改进CS\_BP\_Adaboost算法在空气质量指数预测中的应用

**摘要：**为了更准确的对空气质量指数进行预测，针对其值受到污染物浓度以及气象条件因素影响，首先使用灰色关联分析计算所有的影响条件与AQI值的相关性，分析出气象变化值的相关性优于气象因素值。在此基础上构建BP神经网络模型，并使用Tent初始化以及反向初始化改进的布谷鸟算法去优化神经网络初始参数。最后使用多个临近监测站点的数据搭建5个改进CS-BP模型，通过Adaboost算法构建出最终的强预测器对AQI值进行预测。实验结果表明，模型的均方误差为8.52，相比于BP算法以及CS-BP算法，改进CS-BP-Adaboost算法在模型预测准确率、稳定性以及收敛速度上均取得了更加优秀的效果。

**关键字：**布谷鸟算法（CS）；BP神经网络；Adaboost算法；tent混沌初始化；灰色关联度分析；

# 0 引言

伴随着经济高速发展，人们的生活水平提高，环境恶化问题愈发严重。为了更加准确的对环境空气质量进行评判，《环境空气质量标准（2012）》中提出以SO2、CO、O3、NO2、PM2.5、PM10这六种因素作为环境质量评判的标准，而AQI（空气质量指数）则是将这六种污染物浓度进行评价比较，最后得出较为全面的综合性评判指标。通过对该指数进行预测，能够及时的了解环境空气状态，对污染防治以及提升环境空气质量方面具有重要意义。刘和旺等在对《环境空气质量标准（2012）》进行研究后得出该标准对环境的保护及城市的高质量发展起到了积极的作用[1]。现有空气质量预测方法可以分为两类—数值模拟和统计学习，伴随着人工智能的崛起，机器学习、深度学习方法已经逐渐成为热点[2]。其中神经网络方法因为其自学习、自组织、自适应和非线性表达能力强等特点被广泛应用，但其也具有对初值敏感、容易陷入局部最优等缺点。为了解决神经网络的缺陷，王芳等使用遗传算法优化神经网络的权值和阈值，在大气污染预测上取得了一定的效果[3]。Zhou K等人使用遗传算法和模拟退火算法结合优化神经网络[4]，进一步解决了遗传算法局部搜索能力弱等缺陷，在空气质量指数预测上具有较高的精度。Zhang Y等人利用PSO算法优化BP神经网络，在预测太阳总辐射上与BP模型和统计模型进行比对，证明了PSO\_BP能取得更好的效果[5]。马天成等人使用改进的PSO优化模糊神经网络对空气中PM2.5浓度进行预测，结果表明该算法相比于常规PSO\_BP取得了更为准确的预测效果[6]。本文提出了改进的CS\_BP\_Adaboost算法，将针对贵阳2018年-2019年的空气质量数据完成模型训练，实现对空气质量指数的预测。为了更好的监测环境情况，进一步验证模型的有效性，将选取一款设备完成模型的移植。在综合考虑了设备功能性、适用性以及价格等重要因素之后，本文将使用STM32F767IGT6来完成数据采集、模型移植以及预测判断功能。采集的数据在经过模型进行融合处理以后，达到减少网络传输的数据量、降低设备传输功耗以及摆脱网络依赖等目的。

# 相关算法描述

## 1.1 灰色关联度分析

灰色关联分析是对一个系统发展变化态势的定量描述和比较的方法，其基本思想是通过确定参考数据列和若干个比较数据列的几何形状相似程度来判断其联系是否紧密，反映了曲线间的关联程度。灰色关联分析的具体计算步骤如下：

1. 确定反映系统行为特征的参考数列和影响系统行为的比较数列。

在本题中，参考数列即为各污染物的，比较数列则为数据处理中得到的各项一次预报得到的气象指标以及实测的气象指标。

* 参考数列为
* 比较数列为

1. 无量纲化处理

由于各因素列中的数据因量纲不同，不便于比较，因此在进行灰色关联度分析时，需要先进行数据的无量纲化处理，使用的无量纲化公式如公式（3）所示。



1. 计算关联系数

计算关联度需先计算关联系数，关联系数计算公式如公式所示。其中，称为分辨系数。越小，分辨力越大，当时，分辨力最好，本题中取。关联系数计算公式如公式（4）所示。



记，则



1. 计算关联度

关联系数是比较数列与参考数列在各个时刻的关联程度值，因此关联系数并不唯一，而信息过于分散不便于进行整体性比较。因此需要将将各个时刻的关联系数集中为一个值，即求其平均值，计算公式如公式（5）所示。



为比较数列对参考数列的灰色关联度，越接近1，说明相关性越好。

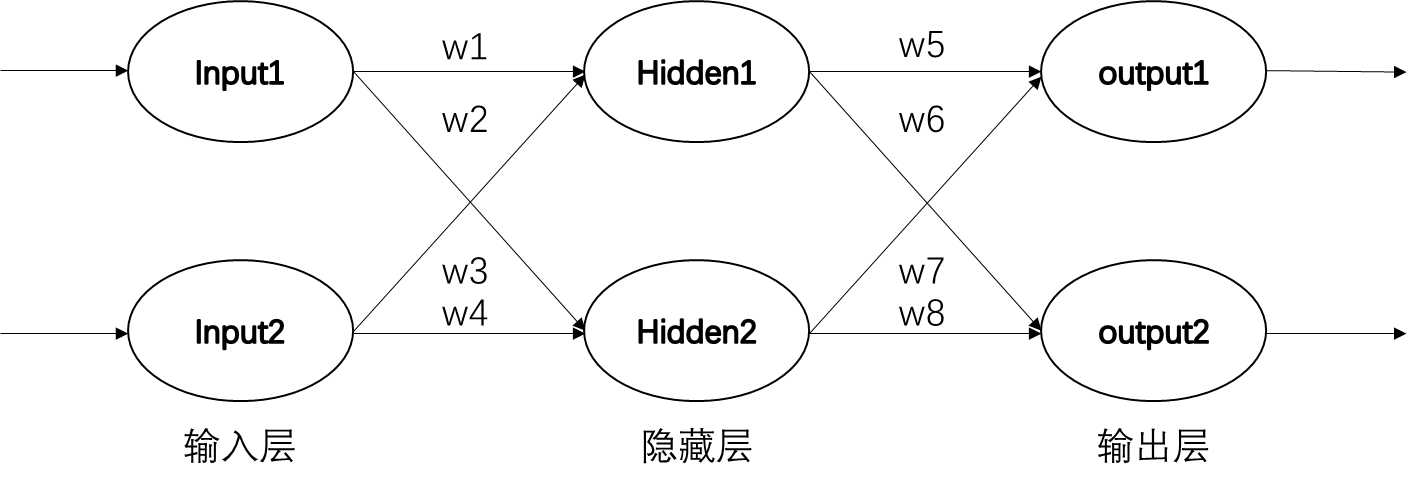
1. 关联度排序

因素间的关联程度，主要通过关联度的大小次序描述，而不仅是关联度的大小。将若干个比较数列对同一参考数列的关联度按大小顺序排列起来，便组成了关联序，反映了对于参考数列来说各比较数列的“优劣”关系。

## 1.2 BP神经网络

BP（back-propagation，反向传播）算法是一种利用输出产生的误差来调整神经元权值的过程，通过不断的反复这个过程使得模型输出逐渐逼近真实值。BP神经网络的特点就是信号的前向传播以及误差的反向传播，其具有很强的非线性映射能力。

