THÈSE DE DOCTORAT
DE L'ECOLE NORMALE SUPÉRIEURE DE CACHAN

Présentée par

Emile RICHARD

pour obtenir le grade de

DOCTEUR DE L'ECOLE NORMALE SUPÉRIEURE DE CACHAN

Domaine : Mathématiques

Sujet de la thèse :

Regularization Methods for Prediction in Dynamic Graphs and e-Marketing Applications

Thèse présentée et soutenue à Cachan le 21/11/2012 devant le jury composé de :

Pr Francis BACH	Ecole Normale Supérieure & INRIA	Président
Pr Theodoros EVGENIOU	INSEAD	Directeur de thèse
Dr Stéphane GAIFFAS	Ecole Polytechnique	Examinateur
Pr Michael I. JORDAN	University of California, Berkeley	Examinateur
M. Thibaut MUNIER	1000mercis	Membre invité
Pr Massimiliano PONTIL	University College London	Rapporteur
Pr Nicolas VAYATIS	Ecole Normale Supérieure de Cachan	Directeur de thèse
Pr Jean-Philippe VERT	Ecole des Mines ParisTech	Rapporteur

Centre des Mathématiques et de Leurs Applications (CMLA)
Ecole Normale Supérieure de Cachan
61, avenue du Président Wilson,
94235 CACHAN CEDEX (France)

Contents

Acknowledgments

First of all I have to thank my uncle Mehrdad Shahshahani who contaminated me with mathematics. Then I would like to thank my advisors Theodoros Evgeniou and Nicolas Vayatis for their patience, insights, and for the profusion of their rich ideas which they generously shared with me. My colleagues Andreas Argyriou, Nicolas Baskiotis, Rodroigo Fernandez, Stéphane Gaïffas, Vianney Perchet and Pierre-André Savalle taught me greatly through many hours of discussion we had about graphs, learning, programming and statistics. I hope I will keep on working with them. Mehrdad, Nicolas V., Pierre-André and Stéphane read parts of this manuscript and I thank them for their feedbacks. It is an honor for me to have Massimiliano Pontil and Jean-Philippe Vert as reviewers, and I was very pleased by their encouraging reviews. My special thanks to Francis Bach, Stéphane Gaïffas and Michael Jordan to have accepted being committee members of this defense. I acknowledge ANRT for the CIFRE contract and financial support. I thank many times the company 1000mercis for providing the data, the problems and financial support, especially Yseulys Costes and Thibaut Munier for their trust in me and their taste for research.

Abstract

Predicting connections among objects, based either on a noisy observation or on a sequence of observations, is a problem of interest for numerous applications such as recommender systems for e-commerce and social networks, and also in system biology, for inferring interaction patterns among proteins. This work presents formulations of the graph prediction problem, in both dynamic and static scenarios, as regularization problems. In the static scenario we encode the mixture of two different kinds of structural assumptions in a convex penalty involving the ℓ_1 and the trace norm. In the dynamic setting we assume that certain graph features, such as the node degree, follow a vector autoregressive model and we propose to use this information to improve the accuracy of prediction. The solutions of the optimization problems are studied both from an algorithmic and statistical point of view. Empirical evidences on synthetic and real data are presented showing the benefit of using the suggested methods.

Overview and contributions

Industrial context and modeling choices

The current work is the result of close collaboration between the web-advertisement and marketing company 1000mercis and the the CMLA academic research laboratory at Ecole Normale Supérieure de Cachan. The collaboration was done in the context of a CIFRE (Conventions Industrielles de Formation par la REcherche) contract between the company and the laboratory. The research project was motivated by data mining challenges 1000mercis had faced. We provide algorithmic tools to solve some of these problems that are frequently met in the context of internet applications and also in other fields.

Predictive algorithms for graph data are needed for the conception of recommender systems, the valuation of network agents, prediction of trends and discovery of hidden patterns in complex real-world networks arising from various application domains such as internet data and gene-expression networks in biology. The purpose of this thesis is to introduce a methodology and related practical implementation techniques for exploiting the temporal and static patterns of complex networks in a predictive perspective. The need for robust and scalable recommender systems and valuation methodology for relational customer relationship management (CRM) data sets such as those arising from purchase histories in customer-to-customer (C-to-C) contexts (see [ZEPR11]), or social networks has motivated and defined the scope of the research project. The contribution of this work, in addition to the descriptive presentation and the analysis of the data provided by the company, is to formulate the problem of prediction in graph sequences as a *regularization* problem and explore the properties of the solutions both empirically and theoretically. In fact even though standard link prediction heuristics may perform well over some databases, the need for more accurate systems, robust measurements and functionality guarantees reveal the importance of developing more powerful tools. We chose to formulate the problem of prediction in graphs with the vocabulary of regularization methods. These methods have received a lot of interest from signal processing, imaging, computer vision, machine learning and high-dimensional statistics communities during the recent years. We can summarize the main motivation of the modeling choices made throughout this work as follows:

- **Complexity measure.** The crucial ingredient for formulating the prediction problem in graphs as a regularization problem is to define adequate complexity measures for graphs. A very basic and insightful observation is that link prediction heuristics can be expressed as spectral functions of the adjacency matrix that flatten the spectrum of it (see Figure 1.5, page 35). Hence spectral measures such as the rank or a convex surrogate, as the nuclear norm, seem to be promising candidates. These spectral measures are closely related to the community structures of the graphs. In fact block-diagonal and overlapping block-diagonal matrices have low rank and they belong to sets of adjacency matrices representing graphs containing sharp clusters or overlapping clusters. The low-rank prior and its convex surrogates [Jam87] (the trace norm [Bac08], the max norm [LRS$^+$10] for instance) have been studied by Srebro [Sre04] in the collaborative filtering context. We adapt them to the framework of highly clustered graph data where low-rank and sparsity effects are mixed.

- **Simultaneous prediction of local and global effects.** Another important direction of investigation is the multi-scale structure of the data: as in many complex, self-organized systems, global and local effects highly interact. In case of dynamic graphs, the idea is that global effects such as life cycles, trends or communities and local connections among

nodes of the networks are inseparable phenomena. The estimation method should therefore take into account the nature of mutual interactions among these objects.

- **Convex optimization tools.** It is a practical choice for guaranteeing the existence and possibly the uniqueness of the solution of the formulated problem to use a convex optimization framework. Such a framework allows in many cases to design algorithms converging to the true or approximate solution in polynomial time, even if the initial problem of interest is non-polynomial. In fact, we know from compressed sensing theory [CT05] that, in some regime, convex relaxation gives the same solution as the corresponding NP hard problem, leading to no loss of information. The very rich literature on optimization provides algorithms for a large body of convex functions that include our functionals. We will provide methodological tools for implementing efficient algorithms, for tuning the parameters, and will provide empirical evidence of the algorithms performance. In matrix factorization problems, the convex formulation leads to computationally intensive iterative algorithms because of the computation of a singular value decomposition at each iteration step. We will argue in a specific matrix factorization task how to replace the convex formulation with a partially convex but more scalable one, and will provide empirical evidence of success of the obtained method. Theoretical guarantees of convergence for these algorithms are missing and constitute a future direction of research.

- **Generative models.** We also introduce generative models of temporal and static graphs that mimic significant properties of the real-world data, and illustrate cases where all the assumptions required for the models to fit the data hold true. After arguing their interesting properties and motivating the modeling choices, we provide theoretical analysis of the predictive models over the synthetic generative graphs. In particular we prove oracle bounds for the regression problem subject to a mixed ℓ_1-trace norm penalty we designed and deduce related bounds for the prediction problem in an autoregressive framework. In terms of methodology, introducing such models allows one to study the impact of parameters on the model separately and to understand their interactions.

