Manifold Regularization
9.520 Class 06, 27 February 2006
Andrea Caponnetto
About this class

**Goal** To analyze the limits of learning from examples in high dimensional spaces. To introduce the semi-supervised setting and the use of unlabeled data to learn the intrinsic geometry of a problem. To define Riemannian Manifolds, Manifold Laplacians, Graph Laplacians. To introduce a new class of algorithms based on Manifold Regularization (LapRLS, LapSVM).
Unlabeled data

Why using unlabeled data?

• labeling is often an “expensive” process

• semi-supervised learning is the natural setting for human learning
Semi-supervised setting

$u$ i.i.d. samples drawn on $X$ from the marginal distribution $p(x)$

$$\{x_1, x_2, \ldots, x_u\},$$

only $n$ of which endowed with labels drawn from the conditional distributions $p(y|x)$

$$\{y_1, y_2, \ldots, y_n\}.$$

The extra $u - n$ unlabeled samples give additional information about the marginal distribution $p(x)$. 
The importance of unlabeled data
Curse of dimensionality and $p(x)$

Assume $X$ is the $D$-dimensional hypercube $[0,1]^D$. The worst case scenario corresponds to uniform marginal distribution $p(x)$.

Two perspectives on curse of dimensionality:

- As $d$ increases, local techniques (e.g., nearest neighbors) become rapidly ineffective.

- Minimax results show that rates of convergence of empirical estimators to optimal solutions of known smoothness, depend critically on $D$. 
Curse of dimensionality and k-NN

- It would seem that with a reasonably large set of training data, we could always approximate the conditional expectation by k-nearest-neighbor averaging.

- We should be able to find a fairly large set of observations close to any $x \in [0,1]^D$ and average them.

- This approach and our intuition breaks down in high dimensions.
Sparse sampling in high dimension

Suppose we send out a cubical neighborhood about one vertex to capture a fraction $r$ of the observations. Since this corresponds to a fraction $r$ of the unit volume, the expected edge length will be

$$e_D(r) = r^{1/D}.$$ 

Already in ten dimensions $e_{10}(0.01) = 0.63$, that is to capture 1% of the data, we must cover 63% of the range of each input variable!

No more "local" neighborhoods!
Distance vs volume in high dimensions
Curse of dimensionality and smoothness

Assuming that the target function \( f^* \) (in the squared loss case) belongs to the Sobolev space

\[
W_s^2([0,1]^D) = \{ f \in L_2([0,1]^D) | \sum_{\omega \in \mathbb{Z}^d} \| \omega \|^{2s} | \hat{f}(\omega) |^2 < +\infty \}
\]

it is possible to show that *

\[
\sup_{\mu, f^* \in W_s^2} \mathbb{E}_S(I[f_S] - I[f^*]) > Cn^{-\frac{s}{D}}
\]

More smoothness \( s \) ⇒ faster rate of convergence

Higher dimension \( D \) ⇒ slower rate of convergence

*A Distribution-Free Theory of Nonparametric Regression, Gyorfi*
Intrinsic dimensionality

Raw format of natural data is often high dimensional, but in many cases it is the outcome of some process involving only few degrees of freedom.

Examples:

- Acoustic Phonetics $\Rightarrow$ vocal tract can be modelled as a sequence of few tubes.
- Facial Expressions $\Rightarrow$ tonus of several facial muscles control facial expression.
- Pose Variations $\Rightarrow$ several joint angles control the combined pose of the elbow-wrist-finger system.

**Smoothness assumption:** $y$’s are “smooth” relative to natural degrees of freedom, **not** relative to the raw format.
Manifold embedding
Riemannian Manifolds

A $d$-dimensional manifold

$$\mathcal{M} = \bigcup_{\alpha} U_{\alpha}$$

is a mathematical object that generalized domains in $\mathbb{R}^d$.

Each one of the “patches” $U_{\alpha}$ which cover $\mathcal{M}$ is endowed with a system of coordinates

$$\alpha : U_{\alpha} \to \mathbb{R}^d.$$  

If two patches $U_{\alpha}$ and $U_{\beta}$, overlap, the transition functions

$$\beta \circ \alpha^{-1} : \alpha(U_{\alpha} \cap U_{\beta}) \to \mathbb{R}^d$$

must be smooth (eg. infinitely differentiable).

- The Riemannian Manifold inherits from its local system of coordinates, most geometrical notions available on $\mathbb{R}^d$: metrics, angles, volumes, etc.
Manifold’s charts
Differentiation over manifolds

Since each point $x$ over $\mathcal{M}$ is equipped with a local system of coordinates in $\mathbb{R}^d$ (its tangent space), all differential operators defined on functions over $\mathbb{R}^d$, can be extended to analogous operators on functions over $\mathcal{M}$.

