1 Heterogeneous data comparison and gene selection with kernel canonical correlation analysis

Yoshihiro Yamanishi
Jean-Philippe Vert
Minoru Kanehisa

The integration and comparison of heterogeneous data such as biochemical pathways, genomes, gene functions, and gene expression data is a major issue in postgenomics. While integration strategies often rely on heuristic approaches specifically adapted to the nature of the data to be integrated — such as sequences, graphs, and vectors — we present in this chapter a systematic approach to the integration and comparison of virtually any types of data, as long as relevant kernel functions can be defined on the data to be compared. Tools to measure the correlation between different heterogeneous data sets and to extract sets of genes which share similarities with respect to multiple biological attributes are proposed. The originality of this approach is the extension of the concept of correlation for nonvectorial data, which is made possible by the use of generalized kernel canonical correlation analysis, and its application to the extraction of groups of genes responsible for the detected correlations.

As an application, this approach is tested on its ability to recognize operons in the *Escherichia coli* genome, from the comparison of three data sets corresponding to *functional* relationships among genes in metabolic pathways, *positional* relationships along the chromosome, and *coexpression* relationships as observed by gene expression data.

1.1 Introduction

Developments in high-throughput technologies have filled biological databases with many sorts of genomic data. Examples include genome sequences, signaling and metabolic pathways (Kanehisa et al., 2002), gene expression data (Eisen et al., 1998), protein-protein interaction data (Ito et al., 2001), phylogenetic profiles (Pellegrini et al., 1999), and several more. Investigating the relationships among these data is an important step toward a better understanding of the functions of the genes and the machinery of the cell. In particular, it is often the case that different data provide different and complementary information about the same underlying objects or processes. To fix the ideas, we focus in this chapter on gene analysis, but the principled approach we follow can easily be applied to the analysis of other biological objects or processes, such as the evolution of a disease, as soon as several different measurements about the objects or processes of interest are available.

Heterogeneous data comparison

Our approach is motivated by the classic idea that comparing different data about the same objects is a way to detect hidden or underlying relationships or phenomena. Let us suppose, for example, that an unusually strong correlation is detected between the presence of a motif in the promoter region of some genes, on the one hand, and the gene expression levels under particular conditions, on the other hand. This correlation might stem from a biological phenomenon linking the sequence and the function of the genes, such as the recognition of the motif by a transcription factor. More generally, comparing different data sets involving the same genes, such as their sequences, expression, promoter regions, or the structure of the encoded proteins, is a way to recognize biological phenomena. Moreover, if a correlation is detected among several data sets, genes mainly responsible for the observed correlation can be detected. One can expect these genes to play a special role in or be affected by the underlying biological phenomenon.

A well-known statistical method to investigate the correlation between different real-valued attributes is canonical correlation analysis (CCA) (Hotelling, 1936). However, classic CCA cannot be applied to nonvectorial genomic data, such as pathways, protein-protein interactions, or gene positions in a chromosome. In this chapter we overcome this issue by using a generalization of CCA, known as kernel CCA (KCCA) proposed by Akaho (2001) and Bach and Jordan (2002), which provides a way to perform a generalized form of CCA between any two types of data as long as kernel functions can be defined on these data. KCCA finds directions simultaneously in the two feature spaces defined by the kernel functions with maximum correlation.

Variants of kernel CCA

As a first contribution we derive two variants of KCCA in order to perform CCA on more than two data sets. The first one, which we call multiple KCCA, is a natural generalization of KCCA to more than two kernel functions. Already suggested by Bach and Jordan (2002), it consists in searching for directions simultaneously in all feature spaces by maximizing the sum of all pairwise correlations between data sets. The second one, which we call integrated KCCA, is a normal KCCA carried out between two kernels which are themselves sums of primary kernels. Integrated KCCA can be useful to extract correlations between two sets of data sets, represented by two sets of kernel functions.

Gene selection

As a second contribution, we propose a method to select genes of interest from the results of CCA. The method consists in ranking the genes in terms of the absolute value of their projection on a canonical direction, typically the first one. Large absolute values correspond to the genes mainly responsible for the detected correlation, hence selecting these genes is likely to provide groups of genes related to the biological phenomenon behind the correlation.

As an application we consider the problem of detecting operons in prokaryotic genomes. Operons are groups of adjacent genes on the genome which are transcribed together on a single messenger RNA (mRNA) molecule. Genes in an operon often code proteins involved in the same biochemical function, such as enzymes catalyzing successive chemical reactions in a pathway. As a result, the presence of operons in prokaryotes is responsible for a form of correlation among several data sets, because genes which form operons tend to be close to each other along chromosomes, to have similar expression profiles, and to catalyze successive reactions in a pathway. Conversely, one can start from three data sets containing the localization of the genes on the genome, their expression profiles, and the chemical reactions they catalyze in known pathways, and look for correlations among these data sets, in order to finally recover groups of genes, which may form operons. We provide experimental results on the unsupervised detection of operons in the *E.coli* genome by detecting correlations among the KEGG/pathways database of metabolic and signaling pathways, the positions of the genes on the genome, and microarray expression data.

The integration of heterogeneous data has been investigated with a variety of approaches so far. Motivated by graph-theoretical arguments, clusters of genes have been extracted from several biological networks using multiple graph comparison by Ogata et al. (2000) and Nakaya et al. (2001). Using classic clustering algorithms with a distance combining information from expression data and biochemical networks, Hanisch et al. (2002) were able to extract coclusters of genes. An approach using direct kernel operations was proposed by Pavlidis et al. (2001) to improve the performance of gene function prediction algorithms from expression data and phylogenetic profiles. The use of KCCA was pioneered by Vert and Kanehisa (2003b) and Vert and Kanehisa (2002) in the context of gene function prediction from gene expression data using biochemical networks as side information, and further investigated by Yamanishi et al. (2003) and Vert and Kanehisa (2003a) as a data mining tool to extract information from heterogeneous data.

1.2 Methods

In this section we present the methodology of our work. We review canonical correlation analysis, its generalization as a kernel algorithm, and propose two variants to handle more than two data sets. We then present a method to select genes from the result of CCA analysis, and finally recall the definition of the diffusion kernel used in the experiment.

1.2.1 Classic CCA

Classic CCA

Canonical correlation analysis was introduced by Hotelling (1936) as a way to measure linear relationships between random multivariate vectors $\mathbf{x}_1$ and $\mathbf{x}_2$, of respective dimension N_1 and N_2. It finds two linear transforms, one for each variable, such that one component within each transformed variable is maximally correlated with a single component in the other. More precisely, the first *canonical variates* are defined as the projections of $\mathbf{x}_1$ and $\mathbf{x}_2$ onto unit norm vectors $\alpha_1 \in \mathbb{R}^{N_1}$ and $\alpha_2 \in \mathbb{R}^{N_2}$ defined by

$$(\alpha_1, \alpha_2) := \underset{||a_1||=||a_2||=1}{\arg\max} \left| \text{corr } \left(a_1^\top \mathbf{x}_1, a_2^\top \mathbf{x}_2\right) \right|, \tag{1.1}$$

where $a^\top$ denotes the transpose of a. The first *canonical correlation* is defined as the maximum value attained in (1.1). Higher-order canonical variates and correlations are defined as in (1.1) under the additional restriction that the kth canonical variate, with $1 \leq k \leq \min(N_1, N_2)$, should be uncorrelated with all canonical variates of lower order. The problem (1.1) has a fairly simple solution (Johnson and Wichern, 1998), where α_1 and α_2 are found by eigenvector decomposition. CCA is a popular tool in exploratory data analysis to investigate the relationship between two kinds of attributes, and has found many applications in economics and medical studies, for example.

1.2.2 Kernel CCA

Kernel CCA is a generalization of CCA using the kernel trick. Proposed independently by Akaho (2001) and Bach and Jordan (2002), it consists in performing a regularized form of CCA in the feature spaces implicitly defined by two different kernels on the same objects. As an example, if objects are genes, kernel CCA can be used to investigate the relationships between gene sequences and gene expression by performing classic CCA between the genes in the features spaces defined respectively by a string kernel and a kernel for expression profiles.