考虑到移植设备的RAM容量和Flash容量限制，本文将采用三层结构的BP神经网络，在保证模型结构简单的同时不会使移植模型过大，并且环境空气数据具有复杂、多维、非线性等特点，三层的神经网络模型已经足够拟合任意情况的非线性问题。如下图1所示，一个简单的BP神经网络拓扑结构分为三个层次，分别是输入层、隐藏层、输出层。



**图1 BP神经网络结构**

在本文中模型的损失函数（也即是布谷鸟算法的适应度函数）选择均方误差（MSE，mean squared error）,损失函数的计算公式如下所示：



式（1）中表示第i个样本的真实值，表示第i个样本的预测值。

隐藏层节点数量的确定有经验公式确定，经验公式如下所示：

（2）

式（2）中H为隐藏层节点个数，m为输入层节点个数，n为输出层节点个数，a为1-10之间的数。

## 1.3 CS算法

布谷鸟算法是群体智能算法的一种，算法思想来源于布谷鸟特殊的生活习性。这种鸟会将生下来的蛋放到其他鸟的鸟巢去，这在算法的寻优当中就是全局最优化，该算法采用莱维飞行来模拟全局随机寻优。但鸟蛋有概率被寄养的鸟发现，发现之后该蛋会被丢弃，布谷鸟则会在蛋被丢弃之后在附加随机选择一个位置重新安置一个新蛋，这里则对应算法的局部寻优部分，采用局部随机行走算法来完成。目前已有许多研究表明布谷鸟算法在优化BP神经网络上取得了不错的效果，如NM Nawi等人将CS\_BP与ABC\_LM、ABC\_BP以及BPNN进行比较，证明了CS\_BP算法在简单性、收敛速度以及准确性上都取得了最优的效果[7]。康亚男使用了改进的CS算法优化BP神经网络，在与其它算法进行对比时也取得了更好的效果[8]。

### 1.3.1 布谷鸟算法的实现过程

在对上述算法思想的总结之后，具体算法的步骤如下：

Step1：在最初可行域内随机生成一组解（布谷鸟）。

Step2：记录这些点的适应度值，并单独记录最优解的位置及其适应度值。

Step3：通过莱维飞行更新这些解的位置，并计算新解的位置，与之前的解对比适应度值，留下更优的解。更新位置的计算公式如下：

(2)



式（2）中为更新的位置，为之前的位置，为步长，代表点乘。

(3)

式（3）中u服从N（0，）正态分布，v服从N（0，）正态分布。

(4)

(5)

式（4）中表示标准伽玛函数,。

Step4：更新后的解有一定的概率被抛弃，被抛弃的解将在附近寻找一个新的位置，没被抛弃的保存原样。这里采用局部随机行走来完成这个过程，更新公式如下：

(6)

式（6）中为更新的位置，为之前的位置，为步长系数，H为跃迁函数（x大于等于1时数值为1，否则为0），和为t时刻随机选取的两个解。

Step5：重新计算适应度值，并保存这一代最好的解和适应度。

Step6：得到新的一组解从Step3重新开始迭代，直到达到适应度要求或是迭代次数完成。

## 1.4 Adaboost算法

Adaboost（Adaptive Boosting）算法的核心思想是使用多个具有一定差异的弱分类器组合，来实现更加有效的效果。它会将前一个分类器分类错误的样本权重增大，分类正确的样本权重减少，这样当后面的分类器在同一个错误样本上再次分类错误的时候将会得到更大的误差，从而导致这次的分类器所占权重减少。后续的分类器将重复这个过程，最后得到的强分类器将具有修正弱分类器分类错误样本的能力。本文中弱分类器就是一个三层BP神经网络，强分类器则为它们的组合。

### 1.4.1 Adaboost算法的实现过程

Step1：初始化训练样本权值D，最开始所有样本权值相同，总和为1。

(7)

式（7）中N为样本总数。

Step2：训练一个弱分类器，并计算该分类器的误差

(8)

式（8）中，表示第m个弱分类器的误差，表示第m个弱分类器的第i个样本权值，I为指示函数，为第m个弱分类器。

Step3：计算该弱分类器权重。

(9)

式（9）中表示第m个弱分类器的权重。

Step4：更新样本权重分布。

(10)

(11)

式（11）中是规范化因子，目的是使中所有元素和为1。

Step5：用更新的权重分布区计算下一个弱分类器的权重。

Step6：构建最终的强分类器。

(12)

式（12）中最终强分类器G(x)为所有弱分类器的线性加权和。

# 2 空气质量判断模型构建

## 2.2 改进的CS算法优化BP神经网络

CS算法相比较其他智能搜索算法有着结构简单、搜索效果好的优势。传统的CS算法使用伪随机数生成器的方式产生初始解，使用这种方式虽然有着简单理解与实现的特点，但它生成的初始解却不够均匀，尤其是当数据维度较高的时候缺陷更加明显。本文在生成CS算法初始解的时候采取了Tent混沌初始化来产生初始解，这样能在解空间中生成更加均匀、更具遍历性、表达能力更强的初始解[9]，之后在Tent算法产生的初始解的基础上取其反向解，并计算这两组解的适应度，依据贪婪法则留下其中的最优，以此构成初始解。

Tent初始化的表达式如下所示：

(13)

式（13）中。

在CS算法进行局部优化的时候，在原本随机生成新鸟蛋的基础上再生成一个其对应的反向鸟蛋，以此增加算法的局部优化效率。并每隔一定进化周期淘汰掉20组鸟蛋当中适应度增长比例最低的5组鸟蛋。

## 2.3 改进的CS\_BP\_Adaboost算法

针对单一模型在预测时总是会存在误差（如异常天气情况下造成的输入数据较为特别的情况），本文引入了Adaboost构建多个弱分类器的方式来提升模型的稳定性与准确性。由于强分类器的性能很大程度上取决于弱分类器的丰富程度[10]，为了使弱分类器之间尽可能具备较大的差异性，将使用相差较大的Tent初值和参数来完成各个弱分类器的CS算法寻优部分，之后使用多个临近检测站点的数据分别完成CS-BP模型的训练，最后按照Adaboost算法的过程求解这些弱分类器的权值，得到最终的强分类器。弱分类器如果一致性过强则会导致生成的强分类器无法得到多大的提升。

