ID Estimation Methods

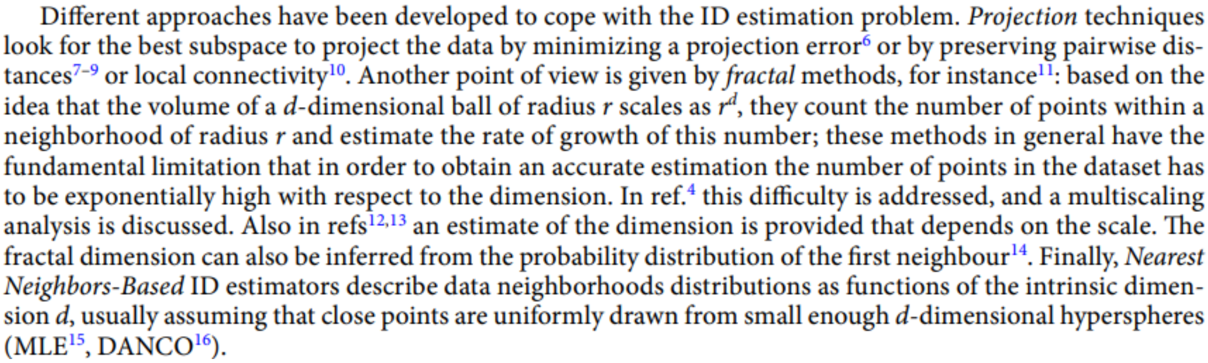
Two Nearest Neighbors:

- It provides an estimate of the ID by considering only the first and second neighbor of each point.
- Since the distance between only 2 Nearest neighbors is considered, it has its benefits:

a. Reduce the effects of density variations
b. Reduce the effects of curvature (of manifold) in the estimation process
c. Low computational cost

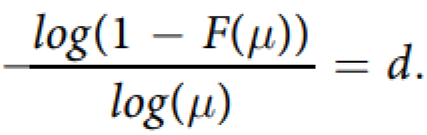
- The method is theoretically correct in uniformly distributed datasets. The effectiveness in non uniformly distributed datasets is not mentioned.

- Assumption of the method - Dataset is locally uniform in density. Locally means in the range of a second neighbor.
- Assumptions of alternative methods of find Intrinsic dimensions
  Nearest neighbor based ID estimator (MLE, DANCO) assumptions - close points are uniformly drawn from small enough d-dimensional hyperspheres



Different approaches have been developed to cope with the ID estimation problem. *Projection* techniques look for the best subspace to project the data by minimizing a projection error[6] or by preserving pairwise distances[7–9] or local connectivity[10]. Another point of view is given by *fractal* methods, for instance[11]: based on the idea that the volume of a d-dimensional ball of radius r scales as r^d, they count the number of points within a neighborhood of radius r and estimate the rate of growth of this number; these methods in general have the fundamental limitation that in order to obtain an accurate estimation the number of points in the dataset has to be exponentially high with respect to the dimension. In ref.[4] this difficulty is addressed, and a multiscaling analysis is discussed. Also in refs[12,13] an estimate of the dimension is provided that depends on the scale. The fractal dimension can also be inferred from the probability distribution of the first neighbour[14]. Finally, *Nearest Neighbors-Based* ID estimators describe data neighborhoods distributions as functions of the intrinsic dimension d, usually assuming that close points are uniformly drawn from small enough d-dimensional hyperspheres (MLE[15], DANCO[16]).

- The equation of ID estimation using 2NN is free from density, i.e the ID estimation is not dependent on the density.



- $$-\frac{log(1 - F(\mu))}{log(\mu)} = d.$$

Mathematical proof

Results

Let i be a point in the dataset, and consider the list of its first k nearest neighbors; let $r_1, r_2, \ldots, r_k$ be a sorted list of their distances from i. Thus, r_1 is the distance between i and its nearest neighbor, r_2 is the distance with its second nearest neighbor and so on; in this definition we conventionally set $r_0 = 0$.

The volume of the hypersferical shell enclosed between two successive neighbors $l - 1$ and l is given by

$$\Delta v_l = \omega_d (r_l^d - r_{l-1}^d), \tag{1}$$

where d is the dimensionality of the space in which the points are embedded and ω_d is the volume of the d-sphere with unitary radius. It can be proved (see SI for a derivation) that, if the density is constant around point i,all the Δv_l are independently drawn from an exponential distribution with rate equal to the density ρ:

$$P(\Delta v_l \in [v, v + dv]) = \rho e^{-\rho v} dv. \tag{2}$$

Consider two shells Δv_1 and Δv_2, and let R be the quantity $\frac{\Delta v_i}{\Delta v_j}$; the previous considerations allow us, in the case of constant density, to compute exactly the probability distribution (pdf) of R:

$$\begin{aligned} & P(R \in [\overline{R}, \overline{R} + d\overline{R}]) \\ & = \int_0^\infty dv_i \int_0^\infty dv_j \rho^2 e^{-\rho(v_i + v_j)} 1_{\left\{ \frac{v_j}{v_i} \in [\overline{R}, \overline{R} + d\overline{R}] \right\}} \\ & = d\overline{R} \frac{1}{(1 + \overline{R})^2}, \end{aligned}$$

where 1 represents the indicator function. Dividing by $d\overline{R}$ we obtain the pdf for R:

$$g(R) = \frac{1}{(1 + R)^2}. \tag{3}$$

The pdf does not depend explicitly on the dimensionality d, which appears only in the definition of R. In order to work with a cdf depending explicitly on d we define quantity $\mu \doteq \frac{r_2}{r_1} \in [1, +\infty)$. R and μ are related by equality

$$R = \mu^d - 1. \tag{4}$$

This equation allows to find an explicit formula for the distribution of μ:

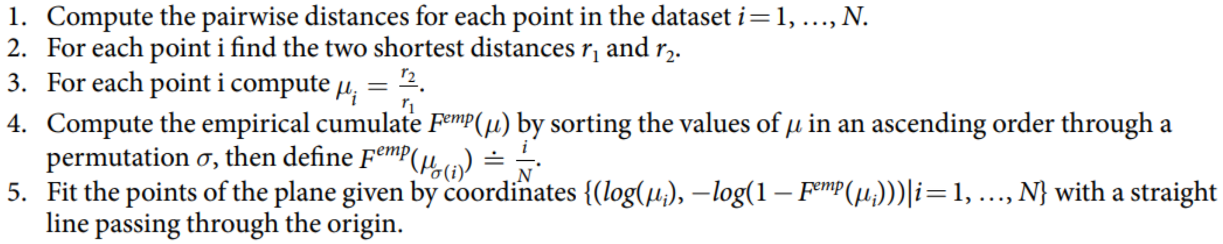
$$f(\mu) = d\mu^{-d-1} 1_{[1, +\infty]}(\mu), \tag{5}$$

while the cumulative distribution (cdf) is obtained by integration:

$$F(\mu) = (1 - \mu^{-d}) 1_{[1, +\infty]}(\mu). \tag{6}$$

Functions f and F are independent of the local density, but depend explicitly on the intrinsic dimension d.

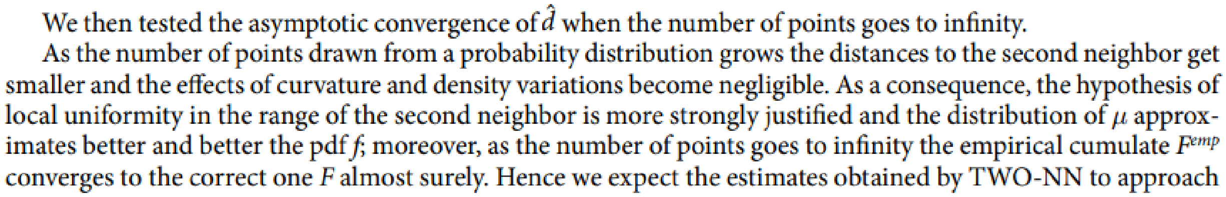
Algorithm



1. Compute the pairwise distances for each point in the dataset $i = 1, \ldots, N$.
2. For each point i find the two shortest distances r_1 and r_2.
3. For each point i compute $\mu_i = \frac{r_2}{r_1}$.
4. Compute the empirical cumulate $F^{emp}(\mu)$ by sorting the values of μ in an ascending order through a permutation σ, then define $F^{emp}(\mu_{\sigma(i)}) \doteq \frac{i}{N}$.
5. Fit the points of the plane given by coordinates $\{(log(\mu_i), -log(1 - F^{emp}(\mu_i)))|i = 1, \ldots, N\}$ with a straight line passing through the origin.

Effect of scaling

- As the number of data points increases, the distance to the second nearest neighbor decreases, reducing the impact of curvature and density variations.
- This strengthens the assumption of local uniformity, meaning the distribution of the second-nearest neighbor distances better approximates the true probability density function (pdf).
- As a result, the empirical cumulative distribution converges to the true cumulative distribution, leading to more accurate estimates of intrinsic dimensionality (ID).
- Therefore, with more data points, the estimate becomes more precise, and the estimated ID tends to increase.



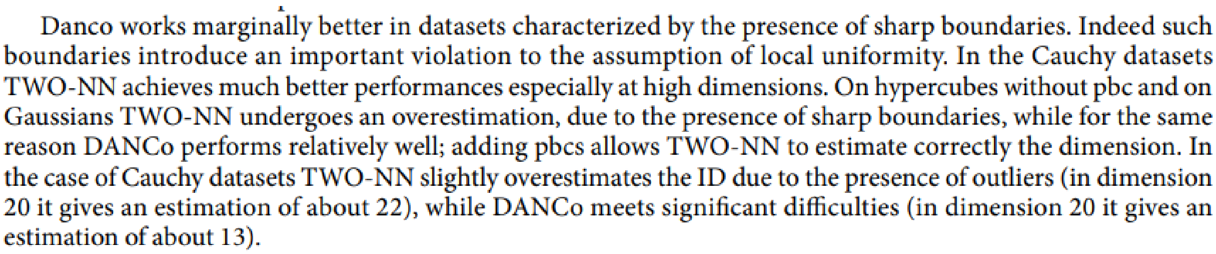
We then tested the asymptotic convergence of $\hat{d}$ when the number of points goes to infinity.

As the number of points drawn from a probability distribution grows the distances to the second neighbor get smaller and the effects of curvature and density variations become negligible. As a consequence, the hypothesis of local uniformity in the range of the second neighbor is more strongly justified and the distribution of μ approximates better and better the pdf f; moreover, as the number of points goes to infinity the empirical cumulate F^{emp} converges to the correct one F almost surely. Hence we expect the estimates obtained by TWO-NN to approach

Danco vs TwoNN:

Danco - considered as one of the best state-of-the-art methods according to *Campadelli, P., Casiraghi, E., Ceruti, C. & Rozza, A. Intrinsic dimension estimation: Relevant techniques and a benchmark framework. Mathematical Problems in Engineering 2015, https://doi.org/10.1155/2015/759567 (2015).*

- DANCo performs slightly better on datasets with sharp boundaries.
- TwoNN is more effective for Cauchy distributions but tends to overestimate intrinsic dimensionality (ID) in high-dimensional datasets, likely due to sensitivity to outliers.
- On hypercubes without periodic boundary conditions (pbc) and Gaussian distributions, TwoNN tends to overestimate ID due to sharp boundaries, while DANCo performs better in these scenarios.
- Adding periodic boundary conditions (pbcs) helps TwoNN estimate the correct dimension.



Danco works marginally better in datasets characterized by the presence of sharp boundaries. Indeed such boundaries introduce an important violation to the assumption of local uniformity. In the Cauchy datasets TWO-NN achieves much better performances especially at high dimensions. On hypercubes without pbc and on Gaussians TWO-NN undergoes an overestimation, due to the presence of sharp boundaries, while for the same reason DANCo performs relatively well; adding pbcs allows TWO-NN to estimate correctly the dimension. In the case of Cauchy datasets TWO-NN slightly overestimates the ID due to the presence of outliers (in dimension 20 it gives an estimation of about 22), while DANCo meets significant difficulties (in dimension 20 it gives an estimation of about 13).