In order to transform CCA into a kernel algorithm, at least two important issues must be addressed:

- Technically, the algorithm to solve CCA must be expressed in a form that only involves the data through their inner products, in order to use the kernel trick (see chapter **??**, subsection **??**) and be able to replace each such inner product by the evaluation of a kernel function.
- Theoretically, classic CCA is not adapted to large-dimensional variables. In particular, when the dimension of each space exceeds the number of points available, perfect canonical correlation can always be found between any sets of variables. While this issue is well-known by practitioners of classic CCA, it becomes problematic with kernels that correspond to high-dimensional feature spaces, such as the

Gaussian kernel. To address this issue, some form of regularization must be added to the CCA definition.

Kernel CCA

Both issues have been addressed in the KCCA algorithm which we now present. Further details and references can be found in Akaho (2001) and Bach and Jordan (2002). The goal is to detect correlations between two data sets $\mathbf{x}_1 = \left(\mathbf{x}_1^{(1)}, \cdots, \mathbf{x}_1^{(n)}\right)$ and $\mathbf{x}_2 = \left(\mathbf{x}_2^{(1)}, \cdots, \mathbf{x}_2^{(n)}\right)$, where n is the number of objects, and each data set $\mathbf{x}_1^{(i)}$ / $\mathbf{x}_2^{(i)}$ belongs to some set $\mathcal{X}_1$ / $\mathcal{X}_2$, for $i = 1, \cdots, n$. In the example treated in this chapter, the objects correspond to genes, and each data set corresponds to one representation of the genes. For example, if $\mathcal{X}_1$ is the set of finite-length nucleotide sequences, and $\mathcal{X}_2$ is a vector space of gene expression profiles, then $\mathbf{x}_1^{(i)}$ could be the sequence of the i-th gene studied and $\mathbf{x}_2^{(i)}$ its expression profile.

In order to detect correlations between the two data sets, the objects $\mathbf{x}_1^{(i)}$ $/\mathbf{x}_2^{(i)}$ are mapped to a Hilbert space $H_1/$ H_2 by a mapping $\phi_1 : \mathcal{X}_1 \to H_1/$ $\phi_2 : \mathcal{X}_2 \to H_2$. Classic CCA can then be performed between the images $\phi_1(\mathbf{x}_1)$ and $\phi_2(\mathbf{x}_2)$ as follows. For any two directions $f_1 \in H_1$ and $f_2 \in H_2$, we can define the projections $u_1 = \left(u_1^{(1)}, \cdots, u_1^{(n)}\right)^\top \in \mathbb{R}^n$ and $u_2 = \left(u_2^{(1)}, \cdots, u_2^{(n)}\right)^\top \in \mathbb{R}^n$ of $\mathbf{x}_1$ and $\mathbf{x}_2$ onto f_1 and f_2 by

$$u_1^{(i)} := \langle f_1, \phi_1(\mathbf{x}_1^{(i)}) \rangle, \quad u_2^{(i)} := \langle f_2, \phi_2(\mathbf{x}_2^{(i)}) \rangle, \tag{1.2}$$

for $i = 1, \cdots, n$, where $\langle ., . \rangle$ denotes the dot products in the Hilbert spaces H_1 and H_2. The sample mean, variance, and covariance of u_1 and u_2 are respectively defined by

$$\begin{aligned} \hat{mean}\,(u_j) &:= \frac{1}{n} \sum_{i=1}^{n} u_j^{(i)}, \\ \hat{var}(u_j) &:= \frac{1}{n} \sum_{i=1}^{n} \left(u_j^{(i)} - \hat{mean}\,(u_j)\right)^2, \\ \hat{cov}(u_1, u_2) &:= \frac{1}{n} \sum_{i=1}^{n} \left(u_1^{(i)} - \hat{mean}\,(u_1)\right) \left(u_2^{(i)} - \hat{mean}\,(u_2)\right), \end{aligned} \tag{1.3}$$

for $j = 1, 2$. The goal of CCA is to find $f_1 \in H_1$ and $f_2 \in H_2$ that maximize the empirical correlation between u_1 and u_2, defined by:

$$\hat{corr}\,(u_1, u_2) := \frac{\hat{cov}(u_1, u_2)}{(\hat{var}(u_1)\hat{var}(u_2))^{\frac{1}{2}}}. \tag{1.4}$$

The solution to this problem, however, is not unique when the dimension of H_1 or H_2 is larger than the number of samples n: indeed, adding to f_1 or f_2 any vector orthogonal to the linear span of the respective points does not change the projections u_1 and u_2. Moreover, the importance of regularization for CCA in high dimension is a well-known fact discussed, for instance, by Hastie et al. (1995) and Leurgans et al. (1993).

Regularization of CCA

A classic way to regularize CCA is to penalize the Hilbert norm of f_1 and f_2 through the maximization of the following functional instead of (1.4):

$$\gamma(f_1, f_2) := \frac{\hat{cov}(u_1, u_2)}{\left(\hat{var}(u_1) + \lambda_1 ||f_1||^2\right)^{\frac{1}{2}} \left(\hat{var}(u_2) + \lambda_2 ||f_2||^2\right)^{\frac{1}{2}}}, \tag{1.5}$$

where λ_1 and λ_2 are regularization parameters. When $\lambda_1 = \lambda_2 = 0$, $\gamma(f_1, f_2)$ reduces to the sample correlation (1.4), but when $\lambda_1 > 0$ and $\lambda_2 > 0$, the pair (f_1, f_2) that maximizes (1.5) finds a tradeoff between maximizing the empirical correlation (1.4) and having small norms $||f_j||/\hat{var}(u_j)$ (for $j = 1, 2$).

By homogeneity, maximizing (1.5) is equivalent to maximizing $\hat{cov}(u_1, u_2)$ under the constraints

$$\hat{var}(u_1) + \lambda_1 ||f_1||^2 \leq 1, \quad \hat{var}(u_2) + \lambda_2 ||f_2||^2 \leq 1.$$

Dual formulation

The solution to this problem is obtained by solving the Lagrangian:

$$\begin{aligned} L(f_1, f_2, \rho_1, \rho_2) &= \hat{cov}(u_1, u_2) \\ &+ \frac{\rho_1}{2}\left(1 - \hat{var}(u_1) - \lambda_1 ||f_1||^2\right) + \frac{\rho_2}{2}\left(1 - \hat{var}(u_2) - \lambda_2 ||f_2||^2\right), \end{aligned} \tag{1.6}$$

where ρ_1 and ρ_2 are Lagrange multipliers. From the conditions that the derivatives of L with respect to f_1 and f_2 be equal to 0, we get that f_1 and f_2 must respectively be in the linear span of $\mathbf{x}_1$ and $\mathbf{x}_2$, that is:

$$f_1 = \sum_{j=1}^{n} \alpha_1^{(j)} \phi_1(\mathbf{x}_1^{(j)}), \quad f_2 = \sum_{j=1}^{n} \alpha_2^{(j)} \phi_2(\mathbf{x}_2^{(j)}), \tag{1.7}$$

for some $\alpha_1 \in \mathbb{R}^n$ and $\alpha_2 \in \mathbb{R}^n$. Supposing now that the points are centered in the feature space, that is, $\sum_{i=1}^{n} \phi_1(\mathbf{x}_1^{(i)}) = \sum_{i=1}^{n} \phi_2(\mathbf{x}_2^{(i)}) = 0$, the sample means $\hat{mean}(u_1)$ and $\hat{mean}(u_2)$ are always null, by (1.2) and (1.3). Plugging (1.7) into (1.3), we can then rewrite the sample variance and covariance of u_1 and u_2 in terms of α_1 and α_2:

$$\begin{aligned} \hat{var}(u_1) &= \frac{1}{n} \alpha_1^\top K_1^2 \alpha_1, \\ \hat{var}(u_2) &= \frac{1}{n} \alpha_2^\top K_2^2 \alpha_2, \\ \hat{cov}(u_1, u_2) &= \frac{1}{n} \alpha_1^\top K_1 K_2 \alpha_2, \end{aligned} \tag{1.8}$$