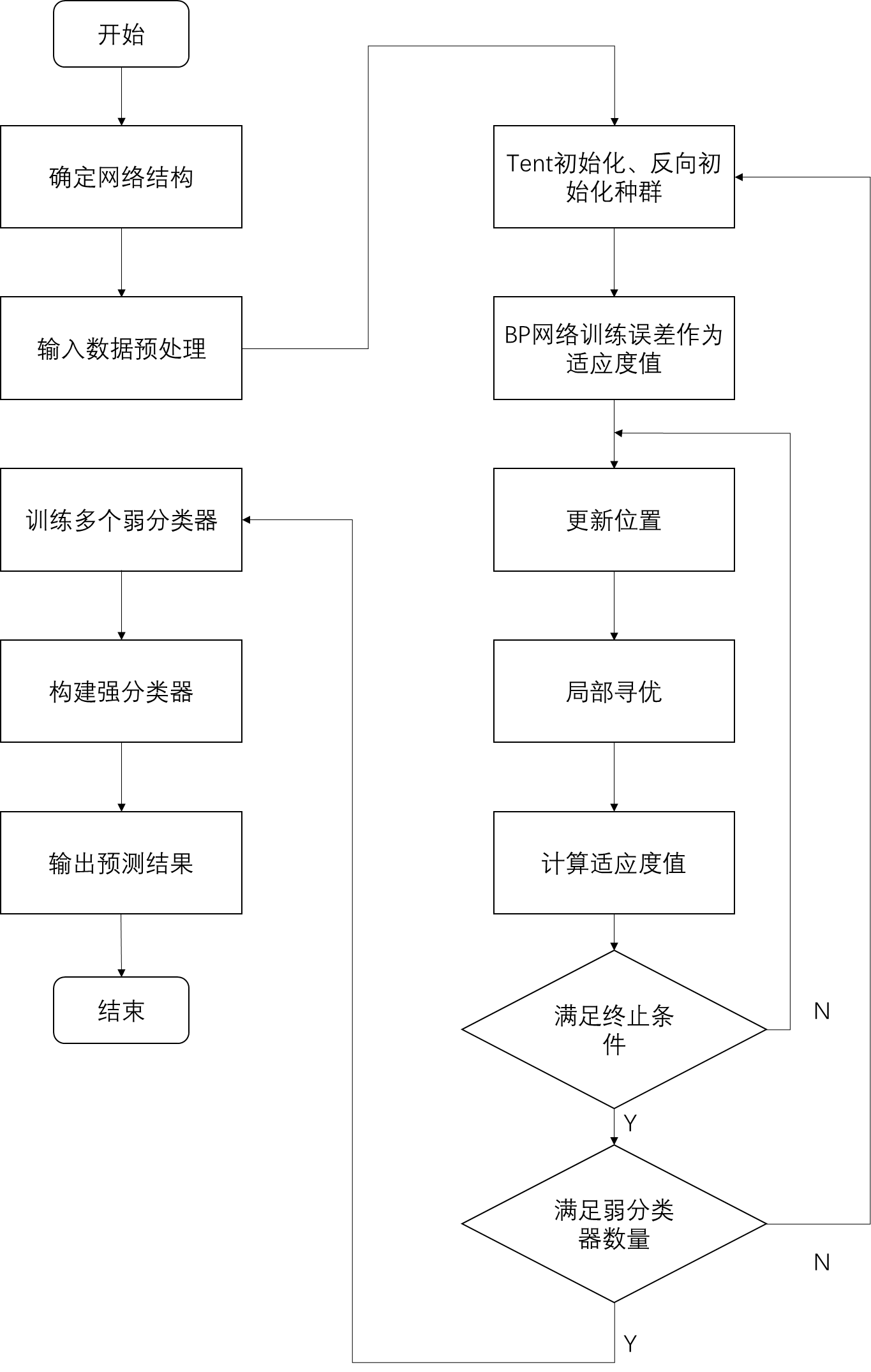
由于Adaboost算法是用来完成二分类情况的，所以对相应的部分作出调整，具体改动如下：

公式（10）中的只适用于二分类情况。本文中将先计算模型预测误差，样本预测值在可接受范围保存为0，样本预测值超出可接受范围保存为1，得到一个样本预测向量Err，最终公式为：

(15)

这样预测正确的样本权重不变，预测错误的样本权重增加。在经过规范化之后预测正确的样本权重减小，预测错误的样本权重增加。

算法总流程图如下图2所示：



**图2算法总流程图**

# 3实验结果与模型移植效果分析

## 3.1 实验环境

实验电脑操作系统为windows，软件部分采用python语言开发，采用TensorFlow2.2构建神经网络。模型移植到STM32部分采用MDK5、STM32CubeMx6.2.0以及ST提供的X-CUBE-AI库完成。

## 3.2 数据选择与预处理

数据采用青悦公开数据（其数据来源为环保部实时空气质量发布系统）提供的贵州省2020年1月到12月的城市空气质量数据以及气候数据。

空气质量指数（AQI）的影响因素为SO2、CO、O3、NO2、PM2.5、PM10这六种因素，而这六种污染物浓度又受到气象条件影响，故本文在污染物数据之外加入了5种气象数据以及前一条数据的AQI值,分别是温度（temp）、湿度（humi）、风向（w\_dir）、风力（wind）、雨量（rain），并对所有因素使用灰色关联度算法分析其与AQI值的关联度，分析结果如下图3所示：

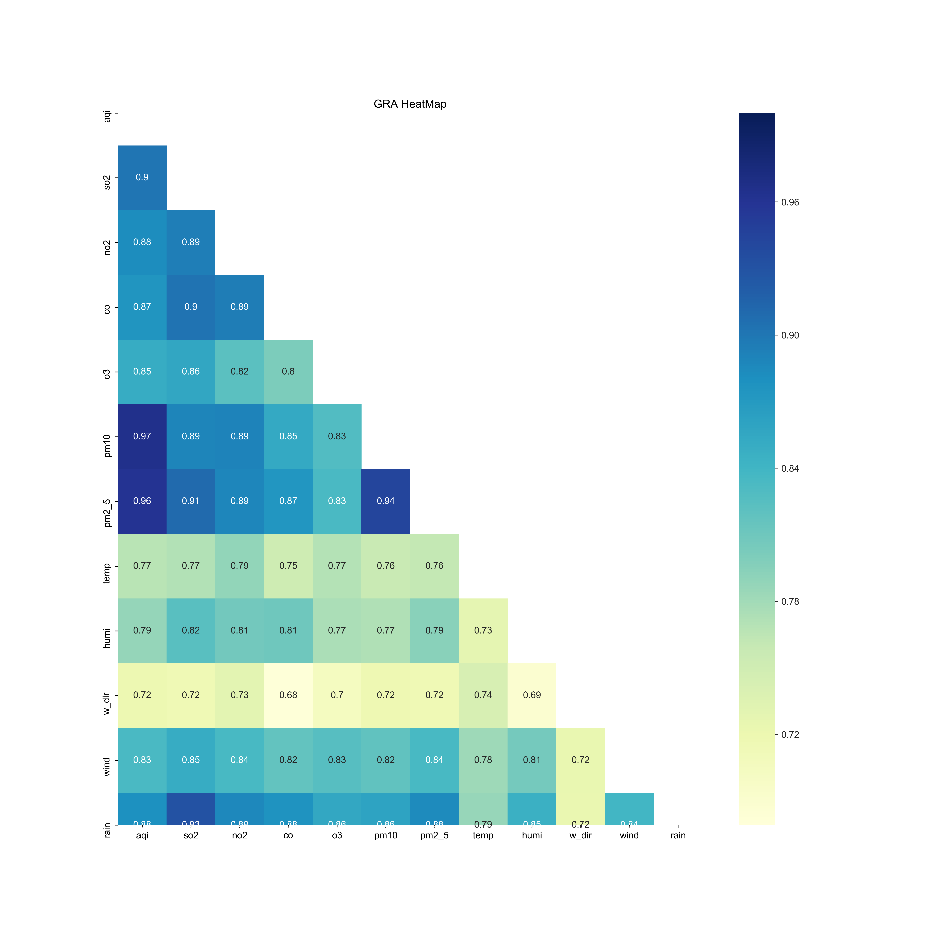


图3 污染物浓度和气象数据与AQI的关联度热力图

由图3可知，相比于六种污染物浓度和AQI的相关性来说来说，部分气象条件的关联度较低。因此，对相邻两组气象数据做差，以气象因素的变化程度取代气象因素值，并分析其与AQI值的关联度，改变后的数据为每小时温差（temp\_diff）、每小时湿度差（humi\_diff）、每小时风向差（w\_dir\_diff）、每小时风力差（wind\_diff）、每小时雨量差（rain\_diff），关联度分析结果如下图4所示：