- One of the findings of the study appeared to be interesting: “In the case of locally uniform distributions of points TWO-NN relies on a robust theoretical framework while in the general case, namely in the presence of curvatures and density variations, TWO-NN is numerically consistent.”

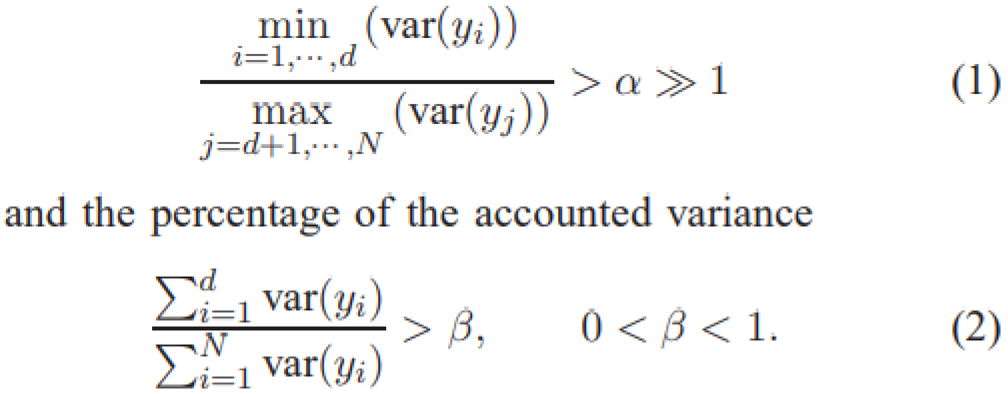
- Finally, TwoNN is a fractal dimension-based method as it estimates the intrinsic dimensionality by analyzing the scaling properties of data.

Principal Component Analysis

- Classical PCA can only be used with datasets with linear structure, not with non linear structure
- The PCA based method for finding ID works with non linear structure as well. The revised PCA method can filter out noise and outliers in data.

- Traditional PCA can identify the main directions of variance in the data, but it assumes a linear structure.
- The C-PCA method adapts this by performing PCA on small subsets of the data, allowing it to handle nonlinear structures.
- It divides the data into smaller subsets, performs local PCA, and aggregates the results to estimate the ID. This method is also suitable for incremental learning and can filter out noise, leading to more stable and accurate ID estimations.

- There are 2 conditions that need to be satisfied for ID estimation with the PCA method



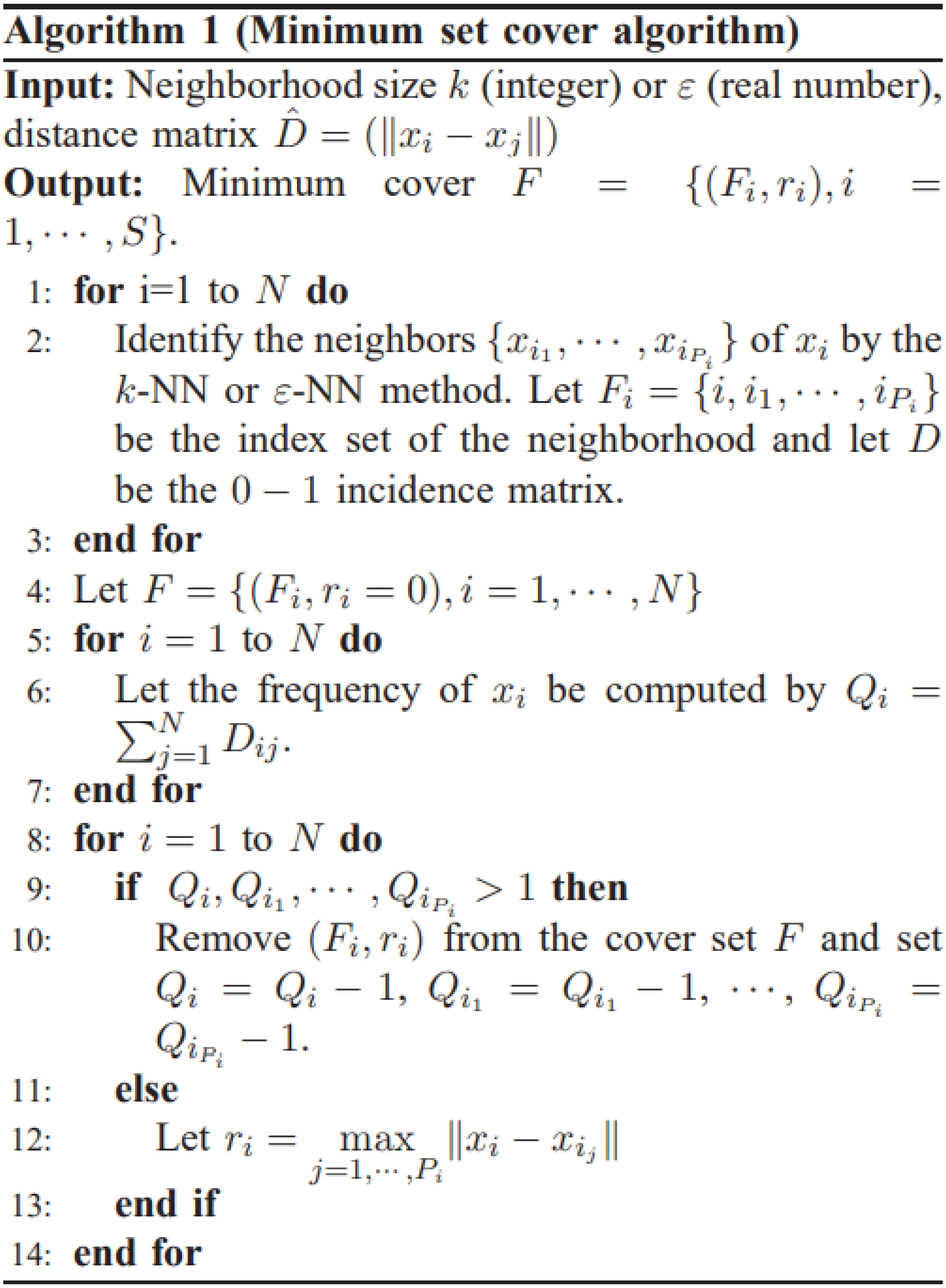
$$\frac{\min_{i=1,\cdots,d}(\mathrm{var}(y_i))}{\max_{j=d+1,\cdots,N}(\mathrm{var}(y_j))} > \alpha \gg 1 \qquad (1)$$

and the percentage of the accounted variance

$$\frac{\sum_{i=1}^{d}\mathrm{var}(y_i)}{\sum_{i=1}^{N}\mathrm{var}(y_i)} > \beta, \qquad 0 < \beta < 1. \qquad (2)$$

Local region selection algorithm

- The data is divided into small regions (subsets) that cover the entire dataset. This is done using either the k-Nearest Neighbors (k-NN) method or the ε-NN method.
- A minimal cover of these subsets is found, which reduces the computational complexity and ensures that the entire dataset is considered in the ID estimation.



Algorithm 1 (Minimum set cover algorithm)

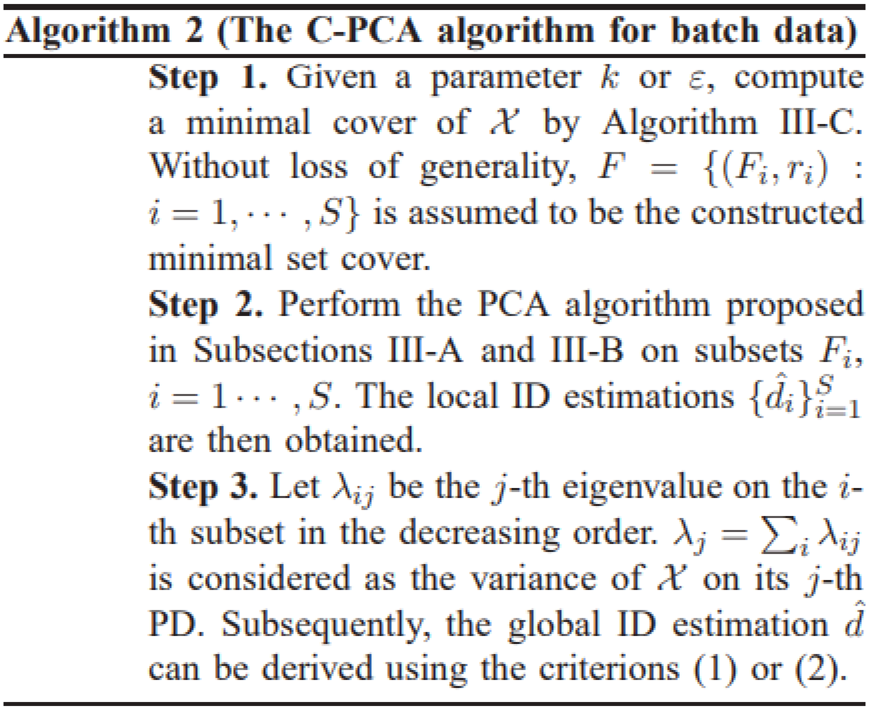
Input: Neighborhood size k (integer) or ε (real number), distance matrix $\hat{D} = (\|x_i - x_j\|)$

Output: Minimum cover $F = \{(F_i, r_i), i = 1, \cdots, S\}$.

```
for i=1 to $N$ do
    Identify the neighbors $\{x_{i_1}, \cdots, x_{i_{P_i}}\}$ of $x_i$ by the $k$-NN or $\varepsilon$-NN method. Let $F_i = \{i, i_1, \cdots, i_{P_i}\}$ be the index set of the neighborhood and let $D$ be the $0-1$ incidence matrix.
end for
Let $F = \{(F_i, r_i = 0), i = 1, \cdots, N\}$
for $i = 1$ to $N$ do
    Let the frequency of $x_i$ be computed by $Q_i = \sum_{j=1}^{N} D_{ij}$.
end for
for $i = 1$ to $N$ do
    if $Q_i, Q_{i_1}, \cdots, Q_{i_{P_i}} > 1$ then
        Remove $(F_i, r_i)$ from the cover set $F$ and set $Q_i = Q_i - 1$, $Q_{i_1} = Q_{i_1} - 1$, $\cdots$, $Q_{i_{P_i}} = Q_{i_{P_i}} - 1$.
    else
        Let $r_i = \max_{j=1,\cdots,P_i} \|x_i - x_{i_j}\|$
    end if
end for
```

Proposed ID estimation algorithm using local PCA on the minimal set cover

- The algorithm starts by dividing the data into subsets, then performs local PCA on each subset, and finally aggregates the results to estimate the global ID.
- Incremental Mode: For streaming or sequential data, the algorithm updates the ID estimation as new data points are added.