where K_1 and K_2 are the $n \times n$ kernel Gram matrix defined by $K_1(i,j) = k_1(\mathbf{x}_1^{(i)}, \mathbf{x}_1^{(j)}) = \langle \phi_1(\mathbf{x}_1^{(i)}), \phi_1(\mathbf{x}_1^{(j)}) \rangle$ and $K_2(i,j) = k_2(\mathbf{x}_2^{(i)}, \mathbf{x}_2^{(j)}) = \langle \phi_2(\mathbf{x}_2^{(i)}), \phi_2(\mathbf{x}_2^{(j)}) \rangle$ for $1 \leq i, j \leq n$. Observing from (1.7) that the square Hilbert norms of f_1 and f_2 can also be expressed in terms of α_1 and α_2 as follows:

$$||f_1||^2 = \alpha_1^\top K_1 \alpha_1, \quad ||f_2||^2 = \alpha_2^\top K_2 \alpha_2,$$

we finally can rewrite the Lagrangian (1.6) as a function of α_1 and α_2 as follows:

$$\begin{aligned} L(\alpha_1, \alpha_2, \rho_1, \rho_2) &= \frac{1}{n}\alpha_1^\top K_1 K_2 \alpha_2 \\ &+ \frac{\rho_1}{2n}\left(n - \alpha_1^\top K_1^2 \alpha_1 - n\lambda_1 \alpha_1^\top K_1 \alpha_1\right) + \frac{\rho_2}{2n}\left(n - \alpha_2^\top K_2^2 \alpha_2 - n\lambda_2 \alpha_2^\top K_2 \alpha_2\right). \end{aligned}$$

Observing that $K^2 + n\lambda K = \left(K + \frac{n\lambda}{2}\mathbf{I}\right)^2$ up to the second order in λ for any square matrix K ($\mathbf{I}$ represents the identity matrix), and imposing that the derivatives with respect to α_1 and α_2 of the first-order approximation of the Lagrangian be equal to 0, we obtain that the values (α_1, α_2) and (ρ_1, ρ_2) that solve the Lagrangian are solution of the following generalized eigenvalue problem;

$$\begin{pmatrix} \mathbf{0} & K_1 K_2 \\ K_2 K_1 & \mathbf{0} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \rho \begin{pmatrix} (K_1 + \frac{n\lambda_1}{2}\mathbf{I})^2 & \mathbf{0} \\ \mathbf{0} & (K_2 + \frac{n\lambda_2}{2}\mathbf{I})^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}. \qquad (1.9)$$

It can be shown (Bach and Jordan, 2002) that the canonical correlations are indeed the $\min(N_1, N_2)$ largest generalized eigenvalues of this problem.

Let $\left(\alpha_1^\top, \alpha_2^\top\right)^\top$ be a generalized eigenvector with generalized eigenvalue ρ. From (1.2) and (1.7) we can recover the canonical variate u_1 and u_2 associated with the canonical correlation ρ as follows:

$$u_1 = K_1 \alpha_1, \qquad u_2 = K_2 \alpha_2.$$

The above derivation is only valid if the points are centered in the feature space. This is not a restriction, however, because for any Gram matrix K of noncentered data points, the Gram matrix $\tilde{K}$ of the centered data points can be computed by $\tilde{K} = N_0 K N_0$ where $N_0 = \mathbf{I} - \frac{1}{n}\mathbf{1}$, where $\mathbf{1}$ is the $n \times n$ matrix composed of ones (Schölkopf et al., 1999).

1.2.3 Multiple KCCA

In this subsection we present an extension of KCCA when more than two kernel matrices are available. This method was suggested by Bach and Jordan (2002) for the purpose of independent component analysis. We refer to it as multiple kernel canonical correlation analysis (MKCCA). It is a natural extension of the ordinary KCCA model described in the previous subsection.

Multiple KCCA

Suppose that we have P data sets $\left(\mathbf{x}_p^{(1)}, \cdots, \mathbf{x}_p^{(n)}\right)$ for $p = 1, 2, \cdots, P$, where $\mathbf{x}_p^{(i)}$ belongs to a set $\mathcal{X}_p$ for $1 \leq p \leq P$ and $1 \leq i \leq n$. For each $p = 1, \cdots, P$, suppose that there is a mapping $\phi_p : \mathcal{X}_p \rightarrow H_p$ to a Hilbert space H_p, and let K_p be the corresponding Gram matrix, that is:

$$K_p(i,j) = k_p(\mathbf{x}_p^{(i)}, \mathbf{x}_p^{(j)}) = \langle \phi_p(\mathbf{x}_p^{(i)}), \phi_p(\mathbf{x}_p^{(j)}) \rangle,$$

for $1 \leq i, j \leq n$. Each set of points is supposed to be centered in the feature space.

The goal of MKCCA is to detect directions $f_p \in H_p \, (p = 1, 2, \cdots, P)$ such that the sum of all pairwise correlations between features

$$u_p^{(i)} = \langle f_p, \phi_p(\mathbf{x}_p^{(i)}) \rangle, \quad p = 1, \cdots, P, \quad i = 1, \cdots, n, \tag{1.10}$$

be the largest possible, that is, to solve the following problem:

$$\max_{(f_1, \cdots, f_P) \in H_1 \times \cdots \times H_P} \sum_{1 \leq p < q \leq P} c\hat{o}rr(u_p, u_q). \tag{1.11}$$

Following the same approach as the one explained in subsection 1.2.2, the problem (1.11) is regularized with regularization parameters $\lambda_p \geq 0$ for $1 \leq p \leq P$ as follows:

$$\max_{(f_1, \cdots, f_P) \in H_1 \times \cdots \times H_P} \sum_{1 \leq p < q \leq P} \frac{c\hat{o}v(u_p, u_q)}{\left(v\hat{a}r(u_p) + \lambda_p ||f_p||^2\right)^{\frac{1}{2}} \left(v\hat{a}r(u_q) + \lambda_q ||f_q||^2\right)^{\frac{1}{2}}}. \tag{1.12}$$

It is then easy to derive that the vectors $(f_1, \cdots, f_P)$ solving (1.12) can be expressed as

$$f_p = \sum_{j=1}^{n} \alpha_p^{(j)} \phi_p(\mathbf{x}_p^{(j)}), \tag{1.13}$$

for some vector $\alpha_p \in \mathbb{R}^n$, for $1 \leq p \leq P$, and that the α_p solve the Lagrangian:

$$L = \frac{1}{n} \sum_{1 \leq p < q \leq P} \alpha_p^T K_p K_q \alpha_q + \sum_{p=1}^{P} \frac{\rho_p}{2n} \left(n - \alpha_p^\top K_p^2 \alpha_p - n \lambda_p \alpha_p^\top K_p \alpha_p \right). \tag{1.14}$$

The estimation of canonical correlation scores (CC scores) is now reduced to the following generalized eigenvalue problem:

$$\begin{pmatrix} \mathbf{0} & \cdots & K_1 K_P \\ \vdots & \ddots & \vdots \\ K_P K_1 & \cdots & \mathbf{0} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_P \end{pmatrix}$$
$$= \rho \begin{pmatrix} (K_1 + \frac{n\lambda_1}{2} I)^2 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & (K_P + \frac{n\lambda_P}{2} I)^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_P \end{pmatrix}.$$

The correlated variates can then be obtained by $u_p = K_p \alpha_p$ (for $p = 1, 2, \cdots, P$).

1.2.4 Integrated KCCA

While theoretically sound, the MKCCA approach presented in subsection 1.2.3 may suffer in practice from the fact that by maximizing (1.12), it detects correlations among *all pairs of data sets*. As the number of data sets increases, it is often the case that no strong signal is present simultaneously in all data sets, except in trivial cases.

In this subsection we propose a variant to perform KCCA on more than two data sets to address this issue. We suppose that the data sets available $(\mathbf{x}_k)_{k=1,\cdots,P}$ are split into two groups, $(\mathbf{x}_p)_{p\in\mathcal{P}}$ and $(\mathbf{x}_q)_{q\in\mathcal{Q}}$, where $\mathcal{P}$ and $\mathcal{Q}$ form a partition of $\{1,\cdots,P\}$. Intuitively, this split should be done in such a way that there is not necessarily a big correlation between the data sets in each split, but that the data sets of one split taken together contain a clear correlation with the data sets of the other split taken together.

