某个污染物浓度值。由于输入数据的维度较高，为了剔除一些与输出值相关性较低的特征，提高模型的精确度，需要对数据做降维处理。而由于 XGBoost 算法目标函数中引入了正则化项，因此在构建模型的过程中，可以量化每个特征与输出值的相关性。首先以 21 维特征构建模型，模型构建完成后可以通过 XGBoost 中的一个打分函数，对 21 维特征进行打分，根据打分结果，适当剔除得分较低的特征。剔除部分特征后，再次对模型进行训练，并用测试集检验模型的准确度是否得到提升。这里以 a1、b2 两个子模型为例，描述数据降维过程。首先构建从未调优的 a1、b2 两个子模型，分别做出对 21 维特征的打分图，如下图 5-3、图 5-4 所示：

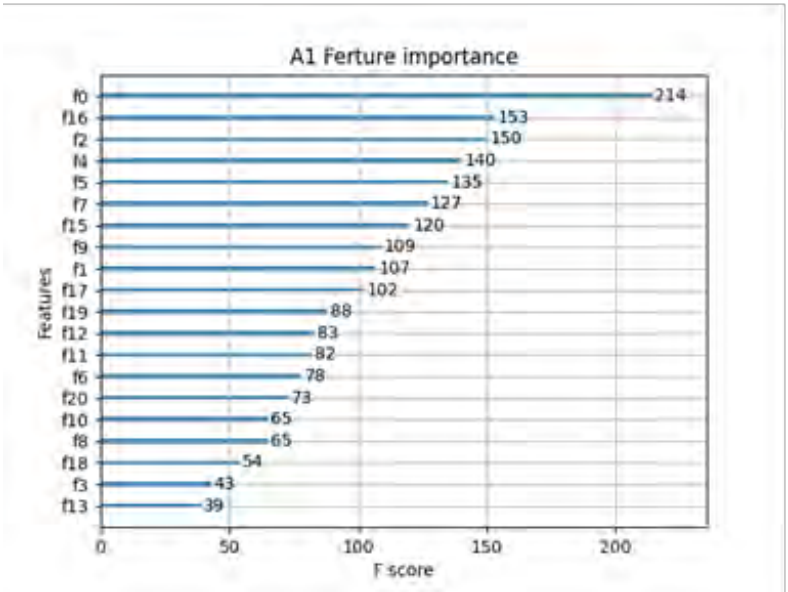


图 5-3 子模型 a1 的 21 维特征打分图

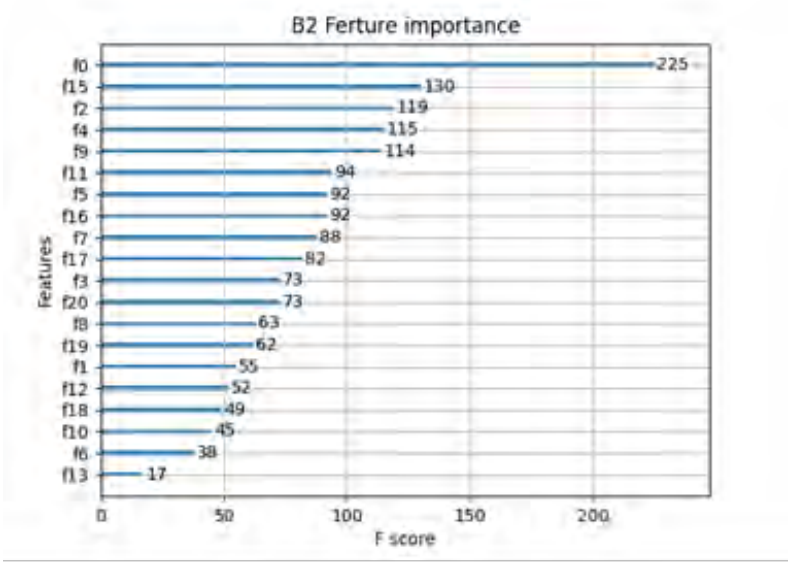


图 5-4 子模型 b2 的 21 维特征打分图

分别观察两图，图 5-3 描绘了构建未调优的 a1 子模型后，XGBoost 算法对第 0~20 维特征的打分情况，其中横坐标表示分数，纵坐标对应 0~20 维特征。可以看到，第 4 列、9 列、11 列、14 列、19 列对应的特征得分较低，说明其在回归预测过程中所对应的正则项值较小，与输出值的相关性较低，因此选择剔除 a1 子模型对应的数据表中的第 4、9、11、14、19 列对应的特征列，其分别是未来第一天各小时的一次预报湿度（%）、一次预报边界层高度（m）、一次预报感通热量（W/m^2）、一次预报短波辐射（W/m^2）、一次预报 PM_{10} 平均浓度($\mu g/m^3$)。同样针对 b2 子模型，剔除其对应数据表中的第 7、11、13、14、19 列对应的特征列，其分别是未来第二天各小时的一次预报雨量（mm）、一次预报感通热量（W/m^2）、一次预报长波辐射（W/m^2）、一次预报短波辐射（W/m^2）、一次预报 PM_{10} 平均浓度($\mu g/m^3$)。a1、b2 子模型分别剔除对应的特征列进行降维后，再次使用 XGBoost 算法构建对应的回归预测模型，观察模型对特征列的打分情况，如图 5-5、图 5-6 所示：

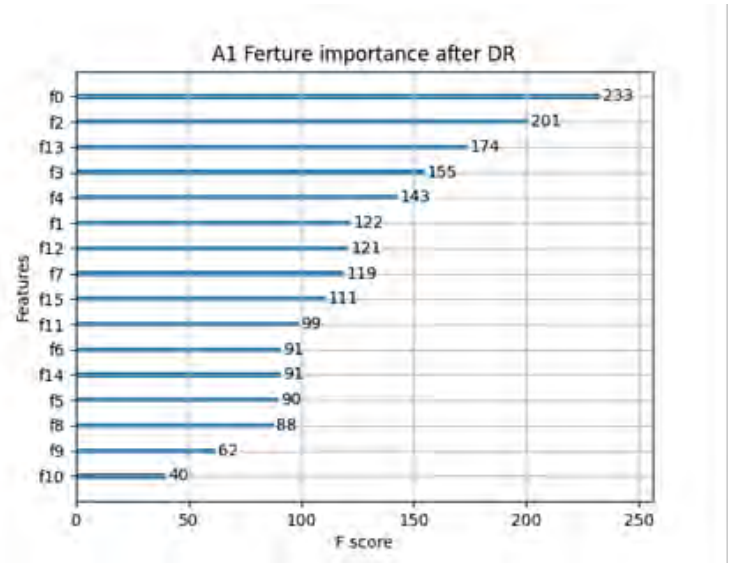


图 5-5 子模型 a1 剔除对应特征列后的打分图

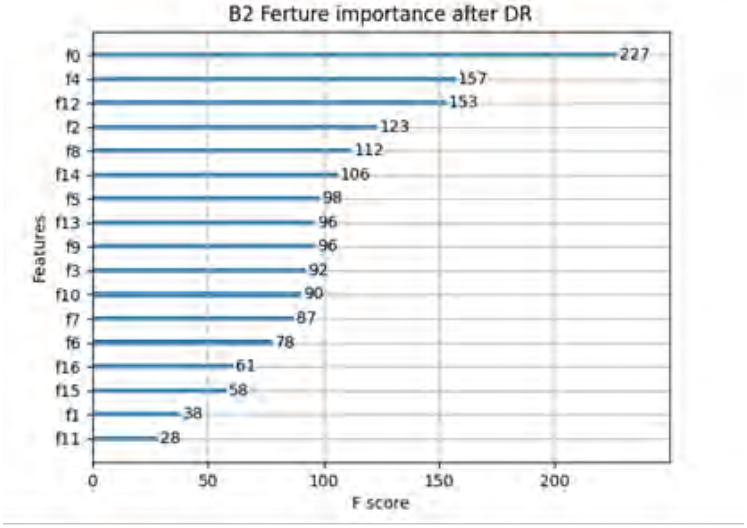


图 5-6 子模型 b2 剔除对应特征列后的打分图

对 a1、b2 子模型进行降维处理后，重新构建回归预测模型，以平均绝对误差作为评价模型好坏的标准，a1 子模型效果提升 8.68%，b2 子模型效果提升 9.09%。对其余 16 个子模型均进行降维处理，对应的子模型效果均有不同程度的提升。下表 5-2 给出了 18 个子模型在进行降维处理后，对应的模型效果变化情况：

表 5-2 降维处理后模型效果变化表

子模型	**a1**	**a2**	**a3**	**a4**	**a5**	**a6**
效果提升	8.68%	4.77%	8.12%	0.94%	3.32%	2.86%
子模型	b1	b2	b3	b4	b5	b6
效果提升	1.57%	9.09%	8.99%	1.21%	7.17%	6.98%
子模型	c1	c2	c3	c4	c5	c6
效果提升	0.24%	10.10%	6.83%	4.73%	4.36%	4.35%

