

一种基于加速聚类法的时域速度计算方法*

Peng Zhang

Wenkai Lu

*翻译自 (P. Zhang, Lu 2016) 一文.

摘要

时域速度和正常时差参数 (moveout parameters) 可以直接从同相轴局部斜率 (local event slopes) 中获得, 而同相轴局部斜率是基于叠前地震道集估计出来的. 在实际应用中, 估计出的局部斜率总是有一定的误差, 特别是在信噪比较低的情况下. 因此, 地下速度信息可能隐藏在速度和其他正常时差参数所决定的图像中. 我们开发了一种加速聚类算法, 可以在事先不知道聚类中心数量的情况下寻找聚类中心. 首先, 实施平面波解构 (plane-wave destruction) 来估计同相轴局部斜率. 对于地震道集中的每一个样本, 我们根据同相轴局部斜率获得速度的估计和它在图像域中的位置. 这些被映射到新的空间中的点展现出了不同类的结构. 我们用混合分布模型来对这些点进行建模. 然后, 对混合模型的聚类中心进行确定, 这些聚类中心就对应着主要地下构造的极大似然速度. 利用估算的速度的不确定约束来选择对应的反射中心. 最后, 对聚类后的采样不均匀的结速度进行插值, 在规则格网上建立高效的速度模型. 在人工合成数据和实际数据上实施该算法, 我们确定了所提出它可以给出精度相对较高的叠加速度模型和时偏移速度模型.

1 简介

速度估计是地震反问题中的一项重要任务. 宏模型的建立通常是从用不同的动校正速度对 CMP 道集进行扫描开始的. 手动选取速度谱上的能量峰值点是一个十分费力的过程, 需要有经验的专业的数据处理人员才能完成这个任务. 在过去的几十年里, 人们采取了各种方法来自动的速度分析. (Toldi 1989) 描述了一种最早的自动速度分析算法, 其中速度模型是由可能的区间速度表示的. 沿着动校正曲线将堆积功率 (stacking power) 最大化, 从而获得最佳的速度模型. 相似谱 (semblance spectra) 上的自动速度选取问题也可以看作是一个射线跟踪问题. 将相似谱上的能量 (Fomel 2009) 作为慢度, 则相似谱顶部到底部具有最小移动时间的射线路径就对应一个最佳的选取速度. 为了克服在 CMP 道集上的速度分析中的反射点分散问题, 可以通过包括各向异性在内的速度延续来建立时偏移速度模型 (Adler 2002; Fomel 2003;

Alkhalifah, Fomel 2011; Burnett, Fomel 2011). 还有一种叫作图像波传播 (image wave propagation) 的类似的方法 (Schleicher, Costa, Novais 2008). 不同速度的重复时偏移 (Yilmaz, Tanir, Gregory 2001) 也是一种替代方法.

另一种方法是使用同相轴局部斜率. 在地震道集上接收到的同相轴局部斜率包含了珍贵的地下信息. 动校正速度 (NMO) 和时偏移速度可以直接从同相轴局部斜率中得出. (Ottolini 1983) 提出了与速度无关的同相轴局部斜率成像. (Fomel 2007) 和 (Cooke, Bóna, Hansen 2009) 表明, 利用叠前反射数据估计的同相轴局部斜率可以完成几乎所有常见的时域成像任务和速度估计. 地震数据或地震图像中包含的同相轴局部斜率和其他局部时差属性也可用于深度域上的速度模型反演, 如立体成像 (Billette, Lambaré 1998; Lambaré 2008) 等.

同相轴局部斜率估计的好坏决定了基于同相轴局部斜率的成像和反演结果. 局部倾斜叠加 (Ottolini 1983) 是提取同相轴局部斜率的标准工具之一. 希尔伯特变换 (Barnes 1996; Cooke, Bóna, Hansen 2009; P. Zhang, Lu, Li 2013; Wang, Lu, P. Zhang 2015) 也可以用来得到等于同相轴局部斜率的相位移. 另一种具有鲁棒性的斜率估计算法是平面波解构 (plane-wave destruction) (PWD) (Fomel 2002; Schleicher, Costa, Santos, et al. 2009). PWD 通过平面波来逼近局部波场. 通过最大限度上扁平化分数延迟滤波器, Chen 等人 (Chen, Fomel, Lu 2013a; Chen, Fomel, Lu 2013b) 进一步加速了 PWD 方法, 并使其适用于陡峭层. 由于估计的同相轴局部斜率总是受到噪声和干扰事件的影响而退化, 所以直接映射算子可能会导致成像部分受到高频振荡的污染, 从同相轴局部斜率得到的速度对噪声很敏感. 虽然图像扭曲的速度图可以得到与同相轴局部斜率相关的时域速度, (Fomel 2007) 但当地震道集中包含多个散射或多路径事件时, 速度图就会变得相当复杂. 速度图上的能量可以作为慢度, 从而通过求解初至射线追踪问题来进行自动速度选取. (Cooke, Bóna, Hansen 2009) 在速度图上使用静音和平均滤波器来抑制多路径事件.

