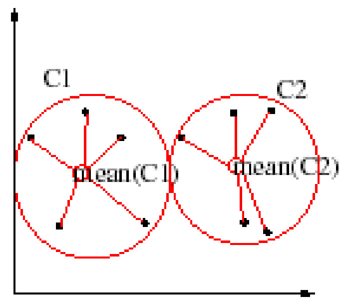
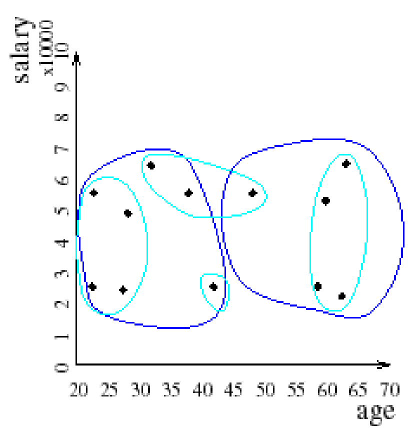
K-means and K-medoids algorithms

- Minimizes the sum of square distances of points to cluster representative
- Efficient iterative algorithms (O(n))



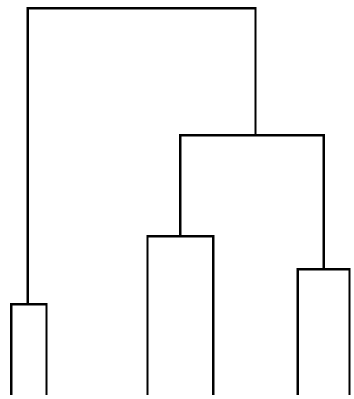
Problems with K-means type algorithms

- Clusters are approximately spherical
- High dimensionality is a problem
- The value of K is an input parameter



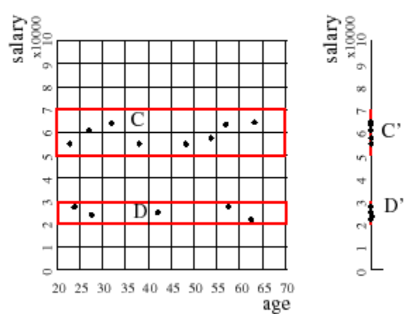
Hierarchical Clustering

- Quadratic algorithms
- Running time can be improved using sampling [Guha et al, SIGMOD 1998] [Kollios et al, ICDE 2001]



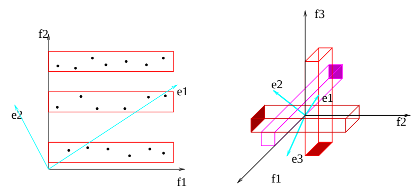
Density Based Algorithms

- Clusters are regions of space which have a high density of points
- Clusters can have arbitrary shapes



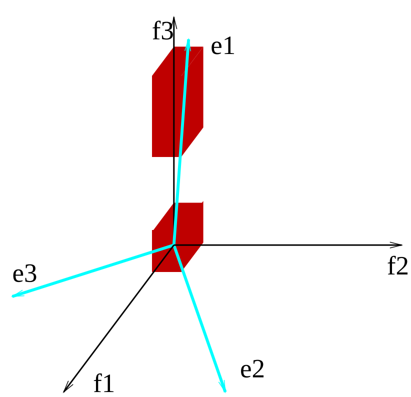
Dimensionality Reduction

- Reduce the dimensionality of the space, while preserving distances
- Many techniques (SVD, MDS)
- May or may not help



Dimensionality Reduction

- Example: SVD decomposition



Speeding up the clustering algorithms: Data Reduction

- Data Reduction:
  - approximate the original dataset using a small representation
  - ideally, the representation must be stored in main memory
  - summarization, compression
- The accuracy loss must be as small as possible.
- Use the approximated dataset to run the clustering algorithms

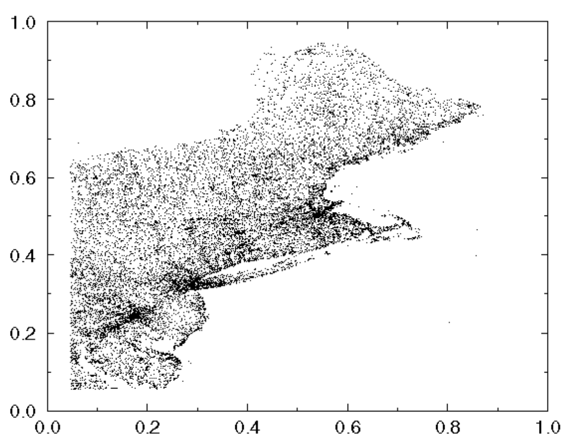
Random Sampling as a Data Reduction Method

- Random Sampling is used as a data reduction method
- Idea: Use a random sample of the dataset and run the clustering algorithm over the sample
- Used for clustering and association rule detection [Ng and Han 94][Toivonen 96][Guha et al 98]
- But:
  - For datasets that contain clusters with different densities, we may miss some sparse ones
  - For datasets with noise we may include significant amount of noise in our sample

