Université Paris 6

Habilitation à Diriger les Recherches

Spécialité: Mathématiques

Kernel Methods in Computational Biology

Jean-Philippe Vert

Habilitation soutenue le 10 décembre 2004 devant le jury composé de:
Robert Azencott
Olivier Catoni
Pascal Massart (rapporteur)
Bernard Prum (rapporteur)
Alexandre Tsybakov
Excusé: Michael I. Jordan (rapporteur)

Preface

This document is made of three parts:

- a curriculum vitae that describes my background, administrative and scientific activities, grants and publications;
- a research summary that describes my main scientific contributions in the period 2001-2004;
- reprints of selected publications, available online at `http://cg.ensmp.fr/~vert/publi`

My work, at the interface between mathematics, computer science and biology, aims at developping mathematical frameworks and computational algorithms to represent, integrate and analyse the growing quantity of data generated by high-throughput technologies in life science. More precisely, my main contributions have been in the field of statistical machine learning methods to 1) process, analyze and classify structured and heterogeneous data, e.g., biological sequences or chemical compounds, and 2) analyze and infer biological networks. A significant part of my work is based on the theory of kernel methods for statistical learning, a increasingly popular approach in machine learning and computational biology.

It is a pleasure to thank the many people who contributed to my investigations through their advices, collaboration or support. Olivier Catoni, as a PhD supervisor, and Robert Azencott have been scientific mentors for a long time now. Bernard Prum had a decisive influence on the orientation of my research by introducing me to bioinformatics. The community of statisticians around, among others, Pascal Massart and Alexandre Tsybakov remains for me a model of scientific excellence where I can always find support. I am therefore very honored to have this Aroepagus as a jury. I am also grateful to Mike Jordan who, besides being a never-ending source of inspiration for my research, accepted to review this manuscript.

My postdoctoral stay in Japan (2001-2002), supported by Minoru Kanehisa, remains an unforgettable experience both scientifically and personally thanks to all members of the Kyoto University's Bioinformatics Center, in particular Tatsuya Akutsu, Yoshihiro Yamanishi, Hiroto Saigo, Yasushi Okuno, Adeline Bichet, Koichi Tonomura, Yoshinobu Igarashi, Shuichi Kawashima. My research also owes a lot to discussions and collaborations with many friends regularly met at bioinformatics and machine learning conferences, in particular Koji Tsuda, Bill Noble, Nello Cristianini, Bernhard Schölkopf, Olivier Bousquet, Jason Weston, Christina Leslie, Olivier Chapelle, Arthur Gretton, Gert Lanckriet, Francis Bach, Florence d'Alche-Buc, Vincent Schächter, Emmanuel Barillot, Regis Vert, Kenji Fukumizu, Risi Kondor, and John Lafferty.

In the last two years we were able to set up a small but dynamic bioinformatics team at the Ecole des Mines, thanks to the motivation and involvement of Christian Lajaunie, Marco Cuturi, Pierre Mahé, Karen Willbrand, Martial Hue, Joannès Vermorel and Franck Rapaport. A very special thought goes to Véronique Stoven, whose energy can overcome any situation. Our group has constantly been supported by Benoit Legait, Michel Schmitt, Jean-Paul Chilès and all members of the Geostatistics Center, to whom I am deeply grateful.

Finally, nothing would have been possible without the constant support of my very understanding wife Yasuko, and without our children Marina, Kenta and Naoki who make life so beautiful.

Jean-Philippe Vert Fontainebleau, December 5, 2004.

Contents

Chapter 1

Curriculum Vitae

Jean-Philippe Vert

Geostatistics Center / Computational Biology group
Ecole des Mines de Paris
35 rue Saint-Honoré
77305 Fontainebleau cedex, France

Tel : +33 1 64 69 47 82
Fax : +33 1 64 69 47 05
E-mail : Jean-Philippe.Vert@mines.org
http ://www.cg.ensmp.fr/~vert

French nationality.
Born January 18th, 1973 in Paris (France).
Married, three children.

EDUCATION

Sep.1998 - Mar.2000 **PhD in Mathematics : "Statistical methods for natural language modelling"**
Supervised by Olivier Catoni, ENS Paris and Paris 6 University, France.
Analysis and development of adaptive statistical algorithms to analyze large textual databases and biological sequences.
Obtained with the highest honours.

1995 - 1998 **"Corps des Mines" Technology, Economy and Policy program (Paris, France)**
Corps des Mines is a selective 3-year MBA-like program, managed by the French Ministry of Industry to train executive managers for large corporations or public bodies.

1996 - 1997 **M. Sc. in Mathematics (DEA Probability and Applications)**
Paris 6 University
M. Sc. Thesis under the direction of Gilles Pages about the Kohonen algorithm.

1992 - 1995 **Ecole Polytechnique (Palaiseau, France)**
General education in mathematics, physics, chemistry, biology, economics, computer science.
Major in applied mathematics (probability and statistics).
Ranked 8th / 400.

1990 - 1992 **Lycée Henri IV College (Paris, France)**
Preparation in two or three years for the entrance exams of universities like Ecole Polytechnique ("Grandes Ecoles").

1990 **Lycée Jacques Prevert High School (Taverny, France)**
Scientific A-Level (Baccalaureate) with highest honours.

Awards

2004 Simon Regnier prize
This prize, given by the "French Society for Classification", recognizes each year one young researcher for his original contribution to the field of classification.

Experiences

Since October 2002 **Researcher, Bioinformatics group leader** **Fontainebleau, France**
Ecole des Mines de Paris.
Responsible for the creation and the development of a research activity in bioinformatics at the Ecole des Mines. Two years after creation, the team is made of 3 faculties, 1 post-doc and 5 PhD students. Research topics : development of algorithms and mathematical methods for post-genomics (analysis of data obtained from high-throughput technologies, integration of heterogeneous data, complex networks analysis, structure and function prediction, systems biology).

2001 - 2002 **Associate researcher (post-doc) in bioinformatics (17 months)** **Kyoto, Japon**
Kanehisa Laboratory, Bioinformatics center, Kyoto University.
Development of new methods for biological data analysis (biological sequences, microarray data, protein structure and function prediction, regulatory and protein interaction networks analysis).

1999 - 2000 **Scientific consultant** **Cachan, France**
Sudimage Co., start-up.
Scientific consultant for natural language processing applications.

1996 - 1997 **Research scientist (10 months)** **Philadelphia, USA**
Elf Atochem North America, Co.
Study and implementation of a monitoring system to control chemical industrial batch processes using multivariate statistical methods.

1995 - 1996 **Consultant (11 months)** **Romorantin, France**
Matra Automonile Co.
Evaluation of various opportunities concerning the industrial strategy of the company.

1995 **Research internship in mathematics (4 months)** **Kyoto, Japan**
Kyoto University, Department of Mathematics
Stochastic calculus, supervised by Professors Shinzo Watanabe and Ichiro Shigekawa.

1994 **Summer internship (1 month)** **Hamamatsu, Japan**
Hamamatsu Photonics (photodiodes department).

1992 - 1993 **Military Service (1 year)** **Monthlery, France**
Officer and platoon leader.

Selected research grants

NIH R33HG003070-01 (PI W. Noble) : Detecting Relations Among Heterogeneous Datasets (2004-2007)
- Collaboration between the University of Washington (PI : W. Noble, co-PI : D. Baker), the Ecole des Mines de Paris (co-PI : JP. Vert), UC Berkeley (co-PI : M. Jordan and L. El Ghaoui) and UC Davis (co-PI : N. Cristianini)
- 1200 KUSD/3years (NIH, USA)
- The major goal of this project is to develop and test novel methods for gene classification and protein interaction prediction by integration of heterogeneous genomic data.

ACI IMPBIO 2004-47 (PI E. Barillot) : Analysis of gene expression and regulatory networks in cancerous cells (2004-2007)
- Collaboration between the Curie Institute (PI : E. Barillot) and the 'Ecole des Mines de Paris (co-PI : JP. Vert)
- 75 KEUR/3years plus a PhD salary for 3 years (French Ministry of Research)
- The major goal of this project is to develop new computational methods to model the activity and evolution of gene regulatory networks in cancerous cells from gene expression data

GEMBIO 2003 (PI JP. Vert) : Analysis of microarray data for bioagnosis (2004-2007)
- Collaboration between the Ecole des Mines de Paris (PI : JP. Vert, co-PI : F. Meyer) and the Ecole des Mines d'Alès (co-PI : M. Crampes)
- 360 KEUR/3years (French Ministry of Industry)
- The major goal is to set up a pipeline for the analysis of microarray data for diagnosis of cancers and other diseases.

ACI NIM (PI A. Tsybakov) : New aggregation strategies for microarray data classification (2003-2006)
- Collaboration between the University Paris 6 (PI A. Tsybakov) and the Ecole des Mines de Paris (co-PI : JP. Vert).
- 20 KEUR/3years plus a PhD salary for 3 years (French Ministry of Industry)
- The goal of this project is to develop new machine learning algorithms based on aggregation strategies for classification of high-dimensional biological data.

PAI SAKURA 2003 (PI JP. Vert) : Analysis of graphs in bio- and chemo-informatics (2003-2004)
- Collaboration between Ecole des Mines de Paris (PI : JP. Vert) and Kyoto University, Japan (PI : T. Akutsu).
- 32 KEUR/2years
- The goal of this project is to develop new methods combining statistical and combinatorial approaches for the analysis and classification of graphs in bio- and chemo-informatics.

ANVAR 2003 (PI JP. Vert) : Machine learning for drug design (2003-2004)
- 60 KEUR/2years
- The goal of this project is to develop and test new approaches for virtual screening of chemical databases in drug design..

Supervised graph inference
J.-P. Vert and Y. Yamanishi. *Advances in Neural Information Processing Systems 17 (NIPS 2004)*, 2004 (In Press). (Oral presentation, 24 accepted papers as oral out of 822 = 3%)

Semigroup kernels on finite sets
M. Cuturi and J.-P. Vert. *Advances in Neural Information Processing Systems 17 (NIPS 2004)*, 2004 (In Press). (Spotlight presentation, 65 accepted papers as spotlight or oral out of 822 = 8%)

Protein network inference from multiple genomic data : a supervised approach
Y. Yamanishi, J.-P. Vert, and M. Kanehisa *Bioinformatics*, vol. 20, p. i363-i370, 2004. (Proceedings of ISMB 2004). (67 accepted papers out of 474 = 14%)

Extensions of marginalized graph kernels
P. Mahé, N. Ueda, T. Akutsu, J.-L. Perret and J.-P. Vert. *Proceedings of the Twenty-First International Conference on Machine Learning (ICML'2004)*, R. Greiner and D. Schuurmans (Eds.), p.552-559, ACM Press, 2004. (118 accepted papers out of 368 = 32%).

A mutual information kernel for sequences
M. Cuturi and J.-P. Vert. *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2004)*, 2004 (in press).

Protein homology detection using string alignment kernels
H. Saigo, J.-P. Vert, T. Akutsu and N. Ueda. *Bioinformatics*, vol.20, p.1682-1689, 2004.

Kernel Methods in Computational Biology
J.-P. Vert, K. Tsuda and B. Schoelkopf (Eds.), MIT Press, 2004.

A primer on kernel methods
J.-P. Vert, K. Tsuda and B. Schoelkopf, in *Kernel Methods in Computational Biology*, B. Schoelkopf, K. Tsuda and J.-P. Vert (Ed.), MIT Press, p.35-70, 2004.

Local alignment kernels for biological sequences
J.-P. Vert, H. Saigo and T. Akutsu. In *Kernel Methods in Computational Biology*, B. Schoelkopf, K. Tsuda and J.-P. Vert (Ed.), MIT Press, p.131-154, 2004.

Diffusion kernels
R. Kondor and J.-P. Vert, in *Kernel Methods in Computational Biology*, B. Schoelkopf, K. Tsuda and J.-P. Vert (Ed.), MIT Press, p.171-192, 2004.

Heterogeneous data comparison and gene selection with kernel canonical correlation analysis
Y. Yamanishi, J.-P. Vert and M. Kanehisa, in *Kernel Methods in Computational Biology*, B. Schoelkopf, K. Tsuda and J.-P. Vert (Ed.), MIT Press, p.209-230, 2004.

Extracting active pathways from gene expression data
J.-P. Vert and M. Kanehisa, *Bioinformatics*, vol. 19, p. i238-ii244, 2003. Proceedings of ECCB 2003. (27 accepted papers out of 124 = 22%).

Extraction of Correlated Gene Clusters from Multiple Genomic Data by Generalized Kernel Canonical Correlation Analysis
Y. Yamanishi, J.-P. Vert, A. Nakaya and M. Kanehisa *Bioinformatics*, vol. 19, p. i323-i330, 2003. Proceedings of ISMB 2003. (35 accepted papers out of 242 = 14%).

Graph-driven features extraction from microarray data using diffusion kernels and kernel CCA
J.-P. Vert and M. Kanehisa. *Advances in Neural Information Processing Systems 15*, (NIPS), Suzanna Becker, Sebastian Thrun and Klaus Obermayer (Eds), p. 1425-1432, MIT Press, Cambridge, MA, 2003.MIT Press, Cambridge, MA, 2003. (207 accepted papers out of 694 = 30%).

A tree kernel to analyze phylogenetic profiles
J.-P. Vert, *Bioinformatics*, vol. 18, p. S276-S284, 2002. Proceedings of ISMB 2002. (42 accepted papers out of 207 = 20%).

Genome informatics for data-driven biology : A report on the twelfth international conference on genome informatics, Tokyo, Japan, December 17-19, 2001
K. Nakai and J.-P. Vert, *Genome Biology*, 3(4) :reports 4010.1-4010.3, 2002.

Support vector machine prediction of signal peptide cleavage site using a new class of kernels for strings
J.-P. Vert, *Proceedings of the Pacific Symposium on Biocomputing 2002*, Altman, R.B., Dunker, A.K., Hunter, L., Lauerdale, K. and Klein, T.E., (Ed.), World Scientific, pp. 649-660, 2002.

Detecting transcriptional cis-regulation from gene expression data
Y. Igarashi, Y. Okuno, J.-P. Vert and M. Kanehisa, Genome Informatics Series No. 12, Genome Informatics 2001, pp.241-242, 2001.

Adaptive context trees and text clustering
J.-P. Vert, *IEEE Trans. Inform. Theory*, vol.47, No.5, pp.1884-1901, Jul. 2001.

Statistical methods for natural language modelling
J.-P. Vert, PhD thesis, Paris 6 University, Mar. 2001.

Text categorization using adaptive context trees
J.-P. Vert, *Proceedings of the CICLing-2001 conference, A. Gelbukh (Ed.), LNCS 2004, Springer-Verlag Berlin Heidelberg*, pp. 423-436, 2001.