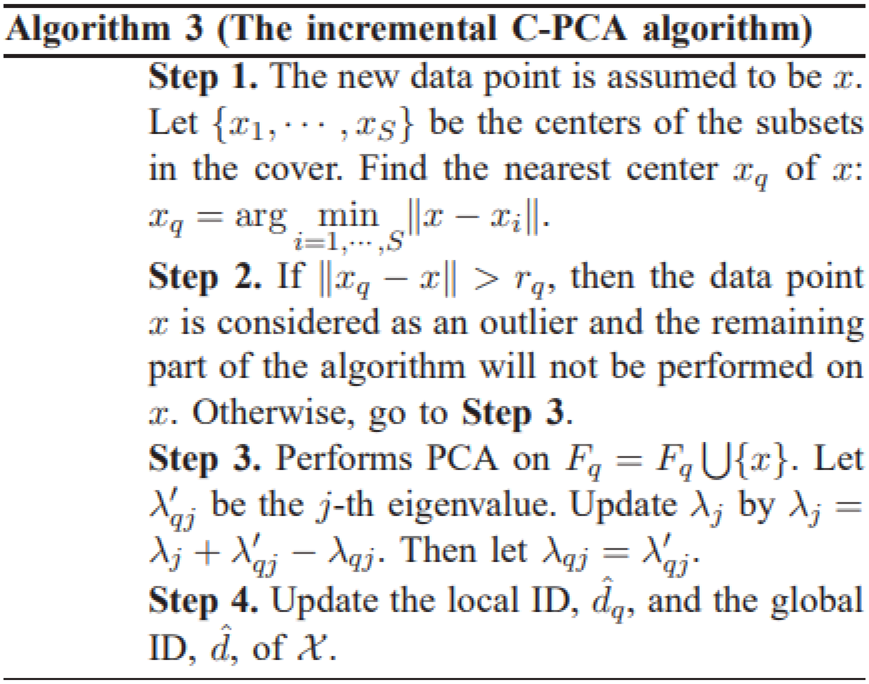


Algorithm 2 (The C-PCA algorithm for batch data)

Step 1. Given a parameter k or ε, compute a minimal cover of $\mathcal{X}$ by Algorithm III-C. Without loss of generality, $F = \{(F_i, r_i) : i = 1, \cdots, S\}$ is assumed to be the constructed minimal set cover.

Step 2. Perform the PCA algorithm proposed in Subsections III-A and III-B on subsets F_i, $i = 1 \cdots, S$. The local ID estimations $\{\hat{d}_i\}_{i=1}^{S}$ are then obtained.

Step 3. Let λ_{ij} be the j-th eigenvalue on the i-th subset in the decreasing order. $\lambda_j = \sum_i \lambda_{ij}$ is considered as the variance of $\mathcal{X}$ on its j-th PD. Subsequently, the global ID estimation $\hat{d}$ can be derived using the criterions (1) or (2).



Algorithm 3 (The incremental C-PCA algorithm)

Step 1. The new data point is assumed to be x. Let $\{x_1, \cdots, x_S\}$ be the centers of the subsets in the cover. Find the nearest center x_q of x: $x_q = \arg\min_{i=1,\cdots,S} \|x - x_i\|$.

Step 2. If $\|x_q - x\| > r_q$, then the data point x is considered as an outlier and the remaining part of the algorithm will not be performed on x. Otherwise, go to **Step 3**.

Step 3. Performs PCA on $F_q = F_q \bigcup \{x\}$. Let λ'_{qj} be the j-th eigenvalue. Update λ_j by $\lambda_j = \lambda_j + \lambda'_{qj} - \lambda_{qj}$. Then let $\lambda_{qj} = \lambda'_{qj}$.

Step 4. Update the local ID, $\hat{d}_q$, and the global ID, $\hat{d}$, of $\mathcal{X}$.

Summarizing, the method,

- It is robust to noise and outliers.
- It converges to a stable ID estimation as the neighborhood size increases.
- It can be efficiently applied to both batch and incremental data.

- The method utilizes the entire dataset, leading to more reliable estimates.

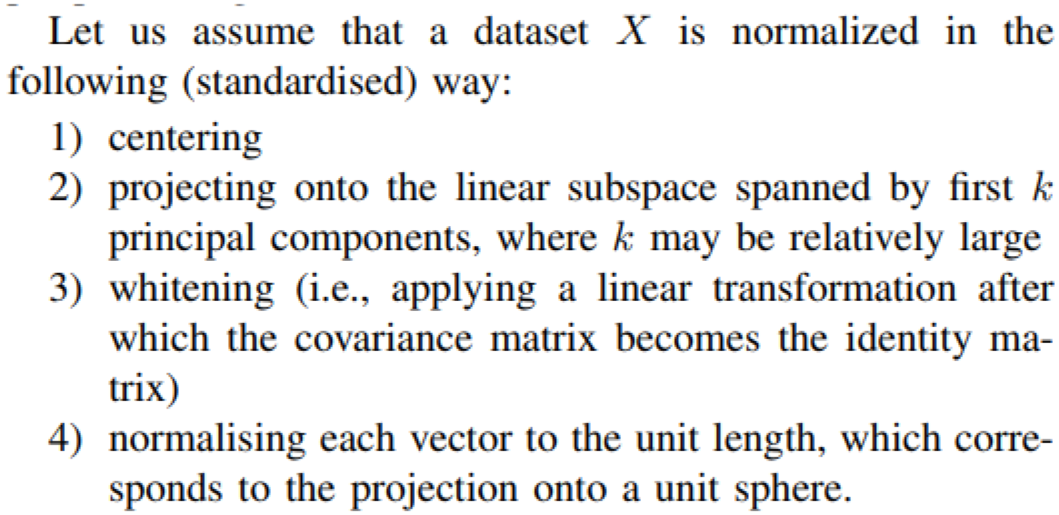
- Finally, PCA based ID estimation is a Combinatorial Method because it involves statistical approaches that analyze the structure of the data using linear combinations.

Fisher separability

- Intrinsic manifold assumption - High-dimensional data lies on a lower-dimensional manifold within the high-dimensional space, meaning the data's essential features can be represented in fewer dimensions.
- Fisher separability allows estimating the intrinsic dimension in situations where the intrinsic manifold assumption is not valid.

Estimating the intrinsic dimensionality of a dataset using separability properties - This approach uses linear separability and probabilistic measures to estimate the intrinsic dimensionality of datasets

- Linear Separability: A point x ∈ Rn is linearly separable from a finite set Y ⊂ Rn if there exists a linear functional l such that l(x) > l(y) for all y ∈ Y. This property is checked using methods like linear Support Vector Machines (SVM) or Fisher's linear discriminant.
- Normalization Process:



Let us assume that a dataset X is normalized in the following (standardised) way:

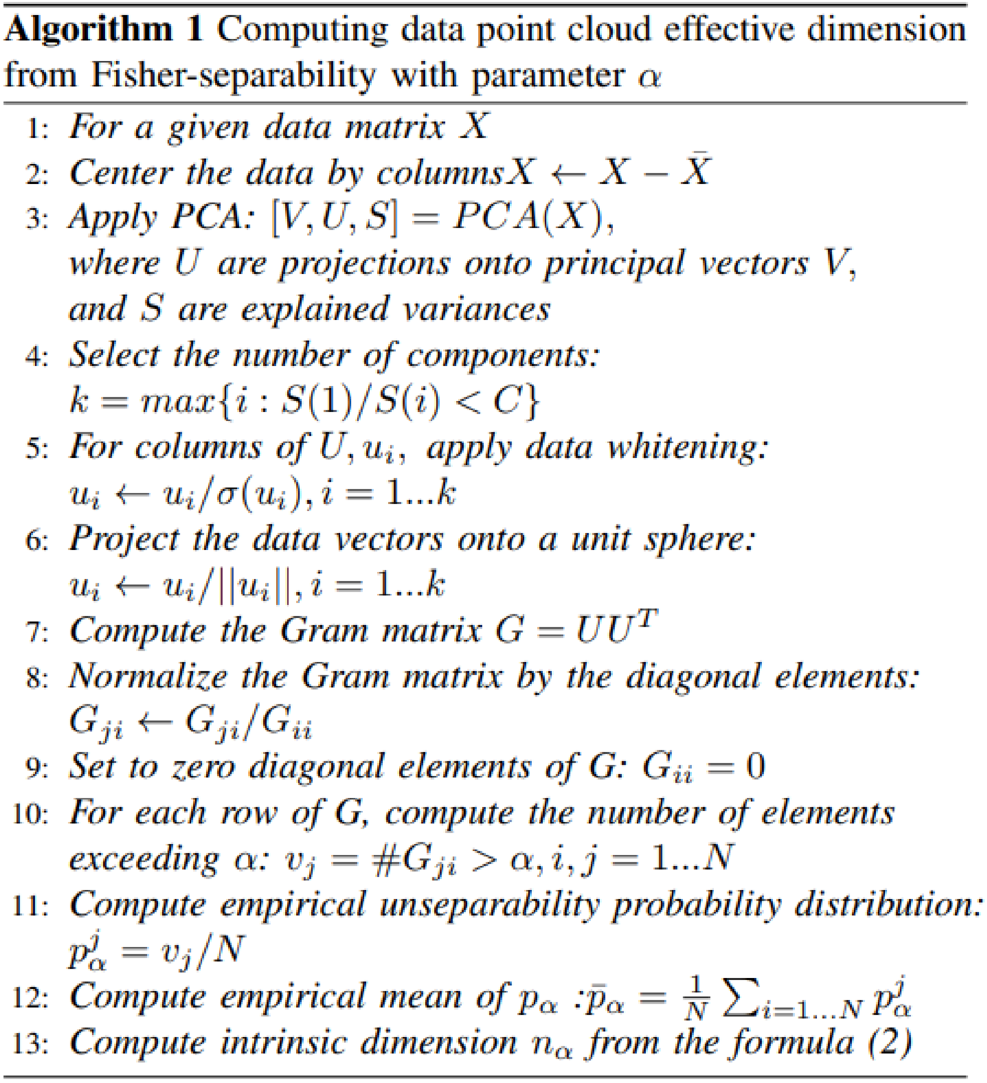
1) centering
2) projecting onto the linear subspace spanned by first k principal components, where k may be relatively large
3) whitening (i.e., applying a linear transformation after which the covariance matrix becomes the identity matrix)
4) normalising each vector to the unit length, which corresponds to the projection onto a unit sphere.

- Fisher Separability: A dataset X is termed Fisher-separable with parameter α for each point x ∈ X, it satisfies (x, y) ≤ α(x, x) for all y ∈ Y, where y is a set of points not equal to x. α ∈ [0, 1), quantifies deviation from perfect separability.

- Effective Dimension Estimation: The effective dimension nα is estimated by comparing the mean separability probability pα to the theoretical value pα bar for a uniform distribution on an n-sphere:

$$n_\alpha = \frac{W\left(\frac{-\ln(1-\alpha^2)}{2\pi\bar{p}_\alpha^2\alpha^2(1-\alpha^2)}\right)}{-\ln(1-\alpha^2)}$$

Where W is the Lambert function. This estimation helps quantify the effective dimensionality of the dataset under different separability conditions α.



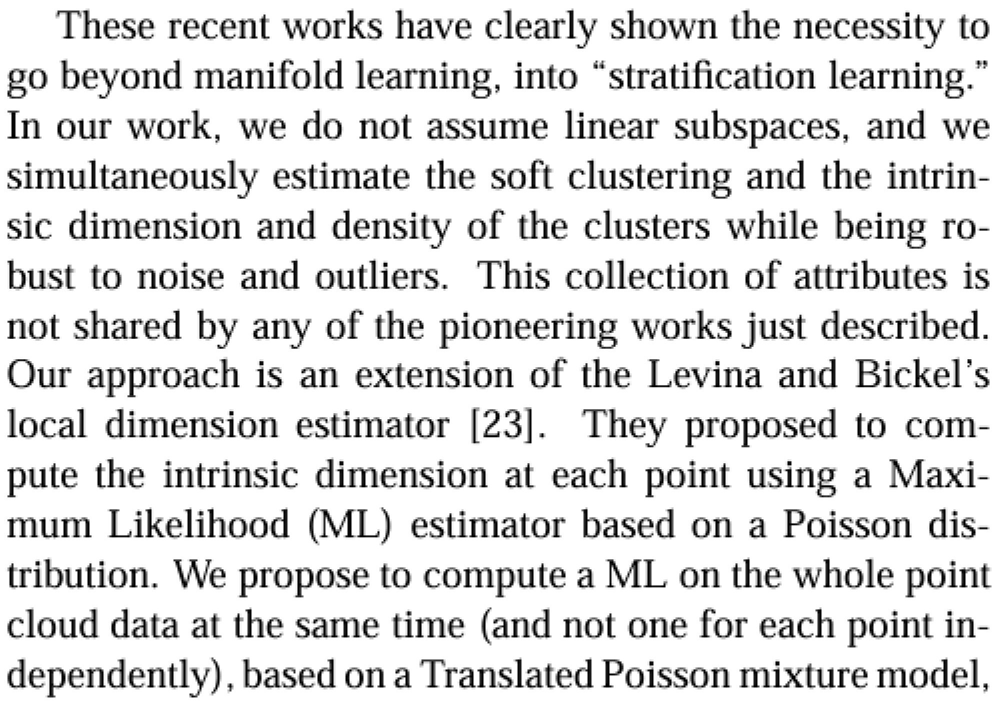
Algorithm 1 Computing data point cloud effective dimension from Fisher-separability with parameter α

```
For a given data matrix $X$
Center the data by columns$X \leftarrow X - \bar{X}$
Apply PCA: $[V, U, S] = PCA(X)$,
    where $U$ are projections onto principal vectors $V$,
    and $S$ are explained variances
Select the number of components:
    $k = max\{i : S(1)/S(i) < C\}$
For columns of $U, u_i$, apply data whitening:
    $u_i \leftarrow u_i/\sigma(u_i), i = 1...k$
Project the data vectors onto a unit sphere:
    $u_i \leftarrow u_i/||u_i||, i = 1...k$
Compute the Gram matrix $G = UU^T$
Normalize the Gram matrix by the diagonal elements:
    $G_{ji} \leftarrow G_{ji}/G_{ii}$
Set to zero diagonal elements of G: $G_{ii} = 0$
For each row of G, compute the number of elements
    exceeding α: $v_j = \#G_{ji} > \alpha, i, j = 1...N$
Compute empirical unseparability probability distribution:
    $p_\alpha^j = v_j/N$
Compute empirical mean of $p_\alpha$ :$\bar{p}_\alpha = \frac{1}{N}\sum_{i=1...N} p_\alpha^j$
Compute intrinsic dimension $n_\alpha$ from the formula (2)
```

- Finally, this method of ID estimation using Fisher separability falls under the typological classification of probabilistic and geometric methods.

