基于预测比较模型的异常检测

> *16周时，开始尝试一些新的思路与原有方法的对比，在比较了传统的滑动平均和现有的指数平均方法之后，我们采用Holt Winters方法从预测的角度来做误差分析。*

新的方法

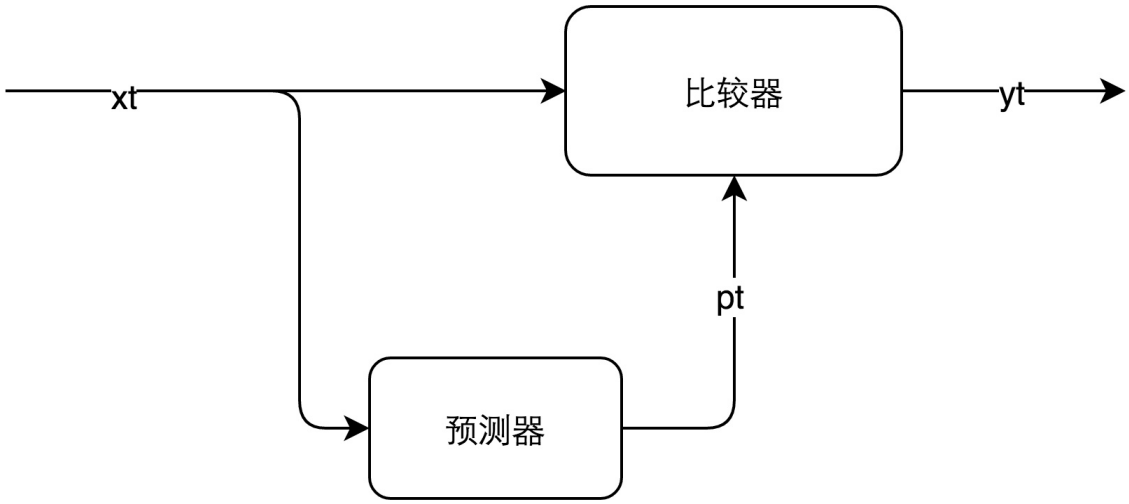
在序列数据的异常检测过程中，我们既可以直接使用对序列进行异常检测的算法，也可以先对序列数据进行特征提取然后转化为传统的离群点检测。

离群点检测方法	方法描述	方法特点
基于统计	大部分的基于统计的离群点检测方法是构建一个概率分布模型，并计算对象符合该模型的概率，把具有低概率的对象视为离群点	基于统计模型的离群点检测方法的前提是必须知道数据集服从什么分布；而对于高维的数据，可能每一维度服从的分布都不太一致，所以通常对高维数据来讲通常效果较差。
基于邻近度	通常可以在数据对象之间定义邻近性度量，把远离大部分点的对象视为离群点。	算法假定离群点是离散的，低维数据我们可以作图观察，而高维数据我们无法观察，所以难以确定有效的参数和全局阈值，效果较差。
基于聚类	一种利用聚类检测离群点的方法是直接丢弃远离其他簇的小簇；另一种是对数据点属于簇的程度进行评价，去除得分较低的点。	聚类算法产生的簇的质量对该算法产生的离群点的质量影响非常大，对数据的可分类性要求较高

之前考虑的算法方向主要是在『基于统计』+『基于聚类』的这个方向来考量。

而如今我发现了一种新的方法可以作为采用与尝试，即上图中『基于临近度』，也是一种使用历史数据判断当前数据的方法。

基于预测的异常检测模型如下图所示，x_t 是真实数据，通过预测器得到预测数据，然后 x_t 和 p_t 分别作为比较器的输入，最终得到输出 y_t，y_t 是一个二元值，可以用+1（+1表示输入数据正常），-1（-1表示输入数据异常）表示。



如果说我们设置异常检测的模型如此，那么我们可以从两个以下方面入手，一是预测器的优化，二是比较器的优化。

预测器优化

同比环比预测器

同比环比是比较常用的异常检测方式，它是将当前时刻数据和前一时刻数据（环比）或者前一天同一时刻数据（同比）比较，超过一定阈值即认为该点异常。如果用图模型来表示，那么预测器就可以表示为用当前时刻前一时刻或者前一天同一时刻数据作为当前时刻的预测数据。