More formally, suppose first that the variables are real-valued vectors, that is, $\mathbf{x}_k \in \mathbb{R}^{N_k}$ for $k = 1,\cdots,P$. Then we propose to concatenate the vector representations in each split to obtain two vector representations $\mathbf{x}_\mathcal{P}$ and $\mathbf{x}_\mathcal{Q}$ of the data of dimensions $\sum_{p\in\mathcal{P}} N_p$ and $\sum_{q\in\mathcal{Q}} N_q$ respectively, and to search for canonical correlations between the resulting two vectors. This amounts to solving the generalized eigenvalue problem (1.9) with K_1 and K_2 replaced by $K_\mathcal{P}$ and $K_\mathcal{Q}$, the kernel matrices of the concatenated vectors $\mathbf{x}_\mathcal{P}$ and $\mathbf{x}_\mathcal{P}$. Now, because $k_\mathcal{P}(\mathbf{x}_\mathcal{P}^{(i)}, \mathbf{x}_\mathcal{P}^{(j)}) = \sum_{p\in\mathcal{P}} k_p(\mathbf{x}_p^{(i)}, \mathbf{x}_p^{(j)})$ and $k_\mathcal{Q}(\mathbf{x}_\mathcal{Q}^{(i)}, \mathbf{x}_\mathcal{Q}^{(j)}) = \sum_{q\in\mathcal{Q}} k_q(\mathbf{x}_q^{(i)}, \mathbf{x}_q^{(j)})$ for $1 \le i,j \le n$, it follows that

$$K_\mathcal{P} = \sum_{p\in\mathcal{P}} K_p, \quad K_\mathcal{Q} = \sum_{q\in\mathcal{Q}} K_q. \tag{1.15}$$

In the more general case where the data sets are not real vector-valued, but rather belong to more general sets endowed with kernel functions, then the same analysis holds for the vector representations in the features spaces associated with the kernels. In particular (1.15) holds for general kernel functions. Observe that summing up kernels is a convenient way to integrate heterogeneous information, which was, for instance, investigated by Pavlidis et al. (2001) in the context of gene function prediction from gene expression and phylogenetic profiles.

Integrated kernel CCA

Plugging (1.15) into (1.9), we see that integrated KCCA (IKCCA)can be performed by solving the following generalized eigenvalue problem:

$$\begin{pmatrix} \mathbf{0} & \sum_{p\in\mathcal{P}} K_p . \sum_{q\in\mathcal{Q}} K_q \\ \sum_{q\in\mathcal{Q}} K_q . \sum_{p\in\mathcal{P}} K_p & \mathbf{0} \end{pmatrix} \begin{pmatrix} \alpha_\mathcal{P} \\ \alpha_\mathcal{Q} \end{pmatrix}$$
$$= \rho \begin{pmatrix} (\sum_{p\in\mathcal{P}} K_p + \frac{n\lambda_\mathcal{P}}{2}\mathbf{I})^2 & \mathbf{0} \\ \mathbf{0} & (\sum_{q\in\mathcal{Q}} K_q + \frac{n\lambda_\mathcal{Q}}{2}\mathbf{I})^2 \end{pmatrix} \begin{pmatrix} \alpha_\mathcal{P} \\ \alpha_\mathcal{Q} \end{pmatrix}.$$

As for KCCA and MKCCA, the correlated variates can again be obtained by $u_\mathcal{P} = K_\mathcal{P}\alpha_\mathcal{P}$ and $u_\mathcal{Q} = K_\mathcal{Q}\alpha_\mathcal{Q}$.

1.2.5 From CCA to Object Selection

Each generalization of CCA presented so far produces several canonical variates for each canonical correlation (2 for KCCA and IKCCA, P for MKCCA). Let us define the *canonical score* $s \in \mathbb{R}^n$ associated with a given canonical correlation to be the absolute value of the average of the corresponding canonical variates. That is, using

Canonical scores

the notations of the previous subsection, we respectively define the canonical scores for KCCA, MKCCA, and IKCCA by

$$s_{KCCA}(i) = \left| \frac{u_1^{(i)} + u_2^{(i)}}{2} \right|, \; s_{MKCCA}(i) = \left| \frac{1}{P} \sum_{j=1}^{P} u_j^{(i)} \right|, \; s_{IKCCA}(i) = \left| \frac{u_{\mathcal{P}}^{(i)} + u_{\mathcal{Q}}^{(i)}}{2} \right|,$$

for $i = 1, \cdots, n$.

The canonical score can be thought of as a quantitative measure of how objects contribute to the canonical correlation. To see this, let us observe, for example, that when u_1 and u_2 are scaled to unit variance

$$\sum_{i=1}^{N} s_{KCCA}(i)^2 = \sum_{i=1}^{N} \left| \frac{u_1^{(i)} + u_2^{(i)}}{2} \right|^2 = \frac{N}{2} \left[1 + c\hat{o}rr(u_1, u_2) \right].$$

Similar results hold for s_{MKCCA} and s_{IKCCA}. This shows that the correlation $c\hat{o}rr(u_1, u_2)$ between canonical variates is the sum of individual canonical scores.

In the case where the canonical correlation is due to some hidden phenomenon, this suggests that objects with large canonical scores are more likely to be involved in the phenomenon than others. If one is interested in the detection of such objects, it therefore makes sense to select those objects that have a canonical score above a given threshold.

Link with spectral clustering

It is worth observing that this method of selecting objects bears some similarity to recently studied spectral clustering methods (Weiss, 1999; Ng et al., 2002) which perform data clustering after embedding the data in a feature space using the first eigenvectors of the kernel Gram matrix. In our case, we use the canonical directions instead of the principal directions to map the data, and need to average over the canonical directions found in different feature spaces in order to obtain a one-dimensional mapping. Selecting the objects with large scores then corresponds to a simple clustering method that separates numbers with large absolute values from the others. Of course, this does not make sense if the correlation is due to the presence of two or more different classes of points that one wants to separate as different clusters, in which case large positive canonical variates should be separated from large negative variates, as most clustering methods would do. However, it makes sense in the cases where the canonical correlation is due to the presence of a number of small "interesting" clusters separated from a large bulk of "noninteresting" objects, and where the goal is to detect the "interesting" objects. We illustrate such a case below, in the problem of detecting operons in a genome.

The link with spectral clustering methods is particularly clear when a single kernel matrix K is considered. Similarly to KCCA, kernel principal component analysis (KPCA) searches a direction f of the Hilbert space H that defines a variate $u^{(i)} = \langle f, \phi\left(\mathbf{x}^{(i)}\right) \rangle$ with maximum variance (where $||f||$ is fixed). As explained in chapter **??**, subsection **??**, this is equivalent to the following problem:

$$\min_{v\hat{a}r(u)=1} ||f||. \tag{1.16}$$

Suppose now that we perform KCCA between the kernel K and itself. From (1.5) it is obvious that the two variates found are equal ($f_1 = f_2$), and that the functional to maximize becomes

$$\gamma_{PCA}(f) = \frac{\hat{var}(u)}{\hat{var}(u) + \lambda||f||^2},$$

where we suppose that $\lambda_1 = \lambda_2 = \lambda$ in (1.5). By homogeneity the maximization of γ_{PCA} is equivalent to the following problem:

$$\max_{\hat{var}(u)=1} \frac{1}{1 + ||f||^2},$$

which is equivalent to (1.16). This shows that KCCA (and any of its generalization to more than two kernels) boils down to KPCA when the kernels are equal.

1.2.6 Diffusion Kernel

In the experiments we perform below, some of the data sets consist of graphs whose nodes are the objects of interest. As an example, metabolic pathways or the organization of the genes on a genome can be represented by graphs with genes as nodes. In order to use such data sets in the KCCA framework, the information contained in the graph must be encoded into a kernel function. We perform this transformation of a graph into a kernel using the diffusion kernel, proposed by Kondor and Lafferty (2002) and reviewed in chapter **??**, which we now briefly recall.