5.3.2.2. 参数调优

数据集的特征与模型的选取决定了预测结果准确度的上限，而对模型参数的调整能够帮助无限接近准确度的上限值。为了进一步提高 XGBoost 回归预测模型对 A、B、C 三个监测点未来三天 6 种污染物浓度二次预报的准确度，同时使得二次预报模型预测结果中 AQI 预报值最大相对误差尽量小，且首要污染物预测准确度尽量高，本文对建立的三个基于 XGBoost 算法的二次预报回归预测模型进行了参数调优。

XGBoost 算法包含两种 booster，一种为树 booster，另一种为线性 booster，本文建立的是 XGBoost 回归预测模型，故选择树 booster。树 booster 的参数类型有三种，分别是通用参数、booster 参数、学习目标参数。本文主要选取 booster 参数中的三个超参数 eta、max_depth 以及 n_estimators 作为优化目标：

(1) eta 为学习率，与传统 GBM 算法中的 learning_rate 参数类似，可用于控制每一步的权重，选取合适的学习率参数可以提高模型的鲁棒性，默认值为 0.3，越小就需要加入越多的弱学习器，取值范围通常在 0.01~0.3。

(2) max_depth 表示基学习器的树最大深度，该参数用于避免过拟合，max_depth 值越大，模型会学习到更具体更局部的样本，默认值为 6，取值范围通常在 3~10。

(3) n_estimators 为梯度增强树 GB 的数量，也等于最大迭代的次数。n_estimators 过小容易欠拟合，而 n_estimators 过大则导致过拟合，默认值为 100。

本文以 a1、a5 子模型为例（a1、a5 子模型分别对应预测未来第一天 SO_2 浓度值、O_3 浓度值），描述参数调优的过程。学习率参数 eta 的取值区间规定在集合[0.0001, 0.001, 0.01, 0.1, 0.2, 0.3]中；树最大深度 max_depth 的取值区间规定在集合[3, 4, 5, 6, 7, 8]中；n_estimators 的取值区间规定在集合[50, 100, 150, 200, 250, 300, 350, 400, 500, 600, 800, 1000]中。模型的评价标准使用平均绝对误差，首先独立观察每个参数变化时模型的效果变化，如下图 5-7、图 5-8、图 5-9 所示：

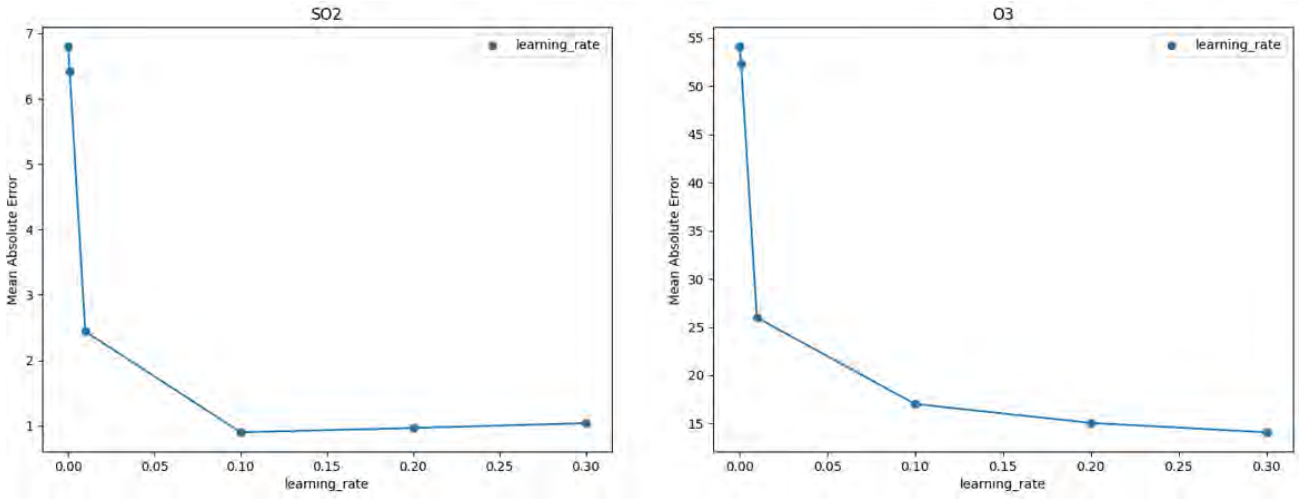


图 5-7 eta 变化时 a1、a5 子模型效果变化

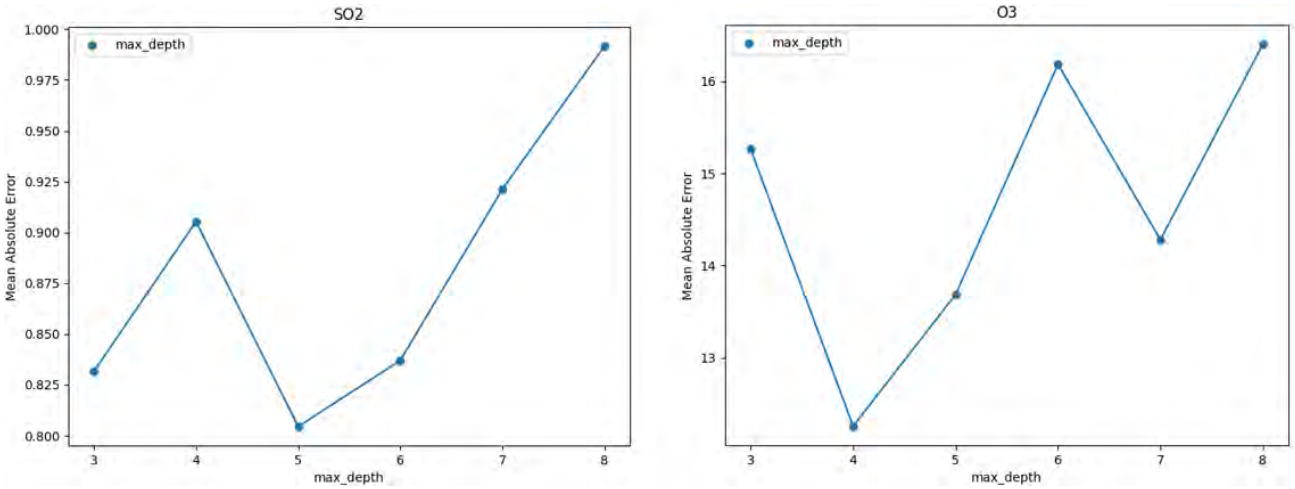


图 5-8 max_depth 变化时 a1、a5 子模型效果变化

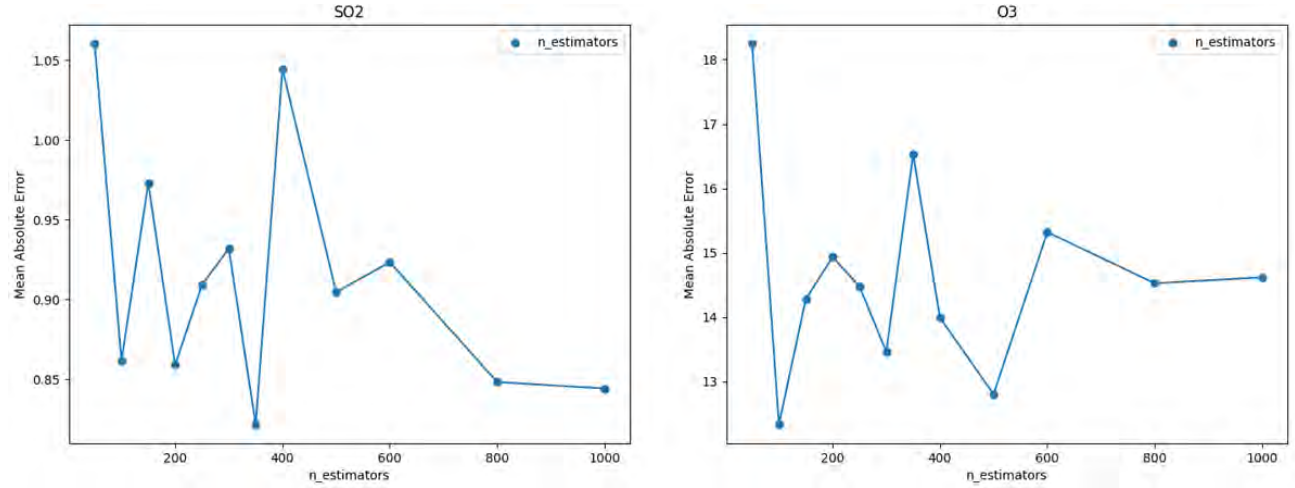


图 5-9 n_estimators 变化时 a1、a5 子模型效果变化

然后观察三个变量同时作用于模型时，模型效果的变化情况，三个参数共 432 种情况，按照对应模型均方根误差大小进行排序，如下图 5-10 所示。最终 a1 子模型对应的最优参数组合为 (learning_rate,n_estimators,max_depth)=(0.1,400,5)， a5 子模型的最优参数组合为 (learning_rate,n_estimators,max_depth)=(0.1,1000,5)。对其他 17 个子模型分别做相同处理，即可找到每个子模型对应的最优参数组合。

