Advanced Machine Learning: from Theory to Practice
Lecture 6
Graphs in Machine Learning

F. d'Alché-Buc and E. Le Pennec

Fall 2016

Introduction

Graphs and Machine Learning

- Graphs as data
  - web, social networks, biological networks, wireless network, molecules, sensor network (IOT)...
  - Recommendation system, Link prediction, Activity prediction
- Data as graphs Today's course
  - data defined by a similarity or affinity matrix
  - use elements of graph theory to achieve clustering, semi-supervised learning, transductive learning

Application to :

- Clustering in unsupervised learning
- Semi-supervised and transductive learning

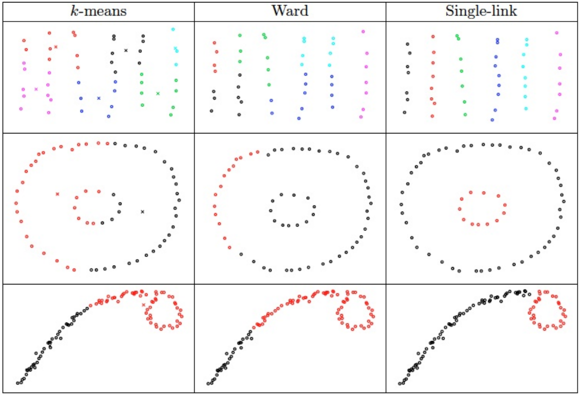
Unlabeled data

- Available data are unlabeled : documents, webpages, clients database ...
- Labeling data is expensive and requires some expertise

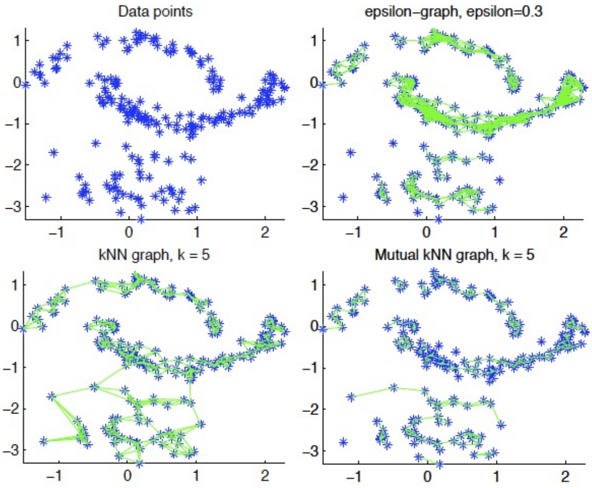
Learning from unlabeled data

- Modeling probability distribution $\rightarrow$ graphical models
- Dimension reduction $\rightarrow$ pre-processing for pattern recognition
- **Clustering** : group data into homogeneous clusters $\rightarrow$ organize your data, make easier access to them, pre and post processing, application in segmentation, document retrieval, bioinformatics ...

Clustering
Different clusterings



Spectral clustering
From data to graphs



Credits :

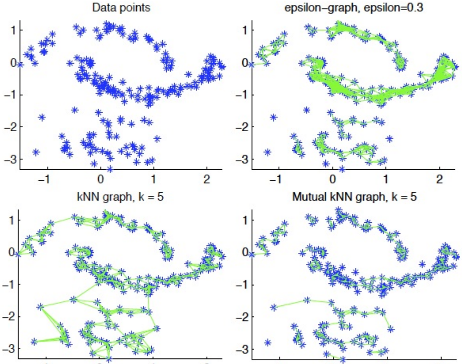
Image : U. V .Luxburg.

- Data $x_1, \ldots, x_n$ with their similarity values $s_{ij} \geq 0$ or with their distance d_{ij} values
- Build a graph $G = (V, E)$
- V : set of vertices. A vertex v_i corresponds to data x_i
- E : set of edges. An edge links two nodes if x_i and x_j are close according to the ε-graph method or the k-nn method
- W : adjacency matrix = binary symmetric matrix
- Definition : $w_{ij} = 1$ if there is an edge between node v_i and node v_j, 0 otherwise.