Suppose that we have an undirected, unweighted graph $\Gamma = (V, E)$. The opposite Laplacian of this graph is the matrix

$$\mathbf{H}_{ij} = \begin{cases} 1 & \text{for } i \sim j, \\ -d_i & \text{for } i = j, \\ 0 & \text{otherwise,} \end{cases} \tag{1.17}$$

where $i \sim j$ means that the ith and jth genes are joined by an edge on the graph, and d_i is the number of edges emanating from the ith vertex. The exponential of the matrix $\beta\mathbf{H}$ is defined as

$$\exp(\beta\mathbf{H}) = \lim_{m\to\infty} \left(\mathbf{I} + \frac{\beta\mathbf{H}}{m}\right)^m, \tag{1.18}$$

where β is a positive constant. This is equivalent to the following expansion:

$$\exp(\beta\mathbf{H}) = \mathbf{I} + \beta\mathbf{H} + \frac{\beta^2}{2}\mathbf{H}^2 + \frac{\beta^3}{3!}\mathbf{H}^3 + \cdots. \tag{1.19}$$

The resulting matrix is symmetric and positive definite. It is therefore a valid kernel called the diffusion kernel (Kondor and Lafferty, 2002), which can be thought of as a generalization of the Gaussian radial basis function (RBF) kernel to a discrete setting.

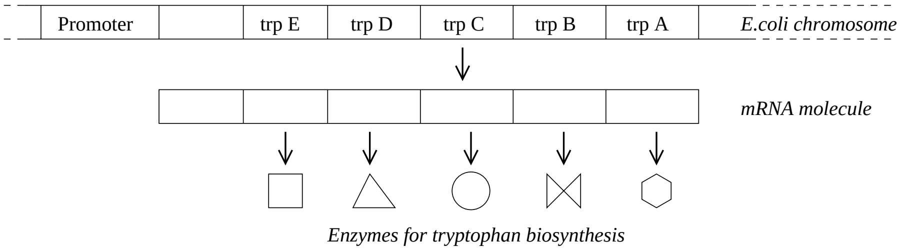

Figure 1.1 The clustered genes in *E.coli* that code for enzymes responsible for the synthesis of the amino acid tryptophan. The five genes are transcribed as a single mRNA molecule, a feature that allows their expression to be controlled coordinately. Such a cluster of genes is called an operon.

1.3 Experimental Results

In order to test the various generalizations of KCCA (section 1.2.3 and section 1.2.4) and the object selection method (section 1.2.5) presented so far on real-world data, we performed a series of experiments with the goal of detecting operons in the *E.coli* genome.

1.3.1 Operon Detection

Operons

In most bacterial genomes, functionally coupled gene clusters are often adjacent to one another on the genome and regulated under the same upstream promoter, therefore transcribed as one long polycistronic mRNA. Such clusters of genes are called *operons*. As an example, figure 1.1 shows the well-studied tryptophan operon which contains five genes translated into five enzymes responsible for the synthesis of tryptophan. Experimental detection or confirmation of operons is time-consuming (Walters et al., 2001) and relatively difficult to implement in the laboratory as a high-throughput process. Computational prediction of operons has therefore gained increased attention in recent years, either by sequence analysis only (Yada et al., 2001; Salgado et al., 2000; Ermolaeva et al., 2001) or by combining multiple information (Ogata et al., 2000; Zheng et al., 2002).

1.3.2 Data

Because genes that code for enzymes in operons are closely located on the genome, are coregulated, and often catalyze related reactions in metabolic pathways, they should be responsible for a form of correlation between three sorts of data: the position of genes on the genome, their expression as measured by DNA microarray, and the position of the chemical reactions they catalyze in metabolic pathways. We therefore tried to automatically detect correlations between these three sorts of data using CCA, and to detect genes likely to belong to operons by selecting the genes mostly responsible for the detected correlations.

We therefore collected three sorts of data for the genes of the bacterium *E.coli* and derived three kernel matrices for the 740 genes common to all three data sets as follows.

Pathways

Pathway data were extracted from the KEGG/LIGAND database of chemical compounds and reactions in biological pathways (Goto et al., 2002), which can be freely downloaded from the KEGG database (Kanehisa et al., 2002). This database contains thousands of metabolic reactions known to take place in various organisms, together with the substrates involved and the classification of the catalyzing enzyme as an EC number. From this database we created an undirected graph with genes of *E.coli* as vertices, where two vertices are linked when the genes encode enzymes that can catalyze two successive reactions in a pathway. The resulting graph, called the *gene metabolic network*, is described in more detail in chapter **??**, subsection **??**. From this graph of genes we built a diffusion kernel as explained in subsection 1.2.6 with the parameter β set to 1.

Gene positions

The positions of the genes on genomes were obtained from the KEGG/GENES database, which contains various genomic information such as gene names, positions along chromosomes, and their amino acid sequences. From this we computed a matrix of pairwise gene distance,where the distance d_{ij} between gene i and gene j is defined by the number of nucleotides between the end of the ith gene and the start of the jth gene along the chromosomes. We then derived a distance kernel by the formula $k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \exp\left(-d_{ij}/h\right)$, where h is a parameter set to 10^5.

Gene expression

Finally the gene expression data for 48 experiments[1] on the genes *E.coli K-12* were downloaded from the KEGG/EXPRESSION database, a repository of expression data for *Saccharomyces cerevisiae*, *E.coli*, and *Bacilla subtilis*. Given the (R,G) fluorescence intensity pairs for each gene on each array (where R=red for Cy5 and G=green for Cy3), we evaluated the expression level by the log ratio $\log(R_S - R_B)/(G_S - G_B)$, where G_B is control-background, G_S is control-signal, R_B is target-background, and R_S is target-signal, respectively. We then used a Gaussian RBF kernel with unit width to obtain the expression Gram matrix.

This results in three 740 × 740 kernel Gram matrices, which we denote by $K_{pathway}$, K_{genome} and $K_{expression}$ below.

1.3.3 Experiments

We performed successively ordinary KCCA (OKCCA) between the three possible pairs of kernels, MKCCA between the three kernels, and IKCCA between all splits of the three kernels into two groups. To confirm the improvement due to the comparison and integration of several attributes, we also performed ordinary kernel PCA on each single data set (e.g., pathway alone, genome alone, and expression alone). Table 1.1 summarizes these experiments.

[1] With identification numbers ex0000287 to ex0000334 in the KEGG/EXPRESSION database.

Table 1.1 List of experiments performed to detect operons in the *E.coli* genome. For OKCCA and IKCCA methods, an ordinary KCCA is performed between the two kernels in the columns Kernel 1 and Kernel 2. For the MKCCA method, a multiple KCCA is performed between the three kernels. For the KPCA method, an ordinary KPCA is performed on Kernel 1. In each case, operons are then predicted by the gene selection method described in subsection 1.2.5

Name	Abbr.	Method	Kernel 1	Kernel 2	Kernel 3
OKCCA-a	O-a	KCCA	$K_{pathway}$	K_{genome}	-
OKCCA-b	O-a	KCCA	K_{genome}	$K_{expression}$	-
OKCCA-c	O-a	KCCA	$K_{expression}$	$K_{pathway}$	-
MKCCA	M	MKCCA	$K_{pathway}$	K_{genome}	$K_{expression}$
IKCCA-a	I-a	IKCCA	$K_{genome} + K_{expression}$	$K_{pathway}$	-
IKCCA-b	I-a	IKCCA	$K_{expression} + K_{pathway}$	K_{genome}	-
IKCCA-c	I-a	IKCCA	$K_{pathway} + K_{genome}$	$K_{expression}$	-
KPCA-a	S-a	KPCA	$K_{pathway}$	-	-
KPCA-b	S-a	KPCA	K_{genome}	-	-
KPCA-c	S-a	KPCA	$K_{expression}$	-	-

We then applied the gene selection procedure described in subsection 1.2.5 with a varying threshold, and compared the set of genes selected at a given threshold with a database of known operons (Ito et al., 1999). By varying the threshold, we computed the number of selected genes that really belong to a known operon (true positives) as a function of the number of selected genes that do not belong to a known operon (false positive). We therefore obtained a receiver operating characteristic (ROC) curve (Gribskov and Robinson, 1996), that is a plot of true positive as a function of false positives, for each CCA method.

