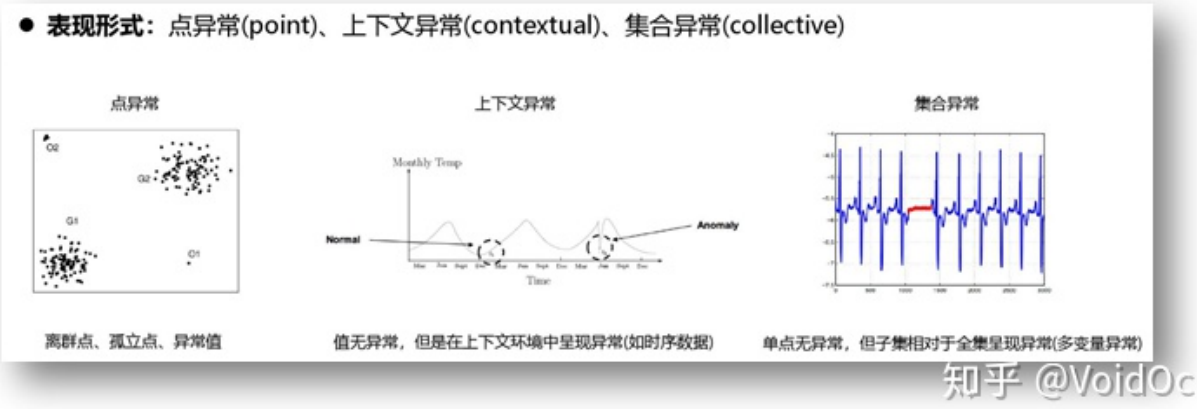
Time Series Anomaly Detection Portrait

基于时间序列的异常检测画像方法

一、介绍

异常检测（Anomaly detection）是目前时序数据分析最成熟的应用之一，定义是从正常的时间序列中识别不正常的事件或行为的过程。

常见的异常类型有点异常、上下文异常以及集合异常。

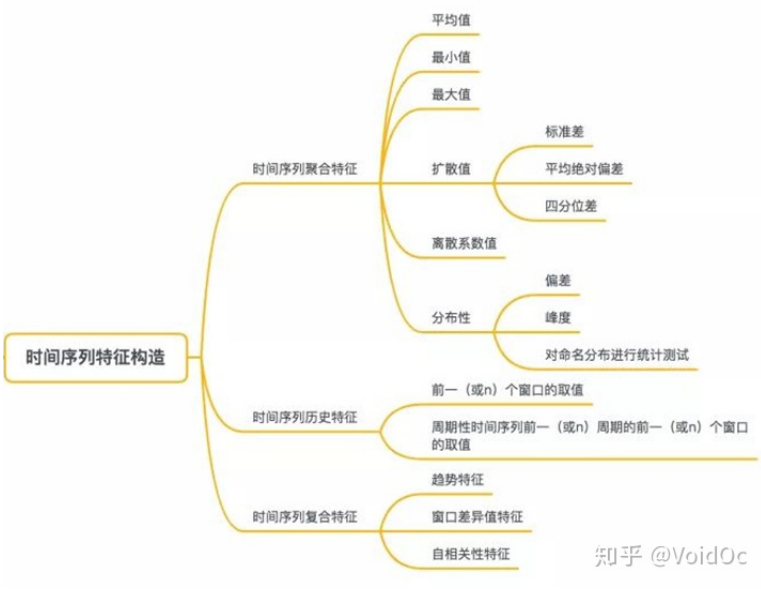


一般来说，异常检测的过程分为3个步骤：特征提取、模型训练和异常判断。在深度学习方法阶段，算法经常将特征提取步骤与模型训练步骤结合到一起，用端到端的方法检测异常。今天我们要讨论的序列模型下的异常检测画像。

在介绍画像方法之前，我先简单介绍一下时间序列的特征构造。

二、基于时间序列下的异常检测画像

在介绍时间序列下的异常检测画像之前，我们先看一下时间序列的特征构造：一般来说，时间序列的特征构造主要分为三类特征：聚合特征（例如：平均值、最大/最小值、标准差等）、历史特征（前1～n个窗口等取值）、复合特征（取汁特征、窗口差异值特征等）。



在查阅了一些基于时间序列下的异常检测论文后，我查阅到的画像方法依据异常检测算法划分主要有以下几种：

基于模型的算法

许多异常检测技术首先建立一个数据模型。异常是那些同模型不能完美拟合的对象。例如，数据分布模型可以通过估计概率分布的参数来创建。如果一个对象不能很好地同该模型拟合，即如果它很可能不服从该分布，则它是一个异常。

- 针对单维数据

  1 集中不等式：集中不等式是数学中的一类不等式，描述了一个随机变量是否集中在某个取值附近。如马尔可夫不等式、切比雪夫不等式等。

  2 统计置信度检验：如3σ-法则。

- 高维数据

  1 马氏距离：用来计算样本X与中心点μ的距离，可以用来做异常分值。马氏距离最强大的地方是引入了数据之间的相关性（协方差矩阵）。 而且马氏距离不需要任何参数，这对无监督学习来说是一件很好的方法。