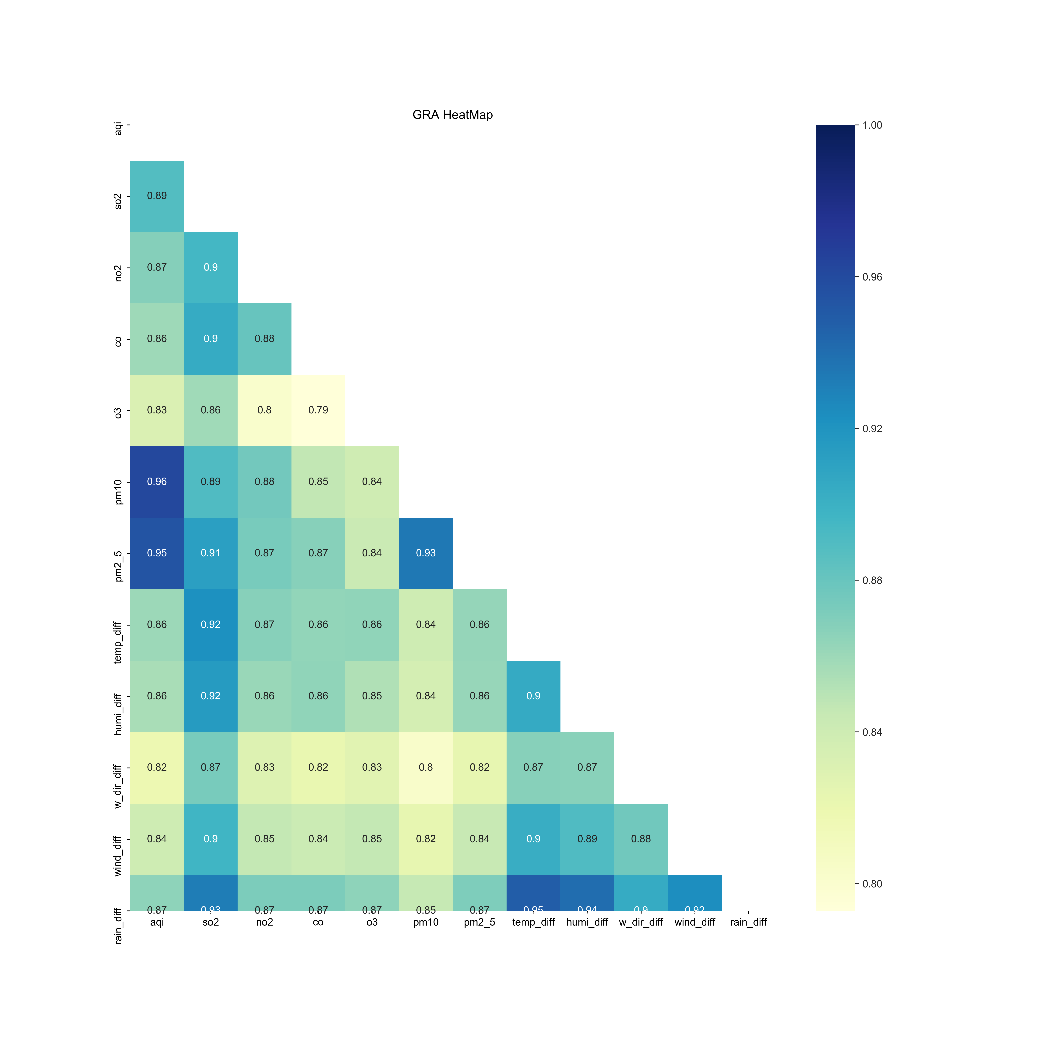


图4 污染物浓度和气象变化数据与AQI的关联度热力图

由图4可以直观的看到相比于气象因素数据，其变化情况与AQI的关联性更高。在对数据进行填充空值以及异常值处理等操作之后，最终得到的部分数据如下表1所示：

表1 实验数据

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| aqi | pre\_aqi | so2 | no2 | co | o3 | pm10 | pm2\_5 | temp\_diff | humi\_diff | W\_dir\_diff | wind\_diff | rain\_diff |
| 48 | 46 | 12 | 27 | 0.7 | 26 | 42 | 33 | 0 | 1 | 0 | 0 | 0 |
| 53 | 48 | 16 | 26 | 0.7 | 29 | 52 | 37 | 1 | -1 | 0 | 0 | 0.1 |
| 52 | 53 | 17 | 17 | 0.6 | 42 | 53 | 36 | 0 | -1 | 0 | 0 | -0.1 |
| 49 | 52 | 15 | 13 | 0.6 | 52 | 49 | 34 | 0 | 0 | 0 | 0 | 0 |
| … | … | … | … | … | … | … | … | … | … | … | … | … |
| 45 | 49 | 16 | 14 | 0.6 | 58 | 45 | 30 | 0 | 1 | 0 | 0 | 0 |

在处理完原始数据后进行零均值以及归一化处理。之后将数据转化为TensorFlow张量，在转化为Dataset，方便后续处理。

## 3.3 实验分析

本次实验的布谷鸟算法初始产生20组初始解，迭代20次，Adaboost算法采取5个弱分类器结合，适应度函数选择mse。

### 3.3.1 模型隐藏层数确定

为了对神经网络隐藏层数进行合理的选择，对经验公式（2）的所有可能情况进行了测试，由于本文的输入维度为12，输出维度为1，所以隐藏层的数量为5-14。在对相应网络结构的神经网络进行100次epoch的训练后，测试结果如下表2所示：

