1

Ranking Data with Graph Laplacian

Ayhan Demiriz
Department of Industrial Engineering, Sakarya University, 54187 TURKEY

CONTENTS

Ranking the data continues to be an important research problem in various fields. The problem can be formulated in many ways. One way is to study ordering the data with ordinal values, for instance ranking the preferences. This is a very useful approach to collaborative filtering and ordinal regression problems. Database marketing mainly focuses on "scoring" the customers to determine the target population. Thus, from a database marketing point of view, ranking is related to scoring the consumer base. On the other hand, we might be interested in finding the "importance" of the data points from an analysis perspective. In this paper, we utilize the spectral properties of graph Laplacian to rank the data. By ranking the data, we mean that determining which points are more important if the data forms a graph. We propose a framework to rank several benchmark datasets and visualize the ranking results.

1.1 Introduction

Ranking the data is an ongoing research area with diverse applications (Cao et al., 2007). We propose a ranking algorithm based on graph Laplacian (Belkin & Niyogi, 2004; Belkin et al., 2006). Our ranking algorithm resembles the algorithm proposed in (Zhou et al., 2004). The primary objective in (Zhou et al., 2004) is to develop an algorithm based on some semi-supervised approach to rank the items for a given query. The method proposed in (Zhou et al., 2004) can exploit the intrinsic manifold structure of the data. The approach in (Zhou et al., 2004) can be seen as an extreme case of semi-supervised learning in which only positive labeled points are provided

1-58488-344-8/04/$0.00+$1.50

to the algorithm.

Formally, ranking is defined as finding a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ that orders the data $X \in \mathbb{R}^n$ correctly. The framework proposed in this paper is based on graph representation of the data. Thus a graph $G = (V,E)$ can be formed from X by Euclidean neighborhood relations where $x \in X$ is represented by the vertices V and the relationships are represented by the edges $E \subseteq V \times V$ on the graph.

In this paper, we utilize spectral graph theory to tackle the ranking problem (Butler, 2006). Essentially, we use spectral properties of normalized Laplacian which is defined as $\mathscr{L} = D^{-1/2}LD^{-1/2} = D^{-1/2}(D-W)D^{-1/2} = I - D^{-1/2}WD^{-1/2}$ where W is the adjacency matrix, D is a diagonal matrix formed by row sums of W, L is traditional Laplacian matrix i.e. $D-W$, and I is the identity matrix. One of the most important spectral properties of the normalized Laplacian is that its eigenvalues vary between 0 and 2. If there are multiple eigenvalues which are equal to 0 then the underlying graph is not connected. An eigenvalue of 2 indicates that the graph is bipartite. On the other hand, we know from the convergence of the random walk that the stationary distribution, π, of the random walk is equivalent to the eigenvector corresponding to eigenvalue 1 of the underlying transition matrix i.e. $P = D^{-1}W$. In other words, the corresponding eigenvector for this transition matrix,P, can easily be shown that is equal to $\pi = \frac{\mathbf{1}D}{\sum_\ell d_\ell}$. This particular stationary distribution is achieved, if the graph is connected.

Practically, there is no need to use the power method to find the stationary distribution once it is shown that the underlying graph is connected. Otherwise, we can utilize an algorithm similar to Google's PageRank (Page et al., 1998) which is not necessarily symmetric (undirected) to find the stationary distribution of the random walk (Zhou et al., 2004). Our approach differs from (Zhou et al., 2004) as we do not attempt a semi-supervised approach. In fact, our approach does not utilize any class labels at all. Our proposed method simply ranks the data. We report ranking results of some benchmark datasets and visualize them by plotting with first-two principal components of the high dimensional data i.e. X.

The organization of the paper is as follows. We give a brief summary of application of the graph Laplacian in machine learning problems in Section 1.2. We then introduce some preliminary concepts and earlier work on using the graph Laplacian in ranking problems in Section 1.3. In Section 1.4, we explain the theoretical justification to our approach by introducing spectral graph theory. The framework and the results on some real datasets are given in Section 1.5. The paper ends with a conclusion in which we also point our future work in Section 1.6.

1.2 Using Graph Laplacian in Machine Learning Problems

The manifold learning via graph Laplacian has recently been studied by several authors. In this section, we will briefly summarize some preliminaries of the graph

Laplacian and show its strengths on various machine learning problems -especially semi-supervised learning problem. The starting point for the graph Laplacian is the adjacency matrix W. We can construct it from the nearest neighbors (binary), Euclidean distance or heat kernel. For instance, $w_{ij} = 1$ if points x_i and x_j are close -i.e. x_j is one of the k-nn of the x_i-, $w_{ij} = 0$ otherwise. The geodesic distance is defined as the shortest distance between two vertices on the adjacency graph. Notice that we can find the shortest geodesic distance of an unlabeled point and a labeled point.

After constructing W accordingly, the question arises how we can utilize such information in our learning process? In order to study spectral properties of the Laplacian, we can compute p eigenvectors corresponding to the smallest eigenvalues for the eigenvector problem :$Le = \lambda e$ where matrix $L = D - W$ is the graph Laplacian for the adjacency graph and D is diagonal matrix with the same size of W in which D_{ii} is equal to sum of corresponding row i in W.