  2 单个或者混合高斯分布： 但由于在某些情况下，很难建立模型，因为数据的统计分布未知或没有训练数据可用。在这些情况下，可以使用其他的不需要模型的技术。

  ## PCA

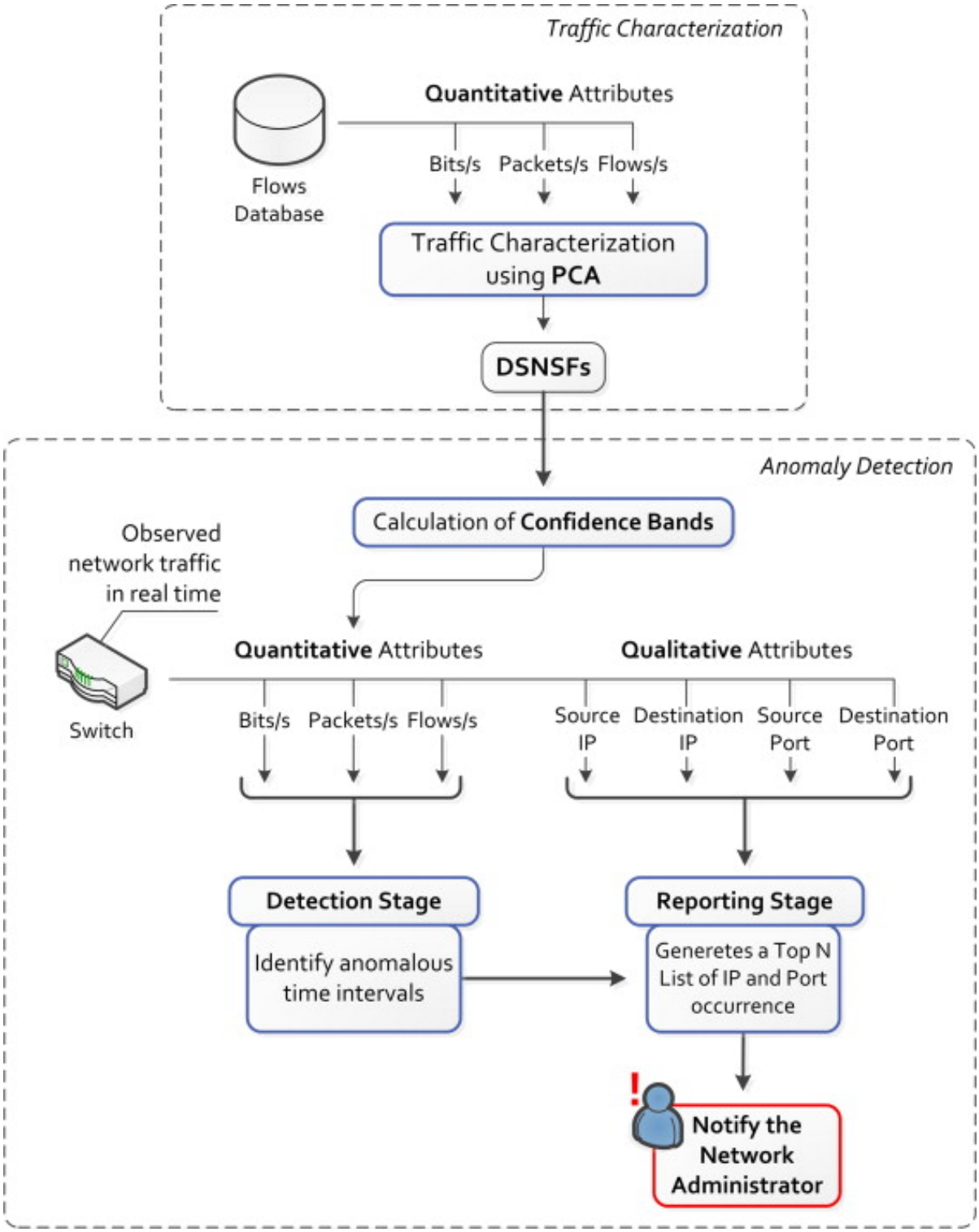
  PCA是一种降维方法，它原理是通过构造一个新的特征空间，把原数据映射到这个新的低维空间里。