Gradient: $\nabla f(x) = \left( \frac{\partial}{\partial x_1} f(x), \ldots, \frac{\partial}{\partial x_d} f(x) \right) \Rightarrow \nabla \mathcal{M} f(x)$

Laplacian: $\triangle f(x) = -\frac{\partial^2}{\partial x_1^2} f(x) - \cdots - \frac{\partial^2}{\partial x_d^2} f(x) \Rightarrow \triangle \mathcal{M} f(x)$
Measuring smoothness over $\mathcal{M}$

Given $f : \mathcal{M} \to \mathbb{R}$

- $\nabla_{\mathcal{M}} f(x)$ represents amplitude and direction of variation around $x$

- $S(f) = \int_{\mathcal{M}} \| \nabla_{\mathcal{M}} f \|^2$ is a global measure of smoothness for $f$

- Stokes’ theorem (generalization of integration by parts) links gradient and Laplacian

$$S(f) = \int_{\mathcal{M}} \| \nabla_{\mathcal{M}} f(x) \|^2 = \int_{\mathcal{M}} f(x) \Delta_{\mathcal{M}} f(x)$$
Example: the circle $S^1$

$\mathcal{M}$: circle with angular coordinate $\theta \in [0, 2\pi)$

$$\nabla_{\mathcal{M}} f = \frac{\partial}{\partial \theta} f, \quad \Delta_{\mathcal{M}} f = -\frac{\partial^2}{\partial \theta^2} f$$

integration by parts: $\int_0^{2\pi} \left( \frac{\partial}{\partial \theta} f(\theta) \right)^2 d\theta = -\int_0^{2\pi} f(\theta) \frac{\partial^2}{\partial \theta^2} f(\theta) d\theta$

eigensystem of $\Delta_{\mathcal{M}}$:

$$\Delta_{\mathcal{M}} \phi_k = \lambda_k \phi_k$$

$$\phi_k(\theta) = \sin k\theta, \ \cos k\theta, \ \lambda_k = k^2 \ \kappa \in \mathbb{N}$$
Manifold regularization *

A new class of techniques which extend standard Tikhonov regularization over RKHS, introducing the additional regularizer \( \|f\|_I^2 = \int_M f(x) \Delta_M f(x) \) to enforce smoothness of solutions relative to the underlying manifold

\[
f^* = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} V(f(x_i), y_i) + \lambda_A \|f\|_K^2 + \lambda_I \int_M f \Delta_M f
\]

- \( \lambda_I \) controls the complexity of the solution in the \textit{intrinsic} geometry of \( \mathcal{M} \).
- \( \lambda_A \) controls the complexity of the solution in the \textit{ambient} space.

*Belkin, Niyogi, Sindhwani, 04
Manifold regularization (cont.)

Other natural choices of $\| \cdot \|_f^2$ exist

- Iterated Laplacians $\int_{\mathcal{M}} f \triangle^s_{\mathcal{M}} f$ and their linear combinations. These smoothness penalties are related to Sobolev spaces

$$\int f(x) \triangle^s_{\mathcal{M}} f(x) \approx \sum_{\omega \in \mathbb{Z}^d} \| \omega \|^{2s} |\hat{f}(\omega)|^2$$

- Frobenius norm of the Hessian (the matrix of second derivatives of $f$)

- Diffusion regularizers $\int_{\mathcal{M}} f e^{t \Delta}(f)$. The semigroup of smoothing operators $G = \{ e^{-t \Delta_{\mathcal{M}}} \mid t > 0 \}$ corresponds to the process of diffusion (Brownian motion) on the manifold.

*Hessian Eigenmaps; Donoho, Grimes 03
Laplacian and diffusion

• If $M$ is compact, the operator $\nabla_M$ has a countable sequence of eigenvectors $\phi_k$ (with non-negative eigenvalues $\lambda_k$), which is a complete system of $L_2(M)$. If $M$ is connected, the constant function is the only eigenvector corresponding to null eigenvalue.

• The function of operator $e^{-t\nabla_M}$, is defined by the eigensystem $(e^{-t\lambda_k}, \phi_k)$, $k \in \mathbb{N}$.

• the diffusion stabilizer $\|f\|_I^2 = \int_M f e^{t\nabla_M}(f)$ is the squared norm of RKHS with kernel equal to Green’s function of heat equation

$$\frac{\partial T}{\partial t} = -\nabla_M T$$
Laplacian and diffusion (cont.)