Spectral clustering
Graph construction

Several ways to construct it :

- ε-graph : connect all points whose pairwise distance is at most ε (alt. whose pairwise similarity is at least ε
- k-nearest-neighbor-graph : connect v_i and v_j if x_i is among the k-nearest-neighbors of x_j OR x_i is among the k-nearest-neighbours of x_j

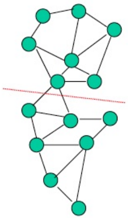


Notations : A and B are two disjoint subsets of the nodes set V that form a partition

- $cut(A, B) = \sum_{t \in A, u \in B} w_{t,u}$
- $vol(A) = \sum_{t \in A, u \in V} w_{t,u}$
- $|A| =$ nb of edges

Mincut problem

- $Cut(A, \bar{A}) = \sum_{i \in A, j \in \bar{A}} w_{ij}$
- Let $f_i \in \{-1, 1\}$ be the index class of x_i
- Clustering :=Find $(f_1, \dots, f_n) \in \{-1, 1\}$ such that $Cut(A, \bar{A})$ is minimized.



For sake of simplicity : $B = \bar{A}$. Ratiocut :

$$\text{Ratiocut}(A, B) = \frac{cut(A, B)}{|A|} + \frac{cut(B, A)}{|B|}$$

Normalized cut

$$\text{Ncut}(A, B) = \frac{cut(A, B)}{vol(A)} + \frac{cut(B, A)}{vol(B)}$$

Some references :

- Courses/slides : Dan Spielman (Godel prize in 2015), Yale, Link
- Spectral Graph Theory, Fan R. K. Chung, Published by AMS , 1997, Link

Definitions

- W matrix : adjacency matrix
- Degree matrix D : $d_{ii} = \sum_j w_{ij}$, if $i \neq j$, $d_{ij} = 0$
- Unnormalized Graph Laplacian : $L = D - W$
- Normalized Graph Laplacians : $L_{sym} = D^{-1/2}(D - W)D^{-1/2}$, $L_{rw} = D^{-1/2}(D - W)$.

Eigenvalue/eigenvectors

1. L is a symmetric and positive semi-definite matrix
2. Vector 1_n is a eigenvector of L with eigenvalue 0.

Proof :
1.

$$
\begin{aligned}
f^T L f &= f^T (D - W) f \\
&= f^T D f - f^T W f \\
&= \sum_i d_i f_i^2 - \sum_{ij} w_{ij} f_i f_j \\
&= \frac{1}{2}\left(\sum_i d_i f_i^2 - 2 \sum_{ij} w_{ij} f_i f_j + \sum_j d_j f_j^2\right) \\
&= \frac{1}{2} \sum_{i,j} w_i j (f_i - f_j)^2
\end{aligned}
$$

2. We notice that : $(D - W)1_n = 0$.

Connected components

- The multiplicity of the smallest eigenvalue (0) of L is the number of connected components in the graph

$$L = \begin{pmatrix} L_1 & & & \\ & L_2 & & \\ & & \ddots & \\ & & & L_k \end{pmatrix}$$

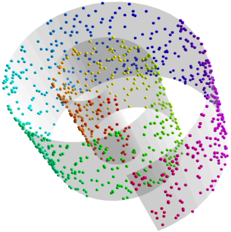
The normalized Laplacians satisfy :

1. For every $f \in \mathbb{R}^n$, $f^T L_{sym} f = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}})^2$.$\lambda$ is an eigenvalue of L_{rw} with eigenvector u iff λ is an eigenvalue of L_{sym} with eigenvector : $v = D^{1/2}u$.

3. λ is an eigenvalue of L_{rw} with eigenvector u iff λ and u solve the generalized eigen problem : $Lu = \lambda Du$.
4. 0 is an eigenvalue of L_{rw} with the constant vector 1_n. 0 is an eigenvalue of L_{sym} with eigenvector $D^{1/2}1$.

A function $f : V \to \mathbb{R}$.
Smoothness of the graph function :

$$\|f\|_L^2 = f^T L f = \sum_{i,j} w_{ij}(f_i - f_j)^2$$



Manifold $\mathcal{M}$: topological space that locally resembles Euclidean space near each point.