  /////

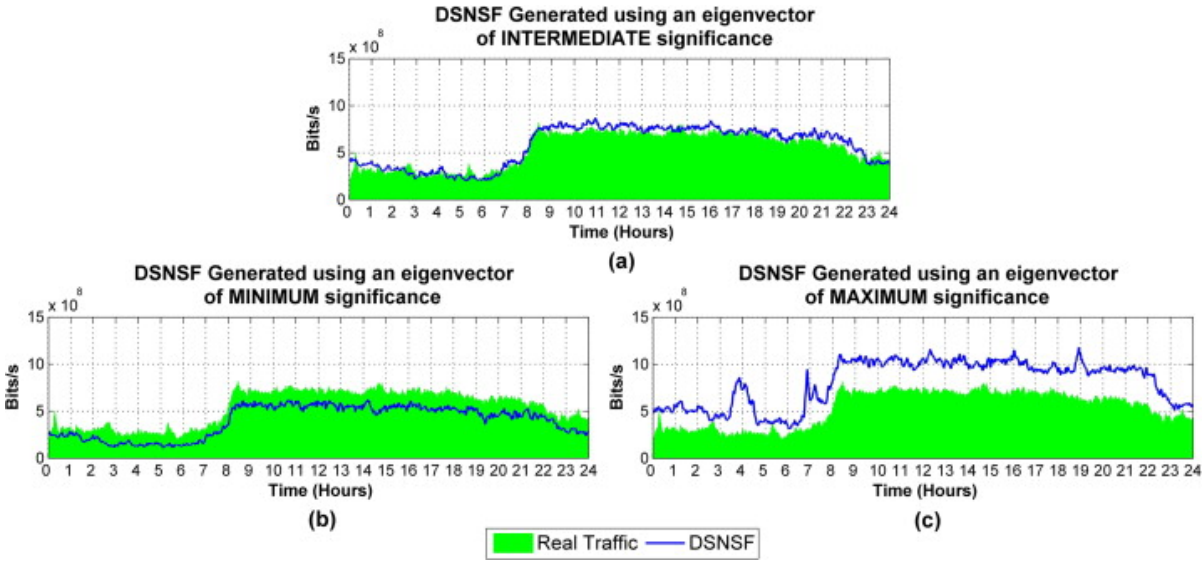
在2015年发布的一篇使用主成分分析和流量分析的基于轮廓的自主异常检测系统论文 Autonomous profile-based anomaly detection system using principal component analysis and flow analysis中，作者提出了一种基于主成分分析（PCA）的自主异常检测系统。**此方法使用流量分析创建一个称为「网络段数字签名」的网络配置文件Digital Signature of Network Segment using Flow Analysis (DSNSF)**，该数字签名是通过对历史数据分析后得到的流量异常检测的阈值，预测网络流量活动是否正常行为。使用到的数据包括了七个流量属性：比特位，数据包、流量数量、源IP地址和目标IP地址和端口。

在标准PCA算法中，输入是一个n×p矩阵，由代表维度（变量）的p列和n条线组成，作为每个变量的n个样本。

该论文从流量记录中收集的数据的排列方式是，每天的流量数据由三个向量表示，这些向量包含与每天24小时相对应的总比特，数据包和流量。然后，分别为每个属性构造流分析算法的输入矩阵。这个nxp的矩阵中的p维是作为生成流分析的基础而选择的p天流量移动数据，n行是从流量记录中提取的每秒传输的比特，数据包或流量数量的n个度量。然后通过一系列计算，得到该数据的协方差矩阵的特征值和特征向量，在计算了所有特征向量和特征值之后，具有最高特征值的特征向量被称为主成分，标准PCA使用它们来构成一个新的简化数据集。



但在论文中比较特别的是，该论文并没有选取具有最高特征值的特征向量作为数字签名，而是使用了中间特征值的特征向量作为数字签名，主要原因是因为具有最大特征值的特征向量表示数据集所有属性之间方差最大的维，并且基于该前提下创建的数字签名将对网段的正常流量模式产生影响，例如不规则数据的合并。



（绿色表示2012年11月8日的24小时的位/秒的实际流量，蓝色表示当天的DSNSF）

基于深度学习的方法

随着贝叶斯与深度网络的结合，能够衡量模型的不确定性的贝叶斯深度网络在各个领域中应用广泛，也诞生了很多专门针对于神经网络计算不确定性的方法，具体文献可以参考Deep and Confident Prediction for Time Series at Uber介绍的Uber用于预测和检测的一个LSTM框架，就是基于不确定性的思路做的。

这篇论文使用美国和加拿大八个代表性大城市的四年每日完成旅行来说明模型性能(we illustrate the model performance using the daily completed trips over four years across eight representative large cities in U.S. and Canada, including Atlanta, Boston, Chicago, Los Angeles, New York City, San Francisco, Toronto, and Washington D.C.)。使用步长为1的滑动窗口构建样本，其中每个滑动窗口包含前28天作为输入，并且旨在预测即将到来的一天。 对原始数据进行对数转换以减轻指数效应。接下来，在每个滑动窗口内，从所有值中减去第一天，以便移除趋势并且训练神经网络用于增量值。 在测试时，可以直接恢复这些转换以获得原始比例的预测。此外，论文为了解释编码器提取的嵌入特征，

基于相似度度量

通常可以在对象之间定义**邻近性度量**，并且许多异常检测方法都基于邻近度。异常对象是那些远离大部分其他对象的对象。这一领域的许多方法都基于距离，**称作基于距离的离群点检测方法**。当数据能够以二维或三维散布图显示时，通过寻找与大部分其他点分离的点，可以从视觉上检测出基于距离的离群点。

- 基于距离，例如KNN（K-nearset-neighbour)
- 基于密度

  参考Eagle: User profile-based anomaly detection for securing Hadoop clusters基于用户配置的Hadoop集群的异常检测Eagle。此篇论文中采用了PCA和密度预测两种机器学习算法，Eagle的数据源来自用于用户活动监视的Hadoop文件系统HDFS和Hive的审计日志，Eagle包含了一个离线训练的组件和一个在线异常检测组件。离线训练组件用于从历史日志提取特征信息中生成用户配置文件，在线组件用于将用户活动与用户配置文件相匹配并检测异常。