本文提出了一种基于从同相轴局部斜率映射来的局部属性聚类的时域速度自动估计方法. 局部属性由一个混合分布模型来表示. 混合分布模型的聚类中心对应于主要地下构造的极大似然速度. 这些聚类中心

对由噪声、干扰事件和多路径事件引起的**同相轴局部斜率**退化具有鲁棒性。我们开发了一种加速聚类算法，从而高效率地找到聚类中心。在聚类后的不均匀采样中心上进行最小二乘插值，建立规则格网上的高效速度模型。通过在人工合成和实际数据上实施，我们评估了所提出的算法的性能，并证实了它符合我们理论上的预期。

2 理论

在本节中，我们通过混合分布模型来说明模型的建立过程。其中加速密度聚类算法是通过一些计算和分析给出的，堆积速度和叠前时偏移速度模型是通过聚类中心进行插值估计出来的。

2.1 混合分布模型

同相轴局部斜率包含地下反射的几何和速度信息。有了估计出的**同相轴局部斜率**，我们可以用简单的一对一映射算子在时域中成像。从反射时差的经典双曲假设开始：

$$t^2(h) = \tau_0^2 + \frac{h^2}{V_{\text{NMO}}^2(\tau_0)} \quad (1)$$

其中 τ_0 为双向零偏距走时， $t(h)$ 代表偏移量 h 处记录的双向走时， $V_{\text{NMO}}(\tau_0)$ 为叠加速度。我们将 $t(h)$ 相对于 h 的导数表示为 p_h (走时斜率)：

$$p_h(t, h) = \frac{dt}{dh} = \frac{h}{t V_{\text{NMO}}^2(\tau_0)} \quad (2)$$

通过上述两个方程，叠加速度 $V_{\text{NMO}}(\tau_0)$ 和零偏距走时 τ_0 可以被有效的表示出来 (Ottolini 1983; Fomel 2007)，如下面的两个公式所示：

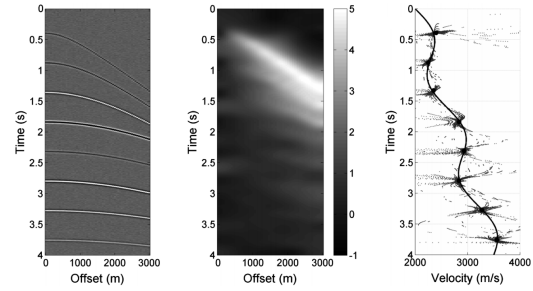
$$V_{\text{NMO}}(\tau_0) = \sqrt{\frac{h}{t p_h(t, h)}} \quad (3)$$

和

$$\tau_0 = \sqrt{t^2 - t p_h(t, h) h} \quad (4)$$

为了在每个 CMP 道集上得到一个稳健的**同相轴局部斜率**估计 p_h ，我们采用了 PWD(Fomel 2002; Chen, Fomel, Lu 2013a) 算法。在具体应用中可以将斜率估计算法用其他算法进行替换。域 $\{t, h\}$ 上的单个 CMP 道集数据点通过方程 (3) 和 (4) 映射到零偏距走时和

叠加速度域 $\{\tau_0, V_{\text{NMO}}\}$ 。在偏移点的附近，**同相轴局部斜率**接近零。因此，我们不使用零偏移点附近的数据。与通过图像扭曲的速度图不同的是 (Fomel 2007)，我们不再使用数据点的振幅。数据点以相等的权重被映射到新的空间中。对于实际数据的应用，可以用从估计斜率中得到的局部相干性来分配每个要映射的数据的权重 (P. Zhang, Lu, Li 2013)。图 1(a) 是使用 8 个