More generally, measure of the smoothness of a function on a manifold :

$$\|f\|_{\mathcal{M}}^2 = \int_{\mathcal{M}} ||\nabla_{\mathcal{M}} f(x)\|^2 p(x) dx$$

- $f_i, i = 1, \dots, n$: membership of data i to cluster 1
- $f_i = 1$ if $x_i \in A$, -1 otherwise Cluster 2 (B)

Balanced Mincut problem

Find $f \in \{-1, 1\}^n$ that minimizes $J(f) = \sum_{i \in A, j \in B} w_{ij}$ such that $|A| = |B|$

Notice that $|A| = |B| \iff \sum_{i=1}^n f_i = 0$ (as many 1's than -1's).
$\sum_{i=1}^n f_i = 0 \iff f \perp 1_n$.

Now $f \in \mathbb{R}^n$

$$\begin{aligned} J(f) &= \sum_{i,j=1}^{n} w_{ij} = \frac{1}{4}\sum_{i,j} w_{ij}(f_i - f_j)^2 \\ &= \frac{1}{4}\sum_{i,j} w_{ij}(f_i^2 + f_j^2 - 2f_i f_j) \\ &= \frac{1}{2} f^T (D - W) f \end{aligned}$$

Constraints :

- Avoiding trivial solution : $f \perp 1_n$
- Controlling the complexity of f (ℓ_2 regularization) : $\sum_i f_i^2 = n$

$\min_{f \in \mathbb{R}^n} f^T L f$
subject to : $f \perp 1$, $||f|| = \sqrt{n}$

- Solve the previous relaxed problem $\rightarrow$ the vector corresponding to the second smallest eigenvalue is solution
- To be convinced : write the First Order Conditions to solve the optimization problem
- Threshold the values of f to get discrete values1 and -1 OR use 2-means (better).

Algorithm

- Solve the previous relaxed problem $\rightarrow$ take the k eigenvectors $(v_2, \ldots, v_{k+1})$ corresponding to the k smallest positive eigenvalues except 0
- Represent your data in the new space spanned by these k vectors : form the matrix V with the v_k's as column vectors
- each row of V represents an individual
- Apply k-means in the k-dimensional space

- Relaxation of Ratiocut
- Relaxation of Mincut

$$\begin{aligned} Ratiocut(A, B) &= \frac{cut(A,B)}{|A|} + \frac{cut(B,A)}{|B|} \\ &= cut(A,B)(\frac{1}{|A|} + \frac{1}{|B|}) \end{aligned}$$

Define (1) :

if $v_i \in A,\ f_i = \sqrt{\frac{|B|}{|A|}}$.

if $v_i \in B$, $f_i = -\frac{\sqrt{|A|}}{\sqrt{|B|}}$

$$
\begin{aligned}
f^T L f &= \frac{1}{2}\sum_{i,j} w_{ij}(f_i - f_j)^2 \\
&= \frac{1}{2}\sum_{i\in A, j\in B} w_{ij}\left(\sqrt{\frac{|B|}{|A|}} + \sqrt{\frac{|A|}{|B|}}\right)^2 + \frac{1}{2}\sum_{i\in B, j\in A}\left(-\sqrt{\frac{|A|}{|B|}} - \sqrt{\frac{|B|}{|A|}}\right)^2 \\
&= cut(A,B)\left(\frac{|B|}{|A|} + \frac{|A|}{|B|} + 2\right) \\
&= cut(A,B)\left(\frac{|A|+|B|}{|A|} + \frac{|A|+|B|}{|B|}\right) \\
&= |V|\, ratiocut(A,B)
\end{aligned}
$$

We have also :

- f as defined for Ratiocut satisfies : $\sum_i f_i = 0$
- $lf \ \ l^2 = n$

Altogether :

