Riemannian Geometry on Graphs and Its Application to Ranking and Classification

Dengyong Zhou
(joint talk with Arthur Gretton)
Department of Empirical Inference
Max Planck Institute for Biological Cybernetics
dengyong.zhou@tuebingen.mpg.de

June 23, 2004

Ranking and classification on graphs

- Ranking: given a graph, in which a (or many) nodes is (are) regarded as the query (queries), ranking the remaining nodes according to their relevances to the query (queries).

- Classification: given a graph, in which some of the nodes are labeled, the task is to predict the labels of the remaining nodes.

Example: Web, social networks, web-log, market-basket analysis, protein/gene networks, documents, images, ...

Our contributions

Two sides

- Mathematical side: build a self-contained differential geometry on graphs.

  Parallel to differential operators on Riemannian manifolds

- Computer science side: propose the powerful classification and ranking algorithms.

  Parallel to variational problems on Riemannian manifolds

In one word, graphs are thought of as Riemannian manifolds!

Some basic notions in graph theory

- A graph $\Gamma = (V, E)$ consists of a set V of vertices and a set of pairs of vertices $E \subseteq V \times V$ called edges.
- A graph is undirected if for each edge $(u, v) \in E$ we also have $(v, u) \in E$.
- A graph is weighted if it is associated with a function $w : E \longrightarrow \mathbb{R}_+$ satisfying $w(u, v) = w(v, u)$.

Some basic notions in graph theory (Cont.)

- The degree function $g : V \longrightarrow \mathbb{R}_+$ is defined to be

$$g(v) := \sum_{u \sim v} w(u, v),$$

where $u \sim v$ denote the set of vertices u connected to v via the edges (u, v). The degree can be regarded as a measure.

The space of functions defined on graphs

- Let $\mathcal{H}(V)$ denote the Hilbert space of real-valued functions endowed with the usual inner product

$$\langle \varphi, \phi \rangle := \sum_{v} \varphi(v)\phi(v),$$

where φ and ϕ denote any two functions in $\mathcal{H}(V)$. Similarly define $\mathcal{H}(E)$. Note that function $\psi \in \mathcal{H}(E)$ need not to be symmetric, i.e., we do not require $\psi(u, v) = \psi(v, u)$.

Gradient (or boundary) operator

- We define the *gradient* operator

$$d : \mathcal{H}(V) \to \mathcal{H}(E)$$

to be

$$(d\varphi)(u, v) := \sqrt{\frac{w(u, v)}{g(u)}}\varphi(u) - \sqrt{\frac{w(u, v)}{g(v)}}\varphi(v).$$

Clearly,

$$(d\varphi)(u, v) = -(d\varphi)(v, u),$$

i.e., $d\varphi$ is skew-symmetric.

Divergence (or co-boundary) operator

- We define the adjoint

$$d^* : \mathcal{H}(E) \to \mathcal{H}(V)$$

of d by

$$\langle d\varphi, \psi \rangle = \langle \varphi, d^*\psi \rangle, \text{ for all } \varphi \in \mathcal{H}(V), \psi \in \mathcal{H}(E).$$

We call d^* the *divergence* operator.

[Note: the inner products are respectively in the space $\mathcal{H}(E)$ and $\mathcal{H}(V)$.]

Divergence (or co-boundary) operator (Cont.)

- We can show that d^* is given by

$$(d^*\psi)(v) = \sum_{u \sim v} \sqrt{\frac{w(u,v)}{g(v)}} \Big(\psi(v,u) - \psi(u,v) \Big).$$

Edge derivative

- The *edge derivative*

$$\left.\frac{\partial}{\partial e}\right|_v : \mathcal{H}(V) \to \mathbb{R}$$

along edge $e = (v, u)$ at vertex v is defined by

$$\left.\frac{\partial \varphi}{\partial e}\right|_v := (d\varphi)(v, u).$$

Local variation

- Define the *local variation* of φ at v to be

$$\|\nabla_v \varphi\| := \left[\sum_{e \vdash v} \left(\left. \frac{\partial \varphi}{\partial e} \right|_v \right)^2 \right]^{1/2},$$

where $e \vdash v$ denotes the set of edges incident on v.

