关于为回归对齐元组

方晨光
清华大学 BNRist
fcg19@mails.tsinghua.edu.cn

宋少旭
BNRist, 清华大学
sxsong@tsinghua.edu.cn

梅一楠
清华大学 BNRist
myn18@mails.tsinghua.edu.cn

叶远
北京理工大学
yuan-ye@bit.edu.cn

王建民
BNRist, 清华大学
jimwang@tsinghua.edu.cn

抽象的

回归模型是通过多个变量学习的，例如，使用发动机扭矩和速度来预测其燃料消耗。在实践中，这些变量的值通常是单独收集的，例如，由车辆中的不同传感器收集，并且需要在学习之前首先在元组中对齐。不幸的是，由于网络延迟等各种问题，同时生成的值可能会记录不同的时间戳，从而使对齐变得困难。根据我们对一家汽车制造商的研究，发动机扭矩、速度和油耗值大多没有用相同的时间戳记录。通过简单地连接具有相同时间戳的变量值来对齐元组会导致学习回归模型的数据有限。为了处理时间戳变化，现有的时间序列匹配技术依赖于值和时间戳的相似性，不幸的是，回归中的变量很可能不存在这些相似性（发动机扭矩和速度值之间没有相似性）。从这个意义上说，我们建议桥接元组对齐和回归。我们没有将相似的值和时间戳对齐，而是将不同变量的值对齐在一个元组中，该元组 (i) 在短时间内记录，即时间约束，更重要的是 (ii) 与回归模型很好地吻合，称为模型约束. 我们的理论和技术贡献包括（1）制定具有时间和模型约束的元组对齐问题，（2）证明问题的 NP 完全性，（3）设计具有性能保证的近似算法，(4) 为算法提出有效的剪枝策略。对现实世界数据集的实验，包括上述由汽车制造商收集的发动机数据，表明我们的提议在对齐精度方面优于现有方法，并提高了回归精度。

CCS 概念

•信息系统→数据管理系统。

关键词

时间序列; 结盟; 回归模型

KDD '22，2022 年 8 月 14-18 日，美国华盛顿特区

978-1-4503-9385-0/22/08。.$15.00
https://doi.org/10.1145/3534678.3539373

ACM 参考格式：
Chenguang Fang、Shaoxux Song、Yinan Mei、Ye Yuan 和 Jianmin Wang。2022. 关于对齐元组以进行回归。在第 28 届 ACM SIGKDD 知识发现和数据挖掘会议 (KDD '22) 会议记录，2022 年 8 月 14 日至 18 日，美国华盛顿特区。ACM，美国纽约州纽约市，11 页。https://doi.org/10.1145/3534678.3539373

1 简介

回归模型是在多个变量上学习的，通常是单独收集的。例如，发动机扭矩和速度由车辆中的不同传感器收集以确定其油耗 [26]。同样，对于图 1 中的三个变量，通过监测家庭中的有功功率和强度来预测电压违规 [15]。为了学习回归模型，一个必要的预处理步骤是对齐一行中的多个变量的值，例如，三个值的对齐元组，有功功率，强度和电压，在图 1 中的时间 480 处由虚线连接。

不幸的是，由于传输或网络延迟以及硬件错误，可以使用不同的时间戳收集同时生成的不同变量的值，从而使元组对齐具有挑战性 [28]。例如，发动机扭矩、速度和燃料消耗值是从车辆的控制器局域网 (CAN) 总线同时收集的，同时由于传输延迟 [25]，被称为时间戳变化，在电子控制单元 (ECU) 中记录不同的时间戳。

简单地通过相等的时间戳对齐多个变量会导致学习回归模型的数据有限。根据某汽车制造商的调查，只有大约 5% 的数据是发动机扭矩、转速和油耗值具有相同的时间戳（实证研究见 4.3.1 节）。

现有的时间序列匹配技术，例如动态时间扭曲 (DTW) [4] 及其变体，不是相等的时间戳，而是主要基于相似值或接近时间戳的时间段来对齐变量。然而，回归中的变量很可能不存在值或时间戳的这种接近性，如图 1 中所示的示例。

另一种可能的替代方法是插值 [23]，即在每个时间戳处插入缺失值（可能同时考虑时间和交叉变量相关性）。显然，基于插值的方法不考虑时间戳变化，而是插值容易出错的新值。

示例 1。图 1 说明了学习回归模型以通过有功功率和强度预测电压的示例。在预处理中，我们首先需要将不同时间戳收集的三个值排成一行。

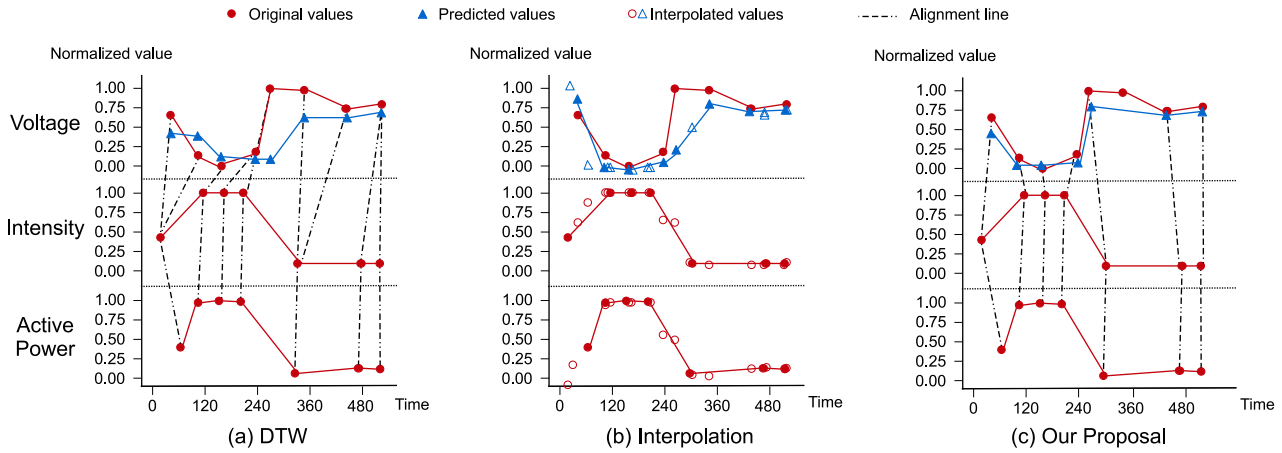

图 1：通过 (a) DTW 变化 [35]、(b) 样条插值 [23] 和 (c) 我们的提议对齐/插值三个现实世界变量的示例。学习回归模型以通过强度和有功功率预测电压。红点是原始值观察值。虚线表示行/元组中值的对齐方式。蓝色三角形是从对齐数据中学习的回归模型预测的值。空心点是插值算法在每个时间戳上计算的插值（因此不应用对齐）。

图 1(a) 显示了基于值和时间相似性的 DTW 变化 [35] 的结果。虽然有功功率和强度的归一化值有些相似，因此可以很好地对齐，但它们与电压值并不相似。从错误排列的有功功率、强度和电压值行中了解到，回归模型不准确，预测（蓝色三角形）与观察值（红点）存在偏差。

图 1(b) 显示了基于样条插值的结果。空心点是样条插值作为时间序列的值。基于插值的方法不考虑时间戳的对齐，而是简单地在每个时间戳处插值（可能不精确）而不进行观察。在不解决时间戳变化但引入新的容易出错的值的情况下，插值预测是不准确的。

在这项研究中，我们创造性地桥接了元组对齐和回归。我们建议在（迭代学习的）回归模型的指导下，将不同变量的值连续对齐，而不是依赖于 DTW 中的相似值。直观上，对齐的行/元组预计与回归模型的预测（在上一次迭代中学习）很好地重合。例如，在图 1(c) 中，对齐的电压观测值（红点）应与相应强度和有功功率的回归模型的预测值（蓝色三角形）一致。实际上，上述 DTW 中考虑的值的相似性可以解释为具有相似值的变量的特殊回归模型，因此包含在我们的解决方案中。在这个意义上说，

因此，我们建议将多个变量的值对齐在一个元组中，其中（1）在短时间内记录，即时间约束，以及（2）与（迭代学习的）回归模型很好地吻合，称为模型约束. 该策略迭代应用，即对齐良好的数据改进模型学习，而改进的回归模型再次建议下一次迭代

结盟。虽然在过去 20 年里已经研究了异相匹配（例如，DTW [4]、GTW [35、36]），但我们的提议是第一个结合模型约束（依赖性）和时间约束（关闭时间戳）用于元组对齐。