Approximating Ratiocut

$\min_f f^T L f$, s.t. $f \perp 1, \|fl = n$

- Normalized cut (avoid isolated subset) :
  $Ncut(A,B) = \frac{cut(A,B)}{vol(A)} + \frac{cut(B,A)}{vol(B)}$
- $f_i = \sqrt{\frac{vol(B)}{vol(A)}}, if \, v_i \in A, \sqrt{\frac{vol(A)}{vol(B)}}, if \, v_i \in B.$
- Notice that :
  - $vol(V) = f^T Df.$
  - $(Df)^T 1 = 0$
  - $f^T Lf = vol(V) Ncut(A,B)$

$\min_{f \in \mathbb{R}^n} \frac{f^T L f}{f^T D f}$
subject to : $f^T D 1_n = 0$

$\min_{f \in \mathbb{R}^n} \frac{f^T L f}{f^T D f}$
subject to : $f^T D 1_n = 0$

Solve the generalized eigenvalue problem :
$(D - W)f = \lambda D f$ which can be re-written as
$D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}} z = \lambda z$
with $z = D^{-\frac{1}{2}} f$.

The problem boils down to find second eigenvector of L_{sym}.

- Importance of the initial graph : several ways to construct it (k-neighbors)
- Able to extract clusters on a manifold
- Consistency (U. Von Luxburg)
- Stability
- Model selection : eigengap

Spectral clustering

Eigengap heuristic

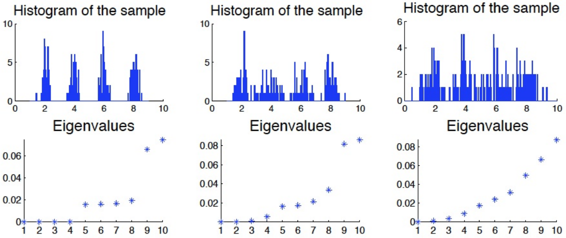
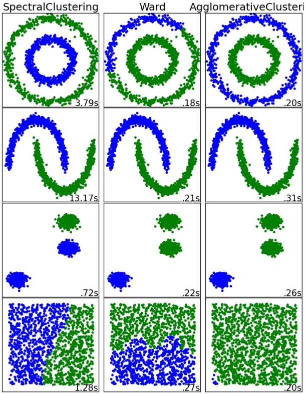


Figure 4: Three data sets, and the smallest 10 eigenvalues of L_{rw}.

- Source Tutorial U. Von Luxburg



- Figure from scikitkearn :

Goal

- Labeled data : $\mathcal{S}_\ell = \{(x_1, y_1), \ldots, (x_\ell, y\ell)\}$
- Unlabeled data : $\mathcal{X}_u = \{x_{\ell+1}, \ldots, x_{\ell+u}\}$, $n = \ell + u$: available during training !
- Usually $\ell << u$
- Goal : find $\hat{y}_{\ell+1}, \ldots, \hat{y}_{\ell+u}$
- No function f : $\mathcal{X} \to \mathcal{Y}$ to be learned ! !

Example : information retrieval

- A user enters a query
- Search engine provides sample documents
- The user labels a subset of returned documents
- Now how to label all the document in the database ?

Example : proteome

- A target organism :
- the set of its proteins (supposedly known)
- Some proteins have a known functional class
- Predict the functional classes for the remaining set of proteins

- c : the number of possible labels (classes)
- a graph on data with adjacency matrix W
- matrix F of size $n \times c$ holds the labeling scores all the datapoints
- matrix Y of same size, is binary such that : $Y_{ij} = 1$ if x_i is initially labeled with label j, 0 otherwise.
- $0 < \alpha < 1$
- $D_{ii} = \sum_j w_{ij}$

The first term is a smoothness constraint. The other term is the data-fitting term.
Reference : Zhou et al. NIPS 2003

$$\mathcal{J}_1(F) = (1/\alpha - 1)\sum_{i=1}^{n} \|F_i - Y_i\|^2 + \sum_{i,j=1}^{n} w_{ij} \| \frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}} \|^2$$

