所在组别	2021年中国高校大数据挑战赛	参赛编号
本科生		bdc210738

智能运维中的异常预测和趋势预测

摘要

在智能运维中，通过与业务、系统、产品关联的KPI业务指标时间序列数据进行分析可以进行异常检测、异常预测和趋势预测。本文研究了运营商基站KPI业务从2021年8月28日0时至9月25日23时共29天5个基站覆盖的58个小区对应的三个核心性能指标，通过DBSCAN密度聚类算法、LOF算法和Prophet模型对三个核心关键指标进行了异常值检测、异常值预测、趋势预测。

在问题一中，题目要求对异常点检测，我们利用寻找异常点的模型进行求解，在众多模型中，我们选取LOF，Isolation Forest(孤立森林)，DBSCAN(密度聚类)三个模型分别对指标进行异常检测，通过比较三个模型的检测效果，发现针对不同的小区，不同指标的数值三个模型的效果各有所优势。考虑到现实的情况，我们不把深夜时间段的数值低谷期当做异常点，而认为这是业务正常现象，在三个模型中综合考虑，认为使DBSCAN密度聚类算法较为合适，但是在对某些小区进行异常点预测时存在误差，需要对异常值检测结果进行修正。而对时间周期的标准定义，我们也是根据实际业务情况，发现指标的时间序列数据有一定的日周期性，为此我们把时间序列数值进行分解求出周期性，而求出的周期的时间跨度来作为异常周期的时间跨度，从而来计算异常周期的个数。

在问题二中，题目要求预测未来的异常点，我们把题目一所找出的异常点进行0-1标记，0代表该时间点的指标数值不是异常，1则表示异常。我们建立神经网络模型进行求解，题目要求模型即要输入时间跨度少，又要在时效性上高，因此为了能确定模型的最佳输入时间跨度和最佳输出时间跨度，我们采用交叉验证法与网格搜索法，通过比较不同的时间跨度的模型算法的F1值来选择最佳时间跨度，最终确定最佳方案是输入时间跨度为10小时，输出时间跨度为5小时，即神经网络的输出是前10个小时的数值，输出是后5个小时的异常点标记。

在问题三中，使用prophet算法，把第一问中检测出的异常数据点和正常数据点作为数据输入，分析数据中的日季节性、假期性和整体趋势性，并建立预测模型，对未来三天三个核心指标的数据进行预测。为了验证模型的预测效果，使用交叉验证方法对模型的预测效果进行评估，计算出预测值的MSE值和MAPE值，其数值均较小，说明模型预测效果较好

关键词：密度聚类 prophet算法 异常检测 时间序列模型 神经网络

一、问题重述

1.1 问题背景

近年来，机器学习理论和研究迅猛发展，不断取得突破，促进了人工智能技术的飞跃。基于机器学习的智能运维，在这几年飞速发展，其中在电信及互联网领域，拥有得天独厚的大数据、标注和应用，更系统的数据采集和标注会帮助智能运维更快发展。众所周知，运营商拥有大量的数据中心，但是要想智能化这些数据中心，面临的挑战也将非常巨大。人工智能是电信运营商在其网络从物理网络演进到基于数据中心的虚拟化基础构架NFV/SDN时必需整合的部分，而对于数据的监控也尤为重要，实时对数据的异常点进行检测，异常点预测，趋势预测，这对发展智能运维起到重大作用。

1.2 问题重述

本题要求我们对运营商基站KPI的性能指标数据进行研究，对其中三个指标进行异常点检测，异常点预测，趋势预测，其中数据中有5个基站覆盖的58个小区29天对应的67个KPI指标。其中重点分析小区内的平均用户数，小区PDCP流量，平均激活用户数三个关键指标。

问题一：对过去的指标数据检测异常点与异常周期，首先以小区为单位，对三个关键指标根据时间序列的数据进行异常点的检测，检测各个小区的异常点所对应的时间点，并根据自己定义的时间周期计算各个小区的异常周期个数，最终汇总在同一个表格中。

问题二：对未来时间跨度中的异常点的预测，对于问题一所检测到异常点的数值，针对该异常点数值前的数据建立预测模型，预测未来的时间段会不会出现异常点，既要考虑模型的复杂度与计算成本，又要考虑模型预测未来时间周期的长度与预测异常点的准确度。

问题三：对未来趋势进行预测，根据2021年8月28日0时至9月25日23时所给的指标数据，分别预测未来三天三个关键指标的数值。

二、问题分析

2.1 问题一的分析

题目要求我们对关于时间序列的三个关键指标进行异常点的检测，对此我们想到异常点检验算法的类别，其中常见的三个分别为：1.基于统计学的方法，根据已知数据来构建一个概率分布模型，把具有低概率的数据视为异常点。2.基于聚类的方法，把偏离聚类的离散点作为异常点。3.基于专门的异常点检测算法，其代表是 One Class SVM 和孤立森林(Isolation Forest)。我们分别用 LOF，孤立森林(Isolation Forest)，DBSCAN 三个模型在对所有小区的三个指标进行异常预测，比较三个模型之间的对异常点检测的效果，由于每个小区、基站之间存在差异，所以对每个小区、基站使用以上三种方法分别单独建立三个异常检测模型，根据实际业务情况从三个模型中选择最适合每个小区的模型。再次考虑到实际业务的因素，三个关键指标有一定的日周期性，我们根据异常点的分布来选取最佳的时间周期，来确定异常点的周期个数。

2.2 问题二的分析

对于问题二，在问题一的基础上对已检测的异常点进行标记，这里我们将异常点标记为1，非异常点标记为0，这样就有一组数据$\{0,1\}$关于时间序列变化。首先对于预测异常来说，输入项是前一段时间的指标的具体数值，输出项是未来时间点可能出现的异常点。模型的选取上既要考虑输入数据的时间跨度，即为了减少模型的复杂度和计算成本应减少输入指标数据量尽可能地少，又要考虑到输出数据的时间跨度（时效性）和数据的准确度，即在保证准确度的情况下，对未来预测的时间跨度应尽可能地长。为此我们选取神经网络进行预测，输入层为数据时间跨度的具体值，输出层为未来预测的异常点标签0或1。