  论文中，提取了用户活动统计信息（包括HDFS数据访问时间，IP地址和访问时区等信息）和用户操作信息（包括用户在任何给定时间间隔执行HDFS操作的频率。间隔可以是数据聚合频率，例如每小时，每天，每周等）来生成配置文件。

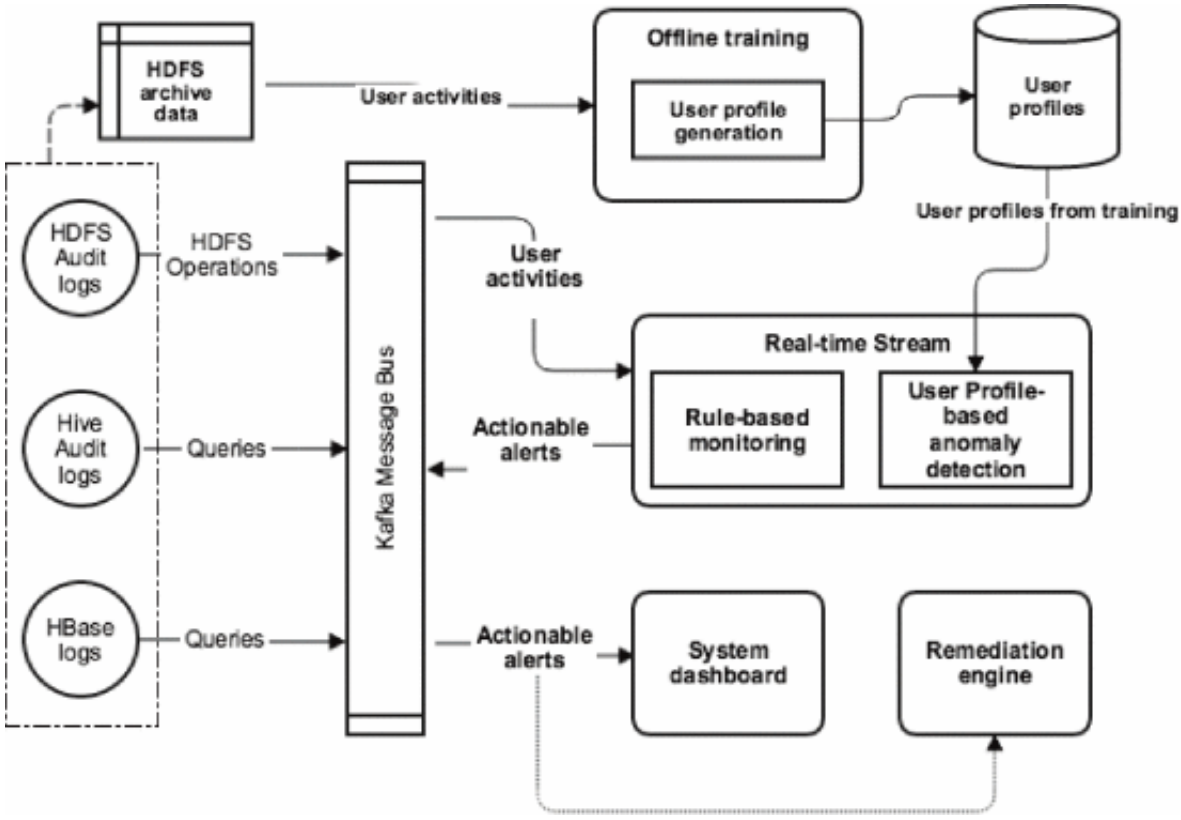
在Eagle的Offline Training Component部分，论文使用了密度预测和PCA两种算法。

在这种算法中，其思想是根据观察到的样本数据点为每个用户评估概率密度函数。每个数据点的尺寸等于所选要素的数量。此方法采用了刚刚提到的HDFS数据访问时间，IP地址和访问时区、定时间间隔执行HDFS操作的频率等15个数值作为特征。然后论文将训练数据集分为两部分：训练集和交叉验证集。在应用算法之前，论文首先通过计算数据集**每个特征的均值和标准差**，然后**从均值中减去实际数据点，然后将结果除以标准差，对训练数据集进行归一化**。然后，论文使用**高斯分布函数作为计算概率密度的方法**。（由于我们的数据集包含15个特征，并且每个特征在条件上彼此独立，因此可以通过分解每个概率密度然后相乘来计算最终的高斯概率密度。）

我们

在online component中，一旦我们计算某个用户的概率，我们将其转换为对数值，并将其与在offline component中该用户用交叉验证集中计算的阈值进行比较，如果概率值低于从AACC计算的阈值，我们将相应的数据点作为异常信号。

$$AAcc = \frac{1}{2}(\frac{TP}{TP+FN} + \frac{TN}{TN+FP})$$



- 基于聚类

  例如K-means等聚类方法进行聚类，哪些远离大簇的对象可以视为异常对象。

- 基于划分：孤立森林，利用一种名为**孤立树**$iTree$的**二叉搜索树结构**来孤立样本机器学习算法（二十）：孤立森林 iForest （Isolation Forest），由于异常值的数量较少且与大部分样本的疏离性，异常值会被更早的孤立出来，也即异常值会距离$iTree$的根节点更近，而正常值则会距离根节点有更远的距离。