Minimizing the cost function $\mathcal{J}$ yields the optimal F :

$$F = (I - \alpha L_{sym})^{-1} Y$$

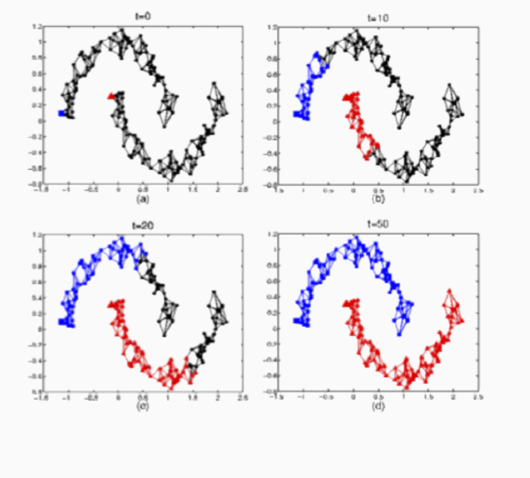
- I : identity matrix of size $n \times n$
- $L_{sym} = D^{-1/2}(D - W)D^{-1/2}$

Too expensive : $\mathcal{O}(n^3)$

Iterative algorithm :

- $F_0 = Y$
- Repeat until convergence :
  - $F_{t+1} = \alpha(I - L_{sym})F_t + (1 - \alpha)Y$

Transductive learning
Example in 2D



Semi-supervised learning

- Benefit from the availability of huge sets of unlabeled data
- Unlabeled data inform us about the probability distribution of the data $p(x)$
- Can we use it ? does it improve the performance of the resulting regressors/classifiers ?

Goal

- Labeled data : $\mathcal{S}_\ell = \{(x_1, y_1), \ldots, (x_\ell, y\ell)\}$
- Unlabeled data : $\mathcal{X}_u = \{x_{\ell+1}, \ldots, x_{\ell+u}\}$, $n = \ell + u$: available during training !
- Usually $\ell << u$
- Test data : $\mathcal{X}_{test} = \{x_{n+1}, \ldots, x_{n+m}\}$: not available during training
- **Learn a function f : $\mathcal{X} \rightarrow \mathcal{Y}$ (regression/classification) that behaves well on test data**

- Learn f from $\mathcal{X}$ to $\mathcal{Y}$ using $\mathcal{S}_\ell = \{(x_1, y_1), \ldots, (x_\ell, y_\ell)\}$ and $\mathcal{X}_u = \{x_{\ell+1}, \ldots, x_{\ell+u}\}$
- Methods
  - Self-training (including generative approaches)
  - Loss-based methods
    - Margin for unlabeled data
    - Smoothness penalty (graph-based semi-supervised learning)

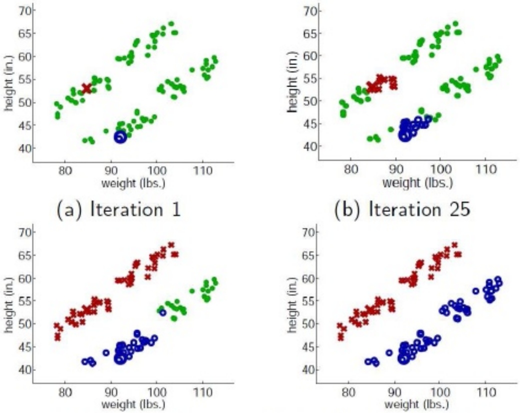
- Any classifier : f

Principle

1. k=0
2. Learn f_k by training on $\mathcal{S}_k = \mathcal{S}$
3. Use f to label $\mathcal{X}_u$ and keep the most confident u_k labeled data and build $\mathcal{S}_{k+1}$ new set of $\ell + u_k$ labeled data
4. Learn f_{k+1} by training on $\mathcal{S}_{k+1}$
5. If $D(f_{k+1}, f_k)$ is small then STOP else GOTO 3

- Two nice clusters without outliers [example Piyush Ray]