Double mixture and universal inference
J.-P. Vert, Technical report DMA-00-15, Departement of Mathematics and Applications, Ecole normale superieure, pp. 1-27, May 2000.

Le consensuisse
M. Chevrel and J.-P. Vert, *Annales des Mines - Realites industrielles*, Feb. 1999, pp. 41-46.

Research in written language processing in Japan
J.-P. Vert, MBA thesis, Ecole des Mines de Paris, Jul. 1998.

Two studies on stochastic calculus
G. Kawabe and J.-P. Vert, Graduation thesis, Ecole Polytechnique, Jul. 1995.

Selected talks

- 15/11/2004 : Bioinformatics seminar, Kyoto University, Kyoto, Japan.
- 5/11/2004 : Statistics seminar, Institute for Statistical Mathematics, Tokyo, Japan.
- 8/10/2004 : "Complex stochastic systems in biology and medicine" workshop (*invited*), Ludwig-Maximilians University Munich, Germany.
- 16/9/2004 : EBI Industry Program Meeting, European Bioinformatics Institute (*invited*), Cambridge, UK.
- 8-10/9/2004 : 11th Meeting of the French-speaking Society of Classification (*invited*), Bordeaux, France.
- 30/7/2004 : "Bioinformatics and Statistical Physics" workshop (*invited*), Glasgow, UK
- 21/5/2004 : "Advanced microarray data analysis" workshop (*invited*), Center for Biological Sequence Analysis, Elsinore, Denmark.
- 6/5/2004 : "Réseaux géniques et métaboliques : vers des modèles intégrés" workshop (*invited*) , INRIA, Grenoble, France
- 14/4/2004 : Computational biology seminar, Univesité de Washington (Genome Science), Seattle, USA.
- 6-9/4/2004 : The Learning Workshop (*invited*) , Snowbird, USA.
- 24-26/3/2004 : PFYY2, 121th Event of the European Federation of Biotechnology (*invited*), Biarritz, France.
- 24/3/2004 : Computational biology workshop (*invited*), Institut Pasteur, France.
- 18/2/2004 : Bioinformatics seminar, Max Planck Institut für Informatik,Saarbrücken, Germany.
- 6/2/2004 : Marketing seminar, INSEAD, Fontainebleau, France.
- 5/2/2004 : Functional genomics seminar, CEA, Evry, France.
- 30/1/2004 : Bioinformatics seminar, INRIA, Nancy, France.
- 26/1/2004 : Geometrica seminar, INRIA, Sophia-Antipolis, France.
- 23/1/2004 : Spatial statistics seminar, INAPG, Paris, France.
- 19/1/2004 : Statistics seminar, Université Paris 6, Paris, France.
- 21/11/2003 : Bioinformatics seminar, Ghent University, Ghent, Belgium
- 27/10/2003 : Bioinformatics seminar, Bioinformatics center, Kyoto University, Japan.
- 17/10/2003 : "Machine Learning in Bioinformatics conference" workshop (*invited*), Bruxelles, Belgium
- 27/9/2003 : European Conference on Computational Biology (**ECCB 2003**), Paris, France.
- 9/9/2003 : Bioinformatics seminar, Institut Curie, Paris, France.
- 27/7/2003 : Statistics seminars, Université le Mirail, Toulouse, France.
- 20/6/2003 : Bioinformatics seminar, INSA, Toulouse, France.
- 16/6/2003 : Applied mathematics seminar, Ecole Polytechnique, Palaiseau, France.
- 5/6/2003 : "Geometric models of biological phenomena" workshop (*invited*), American Institute of Mathematics, Palo Alto, USA.
- 9/5/2003 : Statistics seminar, Université de Californie, Berkeley, USA.
- 6/5/2003 : Statistics seminar, Université de Californie, Davis, USA.
- 4/4/2003 : Mathematics seminar,Université de Bretagne-Sud, Vannes, France
- 31/3/2003 : "Statistical analysis of DNA microarray data" workshop (*invited*), Toulouse, France.
- 15-18/1/2003 : "Mathematical aspects of molecular biology : Towards new constructions" workshop (*invited*), Nara, Japan.
- 15/1/2003 : "Statistical Learning in Classification and Model Selection" workshop (*invited*), EURANDOM, Eindhoven, the Netherlands.
- 14/1/2003 : Mathematics and genomics seminar, INRA, Jouy-en-Josas, France.
- 10/1/2003 : Machine learning seminar, Université Paris Sud, Orsay, France.
- 10/12/2002 : 16th Annual Conference on Neural Information Processing Systems (**NIPS 2002**), Vancouver, Canada.
- 14/11/2002 : "Statistical Learning, Theory and Applications" conference, Paris, France.
- 5/11/2002 : Bioinformatics seminar, INSA, Toulouse, France.
- 7/8/2002 : 10th International Conference on Intelligent Systems for Molecular Biology (**ISMB 2002**, Edmonton, Canada.

- 2/8/2002 : 4th BioPathways Consortium Meeting, Edmonton, Canada.
- 17-19/7/2002 : Bioinformatics seminar, Human genome center, Tokyo University, Japan.
- 12/7/2002 : Applied mathematics seminar, Research Institute for Mathematical Science, Kyoto University, Japan.
- 28/6/2002 : Applied mathematics seminar, Graduate school of mathematics, Kyushu University, Japan.
- 22/4/2002 : Bioinformatics seminar, Bioinformatics center, Kyoto University, Japan.
- 8/3/2002 : Bioinformatics seminar, Genetic knowledge science lab, JAIST, Kanazawa, Japan.
- 2/2/2002 : Mathematical modeling and statistical analysis in biomedical research Symposium (*invited*), Hiroshima university, Hiroshima, Japan.
- 7/1/2002 : Pacific Symposium on Biocomputing (**PSB 2002**), Lihue, Hawaii, USA.
- 21/11/2001 : Bioinformatics seminar, Computational Biology Research Center, Tokyo, Japan.
- 20/11/2001 : "Science et Technologie" workshop, Tokyo, Japan
- 21/5/2001 : Bioinformatics seminar, Bioinformatics center, Kyoto University, Japan.
- 29/3/2001 : Statistics seminar, Ecole normale supérieure, Paris, France.
- 1/3/2001 : Applied mathematics seminar, Université Paris Sud, Orsay, France.
- 22/2/2001 : Second International Conference on Intelligent Text Processing and Computational Linguistics (**CICLing 2001**), Mexico City, Mexico.
- 2/11/2000 : "Mathematical foundations of natural Language modeling" IMA workshop (*invited*), University of Minnesota, Minneapolis, USA.
- 30/8/2000 : "Statistique mathématique et applications" workshop, Garchy, France.
- 14/3/2000 : Statistics and genomics seminar, Université d'Evry, Evry, France.
- 13/12/1999 : Applied mathematics seminar, Ecole normale supérieure, Paris, France.

Teaching

M.Sc "Mathematics, Vision and Learning", ENS Cachan (2003-2004)
– 20h (full course), "Kernel methods in computational biology"

M.Sc. "Probability and applications", Paris 6 University (2003-2004)
– 12h (half-course) "Kernel methods in computational biology"

Ecole Nationale Supérieure des Télécommunications (2002-2004)
– 3h/year "Support Vector Machines"

Student supervision

Marco Cuturi
– Full PhD supervision since 11/2002.
– Marco's research deals with algorithms for the analysis and classification of biological sequences
– Two publications in refereed international conferences (IJCNN 2004 et NIPS 2004).

Pierre Mahé
– Full PhD supervision since 9/2003.
– Pierre's research concerns the use of kernel methods for virtual screening of chemical databases in drug design
– One publication in a refereed international conferenc (ICML 2004).

Martial Hue
– M.Sc (4-8/2003) and full PhD (since 9/2004) supervision.
– Martial's research is about semi-supervised learning applied to protein structure prediction

Joannes Vermorel
– Full PhD supervision since 9/2004.
– Joannès started his research on classification algorithms in non-metric spaces.

Franck Rapaport
– PhD supervision since 9/2004 (50%, in collaboration with Emmanuel Barillot, Institut Curie).
– This thesis, supported by the grant ACI IMPBIO 2004-47, focuses on the classification of tumor cells by integration of microarray data and gene regulatory network models.

Simona Toti
– M.Sc supervision (3-9/2003). (50% in collaboration with Vincent Schächter, Génoscope).
– Simona's thesis was about gene regulatory network reconstruction with Bayesian networks

Selected activities

- Executive board member of the alumi associations of Ecole Polytechnique X-Asia and X-Biotech
- Workshop organization :
  - "Kernel methods in computational biology" workshop (satellite to RECOMB'03), Berlin, Germany, April 14, 2003.
  - "Machine learning and computational biology" workshop (satellite to CAP'03), Laval, France, July 4, 2003.
- Reviewer (journals) : Bioinformatics, Discrete Applied Mathematics, IEEE Trans. Inf. Theory, Journal of Bioinformatics and Computational Biology, Journal of Machine Learning Research.
- Reviewer and/or program committee member(conferences) : COLT 2003 and 2004, GIW 2004, ICML 2004, ISMB 2004, NIPS 2003 and 2004,

Miscallenous

Languages :
- **Native French**
- **Fluent English**
- **Japanese** : good level (spoken and written)
- **German** : good level

Computer skills :
- **Programming** : C, C++, Perl, Ruby, Java, MATLAB, HTML.
- **Environnements** : Unix, Linux, Windows, Mac

Hobbies : travel, reading, music.

Chapter 2

Research Summary

2.1 Introduction

The Feb. 16, 2001, issue of Science and Feb. 15, 2001, issue of Nature contain the first analyses of the working draft human genome sequence (Consortium, 2001; Venter, 2001). Two years later, the 50th anniversary of the publication of the landmark paper by Nobel laureates James Watson and Francis Crick that described DNA's double helix (Watson and Crick, 1953) coincided with the official completion of these drafts, concluding one of the most ambitious scientific undertakings of all time. While genome sequencing has nowadays largely become a well-controlled industrial process, it is just one among a number of technical breakthroughs made possible by the biotechnological progresses of the last decades, that are revolutionarizing biology. Many other less visible technologies are synergetically changing biology by providing new methods to observe, monitor, and test, often on a large scale, biological and biochemical systems. To name just a few, *DNA microarrays* (Schena et al., 1995) allow the monitoring of the expression level of tens of thousands of transcripts simultaneously, opening the door to *functional genomics*, the elucidation of the functions of the genes discovered in the genome (DeRisi et al., 1997); high-throughput clone generation and sequencing also enabled the development of *genome-wide mutagenesis* (Coelho et al., 2000) and *RNA interference* experiments (Kamath et al., 2003), providing complementary informations about gene functions; recent advances in ionization technology have boosted large-scale capabilities of *mass spectrometry* and the rapidly growing and maturing field of *proteomics*, focusing on the systematic, large-scale analysis of proteins (Aebersold and Mann, 2003); miniaturization and progresses in material science led to new assays used in *high-throughput screening* of chemical compounds in the pharmaceutical industry.

This fast accumulation of technical and scientific breakthroughs support the idea that we enter a very exciting period where questions in biology, ranging from the development of innovative and individualized therapies to the molecular description of cognitive functions, are expected to dominate the scientific landscape. Indeed, the apparition of these high-throughput technologies raises new hopes in our capacity to better apprehend and dissect the inherent complexity of living systems, and to transform this new knowledge into practical innovation, in particular in the biomedical field. A common pitfall of most new high-throughput methods, however, is that they tend to deliver much less than the hopes they raise, at least on the short term. While pharmaceutical companies complain that the average cost of developing new drugs only increases as new "-omics" are integrated, the conclusion of the human genome sequencing and of the wide use of DNA microarrays have only highlighted the complexity of living organisms and the difficulty to extract useful information from the flood of data generated by high-throughput technologies. On the good side, however, these techniques and data are laying the foundations of a new quantitative biology, and are changing in depth our representations of living systems, thus paving the way for possible scientific breakthroughs. While its objects of research in the post-genomic era remain living systems, the fast evolving biological science requires a unique integration of concepts and methods from outside of traditional biology; in particular mathematics, computer science and physics play an increasing role in the advances of biology. Data need to be stored, organized, processed, and integrated into models to validate or generate hypothesis, and suggest new experiments.

I was lucky enough to start my scientific carrier in this exciting period. While my education had been dominated by mathematics, I have always kept a strong interest in the applications of mathematics to other fields, and a fascination for biology. My PhD thesis (Vert, 2001b), focused on adaptive statistical estimation methods for text processing, gave me

a first opportunity to manipulate the *E.coli* and *HIV* genomes through a collaboration with my supervisor Olivier Catoni, Bernard Prum and Cécile Cot, from the University of Evry. My postdoctoral studies started a few days after the publications of the first draft of the human genome, with an 18-month visit to Minoru Kanehisa's laboratory in Kyoto University's Bioinformatics Center, followed by a 2-year period as the leader of a newly-created research group on computational biology at the Ecole des Mines of Paris. This period was not sufficient to complete my still limited education in molecular biology, but opened my eyes on the breathtaking scientific stakes of this post-genomic era, and on the new challenges raised at the interface of biology, mathematics and computer science. I decided to focus my energy at this interface, and discovered the difficulties and joys of multi-disciplinary research, torn apart between the requirements of rigor and aesthetic of mathematics, performance and tricks of computer science, results and interpretation of biology. Accordingly my modest contribution to this field, summarized below, has globally been constrained by three requirements which I consider as important safeguards in the development of a useful theoretical framework for post-genomic biology:

- Starting from actual biological problems and using real data;
- Developing methods and concepts that lead to implementable and efficient algorithms;
- Providing a rigorous mathematical framework to represent the data and justify the methods proposed.

My long-term scientific objective is to contribute to the development of rigorous mathematical frameworks useful to conceptualize our rapidly changing representation of life and to make prediction about or understand biological phenomena. In order to resolve the possible contradictions between such an ambitious objective and the need to have a scientific contribution on the short term, I have so far mainly focused my work on two concrete and related issues, which can be considered as major bottlenecks in the current post-genome era, although solutions have started to emerge in the recent years:

1. How to represent data as diverse as genome sequences, protein 3D structures, gene expression data, gene regulatory or interactions networks, etc..., in a common theoretical framework, and develop methods to 1) compare, analyze these data, and 2) predict biological properties by integrating this heterogeneous information? This question is motivated by the current difficulty to make sense out of the wealth of heterogeneous data available and generated everyday.

2. How to formally relate and put in a common theoretical framework data about individual biological objects (genes, proteins, metabolites...), on the one hand, and representations of biological systems, on the other hand, such as a graph with individual biological objects as vertices? This framework should in particular provide efficient methods to infer biological systems from high-throughput data about individual objects. This question is motivated by the vision that has emerged in the last few years in the field of systems biology (Kitano, 2001), that suggests that the complexity of life arises from complex interactions between a finite number of basic elements.