我们的主要贡献总结如下。

(1) 我们形式化了一个新问题，模型约束下的相似性对齐 (SAMC)，这是第一个同时利用时间和模型约束来对齐元组的问题。我们分析问题的 NP 完全性（定理 1）。

(2) 我们设计了一种具有理论性能保证的有效近似算法（命题 2）。

(3) 我们提出具有理论结果的创新策略和结构，以实现高效剪枝，从而显着降低时间成本（命题 3 和命题 7）。

最后，我们对真实世界的数据集进行了综合实验，包括汽车制造商收集的发动机数据。它表明我们的提议在对齐和回归性能方面优于最先进的方法。

该提案已部署在开源时间序列数据库 Apache IoTDB [1] 中。该代码位于 Apache IoTDB [2] 的 Github 存储库中。所有证明1个主要理论结果和实验代码见[3]。

2 元组对齐问题

图 2 概述了为学习回归模型对齐多个变量。考虑<变量)1个,)2个, ...,)<有架构吨8个(*8个,+8个）， 在哪里*8个表示识别变量何时被观察到的时间戳，并且 +8个是相应的值。下面介绍获取对齐实例的方法R有架构R(*1个,*2个,. . . ,*<,+1个,+2个,. . . ,+<). 如图1所示，我们不假设变量的采样率，即支持不规则的时间间隔。通过对属性的投影+1个,+2个,. . . ,+<

超过,我们获得了一组训练数据。回归模型米 可以在对齐的属性上进行训练。

1个限于篇幅，文献[3]中所有证明均在线完成。

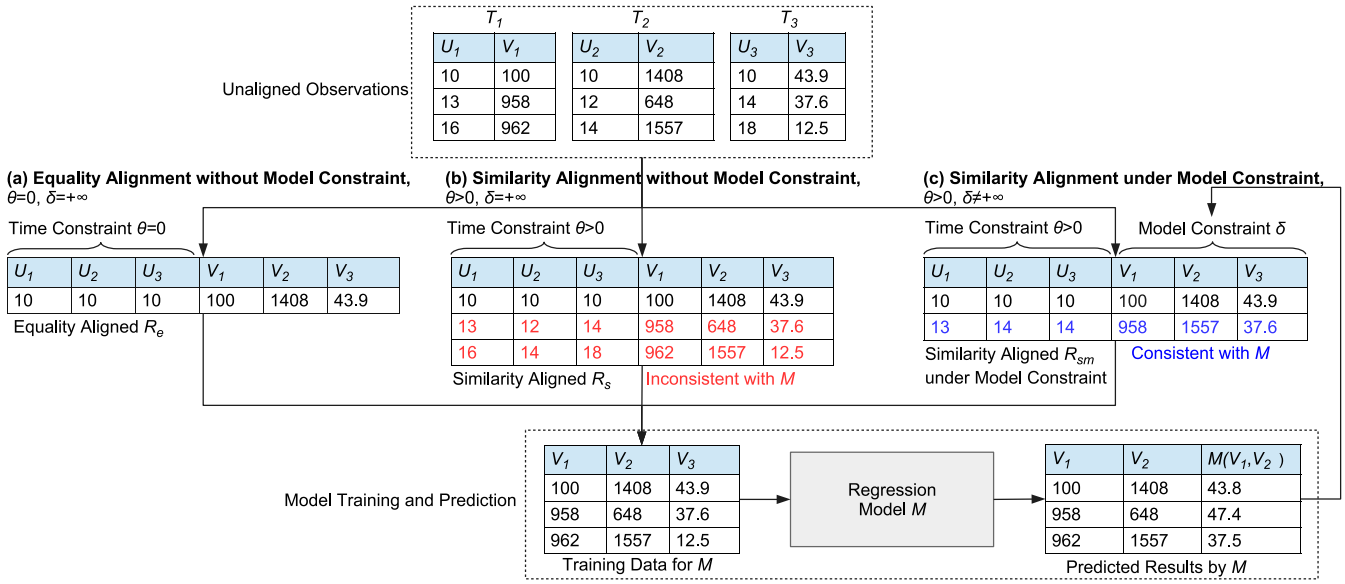

图 2：<= 的各种元组对齐方法的总体3个例如，包括（a）等同对齐（b）相似对齐和（c）模型约束下的相似对齐。等式对齐是具有时间约束的相似对齐的特例 \ =0,即，每个对齐的元组r∈R电子具有相同的时间戳ü1个,ü2个,ü3个的 ）1个,)2个,)3个. 相似性对齐确保时间戳ü1个,ü2个,ü3个在对齐的元组中r∈R秒有距离≤ \彼此。初步对齐的元组（通过相等或相似）作为训练数据来学习初始模型M.这个模型米然后被用作模型约束V1个,V2个,V3个对齐，即预测之间的距离米（一个 [+1个], 一个 [+2个])和一个 [+3个]对齐的元组r∈R小号不应大于阈值X，用Δ(R M）≤X。与模型约束的相似对齐排除了与米 (' 中的红色元组乙在 (b) 中）并找到更一致的对齐方式（蓝色的R小号在（c））。可以应用多次迭代，其中新学到的米作为另一轮对齐的模型约束。

直观地，对齐变量）1个,)2个,. . . ,)<，一个很自然的想法是基于相等的时间戳，即相等对齐。然而，由于传输延迟等问题，相应的结果非常有限。基于相似时间戳的对齐，即相似对齐，在很大程度上丰富了训练数据，但有许多对齐的元组与回归模型不一致。为了排除不一致的结果，在第 2.1 节中，我们建议采用回归模型米作为约束。

为了获得更多用于训练的对齐元组，对齐实例 R预计将在尺寸上最大化。我们表明，在模型约束下获得最佳相似性对齐是 NP-hard（定理 1），并设计了一个近似值（命题 2）。

2.1 相似性对齐和模型约束

在本节中，我们首先提出时间约束，它将时间戳限制在给定的阈值内。然后基于回归模型设计模型约束。结合它们，最终提出了模型约束下的相似性对齐问题。

定义 1。对齐元组 r 的对齐成本∈R wrt timestamp similarity 是在时间属性 U 上定义的1个,ü2个,. . . ,ü<.

$$\Theta(r) = \max_{甲、乙\in\{ü1个,U2个,...,ü米\}} |r[A] - r[B]|。 \quad (1)$$

直观地，对齐元组中的值r∈R预计将在短时间内记录下来。也就是说，他们的时间戳应该有距离不大于阈值\，称为时间限制，

$$|r[U8个] - r[U9]| \leqslant \backslash, 0 \leqslant 8, 9 \leqslant <。 \quad (2)$$

定义 2。我们说一个对齐的实例 R 满足时间限制 \，如果∀r∈R, Θ(r)≤\。

基于相似时间戳的对齐元组可能不准确。除了时间戳的约束ü1个,ü2个,. . . ,ü<，我们进一步引入对对齐值的约束V1个,V2个,. . . ,V<.直观地，回归模型米 (在前面的步骤中学到的）告诉值之间的关系，并可以指导随后的对齐，如图 2（c）所示。

定义 3。对于回归模型“在+上训练1个,+2个,. . . ,+<-1个预测 +<， 这对齐成本一个元组 r∈R 定义为

$$\Delta(r, M) = |M(一个 [+1个], 一个 [+2个],. . . , [+<-1个]) - 一个 [+<]|。 \quad (3)$$

价值对齐代价越大（R M）是，元组中的值越多r与学习到的回归模型不一致M.

定义 4。我们说一个对齐的实例 R 满足模型约束X，如果∀r∈R, 它有 Δ(r, M)≤X。

期望的对齐是获得满足模型约束的对齐元组，即与M.连同前-

说到时间限制，我们将对齐问题陈述如下。

问题 1 (模型约束下的相似性对齐）。给定变量 T1个,吨2个,. . . ,吨<, 时间约束 \ 和模型约束X， 的问题模型约束下的相似性对齐是为了

获得对齐的实例 R，也用 R 表示小号, 这样 (1) 每次属性 U1个,ü2个,. . . ,ü<是R的一个候选键，即T中的每个元组1个,吨2个,. . . ,吨<只能对齐一次，(2) 每个元组 r∈R 满足时间约束 Θ(r)≤ \和模型约束 Δ(r, M)≤
X，和 (3) 对齐元组的数量 |R| 被最大化。

从这个意义上说，初始化米在整个过程中，如图2所示，我们使用相等/相似对齐来获得一些初步的R用于学习回归模型米来指导后续的对齐。类似于EM算法的思想，都是对齐的实例R和模型米可以在迭代中交互改进。（有关实证研究，请参见第 4.2.1 节中的图 5。）

附录 A.2 给出了相等性和相似性对齐的定义，我们还表明它们都是模型约束下所提出的相似性对齐的特例。

2.2 硬度分析

模型约束下的相似性对齐问题在 <⩾3.