(a) 使用随机噪声 (b) 使用 PWD 估计 (c) 从局部属性映射到局部斜率的结果。
人工合成 CMP 计斜率。

图 1: 10dB 随机噪声的人工数据示例。(c) 中，灰色数据点是映射后的数据，黑色圆圈是聚类中心，黑色曲线表示真实速度。

收集器的在速度平滑变化的介质中的简单实验。CMP 道集的 Ricker 子波峰值频率为 20 Hz，接收器的间隔为 50 m，电缆总长度为 3 km，时间间隔为 4 ms，总时间为 4 s，该图是将速度从从 2.0 km/s 开始应用逆动校正获得的。CMP 道集中添加了 10dB 的随机高斯噪声。图 1(b) 描述了 PWD 估计出的**同相轴局部斜率**。可以看出，估计出的**同相轴局部斜率**相当平稳。图 1(c) 显示了数据点映射到 $\{\tau_0, V_{\text{NMO}}\}$ 空间后的分布。

需要注意的是，新空间中的数据点有成簇的性质。鉴于反射时差的双曲假设可以完全描述反射事件的运动，并且能完美估计**同相轴局部斜率**，因此，新空间中的数据点应该对应一次相同位置上的反射事件。在面面对实际数据时，双曲假设不能完全描述非平面反射体，因此，估计出的斜率永远会有偏差，一个反射事件会被映射成一簇。假定双曲假设和斜率估计算法的误差遵循高斯分布，那么，一次反射数据的映射结果可以用一个高斯分布来表示，其期望值为极大似然估计。对于多与一次反射的情况，可以直观的引入高斯混合模型。作为高斯分布的简单线性叠加，高斯混合模型可

以表示更加复杂的密度模型 (Bishop 2006). 高斯混合模型如下:

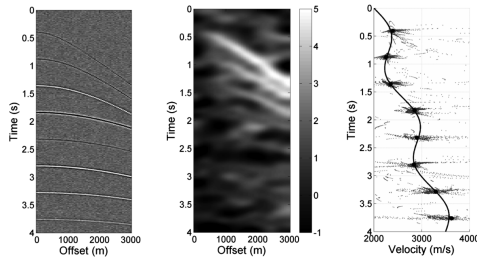
$$P(\mathbf{x}) = \sum_{k=1}^K a_k N_k(\mathbf{x} | \mu_k, \sigma_k) \quad (5)$$

其中, x 为映射数据点在新空间中的位置, $N_k(\mathbf{x} | \mu_k, \sigma_k)$ 为高斯分布, x 的期望为 μ_k , 标准差为 σ_k , K 代表单个高斯分布的总数量, a_k 取混合系数. 分布 $P(x)$ 是联合分布 $P(x|x \in N_k)$ 对所有可能的状态 $P(x \in N_k)$ 的总和, 即

$$P(\mathbf{x}) = \sum_{k=1}^K P(\mathbf{x} \in N_k) P(\mathbf{x} | \mathbf{x} \in N_k) \quad (6)$$

其中 $P(x \in N_k)$ 等于混合系数 a_k . 通过寻找这 K 个聚类中心, 可以有效地计算出每个高斯分布的期望值 μ_k .

在图 1(c) 中, 黑色圆圈代表聚类中心, 它给出了速度的极大似然解, 黑色曲线为真实的 NMO 速度. 可以看出, 所有的聚类中心都落在了真实的 NMO 速度的路径上. 聚类中心的数量为 8 个, 等于反射体的数量. 因此, 混合分布模型可以有效地解决计算从同相轴局部斜率映射的速度值的极大似然解的问题.



(a) 使用随机噪声的人工合成 CMP 道集. (b) 使用 PWD 估计出的同相轴局部斜率. (c) 从局部属性映射到局部斜率的结果.

图 2: 0dB 随机噪声的人工数据示例.

估计出的同相轴局部斜率的质量会因为噪声和干扰而降低. 图 2(a) 显示了与图 1(a) 中相同的 CMP 道集, 但添加了不同水平的随机噪声, 其信噪比为 0dB. 在这种情况下, 信号的功率等于随机高斯噪声的功率. 图 2(b) 展示了 PWD 的估计出的同相轴局部斜率. 尽管在 PWD 之前已经使用了带通滤波器来过滤噪声, 但估计出的同相轴局部斜率仍受到了强噪声的影响. 图 2(c) 显示了在 $\{\tau_0, v_{\text{NMO}}\}$ 空间中的映射数据点, 直接映射算子给出了被高频振荡污染的部分. 在

图 2(c) 中, 黑色圆圈代表聚类中心, 黑色实心曲线是真实的 NMO 速度. 可以看出, 此时所有的聚类中心仍然与真实 NMO 速度的路径相吻合, 这证实了聚类中心对同相轴局部斜率的估计是稳健的.

