UNIVERSITÉ D'ÉVRY VAL D'ESSONNE

ÉCOLE DOCTORALE DES GÉNOMES AUX ORGANISMES

THÈSE

présentée par

Céline BROUARD

pour obtenir le grade de

Docteur en sciences de l'Université d'Évry Val d'Essonne

Spécialité : Bioinformatique, biologie structurale et génomique

---

Inférence de réseaux d'interaction protéine-protéine par apprentissage statistique

Application au réseau d'interaction autour de la protéine CFTR, impliquée dans la mucoviscidose

---

Soutenue le 14 février 2013 devant le jury composé de

Rapporteurs :	François LAVIOLETTE	-	Université Laval
	Jean-Philippe VERT	-	Mines ParisTech
Examinateurs :	Eric ANGEL	-	Université d'Evry
	Franck BROUILLARD	-	Université Paris Descartes
	Christine FROIDEVAUX	-	Université Paris-Sud 11
Directeurs :	Florence D'ALCHÉ-BUC	-	Université d'Evry
	Aleksander EDELMAN	-	Université Paris Descartes

Résumé :

L'objectif de cette thèse est de développer des outils de prédiction d'interactions entre protéines qui puissent être appliqués en particulier chez l'homme, sur les protéines qui constituent un réseau avec la protéine CFTR. Cette protéine, lorsqu'elle est défectueuse, est impliquée dans la mucoviscidose. Le développement de méthodes de prédiction *in silico* peut s'avérer utile pour suggérer aux biologistes de nouvelles cibles d'interaction et pour mieux expliquer les fonctions des protéines présentes dans ce réseau.

Nous proposons une nouvelle méthode pour le problème de la prédiction de liens dans un réseau. Afin de bénéficier de l'information des données non étiquetées, nous nous plaçons dans le cadre de l'apprentissage semi-supervisé. Nous abordons ce problème de prédiction comme une tâche d'apprentissage d'un noyau de sortie, appelée régression à noyau de sortie. Un noyau de sortie est supposé coder les proximités existantes entre les nœuds du graphe et l'objectif est d'approcher ce noyau à partir de descriptions appropriées en entrée. L'utilisation de l'astuce du noyau dans l'ensemble de sortie permet de réduire le problème d'apprentissage à partir de paires à un problème d'apprentissage d'une fonction d'une seule variable à valeurs dans un espace de Hilbert. En choisissant les fonctions candidates pour la régression dans un espace de Hilbert à noyau reproduisant à valeur opérateur, nous développons, comme dans le cas de fonctions à valeurs scalaires, des outils de régularisation. Nous établissons en particulier des théorèmes de représentation dans le cas supervisé et dans le cas semi-supervisé, que nous utilisons ensuite pour définir de nouveaux modèles de régression pour différentes fonctions de coût, appelés IOKR-ridge et IOKR-margin.

Nous avons d'abord testé l'approche développée sur des données artificielles, des problèmes test ainsi que sur un réseau d'interaction protéine-protéine chez la levure *S. Cerevisiae* et obtenu de très bons résultats. Puis nous l'avons appliquée à la prédiction d'interactions entre protéines dans le cas d'un réseau construit autour de la protéine CFTR.

Mots clés : interactions protéine-protéine, prédiction de liens, méthodes à noyaux, apprentissage semi-supervisé, espace de Hilbert à noyaux reproduisants, noyau à valeur opérateur, régression à noyau de sortie, sorties structurées

Abstract :

The aim of this thesis is to develop tools for predicting interactions between proteins that can be applied to the human proteins forming a network with the CFTR protein. This protein, when defective, is involved in cystic fibrosis. The development of *in silico* prediction methods can be useful for biologists to suggest new interaction targets and to better explain the proteins' functions in the network.

We propose a new method to solve the link prediction problem. To benefit from the information of unlabeled data, we place ourselves in the semi-supervised learning framework. Link prediction is addressed as an output kernel learning task, referred as Output Kernel Regression. An output kernel is assumed to encode the proximities of nodes in the target graph and the goal is to approximate this kernel by using appropriate input features. Using the kernel trick in the output space allows one to reduce the problem of learning from pairs to learning a single variable function with output values in a Hilbert space. By choosing candidates for regression functions in a reproducing kernel Hilbert space with operator valued kernels, we develop tools for regularization as for scalar-valued functions. We establish representer theorems in the supervised and semi-supervised cases and use them to define new regression models for different cost functions, called IOKR-ridge and IOKR-margin.

We first tested the developed approach on transductive link prediction using artificial data, benchmark data as well as a protein-protein interaction network of the yeast *S. Cerevisiae* and we obtained very good results. Then we applied it to the prediction of protein interactions in a network built around the CFTR protein.

Keywords : protein-protein interactions, link prediction, kernel methods, semi-supervised learning, reproducing kernel Hilbert space, operator-valued kernel, output kernel regression, structured outputs

Remerciements

Je tiens en premier lieu à remercier Florence pour son encadrement durant ces trois ans de thèse (et même un peu plus avec les stages qui ont précédé). Je te remercie de m'avoir accueilli en stage de dernière année d'école d'ingénieur et de m'avoir fait découvrir le monde de la recherche et le domaine de l'apprentissage statistique. Moi, qui n'avais jamais eu d'idée très précise de ce que je voulais faire plus tard, tu m'as permis de trouver ma voie. Je te remercie pour ta confiance, ta disponibilité et ta bonne humeur. J'ai eu beaucoup de plaisir à travailler avec toi et j'espère avoir le plaisir de continuer à collaborer avec toi dans les années à venir.

Je tiens également à remercier Aleksander pour m'avoir co-encadré. Il n'a pas été facile de se comprendre au début, que ce soit dans un sens ou dans l'autre, mais petit à petit les échanges ont été plus fructueux et j'ai été très motivée de travailler sur ce sujet. Je te remercie en tous cas pour ton enthousiasme sur ce travail. J'en profite pour remercier également Chiara, Serina et Mathieu pour m'avoir accueilli à Necker et m'avoir fait une petite place dans leur bureau lorsque je venais. Chiara, je te remercie également pour tes remarques et tes conseils, toujours très intéressants.

Je remercie Jean-Phillippe Vert et François Laviolette qui m'ont fait l'honneur de rapporter cette thèse. Je remercie également Christine Froidevaux, Eric Angel et Franck Brouillard d'avoir accepté de faire partie de mon jury de thèse.

Je remercie également les membres de l'(ex-)équipe AMIS (Florence, Marie, Jean-Christophe, Farida et Fariza), et plus généralement ceux du laboratoire IBISC pour leur gentillesse et leur disponibilité.

J'aimerais remercier en particulier la team des thésards (actuels ou anciens) du laboratoire IBISC pour les bons moments passés ensemble : Amélie, Samira, Arnaud, Néhémy, Vincent, Stan, Adel, Lukasz, Laurent, Adrien, Ha, Romain. J'y ajoute également Benjamin, Sébastien, Markus, ainsi que les stagiaires : Lise, William, Frédéric, Jean-Baptiste, Guillaume, Sébastien, Artemis (désolée si j'en oublie certains). Je garderai un très bon souvenir des soirées ciné, petit-déjeuners et barbecues qui m'ont permis de décompresser un peu du travail de thèse ! Je remercie également toutes ces personnes pour leur disponibilité, en particulier mes "co-bureau", Arnaud, Néhémy et Adel, qui ont toujours été à l'écoute de mes diverses interrogations.

Enfin je tiens à remercier ma famille, notamment mes parents et mes sœurs, Lisa et Lauren, ainsi que mes "beau-frérots", Eric et Maxime, pour votre soutien et vos encouragements.

Table des matières

Table des figures

Liste des tableaux

Notations

ℓ	nombre d'exemples étiquetés
n	nombre d'exemples non étiquetés
$\mathcal{U}$	ensemble de données en entrée
$\mathcal{U}_\ell$	ensemble d'exemples étiquetés
$\mathcal{U}_{\ell+n}$	union des ensembles d'exemples étiquetés et non étiquetés
$\mathcal{V}$	ensemble de données en sortie
$\mathcal{F}_x$	espace caractéristique en entrée
$\mathcal{F}_y$	espace caractéristique en sortie
d	dimension de $\mathcal{F}_y$
$\mathcal{L}(\mathcal{F}_y)$	ensemble des opérateurs linéaires bornés de $\mathcal{F}_y$ dans lui-même
h	fonction de $\mathcal{U}$ dans $\mathcal{F}_y$
x	fonction caractéristique en entrée, $x : \mathcal{U} \to \mathcal{F}_x$
y	fonction caractéristique en sortie, $y : \mathcal{V} \to \mathcal{F}_y$
$\mathcal{K}_x$	noyau à valeur opérateur, $\mathcal{K}_x : \mathcal{U} \times \mathcal{U} \to \mathcal{L}(\mathcal{F}_y)$
$\mathcal{H}, \mathcal{H}_{\mathcal{K}_x}$	RKHS qui admet $\mathcal{K}_x$ comme noyau reproduisant
$\mathbf{K}_x$	matrice de Gram de $\mathcal{K}_x$
κ_x	noyau scalaire en entrée, $\kappa_x : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$
κ_y	noyau scalaire en sortie, $\kappa_y : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$
X_ℓ	$= (x(u_1), \ldots, x(u_\ell)) = (\mathbf{x}_1, \ldots, \mathbf{x}_\ell)$
$X_{\ell+n}$	$= (x(u_1), \ldots, x(u_{\ell+n})) = (\mathbf{x}_1, \ldots, \mathbf{x}_{\ell+n})$
Y_ℓ	$= (y(v_1), \ldots, y(v_\ell)) = (\mathbf{y}_1, \ldots, \mathbf{y}_\ell)$
K_{X_ℓ}	matrice de Gram de κ_x sur $\mathcal{U}_\ell$, $K_{X_\ell} = X_\ell^T X_\ell$
$K_{X_{\ell+n}}$	matrice de Gram de κ_x sur $\mathcal{U}_{\ell+n}$, $K_{X_{\ell+n}} = X_{\ell+n}^T X_{\ell+n}$
K_{Y_ℓ}	matrice de Gram de κ_y sur $\mathcal{V}_\ell$, $K_{Y_\ell} = Y_\ell^T Y_\ell$
I_p	matrice identité de taille p
$\mathcal{L}$	fonction de perte
A^*	matrice adjointe de A
$A \circ B$	produit terme à terme entre les matrices A et B
$A \otimes B$	produit de kronecker entre les matrices A et B
$\mathrm{vec}(A)$	vecteur formé en empilant les colonnes de la matrice A

Abréviations

ABC : ATP Binding Cassette
ADN : Acide DésoxyriboNucléique
ARN : Acide RiboNucléique
ARNm : Acide RiboNucléique messager
AUC : Area Under the Curve
CFTR : Cystic Fibrosis Transmembrane conductance Regulator
CV : Cross Validation
EM : Expectation-Maximization
FP : False Positive
FRET : Fluorescence Resonance Energy Transfer
GO : Gene Ontology
HSIC : Hilbert-Schmidt Independence criterion
IOKR : Input Output Kernel Regression
KDE : Kernel Dependency Estimation
LOO-CV : Leave-One-Out Cross Validation
MLN : Markov Logic Network
MMR : Maximum Margin Regression
NBD : Nucleotide-Binding Domain
OKR : Output Kernel Regression
PDB : Protein Data Bank
PKA : Protéine kinase A
PKMR : Penalized Kernel Matrix Regression
PPI : Protein-Protein Interaction
PR : Précision-Rappel
PRESS : Predicted Residual Sum of Squares
RE : Réticulum Endoplasmique
RKHS : Reproducing Kernel Hilbert Space
ROC : Receiver Operating Characteristic
SVM : Séparateurs à Vaste Marge
TAP-MS : Tandem Affinity Purification-Mass Spectrometry
TMB : TransMembrane Domain
TP : True Positive
Y2H : Yeast Two-Hybrid

Introduction

La mucoviscidose est une des maladies génétiques létales la plus fréquente dans les populations de type caucasienne. En Europe et en Amérique du Nord, environ un nouveau-né sur 2500 naissances est atteint par cette maladie. Celle-ci affecte principalement les poumons, ainsi que le pancréas, le foie et les intestins. Il n'existe pas de traitement curatif à l'heure actuelle mais la qualité et l'espérance de vie des patients ont été considérablement améliorées ces dernières années grâce aux progrès de la prise en charge.

Cette maladie est due à la présence de mutations sur le gène *CFTR* (cystic fibroses transmembrane conductance regulator), ce qui entraîne une altération de la protéines CFTR codée par ce gène. Depuis 1989, date de la découverte de la localisation du gène [Riordan et al., 1989], près de 2000 mutations différentes ont été identifiées. La plus fréquente d'entre elles est la mutation ΔF508. La mucoviscidose est causée par un seul gène, mais les manifestations cliniques et cellulaires de cette pathologie sont complexes et ne sont pas encore bien comprises. Plusieurs anomalies au niveau cellulaire ont été reportées et la relation entre le génotype et le phénotype reste obscure. On ne peut ainsi expliquer pour l'instant la multiplicité des phénotypes observés chez des patients présentant la même mutation du gène *CFTR* [Ollero et al., 2006]. Ces différents paradoxes indiquent que les manifestations de la mucoviscidose dépendent de plusieurs facteurs cellulaires.

La plupart des protéines réalisent leurs fonctions en interagissant avec d'autres protéines. Ces interactions sont appelées interactions protéine-protéine et correspondent à des contacts physiques ayant lieu dans la cellule entre deux protéines. Dans le cas de la mucoviscidose, la connaissance de ces interactions participe à la compréhension du fonctionnement et de la régulation de CFTR. Cette connaissance apparaît donc comme une source potentielle pour développer de nouvelles stratégies thérapeutiques.

Les interactions protéine-protéine ont été traditionnellement identifiées et caractérisées par des techniques de détection expérimentale à petite échelle. Cependant, celles-ci restent longues et coûteuses à mettre en œuvre. Plus récemment, des approches de détection à grande échelle ont été développées, comme par exemple le double hybride de la levure [Uetz et al., 2000; Ito et al., 2000] pour la détection d'interactions physiques directes entre protéines ou les méthodes basées sur la spectrométrie de masse [Gavin et al., 2002; Ho et al., 2002] pour l'identification de complexes protéiques. Ces méthodes permettent de rechercher les interactions protéine-protéine à l'échelle du protéome. Cependant, les données obtenues sont souvent incomplètes et présentent des taux importants de faux positifs (interactions détectées expérimentalement mais qui n'ont pas lieu dans la cellule) et de faux négatifs (interactions existantes qui n'ont pas été identifiées par la méthode) [von Mering et al., 2002].

Afin de pallier à ces limitations, il est intéressant de développer des méthodes de prédiction *in silico*. Celles-ci permettent de suggérer aux biologistes de nouvelles interactions à vérifier expérimentalement. L'ensemble des interactions protéine-protéine ayant lieu dans un organisme, un organe donné ou un type cellulaire donné, est généralement représenté sous la forme d'un graphe, dans lequel les nœuds représentent des protéines, et où un arc entre deux nœuds signifie l'existence d'une interaction physique entre les deux protéines correspondantes. La prédiction d'interactions entre protéines revient alors à prédire la présence ou l'absence d'arêtes dans ce graphe. La résolution de cette tâche de prédiction nécessite de déterminer quelles sont les informations qui peuvent être pertinentes pour prédire l'existence d'une interaction.

L'objectif de cette thèse est de développer des méthodes de prédiction *in silico* d'interactions protéine-protéine qui puissent être appliquées chez l'homme, mais également de proposer un cadre général pour résoudre ce problème. Nous avons choisi pour cela de nous placer dans le cadre de l'apprentissage statistique. Ce cadre désigne un ensemble de méthodes permettant d'analyser un phénomène et de prédire son comportement futur à partir d'un ensemble d'observations. Celui-ci est de plus en plus présent en bio-informatique car il fournit des outils permettant d'accompagner le biologiste dans une démarche expérimentale en exploitant des données existantes, ainsi que des connaissances relatives au problème étudié. Des méthodes d'apprentissage ont été ainsi développées pour résoudre divers problèmes, tels que l'inférence de réseaux biologiques, la prédiction de structures protéiques, la prédiction de fonctions des protéines ou encore la prédiction de localisations cellulaires des protéines [Larranaga et al., 2006].

Une des problématiques les plus courantes en apprentissage est celle de l'apprentissage supervisé. Dans ce cadre, on dispose d'un ensemble d'observations associées à des sorties, appelées étiquettes. L'objectif de l'apprentissage supervisé consiste à apprendre une fonction capable de prédire correctement l'étiquette associée à une nouvelle observation. Dans le cas du problème de la prédiction de liens, l'apprentissage supervisé vise à construire des classifieurs capables de prédire s'il existe un lien entre deux entités à partir de descriptions de ces entités. L'hypothèse sous-jacente est qu'il existe une relation entre les attributs de chaque élément d'une paire d'entités et la présence d'une arête ou son absence. Les approches d'apprentissage supervisé tirent parti des régularités présentes dans un ensemble dit d'apprentissage, comprenant des exemples de triplet de la forme (entité1, entité2, présence ou absence de lien).

Par ailleurs, des informations supplémentaires peuvent être également disponibles pour les objets non étiquetés, c'est à dire les objets pour lesquels les présences ou absences de lien ne sont pas connues. En biologie par exemple, il est relativement facile d'obtenir une description détaillées des propriétés d'une protéines grâce aux nombreuses bases de données, alors que la détection expérimentale d'interactions physiques entre protéines reste longue et coûteuse. L'apprentissage semi-supervisé [Chapelle et al., 2006] est une problématique d'apprentissage statistique qui

permet d'utiliser à la fois les données étiquetées et les données non étiquetées lors de l'apprentissage. Lorsque que l'on dispose seulement d'une petite quantité de données étiquetées, l'apprentissage semi-supervisé permet généralement d'obtenir de meilleurs résultats que l'apprentissage supervisé. Dans le cas du problème de la prédiction de liens, le nombre de données étiquetées est relativement faible et l'utilisation de l'apprentissage semi-supervisé apparaît donc pertinente. On parle d'apprentissage transductif lorsque l'on dispose d'un ensemble de protéines décrites par des attributs en entrée et que l'on cherche à compléter le réseau partiellement connu entre ces protéines en exploitant ces informations.

Une autre difficulté apparaissant dans ce problème de prédiction concerne le fait que l'on ne dispose pas d'exemples réels d'absence d'interaction, mais d'exemples pour lesquels aucune interaction n'a été détectée jusqu'à présent. Nous disposons donc d'un faible nombre d'interactions et d'un très grand nombre d'absences d'interaction, parmi lesquelles peuvent se trouver des exemples dits "faux négatifs". Dans cette thèse, nous nous concentrons essentiellement sur le problème d'apprentissage semi-supervisé. Nous ne traiterons pas spécifiquement le problème des faux négatifs.

De nombreuses méthodes ont été développées pour résoudre ce problème de prédiction, soit dans le cadre de l'apprentissage supervisé, soit dans le cadre de l'apprentissage transductif. Les méthodes à noyaux, connues pour permettre le traitement de données structurées en entrée, ont montré en particulier de très bonnes performances [Scholkopf & Smola, 2002]. Ces méthodes sont basées sur l'utilisation d'une fonction noyau prenant deux objets en entrée et correspondant à un produit scalaire dans un espace de redescription des données. Parmi les méthodes à noyaux supervisées ayant été proposées dans ce cadre, on distingue deux types de méthodes : celles qui sont basées sur des noyaux définis entre des paires de nœuds [Ben-Hur & Noble, 2005; Martin et al., 2005; Vert et al., 2007] et celles qui utilisent des noyaux entre nœuds [Yamanishi et al., 2004; Vert & Yamanishi, 2005; Geurts et al., 2007; Bleakley et al., 2007]. D'autres méthodes ont été également développées dans le cadre de l'apprentissage transductif [Tsuda et al., 2003; Kato et al., 2005; Yip & Gerstein, 2009; Kashima et al., 2009b].

Dans ce travail de thèse, nous avons développé de nouvelles méthodes à noyaux qui permettent de prédire des sorties structurées. Pour cela, nous avons converti le problème de classification binaire à partir de paires d'objets en un problème d'apprentissage de noyau de sortie comme cela a été fait par Geurts et al. [2006, 2007]. Un noyau de sortie cible est supposé coder une similarité entre les données en tant que nœuds dans le graphe et l'objectif de l'apprentissage est d'approcher cette fonction en utilisant des descripteurs appropriés en entrée. Ainsi, pour une protéine, on va chercher à connaître les proximités relatives existant entre cette protéine et les autres protéines dans le réseau. L'utilisation de l'astuce du noyau dans l'espace de sortie permet de réduire le problème de l'apprentissage à partir de paires d'objets à celui de l'apprentissage d'une fonction d'une seule variable à valeurs dans un espace de Hilbert. Cette tâche de régression supervisée est appelée *régression à noyau de*

sortie [Geurts et al., 2006, 2007].

Les méthodes à base d'arbres développées précédemment dans l'équipe pour résoudre cette tâche de régression fournissent de très bon résultats. Cependant, celles-ci ne sont pas adaptées à l'apprentissage semi-supervisé. Un des moyens les plus rigoureux d'apprendre une fonction dans le cadre de l'apprentissage semi-supervisé consiste à appliquer à cette fonction, une contrainte de régularisation tenant compte des données non étiquetées [Belkin & Niyogi, 2004; Zhou et al., 2004]. En travaillant dans un espace fonctionnel bien choisi, des théorèmes de représentation nous donnent la forme de modèles qui permettent de résoudre une tâche semi-supervisée. Dans le cas de fonctions à valeurs scalaires, des espaces fonctionnels intéressants sont les espaces de Hilbert à noyau reproduisant, que l'on définit à partir de noyaux à valeurs scalaires.

Dans cette thèse, nous nous sommes tournés vers la théorie des espaces de Hilbert à noyau reproduisant dédiée aux fonctions à valeurs dans un espace de Hilbert [Micchelli & Pontil, 2005; Senkene & Tempel'man, 1973]. En se plaçant dans le cadre de cette théorie, nous avons proposé une nouvelle approche, appelée *régression à noyaux d'entrée et de sortie* (IOKR pour *Input and Output Kernel Regression*). Celle-ci permet de bénéficier de la polyvalence des noyaux à coder diverses connaissances structurées dans l'espace d'entrée, comme dans l'espace de sortie. Dans ce cadre théorique, nous établissons un nouveau théorème de représentation dédié à l'apprentissage semi-supervisé. Puis, nous proposons deux modèles, appelés *IOKR-ridge* et *IOKR-margin*, qui ont été obtenus pour deux fonctions de perte différentes.

Enfin, nous étayons la méthode que nous avons développée sur des données artificielles, des problèmes de test, ainsi que deux jeux de données réels : un réseau d'interaction protéine-protéine chez la levure et le réseau d'interaction autour de la protéine CFTR.

Organisation du document

Ce document est organisé en trois parties, qui sont les suivantes :

- **Partie 1 : Etat de l'art.**
  La première partie expose l'état de l'art relatif à ce travail et les différentes notions qui seront utilisées dans ce document. Le **chapitre 1** est consacré au contexte biologique des interactions protéine-protéine. Une section de ce chapitre est également dédiée à la protéine CFTR et son implication dans la mucoviscidose. Dans le **chapitre 2**, nous présentons les principaux concepts de l'apprentissage statistique utiles à cette thèse ainsi que les différents paradigmes d'apprentissage existants. Nous décrivons en particulier les méthodes à noyaux et le cadre de la régularisation, dans lequel nous nous placerons dans la suite de la thèse. Le **chapitre 3** est, lui, dédié au problème général de la prédiction de liens qui a été étudié dans de nombreux domaines, que ce soit pour le problème de la prédiction d'interactions physiques entre des protéines ou

celui de la prédiction de relations sociales entre des individus. Nous décrivons dans ce chapitre les différentes approches existantes et leurs caractéristiques.

– **Partie 2 : Régression à noyaux d'entrée et de sortie pour la prédiction de liens.**
Dans le **chapitre 4**, nous présentons la méthode de régression à noyaux d'entrée et de sortie que nous avons développée au cours de la thèse. Nous montrons également comment celle-ci peut être appliquée de manière générale pour résoudre des problèmes de régression à sortie structurée. Le **chapitre 5** est dédié plus spécifiquement à l'application de la régression à noyaux d'entrée et de sortie au problème de la prédiction de liens. Nous présentons les différents modèles obtenus dans le cadre de l'apprentissage supervisé et de l'apprentissage semi-supervisé. Nous présentons également une méthode par transfert de liens, qui permet de bénéficier de l'information des interactions connues chez plusieurs espèces pour apprendre les interactions protéine-protéine chez une espèce cible.

– **Partie 3 : Résultats numériques.**
Nous présentons dans la dernière partie les différents résultats numériques obtenus avec la méthode de régression à noyaux d'entrée et de sortie. Dans le **chapitre 6**, nous décrivons les résultats obtenus par la méthode sur un ensemble de jeux de données synthétiques, un réseau de co-publication et un réseau d'interaction protéine-protéine utilisé comme problème test par plusieurs approches. Nous présentons ensuite dans le **chapitre 7** des expériences réalisées sur un réseau d'interaction protéine-protéine de la levure actualisé à partir de divers types d'informations utilisés en entrée. Enfin, le **chapitre 8** est dédié à l'application de la méthode que nous avons développée au réseau d'interaction protéine-protéine autour de la protéine CFTR, impliquée dans la mucoviscidose.

La troisième partie est suivie d'une conclusion et d'un ensemble de perspectives.

Première partie

Etat de l'art

Nous exposons dans cette partie les différentes notions qui seront utilisées au cours de cette thèse.

Nous commençons par introduire dans le chapitre 1 le contexte biologique des interactions protéine-protéine. Nous montrons en particulier l'importance de leur identification dans le cas de la protéine CFTR qui joue un rôle central dans la pathologie de la mucoviscidose. Nous présentons un résumé de l'état de l'art concernant les méthodes expérimentales permettant de détecter ces interactions et nous évoquons les méthodes de prédiction *in silico* d'interactions protéine-protéine qui ont été développées afin de compléter les approches expérimentales.

Dans le chapitre 2, nous introduisons les principaux concepts de l'apprentissage statistique utiles à cette thèse. Nous décrivons les trois principaux paradigmes d'apprentissage existants, c'est à dire l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage semi-supervisé. Un autre point présenté concerne le cadre de la régularisation, qui est celui dans lequel nous nous sommes placés dans cette thèse. Les méthodes à noyaux sont également abordées dans ce chapitre, et en particulier deux méthodes à noyaux régularisées qui sont : la *Kernel Ridge Regression* et les *Séparateurs à Vaste Marge*.

Enfin, nous exposons dans le chapitre 3 le problème de la prédiction de liens, qui consiste à prédire l'existence d'un lien entre deux entités appartenant a un réseau. Ce problème a été étudié dans le cadre de l'inférence de réseaux biologiques, et en particulier pour la prédiction d'interactions protéine-protéine. Mais plus généralement, ce problème a été le sujet d'études dans de nombreux domaines tels que les réseaux bibliographiques, les réseaux sociaux ou les réseaux technologiques. Nous décrivons dans ce chapitre les principales approches qui ont été développées pour résoudre ce problème. Celles-ci peuvent être réparties en différentes catégories : les approches basées sur des motifs topologiques, les méthodes basées utilisant des modèles probabilistes, les approches non supervisées et les approches discriminantes.

Chapitre 1

Interactions protéine-protéine

L'objectif de ce chapitre est de présenter le contexte biologique relatif à cette thèse. Nous introduisons les notions de protéine et d'interaction protéine-protéine dans les sections 1.1 et 1.2. La section 1.3 est dédiée plus spécifiquement à la description de la protéine CFTR. Nous présentons ensuite dans la section 1.4 un état de l'art au sujet des méthodes expérimentales de détection d'interactions physiques entre protéines. Un résumé de la littérature concernant les sources de données utilisées par les méthodes de prédiction *in silico* d'interactions protéine-protéine est également présenté dans la section 1.5.

1.1 Protéines

Les protéines sont des macromolécules constituées d'une ou plusieurs chaînes d'acides aminés qui peuvent assurer des fonctions très diverses au sein de la cellule ou de l'organisme. Elles sont assemblées à partir des acides aminés en fonction de l'information contenue dans les gènes.

1.1.1 Du gène à la protéine

Le support physique de l'information génétique d'un individu est contenu dans une molécule, appelée acide désoxyribonucléique (ADN). Cette molécule est composée de deux longues chaînes de nucléotides, chaque nucléotide étant constitué d'un groupe phosphate, d'une molécule de sucre et d'une base azotée. Il existe quatre bases azotées différentes : l'adénine (A), la guanine (G), la cytosine (C) et la thymine (T). L'enchaînement de ces nucléotides est appelé séquence d'ADN. Les deux brins d'ADN sont dits complémentaires car l'adénine d'un brin fait toujours face à la thymine de l'autre brin, et de même la guanine d'un brin fait toujours face à la cytosine de l'autre brin. Les deux brins de l'ADN forment ensemble une double hélice. Chez les eucaryotes, l'ADN est principalement localisé dans le noyau cellulaire sous la forme de différentes molécules. Ces molécules d'ADN sont associées à des protéines, appelées histones, afin de former les chromosomes (voir figure 1.1).

Un gène peut être défini comme un ensemble de séquences d'ADN codant pour un ensemble cohérent de produits fonctionnels (acide ribonucléique ou protéine). Cependant, d'autres définitions d'un gène sont également possibles. L'étape de synthèse d'une molécule d'acide ribonucléique (ARN) à partir de l'ADN s'appelle la *transcription*. Le démarrage de la transcription d'un ARN par une ARN polymérase s'effectue au niveau d'une séquence spécifique de nucléotides, appelée promoteur et une séquence terminatrice marque la fin de la transcription. L'ARN messager (ARNm)

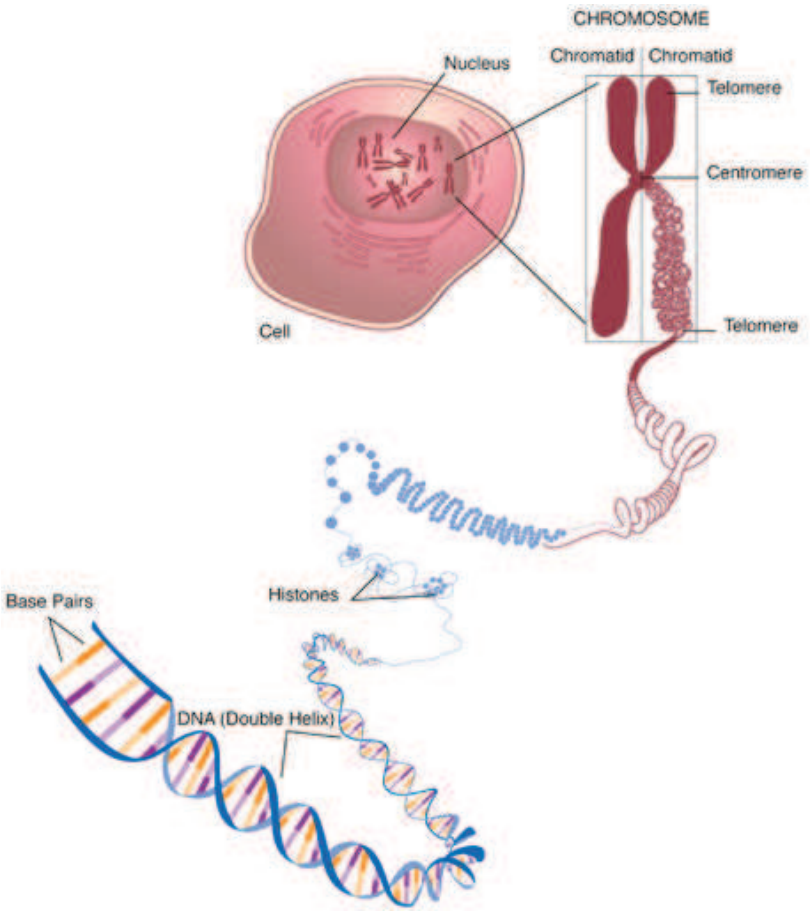

FIGURE 1.1 – **Aspect et localisation de l'ADN.**

est l'ARN qui est utilisé comme intermédiaire pour la synthèse des protéines via le processus de *traduction* (voir figure 1.2). L'information génétique contenue dans la chaîne de nucléotides de l'ARNm est traduite en une chaîne d'acides aminés selon le code génétique. Ce code établit une correspondance entre un triplet de nucléotides sur l'ARNm et un acide aminé.

Dans le cas des organismes eucaryotes, les gènes sont constitués d'une suite d'exons et d'introns alternés. Après l'étape de transcription, l'ARN messager synthétisé va subir un processus d'épissage, au cours duquel les introns vont être excisés de l'ARN afin de donner lieu à un ARNm mature. Le processus d'épissage alternatif de l'ARN messager peut conduire à la production de plusieurs protéines différentes codées par un même gène.

Après sa synthèse, une protéine peut subir au cours de sa vie dans la cellule des modifications chimiques, appelées *modifications post-traductionnelles*. Ces modifications sont le plus souvent réalisées par des enzymes et entraînent généralement des changements de la fonction de la protéine. L'activité, la localisation des protéines ou leurs interactions avec d'autres molécules sont ainsi influencées par ces

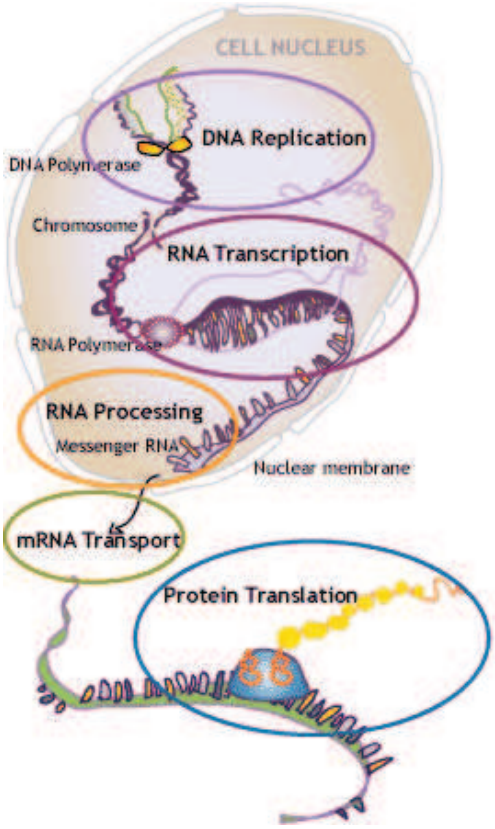

FIGURE 1.2 – **Dogme central de la biologie** : la figure représente les différentes étapes conduisant à la synthèse d'une protéine à partir de l'information contenue dans l'ADN.

modifications.

Des mécanismes de régulation peuvent être mis en œuvre à chaque étape du processus de synthèse des protéines. Ainsi toutes les cellules d'un organisme contiennent le même génome, mais certains gènes ne sont exprimés que dans certaines cellules, sous certaines conditions ou à une période spécifique du développement de l'organisme. Parmi les différents mécanismes existants, on peut citer les modifications post-traductionnelles présentées précédemment, ainsi que les régulations transcriptionnelles. Ces régulations sont effectuées par des protéines, appelées facteurs de transcription. Ces facteurs de transcription se fixent sur les régions promotrices d'autres gènes, ce qui permet d'activer ou d'inhiber l'expression de ces gènes.

1.1.2 Fonctions des protéines

Les protéines sont impliquées dans quasiment toutes les fonctions mises en œuvre dans la cellule. Les principales classes fonctionnelles de protéines incluent [Lesk, 2010] :

– les protéines de structure, qui entrent dans la constitution des tissus ;

- les protéines enzymatiques, qui catalysent les réactions chimiques du métabolisme ;
- les protéines de défense, comme les anticorps du système immunitaire ;
- les protéines régulatrices, telles que les facteurs de transcription qui contrôlent l'expression des gènes ;
- les protéines de signalisation, qui détectent les signaux extérieurs et les transmettent dans la cellule ;
- les protéines de transport, qui contrôlent le trafic à l'intérieur et à l'extérieur des cellules ;
- les protéines motrices, qui permettent aux cellules ou à certains éléments de se mouvoir.

1.1.3 Structures des protéines

La structure des protéines peut être décrite à différents niveaux :

- la structure primaire correspond à la séquence d'acides aminés,
- la structure secondaire désigne l'organisation de groupes d'acides aminés en structures locales, telles que les hélices alpha et les feuillets beta,
- la structure tertiaire correspond au repliement de la protéine dans l'espace tridimensionnel. Cette structure représente les relations spatiales existant entre les différentes structures secondaires,
- la structure quaternaire correspond à l'association de plusieurs protéines entre elles.

1.2 Interactions entre protéines

Les protéines agissent rarement de façon isolée. La plupart d'entre elles réalisent leurs fonctions biologiques en interagissant avec d'autres protéines. Ces interactions correspondent à des contacts physiques entre les protéines qui ont lieu dans la cellule et sont appelées interactions protéine-protéine (PPI). Les signaux extra-cellulaires, par exemple, sont transmis à l'intérieur de la cellule par l'intermédiaire de telles interactions. Ce processus, appelé transduction du signal, joue un rôle très important dans de nombreux processus biologiques, ainsi que dans plusieurs pathologies, dont le cancer. L'identification et la caractérisation de ces interactions sont donc essentielles pour mieux comprendre les mécanismes des processus biologiques au niveau moléculaire.

Les interactions n'ont pas lieu dans toutes les cellules ou à un moment quelconque. Le contexte biologique est, au contraire, un élément essentiel. Ainsi, les interactions dépendent du type de cellule, de la phase du cycle cellulaire, du stade de développement, des conditions environnementales, des modifications post-traductionnelles des protéines, ou encore de la présence d'autres partenaires [De Las Rivas & Fontanillo, 2010].

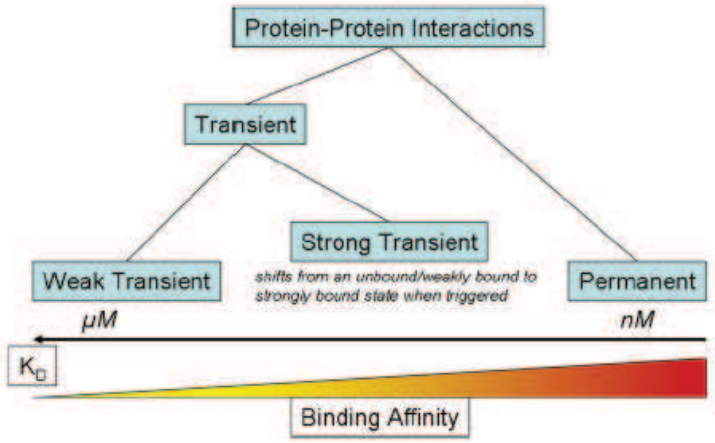

FIGURE 1.3 – **Les différents types d'interaction protéine-protéine sur la base de leurs affinités de liaison** : L'affinité de liaison est inversement reliée à la constante de dissociation. Alors que dans le cas des interactions permanentes, l'affinité de liaison est de l'ordre du nm, celle-ci est de l'ordre du μm dans le cas des interactions transitoires. La catégorie des interactions transitoires fortes illustre le continuum qui existe entre les interactions transitoires les plus faibles et les interactions les plus permanentes [Perkins et al., 2010].

1.2.1 Les types d'interaction

Les interactions entre protéines peuvent être distinguées selon la force de l'interaction [Nooren & Thornton, 2003; Keskin et al., 2008] (voir figure 1.3). Certaines d'entre elles sont qualifiées de permanentes car elles sont en général très stables et irréversibles. Les complexes protéine-protéine permanents sont formés par des protéines qui fonctionnent uniquement lorsqu'elles sont associées dans un complexe. Les homodimères, qui correspondent à des interactions protéine-protéine entre deux protéines identiques, sont ainsi des exemples de complexes permanents. D'autres interactions protéine-protéine sont classées dans la catégorie des interactions transitoires. Ces interactions peuvent s'associer et se dissocier *in vivo*. Elles dépendent de l'état fonctionnel des partenaires impliqués. C'est le cas par exemple des interactions du type hormone-récepteur ou enzyme-inhibiteur. Les interactions permanentes sont en général plus serrées et présentent une meilleur complémentarité de forme. En revanche, les interfaces des complexes transitoires sont moins étendues et les associations entre les protéines sont plus faibles.

Les interactions transitoires sont subdivisées en deux catégories : les interactions transitoires faibles et les interactions transitoires fortes. Cette distinction se fait en fonction de la valeur de la constante de dissociation et de la durée de vie de l'interaction. Cependant, dans la réalité, les interactions forment un continuum allant des interactions transitoires les plus faibles aux interactions permanentes.

Les interactions se différencient également selon leur spécificité. Une protéine réside dans un environnement présentant de nombreux partenaires d'interaction potentiels. La plupart des protéines sont très spécifiques dans leurs choix de partenaires, mais certaines d'entre elles peuvent interagir avec de multiples partenaires [Nooren & Thornton, 2003]. La spécificité dépend de diverses caractéristiques, telles que la complémentarité des formes, les propriétés physico-chimiques, ainsi que la localisation des protéines.

1.2.2 Les domaines

Les protéines interagissent généralement ensemble par l'intermédiaire des *domaines*. Il s'agit d'unités structurales et fonctionnelles des protéines qui sont conservées au cours de l'évolution. Une protéine peut contenir un ou plusieurs domaines, chacun d'entre eux étant généralement associé à une fonction spécifique. La combinaison des domaines d'une protéine détermine sa fonction, sa localisation cellulaire et les interactions dans lesquelles elle est impliquée. En effet, les protéines peuvent interagir ensemble par l'intermédiaire d'une interaction entre deux domaines appartenant à ces protéines ou bien par l'intermédiaire d'une interaction entre un domaine (ou une protéine) et un peptide appartenant à une des protéines et contenant généralement entre 2 et 8 acides aminés.

Cependant, ces deux types d'interaction sont de natures très différentes [Stein et al., 2009]. Les relations qui constituent les différentes voies métaboliques sont, par exemple, souvent mises en œuvre par l'intermédiaire d'interactions entre domaines. Ces interactions sont caractérisées par une grande surface de contact, ce qui permet une forte liaison. Par contre, dans les réseaux de signalisation et de régulation, le fait que certaines interactions soient très dynamiques permet de pouvoir répondre rapidement à des perturbations cellulaires ou à des modifications de l'environnement. Les interactions dans ces types de réseaux sont donc souvent caractérisées par de petites interfaces, dans lesquelles un peptide d'une protéine est lié à un domaine d'une autre protéine.

1.2.3 Réseau d'interaction protéine-protéine

L'ensemble des interactions protéine-protéine ayant lieu dans un organisme, un organe ou un type cellulaire donné est appelé *interactome*. Un interactome est souvent représenté par un réseau, dans lequel les nœuds correspondent à des protéines, et où un arc entre deux nœuds signifie l'existence d'une interaction entre les deux protéines correspondantes. Ce réseau, appelé *réseau d'interaction protéine-protéine*, est non dirigé. Cependant, cette représentation ne tient pas compte de la dynamique et de la localisation des interactions protéine-protéine dans la cellule. La figure 1.4 illustre un exemple de réseau d'interaction protéine-protéine chez la levure du boulanger [Jeong et al., 2001].

Il existe par ailleurs d'autres types de réseau biologique, comme par exemple

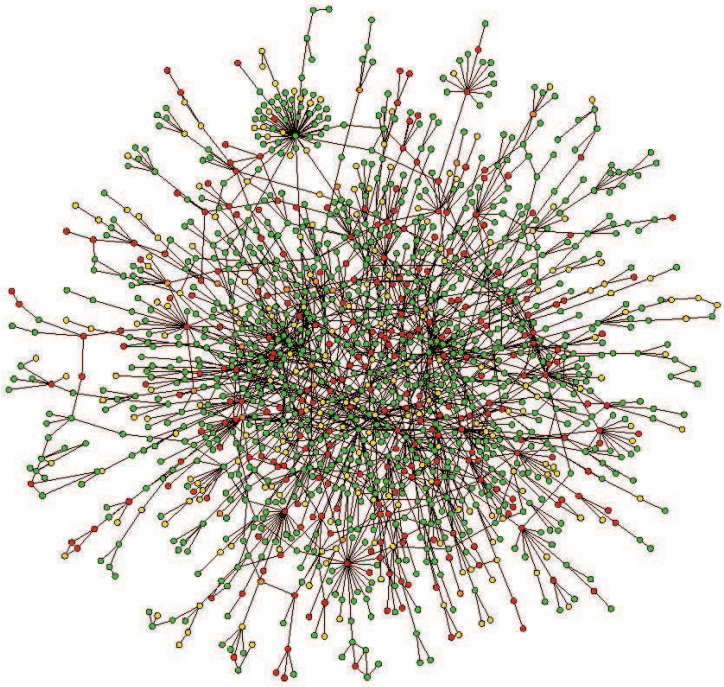

FIGURE 1.4 – **Visualisation d'un réseau d'interaction protéine-protéine chez la levure du boulanger** [Jeong et al., 2001].

le réseau de régulation génétique et le réseau métabolique. Le premier d'entre eux désigne un graphe dirigé, dans lequel les nœuds représentent des gènes, et où il existe un arc dirigé entre le gène A et le gène B si A régule B. Dans ce réseau, un arc peut être étiqueté afin d'indiquer s'il s'agit d'une régulation positive (activation) ou d'une régulation négative (inhibition). Le réseau métabolique, quant à lui, est constitué de noeuds correspondant à des gènes qui codent pour des protéines particulières : les enzymes. La fonction principale de ces protéines est de catalyser les réactions chimiques. Deux enzymes sont connectées dans un réseau de type métabolique si celles-ci catalysent deux réactions successives dans une voie métabolique.

1.3 La protéine CFTR, impliquée dans la mucoviscidose

La mucoviscidose est une maladie létale d'origine génétique qui touche principalement les poumons, le pancréas, le foie, les intestins et le système reproducteur. La mortalité des patients à un âge précoce est principalement due à une inflammation chronique aggravée par des infections causées par des bactéries, telles que *Pseudomonas aeruginosa*. Cette inflammation entraîne alors une détérioration des fonctions pulmonaires. La mucoviscidose est la maladie héréditaire grave la plus fréquente dans les populations européennes et nord-américaines. Environ une personne sur 50 est porteuse du gène muté et un enfant sur 2500 naît atteint de cette maladie.

Cette maladie est causée par des mutations du gène *CFTR* (cystic fibrosis transmembrane conductance regulator) localisé sur le chromosome 7 [Rommens et al., 1989], entraînant une altération de la protéine CFTR codée par ce gène.

1.3.1 La protéine CFTR

La protéine CFTR est une protéine membranaire de 1480 acides aminés. Cette protéine est localisée au niveau de la membrane apicale des cellules épithéliales. L'épithelium est composé de cellules associées les unes aux autres grâce à des jonctions intercellulaires. Il constitue une barrière cellulaire entre les compartiments intra- et extra-cellulaire, et il possède deux fonctions principales : la protection vis à vis du milieu extérieur et le transport transcellulaire.

Structure

Il n'existe pas à ce jour de structure tertiaire résolue de la protéine CFTR, mais l'analyse de la séquence primaire a permis de reconstituer une structure tertiaire probable [Riordan et al., 1989]. Celle-ci est composée de deux domaines membranaires (TMD pour *TransMembrane Domain*) et de trois domaines cytoplasmiques : deux domaines d'interaction avec les nucléotides (NBD pour *Nucleotide-Binding Domain*) et un domaine de régulation (R) (voir figure 1.5).

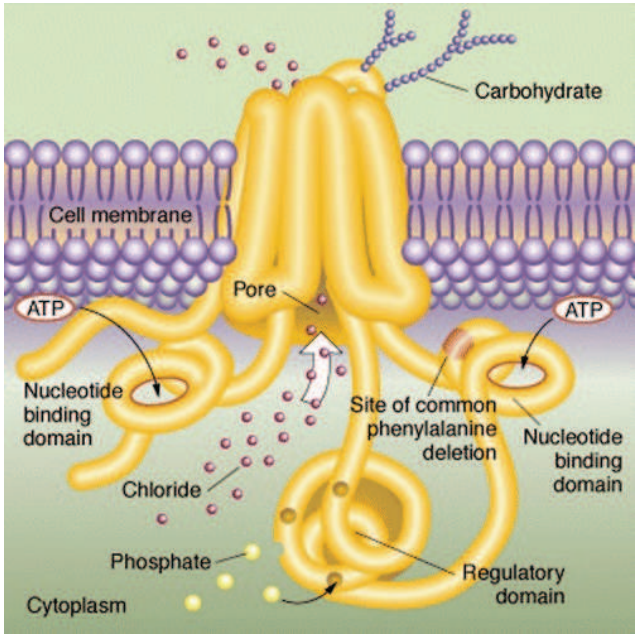

FIGURE 1.5 – **Représentation schématique de la structure tertiaire de la protéine CFTR.**

La protéine CFTR appartient à une famille de protéines membranaires, les *transporteurs ABC* (ATP Binding Cassette). Le rôle des transporteurs ABC consiste à

acheminer de part et d'autre de la membrane cytoplasmique différentes molécules telles que des protéines, des lipides, des sucres, des ions, etc. Tous les transporteurs ABC présentent une organisation commune en quatre domaines : deux domaines TMD sont supposés permettre le passage des différentes molécules à travers la membrane, et deux domaines NBD sont des domaines de liaison à l'ATP (Adénosine triphosphate). Il s'agit d'une molécule dont le rôle est de fournir l'énergie nécessaire aux réactions chimiques du métabolisme.

Fonction

La protéine CFTR est un canal chlorure qui permet de réguler le transport des ions chlorure à travers la membrane cellulaire. L'ouverture du canal constitué par la protéine CFTR se déroule en deux étapes (voir figure 1.6). La première étape correspond à la phosphorylation du domaine R par des protéines kinase A (PKA). Il s'agit d'un mécanisme de régulation, en l'occurrence d'une modification post-traductionnelle, correspondant à l'addition d'un groupe phosphate. La liaison et l'hydrolyse de l'ATP sur les domaines NBD constituent la deuxième étape. Celle-ci induit une modification de la conformation des domaines transmembranaires de la protéine, permettant le passage des ions chlorure Cl^- à travers le canal. La déphosphorylation effectuée par des protéines phosphatases inactive ensuite le canal.

La protéine CFTR régule par ailleurs le fonctionnement d'autres canaux ioniques, comme le canal ENaC (Epithelial Na Channel) ou le canal ORCC (Outwardly Rectifying Chloride Channel), et est impliquée dans le transport de l'ATP, ce qui en fait une protéine multifonctionnelle.

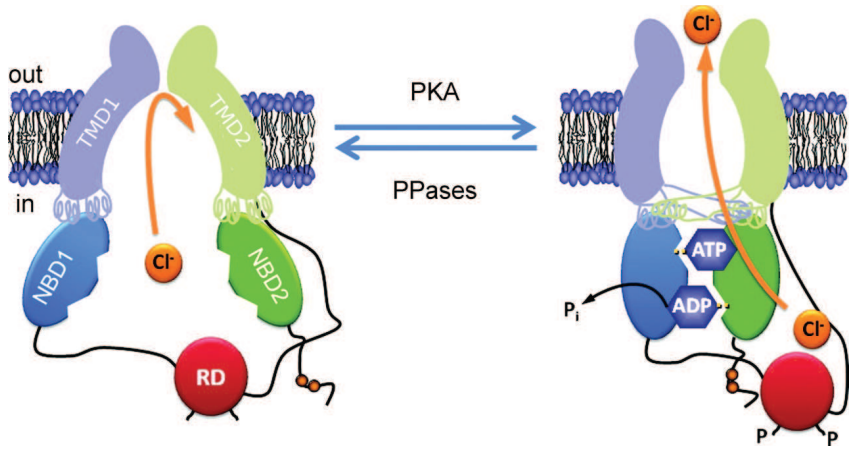

FIGURE 1.6 – **Régulation du canal CFTR** d'après Chen et al. [2006b] : la figure représente le canal CFTR dans l'état de repos (à gauche) et dans l'état activé (à droite). P indique la phosphorylation du domaine R de régulation (noté RD).

1.3.2 Anomalies moléculaires

Comme la plupart des protéines eucaryotes, la protéine CFTR est synthétisée et assemblée dans le réticulum endoplasmique (RE) [Skach, 2000]. Ce compartiment joue un rôle important dans le repliement et la maturation des protéines. Dans le cas de la protéine CFTR sauvage, une fois celle-ci synthétisée dans le RE, elle est transportée vers l'appareil de Golgi puis vers la membrane plasmique des cellules épithéliales.

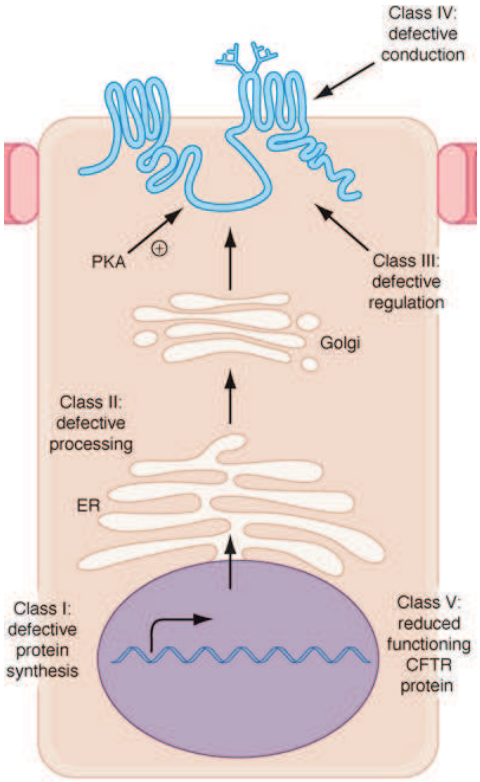

FIGURE 1.7 – **Les différentes anomalies moléculaires liées aux mutations du gène *CFTR*** : Les mutations du gène *CFTR* sont regroupées en différentes classes, qui correspondent à différentes altérations de la biosynthèse de la protéine ou à la perte d'une fonction.

Les mutations du gène *CFTR* peuvent avoir des conséquences variables sur la synthèse de la protéine CFTR ou sur sa fonction. Près de 2000 mutations du gène ont été identifiées à ce jour. Ces mutations ont été séparées en six catégories, selon le type d'anomalie moléculaire engendrée (voir figure 1.7) :

1. altération de la synthèse de la protéine CFTR, entraînant une absence totale ou partielle de la protéine,
2. perturbation du processus de maturation de la protéine CFTR,
3. perturbation de la régulation du canal CFTR,

4. altération de la conduction des ions chlorure par le canal CFTR,
5. altération de la stabilité de l'ARNm CFTR,
6. altération de la stabilité de CFTR à la membrane plasmique.

La plus fréquente des mutations est la mutation ΔF508, qui consiste en une délétion de trois nucléotides, entraînant la suppression d'un acide aminé, la phénylalanine, en position 508. Cette mutation est retrouvée dans près de 70% des cas. Celle-ci appartient à la deuxième catégorie de mutations. Ainsi dans le cas de la protéine CFTR ΔF508, le mécanisme de contrôle qualité du RE détecte un repliement anormal chez la protéine et marque la protéine défectueuse pour qu'elle soit dégradée. Par conséquent, la délétion ΔF508 bloque la production de la protéine de façon précoce, l'empêchant ainsi d'atteindre la membrane pour y exercer ses fonctions. Le défaut de synthèse de la protéine entraîne alors un déficit en chlore extra-cellulaire, provoquant une augmentation de la viscosité du mucus et son accumulation dans les voies respiratoires et digestives.

La mucoviscidose est une maladie multifactorielle qui associe un défaut de transport de fluides dans les épithéliums à des infections et inflammations chroniques, principalement dans les poumons. L'inflammation et l'infection conduisent à l'obstruction des voies aériennes des patients.

Bien que la nature de la mutation détermine en partie la localisation et la fonction de la protéine CFTR défectueuse, la relation entre le génotype et le phénotype reste toujours obscure. En effet, on ne peut expliquer la multiplicité des phénotypes observés chez des patients présentant la même mutation du gène *CFTR*, et cela plus particulièrement dans le cas de la fonction pulmonaire. Même dans le cas de patients appartenant à une même fratrie, une variabilité peut être observée. Les mutations du gène *CFTR* et le dysfonctionnement de la protéine CFTR mutée ne peuvent donc expliquer la pathogenèse de la maladie dans sa totalité [Ollero et al., 2006].

La diversité des phénotypes observés suggère de conduire des analyses globales pour l'etude de la protéine CFTR. L'identification des interactions protéine-protéine peut en particulier permettre de mieux comprendre le lien existant entre le transport de fluide, dont le facteur limitant est CFTR, et l'inflammation.

1.3.3 Protéines d'interaction

Ces dernière années, les méthodes de détection expérimentale ont permis de montrer que de nombreuses protéines interagissent avec CFTR, et cela à différents stades. Depuis sa synthèse jusqu'à sa localisation à la membrane plasmique, et puis sa dégradation finale, CFTR est capable d'interagir avec un grande nombre de protéines et avec des protéines appartenant à différentes classes (voir figure 1.8). Ces interactions ont un impact sur la stabilité, la localisation et la fonction de CFTR.