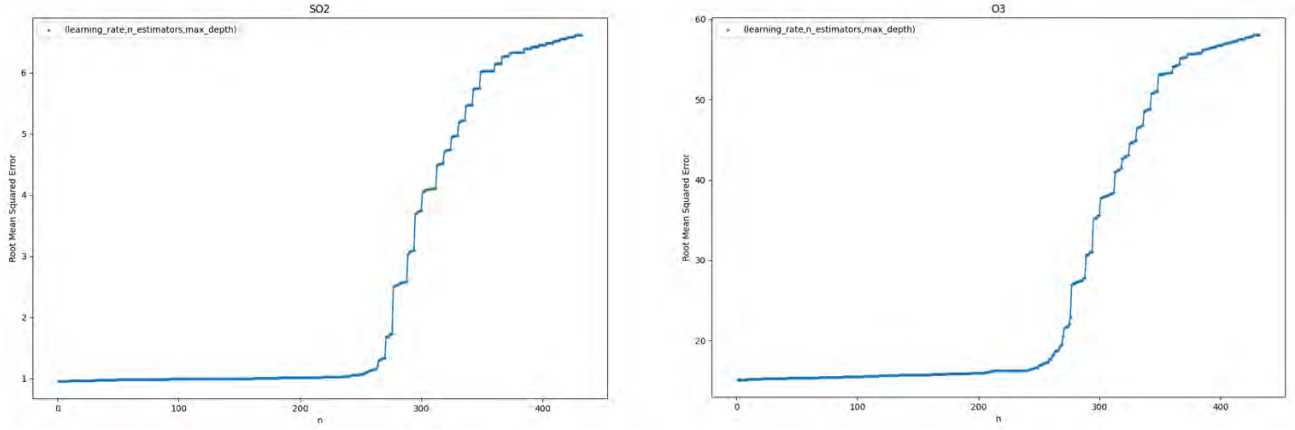


图 5-10 432 种参数组合 a1、a5 模型效果排序图

5.3.3 模型效果

三个模型对应的三个数据表，分别以 8:2 的比例划分训练集与测试集，经过数据降维与参数调优，达到最佳效果。下面本文对建立的二次预报模型进行效果测试，具体方法为挑选 A、B、C 监测点任意连续三日，根据本文建立的 a 模型预测 A、B、C 三点未来第一日各污染物浓度数据，b 模型预测 A、B、C 三点未来第二日各污染物浓度数据，c 模型预测 A、B、C 三点未来第三日各污染物浓度数据。并将 A、B、C 三点这三日的实测各污染物浓度数据与模型输出作对比，观察模型预测效果。这里选取 2020/8/30 、2020/8/31、2020/9/1 三日，对 A、B、C 监测地的各污染物浓度进行二次预报，并与这三日 A、B、C 监测点的实测污染物浓度进行对比，结果如下图 5-11、图 5-12、图 5-13 所示。

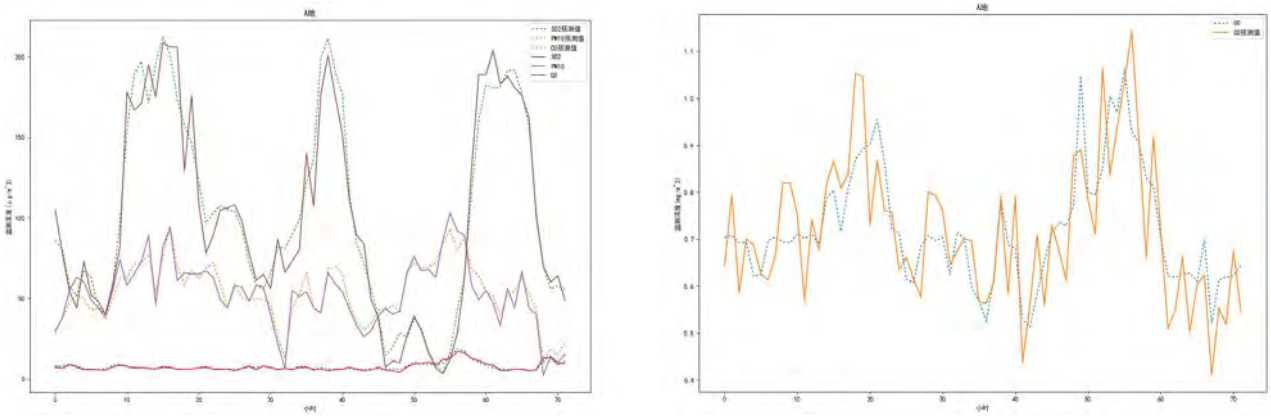


图 5-11 A 点二次预报与实测对比

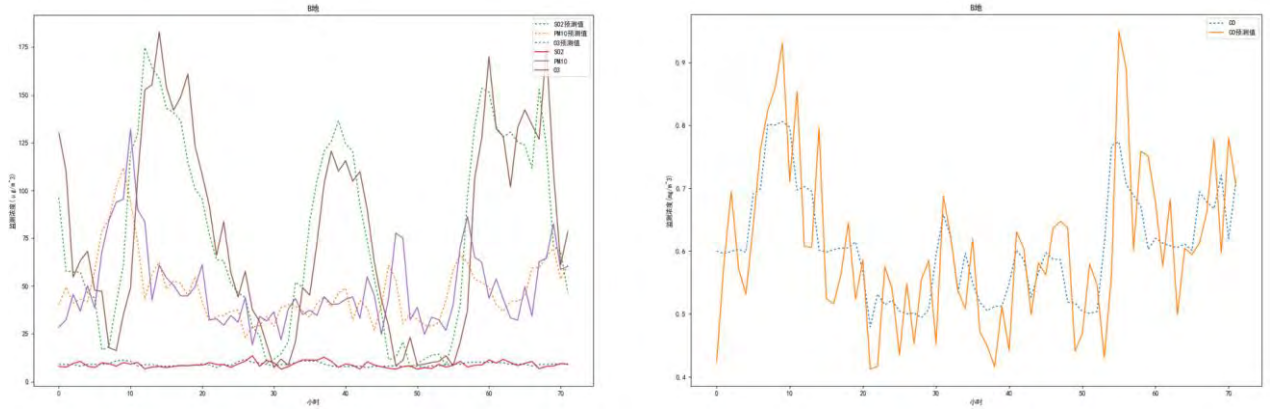


图 5-12 B 点二次预报与实测对比

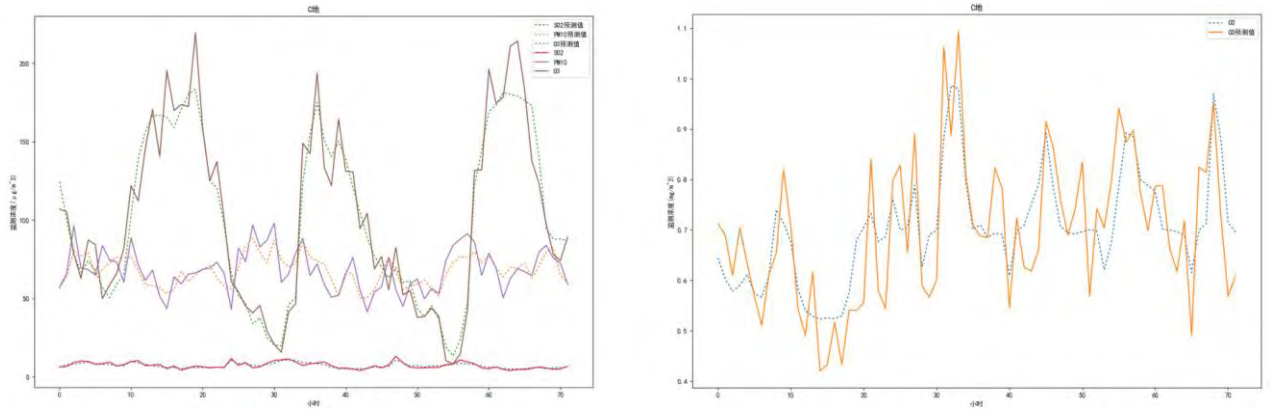


图 5-13 C 点二次预报与实测对比

由于 NO_2 与 SO_2 浓度变化曲线类似，$PM_{2.5}$ 与 PM_{10} 浓度变化曲线类似，为了使得图像更加清晰，删去了 NO_2、$PM_{2.5}$ 浓度值的二次预报与实测对比。此外，由于 CO 浓度单位与其他污染物浓度单位不同，因此将 CO 浓度值的二次预报与实测对比单独作出。

观察 A、B、C 监测点于 2020/8/30 、2020/8/31、2020/9/1 日的二次预报数据与实测数据对比图，虽在拐点处二次预报数据与实测数据偏差较为明显，但总体可以看出三个监测点的二次预报数据与实测数据均能较好的拟合。且经过均值计算得到每日各污染物浓度值后，拐点处偏差的影响效果会适当减小，更加说明在二次预报某监测点某日各污染物平均浓度时，本文建立的二次预报模型具有较高的准确度，即在误差允许的范围内可以认为本文建立的二次预报模型输出的预测值即为该日污染物平均浓度的实测值。

5.3.4 模型预测结果

使用本文建立的二次预报模型预测监测点 A、B、C 在 2021 年 7 月 13 日至 7 月 15 日 6 种常规污染物的单日浓度值，计算相应的 AQI 和首要污染物，将结果展示如下表 5-3：