Total variation

- Let $\mathcal{S}$ denote a functional on $\mathcal{H}(V)$, for any $p \in [1, \infty)$, which is defined to be
$$\mathcal{S}_p(\varphi) := \frac{1}{p} \sum_v \left\| \nabla_v \varphi \right\|^p.$$
The functional $\mathcal{S}_p(\varphi)$ can be thought of as the measure of the *smoothness* of φ.

General variational problem on graphs

- Given a function y in $\mathcal{H}(V)$, the goal is to search for another function f in $\mathcal{H}(V)$, which is not only *smooth* enough on Γ but also *close* enough to the given function y. This idea is formalized via the following optimization problem:

$$\underset{f \in \mathcal{H}(V)}{\operatorname{argmin}} \{\mathcal{S}_p(f) + \frac{\mu}{2} \|f - y\|^2\}.$$

  The first term is the *smoothness term* or *regularizer*, which requires f not to change too much between closely related objects. The second term is the *fitting term*, which says that f should not be far away from y.

Ranking and classification

- Ranking: define $y(v) = 1$ if vertex v is a query and 0 otherwise. Then rank each vertex v according to the corresponding function value $f(v)$ (largest ranked first).

- Classification: define $y(v) = 1$ or -1 if v is labeled as positive or negative and 0 otherwise. Then each vertex v is classified as $\operatorname{sgn} f(v)$.

Connection between ranking and classification

- Classification is essentially built on ranking: for classifying a vertex, compute its ranking score with respect to the queries of different classes, and then the classification depends on which class of ranking score is larger.

Laplacian operator

- By analogy with the Laplace-Beltrami operator on forms on Riemannian manifolds, we define the *graph Laplacian*

$$\Delta : \mathcal{H}(V) \to \mathcal{H}(V)$$

by

$$\Delta := \frac{1}{2} d^* d.$$

Laplacian operator (Cont.)

- Laplacian and smoothness

$$\langle \Delta\varphi, \varphi \rangle = \langle \frac{1}{2} d^* d\varphi, \varphi \rangle = \frac{1}{2} \langle d\varphi, d\varphi \rangle = \mathcal{S}_2(\varphi)$$

$$\Rightarrow \Delta\varphi = \frac{\partial \mathcal{S}_2(\varphi)}{\partial \varphi}.$$

Laplacian operator (Cont.)

Computation of the graph Laplacian

- Substituting the definitions of gradient and divergence operators into that of Laplacian, we have

$$(\Delta\varphi)(v) = \varphi(v) - \sum_{u\sim v} \frac{w(u,v)}{\sqrt{g(u)g(v)}}\varphi(u).$$

[Note: In spectral graph theory, this is directly used as the definition of the graph Laplacian. Our derivation of it, however, is new.]

Laplacian operator (Cont.)

Another equivalent definition of the Graph Laplacian

- We can also define the graph Laplacian with the notion of edge derivative as

$$(\Delta\varphi)(v) := \frac{1}{2}\sum_{e \vdash v} \frac{1}{\sqrt{g}}\left(\frac{\partial}{\partial e}\sqrt{g}\frac{\partial \varphi}{\partial e}\right)\bigg|_v .$$

  This is basically the discrete analogue of another definition of the Laplace-Beltrami operator based on the gradient.

Solving the optimization problem $(p = 2)$

Theorem. *The solution f of the optimization problem*

$$\underset{f \in \mathcal{H}(V)}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{v} \|\nabla_v f\|^2 + \frac{\mu}{2} \|f - y\|^2 \right\}.$$