2.2 加速密度聚类

在所有用于寻找聚类中心的算法中, k-means 可能是应用最广泛的那一个. 为了实施 k-means 算法, 需要提前知道聚类中心的数量, 但这一先验信息通常难以确定 (Hamerly, Elkan 2004). (Wang 2012) 实现了层次聚类和划分, 以便在储层表征中找到正确的聚类中心数量. 通过带分割和合并的 k-means 方法 (Muhr, Granitzer 2009), 可以在合理的运行时间内选取出准确的聚类中心数量. (Lu, Yingqiang, Boran 2014) 应用类似的策略来确定浅水中的衍射地震噪声. 在 k-means 中, 数据点总是被分配到距离最近的中心. 虽然可以用各种距离将其拓展, 但 k-means 还是几乎不能检测到非球形簇.

密度聚类 (Rodriguez, Laio 2014) 则更容易应对更一般的数据分布情况. 聚类簇数也更容易确定. 聚类中心被描述为为密度相对高于相邻点, 而且与其他密度较大的中心点距离较大的点. 对于每个数据点 i , 其局部密度 ρ_i 和其与密度较高的点的最小距离 σ_i 是密度聚类中的两个十分重要的参数. 简单的 ρ_i 定义为:

$$\rho_i = \sum_j \lambda(d_{ij} - d_c) \quad (7)$$

其中, d_{ij} 为数据点 i 与数据点 j 之间的距离, d_c 为截止距离, 如果 $d_{ij} - d_c < 0$, 则 $\lambda = 1$, 否则 $\lambda = 0$. 该算法仅对不同数据点的 ρ_i 敏感, 这意味着其结果对参数 d_c 是稳健的 (Rodriguez, Laio 2014). 而一个数据点到密度较高的点的最小距离 σ_i 为:

$$\sigma_i = \min_{j: \rho_j > \rho_i} (d_{ij}) \quad (8)$$

对于密度最大的点, σ_i 取值为 $\max_j(d_{ij})$. 根据 σ_i 的定义, 我们可以得出 σ_i 只有在全局或局部最大值时才相对较大的. 然后选取局部密度 ρ_i 相对较大、最小距离 σ_i 相对较大的点作为聚类中心. 在实际应用中, 我们可以通过 β 的快速变化来确定中心, 即 β 是 ρ 和 σ 的乘积. 确定聚类中心后, 按照局部密度最高到最低的顺序对数据点进行分配. 数据点总是被分配到

离其最近的局部密度较高的那一类中. 总的来说, 密度聚类对 N 个数据点的计算复杂度为 $O(N^2)$, 详细复杂度分析见附录 A. 即使对于单个 CMP 道集, 在典型的 1000 个时间样本和 100 个源-接收器对的情况下, 也可能有 105 个数据点. 这使得密度聚类算法在实际数据应用也不太合适.

在模式识别中, 输入数据空间中的几何体可能是高度折叠、弯曲或扭曲的. 瑞士卷数据集 (Tenenbaum, De Silva, Langford 2000) 就是这样一类非线性流形. 在大多数地震数据中, 数据点呈现出的结构都是线性的. 因此, 可以用高斯或类似高斯来逼近数据点的分布. 我们提出了一种基于直方图函数的加速密度聚类算法. 在统计学中, 直方图是数据分布的一种表示. 在密度聚类中, 直方图函数来近似. 我们应用二维直方图分析方法 (Gonzalez, Woods, et al. 2002; P. Zhang, Lu, Y. Zhang 2015) 对映射后的零偏距时间和叠加速度进行分析, 得到局部密度的估计.

$$\rho = H(l_1, l_2) = m(l_1, l_2) \quad (9)$$

其中, l_1 和 l_2 分别代表零偏距时间和叠加速度, $H(l_1, l_2)$ 是二维直方图函数, $m(l_1, l_2)$ 是共同在某个组距内的零偏距时间和叠加速度映射后得到的同相轴局部斜率的数目, 这个组距用 l_1, l_2 所代表. 直方图可以用估计出的同相轴局部斜率的相干性来加权, 以保证稳健性, 因为相干性是映射属性可靠性的自然度量.