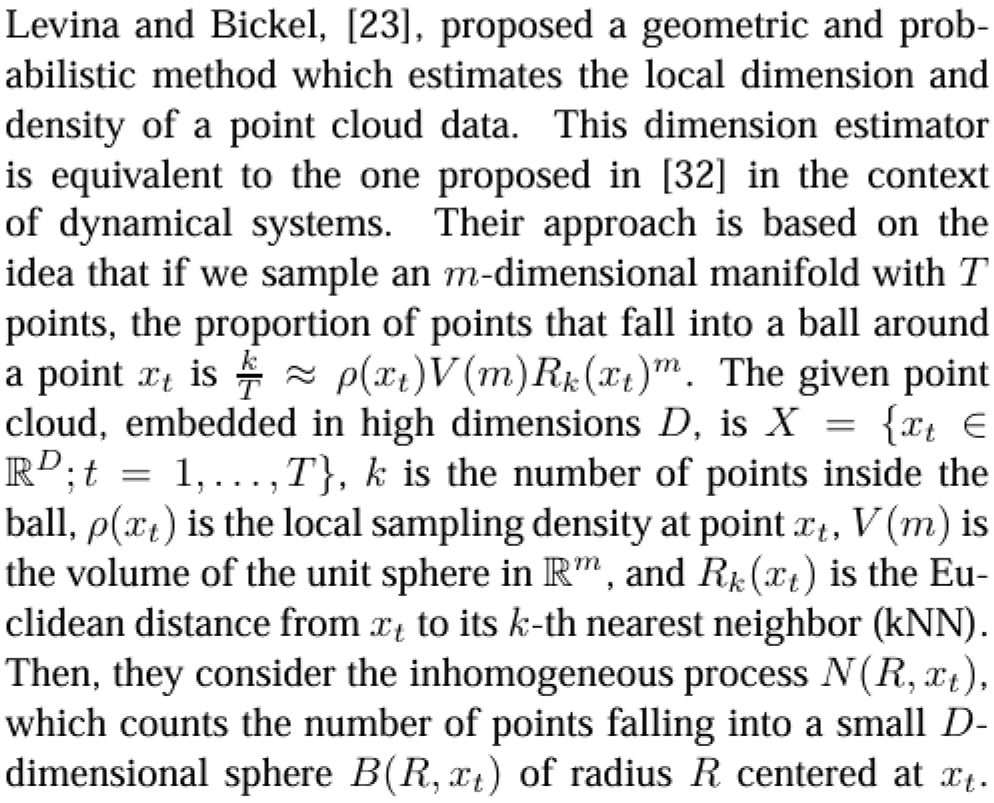
Maximum Likelihood Estimator

- Method for estimating varying dimensionality and density in noisy, high-dimensional data by modeling sample points with Translated Poisson mixtures. It uses regularization to handle noise and identifies mixtures of manifolds, allowing for classification based on both dimensionality and density characteristics.
- Main motivation: Common assumption is that all the points under analysis are samples of the same manifold and thus there is a unique intrinsic dimension. This is often not a correct assumption, which is addressed by MLE.
- Method does not assume linear subspaces, and simultaneously estimate the soft clustering and the intrinsic dimension and density of the clusters while being robust to noise and outliers.
- This method extends previous work focused on "calculating the intrinsic dimension at each point using a Maximum Likelihood (ML) estimator based on a Poisson distribution." Instead of computing ML estimates for each point individually, this approach applies an ML estimator across the entire point cloud simultaneously, utilizing a Translated Poisson mixture model. This model accounts for noise and allows for the identification of different classes, each characterized by its own dimensionality and sampling density.



These recent works have clearly shown the necessity to go beyond manifold learning, into "stratification learning." In our work, we do not assume linear subspaces, and we simultaneously estimate the soft clustering and the intrinsic dimension and density of the clusters while being robust to noise and outliers. This collection of attributes is not shared by any of the pioneering works just described. Our approach is an extension of the Levina and Bickel's local dimension estimator [23]. They proposed to compute the intrinsic dimension at each point using a Maximum Likelihood (ML) estimator based on a Poisson distribution. We propose to compute a ML on the whole point cloud data at the same time (and not one for each point independently), based on a Translated Poisson mixture model,

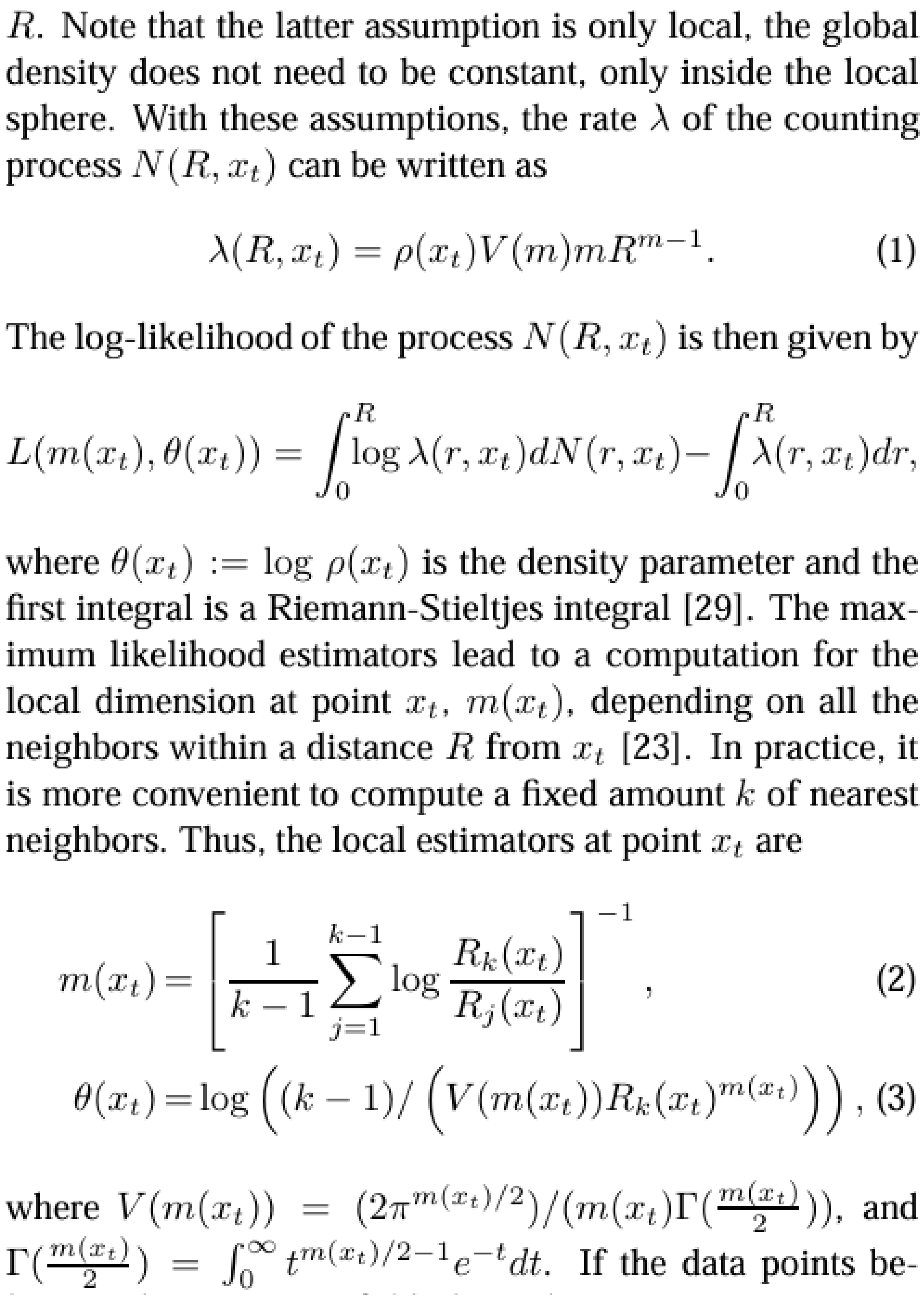
Previous work on Local intrinsic dimension estimation



Levina and Bickel, [23], proposed a geometric and probabilistic method which estimates the local dimension and density of a point cloud data. This dimension estimator is equivalent to the one proposed in [32] in the context of dynamical systems. Their approach is based on the idea that if we sample an m-dimensional manifold with T points, the proportion of points that fall into a ball around a point x_t is $\frac{k}{T} \approx \rho(x_t)V(m)R_k(x_t)^m$. The given point cloud, embedded in high dimensions D, is $X = \{x_t \in \mathbb{R}^D; t = 1, \ldots, T\}$, k is the number of points inside the ball, $\rho(x_t)$ is the local sampling density at point x_t, $V(m)$ is the volume of the unit sphere in $\mathbb{R}^m$, and $R_k(x_t)$ is the Euclidean distance from x_t to its k-th nearest neighbor (kNN). Then, they consider the inhomogeneous process $N(R, x_t)$, which counts the number of points falling into a small D-dimensional sphere $B(R, x_t)$ of radius R centered at x_t.

- Assumptions
  - if T → ∞, k → ∞, and k/T → 0, then the binomial process can be approximated by a Poisson process.
  - Density ρ(xt) is considered constant inside the sphere (global density does not need to be constant)

- Local estimator at point can be derived as



R. Note that the latter assumption is only local, the global density does not need to be constant, only inside the local sphere. With these assumptions, the rate λ of the counting process $N(R, x_t)$ can be written as

$$\lambda(R, x_t) = \rho(x_t)V(m)mR^{m-1}. \tag{1}$$

The log-likelihood of the process $N(R, x_t)$ is then given by

$$L(m(x_t), \theta(x_t)) = \int_0^R \log \lambda(r, x_t) dN(r, x_t) - \int_0^R \lambda(r, x_t) dr,$$

where $\theta(x_t) := \log\, \rho(x_t)$ is the density parameter and the first integral is a Riemann-Stieltjes integral [29]. The maximum likelihood estimators lead to a computation for the local dimension at point x_t, $m(x_t)$, depending on all the neighbors within a distance R from x_t [23]. In practice, it is more convenient to compute a fixed amount k of nearest neighbors. Thus, the local estimators at point x_t are

$$m(x_t) = \left[\frac{1}{k-1} \sum_{j=1}^{k-1} \log \frac{R_k(x_t)}{R_j(x_t)} \right]^{-1}, \tag{2}$$

$$\theta(x_t) = \log \left((k-1) / \left(V(m(x_t)) R_k(x_t)^{m(x_t)} \right) \right), \tag{3}$$

where $V(m(x_t)) = (2\pi^{m(x_t)/2})/(m(x_t)\Gamma(\frac{m(x_t)}{2}))$, and $\Gamma(\frac{m(x_t)}{2}) = \int_0^\infty t^{m(x_t)/2-1} e^{-t} dt$. If the data points be-

Limitations:

- Average over all local estimators is taken in order to obtain a more robust estimator. This assumes that data points belong to the same manifold.
- This does not hold true for more than one manifold with different dimensions.