satisfies

$$\Delta f + \mu(f - y) = 0.$$

Corollary. $f = \mu(\mu I + \Delta)^{-1} y.$

An equivalent iterative algorithm

Isotropic information diffusion

Define

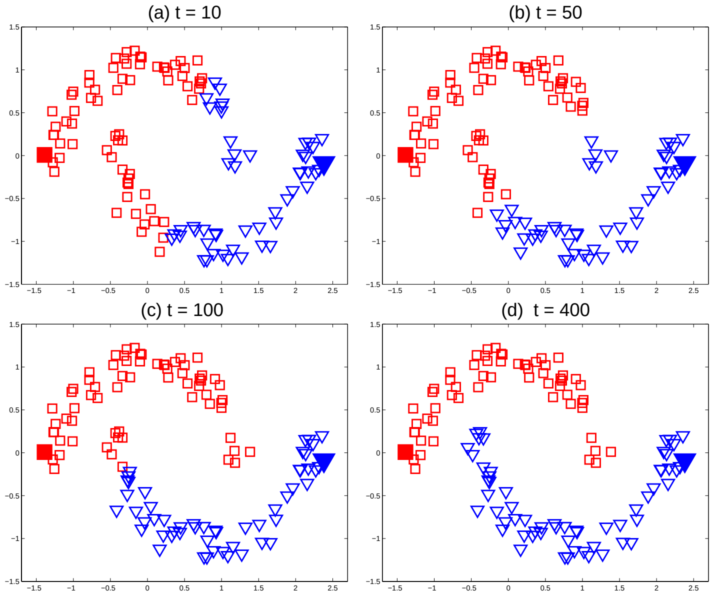
$$p(u, v) = \frac{w(u, v)}{\sqrt{g(u)g(v)}}.$$

Then

$$f^{(t+1)}(v) = \sum_{u \sim v} \alpha p(u, v) f^{t}(u) + (1 - \alpha) y(v),$$

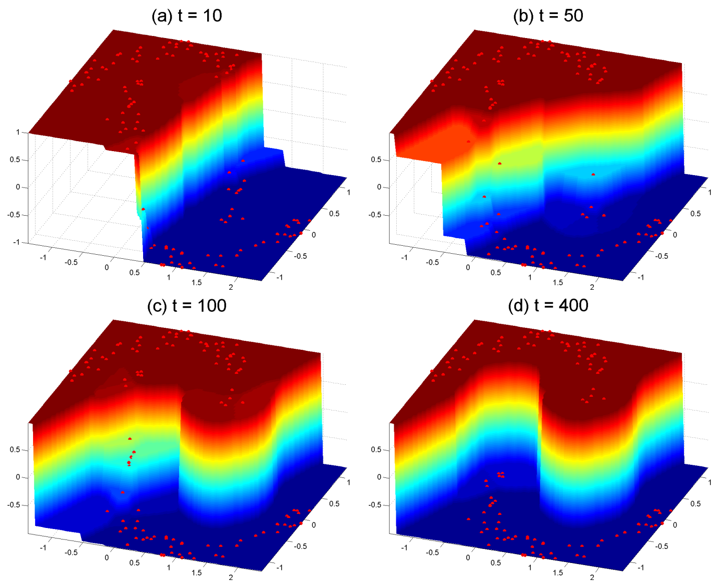
where α is a parameter in $(0, 1)$.

A toy classification example



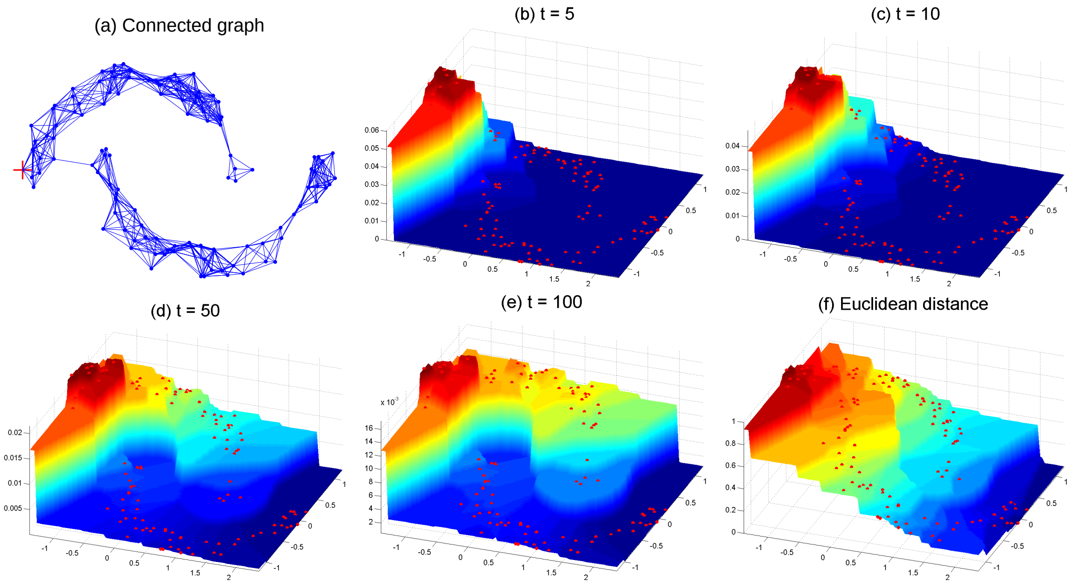
[Note: fully connected graph: $w(u, v) = \exp(-\lambda \|u - v\|)$.]