Les protéines interagissant avec CFTR incluent [Amaral, 2005] :

– les protéines chaperons et co-chaperons qui régulent la biogenèse et la dégradation de CFTR,

- les protéines impliquées dans le routage, la stabilité et l'activité fonctionnelle de CFTR à la membrane plasmique. Ces protéines incluent les protéines PDZ, qui interagissent avec le motif DTRL présent sur la partie C-terminale de CFTR :
  - NHERF1 (Na(+)/H(+) exchange regulatory cofactor NHE-RF1),
  - NHERF2 (Na(+)/H(+) exchange regulatory cofactor NHE-RF2),
  - NHERF3 (Na(+)/H(+) exchange regulatory cofactor NHE-RF3),
  - SHANK2 (SH3 and multiple ankyrin repeat domains protein 2),
  - GOPC (Golgi-associated PDZ and coiled-coil motif-containing protein).

  Ces protéines incluent également les protéines du cytosquelette (Actin cytoplasmic 1, Ezrin, Cytokeratin 8/18) et les protéines SNARE (SNAP23 et STX1A), qui se fixent sur la partie N-terminale de CFTR, entraînant une régulation négative de l'activité du canal CFTR.
- les kinases et phosphatases qui contrôlent l'activité du canal,
- les protéines impliquées dans d'autres processus cellulaires, tels que la signalisation cellulaire et les voies de transport des ions.

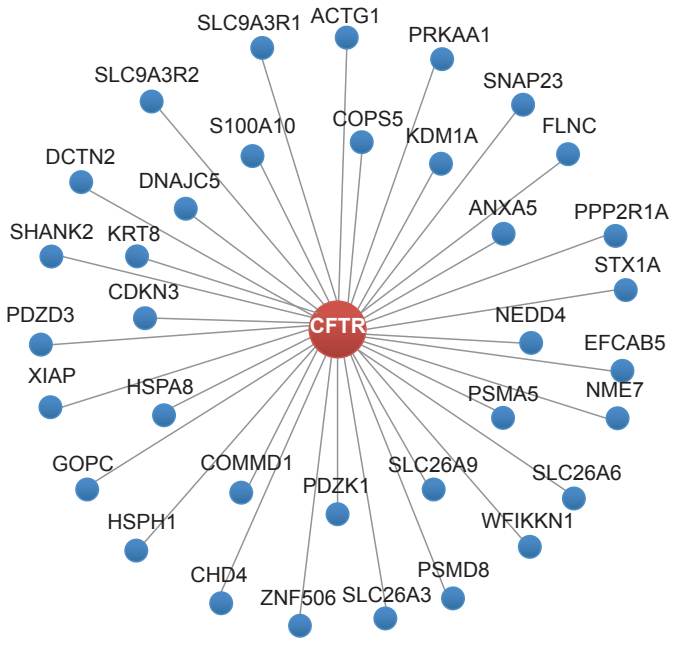

FIGURE 1.8 – **Réseau d'interaction protéine-protéine au premier ordre de la protéine CFTR** : Le réseau d'interaction a été construit à partir des bases de données BioGRID [Stark et al., 2010], DIP [Salwinski et al., 2004], MINT [Licata et al., 2012] et Nextprot [Lane et al., 2012]. Seules les interactions directes avec CFTR ont été sélectionnées. Les interactions présentes dans l'interactome publié par Wang et al. [2006] n'ont pas été considérées ici car il n'y a pas de preuve que celles-ci soient directes.

Un premier interactome autour de la protéine CFTR a été déterminé à partir d'interactions obtenues par des méthodes d'immunoprécipitation avec CFTR [Wang et al., 2006].

1.4 Méthodes de détection expérimentale

Ces dernières années, différentes méthodes de détection expérimentale d'interactions protéine-protéine ont été développées. Certaines de ces méthodes, comme le double hybride de la levure, sont des approches binaires, c'est à dire qu'elles sont capables de mesurer une interaction directe entres deux protéines. Tandis que d'autres méthodes, comme l'immunoprécipitation, permettent d'identifier des complexes protéiques. Ce type de méthode ne permet pas de connaître les protéines étant en contact direct, mais apporte néanmoins une information sur les protéines qui sont trouvées dans un même complexe à un instant donné.

Les techniques expérimentales se différencient également selon le nombre d'interactions détectées. Ainsi les méthodes de détection à grande échelle, comme le double hybride de la levure, permettent de détecter un très grand nombre d'interactions en parallèle tandis que les méthodes de détection à petite échelle se concentrent spécifiquement sur un petit nombre d'interactions. Par ailleurs, les méthodes expérimentales peuvent détecter les interactions *in vitro*, c'est à dire en dehors d'un organisme vivant ou d'une cellule, ou bien *in vivo*, c'est à dire au sein d'un organisme. Les principales méthodes de détection expérimentale et leurs caractéristiques sont listées dans les tables 1.1 et 1.2.

1.4.1 Détection expérimentale à petite échelle

Méthode	**Condition**	**Type**
Cristallographie par rayons X	*in vitro*	complexe
Spectroscopie par résonance magnétique nucléaire	*in vitro*	complexe
Transfert d'énergie fluorescente par résonance	*in vivo*	binaire
Résonance plasmonique de surface	*in vitro*	complexe
Co-immunoprécipitation	*in vitro*/ *in vivo*	complexe

TABLE 1.1 – Méthodes expérimentales de détection d'interactions protéine-protéine à petite échelle [Shoemaker & Panchenko, 2007a].

Traditionnellement, les interactions protéine-protéine ont été étudiées de façon individuelle. Des méthodes de détection à petite échelle, comme les méthodes de cristallographie par rayons X et de spectroscopie par résonance magnétique nucléaire permettent ainsi de caractériser finement les interactions et l'interface d'interaction entre deux protéines.

Parmi ces méthodes, la méthodes de co-immunoprécipitation et le FRET (*Fluorescence Resonance Energy Transfert*) sont deux méthodes de détection d'interactions *in vivo*. La première d'entre elles consiste à isoler un complexe protéique en

utilisant un anticorps dirigé contre un des membres du complexe, et à identifier ensuite les protéines obtenues. Le FRET, quant à lui, est une technique qui permet de détecter la proximité immédiate de deux protéines fluorescentes. Ainsi, lorsque ces deux protéines sont très proches l'une de l'autre, un transfert d'énergie fluorescente est effectué entre les deux protéines, ce qui a pour conséquence de modifier leurs intensités de fluorescence respectives. La combinaison de cette approche avec la technique FLIM (Fluorescence Lifetime Imaging Microscopy) permet de détecter des interactions protéine-protéine directes.

Cependant, les méthodes de détection expérimentale à petite échelle restent en général relativement coûteuses et peuvent nécessiter un temps important. La détection d'une interaction protéine-protéine avec ce type de méthode peut ainsi prendre jusqu'à plusieurs mois.

1.4.2 Détection expérimentale à grande échelle

Méthode	Condition	Type
Double hybride de la levure	*in vivo*	binaire
Purification par affinité couplée à la spectrométrie de masse	*in vitro*	complexe
Puces à protéines	*in vitro*	complexe
Phage display	*in vitro*	complexe

TABLE 1.2 – Méthodes expérimentales de détection d'interactions protéine-protéine à grande échelle [Shoemaker & Panchenko, 2007a].

Ces dernières années, des méthodes de détection à haut débit, capables de détecter un très grand nombre d'interactions, ont été développées. La méthode binaire à grande échelle la plus utilisée est celle du système du double hybride de la levure (Y2H). Deux protéines, appelées protéine appât et protéine cible, sont couplées respectivement à la moitié d'un facteur de transcription et sont exprimées chez la levure. Si les deux protéines entrent en contact, le facteur de transcription sera reconstitué et le gène rapporteur sera transcrit. Cette méthode permet d'identifier des interactions protéine-protéine à l'échelle du protéome. Des cartes d'interaction ont ainsi été identifiées chez la levure *Saccharomyces Cerevisiae* [Uetz et al., 2000; Ito et al., 2000], la bactérie *Helicobacter pylori* [Rain et al., 2001], la mouche *Drosophila melanogaster* [Giot et al., 2003], le ver *Caenorhabditis elegans* [Li et al., 2004] et l'homme [Rual et al., 2005; Stelzl et al., 2005]. Cette technique étant utilisée *in vivo*, elle permet de détecter des interactions transitoires et instables.

Cependant, le principal inconvénient de cette méthode concerne les taux importants de faux positifs (interactions détectées expérimentalement mais qui n'existent pas) et de faux négatifs (interactions existantes qui n'ont pas été détectées par la méthode) [von Mering et al., 2002]. Ces erreurs découlent en partie du fait que les interactions sont testées dans le noyau, or ce n'est pas le compartiment d'origine de beaucoup de protéines. Ainsi, des interactions physiques peuvent être détectées entre des protéines qui ne sont jamais à proximité l'une de l'autre dans la cellule car

elles ont des localisations cellulaires différentes ou bien parce qu'elles sont exprimées à des moments différents du cycle cellulaire. Par ailleurs dans certains cas, il arrive qu'il y ait des faux négatifs du fait que les interactions ne peuvent pas être détectées dans les conditions utilisées.

Parmi les méthodes de détection expérimentale à grande échelle, on trouve également les méthodes de purification par affiné couplées à la spectrométrie de masse (TAP-MS) [Gavin et al., 2002; Ho et al., 2002], qui sont très utilisées pour l'identification de complexes protéiques. Cette technique consiste tout d'abord à marquer individuellement les protéines d'intérêt. Ces protéines sont utilisées pour récupérer par purification biochimique l'ensemble des protéines du complexe. Les différentes protéines du complexe sont ensuite séparées et identifiées par spectrométrie de masse. Cependant, cette technique souffre, elle aussi, de taux importants de faux positifs et faux négatifs.

Les principes de ces deux méthodes de détection expérimentale à grande échelle sont représentés sur la figure 1.9.

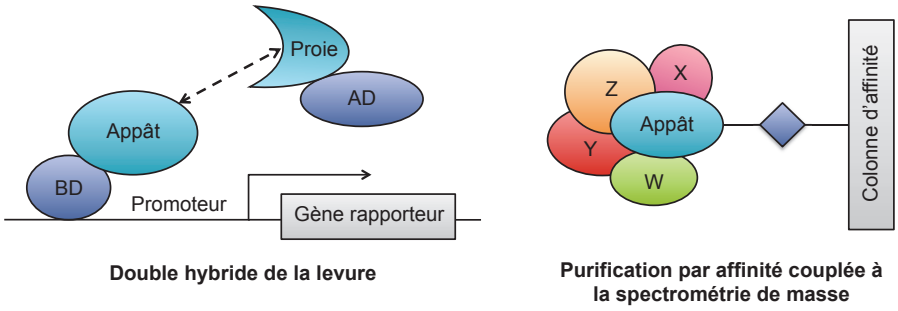

FIGURE 1.9 – **Méthodes de détection expérimentale à grande échelle** : L'idée du systèmes de double hybride de la levure est représentée à gauche de la figure. Un facteur de transcription est découpé en deux parties : un domaine de fixation sur l'ADN (BD ou *Binding Domain*) et un domaine d'activation (AD ou *Activation Domain*). Chaque protéine d'intérêt est couplée à l'un des domaines et si les deux protéines interagissent ensemble, le facteur de transcription devient actif et le gène rapporteur est transcrit.
La méthode de purification par affinité couplée à la spectrométrie de masse est illustrée à droite de la figure : un marqueur (losange violet) est fixé à une protéine d'intérêt, appelée protéine appât, ce qui permet de purifier facilement l'ensemble des protéines du complexe. Les protéines sont ensuite identifiées à l'aide d'un spectromètre de masse.

1.4.3 Bases de données

Différentes bases de données ont été créées ces dernières années afin de compiler l'information des interactions protéine-protéine détectées expérimentalement. Les principales d'entre elles sont :

- DIP (Database of Interacting Proteins) [Salwinski et al., 2004],
- BIND (Biomolecular Interaction Network Database) [Bader et al., 2003],
- MINT (Molecular Interactions Database) [Licata et al., 2012],
- STRING (Search Tool for the Retrieval of Interacting Genes) [Szklarczyk et al., 2011],
- MIPS (Munich Information center for Protein Sequences) [Mewes et al., 2004],
- Intact [Kerrien et al., 2012],
- BioGRID (General Repository for Interaction Datasets) [Stark et al., 2010],
- HPRD (Human Protein Reference Database) [Prasad et al., 2009].

1.5 Prédiction d'interactions protéine-protéine à partir de diverses sources de données

Comme nous l'avons vu dans la section 1.4, les méthodes de détection expérimentale d'interactions protéine-protéine présentent certaines limitations. Le développement de méthodes de prédiction *in silico* permet de compléter ces approches expérimentales. De nombreuses approches ont ainsi été développées pour résoudre le problème de la prédiction d'interactions physiques entre protéines à partir de diverses sources de données indirectes [Shoemaker & Panchenko, 2007b]. On peut classer ces sources d'information en plusieurs catégories : données génomiques, séquences, structures, données expérimentales indirectes et informations topologiques. Dans cette section, nous détaillons les approches appartenant à ces différentes catégories.

1.5.1 Analyses génomiques

Un certain nombre de méthodes analysent le contexte génomique dans différentes espèces (voir figure 1.10) afin d'inférer des associations fonctionnelles entre des gènes, et donc potentiellement des interactions entre les protéines codées par ces gènes. Nous présentons ici quatre d'entre elles.

La première approche [Pellegrini et al., 1999] est basée sur l'hypothèse que des protéines ayant co-évoluées sont susceptibles de présenter des fonctions similaires. Pour cela, cette approche s'intéresse aux protéines ayant des homologues dans les même organismes. Chaque protéine est représentée par un profil phylogénétique, qui correspond à un vecteur dont la longueur est égale au nombre d'espèces considérées. Ce vecteur binaire indique la présence ou l'absence d'un orthologue du gène associé à la protéine dans l'espèce correspondante.

Un deuxième type d'approches [Enright et al., 1999; Marcotte et al., 1999] consiste à analyser la fusion des gènes. Ces méthodes recherchent ainsi des protéines ayant des homologues qui ont fusionnés en une seule protéine dans un autre génome.

D'autres méthodes [Dandekar et al., 1998; Overbeek et al., 1999] se basent sur l'hypothèse que des gènes présentant une association fonctionnelle restent proches

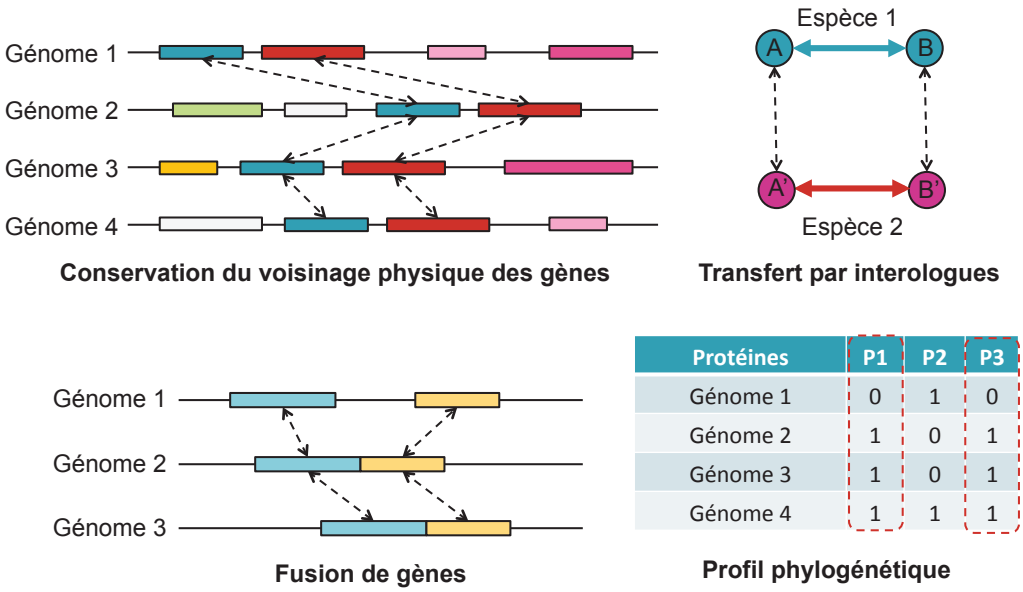

FIGURE 1.10 – **Méthodes basées sur l'analyse génomique**

physiquement entre eux sur le génome. Il est alors possible de prédire des interactions fonctionnelles, et donc des interactions protéine-protéine potentielles, entre des gènes qui sont observés dans un voisinage proche de façon répétitive sur plusieurs génomes. Cependant, les méthodes basées sur la conservation du voisinage physique des gènes peuvent être appliquées uniquement dans le cas des organismes procaryotes.

La méthode du transfert par interologues se base, comme la méthode du profil phylogénétique, sur l'hypothèse que des protéines interagissant ensemble auront tendance à évoluer conjointement. Cette méthode utilise des interactions protéine-protéine connues dans d'autres organismes et cherche à prédire de nouvelles PPI chez un organisme cible. Une interaction protéine-protéine est transférée chez l'espèce cible si les orthologues des deux protéines dans une autre espèce sont connus pour interagir. On parle de transfert par interologues car cette méthode associe relations d'orthologie et interactions [Walhout et al., 2000; Matthews et al., 2001; Michaut et al., 2008]. Cette méthode permet ainsi d'utiliser l'information de l'interactome chez des organismes très étudiés tels que la levure pour inférer des réseaux d'interactions protéine-protéine chez d'autres espèces.

1.5.2 Approches basées sur la séquence d'acides aminés

Plusieurs méthodes de prédiction d'interactions protéine-protéine utilisent la séquence primaire des protéines [Bock & Gough, 2001; Martin et al., 2005; Shen et al., 2007; Guo et al., 2008; Yu et al., 2010]. En plus de la séquence, certaines méthodes considèrent les propriétés physico-chimiques des acides aminés, telles que la charge ou l'hydrophobicité.

D'autres approches utilisent, quant à elles, l'information des domaines présents chez les différentes protéines.

Approches basées sur les domaines

Plusieurs méthodes utilisent l'information des domaines pour prédire de nouvelles interactions protéine-protéine. Celles-ci se basent sur le fait que dans certains cas, les protéines interagissent entre elles par l'intermédiaire d'interactions physiques entre domaines. Cependant, l'inconvénient de ce type d'approche est que le nombre d'interactions domaine-domaine ayant été détectées expérimentalement est limité. Par conséquent, une stratégie couramment utilisée consiste à commencer par identifier des paires de domaines susceptibles d'interagir ensemble à partir d'un ensemble d'interactions protéine-protéine connues. Les interactions prédites entre les domaines sont ensuite utilisées pour prédire de nouvelles interactions entre protéines [Shoemaker & Panchenko, 2007b].

Les méthodes par association s'intéressent à des séquences ou des motifs structuraux caractéristiques permettant de faire la distinction entre les protéines qui interagissent et celles qui n'interagissent pas. Dans le cas particulier des domaines, ces méthodes recherchent les paires de domaines sur-représentées parmi les interactions protéine-protéine connues [Sprinzak & Margalit, 2001; Kim et al., 2002; Ng et al., 2003]. Pour cela, la fréquence de co-occurrence de chaque paire de domaines parmi les paires de protéines interagissant ensemble est calculée.

Une méthode d'estimation du maximum de vraisemblance [Deng et al., 2002] a été par ailleurs proposée afin d'estimer les probabilités d'interaction entre domaines qui sont consistantes avec les interactions entre protéines observées. Cette méthode a ensuite été étendue par Riley et al. [2005], puis par Lee et al. [2006].

Les approches citées précédemment considèrent uniquement des interactions entre deux domaines et supposent que les paires de domaines interagissant ensemble sont indépendantes les unes des autres. Comme les protéines peuvent contenir plusieurs domaines, Han et al. [2003] ont proposé de considérer les interactions entre protéines comme le résultat d'interactions entre des groupes de domaines. La différence entre ces deux types d'approche est illustrée sur la figure 1.11.

Cependant, les approches basées sur les domaines présentent certaines limitations. L'une d'entre elles concerne le fait qu'il n'y pas toujours d'interaction domaine-domaine sous-jacente à une interaction protéine-protéine. Dans certains cas, deux protéines peuvent interagir ensemble par l'intermédiaire d'une interaction entre un domaine et un peptide.

1.5.3 Approches basées sur la structure tertiaire

Les approches basées sur la structure tertiaire des protéines permettent une analyse plus fine des interactions protéine-protéine. Elles sont ainsi capables dans certains cas de déterminer les caractéristiques physiques d'une interaction et le site d'interaction à la surface des protéines. Plusieurs types d'approches utilisent cette

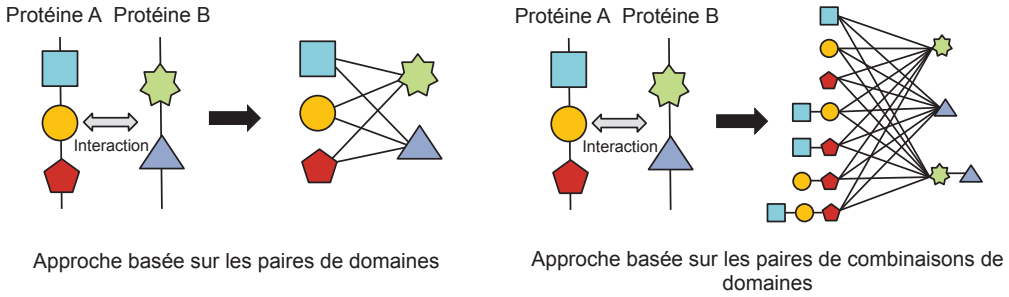

FIGURE 1.11 – **Modèles de prédiction basés sur l'information des domaines.** La figure de gauche illustre les méthodes de prédiction basées sur les interactions entre domaines et celle de droite la méthode [Han et al., 2003] basée sur les interactions entre combinaisons de domaines.

information pour prédire l'existence d'interactions protéine-protéine [Russell et al., 2004].

Docking

Les méthodes de Docking [Smith & Sternberg, 2002] étudient la complémentarité entre des structures connues des protéines. Les méthodes existantes procèdent en deux étapes. La première consiste à générer un grand nombre de conformations possibles pour l'association des deux protéines. Une fonction de score est ensuite utilisée pour classer les différentes conformations. Les méthodes de Docking sont très précises mais présentent l'inconvénient d'être très coûteuses en temps de calcul. De plus, ces approches restent limitées du fait que les structures tertiaires résolues expérimentalement ne sont disponibles que pour une faible proportion des protéines.

Modélisation par homologie

Pour contourner cette limitation, certaines méthodes [Aloy & Russell, 2002; Lu et al., 2003; Hue et al., 2010] utilisent les structures résolues expérimentalement pour modéliser les interactions entre des protéines, pour lesquelles la structure n'est pas connue. L'hypothèse sous-jacente à ces approches est que les protéines présentant une importante homologie de séquence ou de structure interagissent généralement de la même façon. Ces méthodes, contrairement au Docking, peuvent donc être appliquées à l'échelle d'un interactome.

1.5.4 Données expérimentales indirectes et annotations fonctionnelles

Des données expérimentales indirectes, c'est à dire ne correspondant pas directement à des données de PPI, ont été également utilisées pour la prédiction d'interactions. C'est le cas par exemple des données d'expression de gènes et des données

de localisation cellulaire.

Expressions de gènes

Avec le développement des puces à ADN, les expressions de gènes peuvent maintenant être mesurées pour tous les gènes d'un génome. En mesurant l'expression des gènes dans différentes conditions expérimentales, il est possible de détecter des relations fonctionnelles parmi les gènes. Ces relations fonctionnelles peuvent être ensuite utilisées pour inférer des interactions possibles entre les protéines encodées par les gènes considérés. Plusieurs études [Ge et al., 2001; Jansen et al., 2002; Hahn et al., 2005] ont ainsi montré que les gènes codant pour deux protéines interagissant ensemble présentent généralement des profils d'expression similaires.

Localisations cellulaires

Les protéines devant être co-localisées pour pouvoir interagir ensemble, les données de localisation dans les différents compartiments cellulaires ont été utilisées pour prédire des interactions entre protéines.

Gene Ontology

Le Consortium Gene Ontology [Ashburner et al., 2000] fournit un vocabulaire standard structuré permettant de décrire la fonction des gènes et de leurs produits. Gene Ontology (GO) est divisé en trois ontologies : processus biologiques, fonctions moléculaire et composants cellulaires. Les trois ontologies sont représentées par des graphes acycliques dirigés, dans lesquels les nœuds correspondent à des termes et leurs relations sont représentées par des arcs (voir figure 1.12). Dans ces ontologies, chaque nœud peut avoir plusieurs parents ainsi que plusieurs descendants. Les feuilles correspondent aux termes les plus spécifiques et les nœuds proches de la racine aux fonctions principales.

Les produits géniques sont fonctionnellement similaires s'ils ont des fonctions moléculaires comparables ou s'ils sont impliqués dans des processus biologiques similaires. Ces derniers n'ont pas forcément évolué à partir d'un ancêtre commun et par conséquent ne présentent pas nécessairement des similarités de séquence. Les annotations GO capturent l'information fonctionnelle disponible pour un produit génique et par conséquent, deux protéines qui interagissent ensemble sont plus susceptibles d'avoir des annotations GO similaires. Ces annotations ont été ainsi exploitées pour prédire de nouvelles interactions protéine-protéine par plusieurs méthodes [Wu et al., 2006; Jain & Bader, 2010].

1.5.5 Approches topologiques

Les réseaux d'interaction protéine-protéine sont caractérisés par différentes propriétés topologiques. La caractéristique topologique la plus élémentaire d'un nœud

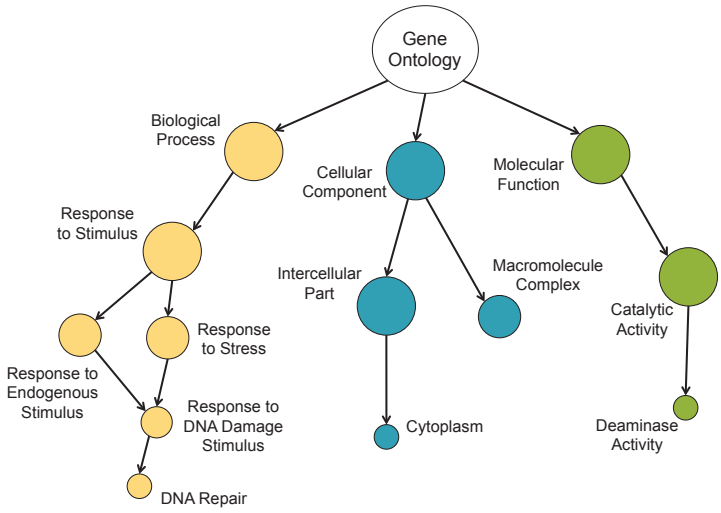

FIGURE 1.12 – **Gene Ontology.** La figure représente les trois ontologies définies par Gene Ontology (processus biologiques, fonctions moléculaires et composants cellulaires), ainsi que les termes les plus généraux de chacune de ces ontologies.

est son degré, c'est à dire le nombre d'interactions existantes entre ce nœud et les autres nœuds dans le réseau. La distribution des degrés d'un réseau est définie par :

$$P(k) = \frac{n_k}{n},$$

où n est le nombre total de nœuds dans le réseau et n_k est le nombre de nœuds de degré k.

Réseaux aléatoires et réseaux libres d'échelle

Le modèle de graphe aléatoire d'Erdős-Renyi [Erdős & Rényi, 1959] est construit en partant de n nœuds initiaux et en connectant chaque paire de nœuds avec une probabilité p. Les degrés des nœuds suivent une distribution de poisson (voir figure 1.13), ce qui indique que la plupart des nœuds ont approximativement le même nombre de liens.

Cependant, la plupart des réseaux biologiques sont libres d'échelle [Barabasi & Oltvai, 2004] (voir figure 1.13), c'est à dire que leur distribution des degrés est proche d'une loi de puissance :

$$P(k) \sim c * k^{-\gamma},$$

où c et γ sont des constantes positives. Une distribution des degrés suivant une loi de puissance signifie que la plupart des nœuds ont un degré faible et qu'il existe un petit nombre de nœuds, appelés *hubs*, qui sont caractérisés par un degré important.

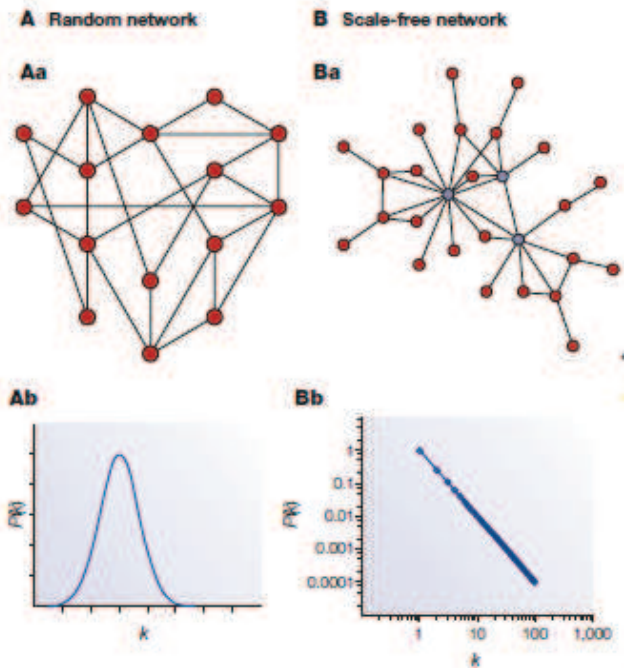

FIGURE 1.13 – **Réseau aléatoire vs. réseau libre d'échelle.** [Barabasi & Oltvai, 2004]
La figure représente un réseau aléatoire de Erdős-Renyi (A) et un réseau libre d'échelle (B). Les distributions de probabilité correspondantes sont représentées respectivement sous chaque réseau.

Effet petit monde

Il a été constaté que beaucoup de réseaux réels vérifient la propriété petit monde [Watts & Strogatz, 1998]. C'est ainsi le cas du web, des réseaux de collaboration scientifique, des réseaux métaboliques ainsi que des réseaux d'interaction protéine-protéine. Les réseaux petit monde présentent deux particularités : les distances moyennes entre les nœuds sont faibles et un grand nombre de structures sont proches de cliques, c'est-à-dire que les voisins d'un nœud donné sont souvent connectés entre eux. Les paramètres qui permettent de caractériser les graphes de type petit monde sont :

- La **longueur caractéristique** d'un chemin, qui correspond à la longueur moyenne des plus courts chemins pour toutes les paires de nœuds possibles ;
- le **diamètre** d'un graphe, qui est la plus grande longueur de chemin possible entre deux nœuds ;
- le **coefficient d'agglomération**, qui permet de mesurer l'inter-connectivité dans le voisinage d'un nœud. Pour chaque nœud, on considère tous ses voisins et on examine tous les arcs possibles entre ces nœuds. Le coefficient d'agglo-

mération est défini comme la proportion de liens existants :

$$C = \frac{2m}{c(c-1)},$$

où c est le nombre de voisins du nœud et m est le nombre de liens existant entre les c nœuds.
Contrairement à la longueur caractéristique et au diamètre, le coefficient d'agglomération permet de différencier les graphes petit monde des graphes aléatoires classiques.

Utilisation des caractéristiques topologiques pour la prédiction de PPI

Plusieurs approches ont exploité les différentes caractéristiques topologiques d'un réseau d'interaction protéine-protéine connu afin d'inférer de nouvelles interactions. Certaines d'entre elles sont basées sur l'hypothèse que deux protéines sont plus susceptibles d'interagir ensemble si elles ont des voisins en commun. C'est le cas de l'approche proposée par Saito et al. [2002] pour détecter les faux positifs parmi des PPI prédites expérimentalement. Scott & Barton [2007] se sont également basés sur cette hypothèse, et l'ont utilisé en combinaison de différentes sources d'information comme les expressions des gènes, les interologues, la localisation, et les domaines. [Goldberg & Roth, 2003] ont, eux aussi, exploité cette propriété, qui peut être déduite de la tendance des réseaux petit monde à présenter de nombreuses structures proches de cliques. Cette propriété a ainsi été utilisée pour évaluer la fiabilité de certaines interactions protéine-protéine prédites par des approches Y2H, ainsi que pour prédire de nouvelles interactions.

D'autres propriétés topologiques ont été exploitées pour la prédiction d'interaction protéine-protéine. Ainsi, Chen et al. [2006a] ont formulé l'hypothèse qu'une interaction protéine-protéine est vraisemblablement plus fiable s'il existe plusieurs chemins alternatifs entre les deux protéines. Lippert et al. [2009] ont pris en compte différentes propriétés topologiques, telles que le degré, la centralité d'intermédiarité (nombre de plus courts chemins passant par un nœud donné), la centralité de proximité (moyenne des longueurs des plus courts chemins entre un nœud et tous les nœuds pouvant être atteints par ce nœud), ainsi que le plus court chemin.

1.5.6 Intégration de données hétérogènes

Plusieurs méthodes ont cherché à intégrer différents types de données [Jansen et al., 2003; Yamanishi et al., 2004; Ben-Hur & Noble, 2005; Rhodes et al., 2005; Qi et al., 2006; Scott & Barton, 2007]. Les sources de données utilisées le plus fréquemment incluent les données d'expression de gène, les données de localisation, les similarités de fonction et l'information des domaines. Ces méthodes de prédiction ont montré que la combinaison de sources de données permettait d'améliorer la prédiction d'interactions protéine-protéine. Qi et al. [2006] ont comparé 17 sources de données réparties en trois catégories : données expérimentales de double hybride et

de spectrométrie de masse, données indirectes (expressions de gènes, interactions protéine-ADN, processus biologiques, localisations, etc. ...) et données basées sur la séquence (information des domaines, fusion de gènes, etc. ...). Les résultats obtenus indiquent que les sources de données les plus pertinentes pour la prédiction d'interactions directes sont les données d'expression de gène, les annotations des trois ontologies de Gene Ontology et les interactions prédites par TAP-MS.

Lu et al. [2005] ont étudié les limites de l'intégration de données pour la prédiction de réseau de PPI en utilisant 16 sources de données. Ils ont observé que l'ajout de sources de données moins pertinentes n'apporte qu'une amélioration marginale par rapport au résultat obtenu en utilisant les sept sources de données les plus prédictives.

Conclusion :
Nous avons introduit dans ce chapitre le contexte des interactions protéine-protéine et présenté l'intérêt de leur identification dans le cas de la protéine CFTR, impliquée dans la mucoviscidose. Nous avons également détaillé les différentes méthodes de détection expérimentale existantes. Nous avons en particulier montré que ces méthodes présentent certaines limitations et que le développement de méthodes de prédiction *in silico* permet de compléter les approches expérimentales. Pour résoudre le problème de la prédiction d'interactions protéine-protéine, les méthodes *in silico* exploitent différentes sources de données présentant un lien potentiel avec la présence d'interaction. Nous avons présenté dans ce chapitre les principales sources d'information qui ont été utilisées dans la littérature.
Dans le chapitre suivant, nous nous intéressons aux méthodes d'apprentissage statistique qui peuvent être utilisées pour la prédiction d'interactions protéine-protéine.

Chapitre 2

Eléments d'apprentissage statistique

L'objectif de ce chapitre consiste à présenter les principaux concepts de l'apprentissage statistique utiles à cette thèse. Nous commençons par introduire les différentes problématiques d'apprentissage existantes dans la section 2.1. Nous décrivons ensuite les méthodes à noyaux et le principe de la régularisation dans la section 2.2. Nous présentons plus particulièrement deux méthodes à noyaux régularisées : la *Kernel Ridge Regression* et les *Séparateurs à Vaste Marge*. Enfin, nous abordons dans la section 2.3 le problème de l'évaluation de l'apprentissage et celui de la sélection de modèle.

2.1 Apprentissage statistique

L'apprentissage statistique désigne un ensemble de méthodes utilisées pour analyser et décrire un phénomène à partir d'une série d'observations, mais également pour inférer le comportement futur de celui-ci. Ces méthodes sont utilisées dans de nombreuses applications comme la reconnaissance de formes, le diagnostic médical, la bio-informatique, les interfaces cerveau-machine, etc. On distingue trois principales problématiques d'apprentissage : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage semi-supervisé. Nous présentons les caractéristiques de chacune d'entre elles dans cette section.

2.1.1 Apprentissage supervisé

Dans le cadre de l'apprentissage supervisé, on considère des observations $\mathbf{x}_i \in \mathcal{X}$ pour $i = 1, \ldots, \ell$ associées à des sorties $y_i \in \mathcal{Y}$, appelées également *étiquettes*. Les observations sont généralement représentées sous la forme d'un vecteur de dimension m. On considère que les ℓ paires $(\mathbf{x}_i, y_i)$ ont été générées d'après une distribution de probabilité jointe P sur $\mathcal{X} \times \mathcal{Y}$ et qu'elles sont indépendantes. L'ensemble $\mathcal{S} = \{(\mathbf{x}_i, y_i)_{i=1,\ldots,\ell}\}$ des paires entrée-sortie est appelé *ensemble d'apprentissage*. Dans le cas où l'espace de sortie $\mathcal{Y}$ est constitué d'un ensemble fini d'étiquettes ou classes ($\mathcal{Y} = \{1, \cdots, q\}$), on parle de classification. Lorsque les sorties y_i peuvent prendre des valeurs continues ($\mathcal{Y} = \mathbb{R}$), on parle de régression.

Ces deux problèmes peuvent être considérés comme des problèmes d'approximation de fonction. L'objectif de l'apprentissage supervisé consiste ainsi à utiliser l'ensemble d'apprentissage $\mathcal{S}$ afin d'apprendre une fonction $f : \mathcal{X} \to \mathcal{Y}$, qui soit

capable de prédire correctement la sortie y associée à une nouvelle entrée $\mathbf{x}$. Afin de mesurer l'erreur entre la prédiction $f(\mathbf{x})$ et la vraie sortie y, une fonction de perte $\mathcal{L}$ est définie pour chaque exemple :

$$\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+.$$

Parmi les fonctions de perte usuelles, on peut citer la fonction de perte charnière, utilisée en classification :

$$\mathcal{L}(f(\mathbf{x}), y) = max(1 - yf(\mathbf{x}), 0),$$

et la fonction de perte des moindres carrés utilisée en régression. Celle-ci correspond au carré de la différence entre l'observation et la prédiction :

$$\mathcal{L}(f(\mathbf{x}), y) = (y - f(\mathbf{x}))^2.$$

Ces deux fonctions de perte sont illustrées sur la figure 2.1.

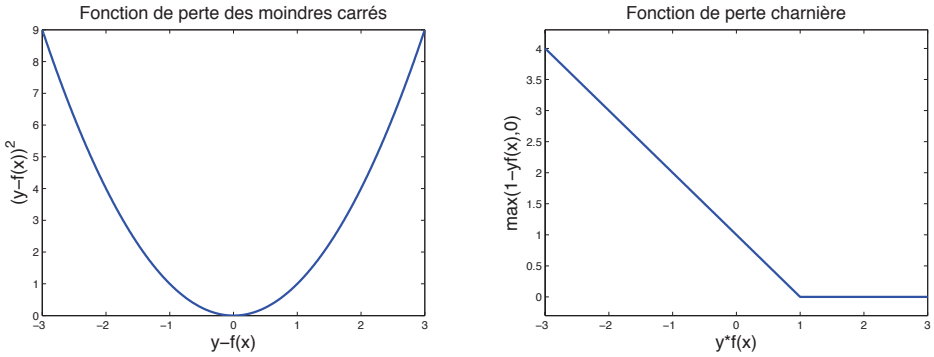

FIGURE 2.1 – **Fonctions de perte** : fonction de perte des moindres carrés (à gauche) et fonction de perte charnière (à droite)

Etant donnés un espace de fonctions $\mathcal{H}$, appelé espace d'hypothèses, et une fonction de perte $\mathcal{L}$, l'apprentissage supervisé consiste à trouver la fonction $f : \mathcal{X} \to \mathcal{Y}$ qui minimise le risque réel R :

$$f^* = \arg\min_{f \in \mathcal{H}} R(f),$$

où

$$R(f) = \mathbb{E}[\mathcal{L}(f(\mathbf{x}), y)] = \int_{\mathcal{X} \times \mathcal{Y}} \mathcal{L}(f(\mathbf{x}), y) d\mathrm{P}(\mathbf{x}, y).$$

Cependant, la distribution de probabilité P n'est pas connue explicitement. On utilise donc en pratique l'ensemble d'apprentissage $\mathcal{S}$ pour estimer le risque réel

par le *risque empirique* :

$$R_{emp}(f) = \frac{1}{\ell} \sum_{i=1}^{\ell} \mathcal{L}(f(\mathbf{x}_i), y_i).$$

Le principe de minimisation du risque empirique consiste à choisir la fonction $f \in \mathcal{H}$ qui minimise $R_{emp}(f)$:

$$f_{\mathcal{S}}^* = \arg\min_{f \in \mathcal{H}} R_{emp}(f).$$

Différentes théories ont pointé le dilemme biais/variance du principe de minimisation du risque empirique lorsque le nombre de données est limité (voir figure 2.2). Minimiser à la fois le biais et la variance est contradictoire. Il faut donc trouver un compromis entre l'adéquation du modèle aux données et la complexité du modèle, qui fait intervenir la dimension de $\mathcal{H}$. Vapnik a proposé de minimiser le risque structurel [Vapnik & Chervonenkis, 1974]. Tikhonov & Arsenin [1977] ont transformé ce problème mal-posé en imposant des contraintes de régularité à la fonction recherchée. L'angle d'attaque choisi dans cette thèse est celui de la régularisation.

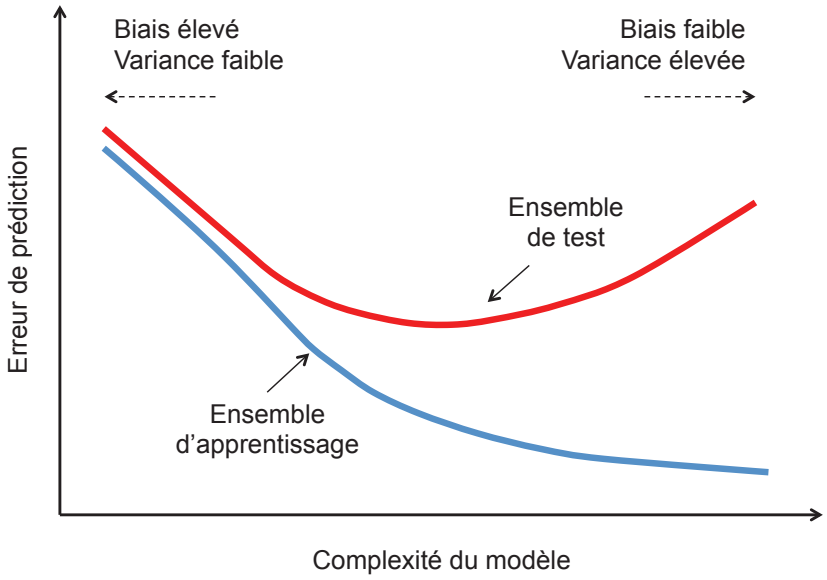

FIGURE 2.2 – **Illustration du dilemme biais-variance**. Lorsque la complexité du modèle augmente, l'erreur commise sur l'ensemble d'apprentissage diminue. Par ailleurs, l'erreur de généralisation mesurée sur un ensemble de test indépendant de l'ensemble d'apprentissage diminue initialement du fait que le biais soit réduit, mais à partir d'un certain point, cette erreur augmente en raison de l'augmentation de la variance.

Régularisation La minimisation du risque empirique peut conduire à des instabilités numériques et à de mauvaises performances en généralisation. Le principe de la

régularisation consiste à restreindre l'espace des hypothèses. La régularisation de Tikhonov [Tikhonov & Arsenin, 1977] contraint indirectement cet espace en ajoutant un terme de pénalisation au problème d'apprentissage :

$$R_{reg}(f) = \underbrace{R_{emp}(f)}_{\text{risque empirique}} + \lambda \underbrace{\text{pen}(f)}_{\text{terme de pénalisation}},$$

où $\lambda \geq 0$, appelé paramètre de régularisation, contrôle le compromis entre l'adéquation du modèle aux données et la complexité du modèle. La régularisation permet ainsi de limiter le sur-apprentissage. Pour contrôler la complexité du modèle, on utilise usuellement la norme de la fonction dans $\mathcal{H}$. La régularisation de Tikhonov peut alors s'écrire de la façon suivante :

$$R_{reg}(f) = \sum_{i=1}^{\ell} \mathcal{L}(f(\mathbf{x}_i), y_i) + \lambda \Omega(\|f\|_{\mathcal{H}}).$$

2.1.2 Apprentissage non supervisé

L'apprentissage non supervisé représente une autre problématique d'apprentissage statistique, dans laquelle les étiquettes des données en entrée ne sont pas connues durant le processus d'apprentissage. L'objectif consiste alors à identifier une structure sous-jacente parmi les données. Un des problèmes étudiés dans ce cadre consiste à utiliser les attributs des données pour les regrouper entre elles de façon à ce que les données similaires se retrouvent dans le même groupe, et les données dissimilaires dans des groupes différents. Un des algorithmes les plus connus pour résoudre ce problème de partitionnement des données est celui des k-moyennes. Un autre problème concerne la réduction de la dimension dans le cas de données de très grandes dimensions.

2.1.3 Apprentissage semi-supervisé et transductif

L'apprentissage semi-supervisé est une problématique d'apprentissage statistique qui utilise lors de l'apprentissage à la fois les données étiquetées, généralement en petite quantité car coûteuses à obtenir, et un grand nombre de données non étiquetées. L'apprentissage semi-supervisé se situe à la frontière entre l'apprentissage non supervisé et l'apprentissage supervisé. Il peut être vu comme de l'apprentissage non supervisé avec des contraintes ou de l'apprentissage supervisé avec des informations supplémentaires sur la distribution des exemples. Il apparaît que les données non étiquetées, lorsqu'elles sont utilisées en conjonction avec une petite quantité de données étiquetées, peuvent améliorer sensiblement la précision de l'apprentissage.

Dans le cas particulier de l'apprentissage transductif, on dispose d'un ensemble d'exemples étiquetés et d'un ensemble d'exemples non étiquetés qui sont connus a priori. L'idée de la transduction est de calculer uniquement les prédictions pour

les points de test, correspondant aux points non étiquetés, contrairement à l'apprentissage inductif qui consiste à trouver une fonction de prédiction générale. La transduction a été introduite par Vapnik [1998], qui pensait qu'elle était préférable à l'induction du fait qu'elle ne requiert pas la résolution d'un problème général (l'inférence d'une fonction) pour résoudre un problème plus spécifique (en l'occurrence, la prédiction des valeurs de sortie pour de nouveaux exemples). Les exemples de test étant connus a priori, l'apprentissage transductif permet d'utiliser l'information contenue dans les attributs de ces exemples.

Les méthodes d'apprentissage semi-supervisé sont généralement basées sur une des trois hypothèses suivantes [Chapelle et al., 2006] :

Hypothèse de régularité Si deux points $\mathbf{x}_1$ et $\mathbf{x}_2$ sont proches dans une région dense alors leurs sorties respectives y_1 et y_2 doivent l'être également. Par contre, si les deux points sont séparés par une région de faible densité, les sorties correspondantes doivent être éloignées l'une de l'autre.

Hypothèse de regroupement Si deux points $\mathbf{x}_1$ et $\mathbf{x}_2$ appartiennent à un même groupe, alors il est vraisemblable qu'ils soient associés à la même étiquette, ou classe. Cette hypothèse n'implique pas que chaque classe corresponde à un unique groupe, mais plutôt que des objets n'appartenant pas à la même classe ne soient pas regroupés dans un même groupe. Cette hypothèse peut se reformuler en disant que la frontière de décision doit se trouver dans une région de faible densité de $\mathcal{X}$ [Chapelle & Zien, 2005]. Les séparateurs à vaste marge transductifs [Vapnik, 1998; Joachims, 1999], par exemple, implémentent l'hypothèse de regroupement en utilisant les données étiquetées et non étiquetées pour rechercher un hyperplan séparateur de marge maximale passant dans des régions de faible densité.

Hypothèse de variété Les données sont supposées appartenir à une variété de dimension inférieure à celle de $\mathcal{X}$. Il s'agit d'un espace topologique, qui, localement, peut être approché par un espace euclidien de même dimension. Pour estimer la variété, un graphe est construit à partir de données étiquetées et non étiquetées. Chaque point $\mathbf{x}_i$ correspond à un nœud dans ce graphe. Deux points $\mathbf{x}_i$ et $\mathbf{x}_j$ sont reliés par un arc s'ils sont proches, par exemple s'ils font respectivement partie des k-plus proches voisins de l'autre. Chaque arrête entre $\mathbf{x}_i$ et $\mathbf{x}_j$ est pondérée par un poids w_{ij}, qui peut être égal à 1, ou bien correspondre à une valeur de similarité. Ce graphe permet ainsi de prendre en compte la structure de la variété en capturant les similarités locales existantes entre les données (voir figure 2.3).

Plusieurs méthodes semi-supervisées ont utilisé la structure du graphe construit pour propager les étiquettes de l'ensemble d'apprentissage aux autres données [Szummer & Jaakkola, 2002; Zhu & Ghahramani, 2002; Zhou et al., 2004]. D'autres approches ont considéré un terme de régularisation basé sur le Laplacien du graphe [Belkin & Niyogi, 2003, 2004; Zhou et al., 2004].

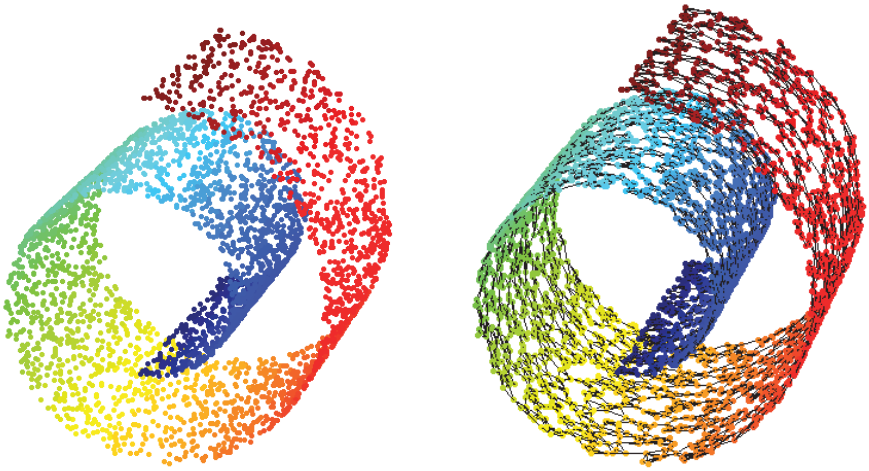

FIGURE 2.3 – **Hypothèse de variété** : la figure de gauche représente les données supposées appartenir à une variété de dimension inférieure à celle de $\mathcal{X}$. La figure de droite représente le graphe construit en utilisant les 4-plus proches voisins pour estimer les similarités locales existantes dans la variété.

2.2 Méthodes à noyaux

Les méthodes à noyaux sont des méthodes très populaires en apprentissage. Elles permettent de définir des fonctions de décision non linéaires tout en utilisant des méthodes linéaires à la base. L'évaluation d'une fonction noyau revient à calculer un produit scalaire dans un espace de redescription des données, généralement de grande dimension. Dans cet espace, les méthodes linéaires peuvent être utilisées.

Une fonction noyau permet également de comparer différents types d'objets, comme par exemple des chaînes de caractères, des images, des signaux audio ou des séries temporelles, et de travailler avec des objets structurés à travers une représentation vectorielle implicite.

2.2.1 Cadre fonctionnel

Nous commençons par présenter les espaces de Hilbert à noyau reproduisant (RKHS, pour *Reproducing Kernel Hilbert Space*), qui peuvent être utilisés comme espaces d'hypothèses pour faire de l'estimation de fonction. Pour cela, nous définissons tout d'abord les notions d'espace de Hilbert et de noyau reproduisant.

Définition 1 : Espace de Hilbert
Un espace de Hilbert $\mathcal{H}$ est un espace vectoriel muni d'un produit scalaire $\langle \cdot, \cdot \rangle_{\mathcal{H}}$, dont on tire une norme, et complet, c'est à dire que toute suite de Cauchy de $\mathcal{H}$ a une limite dans $\mathcal{H}$.

Définition 2 : Noyau reproduisant
Soit $\mathcal{H}$ un espace de Hilbert de fonctions de $\mathcal{X}$ dans $\mathbb{R}$. La fonction $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est appelée noyau reproduisant de $\mathcal{H}$ si :

1. *Pour tout élément $\mathbf{x} \in \mathcal{X}$, la fonction $k(\mathbf{x}, \cdot) : \mathbf{t} \to k(\mathbf{x}, \mathbf{t})$ appartient à $\mathcal{H}$.*
2. *Pour tout $\mathbf{x} \in \mathcal{X}$, $f \in \mathcal{H}$, la propriété reproduisante est vérifiée :*
$$f(\mathbf{x}) = \langle f, k(\mathbf{x}, \cdot)\rangle_{\mathcal{H}}.$$

En particulier, pour toute paire $(\mathbf{x}, \mathbf{x}') \in \mathcal{X} \times \mathcal{X}$, $\langle k(\mathbf{x}, \cdot), k(\mathbf{x}', \cdot)\rangle_{\mathcal{H}} = k(\mathbf{x}, \mathbf{x}')$. Ainsi l'évaluation du noyau en deux points de $\mathcal{X}$ peut s'écrire sous la forme d'un produit scalaire.

Si un noyau reproduisant k existe, $\mathcal{H}$ est appelé *espace de Hilbert à noyau reproduisant* (RKHS).

Nous nous plaçons dans un RKHS pour minimiser le risque empirique régularisé introduit dans la section 2.1. L'intérêt de se placer dans ce cadre est que la fonction minimisant $R_{reg}(f)$ admet une forme explicite qui est donnée par le théorème de représentation [Kimeldorf & Wahba, 1971]. Celui a d'abord été prouvé dans le cas de la fonction de perte des moindres carrés, puis a été étendu à des fonctions de perte plus générales.

Théorème 1 *[Kimeldorf & Wahba, 1971] Soient $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ un noyau reproduisant, $\mathcal{S} = \{\mathbf{x}_i, y_i\}_{i=1}^{\ell} \subseteq \mathcal{X} \times \mathcal{Y}$ un ensemble d'apprentissage et $\mathcal{L}$ une fonction de perte quelconque. Soit $\mathcal{H}$ le RKHS induit par le noyau reproduisant k et $\Omega : [0, \infty) \to \mathbb{R}$ une fonction strictement croissante. Alors toute fonction $f \in \mathcal{H}$ minimisant le risque empirique régularisé :*

$$\mathcal{J}(f) = \sum_{i=1}^{\ell} \mathcal{L}(f(\mathbf{x}_i), y_i) + \lambda\Omega(\|f\|_{\mathcal{H}}),$$

admet une représentation de la forme :

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{\ell} c_i k(\mathbf{x}_i, \mathbf{x}),\ c_1, \ldots, c_\ell \in \mathbb{R}.$$

Le théorème de représentation établit ainsi que l'hypothèse minimisant le risque empirique régularisé se trouve dans le sous-espace vectoriel des fonctions engendrées par les combinaisons linéaires de la fonction noyau appliquée sur les données d'apprentissage. En remplaçant la forme prise par la solution dans le problème d'optimisation, on peut convertir le problème de recherche dans un espace $\mathcal{H}$ de dimension infinie à un problème de recherche de ℓ coefficients c_i.

2.2.2 Noyaux semi-défini positifs

Une façon possible de définir un noyau reproduisant est de considérer un noyau semi-défini positif.

Définition 3 : Noyau semi-défini positif
Un noyau $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ semi-défini positif est une fonction symétrique

$$\forall(\mathbf{x}, \mathbf{x}') \in \mathcal{X} \times \mathcal{X}, k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}', \mathbf{x}),$$

qui vérifie pour tout $n \in \mathbb{N}$:

$$\forall(c_1, \ldots, c_n) \in \mathbb{R}^n, \forall(\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathcal{X}^n, \sum_{i,j=1}^{n} c_i c_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0.$$

On appelle **matrice de Gram** d'un noyau k par rapport à un ensemble d'entrées $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}$ la matrice K de taille $n \times n$ définie de la façon suivante :

$$\forall(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{X} \times \mathcal{X}, \; K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j), \; 1 \leq i, j \leq n.$$

Le théorème suivant fait le lien entre les noyaux semi-défini positifs et les espaces de Hilbert à noyau reproduisant.