Laplacian is a symmetric, semi-definite matrix which can be thought of as an operator on functions defined on vertices of the graph. The eigenfunctions can be interpreted as a generalization of the low frequency Fourier harmonics on the manifold defined by the data points (Belkin & Niyogi, 2004).

$$E = \begin{pmatrix} e_{11} & e_{12} & \dots & e_{1m} \\ e_{21} & e_{22} & \dots & e_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ e_{p1} & e_{p2} & \dots & e_{pm} \end{pmatrix}$$

where m is the total number of points i.e. both labeled (l) and unlabeled (u) data and p is the number of eigenfunctions we wish to employ. The error for the classification problem can be determined by

$$Err(a) = \sum_{i=1}^{l} (y_i - \sum_{j=1}^{p} a_j e_{ji})^2 \tag{1.1}$$

where the sum is taken over all labeled points -labels are denoted by y_i- and the minimization problem is considered over the space of variables $a = (a_1, \dots, a_p)^T$. Then we can find the a as follows (Belkin & Niyogi, 2004):

$$a = (E_{lab}^T E_{lab})^{-1} E_{lab}^T y \tag{1.2}$$

where $y = (y_1, \dots, y_l)^T$ and E_{lab} is constructed just for the labeled points. Then we can classify unlabeled points using following formula (Belkin & Niyogi, 2004):

$$y_i = \begin{cases} 1, & \text{if } \sum_{j=1}^{p} e_{ij} a_j \geq 0 \\ -1, & \text{if } \sum_{j=1}^{p} e_{ij} a_j < 0 \end{cases}$$

1.2.1 Semi-supervised learning via Graph Laplacian

By using unlabeled data, one might intuitively expect understanding of the marginal distribution of $\mathscr{P}_X$ better in the learning process. However there is no direct relationship between $\mathscr{P}_X$ and conditional $\mathscr{P}(y|x)$. The main assumption in (Belkin et al., 2004; Belkin et al., 2006) is that if two points $x_1, x_2 \in X$ are close in intrinsic geometry of $\mathscr{P}_X$, then the conditional distributions $\mathscr{P}(y|x_1)$ and $\mathscr{P}(y|x_2)$ are similar. This is a very useful assumption to motivate semi-supervised learning in a more concrete way.

The standard regularization framework can be summarized by the following optimization problem.

$$f^* = \arg\min_{f \in H_k} \frac{1}{l} \sum_{i=1}^{l} V(x_i, y_i, f) + \gamma \|f\|_K \tag{1.3}$$

This optimization results in a solution as follows (Belkin et al., 2006):

$$f^*(x) = \sum_{i=1}^{l} \alpha_i K(x_i, x)$$

When $\mathscr{P}_X$ is known a priori, the related optimization problem of regularization with additional information from the unlabeled data can be formulated as follows (Belkin et al., 2006):

$$f^* = \arg\min_{f \in H_k} \frac{1}{l} \sum_{i=1}^{l} V(x_i, y_i, f) + \gamma_A \|f\|_K + \gamma_I \|f\|_I \tag{1.4}$$

By utilizing Representer Theorem, we can find the optimum f^* as follows (Belkin et al., 2006):

$$f^*(x) = \sum_{i=1}^{l} \alpha_i K(x_i, x) + \int_{\mathscr{M}} \alpha(y) K(, y) d\mathscr{P}_X(y)$$

where $\mathscr{M} = supp\mathscr{P}_X$ is the support of the marginal $\mathscr{P}_X$.

In the case of $\mathscr{P}_X$ is not known a priori, we can utilize the unlabeled data to estimate the underlying distribution. Particularly, we can estimate empirically $\mathscr{P}_X$ and $\| \; \|_I$. Recently, some work has been conducted when the support of $\mathscr{P}_X$ is a compact submanifold $\mathscr{M} \subset X = \mathbb{R}^d$. In this case a natural choice for $\|f\|_I$ is $\int_{\mathscr{M}} \langle \nabla_{\mathscr{M}} f, \nabla_{\mathscr{M}} f \rangle$. Then we have an optimization problem as follows (Belkin et al., 2006):

$$f^* = \arg\min_{f \in H_k} \frac{1}{l} \sum_{i=1}^{l} V(x_i, y_i, f) + \gamma_A \|f\|_K + \gamma_I \int_{\mathscr{M}} \langle \nabla_{\mathscr{M}} f, \nabla_{\mathscr{M}} f \rangle \tag{1.5}$$

The term $\int_{\mathscr{M}} \langle \nabla_{\mathscr{M}} f, \nabla_{\mathscr{M}} f \rangle$ may be approximated by the usage of labeled and unlabeled data in a graph Laplacian framework. Thus optimization problem takes the following form.

$$f^* = \arg\min_{f\in H_k} \frac{1}{l}\sum_{i=1}^{l} V(x_i,y_i,f) + \gamma_A\|f\|_K + \frac{\gamma_I}{n^2}\sum_{i,j=1}^{n}(f(x_i)-f(x_j))^2 W_{ij} \tag{1.6}$$

$$= \arg\min_{f\in H_k} \frac{1}{l}\sum_{i=1}^{l} V(x_i,y_i,f) + \gamma_A\|f\|_K + \frac{\gamma_I}{n^2}\mathbf{f}^T L\mathbf{f} \tag{1.7}$$

1.2.2 Support Vector Classification and Laplacian Extension

In this part of the paper, we briefly give the formulation for the Laplacian extension of the SVM problem. This is just an illustration to depict the usage of the Laplacian on a well studied SVM problem. In general, the following optimization problem is solved for the SVMs. Note that the classification problem is defined for the labeled points.

$$\min_{f\in H_K} \frac{1}{l}\sum_{i=1}^{l}(1-y_i f(x_i))_+ + \gamma\|f\|_K$$

where the hinge loss is defined as $(1-yf(x))_+ = \max(0,1-yf(x))$ and the labels $y_i \in \{-1,+1\}$.