For reasons detailed below, I started to investigate in 2001 the possible use of positive definite (p.d.) kernels to represent various types of genomic data (Section 2.2). In order to

make this approach practical, I proposed several new kernel functions for particular types of data (Section 2.3), and more generally investigated several systematic approaches to kernel design (Sections 2.3.1, 2.3.4). Part of the bestiary of kernel functions that were invented for specific type of data and applications in computational biology are reviewed in the book B. Schölkopf, K. Tsuda and myself edited recently (Schölkopf et al., 2004). These kernel functions allow most genomic data about a fixed set of genes (typically all genes of a given organism) to be represented simultaneously in a rigorous framework, and enable the use of kernel methods, such as support vector machines (SVM), for various data analysis or inference tasks. Furthermore, heterogeneous data integration, though not a mature fields, can be easily performed by in this framework by performing operations on kernels that conserve the positive definiteness property.

It also turns out that this approach, initially investigated as a possible solution to the first question above, lends itself particularly well to the development of original methods to tackle the second question, that is, to relate biological data with biological systems. In this case, the main challenge is not to develop new p.d. kernels, but rather to imagine methods in this framework to compare biological data with biological systems, or typically infer biological networks from biological data (Section 2.4).

Section 2.5 will conclude this short research summary by mentioning a few research directions I plan to investigate in the future.

2.2 Kernel methods in computational biology

As opposed to other traditionally data-rich fields, data generated in modern biology are often structured (e.g., protein interaction network, gene sequences, evolutionary tree), high-dimensional and noisy if vectorial (e.g., gene expression data measured by microarrays), and heterogeneous (several types of data can represent the same biological objects, such as the sequence and the expression profile of a gene). A recent branch of machine learning, called *kernel methods*, lends itself particularly well to the study of these aspects, making it rather suitable for problems of computational biology. A prominent example of a kernel method is the *support vector machine* (*SVM*) (Boser et al., 1992; Vapnik, 1998), widely used nowadays for pattern recognition and regression problems. The goal of this introductory section is to remind the reader the basics about kernel methods, and why they are useful in computational biology. The following sections will detail in more details my own contributions.

2.2.1 Positive definite kernels and kernel methods

In this section, inspired by (Vert et al., 2004b), we briefly remind the very basics of kernel methods to introduce the framework where most of my research has focused. More complete presentations can be found in several reviews or books, including for example (Vapnik, 1998; Schölkopf and Smola, 2002; Berg et al., 1984; Saitoh, 1988; Cristianini and Shawe-Taylor, 2000; Shawe-Taylor and Cristianini, 2004). The basic philosophy of SVMs and kernel methods is that with the use of a certain type of similarity measure (called a kernel), any type of data can be implicitly embedded in a Hilbert feature space, in which linear methods are used for learning and estimation problems. More precisely, if we assume that the goal is to analyze data from a set $\mathcal{X}$, we have the following definition:

Definition 1 *A function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a* positive definite kernel *(denoted p.d. kernel, or simply kernel below) iff it is symmetric, that is, $K(\mathbf{x}, \mathbf{x}') = K(\mathbf{x}', \mathbf{x})$ for any two objects $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, and positive definite, that is,*

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0$$

for any $n > 0$, any choice of n objects $\mathbf{x}_1, \cdots, \mathbf{x}_n \in \mathcal{X}$, and any choice of real numbers $c_1, \cdots, c_n \in \mathbb{R}$.

The main reason for focusing on p.d. kernels is that they are equivalent to embedding the space $\mathcal{X}$ in a Hilbert space, often called the feature space:

Theorem 2 *For any kernel K on a space $\mathcal{X}$, there exists a Hilbert space $\mathcal{F}$ and a mapping $\phi : \mathcal{X} \to \mathcal{F}$ such that*

$$K(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle, \quad \text{for any } \mathbf{x}, \mathbf{x}' \in \mathcal{X}, \tag{2.1}$$

where $\langle u, v \rangle$ represents the dot product in the Hilbert space between any two points $u, v \in \mathcal{F}$.

This result shows that it is possible to perform implicitly any operations in the feature space that only require inner product computation. This apparently trivial observation, usually referred to as the *kernel trick*, has huge practical implications when one realizes that most linear statistical methods fulfill this constraint: large-margin classification and regression (Vapnik, 1998), Gaussian processes and Kriging (Williams, 1998), principal component analysis (Schölkopf et al., 1999), canonical correlation analysis and independent component analysis (Bach and Jordan, 2002), logistic regression (Zhu and Hastie, 2001) and linear Fisher discriminant (Mika et al., 1999), to name just a few.

A dual realization of the Hilbert structure generated by a kernel, is its reproducing kernel Hilbert space (RKHS) (Saitoh, 1988), that is, the unique Hilbert space $\mathcal{H}_K \subset \mathbb{R}^{\mathcal{X}}$ of functions that satisfy:

$$\begin{cases} \forall \mathbf{x} \in \mathcal{X}, & K(\mathbf{x}, \cdot) \in \mathcal{H}_K \\ \forall (f, \mathbf{x}) \in \mathcal{H}_K \times \mathcal{X}, & \langle f, K(\mathbf{x}, \cdot) \rangle = f(\mathbf{x}) \end{cases}$$

Among other properties, this functional space turns out to be a very convenient space to solve a wide range of optimization problems due to the famous representer theorem first stated with less generality by (Kimeldorf and Wahba, 1971) (see proof for example in (Vert et al., 2004b)):

Theorem 3 *Let $\mathcal{X}$ be a set endowed with a kernel K, and $\mathcal{S} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\} \subset \mathcal{X}$ a finite set of objects. Let $\Psi : \mathbb{R}^{n+1} \to \mathbb{R}$ be a function of $n+1$ arguments, strictly monotonic increasing in its last argument. Then any solution of the problem*

$$\min_{f \in \mathcal{H}_K} \Psi\left(f(\mathbf{x}_1), \cdots, f(\mathbf{x}_n), \| f \|_{\mathcal{H}_K}\right), \tag{2.2}$$

where $(\mathcal{H}_K, \| . \|_{\mathcal{H}_K})$ is the RKHS associated with K, admits a representation of the form

$$\forall \mathbf{x} \in \mathcal{X}, \quad f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(\mathbf{x}_i, \mathbf{x}). \tag{2.3}$$

Theorem 3 shows the dramatic effect of regularizing a problem by including a dependency in $\| f \|_{\mathcal{H}_K}$ in the functional to optimize. First, this norm being penalized in the optimization process, it forces the solution to have a "small" norm, which typically corresponds to being "smooth". Various statistical arguments support this strategy to ensure that the solution may be used successfully to process new points, e.g., to predict a class in the case of SVMs (Vapnik, 1998). While other penalization schemes can be imagined, the representer theorem shows that this penalization also has substantial computational advantages: any solution to (2.2) is known to belong to a subspace of $\mathcal{H}_K$ of dimension at most n, the number of points in $\mathcal{S}$, even though the optimization is carried out over a possibly infinite-dimensional space $\mathcal{H}_K$. A practical consequence is that (2.2) can be reformulated as an n-dimensional optimization problem, by plugging (2.3) into (2.2) and optimizing over $(\alpha_1, \cdots, \alpha_n) \in \mathbb{R}^n$.

As an example, the kernel-PCA algorithm (Schölkopf et al., 1999) consists in minimizing iteratively the following functional:

$$f_i = \underset{f \perp \{f_1, \ldots, f_{i-1}\}}{\arg\max} \frac{1}{n \| f \|_{\mathcal{H}_K}^2} \sum_{i=1}^{n} f(\mathbf{x}_i)^2.$$

which, by application of the representer theorem and basic linear algebra, boils down to diagonalizing the matrix $\tilde{K} = (I - e/n)K(I - e/n)$, where I is the identity matrix and e is the singular matrix with all entries equal to 1 (Vert et al., 2004b).

2.2.2 Kernel methods in computational biology

Kernels methods share several properties that make them potentially suitable for application in computational biology. First, thanks to the kernel trick, they can be applied to the processing of any kind of data as long as p.d. kernels are properly defined on the data to be processed. Hence, processing biological sequences is potentially neither more nor less difficult than processing vectors, graphs, or more complex objects. Second, once a p.d. kernel is defined, the whole machinery of kernel methods can be applied without further effort. This opens the possibility to develop original approaches to difficult problems, such as classification or regression on sequences or graphs. Third, they offer a rigorous mathematical framework to represent biological data by kernel functions. For example, for a given set of genes, the knowledge of their coding sequence in DNA can be represented directly by a kernel function (see Figure 2.1) and not by the set of sequences. Hence this is a first step towards a theoretical framework to represent knowledge about biological systems. Fourth, the set of p.d. kernels on a given space has a rich mathematical structure: it is a convex and pointed cone, closed under point-wise convergence and Schur product (Berg et al., 1984). By representing one biological knowledge (e.g., the data provided by one high-throughput experiment) as one point in this space, i.e., one p.d. function, various mathematical operations can be performed in this space, e.g. to integrate heterogeneous data by taking the center of the corresponding kernels (Pavlidis et al., 2001; Yamanishi et al., 2004a), or by formulating optimization problems in the space of p.d. kernels and using the strong development of semi-definite programming in the recent years (Lanckriet et al., 2004). Fifth and not least, kernel methods are considered at the state-of-the-art level of performance in many real-world applications, which suggests they may be able to provide powerful algorithms useful for biology. Kernel methods, in particular SVM, have indeed invaded the field of computational biology during the last five years (see an a review in (Noble, 2004), and several recent contributions in (Schölkopf et al., 2004)).

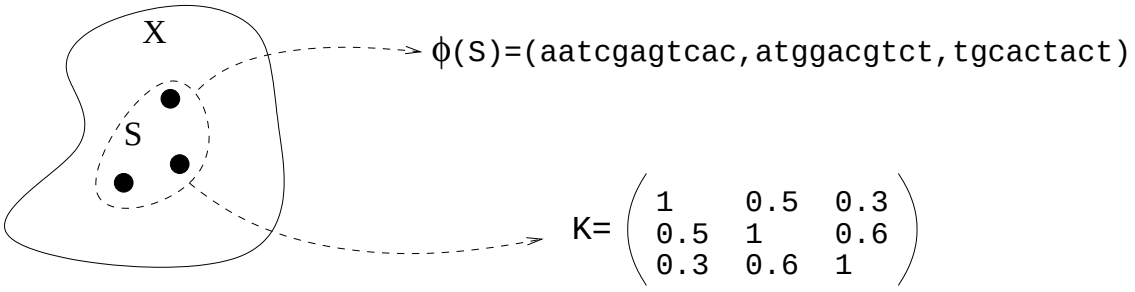


Figure 2.1: Two different representations of the same dataset. $\mathcal{X}$ is supposed to be the set of all oligonucleotides, and $\mathcal{S}$ is a data set of three particular oligonucleotides. The classic way to represent $\mathcal{S}$ is first to define a representation $\phi(\mathbf{x})$ for each element of $\mathbf{x} \in \mathcal{X}$, for example, as a sequence of letters to represent the succession of nucleotides, and then to represent $\mathcal{S}$ as the set $\phi(\mathcal{S})$ of representations of its elements (*upper part*). Kernel methods are based on a different representation of $\mathcal{S}$, as a matrix of pairwise similarity between its elements (*lower part*).

2.3 Kernel design

Kernel methods first attracted my attention in the context of computational biology through two seminal papers of David Haussler's group at UCSC. In the first paper, Jaakkola et al. (2000) proposes a clever approach (the so-called Fisher kernel) to apply the SVM algorithm to the classification of biological sequences, and showed very promising experimental results on the problem of detecting remote homology between protein primary seqences; in the (unpublished) second paper, Haussler (1999) lays the foundations for the use of kernel methods for non-vectorial methods, suggesting in particular new ways to build kernels for strings and trees through an operation of convolution. These papers considerably influenced my research by suggesting to have look in more details at the possibilies offered by kernel methods for non-vectorial data.

2.3.1 P-kernels and graphical models

In two publications (Vert, 2002a,b) I proposed and tested an original approach to define a p.d. kernel on a discrete set endowed with a probability distribution, typically defined as a graphical model (Lauritzen, 1996). The main motivation behind this work is the fact that graphical models are widely used in computational biology to model for example fixed-length sequences with motifs (Gribskov et al., 1990), variable-length sequences with Markov or hidden Markov models, or multiple alignments with phylogenetic trees, i.e., tree graphical models (Durbin et al., 1998).

The approach borrows the concept of P-kernel on a discrete set $\mathcal{X}$, defined by Haussler (1999) as the set of p.d. kernels that satisfy:

$$\begin{cases} \forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2, & K(\mathbf{x}, \mathbf{x}') \geq 0, \\ \sum_{(\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2} K(\mathbf{x}, \mathbf{x}') = 1. \end{cases}$$

In other words, a P-kernel is a p.d. kernel that defines a probability distribution on $\mathcal{X}^2$. The

RKHS defined by a P-kernel is the set of functions $f : \mathcal{X} \to \mathbb{R}$ of the form:

$$\begin{aligned} f(\mathbf{x}) &= \sum_{1=1}^{N} \lambda_i P(\mathbf{x}, \mathbf{x}_i) \\ &= \sum_{1=1}^{N} \lambda_i P(\mathbf{x}_i) P(\mathbf{x}|\mathbf{x}_i), \end{aligned}$$

and their point-wise limits. Two extreme examples of P-kernels are the independent and diagonal kernels (Haussler, 1999), defined with respect to a prior probability distribution p on $\mathcal{X}$ respectively by:

$$K_{ind}(\mathbf{x}, \mathbf{x}') = \begin{cases} p(\mathbf{x}) & \text{if } \mathbf{x} = \mathbf{x}', \\ 0 & \text{otherwise,} \end{cases}$$

and

$$K_{prod}(\mathbf{x}, \mathbf{x}') = p(\mathbf{x}) p(\mathbf{x}').$$

The corresponding RKHS are the limits of functions of the form:

$$f_{ind}(\mathbf{x}) = cte \times \delta(\mathbf{x}, \mathbf{x}_1),$$

where δ is the Dirac symbol, and:

$$f_{prod}(\mathbf{x}) = \left(\sum_{1=1}^{N} \lambda_i P(\mathbf{x}_i) \right) P(\mathbf{x}) = cte \times P(\mathbf{x}),$$

respectively. Hence neither of these kernels is very interesting: the RKHS associated to the independent kernel is "too large", and no learning is possible because all points are orthogonal to each other, while the RKHS associated to the product kernel is "too small", as it basically reduces to the distribution p itself.