如果将不同日期、时刻的监控数据以矩阵方式存储，每一行表示一天内不同时刻的监控数据，每一列表示同一时刻不同日期的监控数据，那么存储矩阵如下图所示：

假如需要预测图中黄色数据，那么环比使用图中的蓝色数据作为预测黄点的源数据，同比使用图中红色数据作为预测黄点的源数据。

基线预测器

同比环比使用历史上的单点数据来预测当前数据，误差比较大。t 时刻的监控数据，与 $t-1, t-2, \ldots$ 时刻的监控数据存在相关性。同时，与$t-k, t-2k, \ldots$ 时刻的数据也存在相关性（k为周期），如果能利用上这些相关数据对t时刻进行预测，预测结果的误差将会更小。

比较常用的方式是对历史数据求平均，然后过滤噪声，可以得到一个平滑的曲线（基线），使用基线数据来预测当前时刻的数据。该方法预测 t 时刻数据（图中黄色数据）使用到的历史数据如下图所示（图中红色数据）：

Holt-Winters预测器

同比环比预测到基线数据预测，使用的相关数据变多，预测的效果也较好。但是基线数据预测器只使用了周期相关的历史数据，没有使用上同周期相邻时刻的历史数据，相邻时刻的历史数据对于当前时刻的预测影响是比较大的。对于 **Holt-winters** 预测期模型，它建议使用黄色点左上方的所有数据。

Holt-Winters是三次指数滑动平均算法，它将时间序列数据分为三部分：残差数据 $a(t)$，趋势性数据 $b(t)$，周期性数据 $s(t)$。使用Holt-Winters预测 t 时刻数据，需要 t 时刻前包含多个周期的历史数据。

详细信息看这里：https://www.otexts.org/fpp/7/5

各部分迭代的简化计算公式如（其中 k 为周期）：

1. $a[t] = \alpha(Y[t] - s[t-k]) + (1-\alpha)a[t-1]$
2. $s[t] = \gamma(Y[t] - a[t]) + (1-\gamma)(s[t-k])$