定理 1。给定变量 T1个,吨2个,. . . ,吨<, <⩾3、时间约束\，模型约束X和常量 ^，问题是 NP-complete 以确定是否存在具有 (1) 个候选键 U 的对齐实例 R1个,ü2个,. . . ,ü<, (2) 时间约束 Θ(r)≤ \和模型约束 Δ(r, M)≤X满足每个 r∈R, (3) |R|⩾^。

证明草图。为了显示问题的 NP 难度，我们从最大3维匹配问题 [17、19、20]。详细证明见[3]。

3 对齐算法

受定理 1 中硬度分析中最大维匹配问题 [17, 19] 的简化的启发，我们设计了一种近似对齐方法。一套R2个候选对齐元组的数量首先在3.1节中得到，参考时间和模型约束，但具有重叠的时间戳，即不满足问题1中的要求（1）。在3.2节中，然后使用局部搜索算法来消除具有重叠时间戳的对齐元组。值得注意的是，我们在命题 2 中展示了近似值的界限。最后，在第 3.3 节中，我们为该算法提出了有效的剪枝策略。[3] 中提供了所有理论结果的证明。

3.1 候选生成

给定变量吨1个,吨2个,. . . ,吨<，我们生成一个集合R2个满足时间约束和模型约束要求的候选对齐元组的数量X。而不是调查巨大的空间 吨1个×吨2个×・・・×吨<，在给定有序变量和时间戳距离的情况下，可以有效地生成候选对象≤\。

附录 A.3 中的算法 1 提供了候选生成算法的概述。吨1个,吨2个,. . . ,吨<，从数据库中读取，自然地按时间戳排序。参考定义 2 中的时间约束，对齐仅发生在具有时间戳距离的元组之间≤\。因此，在大小为 \ 的窗口内考虑候选对齐元组就足够了。受 merge join [37] 思想的启发，我们将窗口滑过吨1个,吨2个,. . . ,吨<生成所有可能对齐的元组 wrt 时间约束 \。

对于每个元组吨k∈吨:,该算法指定所有元组 吨9∈吨9在的窗口吨:大小为 \. 结合所有的吨9在前面的步骤中找到，它会在窗口中生成所有可能对齐的元组。同样，我们对每个变量执行这样的策略
):分别，从而生成满足时间约束的所有可能候选者。这一战略的完整性和正确性是显而易见的。对于每个可能的候选人A，假设

中的最小时间戳A出现在 *: <8= . 然后，A必须找到穿越时）: <8= . 在生成以下候选人之后
时间约束，模型约束X （采用定义4）来过滤与模型冲突的候选项。

在最坏的情况下，会有 |吨1个| ・ |吨2个| . . . |吨<|对齐的元组生成。设回归模型的预测代价为2("), 算法1的时间复杂度为O(|T1个| ・ |吨2个| ・・・ |吨<| ・2（ “））。

3.2 对齐搜索

中的一个元组吨1个,吨2个,. . . ,吨<可能出现在多个候选对齐的元组中R2个.参考问题1中的要求（1），我们需要找到一个子集R小号⊆R2个每个元组在哪里吨1个,吨2个,. . . ,吨<最多对齐一次。此外，问题 1 中的要求 (3) 寻找这样一个大小为 |R小号|尽可能大。

对齐搜索算法采用d-最优局部搜索策略[7, 18]，它是许多匹配问题（例如，k-set packing 问题）采用的启发式算法。思路是先初始化R乙<由一个子集R2个不重叠时间戳，然后通过交换更多元组逐渐扩展集合。图 3 展示了一个运行示例。

首先，对于任何候选对齐的元组r1个= (吨11,吨21,. . . ,吨<1),r2个 = （吨12,吨22,. . . ,吨<2)在Rc, 我们说r1个,r2个在某些变量上重叠，表示为r1个=r2个， 如果∃:,1个≤ : ≤ <,C:1=C:2，即一个元组在r1个,r2个. 在我们的对齐搜索算法中，我们首先初始化R乙<采用贪心策略，即我们不断添加 r2个∈R2个不与任何现有的重叠r小号∈R小号.

接下来，给定一个阈值d要搜索的子集大小，以及未选择的候选者 'D=2,IE， 'D=2= '2个\'乙<, 该算法搜索一个元组集合 '>直流电⊆ 'D=2尺寸？ = 2, 3, . . . ,d外部 '乙<
满足 (1) |'>直流电| = ?, (2) '中的任意两个元组>直流电不要相互重叠，即∀A1个， A2个∈'>直流电,A1个-A2个, 和 (3) 元组在 '>直流电最多重叠？ − 1 个元组（用 ' 表示8=） 在R乙<， IE， |'8=| < |'>直流电|, '8== {A8=|一个8=∈'乙<,∃A>直流电∈'>直流电,A8==A>直流电}. 如果三个条件都满足，因为'>直流电不会引入进一步的冲突，我们可以安全地将它换成R小号通过删除 '8=. R小号然后随着 (|'>直流电|−|'8=|) 大小。当不能交换更多的元组对时R小号，它达到局部最优并返回R小号.

对于每个 '>直流电与 |'>直流电| = ?,我们进行哈希表检查重叠，并用 $(d + |'乙<|) 时间成本。回想一下 |R小号| ≤G因为一个元组在吨1个,吨2个,. . . ,吨<
只能对齐一次。即检查的总时间成本
' 中的元组是否相互重叠>直流电并找到 '8=是 $(d+g) 。对于固定的 ?，至多|') (?2个|'>直流电将被检查。
当'的大小2个很大，我们可以假设d < |'2个|/2，因此我们有? (|')2个| ≤ (|')d,?≤d.此外，迭代（第 8 行）将
跑步d次，因此总时间成本为 $(d (|')d(d+g))。这
当达到局部最优时搜索将停止。同样，由于迭代要么增加 '乙<或停止，以及 '乙<

受限于G，迭代最多执行（第 6 行）G次。到
得出结论，给定的适度大丁，算法因此运行
在 $(gd|') d(d+g))时间。算法 2 给出了伪代码。

命题 2。给定<变量）1个,)2个,. . . ,)<,<⩾3、和阈值d子集大小的近似比b算法 1 和 2 在模型约束下的相似性对齐的边界是

$$b = \frac{<(<-1)_{2个}^{\overline{d+1个}} - <}{2(<-1)_{2个}^{\overline{d+1个}} - <}, \quad 如果d=3,5,7,。. . \qquad (4)$$

$$b = \frac{<(<-1)_{2个}^{d} - 2}{2(<-1)_{2个}^{d} - 2}, \quad 如果d=2,4,6,。. . \qquad (5)$$

回想一下参数d是我们考虑交换的集合的最大尺寸。直观上，更大的d有助于更准确的比对结果。在实践中，一个适度大的d (例如，2 或 3) 就足够了。理论上，根据命题 2，给定 < =3、d = 2，我们有b =2，即factor-2 approximation。根据经验，给定的对齐精度已经很高（F1-score > 0.9）d =2 在第 4.2.2 节图 6 的实验中。在这种情况下d =2、算法2在$(G2个|'2个|2个)时间。

3.3 优化剪枝

如第 3.2 节所示，我们提出的 Align- 的复杂性
换货搜索算法是$(gd $(|')^{2个}$ d(d + g)),它在聚运行
标称值 |'2个|和G。然而，在实践中，候选人的数量明显大于多个变量的长度，即|'2个|≫G。即使给了一个小门槛丁，的大小

$(|'2个|)^{d}$ 仍然失控。什么时候d =2、算法2仍然运行
在 $(G2个|'2个|2个)时间。由于未被选中的候选人有 'D=2= '2个\'乙<，和 '乙<≤G，它还表示 |'D=2|≫ |'乙<|。

在本节中，为了进一步优化和剪枝 Alignment Search 算法，我们从两个方面入手。(1) 根据3.2节，在每次迭代中，Alignment Search算法遍历所有未选中候选的子集'D=2大小为 ?, 导致

$(|')^{2个|}$时间成本。自 |'2个|≫G和 |'乙<| ≤G，是否有可能交易诗的子集 '乙<反而？(2) 而不是遍历'的所有子集乙<, 在遍历中找到它们的交换之前，是否有任何策略可以修剪一些子集？

3.3.1 遍历'乙<代替 'D=2.对于第一个问题，鉴于当前选择的候选人 '乙<, 未被选中的候选人 'D=2= '2个\ '乙<和最佳参数丁，因此我们遍历子集 '8=
的 '乙<尺寸？ = 1, 2, . . . ,d -1 而不是。该算法然后搜索一组 '>直流电⊆ 'D=2满足 (1) |'>直流电|⩾？ +1，以及第 3.2 节中的条件 (2) 和 (3)。图 3 中说明了一个示例。但是，问题是如何找到可能的 '>直流电
无需再次遍历所有可能的子集 'D=2.