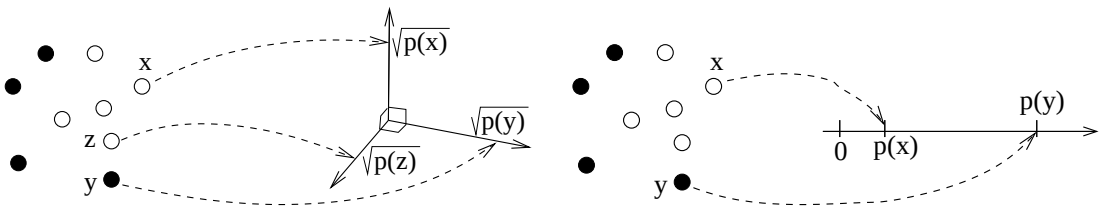
Haussler (1999) and Watkins (2000) proved independently that a possibly interesting P-kernel for gene sequences is defined by pair-HMM (Durbin et al., 1998), under some hypothesis on the parameters of the HMM. In (Vert, 2002a,b) I suggest a different approach to obtain potentially interesting P-kernels by *interpolating* between the independent and product kernels, in the case when there is some structure in the data. More precisely I consider the case where data are n-tuples of discrete variables, that is,

$$\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{X} = \mathcal{A}_1 \times \ldots \times \mathcal{A}_n.$$

Then for any subset of indices $\mathcal{I} \subset [1, \ldots, n]$, the following is a valid P-kernel:

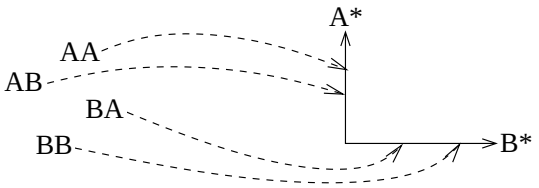
$$K_{\mathcal{I}}(\mathbf{x}, \mathbf{x}') = p(\mathbf{x}_{\mathcal{I}}) \delta(\mathbf{x}_{\mathcal{I}}, \mathbf{x}'_{\mathcal{I}}) \times p(\mathbf{x}_{\mathcal{I}^c}|\mathbf{x}_{\mathcal{I}}) p(\mathbf{x}'_{\mathcal{I}^c}|\mathbf{x}'_{\mathcal{I}}),$$

where $\mathbf{x}_{\mathcal{I}} = (\mathbf{x}_i : i \in \mathcal{I})$. In other words, this $\mathcal{I}$-interpolated kernel applies the independent kernel to the objects restricted to their components in $\mathcal{I}$, and a conditional product kernel to the remaining components. With this kernel, two objects are orthogonal as soon as they do not perfectly match on the components indexed by $\mathcal{I}$. Otherwise, the value of the kernel depends on the conditional probabilities assigned to the other components. Hence this allows to use the product kernel at a finer granularity than the whole space. Figure 2.2 illustrates



(a) Independent kernel

(b) Product kernel



(c) Interpolated kernel

Figure 2.2: Representation of the feature spaces associated to the independent kernel (a), the product kernel (b), and the interpolated kernel (c). The interpolated kernel applies the independent kernel on the first letter, and the product kernel on the second kernel. The dimension of the feature space of the interpolated kernel is between the dimensions of the feature spaces of the product kernel (1) and independent kernel (N).

the difference between the resulting feature space. If enough prior knowledge is available, then one might consider choosing a precise set $\mathcal{I}$ for a given application. However such situations rarely occur, and in case it happens then one might rather split by himself the set of data according to their values on $\mathcal{I}$. A more common and less trivial situation is when one's prior knowledge allows him to define a set of index subsets $\mathcal{V} = \{\mathcal{I}_i, \ldots, \mathcal{I}_p\}$, in the sense that he believes the values of the data restricted to some of these subsets are important for a given problem, without knowing precisely which one. As an example, in the case of fixed-length strings, then one might believe that short-length substrings are important, and consider all subset of indices made of consecutive indices (Vert, 2002a). In order to define a kernel in such a case, I proposed the formula:

$$K_{\mathcal{V}}\left(\mathbf{x}, \mathbf{x}'\right) = \frac{1}{|\mathcal{V}|} \sum_{\mathcal{I} \in \mathcal{V}} K_{\mathcal{I}}\left(\mathbf{x}, \mathbf{x}'\right). \tag{2.4}$$

The resulting kernel is obviously a P-kernel, related to the product kernel by the following (Vert, 2002a):

$$K_{\mathcal{V}}\left(\mathbf{x}, \mathbf{x}'\right) = K_{prod}\left(\mathbf{x}, \mathbf{x}'\right) \times \frac{1}{|\mathcal{V}|} \sum_{\mathcal{I} \in \mathcal{V}} \frac{\delta\left(\mathbf{x}_{\mathcal{I}}, \mathbf{x}'_{\mathcal{I}}\right)}{p\left(\mathbf{x}_{\mathcal{I}}\right)}.$$

This equation highlights the role played both by $\mathcal{V}$ and p in this formulation: *the kernel between two objects increases when they share rare (according to p) common subparts, as defined by the list of index subsets in $\mathcal{V}$.*

In many practical cases the set $\mathcal{V}$ is expected to be large, typically to grow exponentially with the number of variables n. Hence the equation (2.4) is intractable in practice, due to the

possible large number of terms in the sum. It turns out, however, that several choices of sets $\mathcal{V}$ and probabilities p lead to tractable computations, because the kernel can be factorized in different ways. As an example, the following choices lead to linear-time implementation of the kernel with respect to the number of variables n, although $|\mathcal{V}|$ increases exponentially with n:

1. When the variables are independent, i.e.,

$$p(\mathbf{x}) = \prod_{i=1}^{n} p_i(x_i),$$

and

$$\mathcal{V} = \mathcal{P}([1, \ldots, n])$$

is the power set of $[1, \ldots, n]$, that is, the set of all subsets of indices (hence $|\mathcal{V}| = 2^n$).

2. When p is a first-order Markov chain, namely:

$$p(\mathbf{x}) = p_1(x_1) \prod_{i=2}^{n} p_i(x_i|x_{i-1}),$$

and $\mathcal{V}$ is the set of all contiguous subsets of indices, i.e.,

$$\mathcal{V} = \{[i, j] : 1 \leq i \leq j \leq n\} \cup \{\emptyset\}$$

3. When p is a tree graphical model, i.e., when a rooted tree with n nodes numbered from 1 to n is defined and p factorizes as:

$$p(\mathbf{x}) = p(x_{\text{root}}) \prod_{\text{s node}} p\left(x_s|x_{f(s)}\right),$$

where $f(s)$ denotes the unique parent node of node s in the tree, and when $\mathcal{V}$ is the set of all rooted subtrees of the original tree, that is, the set of all connected subgraphs that contain the root node.

4. When p is a tree graphical model, and $\mathcal{V}$ is the set of all subtrees of the original tree, that is, the set of all connected subgraphs of the original tree.

The first two cases are treated in (Vert, 2002a), the third one in (Vert, 2002b), and the fourth one is only a slight generalization of the third one that can easily be implemented (unpublished). In (Vert, 2002b) it is also shown that the probabilistic model may contain hidden variables, and a kernel for the observed variables can still be defined (by summing over the hidden variables to keep the interpretation of the kernel as a probabilistic distribution) and computed with the same linear complexity. The factorizations that lead to linear-time computation of the corresponding kernels are detailed in the mentioned references, and can intuitively be understood as a combination of two tricks: first, the possibility to compute $p(\mathbf{x})$ using a message-passing algorithm (Pearl, 1988) when p is defined by a Bayesian network, and more generally to factorize the computation of one kernel $K_{\mathcal{I}}$ along the branch of the tree, when $\mathcal{I}$ is a subtree; second, a trick to perform a sum over all subtrees of a functional that

factorizes along the branches of the trees, used for example in the context tree weighting compression algorithm (Willems et al., 1995) or in (Vert, 2001a).

Tested on real-world data, these kernels exhibited promising performances on the task of detecting a signal in biological sequences with a moving window (Vert, 2002a), and on the task of prediction of functional class of all genes of the yeast genome using only the information about the presence or absence of homologs in other sequenced genomes (Vert, 2002b). In the later case the kernel is between a set of observed variables at the leaves of a graphical tree model that represent an evolutionary tree. Internal nodes are hidden variables. This application is particularly well motivated because the features of the resulting kernel can be given relevant biological interpretation as evolutionary patterns, knwon to be characteristic of biological functions (Figure 2.3).

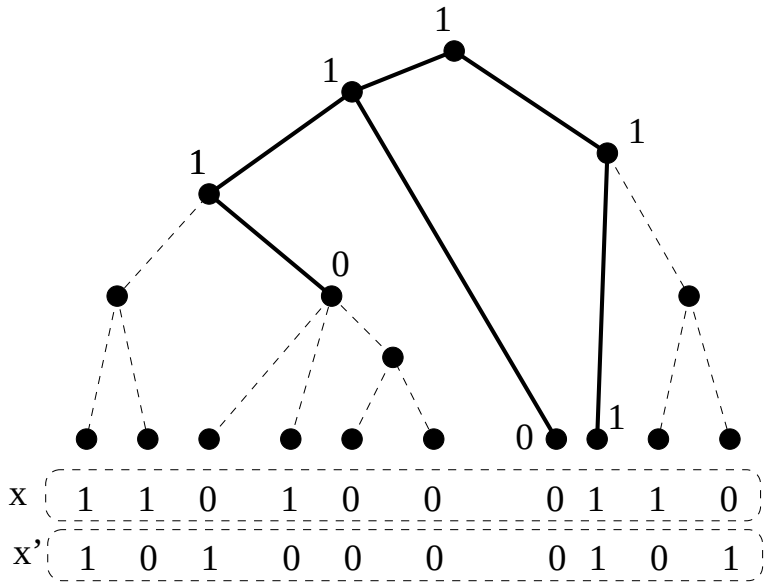


Figure 2.3: In (Vert, 2002b) a kernel between phylogenetic profiles is proposed. The phylogenetic profile of a gene is a string of n bits that indicates the presence or absence of the gene in n sequenced genomes. It can be used to infer the function of a gene, between genes with similar profiles are likely to act in common and therefore to have similar function. The kernel between two profiles $\mathbf{x}$ and $\mathbf{x}'$ is built by considering them as observed variables at the leaves of a tree graphical model, representing the tree of evolution. The features of the kernel are patterns of evolution, i.e., partial assignment of bits to the hidden variables at the node of a rooted subtree. Such patterns are expected to be good characteristics of protein function.

Future work: Kernels defined from probabilistic models, in particular graphical models, are likely to be increasingly useful in the future as graphical models themselves are more and more present to model complex distributions. I therefore plan to continue the investigation of the relationships between both approaches, both in theory and in terms of efficient implementations. More generally, further theory seems to be required to fully understand whether P-kernels have specific properties in terms of learning capabilities, for example, and what consequences this may have on learning algorithms.

2.3.2 Local alignment kernels and remote homology detection

In the last twenty years, nucleotide or amino-acid sequences analysis has by far dominated the field of computational biology. Databases are full of sequences to compare, classify and annotate. In order to process sequences in the framework of kernel methods, as advocated in Section 2.2.2, kernels adapted to such strings must be developed. Potential applications of such a kernel include supervised classification of sequences into functional or evolutionary categories by SVM, analysis of protein families by kernel-PCA, or detection of correlation between the sequence and various properties of the genes with kernel canonical correlation analysis, for example. Given those possibilities, there has recently been a growing interest in the development of kernel functions for biological sequences, including the Fisher kernel (Jaakkola et al., 2000), spectrum kernel (Leslie et al., 2002), mismatch kernel (Leslie et al., 2003), pairwise kernel (Liao and Noble, 2002), and the string kernel proposed by (Lodhi et al., 2002). All these kernels but the last one require the explicit representation of sequences by a finite-dimensional vector.

In collaboration with T. Akutsu, N. Ueda and H. Saigo at Kyoto University, we investigated in (Saigo et al., 2004; Vert et al., 2004a) an alternative approach, based on the following simple idea: because a kernel function can often be thought of as a measure of similarity, why not use as string kernels the measures of similarities traditionally used in computational biology to assess the similarity between biological sequences? The rational behind this idea is that measures of similarity such as the Smith-Waterman score (Smith and Waterman, 1981; Durbin et al., 1998) have been optimized over the years, and are currently accepted as providing a relatively good notion of biological similarity – typically related to the evolutionary distance between sequences in case of orthologs. The question, of course, is to know whether or not such similarity measures are p.d. kernels or not, and under which conditions they can be used by kernel methods.

In order to clarify the concepts, let us first recall in a nutshell the basic definitions of local sequence alignment (see (Vert et al., 2004a) for more details). Roughly speaking, an alignment between two sequences is an arrangement of one sequence on top of the other, some pairs of letters being aligned on top of each others and gaps being inserted when necessary. As an example, the following represents one possible alignment between the strings $\mathbf{x} =$ `GAATCCG` and $\mathbf{x}' =$ `GATTGC` :

```
G-AATCCG-
GAT-T-G-C
```

This example shows 4 aligned pairs: $(G,G), (A,T), (T,T)$ and (C,G), and 3 inserted gaps of length 1 between the first and the last aligned pair (note that we do not count the gaps inserted before the first and after the last aligned positions). The biological relevance of such an alignment is usually assessed by a numeric score, parametrized by a similarity matrix $S : \mathcal{A}^2 \to \mathbb{R}$ between letters and a gap penalty function $f : \mathbb{N} \to \mathbb{R}$, by summing the similarities between aligned letters and removing penalties corresponding to the length of each gap inserted in one sequence or the other, between the first and the last aligned positions. As an example the score of the previous alignment π would be:

$$s\left(\mathbf{x}, \mathbf{x}', \pi\right) = S(G,G) + S(A,T) + S(T,T) + S(C,G) - 3f(1).$$

The name "local alignment" stems from the fact that no penalty for gaps is counted before the first and after the last aligned positions. Finally, the local alignment score – or Smith-

Waterman score – between two sequences is defined as the largest possible score among local alignments, namely:

$$SW\left(\mathbf{x},\mathbf{x}'\right)=\max_{\pi}\ s\left(\mathbf{x},\mathbf{x}',\pi\right) \tag{2.5}$$

This similarity score can be computed by dynamic programming with a complexity $O\left(|\,\mathbf{x}\,|\cdot|\,\mathbf{x}'\,|\right)$, and is widely used in computational biology, either directly or through faster heuristics such as BLAST (Altschul et al., 1997) of FASTA (Pearson, 1990) to query databases.