A toy classification example (Cont.)



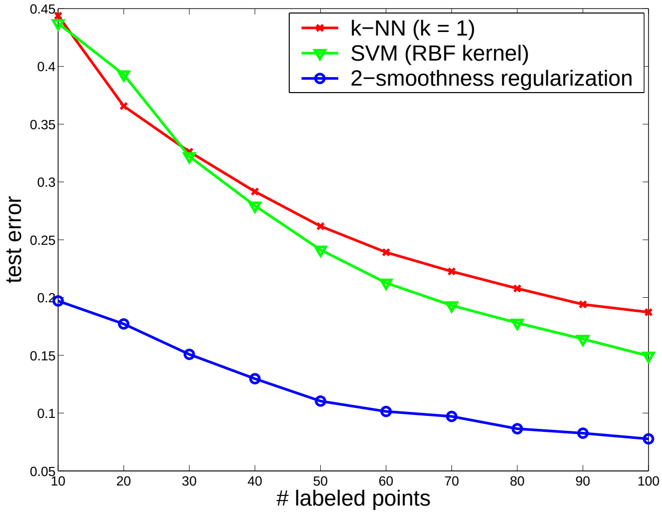
[Note: the function is becoming flatter and flatter.]

A toy ranking example



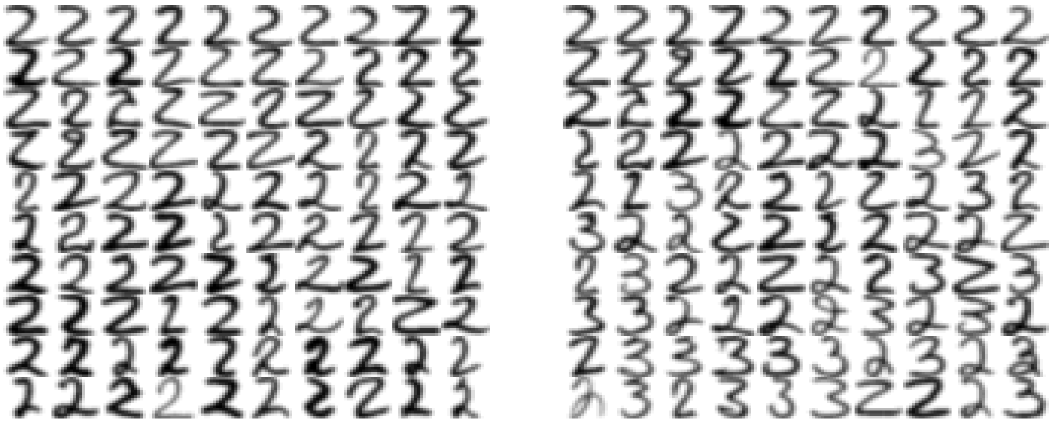
[Note: The shortest path based ranking does not work!]

Handwritten digit recognition



Digit recognition with USPS handwritten 16x16 digits dataset for a total of 9298. The left panel shows test errors for different algorithms with the number of labeled points increasing from 10 to 100.

Handwritten digit ranking



Ranking digits in USPS. The top-left digit in each panel is the query. The left panel shows the top 99 by our method; and the right panel shows the top 99 by the Euclidean distance based ranking. Note that in addition to *3*s there are many more *2*s with knots in the right panel.