- 基于谱/线性模型：**通过与正常谱型进行残差对比，发现异常。**简单的线性模型就是相关性分析。利用一些自变量来预测因变量。比较重要的一个应用就是时序数据或者空间轨迹数据。我们可以利用上一个值或者上几个值来预测当前值，将预测值和实际值的误差作为优化对象，这样就建立了一个正常数据的模型，背离这个模型的就被当作异常值，预测值和实际值的误差也可以作为异常分值

来提供。

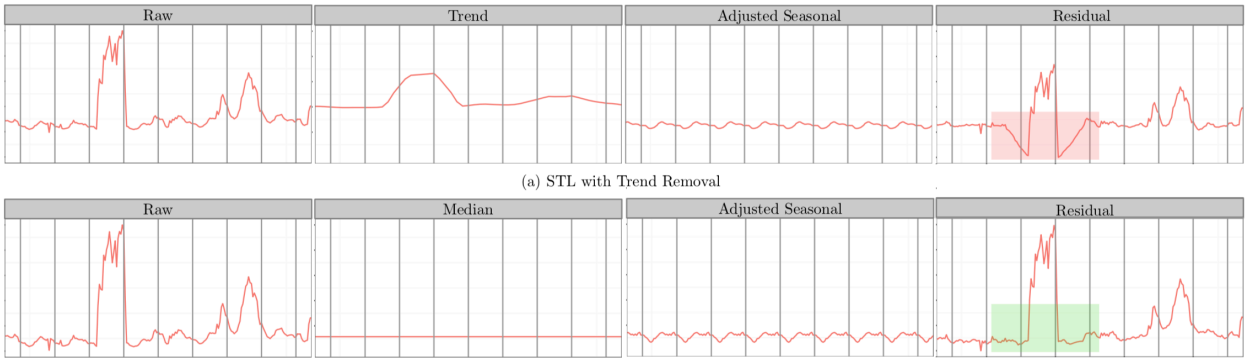
这里，我要针对基于线性模型算法的画像方法做一些介绍。

线性模型/谱

S-ESD

鉴于时间序列数据具有周期性（seasonal）、趋势性（trend），异常检测时不能作为孤立的样本点处理；故而Twitter的工程师提出了S- ESD (Seasonal ESD)与S-H-ESD (Seasonal Hybrid ESD)算法，将ESD扩展到时间序列数据。

将时间序列数据分解为趋势分量、周期分量和余项分量。想当然的解法——将ESD运用于STL分解后的余项分量中，即可得到时间序列上的异常点。但是，我们会发现在余项分量中存在着部分假异常点（spurious anomalies）。



(a) STL with Trend Removal

(b) STL with Median Removal

在红色矩形方框中，向下突起点被误报为异常点。为了解决这种假阳性降低准确率的问题，S-ESD算法用中位数（median）替换掉趋势分量；余项计算公式如下：

$$R_X = X - S_X - \tilde{X}$$

其中，X为原时间序列数据，SX为STL分解后的周期分量，X~为X的中位数。

Shapelets

> 时间序列建模旨在发现按时间顺序排列的数据中的时间关系。 它吸引了广泛的研究领域，如图像对齐[2]、语音识别[3]等。 这里的关键问题是如何提取时间序列的代表性特征。 以前的框架很大一部分从经典的特征工程和表示学习到基于深度学习的模型。 虽然这些方法取得了良好的性能[4,5]，但它们也因缺乏可解释性而受到批评。因此，有人在2020年人工智能领域顶级会议上有文章提出了一种运用Shapelets将时序转化为图用于可解释可推理的异常检测。

Shapelet，代表一类可以在分类场景中提供直接的解释性和见解的时间序列子序列[6]，并且基于Shapelet的模型在各种研究中都被证明是有前景的[7,8,9]。