Base learner: KNN classifier



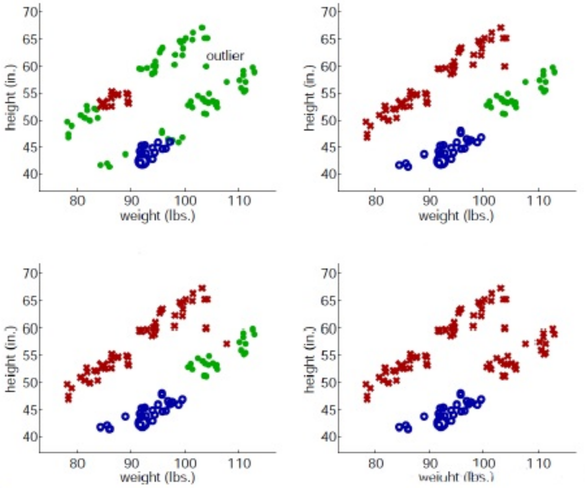
(a) Iteration 1

(b) Iteration 25

(c) Iteration 74

(d) Final labeling of all instances

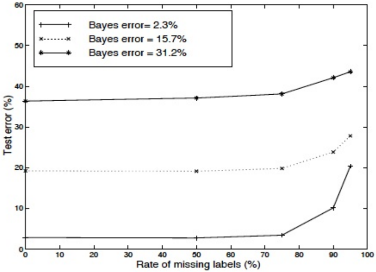
- Two clusters with outliers



- Margin : $\rho(x, y, h) = y.h(x)$
- Which margin for unlabeled data ?
- Reinforce the confidence of the classifier
  - $\rho_2(x, h) = h(x)^2$
  - $\rho_1(x, h) = |h(x)|$
  - **Implicit assumption** : cluster assumption : data in the same cluster share the same label
- Worked for SVM, MarginBoost, ...

- $h_t \in \mathcal{H}$: base classifier
- Boosting model : $H_T(x) = \sum_t \alpha_t h_t(x)$
- Loss function : $J(H_t) = \sum_{i=1}^{\ell} exp(-\rho(x_i, y_i, H_t)) + \lambda \sum_{j=\ell+1}^{n} exp(-\rho_u(x_j, H_t))$

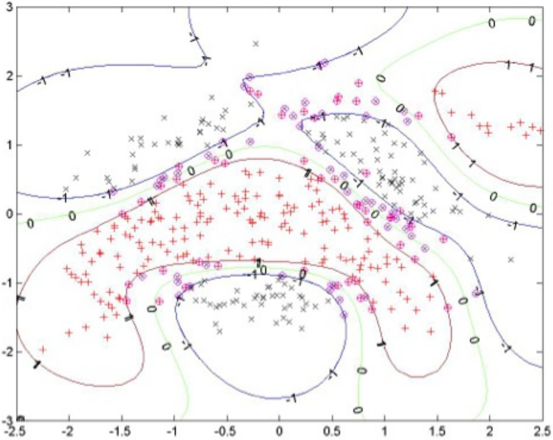
- Toys problems with different level of difficulty (we control Bayes error by mixing more or less the generative models)



[figure : NIPS 2001]

Semi-supervised learning

Data used in the previous sample



Use $y_{\ell+1}, \ldots, y_{\ell+u}$ during learning. Let us call $\mathbf{y}^* = [y_1^*, \ldots, y_u^*]$ the prediction vector.

Joachims proposed a Transductive SVM with a soft margin :

TSVM

$$\underset{\mathbf{w}, y^*, b}{\text{minimize}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{\ell}\xi_i + C^*\sum_{j=1}^{u}\xi_i^*$$

under the constraints

$$y_i(\mathbf{w}^T\mathbf{x}_i + \mathrm{b}) \geq 1 - \xi_i,\ i = 1, \ldots, n$$

$$y_j^*(\mathbf{w}^T\mathbf{x}_{\ell+j} + \mathrm{b}) \geq 1 - \xi_j^*,\ i = 1, \ldots, n$$

$$y_j^* \in \{-1, +1\},\ j = 1, \ldots, u$$

$$\xi_i \geq 0$$

$$\xi_j^* \geq 0$$

Ref : Joachims, 1999.