1. By Taylor expansion of $T(x, t)$ around $t = 0$

$$T(x, t) = T(x, 0) + t \frac{\partial}{\partial t} T(x, 0) + \cdots + \frac{1}{k!} t^k \frac{\partial^k}{\partial t^k} T(x, 0) + \cdots$$

$$= e^{-t\Delta} T(x, 0) = \int K_t(x, x') T(x', 0) dx' = L_K T(x', 0)$$

2. For small $t > 0$, the Green’s function is a sharp gaussian

$$K_t(x, x') \approx e^{-\frac{||x-x'||^2}{t}}$$

3. Recalling relation of integral operator $L_K$ and RKHS norm, we get

$$\|f\|_I^2 = \int f \ e^{t\Delta} (f) = \int f \ L_K^{-1}(f) = \|f\|_K^2$$
An empirical proxy of the manifold

We cannot compute the intrinsic smoothness penalty

$$\|f\|_I^2 = \int_M f(x) \Delta_M f(x)$$

because we don’t know the manifold $M$ and the embedding

$$\Phi : M \rightarrow \mathbb{R}^D.$$ 

But we assume that the unlabeled samples are drawn i.i.d. from the uniform probability distribution over $M$ and then mapped into $\mathbb{R}^D$ by $\Phi$.
Neighborhood graph

Our proxy of the manifold is a \textit{weighted neighborhood graph} $G = (V, E, W)$, with \textbf{vertices} $V$ given by the points $\{x_1, x_2, \ldots, x_u\}$, \textbf{edges} $E$ defined by one of the two following adjacency rules

\begin{itemize}
  \item connect $x_i$ to its $k$ nearest neighborhoods
  \item connect $x_i$ to $\epsilon$-close points
\end{itemize}

and \textbf{weights} $W_{ij}$ associated to two connected vertices

$$W_{ij} = e^{-\frac{\|x_i-x_j\|^2}{\epsilon}}$$

\textbf{Note:} computational complexity $O(u^2)$
Neighborhood graph (cont.)
The **graph Laplacian**

The graph Laplacian over the weighted neighborhood graph \((G, E, W)\) is the matrix

\[
L_{ij} = D_{ii} - W_{ij}, \quad D_{ii} = \sum_j W_{ij}.
\]

\(L\) is the discrete counterpart of the manifold Laplacian \(\triangle_M\)

\[
f^T L f = \sum_{i,j=1}^n W_{ij} (f_i - f_j)^2 \approx \int_M \|\nabla f\|^2 dp.
\]

Analogous properties of the eigensystem: nonnegative spectrum, null space

**Looking for rigorous convergence results**
A convergence theorem *

Operator $\mathcal{L}$: “out-of-sample extension” of the graph Laplacian $\mathcal{L}$

$$\mathcal{L}(f)(x) = \sum_i (f(x) - f(x_i)) e^{-\frac{\|x - x_i\|^2}{\epsilon}} \quad x \in X, \ f : X \to \mathbb{R}$$

**Theorem:** Let the $u$ data points $\{x_1, \ldots, x_u\}$ be sampled from the uniform distribution over the embedded $d$-dimensional manifold $\mathcal{M}$. Put $\epsilon = u^{-\alpha}$, with $0 < \alpha < \frac{1}{2+d}$. Then for all $f \in C^\infty$ and $x \in X$, there is a constant $C$, s.t. in probability,

$$\lim_{u \to \infty} C^\epsilon \frac{d+2}{u} \mathcal{L}(f)(x) = \Delta_\mathcal{M} f(x).$$

**Note:** also stronger forms of convergence have been proved.

*Belkin, Niyogi, 05*
Laplacian-based regularization algorithms *

Replacing the unknown manifold Laplacian with the graph Laplacian $\|f\|_I^2 = \frac{1}{u^2} f^T L f$, where $f$ is the vector $[f(x_1), \ldots, f(x_u)]$, we get the minimization problem

$$f^* = \arg \min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} V(f(x_i), y_i) + \lambda_A \|f\|_K^2 + \frac{\lambda_I}{u^2} f^T L f$$

- $\lambda_I = 0$: standard regularization (RLS and SVM)
- $\lambda_A \to 0$: out-of-sample extension for Graph Regularization
- $n = 0$: unsupervised learning, Spectral Clustering

*Belkin, Niyogi, Sindhwani, 04
The Representer Theorem

Using the same type of reasoning used in Class 3, a Representer Theorem can be easily proved for the solutions of Manifold Regularization algorithms.

The expansion range over all the supervised and unsupervised data points

\[ f(x) = \sum_{j=1}^{u} c_j K(x, x_j). \]
LapRLS

Generalizes the usual RLS algorithm to the semi-supervised setting.

Set $V(w, y) = (w - y)^2$ in the general functional.

By the representer theorem, the minimization problem can be restated as follows

$$c^* = \arg \min_{c \in \mathbb{R}^u} \frac{1}{n} \sum_{i=1}^{n} (y_i - J K c)^T (y_i - J K c) + \lambda_A c^T K c + \frac{\lambda_I}{u^2} c^T K L K c,$$

where $y$ is the $u$-dimensional vector $(y_1, \ldots, y_n, 0, \ldots, 0)$, and $J$ is the $u \times u$ matrix $\text{diag}(1, \ldots, 1, 0, \ldots, 0)$. 
LapRLS (cont.)