2.3 问题三的分析

第三题需要根据29天的历史数据建立模型，预测未来三天的数据，题目以运营商基站KPI的性能指标为研究数据，并且选择其中三个核心指标进行分析，因此使用prophet算法对三个核心指标分别建立数据预测模型进行预测。对三个指标数据的时间序列图进行观察分析发现，指标数据的变化趋势分为日季节性和整体趋势性，因此使用时间序列的分解方法将指标的时间序列数据分为日季节项S_t代表周期性变化，趋势项T_t用于拟合时间序列里的非周期项，剩余项R_t代表了所有未被模型考虑到的误差元素，并将历史数据作为输入，求出以上几项并进行求和得到预测模型并输出预测结果。

三、数据预处理

首先对附件1的数据进行预处理，提取小区内的平均用户数，小区PDCP流量和平均激活用户数三个关键指标，其中小区PDCD流量指标是附件 1 数据中的“小区PDCP层所发送的下行数据的总吞吐量比特"和"小区PDCP层所接收到的上行数据的总吞吐量比特”之和。可以观察到三个关键指标并没有缺失值，故不用对数据进行缺失值填充。为了让数据集简短精炼，我们把这三个关键指标、小区编号和时间指标汇总在同一个表格中，以便后续处理与提取。其中我们把时间指标的类型改为时间戳类型，为后续的时间序列分析和预测做准备，而且我们把数据以基站编号与小区编号为分类标准，把数据进行分类生成相应的透视表，部分数据格式如下：

基站编号	小区编号	时间	小区PDCP流量	小区内的平均用户数	平均激活用户数
1200071	26019009	2021-08-28 00:00:00	15204661328	12.5279	1.0847
		2021-08-28 01:00:00	0	0.0000	0.0000
		2021-08-28 02:00:00	0	0.0000	0.0000
		2021-08-28 03:00:00	0	0.0000	0.0000
		2021-08-28 04:00:00	0	0.0000	0.0000
...	...	...	...	...	...
1200075	26019050	2021-09-25 19:00:00	2073457064	2.3658	0.1929
		2021-09-25 20:00:00	1842170288	2.2769	0.2405
		2021-09-25 21:00:00	3422313584	2.1033	0.1726
		2021-09-25 22:00:00	804307576	1.2836	0.0725
		2021-09-25 23:00:00	428900056	0.4961	0.0308

四、模型的建立与求解

4.1 模型一的建立

问题一需要对关键指标的数据进行异常检测，从常用的异常值检测的模型中，我们选取了 LOF，孤立森林(Isolation Forest)，DBSCAN 三个模型分别对三个关键指标进行异常点的检测，并且与运营商基站的实际业务情况相结合进行分析，我们认为异常点的判别标准应是与实际业务情况不符的时间点，例如在凌晨 0:00 和 1:00 两个时间段之间，小区内的平均用户数明显降低，与白天的指标数值形成较大的数值差，如果依照统计学对异常点的判别标准，在凌晨 0:00 和 1:00 两个时间段内被划分为异常点，但是在时间情况中这是正常现象，不属于异常数值，所以我们把这些时间点归为正常点，即不作为异常点考虑，这就需要对模型进行修正。下面分别介绍 LOF，孤立森林(Isolation Forest)，DBSCAN 三个模型算法的原理和具体实现过程。

1．LOF算法

LOF 算法也叫局部离群因子检测方法，在 LOF 算法中，通过给每个数据点都分配一个依赖于邻域密度的离群因子 LOF，进而判断该数据点是否为离群点。若 LOF > 1，则该数据点为离群点；若 LOF 接近于 1，则该数据点为正常数据点。在对数据进行异常值检测判断时，LOF 算法通过比较每个点p和其邻域点的密度来判断该点是否为异常点，如果点p的密度越低，越可能被认定是异常点。至于密度，是通过点之间的距离来计算的，点之间距离越远，密度越低，距离越近，密度越高。而且，因为 LOF 对密度的是通过点的第k邻域来计算，而不是全局计算，因此得名为“局部”异常因子。在对位于不同邻域的数据进行异常检测时，不会因为数据密度分散情况不同而错误的将正常点判定为异常点。

使用LOF算法进行异常值检测，实现将所有小区的三个关键指标的时间序列数据作为数据集输入，计算出每个数据点p的LOF值，如果这个比值越接近1，说明p的其邻域点密度差不多，p可能和邻域同属一簇；如果这个比值越小于1，说明p的密度高于其邻域点密度，p为密集点；如果这个比值越大于1，说明p的密度小于其邻域点密度，p越可能是异常点，最后将数据集中的异常点检测出来。

2．孤立森林

孤立森林算法适用于连续数据的异常检测，将异常定义为"容易被孤立"，可以理解为发布稀疏且离密度高度的群体较远的点。用统计学来解释，在数据空间里面，发布稀疏的区域表示发生在此区域的概率很低，因此可以认为落在这些区域里的数据是异常的，也适用于智能运维中流量异常分析检测。

iForest 属于Non-parametric和unsupervised的方法，即不用定义数学模型也不需要有标记的训练。对于如何查找哪些点是否容易被孤立（isolated），iForest使用了一套非常高效的策略。假设我们用一个随机超平面来切割（split）数据空间（data space），切一次可以生成两个子空间（详细拿刀切蛋糕一分为二）。之后我们再继续用一个随机超平面来切割每个子空间，循环下去，直到每个子空间里面只有一个数据点为止。直观上来讲，我们可以发现那些密度很高的簇是被切分很多次才会停止切割，但是那些密度很低的点很容易很早就被划分到一个子空间里了。

iForest 算法得益于随机森林的思想，与随机森林由大量决策树组成一样，iForest森林也由大量的二叉树组成，iForest 中的树叫 isolation tree，简称 iTree，iTree 树和决策树不太一样，其构建过程也比决策树简单，是一个完全随机的过程。假设数据集有 N 条数据，构建一颗 Tree时，从 N条数据中均匀抽样（一般是无放回抽样）出 n 个样本出来，作为这棵树的训练样本。在样本中，随机选出一个特征，并在这个特征的所有值范围内（最小值和最大值之间）随机选一个值，对样本进行二叉划分，将样本中小于该值的划分到节点的左边，大于等于该值的划分到节点的右边。由此得到一个分裂条件和左右两边的数据集，然后分别在左右两边的数据集上重复上面的过程，直到数据集只有一条记录或者达到了树的限定高度。由于异常数据较小且特征值和正常数据差别很大。因此，构建 iTree的时候，异常数据离根更近，而正常数据离根更远。一颗iTree的结果往往不可信，iForest算法通过多次抽样，构建多颗二叉树。最后整合所有树的结果，并取平均深度作为最终的输出深度，由此计算数据点的异常分支。