1.3.4 Results

Figure 1.2 shows multiple cross-scatterplots of the first canonical variates obtained with the MKCCA method between pathway, genome, and expression. Figures 1.3 shows scattersplots of the first canonical variates obtained with the IKCCA-a,-b, and -c methods. In these scatterplots each point corresponds to one gene. The diagonal shapes of the clouds of points indicate that correlations have been detected in all cases. The correlations detected are mostly due to the genes with high or low scores, in particular in MKCCA, IKCCA-a, and IKCCA-b.

Operon detection

Operons are likely to form clusters simultaneously in all feature spaces defined by the three kernels considered. As a result, they might be the cause behind the first canonical correlation, in which case genes that form operons are more likely to contribute strongly to the canonical correlation than are others. This motivates the use of the object selection methods described in subsection 1.2.5 with the goal of detecting genes that belong to operons. It should be noted here that we don't try to separate different operons, but rather to separate operon genes from the rest.

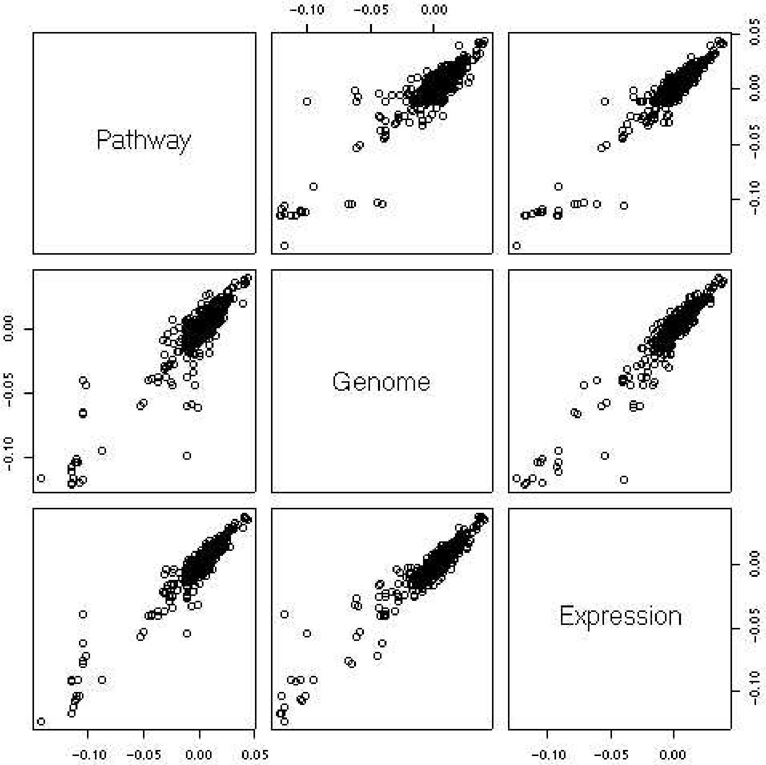

Figure 1.2 Multiple cross-scatterplots of the first canonical variates in MKCCA. In each plot, a circle corresponds to a gene. MKCCA extracts one canonical variate for each of the three data sets: pathway, genome, and expression. These plots highlight the pairwise correlations between these canonical variates.

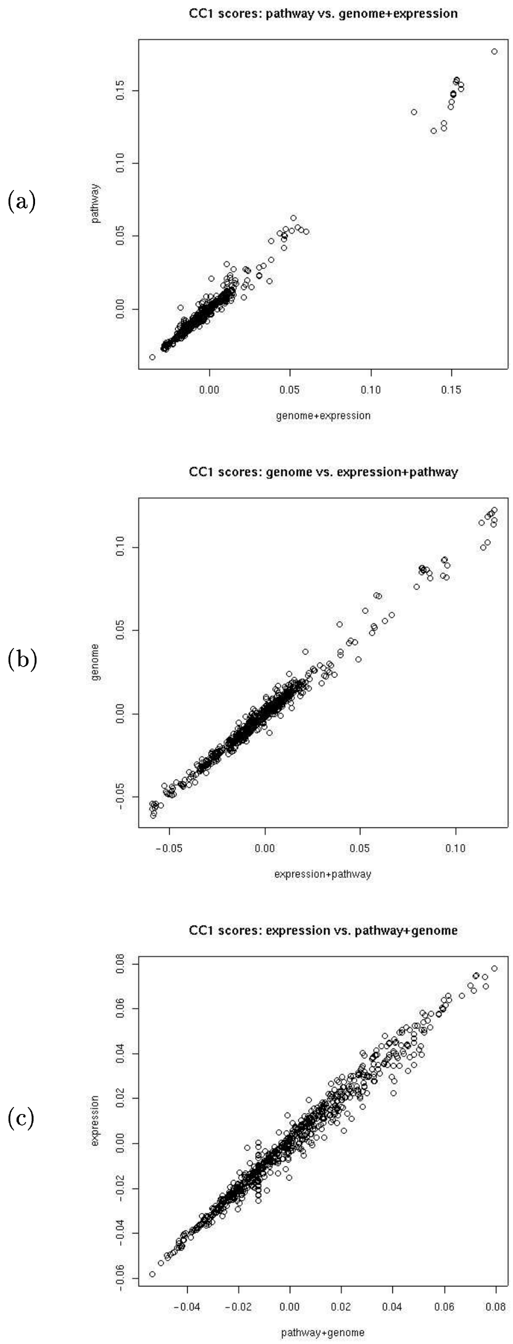

Figure 1.3 Scatterplots of the first canonical variates in IKCCA-a (pathway vs. genome + expression), IKCCA-b (genome vs. expression + pathway), and IKCCA-c (expression vs. pathway + genome). In each case, the plot highlights the canonical correlation between the two variates extracted by IKCCA.

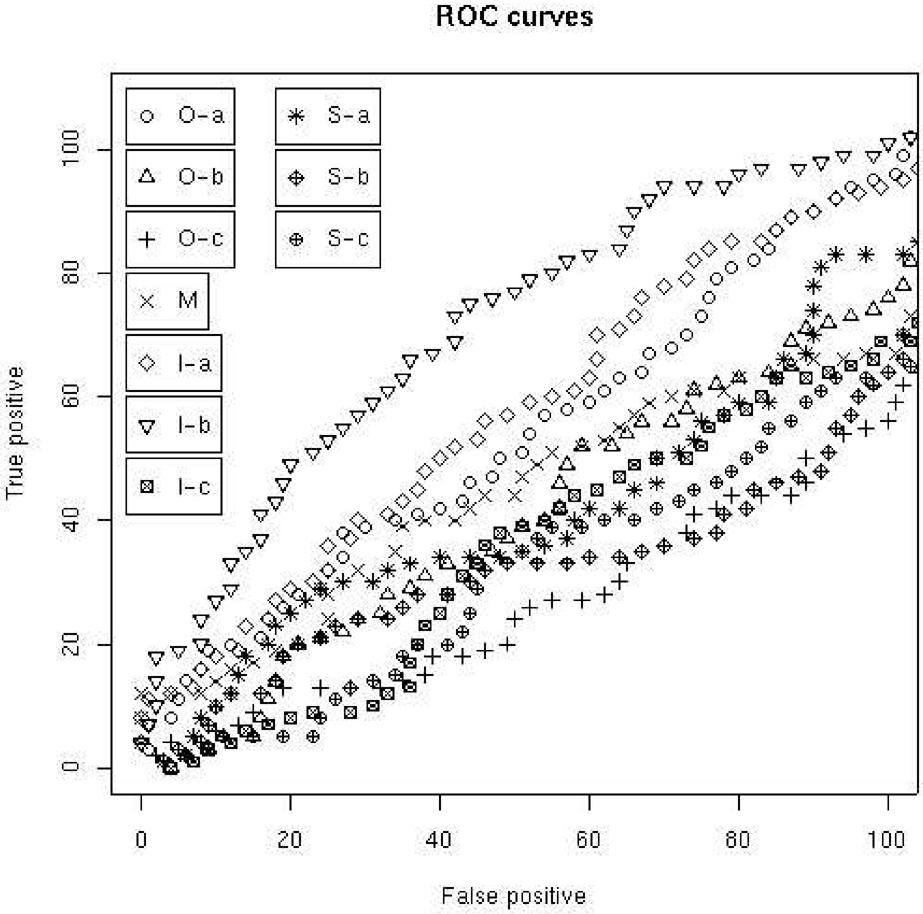

Figure 1.4 ROC curves for the detection of operon genes. O-a, -b, -c indicate OKCCA-a, -b, -c, respectively; M indicates MKCCA; I-a, -b, -c indicate IKCCA-a, -b, -c respectively; S-a, -b, -c indicate KPCA-a, -b, -c respectively. For each method, the gene selection method described in subsection 1.2.5 was performed with a varying threshold in order to vary the number of genes selected. These curves show the number of selected genes that belong to known operons (true positives on the y-axis) as a function of the number of genes selected even though they don't belong to known operons (false positives on the x-axis).