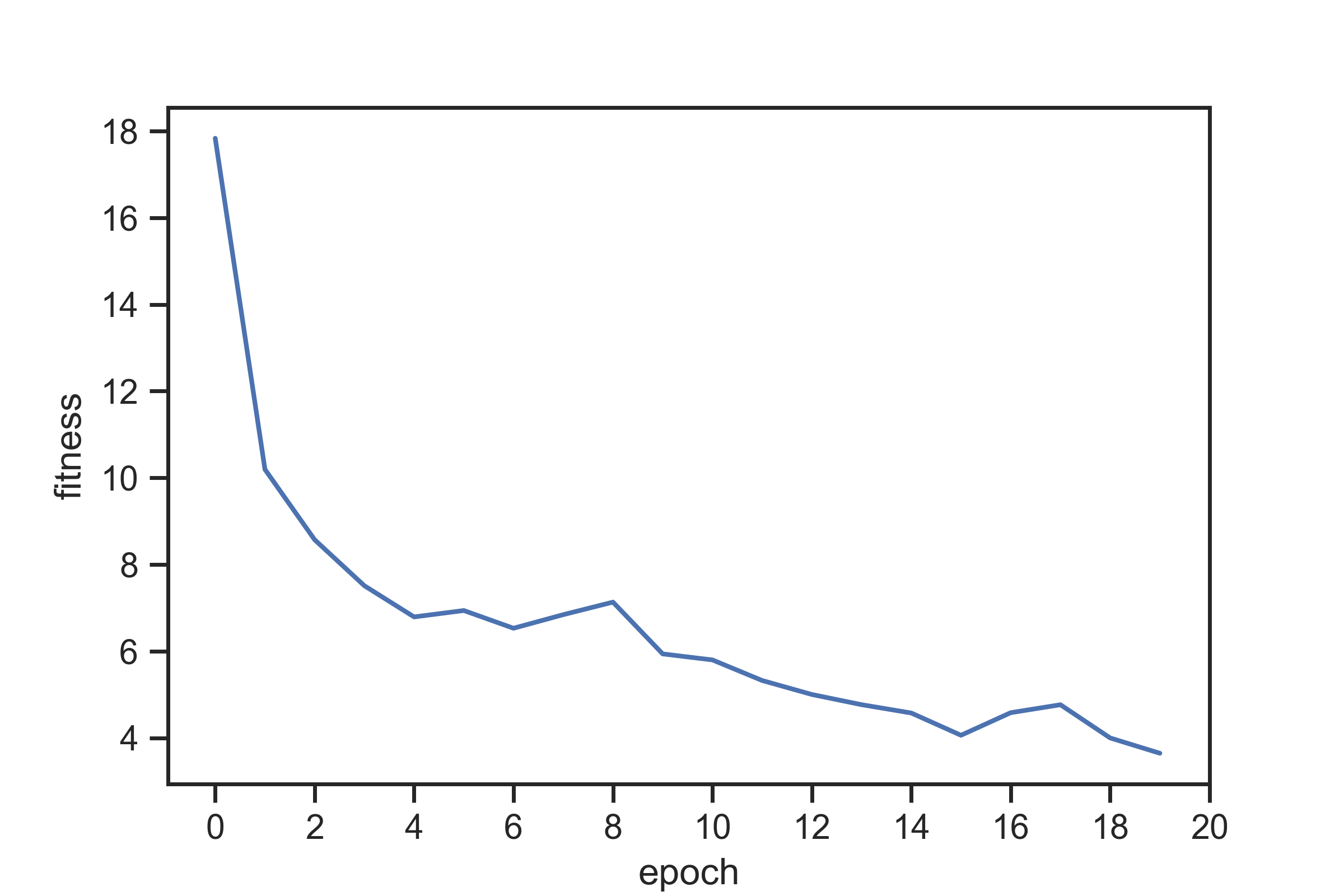
表2 各隐藏层下模型的loss

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 隐藏层数 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| 训练集损失 | 5.94 | 6.68 | 5.90 | 6.17 | 5.52 | 5.45 | 3.65 | 4.98 | 6.72 | 5.78 |

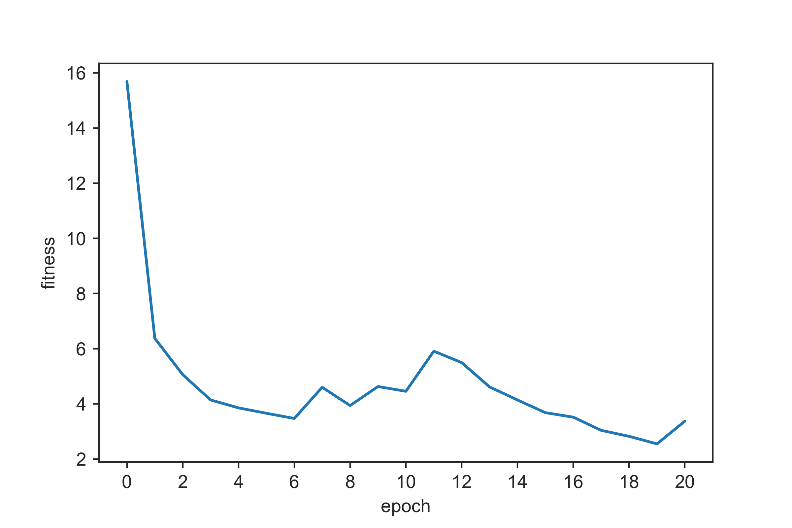
由表2可知最佳的隐藏层数为11，所以本文的神经网络结构为12-11-1。

### 3.3.2 改进CS算法效果分析

对随机初始化的CS算法以及混沌反向初始化的CS算法适应度收敛效果进行对比，下图3是随机初始化方式CS算法的适应度值优化曲线。下图4是5个经过混沌反向初始化的CS算法的适应度值优化曲线，每20个epoch为1个弱分类器结果。可以看到，在经过了Tent混沌初始化结合反向初始化后，布谷鸟算法在适应度收敛效果上取得了不错的提升。



**图5 随机初始化的CS算法适应度优化过程**



**图6 优化CS算法适应度优化效果**

### 3.3.3 AQI预测结果分析

本小节将对BP神经网络模型、CS-BP模型、改进的CS-BP模型、改进的CS-BP-Adaboost模型进行AQI预测分析。神经网络结构皆为12-11-1，训练周期皆为100个epoch。CS算法迭代次数为20，初始生成解个数为20。其中BP神经网络模型的测试集Loss为16.66389，AQI预测效果如下图7所示。CS算法得到的最优训练Loss值为3.64434，在经过100个epoch的神经网络训练后，训练集Loss值为2.4111，测试集Loss值为14.3625，AQI预测效果如下图8所示。而混沌反向初始化后的CS算法得到的最优训练Loss值为2.13257，经过神经网络训练后训练集Loss值为0.5373，测试集Loss为11.7028，AQI预测效果如下图9所示。