算法步骤：该算法的核心在于如何来切割这个数据空间，由于切割是随机的，所以需要用 ensemble 的方法来得到一个收敛值（蒙特卡洛方法），即反复从头开始切，然后对每次切的结果取平均值。IForest 由 t 个 iTree（Isolation Tree）孤立树组成，每个 iTree 是一个二叉树结构，所以首先进行 iTree 树的构建，然后再进行 iForest 树的构建。

构造好iForest之后，对数据进行异常检测，需要综合每棵树的结果，使用PL记录X在每棵树的高度均值，对异常值进行检测。

3．DBSCAN 密度聚类算法

本文使用基于密度的聚类算法 DBSCAN 对小区的关键指标数据进行异常值检测，基于密度的聚类算法假设聚类结构能够通过样本分布的紧密程度确定，以数据集在空间

分布上的稠密程度为依据进行聚类，即只要一个区域中的样本密度大于某个阈值，就把它划入与之相近的簇中。密度聚类从样本密度的角度进行考察样本之间的可连接性，并由可连接样本不断扩展直到获得最终的聚类结果。这类算法可以克服 K-means、BIRCH 等只适用于凸样本集的情况。使用 DBSCAN 算法将数据进行聚类，分为正常数据和异常数据。

DBSCAN 算法流程：

DBSCAN 算法先任选数据集中的一个核心对象作为种子，创建一个簇并找出它所有的核心对象，寻找合并核心对象密度可达的对象，直到所有核心对象均被访问过为止。DBSCAN 的簇中至少包含一个核心对象：如果只有一个核心对象，则其他非核心对象都落在核心对象的 ε-邻域内；如果有多个核心对象，则任意一个核心对象的 ε-邻域内至少有一个其他核心对象，否则这两个核心对象无法密度可达；包含过少对象的簇可以被认为是噪音。

4.2 模型一的求解

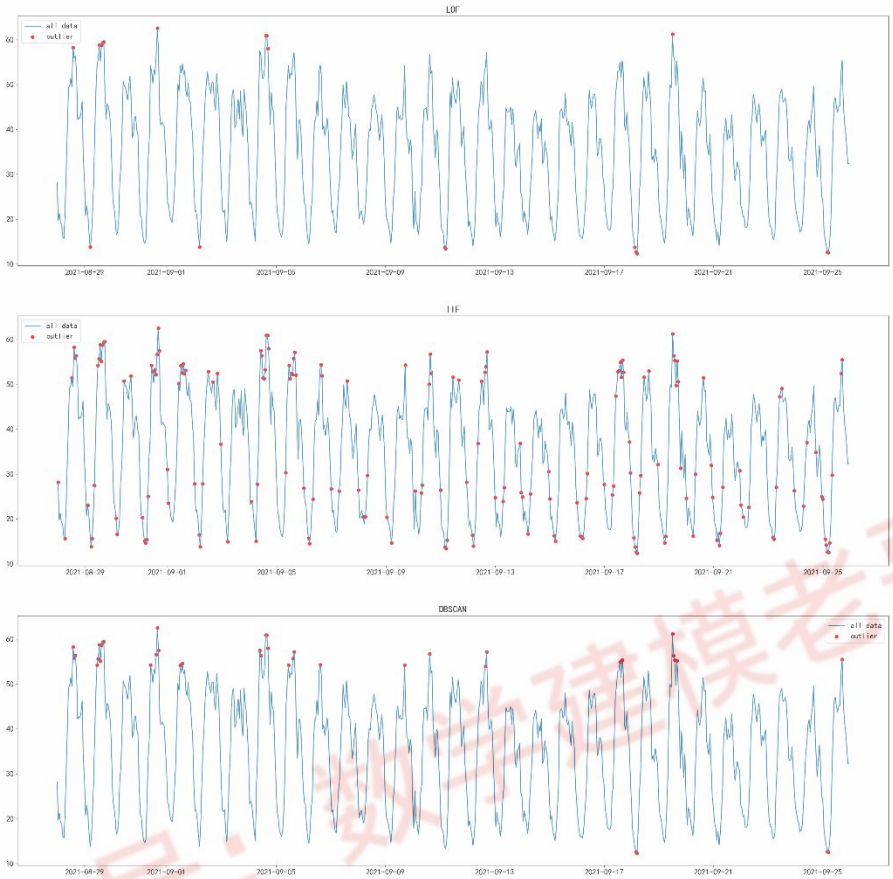
使用以上三种算法分别进行异常检测的过程如下：

使用LOF算法进行异常值检测，首先将所有小区的三个关键指标的时间序列数据作为数据集输入，计算出每个数据点p的LOF值，如果这个比值越接近1，说明p的其邻域点密度差不多，p可能和邻域同属一簇；如果这个比值越小于1，说明p的密度高于其邻域点密度，p为密集点；如果这个比值越大于1，说明p的密度小于其邻域点密度，p越可能是异常点，最后将数据集中的异常点检测出来。

使用孤立森林算法进行异常值检测，将所有小区三个关键指标的数据分别作为数据集，构造生成该数据集的iForest，然后使用生成的iForest来对异常数据进行检测：对每个itree的结果进行分析统计出数据点在每个iTree的高度，然后计算出每个数据点的平均高度，将平均高度小的数据判定为异常数据。

使用DBSCAN算法进行异常值检测，将所有小区的三个关键指标的数据集X作为输入，根据数据集中每个数据点的分布对每个数据点进行划分，得到样本密度聚类空间，将位于聚类空间外的点判断为异常点，将所有异常点输出。

