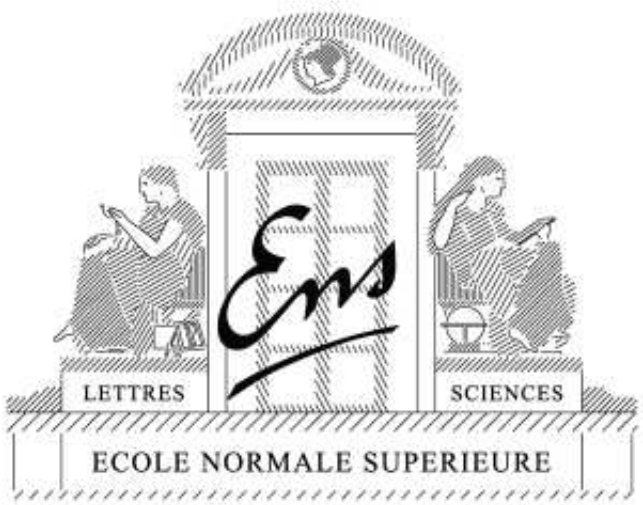
DIFFRAC : a discriminative and flexible framework for clustering

Francis Bach
INRIA - Ecole Normale Supérieure

Zaïd Harchaoui
Telecom Paris



NIPS - November 2007

Summary

- **Discriminative clustering** = find labels that optimize linear separability
- **Square loss** for classification = cost function in closed form
- Optimization of the labels by **convex relaxation**
- Efficient optimization algorithm by **partial dualization**
- Application in **semi-supervised learning**

Classification with square loss

- n points $x_1, \ldots, x_n$ in $\mathbb{R}^d$, represented in a matrix $X \in \mathbb{R}^{n \times d}$.
- Labels $=$ partitions of $\{1, \ldots, n\}$ into $k > 1$ clusters, represented by *indicator matrices*

$$y \in \{0,1\}^{n \times k} \text{ such that } y 1_k = 1_n$$

- Regularized linear regression problem of y given X :

$$J(y, X, \kappa) = \min_{w \in \mathbb{R}^{d \times k},\ b \in \mathbb{R}^{1 \times k}} \frac{1}{n} \|y - Xw - 1_n b\|_F^2 + \kappa \ \text{tr} \ w^\top w,$$

  - Multi-label classification problems with square loss functions
  - Solution in closed form (with $\Pi_n = I_n - \frac{1}{n} 1_n 1_n^\top$):

$$w^* = (X^\top \Pi_n X + n\kappa I_n)^{-1} X^\top \Pi_n y \quad \text{and} \quad b^* = \frac{1}{n} 1_n^\top (y - X w^*)$$

Discriminative clustering cost

- **Discriminative clustering** consists in finding labels such that they lead to best linear separation by a discriminative classifier (Xu et al., 2004, 2005)
- Use square loss for multi-class classification
- Main advantages
  - minimizing the regularized cost in closed form
  - including a bias term by simply centering the data
- Optimal value equal to $J(y, X, \kappa) = \operatorname{tr}\ yy^\top A(X, \kappa)$, where

$$A(X,\kappa) = \frac{1}{n}\Pi_n(I_n - X(X^\top \Pi_n X + n\kappa I)^{-1}X^\top)\Pi_n$$

Diffrac

- Optimization problem: minimize $\operatorname{tr} yy^\top A(X, \kappa)$ with respect to y (indicator matrices)
- The cost function only involves the matrix $M = yy^\top \in \mathbb{R}^{n \times n}$ (= k-class equivalence matrix)
- Convex outer approximation for M
  - M is positive semidefinite (denoted as $M \succcurlyeq 0$)
  - the diagonal of M is equal to 1_n (denoted as $\operatorname{diag}(M) = 1_n$)
  - if M corresponds to at most k clusters, we have $M \succcurlyeq \frac{1}{k} 1_n 1_n^\top$
- Convex set:

$$\mathcal{C}_k = \{M \in \mathbb{R}^{n \times n},\ M = M^\top,\ \operatorname{diag}(M) = 1_n,\ M \geqslant 0,\ M \succcurlyeq \tfrac{1}{k} 1_n 1_n^\top\}$$