Théorème 2 *[Aronszajn, 1950] Soit $\mathcal{X}$ un ensemble. Pour tout noyau semi-défini positif $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, il existe un unique RKHS $\mathcal{H}$ admettant k comme noyau reproduisant.*

Soit k un noyau semi-défini positif. Afin de construire un RKHS admettant k comme noyau reproduisant, on commence par définir l'espace $\mathcal{H}_0$ construit à partir de l'ensemble des fonctions de la forme :

$$f(\cdot) = \sum_{i=1}^{m} \alpha_i k(\mathbf{x}_i, \cdot), \text{ où } m \in \mathbb{N}, \alpha_i \in \mathbb{R} \text{ et } \mathbf{x}_i \in \mathcal{X}.$$

Le produit scalaire sur $\mathcal{H}_0$ entre les fonctions $f(\cdot) = \sum_{i=1}^{m} \alpha_i k(\mathbf{x}_i, \cdot)$ et $g(\cdot) = \sum_{j=1}^{m'} \beta_j k(\mathbf{x}'_j, \cdot)$, avec $m, m' \in \mathbb{N}, \alpha_i, \beta_j \in \mathbb{R}$, $\mathbf{x}_i, \mathbf{x}'_j \in \mathcal{X}$, est défini par :

$$\langle f, g \rangle_{\mathcal{H}_0} = \sum_{i=1}^{m} \sum_{j=1}^{m'} \alpha_i \beta_j k(\mathbf{x}_i, \mathbf{x}'_j).$$

L'espace de Hilbert $\mathcal{H}$ admettant k comme noyau reproduisant est obtenu en complétant l'espace préhilbertien $\mathcal{H}_0$ par les limites des suites de Cauchy.

2.2.3 Astuce du noyau

Par ailleurs, il a été montré que tout noyau semi-défini positif $k(\mathbf{x}, \mathbf{x}')$ peut s'écrire comme un produit scalaire dans un espace de Hilbert [Aronszajn, 1950].

Théorème 3 *[Aronszajn, 1950] k est un noyau semi-défini positif si et seulement si il existe un espace de Hilbert $\mathcal{H}$ muni du produit scalaire $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ et une fonction*

$\phi : \mathcal{X} \to \mathcal{H}$ *tels que :*

$$\forall(\mathbf{x}, \mathbf{x}') \in \mathcal{X} \times \mathcal{X}, k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}}.$$

La transformation ϕ peut être définie par l'intermédiaire du noyau k :

$$\begin{aligned} \phi : \mathcal{X} &\to \mathcal{H} \\ x &\mapsto k(x, \cdot). \end{aligned}$$

Par conséquent, tout algorithme qui utilise uniquement le produit scalaire entre les vecteurs d'entrée peut être effectué implicitement dans un espace de Hilbert, appelé *espace caractéristique*, en remplaçant chaque produit scalaire par l'évaluation d'un noyau. L'astuce du noyau permet ainsi d'opérer dans un espace de plus grande dimension sans avoir à calculer explicitement les coordonnées des données dans cet espace.

Dans de nombreux problèmes d'apprentissage, il n'est pas possible de séparer les différentes classes de façon linéaire. Les méthodes à noyaux, dont l'exemple le plus connu est celui des Séparateurs à Vaste Marge (SVM) [Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1995], fournissent une approche permettant de rendre non linéaires des méthodes linéaires. Ainsi cette approche permet de transformer les données dans un espace de plus grande dimension, dans lequel les données pourront être séparables linéairement (voir figure 2.4).

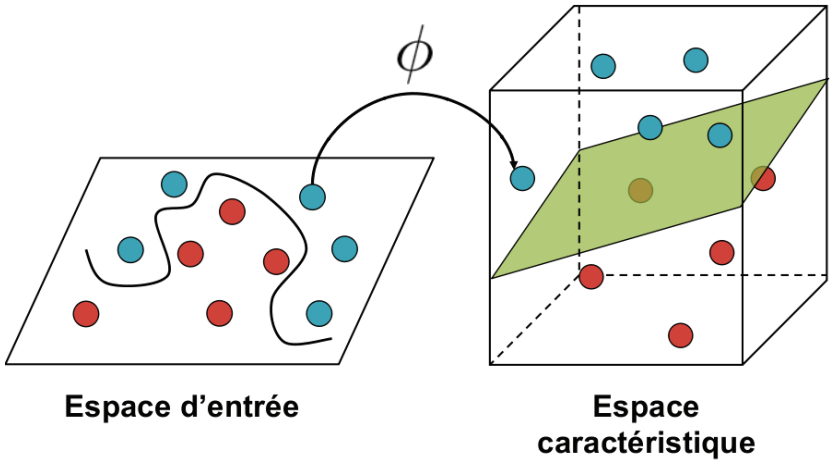

FIGURE 2.4 – **Astuce du noyau**. Cette astuce permet de résoudre un problème non linéaire en transformant l'espace de représentation des données d'entrée en un espace de plus grande dimension, appelé espace caractéristique, et en utilisant un classifieur linéaire dans cet espace.

L'exemple le plus simple de fonction noyau est le noyau linéaire : $k(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$. Parmi les noyaux couramment utilisés, on trouve également :

– le noyau polynomial : $k(\mathbf{x}, \mathbf{x}') = (\langle \mathbf{x}, \mathbf{x}' \rangle + c)^d$,

– le noyau gaussien : $k(\mathbf{x}, \mathbf{x}') = exp(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\sigma^2})$.

Les fonctions noyaux peuvent opérer sur des données non vectorielles comme les données de séquences, de graphes, de textes, ou encore d'images. De plus, les méthodes à noyaux permettent de combiner facilement des sources de données hétérogènes, du fait que la somme de deux noyaux est elle-même un noyau.

Nous présentons dans la suite deux méthodes à noyaux : la *Kernel Ridge Regression* et les *Séparateurs à Vaste Marge*.

2.2.4 Kernel Ridge Regression

La régression Ridge [Hoerl & Kennard, 1970; Tikhonov & Arsenin, 1977] est une méthode de régression linéaire, qui minimise la fonction de perte des moindres carrés pénalisée par la norme ℓ_2 de la fonction. Nous présentons ici la *Kernel Ridge Regression*, qui permet de définir des modèles de régression non linéaires en se plaçant dans le cadre des méthodes à noyaux.

Soit k un noyau semi-défini positif et $\mathcal{H}$ l'espace de Hilbert admettant k comme noyau reproduisant. Le problème d'optimisation correspondant à la Kernel Ridge Regression est le suivant :

$$\arg\min_{f \in \mathcal{H}} \mathcal{J}(f),$$

où

$$\mathcal{J}(f) = \sum_{i=1}^{\ell} (y_i - f(\mathbf{x}_i))^2 + \lambda \|f\|_{\mathcal{H}}^2. \tag{2.1}$$

D'après le théorème de représentation, on sait que la solution peut s'écrire sous la forme $\hat{f}(\mathbf{x}) = \sum_{i=1}^{\ell} c_i k(\mathbf{x}_i, \mathbf{x})$. En remplaçant f par cette expression dans le problème ci-dessus, on obtient :

$$\mathcal{J}(\mathbf{c}) = \|\mathbf{y} - K\mathbf{c}\|_{\mathcal{Y}}^2 + \lambda \mathbf{c}^T K \mathbf{c},$$

où $\mathbf{y} = (y_1, \ldots, y_\ell)^T$, $\mathbf{c} = (c_1, \ldots, c_\ell)^T$, et K est la matrice de Gram associée au noyau k sur l'ensemble d'apprentissage.
Pour obtenir le vecteur $\mathbf{c}$ minimisant $\mathcal{J}(\mathbf{c})$, on annule la dérivée partielle de $\mathcal{J}$ par rapport à $\mathbf{c}$:

$$\begin{aligned} \frac{\partial \mathcal{J}(\mathbf{c})}{\partial \mathbf{c}} = 0 &\Rightarrow -2K^T\mathbf{y} + 2K^T K\mathbf{c} + 2\lambda K\mathbf{c} = 0 \\ &\Rightarrow (K + \lambda I_\ell)\mathbf{c} = \mathbf{y}, \end{aligned}$$

où I_ℓ est la matrice identité de taille ℓ.

On obtient ainsi la solution suivante :

$$\mathbf{c} = (K + \lambda I_\ell)^{-1}\mathbf{y}.$$

2.2.5 Séparateurs à vaste marge

Les machines à vecteurs de support ou séparateurs à vaste marge (SVM) [Boser et al., 1992; Vapnik, 1995] forment un ensemble de méthodes d'apprentissage supervisé, utilisées à l'origine pour la résolution de problèmes de classification binaire ($y_i \in \{+1, -1\}$). Ils peuvent être introduits par la notion de maximisation d'une marge géométrique (voir figure 2.5). Nous les présentons ici dans le cadre de la théorie des RKHS.

Cas séparable : Les séparateurs à vaste marge utilisent la fonction de perte charnière et un terme de pénalisation correspondant à la norme dans le RKHS $\mathcal{H}$. Lorsque les données sont séparables dans l'espace caractéristique, la recherche de l'hyperplan optimal revient à résoudre le problème d'optimisation suivant :

$$\underset{f\in\mathcal{H}}{\operatorname{argmin}} \sum_{i=1}^{\ell} \max\left(1 - y_i f(\mathbf{x}_i), 0\right) + \lambda \|f\|_{\mathcal{H}}^2.$$

Dans la littérature concernant les SVM, le problème d'optimisation est souvent écrit sous la forme suivante :

$$\underset{f\in\mathcal{H}}{\operatorname{argmin}}\, C \sum_{i=1}^{\ell} \max\left(1 - y_i f(\mathbf{x}_i), 0\right) + \frac{1}{2}\|f\|_{\mathcal{H}}^2.$$

En utilisant $C = \frac{1}{2\lambda}$, les deux formulations sont équivalentes.

Ce problème est non différentiable. Celui-ci est donc généralement considéré dans sa formulation primale :

$$\begin{cases} \underset{f\in\mathcal{H}}{\operatorname{argmin}} & \lambda\|f\|_{\mathcal{H}}^2 \\ \text{avec} & y_i f(\mathbf{x}_i) \geq 1, \quad i = 1, \ldots, \ell. \end{cases}$$

Cas non séparable : Dans certains cas, les données ne sont pas linéairement séparables dans l'espace caractéristique. Il se peut par exemple que les données soient bruitées ou qu'elles présentent des erreurs. De plus, il n'est pas toujours souhaitable de classifier correctement toutes les données car cela peut conduire à un sur-apprentissage des données.

Cortes & Vapnik [1995] ont proposé la méthode des SVM à *marge douce*, dont le principe consiste à relâcher les contraintes en introduisant des variables d'écart $\xi_i, i = 1, \ldots, \ell$. La formulation primale de ces SVM est la suivante :

$$\begin{cases} \underset{f\in\mathcal{H}}{\operatorname{argmin}} & \lambda\|f\|_{\mathcal{H}}^2 + \sum_{i=1}^{\ell} \xi_i \\ \text{avec} & y_i f(\mathbf{x}_i) \geq 1 - \xi_i \quad i = 1, \ldots, \ell \\ & \xi_i \geq 0 \quad i = 1, \ldots, \ell \end{cases} \tag{2.2}$$

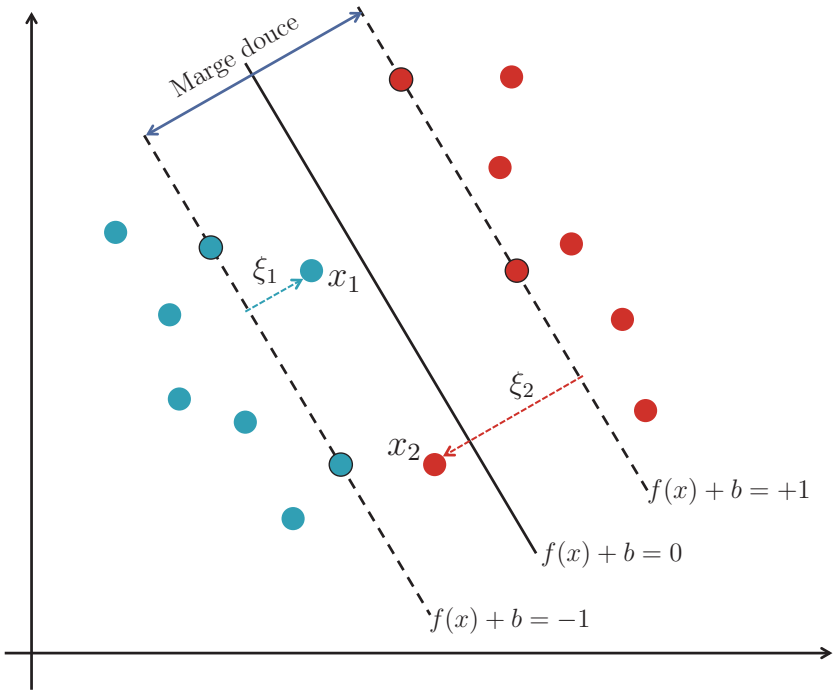

FIGURE 2.5 – **SVM à marge douce.** Un SVM peut être vu comme un problème de recherche d'un hyperplan d'équation $f(\mathbf{x}) + b = 0$ permettant de séparer les exemples positifs des exemples négatifs. Pour cela, la notion de marge géométrique est introduite. Celle-ci représente la distance entre la frontière de séparation et les points les plus proches, appelés *vecteurs de support* (points avec un contour noir sur la figure). L'hyperplan séparateur est choisi de manière à maximiser la marge afin de mieux généraliser à de nouveaux exemples. Dans le cas de données non séparables, des variables d'écart ξ_i sont introduites pour pouvoir tolérer des erreurs de classification.

D'après le théorème de représentation, la solution minimisant le problème d'optimisation s'écrit : $f(\mathbf{x}) = \sum_{i=1}^{\ell} c_i k(\mathbf{x}, \mathbf{x}_i)$.

En dérivant le Lagrangien associé au problème 2.2, et en annulant la dérivée obtenue, on obtient la formulation duale suivante :

$$\begin{cases} \underset{\boldsymbol{\alpha} \in \mathbb{R}^{\ell}}{\text{argmax}} & \sum_{i=1}^{\ell} \alpha_i - \frac{1}{4\lambda} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{avec} & 0 \leq \alpha_i \leq 1, \quad i = 1, \ldots, \ell, \end{cases}$$

où les coefficients α_i sont les multiplicateurs de Lagrange. La fonction de classification s'exprime sous la forme suivante :

$$\forall \mathbf{x} \in \mathcal{X}, f(\mathbf{x}) = \frac{1}{2\lambda} \sum_{i=1}^{\ell} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}).$$

2.2.6 Régularisation semi-supervisée

Dans le cas de l'apprentissage semi-supervisé, on considère un ensemble de ℓ exemples étiquetés $\{\mathbf{x}_i, y_i\}_{i=1}^{\ell} \subseteq \mathcal{X} \times \mathcal{Y}$ et un ensemble supplémentaire de n exemples non étiquetés $\{\mathbf{x}_i\}_{i=\ell+1}^{\ell+n} \subseteq \mathcal{X}$. On se place dans le cas des approches basées sur l'hypothèse de variété, et on appelle W la matrice d'adjacence du graphe construit de sorte à capturer les similarités locales existantes entre les données sur la variété.

Un terme de pénalisation a été introduit dans le cadre de l'apprentissage semi-supervisé afin de contraindre la régularité de la fonction sur la variété [Zhu & Ghahramani, 2002; Belkin et al., 2006]. Ainsi, lorsque deux points $\mathbf{x}_i$ et $\mathbf{x}_j$ sont similaires d'après W, ce terme de régularisation contraint les sorties respectives $f(\mathbf{x}_i)$ et $f(\mathbf{x}_j)$ à être proches et inversement, ce terme contraint les sorties à être éloignées lorsque la similarité entre $\mathbf{x}_i$ et $\mathbf{x}_j$ est faible :

$$\sum_{i,j=1}^{\ell+n} W_{ij}(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 = 2\mathbf{f}^T L\mathbf{f},$$

où $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_{\ell+n})]^T$ et $L = D - W$ est le Laplacien associé à la matrice W, avec D la matrice diagonale de terme général : $D_{ii} = \sum_{j=1}^{\ell+n} W_{ij}$.

Le problème d'optimisation régularisé dans le cas de l'apprentissage semi-supervisé est le suivant :

$$\mathcal{J}(f) = \sum_{i=1}^{\ell} \mathcal{L}(f(\mathbf{x}_i), y_i) + \lambda_1 \|f\|_{\mathcal{H}}^2 + 2\lambda_2 \mathbf{f}^T L\mathbf{f}, \tag{2.3}$$

où $\lambda_1, \lambda_2 \geq 0$ sont deux hyperparamètres.

Comme dans le cas supervisé, la forme de la fonction minimisant le problème d'optimisation peut être donnée par un théorème de représentation. Cependant dans le cas semi-supervisé, la forme de la solution donnée par le théorème de représentation s'écrit comme une combinaison linéaire de fonctions noyaux appliquées aux points correspondant à l'union des exemples étiquetés et des exemples non étiquetés.

Théorème 4 *[Belkin et al., 2006] La fonction minimisant le problème d'optimisation 2.3 admet une représentation de la forme :*

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{\ell+n} c_i k(\mathbf{x}_i, \mathbf{x}),\ c_i \in \mathbb{R}.$$

Au lieu du Laplacien, il est également possible d'utiliser le Laplacien normalisé

$$L_{norm} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} W D^{-1/2},$$

ce qui revient à considérer le terme de régularisation suivant :

$$f^T L_{norm} f = \sum_{i,j=1}^{\ell+n} W_{ij} \left(\frac{f(\mathbf{x}_i)}{\sqrt{D_{ii}}} - \frac{f(\mathbf{x}_j)}{\sqrt{D_{jj}}} \right)^2 .$$

Les itérations du Laplacien L^k sont d'autres choix possibles pour la contrainte de continuité, ainsi que l'exponentielle matricielle $exp(-\beta L)$ où :

$$exp(-\beta L) = \sum_{k=0}^{\infty} \frac{1}{k!} (-1)^k \beta^k L^k.$$

Comme précédemment, nous présentons les modèles obtenus dans le cas de la Kernel Ridge Regression et des séparateurs à vaste marge.

Kernel Ridge Regression

Le problème d'optimisation à résoudre pour la fonction de perte des moindres carrés dans le cas de l'apprentissage semi-supervisé est le suivant :

$$\mathcal{J}(f) = \sum_{i=1}^{\ell} (y_i - f(\mathbf{x}_i))^2 + \lambda_1 \|f\|_{\mathcal{H}}^2 + 2\lambda_2 \, \mathbf{f}^T L \mathbf{f}. \tag{2.4}$$

En remplaçant f par la forme obtenue avec le théorème de représentation dans l'équation ci-dessus, on obtient :

$$\mathcal{J}(f) = \|J^T \mathbf{y} - J^T J K \mathbf{c}\|^2 + \lambda_1 \, \mathbf{c}^T K \mathbf{c} + 2\lambda_2 \, \mathbf{c}^T K L K \mathbf{c},$$

où K est la matrice de Gram du noyau k sur les données étiquetées et non étiquetées, et $J = [I_\ell, 0_{\ell \times n}]$ est une matrice de taille $\ell \times (\ell + n)$ qui contient une matrice identité de dimension $\ell \times \ell$ à gauche et une matrice nulle de dimension $\ell \times n$ à droite.

On annule la dérivée partielle de $\mathcal{J}(f)$ par rapport à $\mathbf{c}$:

$$\begin{aligned} \frac{\partial \mathcal{J}(f)}{\partial \mathbf{c}} = 0 &\Rightarrow -2K^T J^T \mathbf{y} + 2K^T J^T J K \mathbf{c} + 2\lambda_1 K \mathbf{c} + 4\lambda_2 K L K \mathbf{c} = 0 \\ &\Rightarrow (\lambda_1 I_{\ell+n} + J^T J K + 2\lambda_2 L K) \mathbf{c} = J^T \mathbf{y}, \end{aligned}$$

ce qui permet d'obtenir la solution suivante pour le vecteur de coefficients $\mathbf{c}$:

$$\mathbf{c} = (\lambda_1 I_{\ell+n} + J^T J K + 2\lambda_2 L K)^{-1} J^T \mathbf{y}.$$

Séparateurs à vaste marge

La formulation primale des SVM avec le terme de régularisation pour l'apprentissage semi-supervisé est la suivante :

$$\begin{cases} \underset{f\in\mathcal{H}}{\text{argmin}} & \lambda_1\|f\|^2 + 2\lambda_2 \mathbf{f}^T L\mathbf{f} + \sum_{i=1}^{\ell}\xi_i \\ \text{s.t.} & y_i f(\mathbf{x}_i) \geq 1-\xi_i, \quad i=1,\ldots,\ell \\ & \xi_i \geq 0, \quad i=1,\ldots,\ell \end{cases}$$

En utilisant la solution donnée par le théorème de représentation, la formulation primale s'écrit sous la forme suivante :

$$\begin{cases} \underset{\mathbf{c}\in\mathbb{R}^{\ell+n}}{\text{argmin}} & \mathbf{c}^T\left(\lambda_1 K + 2\lambda_2 KLK\right)\mathbf{c} + \sum_{i=1}^{\ell}\xi_i \\ \text{s.t.} & y_i \sum_{j=1}^{\ell+n} c_j k(\mathbf{x}_j,\mathbf{x}_i) \geq 1-\xi_i, \quad i=1,\ldots,\ell \\ & \xi_i \geq 0, \quad i=1,\ldots,\ell \end{cases}$$

et la formulation duale correspondante s'écrit :

$$\begin{cases} \underset{\boldsymbol{\alpha}\in\mathbb{R}^{\ell}}{\text{argmax}} & \sum_{i=1}^{\ell}\alpha_i - \frac{1}{2}\boldsymbol{\alpha}^T Q\boldsymbol{\alpha} \\ \text{s.t.} & 0 \leq \alpha_i \leq 1, \quad i=1,\ldots,\ell \end{cases}$$

où

$$Q = \text{diag}\,(\mathbf{y})JK(2\lambda_1 I_{\ell+n} + 4\lambda_2 LK)^{-1}J^T\,\text{diag}\,(\mathbf{y}).$$

Le vecteur des coefficients $\mathbf{c}$ de la solution donnée par le théorème de représentation est obtenu par l'équation suivante :

$$\mathbf{c} = (2\lambda_1 I_{\ell+n} + 4\lambda_2 LK)^{-1} J^T\,\text{diag}\,(\mathbf{y})\boldsymbol{\alpha}.$$

2.3 Evaluation de l'apprentissage et sélection de modèle

La performance en généralisation d'une méthode d'apprentissage se rapporte à sa capacité de prédiction sur des données de test indépendantes. La problématique de l'évaluation de cette performance apparaît dans deux types de problèmes [Hastie et al., 2008] :

- **La sélection de modèle** : une méthode d'apprentissage est caractérisée par différents hyperparamètres. La sélection de modèle revient à choisir les valeurs de ces hyperparamètres. Plusieurs critères peuvent être utilisés pour faire ce choix, comme la stabilité ou une mesure de performance.
- **L'évaluation d'un modèle** : cette évaluation consiste à estimer l'erreur de prédiction d'un modèle sur de nouvelles données, une fois celui-ci choisi.

Dans le cas où l'on dispose de beaucoup d'exemples, l'approche la plus simple pour les deux problèmes consiste à diviser ces exemples en trois ensembles : un ensemble

d'apprentissage, un ensemble de validation et un ensemble de test. L'ensemble de validation est utilisé pour estimer l'erreur de prédiction pour la sélection de modèle et l'ensemble de test est utilisé pour évaluer l'erreur de généralisation du modèle choisi.

Cependant, en pratique les exemples connus ne sont pas toujours disponibles en quantité suffisante pour permettre l'utilisation de cette approche. La validation croisée permet alors d'estimer l'erreur de prédiction tout en conservant un maximum d'exemples lors de la phase d'apprentissage. La validation croisée en N parties consiste ainsi à :

- découper aléatoirement les données d'apprentissage en N sous-échantillons de mêmes tailles,
- utiliser le $i$ème échantillon pour l'ensemble de test et apprendre le modèle sur les $N-1$ autres échantillons,
- mesurer l'erreur commise sur l'échantillon i,
- recommencer en faisant varier l'échantillon i de 1 à N.

L'estimation de l'erreur finale est obtenue en moyennant les erreurs mesurées pour chaque partie de la validation croisée.

Lorsque les données disponibles sont très peu nombreuses, on peut utiliser la méthode de validation croisée *leave-one-out*, qui consiste à choisir N égal au nombre total d'exemples disponibles. Dans ce cas, le modèle est appris sur toutes les données à l'exception de la $i$ème d'entre elles, et celle-ci est utilisée pour l'ensemble de test. Cette méthode peut être très coûteuse en terme de calculs. Cependant, dans le cas de la Ridge Regression et de la Kernel Ridge Regression, le résultat de la validation croisée leave-one-out peut s'exprimer sous une forme close [Allen, 1974; Stone, 1974].

Conclusion :
Nous avons introduit dans ce chapitre les principaux concepts de l'apprentissage statistique utiles à cette thèse. Nous avons également présenté les méthodes à noyaux, dont deux modèles : la Kernel Ridge Regression et les Séparateurs à Vaste Marge, auxquels nous nous intéresserons plus particulièrement par la suite. Un autre point qui a été abordé dans ce chapitre est celui du problème de l'évaluation de l'apprentissage. Dans le chapitre suivant, nous introduisons le problème de la prédiction de liens et nous parcourons les différentes méthodes d'apprentissage statistique qui ont été développées spécifiquement pour ce problème.

Chapitre 3

Prédiction de liens

Le problème de la prédiction de liens dans un réseau consiste à prédire l'existence d'un lien entre deux entités à partir d'attributs décrivant ces entités. Ce problème a été étudié dans le cadre de l'inférence de réseaux biologiques, et en particulier pour la prédiction d'interactions protéine-protéine. Mais plus généralement, ce problème apparaît dans de nombreux domaines de recherche tels que les réseaux bibliographiques, comme par exemple les réseaux de co-publication ou de co-citation entre auteurs, les réseaux sociaux ou encore les réseaux technologiques comme le Web. La construction d'un système de recommandation peut être également vue comme un problème de prédiction de liens dans un graphe biparti représentant les préférences des utilisateurs ou leurs achats.

Dans la plupart des cas, une partie des liens est observée et l'objectif consiste à prédire des liens qui n'ont pas encore été observés. Dans le cas des réseaux sociaux, l'aspect temporel entre en compte : l'état du réseau à l'instant t est supposé connu et l'objectif consiste à inférer les nouveaux liens qui apparaîtront à l'instant $t+1$.

Le problème de la prédiction de liens reste cependant difficile à résoudre car il s'agit d'un problème très déséquilibré. Il existe en effet très peu d'exemples positifs, c'est à dire de paires de nœuds reliés par un lien, en comparaison du très grand nombre d'exemples négatifs. Une autre difficulté de ce problème, spécifique au cas des réseaux biologiques, est liée au fait que l'on ne dispose pas réellement d'exemples négatifs, mais d'exemples pour lesquels aucun lien n'a été trouvé jusqu'à présent.

Dans ce chapitre, nous présentons les principales approches existantes pour résoudre le problème de la prédiction de liens. Dans la section 3.1, nous décrivons les approches ne faisant pas appel à l'apprentissage. Celles-ci consistent à ordonner les paires de nœuds selon un score indiquant la vraisemblance de l'existence d'un lien entre les deux nœuds [Liben-Nowell & Kleinberg, 2007]. Pour cela, des informations sur la topologie du réseau partiellement connu sont exploitées. La section 3.2 porte sur les différentes méthodes basées sur des modèles probabilistes. Dans la section 3.3, nous présentons les approches qui ont été développées dans le cadre de l'apprentissage non supervisé. Enfin, nous décrivons dans la section 3.4 les approches discriminantes, et plus particulièrement les méthodes à noyaux qui ont été développées pour résoudre ce problème de prédiction.

3.1 Approches basées sur les motifs topologiques

Les approches décrites dans cette section cherchent à exploiter uniquement des motifs topologiques présents dans un réseau partiellement connu. L'existence d'un lien entre deux nœuds u et u' est décidée en calculant une valeur de score pour l'arête (u, u') à partir d'informations sur la topologie du réseau. Un classement des scores obtenus est ensuite effectué sur l'ensemble des différentes paires considérées. Il existe deux familles de mesures de similarités topologiques : les mesures locales basées sur le voisinage des nœuds et les mesures globales basées sur les distances entre nœuds. Liben-Nowell & Kleinberg [2007] ont examiné diverses mesures de similarité appartenant à ces deux familles, et les ont comparées pour le problème de la prédiction de liens dans le cas des réseaux sociaux.

3.1.1 Mesures de similarité locales

On note $\Gamma(u)$ l'ensemble des voisins du nœud u dans le graphe. Les approches utilisant les mesures de similarité locales sont basées sur l'idée que deux nœuds u et u' ayant beaucoup de voisins en commun sont plus susceptibles d'être reliés dans le futur.

Voisins communs. [Newman, 2001] L'approche la plus directe consiste à définir un score entre les nœuds u et u' à partir du nombre de voisins qu'ils ont en commun :

$$\text{score}(u, u') = \mid \Gamma(u) \cap \Gamma(u') \mid .$$

Coefficient de Jaccard. [Chowdhury, 2010] Le coefficient de Jaccard est une version normalisée de la métrique des voisins communs. Il mesure la similarité entre deux nœuds par le nombre de voisins en commun divisé par le nombre total de voisins de ces nœuds :

$$\text{score}(u, u') = \frac{\mid \Gamma(u) \cap \Gamma(u') \mid}{\mid \Gamma(u) \cup \Gamma(u') \mid}.$$

Adamic/Adar. [Adamic & Adar, 2003] La mesure de Adamic/Adar est une autre variante de la métrique des voisins communs. L'idée de cette mesure consiste à introduire une pondération en fonction du nombre de voisins des voisins communs :

$$\text{score}(u, u') = \sum_{p \in \Gamma(u) \cap \Gamma(u')} \frac{1}{\log \mid \Gamma(p) \mid}.$$

Ainsi les voisins les moins connectés sont associés à un poids plus important.

Attachement préférentiel. [Newman, 2001; Barabási et al., 2002] L'attachement préférentiel se base sur le principe que deux nœuds qui ont beaucoup de relations auront tendance à avoir de nouveaux voisins, et seront donc plus susceptibles

d'être liés. Le score associé à la possibilité d'existence d'un lien entre u et u' est le suivant :

$$\text{score}(u, u') = \mid \Gamma(u) \mid \cdot \mid \Gamma(u') \mid .$$

3.1.2 Mesures de similarité globales

Contrairement aux mesures de similarité locales, les mesures globales nécessitent de connaître toute l'information topologique du graphe. Ces approches se basent généralement sur l'hypothèse que dans le cas où il existe plusieurs chemins indirects entre les nœuds u et u', alors il est vraisemblable qu'il existe un lien direct entre ces deux nœuds.

Mesure de Katz. [Katz, 1953] Cette mesure prend en compte tous les chemins reliant les nœuds u et u' et utilise une pondération prenant en compte la longueur de chacun de ces chemins :

$$\text{score}(u, u') = \sum_{\ell=1}^{\infty} \beta^{\ell} \cdot \mid \text{paths}_{u,u'}^{(\ell)} \mid,$$

où $\text{paths}_{u,u'}^{(\ell)}$ est l'ensemble des chemins de longueur ℓ allant de u à u' et où $\beta > 0$ est un paramètre. Cette mesure est identique au noyau de diffusion [Kondor & Lafferty, 2002] utilisé dans les méthodes à noyaux pour définir une similarité entre deux nœuds dans un graphe.

Mesures basées sur les marches aléatoires. Différentes mesures basées sur les marches aléatoires, comme par exemple "Hitting time", "PageRank" [Brin & Page, 1998] ou "SimRank" [Jeh & Widom, 2002], ont été également utilisées pour déterminer la similarité entre deux nœuds d'un graphe.

Cependant, ces approches restent limitées du fait qu'elles utilisent uniquement des informations sur la topologie du réseau pour prédire la présence ou l'absence de liens. Des méthodes ont été développées afin d'utiliser d'autres sources d'information, comme par exemples des descriptions sur les attributs des nœuds. De plus, l'utilisation de l'apprentissage statistique à partir de différentes sources d'information permet d'apprendre à discriminer les instances positives (c'est à dire les paires de nœuds pour lesquelles il existe un lien) des instances négatives. Nous présentons ces différentes approches dans les sections suivantes.

3.2 Modèles probabilistes

Les chercheurs se sont intéressés à l'utilisation de modèles probabilistes pour résoudre le problème de la prédiction de liens [Sarukkai, 2000; Taskar et al., 2003; Getoor & Diehl, 2005; Kashima & Abe, 2006; Miller et al., 2009]. Les approches

basées sur un modèle probabiliste consistent à apprendre à partir du réseau observé un modèle caractérisé par un ensemble de paramètres θ selon une approche de maximisation de la vraisemblance ou une approche d'estimation bayésienne. La probabilité d'existence d'un lien entre deux nœuds u_i et u_j peut alors être estimée par la probabilité conditionnelle $P(e_{ij} = 1|\theta)$.

Différents types de modèles probabilistes ont été développés dans ce cadre. Nous présentons dans la suite plusieurs modèles graphiques probabilistes, des approches à espace latent, ainsi que des modèles plus riches basés sur des représentations relationnelles, tels que les réseaux de Markov relationnels [Taskar et al., 2003] ou les réseaux de Markov logiques [Richardson & Domingos, 2004, 2006].

3.2.1 Modèles graphiques probabilistes

Les modèles graphiques probabilistes sont des graphes, dans lesquels les nœuds représentent des variables aléatoires et les absences de liens des indépendances conditionnelles. Ces modèles permettent de représenter de façon compacte des distributions de probabilité jointes sur un ensemble de variables aléatoires. Il existe principalement deux types de modèle graphique : les modèles graphiques dirigés, appelés réseaux bayésiens, et les modèles graphiques non dirigés, appelés également réseaux de Markov ou champs aléatoires de Markov.

Des classifieurs bayésiens naïfs, qui correspondent à une version simplifiée des réseaux bayésiens, ont été utilisés pour la prédiction d'interactions protéine-protéine chez la levure [Jansen et al., 2003], ainsi que chez l'homme [Rhodes et al., 2005; Xia et al., 2006]. Birlutiu et al. [2012] ont proposé une autre approche bayésienne pour la prédiction de PPI, qui combine des informations sur la topologie du réseau avec des informations sur des attributs décrivant les protéines.

Des réseaux bayésiens [Friedman et al., 2004] ont été par ailleurs utilisés dans le cadre non supervisé pour inférer des réseaux de régulation en cherchant à détecter des relations de causalité entre des données d'expression de gènes.

Un modèle graphique local a été défini par Wang et al. [2007] pour la prédiction temporelle de liens dans un réseau de co-publication. Ils ont cherché à estimer pour une paire de nœuds la probabilité que les nœuds soient reliés dans le futur. Ils ont pour cela utilisé un réseau de Markov afin de modéliser le voisinage local d'une paire de nœuds.

3.2.2 Approches à espace latent

Parmi les approches utilisant un modèle probabiliste pour la prédiction de liens, certaines d'entre elles [Hoff et al., 2002; Hoff, 2008; Miller et al., 2009] considèrent que chaque entité a une représentation binaire latente qui influence ses relations. Ces approches ont été utilisées pour l'inférence de réseaux sociaux, et se basent sur l'observation suivante : la probabilité d'un lien relationnel entre deux individus augmente lorsque les caractéristiques de ces deux individus sont similaires. La probabilité d'un lien est donc modélisée de sorte à être dépendante de la similarité

entre les nœuds dans un espace de caractéristiques latentes non observées. Miller et al. [2009] ont proposé un modèle bayésien non paramétrique, qui permet d'inférer automatiquement la dimension de cet espace latent.

3.2.3 Apprentissage statistique relationnel

L'apprentissage statistique relationnel [Getoor & Taskar, 2007] fait référence à une sous-partie de l'apprentissage statistique qui combine des règles logiques du premier ordre avec des modèles graphiques probabilistes. Les méthodes d'apprentissage relationnel cherchent à capturer les relations complexes existant entre les exemples. Différentes méthodes relationnelles ont été utilisées pour la prédiction de liens, en exploitant à la fois la structure des liens et des descriptions des nœuds. Nous présentons les principales d'entre elles dans la suite.

Modèles graphiques relationnels Les modèles probabilistes relationnels [Friedman et al., 1999] désignent une approche, qui étend les réseaux bayésiens au domaine relationnel. Getoor et al. [2002] ont étendu le cadre des modèles probabilistes relationnels pour modéliser et prédire l'existence de liens. Cette approche a été utilisée par Huang et al. [2004] pour la prédiction de liens dans le contexte spécifique des systèmes de recommandation.

Régression logistique structurée Popescul & Ungar [2003] ont introduit un modèle de régression logistique structurée pour la prédiction de liens. La régression logique structurée combine la régression logistique avec la génération de descriptions relationnelles. Ces descriptions sont construites à partir de requêtes SQL sur les données d'entrée. Cette approche a été appliquée au problème de la prédiction de citations dans le domaine des publications scientifiques.

Réseaux de Markov relationnels Taskar et al. [2003] ont utilisé des réseaux de Markov relationnels qui permettent de définir une distribution de probabilité jointe sur le graphe entier, c'est à dire sur les liens, ainsi que sur les attributs décrivant les nœuds du graphe. Le modèle appris est utilisé pour classifier les données de test de manière collective. Jaimovich et al. [2006] ont appliqué cette approche à la prédiction d'interactions protéine-protéine.

Réseaux de Markov logiques Un réseau de Markov logique $\mathcal{M}$ [Richardson & Domingos, 2004, 2006] est défini par un ensemble de formules $F = \{f_i, i = 1, \cdots, p\}$ et un vecteur de poids $\boldsymbol{w}$ de dimension p, où la clause f_i a un poids associé w_i qui reflète son importance. Un réseau de Markov logique (MLN) permet donc d'assouplir la logique du premier ordre et d'encapsuler l'apprentissage des poids dans un cadre probabiliste. Un réseau de Markov logique associé à un ensemble fini de constantes, parmi lesquelles les variables peuvent prendre leurs valeurs, définit un réseau de Markov.

L'apprentissage d'un MLN comporte deux étapes : l'apprentissage de la structure, c'est à dire l'apprentissage des formules logiques, et l'apprentissage des paramètres, c'est à dire l'apprentissage des poids de chaque formule.

Nous avons utilisé le cadre des réseaux de Markov logiques pour inférer un réseau de régulation génique impliqué dans la bascule entre la prolifération et la différenciation des keratinocytes humains [1] [Brouard et al., 2012]. Nous avons ainsi appris un MLN à partir d'un ensemble de données transcriptomiques expérimentales et de descriptions des gènes encodées sous la forme de prédicats logiques.

3.3 Approches non supervisées

Des approches non supervisées ont été développées dans le cadre de l'inférence de réseaux biologiques et sociaux. Ces approches utilisent uniquement des attributs sur les nœuds pour prédire l'existence de liens. Dans le cas de l'inférence de réseaux de régulation de gène, des séries temporelles de données d'expression ont été utilisées par différentes méthodes. Certaines de ces approches modélisent le réseau de régulation comme un système dynamique et utilisent les équation différentielles ordinaires [Chen et al., 1999; Gardner et al., 2003; di Bernardo et al., 2005; Bansal et al., 2006].

Un autre type approche consiste à prédire un lien entre deux entités lorsque celles-ci sont similaires [Pazos & Valencia, 2001], par exemple en termes d'expression ou de profils phylogénétiques. D'autres méthodes utilisent l'information mutuelle. Certaines de ces approches [Butte et al., 2000; Faith et al., 2007] ont été utilisées pour l'inférence de régulations géniques. Elles sont basées sur le principe qu'une interaction entre un facteur de transcription et un gène régulé par ce facteur de transcription peut être détectée par une relation de dépendance entre leurs niveaux d'expression.

Enfin, Lippert et al. [2009] ont proposé une méthode à noyau non supervisée pour la prédiction de liens. Ils ont pour cela adapté le critère d'indépendance de Hilbert-Schmidt (HSIC) [Gretton et al., 2005] au problème de l'inférence de réseau. Deux noyaux sont définis, l'un d'entre eux porte sur des attributs décrivant les nœuds et l'autre sur des propriétés topologiques du réseau. Le problème d'inférence de réseau consiste alors à trouver la configuration des liens qui maximise la dépendance entre les attributs des nœuds et leurs positions dans le graphe.

3.4 Approches discriminantes

Les approches discriminantes consistent à apprendre une fonction de décision qui associe une paire de nœuds à une étiquette indiquant la présence ou l'absence de lien entre ces deux nœuds. Contrairement aux approches non supervisées, on suppose que des informations sur les attributs des nœuds sont disponibles et que l'on dispose

1. Ce travail a été réalisé en partie durant la thèse, mais pour un choix de cohérence nous ne le présentons pas ici.

également d'un ensemble de liens déjà connus. Dans la suite, nous formalisons ce problème d'apprentissage et nous présentons les différentes approches existantes dans le cas supervisé et dans le cas transductif.

3.4.1 Apprentissage supervisé

On considère un ensemble de nœuds $\mathcal{U}$, correspondant par exemple à un ensemble de protéines ou à un ensemble de personnes. On suppose de plus que chaque nœud $u \in \mathcal{U}$ est décrit par un vecteur $\phi(u) \in \mathbb{R}^p$. On note $E \subset \mathcal{U} \times \mathcal{U}$ l'ensemble des arêtes $e_1, \ldots, e_M$ qui définissent le réseau. Dans le cas de l'inférence supervisée, on dispose lors de la phase d'apprentissage d'un ensemble $\mathcal{S} = ((e_1, y_1), \cdots, (e_m, y_m))$ de paires de nœuds associées à des étiquettes $y_i \in \{0, 1\}$, qui indiquent s'il existe ou non un lien entre les nœuds de e_i. Etant donnés l'ensemble d'apprentissage $\mathcal{S}$ et l'ensemble des descriptions portant sur les nœuds étiquetés, l'objectif de l'inférence supervisée de réseau consiste à prédire pour toutes les paires de nœuds faisant partie de l'ensemble de test si elles interagissent ou non. Dans le cadre supervisé, la prédiction de liens peut se traduire par l'apprentissage d'un classifieur binaire sur des paires d'objets : un classifieur $f : \mathcal{U} \times \mathcal{U} \Rightarrow \{0, 1\}$ prédit ainsi si deux objets interagissent ou non à partir des exemples d'apprentissage.

3.4.1.1 Méthodes d'apprentissage

Plusieurs approches supervisées ont été utilisées pour l'inférence de réseaux biologiques [Qi & Noble, 2011], dont : la régression logistique [Lin et al., 2004], les méthodes à noyaux [Yamanishi et al., 2004; Ben-Hur & Noble, 2005], les arbres de décision [Zhang et al., 2004] et les forêts aléatoires [Qi et al., 2005]. Ces classifieurs ont été comparés pour la prédiction de PPI chez la levure [Qi et al., 2006], et les méthodes qui ont obtenu les meilleures performances sont les forêts aléatoires et les séparateurs à vaste marge.

Dans le cas des réseaux sociaux, Al Hasan et al. [2006] ont également comparé plusieurs algorithmes de classification supervisée pour l'inférence d'un réseau de co-citations entre publications scientifiques à partir d'informations sur la topologie et sur les caractéristiques des nœuds. L'approche utilisant les SVM obtient les meilleurs résultats. Par ailleurs, Lichtenwalter et al. [2010] ont proposé deux méthodes d'ensemble, basées sur le bagging et les forêts aléatoires, pour la prédiction de liens dans le cadre supervisé.

Dans la suite, nous présentons plus en détails les méthodes à noyaux, qui ont été beaucoup utilisées pour l'inférence de réseaux biologiques, car elles permettent de combiner différentes données hétérogènes en entrée. On peut distinguer deux types de méthodes à noyaux parmi celles utilisées dans le cadre de la prédiction de liens : celles basées sur des noyaux entre paires de nœuds, et celles basées sur des noyaux entre nœuds.

3.4.1.2 Méthodes à noyaux basées sur des noyaux entre paires de nœuds

Ben-Hur & Noble [2005], ainsi que Martin et al. [2005], ont proposé une méthode à noyaux pour la prédiction d'interactions protéine-protéine qui est basée sur des noyaux entre paires de protéines. Chaque paire de protéines est considérée comme un exemple et le noyau entre paires mesure la similarité entre deux paires de protéines. Ainsi pour n objets, la matrice de Gram du noyau obtenue contient n^4 entrées. Une manière possible de définir un tel noyau consiste à utiliser un noyau k entre nœuds. Etant donné ce noyau k, Ben-Hur & Noble [2005] et Martin et al. [2005] ont proposé de définir le noyau entre paires de la façon suivante :

$$K((u_1, u_2), (u_3, u_4)) = k(u_1, u_3)k(u_2, u_4) + k(u_1, u_4)k(u_2, u_3). \tag{3.1}$$

D'après cette définition, deux paires de protéines (u_1, u_2) et (u_3, u_4) sont similaires entre elles si les protéines u_1 et u_3 sont similaires et si les protéines u_2 et u_4 le sont également. Ce noyau prend également en compte le fait que u_1 peut être similaire à u_3 ou à u_4. Dans le cas d'un réseau dirigé, seul le premier terme est considéré.

Basilico & Hofmann [2004] ont proposé un noyau similaire pour la prédiction de notes données par des utilisateurs à des objets dans le cadre d'un système de recommandation. Un noyau entre deux paires utilisateur-objet, notées (u, x) et (u', x'), est défini de la façon suivante :

$$K((u, x), (u', x')) = k_U(u, u')k_X(x, x'), \tag{3.2}$$

où k_X désigne un noyau entre objets et k_U un noyau entre utilisateurs.

Il est par ailleurs possible de calculer un noyau entre paires de protéines en utilisant directement des descriptions caractérisant ces paires [Ben-Hur & Noble, 2005] :

$$K((u_1, u_2), (u_3, o_4)) = k(\mathbf{e}_{12}, \mathbf{e}_{34}), \tag{3.3}$$

où $\mathbf{e}_{ij}$ est un vecteur représentant la paire (u_i, u_j).

Plus récemment, Vert et al. [2007] ont considéré le problème de la prédiction de liens supervisée comme un problème d'apprentissage de distance. Une relaxation du problème d'optimisation résultant a conduit à l'algorithme d'un SVM avec un noyau particulier entre paires de nœuds. Comme dans le cas décrit précédemment, ce noyau entre paires est défini à partir d'un noyau k entre nœuds :

$$K((u_1, u_2), (u_3, u_4)) = (k(u_1, u_3) - k(u_1, u_4) - k(u_2, u_3) + k(u_2, u_4))^2. \tag{3.4}$$

k étant un noyau, il peut être écrit comme un produit scalaire dans l'espace caractéristique : $k(u, u') = \phi(u)^T\phi(u')$. Par conséquent, le noyau entre paires peut être

reformulé sous la forme suivante :

$$K((u_1, u_2), (u_3, u_4)) = [(\phi(u_1) - \phi(u_2))^T(\phi(u_3) - \phi(u_4))]^2. \tag{3.5}$$

Les paires de nœuds sont ainsi comparées en utilisant pour chacune d'entre elles la différence dans l'espace caractéristique entre les images par la fonction ϕ des nœuds de la paire.

3.4.1.3 Méthodes à noyaux basées sur des noyaux entre nœuds

Du fait de la complexité en temps de calculs des méthodes basées sur des noyaux entre paires, certains chercheurs ont proposé de travailler directement avec des noyaux entre nœuds. Ces approches nécessitent de connaître complètement une sous-matrice de la matrice d'adjacence du réseau d'interaction (voir figure 3.1).

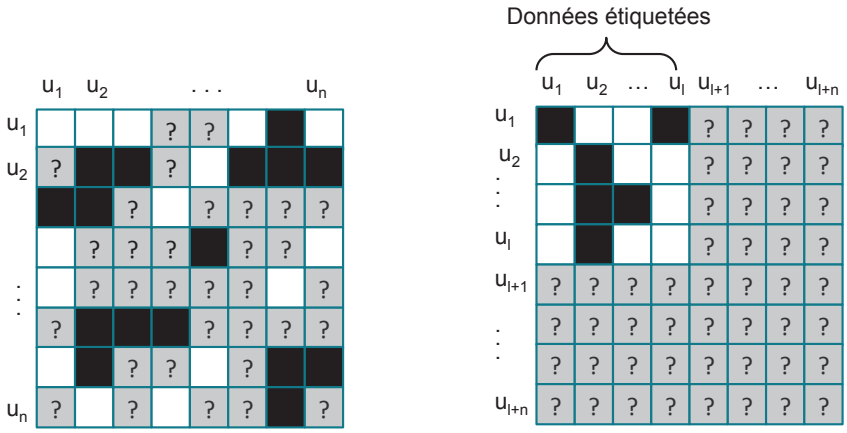

FIGURE 3.1 – **Inférence supervisée de réseau avec les méthodes à noyaux**. Sur cette figure, nous représentons la matrice d'adjacence du réseau partiellement connu pour les deux types de méthodes à noyaux. La matrice d'adjacence de gauche correspond au cas d'une méthode basée sur un noyau entre paires, tandis que celle de droite correspond à une méthode basée sur un noyau entre nœuds. Cette dernière nécessite de connaître complètement une sous-matrice de la matrice d'adjacence. Sur les deux matrices d'adjacence, les liens connus sont représentés en noir, les absences de liens connues en blanc et les paires pour lesquelles la valeur est inconnue sont indiquées en gris.

L'approche développée par Yamanishi et al. [2004] pour l'inférence de réseau de PPI utilise une variante de l'analyse canonique des corrélations à noyau. Cette approche consiste à identifier des corrélations entre le réseau d'interaction partiellement connu et différents attributs décrivant les protéines.

La méthode proposée par Vert & Yamanishi [2005] consiste à utiliser le sous-réseau d'interaction connu pour trouver une nouvelle représentation des nœuds dans un espace vectoriel. La fonction de passage est apprise en imposant la proximité

dans l'espace vectoriel des nœuds adjacents dans le sous-graphe connu. Puis dans cet espace, une interaction est prédite pour les paires de nœuds séparés par une distance plus petite qu'un certain seuil δ.

Régression à noyau de sortie Geurts et al. [2006, 2007] ont proposé une méthode de prédiction de liens basée sur le principe de la régression à noyau de sortie. Cette méthode définit un noyau en sortie, permettant de capturer les proximités entre les nœuds du réseau d'interaction. A partir de la matrice de Gram de ce noyau sur les données d'apprentissage, l'objectif de cette méthode consiste à trouver une approximation des valeurs du noyau pour l'ensemble de test. Des méthodes à base d'arbres ont été utilisées pour résoudre le problème de la régression à noyau de sortie.

Modèles locaux Bleakley et al. [2007] ont introduit une méthode qui utilise des modèles locaux pour l'inférence de réseaux métaboliques et de réseaux de PPI. Cette méthode a été également appliquée à l'inférence de réseaux de régulation génique [Mordelet & Vert, 2008].

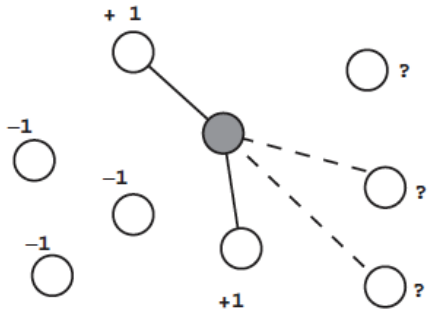

FIGURE 3.2 – Modèle local associé à une protéine [Bleakley et al., 2007]

Dans cette approche, un modèle local est construit pour chaque protéine en utilisant les interactions et absences d'interaction connues avec cette protéine comme exemples positifs et négatifs. Celui-ci est ensuite utilisé pour prédire les interactions manquantes dans le modèle local (voir figure 3.2). Ainsi, un sous-réseau correspondant à chaque protéine est appris individuellement. Une fois les modèles locaux appris pour toutes les protéines, ils sont combinés de façon à obtenir un réseau d'interaction final. Les classifieurs locaux utilisés sont des SVM, et comme chaque nœud a son propre modèle, l'approche nécessite de définir seulement un noyau entre protéines plutôt qu'un noyau entre paires de protéines. Cependant, cette approche présente une limitation, qui est liée au fait qu'elle ne permet pas de prédire des interactions entre deux protéines n'appartenant pas à l'ensemble d'apprentissage.

Inférence de réseaux asymétriques Les approches décrites précédemment concernent l'inférence de réseaux symétriques, mais des approches similaires existent

également dans le cas des réseaux asymétriques. Parmi ces réseaux, les réseaux d'interactions entre protéines et composés chimiques ont été très étudiés ces dernières années [Nagamine & Sakakibara, 2007; Jacob & Vert, 2008; Yamanishi et al., 2008; Bleakley & Yamanishi, 2009]. Un autre cas d'inférence de réseau asymétrique concerne l'inférence de réseau d'interaction protéine-protéine entre différentes espèces. Des méthodes ont été développées en particulier pour identifier les interactions existantes entre des protéines d'un organisme hôte et celles d'un organisme pathogène [Tastan et al., 2009].

3.4.2 Apprentissage transductif

Des méthodes ont été également développées dans le cadre de l'apprentissage transductif afin d'exploiter les descriptions des nœuds faisant partie de l'ensemble de test. Nous décrivons trois de ces approches, qui sont toutes des méthodes à noyaux.

Expansion de l'ensemble d'apprentissage L'approche locale décrite précédemment présente une autre limitation qui concerne la taille de l'ensemble d'apprentissage utilisé pour chaque modèle local. Celle-ci peut en effet être très petite dans le cas où un nœud a peu, voire aucun, voisins connus. Afin de résoudre ce problème, Yip & Gerstein [2009] ont proposé deux méthodes d'apprentissage transductif permettant d'augmenter le nombre d'exemples d'apprentissage. La première d'entre elles consiste à propager les interactions prédites avec confiance par un modèle afin de les utiliser comme exemples d'apprentissage supplémentaires pour d'autres classifieurs locaux. La deuxième méthode, quant à elle, consiste à générer des exemples d'apprentissage auxiliaires lors de la phase d'initialisation. Pour cela, une comparaison des différents objets est effectuée et les paires d'objets les plus similaires, ainsi que les moins similaires, sont utilisées respectivement comme exemples positifs et négatifs supplémentaires.

Complétion de la matrice noyau Kato et al. [2005] ont formulé le problème d'inférence de réseau comme un problème de complétion d'une matrice de Gram associée à un noyau. Cette matrice de Gram est définie à partir de la matrice d'adjacence du réseau partiellement connu. La mesure de divergence de Kulllback-Leibler est utilisée afin de compléter la matrice de Gram en sortie à partir de la matrice de Gram d'un noyau défini en entrée. Lorsque plusieurs noyaux sont disponibles en entrée, un algorithme d'espérance-maximisation est utilisé pour inférer simultanément les entrées manquantes de la matrice d'adjacence et les poids associés à ces différents noyaux [Tsuda et al., 2003].

Yamanishi & Vert [2007] ont considéré le problème de complétion d'une matrice de Gram à partir d'une matrice de Gram auxiliaire du point de vue de la régression. Les matrices de Gram des deux noyaux sont considérées respectivement comme la variable explicative et la variable à expliquer. Cette approche est appelée *Penalized Kernel Matrix Regression*.

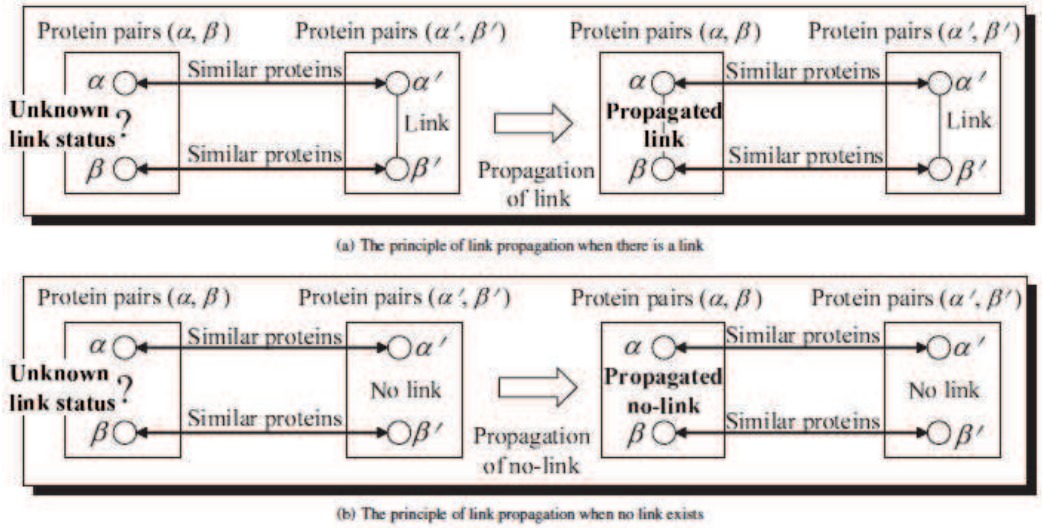

FIGURE 3.3 – Principe de l'approche par propagation de liens pour l'inférence de réseaux métaboliques [Kashima et al., 2009b]

Propagation de liens Kashima et al. [2009a,b] ont étendu au problème de la prédiction de liens le principe de la propagation de l'information des étiquettes utilisé dans l'apprentissage semi-supervisé. Ce principe [Zhu et al., 2003; Zhou et al., 2004] est utilisé à l'origine pour prédire les étiquettes des nœuds non étiquetés et se base sur l'hypothèse que si deux nœuds sont similaires alors il est probable qu'ils aient la même étiquette. Ce principe est adapté au problème de la prédiction de liens de la façon suivante : si deux paires de nœuds sont similaires, alors elles seront vraisemblablement associées au même type de lien (voire figure 3.3). Cette approche a été utilisée pour inférer simultanément des réseaux métaboliques chez plusieurs espèces à partir de données d'expressions de gènes [Kashima et al., 2009b].