为了解决上述问题，我们发现，如果 '>直流电可以换成 '8=, 除了条件 (1), (2) 和 (3), '>直流电与 ' 中的至少一个元组重叠8=, 和每个元组 '8=与 ' 中的至少一个元组重叠>直流电(下面将在命题 3) 中证明。可以进一步利用这一观察来防止耗时的遍历。

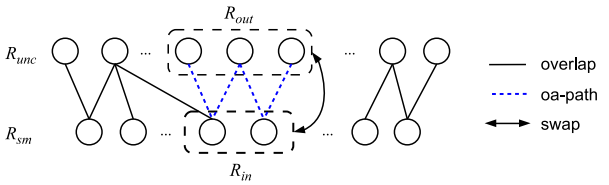

图 3：对齐搜索算法示例。'乙<
和 'D=2分别是当前选择和未选择的候选人。'8=是 ' 的一个子集乙<尺寸？ =2个和 '>直流电是 ' 的一个子集D=2大小 > ?。边表示两个元组（顶点）之间的重叠关系。点缀的蓝色

行形成一个 oa 路径，其中元组在 '8=和 '>直流电交替发生。带箭头的线是交换的尝试。

命题 3。鉴于'D=2, '乙<而如果 '>直流电⊆ 'D=2可以换成 '8=⊆ '乙<有|'8=| = ?,它还应满足以下条件：

(4)∀A>直流电∈'>直流电,∃A8=∈'8=， A>直流电≂A8=.
(5)∀A8=∈'8=,∃A>直流电∈'>直流电,A>直流电≂A8=.

附录 A.3 中的算法 3 概述了改进后的算法。有了命题 3 中的条件（4）和（5），我们可以搜索 A>直流电与 ' 中的元组重叠8=, 无需遍历 ' 的所有子集D=2.重叠元组的搜索可以预先

由哈希映射构建。自 |'乙<| ≤G，搜索的时间复杂度
可能的交换可以从 $(d|') $(d(d+g))
算法 2 的第 8-15 行，对 $(dG) $(d-1个|'2个|G)，在算法的第 8-14 行中-
算式 3. 对于小丁，我们有 $(d|') $(d(d + g))≈ $(d|'2个|d(G))
和$(dG $(d-1个|'2个|G) ≈ $(dgd|'2个|).自 |'2个|≫G，算法 3 与算法 2 相比，大大降低了时间复杂度。

3.3.2 使用 OA 路径和扩展时间约束进行修剪。对于第二个问题，受二部图匹配的交替路径方法的启发，我们在我们的问题中发现了类似的交替路径，即重叠交替路径（定义 5 中的 oa 路径）。那是，A8=∈'8=和A>直流电∈'>直流电或者出现在路径中，如图 3 所示。通过引入 oa-path，我们 (1) 将定义 2 中的时间约束扩展到引理 4 中 oa-path（即一组元组）上的扩展时间约束， (2) 证明对于任何两个元组A1个， A2个∈'8=, 存在一个长度为 oa 的路径≤ 2？ – 在引理 5 和 6 中它们之间有 1 个。

结合这两个发现，命题 7 证明了 '8=在 (2? −1)\ 内。因此，我们采用命题 7 进行修剪 '8=在算法 3 中，这显着降低了 Alignment Search 算法的时间成本。

定义 5. (重叠交替路径，oa-path)。鉴于'8=
和 '>直流电,长度为 的重叠交替路径（oa 路径）；被定义为 (A1个， A2个,. . . ， A;), 有 (1)A8个≂A8+1个, 和 (2)A8个和A8+1个属于 '8=和 '>直流电(或者 '>直流电和 '8=）， 分别，∀8个∈{1, 2, 。. . , ; - 1}。

简而言之，oa-path 是一个元组列表，其中连续的元组彼此重叠，并且元组来自'8=和 '>直流电交替发生。图 3 显示了一个示例，其中顶点是元组 A∈'8=∪'>直流电.它们之间有一条边，如果A8个≂A9.蓝色边缘形成 oa 路径。虽然时间约束最初是在定义 2 中的元组对上定义的，但我们将其扩展为限制 oa 路径上开始和结束顶点的时间差以进行修剪。

引理 4. (oa 路径上的扩展时间限制）。对于 oa 路径 （A1个， A2个,. . . ， A;),的任何时间戳之间的差异A1个和A;
小于;\，即∀: ∈ {1, 2, 。. . ,<}, |吨:1[*:] -C:;[*:]| ≤;\.

为了在引理 4 中应用扩展时间约束，我们识别图中的 oa 路径 wrt 元组重叠≈.

引理 5。对于任意两个元组A0， A1个∈'8=∪'>直流电,如果A0和A1个由路径连接，连接它们的路径必须是 oa-path。

引理 5 指出任何连接元组的路径在 '8=和 '>直流电必须是oa-path，保证利用oapath进行剪枝的正确性。我们进一步调查它的长度。

引理 6。鉴于'8=, '>直流电和 |'8=| = ?,假设算法-rithm 已找到所有可能与 |' 的交换 ‘ 8=| < ,对于 |'8=| = ?, 如果 '8=可以被交换 '>直流电,∀A0， A1个∈'8=, 必须存在一个 oa-path %>0= (A0,. . . ， A1个)有长度≤2？ - 1。

引理 6 表明任意两个元组A0， A1个∈'8=由 oa 路径连接，具有有限的长度。结合引理 4 关于 oa 路径上的扩展时间约束，我们最终将扩展时间约束应用于 '8=在提案 7 中。

提案 7。鉴于'8=和 |'8=| = ?,如果 '8=可以通过 ' 交换>直流电⊆ 'D=2, 我们有 |C:0[*:] -C:1[*:]| ≤ (2？ − 1)\,∀A0， A1个∈'8=, :∈{1, 2, 。. . ,<},即，时间戳差异时间戳任何A0和A1个小于 (2？ − 1)\.

算法 3 提出了经过修剪的优化对齐搜索。根据命题 7，我们修剪不必要的 '8=通过简单地检查'中每个元组对的时间差8=(第 10 行）。时间 co(st o

从 $(dG)f 进一步搜索可能的 swap，在第 3.3 节到 $(dGd−1个d<).再次，正如第 3.2 节末尾所讨论的，一个中等大小的d =2 就足够了，导致时间复杂度 $(克<)算法 3。如第 4.2.2 节所示，在实践中所提出的策略显着提高了时间性能。

4个实验

在本节中，我们通过与现实世界数据集上的最先进方法进行比较来评估我们的提议。实验设置在附录 A.4 中。

4.1 与现有方法的比较

对于每个数据集，我们将我们的 SAMC 与附录 A.4.3 中列出的现有对齐方法与附录 A.4.2 中介绍的不同回归模型进行比较。表 1 报告了比对 F1 分数和模型 RMSE。为了证明算法的差异是显着的，实验进行了 5 次。基于学生配对 t 检验 [29] 在 95% 显着性水平（即 ? < 0.05）的最佳性能在表 1 中以粗体显示。对于对齐精度，基线的结果不会随模型而变化，因为它们的对齐仅利用不考虑回归模型的相似性。如图所示，SAMC 在对齐 F1 分数上优于五个数据集的所有基线。结果并不令人惊讶，因为如第 1 节所述，变量的相似性可以通过特殊的回归模型捕获