The primal form of SVM problem can be reformulated as follows (Belkin et al., 2006):

$$\begin{aligned} &\min_{f,\xi} && \frac{1}{l}\sum_{i=1}^{l}\xi_i + \gamma\|f\|_K \\ &\text{subject to:} && y_i f(x_i) \geq 1-\xi_i \quad i=1,\dots,l \\ & && \xi_i \geq 0 \quad i=1,\dots,l \end{aligned} \tag{1.8}$$

We can write the dual formulation of SVM problem as:

$$\begin{aligned} &\max_{\beta\in\mathbb{R}^l} && \sum_{i=1}^{l}\beta_i - \frac{1}{2}\beta^T Q\beta \\ &\text{subject to:} && \sum_{i=1}^{l} y_i\beta_i = 0 \\ & && 0 \leq \beta_i \leq \frac{1}{l} \quad i=1,\dots,l \end{aligned} \tag{1.9}$$

Then we have the following solution to dual problem

$$\begin{aligned}
Y &= diag(y_1,\ldots,y_l) \\
Q &= Y\left(\frac{K}{2\gamma}\right)Y \\
\alpha^* &= \frac{Y\beta^*}{2\gamma} \qquad (1.10)
\end{aligned}$$

By incorporating the unlabeled data we can rewrite the general optimization formulation as follows (Belkin et al., 2006):

$$\min_{f\in H_K} \frac{1}{l}\sum_{i=1}^{l}(1-y_if(x_i))_+ + \gamma_A\|f\|_K + \frac{\gamma_I}{m^2}\mathbf{f}^T L\mathbf{f}$$

The primal form of SVM problem can be reformulated as follows:

$$\begin{aligned}
&\min_{\alpha\in\mathbb{R}^m,\xi\in\mathbb{R}^l} && \frac{1}{l}\sum_{i=1}^{l}\xi_i + \gamma_A\alpha^T K\alpha + \frac{\gamma_I}{m^2}KLK\alpha \\
&\text{subject to:} && y_i(\sum_{j=1}^{m}\alpha_jK(x_i,x_j)+b) \geq 1-\xi_i \quad i=1,\ldots,l \qquad (1.11) \\
& && \xi_i \geq 0 \quad i=1,\ldots,l
\end{aligned}$$

That will give us a solution for α as follows:

$$\alpha = (2\gamma_A I + 2\frac{\gamma_I}{m^2}LK)^{-1}J^TY\beta^*$$

where J is $l\times l$ diagonal matrix given by $J = diag(|Y|)$ and β^* is the solution to dual SVM problem in which Q is computed by following equation

$$Q = YJK(2\gamma_A I + 2\frac{\gamma_I}{m^2}LK)^{-1}J^TY$$

1.3 Preliminaries and Earlier Work

The idea of the graph Laplacian has been utilized in several earlier work. Most notably, the approach in (Zhou et al., 2004) is very interesting and shows some resemblance to Google's PageRank algorithm (Page et al., 1998). The primary objective in (Zhou et al., 2004) is to develop an algorithm based on some semi-supervised approach to rank the items for a given query. The method can exploit the intrinsic manifold structure of data. Authors emphasize that their approach can be considered as an extreme case of semi-supervised learning in which only positive labeled points are provided to the algorithm. Notice that we do not attempt to develop a semi-supervised learning algorithm in this paper.

We briefly summarize the algorithm proposed in (Zhou et al., 2004). Given a set of points $X = \{x_1, \ldots, x_l, \ldots, x_m\} \subset \mathbb{R}^n$, the first l points are the queries and the rest are the data that we want to rank. Suppose we have distance function $d(x_i, x_j)$ defined between two points. Moreover, define the vector $Y = [y_1, \ldots, y_m]$ where $y_i = 1$ for $i = 1, \ldots, l$ and $y_i = 0$ for $i = l+1, \ldots, m$. Let function $g : X \to \mathbb{R}$ assign a ranking value g_i for each given x_i. The algorithm can be summarized as follows (Zhou et al., 2004):

- Sort the pairwise distances among points in ascending order. Based on this order, connect two points with an edge until a connected graph is obtained.
- Form adjacency matrix W defined by $W_{ij} = exp[-d^2(x_i, x_j)/2\sigma^2]$ if there is an edge linking points x_i and x_j. Note that $W_{ii} = 0$ since loops are not allowed in the graph.
- Symmetrically normalize W by $S = D^{-1/2}WD^{-1/2}$ where D is the diagonal matrix with the entries from the sum of rows of W.
- Iterate $g(t+1) = \alpha S g(t) + (1-\alpha)Y$ until convergence, where $\alpha \in [0,1)$.
- Let g_i^* be the limit of the sequence $\{g_i(t)\}$. Rank each point x_i according to its ranking score g_i^* (largest first)