Contributions

The main contribution of this work is the formulations of the graph prediction problem in the static and dynamic scenarios as convex optimization problems. We provide theoretical and empirical evidence in support of this approach.

1. **Estimation of sparse low-rank matrices and graph denoising.** Previous works in high-dimensional statistics had tackled the problem of estimating a vector having a few nonzero elements. In the same spirit, the estimation of matrices that have low rank has been investigated in an undersampled setting. It is known that the estimation is inaccurate in the first case when *the few active variables are grouped* into sets of highly correlated variables. On the other hand, the standard matrix completion procedure fails when the *coefficients of the low-rank matrix are mostly equal to zero*. It turns out that in many applications the objects of interest are sparse and low-rank simultaneously. Such an object cannot be estimated using each of the existing methodologies alone, as it has the two *undesirable* properties. Namely, only a few nonzero coefficients appear in grouped structures. If the groups are known in advance recent works [BJMO11, BMAP12] have been devoted to the estimation of the coefficients. In many applications the groups themselves are unknown making the task even harder. Block-diagonal matrices are examples of such objects that naturally

arise from clustering or community detection applications. We develop a methodology to estimate, under several scenarios, such matrices. Denote by X a candidate adjacency matrix with coefficients $X_{i,j}$. We introduce a regularizer (see [RSV12]) composed by the sum of two well-studied terms as $X \mapsto \tau\|X\|_* + \gamma\|X\|_1$. We will argue that the mixed norm combines the benefits of each of the two norms and is suitable for estimating sparse low-rank matrices. This penalty can be written as $\tau \sum_i \sigma_i(X) + \gamma \sum_{i,j} |X_{i,j}|$ where $\sigma_i(X)$ represents the ith singular value of X or the square root of the ith largest eigenvalue of $X^\top X$. The set of sparse low-rank matrices obtained by minimizing objectives penalized by such a term turns out to contain matrices that are -up to permutation of rows and columns- block-diagonal and overlapping block-diagonal forms. These matrices can be interpreted as adjacency matrices of networks containing highly connected groups of nodes and therefore are of particular interest for prediction and denoising applications in graph data, and in covariance estimation. Assume we are given a loss function $\ell(\cdot, Y) : \mathbb{R}^{n\times n} \to \mathbb{R}$ that measures for instance the fit to the observed data Y. We use the latter norm in order to relax the hard optimization problem

$$\begin{cases} \min\limits_{X\in\mathcal{S}} \ \ell(X,Y) \\ \text{subject to } \ \|X\|_0 \le q \ \ \text{and} \ \ \text{rank}(X) \le r \end{cases}$$

to its convex surrogate

$$\widehat{X} = \arg\min_{X\in\mathcal{S}} \ \{\ell(X,Y) + \gamma\|X\|_1 + \tau\|X\|_*\} \ .$$

We will provide algorithms to minimize the objective over the convex set of admissible solutions $\mathcal{S}$ for different loss functions ℓ and also provide oracle bounds for the regression loss. In particular we study the problem that we call graph denoising where we consider an ℓ_0 loss $\ell(X,Y) = \|X - Y\|_0$. It consists of finding the fewest edges to flip (turn off if on, and on if off) making the new graph adjacency matrix low-rank and sparse.

We will provide algorithms to solve the convex surrogate of this problem that consists of minimizing the convex nonsmooth objective $X \mapsto \|X - A\|_1 + \tau\|X\|_* + \gamma\|X\|_1$.

2. **Prediction of graphs with autoregressive features.** We also study the graph prediction problem in a dynamic setting [RBEV10, RGV12]. In this case we need to assume that a set of graph descriptors such as degrees or any linear function of the graph evolve smoothly over time. The objective function we design for this problem has two variables: W and A. The variable W is the VAR parameters set that handles the prediction of the features by minimizing the least squares *fit to past data* term $W \mapsto \|\mathbf{X}_T - \mathbf{X}_{T-1}W\|_F^2$ where T is the number of observed snapshots and $\mathbf{X}_T, \mathbf{X}_{T-1}$ represent collections of observed features from time $t = 1, \cdots T$ and $t = 0, \cdots T-1$ respectively. The second variable A represents the adjacency matrix of the graph to be predicted. We argue that the prediction of A can be done through the prediction of the features of A that we compute through a mapping $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$. We refer to this mapping as the feature map. For link prediction heuristics, the *stability* of the distribution of some features explains the success of the methods. We try to generalize this mechanism by designing a methodology where the slow evolution of a set of well-chosen features makes their prediction easier than the prediction of each individual edge directly. Then estimating the graph given a linear observation of it is very similar to matrix completion, or regression with spectral penalty. We use the mixed ℓ_1-trace norm to this end. The objective can be written in the following

A_0 ,	A_1 ,	$\cdots$	A_T		$\widehat{A_{T+1}}$	Adjacency matrices observed $\in \mathbb{R}^{n\times n}$
$\downarrow_\omega$	$\downarrow_\omega$		$\downarrow_\omega$		$\uparrow$	
$\omega(A_0)$,	$\omega(A_1)$,	$\cdots$	$\omega(A_T)$	$\rightarrow_W$	$\widehat{\omega(A_{T+1})}$	Feature vector representations $\in \mathbb{R}^d$

Table 1: General scheme of the method suggested for prediction in dynamic graph sequences through a feature map ω.

form:

$$\mathcal{L}(A,W) \doteq \frac{1}{dT}\|\mathbf{X}_T - \mathbf{X}_{T-1}W\|_F^2 + \kappa\|W\|_1 + \frac{1}{d}\|\omega(A) - W^\top\omega(A_T)\|_2^2 + \tau\|A\|_* + \gamma\|A\|_1$$

This objective function contains a smoothly differentiable convex least squares term interpreted as a loss $(A,W) \mapsto \frac{1}{dT}\|\mathbf{X}_T - \mathbf{X}_{T-1}W\|_F^2 + \frac{1}{d}\|\omega(A) - W^\top\omega(A_T)\|_2^2$, and a penalty term acting on the joint variable (A,W). The penalty term acts differently on the A subset and W subset of variables. The penalty on W is a basic element-wise ℓ_1 term that encourages sparsity of the VAR matrix W. This penalty can be replaced by an ℓ_2 term if the sparsity assumption is not relevant for the domain. On A we incorporate a mixed ℓ_1 plus trace norm penalty encouraging simultaneously sparsity and low rank of A. It is motivated by the block-diagonal and overlapping block-diagonal matrices. Table 1 summarizes the methodology in a scheme where the symbols $\downarrow_\omega$ represent the feature extraction procedure through the map $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$. The prediction in the feature space is represented by $\rightarrow_W$, and is handled in practice by the least squares regression on W, and finally the symbol $\uparrow$ mapping the predicted feature vector $\widehat{\omega(A_{T+1})}$ to $\widehat{A_{T+1}}$ represents the inverse problem that is to be solved through the regression penalized by the mixed penalty. It is important to keep in mind that the mapping ω that is relevant for real world applications may be nonlinear but in the current work we only study linear approximations to such maps. For instance an interesting set of features of graphs is obtained by counting the number of length k ($k = 3, 4, \cdots$) cycles passing through each node, these numbers are obtained algebraically by taking the diagonal elements of A^k where A represents the adjacency matrix of the graph. These mappings are obviously nonlinear. We leave their studies for future work.