- To address these limitations, the paper proposes:
    - Cluster according to dimensionality and then estimate the dimensionality for each cluster.
    - Include this in the process via the simultaneous soft clustering and estimation technique.

Proposed framework to handle noise: Translated Poisson model

Assumption: Each point is translated independently of the others and there are no deletions or insertions in the translation process.

process (these more general cases are also studied in [30]). We have the following critical theorem [30] which says that a translated Poisson process is also a Poisson process:

Theorem (Snyder & Miller [30]). *Let* $\{N(A)\colon A \subseteq X\}$ *be a Poisson process with an integrable intensity function* $\{\lambda(x)\colon x \in X\}$. *Points of this input point process are translated to the output space* Z *to form the output point process* $\{M(B)\colon B \subseteq Z\}$, *where each point is independently translated according to the transition density* $f(z|x)$. *Then, if there are no insertions and deletions,* $\{M(B)\colon B \subseteq Z\}$ *is a Poisson process with intensity*

$$\mu(z) = \int_X f(z|x)\lambda(x)dx.$$

Since the intensity of the Poisson process in our model is parametrized by the Euclidean distances of the points (and not by the points themselves, see previous Section), we are going to consider a random translation in the distances. This means that we do not observe the original distances but noisy distances. Let $f(s|r)$ be the transition density which defines the random process which translates a distance r in the input space to a distance s in the observable space. If $\lambda(r, x_t)$, defined in (1), is the local rate of the Poisson process which defines the counting process in the input space, then $\mu(s)$, the intensity of the Poisson process in the output space is given by

$$\mu(s, x_t) = \int_0^{R'} f(s|r)e^{\theta}V(m)mr^{m-1}dr. \tag{4}$$

R' is different from the radius R considered in the counting process $N(R, x_t)$. We consider $R' > R$ in (4) because, points originally at distance greater than R from x_t can be placed within a distance less than R after the translation process. In practice, the maximum translation is small (just a perturbation because of the noise) and we consider $R' = R + \sigma$ in the particular case of a Gaussian transition density (11). The log-likelihood of the translated Poisson process is

$$L(m(x_t), \theta(x_t)) = \int_0^R \log(\mu(s, x_t))dN(s, x_t) - \int_0^R \lambda(r, x_t)dr.$$

The parameters of the maximum log-likelihood are obtained by solving the system of equations $\partial L/\partial m = 0$ and $\partial L/\partial \theta = 0$. We then obtain the following expression for m when we use the k nearest neighbors (k-NN) instead of the points within distance less to R,

$$m(x_t)=\left[\frac{1}{k-1}\sum_{i=1}^{k-1}\frac{\int_0^{R'} f(R_i(x_t)|r)r^{m-1}\log\frac{R_k(x_t)}{r}dr}{\int_0^{R'} f(R_i(x_t)|r)r^{m-1}dr}\right]^{-1} \tag{5}$$

where, by an abuse of notation, we have identified $m = m(x_t)$ in the right hand side. Note that this expression reduces to the Levina and Bickel estimator [23] in the particular case that $f(s|r) = \delta(s-r)$, i.e., there is no translation of the original points. This corresponds to the ideal case with no noise.

Equation (5) is a nonlinear recursive expression in m which is difficult to solve. Thus, we are going to approximate it by an easier to compute closed expression. Since the translation density is modeling the effect of noise, the effective support of $f(s|r)$ is going to be concentrated around s. Then, we can substitute r^{m-1} in (5) by its Taylor expansion around R_i. Let us write (5) in the following way

$$m(x_t) = I^{-1} = \left[\frac{1}{k-1}\sum_{i=1}^{k-1} I_i\right]^{-1}, \tag{6}$$

and expand r^{m-1} in the integral I_i via its Taylor series

$$\begin{aligned}I_i &:= \frac{\int_0^{R'} f(R_i|r)r^{m-1}\log\frac{R_k(x_t)}{r}dr}{\int_0^{R'} f(R_i|r)r^{m-1}dr}\\ &= \frac{\int_0^{R'} f(R_i|r)\log\frac{R_k(x_t)}{r}dr + \Delta I_{N_i} + \ldots}{\int_0^{R'} f(R_i|r)dr + \Delta I_{D_i} + \ldots} = \frac{I_{N_i}}{I_{D_i}},\end{aligned}$$

where

$$\Delta I_{N_i} := (m-1)R_i^{-1}\int_0^{R'} f(R_i|r)(r-R_i)\log\frac{R_k(x_t)}{r}\,dr, \tag{7}$$

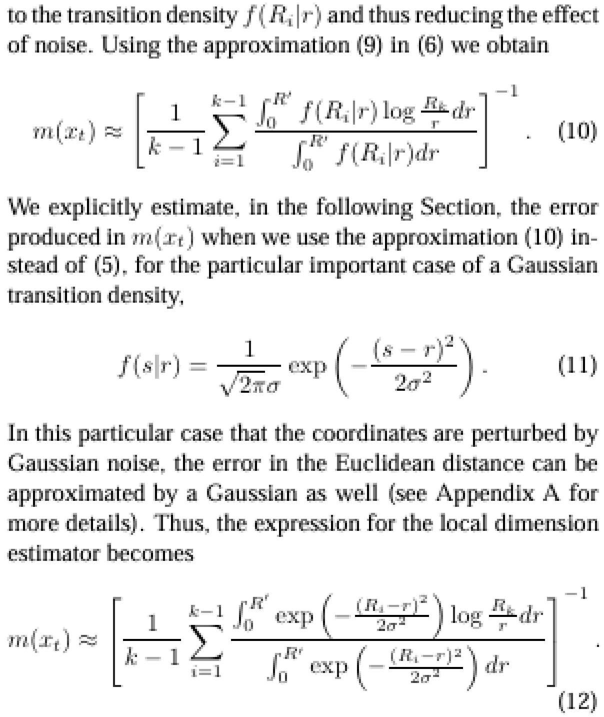
and

$$\Delta I_{D_i} := (m-1)R_i^{-1}\int_0^{R'} f(R_i|r)(r-R_i)dr. \tag{8}$$

These integrals are small since the effective support of $f(R_i|r)$ has the same order than the level of noise (considered not very large), and the quantity $(r-R_i)$ is small in the vicinity of R_i. We can then approximate

$$I_i \approx \frac{\int_0^{R'} f(R_i|r)\log\frac{R_k(x_t)}{r}dr}{\int_0^{R'} f(R_i|r)dr}. \tag{9}$$

Notice that with this approximation of I_i, the estimator (6) still reduces to the noise-free Levina-Bickel estimator (2), that is $I_i = \log\frac{R_k}{R_i}$, when $f(R_i|r) = \delta(R_i - r)$. In the more general case, (9) is the expected value of $\log\frac{R_k}{r}$ according



to the transition density $f(R_i|r)$ and thus reducing the effect of noise. Using the approximation (9) in (6) we obtain

$$m(x_t) \approx \left[\frac{1}{k-1}\sum_{i=1}^{k-1}\frac{\int_0^{R'} f(R_i|r)\log\frac{R_k}{r}dr}{\int_0^{R'} f(R_i|r)dr}\right]^{-1} . \quad (10)$$

We explicitly estimate, in the following Section, the error produced in $m(x_t)$ when we use the approximation (10) instead of (5), for the particular important case of a Gaussian transition density,

$$f(s|r) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(s-r)^2}{2\sigma^2}\right) . \quad (11)$$

In this particular case that the coordinates are perturbed by Gaussian noise, the error in the Euclidean distance can be approximated by a Gaussian as well (see Appendix A for more details). Thus, the expression for the local dimension estimator becomes

$$m(x_t) \approx \left[\frac{1}{k-1}\sum_{i=1}^{k-1}\frac{\int_0^{R'}\exp\left(-\frac{(R_i-r)^2}{2\sigma^2}\right)\log\frac{R_k}{r}dr}{\int_0^{R'}\exp\left(-\frac{(R_i-r)^2}{2\sigma^2}\right)dr}\right]^{-1} . \quad (12)$$

Translation Poisson Mixture Model (TPMM)

- Mixture model with J translated Poisson distributions is considered, each having different dimensions and density parameters.
- **Assumption of Levina-Bickel approach**: constant density within a local region
- **Assumption of TPMM**: constant density within each class.
- The algorithm clusters based on both dimension and density, and if the number of classes is insufficient, it may average the features within a class, leading to under-clustering. However, over-clustering is found to be robust, as extra classes typically remain empty or merge with existing ones.

The observable event is, as in the Levina-Bickel approach, the number of points inside the ball $B(R, x_t)$ of radius R centered at point x_t, denoted by $y_t = N(R, x_t)$. The total number of observations is T' and $Y = \{y_t; t = 1, \dots, T'\}$ is the observation sequence. Often, $T' \equiv T$, all points in the dataset are considered. Let us also denote by $p(\cdot)$ the probability density function and by $P(\cdot)$ the probability. The density function of the Poisson mixture model is given by

$$p(y_t|\psi) = \sum_{j=1}^{J} \pi^j p(y_t|\theta^j, m^j).$$

Since the observations follow a Poisson distribution, and we use the translated Poisson model introduced in the previous section, we have

$$p(y_t|\theta^j, m^j) = e^{\int_0^R \log \mu^j(s)\, dN(s,x_t)} e^{-\int_0^R \lambda^j(r) dr},$$

where $\lambda^j(r) = e^{\theta^j} V(m^j) m^j r^{m^j - 1}$ and $\mu^j(s) = \int_0^{R'} f(s|r) e^{\theta^j} V(m^j) m^j r^{m^j - 1} dr$. If Y contains T statistically independent variables (a standard assumption), then the probability density function of the observation sequence is the product of the individual probability densities, $p(y_t|\psi)$, and the log-likelihood is

$$L(Y|\psi) = \log p(Y|\psi) = \sum_{t=1}^{T} \log p(y_t|\psi). \qquad (14)$$

Let us consider the hidden-state information, that is, which mixture (or expert) generates each observation. We denote by $Z = \{z_t \in C; t = 1, \ldots, T\}$ the set of hidden variables and by $C = \{C^1, C^2, \ldots C^J\}$ the set of class labels. Then, $z_t = C^j$ means that the j-th mixture generates y_t. Using Z we can write the complete data log-likelihood as

$$\log p(Z, Y|\psi) = \sum_{t=1}^{T} \sum_{j=1}^{J} \delta_t^j \log \left[p(y_t|\psi^j)\pi^j \right], \qquad (15)$$

where a set of indicator variables δ_t^j, called membership functions, is used in order to indicate the status of the hidden variables:

$$\delta_t^j \equiv \delta(z_t, C^j) = \begin{cases} 1 & \text{if } z_t = C^j, \\ 0 & \text{otherwise.} \end{cases}$$

The unknown parameters in (15) are: The membership function of an expert (class), δ_t^j, the mixture probabilities, π^j, and the parameters of each expert, m^j and θ^j. Usually, problems involving a mixture of experts are solved by the Expectation Maximization (EM) algorithm [10] [21, Chap. 3]. The EM is based on the following decomposition of the log-likelihood (14):

$$\begin{aligned} L(Y|\psi, H) = & \sum_{t=1}^{T} \sum_{j=1}^{J} h^j(y_t) \log \left[p(y_t|\psi^j)\pi^j \right] \\ & - \sum_{t=1}^{T} \sum_{j=1}^{J} h^j(y_t) \log \left[h^j(y_t) \right], \end{aligned} \qquad (16)$$

where $H = \{h^j(y_t) \leq 1; t = 1, \ldots, T, j = 1, \ldots, J\}$ and $h^j(y_t)$ is the probability that observation t belongs to mixture j: $h^j(y_t) = E_Z[\delta_t^j|y_t, \psi] = P(\delta_t^j = 1|y_t, \psi)$, where $E_Z(\cdot)$ is the expectation with respect to Z. Since the membership functions are indicator variables, the first term in (16) is the expectation of (15) with respect to Z. Also notice that the second term is the entropy of the membership functions.