Conclusion :
Nous avons introduit dans ce chapitre le problème de la prédiction de liens et les différents domaines, dans lesquels il a été étudié. Nous avons également présenté les principales approches existantes pour résoudre ce problème, en l'occurrence les approches exploitant des propriétés topologiques du réseau, les approches basées sur un modèle probabiliste, les approches non supervisées et enfin les approches discriminantes. Dans la partie suivante, nous proposons une nouvelle approche pour le problème de la prédiction de liens que nous avons développée : la régression à noyaux d'entrée et de sortie. Celle-ci s'inscrit dans le cadre de l'apprentissage semi-supervisé.

Deuxième partie

Régression à noyaux d'entrée et de sortie pour la prédiction de liens

Après avoir passé en revue dans le chapitre 3 les différentes approches d'apprentissage statistique permettant de résoudre le problème de la prédiction de liens, nous présentons dans cette partie une nouvelle méthode que nous avons développée, appelée *régression à noyaux d'entrée et de sortie*.

Dans certaines applications comme l'inférence de réseaux d'interaction protéine-protéine, peu de données étiquetées sont généralement disponibles, mais l'on dispose par ailleurs de nombreuses propriétés connues sur les protéines. Dans ce travail, nous avons donc choisi de nous placer dans le cadre de l'apprentissage semi-supervisé pour pouvoir bénéficier de cette information supplémentaire.

De plus, nous considérons que l'ensemble des nœuds d'intérêt peut être supposé connu au début du processus d'apprentissage, et que ce problème peut être vu comme un problème de complétion d'un réseau partiellement connu. Nous avons donc fait le choix de résoudre ce problème de prédiction dans le cadre de l'apprentissage transductif.

Pour cela, nous avons converti le problème de classification binaire à partir de paires d'objets en un problème d'apprentissage d'un noyau de sortie comme Geurts et al. [2006, 2007]. L'objectif de l'apprentissage consiste à approcher un noyau de sortie cible κ_y, supposé encoder la similarité des données en tant que nœuds dans le graphe, à partir de descripteurs appropriés en entrée. Une fois le noyau de sortie appris, la tâche de prédiction de liens est réalisée en seuillant la valeur de ce noyau pour une paire d'entrées :

$$f_\theta(u, u') = \mathrm{sgn}(\widehat{\kappa_y}(u, u') - \theta)\,.$$

L'approximation du noyau de sortie cible est construite à partir du produit scalaire entre les sorties d'une fonction $h : \mathcal{U} \to \mathcal{F}_y$, où $\mathcal{F}_y$ est un espace de Hilbert. L'utilisation de l'astuce du noyau dans l'espace de sortie permet ainsi de réduire le problème d'apprentissage à partir de paires d'objets à un problème d'apprentissage d'une fonction d'une seule variable à valeurs vectorielles dans un espace de Hilbert. Cette tâche de régression supervisée est appelée *régression à noyau de sortie*. Des méthodes à base d'arbres ont été développées pour résoudre cette tâche de régression et ont été appliquées à l'inférence supervisée de réseaux biologiques [Geurts et al., 2007]. L'approche de la régression à noyau de sortie a ainsi donné des résultats prometteurs. Cependant, ces méthodes ne sont pas adaptées pour étendre la régression à noyau de sortie au cas de l'apprentissage semi-supervisé. Nous avons donc recherché un cadre approprié pour cette problématique d'apprentissage.

Comme nous l'avons vu dans le chapitre 2, une approche existante pour la régression semi-supervisée consiste à forcer la régularité de la fonction de prédiction en tenant compte de l'information des exemples étiquetés et non étiquetés. En travaillant dans un espace fonctionnel approprié, en l'occurrence un espace de Hilbert à noyau reproduisant, des théorèmes de représentation nous donnent la forme de modèles permettant de résoudre cette tâche de régression.

Dans le cas de la régression à noyau de sortie, la fonction que l'on cherche à apprendre n'est pas à valeurs réelles, mais à valeurs vectorielles dans un espace de Hilbert. Nous nous sommes donc tournés vers la théorie des espaces de Hilbert à noyau reproduisant dédiée aux fonctions à valeurs dans un espace de Hilbert. En se plaçant dans le cadre de cette théorie, nous proposons une nouvelle approche, appelée *régression à noyaux d'entrée et de sortie*.

Nous présentons cette approche dans le chapitre 4 et nous montrons que celle-ci peut être utilisée de manière plus générale pour résoudre des problèmes de régression ou de classification structurée. Elle permet ainsi d'approcher des fonctions allant d'un ensemble $\mathcal{U}$ dans un espace structuré $\mathcal{V}$, en supposant qu'un noyau peut être défini entre les éléments de $\mathcal{V}$.

Le chapitre 5 est consacré à l'application de la régression à noyaux d'entrée et de sortie au problème de la prédiction de liens. Nous présentons les modèles obtenus dans le cadre de l'apprentissage supervisé et de l'apprentissage semi-supervisé. Nous présentons également une approche par transfert de liens, permettant de bénéficier de l'information des interactions connues chez plusieurs espèces pour prédire des interactions protéine-protéine chez une espèce cible.

Chapitre 4

Théorie des RKHS à noyaux à valeur opérateur pour la régression

Dans ce chapitre, nous considérons le problème d'estimation d'une fonction à valeurs dans un ensemble structuré. Nous supposons qu'un noyau peut être défini entre les éléments de sortie. En partant de cette supposition, nous proposons une nouvelle approche pour la prédiction de sorties structurées, qui consiste à résoudre un problème d'approximation de fonction dans le cadre de la théorie des espaces de Hilbert à noyau reproduisant pour les fonctions à valeurs dans un espace de Hilbert.

Nous commençons par présenter dans la section 4.1 la méthode de la régression à noyaux d'entrée et de sortie pour l'approximation de fonctions à valeurs vectorielles. Dans la section 4.2, nous introduisons la théorie des RKHS dédiée aux fonctions à valeurs vectorielles, ainsi que les noyaux à valeur opérateur. Le problème d'apprentissage régularisé dans le cadre de cette théorie est abordé dans la section 4.3, et dans la section 4.4, nous l'étendons au cadre de l'apprentissage semi-supervisé. Enfin, la section 4.5 est dédiée à un type particulier de noyaux à valeur opérateur, appelés noyaux décomposables [1].

4.1 Régression à noyaux d'entrée et de sortie

On considère la tâche de régression consistant à prédire à partir de données appartenant à un ensemble $\mathcal{U}$, les sorties correspondantes appartenant à un ensemble $\mathcal{V}$. Les sorties sont supposées être structurées, ainsi $\mathcal{V}$ peut par exemple correspondre à un ensemble de séquences, de graphes ou de nœuds dans un graphe. On suppose qu'il est possible de construire une similarité $\kappa_y : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ de telle sorte que κ_y tienne compte de la structure existante entre les éléments de $\mathcal{V}$. Si l'on fait l'hypothèse supplémentaire que κ_y est un noyau semi-défini positif, alors d'après le théorème 3 vu dans le chapitre 2, il existe un espace de Hilbert $\mathcal{F}_y$, appelé espace caractéristique, et une fonction caractéristique y tels que :

$$\forall (v, v') \in \mathcal{V} \times \mathcal{V}, \kappa_y(v, v') = \langle y(v), y(v') \rangle_{\mathcal{F}_y}.$$

Le problème de régression peut alors être décomposé en deux tâches (voir figure 4.1) :

1. Certains auteurs parlent de manière équivalente de noyaux séparables.

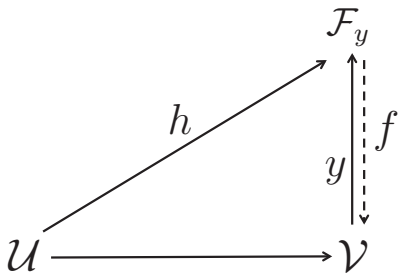

FIGURE 4.1 – Schéma de la régression à noyau de sortie.

- l'apprentissage d'une fonction h allant de l'ensemble $\mathcal{U}$ dans l'espace de Hilbert $\mathcal{F}_y$,
- la définition ou l'apprentissage d'une fonction f de $\mathcal{F}_y$ dans l'ensemble $\mathcal{V}$.

La première tâche est appelée *Régression à noyau de sortie* (OKR, pour *Ouput Kernel Regression*), en référence aux premiers travaux sur la régression à noyau de sortie basés sur des méthodes d'arbres [Geurts et al., 2006, 2007]. La seconde tâche correspond à un *problème de préimage*, pour lequel plusieurs méthodes ont déjà été proposées [Honeine & Richard, 2011]. Dans ce travail, nous nous sommes concentrés sur la tâche de régression à noyau de sortie, et nous avons développé un cadre théorique pour résoudre cette tâche [Brouard et al., 2011].

Dans le cas de la régression à noyau de sortie, la fonction que l'on cherche à approcher est à valeurs vectorielles dans l'espace de Hilbert $\mathcal{F}_y$. Afin de définir un cadre théorique pour OKR, nous avons besoin de déterminer un espace fonctionnel $\mathcal{H}$ approprié, dans lequel le modèle h sera recherché. Pour cela, nous nous sommes tournés vers la théorie des espaces de Hilbert à noyau reproduisant dédiée aux fonctions à valeurs dans un espace de Hilbert. De façon similaire à la théorie des RKHS pour les fonctions à valeurs scalaires, cette théorie permet de construire des modèles non paramétriques dans le contexte de la régularisation.

Dans le cadre de cette théorie, les noyaux ne sont plus à valeurs scalaires, mais définissent pour une paire d'entrées, des opérateurs linéaires sur l'espace de Hilbert de sortie, permettant ainsi de prendre en compte des propriétés complexes des données en entrée. Une fois qu'un noyau à valeur opérateur $\mathcal{K}_x$ est défini, cela détermine de manière unique un espace fonctionnel $\mathcal{H}_\mathcal{K}$, dans lequel on va pouvoir rechercher des modèles non paramétriques en minimisant une certaine fonctionnelle de coût. Ces modèles sont de la forme suivante :

$$\forall u \in \mathcal{U},\ h(u) = \sum_{i=1}^{\ell} \mathcal{K}_x(u_i, u)\mathbf{c}_i,\ \mathbf{c}_i \in \mathcal{F}_y.$$

La famille des méthodes de régression à noyau de sortie et à noyau à valeur opérateur en entrée est appelée *régression à noyaux d'entrée et de sortie* (IOKR, pour *Input Output Kernel Regression*). Ces méthodes nécessitent de définir une paire $(\kappa_y, \mathcal{K}_x)$

pour résoudre le problème de régression à sorties structurées. Nous montrons dans l'annexe A que le même problème de prédiction de sorties structurées, en l'occurrence un problème d'apprentissage multiclasses, peut être résolu de différentes façons selon le choix de la paire $(\kappa_y, \mathcal{K}_x)$.

Dans ce travail, nous avons considéré des noyaux à valeur opérateur construits à partir d'un noyau d'entrée scalaire $\kappa_x : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$. Ces noyaux à valeur opérateur sont appelés noyaux décomposables. Ils sont basés sur le produit entre un noyau d'entrée scalaire κ_x et une matrice semi-définie positive A de taille $dim(\mathcal{F}_y) \times dim(\mathcal{F}_y)$. La régression IOKR basée sur un noyau à valeur opérateur décomposable en entrée et un noyau de sortie généralise le problème de *Kernel Dependency Estimation* (KDE) introduit par Weston et al. [2003] et reformulé de manière plus générale par Cortes et al. [2005]. KDE recherche une fonction allant de $\mathcal{U}$ dans $\mathcal{F}_y$ en construisant une fonction allant de $\mathcal{F}_x$, l'espace caractéristique associé au noyau κ_x, dans $\mathcal{F}_y$ et en la composant avec la fonction caractéristique x associée à κ_x (voir figure 4.2).

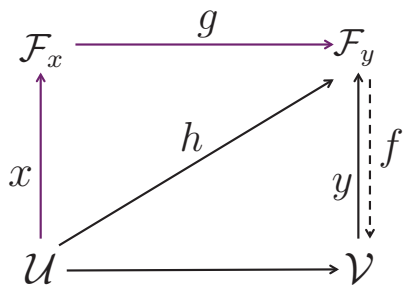

FIGURE 4.2 – **Kernel Dependency Estimation**. La fonction allant de $\mathcal{U}$ dans $\mathcal{V}$ est reconstruite en deux étapes : la fonction h est apprise à partir de la composition des fonctions x et g en utilisant la kernel ridge regression, puis f est déterminée en résolvant un problème de préimage.

Dans la suite, nous montrons comment la théorie des RKHS dédiée aux fonctions à valeurs vectorielles peut fournir un cadre théorique pour la régression à noyau de sortie dans le cadre de l'apprentissage supervisé et de l'apprentissage semi-supervisé.

4.2 Noyaux à valeur opérateur

La théorie des RKHS pour les fonctions à valeurs vectorielles dans un espace de Hilbert a été introduite par Senkene & Tempel'man [1973] puis développée par Micchelli & Pontil [2005], Caponnetto et al. [2008] et Baldassarre et al. [2012]. Une revue de la littérature existant à ce sujet a été décrite par Alvarez et al. [2011].

Dans le cadre de cette théorie, les noyaux sont à valeurs opérateurs et s'appliquent à des vecteurs de l'espace de Hilbert considéré. Dans le cas où la dimension de l'espace $\mathcal{F}_y$ est finie, le noyau $\mathcal{K}_x$ est une fonction à valeurs matricielles. Caponnetto et al. [2008] et Argyriou et al. [2009] ont appliqué cette théorie pour la

résolution de problèmes d'apprentissage multitâches. Celle-ci a été également utilisée par Kadri et al. [2010, 2011] dans le cadre de la régression fonctionnelle.

Nous présentons dans la suite les principaux éléments de cette théorie.

On considère un ensemble $\mathcal{U}$ et un espace de Hilbert $\mathcal{F}_y$ de dimension d doté du produit scalaire $\langle \cdot, \cdot \rangle_{\mathcal{F}_y}$. Dans cette section, nous exposons la théorie générale et nous considérons donc que $\mathcal{F}_y$ est un espace de Hilbert sans lien avec un noyau de sortie. On note $\mathcal{L}(\mathcal{F}_y)$, l'ensemble composé de tous les opérateurs linéaires bornés de $\mathcal{F}_y$ sur lui-même, et A^* la matrice adjointe de $A \in \mathcal{L}(\mathcal{F}_y)$.

Définition 4 *[Senkene & Tempel'man, 1973; Caponnetto et al., 2008] On dit que la fonction $\mathcal{K}_x : \mathcal{U} \times \mathcal{U} \to \mathcal{L}(\mathcal{F}_y)$ est un noyau à valeur opérateur si celle-ci vérifie les propriétés suivantes :*

1. $\forall (u, u') \in \mathcal{U} \times \mathcal{U}$, $\mathcal{K}_x(u, u') = \mathcal{K}_x(u', u)^*$,
2. $\forall m \in \mathbb{N}$, $\forall \{(u_i, \mathbf{y}_i)\}_{i=1}^m \subseteq \mathcal{U} \times \mathcal{F}_y$,

$$\sum_{i,j=1}^{m} \langle \mathbf{y}_i, \mathcal{K}_x(u_i, u_j)\mathbf{y}_j \rangle_{\mathcal{F}_y} \geq 0 \, .$$

Le théorème suivant stipule qu'étant donné un noyau à valeur opérateur, il est possible de construire le RKHS correspondant.

Théorème 5 *[Senkene & Tempel'man, 1973; Micchelli & Pontil, 2005]*
Si $\mathcal{K}_x : \mathcal{U} \times \mathcal{U} \to \mathcal{L}(\mathcal{F}_y)$ est un noyau à valeur opérateur, alors il existe un unique RKHS $\mathcal{H}_{\mathcal{K}_x}$ admettant $\mathcal{K}_x$ comme noyau reproduisant, c'est-à-dire qui vérifie la propriété suivante, correspondant à la propriété reproduisante :

$$\forall u \in \mathcal{U}, \forall \mathbf{y} \in \mathcal{F}_y, \ \langle h, \mathcal{K}_x(\cdot, u)\mathbf{y} \rangle_{\mathcal{H}} = \langle h(u), \mathbf{y} \rangle_{\mathcal{F}_y} \, .$$

Pour construire le RKHS $\mathcal{H}_{\mathcal{K}_x}$, on commence par définir $\mathcal{H}_0$ comme l'espace composé des fonctions de la forme $f(\cdot) = \sum_i \mathcal{K}_x(\cdot, u_i)\boldsymbol{\alpha}_i$, où $u_i \in \mathcal{U}, \boldsymbol{\alpha}_i \in \mathcal{F}_y$. Le produit scalaire sur $\mathcal{H}_0$ entre deux fonctions $f = \sum_i \mathcal{K}_x(\cdot, u_i)\boldsymbol{\alpha}_i$ et $g = \sum_j \mathcal{K}_x(\cdot, u_j)\boldsymbol{\beta}_j$ est défini comme suit :

$$\langle f, g \rangle_{\mathcal{H}_0} = \sum_{i,j} \langle \boldsymbol{\alpha}_i, \mathcal{K}_x(u_i, u_j)\boldsymbol{\beta}_j \rangle_{\mathcal{F}_y}.$$

La norme correspondante sur $\mathcal{H}_0$ est définie par $\|f\|^2_{\mathcal{H}_0} = \langle f, f \rangle_{\mathcal{H}_0}$. Le RKHS $\mathcal{H}_{\mathcal{K}_x}$ admettant $\mathcal{K}_x$ comme noyau reproduisant est ensuite construit en complétant l'ensemble $\mathcal{H}_0$ par les limites des suites de Cauchy. Pour alléger les notations, $\mathcal{H}_{\mathcal{K}_x}$ sera noté $\mathcal{H}$ par la suite.

Comme dans le cas scalaire, les RKHS fournissent un cadre théorique élégant pour les problèmes d'apprentissage régularisés par le biais des théorèmes de représentation.

4.3 Régularisation dans les RKHS pour les fonctions à valeurs vectorielles

Dans le cadre de la théorie des RKHS dédiée aux fonctions à valeurs vectorielles, Micchelli & Pontil [2005] ont prouvé un théorème de représentation dans le cas de l'apprentissage supervisé. Dans cette section, nous présentons ce théorème de représentation, ainsi que le modèle de la Kernel Ridge Regression à noyau à valeur opérateur proposé par Micchelli & Pontil [2005]. Nous exposons également un second modèle de régression à noyau à valeur opérateur, appelé *régression à marge maximale*, que nous avons développé.

On note $S_\ell = \{(u_i, \mathbf{y}_i)\}_{i=1}^{\ell} \subseteq \mathcal{U} \times \mathcal{F}_y$ l'ensemble d'apprentissage et $\mathcal{H}$ le RKHS ayant comme noyau reproduisant $\mathcal{K}_x : \mathcal{U} \times \mathcal{U} \to \mathcal{L}(\mathcal{F}_y)$.

Théorème 6 *[Micchelli & Pontil, 2005] Soit $\mathcal{L}$ une fonction de perte convexe, et $\lambda_1 > 0$ un paramètre de régularisation. La fonction $\hat{h}$ minimisant le problème d'optimisation suivant :*

$$\arg\min_{h \in \mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \mathcal{L}(h(u_i), \mathbf{y}_i) + \lambda_1 \|h\|_{\mathcal{H}}^2 \ ,$$

s'écrit sous la forme :

$$\hat{h}(\cdot) = \sum_{j=1}^{\ell} \mathcal{K}_x(\cdot, u_j)\mathbf{c}_j \ ,$$

où les coefficients $\mathbf{c}_j$ pour $j = 1, \cdots, \ell$ sont des vecteurs appartenant à l'espace de Hilbert $\mathcal{F}_y$.

Ce théorème signifie que la fonction minimisant le risque empirique régularisé $J(h)$ appartient au sous-espace vectoriel des fonctions engendrées par les combinaisons linéaires des fonctions de la forme $\mathcal{K}_x(\cdot, u_j)$, où $j = 1, \ldots, \ell$. Ainsi, on peut remarquer que le choix de la fonction noyau $\mathcal{K}_x$ a un fort impact sur le choix de la norme considérée, et par conséquent sur le type de régularisation imposée dans le problème d'optimisation.

Dans la suite, nous présentons deux modèles : la Kernel Ridge Regression et la régression à marge maximale, dans le cadre des fonctions à valeurs vectorielles. Pour cela, nous utilisons la forme de la solution obtenue avec le théorème de représentation pour la remplacer dans le problème d'optimisation, et ainsi trouver les coefficients $\mathbf{c}_j$ dans le cas des deux modèles.

4.3.1 Kernel Ridge Regression à noyau à valeur opérateur

Dans le cas de la fonction de perte correspondant aux moindres carrés, le problème d'optimisation s'exprime sous la forme suivante :

$$\arg\min_{h\in\mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \|h(u_i) - \mathbf{y}_i\|^2_{\mathcal{F}_y} + \lambda_1 \|h\|^2_{\mathcal{H}} \ . \tag{4.1}$$

Théorème 7 (Micchelli & Pontil [2005]) *Soient $\mathbf{c}_j \in \mathcal{F}_y, j = 1, \cdots, \ell$, les coefficients de l'expansion admise par le minimiseur $\hat{h}$ du problème d'optimisation (4.1). Les vecteurs $\mathbf{c}_j$ sont solutions des équations suivantes :*

$$\sum_{i=1}^{\ell} (\lambda_1 \delta_{ij} + \mathcal{K}_x(u_j, u_i))\mathbf{c}_i = \mathbf{y}_j \ , \tag{4.2}$$

où δ est le symbole de Kronecker : $\delta_{ii} = 1$ et $\forall j \neq i$, $\delta_{ij} = 0$.

La solution du système d'équations linéaires (4.2) peut se réécrire sous la forme :

$$\mathrm{vec}(C) = (\lambda_1 I_{\ell d} + \mathbf{K}_x)^{-1} \mathrm{vec}(Y_\ell), \tag{4.3}$$

où $Y_\ell = (\mathbf{y}_1, \ldots, \mathbf{y}_\ell)$, $C = (\mathbf{c}_1, \ldots, \mathbf{c}_\ell)$ sont des matrices de taille $d \times \ell$ qui contiennent respectivement les vecteur $\mathbf{y}_i$ et $\mathbf{c}_i$ pour $i = 1, \ldots, \ell$. $\mathbf{K}_x$ est une matrice de ℓ blocs par ℓ blocs, le bloc (j, k) de $\mathbf{K}_x$ étant de dimension $d \times d$ et correspondant à $\mathbf{K}_{jk} = \mathcal{K}_x(u_j, u_k), j, k \in \mathbb{N}_\ell$. Cette matrice peut être également vue comme une matrice scalaire de taille $\ell d \times \ell d$. D'après la définition 4, la matrice $\mathbf{K}_x$ est symétrique et semi-définie positive. Enfin, l'opérateur vec appliqué à une matrice A correspond au vecteur formé en empilant les colonnes de A.

La fonction h minimisant le problème d'optimisation (4.1) s'écrit alors sous la forme :

$$h(\cdot) = G_x \,\mathrm{vec}(C) = G_x(\lambda_1 I_{\ell d} + \mathbf{K}_x)^{-1} \mathrm{vec}(Y_\ell),$$

où $G_x = (\mathcal{K}_x(\cdot, u_1), \ldots, \mathcal{K}_x(\cdot, u_\ell))$ est une matrice de dimension $d \times \ell d$.

Dans le cas particulier où $\mathcal{F}_y = \mathbb{R}$, on retrouve la solution de la Kernel Ridge Regression présentée dans le chapitre 2.

4.3.2 Régression à marge maximale à noyau à valeur opérateur

Szedmak et al. [2005] ont proposé une formulation d'un séparateur à vaste marge, appelée *régression à marge maximale* (Maximum Margin Regression ou MMR), dans le cas de fonctions à valeurs vectorielles. Nous avons étendu ce cadre de régression au contexte de la théorie des RKHS pour les fonctions à valeurs vectorielles. Dans le cas supervisé, le problème d'optimisation correspondant est le suivant :

$$\arg\min_{h\in\mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \max(0, 1 - \langle \mathbf{y}_i, h(u_i)\rangle_{\mathcal{F}_y}) + \lambda_1 \|h\|^2_{\mathcal{H}}. \tag{4.4}$$

De façon similaire aux SVM, le problème (4.4) peut être exprimé d'après une formulation primale impliquant l'optimisation de $h \in \mathcal{H}$ et des variables d'écart $\xi_i \in \mathbb{R}$, $i = 1, \ldots, \ell$:

$$\begin{aligned} \min_{h \in \mathcal{H}, \{\xi_i\} \in \mathbb{R}} \quad & \lambda_1 \|h\|^2_{\mathcal{H}} + \sum_{i=1}^{\ell} \xi_i \\ \text{s.t.} \quad & \langle \mathbf{y}_i, h(u_i) \rangle_{\mathcal{F}_y} \geq 1 - \xi_i, i = 1, \ldots, \ell \\ & \xi_i \geq 0, i = 1, \ldots, \ell \end{aligned}$$

ainsi qu'une formulation duale qui est exprimée en fonction des paramètres du Lagrangien. Pour cela, on écrit le Lagrangien en utilisant la propriété reproduisante :

$$\mathcal{L}_a(h, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\eta}) = \lambda_1 \|h\|^2_{\mathcal{H}} + \sum_{i=1}^{\ell} \xi_i - \sum_{i=1}^{\ell} \alpha_i (\langle \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, h \rangle_{\mathcal{H}} - 1 + \xi_i) - \sum_{i=1}^{\ell} \eta_i \xi_i,$$

avec $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_\ell]^T \in \mathbb{R}^\ell$ et $\boldsymbol{\eta} = [\eta_1, \ldots, \eta_\ell]^T \in \mathbb{R}^\ell$ les multiplicateurs de Lagrange.
On annule les dérivées partielles du Lagrangien par rapport à ξ_i et h :

$$(1) \quad \frac{\partial \mathcal{L}_a}{\partial \xi_i} = 0 \Rightarrow 1 - \alpha_i - \eta_i = 0,$$

$$(2) \quad \frac{\partial \mathcal{L}_a}{\partial h} = 0 \Rightarrow h(\cdot) = \frac{1}{2\lambda_1} \sum_{i=1}^{\ell} \alpha_i \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i,$$

ce qui nous permet de formuler un Lagrangien réduit :

$$\mathcal{L}_r(\boldsymbol{\alpha}) = \frac{1}{4\lambda_1} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(u_j, u_i)\mathbf{y}_i, \mathbf{y}_j \rangle_{\mathcal{F}_y} - \frac{1}{2\lambda_1} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(u_j, u_i)\mathbf{y}_i, \mathbf{y}_j \rangle_{\mathcal{F}_y} + \sum_{i=1}^{\ell} \alpha_i.$$

La formulation duale du problème d'optimisation implique la résolution du problème quadratique suivant :

$$\begin{aligned} \min_{\boldsymbol{\alpha} \in \mathbb{R}^\ell} \quad & \frac{1}{4\lambda_1} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(u_j, u_i)\mathbf{y}_i, \mathbf{y}_j \rangle - \sum_{i=1}^{\ell} \alpha_i \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell \end{aligned} \tag{4.5}$$

pour lequel des solveurs efficaces existent.
Nous aurions également pu obtenir cette formulation duale en remplaçant h par l'expression fournie par le théorème de représentation et en dérivant ensuite par rapport aux coefficients $\mathbf{c}_i$ de cette formulation.

Comme dans le cas de la Kernel Ridge Regression, la solution du problème

d'optimisation peut s'écrire sous une forme matricielle :

$$h(\cdot) = \frac{1}{2\lambda_1} G_x \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha})). \tag{4.6}$$

Le vecteur $\boldsymbol{\alpha}$ est alors obtenu par la formulation duale suivante :

$$\begin{array}{ll} \min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^\ell} & \frac{1}{4\lambda_1} \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha}))^T \mathbf{K}_x \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha})) - \boldsymbol{\alpha}^T \mathbf{1} \\ \text{s.t.} & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell \end{array} \tag{4.7}$$

où $\mathbf{1}$ désigne le vecteur de longueur ℓ contenant uniquement des 1.

4.4 Régularisation semi-supervisée pour les fonctions à valeurs vectorielles

Comme nous l'avons vu dans la section 2.2.6, Belkin & Niyogi [2005]; Belkin et al. [2006] ont proposé un théorème de représentation pour des fonctions à valeurs réelles dans le cadre de l'apprentissage semi-supervisé. Un terme de régularisation est ajouté à la fonction de coût de sorte à renforcer la continuité de la fonction : les images d'entrées proches doivent, elles aussi, être proches.

Afin de bénéficier des données non étiquetées, nous établissons un théorème de représentation dédié à l'apprentissage des fonctions à valeurs vectorielles dans le cadre semi-supervisé. Le théorème 8 étend ainsi le théorème de représentation proposé par Belkin et al. [2006] aux fonctions à valeurs vectorielles. Il étend également le théorème 6 au cadre de l'apprentissage semi-supervisé.

Soit $\mathcal{L}$ une fonction de perte convexe. Etant donnés un ensemble de ℓ exemples étiquetés $\{(u_i, \mathbf{y}_i)\}_{i=1}^{\ell} \subseteq \mathcal{U} \times \mathcal{F}_y$ et un ensemble additionnel de n exemples non étiquetés $\{u_i\}_{i=\ell+1}^{\ell+n} \subseteq \mathcal{U}$, on considère le problème d'optimisation suivant :

$$\arg\min_{h\in\mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \mathcal{L}(h(u_i), \mathbf{y}_i) + \lambda_1 \|h\|_{\mathcal{H}}^2 + \lambda_2 \sum_{i,j=1}^{\ell+n} W_{ij} \|h(u_i) - h(u_j)\|_{\mathcal{F}_y}^2, \tag{4.8}$$

où λ_1, $\lambda_2 > 0$ sont deux paramètres de régularisation et où W est une matrice symétrique de taille $(\ell+n) \times (\ell+n)$ à valeurs positives mesurant la similarité entre les objets de l'ensemble d'entrée.
Le problème d'optimisation peut se réécrire sous la forme :

$$\arg\min_{h\in\mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \mathcal{L}(h(u_i), \mathbf{y}_i) + \lambda_1 \|h\|_{\mathcal{H}}^2 + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij} \langle h(u_i), h(u_j) \rangle_{\mathcal{F}_y},$$

où $L = D - W$ est le Laplacien du graphe associé à W, avec D la matrice diagonale de terme général $D_{ii} = \sum_{j=1}^{\ell+n} W_{ij}$.

Théorème 8 *La fonction minimisant le théorème d'optimisation* (4.8) *peut s'écrire sous la forme suivante :*

$$\hat{h}(\cdot) = \sum_{j=1}^{\ell+n} \mathcal{K}_x(\cdot, u_j)\mathbf{c}_j \ ,$$

où $\mathbf{c}_j, j = 1, \cdots, \ell + n$, *sont des vecteurs appartenant à* $\mathcal{F}_y$.

Preuve Définissons l'espace

$$\mathcal{H}_0 = \{h \in \mathcal{H} | h(\cdot) = \sum_{i=1}^{\ell+n} \mathcal{K}_x(\cdot, u_j)\mathbf{c}_j\}.$$

Soit $\mathcal{H}_0^\perp$ le sous-espace de $\mathcal{H}$ orthogonal à $\mathcal{H}_0$:

$$\mathcal{H}_0^\perp = \{g \in \mathcal{H} | \langle g(\cdot), h(\cdot)\rangle_\mathcal{H} = 0, \forall h \in \mathcal{H}_0\}$$

Toute fonction h appartenant à $\mathcal{H}$ admet une décomposition unique

$$h = h_0 + h_0^\perp,$$

où h_0 correspond à la projection de h sur $\mathcal{H}_0$ et $h_0^\perp$ à la projection sur l'espace $\mathcal{H}_0^\perp$.

D'après la propriété reproduisante, l'évaluation de la fonction h en tout point ne dépend pas de $h_0^\perp$:

$$\begin{aligned} \forall i = 1, \cdots, \ell + n,\ h(u_i) &= \langle h, \mathcal{K}_x(\cdot, u_i)\rangle_\mathcal{H} \\ &= \langle h_0, \mathcal{K}_x(\cdot, u_i)\rangle_\mathcal{H} + \langle h_0^\perp, \mathcal{K}_x(\cdot, u_i)\rangle_\mathcal{H} \\ &= \langle h_0, \mathcal{K}_x(\cdot, u_i)\rangle_\mathcal{H} \\ &= h_0(u_i). \end{aligned}$$

En utilisant la propriété d'orthogonalité, on peut décomposer la norme de h dans $\mathcal{H}$ en une somme de deux termes :

$$\|h\|_\mathcal{H}^2 = \|h_0\|_\mathcal{H}^2 + \|h_0^\perp\|_\mathcal{H}^2 + 2\langle h_0, h_0^\perp\rangle_\mathcal{H} = \|h_0\|_\mathcal{H}^2 + \|h_0^\perp\|_\mathcal{H}^2.$$

Ces deux propriétés nous permettent d'exprimer la fonction objectif J sous la forme suivante :

$$\mathcal{J}(h) = \mathcal{J}(h_0) + \lambda_1 \|h_0^\perp\|_\mathcal{H}^2.$$

On en déduit que la solution du problème (4.8) doit vérifier $h_0^\perp = 0$. Par conséquent, la solution s'écrit sous la forme suivante : $h(\cdot) = \sum_{j=1}^{\ell+n} \mathcal{K}_x(\cdot, u_j)\mathbf{c}_j$. ■

Comme précédemment, nous présentons les modèles obtenus dans le cas de la Kernel Ridge Regression et de la régression à marge maximale.

4.4.1 Kernel Ridge Regression à noyau à valeur opérateur

Dans le cas de la Kernel Ridge Regression semi-supervisée, le problème d'optimisation s'écrit sous la forme suivante :

$$\arg\min_{h\in\mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \|h(u_i) - \mathbf{y}_i\|^2_{\mathcal{F}_y} + \lambda_1 \|h\|^2_{\mathcal{H}} + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij} \langle h(u_i), h(u_j)\rangle_{\mathcal{F}_y}. \quad (4.9)$$

Théorème 9 *Les coefficients $\mathbf{c}_j \in \mathcal{F}_y, j = 1, \cdots, \ell+n$, apparaissant dans la forme admise par la solution $\hat{h}$ du problème d'optimisation (4.9) sont solutions du système d'équations suivant :*

$$J_j\, \mathbf{y}_j = J_j \sum_{i=1}^{\ell+n} \mathcal{K}_x(u_j, u_i)\mathbf{c}_i + \lambda_1 \mathbf{c}_j + 2\lambda_2 \sum_{i=1}^{\ell+n} L_{ij} \sum_{m=1}^{\ell+n} \mathcal{K}_x(u_i, u_m)\mathbf{c}_m\ , \quad (4.10)$$

où J_j de dimension $d \times d$ est la matrice identité lorsque $j \leq \ell$ et la matrice nulle lorsque $\ell < j \leq (\ell+n)$.

Preuve Nous remplaçons $h(u_i)$ par l'expression obtenue d'après le théorème de représentation dans le problème d'optimisation (4.9) :

$$\begin{aligned}\mathcal{J}(h) = &\sum_{i=1}^{\ell+n} J_i \| \sum_{j=1}^{\ell+n} \mathcal{K}_x(u_i, u_j)\mathbf{c}_j - \mathbf{y}_i\|^2_{\mathcal{F}_y} + \lambda_1 \sum_{i,j=1}^{\ell+n} \mathbf{c}_i^T \mathcal{K}_x(u_i, u_j)\mathbf{c}_j \\ &+ 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij} \sum_{k,m=1}^{\ell+n} \mathbf{c}_k^T \mathcal{K}_x(u_i, u_k)^T \mathcal{K}_x(u_j, u_m)\mathbf{c}_m.\end{aligned}$$

La dérivée partielle de $\mathcal{J}(h)$ par rapport à c_{j_0}, $1 \leq j_0 \leq \ell+n$, s'écrit :

$$\frac{\partial \mathcal{J}}{\partial c_{j_0}} = 2\sum_{i=1}^{\ell+n} \mathcal{K}_x(u_{j_0}, u_i) \left(J_i \sum_{j=1}^{\ell+n} \mathcal{K}_x(u_i, u_j)\mathbf{c}_j - J_i \mathbf{y}_i + \lambda_1 \mathbf{c}_i + 2\lambda_2 \sum_{j,m=1}^{\ell+n} L_{ij} \mathcal{K}_x(u_j, u_m)\mathbf{c}_m \right).$$

En annulant la dérivée partielle, nous obtenons : $\forall i = 1, \cdots, \ell+n$,

$$J_i \sum_{j=1}^{\ell+n} \mathcal{K}_x(u_i, u_j)\mathbf{c}_j + \lambda_1 \mathbf{c}_i + 2\lambda_2 \sum_{j,m=1}^{\ell+n} L_{ij} \mathcal{K}_x(u_j, u_m)\mathbf{c}_m = J_i\, \mathbf{y}_i.$$

Cela conclut la preuve. ■

Soient $C = (\mathbf{c}_1, \ldots, \mathbf{c}_{\ell+n})$, $Y_\ell = (\mathbf{y}_1, \ldots, \mathbf{y}_\ell)$ et $\mathbf{K}_x$ la matrice de $(\ell+n)$ blocs par $(\ell+n)$ blocs telle que $\mathbf{K}_{jk} = \mathcal{K}_x(u_j, u_k), j, k \in \mathbb{N}^{\ell+n}$. La solution du problème d'optimisation 4.10 peut être exprimée sous la forme matricielle suivante :

$$\mathrm{vec}(C) = \left(\lambda_1 I_{(\ell+n)d} + ((J^T J + 2\lambda_2 L) \otimes I_d)\mathbf{K}_x\right)^{-1} \mathrm{vec}(Y_\ell J), \quad (4.11)$$

où $\otimes$ désigne le produit de Kronecker, $I_{(\ell+n)d}$ la matrice identité de taille $(\ell+n)d \times (\ell+n)d$ et $J = (I_\ell, 0)$ la matrice de dimensions $\ell \times (\ell+n)$ qui contient une matrice identité de taille $\ell \times \ell$ sur le côté gauche et une matrice nulle de taille $\ell \times n$ sur le côté droit.

4.4.2 Régression à marge maximale à noyau à valeur opérateur

En utilisant la fonction de perte charnière, le problème d'optimisation correspondant à la régression à marge maximale dans le cas semi-supervisé s'écrit :

$$\arg\min_{h\in\mathcal{H}} \mathcal{J}(h) = \sum_{i=1}^{\ell} \max(0, 1-\langle \mathbf{y}_i, h(u_i)\rangle_{\mathcal{F}_y}) + \lambda_1 \|h\|^2_{\mathcal{H}} + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij}\langle h(u_i), h(u_j)\rangle_{\mathcal{F}_y}. \tag{4.12}$$

Comme dans le cas de l'apprentissage supervisé, nous donnons les formulations primale et duale du problème (4.12), par rapport aux variables d'optimisation correspondantes. L'expression de la fonctionnelle de coût dans l'espace primal s'écrit de la façon suivante :

$$\begin{aligned} \min_{h\in\mathcal{H},\{\xi_i\}\in\mathbb{R}} \quad & \lambda_1\|h\|^2_{\mathcal{H}} + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij}\langle h(u_i), h(u_j)\rangle_{\mathcal{F}_y} + \sum_{i=1}^{\ell} \xi_i \\ \text{s.t.} \quad & \langle \mathbf{y}_i, h(u_i)\rangle_{\mathcal{F}_y} \geq 1-\xi_i, i = 1, \ldots, \ell \\ & \xi_i \geq 0, i = 1, \ldots, \ell \end{aligned}$$

On écrit le Lagrangien :

$$\begin{aligned} \mathcal{L}_a(h, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\eta}) =& \lambda_1\|h\|^2_{\mathcal{H}} + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij}\langle h(u_i), h(u_j)\rangle_{\mathcal{F}_y} \\ & + \sum_{i=1}^{\ell} \xi_i - \sum_{i=1}^{\ell} \alpha_i(\langle \mathbf{y}_i, h(u_i)\rangle_{\mathcal{F}_y} - 1 + \xi_i) - \sum_{i=1}^{\ell} \eta_i\xi_i \\ & = \lambda_1\|h\|^2_{\mathcal{H}} + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij}\langle \mathcal{K}_x(\cdot, u_j)h(u_i), h\rangle_{\mathcal{H}} \\ & + \sum_{i=1}^{\ell} \xi_i - \sum_{i=1}^{\ell} \alpha_i(\langle \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, h\rangle_{\mathcal{H}} - 1 + \xi_i) - \sum_{i=1}^{\ell} \eta_i\xi_i. \end{aligned}$$

On différencie le Lagrangien par rapport à ξ_i et h :

$$(1) \quad \frac{\partial \mathcal{L}_a}{\partial \xi_i} = 0 \Rightarrow 1 - \alpha_i - \eta_i = 0$$

$$(2) \quad \frac{\partial \mathcal{L}_a}{\partial h} = 0 \Rightarrow 2\lambda_1 h + 4\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij}\mathcal{K}_x(\cdot, u_j)\mathcal{K}_x(u_i, \cdot)h = \sum_{i=1}^{\ell} \alpha_i \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i$$

$$\Rightarrow h = \frac{1}{2} B^{-1} \left(\sum_{i=1}^{\ell} \alpha_i \mathcal{K}_x(\cdot, u_i) \mathbf{y}_i \right),$$

où $B = \lambda_1 I_{(\ell+n)d} + 2\lambda_2 \sum_{i,j=1}^{\ell+n} L_{ij} \mathcal{K}_x(\cdot, u_j) \mathcal{K}_x(u_i, \cdot)$.

On formule un Lagrangien réduit :

$$\begin{aligned}
\mathcal{L}_r(\boldsymbol{\alpha}) &= \frac{1}{4}\lambda_1 \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle B^{-1}\mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j \rangle \\
&\quad + \frac{1}{2}\lambda_2 \sum_{k,m=1}^{\ell+n} L_{k,m} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(\cdot, u_m)\mathcal{K}_x(u_k, \cdot) B^{-1}\mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j \rangle \\
&\quad - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j \rangle + \sum_{i=1}^{\ell} \alpha_i \\
&= \frac{1}{4} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle (\lambda_1 I_{(\ell+n)d} + 2\lambda_2 \sum_{k,m=1}^{\ell+n} L_{k,m}\mathcal{K}_x(\cdot, u_m)\mathcal{K}_x(u_k, \cdot)) B^{-1}\mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j) \rangle \\
&\quad - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j) \rangle + \sum_{i=1}^{\ell} \alpha_i \\
\mathcal{L}_r(\boldsymbol{\alpha}) &= \sum_{i=1}^{\ell} \alpha_i - \frac{1}{4} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j \rangle.
\end{aligned}$$

La formulation duale du problème d'optimisation peut s'exprimer comme :

$$\begin{aligned}
\min_{\boldsymbol{\alpha} \in \mathbb{R}^{\ell}} \quad & \frac{1}{4} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j \langle \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i, B^{-1}\mathcal{K}_x(\cdot, u_j)\mathbf{y}_j \rangle - \sum_{i=1}^{\ell} \alpha_i \\
\text{s.t.} \quad & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell
\end{aligned} \tag{4.13}$$

Afin d'écrire la solution du problème d'optimisation sous forme matricielle, nous utilisons les deux relations suivantes :

$$\begin{aligned}
(1) \quad & \sum_{i,j=1}^{\ell+n} L_{ij}\mathcal{K}_x(\cdot, u_j)\mathcal{K}_x(u_i, \cdot) = G_x(L \otimes I_d)G_x^T, \\
(2) \quad & \sum_{i=1}^{\ell} \alpha_i \mathcal{K}_x(\cdot, u_i)\mathbf{y}_i = G_x \operatorname{vec}(Y_{\ell} \operatorname{diag}(\boldsymbol{\alpha}) J),
\end{aligned}$$

où $G_x = (\mathcal{K}_x(\cdot, u_1), \ldots, \mathcal{K}_x(\cdot, u_{\ell+n}))$.

Nous en déduisons alors que :

$$\begin{aligned} h(\cdot) &= \frac{1}{2}(\lambda_1 I_{(\ell+n)d} + 2\lambda_2 G_x(L \otimes I_d)G_x^T)^{-1} G_x \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha})J) \\ &= \frac{1}{2} G_x(\lambda_1 I_{(\ell+n)d} + 2\lambda_2 (L \otimes I_d)\mathbf{K}_x)^{-1} \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha})J), \end{aligned}$$

et le vecteur $\boldsymbol{\alpha}$ peut être obtenu d'après la formulation duale suivante :

$$\begin{aligned} \min_{\boldsymbol{\alpha}\in\mathbb{R}^\ell} \quad & \frac{1}{4} \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha})J)^T \left(\lambda_1 I_{(\ell+n)d} + 2\lambda_2 \mathbf{K}_x (L \otimes I_d)\right)^{-1} \mathbf{K}_x \operatorname{vec}(Y_\ell \operatorname{diag}(\boldsymbol{\alpha})\, J) \\ & - \boldsymbol{\alpha}^T \mathbf{1} \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell \end{aligned} \tag{4.14}$$

4.5 Noyaux décomposables

Les noyaux décomposables sont des noyaux à valeur opérateur particuliers qui ont été introduits pour la résolution de problèmes d'apprentissage multitâches [Evgeniou et al., 2005; Micchelli & Pontil, 2005; Sheldon, 2008]. Ces noyaux sont également appelés noyaux produit ou noyaux séparables. Les noyaux à valeur opérateur décomposables s'expriment par un produit entre une fonction noyau en entrée et une matrice A :

$$\begin{aligned} \mathcal{K}_x : \quad \mathcal{U} \times \mathcal{U} &\to \mathcal{L}(\mathcal{F}_y) \\ (u, u') &\mapsto \kappa_x(u, u') \times A \end{aligned} \tag{4.15}$$

où κ_x est un noyau scalaire sur $\mathcal{U} \times \mathcal{U}$ et A est une matrice semi-définie positive de taille $d \times d$ qui code la façon dont les composantes de sortie sont reliées entre elles.

Le noyau $\mathcal{K}_x$ satisfait les propriétés d'un noyau semi-défini positif : il est symétrique, et donc hermitien. De plus, la propriété de semi-définie positivité de κ_x et A entraîne la semi-définie positivité de $\mathcal{K}_x$:

$$\begin{aligned} &\forall m \in \mathbb{N}, \forall \{(u_i, \mathbf{y}_i)\}_{i=1}^m \subseteq \mathcal{U} \times \mathcal{F}_y, \\ &\sum_{i,j=1}^m \langle \mathbf{y}_i, \mathcal{K}_x(u_i, u_j)\mathbf{y}_j \rangle_{\mathcal{F}_y} = \sum_{i,j=1}^m \kappa_x(u_i, u_j) \langle \mathbf{y}_i, A\mathbf{y}_j \rangle_{\mathcal{F}_y} \geq 0. \end{aligned}$$

Dans le cadre de l'apprentissage multitâches, $\mathcal{F}_y = \mathbb{R}^d$ est un espace de dimension finie et la matrice A code les relations existant entre les d tâches. Cette matrice peut être estimée à partir des données de l'ensemble d'apprentissage ou bien être apprise en même temps que les autres paramètres du modèle [Dinuzzo et al., 2011].

4.5.1 Noyaux décomposables et régularisation

Dans le cas où un noyau à valeur opérateur décomposable est utilisé, il est possible d'exprimer la norme de la fonction h dans $\mathcal{H}$ en fonction de la matrice A.

Proposition 10 *[Sheldon, 2008] Etant donné un noyau décomposable $\mathcal{K}_x$, dont la forme est donnée en (4.15), alors la norme de toute fonction h dans le RKHS correspondant $\mathcal{H}$ peut s'écrire comme*

$$\|h\|_{\mathcal{H}}^2 = \sum_{\ell,q=1}^{d} A^{\dagger}_{\ell q} \langle h^{\ell}, h^{q} \rangle, \tag{4.16}$$

où h^{ℓ} désigne la ℓ-ième coordonnée de la fonction h et où $A^{\dagger}$ représente la matrice pseudo-inverse de A [Penrose, 1955], c'est à dire la matrice qui satisfait $AA^{\dagger}A = A$.

Cette proposition permet de définir un noyau décomposable, et plus particulièrement la matrice A intervenant dans ce noyau, en spécifiant un terme de régularisation. Différents noyaux décomposables ont été proposés de cette manière. Nous présentons dans la suite des exemples de normes qui s'expriment sous la même forme que celle de l'équation (4.16).

Régularisation basée sur un graphe Sheldon [2008]; Micchelli & Pontil [2004] ont proposé un terme de régularisation, qui en plus du terme de régularisation habituel, force les composantes de h à être plus ou moins proches les unes des autres. Les valeurs de cette similarité sont encodées par une matrice symétrique de poids positifs M, qui est de taille $d \times d$:

$$\Omega(h) = \frac{1}{2}\, \mu \sum_{\ell,q=1}^{d} \|h^{\ell} - h^{q}\|^2 M_{\ell q} + \lambda \sum_{\ell=1}^{d} \|h^{\ell}\|^2 M_{\ell\ell}, \tag{4.17}$$

où μ et λ désignent deux paramètres de régularisation.
Ce terme peut se réécrire sous la forme :

$$\Omega(h) = \sum_{\ell,q=1}^{d} (\mu L_M + \lambda I_d)_{\ell q} \langle h^{\ell}, h^{q} \rangle, \tag{4.18}$$

où $L_M = D - M$, est le laplacien associé à M, avec D la matrice diagonale de terme général $D_{ii} = \sum_{k=1}^{d} M_{ik}$.
En définissant le noyau à valeur opérateur $\mathcal{K}_x(u, u') = \kappa_x(u, u')(\mu L_M + \lambda I_d)^{-1}$ et en utilisant la proposition 10, on trouve que $\|h\|_{\mathcal{H}}^2 = \Omega(h)$.

Régularisation basée sur une partition des composantes Evgeniou et al. [2005] ont introduit dans le cas de l'apprentissage multitâches un autre exemple de terme de régularisation, qui a été également reformulé par Jacob et al. [2008]. L'idée consiste à partitionner les composantes de la fonction à apprendre (ou tâches) en r

clusters disjoints, de sorte que les composantes dans chaque cluster soient similaires. On note m_c le nombre de composantes appartenant au cluster c, et on définit $I(c)$ comme l'ensemble contenant les indices des composantes assignées au cluster c. Le terme de régularisation proposé impose aux composantes appartenant à un même cluster c d'être proches de la moyenne du cluster $\overline{h_c} = \frac{1}{m_c}\sum_{q\in I(c)} h^q$:

$$\Omega(h) = \mu_1 \sum_{c=1}^{r} \sum_{\ell \in I(c)} \|h^\ell - \overline{h_c}\| + \mu_2 \sum_{c=1}^{r} m_c \|\overline{h_c}\|^2, \tag{4.19}$$

où μ_1 et μ_2 sont deux constantes positives.

L'assignation des d tâches aux r clusters est décrite dans une matrice binaire E de taille $d \times r$. On introduit la matrice $M = E(E^T E)^{-1}E^T$. Cette matrice est telle que $M_{\ell q} = 1/m_c$ si ℓ et q appartiennent au même cluster et $M_{\ell q} = 0$ sinon.

Il a été montré [Baldassarre et al., 2012] que le terme de régularisation peut se réécrire sous la forme suivante :

$$\Omega(h) = \sum_{\ell,q=1}^{d} \langle h^\ell, h^q \rangle G_{\ell q},$$

où $G = \mu_1 I_d + (\mu_2 - \mu_1)M$.
Le noyau décomposable correspondant à ce terme de régularisation est donc $\mathcal{K}_x(u,u') = \kappa_x(u,u')G^\dagger$.

Régularisation basée sur une similarité commune Micchelli & Pontil [2004] ont proposé le noyau décomposable suivant :

$$\mathcal{K}_x(u,u') = \kappa_x(u,u')(\frac{1}{\mu}\mathbf{1}_d + I_d), \tag{4.20}$$

où $\mathbf{1}_d$ est une matrice de taille $d \times d$, dont toutes les valeurs sont égales à 1 et I_d est la matrice identité de taille $d \times d$.
Le terme de régularisation associé à ce noyau est :

$$\Omega(h) = \frac{d}{d+\mu}\sum_{\ell=1}^{d}\|h^\ell - \frac{1}{d}\sum_{q=1}^{d} h^q\|^2 + \frac{\mu}{d+\mu}\sum_{\ell=1}^{d}\|h^\ell\|^2. \tag{4.21}$$

Le premier terme impose à chaque composante h^ℓ d'être proche de la moyenne $\overline{h} = \frac{1}{d}\sum_{q=1}^{d} h^q$, tandis que le deuxième terme est le terme standard pour la régularisation. Le paramètre μ contrôle le compromis entre les deux termes. Si μ est petit, les tâches sont liées, tandis qu'une grande valeur de μ indique que les tâches sont apprises indépendamment les unes des autres.

4.5.2 Modèles pour un noyau décomposable général

Nous détaillons à présent les modèles obtenus pour la régression avec un noyau à valeur opérateur décomposable de la forme : $\mathcal{K}_x(u,u') = \kappa_x(u,u') \times A$.

On note $X_\ell = (x(u_1),\dots,x(u_\ell))$ la matrice de dimension $dim(\mathcal{F}_x) \times \ell$ et $X_{\ell+n} = (x(u_1),\dots,x(u_{\ell+n}))$. Soit $K_{X_\ell} = X_\ell^T X_\ell$ et $K_{X_{\ell+n}} = X_{\ell+n}^T X_{\ell+n}$ les matrice de Gram respectives associées au noyau κ_x sur les ensembles $\mathcal{U}_\ell$ et $\mathcal{U}_{\ell+n}$. La matrice de Gram $\mathbf{K}_x$ associée au noyau $\mathcal{K}_x$ s'exprime alors sous la forme $K_{X_\ell} \otimes A$ dans le cas supervisé, et $K_{X_{\ell+n}} \otimes A$ dans le cas semi-supervisé.

Nous utiliserons dans la suite les propriétés du produit de Kronecker et de l'opérateur vec qui sont décrites dans l'annexe D.

Kernel Ridge Regression

La fonction $\hat{h}$ minimisant le problème d'optimisation de la Kernel Ridge Regression dans le cas supervisé (4.1) et pour un noyau à valeur opérateur décomposable peut être exprimée sous la forme :

$$\forall u \in \mathcal{U}, \hat{h}(u) = A \sum_{i=1}^{\ell} \kappa_x(u,u_i)\mathbf{c}_i = ACX_\ell^T x(u).$$

En utilisant la proposition 15 de l'annexe D, ainsi que la solution obtenue à l'équation (4.3), on trouve que

$$\begin{aligned} \forall u \in \mathcal{U}, \hat{h}(u) &= (x(u)^T X_\ell) \otimes A \operatorname{vec}(C) \\ &= (x(u)^T X_\ell \otimes A) \left(\lambda_1 I_{\ell d} + K_{X_\ell} \otimes A\right)^{-1} \operatorname{vec}(Y_\ell). \end{aligned} \tag{4.22}$$

Ainsi le calcul de la solution $\hat{h}$ nécessite de calculer l'inverse de la matrice $(\lambda_1 I_{\ell d} + K_{X_\ell} \otimes A)$, dont la dimension est $\ell d \times \ell d$.

A étant une matrice réelle symétrique, on peut écrire sa décomposition en valeurs propres et vecteurs propres :

$$A = E\Gamma E^T = \sum_{i=1}^{d} \gamma_i \mathbf{e}_i \mathbf{e}_i^T,$$

où $E = (\mathbf{e}_1,\dots,\mathbf{e}_d)$ est une matrice de taille $d \times d$ et $\Gamma = \operatorname{diag}(\gamma_1,\dots,\gamma_d)$ est une matrice diagonale contenant les valeurs propres de A. En considérant la décomposition en valeurs propres de A, on peut montrer que le calcul de la solution $\hat{h}(u)$ revient à résoudre d problèmes indépendants.

Proposition 11 *La fonction $\hat{h}$ minimisant le problème d'optimisation de la régression ridge supervisée dans le cas d'un noyau à valeur opérateur décomposable* (4.1)

peut être exprimée comme :

$$\forall u \in \mathcal{U}, \hat{h}(u) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell (\lambda_1 I_\ell + \gamma_j K_{X_\ell})^{-1} X_\ell^T x(u).$$

Dans le cas semi-supervisé, l'expression de la fonction $\hat{h}$ devient :

$$\forall u \in \mathcal{U}, \hat{h}(u) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell J \left(\lambda_1 I_{\ell+n} + \gamma_j K_{X_{\ell+n}} (J^T J + 2\lambda_2 L)\right)^{-1} X_{\ell+n}^T x(u).$$

Dans le cas supervisé, la complexité algorithmique est de $O((\ell d)^3)$ pour un noyau à valeur opérateur général. Alors que pour un noyau à valeur opérateur décomposable, la détermination de la solution nécessite de calculer la décomposition en vecteurs propres et valeurs propres de A, ainsi que d'inverser une matrice de dimension $\ell \times \ell$. Par conséquent, la complexité algorithmique est dans ce cas de l'ordre de $O(d^3 + \ell^3)$. Dans le cas de l'apprentissage semi-supervisé, les complexités algorithmiques pour ces deux noyaux sont respectivement de $O(((\ell + n)d)^3)$ et de $O(d^3 + (\ell + n)^3)$. L'utilisation du deuxième noyau apparaît donc très intéressante concernant ce point.