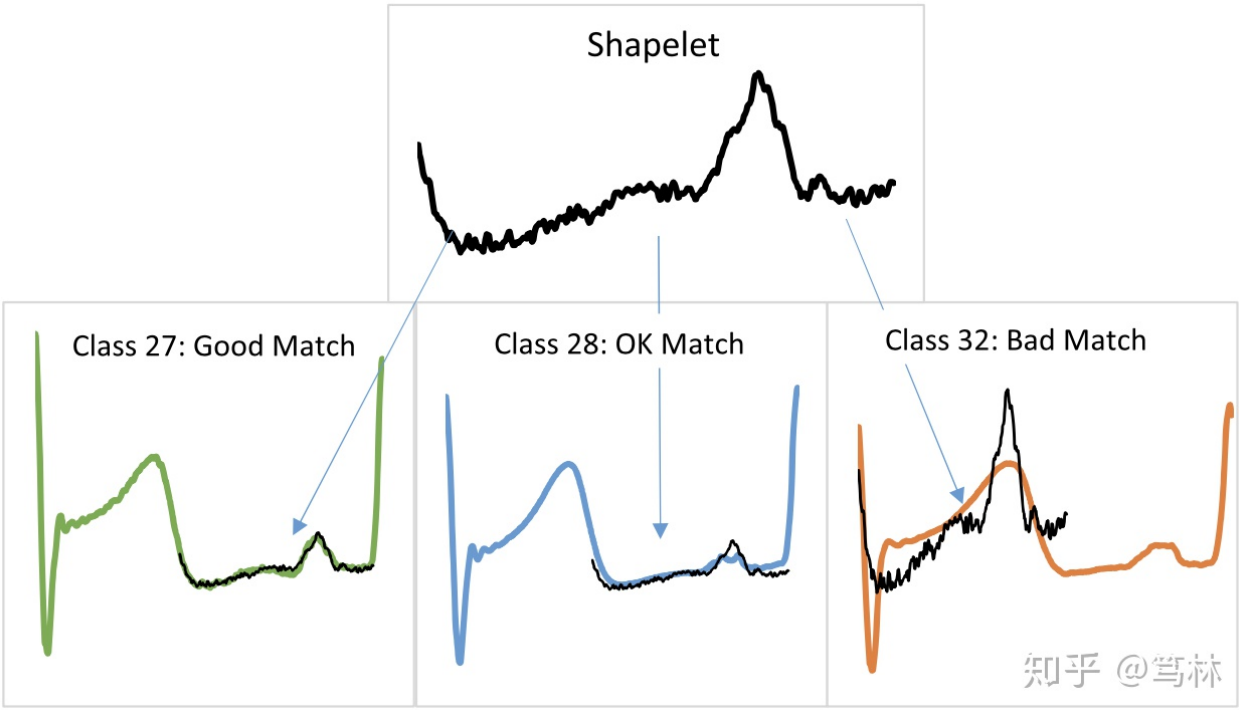
文章通过运用Shapelets（一种可自动挖掘具有代表特征的时序子序列的方法出发），分析不同Shapelet之间的关系，构建图进行表示，提供一种可推理可解释且具有良好表现的时序模型。

如下图所示，其展现了一个Shapelets的案例。每一个Shapelet(或理解为子序列波形)会在整个时序中找到最匹配的位置，以及匹配程度。现有的工作主要着眼于静态的分析Shapelet。但是，在现实世界中，Shapelet通常是动态的，这体现在两个方面：

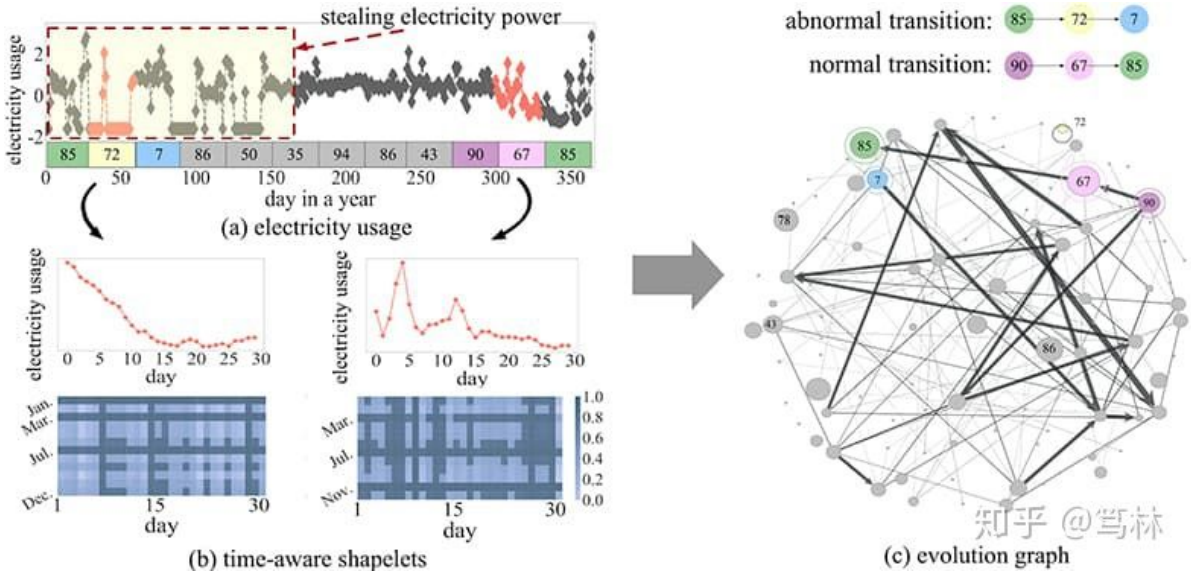
- 首先，出现在不同时间段的同一个Shapelet可能会产生不同的影响。例如，在检测到窃电用户的场

景下，夏季或冬季的低电耗比春季更可疑，因为制冷或供暖设备的使用使得用电量应该更高。

- 其次，确定Shapelet的演化方式对于全面理解时间序列至关重要。实际上，在特定时间具有较小值的Shapelet很难将窃电用户与确实消耗少量电的普通用户区分开。一种替代方法是识别曾经具有高电耗的Shapelet，但突然消耗很少的用户。换句话说，这里的一个重要线索是Shapelet如何随时间演变。