Theorem 1 of (Zhou et al., 2004) states that the sequence $\{g(t)\}$ converges to $g^* = \beta(I - \alpha S)^{-1}Y$ where $\beta = 1 - \alpha$. As seen from the above algorithm, (Zhou et al., 2004) utilizes a variation of the power method to find the steady state distribution. Thus Theorem 1 of (Zhou et al., 2004) provides a formal explanation that the algorithm guaranties finding a steady state distribution that can be used as a ranking function. The idea of using long-term steady state distribution as a ranking function is also utilized in our paper. However, instead of using the labeled points as a query in the power method, we either find the steady state distribution with a closed form solution or start from the closed form solution and use the power method to find the converged distribution.

In (Agarwal, 2006), the ranking preferences problem is solved by using the graph Laplacian. A regularized hinge loss function is minimized via an optimization problem which resembles SVM's quadratic programming formulation. Both undirected and directed graphs are considered in (Agarwal, 2006). Basically, a regularizing real function f is found in (Agarwal, 2006) that its values do not vary rapidly across the neighboring vertices. For the undirected case, solution to the following problem is sought in (Agarwal, 2006):

$$\min_{f:V\to\mathbb{R}} \{\hat{R}_{\ell_h}(f;\Gamma) + \lambda \mathtt{f}^{\mathtt{T}}\mathtt{Lf}\}$$

where ℓ_h is the hinge loss function. A QP model is solved in (Agarwal, 2006). The solution for the QP contains pseudo-inverse Laplacian term. For the directed graph case, the same algorithmic approach can be used as in the undirected case. However, a random walk that converges to a steady state distribution is used as in (Zhou et al., 2005). The implementations in (Zhou et al., 2005; Agarwal, 2006) resemble the PageRank algorithm from the random walk point of view. Teleporting is used again to ensure the convergence of the random walk. Generalization properties are also discussed in (Agarwal, 2006) by showing that column-space of the psuedo-inverse of the Laplacian is an RKHS.

The graph Laplacian is applied to collaborative filtering problems in (Fouss et al., 2007). The problem in (Fouss et al., 2007) is posed again as a random walk problem on graphs and the pseudo-inverse of the Laplacian is used as a similarity measure for collaborative filtering purposes. In the following section we will summarize the spectral properties of Laplacian to present a closed form solution of the steady state distribution of the random walk which upon we build our approach.

1.4 Spectral Graph Theory

As spectrum plays an important role in all physical sciences, the spectrum reveals a lot about the underlying graph. The spectrum is simply equivalent to the eigenvalues of the corresponding matrix. Thus there is an important relationship between the eigenvalues of the matrices and the corresponding graph structures. The study between these two is known as spectral graph theory (Butler, 2006).

In this section, we will introduce some properties of Laplacian graphs from a spectral graph theory perspective. There are three important matrices namely adjacency, Laplacian (L) and normalized Laplacian ($\mathscr{L}$) when it comes to spectral graph theory. As we give the definition above in Section 1.2, the adjacency matrix, W, can be constructed in a binary manner.

We can count the number of walks of length k starting at vertex i and ending at vertex j by simply using $(W^k)_{i,j}$. The trace of W is the sum of its eigenvalues i.e. 0 and the eigenvalues of W^k are the eigenvalues of W raised to the kth power.

The traditional Laplacian L, as we defined above, is equal to $D-W$. There is a special eigenvalue of L namely 0. Since the sum of the rows of L is 0, the eigenvector 1 corresponds to eigenvalue 0. All the other eigenvalues of L are nonnegative in other words L is positive semi-definite (Butler, 2006).

Normalized Laplacian, $\mathscr{L}$, is defined as $\mathscr{L} = D^{-1/2}LD^{-1/2} = D^{-1/2}(D-W)D^{-1/2} = I - D^{-1/2}WD^{-1/2}$. Elementwise $\mathscr{L}$ is defined as follows (Butler, 2006):

$$\mathscr{L}_{i,j} = \begin{cases} 1 & \text{if } i = j; \\ \frac{-1}{\sqrt{d_i d_j}} & \text{if } i \text{ is adjacent to } j; \\ 0 & \text{otherwise} \end{cases}$$

As in the case of the Laplacian, the normalized Laplacian has also non-negative eigenvalues. The difference is that in the Laplacian we can have the eigenvalues as large as possible, however the normalized Laplacian has eigenvalues in the range of 0 and 2, both ends are inclusive.

1.4.1 Random Walks

A random walk on a graph G is considered as a walk that starts from a vertex and moves to an adjacent vertex through a randomly picked edge for a number of steps - as many as required. Randomness is satisfied when the initial state does not affect the current state anymore. In other words, knowing the initial state does not give any significant information about the current state anymore (Butler, 2006). Thus we can easily say that the walk is random if the probability of being in any vertex is equal to its degree (Butler, 2006).