Preuve Nous commençons par montrer la proposition dans le cas semi-supervisé, puis nous en déduirons la solution obtenue dans le cas supervisé. Pour cela, nous partons de l'équation (4.11), en considérant un noyau à valeur opérateur décomposable :

$$\mathrm{vec}(C) = \left(\lambda_1 I_{(\ell+n)d} + M \otimes A\right)^{-1} \mathrm{vec}(Y_\ell J),$$

où $M = ((J^T J + 2\lambda_2 L) K_{X_{\ell+n}})$.
On utilise les matrices de vec-permutation $P_{(\ell+n)d}$ et$P_{d(\ell+n)}$ qui permettent d'exprimer $\mathrm{vec}(C^T)$ en fonction de $\mathrm{vec}(C)$ (voir annexe D) :

$$\begin{aligned}
\mathrm{vec}(C^T) &= P_{d(\ell+n)} \,\mathrm{vec}(C) \\
&= P_{d(\ell+n)} \left(\lambda_1 I_{(\ell+n)d} + P_{(\ell+n)d}(A \otimes M) P_{d(\ell+n)}\right)^{-1} \mathrm{vec}(Y_\ell J) \\
&= \left(\lambda_1 I_{(\ell+n)d} + P_{d(\ell+n)} P_{(\ell+n)d}(A \otimes M)\right)^{-1} P_{d(\ell+n)} \,\mathrm{vec}(Y_\ell J) \\
&= \left(\lambda_1 I_{(\ell+n)d} + A \otimes M\right)^{-1} \mathrm{vec}(J^T Y_\ell^T) \\
&= \left(\lambda_1 I_{(\ell+n)d} + E\Gamma E^T \otimes M\right)^{-1} \mathrm{vec}(J^T Y_\ell^T).
\end{aligned}$$

On multiplie chaque côté de l'équation par $(E^T \otimes I_{\ell+n})$:

$$\begin{aligned}
&(E^T \otimes I_{\ell+n}) \,\mathrm{vec}(C^T) = \\
&\quad (E^T \otimes I_{\ell+n}) \left(\lambda_1 I_{(\ell+n)d} + (E \otimes I_{\ell+n})(\Gamma \otimes M)(E^T \otimes I_{\ell+n})\right)^{-1} \mathrm{vec}(J^T Y_\ell^T).
\end{aligned}$$

En utilisant le fait que $\mathrm{vec}(AXB) = (B^T \otimes A)\,\mathrm{vec}(X)$ et que $E^T E = I_d$, on trouve que :

$$\mathrm{vec}(C^T E) = (\lambda_1 I_{(\ell+n)d} + \Gamma \otimes M)^{-1} \mathrm{vec}(J^T Y_\ell^T E).$$

La matrice $(\lambda_1 I_{(\ell+n)d} + \Gamma \otimes M)$ étant diagonale par blocs, on en déduit que :

$$C^T \mathbf{e}_i = (\lambda_1 I_{\ell+n} + \gamma_i M)^{-1} J^T Y_\ell^T \mathbf{e}_i, \text{ pour } i = 1, \ldots, \ell+n.$$

On peut alors exprimer le modèle h sous la forme suivante :

$$\begin{aligned}\forall u \in \mathcal{U}, \hat{h}(u) &= ACX_{\ell+n}^T x(u) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T C X_{\ell+n}^T x(u) \\ &= \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell J (\lambda_1 I_{\ell+n} + \gamma_j K_{X_{\ell+n}} (J^T J + 2\lambda_2 L))^{-1} X_{\ell+n}^T x(u).\end{aligned}$$

Dans le cas supervisé ($\lambda_2 = 0$), le modèle h s'écrit sous la forme :

$$\forall u \in \mathcal{U}, \hat{h}(u) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell (\lambda_1 I_\ell + \gamma_j K_{X_\ell})^{-1} X_\ell^T x(u).$$

Cela conclut la preuve. ■

Régression MMR

Nous considérons maintenant le problème d'optimisation correspondant à la régression MMR.

Proposition 12 *Etant donné le noyau à valeur opérateur $\mathcal{K}_x(u, u') = \kappa_x(u, u') \times A$, la solution du problème d'optimisation (4.4) s'écrit :*

$$\hat{h}(u) = \frac{1}{2\lambda_1} A Y_\ell \, diag\,(\boldsymbol{\alpha}) X_\ell^T x(u).$$

Le vecteur de coefficients $\boldsymbol{\alpha}$ est obtenu en résolvant le problème suivant :

$$\begin{aligned} \min_{\boldsymbol{\alpha} \in \mathbb{R}^\ell} \quad & \frac{1}{4\lambda_1} \boldsymbol{\alpha}^T (Y_\ell^T A Y_\ell \circ K_{X_\ell}) \boldsymbol{\alpha} - \boldsymbol{\alpha}^T \mathbf{1} \\ s.t. \quad & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell \end{aligned}$$

où $\circ$ désigne le produit terme à terme entre deux matrices.
Dans le cas semi-supervisé, la solution du problème d'optimisation s'écrit :

$$\hat{h}(u) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell \, diag\,(\boldsymbol{\alpha}) J \left(2\lambda_1 I_{\ell+n} + 4\gamma_j \lambda_2 K_{X_{\ell+n}} L\right)^{-1} X_{\ell+n}^T x(u),$$

et la formulation duale permettant d'obtenir $\boldsymbol{\alpha}$ *devient :*

$$\begin{aligned} \min_{\boldsymbol{\alpha}\in\mathbb{R}^\ell} \quad & \frac{1}{2}\boldsymbol{\alpha}^T(\sum_{i=1}^{d}\gamma_i Y_\ell^T \mathbf{e}_i\mathbf{e}_i^T Y_\ell \circ J(2\lambda_1 I_{\ell+n} + 4\lambda_2\gamma_i K_{X_{\ell+n}}L)^{-1}K_{X_{\ell+n}}J^T)\boldsymbol{\alpha} - \boldsymbol{\alpha}^T\mathbf{1} \\ s.t. \quad & 0 \leq \alpha_i \leq 1, i = 1,\ldots,\ell \end{aligned}$$

Preuve Nous définissons $Z_\ell = Y_\ell \operatorname{diag}(\boldsymbol{\alpha})J$. Pour montrer la propriété, nous partons de l'expression du Lagrangien obtenue dans l'équation (4.14) et nous remplaçons A par sa décomposition en vecteurs propres et valeurs propres :

$$\begin{aligned} \mathcal{L}_a(\boldsymbol{\alpha}) &= -\frac{1}{4}\operatorname{vec}(Z_\ell)^T\left(\lambda_1 I_{(\ell+n)d} + 2\lambda_2 K_{X_{\ell+n}}L\otimes A\right)^{-1}(K_{X_{\ell+n}}\otimes A)\operatorname{vec}(Z_\ell) + \boldsymbol{\alpha}^T\mathbf{1} \\ &= -\frac{1}{4}\operatorname{vec}(Z_\ell)^T\left(\lambda_1 I_{(\ell+n)d} + 2\lambda_2 (I_{\ell+n}\otimes E)(K_{X_{\ell+n}}L\otimes\Gamma)(I_{\ell+n}\otimes E^T)\right)^{-1} \\ &\quad (I_{\ell+n}\otimes E)(K_{X_{\ell+n}}\otimes\Gamma)(I_{\ell+n}\otimes E^T)\operatorname{vec}(Z_\ell) + \boldsymbol{\alpha}^T\mathbf{1}. \end{aligned}$$

D'après la propriété 14, $(I_{\ell+n}\otimes E^T)\operatorname{vec}(Z_\ell) = \operatorname{vec}(E^T Z_\ell I_{\ell+n})$, d'où

$$\mathcal{L}_a(\boldsymbol{\alpha}) = -\frac{1}{4}\operatorname{vec}\left(E^T Z_\ell\right)^T\left(\lambda_1 I_{(\ell+n)d} + 2\lambda_2 K_{X_{\ell+n}}L\otimes\Gamma\right)^{-1}(K_{X_{\ell+n}}\otimes\Gamma)\operatorname{vec}(E^T Z_\ell) + \boldsymbol{\alpha}^T\mathbf{1}.$$

En utilisant les matrices de vec-permutation, on peut montrer que :

$$\mathcal{L}_a(\boldsymbol{\alpha}) = -\frac{1}{4}\operatorname{vec}\left(Z_\ell^T E\right)^T\left(\lambda_1 I_{(\ell+n)d} + 2\lambda_2\Gamma\otimes K_{X_{\ell+n}}L\right)^{-1}(\Gamma\otimes K_{X_{\ell+n}})\operatorname{vec}\left(Z_\ell^T E\right) + \boldsymbol{\alpha}^T\mathbf{1}.$$

Comme $(\lambda_1 I_{(\ell+n)d} + 2\lambda_2\Gamma\otimes K_{X_{\ell+n}}L)$ est une matrice diagonale par blocs, on peut écrire :

$$\begin{aligned} \mathcal{L}_a(\boldsymbol{\alpha}) &= -\frac{1}{4}\sum_{i=1}^{d}\mathbf{e}_i^T Z_\ell(\lambda_1 I_{\ell+n} + 2\lambda_2\gamma_i K_{X_{\ell+n}}L)^{-1}\gamma_i K_{X_{\ell+n}}Z_\ell^T\mathbf{e}_i + \boldsymbol{\alpha}^T\mathbf{1} \\ &= -\frac{1}{4}\sum_{i=1}^{d}\gamma_i\operatorname{trace}\left(Y_\ell^T\mathbf{e}_i\mathbf{e}_i^T Y_\ell\operatorname{diag}(\boldsymbol{\alpha})J(\lambda_1 I_{\ell+n} + 2\lambda_2\gamma_i K_{X_{\ell+n}}L)^{-1}K_{X_{\ell+n}}J^T\operatorname{diag}(\boldsymbol{\alpha})\right) \\ &\quad + \boldsymbol{\alpha}^T\mathbf{1}. \end{aligned}$$

En utilisant le fait que $\mathbf{y}^T(A\circ B)\mathbf{x} = \operatorname{trace}(\operatorname{diag}(\mathbf{y})^T A\operatorname{diag}(\mathbf{x})B^T)$, le Lagrangien peut se réécrire sous la forme :

$$\mathcal{L}_a(\boldsymbol{\alpha}) = -\frac{1}{4}\sum_{i=1}^{d}\gamma_i\,\boldsymbol{\alpha}^T(Y_\ell^T\mathbf{e}_i\mathbf{e}_i^T Y_\ell\circ J(\lambda_1 I_{\ell+n} + 2\lambda_2\gamma_i K_{X_{\ell+n}}L)^{-1}K_{X_{\ell+n}}J^T)\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}.$$

Dans le cas supervisé ($\lambda_2 = 0$), l'expression du Lagrangien devient :

$$\mathcal{L}_a(\boldsymbol{\alpha}) = -\frac{1}{4\lambda_1}\sum_{i=1}^{d}\gamma_i\,\boldsymbol{\alpha}^T(Y_\ell^T\mathbf{e}_i\mathbf{e}_i^TY_\ell \circ K_{X_\ell})\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}$$
$$= -\frac{1}{4\lambda_1}\boldsymbol{\alpha}^T(Y_\ell^TAY_\ell \circ K_{X_\ell})\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\mathbf{1}.$$

Cela conclut la preuve. ■

4.5.3 Sélection des hyperparamètres

Les modèles que nous avons présentés dépendent du choix des hyperparamètres, tels que les paramètres des noyaux ou les paramètres de régularisation. Pour sélectionner ces paramètres, on peut choisir a posteriori ceux qui minimisent l'erreur mesurée par validation croisée selon un certain critère (voir chapitre 2). Dans le cas des noyaux scalaires, l'erreur obtenue par la validation croisée leave-one-out (LOO-CV) peut être exprimée par une formule close [Golub et al., 1979]. Cette procédure de validation croisée consiste à retirer une observation de l'ensemble d'apprentissage et à prédire sa valeur à partir du reste de l'ensemble d'apprentissage. Cette étape est répétée pour chaque exemple de l'ensemble d'apprentissage.

Le critère PRESS (Predicted Residual Sum of Squares) [Allen, 1974; Stone, 1974] correspond à la somme des différences au carré entre la vraie valeur et la valeur prédite sur l'ensemble d'apprentissage.

Soit $\mathcal{S} = \{(u_1, \mathbf{y}_1), \ldots, (u_\ell, \mathbf{y}_\ell)\}$ un ensemble de ℓ points étiquetés. On définit $\mathcal{S}^i$, $1 \leq i \leq \ell$, comme l'ensemble d'apprentissage auquel le i^{th} exemple a été retiré :

$$\mathcal{S}^i = \{(u_1, \mathbf{y}_1), \ldots, (u_{i-1}, \mathbf{y}_{i-1}), (u_{i+1}, \mathbf{y}_{i+1}), \ldots, (u_\ell, \mathbf{y}_\ell)\}.$$

On note la valeur au point u_i de la fonction h apprise sur $\mathcal{S}^i$ par $h_{\mathcal{S}^i}(u_i)$. La i-ième erreur de la validation croisée leave-one-out correspond à $\|\mathbf{y}_i - h_{\mathcal{S}^i}(u_i)\|^2_{\mathcal{F}_y}$. L'expression du critère PRESS est la suivante :

$$PRESS = \sum_{i=1}^{\ell}\|\mathbf{y}_i - h_{\mathcal{S}^i}(u_i)\|^2_{\mathcal{F}_y}. \tag{4.23}$$

Dans le cas des fonctions à valeurs scalaires, il a été montré qu'il était possible d'évaluer ce critère sans avoir à calculer explicitement $h_{\mathcal{S}^i}(u_i)$ pour $i = 1, \ldots, \ell$ et pour chaque valeur de la grille des paramètres [Allen, 1974; Stone, 1974; Rifkin & Lippert, 2007]. Dans la suite, nous étendons cette propriété au cas des fonctions à valeurs vectorielles. Nous considérons pour cela le cas où le noyau à valeur opérateur peut être écrit sous la forme d'un noyau décomposable.

Supposons que l'on connaisse $h_{\mathcal{S}^i}$, on peut alors définir la matrice $Y_\ell^i =$

$(\mathbf{y}_1^i, \ldots, \mathbf{y}_\ell^i)$, où le vecteur $\mathbf{y}_j^i$ est donné par :

$$\mathbf{y}_j^i = \begin{cases} \mathbf{y}_j & \text{si } j \neq i \\ h_{\mathcal{S}^i}(u_i) & \text{si } j = i \end{cases}.$$

En utilisant Y_ℓ^i à la place de Y_ℓ dans le problème d'optimisation, la solution optimale sera $h_{\mathcal{S}^i}$, c'est à dire :

$$\forall i = 1, \ldots, \ell,\ h_{\mathcal{S}^i}(u_i) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell^i (B_j X_\ell)_i,$$

où la valeur deB_j est donnée dans le tableau 4.1.

$B_j =$	Apprentissage supervisé	Apprentissage semi-supervisé
IOKR-ridge	$(\lambda_1 I_\ell + \gamma_j K_{X_\ell})^{-1} X_\ell^T$	$J(\lambda_1 I_{\ell+n} + \gamma_j K_{X_{\ell+n}}(J^T J + 2\lambda_2 L))^{-1} X_{\ell+n}^T$
IOKR-margin	$\frac{1}{2\lambda_1}\,\mathrm{diag}\,(\alpha) X_\ell^T$	$\mathrm{diag}\,(\alpha) J(2\lambda_1 I_{\ell+n} + 4\gamma_j \lambda_2 K_{X_{\ell+n}} L)^{-1} X_{\ell+n}^T$

TABLE 4.1 – Les différents modèles de la méthode IOKR dans le cas d'un noyau décomposable général.

Pour obtenir une expression de $h_{\mathcal{S}^i}(u_i)$, on calcule la différence entre $h_{\mathcal{S}^i}(u_i)$ et $h_{\mathcal{S}}(u_i)$:

$$\begin{aligned} h_{\mathcal{S}^i}(u_i) - h_{\mathcal{S}}(u_i) &= \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T (Y_\ell^i - Y_\ell)(B_j X_\ell)_i \\ &= \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T \sum_{k=1}^{\ell} (\mathbf{y}_k^i - \mathbf{y}_k)(B_j X_\ell)_{ki} \\ h_{\mathcal{S}^i}(u_i) - h_{\mathcal{S}}(u_i) &= \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T (h_{\mathcal{S}^i}(u_i) - \mathbf{y}_i)(B_j X_\ell)_{ii}, \end{aligned}$$

ce qui conduit à :

$$h_{\mathcal{S}^i}(u_i) - \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T h_{\mathcal{S}^i}(u_i)(B_j X_\ell)_{ii} = h_{\mathcal{S}}(u_i) - \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T \mathbf{y}_i (B_j X_\ell)_{ii}.$$

On note $LOOV = (h_{\mathcal{S}^1}(u_1), \ldots, h_{\mathcal{S}^\ell}(u_\ell))$ la matrice contenant les valeurs des vecteurs obtenus par LOO-CV pour chaque exemple de l'ensemble d'apprentissage.

L'équation ci-dessus peut se réécrire :

$$LOOV - \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T LOOV \operatorname{diag}(B_j X_\ell) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T Y_\ell (B_j X_\ell - \operatorname{diag}(B_j X_\ell)).$$

En appliquant l'opérateur vec de part et d'autre de l'équation, on trouve que :

$$\left(I - \sum_{j=1}^{d} \gamma_j (\operatorname{diag}(B_j X_\ell)) \otimes (\mathbf{e}_j \mathbf{e}_j^T) \right) \operatorname{vec}(LOOV) = \sum_{j=1}^{d} \gamma_j (B_j X_\ell - \operatorname{diag}(B_j X_\ell)) \otimes (\mathbf{e}_j \mathbf{e}_j^T) \operatorname{vec}(Y_\ell).$$

La statistique PRESS d'Allen peut alors être exprimée sous la forme suivante :

$$\begin{aligned} PRESS &= \| \operatorname{vec}(Y_\ell) - \operatorname{vec}(LOOV) \|^2 \\ &= \| \operatorname{vec}(Y_\ell) - (I - \sum_{j=1}^{d} \gamma_j \operatorname{diag}(B_j X_\ell) \otimes \mathbf{e}_j \mathbf{e}_j^T)^{-1} \\ &\qquad (\sum_{j=1}^{d} \gamma_j (B_j X_\ell - \operatorname{diag}(B_j X_\ell)) \otimes (\mathbf{e}_j \mathbf{e}_j^T) \operatorname{vec}(Y_\ell)) \|^2 \\ &= \| \left(I - \sum_{j=1}^{d} \gamma_j \operatorname{diag}(B_j X_\ell) \otimes \mathbf{e}_j \mathbf{e}_j^T \right)^{-1} \left(I - \sum_{j=1}^{d} \gamma_j (B_j X_\ell) \otimes \mathbf{e}_j \mathbf{e}_j^T \right) \operatorname{vec}(Y_\ell) \|^2. \end{aligned}$$

Nous avons donc montré qu'il était possible d'évaluer le critère PRESS à partir d'une expression sous forme close dans le cas de fonctions à valeurs vectorielles et cela, lorsqu'un noyau à valeur opérateur décomposable est utilisé.

Conclusion :
Dans ce chapitre, nous avons présenté une nouvelle méthode pour le problème de la prédiction de sorties structurées dans le cadre de l'apprentissage semi-supervisé. Les noyaux à valeur opérateur et la théorie des espaces de Hilbert à noyau reproduisant pour les fonctions à valeurs vectorielles nous ont fourni un cadre général pour adresser le problème d'approximation de fonctions à valeurs dans un espace de Hilbert. Dans le cadre de cette théorie, nous avons établi et prouvé un nouveau théorème de représentation dédié à l'apprentissage semi-supervisé. Cela nous a permis de développer les solutions de la régression pénalisée pour deux fonctions de perte dans le cas supervisé et le cas semi-supervisé : les moindre carrés et la fonction de perte charnière. Dans le chapitre suivant, nous utilisons cette approche pour résoudre le problème de la prédiction de liens.

Chapitre 5

Prédiction de liens supervisée et semi-supervisée

Dans ce chapitre, nous utilisons la régression à noyaux d'entrée et de sortie (IOKR) pour résoudre le problème de la prédiction de liens. Dans la section 5.1, nous décrivons le principe de cette approche dans le cas de la prédiction de liens. Nous abordons en particulier le choix d'un noyau à valeur opérateur d'entrée et d'un noyau de sortie adaptés à ce problème. La section 5.2 est dédiée à la description des modèles IOKR-ridge et IOKR-margin obtenus dans le cas supervisé et semi-supervisé pour la Kernel Ridge Regression et la régression MMR. Enfin, nous introduisons dans la section 5.3 une approche par transfert de liens basée sur la régression à noyaux d'entré et de sortie pour la prédiction d'interactions protéine-protéine. Celle-ci permet d'exploiter l'information des réseaux d'interactions protéine-protéine connus chez différentes espèces, afin d'améliorer la prédiction du réseau d'interaction chez une espèce cible.

5.1 Régression à noyaux d'entrée et de sortie pour la prédiction de liens

Dans le cadre de la prédiction de liens, l'ensemble $\mathcal{V}$ correspond à l'ensemble $\mathcal{U}$ et désigne un ensemble de nœuds d'intérêt. L'objectif est d'estimer une relation existant entre ces nœuds, comme par exemple une relation d'ordre social ou une interaction physique dans le cas des protéines. La prédiction de liens dans le cadre supervisé consiste à apprendre un classifieur binaire $f : \mathcal{U} \times \mathcal{U} \to \{0, 1\}$ qui prend en entrée une paire d'objets et prédit si ces deux objets interagissent ou non à partir de descriptions en entrée sur les attributs de ces objets.

Nous allons utiliser le formalisme précédent à l'aide d'un régresseur à noyau à valeur opérateur en entrée et un noyau scalaire en sortie. Concernant le noyau de sortie, on ne dispose pas d'un noyau κ_y défini sur $\mathcal{U} \times \mathcal{U}$, mais d'une matrice de Gram K_{Y_ℓ} définie sur $\mathcal{U}_\ell$. En terme de noyau d'entrée, on va supposer que l'on dispose de noyaux qui vont coder pour différentes propriétés des nœuds du réseau.

5.1.1 Noyau de sortie

Durant la phase d'apprentissage, on dispose d'un graphe non orienté $\mathcal{G}_\ell = (\mathcal{U}_\ell, A_\ell)$ défini par le sous-ensemble $\mathcal{U}_\ell \subseteq \mathcal{U}$ et la matrice d'adjacence A_ℓ de taille $\ell \times \ell$. Pour cette tâche, nous définissons une matrice de Gram K_{Y_ℓ} de telle sorte que

celle-ci encode les proximités entre les nœuds du graphe. Dans le cas présent, nous avons choisi d'utiliser la matrice de Gram du noyau de diffusion [Kondor & Lafferty, 2002], qui est définie à partir de la matrice d'adjacence du graphe A_ℓ connue durant la phase d'apprentissage :

$$K_{Y_\ell} = \exp(-\beta L_{Y_\ell}),$$

où $L_{Y_\ell} = D_\ell - A_\ell$ est le laplacien du graphe, avec D_ℓ la matrice diagonale des degrés. Néanmoins, d'autres noyaux peuvent être également envisagés pour le choix de ce noyau de sortie.

La matrice de Gram de sortie K_{Y_ℓ} peut être associée à un noyau $\kappa_{y,\ell} : \mathcal{U}_\ell \times \mathcal{U}_\ell \to \mathbb{R}$, qui définit de manière unique un RKHS, noté $\mathcal{F}_{y,\ell}$:

$$\forall (u, u') \in \mathcal{U}_\ell \times \mathcal{U}_\ell,\ K_{Y_\ell}(u, u') = \kappa_{y,\ell}(u, u').$$

5.1.2 Noyau d'entrée à valeur opérateur

Afin d'utiliser la méthode IOKR introduite dans le chapitre précédent, nous avons besoin de définir un noyau à valeur opérateur adapté au contexte de la prédiction de liens. Pour cela, nous supposons que les données peuvent être décrites par un noyau scalaire en entrée $\kappa_x : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$. Etant donné κ_x, il existe un espace de Hilbert $\mathcal{F}_x$ et une fonction caractéristique $x : \mathcal{U} \to \mathcal{F}_x$ tels que :

$$\forall (u, u') \in \mathcal{U} \times \mathcal{U}, \kappa_x(u, u') = \langle x(u), x(u') \rangle_{\mathcal{F}_x}.$$

Mais contrairement au noyau de sortie, la fonction noyau en entrée κ_x est supposée complètement connue. On considère le noyau à valeur opérateur décomposable suivant :

$$\begin{aligned} \mathcal{K}_x : \quad & \mathcal{U} \times \mathcal{U} \to \ \mathcal{L}(\mathcal{F}_{y,\ell}) \\ & (u, u') \ \mapsto \ \kappa_x(u, u') \times I_{dim(\mathcal{F}_{y,\ell})} \end{aligned} \tag{5.1}$$

où $I_{dim(\mathcal{F}_{y,\ell})}$ est la matrice identité de taille $dim(\mathcal{F}_{y,\ell}) \times dim(\mathcal{F}_{y,\ell})$.

Le théorème (5) établit qu'un RKHS $\mathcal{H}$ peut être construit à partir de ce noyau.

Il est important de souligner que même si le noyau choisi peut sembler simple, il faut tenir compte du fait que dans cette tâche, l'espace de Hilbert de sortie correspond à l'espace caractéristique associé au noyau $\kappa_{y,\ell}$ et celui-ci est donc potentiellement de dimension infinie.

5.1.3 Régression à noyaux d'entrée et de sortie pour la prédiction de liens

Si l'on construit une approximation $\hat{\kappa_y}$ de la fonction noyau $\kappa_y : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ telle que

$$\forall (u, u') \in \mathcal{U}_\ell \times \mathcal{U}_\ell, \hat{\kappa_y}(u, u') = \kappa_{y,\ell}(u, u'),$$

on disposera alors d'informations concernant la proximité des objets de $\mathcal{U}$ entre eux, en tant que nœuds dans le graphe total.

$\kappa_{y,\ell}$ étant un noyau semi-défini positif, il existe une fonction caractéristique $y_\ell : \mathcal{U}_\ell \to \mathcal{F}_{y,\ell}$ telle que :

$$\forall (u,u') \in \mathcal{U}_\ell \times \mathcal{U}_\ell, \kappa_{y,\ell}(u,u') = \langle y_\ell(u), y_\ell(u') \rangle_{\mathcal{F}_{y,\ell}} .$$

On considère le problème de l'apprentissage d'une fonction $h : \mathcal{U} \to \mathcal{F}_{y,\ell}$, où h est élément de $\mathcal{H}$, qui minimise la fonctionnelle de coût suivant :

$$\mathcal{J}(h) = \sum_{i=1}^{\ell} \mathcal{L}(h(u_i), y_\ell(u_i)) + \lambda_1 \|h\|^2_{\mathcal{H}} + \lambda_2 \sum_{i,j=1}^{\ell+n} W_{ij} \|h(u_i) - h(u_j)\|^2_{\mathcal{F}_{y,\ell}}.$$

Ce problème peut être résolu à l'aide des outils développés au chapitres 4.

Une fois cette fonction apprise, on définit un classifieur binaire de la façon suivante :

$$\forall (u,u') \in \mathcal{U} \times \mathcal{U},\ f_\theta(u,u') = \mathrm{sgn}(\langle h(u), h(u') \rangle_{\mathcal{F}_{y,\ell}} - \theta) . \tag{5.2}$$

Si h minimise $\mathcal{J}$, alors $h(u)$ est une bonne approximation de $y_\ell(u)$ pour u élément de $\mathcal{U}_\ell$ et par conséquent le produit scalaire $\langle h(u), h(u') \rangle_{\mathcal{F}_{y,\ell}}$ approche $\kappa_{y,\ell}(u,u')$ pour (u,u') élément de $\mathcal{U}_\ell \times \mathcal{U}_\ell$. Dans le cas général, on va supposer que pour (u,u') élément de $\mathcal{U} \times \mathcal{U}$, ce produit scalaire approche un noyau κ_y défini sur $\mathcal{U} \times \mathcal{U}$ de telle sorte que ce noyau restreint à $\mathcal{U}_\ell \times \mathcal{U}_\ell$ est égal à $\kappa_{y,\ell}$.

Dans la section suivante, nous présentons les modèles obtenus pour ce noyau à valeur opérateur dans le cadre supervisé.

5.2 Prédiction de liens dans le cadre supervisé et semi-supervisé

Nous présentons dans cette section les modèles obtenus pour la Kernel Ridge Regression et la régression à marge maximale dans le cas du noyau à valeur opérateur décomposable défini dans l'équation (5.1). Ces modèles sont appelés *IOKR-ridge* et *IOKR-margin*.

5.2.1 Modèles dans le cas d'un noyau décomposable identité

IOKR-ridge Lorsque l'on utilise le noyau à valeur opérateur $\mathcal{K}_x(u,u') = \kappa_x(u,u') \times I_d$, la fonction $\hat{h}$ minimisant le problème d'optimisation de la Kernel Ridge Regression supervisée (4.1) s'exprime de la façon suivante :

$$\forall u \in \mathcal{U}, \hat{h}(u) = C X_\ell^T x(u) = Y_\ell (\lambda_1 I_\ell + K_{X_\ell})^{-1} X_\ell^T x(u). \tag{5.3}$$

On peut remarquer que l'on retrouve le modèle linéaire proposé par Cortes et al. [2005] dans le cadre de la reformulation de *Kernel Dependency Estimation* (KDE). La méthode IOKR coïncide ainsi avec cette approche dans le cas supervisé et pour un noyau à valeur opérateur particulier ($k(u,u') \times I_{dim(\mathcal{F}_{y,\ell})}$). Par conséquent, la méthode que nous avons développée généralise le modèle proposé par Cortes et al. [2005].

Dans le cas semi-supervisé (4.9), cette expression devient :

$$\forall u \in \mathcal{U}, \hat{h}(u) = Y_\ell J \left(\lambda_1 I_{\ell+n} + K_{X_{\ell+n}}(J^T J + 2\lambda_2 L)\right)^{-1} X_{\ell+n}^T x(u). \tag{5.4}$$

IOKR-margin La solution du problème d'optimisation de la régression supervisée à marge maximale à noyau à valeur opérateur (4.4) est donnée par :

$$\hat{h}(u) = \frac{1}{2\lambda_1} Y_\ell \operatorname{diag}(\boldsymbol{\alpha}) X_\ell^T x(u), \tag{5.5}$$

où $\boldsymbol{\alpha}$ est obtenu en résolvant le problème suivant :

$$\begin{array}{ll} \min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^\ell} & \frac{1}{4\lambda_1}\boldsymbol{\alpha}^T(K_{Y_\ell} \circ K_{X_\ell})\boldsymbol{\alpha} - \boldsymbol{\alpha}^T\mathbf{1} \\ \text{s.t.} & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell \end{array} \tag{5.6}$$

En apprenant h à l'aide d'un noyau à valeur opérateur pour la fonction de perte charnière, on retrouve dans le cadre supervisé le problème d'optimisation *Maximum Margin Regression* proposé par Szedmak et al. [2005] et appliqué dans un autre contexte, celui de la classification multiclasses.

Pour le problème d'optimisation dans le cas semi-supervisé (4.12), la solution s'écrit :

$$\hat{h}(u) = Y_\ell \operatorname{diag}(\boldsymbol{\alpha}) J (2\lambda_1 I_{\ell+n} + 4\lambda_2 K_{X_{\ell+n}} L)^{-1} X_{\ell+n}^T x(u), \tag{5.7}$$

où $\boldsymbol{\alpha}$ est obtenu en résolvant le problème ci-dessous :

$$\begin{array}{ll} \min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^\ell} & \frac{1}{2}\boldsymbol{\alpha}^T(K_{Y_\ell} \circ J(2\lambda_1 I_{\ell+n} + 4\lambda_2 K_{X_{\ell+n}} L)^{-1} K_{X_{\ell+n}} J^T)\boldsymbol{\alpha} - \boldsymbol{\alpha}^T\mathbf{1} \\ \text{s.t.} & 0 \leq \alpha_i \leq 1, i = 1, \ldots, \ell \end{array} \tag{5.8}$$

5.2.2 Retour à la prédiction de liens

Le problème de la régression supervisée et semi-supervisée à noyau de sortie étant résolu, nous revenons maintenant au problème de la prédiction de liens en construisant le classifieur suivant, comme annoncé dans la section 5.1 :

$$\forall (u,u') \in \mathcal{U}, f_\theta(u,u') = \operatorname{sgn}(\widehat{\kappa_y}(u,u') - \theta). \tag{5.9}$$

En utilisant le noyau à valeur opérateur défini à l'équation (5.1), l'approximation

du noyau de sortie peut s'écrire de la façon suivante :

$$\widehat{\kappa_y}(u,u') = \langle \hat{h}(u), \hat{h}(u') \rangle_{\mathcal{F}_{y,\ell}} = x(u)^T B^T K_{Y_\ell} B x(u'),$$

où B est une matrice de taille $\ell \times dim(\mathcal{F}_x)$, qui dépend de la fonction de perte et du paradigme d'apprentissage (voir tableau 5.1).

$B =$	Apprentissage supervisé	Apprentissage semi-supervisé
IOKR-ridge	$(\lambda_1 I_\ell + K_{X_\ell})^{-1} X_\ell^T$	$J(\lambda_1 I_{\ell+n} + K_{X_{\ell+n}}(J^T J + 2\lambda_2 L))^{-1} X_{\ell+n}^T$
IOKR-margin	$\frac{1}{2\lambda_1}$ diag $(\alpha) X_\ell^T$	diag $(\alpha) J(2\lambda_1 I_{\ell+n} + 4\lambda_2 K_{X_{\ell+n}} L)^{-1} X_{\ell+n}^T$

TABLE 5.1 – Les différents modèles de la méthode IOKR dans le cas d'un noyau décomposable de la forme $\kappa_x(u,u') \times I_d$.

On constate qu'il n'est pas nécessaire de connaître explicitement les expressions des sorties $y(u)$ pour calculer ce produit scalaire.

Par ailleurs, cette formulation montre que l'expression du produit scalaire $y(u)^T y(u')$ est approchée par un produit scalaire modifié entre les entrées.

5.3 Transfert de liens pour la prédiction d'interactions protéine-protéine à l'aide de plusieurs espèces

Dans cette section, une nouvelle méthode de prédiction de liens, appelée *transfert de liens*, est définie. Cette méthode a été étudiée dans le cadre d'un séjour d'Adriana Birlutiu au laboratoire[1]. Nous avons participé à ce travail en le formulant dans le cadre des RKHS.

La méthode par transfert de liens consiste à utiliser les réseaux de PPI connus chez différentes espèces, appelées espèces de référence, afin de contraindre l'apprentissage d'une fonction de prédiction d'interactions protéine-protéine chez une espèce cible. Nous formulons cette méthode dans le cadre de la régression à noyaux d'entrée et de sortie. Afin d'incorporer l'information disponible chez les espèces de référence lors de l'inférence du réseau de PPI cible, la régression IOKR est utilisée à deux reprises : une première fois pour convertir les fonctions caractéristiques de sortie d'une espèce de référence à l'espèce cible et une seconde fois pour apprendre le réseau d'interaction cible. L'intérêt de la première étape consiste à augmenter l'ensemble d'apprentissage chez l'espèce cible en apprenant la relation existant entre l'espace de sortie de l'espèce de référence et l'espace de sortie de l'espèce cible.

Nous considérons pour le moment le cas d'une seule espèce de référence. Soit $\mathcal{U}$ l'ensemble des protéines chez l'espèce cible et $\mathcal{U}_\ell$ le sous-ensemble de protéines ap-

1. voir thèse d'Adriana Birlutiu [Birlutiu, 2011] ou article en préparation.

partenant à l'ensemble d'apprentissage. Durant la phase d'apprentissage, on dispose chez l'espèce cible de :

- la matrice de Gram $K_{Y_\ell}^{(c)}$ en sortie construite à partir la matrice d'adjacence partiellement connue $A_\ell^{(c)}$. On note $\kappa_y^{(c)}$ la fonction noyau associée avec ${\mathcal{F}_y}^{(c)}$ et $y^{(c)}(\cdot)$ l'espace et la fonction caractéristiques correspondants.
- la matrice de Gram $K_X^{(c)}$ en entrée, dont les coefficients sont supposés être définis à partir d'une fonction noyau : $\forall i, j = 1, \ldots, \ell, K_X^{(c)}(i,j) = \kappa_x^{(c)}(u_i, u_j)$.

Dans le cas de l'espèce de référence, on connaît la matrice d'adjacence $A^{(r)}$, à partir de laquelle on peut construire une matrice de Gram $K_{Y_\ell}^{(r)}$. On note $\kappa_y^{(r)}$ la fonction noyau associée avec ${\mathcal{F}_y}^{(r)}$ et $y^{(r)}(\cdot)$ l'espace et la fonction caractéristiques correspondants. Par contre, on ne dispose pas d'une matrice de Gram en entrée. Par ailleurs, on connaît les protéines de l'espèce cible qui ont des orthologues dans l'espèce de référence. On assimile l'ensemble des orthologues entre les deux espèces à un sous-ensemble de $\mathcal{U}$. Cet ensemble peut être divisé en deux sous-ensembles : $\mathcal{U}_1 \subseteq \mathcal{U}_\ell$ et $\mathcal{U}_2 \nsubseteq \mathcal{U}_\ell$.

On remarque que les matrices de Gram de sortie chez l'espèce de référence et l'espèce cible définissent deux espaces de Hilbert différents : ${\mathcal{F}_y}^{(r)}$ et ${\mathcal{F}_y}^{(c)}$. La première étape consiste par conséquent à apprendre une fonction $h_{r\to c} : {\mathcal{F}_y}^{(r)} \to {\mathcal{F}_y}^{(c)}$ allant de l'espace caractéristique de référence à l'espace caractéristique cible. Cette fonction de conversion est apprise sur $\mathcal{U}_1$, c'est à dire sur l'ensemble des orthologues, pour lesquels les liens sont connus à la fois dans l'espèce de référence et l'espèce cible :

$$\sum_{i \in \mathcal{U}_1} \|h_{r\to c}(y^{(r)}(u_i)) - y^{(c)}(u_i)\|^2_{\mathcal{F}_y} + \lambda_c \|h_{r\to c}\|^2_{\mathcal{H}}, \tag{5.10}$$

où $\lambda_c \geq 0$ est une constante de régularisation.
D'après l'équation (5.3), la fonction $h_{r\to c}$ optimale s'écrit sous la forme suivante :

$$h_{r\to c}(y^{(r)}(u)) = Y_1^{(c)}(\lambda_c I_{|\mathcal{U}_1|} + K_{Y_1}^{(r)})^{-1}(Y_1^{(r)})^T y^{(r)}(u), \tag{5.11}$$

où $Y_1^{(c)} = \{y^{(c)}(u_i), u_i \in \mathcal{U}_1\}$, $Y_1^{(r)} = \{y^{(r)}(u_i), u_i \in \mathcal{U}_1\}$ et où $I_{|\mathcal{U}_1|}$ est la matrice identité de dimension $|\mathcal{U}_1|$, c'est à dire le nombre de protéines appartenant à l'ensemble $\mathcal{U}_1$.

On utilise ensuite la fonction apprise afin d'améliorer la prédiction de liens chez l'espèce cible (voir figure 5.1). Pour cela, on cherche la fonction $h : \mathcal{U} \to {\mathcal{F}_y}^{(c)}$ appartenant à l'espace de Hilbert $\mathcal{H}$ de noyau reproduisant $\mathcal{K}_x$, qui minimise le critère suivant :

$$\mathcal{J}(h) = \sum_{i \in \mathcal{U}_\ell} \|h(u_i) - y^{(c)}(u_i)\|^2_{{\mathcal{F}_y}^{(c)}} + \lambda \|h\|^2_{\mathcal{H}} + \gamma \sum_{i \in \mathcal{U}_2} \|h(u_i) - h_{r\to c}(y^{(r)}(u_i))\|^2_{{\mathcal{F}_y}^{(c)}}, \tag{5.12}$$

où $\lambda, \gamma \geq 0$ sont deux constantes de régularisation. Le dernier terme transfère l'information de l'espèce de référence à l'espèce cible.
Nous montrons qu'il existe un théorème de représentation pour ce problème d'opti-

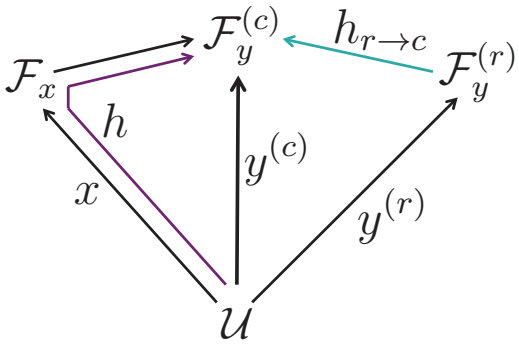

FIGURE 5.1 – **Schéma de la méthode par transfert de liens** : la régression IOKR est utilisée à deux reprises : une première fois pour apprendre la relation existant entre les espaces caractéristiques de sortie correspondant à l'espèce cible (c) et à l'espèce de référence (r), c'est à dire la fonction $h_{r\to c}$, et une seconde fois pour apprendre le réseau d'interaction cible à partir d'un noyau défini en entrée, c'est à dire la fonction h.

misation.

Théorème 13 *(Théorème de représentation) La solution minimisant le problème d'optimisation 5.12 admet l'expression suivante :*

$$\forall u \in \mathcal{U}, h(u) = \sum_{i \in \mathcal{U}_\ell \cup \mathcal{U}_2} \mathcal{K}_x(u, u_i)\mathbf{c}_i,$$

avec $\mathbf{c}_i \in {\mathcal{F}_y}^{(c)}$ *pour* $i \in \mathcal{U}_\ell \cup \mathcal{U}_2$.

Preuve Soit $\mathcal{H}_0$ l'espace défini par :

$$\mathcal{H}_0 = \{h \in \mathcal{H} | h(\cdot) = \sum_{i \in \mathcal{U}_\ell \cup \mathcal{U}_2} \mathcal{K}_x(\cdot, u_j)\mathbf{c}_j\},$$

et $\mathcal{H}_0^\perp$ l'espace orthogonal à $\mathcal{H}_0$:

$$\mathcal{H}_0^\perp = \{g \in \mathcal{H} | \langle g(\cdot), h(\cdot)\rangle_\mathcal{H}, \forall h \in \mathcal{H}_0\}.$$

Toute fonction h de $\mathcal{H}$ admet une décomposition unique $h = h_0 + h_0^\perp$, où h_0 est la projection de h sur $\mathcal{H}_0$ et $h_0^\perp$ celle sur l'espace $\mathcal{H}_0^\perp$.

D'après la propriété reproduisante, l'évaluation de la fonction h en tout point ne dépend pas de $h_0^\perp$:

$$\begin{aligned}
\forall u \in \mathcal{U}, h(u) &= \langle h, \mathcal{K}_x(\cdot, u)\rangle_\mathcal{H} \\
&= \langle h_0, \mathcal{K}_x(\cdot, u)\rangle_\mathcal{H} + \langle h_0^\perp, \mathcal{K}_x(\cdot, u)\rangle_\mathcal{H} \\
&= \langle h_0, \mathcal{K}_x(\cdot, u)\rangle_\mathcal{H} \\
&= h_0(u).
\end{aligned}$$

De plus,

$$\|h\|^2_{\mathcal{H}} = \|h_0\|^2_{\mathcal{H}} + \|h_0^{\perp}\|^2_{\mathcal{H}}.$$

Par conséquent,

$$\mathcal{J}(h) = \mathcal{J}(h_0) + \lambda\|h_0^{\perp}\|^2_{\mathcal{H}}.$$

On en déduit que la solution du problème (5.12) appartient à $\mathcal{H}_0$ et peut donc s'écrire sous la forme : $\forall u \in \mathcal{U}, h(u) = \sum_{i\in\mathcal{U}_\ell\cup\mathcal{U}_2} \mathcal{K}_x(u,u_i)\mathbf{c}_i$. ■

Comme précédemment, on considère le noyau à valeur opérateur décomposable de la forme $\mathcal{K}_x(u,u') = \kappa_x(u,u') \times I_{dim(\mathcal{F}_y)}$. La fonction h peut alors s'écrire sous la forme :

$$\forall u \in \mathcal{U}, h(u) = CX^T x(u), \text{ où } X = (X_\ell, X_2)\,.$$

En remplaçant cette expressions dans (5.12), on obtient :

$$\mathcal{J}(C) = \|CX^TX_\ell - Y_\ell^{(c)}\|^2 + \lambda\|CX^T\|^2 + \gamma\|CX^TX_2 - Y_1^{(c)}(\lambda_c I_{|\mathcal{U}_1|} + K_{Y_1}^{(r)})^{-1}(Y_1^{(r)})^TY_2^{(r)}\|^2.$$

On définit les matrices :

$$Y_{\mathcal{D}} = \left(Y_\ell^{(c)}, \sqrt{\gamma}\; Y_1^{(c)}(\lambda_c I_{|\mathcal{U}_1|} + K_{Y_1}^{(r)})^{-1}(Y_1^{(r)})^TY_2^{(r)}\right)$$

et

$$X_{\mathcal{D}} = (X_\ell, \sqrt{\gamma}\; X_2)\,.$$

$\mathcal{J}$ peut alors se réécrire sous la forme suivante :

$$\mathcal{J}(C) = \|CX^TX_{\mathcal{D}} - Y_{\mathcal{D}}\|^2 + \lambda\|CX^T\|^2.$$

On en déduit alors que :

$$\forall u \in \mathcal{U}, h(u) = Y_{\mathcal{D}}(\lambda I_{|\mathcal{U}_\ell|+|\mathcal{U}_2|} + K_{X_{\mathcal{D}}})^{-1}X_{\mathcal{D}}^T\, x(u). \tag{5.13}$$

La méthode peut être également étendue pour inclure l'information provenant de plusieurs espèces de référence en ajoutant un terme supplémentaire pour chaque espèce dans le problème d'optimisation (5.12).

Conclusion :
Dans ce chapitre, nous avons considéré l'application de la méthode de régression à noyaux d'entrée et de sortie au problème de la prédiction de liens. Nous avons présenté les modèles obtenus dans le cadre supervisé et semi-supervisé. Nous avons en particulier montré que l'on retrouve pour un coût quadratique pénalisé, une solution analytique qui étend la reformulation de la méthode KDE proposée par Cortes et al. [2005].
Par ailleurs, nous avons introduit dans la dernière section une approche par transfert de liens basée sur l'approche IOKR. Cette approche permet d'exploiter l'information des réseaux d'interaction protéine-protéine chez différentes espèces afin d'améliorer la prédiction du réseau d'interaction chez une espèce cible.

Troisième partie

Résultats numériques

Dans la troisième partie, nous présentons les différents résultats numériques que nous avons obtenus pour plusieurs problèmes de prédiction de liens.

Dans le chapitre 6, nous comparons tout d'abord les modèles IOKR-ridge et IOKR-margin introduits dans le chapitre 5 et nous étudions pour chacun d'entre eux le comportement de l'approche semi-supervisée par rapport à l'approche supervisée. Pour cela, nous décrivons les résultats obtenus par les deux modèles sur un ensemble de jeux de données synthétiques, un réseau de co-publication et un réseau d'interaction protéine-protéine chez la levure utilisé comme problème test par plusieurs approches. Sur le dernier réseau, nous présentons une comparaison dans le cadre supervisé avec différentes méthodes d'inférence de réseaux, les résultats obtenus dans le cadre transductif par les modèles IOKR-ridge et IOKR-margin, ainsi que les résultats pour l'approche par transfert de liens.

Nous présentons ensuite dans le chapitre 7 des expériences réalisées sur un réseau d'interaction protéine-protéine de la levure actualisé. Ce réseau a été construit à partir d'une base de données et nous avons collecté différentes propriétés appropriées sur les protéines telles que la localisation, la séquence ou les annotations de Gene Ontology.

Enfin, le chapitre 8 est dédié à l'application de la méthode que nous avons développée au réseau d'interaction protéine-protéine autour de la protéine CFTR, impliquée dans la mucoviscidose. Comme pour le réseau de la levure, nous avons construit divers types de descriptions à partir d'informations extraites de différentes bases de données.

Plusieurs difficultés apparaissant dans le cas de l'inférence de réseaux biologiques ont été soulevées précédemment, comme l'absence d'exemples négatifs à proprement parler, et donc le fait que des exemples considérés comme négatifs peuvent correspondre à des liens n'ayant pas été détectés expérimentalement jusqu'à présent. Une autre difficulté concerne le fait que l'approche par complétion de réseau nécessite de connaître complètement un sous-réseau. Nous abordons ces deux difficultés dans cette partie.

Chapitre 6

Application à la prédiction de liens

Dans ce chapitre, nous étudions le comportement des modèles IOKR-ridge et IOKR-margin présentés dans le chapitre 5. Pour cela, nous avons réalisé des expériences sur un ensemble de jeux de données synthétiques et deux réseaux réels fréquemment utilisés comme problème test : un réseau de co-publication et un réseau d'interaction protéine-protéine chez la levure. Dans les expériences, nous avons évalué les deux méthodes semi-supervisées dans un cadre transductif : nous avons ainsi supposé que tous les nœuds étaient connus au début de la phase d'apprentissage et que seul un ensemble de noeuds, correspondant à un sous-graphe, était étiqueté. L'objectif est de compléter le reste du réseau. Pour les différentes applications, nous nous sommes placés dans des cas extrêmes en considérant un très faible nombre d'exemples étiquetés.

6.1 Protocole expérimental

Nous commençons par décrire les mesures d'évaluation et le protocole expérimental qui ont été utilisés pour les expériences numériques.

6.1.1 Mesures d'évaluation

La tâche de prédiction de liens s'apparente à une tâche de classification binaire qui peut être évaluée par le calcul de mesures ROC (Receiver Operating Characteristic) [Fawcett, 2005]. Une courbe ROC représente le comportement du taux de vrais positifs (également appelé *rappel*), $TPR = \frac{TP}{P}$, en fonction du taux de faux positifs, $FPR = \frac{FP}{N}$, où N et P représentent respectivement le nombre d'exemples négatifs et d'exemples positifs. Ce type de courbe exprime le prix à payer en termes d'exemples négatifs mal prédits pour retrouver correctement un certain nombre d'exemples positifs.

Les courbes Précision-Rappel (PR) sont également utilisées pour l'évaluation des performances. Ce type de courbe montre le comportement de la *précision*, $Precision = \frac{TP}{TP+FP}$, en fonction du *rappel*. Cette courbe exprime ainsi la confiance que l'on peut avoir dans les prédictions positives. Pour les résultats numériques présentés dans cette partie, nous avons évalué les courbes précision-rappel de façon similaire à ce qui a été proposé par Davis & Goadrich [2006].

Rappelons que le modèle fournit en sortie une estimation $\hat{\kappa_y}(u, u')$ et que cette quantité doit être seuillée pour prendre une décision (voir équation 5.2). Les deux types de courbe sont obtenus en faisant varier le seuil θ de 0 à 1. Nous avons utilisé comme mesures d'évaluation l'aire sous la courbe ROC et l'aire sous la courbe PR, notées respectivement AUC-ROC et AUC-PR.

6.1.2 Protocole

Nous avons supposé que le graphe était connu. Pour différentes valeurs du nombre ℓ de nœuds étiquetés, nous avons échantillonné uniformément un sous-ensemble de nœuds pour former l'ensemble d'apprentissage. Nous avons utilisé les nœuds restants comme exemples non étiquetés et exemples de test. Les interactions étiquetées correspondent aux interactions entre deux nœuds de l'ensemble d'apprentissage et l'objectif est de compléter la matrice d'interactions. Ainsi, lorsque 10% des nœuds sont sélectionnés pour l'ensemble d'apprentissage, cela correspond à seulement 1% d'interactions étiquetées. Les performances ont été évaluées en calculant les valeurs moyennes des aires sous les courbe ROC et PR pour dix choix aléatoires de l'ensemble d'apprentissage.

La matrice de Gram $K_{X_{\ell+n}}$ du noyau d'entrée κ_x a été construite en utilisant un noyau gaussien. Le paramètre σ de ce noyau et l'hyperparamètre λ_1 ont été sélectionnés par une procédure de validation croisée leave-one-out sur l'ensemble d'apprentissage décrite dans le chapitre 4. Un noyau de diffusion de paramètre β a été choisi pour le noyau de sortie. Un autre noyau de diffusion de paramètre β_2 a été utilisé dans la contrainte de continuité au lieu du Laplacien du graphe. Les expériences ont montré que les valeurs de β et β_2 avaient une influence limitée sur les performances, nous avons donc fixé ces paramètres à 1. Enfin, nous avons fixé $W = K_{X_{\ell+n}}$ et défini L comme $\exp(-\beta L_W)$, où L_W est le laplacien associé à la matrice W.

6.2 Réseaux synthétiques

Avant d'employer nos méthodes sur des données réelles, nous avons voulu vérifier si lorsque la matrice de Gram en entrée contient presque toute l'information de sortie, les algorithmes et modèles proposés obtiennent de très bonnes performances. Nous avons en particulier voulu mesurer l'apport des données non étiquetées sur ces réseaux dans le cas où la proportion de nœuds étiquetés est faible.

6.2.1 Données

Nous avons généré les données synthétiques en échantillonnant des graphes aléatoires à partir d'une loi de Erdős-Renyi [Erdős & Rényi, 1959]. Dans les graphes aléatoires générés selon cette distribution de probabilité, chaque arête est présente avec la probabilité p et absente avec la probabilité $1 - p$, cela indépendamment du

statut des autres arêtes. La même étude aurait pu être réalisée à partir de mélanges d'Erdős-Renyi.

Les graphes échantillonnés contiennent 700 nœuds et leurs densités[1] ont été respectivement fixées à 0.007, 0.01 et 0.02, c'est à dire des valeurs qui sont représentatives des densités caractérisant les réseaux réels. Les vecteurs caractéristiques en entrée ont été obtenus en appliquant l'analyse en composantes principales à noyau sur le noyau de diffusion associé au graphe. Le paramètre de ce noyau de diffusion a été sélectionné de sorte à maximiser un critère d'information. Enfin, nous avons utilisé les premières composantes permettant de capturer 95% de la variance pour définir les vecteurs caractéristiques en entrée. La matrice de Gram du noyau d'entrée est construite à partir de ces vecteurs caractéristiques[2].

6.2.2 Résultats

Les figures 6.1 et 6.2 reportent respectivement les valeurs moyennes et les écarts-types des AUC-ROC et AUC-PR obtenues pour différentes densités de graphe en fonction de la valeur du paramètre de régularisation λ_2.

Comparaison dans le cas supervisé

Concernant le cas supervisé, c'est à dire dans le cas où λ_2 est égal à 0, on observe que la méthode IOKR-ridge obtient de meilleurs résultats que la méthode IOKR-margin. La différence entre les résultats obtenus par les deux méthodes s'accentue lorsque la densité du réseau augmente. Cette différence varie également selon le pourcentage de données étiquetées considéré. Elle est ainsi très significative pour 10% et 20% de données étiquetées. Cette observation peut s'expliquer par la différence de complexité existant entre les deux modèles dans le cas supervisé. En effet, les solutions obtenues dans le cas supervisé pour les deux modèles s'écrivent sous la forme $\hat{h}(u) = CX_\ell^T x(u)$. Pour le modèle IOKR-ridge,

$$C = Y_\ell(\lambda_1 I_\ell + K_{X_\ell})^{-1},$$

tandis que pour le modèle IOKR-margin, on a

$$C = \frac{1}{2\lambda_1} Y_\ell \operatorname{diag}(\alpha).$$

Apport du semi-supervisé pour IOKR-margin

Dans le cas du modèle IOKR-margin, on peut observer que l'approche semi-supervisée permet une amélioration importante des performances par rapport à l'approche supervisée en termes d'AUC-ROC et d'AUC-PR, et cela pour 5%, 10% ou 20% de données étiquetées. Cela peut s'expliquer par le fait que les modèles correspondant à cette approche dans le cas supervisé et semi-supervisé n'ont pas la

1. La densité d'un graphe correspond à la probabilité de présence d'arcs dans le graphe
2. Cette méthode a été discutée précédemment avec Pierre Geurts pour d'autres travaux.