得到分别使用三种算法进行异常值检测的结果如下图所示：

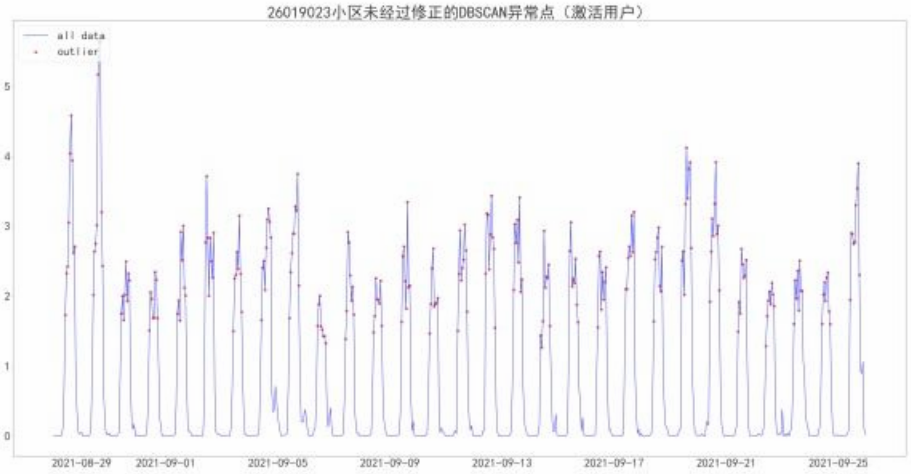


对三种算法异常值检测的结果分别进行分析：

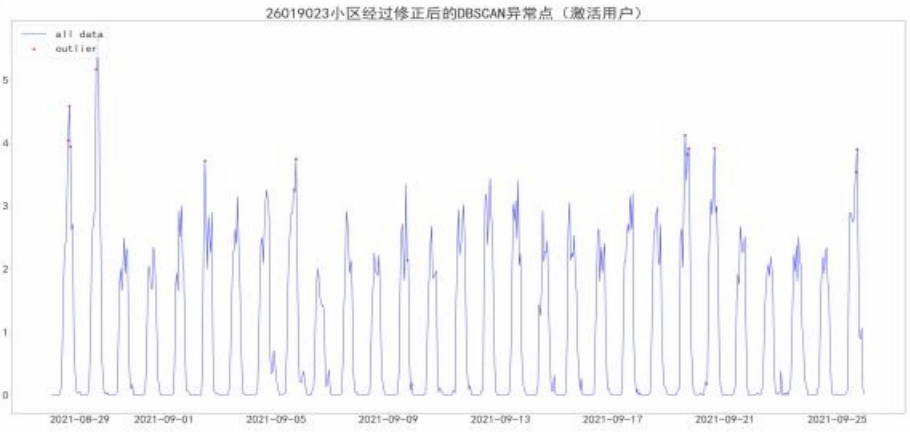
观察异常点检测的结果图像可以发现，使用LOF算法的检测结果图中明显偏离正常点的值都被标记出来，使用LOF算法可以把具有显著性的异常值都检测出来，但是检测出来的异常值过少，召回率偏低，并不能将所有的异常点都全部检测出来；孤立森林算法的检测结果图中偏高和偏低的数据点都被判断为异常值标记了出来，基本上能够把所有的异常点都检测出来，但是将很多位于正常范围的正常数据点也判定为异常点标记出来，准确率偏低；DBSCAN的检测结果图中基本上能够把异常点都检测出来，而且检测出的异常点基本上都明显偏离正常范围，准确率和召回率都较高，F1值高于前两个模型。因此我们认为DBSCAN模型在对小区的关键指标进行异常检测时效果更好，对附件的指标数据使用DBSCAN模型进行异常值检测。

4.3 模型一的修正

以26019023小区的平均激活用户数作为数据集，使用DBSCAN模型检测出该数据中的异常数值，结果如下图：



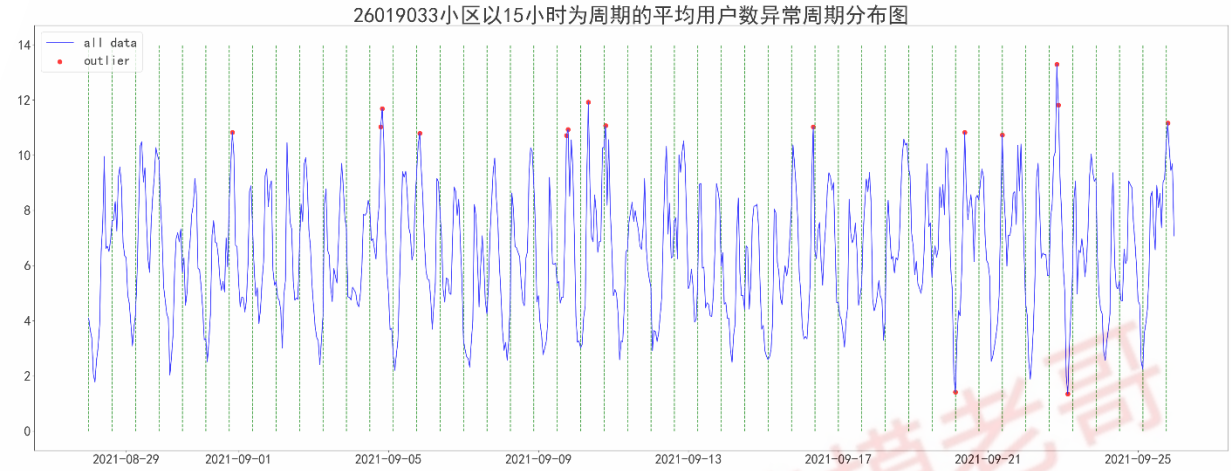
分析DBSCAN模型的异常值检测结果可以发现，一些位于凌晨一点到六点和深夜十一点的在数据分布上具有异常性的数据点被标记为异常值。但是考虑到实际业务情况，平均激活用户数表示该基站覆盖的小区在一个时间段内注册过无线网络的平均人数，而在深夜和凌晨时间段，使用无线网络的人数明显减少，平均激活用户数随之减少；在凌晨到白天和白天到深夜的时间段内，使用无线网络的人数会有明显的变化，平均激活用户数也会随之有明显的变化，这属于业务的正常现象，但是在DBSCAN模型中，这些数据的变化情况被判定为异常变化，这些时间段内的一些数据点也被判定为异常点。所以本文在DBSCAN模型的基础上根据运营商基站的实际业务情况对异常点的检测结果进行修正，将符合实际业务情况的数据点标记为正常点，进行修正后结果如下：