表 5-3 二次预报模型预测结果表

预报日期	地点	二次模型日值预测							
		SO_2 ($\mu g/m^3$)	NO_2 ($\mu g/m^3$)	PM_{10} ($\mu g/m^3$)	$PM_{2.5}$ ($\mu g/m^3$)	O_3 最大八小时滑动平均 ($\mu g/m^3$)	CO (mg/m^3)	AQI	首要污染物
2021/7/13	监测点 A	6	17	21	6	106	0.5	55	O_3
2021/7/14	监测点 A	5	34	22	8	117	0.5	65	O_3
2021/7/15	监测点 A	5	23	24	8	120	0.6	67	O_3
2021/7/13	监测点 B	6	12	20	5	61	0.5	31	无首要污染物
2021/7/14	监测点 B	7	12	19	5	85	0.5	43	无首要污染物
2021/7/15	监测点 B	7	13	21	5	95	0.5	48	无首要污染物
2021/7/13	监测点 C	8	25	41	23	133	0.5	78	O_3
2021/7/14	监测点 C	11	24	37	20	134	0.5	79	O_3
2021/7/15	监测点 C	10	25	45	31	303	0.5	208	O_3

5.4 模型总结分析

下图 5-14 展示了本文建立的基于 XGBoost 算法的二次预报模型与题中所给的一次预报模型之间的对比情况。比较了一次预报模型、二次预报模型对 SO_2 单日浓度、NO_2 单日浓度、PM_{10} 单日浓度、$PM_{2.5}$ 单日浓度、O_3 单日浓度、CO 单日浓度、AQI 的预测值与实测数据之间的最大相对误差，同时也比较了一次预报模型与二次预报模型计算得出的每日首要污染物的平均准确率。可以看出，在对单日污染物浓度进行预报时，二次预报模型在 SO_2 单日浓度、NO_2 单日浓度、PM_{10} 单日浓度、O_3 单日浓度、AQI 的预测上均较一次模型有不同程度的提升，尤其是对 O_3 单日浓度以及 AQI 的预测与一次预报相比误差明显降低。

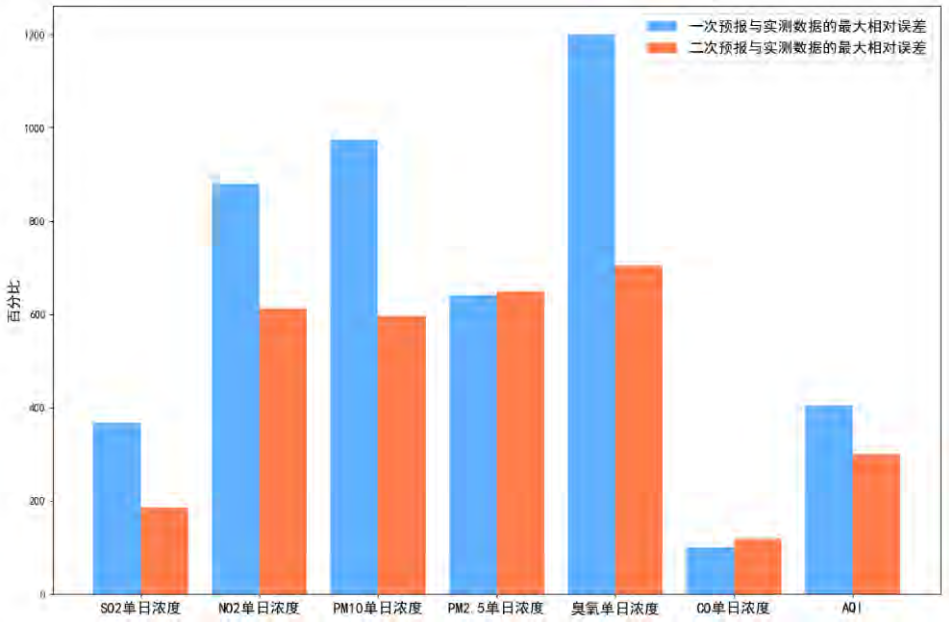


图 5-14 二次预报模型与一次预报模型对比图

首要污染物的准确率计算方式：当实际首要污染物只有一类时，预报正确的准确率即为100%，否则为 0%；当实际不存在首要污染物时，预报正确的准确率即为 100%，否则为 0%；当实际首要污染物有>1 类时，预报准确率=准确预报的污染物种类数/实际首要污染物种类数 * 100%。最后计算所有预报数据的平均准确率。从表 5-4 中可以看出，二次预报模型对于每日首要污染物预报的平均准确率相较于一次预报模型也有很大提升。

表 5-4 二次预报模型与一次预报模型对比表

	一次预报	二次预报	二次预报相比一次预报的提升
首要污染物预报平均准确率	7.69%	40.05%	32.36%

观察图 5-14，在二次预报模型中 $PM_{2.5}$ 单日浓度、CO 单日浓度的预测效果不如一次预报模型，误差不降反升。主要原因可能是 CO 浓度量级较小，且时间安排较紧张，数据量较大，没有对二次预报模型进行充分的训练与测试，导致二次预报模型在预测 $PM_{2.5}$ 单日浓度、CO 单日浓度时出现了较大偏差。我们会在未来的工作中，继续完善二次预报模型的精度，探究造成预测 $PM_{2.5}$ 单日浓度、CO 单日浓度时出现较大偏差的具体原因。

综上，本文建立了三个基于 XGBoost 算法的数据回归预测模型，分别用于在当日预测未来第一天各污染物浓度、未来第二天各污染物浓度、未来第三天各污染物浓度。每个模型又根据预测污染物种类的不同，各分为 6 个子模型，分别对应对 SO_2 单日浓度、NO_2 单日浓度、PM_{10} 单日浓度、$PM_{2.5}$ 单日浓度、O_3 单日浓度、CO 单日浓度的预测。本文对二次预报模型进行了较为充分的训练与测试，具体阐述了模型建立与调优的过程，在 5.3.3 小节中选取任意连续三天对 A、B、C 监测点各污染物浓度进行预测，并与其对应的实测值进行对比；然后，在 5.3.4 小节中对问题 3 中提出的问题进行了回答；此外，给出了二次预报模型的整体效果，其在多个污染物浓度预测上都优于一次预报模型，尤其是对 O_3 单日浓度值、AQI 的预测，且对首要污染物预报的平均准确率也较一次预报模型有一定提升；最后，分析了本文建立的二次预告模型目前存在的不足，并对未来的工作方向进行了讨论。

6. 问题 4 分析与求解

6.1 思路分析

在本题中，要求使用附件 1、3 中的数据，根据 A、A1、A2、A3 四个监测点的一次预报数据与实测数据，建立能够用于四个监测点的协同预报模型，用来预测 6 种常规污染物的单日浓度值，目的是利用 A、A1、A2、A3 四个监测点的位置相关性和天气条件相似性，让 AQI 与首要污染物的预报值的预测准确度相比于独立二次预报模型可以进一步提高。并最终使用该模型预报三个监测点在 2021 年 7 月 13 日至 7 月 15 日 6 种常规污染物的单日浓度值、相应的 AQI 和首要污染物。

观察附件 1、3 的数据可以发现问题 4 的样本数据格式基本与问题 3 基本相同，区别只是多了三个监测点 A1、A2、A3 的数据，经过第 5 章的结果分析可以发现已有的独立预测模型具有较好的预测性能，因此在本章的建模中仍然沿用解决问题 3 时所使用的 XGboost 回归预测模型。但为了实现协同预报，需要对数据进行一定的属性转换以及拆分等预处理操作，并建立二次预报的协同预测模型，该模型包括四个分别用来预测 A、A1、A2、A3 监测点的污染物浓度的子模型。

其中，数据预处理操作包括如下步骤：

step1. 首先需要将附件 1、3 的 监测点 A、A1、A2、A3 的一次预报数据进行纵向合并。

step2. 执行 4.2、5.2 章的数据预处理流程，包括异常值与缺失值处理，数据拆分与重新构建等。

step3. 为利用 A、A1、A2、A3 四个监测点的位置和天气条件的相关性，如下图 6-1 所示，计算各个点之间的距离、某点在某个风向时对其他点的风力影响距离，对 ***step1*** 中的数据中引入相应的“绝对距离”、“风力影响距离”字段，并删除原数据“风向”字段。绝对距离和风力影响距离可以大致描绘两监测点间的位置相关性，进而可以决定两个监测点天气条件的相互影响。