具体案例解释



本文开始展示的图中显示了一个来自真实用电记录的具体示例，它可以更好地展现该文章构建图的动机：图a显示了窃电用户在一年中的用电情况：1月到5月是没被抓时的用电曲线，之后是被抓后正常的用电情况。文章给每个月分配了最有代表性的Shapelet，并在图b中显示了两个的Shapelet（＃72和＃67，一个Shaplet v 是代表某个类别的时序片段。），以及它们的时间意识因素，其中深色区域表示Shapelet相对于浅色区域更具区分性。Shapelet演变图如图c所示，说明了Shapelet在正常情况下如何从一个转移到另一个：对于正常的用电量记录，其Shapelet的过渡有一条清晰的路径（＃90→＃67→＃ 85）。但是，对于异常数据，他们走一条不明显的路径（＃85→＃72→＃7），这表明Shapelet转移路径的连通性可以为检测异常时间序列提供了参考标

准。最后，文章将学习到的Shapelet和时间序列表示的问题转化为图嵌入问题，并用图算法进行解决。

方法：

捕捉具有时间意识的Shapelet

正式地，一个Shaplet v 是代表某个类别的时序片段。更确切地说，它可以通过某些特定的指标，将时序 T 分成两个较小的集合，一个接近 v 而另一个远离，例如对于时间序列分类任务，可以将正样本和负样本放入不同的组中，具体可以形式化为 $\mathcal{L}=-g(S_{pos}(v,T),S_{neg}(v,T))$

这里 $S_*(v,T)$ 表示相对于特定组 T_* 的距离，函数 g 接受两个有限集作为输入，返回一个标量值以指示这两个集合有多远，它可以是信息增益或集合上的一些不相似性度量，即KL散度。

为了捕获Shapelet 的动态性，文章定义了两个因素来定量测量Shapelet在不同水平上的时序影响。具体来说，文章介绍一个**局部因素** w_n

来表示特定Shapelet的第n个元素的内部重要性。之后，Shapelet 与时序片段之间的距离可以被重新定义为 $\hat{d}(v,s|w)=\tau(v,s|a^*,w)=(\sum_{k=1}^{p}\ w_{a_1^*(k)}\cdot(v_{a_1^*(k)}-s_{a_2^*(k)})^2)^{\frac{1}{2}}$

这里 a^* 指的是DTW距离的最佳对齐。另一方面，在**全局范围**内，文章旨在衡量跨段的时间效应对Shapelet的判别力的影响。直觉上Shapelet在不同的时间步长可能代表完全不同的含义，这里可以直接通过添加段级权重来测量此类偏差。正式地，文章里设定了一个全局因素 u_m 捕获跨段影响，然后Shapelet v 与时序 t 之间的距离可以写为

$$\hat{D}(v,t|w,u)=\min_{1\le k\le m}u_k\cdot\hat{d}(v,s_k|w)$$

然后给定一个分类任务，这里可以建立一套监督学习方法，以选择最重要的具有时间意识的Shapelet，并可以为每一个Shapelet学习其相应的时间因素 w_i 和 u_i 。特别地，这里有一个从所有子序列中选择出的作为shapelet候选者的片段池，以及一组带标签的时间序列 T 。对于每个候选者 v ，都有以下目标函数： $\hat{\mathcal{L}}=-g(S_{pos}(v,T),S_{neg}(v,T))+\lambda||w||+\epsilon||u||$

在分别从Shapelet候选者那里学习了时序因素之后，文章选择损失最小的前K个shapelet作为最终的具有时间意识的Shapelet。

构建Shapelet演化图

Shapelet 演化图是有向加权图 $G=(V,E)$ ，其中图由K个顶点所组成，每个顶点表示一个shapelet，每个有向边 $e_{i,j}\in E$ 与其权重 $w_{i,j}$ ，表示在相同的时间序列中，shapelet $v_i\in V$ 跟着另一个shapelet $v_j\in V$ 的出现概率。这里的关键思想是，图中的路径可以自然反映出shapelet的演变及其过渡模式，然后可以将图嵌入算法应用于学习shapelet以及时间序列表示。文章首先分配每个时序片段 s_i 到距离最近的几个Shapelets。详细地说，这里将shapelet的赋值概率标准化为

$$p_{i,j}=\frac{\max(\hat{d_{i,*}}(v_{i,*},s_i))-\hat{d_{i,j}}(v_{i,j},s_i)}{\max(\hat{d_{i,*}}(v_{i,*},s_i))-\min(\hat{d_{i,*}}(v_{i,*},s_i))}$$ 这里有

$$\hat{d_{i,*}}(v_{i,*},s_i)=u_*[i]*\hat{d}(v_{i,*},s_i|w_*)$$

> 的预定义约束， $\hat{d_{i,*}} \leq \delta$ 。然后，对于每个对 (j,k) ，文章从shapelet创建加权边 $v_{i,j}$ 到 $v_{i+1,j}$ ，并通过权重 $p_{i,j} \cdot p_{i+1,k}$ 合并所有的重复边。最后，将从每个节点获得的边缘权重归一化为1，这自然会解释每对节点之间的边缘权重。

环比

在基于时间序列异常检测模型的画像中，有的论文使用了计算时间序列的环比等统计特征进行画像。例如针对360的lvs流量异常检测的阈值设置方法中，通过计算时间序列的环比，EWMA（**指数加权移动平均**），同比，振幅，以及检测点的孤立森林的分数，再与各个指标的阈值进行比较判断检测点是否为异常点。各个指标的阈值是利用对17000余个vip流量数据进行统计，通过数据的前5%的值作为初始阈值。