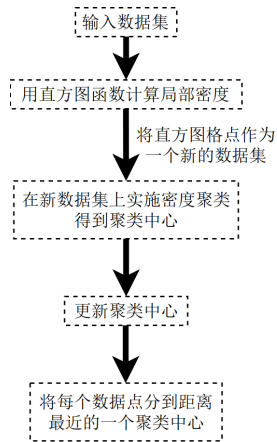


图 3: 加速密度聚类算法流程图.

加速密度聚类算法的流程图如图 3 所示. 为了证明加速密度算法的性能, 我们将其应用于 “S” 数据

集 (Fränti, Virtajoki 2006), 如图 4(a) 所示. 二维的

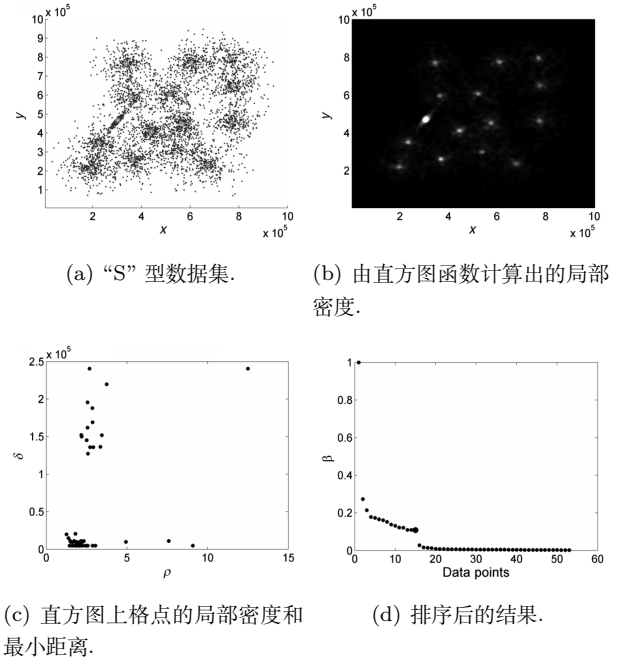
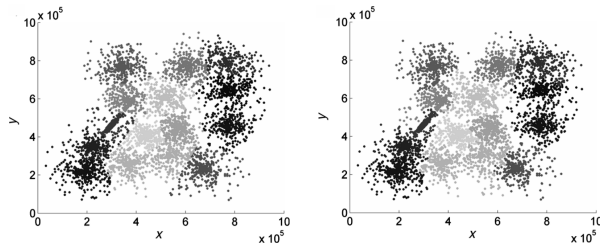


图 4: 加速密度聚类算法. 图 (a) 中的白色圆圈和白色星号分别是密度聚类 and 加速密度聚类结果的聚类中心. 图 (d) 中较大的圆点代表聚类中心的决策点.

“S” 型数据集的在空间分布上具有不同的复杂性. 它有 15 个预定义的聚类中心. 从图 4(a) 中我们可以看到, 数据点具有非球形的概率分布, 且不同聚类中心所属的概率分布重叠严重. 白色圆圈是密度聚类算法确定的聚类中心. 白色星号是由加速密度聚类算法确定的聚类中心. 可以看到, 加速密度聚类算法计算出的聚类中心与原始密度聚类算法计算出的聚类中心完全吻合. 这个结果表明加速密度聚类可以找到与原始密度聚类相同的极大似然解. 图 4(b) 显示了直方图函数所估计出的局部密度. 二维直方图上的密度峰值点保留了中心信息. 通过将直方图的格点作为一个新的数据集, 大大降低了计算成本, 因为新数据集中的点数量很少. 图 4(c) 给出了每个数据点的局部密度 ρ_i 和新数据集中密度较高的点的最小距离 σ_i . 图 4(d) 是 β 的排序结果, 使用的是归并排序 (Satish et al. 2010). 在对 β 进行排序后, 选择聚类中心为那些 β 相对较大的点. 通常情况下, 聚类中心与其他普通点之间会有一个突变. 利用这一特性, 在排序后的 β 上应用梯度算子来检测突变. 在图 4(d) 中, 有突变的点用一个较大的圆点表示. β 值大于或等于这个点的