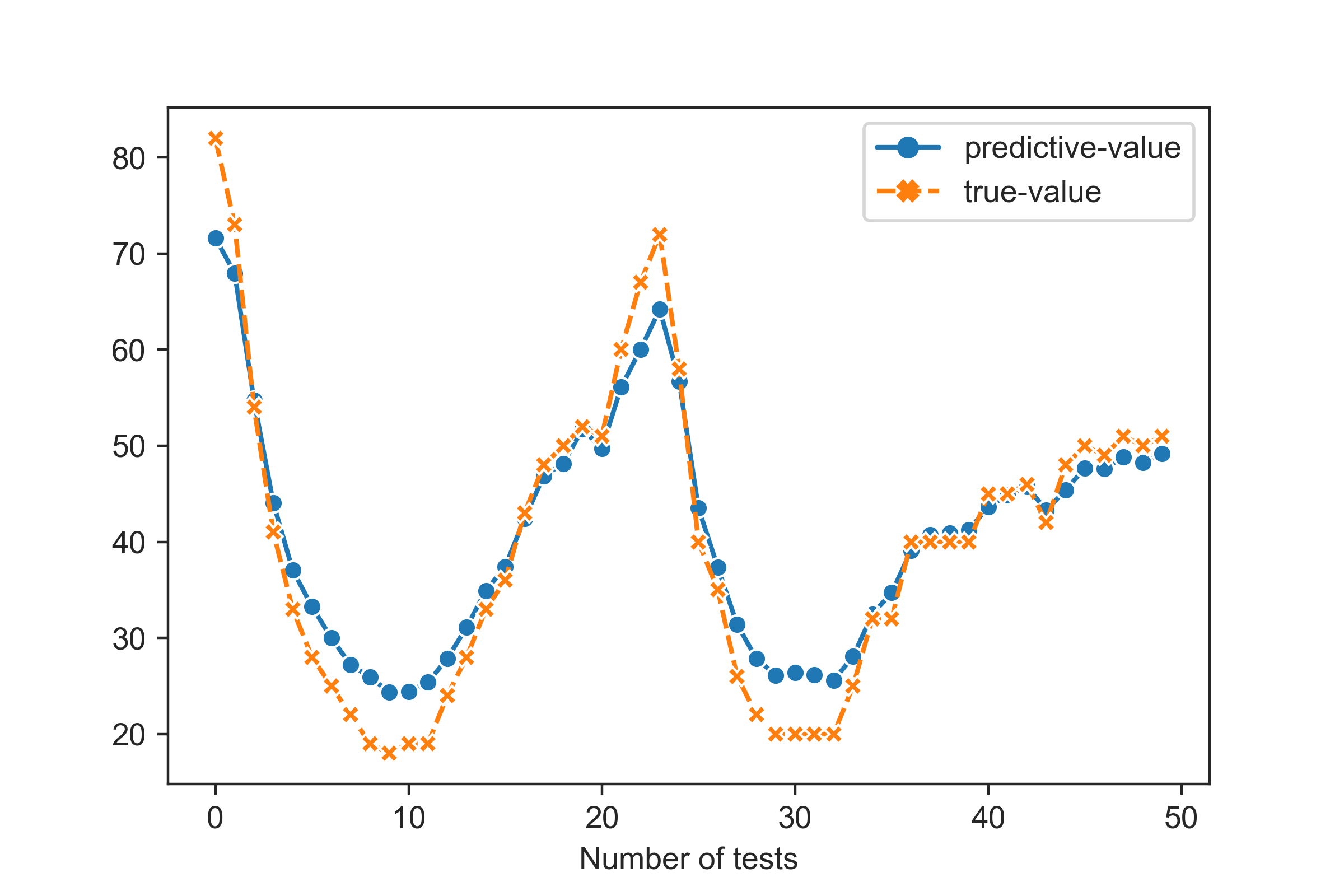


图7 BP神经网络AQI预测效果

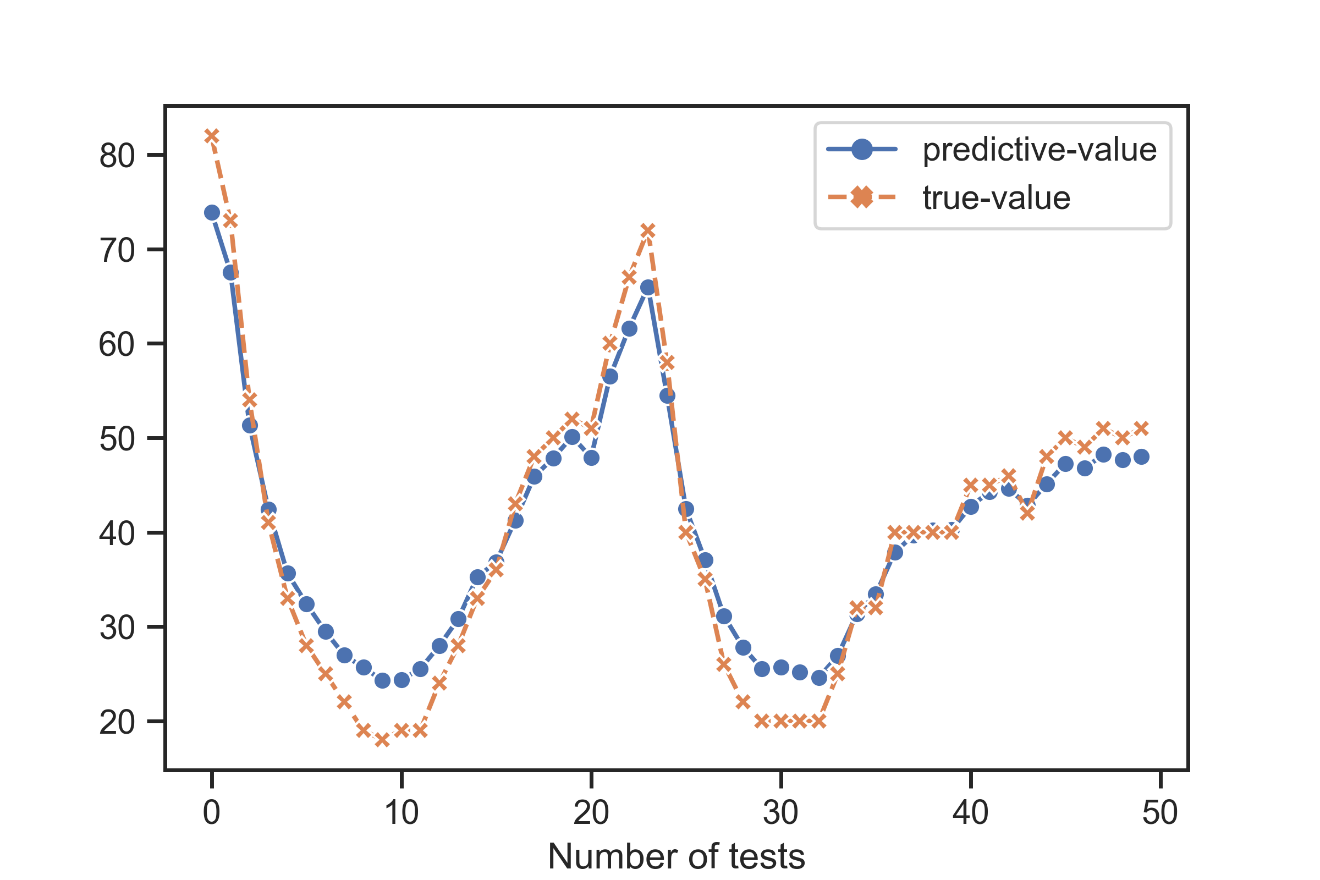


图8 CS-BP模型AQI预测效果

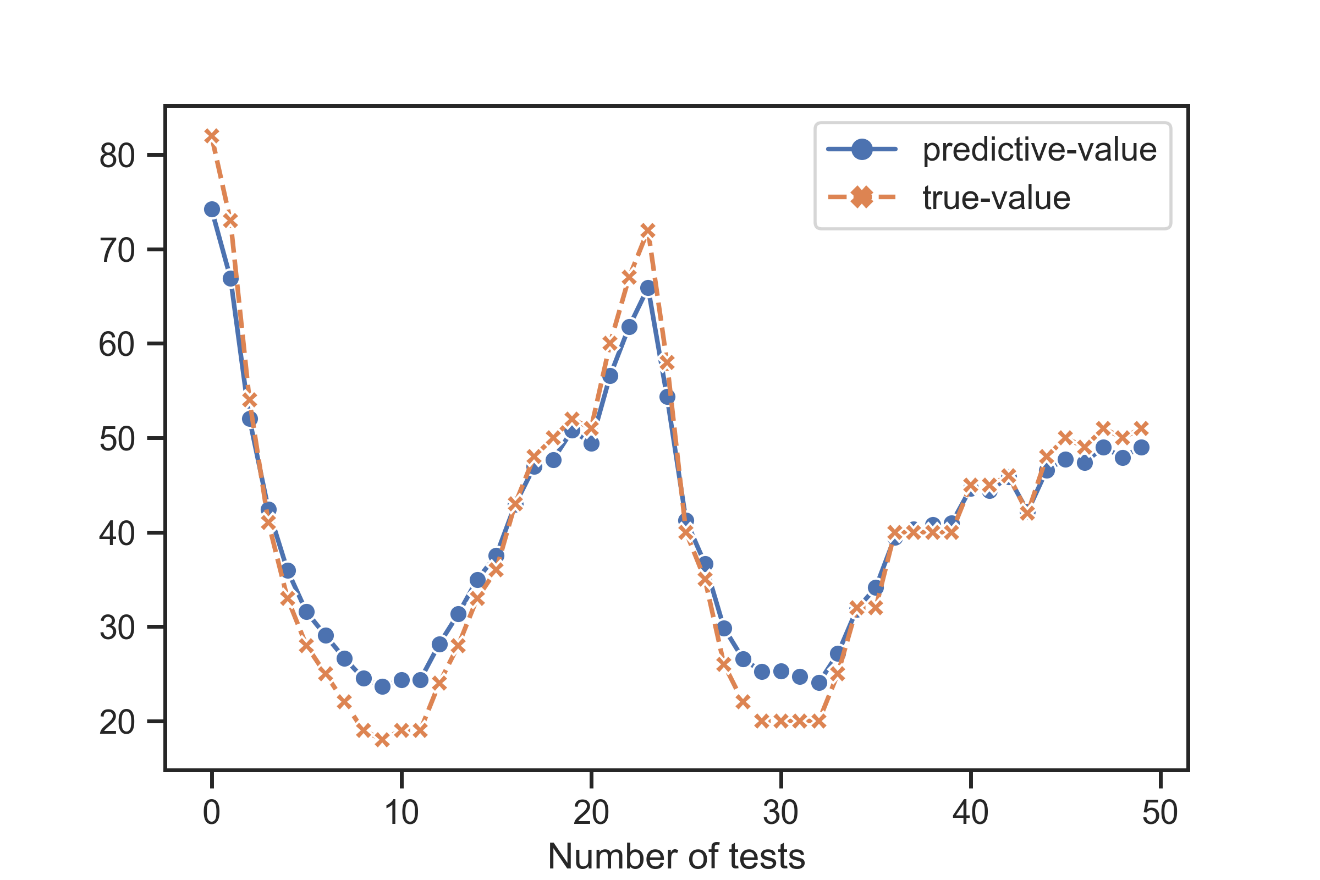


图9 改进CS-BP模型AQI预测效果

为了进一步提升模型效果，本文选取贵阳多个站点采集的污染物数据以及其对应的气候数据，搭建了5个改进的CS-BP模型。