3. **Algorithmic and empirical evaluation of the models.** The algorithms introduced for solving the convex but nonsmooth optimization problems defined in both static and dynamic contexts arise from the family of forward-backward algorithms [CP11] and their generalizations. Specifically, minimizing an objective function containing a smoothly differentiable convex loss plus the mixed penalty $\tau\|X\|_* + \gamma\|X\|_1$ can be done through the generalized forward-backward algorithm [RFP11], and the same algorithmic scheme can be used in the dynamical context. A Douglas-Rachford scheme using a generalization of the soft-thresholding operator to piecewise affine functions handles the case where the loss function is the graph denoising loss $\ell : X \mapsto \|X - A\|_1$. The need for scalability and the high computational cost of the shrinkage operator through a soft-thresholded singular value decomposition suggests the use of the variational characterization of the trace norm $\|X\|_* = \frac{1}{2}\min_{UV^\top = X}\|U\|_F^2 + \|V\|_F^2$ in the objective function rather that its closed-form expression. Hence we also wrote alternate minimization algorithms for optimizing surrogate objectives. These algorithms are known to perform well in practice [Kor08], to be scalable and implementable in a parallel architecture [RR11] but the lack of convexity

makes their analysis more challenging. We used the MATLAB implementations of the algorithms for evaluating the suggested models on both real and synthetic data and report empirical results illustrating the comparative superiority of the suggested methods relative to the standard ones and also describing regimes of noise and regularity where accurate predictions can be expected as phase transition diagrams.

4. **Theoretical guarantees.** The formulation of the problems as convex optimization procedures allows one to benefit from the large body of work on the analysis of these classes of problems. In particular the oracle bounds that we were eager to obtain have been derived by using the tools introduced for studying the Lasso (ℓ_1 penalized least squares regression) and the trace-norm penalized regression. In our static setting we consider a *low dimensional* linear measurement $\omega(A_0)$ of the target matrix A_0 to be observed after corruption by noise: $y = \omega(A_0) + \epsilon$. Let $\widehat{A}$ denote the estimator obtained by minimizing the convex surrogate objective. In this setting these inequalities bound the prediction error of the estimator $\|\omega(\widehat{A}) - \omega(A_0)\|_2$ by the the infimum over the set of admissible solutions of the squared error plus a complexity term involving both the rank and sparsity index of the candidate. The inequality shows in particular that optimizing the convex surrogate leads us to the best solution, if we were able to afford the high cost of minimizing the NP-hard real problem containing a rank and a sparsity index constraint. Technically, instead of directly decomposing the noise into the estimator support and the orthogonal, we first split the noise into two parts and perform the projections on each part separately. The splitting constant $\alpha \in (0, 1)$ appears in the propositions' assumptions as a trade-off constant between the two tuning parameters τ and γ. Using this mechanism, all the inequalities obtained can be interpreted as smooth interpolation between the corresponding inequalities for the Lasso and for the trace-norm penalized matrix completion, and shows that our estimation procedure performs at least as well as the best of the two. We also introduce the restricted isometry constant, the cone of restriction and restricted eigenvalue constants that extend the existing definitions to our framework.

   In a nutshell, the oracle bounds we provide for the time-dependent setup hold the following message: the prediction error and the estimation error can be simultaneously bounded by the sum of three terms that involve homogeneously (a) the sparsity, (b) the rank of the adjacency matrix A_{T+1}, and (c) the sparsity of the VAR model matrix W_0. The tight bounds we obtain are similar to the bounds of the Lasso and are upper bounded by:

$$C_1\sqrt{\frac{\log d}{Td^2}}\|W_0\|_0 + C_2\sqrt{\frac{\log n}{d}}\|A_{T+1}\|_0 + C_3\sqrt{\frac{\log n}{d}}\,\mathrm{rank}\,A_{T+1} \quad .$$

   The positive constants C_1, C_2, C_3 are proportional to the noise level σ. The interplay between the rank and sparsity constraints on A_{T+1} are reflected in the observation that the values of C_2 and C_3 can be changed as long as their sum remains constant. The bounds obtained for regression with the mixed penalty are similar to the latter results for $C_1 = 0$.

Organization

This dissertation is organized as follows. The Chapter 1 is a general introduction. In Chapter 2 we introduce tools to explore dynamic graph data sets. We provide definitions of some topological features of graphs that allow one to measure quantities of interest and illustrate their distributions over several real data sets, discussing implications on various properties of the data we can read from the plots and data. We point out the connections between specific

distribution of graph features and heuristics approaches used for predicting links in networks. We also review standard random graph generative models before introducing some random graph models that reproduce the properties of real-world graphs we are interested in. In particular, these models allow us to test the performance of our algorithm in a framework where the working hypotheses are rigorously fulfilled. This is a standard way of validating an estimation methodology before testing the validity of the working assumptions. In Chapter 3 we introduce a new regularization tool for estimating sparse low-rank matrices. This regularizer is a norm formed by taking the positive linear combination of two norms, each being used separately in previous applications. The ℓ_1 norm has been used for estimating sparse vectors independently from the dependence structure among variables, and the trace norm, or the sum of singular values, is used for estimating low-rank matrices.

Finally in Chapter 4 we formulate the problem of prediction in a sequence of graphs as a regression problem with a mixed penalty term. We provide optimization algorithms for joint prediction and VAR estimation and prove theoretical bounds guaranteeing the quality of the estimator under natural assumptions on the noise process. We point out that several simplifications had to be made in the models we study in order to formulate it using the tools of convex optimization. This formulation has the benefit of relating the work to a large body of theoretical and numerical analysis. The current work should be seen as a first attempt to develop prediction models for dynamic graphs using convex estimation tools.

Chapter 1

Introduction

1.1 Motivations

1.1.1 Prediction for anticipation, discovery and serendipity

The methods and algorithms studied in the current thesis are devoted to explore some of the strategic research and development axes identified by the company 1000mercis that is our industrial partner. 1000mercis, a pioneer in interactive marketing and advertisement founded in Paris in 2000, provides acquisition and retention tools through different channels: emailing, mobile and viral games. The main challenges motivating our collaboration with 1000mercis can be summarized as follows.

1. **Reducing promotional pressure.** Emailing has no cost, therefore the immediate danger many emailing service providers face is to overwhelm consumers. The mission of a responsible company is to struggle against such abusive practices by trading the amount of advertisements sent with their quality. The obvious benefit of such a sustainable strategy is to avoid long-term fatigue of users and reducing churn rate, *i.e.* the ratio of users who unsubscribe from a promotional campaign.

2. **Serendipity.** Web 2.0 places the customer at the heart of its development. By providing socio-demographic as well as browsing and purchase data of users to the websites, many potential services involving personalized advertisement have been made possible. The recommender systems, that are one of the main application of the current work, aim precisely at using past users data in order to provide the users information about products they may like and do not know about.

3. **Optimize immediate efficiency.** The last but not the least point is the immediate benefit generated by using powerful tools of modern statistics. Some preliminary tests in 1000mercis had shown that the efficiency of advertisement campains can be improved by a factor of 5 by tuning the parameters of a basic algorithm.

The purpose of storing exponentially growing amounts of data is to use the past data in order to make better decisions. We argue that the key to the three before-mentionned challenges is to predict several types of phenomena by forecasting users activity levels and trends in dynamic relational databases.

