Fast low-rank metric learning

Dan-Theodor Oneaţă

July 25, 2011

Neighbourhood component analysis

- Learns a Mahalanobis-like distance metric $\mathbf{S}$:

$$d_{\mathbf{S}}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}} \mathbf{S} (\mathbf{x}_i - \mathbf{x}_j)}.$$

- Equivalent to a linear transformation: $\mathbf{S} = \mathbf{A}^{\mathrm{T}}\mathbf{A} \Rightarrow$

$$d_{\mathbf{S}}(\mathbf{x}_i, \mathbf{x}_j) = d_{\mathbf{I}}(\mathbf{A}\mathbf{x}_i, \mathbf{A}\mathbf{x}_j).$$

- Objective function: expected number of correctly classified points $f(\mathbf{S}) = \sum_i p_i$, where $p_i = \frac{\sum_{j \in C_i} \exp\{-d_{\mathbf{S}}(\mathbf{x}_i, \mathbf{x}_j)\}}{\sum_k \exp\{-d_{\mathbf{S}}(\mathbf{x}_i, \mathbf{x}_k)\}}$.

Positive things:

- Improves classification
- Dimensionality reduction
- Assumption-free.

Possible issues:

- $f(\mathbf{S})$ is not convex
- Gradient is expensive to evaluate $\mathcal{O}(N^2 D^2)$
- How to do classification.

Sub-sampling

- Easy to implement.
- Reduces the general cost: $\mathcal{O}(n^2 D^2)$.

Mini-batches

1. Randomly selected mini-batches
2. Batches formed by clustering:
   - Farthest-point clustering
   - Recursive projection clustering
   - Agglomerative clustering.

Stochastic learning

- Instead of updating $\mathbf{A}$ after one sweep through the entire data set, update it more often.

Approximate computations & k-d trees

NCA as kernel denisity estiamtion

Exact computations & compact support kernels

Next...