这5个模型在进行CS算法优化的过程当中，也采用了具备明显差异的Tent初始化的初值和参数，保证了模型之间的差异性。最终模型的测试集Loss为8.52，预测结果如下图10所示：

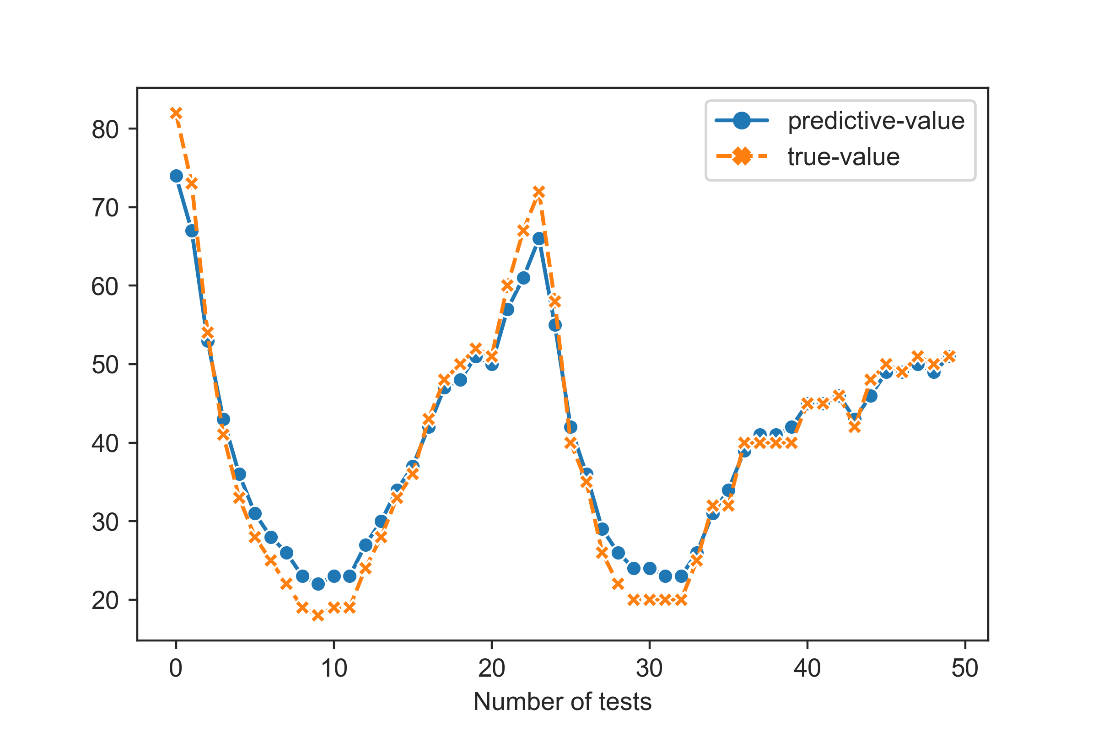


图10 改进CS-BP-Adaboost模型AQI预测效果

为了更加清晰的观察到模型优化效果，将改进CS-BP模型和改进CS-BP-Adaboost模型预测结果进行量化，将预测结果差值在2以内的设置为0类，误差在2-4的设置为1类，后续的以此类推，最终量化后的预测结果如下图11和下图12所示，量化后的对比情况如下表3所示。