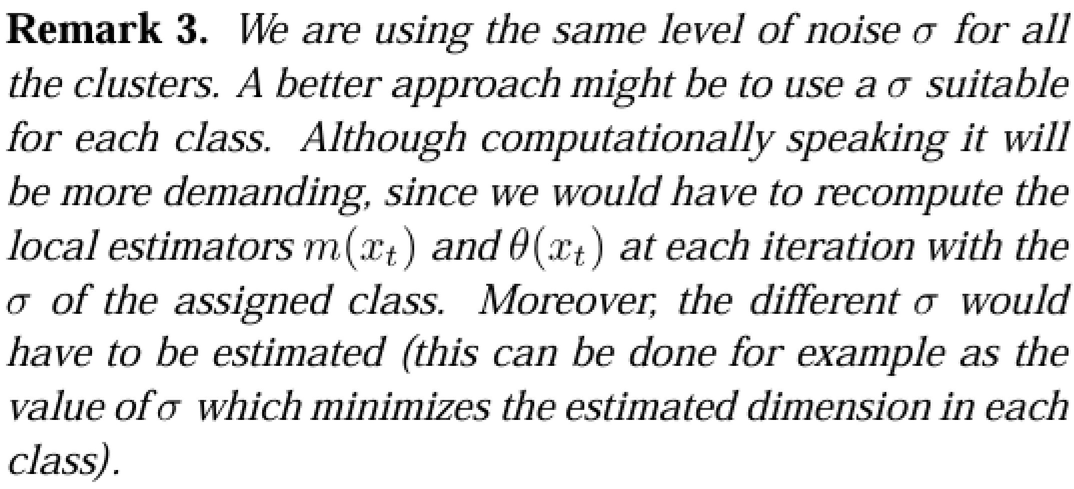
An interesting interpretation of the EM algorithm is introduced in [17], where the EM is seen as an alternate optimization algorithm of the log-likelihood (16). Then, the E-step is nothing else than the maximization of $L(Y|\psi, H)$ with respect to H with the additional constraint that $\sum_{j=1}^{J} h^j(y_t) = 1$ for each observation $t = 1, \ldots, T$. Thus, the variables $h^j(y_t)$ at step $n+1$ of the optimization algorithm are

$$h_{n+1}^j(y_t) = \frac{p(y_t|m_n^j, \theta_n^j)\pi_n^j}{\sum_{l=1}^{J} p(y_t|m_n^l, \theta_n^l)\pi_n^l}. \qquad (17)$$

In the same way, variables ψ are obtained by maximizing $L(Y|\psi, H)$ with respect to ψ with an additional constraint for the mixture probabilities: $\sum_{j=1}^{J} \pi^j{=}1$. This gives equations (21)-(23) for the variables at step $n+1$. In order

Regularized TMPP

- This is an extension of the TPMM (Translated Poisson Mixture Model) algorithm, introducing regularization to improve soft clustering based on dimensionality and density, particularly in noisy data and near manifold edges.
- This regularization adds a spatial term to the log-likelihood function, inspired by the Neighborhood EM (NEM) algorithm, which incorporates spatial constraints.
- The regularization is controlled by a parameter α, balancing the spatial term and likelihood. The algorithm uses a modified EM-type approach, incorporating a dissimilarity measure that considers neighborhood relationships between observations.
- An interesting remark observed here is:

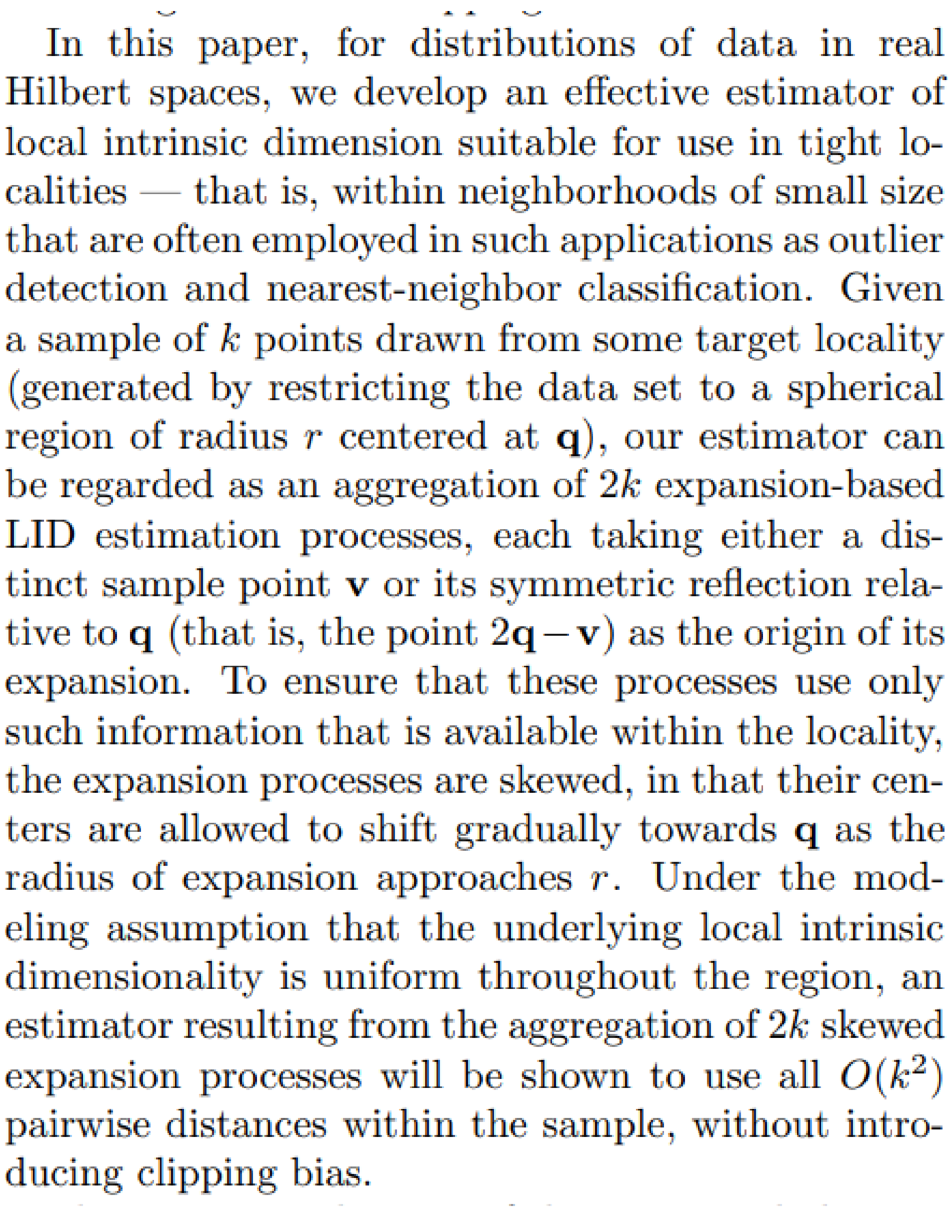


Remark 3. *We are using the same level of noise σ for all the clusters. A better approach might be to use a σ suitable for each class. Although computationally speaking it will be more demanding, since we would have to recompute the local estimators $m(x_t)$ and $\theta(x_t)$ at each iteration with the σ of the assigned class. Moreover, the different σ would have to be estimated (this can be done for example as the value of σ which minimizes the estimated dimension in each class).*

```
ENSURE: Regularized soft clustering according to dimensionality and density.

1. Compute the local estimators

   $m(x_t) = \left[ \frac{1}{k-1} \sum_{j=1}^{k-1} \frac{\int_0^{R'} f(R_i(x_t)|r) \log \frac{R_k(x_t)}{r} dr}{\int_0^{R'} f(R_i(x_t)|r) dr} \right]^{-1}$
   $\theta(x_t) = \log \left( (k-1) / \left( V(m(x_t)) R_k(x_t)^{m(x_t)} \right) \right)$

   In particular, we use the definition of $f$ given in (11).

2. Initialize $\psi_0 = \{\pi_0^j, m_0^j, \theta_0^j\}$ and $\bar{\psi}_0 = \{\bar{\pi}_0^j, \bar{m}_0^j, \bar{\theta}_0^j\}$ to any set of values which ensures that $\sum_j \pi_0^j = \sum_j \bar{\pi}_0^j = 1$ and $\bar{H}_0 = \{\bar{h}_0^j(y_t) = 1/J;\ j = 1, \ldots, J,\ t = 1, \ldots, T\}$.

3. Iterations on $l$,

   3A. If $l$ is odd
           Set $\bar{m}_l^j = m_0^j$ and $\bar{\theta}_l^j = \theta_0^j$, for all $j = 1, \ldots, J$.
     Else
           Set $\bar{\pi}_l^j = 1/J$, for all $j = 1, \ldots, J$.

   3B. Iterations on $n$,

      For all $j = 1, \ldots, J$:

      3B.1: Compute, for all $t = 1, \ldots, T$,

         $h_{n+1}^j(y_t) = \frac{p(y_t | m_n^j, \theta_n^j) \pi_n^j e^{-\alpha \mathcal{D}'(t,j,X,H_n)}}{\sum_{l=1}^J p(y_t | m_n^l, \theta_n^l) \pi_n^l e^{-\alpha \mathcal{D}'(t,l,X,H_n)}},$

         where $H_n = \{h_n^j(y_t);\ j = 1, \ldots, J,\ t = 1, \ldots, T\}$.

      3B.2: Compute

         $\pi_{n+1}^j = \frac{1}{T} \sum_{t=1}^T h_n^j(y_t)$    (21)

         $m_{n+1}^j = \left[ \sum_{t=1}^T h_n^j(y_t) m(x_t)^{-1} / \sum_{t=1}^T h_n^j(y_t) \right]^{-1}$    (22)

         $\rho_{n+1}^j = e^{\theta_{n+1}^j} = \left[ \sum_{t=1}^T h_n^j(y_t) f(x_t)^{-1} / \sum_{t=1}^T h_n^j(y_t) \right]^{-1}$    (23)

         where $\rho(x_t) = e^{\theta(x_t)}$.

      Until convergence of $\psi_n$, that is, when $||\psi_{n+1} - \psi_n||_2 < \epsilon$, for a certain small value $\epsilon$.

    Set $\bar{\psi}_{l+1} = \psi_n$ and $\bar{H}_{l+1} = H_n$.

    Until $||\bar{\psi}_{l+1} - \bar{\psi}_l||_2 < \epsilon$, $||\bar{H}_{l+1} - \bar{H}_l||_2 < \epsilon$ or $l = l_{\text{max}}$.[3]
```