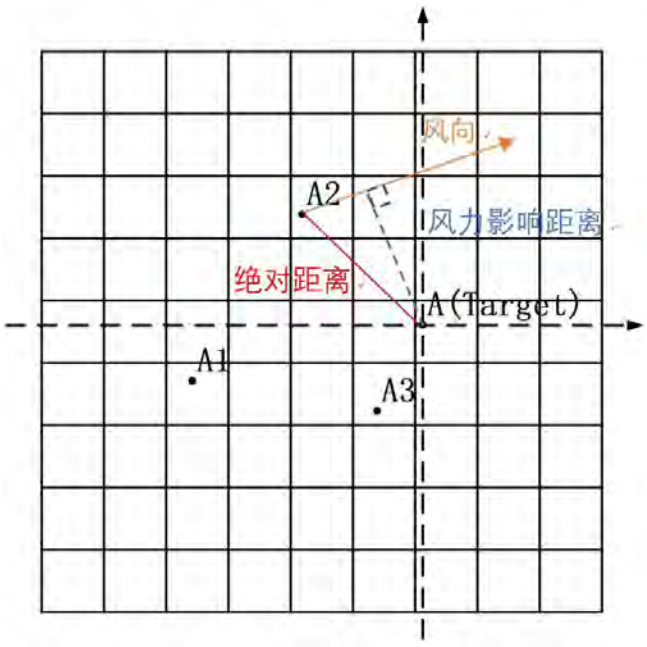


图 6-1 监测点间绝对距离、风向与风力影响距离示例

模型输入数据构建好后，本文基于第 5 章中的回归预测模型进行参数调整及优化，构建协同预测模型.该模型包括四个子模型，每个子模型结构与 5.3 节中的回归预测模型相同，但其输入的样本数据互不相同，具体区别将在 6.3 节中进行叙述。

综上所述，解决问题 4 的技术路线如下图 6-2 所示：

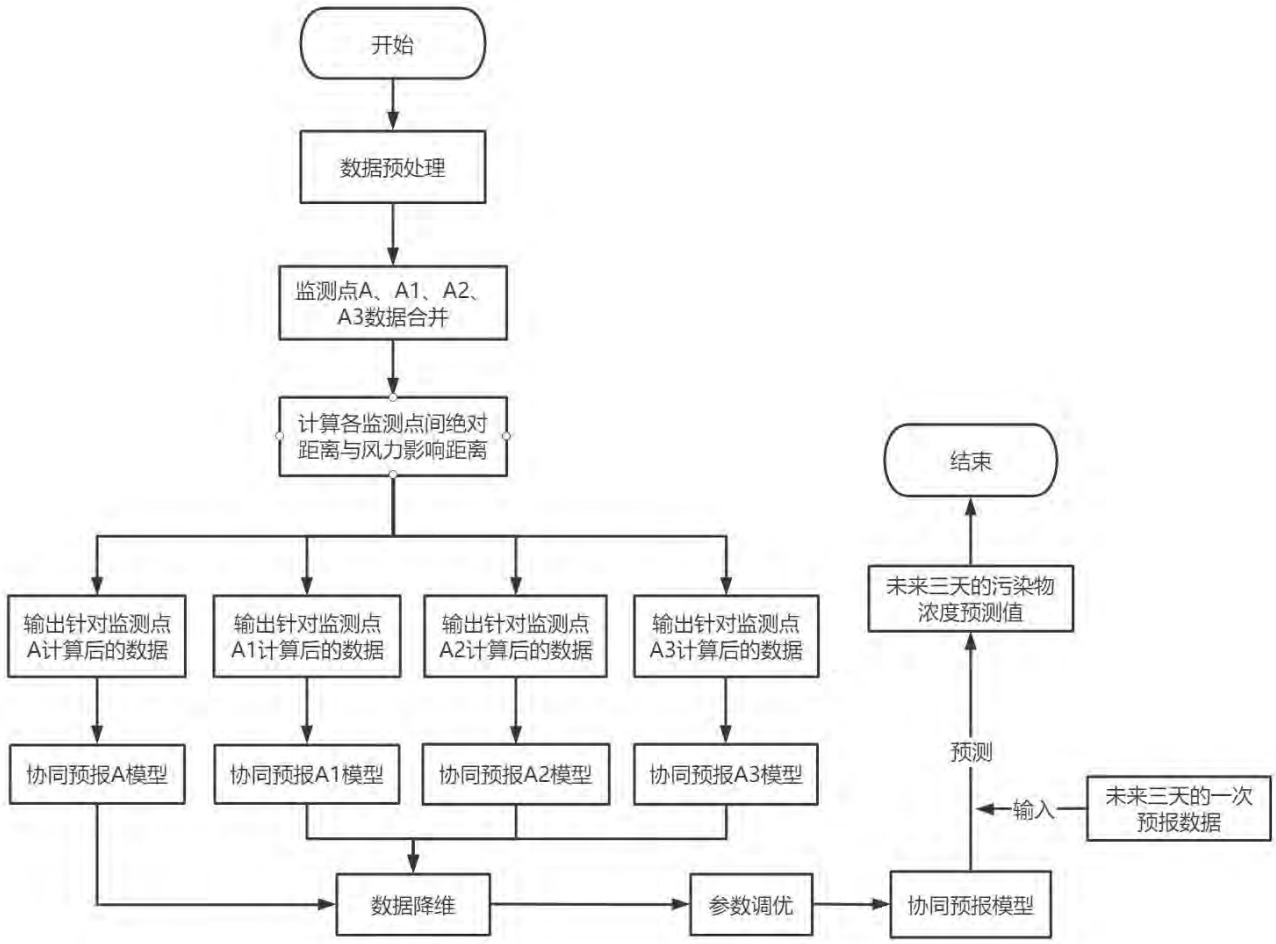


图 6-2 问题 4 技术路线

6.2 数据预处理

同样的，监测点 A、A1、A2、A3 的逐小时污染物浓度与气象实测数据同样存在缺失值与异常值，并且需要同样的拆分与重新构建，因此需要先执行 4.2 节和 5.2 节中叙述的数据预处理方案，由于过程相同，不在此赘述，本节将介绍其余的数据预处理步骤。

6.2.1 监测点 A、A1、A2、A3 数据纵向合并

将附件 1、3 中监测点 A、A1、A2、A3 的逐小时污染物浓度一次预报数据纵向连接，之后再与监测点 A、A1、A2、A3 的逐小时污染物浓度实测数据纵向合并。预测部分的字段将作为模型的样本数据，而实测部分的字段将作为模型的目标数据。

6.2.2 绝对距离计算

由于我们假设题目中监测点 A、A1、A2、A3 位置关系可以在二维平面上近似表示，所以两个监测点A_i，A_j间的直线绝对距离 d 可以由下式 6-1 计算：

$$d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{6-1}$$

其中，(x_i, y_i)表示A_i在二维坐标系上的坐标，同理，(x_j, y_j)表示A_j的坐标。

经计算，监测点 A、A1、A2、A3 间的绝对距离如下表 6-1 所示（单位：km）：

表 6-1 监测点 A、A1、A2、A3 间绝对距离

A				
A1	14.62			
A2	10.11	12.35		
A3	6.03	11.54	13.04	
	A	**A1**	**A2**	**A3**

6.2.3 风向角度计算与影响距离转换

参考下图 6-3，计算监测点A_i受 A_j风力影响的距离 w 可以通过角 α 的正弦与绝对距离 d 相乘求得。

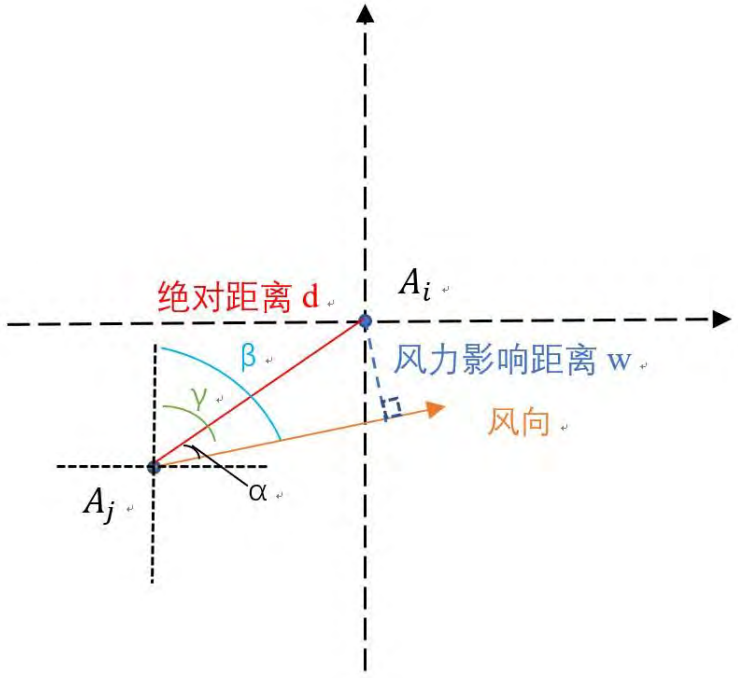


图 6-3 监测点间风向角度计算与影响距离转换示意图

如式 6-2 所示：

$$w = dsin(\alpha) \tag{6-2}$$

而α由可以通过$\beta - \gamma$求得，角γ即为数据中的风向角度，角β满足式 6-3：

$$\beta = tan^{-1}(\frac{x_i - x_j}{y_i - y_j}) \tag{6-3}$$

因此 w 的计算公式 6-4 如下：

$$w = dsin(tan^{-1}\left(\frac{x_i - x_j}{y_i - y_j}\right) - \gamma) \tag{6-4}$$