- **Forecasting users activity levels and trends** one can adjust promotional pressure at individual and global levels, anticipate important variations in the supply-chain, and push fragile tendencies

- **Predicting users preferences** filters information for adapting the content of advertisement to individuals tastes and also to share similar users preferences making one discover new items.

- **Fitting predictive models to the market data** allows to build valuation tools at micro and macro levels, and hence attribute the promotional assets optimally.

In statistics and machine learning the graphs have classically been used for modeling dependence among variables. Numerous examples can be given from biology applications where the huge number of variables imposes to find sparse dependency patterns among them. The new type of problems involving these objects uses them in a very different fashion. In fact the growth of internet has produced considerable amounts of relational data (social networks, purchase histories, communication networks, hyperlinked websites) where the observations are modeled directly as graphs. Graphs are in this case not anymore artifacts used for better visualizing or understanding dependence structures among variables, but rather constitute the observation and the output variable of interest.

1.1.2 Growing graphs of the world wide web

The data produced nowadays by humans every year overpass by an order of magnitude the amount of data produced up to 2005. The cost of storing 1Tera-Byte of data in 1980 was $14M and it is not more than $30 in 2010. What *The Economist* calls the 'The Data Deluge' [The10] opens the door to a wide spectrum of blue oceans of scientific research and economic development. These new range of data, generated by various types of applications including governments records, scientific data, internet websites, raise many original challenges for statisticians, computer scientists and mathematicians. An important portion of the data generated by internet can be viewed as network data evolving over time. In fact they contain pairs of connected objects such as websites connected by hyperlinks, users linked to websites through theirs surfing and clicks, users associated to products through purchases or users tied through social networks. The link structure among these entities, and its evolution dynamics contains highly relevant information. The value of this information can be highlighted by noticing the success of those who were able to use it in a proper way. For instance

- **Google**'s pagerank [BP98] that is the basic ingredient of the search engine aims at quantifying the relevance of each website given the hyperlink structure of its surrounding.

- **Amazon** was the first to use purchase history as a mirror for predicting future purchases. It has used its valuable database for earning up to 35%of its turnover through the recommender engine [LSY03].

- **Facebook Open Graph** is a protocol[1] that aims at superimposing various internet graphs (mainly user-products and user-users) in order to use the social link structure for more efficient advertisement targeting.

The emergence of internet as a new communication medium has fundamentally changed the nature of advertisement by placing the user at the heart of the communication chain. For the advertiser, besides making communication almost cost-free and instantaneous, two major revolutionary assets of internet as a communication medium are the measurability of the users feedbacks to ads and the availability of inter-users communication traces. In fact, in comparison with the standard offline Business-to-Customer, the clicks or purchases initiated by an

[1] http://developers.facebook.com/docs/reference/api/

advertisement can be interpreted as the success of the ad or the promoted product and are a golden key for the marketers in their product design. On the other hand the growing popularity of user generated websites and social networks has made the management of viral marketing[2] possible. In fact the former non-measurable word-of-mouth advertisement has turned out to be observable and measurable at individual and aggregated levels, and given its tremendous efficiency, marketers run after tools allowing them to master related technologies. Note that despite being rich in content and widely available, the data collected from internet are not always smoothly useable raw materials for advertisers. In fact most declarative data as questionnaires, turn out to contain many missing and noisy values, and the spontaneous surfing and behavior data is high-dimensional, noisy and incomplete. A common point among many types of data collected from internet, such as purchase history, surfing data, communication, collaboration and social media data, are that they can be modeled as connections among users, or connections of users and items (webpages or products), or connections among objects. The technical application of the thesis is to suggest methodologies to handle these types of data for predictive goals.

The complexity and the novelty of graph data make their statistical analysis, engineering and retrieval non-trivial. As graph data can not be easily embedded onto a usual vector space, the standard tools of statistics and machine learning do not apply to them when one needs to achieve clustering, classification or prediction. The goal of the current thesis is to introduce a set of mathematical tools enabling the formulation of the problem of prediction in dynamic graphs such as those produced massively by web applications as a regularization problem, and suggest methods to solve them efficiently.

1.1.3 Graphs of multivariate data

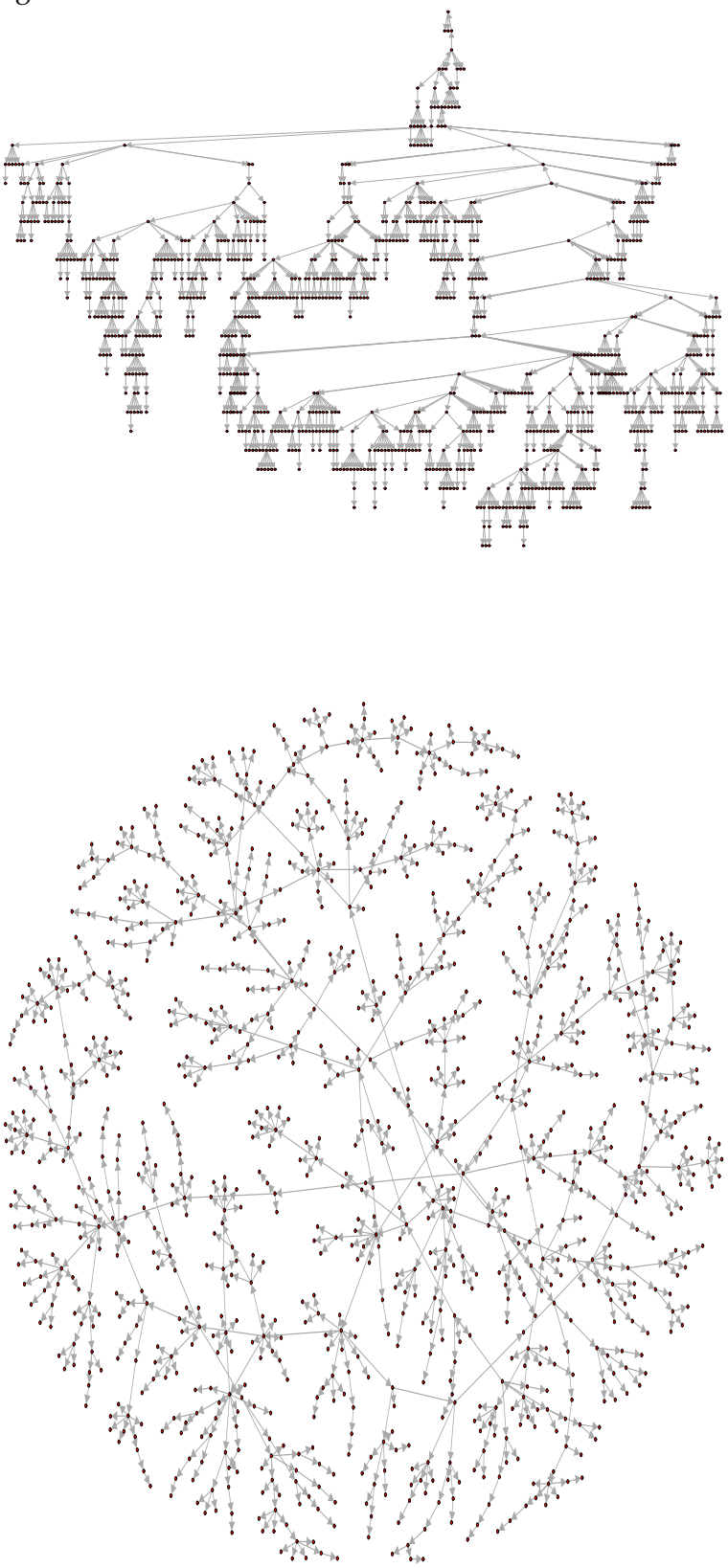
When observations are modeled as vector data, graphs are useful objects for describing high-dimensional distributions. In fact when the number of variables grows, not only the study of each individual variable is of interest, but in many cases the interaction structure among different variables turns out to be both informative and interesting by its own. Forecasting the behavior of systems with multiple responses has been a challenging problem in the context of many applications such as collaborative filtering, financial markets, or bioinformatics, where responses may be, respectively, movie ratings, stock prices, or activity of genes within a cell. Statistical modeling techniques have been widely applied for learning *multivariate time series* either in the multiple linear regression setting [BF97] or with autoregressive models [Tsa05]. More recently, kernel-based regularized methods have been developed for multitask learning [EMP05, APMY07, LPTvdG11]. These approaches share in common the use of the interactions between input variables to enhance prediction of every single output.