The functional is differentiable, strictly convex and coercive. The derivative of the object function vanishes at the minimizer $c^*$

$$\frac{1}{n}KJ(y - JKc^*) + (\lambda_A K + \frac{\lambda_I n}{u^2}KLK)c^* = 0.$$  

From the relation above and noticing that due to the positivity of $\lambda_A$, the matrix $M$ defined below, is invertible, we get

$$c^* = M^{-1}y,$$

where

$$M = JK + \lambda_A nI + \frac{\lambda_I n^2}{u^2}LK.$$
LapSVM

Generalizes the usual SVM algorithm to the semi-supervised setting.

Set $V(w, y) = (1 - yw)_+$ in the general functional above.

Applying the representer theorem, introducing slack variables and adding the unpenalized bias term $b$, we easily get the primal problem

$$c^* = \arg \min_{c \in \mathbb{R}^u, \xi \in \mathbb{R}^n} \frac{1}{n} \sum_{i=1}^{n} \xi_i + \lambda A c^T K c + \frac{\lambda I}{u^2} c^T KLK c$$

subject to:

$$y_i \left( \sum_{j=1}^{u} c_j K(x_i, x_j) + b \right) \geq 1 - \xi_i \quad i = 1, \ldots, n$$

$$\xi_i \geq 0 \quad i = 1, \ldots, n$$
LapSVM: forming the Lagrangian

As in the analysis of SVM, we derive the Wolfe dual quadratic program using Lagrange multiplier techniques:

\[
L(c, \xi, b, \alpha, \zeta) = \frac{1}{n} \sum_{i=1}^{n} \xi_i + \frac{1}{2} c^T \left( 2 \lambda_A K + 2 \frac{\lambda_I}{u^2} KLK \right) c \\
- \sum_{i=1}^{n} \alpha_i \left( y_i \left\{ \sum_{j=1}^{u} c_j K(x_i, x_j) + b \right\} - 1 + \xi_i \right) \\
- \sum_{i=1}^{n} \xi_i \xi_i
\]

We want to minimize \( L \) with respect to \( c, b, \) and \( \xi, \) and maximize \( L \) with respect to \( \alpha \) and \( \zeta, \) subject to the constraints of the primal problem and nonnegativity constraints on \( \alpha \) and \( \zeta. \)
LapSVM: eliminating \( b \) and \( \xi \)

\[
\frac{\partial L}{\partial b} = 0 \implies \sum_{i=1}^{n} \alpha_i y_i = 0
\]

\[
\frac{\partial L}{\partial \xi_i} = 0 \implies \frac{1}{n} - \alpha_i - \zeta_i = 0
\]

\[
\implies 0 \leq \alpha_i \leq \frac{1}{n}
\]

We write a reduced Lagrangian in terms of the remaining variables:

\[
L^R(c, \alpha) = \frac{1}{2} c^T \left( 2\lambda_A K + 2\frac{\lambda_I}{u^2} KLK \right) c - c^T K J^T Y \alpha + \sum_{i=1}^{n} \alpha_i,
\]

where \( J \) is the \( n \times u \) matrix \((I \ 0)\) with \( I \) the \( n \times n \) identity matrix and \( Y = \text{diag}(y) \).
LapSVM: eliminating $c$

Assuming the $K$ matrix is invertible,

\[
\frac{\partial L^R}{\partial c} = 0 \implies \left(2\lambda_A K + 2\frac{\lambda I}{u^2} K L K\right)c - K J^T Y \alpha = 0
\]

\[
\implies c = \left(2\lambda_A I + 2\frac{\lambda I}{u^2} L K\right)^{-1} J^T Y \alpha
\]

Note that the relationship between $c$ and $\alpha$ is no longer as simple as in the SVM algorithm.
LapSVM: the dual program

Substituting in our expression for $c$, we are left with the following “dual” program:

$$\alpha^* = \arg \max_{\alpha \in \mathbb{R}^n} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \alpha^T Q \alpha$$

subject to:

$$\sum_{i=1}^{n} y_i \alpha_i = 0$$

$$0 \leq \alpha_i \leq \frac{1}{n} \quad i = 1, \ldots, n$$

Here, $Q$ is the matrix defined by

$$Q = YJ K \left(2\lambda_A I + 2\frac{\lambda_I}{\alpha^2} L K\right)^{-1} J^T Y.$$ 

One can use a standard SVM solver with the matrix $Q$ above, hence compute $c$ solving a linear system.
Numerical experiments

- Two Moons Dataset
- Handwritten Digit Recognition
- Spoken Letter Recognition

*http://manifold.cs.uchicago.edu/manifold_regularization*