The question of whether or not it defines a p.d. kernel for sequences was investigated in (Vert et al., 2004a), where it is shown that it can be a p.d. kernel for particular choices of S and f, but that it is generally not p.d. with classical parameter settings used in practice. However, we proved that the following slightly different version of the alignment score is a p.d. kernel:

Theorem 4 *If S is symmetric and conditionally positive definite*[1] *then the following equation defines a valid p.d. kernel, called the* local alignment kernel *(or LA kernel) for any $\beta\geq 0$:*

$$K_{LA}^{(\beta)}=\sum_{\pi}\exp\left\{\beta s\left(\mathbf{x},\mathbf{x}',\pi\right)\right\}.$$

The complete proof of this result, which can be found in (Vert et al., 2004a), relies on the expression of K as a sum of convolutions kernels (Haussler, 1999). This results shows that a valid p.d. kernel is obtained when the contributions of all possible alignments are summed up after exponentiation, while the Smith-Waterman (2.5) score only keeps the contribution of the highest-scoring alignment. Both are related by the following equality:

$$\forall(\mathbf{x},\mathbf{x}')\in\mathcal{X}^2,\quad \lim_{\beta\to+\infty}\frac{1}{\beta}\ln K_{LA}^{(\beta)}(\mathbf{x},\mathbf{x}')=SW(\mathbf{x},\mathbf{x}'). \tag{2.6}$$

Just like the Smith-Waterman score, the LA kernel between two strings can be computed with a complexity $O\left(|\,\mathbf{x}\,|\cdot|\,\mathbf{x}'\,|\right)$ (Saigo et al., 2004; Vert et al., 2004a). It basically boils down to translating the Smith-Waterman dynamic programming algorithm from the $(\min,+)$ tropical semiring to the $(\oplus_{\log},+)$ log semiring[2] (Kuich and Salomaa, 1986).

In spite of this theoretical results the LA kernels turns out to be a bad choice of kernel in practice. The reason is that, like several other string kernels, its values vary on an exponential scale. More precisely, they roughly increase exponentially with the length of meaningful alignments, and decreases exponentially with the number of mismatches or gaps between two sequences. The RKHS associated with such kernels is very "large", in the sense that most pairs of points are almost orthogonal; learning from examples in such spaces is difficult because it requires a lot of examples, to "cover" the whole space (Schölkopf et al., 2002). In order to overcome this difficulty, we proposed to take the logarithm of the LA kernel as a candidate string kernel:

$$\tilde{K}_{LA}^{(\beta)}(\mathbf{x},\mathbf{x}')=\frac{1}{\beta}\ln K_{LA}^{(\beta)}(\mathbf{x},\mathbf{x}'). \tag{2.7}$$

An obvious caveat with this operation is that the logarithm of a positive definite kernel is not a positive definite kernel in general (Berg et al., 1984). In concrete application, a post-processing might therefore be required to ensure that the pairwise kernel matrix computed on

[1] A symmetric function $f:\mathcal{X}^2\to\mathbb{R}$ is conditionally positive definite if and only if $\sum_{i,j=1}^{n}\alpha_i\alpha_j f\left(\mathbf{x}_i,\mathbf{x}_j\right)\geq 0$ for any $n\in\mathbb{N}$, any $\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n\in\mathcal{X}^n$, and any $\alpha_1,\alpha_2,\ldots,\alpha_n\in\mathbb{R}^n$ such that $\sum_{i=1}^{n}\alpha_i=0$.

[2] Where, $x\oplus_{\log}y=\log e^x+e^y$

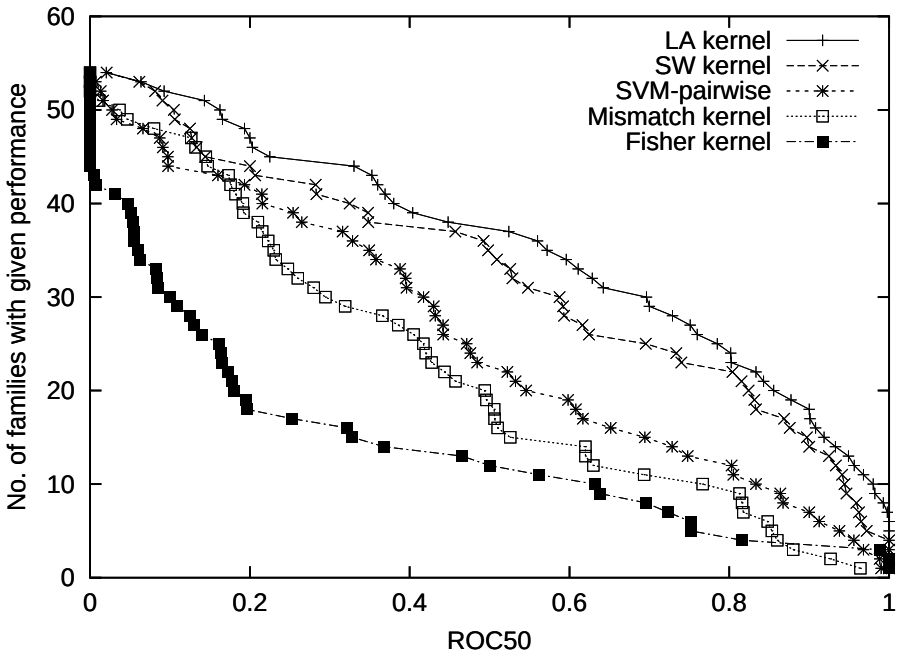


Figure 2.4: Comparison of different methods for a benchmark experiment of remote homology detection (from (Saigo et al., 2004)). Each curve corresponds to a different method, including SVM with a Fisher kernel (Jaakkola et al., 2000), SVM-pairwise (Liao and Noble, 2002), and SVM with a mismatch kernel (Leslie et al., 2003), and the proposed LA kernel with β set to 0.5 and $+\infty$. The curve shows how many protein domain families, from a selection of 53 families in this benchmark, can be assigned to their correct superfamily with a given measure of performance measured by the ROC_{50} index. In short, the higher the curve the better. The proposed methods perform surprisingly well on this benchmark. See (Saigo et al., 2004; Vert et al., 2004a) for more details.

a set of examples be positive definite; this might be done by removing the smallest negative eigenvalue from the diagonal of the matrix, for example.

The method, tested on a benchmark experiment that stimulates the problem of detecting remote homology between protein sequences (Jaakkola et al., 2000), compares surprisingly well with other state-of-the-art SVM-based classifiers (see Figure 2.4) and validates the initial motivation that a careful choice of kernel based on domain-specific knowledge, in our case the use of classical measures of similarities used in the field, can lead to performance improvement for SVM-based classification tasks. Obviously, the state-of-the-art level increases very rapidly and new methods are expected to outperform the results shown in the coming months or so. An important limitation of the proposed kernel is the quadratic cost of dynamic programming, which is likely to be too slow for large-scale real-world applications involving hundreds of thousands of sequences. In the precision/speed balance, local alignment kernels are definitely geared toward higher precision at the expense of speed.

Future work: This work suggests different research opportunities. First the good results obtained on remote homology detection show that the resulting algorithm reaches a state-of-the-art performance, suggesting the investigation of new applications for this algorithm for biological sequence classification. Second, more work needs to be done regarding the choice

of parameters. The fact the similarity score obtained, as opposed to the Smith-Waterman score, is differentiable with respect to the parameters suggests various parameter optimization procedures for any given application. Finally the experiments confirm that the condition of positive definiteness, although required in theory, are not necessary to obtain good results. A theoretical framework to justify this has to be built.

2.3.3 Mutual information kernels and string compression

With M. Cuturi, who started his PhD at the Ecole des Mines in November, 2002, we started investigating a different direction to kernel design for strings. Our initial motivation was to develop fast kernels, i.e., with a linear complexity with respect to the total length of the sequences involved, and to investigate possible connections between string kernels and source coding used for text compression. Roughly speaking, if a compression algorithm can compress a sequence $\mathbf{x}$ in $l(\mathbf{x})$ bits, we wanted to study under which conditions the length of the code of the concatenation of two strings $l(\mathbf{x}\mathbf{x}')$ can be used as (the opposite of) a p.d. kernel for strings.

The rational behind this approach is that many compression algorithms are efficient for sequences with repeated patterns or homogeneities. In other words, if $\mathbf{x}'$ is "similar" to $\mathbf{x}$ according to some algorithm-specific criterion, then both sequences are better compressed taken together – e.g., concatenated – than taken apart : $l(\mathbf{x}\mathbf{x}') \leq l(\mathbf{x}) + l(\mathbf{x}')$. Hence, if an algorithm is known to be efficient to compress particular strings (e.g., biological sequences, natural language texts, financial time series...), then it might make sense to assess the similarity between two strings by the length of the code of the two strings compressed together, because this would mean that the algorithm has detected common features between the two strings that are known to be relevant in the field concerned.

Most compression algorithms, however, do not easily lead to p.d. kernels. We focused on a particular class of algorithm that first define a coding probability P_c on the set of strings, and then use arithmetic coding to form a code that compresses any sequence $\mathbf{x}$ into approximately $\log P_c(\mathbf{x})$ bits (Cover and Thomas, 1990). Among this class of algorithms, a particular subclass defines the coding probability as a mixture probability:

$$P_c(\mathbf{x}) = \int P_\theta(\mathbf{x})\, w(d\theta),$$

where $\{P_\theta : \theta \in \Theta\}$ is a prior family of probability distributions and w is a prior probability on the measurable set Θ. Such algorithms include for example the Laplace or Krichesky-Trofimov coding probabilities (Krichevsky and Trofimov, 1981), as well as the more complex context-tree weighting (CTW) algorithms (Willems et al., 1995).

For such algorithms, one can easily define a p.d. kernel by the formula:

$$K_c(\mathbf{x}, \mathbf{x}') = \int P_\theta(\mathbf{x})\, P_\theta(\mathbf{x}')\, w(d\theta). \tag{2.8}$$

This obviously defines a p.d. kernel, as soon the function $f_{\mathbf{x}} : \Theta \to \mathbb{R}$ defined by:

$$\forall \theta \in \Theta, \quad f_{\mathbf{x}}(\theta) = P_\theta(\mathbf{x})$$

belongs to $L_2(\Theta, w()d\theta)$. It corresponds to the inner product in this space, and the features representing a sequence $\mathbf{x}$ are therefore exactly $(P_\theta(\mathbf{x}), \theta \in \Theta)$. This provides an intuitive

interpretation of how the kernel behaves : *the kernel between two strings is large when both strings have simultaneously large probabilities under models with large a priori probabilities.* We note that independently (Seeger, 2002) introduced the similar concept of covariance kernels. Hence potentially interesting kernels can be formed if 1) domain-specific relevant models P_θ are used, and 2) the mixture probability (2.8) can be efficiently implemented.

In (Cuturi and Vert, 2004) we present a particular case of such a kernel that solves both issues simultaneously. Based on previous evidences that variable-length Markov models for strings are relevant models for protein superfamilies (Bejerano and Yona, 1999; Eskin et al., 2000; Bejerano and Yona, 2001) and that mixture of Dirichlet distributions with domain-specific optimized parameters are relevant prior probabilities amino-acid distributions (Brown et al., 1993), we define a kernel for strings that involves a 3-stage mixture over suffix tree models, Dirichlet priors and multinomial parameters. More precisely, we first define a set of variable-length Markov probability distributions $P_{\mathcal{D},\beta,\theta}$ over strings, where $\mathcal{D}$ is a complete suffix tree that defines the set of suffixes to use in the variable-length Markov model, β is a discrete variable that specifies the index of the Dirichlet component attached to each leaf of the suffix tree – i.e., an integer between 1 and m is specified on each leaf, where m is the number of components in the Dirichlet mixture considered –, and θ is a set of multinomial parameters, one for each leaf of the suffix tree, that defines the conditional distribution of the next letter given a particular suffix. We then define a prior probability π over the parameters that factorizes as:

$$\pi\left(\mathcal{D},\beta,\theta\right)=\pi\left(\mathcal{D}\right)\pi\left(\beta|\mathcal{D}\right)\pi\left(\theta|\beta\right)$$

where $\pi\left(\mathcal{D}\right)$ is a prior probability on suffix trees, typically the distribution of a Galton-Watson branching process; $\pi\left(\beta|\mathcal{D}\right)$ is the distribution of the Dirichlet mixture used at each leaf of the tree $\mathcal{D}$, and typically decomposes as a product of independent multinomials on each leaf; $\pi\left(\theta|\beta\right)$ is a conditional distribution that typically decomposes as a product of Dirichlet distributions with parameters β. The resulting kernel is obtained by a 3-stage mixture over suffix trees, Dirichlet mixture components and multinomial parameters as follows

$$K_c\left(\mathbf{x},\mathbf{x}'\right)=\sum_{\mathcal{D}}\pi\left(\mathcal{D}\right)\left\{\sum_{\beta}\pi\left(\beta|\mathcal{D}\right)\left[\int_\theta P_{\mathcal{D},\beta,\theta}\left(\mathbf{x}\right)P_{\mathcal{D},\beta,\theta}\left(\mathbf{x}'\right)\pi\left(d\theta|\beta\right)\right]\right\}.$$

In spite of the apparently complex mixture to perform, this kernel can be computed exactly with a linear complexity with respect to the length of the sequences. This results from the extension of the CTW algorithm (Willems et al., 1995), that performs a double mixture over trees and multinomial parameters in linear time, to also include a step that performs the average over Dirichlet mixture components, thus resulting in a linear-time triple-mixture algorithm. As a by-product, this algorithm can also be used as an extension of the CTW algorithm for compression purpose.

When tested on a benchmark experiment on remote protein homology detection, the performance of this kernel was a bit disappointing – typically of the order the mismatch kernel (Leslie et al., 2003), that is faster to run (both kernels are linear with respect to the total sequence length, but the CTW kernel requires more operations). Some reasons for the difficulty to outperform the mismatch kernel are discussed in (Cuturi and Vert, 2004).

Future work: the main contribution of this research, to our opinion, is to open new connections between machine learning and information theory, with important practical implications. This suggests that other compression algorithms may be used as kernel, and that

more theoretical results on kernel representation may be obtained from information theory. For example, information theory is likely to offer useful concepts to study the problem of diagonal dominance for string kernels, highlighting the trade-off between having a kernel that can "sufficiently" discriminate any pair of sequence, and having a kernel with large enough values.

2.3.4 Semigroup kernels for finite sets

Following our investigations on the link between compression algorithms and string kernels, we extended the approach to a more abstract and general setting that goes beyond string kernels. This research was mainly performed by M. Cuturi, under my supervision. The initial motivation was that the CTW kernel operates on a "bag-of-words"[3] representation of sequences, because the probability of a sequence under a Markov model of order less than $d-1$ is only based on the counts of d-grams. The operation of concatenation of sequences in fact corresponds to the union of bag-of-words, and the CTW kernel has therefore the following property:

$$K_{CTW}\left(\mathbf{x},\mathbf{x}'\right) = f\left(B\left(\mathbf{x}\right) \cup B\left(\mathbf{x}'\right)\right),$$

where $B\left(\mathbf{x}\right)$ represents the bag-of-word representation of $\mathbf{x}$, and $\cup$ represents the union with repetition.