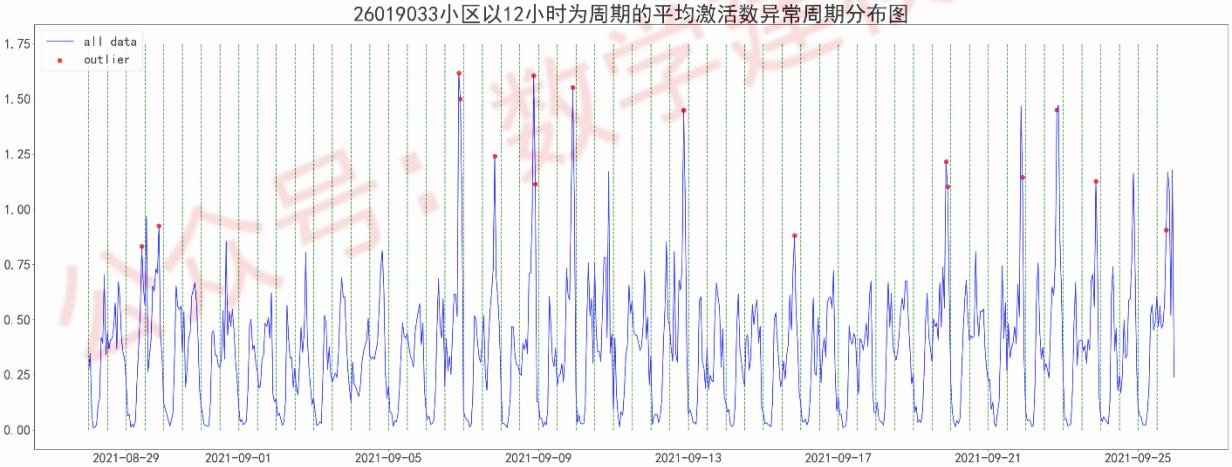


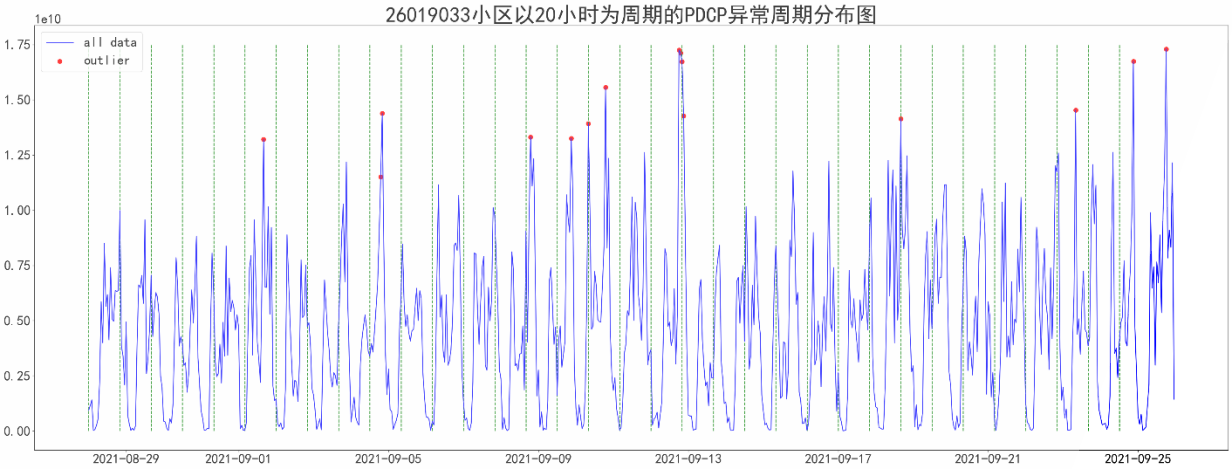
观察修正后的数据结果图像后可以发现之前被标记为异常但是实际上符合业务情况的点没有再被标记为异常点，经过修正后的DBSCAN模型能够正确检测出实际业务情况中的异常点。

异常周期的检测

对异常周期的检测需要选定异常周期的时间范围段，将有多个异常值的时间段划分为异常周期。对附件中三个关键指标进行分析，发现三个指标的数据分布具有周期性，因此对数据进行统计计算出三个指标的周期，其中小区内的平均用户数的周期为15小时，小区DPCP流量的周期为20小时，平均激活用户数的周期为15小时。以编号为26019033的小区的三个指标的数据为例，画出其时间序列图并标出其中的异常点，按照其周期将其数据点进行划分，结果如下图所示：







图中被标记为红色的为使用模型检测出的异常点，绿色的竖线为指标数据的周期划分线，将每个指标数据中一个周期内含有两个及以上异常值的周期数进行统计，记为该小区指标的异常周期个数。对其他小区使用相同方法统计出所有小区三个指标数据的异常周期数并进行汇总，得到所有小区的异常情况。

表:异常检测汇总表

	时间周期选择标准	异常孤立点的个数	异常周期个数
小区内的平均用户数	15小时	1856	126
小区PDCP流量	20小时	1516	67
平均激活用户数	12小时	1208	139

4.4 模型二的建立

本题要求我们根据过去的数值对未来异常点进行预测，针对这个问题，看似是一种分类判别模型，但是本题是对多个未来时间序列的同时进行检测，若用传统的分类判别只对一个时间点进行预测，大大降低了模型的时效性。考虑到既能同时预测多个时间点的异常点，又能进行判别分类，为此我们选取多输入多输出的神经网络模型进行预测，这样可以增加模型的时效性，而且更改输入输出的时间跨度非常方便。

我们先对神经网络进行简单了解，神经网络是由具有适应性的简单单元组成的广泛并行互连的网络，神经网络中最基本的成分是神经元模型，神经元接收到来自n个其他神经元传递过来的输入信号，这些输入信号通过带权重的连接进行传递 ，神经元接收到的总输入值将与神经元的阀值进行比较，然后通过“激活函数”处理以产生神经元的输出。把许多个神经元按一定的层次结构连接起来，就得到了神经网络。具有多层神经元的神经网结构被称为“多层前馈神经网络”。神经网络的学习过程，就是根据训练数据来调整神经元中间的“连接权”以及每个功能神经元的阈值。

4.5 模型二的求解

首先要对三个关键指标的数据进行标准化，标准化方程如下：

$$x' = \frac{x - x_{mean}}{x_{max}}$$

其中，x'为标准化后的数值，x为原数值，x_{mean}为该变量的平均值，x_{max}为该变量的最大值。本题我们以三个关键指标中的小区内的平均用户为例，为了确定神经网络的输入的时间跨度和输出的时间跨度，我们利用交叉验证法和网格搜索法进行选取，我们列举了几个时间跨度的例子(见下表)，通过混淆矩阵分别进行 F1 值的计算。计算公式如下：