In some applications the interactions among variables constitute the main object of interest. For instance, a problem of major interest in functional genomics, systems biology and drug discovery is the estimation of regulatory networks in gene expression, and interaction networks among proteins that could explain genetic variability in cancer and other diseases (see [BO04, HRVSN10, JV08] for explanations and the figure 1.1.3 for illustration [3]). Nodes represent genes or proteins and edges interactions. The studies in these fields aim at discovering the mechanisms leading to diseases such as the development of cancer cells. Understanding the

[2]Viral marketing, also called viral advertisement or buzz marketing, is a marketing technique based on word-of-mouth type of exchanges among users, leading to a spread of information about a brand that is similar to virus propagation in a social network.

[3]Source : http://www.bioquicknews.com/ *Vast New Regulatory Network Discovered in Mammalian Cells*, 10/14/2011

Figure 1.1: These two representations of the same graph represent a communication network in a viral marketing situation.



functions of the central nervous system and the brain [VLEM01] also involves a large number of objects interacting together. Based on the observation of gene expressions, protein interactions, or cell activities statistical tools are designed to infer the correlation mechanisms, and at a higher level, measure causality among different expressions. For this goal methodologies involving Bayesian Networks [Jen02] and causality networks have been developed, both in static and dynamic [SKX09] cases. Bayesian Networks model the random variables as nodes of a graph that are connected through an edge if they are dependent. If no cycle appears in the graph of variables, then the joint density function of the d variables $x_1, \cdots, x_d$ can be written as

$$p(x_1, \cdots, x_d) = \prod_{(a,b) \in E} p(a|b)$$

where a and b are pairs of variables and E represents the *edge* set that contains all the pairwise connections. The undirected Bayesian Networks are also known as Markov Random Fields. They have the power of modeling cyclic dependency behavior and have been used in computer vision and image processing [KNG11]. Even though causality remains a very delicate notion to measure, concise definitions have attempted to recover this notion. *Granger causality* for instance is defined for a pair of temporal random variable (X, Y) as follows. If the prediction of Y given past values of both X and Y outperforms the prediction of Y given only its own past values in an autoregressive way, then X is said to be Granger-causal for Y. Networks of Granger Causality describe the paths and mechanisms leading to specific expressions in living metabolisms [SM10].

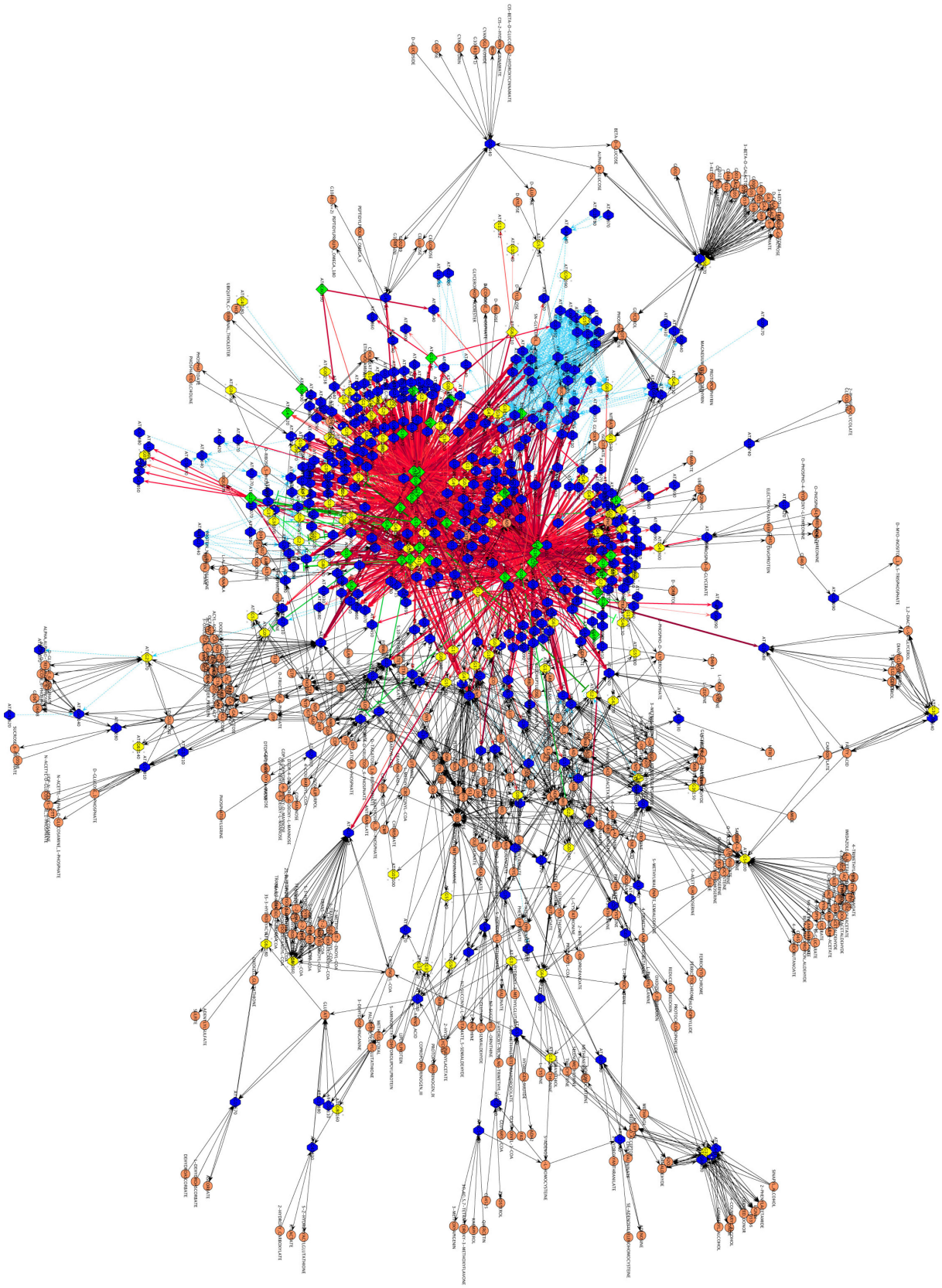
For studying interactions among different variables a common practice is to introduce matrices that have entries quantifying pairwise interactions. Two matrices are of particular interest in this context: the covariance matrix itself and its inverse that is also called the precision matrix. The correlation matrix can be seen as a dimensionless covariance matrix. The precision matrix entries are interpreted as partial covariance between two variables given all the others are set to fixed values. In case the data is *i.i.d.* multivariate gaussian, it turns out that the sparsity pattern of the precision matrix determines the pairwise dependence structure among random variables. Even though estimating the empirical covariance matrix of a sample is straightforward, when the number of variables is larger than the sample size or in presence of noise, the empirical sample covariance is a poor estimation of the true covariance matrix, and additional prior knowledge is needed to estimate the true covariance or precision matrix.
Efficient procedures developed in the context of sparse model estimation mostly rely on the use of ℓ_1-norm regularization [Tib96, BRT09]. Natural extensions include cases where subsets of related variables are known to be active simultaneously [YL06]. These methods are readily adapted to matrix valued data and have been applied to covariance estimation [EK08, BT10] and graphical model structure learning through the estimation of the inverse covariance by using the sample covariance as the input of an estimation procedure [BEGd07, FHT08].