Minimum cluster sizes

- Avoid trivial solution by imposing a minimum size λ_0 for each cluster, through:
  - **Row sums**: $M1_n \geqslant \lambda_0 1_n$ and $M1_n \leqslant (n-(k-1)\lambda_0)1_n$ (same constraint as Xu et al., 2005).
  - **Eigenvalues**: The sizes of the clusters are exactly the k largest eigenvalues of $M \Rightarrow$ constraint equivalent to $\sum_{i=1}^{n} 1_{\lambda_i(M)\geqslant\lambda_0} \geqslant k$, where $\lambda_1(M), \ldots, \lambda_n(M)$ are the n eigenvalues of M.
    - Non convex constraint
    - Relaxed as $\sum_{i=1}^{n} \phi_{\lambda_0}(\lambda_i(M)) \geqslant k$, where $\phi_{\lambda_0}(\kappa) = \min\{\kappa/\lambda_0, 1\}$

- **Final convex relaxation**: minimize $\operatorname{tr} A(X,\kappa)M$ such that $M = M^\top,\ \operatorname{diag}(M) = 1_n,\ M \geqslant 0,\ M \succcurlyeq \frac{1}{k}1_n 1_n^\top,\ \sum_{i=1}^{n} \phi_{\lambda_0}(\lambda_i(M)) \geqslant k$

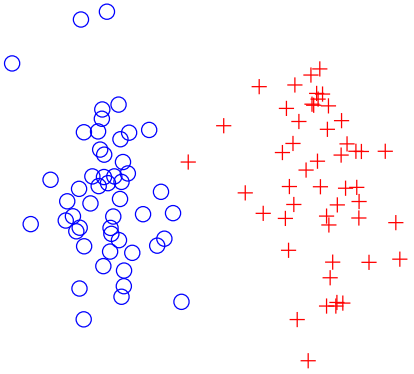
Comparison with K-means

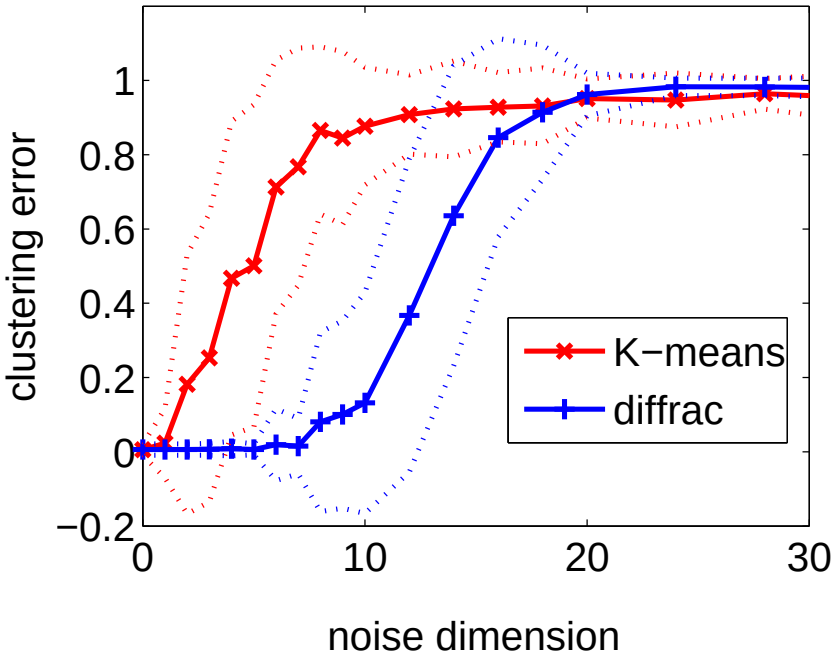
- **DIFFRAC** ($\kappa = 0$): minimize

$$\text{tr } \Pi_n (I_n - X(X^\top \Pi_n X)^{-1} X^\top) \Pi_n y y^\top$$

- **K-Means**: minimize (Zha et al., 2002, Bach & Jordan, 2004)

$$\min_{\mu \in \mathbb{R}^{k \times d}} \|X - y\mu\|_F^2 = \text{tr}(I_n - y(y^\top y)^{-1} y^\top)(\Pi_n X)(\Pi_n X)^\top$$





Kernels

- The matrix $A(X, \kappa)$ can be expressed only in terms of the Gram matrix $K = XX^\top$.

$$\boxed{A(K, \kappa) = \kappa \Pi_n (\widetilde{K} + n\kappa I_n)^{-1} \Pi_n}$$

  where $\widetilde{K} = \Pi_n K \Pi_n$ is the "centered Gram matrix" of the points X.