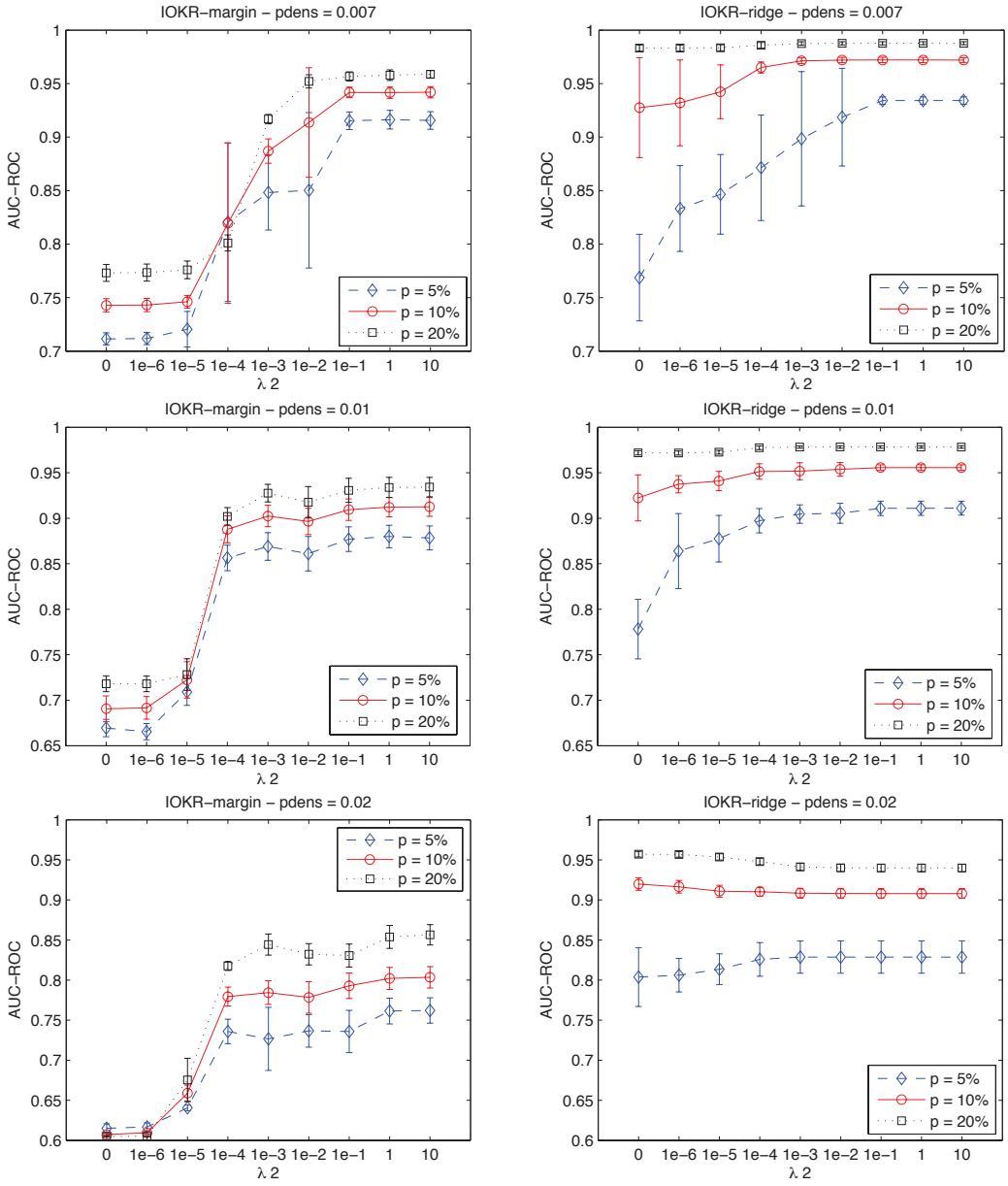

FIGURE 6.1 – Valeurs moyennes et écarts-types des AUC-ROC obtenues pour la reconstruction de trois réseaux synthétiques étant donné un pourcentage de nœuds étiquetés de 5%, 10% et 20% dans le cas du modèle IOKR-margin (colonne de gauche) et du modèle IOKR-ridge (colonne de droite). Les lignes correspondent à différentes densités de réseaux (notées pdens) : 0.007, 0.01 et 0.02.

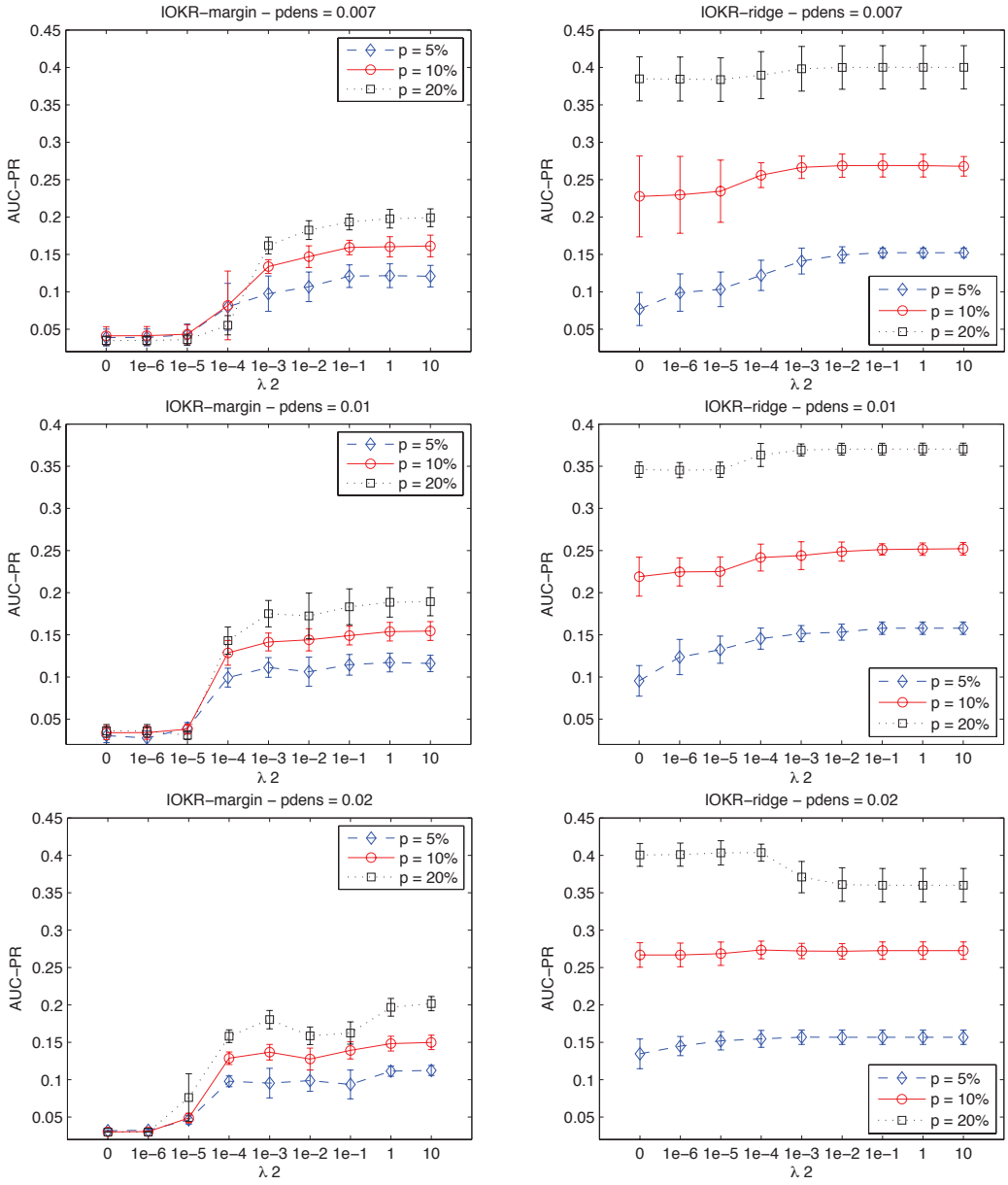

FIGURE 6.2 – Valeurs moyennes et écarts-types des AUC-PR obtenues pour la reconstruction de trois réseaux synthétiques étant donné un pourcentage de nœuds étiquetés de 5%, 10% et 20% dans le cas du modèle IOKR-margin (colonne de gauche) et du modèle IOKR-ridge (colonne de droite). Les lignes correspondent à différentes densités de réseaux (notées pdens) : 0.007, 0.01 et 0.02.

même complexité. En effet, la solution obtenue dans le cas supervisé s'écrit sous la forme $\hat{h}(u) = CX_\ell^T x(u)$ avec :

$$C = \frac{1}{2\lambda_1} Y_\ell \operatorname{diag}(\alpha).$$

Alors que dans le cas semi-supervisé, la solution s'écrit $\hat{h}(u) = CX_{\ell+n}^T\, x(u)$, où C est une matrice de paramètres beaucoup plus riche :

$$C = Y_\ell \operatorname{diag}(\alpha) J(2\lambda_1 I_{\ell+n} + 4\lambda_2 K_{X_{\ell+n}} L)^{-1}.$$

Apport du semi-supervisé pour IOKR-ridge

Pour le modèle IOKR-ridge, on observe également une amélioration des performances de l'approche semi-supervisée pour un faible pourcentage de données étiquetées (jusqu'à 10%), mais cette amélioration est moins importante que pour IOKR-margin. On peut donc faire la supposition que pour ce modèle relativement riche, l'utilisation de données non étiquetées améliore les AUC pour de faibles pourcentages de données étiquetées, mais lorsque les données étiquetées présentent suffisamment d'information, l'apprentissage semi-supervisé ne permet plus d'améliorer les performances.

Comparaison dans le cas semi-supervisé

En ce qui concerne la comparaison des résultats obtenus par les deux méthodes dans le cas semi-supervisé, on constate que la méthode IOKR-ridge obtient de meilleurs résultats que IOKR-margin. Cette amélioration est particulièrement significative pour les AUC-PR et est d'autant plus importante lorsque la densité du réseau est forte et dans le cas où le pourcentage de données étiquetées est élevé.

Différences observées selon la densité de réseau

Selon la densité de réseau considérée, les AUC obtenues par les deux méthodes peuvent varier de façon importante. A partir de ces résultats, on peut formuler l'hypothèse que la prédiction de lien supervisée est plus difficile dans le cas de réseaux plus denses et que la contribution des données non étiquetées semble être moins utile dans ce cas. Ainsi, dans le cas d'un réseau de densité égale à 0.007, avec 5% de données étiquetées, l'AUC-ROC est améliorée de 0.21 points pour l'approche IOKR-margin, et de 0.16 points environ pour l'approche IOKR-ridge, alors que ces améliorations sont respectivement de l'ordre de 0.15 et 0.02 pour le réseau de densité égale à 0.02.

6.3 Réseau de co-publication

Nous présentons maintenant l'application de notre méthode à un jeu de données contenant des informations sur les publications des conférences NIPS de 1988 à 2003

[Globerson et al., 2007]. Les nœuds du réseau représentent ici des chercheurs et un arc relie deux chercheurs entre eux s'ils sont co-auteurs d'au moins une publication. Parmi les 2865 chercheurs faisant partie des données d'origine, seuls ceux ayant au moins deux liens dans le réseau ont été pris en compte et nous avons ainsi considéré un réseau contenant 2026 chercheurs avec une densité de lien empirique de 0.02. Chaque chercheur est décrit par un vecteur de 14036 valeurs, qui correspondent aux fréquences avec lesquelles les mots du corpus considéré ont été utilisés dans ses publications. Les résultats moyens des AUC obtenus pour différentes valeurs du paramètre λ_2 sont représentés sur la figure 6.3. Comme précédemment, on peut observer que l'approche semi-supervisée permet améliorer les performances par rapport à l'approche supervisée pour les deux modèles. Pour les AUC-ROC, cette amélioration est d'autant plus significative que le pourcentage de données étiquetées utilisé est faible. Par ailleurs, le modèle IOKR-ridge surpasse le modèle IOKR-margin en termes d'AUC-ROC et d'AUC-PR et cela plus particulièrement lorsque le pourcentage de données étiquetées est important.

6.4 Réseau d'interaction protéine-protéine

Nous avons également réalisé des expériences sur un réseau d'interaction protéine-protéine chez la levure *S. Cerevisiae* construit par Kato et al. [2005] à partir des données de von Mering et al. [2002]. Ces données ont été obtenues en combinant des interactions détectées par des expériences de double-hybride [Uetz et al., 2000; Ito et al., 2000] et de purification de complexes protéiques [Gavin et al., 2002; Ho et al., 2002], ainsi que des données d'associations fonctionnelles. Ces dernières correspondent à des corrélations de profils d'expression d'ARNm, des données d'interaction génétique, ainsi que des interactions prédites *in silico* à partir de l'analyse du contexte génétique (fusion de gènes, voisinage des gènes, co-occurrences des gènes et profils phylogénétiques). Seules les interactions ayant été prédites par au moins deux méthodes différentes ont été considérées pour obtenir un réseau composé de 984 protéines reliées entre elles par 2438 interactions.

Les descripteurs suivants ont été utilisés en entrée pour l'inférence de ce réseau [Yamanishi et al., 2004; Kato et al., 2005; Geurts et al., 2006, 2007; Bleakley et al., 2007] :

- expressions de gènes [Eisen et al., 1998; Spellman et al., 1998],
- profils phylogénétiques,
- localisations cellulaires [Huh et al., 2003],
- interactions protéine-protéine dérivées d'expériences de double hybride de la levure [Uetz et al., 2000; Ito et al., 2000].

Ce réseau biologique a été utilisé comme problème test par plusieurs méthodes supervisées d'inférence de réseau. Cependant, nous soulignons le fait que des informations redondantes sont utilisées entre les entrées et les sorties ainsi que le fait que le réseau considéré n'est pas uniquement composé d'interactions physiques entre protéines ayant été observées expérimentalement. Nous nous sommes servis

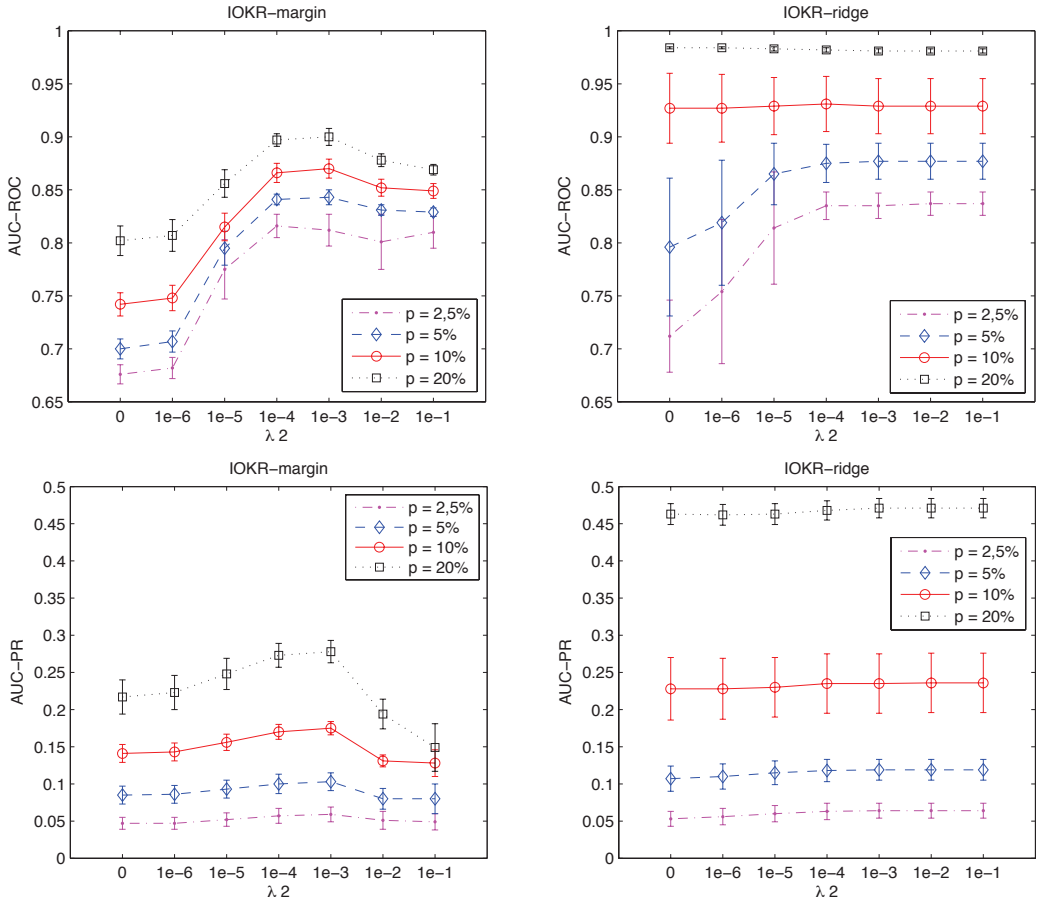

FIGURE 6.3 – Reconstruction du réseau de co-publication NIPS pour les auteurs ayant au moins deux liens dans le réseau. La figure reporte les moyennes et les écarts-types des AUC-ROC et des AUC-PR dans les cadres supervisé ($\lambda_2 = 0$) et transductif.

de ce jeu de données comme moyen de comparaison avec les autres méthodes mais nous consacrons le chapitre suivant à l'élaboration d'un jeu de données plus pertinent et à l'évaluation des performances de notre méthode sur ces données.

Nous présentons dans un premier temps une comparaison réalisée dans le cadre supervisé avec plusieurs méthodes d'inférence de réseau. Puis nous décrivons ensuite les résultats obtenus avec notre approche dans le cadre de l'apprentissage transductif.

6.4.1 Cadre supervisé

Nous avons complété la comparaison réalisée par Bleakley et al. [2007] avec la méthode de régression à noyaux d'entrée et de sortie (IOKR) et les arbres à sortie

noyau avec la méthode des extra-trees (OK3+ET) [Geurts et al., 2007]. Le protocole décrit dans Bleakley et al. [2007] a été utilisé : chaque méthode a été évaluée par une procédure de validation croisée en cinq parties (5-CV) et les hyperparamètres ont été sélectionnés sur l'ensemble d'apprentissage par une deuxième procédure de 5-CV. Les AUC-ROC et AUC-PR ont été mesurées uniquement pour les interactions possibles entre des protéines de l'ensemble de test et des protéines de l'ensemble d'apprentissage [3]. Les méthodes ont été évaluées pour chacun des descripteurs définis plus haut, ainsi que pour le noyau obtenu en sommant les noyaux correspondant à chacun des descripteurs :

$$\forall u, u' \in \mathcal{U}, \kappa_{int}(u, u') = \frac{1}{4}\left(\kappa_{exp}(u, u') + \kappa_{loc}(u, u') + \kappa_{phy}(u, u') + \kappa_{y2h}(u, u')\right).$$

La table 6.1 reporte les résultats obtenus par Bleakley et al. [2007], ainsi que ceux obtenus avec les méthodes OK3+ET et IOKR, dans le cas de la fonction de perte des moindres carrés. Les résultats correspondant à la méthode *Pkernel*, qui est basée sur un noyau entre paires [Ben-Hur & Noble, 2005], n'ont pas été reportés car celle-ci nécessite de définir une matrice de Gram composée de $O(n^4)$ éléments. Dans le cas présent, le nombre d'éléments est de l'ordre de 10^{12}, ce qui soulève des problèmes pratiques en termes de stockage de la mémoire et de temps de convergence.

Concernant les valeurs d'AUC-ROC, la méthode IOKR-ridge donne les meilleurs résultats pour les données d'expression et la combinaison de noyaux, tandis que les performances ne sont pas aussi bonnes pour les autres descripteurs. En terme d'AUC-PR, la méthode IOKR-ridge obtient des résultats du même ordre que les autres méthodes pour les données de localisation des protéines et les profils phylogénétique, et présente de plutôt bonnes performances pour les données d'expression de gène, les interactions protéine-protéine obtenues par double-hybride et la combinaison de noyaux.

6.4.2 Cadre transductif

Nous présentons maintenant les expériences réalisées dans le cadre transductif en utilisant le protocole expérimental décrit dans la section 6.1. La figure 6.4 reporte les valeurs moyennes et les écarts-types des AUC-ROC et AUC-PR obtenues pour l'inférence du réseau PPI de la levure, et cela pour différentes valeurs du paramètre λ_2 et pour différents pourcentages de nœuds étiquetés.

On peut constater que l'approche semi-supervisée améliore significativement l'AUC-ROC par rapport à l'approche supervisée dans le cas de IOKR-margin, et cela plus particulièrement lorsque le pourcentage de nœuds étiquetés est faible. En effet, avec 5% de nœuds étiquetés, l'amélioration peut être en moyenne de 0.8 points. Une amélioration est également constatée pour le modèle IOKR-ridge. Il est intéressant de noter que les problèmes d'inférence de réseaux d'interaction protéine-protéine sont caractérisés par un petit nombre de protéines étiquetées, et que l'on peut observer

3. Cela provient du fait que la méthode locale [Bleakley et al., 2007] ne peut être utilisée pour la prédiction d'interactions entre deux protéines de l'ensemble de test.

a) AUC-ROC :

Méthodes	exp	loc	phy	y2h	int
kCCA	81.4 ± 1.1	49.1 ± 14.6	67.8 ± 2.2	48.1 ± 2.4	87.8 ± 0.9
kML	82.9 ± 1.2	76.3 ± 1.1	71.7 ± 1.8	64.4 ± 1.9	88.1 ± 1.2
EM	80.6 ± 1.1	76.7 ± 3.8	71.0 ± 1.3	57.2 ± 2.7	89.3 ± 1.1
Local	78.1 ± 1.1	77.1 ± 2.9	$\mathbf{75.5 \pm 2.4}$	77.8 ± 1.2	87.6 ± 1.8
OK3+ET	82.1 ± 1.1	$\mathbf{81.0 \pm 1.3}$	75.1 ± 1.9	$\mathbf{80.6 \pm 1.6}$	89.2 ± 1.2
IOKR-ridge	$\mathbf{83.3 \pm 2.1}$	74.7 ± 3.6	69.6 ± 1.5	60.8 ± 3.5	$\mathbf{91.0 \pm 0.4}$

b) AUC-PR :

Méthodes	exp	loc	phy	y2h	int
kCCA	8.0 ± 1.9	2.1 ± 3.8	1.9 ± 0.4	2.1 ± 0.6	11.4 ± 1.5
kML	7.0 ± 2.4	1.2 ± 0.1	1.5 ± 0.2	0.8 ± 0.1	6.5 ± 2.2
EM	6.3 ± 1.2	5.5 ± 1.1	3.2 ± 0.5	10.4 ± 1.0	19.1 ± 1.3
Local	2.6 ± 0.4	3.7 ± 0.9	2.1 ± 0.3	7.6 ± 1.6	25.5 ± 3.4
OK3+ET	10.3 ± 3.7	$\mathbf{8.0 \pm 1.5}$	$\mathbf{3.7 \pm 0.9}$	5.7 ± 1.2	14.3 ± 3.8
IOKR-ridge	$\mathbf{13.7 \pm 4.4}$	7.0 ± 1.3	2.6 ± 0.4	$\mathbf{12.9 \pm 2.9}$	$\mathbf{27.2 \pm 6.5}$

TABLE 6.1 – **Valeurs des AUC-ROC et AUC-PR estimées par 5-CV pour la reconstruction du réseau PPI de la levure dans le cadre supervisé.** Les AUC ont été estimées à partir de données d'expression (exp), de données de localisation (loc), de profils phylogéniques (phy), de données d'interactions dérivées d'expériences de double hybride (y2h) ainsi que pour la combinaison des noyaux d'entrée (int). Les quatre premières lignes proviennent de Bleakley et al. [2007] : *kCCA* désigne la méthode par analyse des corrélations canoniques [Yamanishi et al., 2004] ; *kML* la méthode à noyau d'apprentissage de métrique [Vert & Yamanishi, 2005] ; *EM* la méthode de projection EM [Kato et al., 2005] ; *Local* les modèles locaux avec SVM [Bleakley et al., 2007]. Les résultats obtenus pour *OK3+ET* [Geurts et al., 2007] et *IOKR-ridge* sont également indiqués.

que l'utilisation de l'approche semi-supervisée permet d'améliorer les performances dans ce cas.

Par ailleurs, pour ce réseau, les AUC-ROC obtenues pour IOKR-ridge ne sont pas significativement différentes de celles obtenues avec IOKR-margin. Par contre, en terme d'AUC-PR, la méthode IOKR-ridge présente de meilleurs résultats que IOKR-margin lorsque le pourcentage de données étiquetées est plus important.

6.4.3 Transfert de liens

Afin de présenter un travail complet, nous présentons également dans cette section les résultats préliminaires obtenus par Adriana Birlutiu pour l'approche par

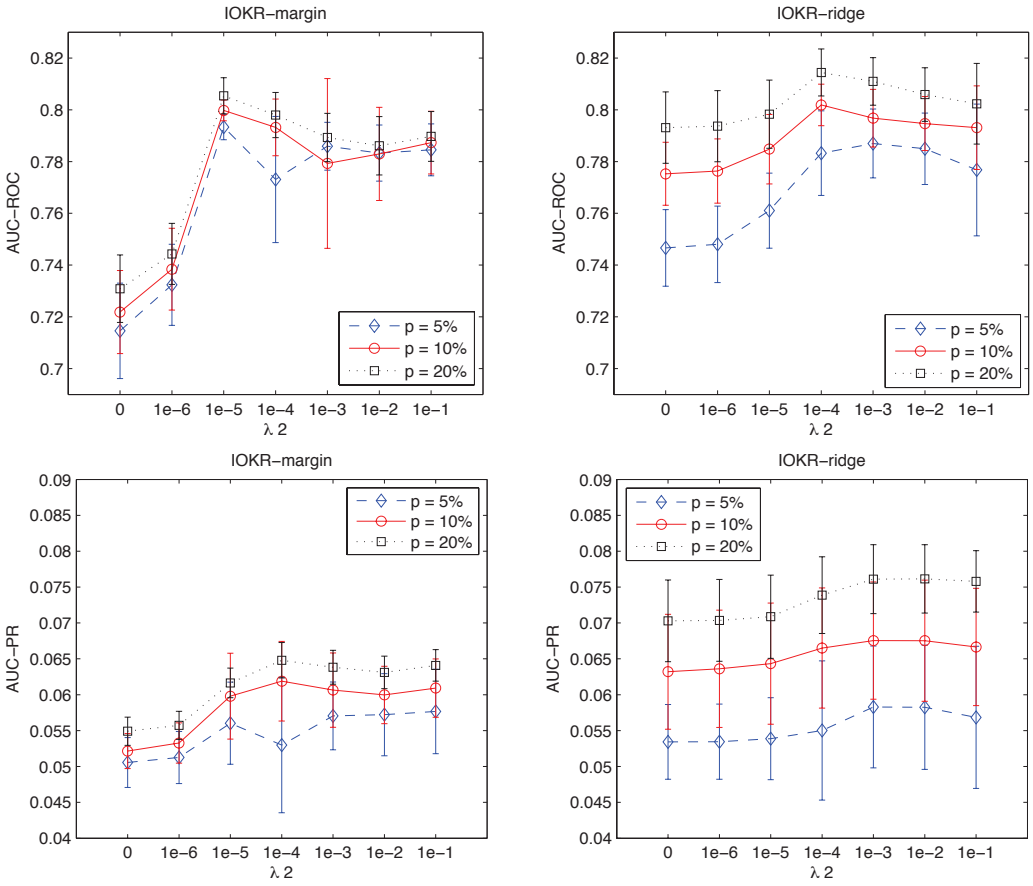

FIGURE 6.4 – Valeurs moyennes des AUC-ROC et AUC-PR obtenues pour la reconstruction du réseau PPI de la levure dans le cas des approches IOKR-ridge et IOKR-margin.

transfert de liens [Birlutiu, 2011].

Données

La levure *Saccharomyces cerevisiae* a été choisie comme organisme cible et les données d'interaction protéine-protéine considérées sont les mêmes que celles utilisées précédemment. Chaque protéine est décrite par des données d'expression de gène, de localisation et de profils phylogénétiques.

Trois espèces de référence ont été considérées : la levure à fission *Schizosaccharomyces pombe*, la souris *Mus Musculus* et la plante *Arabidopsis thaliana*. Les réseaux d'interaction protéine-protéine de ces espèces de référence ont été extraits d'après la base de données String [Szklarczyk et al., 2011]. Seules les interactions validées par des expériences en laboratoire ont été utilisées. L'ensemble des orthologues entre l'espèce cible et chacune des espèces de référence a été obtenu à partir de la base de

données Inparanoid [O'Brien et al., 2005] : la levure à fission a 271 orthologues avec l'espèce cible, la souris en a 147 et la plante 120. En plus des réseaux d'interaction protéine-protéine de ces trois espèces de référence, un réseau construit de façon artificielle a été également considéré. Les nœuds de ce réseau correspondent aux nœuds de la levure ayant un orthologue avec la levure à fission, et sont donc au nombre de 271. Les présences/absences de liens dans ce réseau artificiel correspondent, quant à elles, aux présences/absences de liens connues dans le réseau cible.

Protocole

Des expériences sur les données décrites ci-dessus ont été réalisées dans le cadre supervisé afin de déterminer si l'approche par transfert de liens permet d'améliorer les performances. Pour cela, les performances on été évaluées en fonction du paramètre γ. A l'exception de γ, les paramètres du modèle ont été fixés à la valeur optimale obtenue dans le cas où le terme de transfert n'est pas utilisé. Le paramètre du noyau de diffusion κ_y a ainsi été fixé à 3, le paramètre du noyau gaussien κ_x a été fixé à 4, le paramètre de régularisation λ à 0.9 et λ_c à 0.1. Les expériences ont été réalisées pour dix partitions aléatoires des données entre l'ensemble d'apprentissage et l'ensemble de test, ainsi que pour différents pourcentages de données étiquetées. Le modèle a été appris sur l'ensemble d'apprentissage pour différentes valeurs de γ et les performances ont été mesurées en utilisant l'aire sous la courbe ROC.

Résultats

La figure 6.5 représente les performances obtenues avec l'approche par transfert de liens en fonction du paramètre γ, et cela pour différentes proportions de données étiquetées : 10%, 15% et 20%. Pour chaque ensemble d'apprentissage, l'amélioration apportée par les différentes espèces de référence ainsi que pour l'espèce construite de façon artificielle, a été évaluée. On observe une amélioration de l'AUC-ROC pour les valeurs de γ supérieures à 0 par rapport au cas sans transfert, correspondant au cas où $\gamma = 0$. Cette amélioration est particulièrement significative pour les espèces de référence les plus proches de l'espèce cible, c'est à dire la levure à fission et l'espèce de référence artificielle.

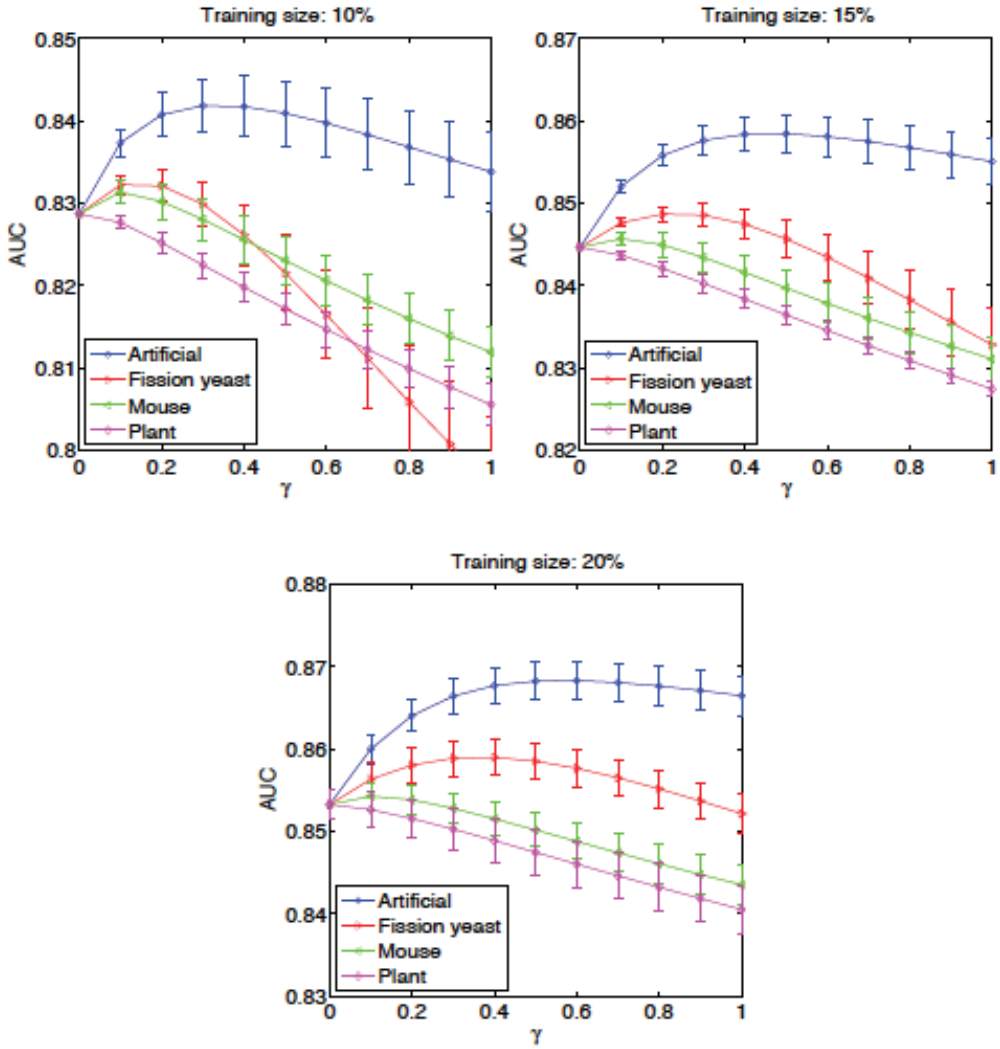

FIGURE 6.5 – **Résultats pour la méthode par transfert de liens sur le réseau PPI de la levure** [Birlutiu, 2011]. Les valeurs moyennes des AUC-ROC sont représentées en fonction de la valeur du paramètre γ pour dix ensembles d'apprentissage et de test sélectionnés aléatoirement. Les trois figures correspondent à trois valeurs de pourcentage de données étiquetées : 10%, 15% et 20%. Les différentes courbes correspondent aux espèces de référence : la levure à fission, la souris, la plante et l'espèce de référence artificielle.

Conclusion :
Dans ce chapitre, nous avons étudié le comportement des modèles IOKR-ridge et IOKR-margin sur des données artificielles et deux jeux de données réelles : la complétion d'un réseau de co-publication et celle d'un réseau d'interaction protéine-protéine. Les résultats numériques obtenus indiquent que l'utilisation des données non étiquetées permet d'améliorer les performances dans les cas extrêmes où il y a très peu de données étiquetées. On constate également que le modèle IOKR-ridge présente de meilleurs résultats en termes d'AUC-ROC et d'AUC-PR que IOKR-margin sur les différents réseaux. Dans le chapitre suivant, nous choisissons par conséquent d'utiliser IOKR-ridge comme méthode.

Chapitre 7

Application au réseau d'interaction protéine-protéine de la levure

Nous présentons dans ce chapitre l'application de notre méthode à un réseau d'interaction protéine-protéine de la levure *S. cerevisiae* actualisé. Pour cela, nous avons construit un réseau d'interaction protéine-protéine à partir d'une base de données. Puis nous avons collecté diverses caractéristiques sur les protéines utilisées dans la littérature, telles que la séquence, la localisation ou la fonction des protéines. Ce travail est motivé par les problèmes soulevés dans le chapitre précédent au sujet des données de la levure utilisées comme problème test.

7.1 Données

Nous énumérons dans un premier temps les descriptions des données que nous avons utilisées en entrée et nous présentons le réseau d'interaction protéine-protéine de la levure que nous avons construit.

7.1.1 Descriptions en entrée des données

Nous avons extrait un ensemble de douze descriptions à partir de différentes sources de données présentant un lien potentiel avec la présence d'interaction entre deux protéines. Le choix de ces descriptions a été effectué à partir d'une étude de la littérature existante dans ce domaine (voir section 1.5). Les descriptions utilisées sont résumées dans le table 7.1. Nous les présentons plus en détails dans la suite.

Expressions de gènes Nous avons utilisé les données d'expression de gènes de Eisen et al. [1998] et de Spellman et al. [1998]. Dans le premier cas, les données correspondent à la mesure de l'expression des gènes de la levure durant la division cellulaire, la sporulation, ainsi que durant des modifications des conditions expérimentales telles que l'oxygène et la température. Au total, 80 conditions expérimentales ont été considérées. Dans le deuxième cas, les données d'expression ont été mesurées durant les différentes phases du cycle cellulaire, et cela, pour 77 pas de temps. Les données d'expression sont parfois manquantes pour certains gènes. Dans le cas où un gène présente au maximum cinq valeurs manquantes, nous avons

Informations	Type de noyau
Expressions de gènes (Eisen)	gaussien
Expressions de gènes (Spellman)	gaussien
Localisations cellulaires	gaussien
Interactions génétiques	gaussien
Séquence primaire	linéaire (k-spectrum)
Interactions domaine-domaine	diffusion
Facteurs de transcription	gaussien
Processus biologiques (GO)	gaussien
Fonctions moléculaires (GO)	gaussien
Composants cellulaires (GO)	gaussien
Interologues	diffusion
Profils phylogénétiques	gaussien

TABLE 7.1 – Descriptions utilisées en entrée pour l'inférence du réseau PPI de la levure

remplacé ces valeurs par la valeur moyenne de l'expression dans les conditions expérimentales correspondantes. Nous avons considéré que les gènes présentant plus de cinq valeurs manquantes n'étaient pas annotés pour cette description.

Nous avons utilisé deux noyaux gaussiens sur les informations concernant l'expression des gènes. Pour rappel, un noyau gaussien $\kappa_x : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ a pour expression :

$$\forall u, u' \in \mathcal{U},\ \kappa_x(u, u') = \exp(-\gamma\ \|x(u) - x(u')\|^2_{\mathcal{F}_x}),$$

où $\mathcal{F}_x$ et $x : \mathcal{U} \to \mathcal{F}_x$ correspondent respectivement à l'espace caractéristique et la fonction caractéristique associés au noyau κ_x.

Localisations cellulaires Les localisations cellulaires des protéines de la levure ont été extraites à partir du catalogue MIPS correspondant [Mewes et al., 2004]. Les données contenues dans ce catalogue ont été obtenues à l'aide d'outils pour la fouille de texte, et ont été ensuite vérifiées manuellement. Après extraction de cette information, nous avons décrit les protéines par des vecteurs binaires indiquant la présence des protéines dans 54 compartiments cellulaires et nous avons défini un noyau gaussien à partir de ces vecteurs.

Interactions génétiques On dit que deux gènes interagissent génétiquement entre eux si le phénotype généré lorsque les deux gènes sont mutés diffère de celui attendu en combinant les phénotypes obtenus individuellement pour chaque gène muté. Une interaction génétique peut être qualifiée de négative ou de positive selon si le phénotype observé avec le double mutant est plus grave ou moins grave que

celui attendu. Dans le cas extrême, la combinaison des mutations sur les deux gènes entraîne la mort de la cellule, alors que si un seul des deux gènes est muté, la cellule est viable. On parle dans ce cas de mutation létale synthétique.

Nous avons extrait un ensemble d'interactions génétiques positives et négatives pour les gènes de la levure à partir de la base de données BioGRID [Stark et al., 2010]. Chaque gène a été représenté par un vecteur correspondant à un profil d'interactions avec les autres gènes. Une valeur de -1 ou 1 dans ce vecteur indique qu'il existe une interaction génétique négative ou positive avec le gène correspondant, et dans le cas contraire, une valeur nulle est indiquée. Nous avons ensuite utilisé un noyau gaussien pour mesurer les similarités entre ces profils d'interactions génétiques.

Séquence primaire Pour calculer les valeurs des similarités entre les séquences primaires des protéines, nous avons utilisé le noyau k-spectrum [Leslie et al., 2002], qui permet de comparer des séquences de différentes longueurs. Le principe de ce noyau consiste à construire une fonction caractéristique basée sur le k-spectrum d'une séquence, c'est à dire sur l'ensemble de toutes les sous-séquences de longueur k qu'elle contient. Etant donné $\mathcal{A}$ un alphabet de taille finie l (ici l'alphabet des acides aminés avec $l = 20$), la fonction caractéristique est définie de la façon suivante :

$$\begin{aligned} x^{(k)} : \mathcal{U} &\to \mathbb{R}^{l^k} \\ u &\mapsto (\phi_a(u))_{a \in \mathcal{A}^k}, \end{aligned}$$

où $\phi_a(u)$ est le nombre de fois où a apparaît dans la séquence primaire de u.
Le noyau k-spectrum entre deux séquences est ensuite défini en utilisant un noyau linéaire :

$$\kappa_x^{(k)}(u, u') = \langle x^{(k)}(u), x^{(k)}(u') \rangle_{\mathbb{R}^{l^k}}.$$

Nous avons fixé la valeur du paramètre k de ce noyau à 3, car c'est la valeur qui est généralement utilisée dans la littérature dans le cas des séquences protéiques.

Interactions domaine-domaine Nous avons utilisé les domaines définis dans la base de données Pfam [Punta et al., 2012], qui contient une grande collection de familles de domaines protéiques. Les domaines Pfam associés aux protéines d'intérêt ont été récupérés à l'aide du logiciel PfamScan, et les interactions domaine-domaine ont été obtenues à partir de la base de données Domine [Yellaboina et al., 2011]. Seules les interactions déduites des structures tridimensionnelles de la base PDB (Protein Data Bank) [Berman et al., 2000] ont été considérées.

A partir de ces données, nous avons construit un graphe en associant chacune des protéines à un nœud et en ajoutant une arête entre deux protéines si au moins deux domaines appartenant respectivement à ces protéines sont connus pour interagir ensemble. Nous avons ensuite défini un noyau de diffusion à partir de la matrice d'adjacence A du graphe obtenu. La matrice de Gram de ce noyau s'exprime sous la forme suivante :

$$K_X = \exp(-\beta L),$$

où exp désigne l'exponentielle matricielle, β le paramètre de diffusion et $L = D - A$ le Laplacien du graphe, avec D la matrice diagonale des degrés.

Fixation des facteurs de transcription Les données de fixation de facteurs de transcription utilisées ont été extraites de la base de données YEASTRACT [Teixeira et al., 2005]. Chaque protéine a été décrite par un vecteur binaire dont la dimension est égale au nombre de facteurs de transcription considérés, en l'occurrence 183. Une valeur de 1 indique que le facteur de transcription correspondant se fixe sur le gène qui code la protéine, et une valeur de 0 le contraire. Nous avons ensuite défini un noyau gaussien à partir de ces vecteurs caractéristiques.

Gene Ontology Nous avons également choisi d'utiliser un noyau gaussien pour chacune des trois ontologies de GO. Chaque protéine est codée par un vecteur caractéristique dont la dimension est égale au nombre total de termes existants dans l'ontologie considérée (3790 pour les processus biologiques, 1935 pour les fonctions moléculaires et 732 pour les composants cellulaires). Si une protéine est annotée par un terme, la valeur du vecteur à l'indice correspondant au terme sera égale à

$$-ln\left(\frac{\text{nombre de protéines annotées par ce terme}}{\text{nombre total de protéines}}\right).$$

Contrairement à l'utilisation d'une valeur binaire, cette expression présente l'intérêt de tenir compte de la spécificité du terme dans l'ontologie. Ainsi une annotation pour un terme relativement spécifique, et donc pour lequel peu de protéines sont annotées, sera associé à une valeur plus importante qu'un terme assez général apportant peu d'information.

Les annotations GO inférées automatiquement et non vérifiées par un expert, appelées IEA (*Inferred Electronic Annotation*), n'ont pas été prises en compte dans la construction de ces noyaux. Afin d'éviter d'introduire un biais dans l'évaluation, nous avons pas non plus tenu compte des annotations inférées à partir d'interactions protéine-protéine.

Interologues Nous avons par ailleurs exploité l'information des interologues. Pour cela, nous avons considéré plusieurs espèces source : *C. elegans*, *D. melanogaster*, *E. coli*, *A. thaliana*, *H. sapiens*, *S. pombe* et *D. rerio*. Les relations d'orthologie entre les gènes des espèces source et ceux de l'espèce cible (*S. cerevisiae*) ont été déterminées d'après la base de données Inparanoid [O'Brien et al., 2005]. Les interactions protéine-protéine des différentes espèces source ont été collectées à partir des bases de données DIP, MINT et BioGRID puis ont été agrégées afin d'obtenir un réseau d'interaction pour chaque espèce. Seules les interactions, pour lesquelles les deux protéines provenaient d'une même espèce, ont été considérées.

Nous avons ensuite construit un graphe entre les protéines considérées en ajoutant une arête entre deux protéines lorsque celles-ci avaient chacune un orthologue dans une des espèces considérées, et que ces deux orthologues étaient connus pour

interagir ensemble. Puis nous avons défini un noyau de diffusion sur le graphe obtenu et utilisé celui-ci comme noyau d'entrée.

Profils phylogénétiques Nous avons utilisé l'algorithme Phylopro [Xiong et al., 2011] pour générer les profils phylogénétiques. Celui-ci recherche les relations d'homologie présentes entre les protéines de la levure et celles des 120 organismes eucaryotes, pour lesquels le génome a été entièrement séquencé, en se basant sur la base de données Inparanoid. Nous avons construit un noyau gaussien à partir des profils phylogénétiques obtenus.

7.1.2 Réseau d'interaction

Nous avons construit un réseau d'interaction de la levure *S. cerevisiae* à partir de la base de données DIP (*Database of Interacting Proteins*) [Salwinski et al., 2004]. Cette base contient des interactions obtenues expérimentalement et ayant été vérifiées manuellement. Nous avons utilisé plus particulièrement les interactions présentes dans le sous-ensemble "core" [Deane et al., 2002] de la base DIP. Ce sous-ensemble est constitué des interactions

- déterminées par au moins une expérience à petite échelle,
- détectées par au moins deux expériences indépendantes,
- pour lesquelles il existe une interaction paralogue dans une autre espèce.

Ce sous-ensemble a été défini de sorte à améliorer considérablement la qualité des interactions, et en particulier pour éviter le nombre important de faux positifs obtenus avec les méthodes de détection expérimentale à grande échelle.

Nous avons considéré l'ensemble des protéines étant annotées pour chacun des noyaux d'entrée et étant impliquées dans au moins une interaction. Le réseau d'interaction protéine-protéine obtenu est constitué de 815 nœuds et présente une densité de liens de 0.0054. Celui-ci est représenté sur la figure 7.1.

7.2 Protocole

Le protocole, que nous avons utilisé, est similaire à celui décrit dans le chapitre précédent. Ainsi pour différents pourcentages de nœuds étiquetés, nous avons échantillonné uniformément un sous-ensemble de nœuds pour constituer l'ensemble d'apprentissage, et utilisé les nœuds restants comme exemples non-étiquetés. Nous avons ainsi évalué la méthode dans le cadre transductif. Les performances ont été mesurées par les valeurs moyennes de l'AUC-ROC et de l'AUC-PR obtenues sur dix ensembles d'apprentissage sélectionnés aléatoirement.

Pour chacun des noyaux d'entrée, nous avons sélectionné le paramètre du noyau de sorte à maximiser un critère d'information. Un noyau de diffusion de paramètre $\beta = 1$ a été choisi pour le noyau de sortie.

Etant donnés les résultats numériques présentés dans le chapitre précédent au sujet de la comparaison des modèles IOKR-ridge et IOKR-margin, nous avons choisi d'utiliser ici le modèle IOKR-ridge.

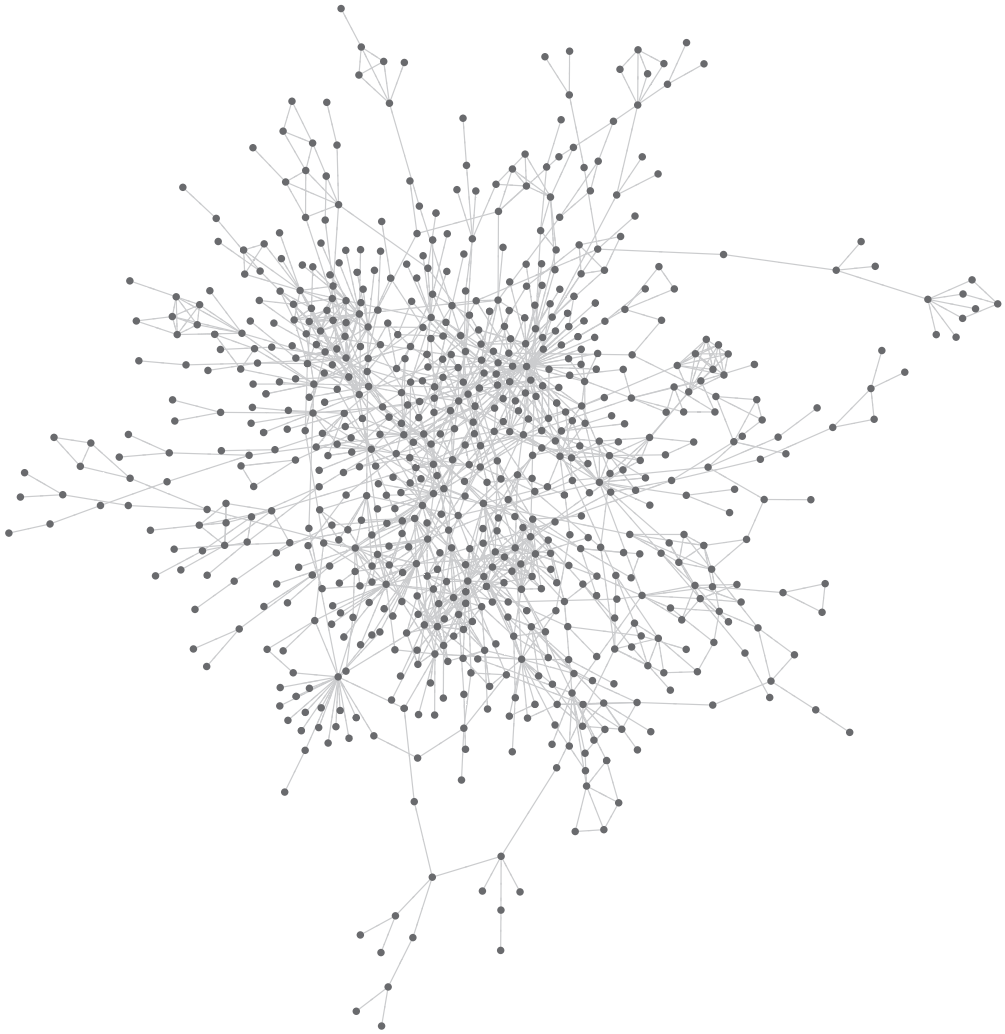

FIGURE 7.1 – Réseau d'interaction protéine-protéine de la levure *S. cerevisiae* obtenu à partir du sous-ensemble "core" de la base de données DIP [Salwinski et al., 2004]. Ce réseau a été visualisé à l'aide du logiciel Cytoscape [Shannon et al., 2003].

7.3 Résultats

Nous présentons dans cette section les résultats numériques que nous avons obtenus sur ce réseau d'interaction protéine-protéine. Ces résultats se décomposent en deux parties : la première d'entre elles porte sur l'apport lié à l'utilisation des données non étiquetées et la seconde concerne la comparaison de IOKR avec deux autres approches transductives.

7.3.1 Apport de l'apprentissage semi-supervisé

Dans un premier temps, nous exposons les résultats concernant l'apport de l'approche semi-supervisée par rapport à l'approche supervisée sur ces données. Les figures 7.2, 7.3 et 7.4 représentent respectivement les valeurs moyennes et les écarts-types des AUC-ROC et AUC-PR obtenues pour 5%, 10% et 20% de nœuds étiquetés. Sur chaque figure, les AUC obtenues sont représentées en fonction de la valeur de l'hyperparamètre λ_2, qui contrôle le poids associé à l'utilisation des exemples non étiquetés. L'hyperparamètre λ_1 a été fixé à 1 et nous avons utilisé pour la contrainte de continuité, un noyau de diffusion sur le graphe construit en utilisant les k-plus proches voisins avec $k = 50$. Les résultats obtenus pour chacun des noyaux d'entrée sont représentés. Nous avons également représenté le résultat correspondant à la combinaison des prédictions obtenues pour chaque noyau d'entrée, après normalisation de ces prédictions. Le noyau de sortie appris dans ce cas-là est le suivant :

$$\forall u, u' \in \mathcal{U},\ \hat{\kappa_y}(u, u') = \frac{1}{p} \sum_{j=1}^{p} \hat{\kappa_y}^{(j)}(u, u'),$$

où $\hat{\kappa_y}^{(j)}$ correspond à l'approximation du noyau de sortie obtenue lorsque le j-ème noyau est utilisé en entrée, et p au nombre de noyaux considérés.

On observe sur ces figures que les meilleures valeurs d'AUC-ROC et AUC-PR sont obtenues pour les noyaux GO concernant les processus biologiques, les fonctions moléculaires et les composants cellulaires, ainsi que pour le noyau sur les localisations cellulaires. La combinaison extérieure des différentes prédictions permet d'améliorer les résultats obtenus pour les noyaux individuels. L'amélioration observée pour la combinaison des prédictions est particulièrement significative pour l'AUC-PR. Cependant, les valeurs d'AUC-PR obtenues restent relativement faibles. Il faut ainsi atteindre un niveau de 20% de protéines étiquetées pour obtenir une valeur d'AUC-PR égale à 0.15.

Pour la plupart des noyaux utilisés en entrée, ainsi que pour la combinaison des approximations des noyaux de sortie, l'approche semi-supervisée présente de meilleures performances que l'approche supervisée, que ce soit en termes d'AUC-ROC ou d'AUC-PR. Comme pour les résultats présentés dans le chapitre précédent, on observe que cette amélioration est d'autant plus importante lorsque le pourcentage de données étiquetées est faible. Par ailleurs, cette amélioration peut varier selon le noyau d'entrée utilisé. Elle est ainsi relativement importante pour les noyaux d'entrée sur les informations de GO relatives aux processus biologiques et aux composants cellulaires.

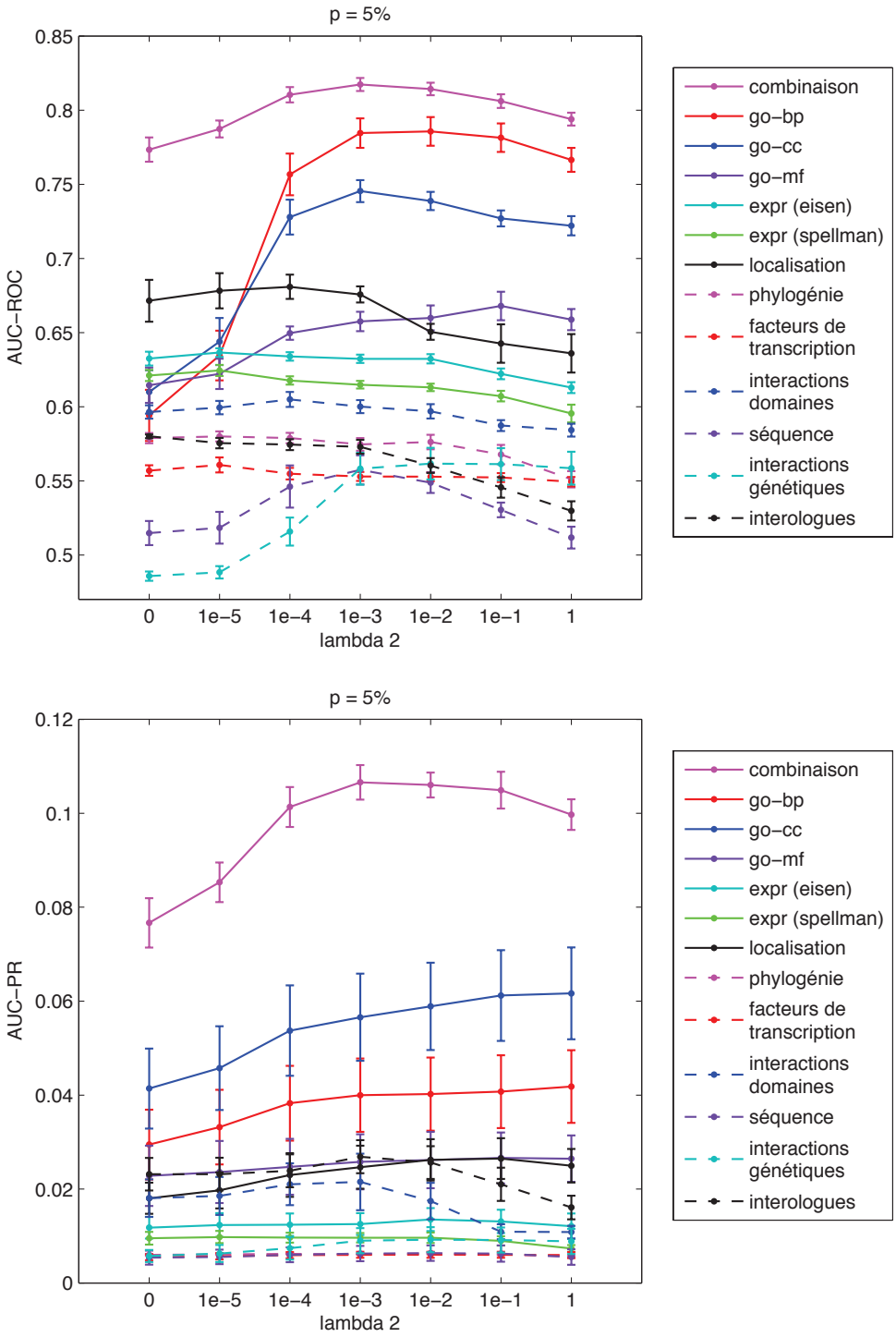

FIGURE 7.2 – **Reconstruction du réseau PPI de la levure à partir de différents descripteurs pour 5% de données étiquetées.** Les valeurs moyennes et les écarts-types des AUC-ROC et AUC-PR obtenues pour dix choix aléatoires de l'ensemble d'apprentissage sont représentés en fonction de la valeur prise par l'hyperparamètre λ_2. "go-bp", "go-cc" et "go-mf" font respectivement référence aux noyaux construits à partir des annotations GO sur les processus biologiques, les composants cellulaires et les fonctions moléculaires, tandis que "expr" désigne les noyaux basés sur l'information des données d'expression de gènes.