This suggests to consider the more general setting where $\mathbf{x}$ is a finite set of points of a set $\mathcal{U}$, with possible repetitions. Equivalently $\mathbf{x}$ can be defined as a finite sum of Dirac measures:

$$\mathbf{x} = \sum_{i=1}^{n} \alpha_i \delta_{u_i},$$

where $u_1, u_2, \ldots, u_n \in \mathcal{U}^n$. In particular, in this representation, $\mathbf{x}$ is a finite Radon measure on the set $\mathcal{U}$, and the operation of "bag-of-word unions with repetition" of two objects simply boils down to the addition of measures. As a second extension to this framework, we therefore investigated the following problems: if $\mathcal{U}$ is a Hausdorff space, and $\mathcal{X} = \mathcal{M}_+^b\left(\mathcal{U}\right)$ is the set of finite Radon measures on $\mathcal{U}$ (Berg et al., 1984), under which condition on $f : \mathcal{X} \to \mathbb{R}$ is the following a valid p.d. kernel:

$$K_f\left(\mathbf{x},\mathbf{x}'\right) = f\left(\mathbf{x}+\mathbf{x}'\right).$$

Stated like this, the general theory of harmonic analysis on semigroups can be applied to the semigroup $(\mathcal{X},+)$ endowed with the identity involution (Berg et al., 1984) to prove the following answer to this question, at least for continuous functions f (Cuturi and Vert, 2005)

Theorem 5 *Let f be a continuous function on $\mathcal{M}_+^b\left(\mathcal{U}\right)$ endowed with the weak topology. Then the function $K_f : \mathcal{X}^2 \to \mathbb{R}$ defined by:*

$$K_f\left(\mathbf{x},\mathbf{x}'\right) = f\left(\mathbf{x}+\mathbf{x}'\right)$$

is a p.d. kernel if and only if f has an integral representation of the form:

$$f\left(\mathbf{x}\right) = \int_{C\left(\mathbb{R}^{\mathcal{U}}\right)} e^{\mathbf{x}[h]} d\nu(h)$$

[3] Or more precisely a "bag-of-n-grams" representation, that is, the unordered list of all n-grams of the sequence (with their number of occurrence)

where ν is a uniquely determined positive radon measure on $C\left(\mathbb{R}^{\mathcal{U}}\right)$, the space of continuous functions of $\mathbb{R}^{\mathcal{U}}$, endowed with the topology of pointwise convergence.

For different choices of ν, this theorem results in various p.d. kernel between finite Radon measures, that generalize in particular the CTW kernel. As an example, we present in (Cuturi and Vert, 2005) (and in a forthcoming publication) three particular semigroup kernels for finite Radon measures, or simply sets of points.

1. Let us consider the set of probability measures on a set $\mathcal{U}$, absolutely continuous with respect to a reference measure. In this context, let us consider $\mathcal{X}$ to be the set of densities with finite entropy:

$$h\left(\mathbf{x}\right) = -\int_{\mathcal{U}} \mathbf{x} \ln \mathbf{x}.$$

Then the following holds:

Theorem 6 *The function*

$$K\left(\mathbf{x}, \mathbf{x}'\right) = -h\left(\frac{\mathbf{x}+\mathbf{x}'}{2}\right)$$

is conditionally positive definite, making the function

$$K_{ent}\left(\mathbf{x}, \mathbf{x}'\right) = e^{-\beta h\left(\frac{\mathbf{x}+\mathbf{x}'}{2}\right)}$$

a p.d. kernel for any $\beta > 0$, called the entropy kernel.

The entropy kernel has the following intuitive meaning: if two distribution are similar to each other, then their average will be less scattered and thus have a lower entropy than if they are very different and do not put large probabilities on the same points.

2. Let $\mathcal{U} = \mathbb{R}^d$, $\mathcal{X}$ be the set of finite sets of points of $\mathbb{R}^n$, $(\mu_{\mathbf{x}}, \Sigma_{\mathbf{x}})$ be the parameters (mean and covariance matrix) of the maximum likelihood Gaussian distribution that fits the points of $\mathbf{x}$.

Theorem 7 *For any two sets of points $\mathbf{x}$ and $\mathbf{x}'$, the function:*

$$K\left(\mathbf{x}, \mathbf{x}'\right) = \frac{1}{|\,\Sigma_{\mathbf{x}+\mathbf{x}'}\,|}$$

is a p.d. kernel.

Once again the intuition is very clear: given two clouds of points, they are considered "similar" if, when taken together, one can adjust on them a Gaussian distribution as small as possible, as measured by the determinant of the covariance matrix. We note that the choice of Gaussian distribution can be generalized to any exponential model, and that the final kernel involves in general the entropy of the maximum likelihood estimate from the model.

3. The previous case can be considerably generalized using the kernel trick on the space $\mathcal{U}$ itself

Theorem 8 *Let us suppose that $\mathcal{U}$ is a measurable set endowed with a p.d. kernel K_u. For any set of points $\mathbf{x}$, let $K_{\mathbf{x}}$ be the kernel Gram matrix, that is, the square matrix of pairwise kernel evaluation. Then the function:*

$$K\left(\mathbf{x},\mathbf{x}'\right) = \frac{1}{\mid K_{\mathbf{x}+\mathbf{x}'} + \lambda I \mid}$$

is a p.d. kernel, for any $\lambda > 0$.

We can observe that this kernel formulation depends on the eigenvalues of the kernel matrix, which highlights a strong link with between this kernel and the kernel-PCA decomposition (Schölkopf et al., 1999). The interpretation of this kernel is summarized on Figure 2.5

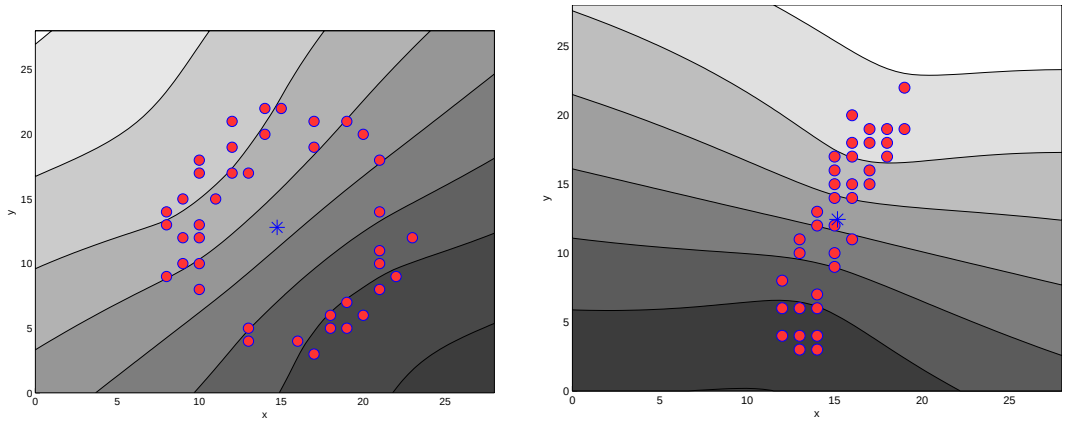
Future work: This work has shown how an algebric structure can translate into a kernel. Similar approaches may be investigated in other applications where invariant and algebric structures are present, such as the comparison of 3-dimensional structures of molecules.

2.3.5 Graph kernels for chemo-informatics

In parallel to the recent explosion in the amount of data about genes and proteins, an increasing attention is paid to data about smaller organic molecules and their interactions with DNA and proteins. As an example, the traditional field of glycobiology is giving rise to the fast-developing field of glyco-bioinformatics, focusing of the understanding of different properties of sugars with long chains (Cooper et al., 2001). More classically, drug discovery in pharmaceutical companies involves increasingly the use of large databases of chemical compounds (typically of the order of 1 millions), and high-throughput screening technologies can now generate vast amounts of data about different properties of these molecules. Moreover, integrated analysis of genomic and chemical information is expected to be a further step towards better understanding of life at the molecular level in the coming years.

Through a collaboration with T. Akutsu, J.-L. Perret and N. Ueda at Kyoto University, we began with P. Mahé, who was starting his PhD under my supervision, to investigate the possibility to use kernel methods for small molecules, i.e., molecules with up to a few tens of atoms. Focusing on the planar (2D) description of molecules, the problem was then to develop kernels for labeled graph, vertices being atoms and edges being covalent bonds. Kashima et al. (2003, 2004) had just proposed such a kernel, called "marginalized graph kernel" because it belongs to the class of kernels obtained through marginalization of a variable (Tsuda et al., 2002). This kernel is defined as follows. To each graph $\mathbf{x}$ is first associated a probability distribution $p_{\mathbf{x}}$ on the set $\mathcal{S}$ of finite-length sequences $s = v_1 e_1 \dots e_{n-1} v_n$ that alternate a vertex label with a bond label. In the case of molecules, such a sequence might for example be "`C = C - C - O - N`". This probability distribution is the image of a first-order Markov random walk on the graph, killed after each step with some probability, by the operation of taking the sequence of labels of each walk on the graph. In other words, the probability of a sequence of labels s is the probability that the random walk follows a path with exactly this sequence of labels. The kernel between two graphs $\mathbf{x}$ and $\mathbf{x}'$ is then defined as a kernel between the corresponding distribution $p_{\mathbf{x}}$ and $p_{\mathbf{x}'}$:

$$K\left(\mathbf{x},\mathbf{x}'\right) = \sum_{s\in\mathcal{S}} p_{\mathbf{x}}\left(s\right) p_{\mathbf{x}'}\left(s\right).$$



(a) Data sampled from a '0'

(b) Data sampled from a '1

(c) Data superposed

Figure 2.5: In order to compare two sets of points, such as pixels on two images, the sets of points are superimposed and the eigenvalues of kernel-PCA are computed. The grey levels in these pictures represent the level sets of the first kernel principal components. If both sets of points superimpose well, then the eigenvalues will tend to be smaller than if they do not. This property can be used to define a semigroup kernel on the sets of points (from (Cuturi and Vert, 2005))

This kernel is parametrized by the parameters of the random walk: distribution of the initial vertex (typically uniform over vertices), transition probabilities between each node and its neighbors (typically uniform among the neighbors), and death probability (typically constant at each step). An interesting property of this kernel is that it can be computed in polynomial time with respect to the product of the numbers of atoms in each graph to be compared[4].

In (Mahé et al., 2004) we propose two extensions to this graph kernel. First the alphabet of the vertices labels is extended, typically to augment the original vertex label – such as the type of an atom – with contextual informations – such as the degree of covalence, or the type of neighborhood atoms. Although various choices of label enrichment are possible, we test the use of an index known in chemoinformatics as the Morgan index, that roughly characterizes the number of atoms in a given range of each atom. Label enrichment has two important consequences. First, as the label specificity increases, the number of common paths between two graphs automatically decreases. This not only mean that the graphs becomes increasingly orthogonal, but also that the kernel becomes increasingly fast to compute – the complexity of the kernel computation depends on the number of matches between the vertices of both graphs. Second, as the environment of each atom is taken into account to build the labels, the feature space of the kernel increases and more relevant features for a given classification problem than only sequences of atoms are likely to appear. There is obviously a trade-off to be found between the computation time and a variety of features encoded in the kernel, that favor large label alphabets, and statistical learning, that favors small alphabets which result in large amount of features shared between different molecules. Tested on a benchmark experiment of binary molecule classification (mutagenicity or not), we indeed found an optimal alphabet size that results in a small performance improvement, while the computation time is reduced by two orders of magnitude.

The second extension aims at removing paths that "totter" from the random walk model, i.e., paths that go from a node A to a node B, and then to A again. We hypothesized that such path might generate misleading features for the graph kernel. For example, the feature "`C - C - C`" might correspond to the presence of 3 carbon atoms in a row, or simply to 2 carbon atoms. By preventing tottering paths in the random walk model, we therefore expected to "improve" the chemical relevance of the features. A possibility to prevent totters is to modify the random walk model, and make it second-order Markov. In order to implement it we showed that the probability distribution of the paths label sequences under such a second-order Markov model can be obtained from a first-order Markov model on a bigger graph (see Figure 2.6), just like, for discrete time-series, a d-th-order Markov model reduces to a first-order Markov model on an augmented variable. This modification, of course, has the drawback to increase the computation time of the kernel. Tested on two benchmark experiments of molecule classification, this expensive extension to the original graph kernel led to no significant performance improvement, for a longer computation time.

Future work: We are currently validating the method on larger chemical datasets, for various problems of drug-likeness and activity prediction, which are among the dominant applications in chemical dataset analysis for drug discovery. From this validation phase, new developments will be planed. In particular, parameter optimization and feature selection might be relevant in this case.

[4]More precisely it involves the inversion of a sparse $|\mathbf{x}| \times |\mathbf{x}|$ matrix.

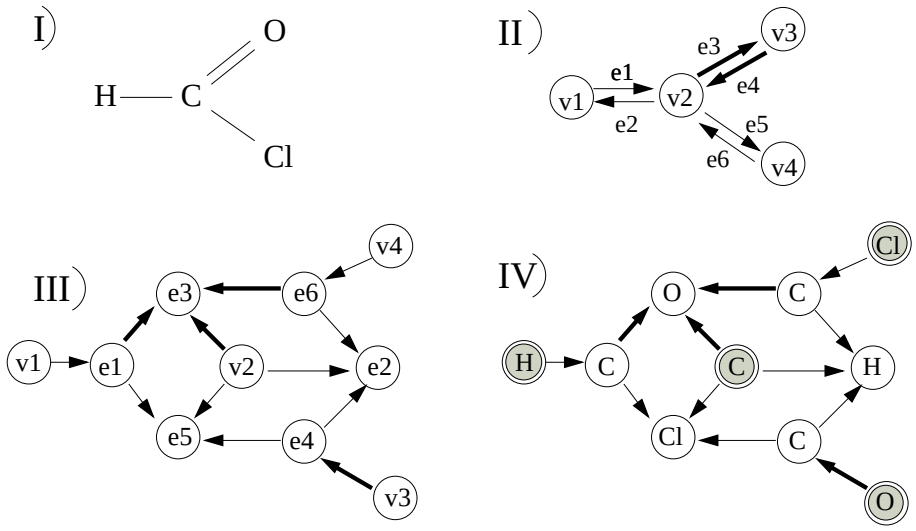


Figure 2.6: A graph transformation that transforms non-tottering second-order Markov random walks into first-order Markov random walk, enabling the fast computation of the modified kernel (from (Mahé et al., 2004)).

2.4 Kernels in systems biology

The term "systems biology" refers here to the fast-growing discipline that consists in studying biology at the level of systems of basic entities that interact together. Roughly speaking, this encompasses the reconstruction of biological network – such as regulatory, interaction or metabolic networks –, the analysis of their properties, the study of their evolution, and the prediction of their responses to external action. In the kernel approach, a set of points, e.g., a set of genes, is represented by a set of points in a Hilbert space. This defines an elementary system, where "interactions" between points basically refers to their relative positions in the Hilbert space, and where the naivety of the approach is partly compensated by the power of kernel methods to provide useful methods. We mention below two contributions we did in this field.