- Bennet and Demiriz 1999, 2001
- Bennet and Demiriz proposed $\rho_1(x, h) = |h(x)|$ and an implementation of S3VM based on Mangasarian's work.
- Robust Linear Programming

SVM formulation :

$$\min_{w,b,\eta} C \sum_{i=1}^{t} \eta_i + \frac{1}{2}\|w\|^2$$

$$s.t.\ y_i[wx_i - b] + \eta_i \geqslant 1$$

$$\eta_i \geqslant 0, i = 1, \ldots, l$$

S3VM formulation (Bennet and Demiriz) :

$$\begin{array}{ll} \min_{\mathbf{w},b,\eta,\xi,z} & C\left[\sum_{i=1}^{\ell}\eta_i + \sum_{j=\ell+1}^{\ell+k} min(\xi_j, z_j)\right] + \|\mathbf{w}\| \\ \textit{subject to} & y_i(\mathbf{w}\cdot x_i + b) + \eta_i \geq 1 \quad \eta_i \geq 0 \quad i = 1,\ldots,\ell \\ & \mathbf{w}\cdot x_j - b + \xi_j \geq 1 \quad \xi_j \geq 0 \quad j = \ell+1,\ldots,\ell+k \\ & -(\mathbf{w}\cdot x_j - b) + z_j \geq 1 \quad z_j \geq 0 \end{array}$$

With integer variables $d_i = 0 or 1$ according it belongs to class 1 or class -1 (d has to be learned as well) :

$$\begin{array}{ll} \min_{\mathbf{w},b,\eta,\xi,z,d} & C\left[\sum_{i=1}^{\ell}\eta_i + \sum_{j=\ell+1}^{\ell+k} (\xi_j + z_j)\right] + \|\mathbf{w}\| \\ \textit{subject to} & y_i(\mathbf{w}\cdot x_i - b) + \eta_i \geq 1 \quad \eta_i \geq 0 \quad i = 1,\ldots,\ell \\ & \mathbf{w}\cdot x_j - b + \xi_j + M(1-d_j) \geq 1 \quad \xi_j \geq 0 \quad j = \ell+1,\ldots,\ell+k \\ & -(\mathbf{w}\cdot x_j - b) + z_j + Md_j \geq 1 \quad z_j \geq 0 \quad d_j = \{0,1\} \end{array}$$

Mixed integer programming.

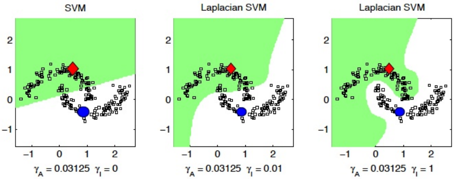
Let k be a positive definite kernel and $\mathcal{H}_k$ the unique RKHS induced by k.

Smoothness constraint / Manifold regularization 1/2

- Training data : $\mathcal{S}_\ell = \{(x_i, y_i, i =, \dots \ell\}$ and $\mathcal{S}_u = \{x_{\ell+1}, \dots, x_{\ell+u}\}$
- For $f \in \mathcal{H}_k$ and W a similarity matrix between data
- Impose an additional penalty that ensures smoothness of function f : for two close inputs, f takes close values
- Ref : Belkin, Nyogi and Sindwani (2006)

The key ideas :

- We assume that a better knowledge of the marginal distribution $P_x(x)$ will give us bette knowledge of $P(Y|x)$.
- If two points x_1 and x_2 are close in the intrinsic geometry of P_x then the conditional distribution $P(y|x_1)$ and $P(y|x_2)$ will be close.



- If $\mathcal{M}$, the support of P_x is a submanifold $\subset \mathbb{R}^p$, then we can try to minimize the penalty :

$$||f||_I^2 = \int_{\mathcal{M}} ||\nabla_{\mathcal{M}} f||^2 p(x) dx$$

- $\nabla_{\mathcal{M}} f$ is the gradient of f along the manifold $\mathcal{M}$
- Approximation of $||f||_I^2$:

$$||f||_I^2 \approx \sum_{ij} w_{ij} (f(x_i) - f(x_j))^2,$$

where W is the adjacency matrix of the data graph.

Let k be a positive definite kernel and $\mathcal{H}_k$ the unique RKHS induced by k.

