Efficient nearest-neighbor search and its applications for nonlinear time series analysis

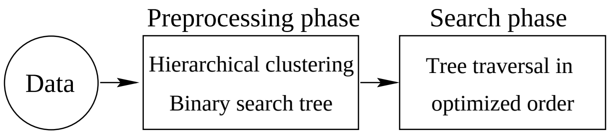
Jörg D. Wichard, Christian Merkwirth, Ulrich Parlitz and Werner Lauterborn
Drittes Physikalisches Institut, Georg-August-Universität Göttingen,
D-37073 Göttingen, Germany

Problems of nearest-neighbor search

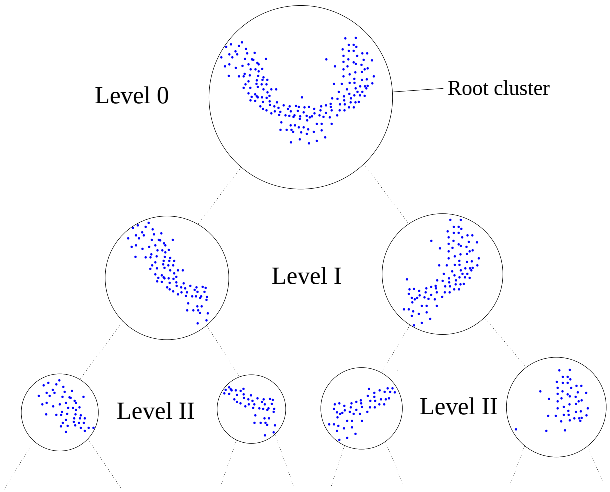
- Complexity increases with the number of data points
- Naive implementations are time consuming for large data sets

Hierarchical clustering scheme

During the *preprocessing phase*, a database (cluster tree) is created which accelerates finding nearest neighbors in the *search phase*.



Building a typical cluster tree:



Search efficiency

We define the *search efficiency* as:

$$\frac{\text{Total number of points}}{\text{Average number of distance calculations per query}}$$

The *search efficiency* of a brute force implementation is always 1, the *search efficiency* of this algorithm is $O(N/\log(N))$.

Applications of nearest-neighbor search

- Modelling and prediction of time series
- Fast correlation sum computation (dimension analysis, generalized mutual information etc.)
- Lyapunov exponents
- Nonlinear noise reduction
- Recurrence plots
- Continuity test for mappings

High dimensional queries

Time series of length 100000 from the Lorenz system , the Hénon map and Gaussian noise are embedded in 30 dimensions with lag 1 and a histogram of distances is computed between 100 reference points and the rest of the dataset. On large scales the dimension of the Hénon attractor is over estimated, because of a strongly folded structure of the high-dimensional reconstruction. Continuous systems on the other hand can be sampled with arbitrary small sampling times which avoids strong folding in the reconstruction. Therefore, the correct dimension becomes visible already on medium scales as can be seen for the Lorenz data.