2.4.1 Supervised graph inference

The problem of graph inference, or graph reconstruction, is to predict the presence or absence of edges between a set of points known to form the vertices of a graph, the prediction being based on observations about the points. This problem has recently drawn a lot of attention in computational biology, where the reconstruction of various biological networks, such as gene or molecular networks from genomic data, is a core prerequisite to the recent field of systems biology that aims at investigating the structures and properties of such networks. As an example, the *in silico* reconstruction of protein interaction networks (Jansen et al., 2003), gene regulatory networks (Friedman et al., 2000) or metabolic networks (Kanehisa, 2001) from large-scale data generated by high-throughput technologies, including genome sequencing or microarrays, is one of the main challenges of current systems biology.

Various approaches have been proposed to solve the network inference problem. Bayesian

(Friedman et al., 2000) or Petri networks (Doi et al., 2000) are popular frameworks to model the gene regulatory or the metabolic network, and include methods to infer the network from data such as gene expression of metabolite concentrations (Friedman et al., 2000). In other cases, such as inferring protein interactions from gene sequences or gene expression, these models are less relevant and more direct approaches involving the prediction of edges between "similar" nodes have been tested (Marcotte et al., 1999; Pazos and Valencia, 2001).

In collaboration with Y. Yamanishi, currently PhD student at Kyoto University, we investigated a radically different approach for the problem of graph inference, and proposed two related algorithms based on kernel methods that gave very promising results on the problem of reconstructing the metabolic network of a simple organism (Yamanishi et al., 2004b; Vert and Yamanishi, 2005). Our key contributions are 1) the observation that most problems of network inference in computational biology can in fact be considered as supervised learning problem, and 2) a general methodology to solve the problem of supervised network inference, resulting in two algorithms.

The first point is based on the observation that many problems of interest in systems biology concern the completion of a network that is partially known. For example, a significant number of classical metabolic pathways can be reconstructed, together with the enzymes, on most newly sequenced organisms, and a challenging problem is to find new pathways or enzymes (i.e., genes) missing is the known pathways. Similarly, while limited parts of gene regulatory networks and protein interaction networks are known with high evidence from experimental validation, the challenge is to extend this limited networks to a the whole-genome or proteome scale.

This observation suggests to formalize this problem as a supervised learning problem: given a limited network of genes, together with data about each gene, how to extend this network to new genes, for which only data are available? In order to solve such problems, we propose a general two-step strategy:

1. first map the data points (graph vertices) to a Euclidean space with a mapping $\Phi : \mathcal{X} \to \mathbb{R}^d$;

2. then connect any pair of points $(\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2$ with an edge whenever the Euclidean distance $\| \Phi(\mathbf{x}) - \Phi(\mathbf{x}') \|$ is smaller than a given or estimated threshold δ.

Within this strategy we proposed to restrict the supervised learning problem to the selection of the mapping Φ in the first step. This amounts to solving the following problem: given a set of points in $\mathcal{X}$ together with edges between the points, how to find a mapping from $\mathcal{X}$ to a Euclidean (or Hilbert) space such that the edges link points close to each other in this space? Following a classical approach in statistics and machine learning, we define an empirical loss function $R_n(\Phi)$ that quantifies how well a candidate mapping Φ fulfills this goal, and a regularizer $\Omega(\phi)$ that measures the complexity of the mapping Φ; the mapping learned from the known network is then the solution of:

$$\min_{\Phi} \{R_n(\Phi) + \lambda \Omega(\phi)\},$$

where λ is the parameter that controls the trade-off between fitting the known network and finding a low complexity mapping, expected to allow the reconstruction of edges with new vertices. As an example, we take in (Vert and Yamanishi, 2005) each dimension of Φ to be an

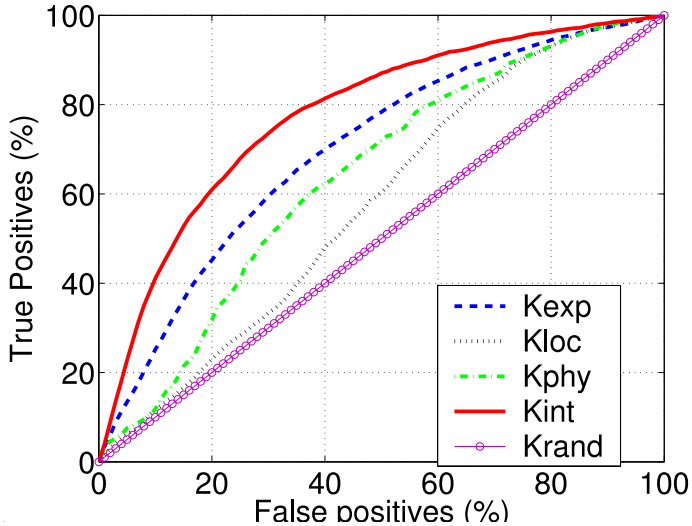


Figure 2.7: Performance of the network inference algorithm with different genomic kernels. Each curve plots the performance of one particular kernel, obtained from gene expression data (Kexp), protein localization (Kloc), gene phylogenetic profiles (Kphy), data integration (Kint), and random kernel (Krand). The performance is measured by the number of correctly predicted edges as a function of wrongly predicted edges, for different thresholds in the algorithm. We observe that the data integration through kernels improves the performance over each data taken separately. From (Vert and Yamanishi, 2005).

element of the RKHS associated with a kernel on $\mathcal{X}$, the regularizer to be the square norm in the RKHS, and the loss function to be quadratic:

$$R_n(\Phi) = \sum_{i \sim j} \left[\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)\right]^2 .$$

The optimization is performed iteratively for each dimension of Φ, under the additional constraints that each dimension be orthogonal to all previous dimensions and have unit norm in the RKHS. Other choices for $R_n(\Phi)$ and $\Omega(\Phi)$ are obviously possible, and we are currently investigating several alternative both in theory and in practice.

Thus stated this method can be applied on any space $\mathcal{X}$ endowed with a p.d. kernel. Given the increasing number of such kernels for genes, this provides a principled approach to network learning from several heterogeneous data simultaneously, by first integrating the different kernels through operations in the space of p.d. kernels (such as addition, for example), and then applying the supervised network inference with the resulting kernel. We tested in (Vert and Yamanishi, 2005) the possibility to infer a particular network, the metabolic gene network, from various genomic data including gene expression data, phylogenetic profiles, protein localization in the cell and predicted protein-protein interactions from large-scale high-throughput experiments. We observed that the supervised approach clearly outperforms the unsupervised approach, and that the prediction using the integration of all data outperforms the prediction using each data independently (see Figure 2.7).

Future work. I plan to investigate this topic further in the future for two reasons: first because it deals with a major problem in current systems biology – how to infer a network from

noisy and heterogeneous data –, and second because the approach we proposed are clearly in their infancy. No theoretical work has been done about the statistical issue of estimating a network. The choice of loss function and regularizer we did was largely arbitrary, and should be compared both in theory and in practice with other choices, that would lead to different optimization problems. The validation on real-world data including prediction of protein-protein interaction, metabolic and regulatory pathways, will be a priority in the near future. Finally, many connections exist between the formulation we proposed and other problems in machine learning, including distance learning and data-driven regularization (Sindhwani et al., 2004), for which a unifying point of view remains to be clarified.

2.4.2 Graph-driven feature extraction

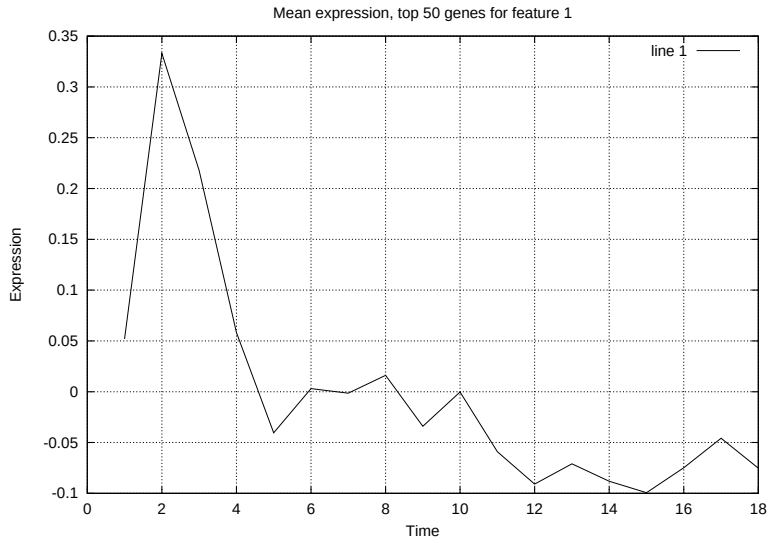
A by-product of the supervised network inference method summarized in the previous section is the mapping Φ that maps the original data to a Euclidean space where the structure of the graph (presence of absence of edges) roughly corresponds to similarity between points in terms of Euclidean distance. Φ can therefore be seen as a regularized embedding of the training graph into a Euclidean space. This embedding can be given a particular interpretation in the case of gene network, that we explored in (Vert and Kanehisa, 2003b,a) and summarize now.

Consider the simple case when genes are characterized by expression profiles, i.e., to each gene is associated a series of numbers that corresponds to its level of expression in different experimental conditions. In this case, we take $\mathcal{X} = \mathbb{R}^d$, where d is the number of experimental conditions. Let now (V, E) be a known graph of genes, such as a physical interaction network or the metabolic gene network, and consider the linear p.d. kernel on $\mathcal{X}$: $K(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$. Each element of the RKHS is then a linear function, of the form $f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}$. By construction of the linear embedding Φ of the graph (V, E) into a Euclidean space, connected vertices should be close to each others in the Euclidean space. Another way to express this is to say that each coordinate of Φ, as a function from $\mathcal{X}$ to $\mathbb{R}$, should vary smoothly on the graph: connected vertices should have similar values.

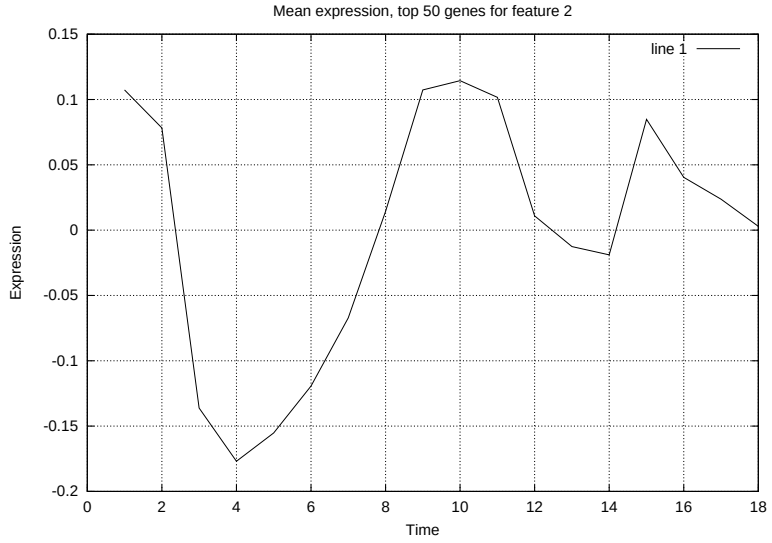
In other words, this linear embedding is an automatic way to extract vectors $\mathbf{w}_1, \mathbf{w}_2, \ldots$, such that the functions $f_i(\mathbf{x}) = \mathbf{w}_i \cdot \mathbf{x}$ vary as smoothly as possible on the network, while remaining of low complexity. With the linear kernel, the low complexity simply means that f_i should capture enough of the variation in the data, the first principal component being therefore the function with smallest complexity. These smoothly varying features are of particular importance to interpret experimental measurements, such as gene expression data, in terms of prior knowledge such as metabolic networks. Indeed the extracted profiles $\mathbf{w}_i$ correspond to particular weightings of the different experiment, such that the corresponding features f_i exhibit a particular coherence with the network. Here the coherence is expressed in terms of smoothness, which typically means that the feature is likely to have clusters of rather positive or negative values on the graph. These clusters can in turn be interpreted as a set of genes working together, either because they are involved in the same metabolic pathway (when the graph is the metabolic network), or because they form complexes (when the graph is a pairwise interaction graph). One might therefore associate functions to these features.

Examples on real-world data were proposed in (Vert and Kanehisa, 2003a), using a slightly different method. In order to interpret gene expression data, we applied the method described earlier and observed the features extracted (Figure 2.8), together with the clusters in the network where the features are particularly positive or negative (Table 2.1).

In this particular example, a strong experimental bias, never reported before to our knowl-



(a) First profile



(b) Second profile

Figure 2.8: First 2 profiles extracted (α factor data set, from (Vert and Kanehisa, 2003a))

Feature	Correlation	Main pathways and genes
1	+	Glycolysis / Gluconeogenesis (PGK1, GPM2, ALD4,6), TCA cycle (CIT2, MDH1,2, SDH1, LSC1), Pentose phosphate pathway (RBK1, SOL4, ZWF1, YGR043C), Glycerolipid metabolism (GPD1,2,3, ALD4,6), Glyoxylate and dicarboxylate metabolism (MDH1,2, CIT2, ICL2), Sulfur metabolism (MET2,14,16,17).
1	-	Pyrimidine metabolism (RPA12,34,49,190, RPB2,5, RPC53, DUT1, TRR1, POL5, URK1, MIP1, PUS1), Purine metabolism (RPA12,34,49,190, RPB2,5, RPC53, CDC19, APT2, POL5, MIP1), Aminoacyl-tRNA biosynthesis (ILS1, FRS2, MES1, YHR020W, GLN4, ALA1, CDC60), Starch and sucrose metabolism (MPS1, HPR5, SWE1, HSL1, EXG1).
2	+	Pyrimidine metabolism (DEG1, PUS1,3,4, URA1,2, CPA1,2,FCY1), Folate biosynthesis (ENA1,5, BRR2, HPR5, FOL1), Starch and sucrose metabolism (ENA1,5, BRR2, HPR5, PGU1), Phenylalanine, tyrosine and tryptophan biosynthesis (TRP2,3,4, ARO2,7), Sterol biosynthesis (ERG7,12, HGM1,2).
2	-	Starch and sucrose metabolism (CDC7, ENA1, GIN4, HXK2, HPR5, SWE1, UGP1, HSL1, FKS1, MEK1), Purine and pyrimidine metabolism (POL12, ADK2, DUT1, RNR2, HYS2, YNK1, CDC21), Fructose and mannose metabolism (MNN1, PMI40, SEC53, HXK2), Cell cycle (CDC7, GIN4, SWE1, HSL1).