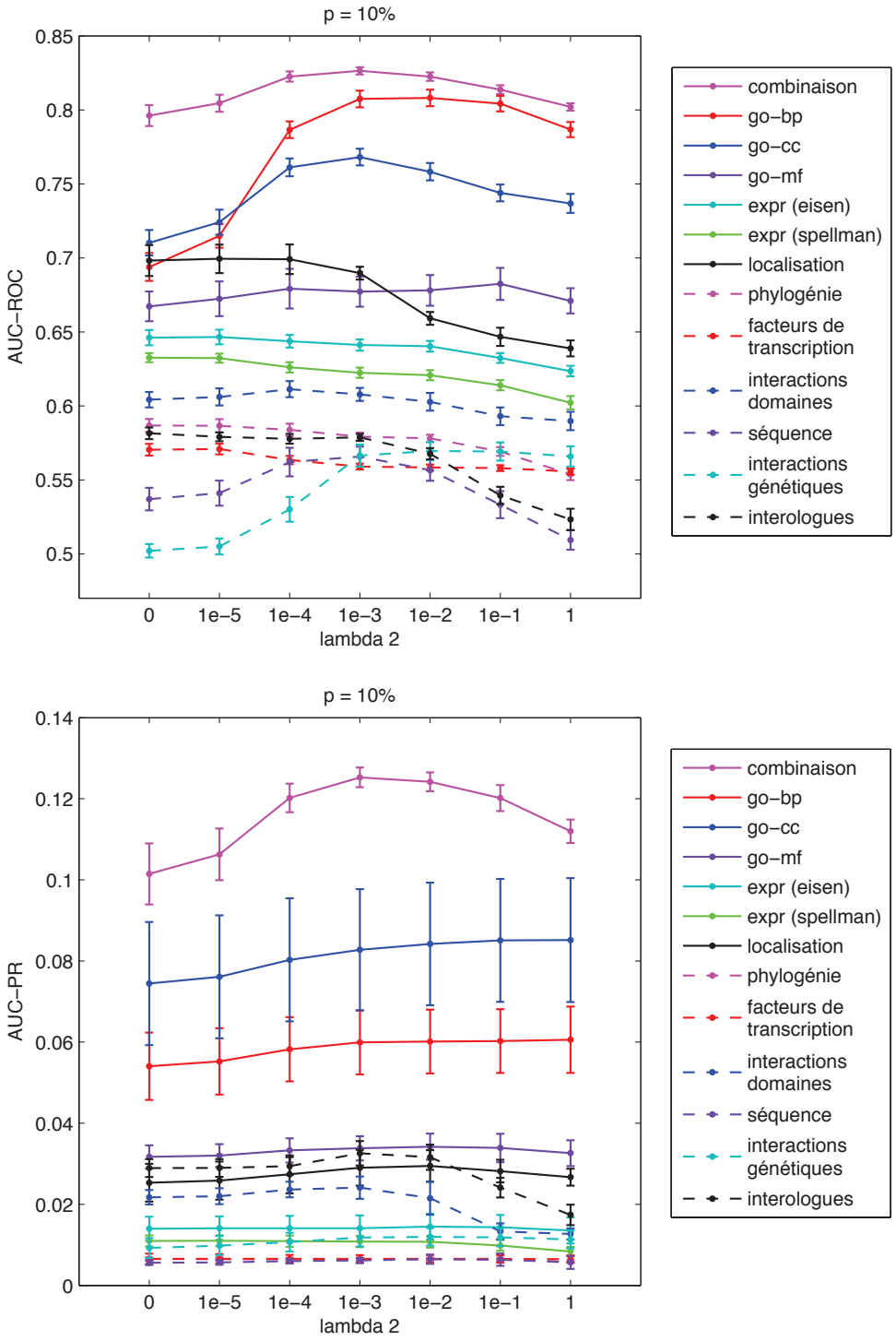

FIGURE 7.3 – **Reconstruction du réseau PPI de la levure à partir de différents descripteurs pour 10% de données étiquetées.**

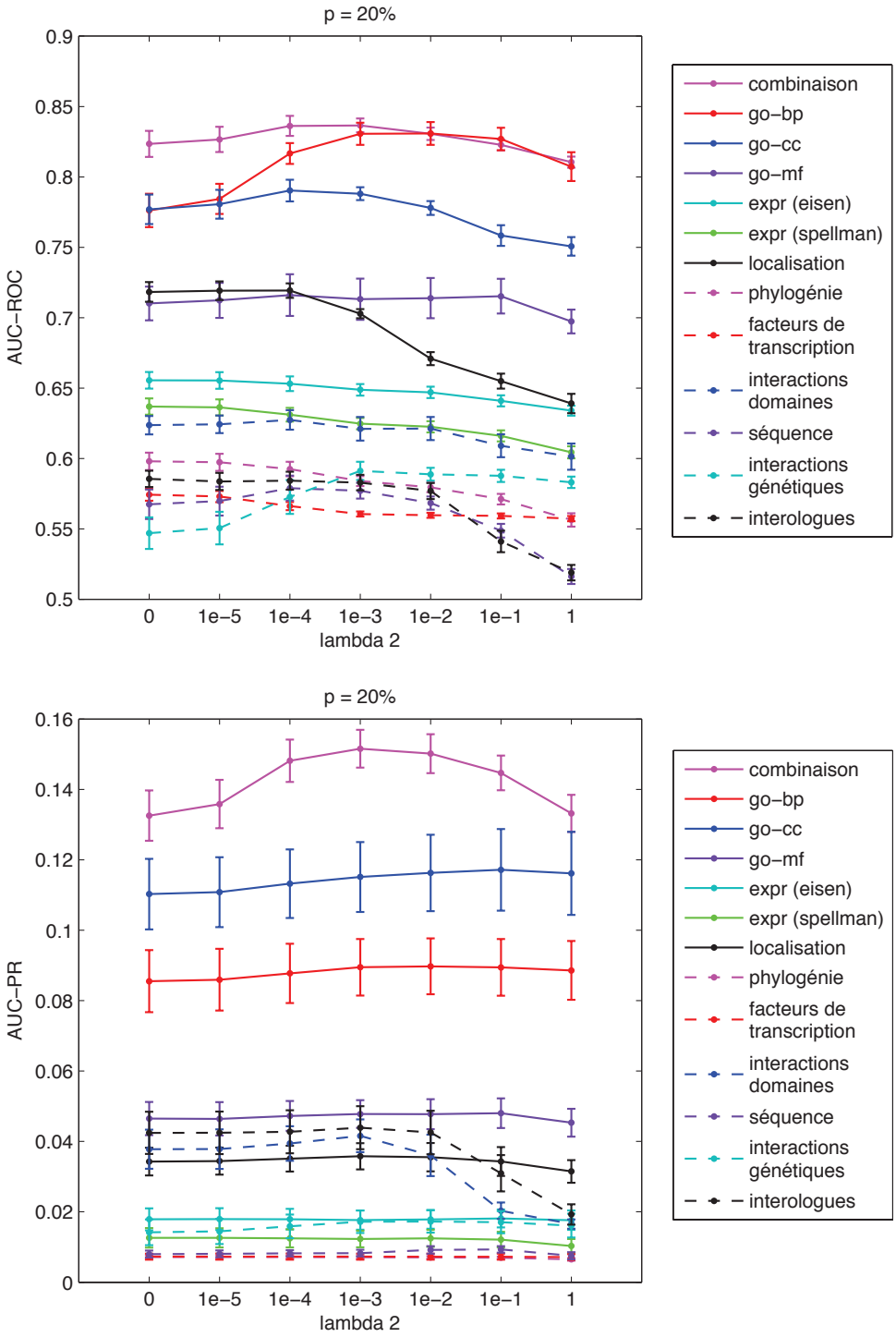

FIGURE 7.4 – **Reconstruction du réseau PPI de la levure à partir de différents descripteurs pour 20% de données étiquetées.**

7.3.2 Comparaison avec des approches transductives

Nous présentons maintenant la comparaison que nous avons effectuée entre la méthode IOKR-ridge et deux méthodes de prédiction de liens transductives [1] : EM [Kato et al., 2005] et *Penalized Kernel Matrix Regression* (PKMR) [Yamanishi & Vert, 2007]. Concernant les méthodes IOKR et PKMR, les paramètres de régularisation ont été sélectionnés par une procédure de validation croisée en trois parties sur l'ensemble d'apprentissage. Nous n'avons pas utilisé la procédure de validation croisée leave-one-out utilisée jusqu'ici afin de pouvoir comparer de manière identique les deux méthodes. Le paramètre β du noyau de diffusion correspondant au noyau de sortie a été fixé à 1. Les paramètres des noyaux d'entrée ont été selectionnés de sorte à maximiser un critère d'information.

Les résultats obtenus pour la comparaison des trois approches sont représentés sur la figure 7.5. Il s'agit des résultats correspondant à la combinaison des prédictions obtenues pour chacun des noyaux utilisées en entrée. On peut remarquer sur cette figure que la méthode IOKR-ridge obtient de meilleurs résultats en terme d'AUC-ROC que les méthodes EM et PKMR, et cela, quelle que soit la valeur du pourcentage de données étiquetées. Cependant, à partir de 10% de données étiquetées, cette amélioration n'est pas significative par rapport aux performances obtenues avec l'approche PKMR. En ce qui concerne les AUC-PR, la méthode EM présente de meilleures performances que les deux autre méthodes lorsque le pourcentage de données étiquetées est faible (5 et 10%). Par contre, à partir de 20% de nœuds étiquetés, la méthode IOKR-ridge améliore un peu les performances par rapport à celles obtenues pour les méthodes EM et PKMR.

Conclusion :
Nous avons présenté dans ce chapitre les résultats obtenus pour l'inférence d'un réseau d'interaction protéine-protéine de la levure à partir de diverses informations que nous avons extraites de plusieurs bases de données. Nous avons montré dans un premier temps que l'approche semi-supervisée permettait d'améliorer les performances par rapport à l'approche supervisée, et cela plus particulièrement pour une faible proportion de données étiquetées. Nous avons également montré que la combinaison des prédictions obtenues pour chacun des noyaux d'entrée présentait de meilleurs résultats. Nous avons par ailleurs comparé la méthode IOKR-ridge avec deux méthodes transductives d'inférence de réseau (EM et PKMR).

1. Une comparaison avec le cadre de la *propagation de liens* proposé par Kashima et al. [2009a,b] ne serait pas appropriée ici du fait que les interaction étiquetées peuvent être considérées de façon arbitraire, alors que le cadre de la méthode IOKR nécessite de connaître un sous-réseau.

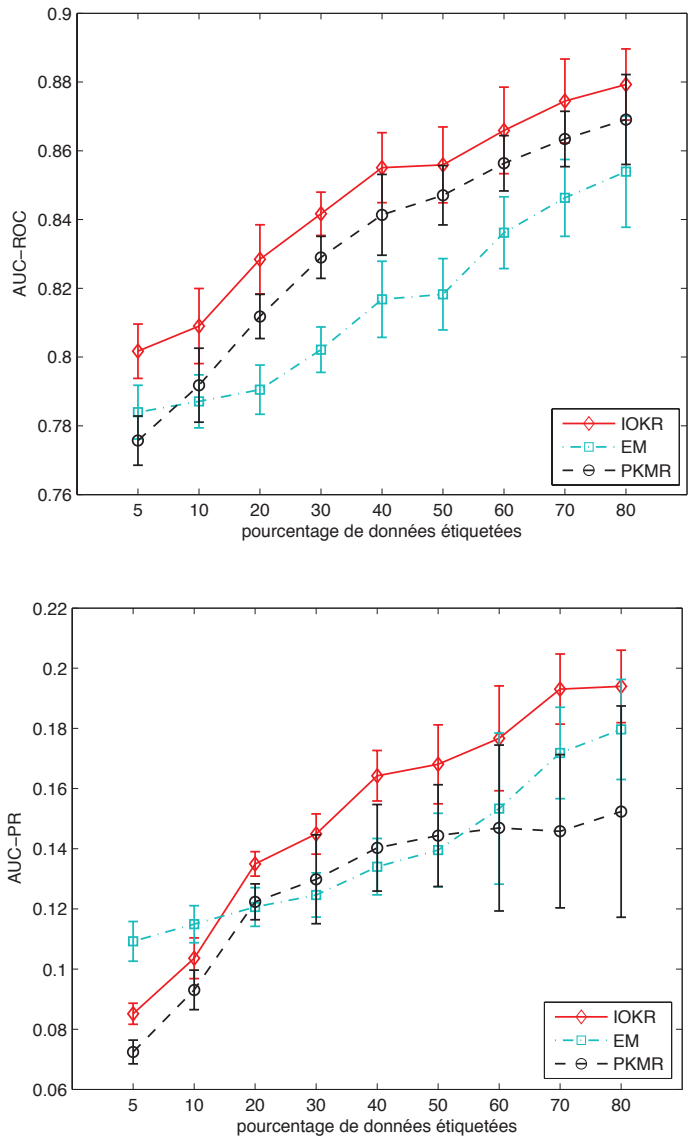

FIGURE 7.5 – **Comparaison des valeurs moyennes d'AUC-ROC et AUC-PR obtenues pour la reconstruction du réseau PPI de la levure**. Les AUC obtenues pour les méthodes IOKR-ridge, EM et PKMR sont représentées en fonction du pourcentage de données étiquetées. Elles ont été évaluées à partir de la combinaison des prédictions obtenues pour chaque descripteur, et cela sur dix ensembles d'apprentissage sélectionnés aléatoirement.

Chapitre 8

Inférence du réseau d'interaction autour de la protéine CFTR

Ce chapitre porte sur l'application de la régression à noyaux d'entrée et de sortie au problème d'inférence du réseau d'interaction protéine-protéine autour de la protéine CFTR, qui a été introduit au début du manuscrit. Pour cela, nous avons tout d'abord construit un réseau d'interaction protéine-protéine en considérant les interactions directes et les interactions du second ordre impliquant la protéine CFTR. Afin de compléter ce réseau d'interaction, nous avons collecté divers types de descriptions portant sur les protéines de ce réseau.

8.1 Données

Dans un premier temps, nous présentons les différents types d'information et les bases de données qui ont servi à définir les matrices noyaux utilisées en entrée. Nous décrivons également le réseau d'interaction protéine-protéine que nous avons construit.

8.1.1 Informations et codage

Comme pour le réseau de la levure décrit dans le chapitre précédent, nous avons collecté diverses descriptions pour résoudre ce problème d'inférence de réseau. A partir de ces informations, nous avons construit plusieurs noyaux d'entrée. A l'exception du noyau sur les expressions des protéines en fonction du tissu, les types de noyau d'entrée sont identiques à ceux utilisés dans le chapitre 7. La liste de ces noyaux est donnée dans la table 8.1.

Expressions de gènes Nous avons utilisé les données d'expression de gène mesurées par Su et al. [2004]. Ces données correspondent à l'expression de 44775 transcrits humains dans 79 tissus humains sains.

Expressions des protéines selon le tissu La présence d'une protéine dans un tissu est une propriété importante pour déterminer l'existence ou l'absence d'une interaction entre deux protéines. Nous avons donc extrait les distributions spatiales des protéines dans 46 tissus humains à partir de la base de données Human Protein Atlas [Uhlen et al., 2010]. Cette base répertorie des données obtenues par immunohistochimie, qui est une méthode permettant de localiser les protéines dans les

Informations	Type de noyau
Expressions des gènes	gaussien
Expressions des protéines selon les tissus	gaussien
Localisations cellulaires	gaussien
Séquence primaire	linéaire (k-spectrum)
Interactions domaine-domaine	diffusion
Processus biologiques (GO)	gaussien
Fonctions moléculaires (GO)	gaussien
Composants cellulaires (GO)	gaussien
Interologues	diffusion
Profil phylogénétique	gaussien

TABLE 8.1 – Informations extraites à partir de différentes bases de données et liste des noyaux d'entrée utilisés pour l'inférence du réseau d'interaction autour de la protéine CFTR

cellules d'un tissu grâce à des anticorps. Nous avons codé l'information obtenue sous forme de vecteurs binaires et nous avons utilisé ces vecteurs pour construire un noyau gaussien.

Localisations cellulaires Nous avons extrait de la base de données Human Protein Atlas [Uhlen et al., 2010] des données de localisation dans seize compartiments cellulaires différents. Cette information est codée par un vecteur binaire pour chacune des protéines.

Séquence primaire Comme dans le chapitre précédent, nous avons utilisé le noyau k-spectrum pour mesurer les valeurs de similarité entre les séquences d'acides aminés des protéines. Nous avons fixé la valeur du paramètre k de ce noyau à 3, car c'est la valeur qui est généralement utilisée dans la littérature.

Interactions domaine-domaine Nous avons extrait les domaines Pfam pour chacune des protéines et nous avons collecté les interactions domaine-domaine déduites des structures tridimensionnelles des protéines à partir de la base de données Domine. A partir de ces données, nous avons construit un graphe en associant chaque protéine d'intérêt à un nœud et en ajoutant un arc entre deux protéines si au moins deux domaines appartenant respectivement à ces protéines sont connus pour interagir ensemble. Nous avons utilisé ensuite un noyau de diffusion sur le graphe obtenu.

Gene Ontology Nous avons construit un noyau gaussien pour les trois ontologies de Gene Ontology. Pour cela, nous avons tout d'abord défini un vecteur caractéris-

tique pour chaque protéine de façon identique à celle décrite dans le chapitre 7. Ces vecteurs caractéristiques sont de dimension 2742 dans le cas des processus biologiques, de dimension 446 pour les fonctions moléculaires et 319 pour les composants cellulaires.

Interologues Nous avons utilisé l'information des interologues pour plusieurs espèces : la souris *M. Musculus*, le rat *R. norvegicus*, le poisson *D. rerio*, le ver *C. elegans*, la drosophile *D. melanogaster*, la plante *A. thaliana* ainsi que les levures *S. cerevisiae* et *S. pombe*. Pour chacune de ces espèces, nous avons construit un réseau d'interaction protéine-protéine à partir des bases de données BioGRID, DIP, MINT et Intact et extrait les relations d'orthologie existant entre les protéines de ces espèces et les protéines humaines. Comme dans le chapitre précédent, nous avons construit un graphe à partir de ces différentes informations

Profils phylogénétiques Les profils phylogénétiques des différentes protéines ont été construits en utilisant l'algorithme de comparaison de séquences BLASTP (protein-protein BLAST) [Altschul et al., 1990]. Ces profils indiquent la présence ou l'absence d'orthologues dans 1098 organismes.

8.1.2 Réseau d'interaction protéine-protéine

Un réseau d'interaction protéine-protéine a été construit à partir des informations contenues dans les bases de données BioGRID, DIP, MINT, Intact et NextProt [Lane et al., 2012]. Seules les interactions détectées par résonance plasmonique de surface, double hybride de la levure et pull down ont été sélectionnées. Chaque interaction a été vérifiée manuellement dans la littérature afin de s'assurer que les interactions utilisées sont bien des interactions directes entre deux protéines.

Le réseau construit contient un premier ensemble de protéines qui interagissent directement avec la protéine CFTR, ainsi que les protéines qui interagissent directement avec les protéines du premier ensemble. Le réseau obtenu contient 198 protéines reliées entre elles par 248 arcs. Celui-ci est représenté sur la figure 8.1. La liste des protéines constituant ce réseau est détaillée dans l'annexe B.

8.2 Protocole

Nous décrivons maintenant le protocole que nous avons utilisé pour les expériences numériques. Celui-ci adresse deux difficultés : la présence d'annotations manquantes pour certains noyaux et l'absence d'exemples négatifs connus.

8.2.1 Problème des annotations manquantes

Dans le cas du réseau de la levure présenté dans le chapitre précédent, nous avions considéré comme exemples les protéines qui étaient complètement annotées pour les différentes sources d'information. Cependant, cela pose un problème dans le

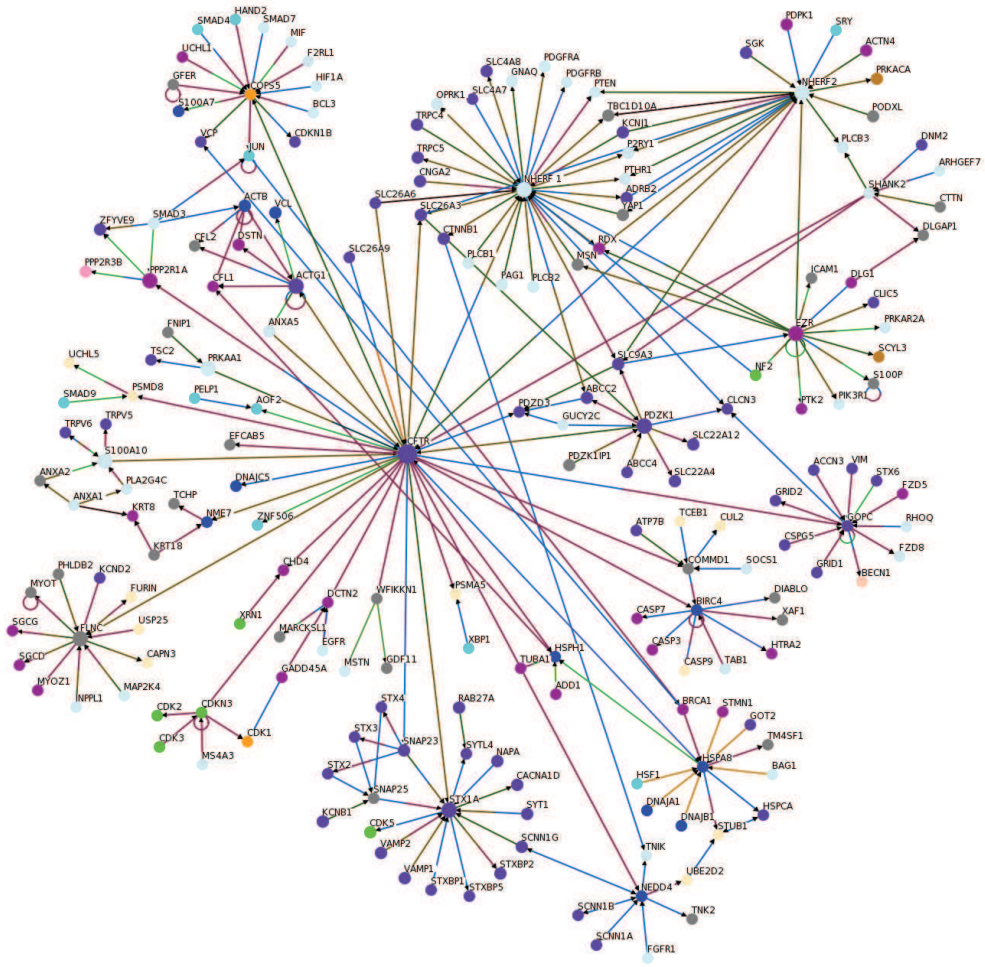

FIGURE 8.1 – **Réseau d'interaction protéine-protéine autour de la protéine CFTR**. Les différentes couleurs des arêtes représentent les différentes méthodes expérimentales utilisées pour détecter les interactions protéine-protéine, tandis que les différentes couleurs utilisées pour les nœuds désignent différentes annotations de Gene Ontology.

cas du réseau autour de CFTR car le nombre de protéines que l'on peut annoter pour toutes les sources d'information est relativement faible. Nous avons donc conservé l'ensemble des 198 protéines présentes dans le réseau d'interaction et nous avons établi un protocole adapté à ces données afin de pouvoir tester l'approche IOKR dans le cadre transductif.

Pour chacun des noyaux d'entrée, nous avons choisi d'apprendre la fonction de prédiction sur les protéines de l'ensemble d'apprentissage étant annotées pour ce noyau, et d'appliquer la fonction sur l'ensemble de test, auquel on retire les protéines n'étant pas annotées. On note A_j l'ensemble des protéines étant annotées pour le j-ième noyau. Si $\mathcal{S}$ désigne l'ensemble d'apprentissage et $\mathcal{R} \times \mathcal{T}$ représente l'ensemble des paires de protéines pour lesquelles on veut tester la présence d'interaction, alors pour le j-ième noyau, la fonction de prédiction sera apprise sur l'ensemble $\mathcal{S} \cap A_j$ et testée sur les paires appartenant à $(\mathcal{R} \cap A_j) \times (\mathcal{T} \cap A_j)$.

On s'intéresse également à la combinaison des noyaux de sortie appris pour chacun des noyaux en entrée. Pour cela, on effectue une moyenne pondérée de ces noyaux de sortie en tenant compte des valeurs manquantes pour certains noyaux d'entrée :

$$\forall u, u' \in \mathcal{U},\ \hat{\kappa}_y(u, u') = \frac{\sum_{j=1}^{p} \hat{\kappa}_y^{(j)}(u, u') * \delta_{u \in A_j} * \delta_{u' \in A_j}}{\sum_{j=1}^{p} \delta_{u \in A_j} * \delta_{u' \in A_j}},$$

où $\hat{\kappa_y}^{(j)}$ désigne le noyau de sortie appris en utilisant le j-ième noyau d'entrée, et où $\delta_{u \in A_j}$ vaut 1 si la protéine u est annotée pour ce noyau d'entrée, et vaut 0 sinon.

8.2.2 Prise en compte de l'absence d'exemples négatifs connus

Par ailleurs, nous avons vu précédemment que la méthode IOKR nécessite de connaître complètement un sous-réseau du réseau d'interaction. Dans le cas du réseau autour de la protéine CFTR, nous ne disposons pas d'exemples négatifs d'interaction mais d'exemples pour lesquels la présence d'un lien n'a pas été identifiée jusqu'à présent et notre objectif consiste à identifier des interactions potentielles parmi ces interactions. Nous décrivons dans la suite le protocole que nous avons établi afin de tenir compte de cette difficulté.

Soit S l'ensemble des protéines contenant la protéine CFTR et les protéines interagissant directement avec celle-ci. On note T l'ensemble des protéines qui interagissent avec les protéines de l'ensemble S. Nous avons choisi de prédire les interactions entre les protéines de l'ensemble S et celles appartenant à l'ensemble T. Nous avons fait ce choix-là car nous considérons que les interactions entre les protéines de S ont été mieux étudiées par les biologistes et sont donc plus fiables.

Pour cela, nous avons procédé en plusieurs itérations. A la i-ième itération, l'ensemble T est divisé uniformément en deux sous-ensembles, notés $T_{1,i}$ et $T_{2,i}$. Le classifieur est appris sur l'ensemble composé des protéines de S et de $T_{1,i}$, tandis que $T_{2,i}$ constitue l'ensemble de données non étiquetées. Le classifieur est ensuite

utilisé pour prédire les interactions entre les protéines de S et les protéines du sous-ensemble $T_{2,i}$ (voir figure 8.2). Une fois les différentes itérations effectuées, les prédictions d'interactions entre les protéines appartenant à S et une protéine de T sont obtenues en combinant les prédictions des itérations, pour lesquelles cette protéine faisait partie de l'ensemble de test. Ainsi,

$$\forall (u,u') \in S \times T,\ \hat{\kappa}_y(u,u') = \frac{\sum_{i=1}^{m} \hat{\kappa}_i(u,u') * \delta_{u' \in T_{2,i}}}{\sum_{i=1}^{m} \delta_{u' \in T_{2,i}}}, \tag{8.1}$$

où $\hat{\kappa}_i(u,u')$ désigne la valeur du noyau de sortie appris entre les protéines u et u' à la i-ième itération :

$$\forall (u,u') \in S \times T,\ \hat{\kappa}_i(u,u') = \frac{\sum_{j=1}^{p} \hat{\kappa}_i^{(j)}(u,u') * \delta_{u \in A_j} * \delta_{u' \in A_j}}{\sum_{j=1}^{p} \delta_{u \in A_j} * \delta_{u' \in A_j}}.$$

D'autres schémas de vote que celui proposé à l'équation 8.1 sont envisageables pour le calcul de $\hat{\kappa}_y(u,u')$. Dans notre cas, nous avons fait ce choix-là mais nous sommes conscients que le schéma de vote et la normalisation utilisée, pour laquelle le choix n'est pas du tout évident, mériteraient d'être étudiés de façon plus approfondie.

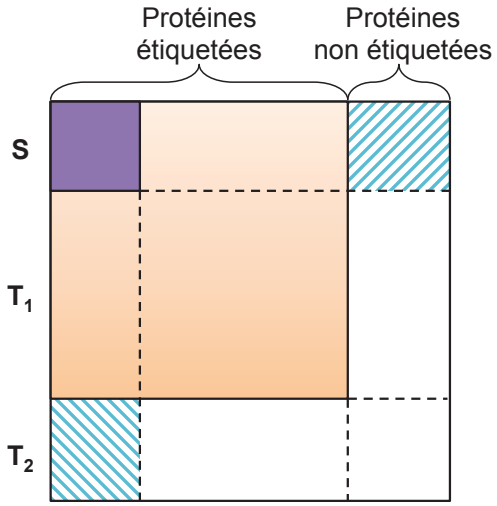

FIGURE 8.2 – **Protocole pour l'inférence du réseau d'interaction autour de la protéine CFTR.** A l'itération i, l'ensemble T est partitionné uniformément en deux sous-ensembles $T_{1,i}$ et $T_{2,i}$. L'union des ensembles S et $T_{1,i}$ est utilisée comme ensemble d'apprentissage et l'ensemble $T_{2,i}$ constitue l'ensemble des exemples non étiquetées. L'ensemble de test correspond à l'ensemble des interactions potentielles entre S et $T_{2,i}$ (représenté par des hâchures vertes sur la figure).

Ce processus est répété autant de fois qu'il y a d'exemples dans l'ensemble T. Dans le cas présent, l'ensemble S est composé de 35 protéines, et l'ensemble T de 163 protéines. Nous avons choisi de partitionner les protéines de l'ensemble T de

telle sorte que l'ensemble $T_{1,i}$ contienne 130 protéines et que $T_{2,i}$ en contienne 33.

Concernant les hyper-paramètres, les deux paramètres de régularisation λ_1 et λ_2 ont été sélectionnés selon la procédure de validation croisée leave-one-out décrite dans la section 4.5.

8.3 Résultats

Nous présentons dans cette section les résultats obtenus pour la prédiction d'interactions protéine-protéine dans le réseau décrit précédemment.

8.3.1 Prédiction d'interactions connues

Nous comparons tout d'abord les prédictions obtenues en se plaçant dans le cadre du protocole décrit ci-dessus avec l'ensemble des interactions du réseau déjà connues. Pour cela, nous avons ordonné toutes les prédictions d'interactions entre des paires de protéines $(u, u') \in S \times T$ en fonction de la valeur prise par $\hat{\kappa}_y(u, u')$. Puis nous avons mesuré le taux de vrais positifs obtenus lorsque l'on considère les n premières prédictions. La figure 8.3 représente l'évolution de ce taux lorsque l'on fait varier la valeur du nombre n.

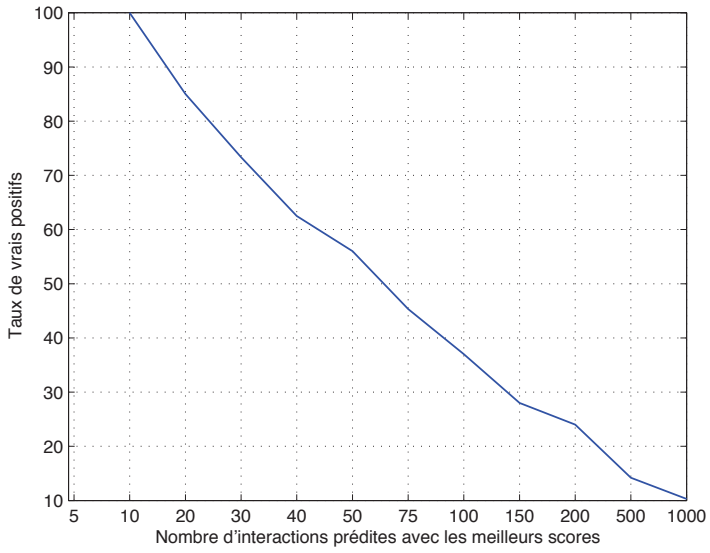

FIGURE 8.3 – Taux de vrais positifs parmi les n premières interactions prédites en fonction de la valeur de n.

On peut observer sur la figure que parmi les prédictions d'interaction correspondant aux valeurs les plus importantes du noyau de sortie appris, on retrouve une proportion importante de vrais positifs. Ainsi les dix premières interactions prédites correspondent toutes à des interactions existantes dans le réseau d'origine. Pour les

vingt premières, le taux de vrais positifs passe à 85% et il descend à 73% pour les trente premières interactions, c'est à dire des valeurs qui restent très satisfaisantes pour cette tâche de prédiction très déséquilibrée.

8.3.2 Prédiction de nouvelles interactions

Nous nous intéressons maintenant aux prédictions présentes parmi les n premières prédictions d'interaction et pour lesquelles il n'existe pas d'arête connue dans le réseau considéré. La liste des nouvelles interactions obtenues parmi les 100 premières interactions prédites pour l'inférence du réseau de PPI est donnée dans l'annexe C.

Nous avons réalisé une première étape de validation de ces prédictions en les confrontant avec la littérature existante sur les interactions protéine-protéine.

Prot 1	**Prot 2**	**Méthode**	**Référence**	**pubmed**
XIAP	PTEN	étude enzymatique	Van Themsche C (2009)	19473982
NEDD4	PTEN	pull down	Wang X (2008)	18498243
NEDD4	PTEN	étude enzymatique	Wang X (2008)	18498243
SNAP23	VAMP2	pull down	Kawanishi M (2000)	10713150
SNAP23	VAMP1	double hybride	Ravichandran V (1996)	8663154
SNAP23	STX6	pull down	Martin-Martin B (2000)	11001914
SNAP23	STXBP2	in vivo	Schraw TD (2003)	12773094
DNAJC5	STUB1	affinity capture western	Schmidt BZ (2009)	19098309
ANXA5	ANXA1	co-localisation	Arur S (2003)	12689596

Table 8.2 – Liste des interactions établie par une étude de la littérature pour les 100 premières prédictions obtenues.

Nous avons listé dans la table 8.2 les interactions que nous avons retrouvées dans la littérature et les différentes bases de données de PPI pour les nouvelles interactions présentes parmi les 100 premières prédictions d'interaction. Celle-ci contient neuf interactions, dont au moins six d'entre elles correspondent à des interactions directes entre deux protéines. Les résultats obtenus pour cette première esquisse de validation des interactions prédites sont donc très encourageants.

La prochaine étape de validation envisagée concerne la validation expérimentale par des biologistes de quelques unes des interactions parmi les différentes prédictions obtenues.

Conclusion :
Nous avons présenté dans ce chapitre les résultats obtenus pour la méthode IOKR-ridge sur un réseau d'interaction protéine-protéine autour de la protéine CFTR. Nous avons tout d'abord décrit les différentes descriptions que nous avons extraites de plusieurs bases de données, ainsi que le réseau que nous avons construit à partir d'une vérification manuelle de la littérature. Nous avons établi un protocole permettant de tenir compte de l'absence d'annotations de protéines pour certains noyaux ainsi que de l'absence d'exemples négatifs connus. Puis nous avons étudié les prédictions d'interaction obtenues en les comparant avec les interactions connues du réseau. Nous avons également esquissé une première étape de validation en recherchant dans la littérature des interactions détectées expérimentalement parmi les interactions prédites pour les meilleures valeurs de score.

Conclusion et perspectives

Concernant les contributions théoriques, nous avons introduit dans cette thèse un nouveau cadre pour la régression à sorties structurées, que nous avons appelé *régression à noyaux d'entrée et de sortie* (IOKR). Ce cadre fait intervenir des outils encore peu connus, qui sont les noyaux à valeur opérateur et les espaces de Hilbert à noyau reproduisant pour les fonctions à valeurs vectorielles dans un espace de Hilbert. L'approche que nous avons développée vient enrichir les premiers outils de régression à noyau de sortie qui ont été établis par Geurts et al. [2006, 2007].

Afin de résoudre le problème de la régression à sorties structurées, l'approche IOKR nécessite de définir un noyau à valeur opérateur en entrée et un noyau scalaire en sortie. Dans ce travail, nous avons utilisé en entrée des noyaux à valeur opérateur décomposables, c'est à dire des noyaux basés sur le produit de Kronecker entre un noyau scalaire d'entrée et une matrice semi-définie positive A. En utilisant la fonction de perte des moindres carrés et la fonction de perte charnière, nous avons proposé deux modèles, appelés IOKR-ridge et IOKR-margin. Nous avons montré que le modèle IOKR-ridge généralise le modèle linéaire proposé par Cortes et al. [2005], tandis que le modèle IOKR-margin présente une généralisation de celui proposé par Szedmak et al. [2005] pour la régression à marge maximale.

L'approche de régression à noyaux d'entrée et de sortie permet d'aborder le problème de la régression à sorties structurées dans le cadre de la régularisation. Ce cadre nous a permis d'étendre de manière rigoureuse l'approche IOKR au cas de l'apprentissage semi-supervisé, et donc de tirer partie des données non étiquetées. Pour cela, nous avons considéré un terme de régularisation inspiré de celui utilisé pour les fonctions à valeurs scalaires, et qui permet d'imposer la régularité de la fonction de prédiction sur un graphe décrivant les similarités entre entrées [Belkin & Niyogi, 2004; Zhou et al., 2004].

Une autre contribution de cette thèse concerne la problématique de la prédiction d'interactions protéine-protéine. L'approche que nous avons proposée convertit le problème de classification binaire à partir de paires d'objets en un problème d'apprentissage d'une fonction d'une seule variable à valeurs vectorielles et permet de bénéficier des données non étiquetées.

Nous avons illustré cette approche dans le cadre transductif par des résultats numériques sur des données artificielles, des problèmes test, ainsi que sur des données pertinentes pour la biologie. Nous avons ainsi construit deux réseaux d'interaction protéine-protéine à partir des interactions présentes dans différentes bases de données. Le premier d'entre eux est un réseau d'interaction chez une espèce qui a été très étudiée, en l'occurrence il s'agit de la levure *S. cerevisiae*. Le deuxième réseau, quant à lui, comporte des protéines humaines et a été construit autour de la protéine CFTR, qui joue un rôle important dans la pathologie de la mucoviscidose. Ces expériences ont montré que l'utilisation des données non étiquetées permettait d'améliorer les performances, en particulier lorsque seul un faible pourcentage

de nœuds sont étiquetés. Dans le cas du réseau d'interaction protéine-protéine autour de la protéine CFTR, nous avons esquissé une première étape de validation en confrontant les prédictions obtenues avec la littérature existante.

Perspectives

Dans la continuité de ces travaux de thèse, nous envisageons différentes perspectives. Une perspective importante envisagée à court terme concerne la validation expérimentale par des biologistes de quelques-unes des interactions parmi les interactions prédites avec le plus de confiance dans le réseau autour de la protéine CFTR.

Nous exposons dans la suite un ensemble de perspectives. Certaines d'entre elles concernent la méthode d'apprentissage, tandis que d'autres portent sur les expériences et les résultats numériques.

Choix du noyau de sortie

Le noyau de sortie a été peu étudié dans cette thèse. Nous avons choisi le noyau de diffusion comme noyau de sortie, mais il est possible qu'un autre noyau permette d'obtenir de meilleurs résultats. Il serait donc intéressant d'étudier par la suite l'impact du choix du noyau de sortie sur les performances, et en particulier dans le cas où des exemples faux négatifs sont présents dans l'ensemble d'apprentissage.

Robustesse face aux faux négatifs

Une première étude a été réalisée afin d'étudier la robustesse de la méthode IOKR face aux faux négatifs. Pour cela, nous avons réalisé une expérience numérique consistant à remplacer une partie des interactions protéine-protéine présentes dans l'ensemble d'apprentissage par des absences d'interaction. Nous avons observé que même lorsqu'une partie importante des exemples positifs étaient considérés comme des exemples négatifs, les performances de la méthode ne diminuent que légèrement. Cette robustesse face aux faux négatifs mériterait donc d'être étudiée plus en détails.

Choix du noyau d'entrée

Nous envisageons également d'étendre l'approche IOKR de sorte à pouvoir considérer plusieurs noyaux à valeur opérateur en entrée au lieu d'un seul. L'algorithme d'apprentissage par noyaux multiples (*Multiple Kernel Learning*) [Lanckriet et al., 2004; Rakotomamonjy et al., 2008] considère un noyau d'entrée scalaire κ_x, qui est défini comme une combinaison linéaire de plusieurs noyaux :

$$\forall u, u' \in \mathcal{U},\ \kappa_x(u, u') = \sum_{j=1}^{p} \mu_j \kappa_j(u, u') \text{ avec } \mu_j \geq 0.$$

Récemment, Kadri et al. [2012] ont proposé une méthode d'apprentissage par noyaux multiples dans le cas des noyaux à valeur opérateur, et cela pour la Kernel Ridge Regression.

Nous avons déjà commencé à travailler sur cette problématique et les premiers résultats obtenus montre que la combinaison des noyaux d'entrée présente de moins bonne performances que la combinaison des régresseurs.

Une autre piste éventuelle concernant le choix du noyau d'entrée serait de définir des descriptions plus riches. Des connaissances ou hypothèses de haut niveau, formulées par exemple sous la forme de règles logiques, sont peut-être susceptibles d'apporter des informations supplémentaires par rapport aux différents types de description que nous avons utilisés dans cette thèse.

Méthodes d'ensemble

Dans cette thèse, nous avons présenté deux régresseurs de base (IOKR-ridge et IOKR-margin). On peut envisager de composer la méthode IOKR avec une "randomisation" des échantillons et des noyaux à l'image des approches de type Extra-Trees [Geurts et al., 2006, 2007] et des récents travaux de Haury et al. [2012] sur différents régresseurs linéaires.

On peut également envisager d'utiliser des méthodes d'ensemble telles que le boosting et le bagging sur les deux régresseurs de base.

Difficultés de la tâche biologique

Nous avons vu dans les chapitres 6 et 7 que les AUC-PR obtenues pour l'inférence de réseaux d'interaction protéine-protéine étaient relativement faibles. Ces valeurs peuvent peut-être s'expliquer par des limitations intrinsèques à la méthode ou éventuellement par le contexte biologique du problème de prédiction d'interaction. On peut se poser la question de savoir si les attributs en entrée sont appropriés pour cette tâche, et s'il ne faudrait pas exploiter d'autres sources d'information.

Prédiction de fonction

L'identification des fonctions des protéines est un problème majeur actuellement en biologie car elle permet une meilleure compréhension des mécanismes moléculaires mis en œuvre dans la cellule. Les fonctions biologiques restent encore inconnues pour une proportion importante des protéines. Même chez les organismes les plus étudiés comme la levure, près d'un quart des protéines ne sont pas encore caractérisées par une fonction [Sharan et al., 2007]. Dans le cas présent, la prédiction de fonctions serait intéressante pour les protéines proches de CFTR car elle permettrait de mieux caractériser les interactions présentes dans le réseau.

L'information des interactions protéine-protéine a été utilisée auparavant pour la prédiction de fonction. C'est également le cas des données d'expression de gène, d'interactions génétiques, de profils phylogénétiques, de séquence et de localisation [Noble & Ben-Hur, 2007].

Certaines annotations de fonction sont organisées selon une hiérarchie. C'est le cas par exemple de Gene Ontology, du catalogue FunCat de MIPS, ainsi que celui de la nomenclature EC (Enzyme Commission) [Bairoch, 2000], qui est une classification des enzymes, basée sur la réaction chimique qu'elles catalysent. La régression à noyaux d'entrée et de sortie peut donc être utilisée pour résoudre ce problème d'apprentissage structuré. Nous montrons d'ailleurs en annexe une application de cette méthode à la classification multiclasses, pour des classes organisées selon une hiérarchie.

Apprentissage multitâches et apprentissage par transfert

Une autre perspective concerne l'apprentissage conjoint de la fonction des protéines et des interactions protéine-protéine. Ce problème peut être abordé dans le cas de l'apprentissage multitâches. Il s'agit d'une approche d'apprentissage statistique qui permet d'apprendre en même temps plusieurs problèmes reliés entre eux, en utilisant une représentation des données d'entrée commune pour les différents problèmes. L'utilisation de ce type d'approche permet de tirer partie d'éléments communs aux différentes tâches. La prédiction de fonction et la prédiction d'interactions protéine-protéine étant deux problèmes très liés, et pour lesquels les mêmes types de descripteurs sont généralement utilisés en entrée, il paraît tout à fait pertinent d'aborder ces problèmes dans le cadre de l'apprentissage multitâches. De plus, le cadre des modèles à noyaux à valeur opérateur est bien adapté à ce type de problèmes d'apprentissage.

Par ailleurs, une autre perspective concerne l'utilisation de l'apprentissage multitâches pour s'appuyer sur plusieurs réseaux d'interaction protéine-protéine chez l'homme et construire des régresseurs qui partagent une partie de leurs paramètres.

Annexes

Annexe A

Régression à noyaux d'entrée et de sortie pour la classification structurée

A.1 Description

Dans certains problèmes de classification multiclasse, les classes ne sont pas disjointes et il peut exister des relations entre ces classes, par exemple des relations d'inclusion ou de similarité. Ce problème de classification est généralement appelé classification structurée. La catégorisation de documents, ainsi que la prédiction de fonctions des protéines sont deux exemples de problèmes de classification structurée.

Le problème de la classification multiclasse peut être vu comme un problème d'apprentissage d'une fonction à valeurs vectorielles. Cette approche présente l'avantage de permettre la prise en compte des corrélations existant entre les tâches. Dans ce problème de classification, $\mathcal{F}_y = \mathbb{R}^d$, où d correspond au nombre de classes.

A.2 Application à la classification de documents

Nous nous sommes intéressés au problème de la classification structurée de documents. Pour cela, nous avons utilisé la base de données Reuters Corpus Volume 1 (RCV1) [Lewis et al., 2004], qui contient des articles d'information classés en quatre catégories principales : entreprise, politique, marché et économie. Ces catégories sont elles-même divisées en sous-catégories et l'ensemble des catégories est organisé selon une hiérarchie, appelée H.

A.2.1 Jeu de données

Le jeu de données utilisé correspond à la version *topics*[1] de la librairie LIBSVM [Chang & Lin, 2011]. Les descripteurs en entrée correspondent aux mots utilisés dans les articles. Seuls les mots et les catégories, pour lesquels la fréquence était supérieure ou égale à 0.75 ont été considérés. Nous avons ensuite utilisé 50 données étiquetées et 1000 données non étiquetées pour l'ensemble d'apprentissage, et 1000 données pour l'ensemble de test.

1. http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/multilabel.html. Le jeu de données utilisé correspond à l'ensemble d'apprentissage de RCV1v2(topics, full sets).

A.2.2 Protocole

Un noyau linéaire a été utilisé en entrée. Un noyau de diffusion associé à la matrice d'adjacence de la hiérarchie H des catégories avec un paramètre de diffusion égal à 1 a été défini pour la matrice A intervenant dans le noyau à valeur opérateur décomposable. Pour W, nous avons considéré la matrice d'adjacence du graphe obtenue avec les k-plus proches voisins pour $k = 10$, et en utilisant les coefficients provenant d'un noyau gaussien sur les entrées. L est choisi comme le noyau de diffusion de la matrice d'adjacence construite. Le paramètre β de ce noyau de diffusion est sélectionné par validation croisée leave-one-out, ainsi que le paramètre λ_1. Enfin, nous avons utilisé la fonction de perte des moindres carrés.

Les performances ont été mesurées en évaluant la similarité entre $y(v)$ et $\hat{y}(v)$:

$$\left\langle \frac{y(v)}{\|y(v)\|}, \frac{\hat{y}(v)}{\|\hat{y}(v)\|} \right\rangle.$$

Nous avons comparé deux modèles pour résoudre ce problème de classification structurée :

1. $\hat{y}_1(v) = A \times \hat{h}_1(u)$, avec le noyau décomposable identité $\mathcal{K}_x(u, u') = \kappa_x(u, u') \times I_d$,
2. $\hat{y}_2(v) = \hat{h}_2(u)$, avec le noyau décomposable $\mathcal{K}_x(u, u') = \kappa_x(u, u') \times A$.

Dans un cas, nous avons :

$$\forall u \in \mathcal{U}, \hat{y}_1(v) = AV_\ell J \left(\lambda_1 I_{\ell+n} + K_{X_{\ell+n}}(J^T J + 2\lambda_2 L)\right)^{-1} X_{\ell+n}^T x(u),$$

et dans l'autre :

$$\forall u \in \mathcal{U}, \hat{y}_2(v) = \sum_{j=1}^{d} \gamma_j \mathbf{e}_j \mathbf{e}_j^T V_\ell J(\lambda_1 I_{\ell+n} + \gamma_j K_{X_{\ell+n}}(J^T J + 2\lambda_2 L))^{-1} X_{\ell+n}^T x(u).$$

A.2.3 Résultats

Nous avons comparé les deux modèles entre eux, ainsi qu'avec le cas où $A = I_d$, c'est à dire lorsque la structure de la hiérarchie H n'est pas prise en compte. Les résultats présentés sur la figure A.1 ont été obtenus pour dix choix aléatoires de l'ensemble d'apprentissage. On peut observer que les performances obtenues pour le modèle n°1 sont bien supérieures par rapport au modèle de base ne tenant pas compte de la hiérarchie sur les catégories de sujet. Par ailleurs, le modèle n°2 est bien moins compétitif que le modèle n°1, ainsi que le modèle de base.

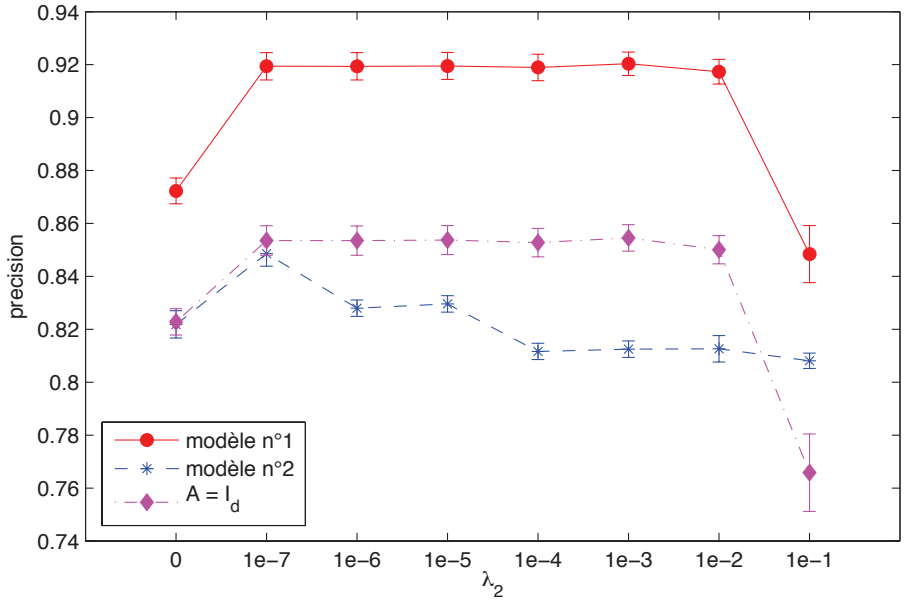

FIGURE A.1 – **Comparaison des différents modèles sur les données RCV1.** Les résultats ont été moyennés pour dix choix aléatoires de l'ensemble d'apprentissage et sont indiqués en fonction du paramètre de régularisation λ_2. Les deux modèles sont comparés entre eux, ainsi qu'avec le cas où $A = I_d$ pour montrer l'influence de l'utilisation de la hiérarchie sur les résultats de la classification.