We can practically utilize the random walks to study ranking the data. The convergence of the random walks will yield us the theoretical steady state distribution of the vertices i.e. data points. In order to find the steady state distribution, we need a transition matrix which is defined as $D^{-1}W$. In other words,

$$(D^{-1}W)_{i,j} = \begin{cases} 0 & \text{if } i \text{ is not adjacent to } j; \\ \frac{1}{d_i} & \text{if } i \text{ is adjacent to } j; \end{cases}$$

is the probability of moving from vertex i to vertex j. The probability distribution after k steps will be $f(D^{-1}W)^k$.

The relationship between the normalized Laplacian and the transition matrix can be given by the following equation.

$$D^{-1/2}(I-\mathscr{L})D^{1/2} = D^{-1/2}(D^{-1/2}WD^{-1/2})D^{1/2} = D^{-1}W.$$

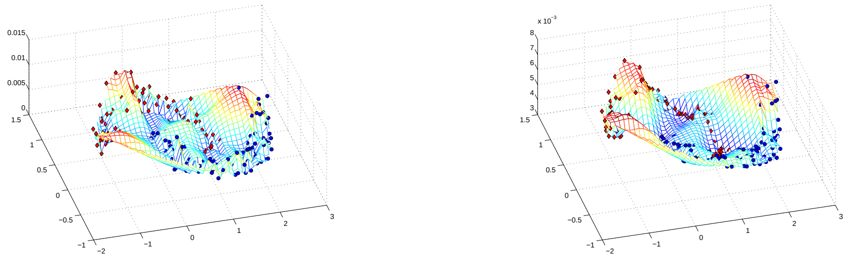
If λ is an eigenvalue of $\mathscr{L}$ then $(1-\lambda)$ is an eigenvalue of $I-\mathscr{L}$ (Butler, 2006) with the same eigenvector. Particularly, since $\mathbf{0}$ is an eigenvalue of $\mathscr{L}$ then $\mathbf{1}$ must be an eigenvalue of $D^{-1}W$. The corresponding *left* eigenvector can be shown to be $\mathbf{1}D$ (Butler, 2006). The stationary (steady state) distribution of a well-connected aperiodic graph is equal to

$$\pi = \frac{\mathbf{1}D}{\sum_\ell d_\ell}$$

In other words, random walk will converge to the above distribution. Let ϕ_i be an orthonormal set of eigenvectors corresponding to λ_i for $\mathscr{L}$. Notice that $0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_{n-1} \leq 2$. By above argument, ϕ_i is also related to $1-\lambda_i$ for $D^{-1/2}WD^{-1/2}$. Then it is easy to show that $\phi_0 = \mathbf{1}D^{1/2}/\sqrt{\sum_\ell d_\ell}$. Since we have the full set of orthonormal eigenvectors, we can use the idea of projections onto eigenspaces to write (Butler, 2006)

$$D^{-1/2}WD^{-1/2} = \sum_i (1-\lambda_i)\phi_i^T \phi_i$$

To check how close we are after k-steps to convergence, we can use L^2-norm



(a) (b)

FIGURE 1.1

Two Moons Dataset Ranking Results

distance measure (Butler, 2006).

$$
\begin{aligned}
\|f(D^{-1}W)^k - \frac{\mathbf{1}D}{\sum_\ell d_\ell}\| &= \|fD^{-1/2}(D^{-1/2}WD^{-1/2})^k D^{1/2} - \frac{\mathbf{1}D}{\sum_\ell d_\ell}\| \\
&= \|fD^{-1/2}(\sum_i (1-\lambda_i)\phi_i^T\phi_i)^k D^{1/2} - \frac{\mathbf{1}D}{\sum_\ell d_\ell}\| \\
&= \|fD^{-1/2}(\sum_i (1-\lambda_i)^k\phi_i^T\phi_i) D^{1/2} - \frac{\mathbf{1}D}{\sum_\ell d_\ell}\| \\
&= \|fD^{-1/2}(\sum_{i\neq 0} (1-\lambda_i)^k\phi_i^T\phi_i) D^{1/2}\| \\
&\le \max_{i\neq 0} |1-\lambda_i|^k \frac{\max_i \sqrt{d_i}}{\min_j \sqrt{d_j}}.
\end{aligned}
$$

The last inequality enables the usage of eigenvalues in approximating the error of convergence after k-step. In addition, we can easily conclude that the more closely eigenvalues are gathered around 1 for the normalized Laplacian, $\mathscr{L}$, the faster convergence we should expect to the steady state distribution. Since the eigenvalues are in between 0 and 2 for $\mathscr{L}$, the inequality $\max_{i\neq 0}|1-\lambda_i| \le 1$ holds.

The term $\max_{i\neq 0}|1-\lambda_i|$ will be equal to 0 in two cases. If there are multiple zeroes as eigenvalues and the largest eigenvalue is two. In the first case it can be shown that the graph is not connected. In the second case, the graph is said to be bipartite.

1.5 Random Walk Framework and Experimental Evaluation

Even though by analyzing the spectrum of $\mathscr{L}$ we can come to some conclusions about the underlying graph structure and the convergence of the steady state distri-

TABLE 1.1
The Summary of the Datasets

Name	# of Rows	Dimensionality	Type
Two Moons	200	2	Classification
Bank Notes	200	6	Classification
NBA	453	12	Ranking
MLB	607	17	Ranking
USPS	1200	100	Classification

bution, we can still utilize a PageRank algorithmic approach to find the steady state distribution π regardless of a spectral analysis.