1.2 Context

1.2.1 Users behavior modeling with redundant and missing data

Modeling users behavior is of significant interest for marketers. The use of electronic devices such as internet as advertisement platforms has drastically changed the marketers job by suggesting new tools to measure users sensitivities to different stimuli. In fact, as opposite to traditional advertisement campaigns, the feedbacks of the electronic media users to advertisement campaigns can now be stored at individual level and give access to measurements of

Figure 1.2: Example of regulatory network.



the success of each advertisement campaign at the individual levels. Furthermore the individual browsing and reaction behaviors of users are a key to design personalized advertisements. The key ingredient for designing individual level ads is to have access to users preference data, which are often incomplete or noisy. A common remedy to the missing data problem [Mar08] is to use the redundancies in users behaviors in order to infer the missing features. This controvertial but still rich method has a crucial benefit: it exploits the redundancy of the users behavior for considerably reducing the search space, or equivalently, multiplying data points. In mathematical terms, the manifold on which the users descriptor parameters lie has a much lower dimension than the number of apparent characteristics and the number of users. This implies that by using such prior knowledge on the domain one can infer the missing entities by detecting similarities among users and hence boost the quality of the database artificially. Techniques of the same spirit have extensively been used in multivariate regression [BF97], multitask learning [Arg06] and matrix completion [SRJ05].

1.2.2 Estimation for valuation

Customer valuation consists of scoring the clients and prospects by using their past actions for estimating their expected spendings, and allocate accordingly the promotional assets. In fact, given the unbalanced power-law distribution of consumer spendings, or the Pareto 80-20 rule that states that 80 % of the revenue is generated by 20% of the customers, it is crucial to detect or predict the small portion of the population that generates the majority of the company's income for privileging them.
Standard tools evaluate the customer value by simply averaging customers past activities or tracking the corresponding curves. In many modern data sets, the network structure makes the prediction of many actions much harder due to interactions among the users that strongly impact their behavior. For highlighting the relevance of this application of our work, we show how to valuate buyers and sellers in a C-to-C (costumer-to-costumer) market seen as a bipartite network, and refer to the paper [ZEPR11] for further details on the methodology. Here is a brief description of the data and the major elements of the estimation tools used in the paper.

Seller and buyers valuation in a C-to-C book market based on sales time series. We used approximately 6 years historic data from an online C-to-C bookstore. The market is formed by 1.3M buyers, 0.3M sellers, 1.5M products tracked over 322 weeks, representing an overall 8.6M transactions, having generated 80 M euros turnover (*i.e.* 19 M euros commision. Average weekly values :

Variable	Weekly Average
Sales Volume	3.13e+04
Commission	7.17e+04
Number of New Buyers	2.03e+03
Number of New Products	2.09e+03
Number of New Sellers	1.71e+02
Number of Returning Buyers	2.93e+04
Number of Returning Sellers	3.12e+04

We did not take the December data into account due to the particular purchase behavior of users that should be addressed independently from the rest of the year. We also did not

include the first week of january as it may have less than 7 days.

Following standard econometric methodology we suggested in [ZEPR11], we estimate using an ordinary least squares regression the matrix autoregressive coefficients of the multivariate time series formed by the number of new buyers, returning buyers, new sellers, returning sellers and the turnover. The obtained matrix is then used for computing the cumulative turnover, after a period of time, resulting from shocks on each actor (new/returning, buyer/seller). The interpretation is that the unit actor at time $t = 0$ is seen as a perturbation to the system, and one measures the cumulative value generated by a small perturbation on each of the actors volumes. This gives an estimate of the value of each of the users of the C-to-C market, and the cumulative impulse response of market users are interpreted as valuations of the *network effect*. We refer to [ZEPR11] and references therein for details on the methodology, and to 1.2.2 for the cumulative impulse response curves obtained in our experiments, and see table 1.1 for the table of results obtained in [ZEPR11]. We highlight that the purpose of the current thesis is to develop an estimation methodology taking into account both time series and network data in order to capture the effects of each on the other thanks to a joint estimation.

	Revenue contribution	Network effect: new sellers	Network effect: new buyers
New seller	11	8	11
New buyer	12	11	12
Returning seller	6	6	5
Returning buyer	1	1	1

Table 1.1: Revenue contribution and network effects measured by cumulative impulse responses at 95 % of the cumulative levels.

Formulation as a prediction problem in evolving graph data. It is very natural to use network data for valuation purposes because the revenue generated by an e-commerce network is a simple linear function of the market graph. Therefore we argue that fitting to the graph dynamics allows to extract wealth generating patterns in the graph and hence establish the proper value of each of the market actors.

For classical data sets available for econometric measurements only the time series of some variable are available and therefore vector autoregressive (VAR) models are the most appropriate tools. In the case of e-commerce data sets, individual level data are available, and therefore the estimation method can take them into account in order to get a more accurate estimation. Using the vocabulary of dynamic graph sequences we formulate the problem of consumer and contributor valuation for the C-to-C data presented below.
Consider a C-to-C market formed by Sellers (S), Products (P) and Buyers (B). We represent the market by a dynamic tripartite graph. The three sets of nodes are S,P and B. A transaction is modeled by a pair of links between a buyer and a product, and a seller to the latter product.

Let n_B, n_P, n_S design the number of buyers, products and seller, and $n = n_B + n_P + n_S$ be the overal number of actors in the market.

We can represent the transactions at each date by 3 incidence matrices B_P of size $n_B \times n_P$, B_S of size $n_B \times n_S$ and P_S of size $n_P \times n_S$. The (i, j) entry of a matrix X_Y is non-zero if a