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2Precision * Recall}{Precision + Recall}$$

其中混淆矩阵表示如下：

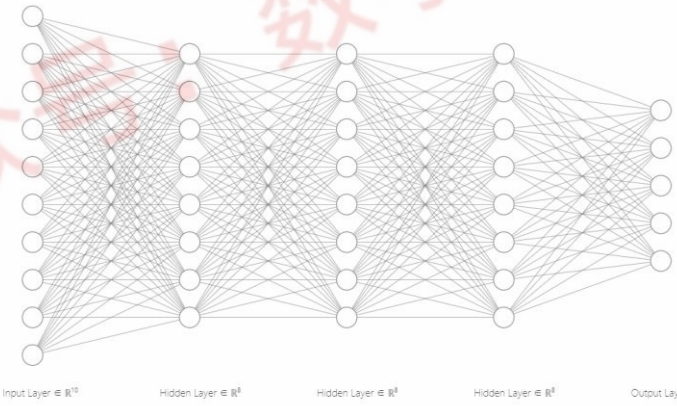
	预测值	
真实值	*Positive*	*Negative*
Positive	*TP*	*FN*
Negative	*FP*	*TN*

表1. 混淆矩阵

输入数据时间跨度	输出数据时间跨度	F1值
48小时	24小时	0.58
24小时	12小时	0.65
12小时	6小时	0.86
10小时	5小时	0.95
8小时	4小时	0.96

表2. 神经网络不同输入和输出的时间跨度与F1值

根据上表结果分析可得：当输入数据的时间跨度为 10 小时和 8 小时，输出数据时间度为 5 小时和 4 小时的 F1 值较高，由于 F1 值的差距较小，为了增加神经网络的时效性，最终我们选取输入数据的时间跨度为 10 小时，输出数据的时间跨度为 5 小时的方案，故本文选用的神经网络结构输入层的神经元为 10 个，输出层的神经元为 5 个，结构图如下：



该神经网络为四层数据网络，隐藏层为三层，隐藏层节点为8个。

BP 算法的工作流程如下：对每个训练样例，BP 算法执行一下操作:先将输入示例提供给输入层神经元，然后逐层将信号前传，直到产生输出层的结果;然后计算输出层的误差，再将误差逆向传播至隐藏层神经元，最后根据隐藏层神经元的误差来更新连接权和阈值进行调整。该迭代过程循环进行，直到达到某些停止条件为止，例如训练误差已达到一个很小的值。

算法流程

输入：训练集 $\boldsymbol{D}=(\boldsymbol{x_k},\boldsymbol{y_k})_{k=1}^{m}$;学习率
过程：
1：在（0,1）范围内设计初始化网络中所有连接权和阈值
2：repeat
3： for all $(\boldsymbol{x_k},\boldsymbol{y_k})\in \boldsymbol{D}$ do
4： 根据当前参数和隐藏层输入函数计算当前样本的输出$\boldsymbol{y_k}$;
5： 计算出输出层神经元的梯度项$\boldsymbol{g_j}$;
6： 计算出隐藏层神经元的梯度项$\boldsymbol{e_h}$;
7： 根据公式更新连接权$\boldsymbol{w_{hj}},\boldsymbol{v_{ih}}$和阈值$\boldsymbol{\theta_j},\boldsymbol{\gamma_h}$
8： end for
9： until 达到停止条件
输出：连接权与阈值确定的多层前馈神经网络

4.6 模型三的建立

问题三需要将三个指标 29 天内的历史数据进作为输入，分析三个指标的变化趋势，建立预测模型，预测未来三天三个指标的数据取值。对三个指标 29 天的时间序列数据进行分析，画出其时间序列图像，观察图像可以发现：29 天内每个指标数据每天的变化相似，具有周期性，于是将三个指标的数据变化趋势分解为日季节性和整体趋势进行分析。本文使用 Prophet 模型来分析三个指标数据的变化趋势，并建立预测模型对指标数据进行预测。

算法实现：
在进行对时间序列数据的分析时，使用时间序列的分解方法，通常将时间序列数据y_t分成几个部分，分别是季节项S_t，趋势项T_t，剩余R_t。也就是说对于所有的$t\geq 0$，都有

$$y_t = S_t + T_t + R_t$$

这里我们从三个指标中，选小区内的平均用户历史数据为例，在数据分析中，需要考虑到日季节项、趋势项和剩余项，所以在 prophet 算法中，本文考虑了以上三项，也就是：

$$y_t = g(t) + s(t) + \epsilon_t$$

其中$g(t)$表示趋势项，它表示小区内的平均用户数时间序列数据在非周期上面的变化趋势，即长期的变化趋势;；$s(t)$它表示日季节项，或者叫日周期项，它表示时间序列数据在一个周期内即一天内的变化趋势，$\epsilon(t)$表示误差项或者称为剩余项。通过对上述几项数据进行拟合，最后将其累加起来就得到了时间序列数据的预测值。

趋势项模型$\boldsymbol{g(t)}$：

在 Prophet 算法中，对趋势项可以使用基于逻辑回归函数的方法进行拟合，逻辑回归函数的形式为：

$$\sigma(x)=\frac{1}{1+e^{-x}}$$

它的导数是$\sigma'(x)=\sigma(x)\times(1-\sigma(x))$,并且$\lim_{x\to+\infty}\sigma(x)=1,\lim_{x\to-\infty}\sigma(x)=0$,经过变形，最终可变为：

$$f(x)=\frac{C}{1+e^{-k(x-m)}}$$

其中C,k,m为常数，但考虑到时间序列数据中，C,k,m会随时间的迁移而变化，所以令$C=C(t),k=k(t),m=m(t)$,除此之外，考虑到数据中存在潜在的周期曲线会发生变化，因此在 Prophet 算法中，为了增加模型的准确性，需加入变点的位置，为此我们把问题一中的标记的异常点作为变点的位置。下面利用数学理论来说明变点数据的用处，假设我们给定的变点有S个，且时间位置设为s_j，$1\le j\le S$,那么在这些时间戳上，我们就需要给出增长率的变化。可以假设有一个向量：$\delta\in R^S$，其中δ_j表示在时间戳s_j上的增长率的变化量。如果一开始的增长率我们使用K来代替的话，那么在时间戳t上的增长率就是$K+\sum_{j:t>s_j}\delta_j$，通过一个指示函数$a(t)\in\{0,1\}^S$，即

$$a_j(t)=\begin{cases}1 & ,if \quad t\ge s_j,\\ 0 & ,otherwise.\end{cases}$$

那么在时间戳t上面的增长率就是$k+a^T\delta$。一旦变化量k确定了，另外一个参数m也要随之确定，在这里学要把线段的边界处理好，因此通过数学计算可得：

$$\gamma_j=(s_j-m-\sum_{\iota<j}\gamma_\iota)\cdot(1-\frac{k+\sum_{\iota<j}\delta_\iota}{k+\sum_{\iota<j}\delta_\iota})$$