预测值：$Y[t+h] = a[t] + s[t-k+1+(h-1)\ mod\ k]$

为了将算法应用到线上的实时预测，我们可以将Holt-Winters算法拆分为两个独立的计算过程：

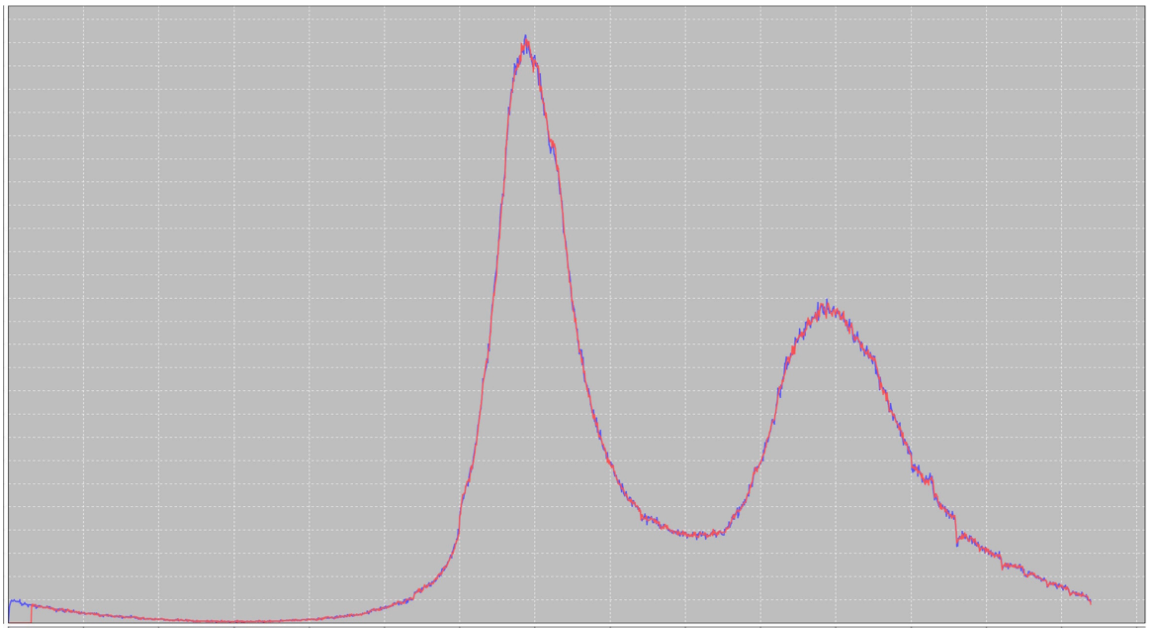
1. 定时任务计算序列的周期数 $s(t)$。

   $S(t)$ 不需要实时计算，只用按照周期性更新即可，使用 Holt-Winters 公式计算出时间序列的周期性数据。

2. 对残差序列做实时预测。

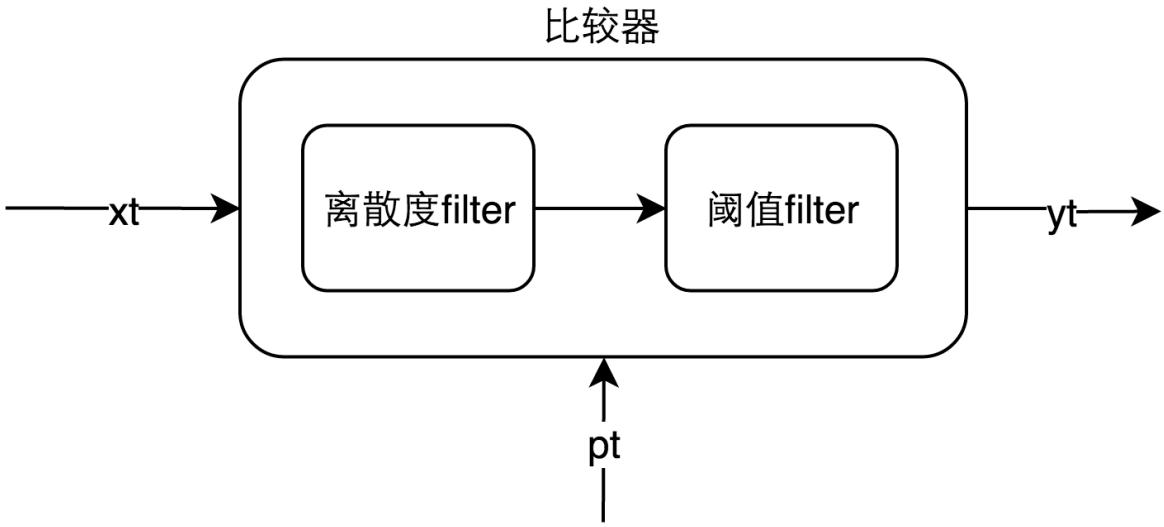
   计算出周期数据后，下一个目标就是对残差数据的预测。使用下面的公式，实际监控数据与周期数据相减得到残差数据，对残差数据做一次滑动平均，预测出下一刻的残差，将该时刻的残差、周期数据相加即可得到该时刻的预测数据。对于分钟数据，则将残差序列的长度设为60，即可以得到比较准确的预测效果。

   > *红线为预测数据，蓝线为真实数据*



比较器优化

预测器预测出当前时刻传感器的预测值后，还需要与真实值比较来判断当前时刻数据是否异常。一般的比较器都是通过阈值法，比如实际值超过预测值的一定比例就认为该点出现异常，进行报警。这种方式错误率比较大。在传感器数值模型的报警检测中没有使用这种方式，而是使用了两个串联的Filter，只有当两个Fliter都认为该点异常时，才进行报警，下面简单介绍一下两个Filter的实现。