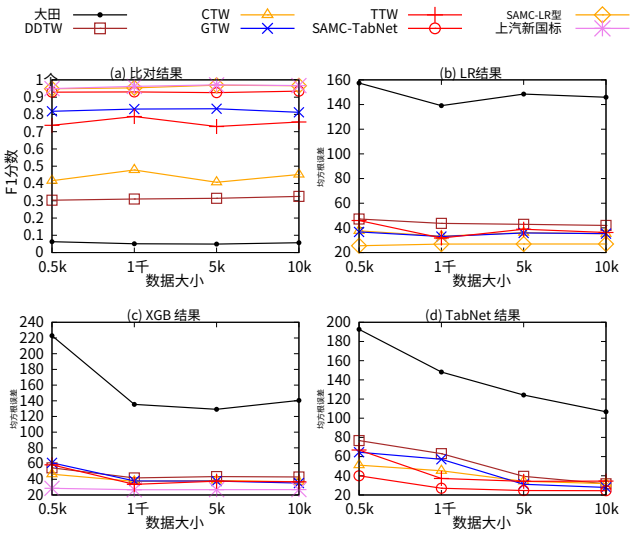

图 4：将我们的 SAMC 与 Fuel 数据集上的现有对齐方法在 (a) 对齐结果的 F1 分数，(b) LR 模型的 RMSE 损失，(c) XGB 模型的 RMSE 损失和 (d) RMSE 上进行比较TabNet 模型的损失

在具有相似值的变量上。相反，在现有的比对方法中没有考虑对不同变量的更一般的回归。验证了我们研究的直觉，回归模型确实指导了多变量的对齐。

表 1 还报告了基于测试数据的模型 RMSE。通常，更准确对齐的数据会带来更好的模型学习性能。因此，我们的提议实现了最低的模型 RMSE。与表 1 中在插值时间序列上训练的模型相比，我们的建议也显示出具有竞争力的性能。结果并不令人惊讶，因为插值不考虑时间戳变化，而是插值容易出错的新值，如简介中图 1(b) 中所示的示例。

图 4 报告了不同数据大小（即序列长度）的比对和模型准确度。结果在各种数据大小上都是稳定的，并且通常与表 1 相似。

4.2 拟议技术的评估

该实验评估了应用迭代、优化和修剪策略的有效性。由于空间有限，我们在 [3] 中的所有数据集上报告了类似的结果。

4.2.1 改变迭代次数。如图 2 所示和第 2.1 节末尾讨论的那样，通过相等对齐的实例R4个或相似性R乙可以服务于初始化得到初步的回归模型M.然后将其用于模型约束下的相似性对齐，称为米由初始化R4个

和米由初始化R乙,分别。新的预测模型米 ‘
可以在模型约束下的相似性对齐的另一迭代中进一步用于模型约束米 ‘.

图 5 显示了不同迭代次数下的结果。使用相似性对齐也就不足为奇了R乙初始化米 具有更好的性能。更多的迭代会带来更好的性能。

表 1：与五个数据集上的现有方法的比较。基于 95% 显着性水平的学生配对 t 检验的最佳表现（即？ <0.05)大胆。

数据集	模型	比对 F1 分数		测试数据的回归 RMSE		
		DTW DDTW CTW GTW TTW SAMC		DTW DDTW CTW GTW TTW SAMC		插补
房子	左轮	0.222	0.024 0.007 0.675 0.0490.929 0.024	17.937	5.223 3.8253.359 3.390 3.222	3.690
	XGB	0.222	0.007 0.675 0.0490.932 0.024 0.007	29.367	8.594 5.464 5.015 4.8844.337	4.659
	标签网	0.222	0.675 0.0490.928	27.465	7.778 5.2324.5195.1904.152	4.714
遥测	左轮	0.547	0.301 0.580 0.148 0.5020.829 0.301	0.255	0.253 0.257 0.259 0.2560.221 0.171	0.283
	XGB	0.547	0.580 0.148 0.5020.889 0.301 0.580	0.169	0.170 0.1710.165 0.157 0.231 0.248	0.240
	标签网	0.547	0.148 0.5020.855	0.240	0.2240.2670.212	0.263
水	左轮	0.140	0.629 0.092 0.061 0.0020.948 0.629	0.239	0.210 0.204 0.195 1.0810.057 0.037	0.279
	XGB	0.140	0.092 0.061 0.0020.964 0.629 0.092	0.037	0.036 0.036 0.1730.031 0.072 0.085	0.051
	标签网	0.140	0.061 0.0020.939	0.106	0.112 1.4000.038	0.059
空气质量	左轮	0.190	0.056 0.048 0.037 0.0010.693 0.056	50.571	47.215 43.133 50.146 49.07235.670	38.056
	XGB	0.190	0.048 0.037 0.0010.731 0.056 0.048	44.748	43.907 40.608 39.231 54.10537.281	38.409
	标签网	0.190	0.037 0.0010.727	55.836	78.012 66.054 67.821 83.83650.045	54.742
燃料	左轮	0.057	0.325 0.452 0.812 0.7880.966 0.325	139.077	37.881 31.075 30.926 31.80125.132	71.647
	XGB	0.057	0.452 0.812 0.7880.967 0.325 0.452	151.752	40.515 33.410 32.260 33.55824.636	181.011
	标签网	0.057	0.812 0.7880.935	154.779	38.814 33.137 31.673 37.21524.585	54.154

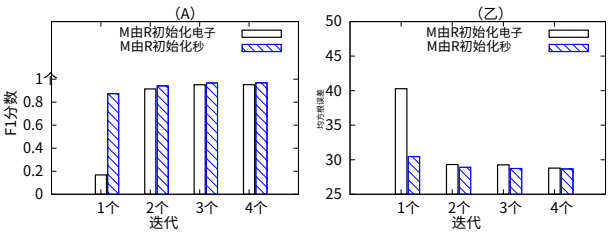

图5：时间约束下相似性对齐多次迭代的对齐精度和模型预测性能120和模型约束X =35关于燃料

结果不会通过进行甚至更多的迭代（即收敛）而进一步改善。也就是说，大多数对齐的元组符合回归并具有相似的时间戳，即使其中一些不符合基本事实。

4.2.2 评估优化和修剪策略。表 2 报告了具有不同优化和修剪策略的对齐搜索算法的运行时间。我们对不同长度的数据集 House 进行实验G的变量。AS 是原始的对齐搜索算法。AS+Opt 代表 3.3.1 节介绍的 Alignment Search with optimization strategy。AS+Opt+Pruning 表示 3.3.2 节中同时采用优化和剪枝策略的 Alignment Search。

表2表明优化策略降低了原算法的时间成本。修剪策略不执行时d =2，即'中只有一个元组8=. 尽管如此，对于d >2、剪枝策略显着提高了效率（减少85%的时间|'2个|≈104个).

图 6 通过不同的方式说明了我们提议的对齐性能丁，我们将考虑交换的集合的最大尺寸。结果验证了适度大d正如第 3.2 节末尾所讨论的那样就足够了，因为

表 2：Alignment Search 算法使用不同优化和修剪策略对具有 \ = 的 House 数据集的运行时间（s）80和X =8个

d	G	|'2个|	作为	AS+选择	AS+Opt+剪枝
2个	100	986	1.56	0.03	0.03
3个	100	986	525.93	0.17	0.06
2个	1000	9686	112.02	0.32	0.32
3	1000	9686	> 1 小时	15.61	2.17