数据点被识别为新数据集的聚类中心. 这些中心给出了原始数据集的真实聚类中心的大致位置. 然后进行聚类中心的更新, 找到真正的中心. 对于一个由 N 个数据点组成的数据集, 加速密度聚类的计算复杂度为 $O(1/(K(r_M/r_{clu})^4 N^2))$, 其中 K 为聚类中心的数量, r_M 和 r_{clu} 分别为直方图中组距的平均半径和聚类簇的平均半径. 加速密度聚类算法的详细计算复杂度分析见附录 A. 而密度聚类算法的计算复杂度为 $O(N^2)$. 加速密度聚类算法比原密度聚类算法快 $K(r_{clu}/r_M)^4$ 倍. 在“S”型数据集的例子中, 有 5000 个数据点, 在 Matlab 中编写的新算法在单核 CPU 上的运行时间为 0.3 s, 而原密度聚类算法的运行时间为 22 s, 加速后的密度聚类算法速度约原来的为 73 倍.

为了更加完整地表示算法的性能, 我们给出了原始数据点的分配结果. 图 5(a) 和图 5(b) 分别是我们的



(a) 加速密度聚类算法的分配结果. (b) 原始密度聚类算法的分配结果.

图 5: “S” 型数据中数据点的分配结果. 不同灰度表示不同簇.

的算法和原始密度聚类算法的结果. 除了在组的边界处有细微的差异外, 它们的结果十分相似, 这也证明了我们的算法良好性能.

2.3 自动速度估计

通过混合分布模型和加速密度聚类算法的介绍, 我们可以有效地从主要地下构造的同相轴局部斜率中找到映射后的极大似然速度. 主要的反射体通常具有较高的信噪比, 可以很好地被 PWD 捕获. 通过计算单个 CMP 道集的映射后的叠加速度 $V_{NMO}(\tau_0)$ 和零偏距走时 τ_0 的聚类中心, 我们可以得到了作为结点速度的主反射体的速度. 在所有 CMP 道集上重复这一过程后, 我们就得到了整个测量的结点速度. 因为假设叠加速度是横向连续的, 所以从相邻 CMP 道集点

映射得到的局部属性可以添加到当前位置作为聚类的输入.

将自动叠加速度估计扩展到时偏移速度分析上是很直观的. 只需要从堆叠前空间 $\{t, h, y\}$ 映射到时偏移图像空间 $\{\tau, x, V_{mig}\}$ (Fomel 2007), 其中 y 为中点坐标, τ 为时偏移双向垂直走时, V_{mig} 为时偏移速度, x 为时偏移图像坐标. 以映射后的局部属性为输入数据, 对每一个成像点道集 (CIG) 实施加速密度聚, 来提取整个事件的结时偏移速度. 同样的, 对结果进行插值, 建立时偏移速度模型.

为了将不均匀采样的聚类后的结点速度插入到格网中, 我们可以通过计算一个插值问题:

$$\mathbf{G}\mathbf{v}_{grid} = \mathbf{v}_{knot} \quad (10)$$

其中 \mathbf{v}_{grid} 是规则格网上的速度, \mathbf{v}_{knot} 取不均匀采样上的结点速度, \mathbf{G} 代表插值算子, 在我们的例子中是立方 B-splines 插值. 由于我们的目标是建立平稳的时域速度模型, 所以在最小二乘目标函数中加入了一个正则化项 ϵ :

$$\|\mathbf{G}\mathbf{v}_{grid} - \mathbf{v}_{knot}\|_2^2 + \epsilon^2 \|\mathbf{L}\mathbf{v}_{grid}\|_2^2 \quad (11)$$

其中 \mathbf{L} 代表一个粗化算子 (如 Laplacian 算子). 目标函数的最小化结果为:

$$\mathbf{v}_{grid} = (\mathbf{G}^T \mathbf{G} + \epsilon^2 \mathbf{L}^T \mathbf{L})^{-1} \mathbf{G}^T \mathbf{v}_{knot} \quad (12)$$

通过上述公式, 我们建立了规则格网上的速度模型. 这个时域速度模型可以进一步用于时域成像任务.

3 在数据集上进行测试

为了证明加速密度聚类算法对速度进行自动估计的性能, 我们将其应用在合成和真实数据集上. 在这两种情况下, 叠加速度和时偏移速度都是自动估计的. 此外, 我们还利用估算的速度进行了 NMO 校正和时间成像.