离散度Filter

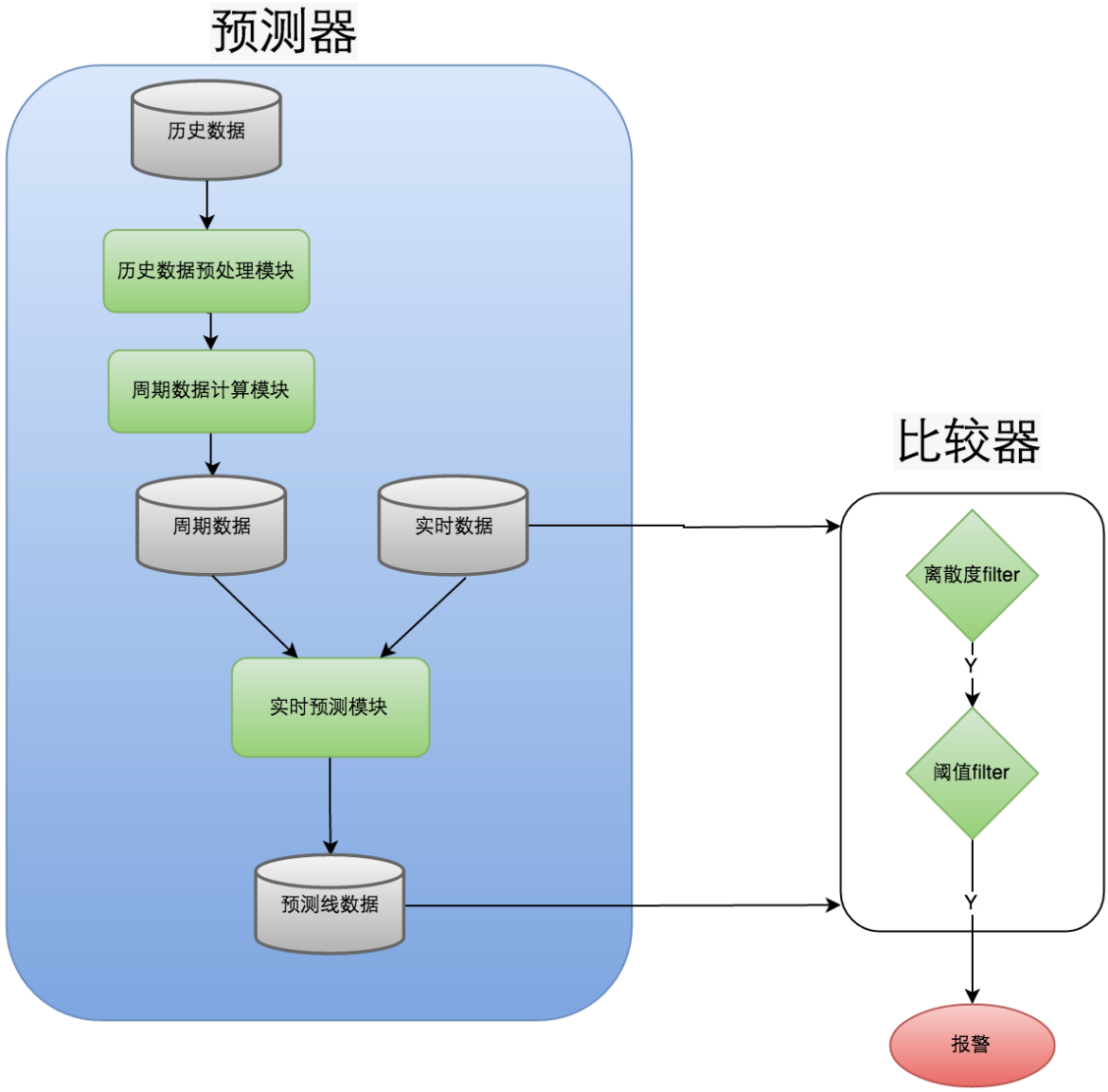
根据预测误差曲线离散程度过滤出可能的异常点。一个序列的方差表示该序列离散的程度，方差越大，表明该序列波动越大。如果一个预测误差序列方差比较大，那么我们认为预测误差的报警阈值相对大一些才比较合理。离散度 Filter 利用了这一特性，取连续 15 分钟的预测误差序列，分为首尾两个序列（e1,e2），如果两个序列的均值差大于 e1 序列方差的某个倍数，我们就认为该点可能是异常点。

阈值Filter

根据误差绝对值是否超过某个阈值过滤出可能的异常点。利用离散度 Filter 进行过滤时，报警阈值随着误差序列波动程度变大而变大，但是在输入数据比较小时，误差序列方差比较小，报警阈值也很小，容易出现误报。所以设计了根据误差绝对值进行过滤的阈值 Filter。阈值 Filter 设计了一个分段阈值函数 $y = f(x)$，对于实际值 x 和预测值 p，只有当 $|x - p| > f(x)$ 时报警。实际使用中，可以寻找一个对数函数替换分段阈值函数，更易于参数调优。

模型最终架构

每天定时抽取历史10天数据，经过预处理模块，去除异常数据，经过周期数据计算模块得到周期性数据。对当前时刻预测时，取60分钟的真实数据和周期性数据，经过实时预测模块，预测出当前传感器数值。将连续15分钟的预测值和真实值通过比较器，判断当前时刻是否异常。



> *参考来源：*

1. https://www.jianshu.com/p/6fb0408b3f54
2. https://www.otexts.org/fpp/7/5