通过 6.2.2 中计算出的绝对距离与数据中的风向字段，即可计算风力影响距离。计算完毕后将此结果替换数据集中的风向字段。

最终构建好的数据集共 114 个特征字段，如下表 6-2 所示：

表 6-2 监测点 A、A1、A2、A3 合并数据集特征表

属性名称	变量名	属性名称	变量名
模型运行日期	date	风力影响距离（km）(4 列)	w_Ai(0≤i＜4)
预测时间	predict_date	绝对距离（km）(4 列)	d_Ai(0≤i＜4)
SO_2 预报浓度(μ g/m³)	SO2	大气压（Kpa）	Kpa
NO_2 预报浓度(μ g/m³)	NO2	感热通量（W/m²）	sense_flux
PM_{10} 预报浓度(μ g/m³)	PM10	潜热通量（W/m²）	latent_flux
$PM_{2.5}$ 预报浓度(μ g/m³)	PM25	长波辐射（W/m²）	long_wave
O_3 预报浓度(μ g/m³)	O3	短波辐射（W/m²）	short_wave
CO 预报浓度(mg/m³)	CO	地面太阳能辐射（W/m²）	solar_radiation
近地 2 米温度（℃）	2m_temp	监测点 A1 的一次预报数据（20 列）	xxx_A1
地表温度（K）	sur_temp	监测点 A2 的一次预报数据（20 列）	xxx_A2
比湿（kg/kg）	spec_humid	监测点 A3 的一次预报数据（20 列）	xxx_A3
湿度（%）	humid	监测点 A 的实测数据（6 列）	xxx_A_real
近地 10 米风速（m/s）	near_speed	监测点 A1 的实测数据（6 列）	xxx_A1_real
雨量（mm）	rain	监测点 A2 的实测数据（6 列）	xxx_A2_real
云量	cloud	监测点 A3 的实测数据（6 列）	xxx_A3_real
边界层高度（m）	height		

6.3 协同预测模型建模

6.3.1 建立模型

本协同预测模型包含四个分别针对于监测点 A、A1、A2、A3 子模型：

A 模型：使用的输入数据中，w_A＝0，w_A1、w_A2、w_A3 根据风向数据计算监测点 A1、A2、A3 对监测点 A 的风力影响距离计算而得。d_A＝0，d_A1、d_A2、d_A3 根据监测点 A1、A2、A3 与监测点 A 的直线绝对距离计算而得。

A1 模型：使用的输入数据中，w_A1＝0，w_A、w_A2、w_A3 根据风向数据计算监测点 A、A2、A3 对监测点 A1 的风力影响距离计算而得。d_A1＝0，d_A、d_A2、d_A3 根据监测点 A、A2、A3 与监测点 A1 的直线绝对距离计算而得。

A2 模型：使用的输入数据中，w_A2＝0，w_A、w_A1、w_A3 根据风向数据计算监测点 A、A1、A3 对监测点 A2 的风力影响距离计算而得。d_A2＝0，d_A、d_A1、d_A3 根据监测点 A、A1、A3 与监测点 A2 的直线绝对距离计算而得。

A3 模型：使用的输入数据中，w_A3＝0，w_A、w_A1、w_A2 根据风向数据计算监测点 A、A1、A2 对监测点 A3 的风力影响距离计算而得。d_A3＝0，d_A、d_A1、d_A2 根据监测点 A、A1、A2 与监测点 A3 的直线绝对距离计算而得。

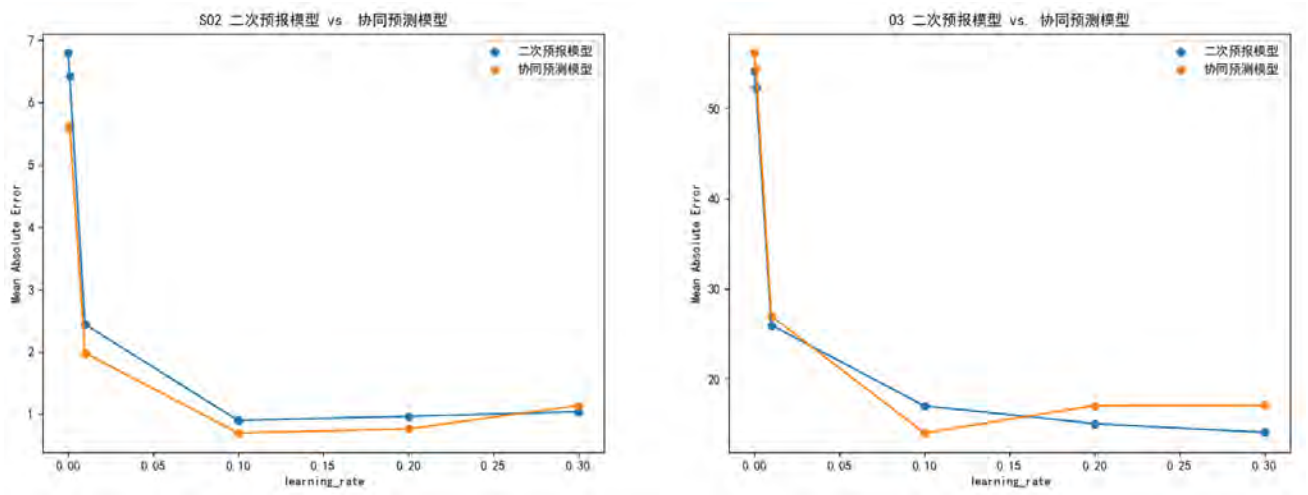


图 6-4 A 模型 learning_rate 参数调优过程

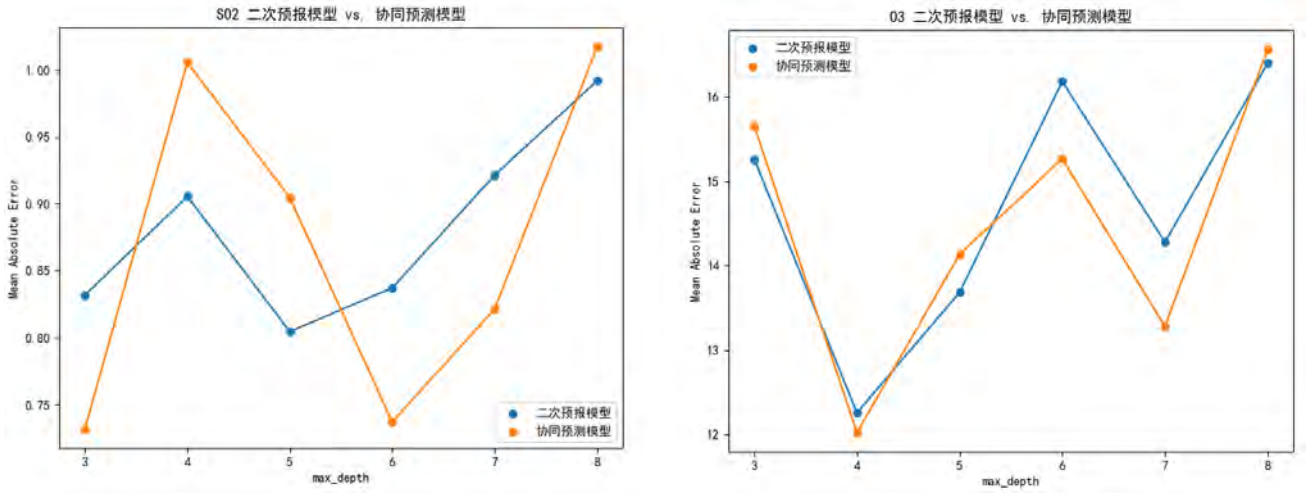


图 6-5 A 模型 max_depth 参数调优过程

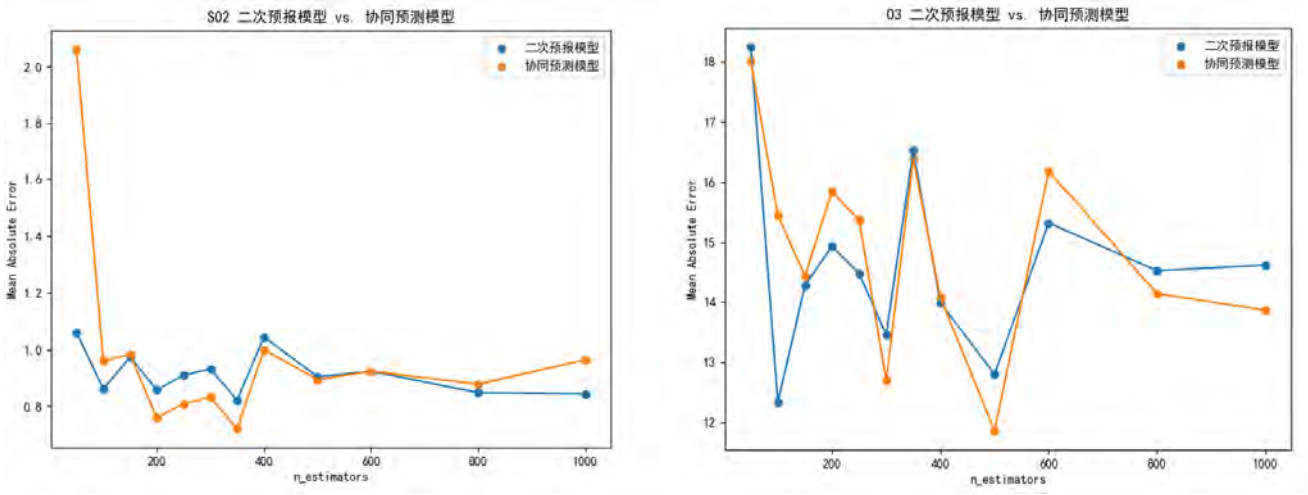


图 6-6 A 模型 n_estimators 参数调优过程

如图 6-4、6-5、6-6 所示为 A 模型为样例的模型参数调优的过程，并与上章独立二次预报模型的过程做了对比，从中可以看到，不同参数下，模型的误差最小值均得到了优化。