Curvature operator

- By analogy with the curvature of a surface which is measured by the change in the unit normal, we define the *graph curvature*

$$\kappa : \mathcal{H}(V) \to \mathcal{H}(V)$$

by

$$\kappa\varphi := d^*\left(\frac{d\varphi}{\|\nabla\varphi\|}\right).$$

Curvature operator (Cont.)

- Computation of the graph curvature

$$(\kappa\varphi)(v) = \sum_{u\sim v} \frac{w(u,v)}{\sqrt{g(v)}} \left(\frac{1}{\|\nabla_v \varphi\|} + \frac{1}{\|\nabla_u \varphi\|} \right) \left(\frac{\varphi(v)}{\sqrt{g(v)}} - \frac{\varphi(u)}{\sqrt{g(u)}} \right)$$

Unlike the graph Laplacian, the graph curvature is a non-linear operator.

Curvature operator (Cont.)

- Another equivalent definition of the graph curvature based on the gradient:

$$(\kappa\varphi)(v) := \sum_{e \vdash v} \frac{1}{\sqrt{g}} \left(\frac{\partial}{\partial e} \frac{\sqrt{g}}{\|\nabla\varphi\|} \frac{\partial \varphi}{\partial e} \right) \bigg|_v .$$

- An elegant property of the graph curvature

$$\kappa\varphi = \frac{\partial \mathcal{S}_1(\varphi)}{\partial \varphi}.$$

Solving the optimization problem $(p = 1)$

Theorem. *The solution of the optimization problem*

$$\underset{f \in \mathcal{H}(V)}{\operatorname{argmin}} \left\{ \sum_v \|\nabla_v f\| + \frac{\mu}{2} \|f - y\|^2 \right\}$$

satisfies

$$\kappa f + \mu(f - y) = 0.$$

No closed form solution.

An iterative algorithm

Anisotropic information diffusion

$$f^{(t+1)}(v) = \sum_{u \sim v} p^{(t)}(u,v) f^{(t)}(v) + p^{(t)}(v,v) y(v), \ \forall v \in V$$

[Note: Compared with the iterative algorithm corresponding to the 2-smoothness regularizer, the weight coefficients $p(u,v)$ in this iteration procedures are adaptively updated at each iteration, in addition to the classifying function being updated. This weight update causes the diffusion inside clusters to be enhanced, and the diffusion across clusters to be reduced.]

An iterative algorithm (Cont.)

Weight coefficients update:

$$p(u,v) = \frac{\dfrac{m(u,v)}{\sqrt{g(u)g(v)}}}{\displaystyle\sum_{u\sim v} \frac{m(u,v)}{g(v)} + \mu}, \text{ if } u \neq v;$$

and

$$p(v,v) = \frac{\mu}{\displaystyle\sum_{u\sim v} \frac{m(u,v)}{g(v)} + \mu}.$$

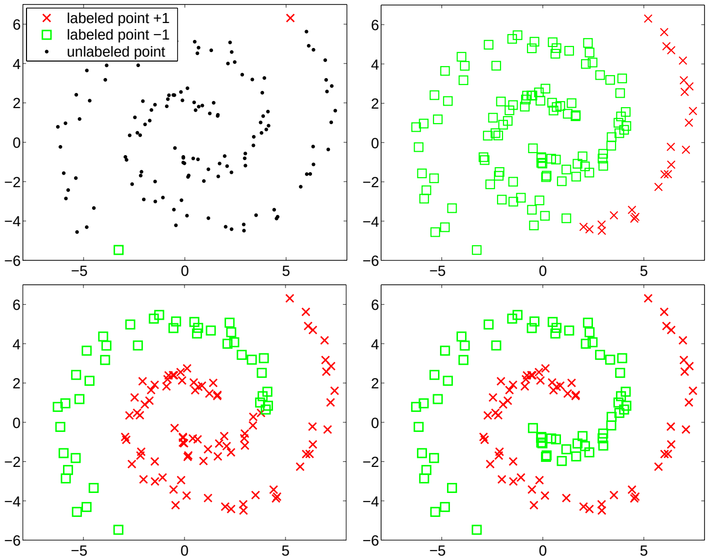
An iterative algorithm (Cont.)