A 密度聚类 and 加速密度聚类的复杂性分析

考虑对 N 个数据点进行聚类, 需要计算任意两点之间的距离, 得到截止局部密度 ρ_i . 因此需要的运算总数为

$$N + (N-1) + (N-2) + \cdots + 3 + 2 + 1 = \frac{N(1+N)}{2} \quad (13)$$

可以看到, 其复杂度为 $O(N^2)$. 然后, 我们需要将一个数据点的局部密度与其他所有点的局部密度进行比较, 得到最小距离 σ_i . 为了加快最小距离的计算, 归并排序 (Satish et al. 2010) 被实现在局部密度的序列上, 这个操作的复杂度是 $O(N \log N)$. 只有对局部密度高于当前点的数据点才需要进行比较, 需要的次数与公式 13 相同, 即为 $O(N^2)$. 乘积 β 将需要 N 次的乘法. 对 β 进行归并排序, 选择聚类中心, 复杂度为 $O(N \log N)$. 经过 N 次减法, 对排序后的 β 进行求梯度运算, 检测其突变. 确定聚类中心后, 进行对数据点进行分配, 复杂度为 $O(N)$. 因此, 密度聚类的计算复杂度近似为 $O(N^2)$.

加速密度聚类算法从复杂度为 $O(N)$ 的对直方图进行处理开始, 将直方图的格点作为一个新的数据集进行聚类. 对于二维的情况, 周长数为 $M1, M2$, $M1, M2$ 分别为时间和偏移维度, $M1, M2$ 远小于 N . 然后对新数据集进行密度聚类, 得到聚类中心, 这一步骤的复杂度为 $O((M1M2)^2)$. 每个聚类中心位置更新是通过重新计算组距内密度最高的点来完成的. 粗略估计每个聚类中心的组距中的数据点数量为 $1/(K(r_M/r_{clu})^2N)$, 其中 K 为聚类中心的个数, r_M 和 r_{clu} 分别为直方图中组距的平均半径和簇的平均半径. 更新过程中需要对每个聚类中心进行距离计算. K 个聚类中心的距离计算总数为

$$K \sum_{i=1}^{\frac{1}{K} \left(\frac{r_M}{r_{clu}}\right)^2 N} i = \frac{1}{2} \left[\frac{1}{K} \left(\frac{r_M}{r_{clu}}\right)^2 N + 1 \right] \left(\frac{r_M}{r_{clu}}\right)^2 N \quad (14)$$

这个更新聚类中心过程的复杂度为 $O(1/K(r_M/r_{clu})^4 N^2)$. 最后一步是用最近邻原则将每个原始数据点分配到更新后的聚类中心, 其复杂度为 $O(N)$. 通过省略低阶项, 得到加速密度聚类的近似复杂度 $O(1/K(r_M/r_{clu})^4 N^2)$.

加速的密度聚类算法可以比原算法快 $K(r_{clu}/r_M)^4$ 倍. 在“S”型数据集的例子中, 有

5000 个数据点, 15 个聚类中心. 速率 r_{clu}^2/r_M^2 约为 2, 因此, 大约比原算法提速了 60 倍.

参考文献

- [1] ZHANG P, LU W. Automatic time-domain velocity estimation based on an accelerated clustering method[J]. *Geophysics*, 2016, 81(4): U13-U23.
- [2] TOLDI J L. Velocity analysis without picking[J]. *Geophysics*, 1989, 54(2): 191-199.
- [3] FOMEL S. Velocity analysis using AB semblance[J]. *Geophysical Prospecting*, 2009, 57(3): 311-321.
- [4] ADLER F. Kirchhoff image propagation[J]. *Geophysics*, 2002, 67(1): 126-134.
- [5] FOMEL S. Time-migration velocity analysis by velocity continuation[J]. *Geophysics*, 2003, 68(5): 1662-1672.
- [6] ALKHALIFAH T, FOMEL S. The basic components of residual migration in VTI media using anisotropy continuation[J]. *Journal of Petroleum Exploration and Production Technology*, 2011, 1(1): 17-22.
- [7] BURNETT W, FOMEL S. Diffraction Imaging Using 3D Azimuthally-anisotropic Velocity Continuation[C]//73rd EAGE Conference and Exhibition incorporating SPE EUROPEC 2011. [S.l. : s.n.], 2011: cp-238.
- [8] SCHLEICHER J, COSTA J, NOVAIS A. Time-migration velocity analysis by image-wave propagation of common-image gathers[J]. *Geophysics*, 2008, 73(5): VE161-VE171.
- [9] YILMAZ Ö, TANIR I, GREGORY C. A unified 3-D seismic workflow[J]. *Geophysics*, 2001, 66(6): 1699-1713.
- [10] OTTOLINI R. Velocity independent seismic imaging[J]. *Stanford Exploration Project*, 1983, 37: 59-68.