Smoothness constraint / Manifold regularization

Minimize $J(f)$ in $\mathcal{H}_k$:

$$J(f) = \frac{1}{\ell}\sum_{i=1}^{\ell} V(x_i, y_i, f) + \lambda||f||_k^2 + \lambda_u \sum_{ij} w_{ij}(f(x_i) - f(x_j))^2$$

Let k be a positive definite kernel and $\mathcal{H}_k$ the unique RKHS induced by k.

Smoothness constraint / Manifold regularization

Minimize $J(f)$ in $\mathcal{H}_k$:

$$
\begin{aligned}
J(f) &= \frac{1}{\ell}\sum_{i=1}^{\ell} V(x_i, y_i, f) + \lambda ||f||_k^2 + \lambda_u \sum_{ij} w_{ij}(f(x_i) - f(x_j))^2 \\
&= \frac{1}{\ell}\sum_{i=1}^{\ell} V(x_i, y_i, f) + \lambda ||f||_k^2 + \lambda_u f^T L f
\end{aligned}
$$

$$
\begin{aligned}
J(f) &= \frac{1}{\ell}\sum_{i=1}^{\ell} V(x_i, y_i, f) + \lambda ||f||_k^2 + \lambda_u \sum_{ij=1}^{\ell+u} w_{ij}(f(x_i) - f(x_j))^2 \\
&= \frac{1}{\ell}\sum_{i=1}^{\ell} V(x_i, y_i, f) + \lambda ||f||_k^2 + \lambda_u f^T L f
\end{aligned}
$$

Any minimizer of $J(f)$ admits a representation
$\hat{f}(\cdot) = \sum_{i=1}^{\ell+u} \alpha_i k(x_i, \cdot)$

- Closed-from solution : extension of ridge regression

$$
\begin{aligned}
V(x_i, y_i, f) &= (y_i - f(x_i))^2 \\
\lambda_L = \frac{\lambda_u}{u+\ell} \\
\hat{\alpha} &= (JK + \lambda\ell Id + \frac{\lambda_u \ell}{(u+\ell)^2} LK)^{-1} Y
\end{aligned}
$$

K : Gram matrix for all data
J : $(\ell+u)\times(\ell+u)$ diagonal matrix with the first ℓ values equal to 1 and the remaining ones to 0.

We choose the hinge loss functions :

$$\min_{f \in \mathcal{H}_k} \frac{1}{\ell} \sum_{i=1}^{\ell} (1 - y_i f(x_i))_+ + \lambda ||f||_k^2 + \frac{\lambda_u}{u+\ell} f^T L f$$

We benefit from the representer theorem.

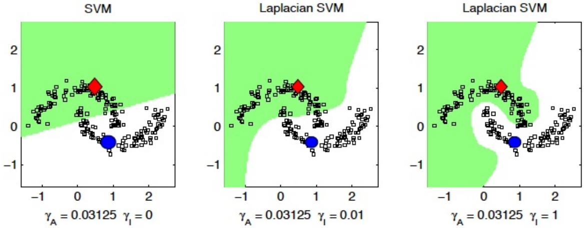
In practise, we solve :

$$\min_{\alpha \in \mathbb{R}^{l+u}, \xi \in \mathbb{R}^{l}} \frac{1}{l} \sum_{i=1}^{l} \xi_i + \gamma_A \alpha^T K \alpha + \frac{\gamma_I}{(u+l)^2} \alpha^T K L K \alpha$$

$$\text{subject to: } y_i\Big(\sum_{j=1}^{l+u} \alpha_j K(x_i, x_j) + b\Big) \geq 1 - \xi_i, \quad i = 1, \ldots, l$$

$$\xi_i \geq 0 \quad i = 1, \ldots, l.$$

Results : Belkin et al. 2006, JMLR.



- Code the Laplacian SVM or the Laplacian Kernel Ridge regressor in scikitlearn
- Elaborate ideas to scale up Spectral clustering
- Ng, Jordan, Spectral Clustering, 2001.
- Belkin et al. 2006, JMLR
- Book : Semi-supervised learning, Chapelle, Scholpkoft, Zien,MIT