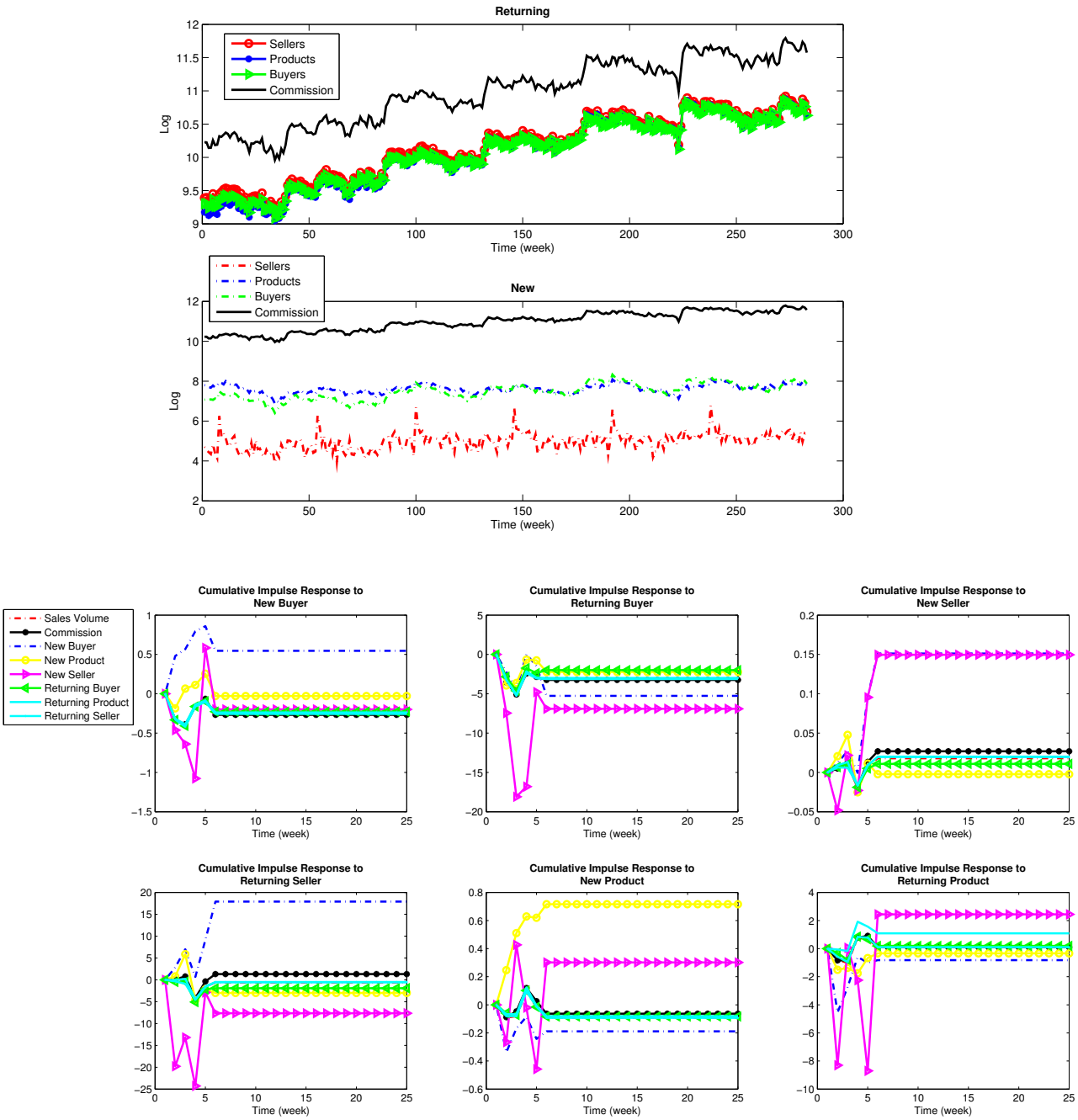


Figure 1.3: Top: time series of the number of different types of actors of a C-to-C market and the corresponding commission generated over weeks.. Bottom: impulse response curves of different times series of the marketplace.

transaction link relates the i-th user/item of set X to the j-th user/item of set Y. Following the application we may either take binary matrices or weighted adjacency matrices where the weight of an edge equals the amount of transaction or commission generated by the transaction. Note that $X_Y^\top = Y_X$. The n-by-n symmetric adjacency matrix

$$A_t = \begin{pmatrix} \mathbf{0} & B_P & B_S \\ B_P^\top & \mathbf{0} & P_S \\ B_S^\top & P_S^\top & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{0} & B_P & B_S \\ P_B & \mathbf{0} & P_S \\ S_B & S_P & \mathbf{0} \end{pmatrix}$$

summarizes all the transaction informations at time t.

Note that the following quantities are linear functions of the adjacency matrices :

1. The overall turnover / commission generated by the transactions
2. Number of Buyers/Products/Sellers participating to a transaction at time t
3. Number of Buyers/Products/Sellers participating to a transaction *up to* time t if we either consider cumulated adjacencies or a function of the set of matrices $A_1, \cdots, A_t$
4. Individual user activity / product popularity indicators : number or amount of transactions at time t

We can denote by $\omega(A_t) \in \mathbb{R}^d$ the vector containing the set of all such scalar variables of interest. Given the observation of $\omega(A_t)$, we aim at predicting $\omega(A_{t+1})$. For this end we use the linear autoregressive process

$$\omega(A_{t+1}) = f^*(\omega(A_t)) + \epsilon_{t+1} \tag{1.1}$$

where ϵ_{t+1} represents noise, that must be assumed to be i.i.d. and to have zero mean. The function f, taken linear at the first glance, can be estimated not only based on the set of vectors $(\omega(A_t))_t$, but also using the network information, namely the adjacency matrix sequence.

To summarize, in the setting of our problem we consider

- Adjacency matrices $A_t \in \mathbb{R}^{n\times n}$ representing a graph sequence for $t \in \{1, 2, ..., T\}$
- A feature map $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$. We refer to $\omega(A_t)$ as the feature vector of A_t.

and we want to find

- A matrix $A \in \mathbb{R}^{n\times n}$ representing a link discovery score, or the estimate of A_{T+1},
- A prediction function $\hat{f} : \mathbb{R}^d \to \mathbb{R}^d$ such that $\hat{f}(\omega(A_t)) \simeq \omega(A_{t+1})$

The methodology suggested allows to not only results in more accurate estimation of the system parameter but also allows to valuate customers at individual level.

One of the contribution of the current thesis is to suggest a methodology for simultaneously estimating f and predicting A_{T+1}. We just discussed why the accurate knowledge of the multivariate function f is of interest for valuation. In the next section we see why the prediction of A_{T+1} is of interest.

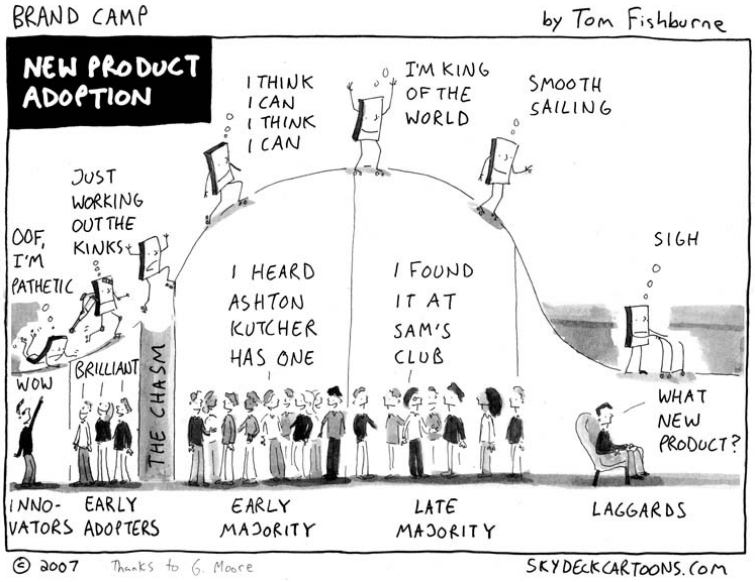


Figure 1.4: Cartoon of the innovation curve of a product.