In this section, we will show how we can utilize the theoretical steady state distribution of a well connected and aperiodic graph i.e. $\frac{\mathbf{1}D}{\sum_\ell d_\ell}$ in ranking the data. We present the ranking results visually by plotting them with the first two principle components of the each benchmark datasets. Our framework can be summarized as follows:

1. Given dataset $X \in \mathbb{R}^n$ construct adjacency matrix W and D
2. Compute the theoretical steady state distribution $\pi = \frac{\mathbf{1}D}{\sum_\ell d_\ell}$
3. By using π as the query vector, run the PageRank algorithm to find the steady state distribution
4. Visualize the ranking results using principal components.

As we mentioned in Section 1.3, we use the closed form solution of the stationary distribution, π to start the PageRank algorithm. Thus instead of choosing a random start, we prefer running the PageRank algorithm with π as the query vector.

There are several ways that we can construct the adjacency matrix W and D. The neighborhood boundaries can be defined by k-nn and ε-ball. Throughout the paper we only utilize 8 nearest neighbors based on the Euclidean distance to determine the neighborhood boundaries. In addition, W can be constructed by binary relationship, Euclidean distance measure and heat kernel i.e. $exp[-d^2(x_i,x_j)/2\sigma^2]$ between point x_i and point x_j. We utilize the Euclidean distance measure in constructing W to represent the strength of the relationship between points.

1.5.1 Evaluation of the Framework

We use some real and benchmark datasets to study the applicability of the framework given above. The datasets vary in the size and type (i.e. task). The summary of the datasets are given in Table 1.1.

We first run the ranking on a well-known toy problem two moons dataset. Note that our framework does not involve any learning step, however we use the class information for the visualization purposes. There are 200 hundred points in two-dimensional space. Figure 1.1 depicts the ranking results of the two moons dataset.

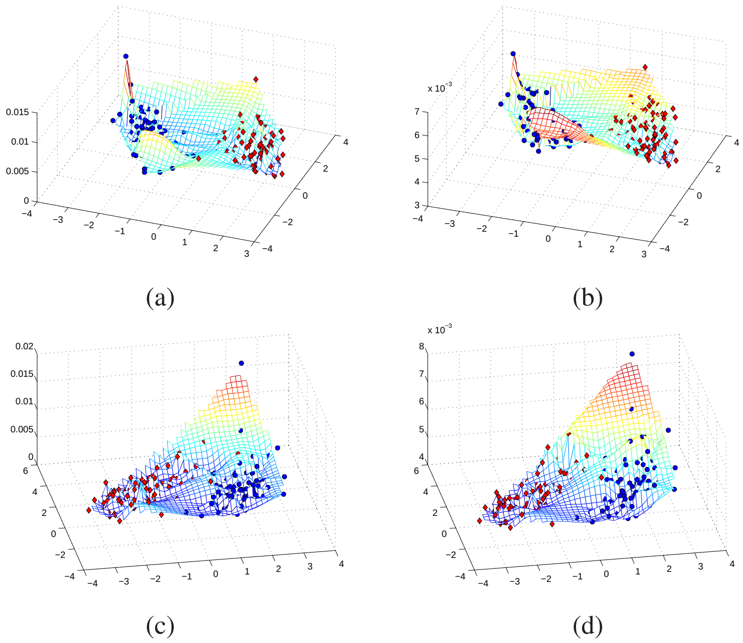
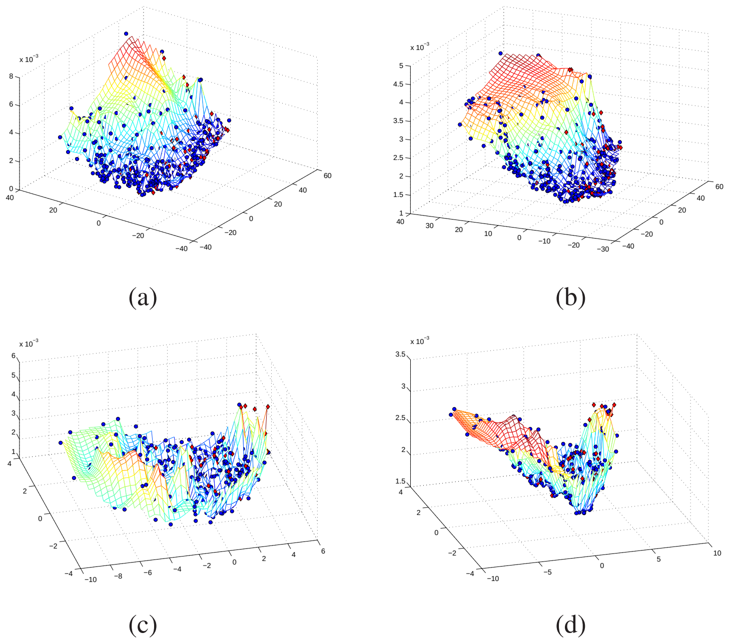


FIGURE 1.2

Bank Notes Dataset Ranking Results



(a) (b)

(c) (d)