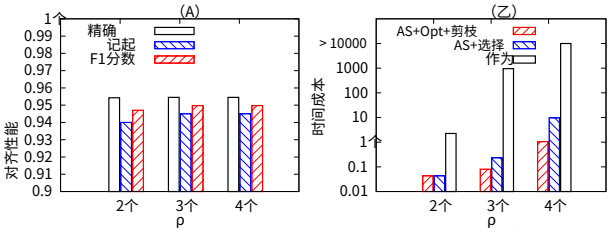

图 6：(a) 对齐性能和 (b) SAMC 的时间成本通过改变d使用 \ = 在 House 数据集上80和X =8个

建议与d >2 仅显示出轻微的改善，同时遭受效率损失，如表 2 所示。

4.3 自动约束确定

在本节中，我们首先说明时间约束和模型约束如何X影响对齐和模型学习性能。然后，我们介绍适当的自动确定 \和X在实践中没有基本事实。

4.3.1 变时约束 (对于每个固定X，）毫不奇怪，增加 \ 会导致图 7(b) 中对齐的元组更多。更多时间戳距离更大的元组对可以满足

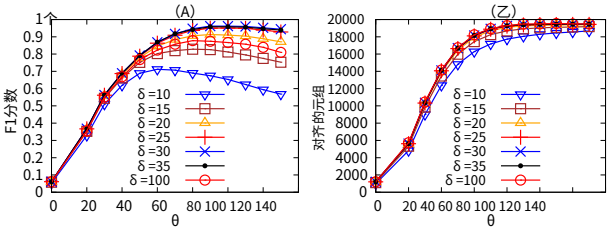

图 7：（a）对齐精度和（b）在各种时间约束和模型约束下对齐的元组数量X在燃料数据集上

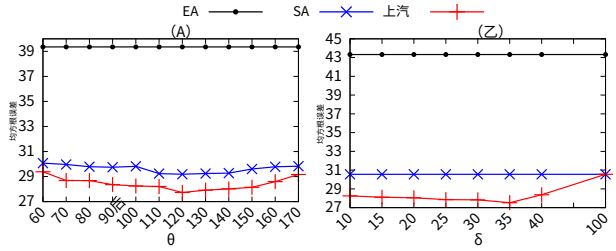

图 8：（a）各种时间约束下的模型精度 \用固定的X =35,(b) 各种模型约束X 用固定的\=100使用 XGB 处理燃料数据

具有较大距离阈值的时间约束的要求\。图 7(a) 说明了通过增加 \，F1-score 先增加然后下降。对于较小的 \（极端情况是 \ = 0 的等式对齐），对齐的元组数量有限，导致对齐召回率低。另一方面，对于一个大的\，太远的元组没有出现在同一时期被错误地对齐，精度低。因此，预计会有一个适度大的时间约束（参见第 4.3 节关于确定适当的 \）。

图 8(a) 展示了 SA 和 SAMC 在各种时间约束下的模型精度结果。具有固定 \ = 0 的 EA 结果也被报告为基线。模型性能通常类似于图 7(a) 中的对齐精度。\ = 0 的 EA 具有低对齐 F1 分数和高模型预测 RMSE。时间限制 \ 具有更好的对齐 F1-score，例如，在 [90, 130] 范围内X =图 7(a) 中的 35，在图 8(a) 中也显示了较低的模型预测 RMSE。

4.3.2 改变模型约束X。对于一个小X，即，严格的模型约束，更少的元组可以满足如图7(b)所示的要求。又是一个小X导致有限数量的对齐元组具有低召回率。对于大X （极端情况是没有模型约束的相似比对X = +∞),对齐的元组不符合模型米将被考虑，导致对准精度低。如图 7(a) 所示，在 Fuel 数据集上，X在 [25, 35] 范围内显示最佳对齐精度（另请参阅确定X在第 4.3 节）。

同样，图 8(b) 报告了等同性和相似性对齐X = +∞作为基线。同样，SAMC 在模型约束下X =35，在图 7（a）中显示出更好的对准精度，导致图 8（b）中模型 RMSE 较低。我们观察到一个小 X导致模型不准确。这并不奇怪，因为一个小X 忽略了太多的对齐方式，因此对于学习来说是不够的。

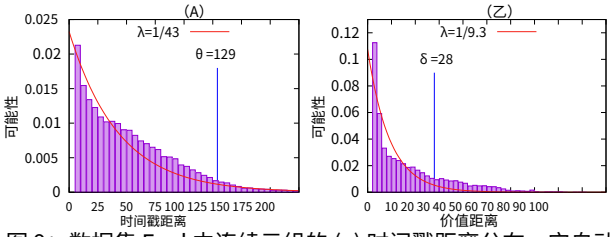

图 9：数据集 Fuel 中连续元组的 (a) 时间戳距离分布，它自动确定时间约束 \ =129、(b) 对齐候选者与其模型预测之间的值距离，这会自动确定模型约束X =28

4.3.3 确定时间约束\。为了确定对齐元组之间时间戳距离的适当阈值，我们观察输入数据集中连续元组的时间戳距离分布。将连续的元组解释为时间戳上的最近邻居，我们可以使用泊松过程来近似最近邻居的出现 [27]。也就是说，连续元组的时间戳距离用指数分布建模，表示为dü~Exp(_),具有概率密度函数 (PDF)f (dü| _) = _ exp(−_dü). 参数-

eter _ 由 _ 估计≈1个 ~~MTI~~,在哪里MTI是所有的中位数时间戳距离。图 9(a) 显示了实验中使用的数据集在估计参数 _ = 上的分布1个 ~~43~~.

一旦连续元组的时间戳距离被指数分布建模，我们确定可以在置信水平 0.95 [33] 下对齐的时间戳距离的阈值，即找到具有累积分布函数（CDF）的 \ F(\ | _) =0.95。对于图 9(a) 中的分布，我们确定 \=129. 如图 7(a) 和 8(a) 所示，这样的时间限制会带来更好的对齐精度和模型预测性能。

4.3.4 确定模型约束X。同样，找到一个合适的模型约束阈值X，我们观察对齐的候选元组之间的值距离r和预测米,IE， Δ(R M）在公式 3 中。同样，我们可以通过指数分布 [27] 对最近邻居的距离进行建模，dv~Exp(_),具有概率密度函数 (PDF)f (dv| _) = _exp(−_dv). 图 9(b) 显示了在数据集 Fuel 中观察到的分布

估计参数 _ =1个 ~~9.3~~. 置信度为 0.95 [33]，我们决定X=28. 如图7(a)和8(b)所示，对应的对齐精度高，而模型RMSE低。此外，如果因变量无法预测，即回归关系不成立，则模型约束下的相似性对齐问题（第2.1节）等同于相似性对齐问题，即前面提到的简单方法只比较时间戳。在这种情况下，我们可以设置X到 +∞该提案仍然有效。由于参数 \和X通过观察数据分布来确定，该提案可以在一定程度上容忍噪声。

5 相关工作

除了在第 4.1 节（并在附录 A.4.3 中介绍）的实验中比较的典型方法外，类似的方法也可用于对齐时间序列。ERP [9] 和 EDR [10]

利用动态规划的编辑距离来评估时间序列的相似性。LCSS [32] 计算有界时间窗口中序列的相似性，并考虑值的相似性。尽管这些方法中的大多数不需要有界匹配窗口，但它们高度依赖时间序列之间的相似性。不幸的是，相似值和时间戳的这种近似值很可能在回归变量中不存在，例如，在图 1 中。出于同样的原因，方法 [12、16] 也强调时间序列集群之间的相似性不申请。

除了翘曲，Dignös 等人。[13] 研究时间对齐，并提出了一种关系代数，用于在 DBMS 中提供顺序语义。然而，它不能应用于我们的情况，因为（1）该方法处理持续时间，例如，在酒店逗留的时间，而不是离散时间，例如，从传感器收集的即时数据，以及（2）它侧重于两个时间关系之间的查询操作，而不是多个时间序列。

与指示隐藏的时间依赖性的时间滞后发现问题 [31] 不同，我们研究由意外时间戳变化（例如传输延迟）引起的未对齐变量。这种延迟不能建模为时间依赖性。要为下游任务学习高质量的回归模型，对变量进行对齐是可能且必要的。

多个对齐变量之间的回归进一步指导后续对齐迭代中的对齐，类似于时间序列数据清理的时间约束 [30]。处理对齐实例的不一致R对于模型，我们使用模型米作为参考数据。这个想法类似于用主数据修复数据[14]。而范等人。[14]建议修改数据值以消除违规，我们的研究发现对齐变量符合M.