6.3.2 AQI 与首要污染物计算

使用本文建立的协同预报模型预测监测点 A、A1、A2、A3 在 2021 年 7 月 13 日至 7 月 15 日 6 种常规污染物的单日浓度值，计算相应的 AQI 和首要污染物，将结果展示如下表 6-3：

表 6-3 二次预报模型预测结果表

预报日期	地点	二次模型日值预测							
		SO_2 ($\mu g/m^3$)	NO_2 ($\mu g/m^3$)	PM_{10} ($\mu g/m^3$)	$PM_{2.5}$ ($\mu g/m^3$)	O_3 最大八小时滑动平均 ($\mu g/m^3$)	CO (mg/m^3)	AQI	首要污染物
2021/7/13	监测点 A	5	21	25	10	90	0.5	45	无首要污染物
2021/7/14	监测点 A	5	27	23	9	112	0.4	60	O_3
2021/7/15	监测点 A	5	20	37	13	128	0.5	74	O_3
2021/7/13	监测点 A1	7	20	37	11	73	0.5	37	无首要污染物
2021/7/14	监测点 A1	8	24	66	13	128	0.3	74	O_3
2021/7/15	监测点 A1	8	19	45	17	99	0.4	50	O_3
2021/7/13	监测点 A2	5	28	33	9	110	0.5	59	O_3
2021/7/14	监测点 A2	4	22	68	8	97	0.4	59	PM_{10}
2021/7/15	监测点 A2	5	25	41	11	89	0.4	45	无首要污染物
2021/7/13	监测点 A3	5	14	19	8	90	0.4	45	无首要污染物
2021/7/14	监测点 A3	6	10	24	10	67	0.6	34	无首要污染物
2021/7/15	监测点 A3	3	7	17	9	121	0.5	68	O_3

6.3.3 结果分析

如图 6-7 所示是协同二次预报模型与一次预报模型、独立二次预报模型的最大相对误差

比较图，所用的数据来自于监测点 A。

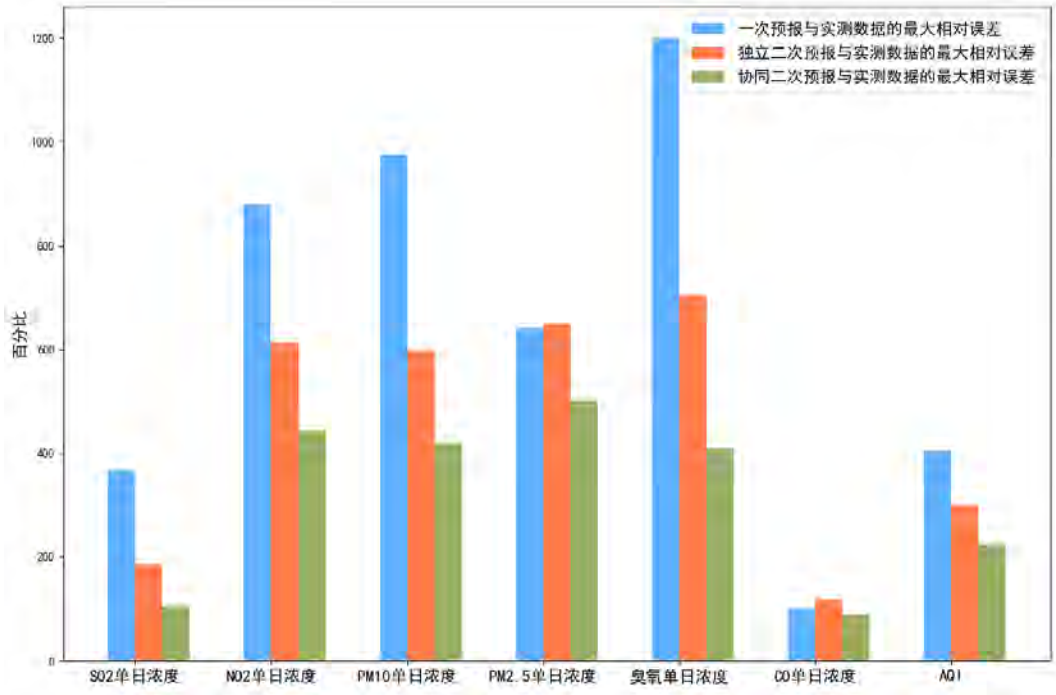


图 6-7 协同二次预报模型最大相对误差比较图

由该图可以基本得出结论：得益于协同预报模型综合考虑了监测点 A1、A2、A3 与监测点 A 的位置、天气条件的相关性，将监测点附近地区的风力、温度、湿度等天气影响因子能够更准确地作用于模型之中，使得预报的各项指标误差均有所降低，二次预报模型的准确度有了进一步的提升。同时，由于某地区的污染物浓度也与附近地区的污染物浓度息息相关，所以将风力影响距离引入模型后，其污染物浓度二次预报准确度也有了提升。

7. 模型评价

7.1 模型的优点

对于问题二，本文采用了拉格朗日插值法、绘制箱线图等方法对原数据集的缺失值、异常值进行预处理，以避免这些值对模型的构建产生负影响。根据题意，为了发现气象条件的变化对各污染物浓度的变化的影响，本文根据预处理后的数据集构建了新的 22 维数据集，规定三天为一个时间周期，记录每个时间周期内气象条件的变化以及各污染物浓度的变化。在做聚类分析之前，本文先对每个气象条件与每个污染物浓度做了相关性分析。本文使用基于 EM 的 GMM 对新数据集做聚类分析。得到聚类结果后，将相关性分析结果与气象相关文献结合，对聚类结果进行解读，使得聚类结果中气象条件的分类更加合理，对各类气象条件的特征阐述更加具体。

对于问题三，题目中提及一次预报对邻近日期的准确度较高，理论上二次预报对邻近日期的准确度也较高。因此本文将预处理后的数据集分为三个，提出了三个基于 XGBoost 算法的回归预测模型，分别对应预测未来第一天的数据、第二天的数据、第三天的数据，并且每个模型分为 6 个子模型，对应每个污染物浓度值的预测，这样做可以使得每个子模型能更好的学习每个数据集的特征，从而得到更准确的预测值。此外，XGBoost 高效、精度高的特点也使得模型的预测更加贴近实测值。本文描述了一个具体的测试方法对模型效果进行测试，并对结果进行可视化展示，并通过最大相对误差、均方根误差、平均绝对误差对模型进行评价，最终证实了本文建立的预测模型的优越性，尤其是对臭氧浓度、AQI、首要污染物的预测。

对于问题四，题目中给出了监测点 A、A1、A2、A3 在二维平面上的位置。而在天气系统中，临近地区的天气条件通常具有空间和时间上的连续性。受天气条件的影响，某地区的污染物浓度也与附近地区的污染物浓度息息相关。因此为了能够充分利用四个监测点的位置、天气条件的相关性，本文首先将四个监测点的数据集纵向合并，这是能够建立协同预报模型的基础。其次，加入了监测点间直线距离变量，又将风向变量转化为更容易度量的风力影响距离变量，这样监测点附近地区的风力、温度、湿度等天气影响因子便能更准确地作用于模型之中，使得预报的各项指标误差均有所降低，预报模型的准确度有了进一步的提升，最终证实了本文建立的协同预报模型的优越性，同时说明了协同预报模型能够提升针对监测点 A 的污染物浓度预报准确度。

7.2 模型的缺点

对于问题二，本文使用基于 EM 算法的高斯混合模型对构建的新数据集进行聚类处理，这里基于 EM 算法的 GMM 是需要预先设置簇的个数的。而本文仅是结合相关性分析的结果与气象相关文献，直接指定簇个数为 6，这样处理会对聚类结果造成一定的影响，可能无法发现气象变化对污染物浓度变化所产生的更细微、具体的影响。

对于问题三，本文建立的基于 XGBoost 的回归预测模型在对极个别污染物浓度进行预测时，会发生负优化的现象，这是由于部分污染物浓度量级较小，且由于时间安排较为紧张，没有对建立的二次预报模型进行充分的训练与测试。

7.3 未来工作

针对模型的第一个不足，我们会在未来依据 AIC 及 BIC 准则，计算出一个更准确的簇的个数，从而降低人为提供的簇树对模型造成的误差，提高聚类模型的精确度；针对模型的第二

个不足，我们会在未来的工作中，针对量级较小的污染物浓度预测建立更精确的二次预报模型，对模型进行充分的训练与测试，提高模型的鲁棒性，进一步提高模型的精确度。

8. 参考文献

[1] 杨卫芬, 夏京, 赵亚芳,等. WRF CMAQ 模式对常州市空气质量预报效果的评估[J]. 四川环境, 2019, 038(005):119-125.