Table 1.2 Number of correctly detected genes based on the first canonical scores in each KCCA. We set the threshold such that 10% of all genes with high scores (74 out of 740 genes) are selected.

Operon (# of genes)	O-a	O-b	O-c	M	I-a	I-b	I-c
Biotin metabolism (3)	3	1	0	3	3	3	0
Fatty acid (short-chain) metabolism (3)	0	3	0	2	0	3	3
Fumarate reductase (4)	4	0	2	4	4	4	0
Galactose metabolism (4)	4	0	0	4	3	4	1
Glycerol-3-phosphate dehydrogenase (3)	0	3	3	3	3	3	3
Menaquinone (vitamin K_2) biosynthesis (5)	0	3	0	0	4	0	0
NADH dehydrogenase (13)	0	0	0	0	0	13	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Total number (280)	39	34	27	37	42	52	28

Figure 1.4 shows the ROC curves for the task of detecting operon genes with the object selection method described in subsection 1.2.5 applied to each CCA method in table 1.1. Compared with the performance of the approach applied to a single data set, the detection rates have been improved by the comparison and integration of several data sets. Table 1.2 shows the number of genes correctly selected by each method for several known operons when we set the threshold such that 10% of all genes (74 out of 740 genes) are selected, for instance.

The best operon detection performance is obtained by IKCCA-b, which corresponds to correlations between K_{genome} and $K_{pathway} + K_{expression}$. Next are IKCCA-a, corresponding to correlations between $K_{pathway}$ and $K_{genome} + K_{expression}$, and OKCCA-a, corresponding to correlations between K_{genome} and $K_{pathway}$. The worst methods are OKCCA-c, IKCCA-c, and OKCCA-b, which correspond to correlations between $K_{expression}$ and another kernel involving $K_{pathway}$ or K_{genome}, or both.

Kernel hierarchy

These results suggest several remarks. First, a clear hierarchy appears between the three kernels. K_{genome} is the one that contains the most information about operons, as seen from the good performance of the methods that detect correlations between K_{genome} alone and other kernels. It is closely followed by $K_{pathway}$. $K_{expression}$ is clearly less related to operons, as shown by the poor performance of the experiments where canonical correlations were driven by $K_{expression}$. The major contribution of K_{genome} in the detection of operons makes sense, by the definition itself of operons, which are clusters of genes on the genome. The relatively poor performance of OKCCA-b (K_{genome} vs. $K_{expression}$), and more generally of all experiments involving $K_{expression}$ alone, seems to indicate that the quality of the expression data used is poor, since genes in an operon are supposed to be coregulated. In contrast, the good performance of $K_{pathway}$ suggests that the pathway database is of reasonable quality.

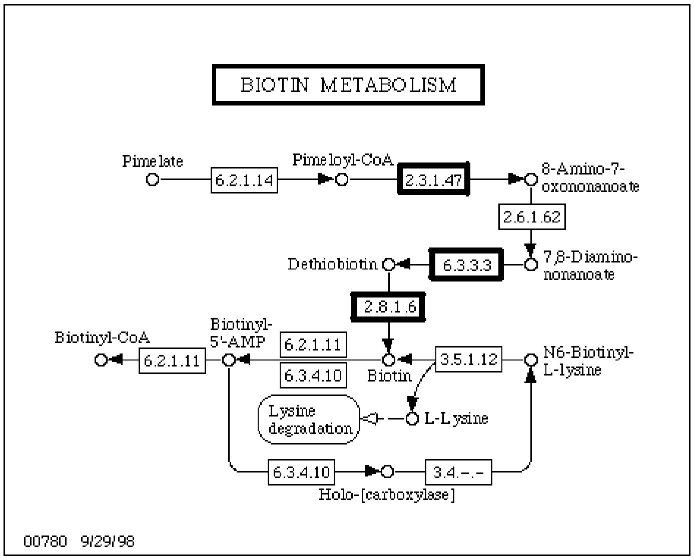

Figure 1.5 An example of known operons in the operon data library. The genes in known operons are represented by the corresponding EC numbers, and are outlined in bold boxes.

Advantage of kernel combination

Second, in spite of the poor quality of the expression data, it appears that the best performance is obtained by using the three kernels in the form of canonical correlations between K_{genome} and $K_{pathway}+K_{expression}$. This means that IKCCA-b is able to somehow denoise the expression data and extract from a combination of pathway information and expression data a meaningful correlation with the genome data that outperforms the correlation detected by each data set alone with the genome data. This experiment is a typical example where IKCCA is more relevant than MKCCA and OKCCA, because of the difference in the information about operons contained in each data set.

Visualization

For visualization the genes detected can be mapped to the KEGG/pathway visualization tool. As an example, figure 1.5 shows a very local picture of the large metabolic network, namely biotin metabolism, together with 3 genes known to form an operon (the genes respectively marked 2.3.1.47, 6.3.3.3, and 2.8.1.6). Figure 1.6, on the other hand, shows the genes selected by the IKCCA-a method which belong to the biotin metabolism, when the threshold of the gene selection procedure is set in such a way that 10% (74) of all genes are selected. The three known operon genes are selected, as well as a fourth gene (*JW0757*) annotated 2.6.1.62. Figure 1.7 shows the positions of the four selected genes on the genome, where genes *JW0757*, *JW0758*, *JW0759*, and *JW0761* on the genome correspond to their product enzymes EC 2.6.1.62, EC 2.8.1.6, EC 2.3.1.47, and EC 6.3.3.3 respectively in the pathway. One can observe that the four genes selected catalyze four successive reactions in the biotin pathway, and they are adjacent on the genome. The reason why the gene *JW0757* does not belong to the operon formed by the three other genes is that its

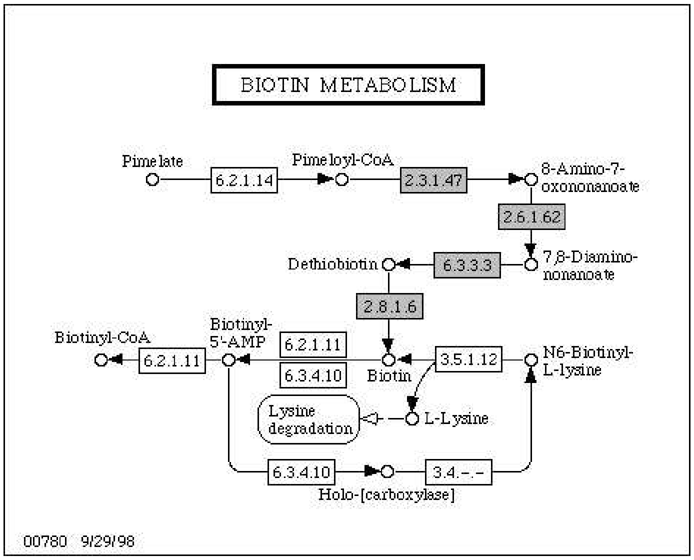

Figure 1.6 An example of operons predicted by IKCCA. The genes selected are represented by the corresponding EC numbers, and colored in gray.

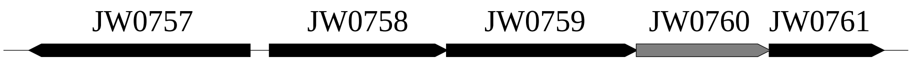

Figure 1.7 The three genes *JW0758*, *JW0759*, and *JW0761* (corresponding to EC 2.8.1.6, EC 2.3.1.47, and EC 6.3.3.3 respectively in the biotin pathway) are part of an operon. Our gene selection method included the gene *JW0757* (corresponding to EC 2.6.1.62 in the pathway) in the operon because it is close to the other genes in the genome and has a similar function. This is a mistake, however, because the orientation of this gene, which corresponds to the direction of translation, is opposite that of the remaining genes.

translation direction is different from that of the other genes. This difference is an important factor in the mechanism of transcription, because a transcription starts from the promoter at the beginning of genes in the same direction. This suggests that further improvements might result from taking into account the direction of the genes in the genome kernel function, which currently only contains distance information.