Annexe B

Protéines du réseau d'interaction autour de CFTR

Symbole	ID Uniprot	Nom
ABCC2	Q92887	Canalicular multispecific organic anion transporter 1
ABCC4	O15439	Multidrug resistance-associated protein 4
ACCN3	Q9UHC3	Acid-sensing ion channel 3
ACTB	P60709	Actin, cytoplasmic 1
ACTG1	P63261	Actin, cytoplasmic 2
ACTN4	O43707	Alpha-actinin-4
ADD1	P35611	Alpha-adducin
ADRB2	P07550	Beta-2 adrenergic receptor
ANXA1	P04083	Annexin A1
ANXA2	P07355	Annexin A2
ANXA5	P08758	Annexin A5
ARHGEF7	Q14155	Rho guanine nucleotide exchange factor 7
ATP7B	P35670	Copper-transporting ATPase 2
BAG1	Q99933	BAG family molecular chaperone regulator 1
BCL3	P20749	B-cell lymphoma 3 protein
BECN1	Q14457	Beclin-1
BRCA1	P38398	Breast cancer type 1 susceptibility protein
CACNA1D	Q01668	Voltage-dependent L-type calcium channel subunit alpha-1D
CAPN3	P20807	Calpain-3
CASP3	P42574	Caspase-3
CASP7	P55210	Caspase-7
CASP9	P55211	Caspase-9
CDK1	P06493	Cyclin-dependent kinase 1
CDK2	P24941	Cyclin-dependent kinase 2
CDK3	Q00526	Cyclin-dependent kinase 3
CDK5	Q00535	Cyclin-dependent kinase 5
CDKN1B	P46527	Cyclin-dependent kinase inhibitor 1B
CDKN3	Q16667	Cyclin-dependent kinase inhibitor 3
CFL1	P23528	Cofilin-1
CFL2	Q9Y281	Cofilin-2
CFTR	P13569	Cystic fibrosis transmembrane conductance regulator
CHD4	Q14839	Chromodomain-helicase-DNA-binding protein 4
CLCN3	P51790	H(+)/Cl(-) exchange transporter 3
CLIC5	Q9NZA1	Chloride intracellular channel protein 5
CNGA2	Q16280	Cyclic nucleotide-gated olfactory channel

COMMD1	Q8N668	COMM domain-containing protein 1
COPS5	Q92905	COP9 signalosome complex subunit 5
CSPG5	O95196	Chondroitin sulfate proteoglycan 5
CTNNB1	P35222	Catenin beta-1
CTTN	Q14247	Src substrate cortactin
CUL2	Q13617	Cullin-2
DCTN2	Q13561	Dynactin subunit 2
DIABLO	Q9NR28	Diablo homolog, mitochondrial
DLG1	Q12959	Disks large homolog 1
DLGAP1	O14490	Disks large-associated protein 1
DNAJA1	P31689	DnaJ homolog subfamily A member 1
DNAJB1	P25685	DnaJ homolog subfamily B member 1
DNAJC5	Q9H3Z4	DnaJ homolog subfamily C member 5
DNM2	P50570	Dynamin-2
DSTN	P60981	Destrin
EFCAB5	A4FU69	EF-hand calcium-binding domain-containing protein 5
EGFR	P00533	Epidermal growth factor receptor
EZR	P15311	Ezrin
F2RL1	P55085	Proteinase-activated receptor 2
FGFR1	P11362	Fibroblast growth factor receptor 1
FLNC	Q14315	Filamin-C
FNIP1	Q8TF40	Folliculin-interacting protein 1
FURIN	P09958	Furin
FZD5	Q13467	Frizzled-5
FZD8	Q9H461	Frizzled-8
GADD45A	P24522	DNA damage-inducible transcript 1 protein
GDF11	O95390	Growth/differentiation factor 11
GFER	P55789	FAD-linked sulfhydryl oxidase ALR
GNAQ	P50148	Guanine nucleotide-binding protein G(q) subunit alpha
GOPC	Q9HD26	Golgi-associated PDZ and coiled-coil motif-containing protein
GOT2	P00505	Aspartate aminotransferase, mitochondrial
GRID1	Q9ULK0	Glutamate receptor delta-1 subunit
GRID2	O43424	Glutamate receptor delta-2 subunit
GUCY2C	P25092	Heat-stable enterotoxin receptor
HAND2	P61296	Heart- and neural crest derivatives-expressed protein 2
HIF1A	Q16665	Hypoxia-inducible factor 1-alpha
HSF1	Q00613	Heat shock factor protein 1
HSP90AA1	P07900	Heat shock protein HSP 90-alpha
HSPA8	P11142	Heat shock cognate 71 kDa protein
HSPH1	Q92598	Heat shock protein 105 kDa
HTRA2	O43464	Serine protease HTRA2, mitochondrial
ICAM1	P05362	Intercellular adhesion molecule 1
INPPL1	O15357	Inositol polyphosphate phosphatase-like protein 1
JUN	P05412	Transcription factor AP-1
KCNB1	Q14721	Potassium voltage-gated channel subfamily B member 1
KCND2	Q9NZV8	Potassium voltage-gated channel subfamily D member 2
KCNJ1	P48048	ATP-sensitive inward rectifier potassium channel 1

KDM1A	O60341	Lysine-specific histone demethylase 1A
KRT18	P05783	Keratin, type I cytoskeletal 18
KRT8	P05787	Keratin, type II cytoskeletal 8
MAP2K4	P45985	Dual specificity mitogen-activated protein kinase kinase 4
MARCKSL1	P49006	MARCKS-related protein
MIF	P14174	Macrophage migration inhibitory factor
MS4A3	Q96HJ5	Membrane-spanning 4-domains subfamily A member 3
MSN	P26038	Moesin
MSTN	O14793	Growth/differentiation factor 8
MYOT	Q9UBF9	Myotilin
MYOZ1	Q9NP98	Myozenin-1
NAPA	P54920	Alpha-soluble NSF attachment protein
NEDD4	P46934	E3 ubiquitin-protein ligase NEDD4
NF2	P35240	Merlin
NME7	Q9Y5B8	Nucleoside diphosphate kinase 7
OPRK1	P41145	Kappa-type opioid receptor
P2RY1	P47900	P2Y purinoceptor 1
PAG1	Q9NWQ8	Transmembrane adapter protein PAG
PDGFRA	P16234	Platelet-derived growth factor receptor alpha
PDGFRB	P09619	Platelet-derived growth factor receptor beta
PDPK1	O15530	3-phosphoinositide-dependent protein kinase 1
PDZD3	Q86UT5	Na(+)/H(+) exchange regulatory cofactor NHE-RF4
PDZK1	Q5T2W1	Na(+)/H(+) exchange regulatory cofactor NHE-RF3
PDZK1IP1	Q13113	PDZK1-interacting protein 1
PELP1	Q8IZL8	Proline-, glutamic acid- and leucine-rich protein 1
PHLDB2	Q86SQ0	Pleckstrin homology-like domain family B member 2
PIK3R1	P27986	Phosphatidylinositol 3-kinase regulatory subunit alpha
PLA2G4C	Q9UP65	Cytosolic phospholipase A2 gamma
PLCB1	Q9NQ66	Phosphoinositide phospholipase C-beta-1
PLCB2	Q00722	Phosphoinositide phospholipase C-beta-2
PLCB3	Q01970	Phosphoinositide phospholipase C-beta-3
PODXL	O00592	Podocalyxin
PPP2R1A	P30153	PP2A subunit A isoform PR65-alpha
PPP2R3B	Q9Y5P8	PP2A subunit B isoform PR48
PRKAA1	Q13131	5'-AMP-activated protein kinase catalytic subunit alpha-1
PRKACA	P17612	cAMP-dependent protein kinase catalytic subunit alpha
PRKAR2A	P13861	cAMP-dependent protein kinase type II-alpha regulatory subunit
PSMA5	P28066	Proteasome subunit alpha type-5
PSMD8	P48556	26S proteasome non-ATPase regulatory subunit 8
PTEN	P60484	Phosphatase and tensin homolog
PTH1R	Q03431	PTH/PTHrP type I receptor
PTK2	Q05397	Focal adhesion kinase 1
RAB27A	P51159	Ras-related protein Rab-27A
RDX	P35241	Radixin
RHOQ	P17081	Rho-related GTP-binding protein RhoQ
S100A10	P60903	Protein S100-A10
S100A7	P31151	Protein S100-A7

S100P	P25815	Protein S100-P
SCNN1A	P37088	Amiloride-sensitive sodium channel subunit alpha
SCNN1B	P51168	Amiloride-sensitive sodium channel subunit beta
SCNN1G	P51170	Amiloride-sensitive sodium channel subunit gamma
SCYL3	Q8IZE3	Ezrin-binding protein PACE-1
SGCD	Q92629	Delta-sarcoglycan
SGCG	Q13326	Gamma-sarcoglycan
SGK1	O00141	Serine/threonine-protein kinase Sgk1
SHANK2	Q9UPX8	SH3 and multiple ankyrin repeat domains protein 2
SLC22A12	Q96S37	Solute carrier family 22 member 12
SLC22A4	Q9H015	Solute carrier family 22 member 4
SLC26A3	P40879	Chloride anion exchanger
SLC26A6	Q9BXS9	Solute carrier family 26 member 6
SLC26A9	Q7LBE3	Solute carrier family 26 member 9
SLC4A7	Q9Y6M7	Sodium bicarbonate cotransporter 3
SLC4A8	Q2Y0W8	Electroneutral sodium bicarbonate exchanger 1
SCL9A3	P48764	Sodium/hydrogen exchanger 3
SLC9A3R1	O14745	Na(+)/H(+) exchange regulatory cofactor NHE-RF1
SLC9A3R2	Q15599	Na(+)/H(+) exchange regulatory cofactor NHE-RF2
SMAD3	P84022	Mothers against decapentaplegic homolog 3
SMAD4	Q13485	Mothers against decapentaplegic homolog 4
SMAD7	O15105	Mothers against decapentaplegic homolog 7
SMAD9	O15198	Mothers against decapentaplegic homolog 9
SNAP23	O00161	Synaptosomal-associated protein 23
SNAP25	P60880	Synaptosomal-associated protein 25
SOCS1	O15524	Suppressor of cytokine signaling 1
SRY	Q05066	Sex-determining region Y protein
STMN1	P16949	Stathmin
STUB1	Q9UNE7	E3 ubiquitin-protein ligase CHIP
STX1A	Q16623	Syntaxin-1A
STX2	P32856	Syntaxin-2
STX3	Q13277	Syntaxin-3
STX4	Q12846	Syntaxin-4
STX6	O43752	Syntaxin-6
STXBP1	P61764	Syntaxin-binding protein 1
STXBP2	Q15833	Syntaxin-binding protein 2
STXBP5	Q5T5C0	Syntaxin-binding protein 5
SYT1	P21579	Synaptotagmin-1
SYTL4	Q96C24	Synaptotagmin-like protein 4
TAB1	Q15750	TGF-beta-activated kinase 1-binding protein 1
TBC1D10A	Q9BXI6	TBC1 domain family member 10A
TCEB1	Q15369	Transcription elongation factor B polypeptide 1
TCHP	Q9BT92	Trichoplein keratin filament-binding protein
TM4SF1	P30408	Transmembrane 4 L6 family member 1
TNIK	Q9UKE5	TRAF2 and NCK-interacting protein kinase
TNK2	Q07912	Activated CDC42 kinase 1
TRPC4	Q9UBN4	Short transient receptor potential channel 4

TRPC5	Q9UL62	Short transient receptor potential channel 5
TRPV5	Q9NQA5	Transient receptor potential cation channel subfamily V member 5
TRPV6	Q9H1D0	Transient receptor potential cation channel subfamily V member 6
TSC2	P49815	Tuberin
TUBA4A	P68366	Tubulin alpha-4A chain
UBE2D2	P62837	Ubiquitin-conjugating enzyme E2 D2
UCHL1	P09936	Ubiquitin carboxyl-terminal hydrolase isozyme L1
UCHL5	Q9Y5K5	Ubiquitin carboxyl-terminal hydrolase isozyme L5
USP25	Q9UHP3	Ubiquitin carboxyl-terminal hydrolase 25
VAMP1	P23763	Vesicle-associated membrane protein 1
VAMP2	P63027	Vesicle-associated membrane protein 2
VCL	P18206	Vinculin
VCP	P55072	Transitional endoplasmic reticulum ATPase
VIM	P08670	Vimentin
WFIKKN1	Q96NZ8	Growth and differentiation factor-associated serum protein 2
XAF1	Q6GPH4	XIAP-associated factor 1
XBP1	P17861	X-box-binding protein 1
XIAP	P98170	E3 ubiquitin-protein ligase XIAP
XRN1	Q8IZH2	5'-3' exoribonuclease 1
YAP1	P46937	Yorkie homolog
ZFYVE9	O95405	Zinc finger FYVE domain-containing protein 9
ZNF506	Q5JVG8	Zinc finger protein 506

Annexe C

Liste des interactions protéine-protéine prédites

Le tableau suivant contient les interactions nouvelles parmi les 100 interactions obtenues avec les meilleurs scores pour l'inférence du réseau de PPI autour de la protéine CFTR (voir chapitre 8 pour plus de détails).

No	Protéine 1	Protéine 2	No	Protéine 1	Protéine 2
1	STX1A	STX3	33	SLC9A3R1	BECN1
2	STX1A	STX2	34	SLC9A3R1	STXBP1
3	STX1A	STX4	35	SLC26A9	HSP90AA1
4	XIAP	PTEN	36	S100A10	S100P
5	NEDD4	PTEN	37	SLC9A3R1	FZD5
6	COPS5	TNIK	38	DNAJC5	TNIK
7	COPS5	SMAD3	39	SLC9A3R1	FZD8
8	HSPA8	TNIK	40	SLC26A6	HSP90AA1
9	DNAJC5	SNAP25	41	SLC9A3R1	STUB1
10	SNAP23	VAMP2	42	SLC9A3R1	BAG1
11	STX1A	STX6	43	S100A10	SYTL4
12	SLC9A3R1	ICAM1	44	NEDD4	VAMP2
13	SLC9A3R1	SOCS1	45	S100A10	PLCB2
14	DNAJC5	BAG1	46	GOPC	DSTN
15	SNAP23	VAMP1	47	S100A10	SYT1
16	SNAP23	STX6	48	S100A10	PLCB3
17	SLC9A3R1	DSTN	49	SLC9A3R1	SNAP25
18	STX1A	RAB27A	50	S100A10	S100A7
19	SNAP23	STXBP2	51	STX1A	PRKAR2A
20	DNAJC5	DNAJB1	52	SLC9A3R1	HTRA2
21	SNAP23	STXBP1	53	GOPC	TNIK
22	HSPA8	SNAP25	54	SLC9A3R1	CDKN1B
23	NEDD4	NAPA	55	COPS5	CFL1
24	ACTG1	JUN	56	ANXA5	ANXA2
25	SNAP23	SNAP25	57	SLC26A3	HSP90AA1

26	DNAJC5	DNAJA1	58	S100A10	PLCB1
27	SLC9A3R1	TNIK	59	ANXA5	ANXA1
28	HSPH1	DNAJB1	60	HSPH1	BAG1
29	HSPH1	DNAJA1	61	SLC9A3R2	DSTN
30	SLC9A3R1	CFL	62	COPS5	DSTN
31	DNAJC5	STUB1	63	SLC26A9	ABCC4
32	SLC9A3R1	ACTB			

Annexe D

Produit de Kronecker et opérateur vec

Nous présentons dans cette annexe les principales propriétés du produit de Kronecker et de l'opérateur vec, qui ont été utilisées dans le chapitre 4.

Produit de Kronecker

Soient $A \in \mathbb{R}^{m\times n}$ et $B \in \mathbb{R}^{p\times q}$. Le produit de Kronecker entre A et B est une matrice de $m \times n$ blocs, chacun de ces blocs étant de taille $p \times q$, qui est définie de la façon suivante :

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix} \in \mathbb{R}^{mp\times nq}.$$

Le produit de Kronecker vérifie les propriétés suivantes :

1. Bilinéarité et associativité :
   - $A \otimes (B + C) = A \otimes B + A \otimes C$
   - $(A \otimes B) \otimes C = A \otimes (B \otimes C)$
2. Produit usuel :
   - $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$
3. Inverse :
   - $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$
4. Transposition :
   - $(A \otimes B)^T = A^T \otimes B^T$

Opérateur Vec

Il est parfois commode de réarranger les éléments d'une matrice A de taille $m\times n$ en un vecteur colonne de dimension mn. L'opérateur vec consiste à empiler les n colonnes $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n$ de A les unes sous les autres :

$$\mathrm{vec}(A) = \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_n \end{pmatrix}$$

Pour toutes matrices A et B de même taille, $\text{vec}(A+B) = \text{vec}(A) + \text{vec}(B)$.

Proposition 14 *Pour toutes matrices $A \in \mathbb{R}^{m\times n}, X \in \mathbb{R}^{n\times p}$ et $B \in \mathbb{R}^{p\times q}$,*

$$\text{vec}(AXB) = (B^T \otimes A)\,\text{vec}(X).$$

Proposition 15 *Comme pour tout vecteur colonne $\mathbf{a}$, $\text{vec}(\mathbf{a}^T) = \text{vec}(\mathbf{a}) = \mathbf{a}$, on peut déduire que pour tout $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{n\times q}$ et $\mathbf{x} \in \mathbb{R}^{p\times 1}$:*

$$\begin{aligned} AB\mathbf{x} &= \text{vec}(AB\mathbf{x}) = \text{vec}(\mathbf{x}^T B^T A^T) \\ &= (\mathbf{x}^T \otimes A)\,\text{vec}(B) = (A \otimes \mathbf{x}^T)\,\text{vec}(B^T). \end{aligned}$$

Proposition 16 *Pour toutes matrices $A, B \in \mathbb{R}^{m\times n}$,*

$$\text{trace}(A^T B) = \text{vec}(A)^T\,\text{vec}(B).$$

Proposition 17 *$\forall A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{m\times p}, C \in \mathbb{R}^{p\times q}$ et $D \in \mathbb{R}^{n\times q}$,*

$$\text{trace}(A^T BCD^T) = \text{vec}(A)^T (D \otimes B)\,\text{vec}(C).$$

Matrice de "Vec-permutation"

Soit A une matrice de dimension $m \times n$. $\text{vec}(A)$ et $\text{vec}(A^T)$ sont tous les deux des vecteurs colonne de longueur mn qui ont été obtenus en réarrangeant les éléments de la matrice A. Cependant, dans le cas de $\text{vec}(A^T)$ les éléments sont réarrangés ligne par ligne, alors qu'ils sont réarrangés colonne par colonne dans le cas de $\text{vec}(A)$. Il existe une matrice de permutation de dimension $mn \times mn$, notée P_{mn}, telle que :

$$\text{vec}(A^T) = P_{mn}\,\text{vec}(A).$$

Par exemple, dans le cas où $m = 2, n = 3$:

$$P_{23} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Cette matrice de permutation vérifie les propriétés suivantes :
- $\text{vec}(A) = P_{nm}\,\text{vec}(A^T) = P_{nm}P_{mn}\,\text{vec}(A)$
- $I_{mn} = P_{nm}P_{mn}$
- $B \otimes A = P_{pm}(A \otimes B)P_{nq}$, où B est une matrice de taille $p \times q$.

Bibliographie

Adamic, L. A. and Adar, E. Friends and neighbors on the web. *Social Networks*, 25 (3) :211 – 230, 2003.

Al Hasan, M., Chaoji, V., Salem, S., and Zaki, M. Link prediction using supervised learning. In *Proceedings of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

Allen, D. M. The relationship between variable selection and data augmentation and a method for prediction. *Technometrics*, 16(1) :125–127, 1974.

Aloy, P. and Russell, R. B. Interrogating protein interaction networks through structural biology. *Proceedings of the National Academy of Sciences*, 99(9) :5896–5901, 2002.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3) :403–410, 1990.

Alvarez, M., Rosasco, L., and Lawrence, N. D. Kernels for vector-valued functions : a review. In *Technical report MIT-CSAIL-TR-2011-033*, 2011.

Amaral, M. D. Processing of cftr : Traversing the cellular maze - how much cftr needs to go through to avoid cystic fibrosis. *Pediatric Pulmonology*, 39 :479–491, 2005.

Argyriou, A., Micchelli, C. A., and Pontil, M. When is there a representer theorem ? Vector vs matrix regularizers. *Journal of Machine Learning Research*, 10 :2507–2529, 2009.

Aronszajn, N. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3) :337–404, 1950.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. Gene ontology : tool for the unification of biology. *Nature Genetics*, 25(1) :25–29, 2000.

Bader, G. D., Betel, D., and Hogue, C. W. V. Bind : the biomolecular interaction network database. *Nucleic Acids Research*, 31(1) :248–250, 2003.

Bairoch, A. The enzyme database in 2000. *Nucleic Acids Research*, 28(1) :304–305, 2000.

Baldassarre, L., Rosasco, L., Barla, A., and Verri, A. Multi-output learning via spectral filtering. *Machine Learning*, 87 :259–301, 2012.

Bansal, M., Gatta, G. D., and di Bernardo, D. Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, 22(7) :815–822, 2006.

Barabasi, A.-L. and Oltvai, Z. N. Network biology : understanding the cell's functional organization. *Nature Review Genetics*, 5 :101–113, 2004.

Barabási, A. L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., and Vicsek, T. Evolution of the social network of scientific collaborations. *Physica A : Statistical Mechanics and its Applications*, 311(3-4) :590 – 614, 2002.

Basilico, J. and Hofmann, T. Unifying collaborative and content-based filtering. In *Proceedings of the 21st International Conference on Machine Learning (ICML)*, 2004.

Belkin, M. and Niyogi, P. Using manifold structure for partially labeled classification. In Becker, S., Thrun, S., and Obermayer, K. (eds.), *Advances in Neural Information Processing Systems 15*, Cambridge, MA, 2003. MIT Press.

Belkin, M. and Niyogi, P. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56(1-3) :209–239, 2004.

Belkin, M. and Niyogi, P. Towards a theoritical foundation for laplacian-based manifold methods. In *Proceedings of the 18th Annual Conference on Learning Theory*, pp. 486–500, 2005.

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

Ben-Hur, A. and Noble, W. S. Kernel methods for predicting protein–protein interactions. *Bioinformatics*, 21(1) :38–46, 2005.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. The protein data bank. *Nucleic Acids Research*, 28(1) :235–242, 2000.

Birlutiu, A. *Machine learning for pairwise data : applications for preference learning and supervised network inference*. PhD thesis, Radboud University Nijmegen, 2011.

Birlutiu, A., d'Alché Buc, F., and Heskes, T. Combining protein and network topology information for predicting protein-protein interactions : a bayesian approach. submitted, 2012.

Bleakley, K. and Yamanishi, Y. Supervised prediction of drug-target interactions using bipartite local models. *Bioinformatics*, 25(18) :2397–2403, 2009.

Bleakley, K., Biau, G., and Vert, J.-P. Supervised reconstruction of biological networks with local models. *Bioinformatics*, 23(13) :i57–i65, 2007.

Bock, J. R. and Gough, D. A. Predicting protein-protein interactions from primary structure. *Bioinformatics*, 17(5) :455–460, 2001.

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

Brin, S. and Page, L. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1-7) :107–117, 1998.

Brouard, C., d'Alché-Buc, F., and Szafranski, M. Semi-supervised penalized output kernel regression for link prediction. In *Proceedings of the 28th International Conference on Machine Learning*, 2011.

Brouard, C., Vrain, C., Dubois, J., Castel, D., Debily, M.-A., and d'Alché Buc, F. Learning a markov logic network for supervised gene regulatory network inference. submitted, 2012.

Butte, A. J., Tamayo, P., Slonim, D., Golub, T. R., and Kohane, I. S. Discovering functional relationships between rna expression and chemotherapeutic susceptibility using relevance networks. *Proceedings of the National Academy of Sciences*, 97(22) :12182–12186, 2000.

Caponnetto, A., Micchelli, C. A., Pontil, M., and Ying, Y. Universal multitask kernels. *Journal of Machine Learning Research*, 9 :1615–1646, 2008.

Chang, C.-C. and Lin, C.-J. LIBSVM : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chapelle, O. and Zien, A. Semi-supervised classification by low density separation. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, 2005.

Chapelle, O., Schölkopf, B., and Zien, A. (eds.). *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

Chen, J., Hsu, W., Lee, M. L., and Ng, S.-K. Increasing confidence of protein interactomes using network topological metrics. *Bioinformatics*, 22(16) :1998–2004, 2006a.

Chen, J.-H., Cai, Z., Li, H., and Sheppard, D. N. Function of cftr protein : ion transport. In Bush, A., Alton, E. W. F. W., Davies, J. C., Griesebach, U., and Jaffe, A. (eds.), *Cystic Fibrosis in the 21st Century*, pp. 38–44. Karger, 2006b.

Chen, T., He, H. L., and Church, G. M. Modeling gene expression with differential equations. *Pacific Symposium on Biocomputing*, pp. 29–40, 1999.

Chowdhury, G. *Introduction to Modern Information Retrieval, Third Edition*. Facet Publishing, 3rd edition, 2010.

Cortes, C. and Vapnik, V. Support vector networks. *Machine Learning*, 20 :273–297, 1995.

Cortes, C., Mohri, M., and Weston, J. A general regression technique for learning transductions. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 153–160, New York, NY, USA, 2005. ACM.

Dandekar, T., Snel, B., Huynen, M., and Bork, P. Conservation of gene order : a fingerprint of proteins that physically interact. *Trends in biochemical sciences*, 23 (9) :324–328, 1998.

Davis, J. and Goadrich, M. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning ; Pittsburgh*, pp. 233–240, 2006.

De Las Rivas, J. and Fontanillo, C. Protein-protein interactions essentials : key concepts to building and analyzing interactive networks. *PLOS Computational Biology*, 6(6) :e1000807, 06 2010.

Deane, C. M., Salwinski, L., Xenarios, I., and Eisenberg, D. Protein interactions : two methods for assessment of the reliability of high throughput observations. *Molecular & Cellular Proteomics*, 1(5) :349–356, 2002.

Deng, M., Mehta, S., Sun, F., and Chen, T. Inferring domain-domain interactions from protein-protein interactions. *Genome Research*, 12(10) :1540–1548, 2002.

di Bernardo, D., Thompson, M. J., Gardner, T. S., Chobot, S. E., Eastwood, E. L., Wojtovich, A. P., Elliott, S. J., Schaus, S. E., and Collins, J. J. Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nature Biotechnology*, 23 :377–383, 2005.

Dinuzzo, F., Ong, C. S., Gehler, P., and Pillonetto, G. Learning output kernels with block coordinate descent. In *Proceedings of the 28th International Conference on Machine Learning*, 2011.

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25) :14863–14868, 1998.

Enright, A. J., Iliopoulos, I., Kyrpides, N. C., and Ouzounis, C. A. Protein interaction maps for complete genomes based on gene fusion events. *Nature*, 402 :86–90, 1999.

Erdős, P. and Rényi, A. On random graphs. *Publicationes Mathematicae*, 6 :290–297, 1959.

Evgeniou, T., Micchelli, C. A., and Pontil, M. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6 :615–637, 2005.

Faith, J. J., Hayete, B., Thaden, J. T., Mogno, I., Wierzbowski, J., Cottarel, G., Kasif, S., Collins, J. J., and Gardner, T. S. Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biology*, 5(1) :e8, 2007.

Fawcett, T. An introduction to roc analysis. *Pattern Recognition Letters*, 27 : 861–874, 2005.

Friedman, N., Getoor, L., Koller, D., and Pfeffer, A. Learning probabilistic relational models. In *Proceedings of the 16th international joint conference on artificial intelligence*, IJCAI'99, 1999.

Friedman, N., Linial, M., Nachman, I., and Pe'er, D. Using bayesian networks to analyse expression data. *Journal of Computational Biology*, 7, 2004.

Gardner, T. S., di Bernardo, D., Lorenz, D., and Collins, J. J. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301(5629) :102–105, 2003.

Gavin, A.-C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J.M., Michon, A.-M., Cruciat, C.-M., Remor, M., Hofert, C., Schelder, M., Brajenovic, M., Ruffner, H., Merino, A., Klein, K., Hudak, M., Dickson, D., Rudi, T., Gnau, V., Bauch, A., Bastuck, S., Huhse, B., Leutwein, C., Heurtier, M.-A., Copley, R.R., Edemann, A., Querfurth, E., Rybin, V., Drewes, G., Raida, M., Bouwmeester, T., Bork, P., Seraphin, B., Kuster, B., Neubauer, G., and Superti-Furga, G. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415 :141–147, 2002.

Ge, H., Liu, Z., Church, G. M., and Vidal, M. Correlation between transcriptome and interactome mapping data from saccharomyces cerevisiae. *Nature Genetics*, 29(4) :482–486, 2001.

Getoor, L. and Diehl, C. P. Link mining : a survey. *SIGKDD Explorations Newsletters*, 7(2) :3–12, 2005.

Getoor, L. and Taskar, B. (eds.). *Introduction to statistical relational learning*. MIT Press, 2007.

Getoor, L., Friedman, N., Koller, D., and Taskar, B. Learning probabilistic models of link structure. *Journal of Machine Learning Research*, 3 :679–707, 2002.

Geurts, P., Wehenkel, L., and d'Alché Buc F. Kernelizing the output of tree-based methods. In *Proceedings of the 23th International Conference on Machine learning*, pp. 345–352, 2006.

Geurts, P., Touleimat, N., Dutreix, M., and d'Alché-Buc, F. Inferring biological networks with output kernel trees. *BMC Bioinformatics (PMSB06 special issue)*, 8(Suppl 2) :S4, 2007.

Giot, L., Bader, J. S., Brouwer, C., Chaudhuri, A., Kuang, B., Li, Y., Hao, Y. L., Ooi, C. E., Godwin, B., Vitols, E., Vijayadamodar, G., Pochart, P., Machineni, H., Welsh, M., Kong, Y., Zerhusen, B., Malcolm, R., Varrone, Z., Collis, A., Minto, M., Burgess, S., McDaniel, L., Stimpson, E., Spriggs, F., Williams, J., Neurath, K., Ioime, N., Agee, M., Voss, E., Furtak, K., Renzulli, R., Aanensen, N., Carrolla, S., Bickelhaupt, E., Lazovatsky, Y., DaSilva, A., Zhong, J., Stanyon, C. A., Finley, R. L., White, K. P., Braverman, M., Jarvie, T., Gold, S., Leach, M., Knight, J., Shimkets, R. A., McKenna, M. P., Chant, J., and Rothberg, J. M. A protein interaction map of drosophila melanogaster. *Science*, 302(5651) :1727–1736, 2003.

Globerson, A., Chechik, G., Pereira, F., and Tishby, N. Euclidean embedding of co-occurrence data. *Journal of Machine Learning Research*, 8 :2265–2295, 2007.

Goldberg, D. S. and Roth, F. P. Assessing experimentally derived interactions in a small world. *Proceedings of the National Academy of Sciences*, 100(8) :4372–4376, 2003.

Golub, G. H., Heath, M., and Wahba, G. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21 :215–223, 1979.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. Measuring statistical dependence with hilbert-schmidt norms. In Jain, S., Simon, H., and Tomita, E. (eds.), *Algorithmic Learning Theory*, volume 3734 of *Lecture Notes in Computer Science*, pp. 63–77. Springer Berlin Heidelberg, 2005.

Guo, Y., Yu, L., Wen, Z., and Li, M. Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. *Nucleic Acids Research*, 36(9) :3025–3030, 2008.

Hahn, A., Rahnenfuhrer, J., Talwar, P., and Lengauer, T. Confirmation of human protein interaction data by human expression data. *BMC Bioinformatics*, 6(1) : 112, 2005.

Han, D., Kim, H.-S., Seo, J., and Jang, W. A domain combination based probabilistic framework for protein-protein interaction prediction. *Genome Informatics*, 14 : 250–259, 2003.

Hastie, R., R., Tibshirani ;, and Friedman, J. *The elements of statistical learning*. Springer-Verlag, 2008.

Haury, A.-C., Mordelet, F., Vera-Licona, P., and Vert, J.-P. Tigress : Trustful inference of gene regulation using stability selection. *BMC Systems Biology*, 6(1) : 145, 2012.

Ho, Y., Gruhler, A., Heilbut, A., Bader, G. D., Moore, L., Adams, S. L., Millar, A., Taylor, P., Bennett, K., Boutilier, K., Yang, L., Wolting, C., Donaldson, I., Schandorff, S., Shewnarane, J., Vo, M., Taggart, J., Goudreault, M., Muskat, B., Alfarano, C., Dewar, D., Lin, Z., Michalickova, K., Willems, A. R., Sassi, H., Nielsen, P. A., Rasmussen, K. J., Andersen, J. R., Johansen, L. E., Hansen, L. H., Jespersen, H., Podtelejnikov, A., Nielsen, E., Crawford, J., Poulsen, V., Sorensen, B. D., Matthiesen, J., Hendrickson, R. C., Gleeson, F., Pawson, T., Moran, M. F., Durocher, D., Mann, M., Hogue, C. W., Figeys, D., and Tyers, M. Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. *Nature*, 415(6868) :180–183, 2002.

Hoerl, A. E. and Kennard, R. W. Ridge regression : biases estimation for nonorthogonal problems. *Technometrics*, 12(1) :55–67, 1970.

Hoff, P. D. Multiplicative latent factor models for description and prediction of social networks. *Computational and Mathematical Organization Theory*, 2008.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97 :1090–1098, 2002.

Honeine, P. and Richard, C. Preimage problem in kernel-based machine learning. *IEEE Signal Processing Magazine*, 28(2) :77–88, 2011.

Huang, Z., Zeng, D., and Chen, H. A unified recommendation framework based on probabilistic relational models. In *14th annual Workshop on Information Technologies and Systems (WITS)*, pp. 8–13, 2004.

Hue, M., Riffle, M., Vert, J.-P., and Noble, W. Large-scale prediction of protein-protein interactions from structures. *BMC Bioinformatics*, 11(1) :144, 2010.

Huh, W.-K., Falvo, J. V., Gerke, L. C., Carroll, A. S., Howson, R. W., Weissman, J. S., and O'Shea, E. K. Global analysis of protein localization in budding yeast. *Nature*, 425 :686–691, 2003.

Ito, T., Tashiro, K., Muta, S., Ozawa, R., Chiba, T., Nishizawa, M., Yamamoto, K., Kuhara, S., and Sakaki, Y. Toward a protein-protein interaction map of the budding yeast : A comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proceedings of the National Academy of Sciences*, 97(3) :1143–1147, 2000.

Jacob, L. and Vert, J.-P. Protein-ligand interaction prediction : an improved chemogenomics approach. *Bioinformatics*, 24(19) :2149–2156, 2008.

Jacob, L., Bach, F., and Vert, J. Clustered multi-task learning : A convex formulation. In *Advances in Neural Information Processing Systems*, 2008.

Jaimovich, A., Elidan, G., Margalit, H., and Friedman, N. Towards an integrated protein-protein interaction network : a relational markov network approach. *Journal of Computational Biology*, 13 :145–164, 2006.

Jain, S. and Bader, G. D. An improved method for scoring protein-protein interactions using semantic similarity within the gene ontology. *BMC Bioinformatics*, 11(1) :562, 2010.

Jansen, R., Greenbaum, D., and Gerstein, M. Relating whole-genome expression data with protein-protein interactions. *Genome Research*, 12 :37–46, 2002.

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

Jeh, G. and Widom, J. Simrank : a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pp. 538–543, New York, NY, USA, 2002. ACM.

Jeong, H., Mason, S. P., Barabasi, A. L., and Oltvai, Z. N. Lethality and centrality in protein networks. *Nature*, 411(6833) :41–42, May 2001.

Joachims, T. Transductive inference for text classification using support vector machines. In *Proceedings of the International Conference on Machine Learning*, 1999.

Kadri, H., Duflos, E., Preux, P., Canu, S., and Davy, M. Nonlinear functional regression : a functional rkhs approach. In *JMLR Proceedings of International Conference on Artificial Intelligence and Statistics*, volume 9, 2010.

Kadri, H., Rabaoui, A., Preux, P., Duflos, E., and Rakotomamonjy, A. Functional regularized least squares classification with operator-valued kernels. In *Proceedings of the 28th International Conference on Machine Learning*, 2011.

Kadri, H., Rakotomamonjy, A., Bach, F., and Preux, P. Multiple operator-valued kernel learning. In *Advances in Neural Information Processing Systems*, 2012.

Kashima, H. and Abe, N. A parameterized probabilistic model of network evolution for supervised link prediction. In *Sixth International Conference on Data Mining (ICDM)*, pp. 340–349, 2006.

Kashima, H., Kato, T., Yamanishi, Y., Sugiyama, M., and Tsuda, K. Link propagation : A fast semi-supervised learning algorithm for link prediction. In *Proceedings of the 9th SIAM International Conference on Data Mining*, pp. 1099–1110, 2009a.

Kashima, H., Yamanishi, Y., Kato, Ts., Sugiyama, M., and Tsuda, K. Simultaneous inference of biological networks of multiple species from genome-wide data and evolutionary information. *Bioinformatics*, 25(22) :2962–2968, 2009b.

Kato, T., Tsuda, K., and Asai, K. Selective integration of multiple biological data for supervised network inference. *Bioinformatics*, 21(10) :2488–2495, 2005.

Katz, L. A new status index derived from sociometric analysis. *Psycometrika*, 18 (1) :39–43, 1953.

Kerrien, S., Aranda, B., Breuza, L., Bridge, A., Broackes-Carter, F., Chen, C., Duesbury, M., Dumousseau, M., Feuermann, M., Hinz, U., Jandrasits, C., Jimenez, R. C., Khadake, J., Mahadevan, U., Masson, P., Pedruzzi, I., Pfeiffenberger, E., Porras, P., Raghunath, A., Roechert, B., Orchard, S., and Hermjakob, H. The intact molecular interaction database in 2012. *Nucleic Acids Research*, 40(D1) : D841–D846, 2012.

Keskin, O., Gursoy, A., Ma, B., and Nussinov, R. Principles of protein-protein interactions : what are the preferred ways for proteins to interact ? *Chemical Reviews*, 108 :1225–1244, 2008.

Kim, W. K., Park, J., and Suh, J. K. Large scale statistical prediction of protein-protein interaction by potentially interacting domain (pid) pair. *Genome Informatics*, 13 :42–50, 2002.

Kimeldorf, G. and Wahba, G. Some results on tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33 :82–95, 1971.

Kondor, R. I. and Lafferty, J. D. Diffusion kernels on graphs and other discrete input spaces. In *Proceedings of the 19th International Conference on Machine Learning*, pp. 315–322, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

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

Lane, L., Argoud-Puy, G., Britan, A., Cusin, I., Duek, P. D., Evalet, O., Gateau, A., Gaudet, P., Gleizes, A., Masselot, A., Zwahlen, C., and Bairoch, A. nextprot : a knowledge platform for human proteins. *Nucleic Acids Research*, 40(D1) :D76–D83, 2012.

Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano, J. A., Armananzas, R., Santafé, G., Pérez, A., and Robles, V. Machine learning in bioinformatics. *Briefings in Bioinformatics*, 7(1) :86–112, 2006.

Lee, H., Deng, M., Sun, F., and Chen, T. An integrated approach to the prediction of domain-domain interactions. *BMC Bioinformatics*, 7(1) :269, 2006.

Lesk, A. M. *Introduction to protein science*. Oxford university press, 2010.

Leslie, C. S., Eskin, E., and Noble, W. S. The spectrum kernel : a string kernel for svm protein classification. In *Proceedings of the Pacific Symposium on Biocomputing*, pp. 564–575, 2002.

Lewis, D. D., Yang, Y., Rose, T. G., and Li, F. RCV1 : A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5 :361–397, 2004.

Li, S., Armstrong, C. M., Bertin, N., Ge, H., Milstein, S., Boxem, M., Vidalain, P.-O., Han, J.-D. J., Chesneau, A., Hao, T., Goldberg, D. S., Li, N., Martinez, M., Rual, J.-F., Lamesch, P., Xu, L., Tewari, M., Wong, S. L., Zhang, L. V., Berriz, G. F., Jacotot, L., Vaglio, P., Reboul, J., Hirozane-Kishikawa, T., Li, Q., Gabel, H. W., Elewa, A., Baumgartner, B., Rose, D. J., Yu, H., Bosak, S., Sequerra, R., Fraser, A., Mango, S. E., Saxton, W. M., Strome, S., van den Heuvel, S., Piano, F., Vandenhaute, J., Sardet, C., Gerstein, M., Doucette-Stamm, L., Gunsalus, K. C., Harper, J. W., Cusick, M. E., Roth, F. P., Hill, D. E., and Vidal, M. A map of the interactome network of the metazoan c. elegans. *Science*, 303(5657) : 540–543, 2004.

Liben-Nowell, D. and Kleinberg, J. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58(7) : 1019–1031, 2007.

Licata, L., Briganti, L., Peluso, D., Perfetto, L., Iannuccelli, M., Galeota, E., Sacco, F., Palma, A., Nardozza, A. P., Santonico, E., Castagnoli, L., and Cesareni, G. Mint, the molecular interaction database : 2012 update. *Nucleic Acids Research*, 40(D1) :D857–D861, 2012.

Lichtenwalter, R. N., Lussier, J. T., and Chawla, N. V. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pp. 243–252, 2010.

Lin, N., Wu, B., Jansen, R., Gerstein, M., and Zhao, H. Information assessment on predicting protein-protein interactions. *BMC Bioinformatics*, 5(1) :154, 2004.

Lippert, C., Stegle, O., Ghahramani, Z., and Karsten, M. B. A kernel method for unsupervised structured network inference. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 368–375, 2009.

Lu, L., Arakaki, A. K., Lu, H., and Skolnick, J. Multimeric threading-based prediction of protein-protein interactions on a genomic scale : application to the saccharomyces cerevisiae proteome. *Genome Research*, 13 :1146–1154, 2003.

Lu, L. J., Xia, Y., Paccanaro, A., Yu, H., and Gerstein, M. Assessing the limits of genomic data integration for predicting protein networks. *Genome Research*, 15 (7) :945–953, 2005.

Marcotte, E. M., Pellegrini, M., Ng, H.-L., Rice, D. W., Yeates, T. O., and Eisenberg, D. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285(5428) :751–753, 1999.

Martin, S., Roe, D., and Faulon, J. L. Predicting protein-protein interactions using signature products. *Bioinformatics*, 21 :218–226, 2005.

Matthews, L. R., Vaglio, P., Reboul, J., Ge, H., Davis, B. P., Garrels, J., Vincent, S., and Vidal, M. Identification of potential interaction networks using sequence-based searches for conserved protein-protein interactions of 'interologs'. *Genome Research*, 11(12) :2120–2126, 2001.

Mewes, H., Amid, C., Arnold, R., Frishman, D., Guldener, U., Mannhaupt, G., Münsterkötter, M., Pagel, P., Strack, N., Stümpflen, V., Warfsmann, J., and Ruepp, A. Mips : analysis and annotation of proteins from whole genomes. *Nucleic Acids Research*, 32(Database issue) :D41–4, 2004.

Micchelli, C. A. and Pontil, M. Kernels for multi-task learning. In Saul, L. K., Weiss, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems 17*, pp. 921–928. MIT Press, 2004.

Micchelli, C. A. and Pontil, M. On learning vector-valued functions. *Neural Computation*, 17 :177–204, 2005.

Michaut, M., Kerrien, S., Montecchi-Palazzi, L., Chauvat, F., Cassier-Chauvat, C., Aude, J.-C., Legrain, P., and Hermjakob, H. Interoporc : automated inference of highly conserved protein interaction networks. *Bioinformatics*, 24(14) :1625–1631, 2008.

Miller, K., Griffiths, T., and Jordan, M. Nonparametric latent feature models for link prediction. In *Advances in Neural Information Processing Systems 22*, pp. 1276–1284, 2009.

Mordelet, F. and Vert, J.-P. Sirene : supervised inference of regulatory networks. *Bioinformatics*, 24(16) :i76–i82, 2008.

Nagamine, N. and Sakakibara, Y. Statistical prediction of protein-chemical interactions based on chemical structure and mass spectrometry data. *Bioinformatics*, 23(15) :2004–2012, 2007.

Newman, M. E. J. Clustering and preferential attachment in growing networks. *Physical Review E*, 64 :025102, 2001.

Ng, S.-K., Zhang, Z., and Tan, S.-H. Integrative approach for computationally inferring protein domain interactions. *Bioinformatics*, 19(8) :923–929, 2003.

Noble, W. and Ben-Hur, A. Integrating information for protein function prediction. In Lengauer, T. (ed.), *Bioinformatics-From Genomes to Therapies*. WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim, Germany, 2007.

Nooren, I. M. A. and Thornton, J. M. Diversity of protein-protein interactions. *EMBO Journal*, 22(14) :3486–3492, 2003.

O'Brien, K. P., Remm, M., and Sonnhammer, E. L. L. Inparanoid : a comprehensive database of eukaryotic orthologs. *Nucleic Acids Research*, 33(suppl 1) :D476–D480, 2005.

Ollero, M., Brouillard, F., and Edelman, A. Cystic fibrosis enters the proteomics scene : New answers to old questions. *Proteomics*, 6(14) :4084–4099, 2006.

Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G. D., and Maltsev, N. The use of gene clusters to infer functional coupling. *Proceedings of the National Academy of Sciences*, 96(6) :2896–2901, 1999.

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

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

Penrose, R. A generalized inverse for matrices. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 51, pp. 406–413, 1955.

Perkins, J. R., Diboun, I., Dessailly, B. H., Lees, J. G., and Orengo, C. Transient protein-protein interactions : structural, functional, and network properties. *Structure*, 18(10) :1233–1243, 2010.

Popescul, A. and Ungar, L. H. Statistical relational learning for link prediction. In *Proceedings of the workshop on learning statistical models from relational data at IJCAI*, 2003.

Prasad, T. S. K., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S., Telikicherla, D., Raju, R., Shafreen, B., Venugopal, A., Balakrishnan, L., Marimuthu, A., Banerjee, S., Somanathan, D. S., Sebastian, A., Rani, S., Ray, S., Kishore, C. J. H., Kanth, S., Ahmed, M., Kashyap, M., Mohmood, R., Ramachandra, Y. L., Krishna, V., Rahiman, A. B., Mohan, S., Ranganathan, P., Ramabadran, S., Chaerkady, R., and Pandey, A. Human protein reference database - 2009 update. *Nucleic Acids Research*, 37 :D767–D772, 2009.

Punta, M., Coggill, P. C., Eberhardt, R. Y., Mistry, J., Tate, J., Boursnell, C., Pang, N., Forslund, K., Ceric, G., Clements, J., Heger, A., Holm, L., Sonnhammer, E. L. L., Eddy, S. R., Bateman, A., and Finn, R. D. The pfam protein families database. *Nucleic Acids Research*, 40 (Database Issue) :D290–D301, 2012.

Qi, Y. and Noble, W. S. *Protein interaction networks : protein domain interaction and protein function prediction*, pp. 427–459. Springer, 2011.

Qi, Y., Klein-seetharaman, J., and Bar-joseph, Z. Random forest similarity for protein-protein interaction prediction from multiple sources. *Proceedings of the Pacific Symposium on Biocomputing*, 10 :531–542, 2005.

Qi, Y., Bar-joseph, Z., and Klein-seetharaman, J. Evaluation of different biological data and computational classification methods for use in protein interaction prediction. *Proteins*, 63 :490–500, 2006.

Rain, J.-C., Selig, L., De Reuse, H., Battaglia, V., Reverdy, C., Simon, S., Lenzen, G., Petel, F., Wojcik, J., Schachter, V., Chemama, Y., Labigne, A., and Legrain, P. The protein-protein interaction map of helicobacter pylori. *Nature*, 409 :211–215, 2001.

Rakotomamonjy, A., Bach, F. R., Canu, S., and Grandvalet, Y. Simplemkl. *Journal of Machine Learning Research*, 9 :2491–2521, 2008.

Rhodes, D. R., Tomlins, S. A., Varambally, S., Mahavisno, V., Barrette, T., Kalyana-Sundaram, S., Ghosh, D., Pandey, A., and Chinnaiyan, A. M. Probabilistic model of the human protein-protein interaction network. *Nature Biotechnology*, 23 :951–959, 2005.

Richardson, M. and Domingos, P. Markov logic : a unifying framework for statistical relational learning. In *Proceedings of the ICML-2004 Workshop on Statistical Relational Learning and its Connections to Other Fields*, pp. 49–54, 2004.

Richardson, M. and Domingos, P. Markov logic networks. *Machine Learning*, 62 (1-2) :107–136, 2006.

Rifkin, R. M. and Lippert, R. A. Notes on regularized least-squares. Technical report, MIT, Computer Science and Artificial Intelligence Laboratory, 2007.

Riley, R., Lee, C., Sabatti, C., and Eisenberg, D. Inferring protein domain interactions from databases of interacting proteins. *Genome Biology*, 6(10) :R89, 2005.

Riordan, J. R., Rommens, J. M., Kerem, B., Alon, N., Rozmahel, R., Grzelczak, Z., Zielenski, J., Lok, S., Plavsic, N., and Chou, J. L. Identification of the cystic fibrosis gene : cloning and characterization of complementary dna. *Science*, 245 : 1066–1073, 1989.

Rommens, J. M., Iannuzzi, M. C., Kerem, B., Drumm, M. L., Melmer, G., Dean, M., Rozmahel, R., Cole, J. L., Kennedy, D., and Hidaka, N. Identification of the cystic fibrosis gene : chromosome walking and jumping. *Science*, 245 :1059–1065, 1989.

Rual, J. F., Venkatesan, K., Hao, T., , Hirozane-Kishikawa, T., Dricot, A., Li, N., Berriz, G. F., Gibbons, F. D., Dreze, M., Ayivi-Guedehoussou, N., Klitgord, N., Simon, C., Boxem, M., Milstein, S., Rosenberg, J., Goldberg, D. S., Zhang, L. V., Wong, S. L., Franklin, G., Li, S., Albala, J. S., Lim, J., Fraughton, C., Llamosas,

E., Cevik, S., Bex, C., Lamesch, P., Sikorski, R. S., Vandenhaute, J., Zoghbi, H. Y., Smolyar, A., Bosak, S., Sequerra, R., Doucette-Stamm, L., Cusick, M. E., Hill, D. E., Roth, F. P., and Vidal, M. Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, 437(7062) :1173–1178, 2005.

Russell, A. B., Alber, F., Aloy, P., Davis, F. P., Korkin, D., Pichaud, M., Topf, M., and Sali, A. A structural perspective on protein-protein interactions. *Current Opinion in Structural Biology*, 14(3) :313–324, 2004.

Saito, R., Suzuki, H., and Hayashizaki, Y. Interaction generality, a measurement to assess the reliability of a protein-protein interaction. *Nucleic Acids Research*, 30 (5) :1163–1168, 2002.

Salwinski, L., Miller, C. S., Smith, A. J., Pettit, F.K., Bowie, J. U., and Eisenberg, D. The database of interacting proteins : 2004 update. *Nucleic Acids Research*, 32 (Database Issue) :D449–D451, 2004.

Sarukkai, R. R. Link prediction and path analysis using markov chains. *Computer Networks*, 33 :377–386, 2000.

Scholkopf, B. and Smola, A. J. *Learning with kernels*. MIT Press, 2002.

Scott, M. and Barton, G. Probabilistic prediction and ranking of human protein-protein interactions. *BMC Bioinformatics*, 8(1) :239, 2007.

Senkene, E. and Tempel'man, A. Hilbert spaces of operator-valued functions. *Lithuanian Mathematical Journal*, 13(4) :665–670, 1973.

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B., and Ideker, T. Cytoscape : A software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13 (11) :2498–2504, 2003.

Sharan, R., Ulitsky, I., and Shamir, R. Network-based prediction of protein function. *Molecular Systems Biology*, 3(88), 2007.

Sheldon, D. Graphical multi-task learning. Technical report, Cornell University, 2008. URL `http://web.engr.oregonstate.edu/~sheldon/`.

Shen, J., Zhang, J., Luo, X., Zhu, W., Yu, K., Chen, K., Li, Y., and Jiang, H. Predicting protein-protein interactions based only on sequences information. *Proceedings of the National Academy of Sciences*, 104(11) :4337–4341, 2007.

Shoemaker, B. A. and Panchenko, A. R. Deciphering protein-protein interactions. part i. experimental techniques and databases. *PLoS Computational Biology*, 3 (3) :e42, 2007a.

Shoemaker, B. A. and Panchenko, A. R. Deciphering protein-protein interactions. part ii. computational methods to predict protein and domain interaction partners. *PLoS Computational Biology*, 3(4) :e43, 2007b.

Skach, W. R. Defects in processing and trafficking of the cystic fibrosis transmembrane conductance regulator. *Kidney International*, 57 :825–831, 2000.

Smith, G. R. and Sternberg, M. J. E. Prediction of protein-protein interactions by docking methods. *Current Opinion in Structural Biology*, 12(1) :28–35, 2002.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, B., and Futcher, B. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, 9(12) :3273–3297, 1998.

Sprinzak, E. and Margalit, H. Correlated sequence-signatures as markers of protein-protein interaction. *Journal of Molecular Biology*, 311(4) :681–692, 2001.

Stark, C., Breitkreutz, B. J., Chatr-Aryamontri, A., Boucher, L., Oughtred, R., Livstone, M. S., Nixon, J., Van Auken, K., Wang, X., Shi, X., Reguly, T., Rust, J. M., Winter, A., Dolinski, K., and Tyers, M. The biogrid interaction database : 2011 update. *Nucleic Acids Research*, 2010.

Stein, A., Pache, R. A., Bernadó, P., Pons, M., and Aloy, P. Dynamic interactions of proteins in complex networks : a more structured view. *FEBS Journal*, 276 (19) :5390–5405, 2009.

Stelzl, U., Worm, U., Lalowski, M., Haenig, C., Brembeck, F. H., Goehler, H., Stroedicke, M., Zenkner, M., Schoenherr, A., Koeppen, S., Timm, J., Mintzlaff, S., Abraham, C., Bock, N., Kietzmann, S., Goedde, A., Toksoz, E., Droege, A., Krobitsch, S., Korn, B., Birchmeier, W., Lehrach, H., and Wanker, E. E. A human protein-protein interaction network : a resource for annotating the proteome. *Cell*, 122(6) :957–968, 2005.

Stone, M. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society*, 36(2) :111–147, 1974.

Su, A. I., Wiltshire, T., Batalov, S., Lapp, H., Ching, K. A., Block, D., Zhang, J., Soden, R., Hayakawa, M., Kreiman, G., Cooke, M. P., Walker, J. R., and Hogenesch, J. B. A gene atlas of the mouse and human protein-encoding transcriptomes. *Proceedings of the National Academy of Sciences*, 101(16) :6062–6067, 2004.

Szedmak, S., Shawe-Taylor, J., and Parado-Hernandez, E. Learning via linear operators : Maximum margin regression. Technical report, University of Southampton, UK, 2005.

Szklarczyk, D., Franceschini, A., Kuhn, M., Simonovic, M., Roth, A., Minguez, P., Doerks, T., Stark, M., Muller, J., Bork, P., Jensen, L.J., and von Mering, C. The string database in 2011 : functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Research*, 39(Database issue) :D561–D568, 2011.

Szummer, M. and Jaakkola, T. Partially labeled classification with markov random walks. In Dietterich, T. G., Becker, S., and Ghahramani, Z. (eds.), *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

Taskar, B., Wong, M., Abbeel, P., and Koller, D. Link prediction in relational data. In *Advances in Neural Information Processing Systems 15*, 2003.

Tastan, O., Qi, Y., Carbonell, J. G., and Klein-Seetharaman, J. Prediction of interactions between hiv-1 and human proteins by information integration. In *Pacific Symposium on Biocomputing (PSB*, volume 14, pp. 516–527, 2009.

Teixeira, M. C., Monteiro, P., Jain, P., Tenreiro, S., Fernandes, A. R., Mira, N. P., Alenquer, M., Freitas, A. T., Oliveira, A. L., and Sá-Correia, I. The yeastract database : a tool for the analysis of transcription regulatory associations in saccharomyces cerevisiae. *Nucleic Acids Research*, 34(suppl 1) :D446–D451, 2005.

Tikhonov, A. N. and Arsenin, V. Y. *Solutions of ill posed problems*. Wilson, W. H. and Sons, 1977.

Tsuda, K., Akaho, S., and Asai, K. The em algorithm for kernel matrix completion with auxiliary data. *Journal of Machine Learning Research*, 4 :67–81, 2003.

Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., A Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M., Johnston, M., Fields, S., and Rothberg, J. M. A comprehensive analysis of protein–protein interactions in saccharomyces cerevisiae. *Nature*, 403 :623–627, 2000.

Uhlen, M., Oksvold, P., Fagerberg, L., Lundberg, E., Jonasson, K., Forsberg, M., Zwahlen, M., Kampf, C., Wester, K., Hober, S., Wernerus, H., Björling, L., and Ponten, F. Towards a knowledge-based human protein atlas. *Nature Biotechnology*, 28 :1248–50, 2010.

Vapnik, V. *The nature of statistical learning theory*. Springer, NY, 1995.

Vapnik, V. *Statistical learning theory*. Wiley, 1998.

Vapnik, V. and Chervonenkis, A. Ordered risk minimization (i and ii). *Automation and Remote Control*, 34 :1226–1235, 1974.

Vert, J.-P. and Yamanishi, Y. Supervised graph inference. In *Advances in Neural Information Processing Systems*, pp. 1433–1440, 2005.

Vert, J.-P., Qiu, J., and Noble, W. A new pairwise kernel for biological network inference with support vector machines. *BMC Bioinformatics*, 8(Suppl 10) :S8, 2007.

von Mering, C., Krause, R., Snel, B., Cornell, M., Oliver, S. G., Fields, S., and Bork, P. Comparative assessment of large-scale data sets of protein–protein interactions. *Nature*, 417 :399–403, 2002.

Walhout, A. J. M., Sordella, R., Lu, X., Hartley, J. L., Temple, G. F., Brasch, M. A., Thierry-Mieg, N., and Vidal, M. Protein interaction mapping in c. elegans using proteins involved in vulval development. *Science*, 287(5450) :116–122, 2000.

Wang, C., Satuluri, V., and Parthasarathy, S. Local probabilistic models for link prediction. In *Proceedings of the 7th IEEE ICDM*, pp. 322–331, 2007.

Wang, X., Venable, J., LaPointe, P., Hutt, D. M., Koulov, A. V., Coppinger, J., Gurkan, C., Kellner, W., Matteson, J., Plutner, H., Riordan, J. R., Kelly, J. W., Yates, J. R. 3rd, and Balch, W. E. Hsp90 cochaperone aha1 downregulation rescues misfolding cftr in cystic fibrosis. *Cell*, 127(4) :803–815, 2006.

Watts, D. J. and Strogatz, S. H. Collective dynamics of /'small-world/' networks. *Nature*, 393 :440–442, 1998.

Weston, J., Chapelle, O., Elisseef, A., Schölkopf, B., and Vapnik, V. Kernel dependency estimation. In *Advances in Neural Information Processing Systems 15*, 2003.

Wu, X., Zhu, L., Guo, J., Zhang, D.-Y., and Lin, K. Prediction of yeast protein-protein interaction network : insights from the gene ontology and annotations. *Nucleic Acids Research*, 34(7) :2137–2150, 2006.

Xia, K., Dong, D., and Han, J.-D. Intnetdb v1.0 : an integrated protein-protein interaction network database generated by a probabilistic model. *BMC Bioinformatics*, 7(1) :508, 2006.

Xiong, X., Song, H., On, T., Lochovsky, L., Provart, N. J., and Parkinson, J. Phylopro : a web-based tool for the generation and visualization of phylogenetic profiles across eukarya. *Bioinformatics*, 27(6) :877–878, 2011.

Yamanishi, Y. and Vert, J.-P. Kernel matrix regression. In *Proceedings of the 12th International Conference on Applied Stochastic Models and Data Analysis (ASMDA)*, 2007.

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

Yamanishi, Y., Araki, M., Gutteridge, A., Honda, W., and Kanehisa, M. Prediction of drug-target interaction networks from the integration of chemical and genomic spaces. *Bioinformatics*, 24(13) :i232–i240, 2008.

Yellaboina, S., Tasneem, A., Zaykin, D. V., Raghavachari, B., and Jothi, R. Domine : A comprehensive collection of known and predicted domain-domain interactions. *Nucleic Acids Research*, 39 (Database Issue) :D730–D735, 2011.

Yip, K. Y. and Gerstein, M. Training set expansion : an approach to improving the reconstruction of biological networks from limited and uneven reliable interactions. *Bioinformatics*, 25(2) :243–250, 2009.

Yu, C.-Y., Chou, L.-C., and Chang, D. Predicting protein-protein interactions in unbalanced data using the primary structure of proteins. *BMC Bioinformatics*, 11(1) :167, 2010.

Zhang, L., Wong, S., King, O., and Roth, F. Predicting co-complexed protein pairs using genomic and proteomic data integration. *BMC Bioinformatics*, 5 :38–53, 2004.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16*, Cambridge, MA, 2004. MIT Press.

Zhu, X. and Ghahramani, Z. Learning from labeled and unlabeled data with label propagation. Technical report, Carnegie Mellon University, Pittsburgh, 2002.

Zhu, X., Ghahramani, Z., and Lafferty, J. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine learning (ICML)*, 2003.