[2] 王桂红.郑州市空气质量变化特征及其与气象要素的关系[J].河南科学,2021,39(09):1497-1503.

[3] 张迎春,付虹,李迪,明镇洋,刘岳军.基于CMAQ模型分析成都市O_3对气象因子的灵敏度[J].中国资源综合利用,2021,39(09):31-35.

[4] 卢亚灵,李勃,范朝阳,王建童,张鸿宇,蒋洪强.空气质量预测模拟技术演变与发展研究[J].中国环境管理,2021,13(04):84-92.

[5] Xuan G, Zhang W, Chai P. EM algorithms of Gaussian mixture model and hidden Markov model[C]//Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205). IEEE, 2001, 1: 145-148.

[6] Van der Maaten L, Hinton G. Visualizing data using t-SNE[J]. Journal of machine learning research, 2008, 9(11).

[7] Chen T, He T, Benesty M, et al. Xgboost: extreme gradient boosting[J]. R package version 0.4-2, 2015, 1(4): 1-4.

9. 附录

[问题 1：数据计算]算法程序

```
# 浓度限值表
limitTable = {
    'IAQI': [0, 50, 100, 150, 200, 300, 400, 500],
    'SO2': [0, 50, 150, 475, 800, 1600, 2100, 2620],
    'NO2': [0, 40, 80, 180, 280, 565, 750, 940],
    'PM10': [0, 50, 150, 250, 350, 420, 500, 600],
    'PM25': [0, 35, 75, 115, 150, 250, 350, 500],
    'O3': [0, 100, 160, 215, 265, 800],
    'CO': [0, 2, 4, 14, 24, 36, 48, 60]
}
# 污染物种类
pollution = set(['SO2', 'NO2', 'PM10', 'PM25', 'O3', 'CO'])

# AQI 计算
def calAQI(row):
    maxpollution = []
    maxIAQI = 0
    for key in row.keys():
        if key in pollution:
            C = row[key]
            Hi = -1
            for i in range(1, len(limitTable[key])):
                if C < limitTable[key][i]:
                    Hi = i
                    break
            if Hi < 0:
                print(key, "超出范围", C)
                return
            IAQIhi = limitTable['IAQI'][Hi]
            IAQIlo = limitTable['IAQI'][Hi-1]
            BPhi = limitTable[key][Hi]
            BPlo = limitTable[key][Hi-1]
            IAQI = math.ceil((IAQIhi - IAQIlo) / (BPhi - BPlo) * (C - BPlo) + IAQIlo)
            if IAQI > maxIAQI:
                maxIAQI = IAQI
                maxpollution = [key]
            elif IAQI > 0 and IAQI == maxIAQI:
                maxpollution.append(key)
    if maxIAQI < 50:
        maxpollution = '无首要污染物'
    return maxIAQI, maxpollution
```

[问题 2：气象条件分类]部分算法程序

```
import numpy as np
from scipy.stats import multivariate_normal


# 第 k 个单一高斯模型的密度函数
def phi(Y, mu_k, cov_k):
    norm = multivariate_normal(mean=mu_k, cov=cov_k)
    return norm.pdf(Y)


# EM 算法 step1
def getExpectation(Y, mu, cov, alpha):
    # 样本数
    N = Y.shape[0]
    # 模型数
    K = alpha.shape[0]

    gamma = np.mat(np.zeros((N, K)))

    prob = np.zeros((N, K))
    for k in range(K):
        prob[:, k] = phi(Y, mu[k], cov[k])
    prob = np.mat(prob)

    # 计算每个模型对每个样本的响应度
    for k in range(K):
        gamma[:, k] = alpha[k] * prob[:, k]
    for i in range(N):
        gamma[i, :] /= np.sum(gamma[i, :])
    return gamma


# EM 算法 step2
def maximize(Y, gamma):
    # 样本数和特征数
    N, D = Y.shape
    # 模型数
    K = gamma.shape[1]

    #初始化参数值
    mu = np.zeros((K, D))
```

```
    cov = []
    alpha = np.zeros(K)

    # 更新模型参数
    for k in range(K):
        # 第 k 个模型对所有样本的响应度之和
        Nk = np.sum(gamma[:, k])
        mu[k, :] = np.sum(np.multiply(Y, gamma[:, k]), axis=0) / Nk
        cov_k = (Y - mu[k]).T * np.multiply((Y - mu[k]), gamma[:, k]) / Nk
        cov.append(cov_k)
        alpha[k] = Nk / N
    cov = np.array(cov)
    return mu, cov, alpha


# 归一化处理
def scale_data(Y):
    for i in range(Y.shape[1]):
        max_ = Y[:, i].max()
        min_ = Y[:, i].min()
        Y[:, i] = (Y[:, i] - min_) / (max_ - min_)
    return Y


# 初始化参数
def init_params(shape, K):
    N, D = shape
    mu = np.random.rand(K, D)
    cov = np.array([np.eye(D)] * K)
    alpha = np.array([1.0 / K] * K)
    return mu, cov, alpha



# 基于 EM 的 GMM 模型
def GMM_EM(Y, K, times):
    Y = scale_data(Y)
    mu, cov, alpha = init_params(Y.shape, K)
    for i in range(times):
        gamma = getExpectation(Y, mu, cov, alpha)
        mu, cov, alpha = maximize(Y, gamma)
    return mu, cov, alpha
```

[问题 3：建立二次预报数学模型]部分算法程序

```
import pandas as pd
from sklearn.model_selection import train_test_split
import xgboost as xgb
from sklearn.metrics import mean_absolute_error
from sklearn.model_selection import GridSearchCV
import matplotlib.pyplot as plt

# 读取预处理后的数据集
excel_reader=pd.ExcelFile('A_final.xlsx')
sheet_names = excel_reader.sheet_names
data =    excel_reader.parse(sheet_name=sheet_names[0])
cols = data.shape[1]
my_cols = cols - 6

# 将模型的输入输出数据分别放入 X、Y 中
X = data.iloc[:,3:my_cols]
Y = data.iloc[:,my_cols]

# 在构建的新数据集中划分训练集、测试集,划分比例 8 : 2
train_X, test_X, train_y, test_y = train_test_split(X.values, Y.values, test_size=0.2)

# 使用 XGBoot 模型，输入训练集数据对模型进行训练
my_model = xgb.XGBRegressor(objective='reg:squarederror', learning_rate='0.1', n_estimators=100,
max_depth=8)
my_model.fit(train_X, train_y, verbose=False)

# 对特征列进行打分，观察各列对输出的影响程度，从而进行数据降维
# xgb.plot_importance(my_model,title='A1 Ferture importance')
# plt.show()
xgb.plot_importance(my_model,title='A1 Ferture importance after DR')
plt.show()

# 参数调优部分
learning_rate = [0.0001, 0.001, 0.01, 0.1, 0.2, 0.3]
n_estimators = [50, 100, 150, 200, 250, 300, 350, 400, 500, 600, 800, 1000]
max_depth = [3,4,5,6,7,8]
param_grid = dict(learning_rate=learning_rate,n_estimators=n_estimators,max_depth=max_depth)
grid_search = GridSearchCV(my_model, param_grid, scoring="neg_mean_absolute_error", n_jobs=-
1)
grid_result = grid_search.fit(train_X, train_y)
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))

means = grid_result.cv_results_['mean_test_score']
```

```
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))

# 使用模型对测试集数据进行预测
predictions = my_model.predict(test_X)
print(predictions)
print(predictions.shape)

# 使用平均绝对误差对模型进行评价
print("Mean Absolute Error : " + str(mean_absolute_error(predictions, test_y)))
```

[问题 4：建立区域协同预报模型]算法程序

```
# 计算逐日数据
new = pd.read_excel('chazhi_C.xlsx', sheet_name=0, header=0)
old = pd.read_excel('BC.xlsx', sheet_name=5, names=['date', 'location'] + pollution)
old['AQI'] = [0] * len(old)
old['prime'] = [''] * len(old)
for index, row in old.iterrows():
    if row['date'] < datetime.datetime.strptime('2020-7-23 00:00:00', '%Y-%m-%d %H:%M:%S'):
        continue
    if row['date'] == datetime.datetime.strptime('2021-7-13 00:00:00', '%Y-%m-%d %H:%M:%S'):
        break
    startHour = row['date']
    endHour = startHour + datetime.timedelta(days=1)
    hoursData = new[startHour <= new['date']]
    hoursData = hoursData[hoursData['date'] < endHour]
    hoursData.index = range(len(hoursData))
    if len(hoursData) < 8:
        print(startHour, '单天数据<8')
        old['location'][index] = 'nan'
        continue
    for key in pollution:
        if key == "O3":
            end = 8
            maxO3 = o3 = hoursData[key][0:8].sum()
            while end < len(hoursData):
                o3 += (hoursData.loc[end]['O3'] - hoursData.loc[end - 8]['O3'])
                end += 1
                if o3 > maxO3:
                    maxO3 = o3
            old[key][index] = round(maxO3 / 8)
```

```
            else:
                old[key][index] = round(hoursData[key].mean())
        old['AQI'][index], old['prime'][index] = calAQI(old.loc[index])

old.to_excel("C_day_fill.xlsx", index=False)
```