- [11] FOMEL S. Velocity-independent time-domain seismic imaging using local event slopes[J]. *Geophysics*, 2007, 72(3): S139-S147.
- [12] COOKE D, BÓNA A, HANSEN B. Simultaneous time imaging, velocity estimation, and multiple suppression using local event slopes[J]. *Geophysics*, 2009, 74(6): WCA65-WCA73.
- [13] BILLETTE F, LAMBARÉ G. Velocity macro-model estimation from seismic reflection data by stereotomography[J]. *Geophysical Journal International*, 1998, 135(2): 671-690.
- [14] LAMBARÉ G. Stereotomography[J]. *Geophysics*, 2008, 73(5): VE25-VE34.
- [15] BARNES A E. Theory of 2-D complex seismic trace analysis[J]. *Geophysics*, 1996, 61(1): 264-272.
- [16] ZHANG P, LU W, LI F. Supertrace-based coherence algorithm with robust slope estimation[C]//75th EAGE Conference & Exhibition incorporating SPE EUROPEC 2013. [S.l. : s.n.], 2013: cp-348.
- [17] WANG Y, LU W, ZHANG P. An improved coherence algorithm with robust local slope estimation[J]. *Journal of Applied Geophysics*, 2015, 114: 146-157.
- [18] FOMEL S. Applications of plane-wave destruction filters[J]. *Geophysics*, 2002, 67(6): 1946-1960.
- [19] SCHLEICHER J, COSTA J, SANTOS L, et al. On the estimation of local slopes[J]. *Geophysics*, 2009, 74(4): P25-P33.
- [20] CHEN Z, FOMEL S, LU W. Accelerated plane-wave destruction[J]. *Geophysics*, 2013, 78(1): V1-V9.
- [21] CHEN Z, FOMEL S, LU W. Omnidirectional plane-wave destruction[J]. *Geophysics*, 2013, 78(5): V171-V179.
- [22] BISHOP C M. Pattern recognition and machine learning[M]. [S.l.]: springer, 2006.
- [23] HAMERLY G, ELKAN C. Learning the k in k-means[J]. *Advances in neural information processing systems*, 2004, 16: 281-288.
- [24] WANG Y. Reservoir characterization based on seismic spectral variations[J]. *Geophysics*, 2012, 77(6): M89-M95.
- [25] MUHR M, GRANITZER M. Automatic cluster number selection using a split and merge k-means approach[C]//2009 20th International Workshop on Database and Expert Systems Application. [S.l. : s.n.], 2009: 363-367.
- [26] LU W, YINGQIANG Z, BORAN Z. Automatic source localization of diffracted seismic noise in shallow water[J]. *Geophysics*, 2014, 79(2): V23-V31.
- [27] RODRIGUEZ A, LAIO A. Clustering by fast search and find of density peaks[J]. *Science*, 2014, 344(6191): 1492-1496.
- [28] TENENBAUM J B, DE SILVA V, LANGFORD J C. A global geometric framework for nonlinear dimensionality reduction[J]. *Science*, 2000, 290(5500): 2319-2323.
- [29] GONZALEZ R C, WOODS R E, et al. Digital image processing[Z]. 2002.
- [30] ZHANG P, LU W, ZHANG Y. Velocity analysis with local event slopes related probability density function[J]. *Journal of Applied Geophysics*, 2015, 123: 177-187.
- [31] FRÄNTI P, VIRMAJOKI O. Iterative shrinking method for clustering problems[J]. *Pattern Recognition*, 2006, 39(5): 761-775.
- [32] SATISH N, KIM C, CHHUGANI J, et al. Fast sort on CPUs and GPUs: a case for bandwidth oblivious SIMD sort[C]//Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. [S.l. : s.n.], 2010: 351-362.