The function $m : E \longrightarrow \mathbb{R}$ is defined by

$$m(u, v) = w(u, v)\left(\frac{1}{\|\nabla_u f\|} + \frac{1}{\|\nabla_v f\|}\right).$$

The smoother the function f at nodes u and v, the larger the function m at edge (u, v).

A toy classification example

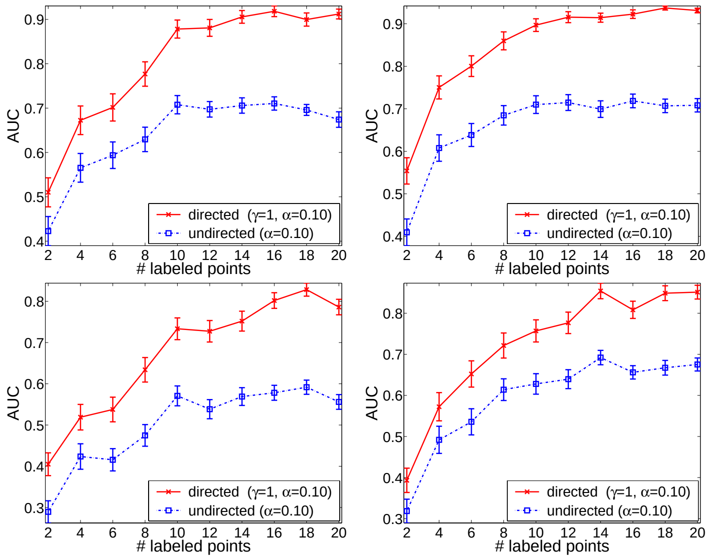


Classification on the spiral toy data. Top-left: toy data; top-right: spectral clustering; bottom-left: 2-smoothness; bottom-right: 1-smoothness.

Ranking and classification on directed graphs

- The differential geometry on undirected graphs can be naturally generalized to directed graphs. (See the references listed in the last slide.)

The importance of directionality



Classification on the WebKB dataset: *student* vs. the rest in each university. Taking the directionality of edges into account can yield substantial accuracy gains.

References

Available at http://www.kyb.mpg.de/~zhou

- **Riemannian Geometry** D. Zhou and B. Schölkopf. *Transductive Inference with Graphs*. Submitted to NIPS 2004.
- **Directed Graphs** D. Zhou, B. Schölkopf and T. Hofmann. *Semi-supervised Learning on Directed Graphs*. Submitted to NIPS 2004.
- **Undirected Graphs** D. Zhou, O. Bousquet, T.N. Lal, J. Weston and B. Schölkopf. *Learning with Local and Global Consistency*. NIPS 2003.
- **Ranking** D. Zhou, J. Weston, A. Gretton, O. Bousquet and B. Schölkopf. *Ranking on Data Manifolds*. NIPS 2003.

Conclusions

- Built a self-contained differential geometry on graphs.
- Proposed the powerful classification and ranking algorithms.

Nothing is more practical than a good theory. — Vapnik

Some searching results from Google

Google can provide us answers to many interesting questions

- "friend" vs. "enemy": 92,700,000 vs. 9,290,000: **good news! so many friends!** "war" vs. "peace" : 97,500,000 vs. 28,500,000: **however, more wars in our world! what's wrong?**
- "love" vs. "sex": 117,000,000 vs. 215,000,00: **too much sex without love**; "love" vs. "marriage" 117,000,000 vs. 19,600,000: **at present people consider getting married very seriously . (Note also compare sex vs. marriage!)**
- "man" vs. "woman": 233,000,000 vs. 57,600,000 : **more men;** "penis" vs. "pussy": 30,500,000 vs. 40,300,000: **but more pussies.**
- "start" vs. "finish": 118,000,000 vs. 18,700,000: **people always have many "great" plans, but only a small part of them are finished!**
- "today" vs. "tomorrow" : 22,900,000 vs. 16,900,000: **today is more important than tomorrow. Don't be always dreaming of tomorrow!**