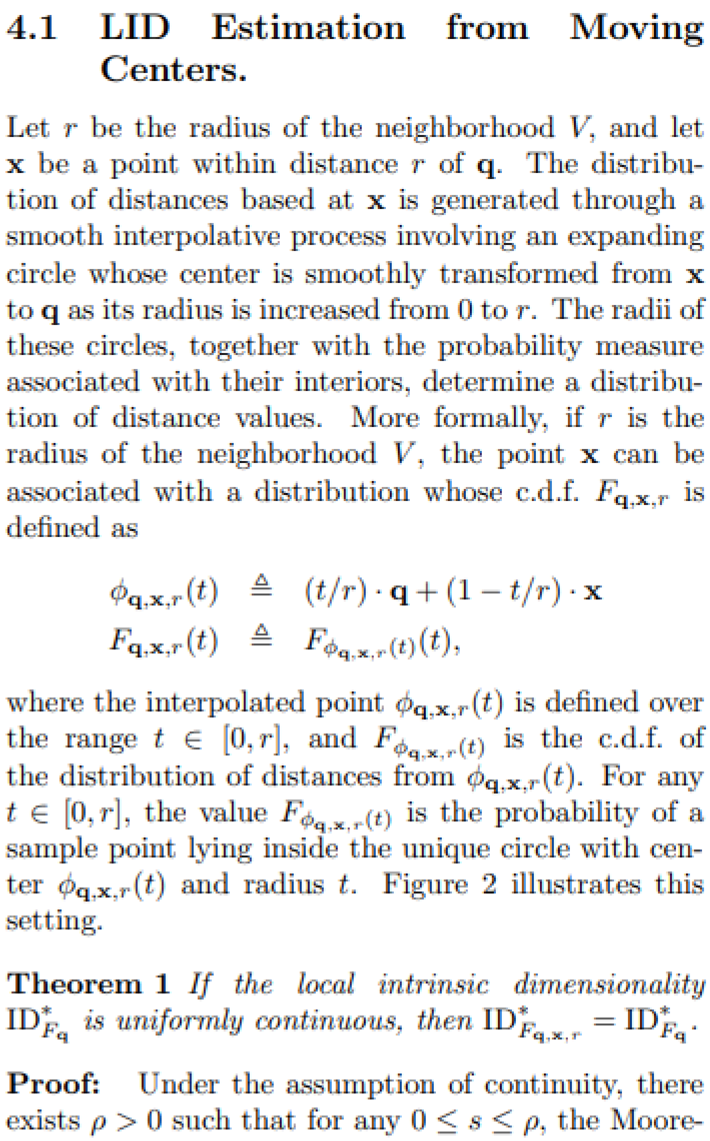
Tight Localities Estimation

- Method designed for small datasets, where traditional ID estimators, which require large sample sizes, may be ineffective.
- It estimates intrinsic dimensionality by applying MLE techniques to distances between pairs of points in small groups, based on the Local Intrinsic Dimension model.
- The estimator works by combining multiple expansion processes, each centered on either a sample point or its mirror image.
- These processes gradually shift their focus toward the center of the neighborhood as they expand, ensuring that they only use information available within the locality.
- The method takes into account all pairwise distances between the sample points, avoiding the introduction of bias that can occur in other methods.
- **Assumption:** intrinsic dimensionality is uniform across the region being analyzed.



In this paper, for distributions of data in real Hilbert spaces, we develop an effective estimator of local intrinsic dimension suitable for use in tight localities — that is, within neighborhoods of small size that are often employed in such applications as outlier detection and nearest-neighbor classification. Given a sample of k points drawn from some target locality (generated by restricting the data set to a spherical region of radius r centered at $\mathbf{q}$), our estimator can be regarded as an aggregation of $2k$ expansion-based LID estimation processes, each taking either a distinct sample point $\mathbf{v}$ or its symmetric reflection relative to $\mathbf{q}$ (that is, the point $2\mathbf{q}-\mathbf{v}$) as the origin of its expansion. To ensure that these processes use only such information that is available within the locality, the expansion processes are skewed, in that their centers are allowed to shift gradually towards $\mathbf{q}$ as the radius of expansion approaches r. Under the modeling assumption that the underlying local intrinsic dimensionality is uniform throughout the region, an estimator resulting from the aggregation of $2k$ skewed expansion processes will be shown to use all $O(k^2)$ pairwise distances within the sample, without introducing clipping bias.

Mathematical formulations



4.1 LID Estimation from Moving Centers.

Let r be the radius of the neighborhood V, and let $\mathbf{x}$ be a point within distance r of $\mathbf{q}$. The distribution of distances based at $\mathbf{x}$ is generated through a smooth interpolative process involving an expanding circle whose center is smoothly transformed from $\mathbf{x}$ to $\mathbf{q}$ as its radius is increased from 0 to r. The radii of these circles, together with the probability measure associated with their interiors, determine a distribution of distance values. More formally, if r is the radius of the neighborhood V, the point $\mathbf{x}$ can be associated with a distribution whose c.d.f. $F_{\mathbf{q},\mathbf{x},r}$ is defined as

$$\begin{aligned} \phi_{\mathbf{q},\mathbf{x},r}(t) &\triangleq (t/r)\cdot\mathbf{q} + (1-t/r)\cdot\mathbf{x} \\ F_{\mathbf{q},\mathbf{x},r}(t) &\triangleq F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}(t), \end{aligned}$$

where the interpolated point $\phi_{\mathbf{q},\mathbf{x},r}(t)$ is defined over the range $t \in [0,r]$, and $F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}$ is the c.d.f. of the distribution of distances from $\phi_{\mathbf{q},\mathbf{x},r}(t)$. For any $t \in [0,r]$, the value $F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}$ is the probability of a sample point lying inside the unique circle with center $\phi_{\mathbf{q},\mathbf{x},r}(t)$ and radius t. Figure 2 illustrates this setting.

Theorem 1 *If the local intrinsic dimensionality* $\mathrm{ID}^*_{F_\mathbf{q}}$ *is uniformly continuous, then* $\mathrm{ID}^*_{F_{\mathbf{q},\mathbf{x},r}} = \mathrm{ID}^*_{F_\mathbf{q}}$.

Proof: Under the assumption of continuity, there exists $\rho > 0$ such that for any $0 \leq s \leq \rho$, the Moore-

Osgood theorem implies that:

$$\begin{aligned}\mathrm{ID}^*_{F_{\mathbf{q}}} &= \lim_{t\to r^-} \mathrm{ID}^*_{F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}} \\ &= \lim_{t\to r^-} \lim_{s\to 0+} \mathrm{ID}_{F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}}(s) \\ &= \lim_{s\to 0^+} \lim_{t\to r^-} \mathrm{ID}_{F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}}(s) \,.\end{aligned}$$

The inner limit can be replaced by a subsequence of values of $\mathrm{ID}_{F_{\phi_{\mathbf{q},\mathbf{x},r}(t)}}(s)$, for choices of t that tend to r as s tends to zero. Here, we simply choose $t = r - s$. Noting that $\phi_{\mathbf{q},\mathbf{x},r}(r) = \mathbf{q}$, we obtain

$$\mathrm{ID}^*_{F_{\mathbf{q}}} \;=\; \lim_{s\to 0+} \mathrm{ID}_{F_{\phi_{\mathbf{q},\mathbf{x},r}(r-s)}}(s) \;=\; \mathrm{ID}^*_{F_{\mathbf{q},\mathbf{x},r}} \,.$$

□

Under the assumption of continuity, the local ID at q can therefore be estimated from the distribution $F_{\mathbf{q},\mathbf{x},r}$ for any location $\mathbf{x}$ falling within a sufficiently small neighborhood of $\mathbf{q}$. For the purpose of estimation, the distance value associated with a sample point $\mathbf{v} \in V$ is determined by the radius of the expanding circle at the time its boundary encounters $\mathbf{v}$, and not the actual distance from $\mathbf{x}$ to $\mathbf{v}$. This distance — which we will denote by $d_{\mathbf{q},r}(\mathbf{x},\mathbf{v})$ — is given by the value t such that

$$d_{\mathbf{q},r}(\mathbf{x},\mathbf{v}) \;\triangleq\; t \;=\; \|\phi_{\mathbf{q},\mathbf{x},r}(t) - \mathbf{v}\|.$$

We observe that when $\mathbf{v} = \mathbf{x}$, this expression reduces to $t = t\|\mathbf{q} - \mathbf{x}\|/r$, in which case either $t = 0$, or $\|\mathbf{q} - \mathbf{x}\| = r$ and t is indeterminate. For this reason, we deem $d_{\mathbf{q},r}(\mathbf{x},\mathbf{v})$ to be zero whenever $\mathbf{v} = \mathbf{x}$. Similarly, when $\mathbf{q} = \mathbf{x}$, the expression reduces to $t = \|\mathbf{q} - \mathbf{v}\|$, the distance that would be achieved if the expansion center were stationary at $\mathbf{q}$. When $\mathbf{v} = \mathbf{q}$, the expression yields $t = (1 - t/r)\|\mathbf{q} - \mathbf{x}\|$, in which case

$$t = \frac{r\|\mathbf{q} - \mathbf{x}\|}{r + \|\mathbf{q} - \mathbf{x}\|} \,.$$

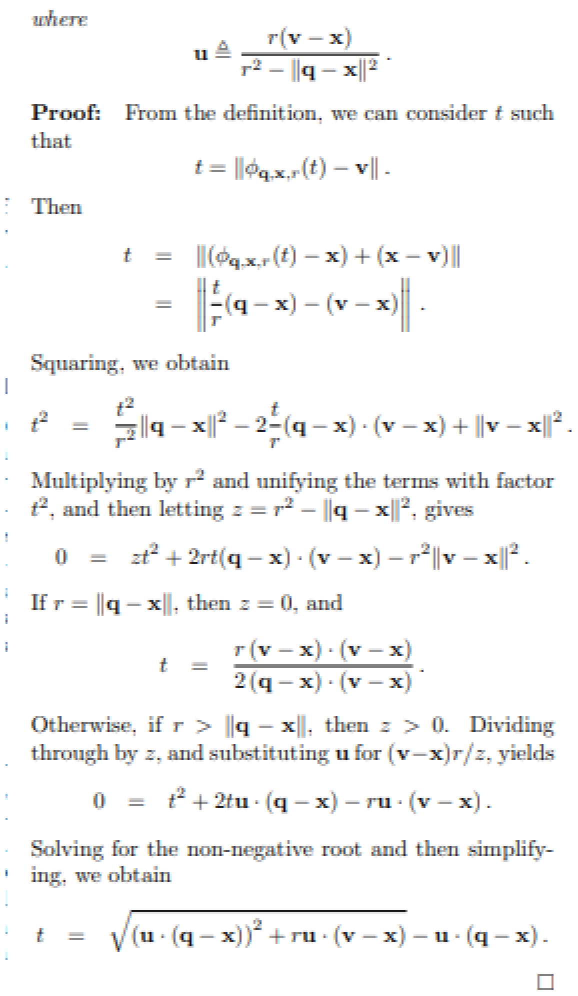
The remaining cases are accounted for by the following theorem.