FIGURE 1.3

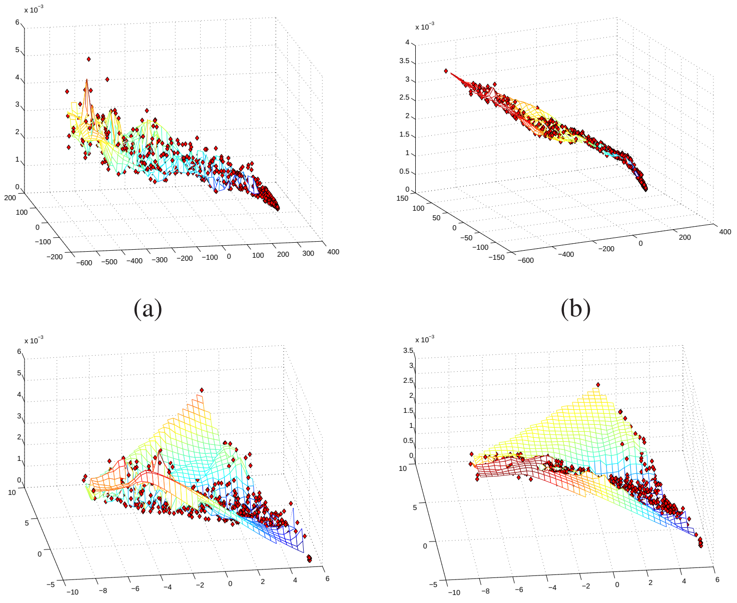
NBA Dataset Ranking Results

Basically Figure 1.1(a) visualizes the theoretical steady state distribution π. The steady state distribution found by PageRank is visualized in Figure 1.1(b). Since the dataset is two-dimensional, the original dimensions are used in the plot. The figures reveal that a smoother stationary distribution is found after running the PageRank algorithm as one might expect. Due to the nature of ranking, we may consider the points with higher ranking values as cluster centers. Thus an indirect clustering is apparent from the figure. Notice that clustering is based on the geodesic distance (see Section 1.2).

We apply our ranking framework to some benchmark datasets. The bank notes dataset (Flury & Riedwyl, 1988) contains two hundred data points: 100 of them belong to "forged" Swiss bank notes and 100 of them belong to "genuine" Swiss bank notes. There are six features in the bank notes dataset. We also collected the data from NBA rankings (2006-2007 season)[*] and MLB batting rankings (year 2006)[†]. There are 12 statistics collected for the 453 NBA players compared to 17 collected for the 607 MLB players. The purpose of using sports ranking data is to see how our framework will behave on naturally ranked data. Notice that sports rankings are also a form of preferences e.g. MVP has better stats than all the other players. We

[*] www.NBA.com

[†] www.usatoday.com/sports/baseball/stats-archive.htm



(a) (b)

(c) (d)

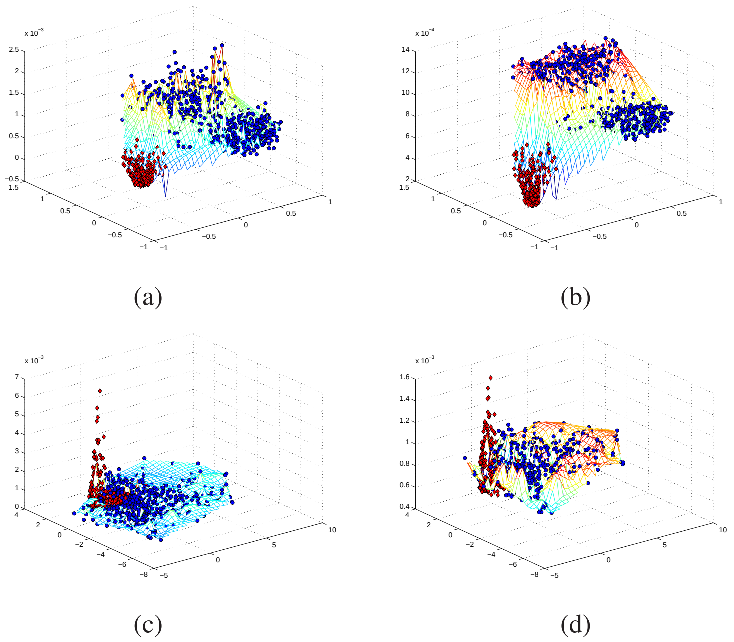
FIGURE 1.4

MLB Dataset Ranking Results

can then actually evaluate our framework on the preference (ordinal) data. We also use well-known benchmark dataset USPS (Asuncion & Newman, 2007): one digit (0) versus two other digits (2 and 7) which corresponds to 1200 total data points.

Figure 1.2 depicts the ranking results of the bank notes dataset. This dataset is linearly separable. Since the dataset is high dimensional as in the other benchmark datasets, the ranking values are plotted versus first two principal components. Again the Figures 1.2(a) and 1.2(b) depict ranking results based on the theoretical stationary distribution π and stationary distribution after the PageRank algorithm respectively. The issue of normalization (e.g. standard normalization) is highly important in analyzing multivariate data. The Figures 1.2(c) and 1.2(d) depicts stationary distributions using standard normalized data (X) in finding adjacency matrix W.