1.4 Discussion and Conclusion

In this chapter we proposed various approaches to investigate the correlation between heterogeneous genomic data. We proposed several generalized formulations of ordinary KCCA and derived a gene selection procedure based on the newly introduced canonical score. The integration of different types of genomic data (e.g., biochemical pathways, genomes, and expression data) is a key problem in computational biology nowadays. When data types are different (e.g., graphs, strings, and vectors), integration strategies often rely on various heuristic approaches, which depend on the types of data. The originality of our approach is the extension of the concept of correlation for nonvectorial data and integration of genomic data in a rigorous mathematical framework common to all types.

The proposed methods enable us to automatically find correlated directions, along which high/low scoring genes tend to share similarities with respect to multiple biological attributes. These methods give encouraging results on the problem of recognizing the genes that belong to operons in the *E. coli* genome, by comparing three data sets corresponding to *functional* relationships between genes in metabolic pathways, *positional* relationships along the chromosome, and *coexpression* relationships as observed by gene expression data. We observed that generalized KCCAs (MKCCA and IKCCA) outperform ordinary KCCAs in this context. In our preliminary results the number of correct operon candidates selected by MKCCA at a given rate of false predictions tends to be smaller than that selected by the best choice of IKCCA, that is, when the genome data set is compared to the combination of the pathway and the expression data sets. One explanation for this difference in performance might be the fact that MKCCA looks for correlations simultaneously among all pairs of data sets. It would work well if the genes in an operon were systematically similar to each other with respect to all three sources of information we used. To the contrary, in our IKCCA setting, we relax the constraint of having a correlation between gene positions in the pathways and gene expression (which alone gave the worst results), and rather focus on detection of correlations between positions on the genome on the one hand, and positions on the pathways *or* expression profile on the other. Due to noise and errors in the data, this less constrained problem might detect biological phenomena (operons in our case) more easily than the MKCCA approach. We conjecture that as the number of data sets increases, the performance of MKCCA might decrease because it becomes too difficult to impose correlation constraints between any two data sets. In that case it might be more efficient to try to detect correlations between a smaller number of data sets, obtained themselves by combining the initial data sets available, as we did in IKCCA.

From the viewpoint of algorithms, much work remains to be done on testing the influence of kernel parameters on the final performance of the methods. Any real-world application of these methods might require a fine-tuning of each kernel, as well as of the regularization parameters used in the KCCA algorithms.

Finally, it should be pointed out that the canonical variates extracted from the comparison of several data sets can be used as new representations of the genes themselves. This avenue was investigated by Vert and Kanehisa (2003b) with promising results for gene function prediction from heterogeneous data.

Acknowledgments

Y.Y. and M.K. were supported by grants from the Ministry of Education, Culture, Sports, Science and Technology of Japan, the Japan Society for the Promotion of Science, and the Japan Science and Technology Corporation. The computational resource was provided by the Bioinformatics Center, Institute for Chemical Research, Kyoto University.

References

S. Akaho. A kernel method for canonical correlation analysis. In *Proceedings of the 2000 Workshop on Information-Based Induction Sciences (IBIS2000),* 17–18 July 2000, Izu, Japan, pages 123–128, 2001.

F. R. Bach and M. I. Jordan. Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48, 2002.

M. Eisen, P. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 95:14863–14868, 1998.

M.D. Ermolaeva, O. White, and S.L. Salzberg. Prediction of operons in microbial genomes. *Nucleic Acids Research*, 29:1216–1221, 2001.

S. Goto, Y. Okuno, M. Hattori, T. Nishioka, and M. Kanehisa. LIGAND: Database of chemical compounds and reactions in biological pathways. *Nucleic Acids Research*, 30:402–404, 2002.

M. Gribskov and N. L. Robinson. Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. *Computers and Chemistry*, 20(1):25–33, 1996.

D. Hanisch, A. Zien, R. Zimmer, and T. Lengauer. Co-clustering of biological networks and gene expression data. *Bioinformatics*, 18:S145–S154, 2002.

T. Hastie, A. Buja, and R. Tibshirani. Penalized discriminant analysis. *Annals of Statistics*, 23:73–102, 1995.

H. Hotelling. Relation between two sets of variates. *Biometrika*, 28:321–377, 1936.

T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proceedings of the National Academy of Sciences of the United States of America*, 98(8): 4569–4574, 2001.

T. Ito, K. Takemoto, H. Mori, and T. Gojobori. Evolutionary instability of operon structures disclosed by sequence comparisons of complete microbial genomes. *Molular Biology and Evolution*, 16:332–346, 1999.

R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis*. Paramus, NJ, Prentice Hall, 1998.

M. Kanehisa, S. Goto, S. Kawashima, and A. Nakaya. The KEGG databases at genomenet. *Nucleic Acids Research*, 30:42–46, 2002.

R. I. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete structures. In C. Sammut and A. G. Hoffmann, editors, *Machine Learning, Proceedings of the 19th International Conference (ICML 2002)*, pages 315–322. San Francisco, Morgan Kaufmann, 2002.

S. Leurgans, R. Moyeed, and B. Silverman. Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society*, B55:725–740, 1993.

A. Nakaya, S. Goto, and M. Kanehisa. Extraction of correlated gene clusters by multiple graph comparison. In H. Matsuda, S. Miyano, T. Takagi, and L. Wong, editors, *Genome Informatics 2001*, pages 44–53. Tokyo, Universal Academy Press, 2001.

A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T.G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14, pages 849–856, Cambridge, MA, MIT Press, 2002.

H. Ogata, W. Fujibuchi, S. Goto, and M. Kanehisa. A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic Acids Research*, 28:4021–4028, 2000.

P. Pavlidis, J. Weston, J. Cai, and W. N. Grundy. Gene functional classification from heterogeneous data. In *Proceedings of the Fifth Annual International Conference on Computational Biology (RECOMB)*, pages 249–255. New York, ACM Press, 2001.

M. Pellegrini, E. M. Marcotte, M. J. Thompson, D. Eisenberg, and T. O. Yeates. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 96(8):4285–4288, 1999.

H. Salgado, G. Moreno-Hagelsieb, T.F. Smith, and J. Collado-Vides. Operons in *Escherichia coli:* Genomic analysis and prediction. *Proceedings of the National Academy of Sciences of the United States of America*, 97:6652–6657, 2000.

B. Schölkopf, A. J. Smola, and K.-R. Müller. Kernel principal component analysis. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 327–352. MIT Press, 1999.

J.-P. Vert and M. Kanehisa. Graph-driven features extraction from microarray data. Technical report, Arxiv, June 2002.

J.-P. Vert and M. Kanehisa. Extracting active pathways from gene expression data. *Bioinformatics*, 19:238ii–234ii, 2003a.

J.-P. Vert and M. Kanehisa. Graph-driven features extraction from microarray data using diffusion kernels and kernel CCA. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15, pages 1425–1432. Cambridge, MA, MIT Press, 2003b.

D.M. Walters, R. Russ, H.J. Knackmuss, and P.E. Rouviere. High-density sampling of a bacterial operon using mRNA differential display. *Gene*, 273(2):305–315, 2001.

Y. Weiss. Segmentation using eigenvectors: a unifying view. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 975–982. Los Alamitos, CA, IEEE Computer Society, 1999.

T. Yada, M. Nakao, Y. Totoki, and K. Nakai. Modeling and predicting transcriptional units of *Escherichia coli* genes using hidden Markov models. *Bioinformatics*, 15:987–993, 2001.

Y. Yamanishi, J.-P. Vert, A. Nakaya, and M. Kanehisa. Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis. *Bioinformatics*, 19:i323–i330, 2003.

Y. Zheng, J.D. Szustakowski, L. Fortnow, R.J. Roberts, and S. Kasif. Computational identification of operons in microbial genomes. *Genome Research*, 12(8): 1221–1230, 2002.