六，结论

在本文中，为了在相应变量上对齐用于学习回归模型的元组，我们提出了一种新的模型约束下的相似性对齐（SAMC）。虽然时间约束确保对齐的元组中的值在时间上与变量中的相似时间戳接近，但模型约束进一步排除了与（迭代学习的）回归模型不一致的对齐。我们证明了对齐问题的NP完全性，并提出了一种具有理论性能保证的近似算法。提出了新的修剪策略以提高效率。对真实数据集的实验表明，根据我们的建议，对齐元组和学习回归模型在迭代中相互改进。改进验证了对齐元组学习回归模型的必要性。

致谢

这项工作得到了国家自然科学基金（62072265、62021002）、国家重点研发计划（2021YFB3300500、2019YFB1705301、2019YFB1707001）、北京国家信息科学技术研究中心（BNR2022RC01011）和阿里巴巴的部分支持通过阿里巴巴创新研究（AIR）计划进行分组。宋绍旭（https://sxsong.github.io/）为通讯作者。

参考

[1] http://iotdb.apache.org。
[2] https://github.com/apache/iotdb/tree/research/alignment-model。
[3] https://github.com/fangfcg/SAMC。
[4] 动态时间规整 (DTW)。在 SZ Li 和 AK Jain 的编辑中，生物识别百科全书，第 231 页。施普林格美国，2009 年。
[5] Scikit-learn，https://scikit-learn.org/stable/。
[6] SO Arık 和 T. Pfister。Tabnet：细心的可解释表格学习。在 美国汽车协会，第 6679-6687 页。美国人工智能出版社，2021 年。
[7] EM Arkin 和 R. Hassin。在本地搜索加权k-设置包装。数学。歌剧。水库，23(3):640–648, 1998。
[8] PC Arocena、B. Glavic、G. Mecca、RJ Miller、P. Papotti 和 D. Santoro。搞乱 BART：评估数据清理算法的错误生成。过程。VLDB 基金会，9(2):36–47，2015。
[9] L. Chen 和 RT Ng。关于 lp 规范与编辑距离的结合。在超低密度数据库， 第 792-803 页。摩根考夫曼，2004 年。
[10] L. Chen、MT Özsu 和 V. Oria。对移动物体轨迹进行稳健且快速的相似性搜索。在SIGMOD大会，第 491-502 页。美国计算机学会，2005 年。
[11] T. Chen 和 C. Guestrin。Xgboost：一个可扩展的树提升系统。在, 第 785-794 页。美国计算机学会，2016 年。
[12] D. Chudova、S. Gaffney、E. Mjolsness 和 P. Smyth。用于曲线聚类的平移不变混合模型。在,第 79-88 页。美国计算机学会，2003 年。
[13] A. Dignös、MH Böhlen 和 J. Gamper。时间对齐。在SIGMOD大会，第 433-444 页。美国计算机学会，2012 年。
[14] W. Fan、J. Li、S. Ma、N. Tang 和 W. Yu。使用编辑规则和主数据进行某些修复。过程。VLDB 基金会，3(1):173–184, 2010。
[15] A. Furlani Bastos、S. Santoso、VK Krishnan 和 Y. Zhang。基于机器学习的配电网电压预测和传感器分配。技术报告，国家可再生能源实验室。(NREL)，2020 年。
[16] S. 加夫尼和 P. 史密斯。联合概率曲线聚类和对齐。在 尼普斯,第 473–480 页，2004 年。
[17] P. Haxell 和 L. Narins。三方 3-图中匹配的稳定性定理。 组合学、概率与计算，27(5):774–793, 2018。
[18] CAJ Hurkens 和 A. Schrijver。关于集合系统的大小，其中每个 t 都有一个 sdr，并将启发式算法应用于包装问题的最坏情况比率。暹罗 J. 谨慎。数学。，2(1):68–72, 1989。
[19] V. Kann。最大有界 3 维匹配是 MAX snp-complete。信息。过程。利特，37(1):27–35, 1991。
[20] RM 卡普。组合问题之间的可还原性。在计算机计算的复杂性，第 85-103 页。全会出版社，纽约，1972 年。
[21] EJ Keogh 和 MJ Pazzani。导数动态时间扭曲。在数据管理,第 1-11 页。暹罗，2001 年。
[22] S. Khorram、MG McInnis 和 EM Provost。可训练的时间扭曲：在连续时间域中对齐时间序列。在ICASSP，第 3502–3506 页，2019 年。
[23] M. Lepot, J.-B. 奥宾和 FH 克莱门斯。时间序列插值：对现有方法、它们的性能标准和不确定性评估的介绍性概述。水，9(10):796, 2017。
[24] HR Lourenço、OC Martin 和 T. Stützle。迭代局部搜索。在元启发式手册，第 320-353 页。施普林格，2003 年。
[25] SE Marx、JD Luck、SK Pitla 和 RM Hoy。比较控制器局域网（CAN）总线数据采集的各种硬件/软件解决方案和转换方法。农业计算机和电子产品，128:141–148, 2016。
[26] Y. Min, Y. Xiaogang, J. Shengjie. 通过参数匹配优化车载混凝土泵的能量。能源程序，88:574–580, 2016。
[27] Y. Noh、FC Park 和 DD Lee。自适应最近邻分类的扩散决策。在尼普斯,第 1934–1942 页，2012 年。
[28] P. Philipp 和 S. Altmannshofer。具有不同步时钟的网络控制系统的新移动水平估计器方法的实验验证。在行政协调会，第 4939-4944 页。IEEE，2012 年。
[29] MD Smucker、J. Allan 和 B. Carterette。信息检索评价统计显着性检验的比较。在中科院，第 623–632 页，2007 年。
[30] S. Song、A. Zhang、J. Wang 和 PS Yu。屏幕：速度限制下的流数据清理。在 SIGMOD大会，第 827-841 页。美国计算机学会，2015 年。
[31] L. Tang、T. Li 和 L. Shwartz。发现时间依赖性的滞后间隔。在,第 633-641 页。美国计算机学会，2012 年。
[32] M. Vlachos、D. Gunopulos 和 G. Kollios。发现相似的多维轨迹。在集成电路设计器，第 673-684 页。IEEE 计算机协会，2002 年。
[33] JH扎尔。生物统计分析。培生教育印度公司，1999 年。
[34] F. Zhou 和 FD la Torre。用于对齐人类行为的规范时间扭曲。在尼普斯,第 2286-2294 页。柯伦联合公司，2009 年。
[35] F. Zhou 和 FD la Torre。用于人体运动多模式对齐的广义时间扭曲。在简历，第 1282-1289 页。IEEE 计算机协会，2012 年。
[36] F. Zhou 和 FD la Torre。广义规范时间扭曲。IEEE 跨。模式肛门。马赫。智力.，38(2):279–294, 2016。
[37] 周杰. 排序合并连接。在数据库系统百科全书（第 2 版）。施普林格，2018 年。

A 补充材料

A.1 符号

表 3 总结了本文中一些常用的符号。

表 3：符号

象征	描述
) 1个,)2个,. . . ,)<	<变量
G	由 EA、SA 和 SAMC 对齐的时间和模型约束元
\,X	组的每个变量阈值的最小值数
'4个, '乙, '乙<	
'2个, 'D=2	' 的所有候选人和未选择的候选人子集
'8=, '>直流电	乙<和 'D=2要交换的搜索子集大小的阈
d	值

A.2 特殊情况

问题 1 定义了模型约束下的相似性对齐 (SAMC)。除了定义 2 中的时间约束限制相似的时间戳外，定义 4 中的模型约束确保 Δ(R M）≤X每个r∈R。

正如我们在第 2 节中介绍的，相等对齐是对齐元组的自然想法，它基于相等的时间戳。相似性对齐采用时间约束，进一步丰富训练数据。事实上，我们将证明它们都是所提出问题的特例。根据模型约束下的相似性对齐（SAMC），我们将给出等式对齐（EA）问题和相似性对齐（SA）问题的定义。图 2 说明了它们的关系和使用这些方法的管道的示例。

相等对齐 (EA) 问题在对齐的元组中强制执行相同的时间戳，这是一种直观的对齐策略。相等对齐的定义如下。

问题 2 (平等对齐）。给定变量）1个,)2个,. . . ,)<， 这平等对齐问题是获得一个对齐的实例R，即R4个,这样 (1) 每次属性 U我是 R 的一个候选键，即) 中的每个元组1个,)2个,. . . ,)<只能对齐一次，(2) 每个元组 r∈R 具有相等的时间戳 r[U1个] =r[U2个] = • • • =r[U<]，和（3）对齐元组的数量|R| 被最大化。

解决相等对齐问题很简单。通过执行连接操作R = T1个⊲⊳吨2个⊲⊳ 。. . ⊲⊳吨<在ü1个=ü2个= • • • =ü米, 它自然地最大化对齐元组的数量 |R|。

相似性对齐 (SA) 问题仅使用时间约束来过滤对齐的元组。

问题 3 (相似对齐）。给定变量 T1个,吨2个,. . . ,吨<和时间限制\，相似比对问题是获得对齐的实例 R，即 '乙,这样 (1) 每次属性 U1个,ü2个,. . . ,ü<是R的一个候选键，即T中的每个元组1个,吨2个,. . . ,吨<可以对齐一次，（2）每个元组r∈R满足时间约束 Θ(r)≤\,和 (3) 对齐元组的数量 |R| 被最大化。

而不是平等对齐中的时间戳相等，相似对齐中的时间约束，定义为时间约束 Θ(r)≤ \在定义 2 中，考虑相似的时间戳。它确保在每个对齐的元组r∈,任意两个时间戳之间的距离不大于\。为了解决相似性对齐问题，我们还应用了我们提出的候选生成（算法 1）和对齐搜索（算法 2），忽略了模型约束部分，以及算法 3 中的修剪策略。