- Additional relaxation to kernel PCA:

  1. relaxing the constraints $M \succcurlyeq \frac{1}{k} 1_n 1_n^\top$ into $M \succcurlyeq 0$
  2. relaxing $\operatorname{diag}(M) = 1_n$ into $\operatorname{tr} M = n$
  3. removing the constraint $M \geqslant 0$ and the constraints on the row sums.

- Important constraint: $\operatorname{diag}(M) = 1$

Optimization by partial dualization - I

- Optimization problem:

$$\min \operatorname{tr} AM \quad \text{such that} \quad \begin{array}{l|r} M = M^\top,\ M \succcurlyeq 0,\ \operatorname{tr} M = n & \\ \Phi_{\lambda_0}(M) \geqslant k & \\ \operatorname{diag}(M) = 1_n & \beta_1 \\ M1_n \leqslant (n-(k-1)\lambda_0)1_n,\ M1_n \geqslant \lambda_0 1_n & \beta_2, \beta_3 \\ M \geqslant 0 & \beta_4 \\ M \succcurlyeq \frac{1_n 1_n^\top}{k} & \beta_5, \beta_6 \end{array}$$

- Partial dualization of constraints
  - Kept constraints lead to simple **spectral problem**

Optimization by partial dualization - II

- Lagrangian equal to $\operatorname{tr} B(\beta)M - b(\beta)$ with

$$B(\beta) = A + \operatorname{Diag}(\beta_1) - \tfrac{1}{2}(\beta_2 - \beta_3)1^\top - \tfrac{1}{2}1(\beta_2 - \beta_3)^\top - \beta_4 + \tfrac{1}{2}\frac{\beta_5\beta_5^\top}{\beta_6}$$

$$b(\beta) = \beta_1^\top 1 - (n - (k-1)\lambda_0)\beta_2^\top 1 + \lambda_0\beta_3^\top 1 + k\beta_6/2 + \beta_5^\top 1,$$

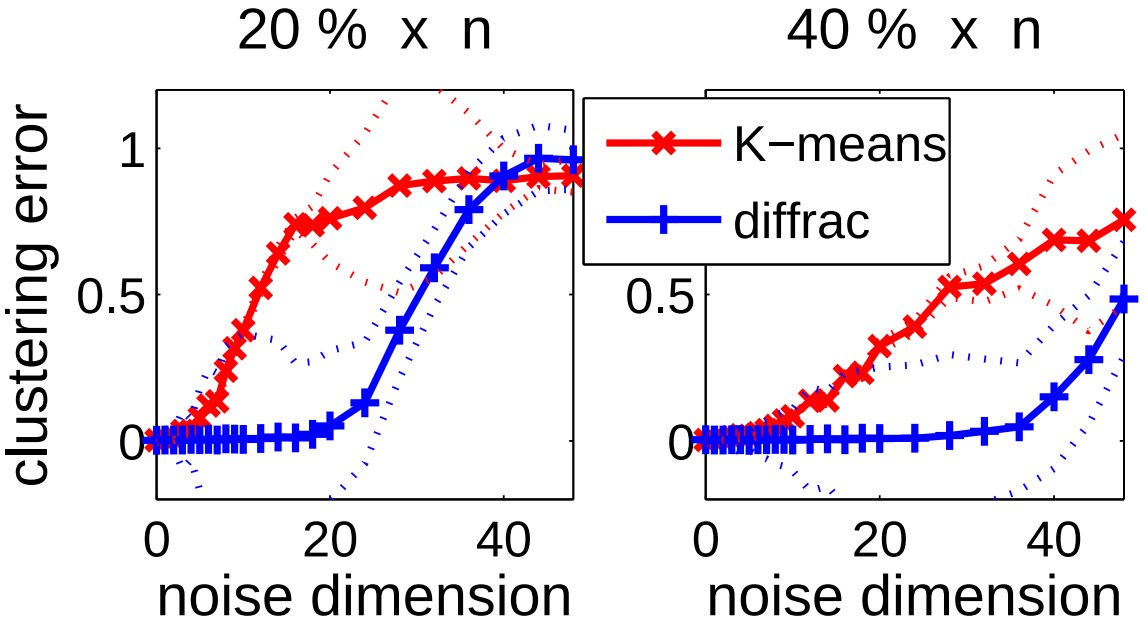
- Primal variable M, dual variables β_1, β_2, β_3, β_4, (β_5, β_6)
- **Dual problem**: $\max_{\beta} \left\{ \min_{M \succcurlyeq 0, \operatorname{tr} M = n, \Phi_{\lambda_0}(M) \geqslant k} \operatorname{tr} B(\beta)M - b(\beta) \right\}$
- Minimization with respect to M leads to convex non differentiable spectral function in β
- Maximization with respect to β by projected subgradient or projected gradient (after smoothing)