Using sports ranking data from NBA and MLB, we can evaluate our framework on “real” ranking problems. Sports rankings are highly popular domain for statistics. In a short explanation, the players with better statistics rank higher. For instance for the 2006 and 2007 NBA season Kobe Bryant has the best statistics, so he is ranked number 1. For the NBA dataset, we created a new label to differentiate rookie players from the regular player. So there are 77 rookie players and 376 regular players for the 2006-2007 season. Figures 1.3 and 1.4 depict the ranking results of our framework on sports ranking data. As in the bank notes dataset, we use the standard normalized data in parts (c) and (d) of the corresponding figures. After using NBA dataset, we



(a) (b)

(c) (d)

FIGURE 1.5

USPS Dataset Ranking Results

get ranking results in Figures 1.3(a) and 1.3(b) that Peja Stojakovic has the highest ranking. Notice that as in the case of web page ranking, the importance is determined by the size of the inbound links to a node. Thus we can conclude that Peja Stojakovic is in the middle of a larger cluster. For the standard normalized data (see Figure 1.3(c)), James Augustine who is a rookie player and is ranked number 438 has the highest ranking result. Since this player is in the middle of a large group of players who have poor performances - in other words the most of the statistics are near 0-, he has the highest ranking results from our ranking framework. However, after running PageRank algorithm (see Figure 1.3(d)) the player Pau Gasol who is ranked number 21 ends up with the highest ranking results according to our ranking framework.

Similar to NBA players, MLB players are also ranked with our framework in Figure 1.4 according to their batting statistics. Matthews has the highest ranking using unnormalized data in Figures 1.4(a) and 1.4(b). But Patterson and Reyes have the highest rankings in Figures 1.4(c) and 1.4(d) respectively on standard normalized data. Notice that these players are not the best ones in the league. Again the more inbound connection the player has, the higher ranking results he gets.

We also use USPS dataset (Asuncion & Newman, 2007) in our experiments. As mentioned above, we use only three digits in our experiment: one digit vs. two other digits. There are totally 1200 data points in 100 dimensional space for our particular experiment. Surprisingly, Figures 1.5(a) and 1.5(b) show that there is a clear ranking difference among digits. Although the rest of the datasets used in our experiments have yielded mixed ranking results between classes, we observe a clear separation in ranking results from one class to another one in USPS dataset. We can still see the separation on normalized data in Figure 1.5(c), but we loose this separation after running PageRank algorithm (see Figure 1.5(d)).

1.6 Conclusion and Future Work

We introduced a framework to rank the data using ideas from the graph Laplacian. We also utilized this framework on some real and benchmark datasets. The approach certainly has some interesting advantages. The important finding is that there is no need to use a power method such as PageRank algorithm to find the stationary distribution when we deal with undirected graphs. We can simply use a closed form solution for the stationary distribution. Even if the PageRank algorithm is used to compute the stationary distribution, it is well know that it will converge for the directed graphs. Therefore we can safely use the stationary distribution to rank the data (i.e. nodes) by their structural effects on the graph.

Since the ranking framework introduced in this paper lists the nodes (states) of the graphs by their structural importance, our ranking results can be used as an input to develop some search algorithms on the graph domain. For example, a possible application domain might be the famous traveling salesman problem. Similar network problems such as transportation problem is also a likely application area of the ranking results. In other words, we can develop certain search algorithms that use ranking scores to search the solution space more efficiently.

Acknowledgement

Some parts of the work for this paper were completed when the author was visiting University College of London through funding from EU PASCAL Network of Excellence. The author also would like to thank to John Shawe-Taylor for reviewing an early draft of this paper.

Bibliography

Agarwal, S. (2006). Ranking on graph data. *Proceedings of the 23rd International Conference on Machine Learning*.

Asuncion, A., & Newman, D. (2007). UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

Belkin, M., & Niyogi, P. (2004). Semi-supervised learning on riemannian manifolds. *Journal of Machine Learning*, *56*, 209–239.

Belkin, M., Niyogi, P., & Sindhwani, V. (2004). *Manifold regularization: A geometric framework for learning from examples* (Technical Report TR-2004-06). Dept. of Computer Science, University of Chicago.

Belkin, M., Niyogi, P., & Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, *7*, 2399–2434.

Butler, S. (2006). Spectral graph theory: Three common spectra. Talks given at the Center for Combinatorics, Nankai Unniversity, Tianjin.

Cao, Z., Qin, T., Liu, T., Tsai, M., & Li, H. (2007). Learning to rank: from pairwise approach to listwise approach. In Z. Ghahramani (Ed.), *Proceedings of the 24th annual international conference on machine learning (icml 2007)*, 129–136. Omnipress.

Flury, B., & Riedwyl, H. (1988). *Multivariate statistics: A practical approach*. London: Chapman and Hall.

Fouss, F., Pirotte, A., & Saerens, M. (2007). Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *19*, 355–369.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The pagerank citation ranking: Bringing order to the web.

Zhou, D., Huang, J., & Schölkopf, B. (2005). Learning from labeled and unlabeled data on a directed graph. *Proceedings of the 22nd International Conference on Machine Learning*.

1-58488-344-8/04/$0.00+$1.50

Zhou, D., Weston, J., Gretton, A., Bousquet, O., & Schölkopf, B. (2004). Ranking on data manifolds. In S. Thrun, L. Saul and B. Schölkopf (Eds.), *Advances in neural information processing systems 16*. Cambridge, MA: MIT Press.