Theorem 2 *Assume that* $\mathbf{q}$, $\mathbf{x}$ *and* $\mathbf{v}$ *are points where* $\mathbf{x} \neq \mathbf{q}$ *and* $\mathbf{x} \neq \mathbf{v}$. *Furthermore, assume that* $0 < \|\mathbf{q} - \mathbf{x}\| \leq r$ *and* $0 \leq \|\mathbf{q} - \mathbf{v}\| \leq r$ *for some positive radius* $r > 0$. *If* $\|\mathbf{q} - \mathbf{x}\| = r$, *then*

$$d_{\mathbf{q},r}(\mathbf{x},\mathbf{v}) = \frac{r\,(\mathbf{v} - \mathbf{x}) \cdot (\mathbf{v} - \mathbf{x})}{2\,(\mathbf{q} - \mathbf{x}) \cdot (\mathbf{v} - \mathbf{x})} \,.$$

Otherwise, if $\|\mathbf{q} - \mathbf{x}\| < r$, *then*

$$d_{\mathbf{q},r}(\mathbf{x},\mathbf{v}) = \sqrt{(\mathbf{u} \cdot (\mathbf{q} - \mathbf{x}))^2 + r\mathbf{u} \cdot (\mathbf{v} - \mathbf{x})} - \mathbf{u} \cdot (\mathbf{q} - \mathbf{x}) \,,$$



where

$$\mathbf{u} \triangleq \frac{r(\mathbf{v}-\mathbf{x})}{r^2 - \|\mathbf{q}-\mathbf{x}\|^2}.$$

Proof: From the definition, we can consider t such that

$$t = \|\phi_{\mathbf{q},\mathbf{x},r}(t) - \mathbf{v}\|.$$

Then

$$\begin{aligned} t &= \|(\phi_{\mathbf{q},\mathbf{x},r}(t) - \mathbf{x}) + (\mathbf{x}-\mathbf{v})\| \\ &= \left\|\frac{t}{r}(\mathbf{q}-\mathbf{x}) - (\mathbf{v}-\mathbf{x})\right\|. \end{aligned}$$

Squaring, we obtain

$$t^2 = \frac{t^2}{r^2}\|\mathbf{q}-\mathbf{x}\|^2 - 2\frac{t}{r}(\mathbf{q}-\mathbf{x})\cdot(\mathbf{v}-\mathbf{x}) + \|\mathbf{v}-\mathbf{x}\|^2.$$

Multiplying by r^2 and unifying the terms with factor t^2, and then letting $z = r^2 - \|\mathbf{q}-\mathbf{x}\|^2$, gives

$$0 = zt^2 + 2rt(\mathbf{q}-\mathbf{x})\cdot(\mathbf{v}-\mathbf{x}) - r^2\|\mathbf{v}-\mathbf{x}\|^2.$$

If $r = \|\mathbf{q}-\mathbf{x}\|$, then $z = 0$, and

$$t = \frac{r\,(\mathbf{v}-\mathbf{x})\cdot(\mathbf{v}-\mathbf{x})}{2\,(\mathbf{q}-\mathbf{x})\cdot(\mathbf{v}-\mathbf{x})}.$$

Otherwise, if $r > \|\mathbf{q}-\mathbf{x}\|$, then $z > 0$. Dividing through by z, and substituting $\mathbf{u}$ for $(\mathbf{v}-\mathbf{x})r/z$, yields

$$0 = t^2 + 2t\mathbf{u}\cdot(\mathbf{q}-\mathbf{x}) - r\mathbf{u}\cdot(\mathbf{v}-\mathbf{x}).$$

Solving for the non-negative root and then simplifying, we obtain

$$t = \sqrt{(\mathbf{u}\cdot(\mathbf{q}-\mathbf{x}))^2 + r\mathbf{u}\cdot(\mathbf{v}-\mathbf{x})} - \mathbf{u}\cdot(\mathbf{q}-\mathbf{x}).$$

□

- Previous work has done three types of measurements for LID estimation: non-central (using neighborhood samples), central (using the neighborhood center), and reflected (using the symmetric reflection of samples through the center).
- This method extends on previous work by introducing new estimation strategies that combine these measurements in different ways to improve the stability and accuracy of the LID estimation:

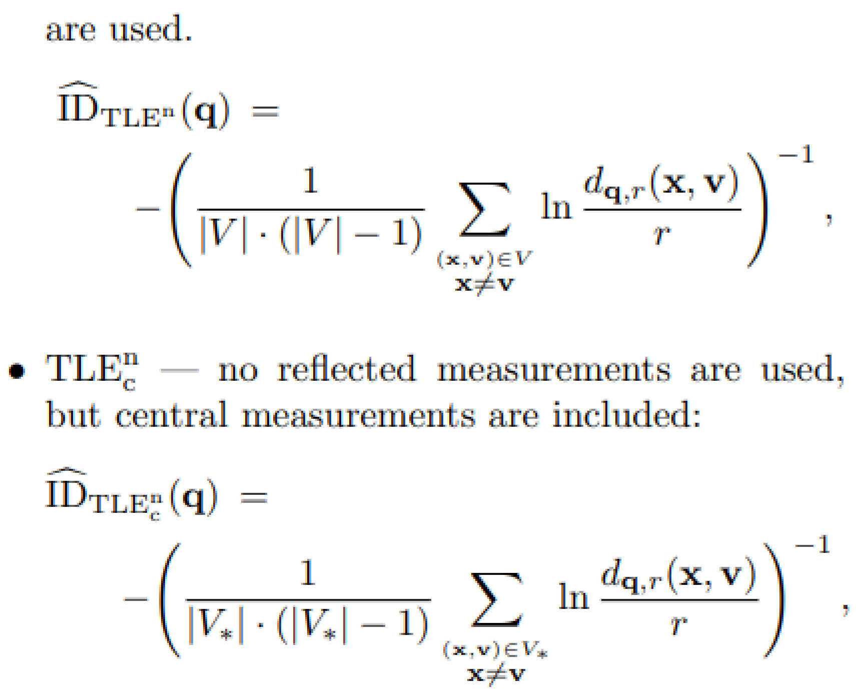
- TLE (the default estimator proposed in this paper) — reflected measurements are used, but central measurements are not:

  $$\widehat{\mathrm{ID}}_{\mathrm{TLE}}(\mathbf{q}) = -\left(\frac{1}{2|V|\cdot(|V|-1)}\cdot \sum_{\substack{(\mathbf{x},\mathbf{v})\in V\\ \mathbf{x}\neq\mathbf{v}}} \left[\ln\frac{d_{\mathbf{q},r}(\mathbf{x},\mathbf{v})}{r}+\ln\frac{d_{\mathbf{q},r}(\mathbf{2q}-\mathbf{x},\mathbf{v})}{r}\right]\right)^{-1}$$

- $\mathrm{TLE}_{\mathrm{c}}$ (the variant originally proposed in [3]) — central measurements are included together with reflected measurements:

  $$\widehat{\mathrm{ID}}_{\mathrm{TLE}_{\mathrm{c}}}(\mathbf{q}) = -\left(\frac{1}{2|V_*|\cdot(|V_*|-1)}\cdot \sum_{\substack{(\mathbf{x},\mathbf{v})\in V_*\\ \mathbf{x}\neq\mathbf{v}}} \left[\ln\frac{d_{\mathbf{q},r}(\mathbf{x},\mathbf{v})}{r}+\ln\frac{d_{\mathbf{q},r}(\mathbf{2q}-\mathbf{x},\mathbf{v})}{r}\right]\right)^{-1}$$

- $\mathrm{TLE}^{\mathrm{n}}$ — no reflected or central measurements



are used.

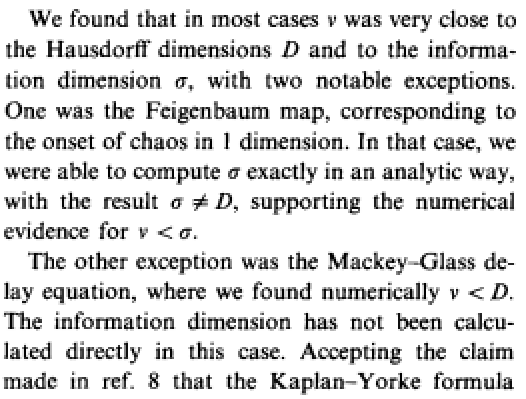
$$\widehat{\mathrm{ID}}_{\mathrm{TLE^n}}(\mathbf{q}) = -\left(\frac{1}{|V|\cdot(|V|-1)}\sum_{\substack{(\mathbf{x},\mathbf{v})\in V\\ \mathbf{x}\neq\mathbf{v}}} \ln\frac{d_{\mathbf{q},r}(\mathbf{x},\mathbf{v})}{r}\right)^{-1},$$

- TLE$^{\mathrm{n}}_{\mathrm{c}}$ — no reflected measurements are used, but central measurements are included:

$$\widehat{\mathrm{ID}}_{\mathrm{TLE^n_c}}(\mathbf{q}) = -\left(\frac{1}{|V_*|\cdot(|V_*|-1)}\sum_{\substack{(\mathbf{x},\mathbf{v})\in V_*\\ \mathbf{x}\neq\mathbf{v}}} \ln\frac{d_{\mathbf{q},r}(\mathbf{x},\mathbf{v})}{r}\right)^{-1},$$

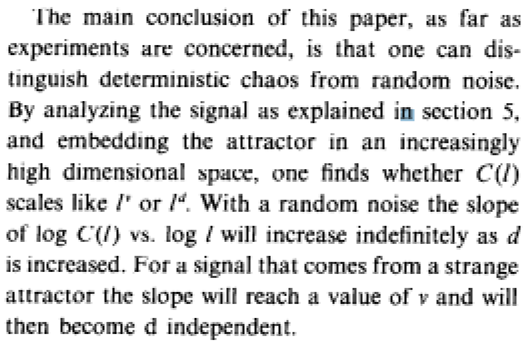
Correlation Integral

- The method uses the correlation exponent v to analyze strange attractors in chaotic systems.
- v is closely related to other measures of an attractor's local structure and can be calculated efficiently using time series data, even with moderate-length series.
- Compared to previous box-counting methods, their approach is more efficient and avoids systematic errors due to finite iterations.
- v often closely approximates other dimensions like the Hausdorff dimension D and the information dimension α, although notable exceptions are identified, such as in the Feigenbaum map and the Mackey-Glass delay equation.
- Correlation exponent can effectively distinguish between deterministic chaos and random noise by analyzing how the correlation integral C(l) scales with l in different embedding dimensions.



We found that in most cases ν was very close to the Hausdorff dimensions D and to the information dimension σ, with two notable exceptions. One was the Feigenbaum map, corresponding to the onset of chaos in 1 dimension. In that case, we were able to compute σ exactly in an analytic way, with the result $\sigma \neq D$, supporting the numerical evidence for $\nu < \sigma$.

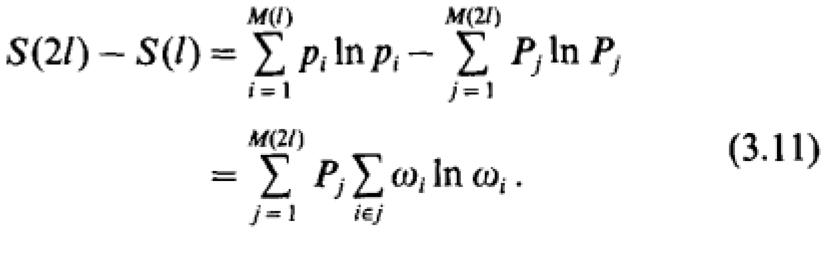
The other exception was the Mackey–Glass delay equation, where we found numerically $\nu < D$. The information dimension has not been calculated directly in this case. Accepting the claim made in ref. 8 that the Kaplan–Yorke formula



The main conclusion of this paper, as far as experiments are concerned, is that one can distinguish deterministic chaos from random noise. By analyzing the signal as explained in section 5, and embedding the attractor in an increasingly high dimensional space, one finds whether $C(l)$ scales like l^ν or l^d. With a random noise the slope of log $C(l)$ vs. log l will increase indefinitely as d is increased. For a signal that comes from a strange attractor the slope will reach a value of ν and will then become d independent.

- Assumption in finding relationships between v, σ and D:

$$\frac{C(l)}{C(2l)} = \frac{\sum_j P_j^2 \sum_{i\in j} \omega_i^2}{\sum_j P_j^2}, \qquad (3.10)$$



$$S(2l) - S(l) = \sum_{i=1}^{M(l)} p_i \ln p_i - \sum_{j=1}^{M(2l)} P_j \ln P_j$$

$$= \sum_{j=1}^{M(2l)} P_j \sum_{i\in j} \omega_i \ln \omega_i \, . \qquad (3.11)$$

In order to estimate eq. (3.10) in terms of eq. (3.11), we have to introduce a new assumption. We assume that the ω_i's are distributed independently of the P_j. This means essentially that locally the attractor looks the same in regions where it is rather dense (P_j large) as in regions where P_j is small. Although we cannot further justify this assumption, it seems to us very natural. It leads immediately to

$$\frac{C(l)}{C(2l)} = \frac{\langle \omega^2 \rangle}{\langle \omega \rangle} = 2^D \langle \omega^2 \rangle \,, \tag{3.12}$$

and to

$$S(2l) - S(l) = 2^D \langle \omega \ln \omega \rangle \,. \tag{3.13}$$