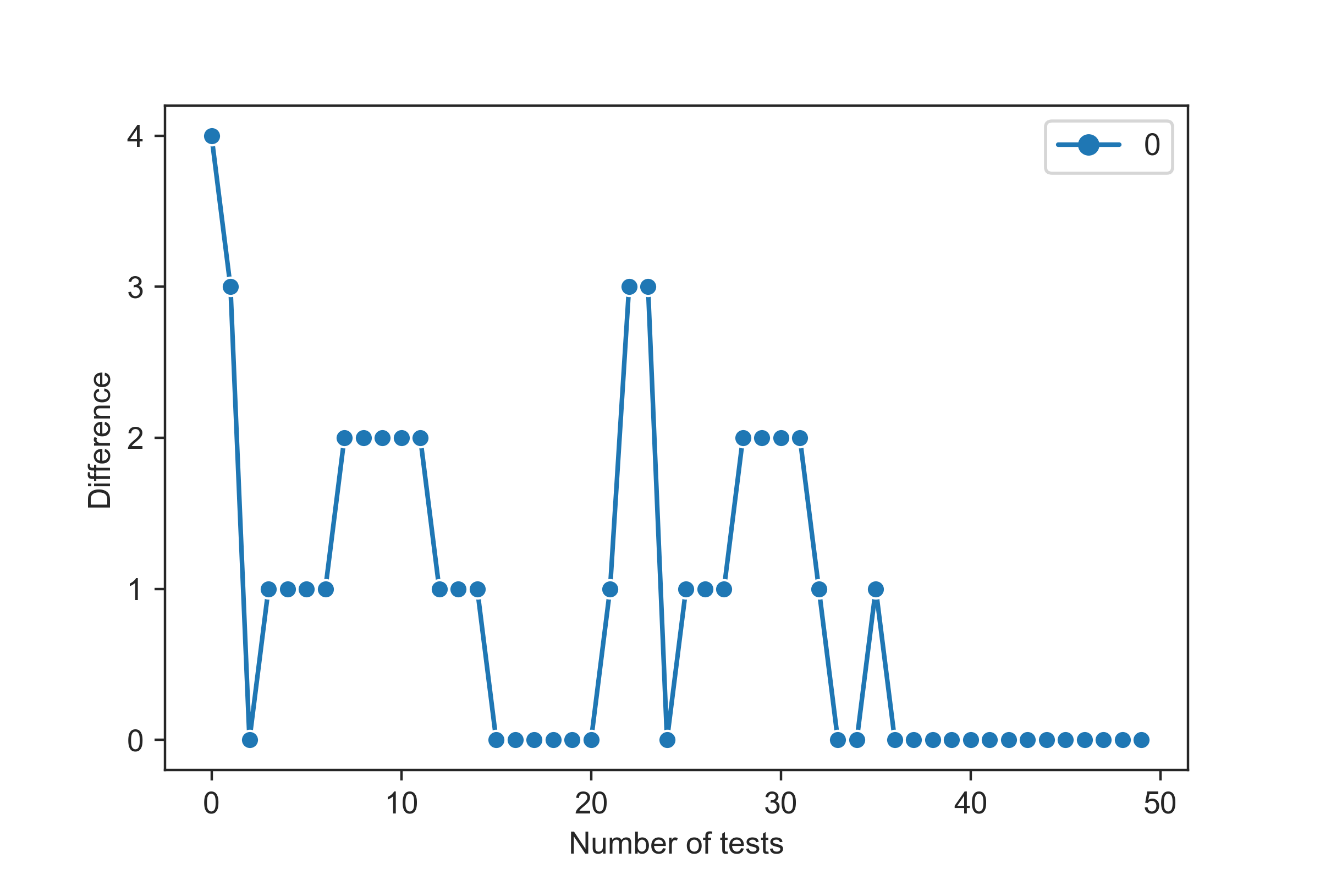


图11 改进CS-BP模型量化后AQI预测效果

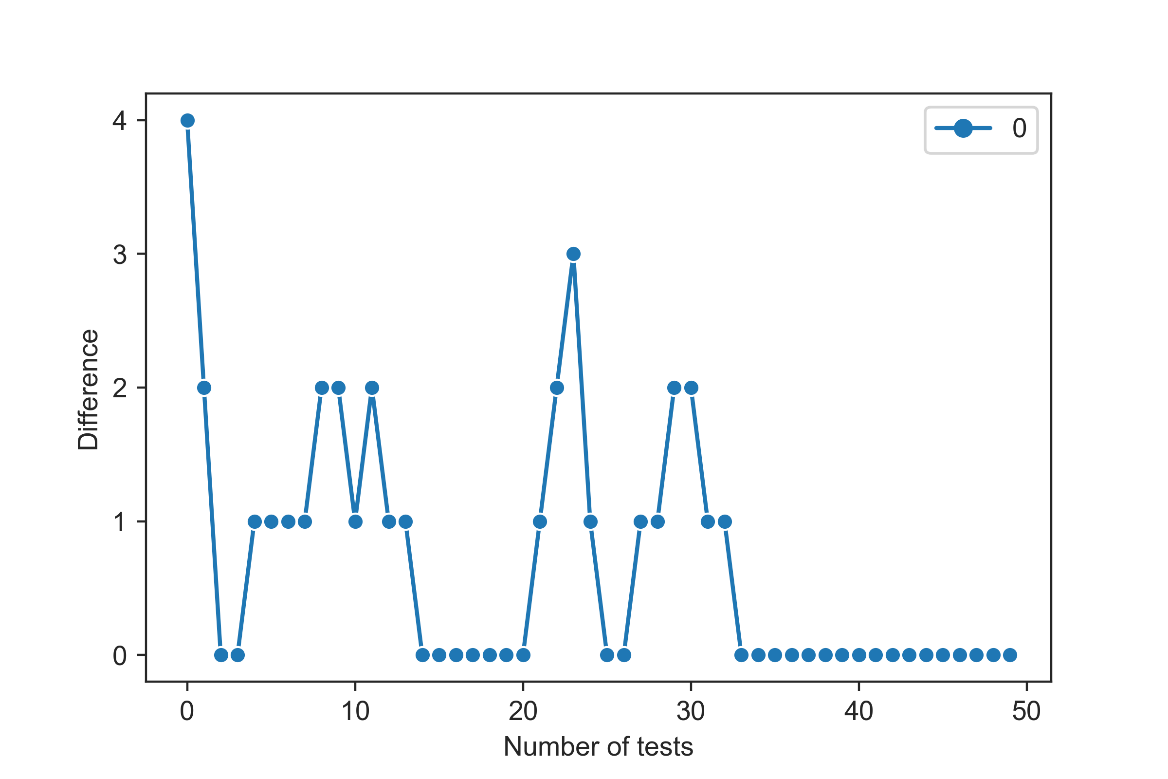


图12 改进CS-BP-Adaboost模型量化后AQI预测效果

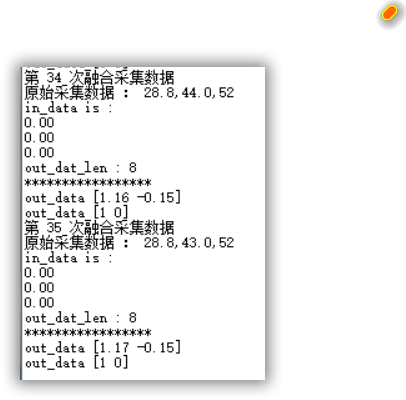
表3 量化后的模型预测效果对比

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0类 | 1类 | 2类 | 3类 | 4类 |
| 改进CS-BP | 24 | 13 | 9 | 3 | 1 |
| 改进CS-BP-Adaboost | 28 | 13 | 7 | 1 | 1 |

通过观察图11、图12以及表3可以较为明显的发现，经过Adaboost算法优化后模型性能得到了较大的提升，能够消除或是减弱单一模型的预测误差，模型预测精度较高。综上，本文提出的改进CS-BP-Adaboost模型在空气质量指数预测上效果较好。

## 3.4 模型移植效果分析

移植设备选择STM32F767IGT6，该芯片的RAM和Flash足以支撑参数量达到千万级的模型，对于本文的应用绰绰有余。芯片驱动so2、no2、co、o3、pm10、pm2.5六种传感器实时采集数据，并将采集到的数据进行预处理之后输入到移植的改进CS\_BP\_Adaboost模型当中，具体预测效果如下图所示：



从图中看到，移植效果比较稳定，预测结果与数据集进行比对后发现基本一致，说明了该方式具备一定的实用性，可以应用在一些现实问题的解决当中。

# 4结语

# 5参考文献

[1]刘和旺,刘池,郑世林.《环境空气质量标准(2012)》的实施能否助推中国企业高质量发展?[J].中国软科学,2020(10):45-55.

[2]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.

[3]王芳,程水源,李明君,范清.遗传算法优化神经网络用于大气污染预报[J].北京工业大学学报,2009,35(09):1230-1234.

[4] Zhou K , Qu Z . Application of BP neural network optimized by genetic simulated annealing algorithm to prediction of air quality index in Lanzhou[C]// 2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA). IEEE, 2017.

[5] Zhang Y , Cui N , Feng Y , et al. Comparison of BP, PSO-BP and statistical models for predicting daily global solar radiation in arid Northwest China[J]. Computers and Electronics in Agriculture, 2019, 164:104905.

[6]马天成,刘大铭,李雪洁,孙川川.基于改进型PSO的模糊神经网络PM\_(2.5)浓度预测[J].计算机工程与设计,2014,35(09):3258-3262.

[7] Nawi N M , Khan A , Rehman M Z . A New Back-Propagation Neural Network Optimized with Cuckoo Search Algorithm[C]// International Conference on Computational Science & Its Applications-volume. Springer Berlin Heidelberg, 2013.

[8]康亚男.CS优化BP神经网络的高速公路流量预测[J].公路,2017,62(05):194-198.

[9]张浩,张铁男,沈继红,李阳.Tent混沌粒子群算法及其在结构优化决策中的应用[J].控制与决策,2008(08):857-862.

[10] Schapire R E . Explaining AdaBoost[J]. empirical inference, 2013.