所以分段的逻辑回归增长模型就是：

$$g(t)=\frac{C(t)}{1+e^{-(k+a(t)^t\delta)\cdot(t-(m+a(t)^T\gamma))}}$$

其中$a(t)=(a_1(t),\cdots,a_s(t))^T,\delta=(\delta_1,\cdots,\delta_s)^T,\gamma=(\gamma_1,\cdots,\gamma_s)^T.$

所以只需提前设置好$C(t)$的取值即可，便可确定逻辑回归增长模型。

日季节性趋势

Prophet可以通过在ds列中传递一个带有时间戳的dataframe来对时间序列进行子日（Sub-daily）观测。当使用子日数据时，日季节性将自动匹配。用时间间隔为一小时的小区内的平均用户数数据对Prophet进行数据匹配，得到日季节性的趋势变化图。

Prophet模型利用正弦余弦函数来表示区间内的周期性，使用其傅里叶级数来模拟时间序列的周期性，假设P表示时间序列的周期，则其傅里叶级数为

$$s(t)=\sum_{n=1}^{N}(a_n\cos(\frac{2\pi nt}{P})+b_n\sin(\frac{2\pi nt}{P}))$$

这里令

$$\beta=(a\_1,b\_1,\cdots,a\_N,b\_N)\text{^}T,X(t)=[\cos\left(\frac{2\pi nt}{P}\right)+\sin\left(\frac{2\pi nt}{P}\right]]$$

则时间序列的季节项就是：

$$s(t)=X(t)\beta$$

其中β的初始化是β服从正态分布$N(0,\sigma^2)$,σ是来决定季节的效应的强弱的，σ的值越大，表示季节的效应越明显。附件一的数据有明显的日周期性，所以我们在拟合时，需要把季节项的趋势设置为日季节项周期。

模型拟合(Model Fitting)