Table 2.1: Pathways and genes with highest and lowest scores on the first 2 features extracted. The first profile corresponds to an experimental bias: energy-producing pathways are strongly activated following the beginning of the experiment, which most pathways involved in the synthesis of macromolecules are momentarily inhibited. The second profile corresponds to the cell cycle, which was the motivation for this studies (from (Vert and Kanehisa, 2003a)).

edge, was easily detected in widely-used yeast cell cycle data (Spellman et al., 1998): the goal of the experiment being to study genes with a cyclic regulation in the cell cycle, a colony of yeast had to be synchronized at a given point of the cell cycle, and this synchronization plays a major role in the gene expression data easily described in terms of metabolic pathways.

Future work: this original approach gave promising results on widely-used public data, and partially answers a major question in today's biomedical research: how to make sense out of gene expression data, particularly in terms of metabolic pathways? We are currently testing this method on two more challenging real-world issues: the first one, through a collaboration with P. David and J.Y. Coppée's group at the Pasteur Institute, concerns the analysis of the metabolism of *Plasmodium falciparum*, the agent responsible for malaria, and how it reacts to various drugs; the second one, through a collaboration with E. Barillot and F. Radvanyi's groups, at the Curie Institute, to characterize progresses in tumor progression in cancer research.

2.5 Conclusion and perspective

My research in the last three years has focused on the use of p.d. kernels to represent heterogeneous data in computational biology, and perform various inference tasks, including gene supervised classification and gene network inference. Several new ideas and methods were introduced, which have certainly not been fully exploited neither in theory nor practice, and which constitute future work both for me and for my collaborators, including the PhD students I have the honors tu supervise.

My first focus will therefore be, both in theory and in practise, to continue the development of the approaches we pioneered for different applications. Several collaborations have been set up in the last two years, in particular to secure possibilities of real-world application and experimental validation of diverse predictions:

- with the Pasteur Institute (in particular P. David, and J.-Y. Coppée's group) to perform functional analysis and metabolic pathway monitoring from gene expression data on *P. falciparum*, the agent of malaria. The ultimate goal of this research is to develop new drugs against malaria;
- with the Curie Institute (in particular F. Radvanyi and E. Barillot's group), to characterize cancer tumors by integrating gene expression data, CGH data and gene regulatory networks;
- with the biotechnological and pharmaceutical industry to validate the new methods on virtual screening of chemical compounds in drug design;
- with Kyoto University (in particular T. Akutsu and M. Kanehisa's groups) to propose new methods for the analysis of metabolic pathways stored in the KEGG database.
- with University of Washington (B. Noble and D. Baker), UC Berkeley (M. Jordan and L. El Ghaoui) and UC Davis (N. Cristianini), in order to develop and validate on yeast genes various kernel-based gene function and protein interaction prediction methods.

A second research direction to be followed in parallel concerns further developments in the problem of learning from data, both in theory and in practice. Our recent work on semi-supervised learning (Hue, 2004) and active learning (Abernethy et al., 2004) convinced me of

the practical performance and theoretical lack of understandings of these fields, which I would like to be able to devote time on.

Finally, my long-term objective remains to develop rigorous theoretical frameworks and useful algorithms for post-genomic biology. I expect this goal to be broad and ambitious enough to motivate my and other's research in the long term.

Bibliography

J. Abernethy, T. Evgeniou, and J.-P. Vert. An optimization framework for adaptive conjoint questionnaire design. Technical report, INSEAD, 2004.

R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422(6928):198–207, Mar 2003.

S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. Gapped blast and psi-blast: A new generation of protein database search programs. *Nucleic Acids Research*, 25:3389–3402, 1997.

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

G. Bejerano and G. Yona. Modeling protein families using probabilistic suffix trees. In *Proceedings of RECOMB 1999*, pages 15–24. ACM Press, 1999.

G. Bejerano and G. Yona. Variations on probabilistic suffix trees: statistical modeling and prediction of protein families. *Bioinformatics*, 17:23–43, 2001.

C. Berg, J.P.R. Christensen, and P. Ressel. *Harmonic analysis on semigroups*. Springer-Verlag, New-York, 1984.

B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th annual ACM workshop on Computational Learning Theory*, pages 144–152. ACM Press, 1992.

M.P. Brown, R. Hughey, A. Krogh, I.S. Mian, K. Sjolander, and D. Haussler. Using dirichlet mixture priors to derive hidden markov models for protein families. In *Proc. First International Conference on Intelligent Systems for Molecular Biology (ISMB 1993)*, 1993.

P.S. Coelho, A. Kumar, and M. Snyder. Genome-wide mutant collections: toolboxes for functional genomics. *Curr. Opin. Microbiol.*, 3:309–315, 2000.

International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature Nature*, 409(6822):860–921, Feb 2001.

C. Cooper, M. Harrison, M. Wilkins, and N. Packer. Glycosuitedb: a new curated relational database of glycoprotein glycan structures and their biological sources. *Nucleic Acids Res.*, 29:332–335, 2001.

T.M. Cover and J.A. Thomas. *Elements of Information Theory*. John Wiley, New-York, 1990.

N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

M. Cuturi and J.-P. Vert. A mutual information kernel for strings. In *Proceedings of IJCNN 2004*, 2004.

M. Cuturi and J.-P. Vert. Semigroup kernels on finite sets. In *Advances in Neural Information Processing Systems*, volume 17, 2005.

J.L. DeRisi, V.R. Iyer, and P.O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278(5338):680–686, 1997.

A. Doi, H. Matsuno, M. Nagasaki, and S. Miyano. Hybrid petri net representation of gene regulatory network. In *Proceedings of the Pacific Symposium on Biocomputing*, volume 5, pages 341–352, 2000.

R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

E. Eskin, W.N. Grundy, and Y. Singer. Protein family classification using sparse markov transducers. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB 2000)*, pages 134–145, 2000.

N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4):601–620, 2000.

M. Gribskov, R. Lüthy, , and D. Eisenberg. Profile analysis. *Methods in Enzymology*, 183: 146–159, 1990.

D. Haussler. Convolution kernels on discrete structures. Technical report, UC Santa Cruz, 1999.

M. Hue. Semi-supervised learning for protein structure prediction. Master's thesis, Ecole des Mines de Paris, 2004.

T. Jaakkola, M. Diekhans, and D. Haussler. A discriminative framework for detecting remote protein homologies. *Journal of Computational Biology*, 7(1,2):95–114, 2000.

R. Jansen, H. Yu, D. Greenbaum, Y. Kluger, N.J. Krogan, S. Chung, A. Emili, M. Snyder, J.F. Greenblatt, and M. Gerstein. A bayesian networks approach for predicting protein-protein interactions from genomic data. *Science*, 302(5644):449–453, 2003.

R.S. Kamath, A.G. Fraser, Y. Dong, G. Poulin, R. Durbin, M. Gotta, A. Kanapin, N. Le Bot, S. Moreno, M. Sohrmann, D.P. Welchman, P Zipperlen, and J. Ahringer. Systematic functional analysis of the caenorhabditis elegans genome using rnai. *Nature*, 421(6920):231–237, Jan 2003.

M. Kanehisa. Prediction of higher order functional networks from genomic data. *Pharmacogenomics*, 2(4):373–385, 2001.

H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In T. Faucett and N. Mishra, editors, *Proceedings of the Twentieth International Conference on Machine Learning*, pages 321–328. AAAI Press, 2003.

H. Kashima, K. Tsuda, and A. Inokuchi. Kernels for graphs. In B. Schölkopf, K. Tsuda, and J.P. Vert, editors, *Kernel Methods in Computational Biology*, pages 155–170. MIT Press, 2004.

G.S. Kimeldorf and G. Wahba. Some results on tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33:82–95, 1971.

H. Kitano. *Foundations of Systems Biology*. MIT Press, 2001.

R. Krichevsky and V. Trofimov. The performance of universal encoding. *IEEE Trans. Inform. Theory*, 27(2):199–207, Mar 1981.

W. Kuich and A. Salomaa. Semirings, automata, languages. In *EATCS Monographs on Computer Science*, volume 5. Springer-Verlag, 1986.

G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, and M.I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5: 27–72, 2004.

S. Lauritzen. *Graphical Models*. Oxford, 1996.

C. Leslie, E. Eskin, and W.S. Noble. The spectrum kernel: a string kernel for svm protein classification. In Russ B. Altman, A. Keith Dunker, Lawrence Hunter, Kevin Lauerdale, and Teri E. Klein, editors, *Proceedings of the Pacific Symposium on Biocomputing 2002*, pages 564–575. World Scientific, 2002.

C. Leslie, E. Eskin, J. Weston, and W.S. Noble. Mismatch string kernels for svm protein classification. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems 15*. MIT Press, 2003.

Li Liao and William Stafford Noble. Combining pairwise sequence similarity and support vector machines for remote protein homology detection. In *Proceedings of the Sixth International Conference on Computational Molecular Biology*, 2002.

H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. Text classification using string kernels. *Journal of Machine Learning Research*, 2:419–444, 2002.

P. Mahé, N. Ueda, T. Akutsu, J.-L. Perret, and J.-P. Vert. Extensions of marginalized graph kernels. In R. Greiner and D. Schuurmans, editors, *Proceedings of the Twenty-First International Conference on Machine Learning (ICML 2004)*, pages 552–559. ACM Press, 2004.

E.M. Marcotte, M. Pellegrini, H.-L. Ng, D.W. Rice, T.O. Yeates, and D. Eisenberg. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285: 751–753, 1999.

S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.R. Müller. Fisher discriminant analysis with kernels. In Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, editors, *Neural Networks for Signal Processing IX*, pages 41–48. IEEE, 1999.

W. S. Noble. Support vector machine applications in computational biology. In B. Schölkopf, K. Tsuda, and J.P. Vert, editors, *Kernel Methods in Computational Biology*, pages 71–92. MIT Press, 2004.

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*, pages 249–255, 2001.

F. Pazos and A. Valencia. Similarity of phylogenetic trees as indicator of protein-protein interaction. *Protein Engineering*, 9(14):609–614, 2001.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, 1988. The classic original book on belief networks, which was certainly motivated by the idea that belief networks might have relevance to brains.

W.R. Pearson. Rapid and sensitive sequence comparisons with FASTP and FASTA. *Methods in Enzymology*, 183:63–98, 1990.

H. Saigo, J.-P. Vert, N. Ueda, and T. Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(11):1682–1689, 2004.

S. Saitoh. *Theory of reproducing Kernels and its applications*. Longman Scientific & Technical, Harlow, UK, 1988.

M. Schena, D. Shalon, R.W. Davis, and P.O. Brown. Quantitative monitoring of gene expression patterns with a complimentary DNA microarray. *Science*, 270:467–470, 1995.

B. Schölkopf and A.J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, 2002.

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.

B. Schölkopf, K. Tsuda, and J.P. Vert. *Kernel Methods in Computational Biology*. MIT Press, 2004.

B. Schölkopf, J. Weston, E. Eskin, C. Leslie, and W.S. Noble. A kernel approach for learning from almost orthogonal patterns. In *Proceedings of ECML 2002*, 2002.

M. Seeger. Covariance kernels from bayesian generative models. In *Advances in Neural Information Processing Systems*, volume 14, pages 905–912, 2002.

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

V. Sindhwani, P. Niyogi, and M. Belkin. Manifold regularization: A geometric framework for learning from examples. Technical Report TR-2004-06, The University of Chicago, 2004.

T. Smith and M. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

P.T. Spellman, G. Sherlock, M.Q. Zhang, V.R. Iyer, K. Anders, M.B. Eisen, P.O. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Mol. Biol. Cell*, 9:3273–3297, 1998.

K. Tsuda, T. Kin, and K. Asai. Marginalized kernels for biological sequences. *Bioinformatics*, 18:S268–S275, 2002.

V.N. Vapnik. *Statistical Learning Theory*. Wiley, New-York, 1998.

J. Craig et al. Venter. The sequence of the human genome. *Science*, 291(5507):1304–1351, 2001.

J.-P. Vert. Adaptive context trees and text clustering. *IEEE Trans. Inform. Theory*, 47(5): 1884–1901, Jul 2001a.

J.-P. Vert. *Statistical Methods for Natural Language Modelling*. PhD thesis, Paris 6 University, 2001b.

J.-P. Vert. Support vector machine prediction of signal peptide cleavage site using a new class of kernels for strings. In Russ B. Altman, A. Keith Dunker, Lawrence Hunter, Kevin Lauerdale, and Teri E. Klein, editors, *Proceedings of the Pacific Symposium on Biocomputing 2002*, pages 649–660. World Scientific, 2002a.

J.-P. Vert. A tree kernel to analyze phylogenetic profiles. *Bioinformatics*, 18:S276–S284, 2002b.

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 Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems*, pages 1449–1456. MIT Press, 2003b.

J.-P. Vert, H. Saigo, and T. Akutsu. Local alignment kernels for biological sequences. In B. Schölkopf, K. Tsuda, and J.P. Vert, editors, *Kernel Methods in Computational Biology*, pages 131–154. MIT Press, 2004a.

J.-P. Vert, K. Tsuda, and B. Schölkopf. A primer on kernel methods. In B. Schölkopf, K. Tsuda, and J.P. Vert, editors, *Kernel Methods in Computational Biology*, pages 35–70. MIT Press, 2004b.

J.-P. Vert and Y. Yamanishi. Supervised graph inference. In *Advances in Neural Information Processing Systems*, volume 17, 2005.

C. Watkins. Dynamic alignment kernels. In A.J. Smola, P.L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 39–50. MIT Press, Cambridge, MA, 2000.

J.D. Watson and F.H.C. Crick. A structure for deoxyribose nucleic acid. *Nature*, 171:737, 1953.

F.M.J. Willems, Y.M. Shtarkov, and T.J. Tjalkens. The context tree weighting method: Basic properties. *IEEE Trans. Inform. Theory*, 41(3):653–664, May 1995.

C.K.I. Williams. Prediction with gaussian processes: From linear regression to linear prediction and beyond. In M.I. Jordan, editor, *Learning and Inference in Graphical Models*. Kluwer Academic Press, 1998.

Y. Yamanishi, J.-P. Vert, and M. Kanehisa. Heterogeneous data comparison and gene selection with kernel canonical correlation analysis. In B. Schölkopf, K. Tsuda, and J.P. Vert, editors, *Kernel Methods in Computational Biology*, pages 209–230. MIT Press, 2004a.

Y. Yamanishi, J.-P. Vert, and M. Kanehisa. Protein network inference from multiple genomic data: a supervised approach. *Bioinformatics*, 20:i363–i370, 2004b.

J. Zhu and T. Hastie. Kernel logistic regression and the import vector machine. In *Advances in Neural Information Processing Systems*, 2001.

Chapter 3

Selected Publications