> ### 环比：
>
> 通过计算检测点与过去几个时刻点数据差的绝对值，得到环比列表，再与环比阈值比较，多数超过阈值则说明检测点是异常点，否则检测点为正常点。若检测点异常而且检测点高于上一时刻点，则为突增情况，否则为突降情况。对17000余个vip流量数据进行统计，并获取每个vip的环比列表最大值，取前5%的值作为初始阈值。
>
> ### EWMA：
>
> EWMA是以指数式递减加权的移动平均。各数值的加权影响力随时间而指数式递减，越近期的数据加权影响力越重，但较旧的数据也给予一定的加权值。加权的程度以常数α决定，α数值介乎0至1，本文中α的取值为1/21。具体的EWMA公式如下：
>
> 利用上述公式能够计算变换后的序列的均值和方差，如果检测点在3sigma以外，那么该点是异常点，否则为正常点。如果检测点是异常点，而且检测点高于上一时刻点，则为突增情况，否则为突降情况。
>
> ### 同比：
>
> 同比的判断与环比类似，也是通过计算获取一个同比列表，但是异常点的判断方式略有不同。同比的判断是：如果检测点比上一时刻点数值大，而且检测点值大于同比中的最大值与同比上限阈值的乘积要大，那么检测点是突增异常点；如果检测点比上一时刻点数值小，而且检测点小于同比中的最小值与同比下限阈值的乘积，那么检测点是突降异常点；若检测点不满足上述两种情况，那么检测点是正常点。获取vip数据的同比上/下限的前5%作为初始阈值。
>
> ### 振幅：
>
> 通过计算过去几天检测时刻与上一时刻的增幅的比例得到振幅列表。振幅指标的判断与同比类似，也需要上/下限两个阈值辅助判定，这里就不详细展开了，获取vip数据的振幅上/下限的前5%作为初始阈值。
>
> ### iForest算法：
>
> iForest算法是周志华老师于2010年设计的一种异常点检测算法，该方法同过利用数据构建iTree，进而构建iForest，是一种无监督的检测方法，具有很好的效果，具体算法可参见：http://www.cnblogs.com/fengfenggirl/p/iForest.html

三、变量处理方法

主要针对的是非数值型变量，因为极值分析和统计算法依赖于统计量化，例如均值或者标准差，对于非数值型变量，这些统计量化将不再有意义；但通过一些改变我们就能将上面介绍的模型拓展为适用于非数值型变量的模型。

4.1 统计概率模型：

唯一的区别就是变量不再默认服从特定分布（如高斯），而需要单独定义概率分布（按比例），并按乘积方式与数值变量组合以创建单个多元分布。

4.2 线性模型：

1.One-hot码二进制转换，一个值对应一个种类，但容易维度爆炸，且无法体现不同类别的不同权重。可以通过将每列除以其标准偏差（deviation）来进行归一化。

2.潜在语义分析（Latent Semantic Analysis）

4.3 基于相似度量模型：

a) 基于距离：

1.基于非数值属性的统计频率计算相似度，比如稀有属性的匹配比常规属性的匹配权重要高。

2.基于数据的统计邻域计算相似度，比如文本变量中“红色”和“橙色”比“红色”和“蓝色”更相近，但要求人为区分属性值之间的语义关系。

b) 基于密度：基于密度的方法可以自然地扩展到离散数据，因为通常数值属性也将离散化以创建频率曲线。

4.4 数据降维度

这里我主要介绍两类：一类是运用主成分分析PCA（Principal Component Analysis）方法。还有一类是运用Shapelets。

> 在提出的算法中，创建仅使用一个主分量（特征向量）的数字签名。但是，不是选择具有最高特征值的特征向量，而是选择具有相应中间值特征值的特征向量。这是因为最重要的特征向量表示数据集所有组件之间方差最大的维，并且基于该前提创建数字签名将对网段的正常流量模式产生影响，例如不规则的合并。当天的流量中存在异常值。同样，使用低重要性特征向量可能涉及服务器崩溃或断电的情况的合并。

> References
>
> [1]王志国, 章毓晋. 监控视频异常检测：综述. 清华大学学报(自然科学版), 2020, 60(6): 518-529.
>
> [2]Elastic开发者大会2018 - 基于 Elasticsearch 的 AI 异常检测和画像系统 - 朱彦安，安恒信息
>
> [3]郭小芳,李锋,宋晓宁.一种基于PCA的时间序列异常检测方法[J].江西师范大学学报(自然科学版),2012,36(03):280-283.
>
> [4]Gilberto Fernandes, Joel J.P.C. Rodrigues, Mario Lemes Proença, Autonomous profile-based anomaly detection system using principal component analysis and flow analysis, Applied Soft Computing, Volume 34, 2015,Pages 513-525,ISSN 1568-4946,

> https://zhuanlan.zhihu.com/p/142320349
>
> https://zhuanlan.zhihu.com/p/142525791
>
> https://petecheng.github.io/Time2Graph/?spm=a2c4e.10696291.0.0.643c19a4mreJAk shapeless
>
> https://blog.csdn.net/weixin_39910711/article/details/107754032#1.9%C2%A0基于深度学习的方法