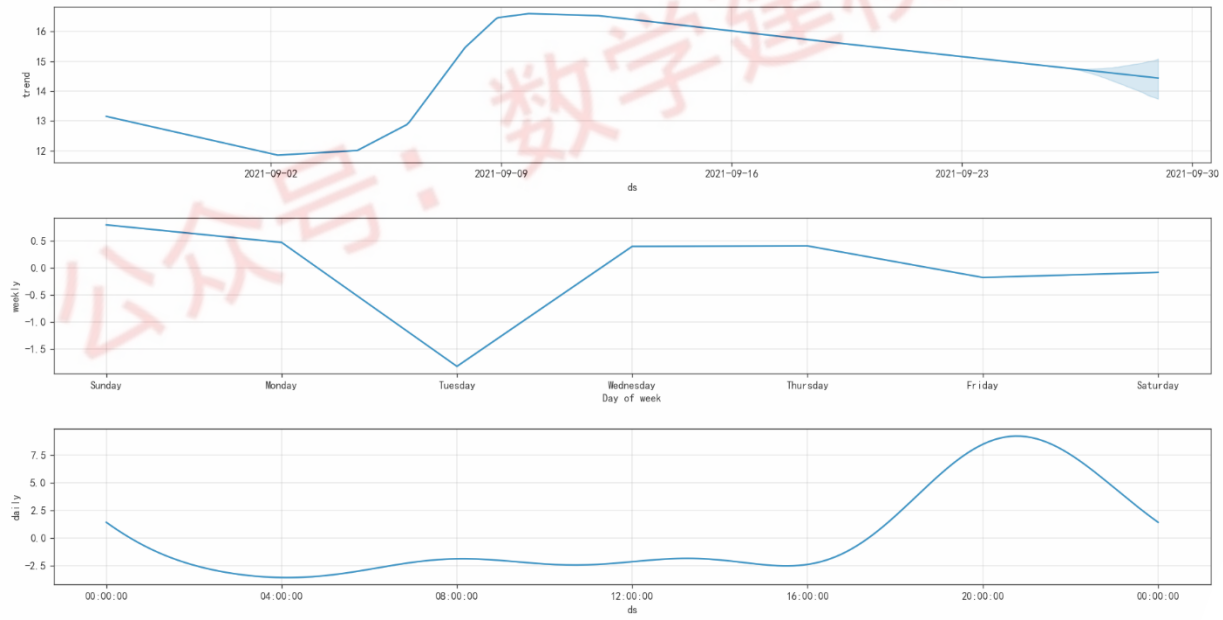
综上推理，我们的时间序列已经可以通过趋势项，季节项来构建了，公式如下：

$$y(t) = g(t) + s(t) + \epsilon$$

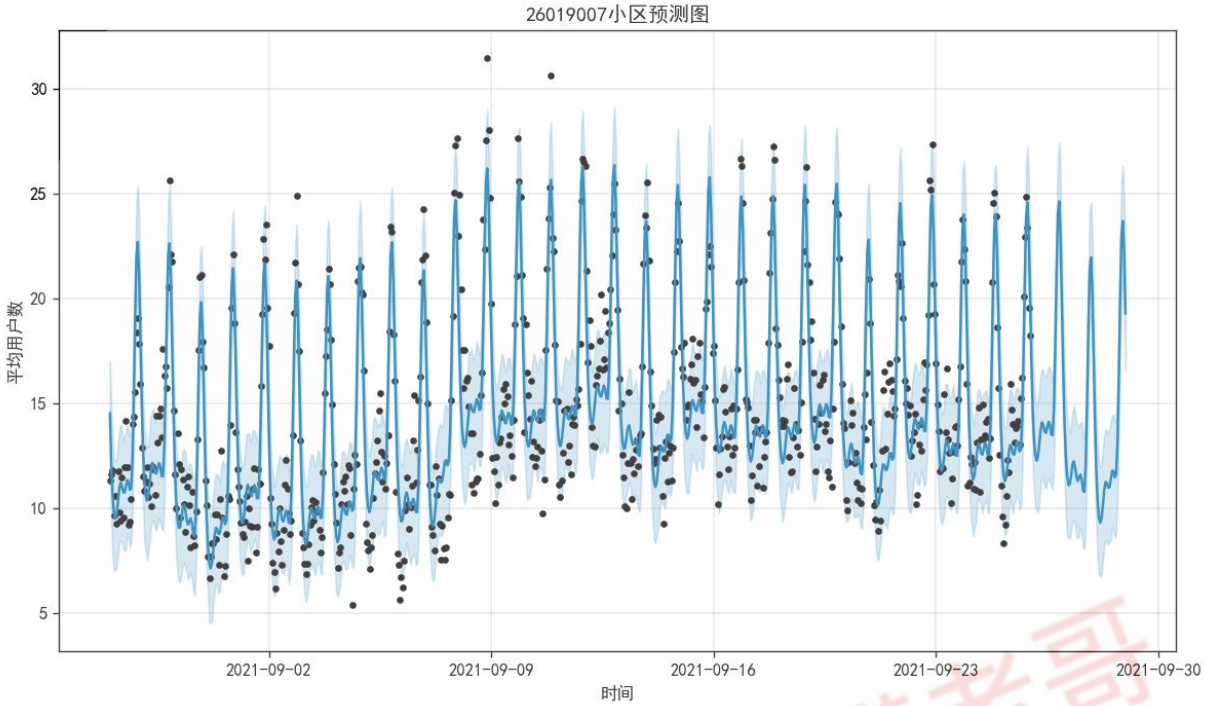
Prophet 包含也时间序列交叉验证功能，以测量使用历史数据的预测误差。这是通过在历史记录中选择截止点来完成的，对于每一个都只使用该截止点之前的数据来拟合模型。然后，我们可以将预测值与实际值进行比较。

4.7 模型三的求解

本文以编号为26019010的小区的小区内的平均用户数指标为例，使用Porphet模型预测未来三天的数据。
首先把该指标29天内的时间序列数据作为输入，分析数据的变化趋势，得到该数据的趋势项、日季节项，画出图像如下：

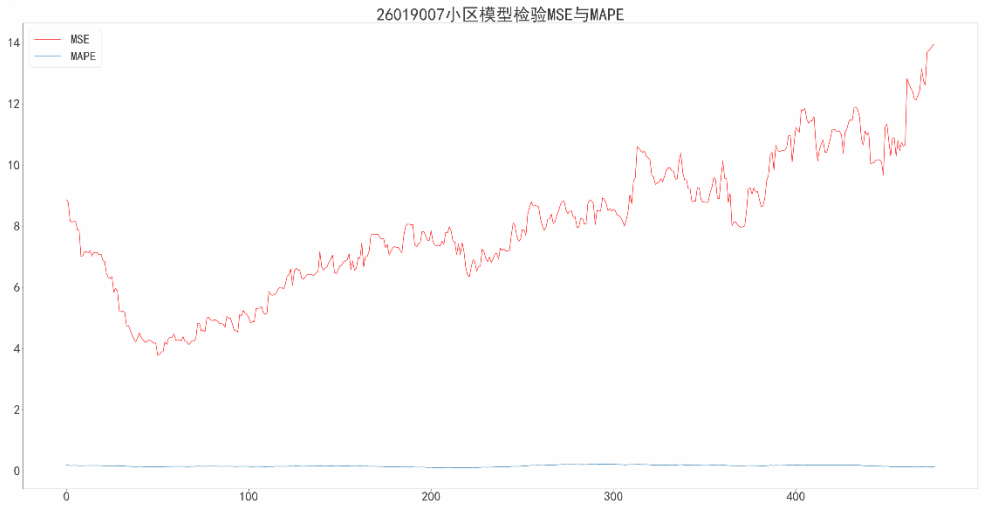


观察该数据的趋势项可以发现，该指标数据在时间的前半部分逐渐增加，然后缓慢下降；观察日季节项可以发现在晚上时指标数据明显高于白天，这符合夜晚手机在线人数高于白天的手机情况。将求得的季节项和趋势项进行求和，得到基于该指标历史数据的Prophet预测模型,预测未来三天的数据，画出其图像如下：



预测结果图像中黑点为历史数据点，黑点附件的蓝线为模型对历史数据点进行拟合得到的曲线，图像最右边的曲线为预测数据点画出的曲线。观察图像可以发现，预测数据值符合历史数据值图像的趋势性和周期性。

为了检验该模型对于指标数据的预测效果，我们使用时间序列交叉验证方法，来测量使用历史数据的预测误差，通过在历史数据中选取截断点，将截断点前的数据作为输入建立预测模型，将预测出的数值与截断点后的实际值进行比较。选取29天的历史数据，把第15天的数据作为截断点，然后每一天进行一次预测，得到交叉验证的交过，并计算出预测数据值和真实值的均方误差MSE和绝对均方误差MAPE，画出其图像如下：



可以看出，预测值的MSE和MAPE均较小，说明预测效果较好。

五、结束语

本文以现实生活中的运营商业务智能运维为场景，以三个性能核心指标数据为研究对象，使用了LOF模型、孤立森林模型、DBSCAN模型、神经网络模型、Prophet模型对运营商基站覆盖小区核心指标数据的异常检测、异常预测、趋势预测等问题进行了研究，对三个指标29天的数据和使用模型得到的数据结果进行了可视化操作，并结合考虑了实际业务情况的合理性对得到的结果进行了处理和修正，使得到的结果符合实际情况，更加具有合理性和实际应用价值。

参考文献

[1] 纪宁,张华伟,刘远岗,李佳.基于AI人工智能化网络系统运维[J].石油知识,2021(05):46-47

[2] 李远.基于大数据的云计算中心智能运维系统的应用[J].电力设备管理,2021(08):37-38+41.

[3] 徐友恒.数据中心一体化智能运维管理平台建设研究[J].中国管理信息化,2021,24(18):105-106.

[4] 史浩鹏,马凡琳,杨轲.基于人工智能识别技术的运维检修监测系统设计[J].电气应用,2021,40(08):86-91.

附录

代码名称	代码解决的问题	代码电子版所在的位置
数据预处理与第一问第二问.ipynb	第一问与第二问	支撑材料中
第三问.ipynb	第三问	支撑材料中