提案 8。鉴于X = +∞,即，没有模型约束，Problem 1等同于相似比对。和...一起X =0,这确实是平等对齐。

A.3 算法

算法 1 提供了候选生成的概述。让 吨1个,吨2个,. . . ,吨<按时间戳自然排序。该算法首先指定所有的元组吨9∈吨9在的窗口吨:大小 \ (第 3 行）。模型约束X用于过滤与模型冲突的候选者”（第 6 行）。如前所述，在整个过程中，等同/相似对齐用于初始化初步R用于学习回归模型米指导后续对齐（如图 2 所示）。

算法 2 给出了对齐搜索算法。第 3 行初始化R乙<采用贪心策略，即我们不断添加 r2个∈R2个不与任何现有的重叠r小号∈R小号. 第 10-12 行根据条件过滤可能的交换，并生成符合模型的时间窗口中所有可能的对齐元组M.第 14 行最终进行交换。它遵循以下框架d-最优局部搜索算法：-SP 问题 [18, 24]，具有有界近似比（命题 2）。

算法 3 概述了具有修剪策略的对齐搜索算法。第 9 行由命题 3 证明，遍历 '乙<
代替 'D=2.第 10 行使用具有扩展时间约束的 oa-path 来进一步过滤候选者，这在命题 7 中得到了证明。

算法 1：候选生成（T_List, \, M,X）

```
输入：变量列表T_列表 = ()1个,)2个,. . . ,)<) ， 时间
      约束\，模型约束（米,X） 输出：对
齐元组的候选者R2个
1个R2个←∅;
2个为了每个 ）:∈ )_;8BC做
3个   R:← {(吨1个,吨2个,. . . ,吨<)|∀1个≤8个≤ <,吨8个∈吨8个,吨:[ü:]
        ≤ 吨8个[ü8个] ≤吨:[ü:] +\}; R2个←R2个∪R:;
4个
5个为了每个 r∈R2个做
6个   如果Δ(r, M) >X然后
7     R2个←R2个\ {A}
8个返回R2个;
```

A.4 实验设置

A.4.1 数据集。(1) 房子2个是关于电力消耗在一个家庭中。回归模型基于有功功率、无功功率和强度属性来预测电压。(2) 遥测3个来自环境传感器遥测数据。回归模型建立在一氧化碳、湿度和烟雾属性之上

2个https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption
3个https://www.kaggle.com/garystafford/environmental-sensor-data-132k

算法 2：对齐搜索 (R2个,四)

输入：一致的候选人Rc， 临界点d
输出：对齐的元组R小号没有重叠的时间戳

```
1个R小号←∅;
2个为了rc∈Rc做
3个  如果∀r小号∈R小号 , r2个-r小号然后
4个    R小号←R小号∪{rc};
5个局部最优←错误的;
6个尽管不是局部最优做
7    局部最优←真的; 为了
8个  2个≤p≤d做
9      'D=2←Rc\R小号;
10     为了 '>直流电⊆ 'D=2, |'>直流电| =? 做
11       如果∀A0， A1个∈'>直流电,A0-A1个然后
12         '8=← {A8=∈RZ<|∃A>直流电∈'>直流电,A>直流电=A8=};
13         如果 |'8=| ⩽ ? −1个然后
14           'Z<← 'Z<\'8=∪'>直流电;
15           局部最优←错误的;
16返回R小号;
```

算法 3：对齐搜索修剪（R2个,d, \)

输入：一致的候选人Rc， 临界点丁，时间限制 \ 输出：对齐的元组R小号没有重叠的时间戳

```
1个R小号←∅;
2个为了rc∈Rc做
3个  如果∀r小号∈R小号, r2个-r小号然后
4个    R小号←R小号∪{rc};
5个局部最优←错误的;
6个尽管不是局部最优做
7    局部最优←真的;
8个  为了1个≤p≤d -1个做
9      为了 '8=⊆R小号, |'8=| =? 做
10       如果∀A0∈'8=,∀A1个∈'8=,∀8个∈
           {1, 2, . . . ,<}, |C80−C81| ⩽ (2? − 1)\然后
11         '>直流电← {A>直流电∈'D=2|∃A8=∈'8=， A>直流电=
           A8=}∩{A>直流电∈'D=2|∀r∈RZ<\R8=,r -
           A>直流电};
12         如果 |'>直流电|⩾? +1个然后
13           'Z<← 'Z<\'8=∪'>直流电;
14           局部最优←错误的;
15 返回R小号;
```

来预测温度。(3) 水4个由水质监测站的传感器收集。回归模型建立在电导率、水温和浊度的基础上以预测水位。(4) 空气质量5个是北京空气质量监测点的空气污染物数据。它是一个具有11个属性的高维数据集，回归模型是通过其他属性来预测PM10。(5) Fuel 为车辆实际油耗数据，由汽车制造商收集。回归模型基于发动机扭矩和速度来预测油耗。

4个https://www.kaggle.com/ivivan/real-time-water-quality-data
5个https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data

对于 House、Telemetry 和 Air Quality，它们的时间戳都是自n然对齐的，可以作为基本事实。我们通过在时间戳上引入干扰来模拟未对齐的变量 [8]。对于 Fuel 来说，由于前面提到的传输延迟等问题，只有大约 5% 的数据是自然对齐的。类似地，对于 Water 数据集，在 29k 元组中，只有大约 13k 元组自然对齐。因此，我们将自然对齐元组的部分视为评估的基本事实，表示为R真相.

A.4.2 回归模型。我们采用回归模型，包括线性回归 (LR) 和 XGBoost (XGB [11])，由 scikit-learn [5] 实现。TabNet [6] 是一种更新的规范深度表格数据学习架构，用于多元回归。

A.4.3 对齐方法。我们比较我们提出的小号相似性 A排列在米奥德尔C现有方法的约束（SAMC）：（1）动态时间规整（DTW）[4]，一种基于动态规划的算法；（2）导数动态时间规整（DDTW）[21]，考虑数据局部导数的DTW扩展；(3) 典型时间扭曲 (CTW) [34]，是人体运动时空对齐的典型相关分析 (CCA) 的扩展；(4) 广义时间扭曲 (GTW) [35, 36]，它扩展了 DTW，用于在时间上对齐来自多个主题的多模态序列；(5) trainable time warping (TTW) [22] 利用 sinc 卷积核和基于梯度的优化技术进行多重对齐。由于我们的场景侧重于对齐多个变量，对于仅针对时间序列对提出的 DTW、DDTW 和 CTW，

A.4.4 评估指标。我们建议评估方法对齐元组的准确性（对齐精度）以及从对齐数据中学习模型的准确性（模型精度）。

为了对齐精度，我们比较实例,对齐 吨1个,吨2个,. . . ,吨<通过各种算法，求真R真相在 A.4.1 节中介绍。对齐精度由下式给出

$$\text{F1-sc矿石} = 2 * \frac{?A428B8>=*A420;;}{?A428B8>= +A420;;},$$

在哪里A420;; $=\frac{R_{真相} \cap R_{\text{—}}}{R_{真相}}$， 和 ？ A428B8>==$\frac{R_{真相} \cap R.}{R}$.

为了模型的准确性，我们保留每个数据集的 20% 自然对齐的元组作为测试数据 'C4BC用于评估学习到的模型。对于学习回归模型米,给定自变量
+1个,+2个,. . . ,+<-1个和因变量 +<，我们使用 RMSE，即

$$\text{'"((男，'C4BC) =} \frac{\sqrt{\sum_{A \in \text{'C4BC}} (\text{" (A [+1]个..., A [+<-1个]) - 一个 [+<])2个}}}{|\text{'C4BC}|}.$$

RMSE 越低，对齐和学习模型越好。

A.4.5 实施细节。在我们的小号相似性A排列在 米奥德尔C约束（SAMC），模型“由小号相似性 A对齐 (SA)，如图 2 所示。学习的模型“然后在 SAMC 中使用和更新，如第 2.1 节所述。

实验在具有 16 个 2.1GHZ 内核和 128 GB 内存的 Ubuntu 16.04 LTS 机器上进行。大规模数据对齐是在 Apache Spark 2.2.3 的 3 节点集群上执行的。每个节点有 32 GB 内存和 16 个 2.6GHz 内核。实验代码和数据以及证明可在 [3] 中找到。