A better idea: Biased Sampling

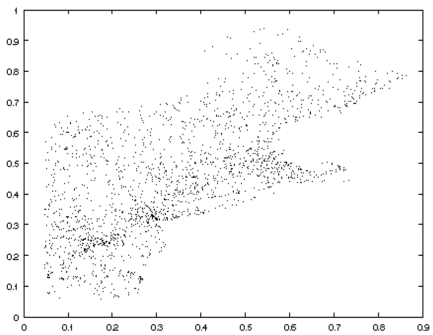
- Use biased sampling instead of random sampling
- In biased sampling, the prob that a point is included in the sample depends on the local density
- We can oversample or undersample regions in our datasets depending on the DM task at hand

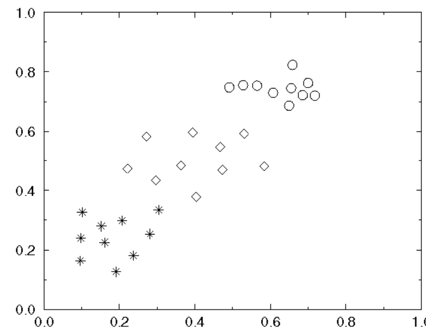
Example: NorthEast Dataset



NorthEast Dataset, 130K postal addresses in North Eastern USA

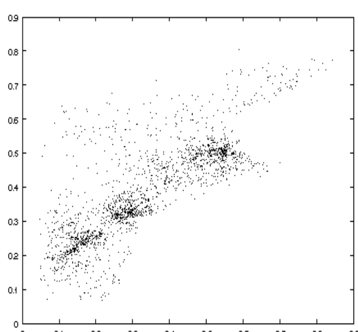
Random Sample

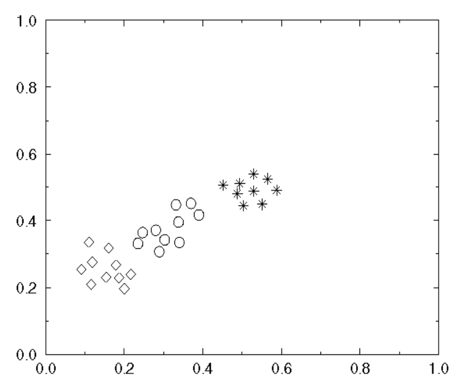




Random Sampling fails to find the clusters

Biased Sampling





Biased Sampling finds the clusters

The Biased Sampling Technique

- Basic idea:
  - First compute an approximation of the density function of the dataset
  - Use the density function to define the bias for each point and perform the sampling

  [Kollios et al, ICDE 2001]

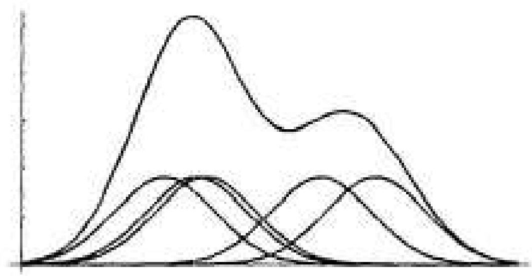
  [Domeniconi and Gunopulos, ICML 2001]

  [Palmer and Faloutsos, SIGMOD 2000]

Density Estimation

- We use kernels to approximate the probability density function (pdf)
- We scan the dataset and we compute an initial random sample and standard deviation
- For each sample we use a kernel. The approximate pdf is the sum of all kernels

Kernel Estimator



Example of a Kernel Estimator

The sampling step

- Let ***f(p)*** the pdf value for the point $p \in D$
  $p = (x_1, x_2, \ldots, x_d)$

- We define $L(p) = f(p)^{\alpha}$, where α is parameter
- We compute the normalization parameter k (in one scan):

$$k = \sum_{p \in D} L(p)$$

The sampling step (cont.)

- The sampling bias is proportional to:

$$\frac{b}{k} L(\mathbf{p})$$

Where b is the size of the sample and k the normalization factor

- In another scan we perform the sampling (two scans)
- We can combine the above two steps into one scan

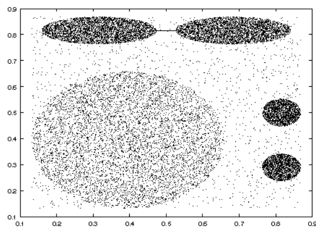
The variable α

- If $\alpha = 0$ then we have uniform random sampling bias: $\frac{b}{n}$
- If $\alpha > 0$ then regions with higher density are sampled at a higher rate
- If $\alpha < 0$ then regions with higher density are sampled at a lower rate

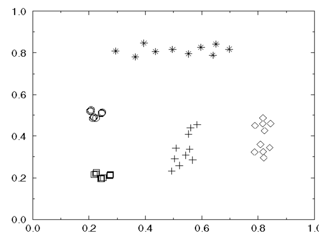
Bias ~ $\frac{b}{k} f(\mathbf{p})^{a}$

- We can show that if $\alpha > -1$, relative densities are preserved in the sample

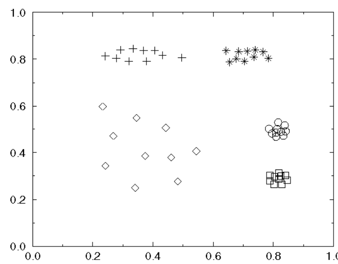
Biased vs Uniform random sampling



DataSet 5 clusters



With 1000 Uniform RS



With 1000 Biased RS, a=-0.5