Computational complexity - Rounding

- Constant times the matrix-vector operation with the matrix A
- **Linear complexity** in the number n of data points.
- For linear kernels with dimension d: $O(d^2 n)$
- For general kernels: $O(n^3)$ or $O(m^2 n)$ using an incomplete Cholesky decomposition of rank m
- Rounding
  - After the convex optimization, we obtain a low-rank matrix $M \in \mathcal{C}_k$ which is pointwise nonnegative with unit diagonal
  - Spectral clustering algorithm on the matrix M (Ng & al., 2001)
  - NB : Diffrac works better than doing spectral clustering on A or K!

Semi-supervised learning

- Equivalence matrices M allows simple inclusion of prior knowledge (Xu et al., 2004, De Bie and Cristianini, 2006)
- "must-link" constraints (positive constraints) : $M_{ij} = 1$
  - With a square loss $\Rightarrow$ equivalent to grouping into chuncks
- "must-not-link" constraints (negative constraints) : $M_{ij} = 0$



Simulations

- Clustering classification datasets
  - Performance measured by clustering error between 0 and $100(k-1)$
  - Comparison with K-means and RCA (Bar-Hillel et al., 2003)

Dataset	K-means	DIFFRAC	RCA
Mnist-linear 0%	$\mathbf{5.6 \pm 0.1}$	6.0 ± 0.4	
Mnist-linear 20%	4.5 ± 0.3	3.6 ± 0.3	$\mathbf{3.0 \pm 0.2}$
Mnist-linear 40%	2.9 ± 0.3	2.2 ± 0.2	$\mathbf{1.8 \pm 0.4}$
Mnist-RBF 0%	5.6 ± 0.2	$\mathbf{4.9 \pm 0.2}$	
Mnist-RBF 20%	4.6 ± 0.0	$\mathbf{1.8 \pm 0.4}$	4.1 ± 0.2
Mnist-RBF 40%	4.9 ± 0.0	$\mathbf{0.9 \pm 0.1}$	2.9 ± 0.1
Isolet-linear 0%	$\mathbf{12.1 \pm 0.6}$	12.3 ± 0.3	
Isolet-linear 20%	10.5 ± 0.2	$\mathbf{7.8 \pm 0.8}$	9.5 ± 0.4
Isolet-linear 40%	9.2 ± 0.5	$\mathbf{3.7 \pm 0.2}$	7.0 ± 0.4
Isolet-RBF 0%	11.4 ± 0.4	$\mathbf{11.0 \pm 0.3}$	
Isolet-RBF 20%	10.6 ± 0.0	$\mathbf{7.5 \pm 0.5}$	7.8 ± 0.5
Isolet-RBF 40%	10.0 ± 0.0	$\mathbf{3.7 \pm 1.0}$	6.9 ± 0.6

Simulations

- Semi-supervised classification
- Diffrac works with any amount of supervision
  - Diffrac works with any amount of supervision
  - Comparison with LDS (Chapelle & Zien, 2004)