1.2.3 Diffusion curves and network effect

The diffusion or penetration of a product refer to the different cycles a product goes across during its life, starting from the innovation, and ending by obsolescence. Studies in marketing [Bas63, Rog62, MMF90, SJT09, VFK04] have attempted to characterize the cycles of life of products. The parametric models introduced, despite being simplistic, provide interpretable schemes. Bass modeled the diffusion of innovations in networks of individual [Bas63] by stating that "the probability of adopting by those who have not yet adopted is a linear function of those who had previously adopted". In mathematical terms, these models that aim at also including the models of epidemic propagation in networks can be stated as the following discrete time equation, involving the fraction of infected population at time t:

$$F(t) = F(t-1) + p(1 - F(t-1)) + q(1 - F(t-1))F(t-1)$$

where p is the coefficient of innovation and q the coefficient of imitation. This model results in the following differential equation

$$\frac{\partial F}{\partial t} = (p + qF(t))(1 - F(t)) \ , \ \text{with} \ F(0) = 0$$

which admits a closed form solution:

$$F(t) = \frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p}e^{-(p+q)t}}$$

It can be easily seen that when $p > q$, $F(t)$ is a concave function of time, and if $p < q$, then the curve has an inflection point and is said to have an S-shape. In the *S-curves* sales volumes are said to follow over time, different phases of development, innovation, early adoption, early majority late majority and lagged sales are delimited by quantiles of a standard gaussian distribution. More recent works on the topic (*e.g.* [SJT09, VFK04]) have attempted to relax this very rigid parametric model of diffusion.

In marketing and economics, the notion of *network effect* is used to design the surmodular [Top98] increase of value of a network of people with the population size. In fact the potential number of connections among network nodes increasing quadratically, it has been observed that the responses of a network of individuals is richer than the sum of individual response. A more data-oriented study by Leskovec and coauthors [LKF05] rather suggests that in many real-world graphs the evolution of the number of edges is related to the number of nodes by a densification power law:

$$E(t) = N(t)^a$$

with a coefficient $a \in [0; 1]$. The network effect has been a leading concept in the study of *word of mouth* in marketing and recommendation[VYH08, LAB07] and also C-to-C two-sided markets[ZEPR11].

1.2.4 Design of recommender systems

One of the motivating applications of predictive algorithms on graph data is the design of recommender systems. Predicting univariate binary outputs in user behavior, such as clicks on ads, or opening advertisement emails can be handled by standard statistical tools, and for which measurement methodologies have been studied extensively in statistics and information retrieval literature. Note that the statistical tools needed for building recommender systems can not be found in standard statistics and data-mining methods. The reason is the particular characteristics of the data. In fact a naive approach would consist of treating the recommendation problem as a multi-class classification or multi-class ranking problem for each individual taken separately, the classes being the different items. In this case the number of classes being extremely high, and the number of observations very small, the problem is highly underdetermined and the algorithm will perform very poorly. A common remedy to deal with the very small number of observation by individual is to form groups of similar individuals and to complete unobserved features of each by the observed features of the others. The rigorous definition of functional spaces that handle the task of measuring similarity among partially observed data points and completing the missing data turns out to be very challenging. The objective of the current thesis is to attempt to answer some questions risen in this direction. Developed first by Greg Liden at Amazon in early 2000's [LSY03], recommender systems provide lists of items that users of an e-commerce website are likely to purchase, given the past collective purchase history. The philosophy behind such systems is that given the overwhelming amount of information available online, filtering the content based on the past users declaration, surfing and purchase history is a useful service. The goal being to emphasize for each user the very few items he is likely to purchase as a recommendation list rather than facing him to the whole items catalogue equally weighted. Even though personalized filtering of web information at user level sounds seducing, the use of purely predictive methodologies for such a goal has been subject to criticism [Bod08, FH09, Par11]. A main argument that predictive methods face is that by fitting to the tastes of a user the system does not make him discover new items. In other words, if one knows that a user is likely to buy an item, there is no need to recommend him that item. As one of the major benefits of recommender systems is serendipity (discovery of unexpected good things), fitting to the users tastes is indeed undesirable. We argue that once we acquired the technical tools allowing to fit the tastes of users, making them discover new items can be handled efficiently. Therefore we emphasize that our goal in the current thesis is to explore methods that allow to fit to the evolution of dynamic graphs. Following the applications in mind, it is up to the practitioner to use the suggested methodology in the convenient way.

1.3 Related works

The problem of predicting the evolution of a graph sequence is a hard problem. Similar problems have been widely studied in different fields of Physics, Mathematics, Computer Science, Signal Processing and Statistics. This is why before stating our approach to the problem we overview related ideas in various scientific disciplines.
The most classical related problem is possibly the n-body problem which aims at forecasting the trajectory of n planets. The planets interaction is subject to the gravitation whose closed-form expression is known. Nevertheless solving the system of these n dynamical equations turns out to be extremely challenging for $n \geq 3$. In our problem the inherent dimensionality of the problem is larger that 3, the closed form expression of the interaction is unknown and only a partially observed and noisy observation is available.

In the following we review some of the related problems and ideas. We quickly recall some vector and matrix operators needed for introducing the objects of interest. For a matrix $X = (X_{i,j})_{i,j}$, we set the following matrix norms:

$$\|X\|_1 \doteq \sum_{i,j} |X_{i,j}| \text{ and } \|X\|_* \doteq \sum_{i=1}^{\text{rank}(X)} \sigma_i,$$

where σ_i are the singular values of X and rank(X) is the rank of X. We consider the following setup. In the sequel, the projection of a matrix Z onto $\mathcal{S}$ is denoted by $P_{\mathcal{S}}(Z)$. The matrix $(M)_+$ is the componentwise positive part of the matrix M, and sgn(M) is the sign matrix associated to M with the convention sgn$(0) = 0$. The component-wise product of matrices is denoted by $\circ$. The class S_n^+ of matrices is the convex cone of positive semidefinite matrices in $\mathbb{R}^{n\times n}$. The sparsity index of M is $\|M\|_0 = |\{M_{i,j} \neq 0\}|$ and the Frobenius norm of a matrix M is defined by $\|M\|_F^2 = \sum_{i,j} M_{i,j}^2$. We also use $\|M\|_{\text{op}} = \sup_{x\,:\,\|x\|_2=1} \|Mx\|_2$ and $\|M\|_\infty = \max |M_{i,j}|$.

1.3.1 Underdetermined linear systems

Using first order local approximation of smooth functions, we can approximate a large class of problems by solving linear systems. In their general form, linear systems can be formulated as $Xw = y$ where $w \in \mathbb{R}^d$ is the unknown, $X \in \mathbb{R}^{N\times d}$ is the covariate matrix, N is the number of observations, and $y \in \mathbb{R}^N$ is the output vector. In such problems, the pair of observed input-output (X, y) are used to infer the vector w. In general $X \in \mathbb{R}^{N\times d}$ is not invertible. In case there are more observations than unknowns, $d < N$, the problem is said to be over-determined. We may therefore prefer solving the problem $X^\top Xw = X^\top y$ instead. When the number of observations is smaller than the number of variables $N < d$, the problem may have infinitely many solutions. Consequently, the optimization problem

$$\min_{w\in\mathbb{R}^d} \|Xw - y\|_2 \tag{1.2}$$

suffers from numerical instability of solutions. In 1943, for stabilizing the solutions, Tikhonov suggested to solve the following regularized problem :

$$\min_{w\in\mathbb{R}^p} \|Xw - y\|_2^2 + \|\Gamma w\|_2^2 \tag{1.3}$$

for a well-chosen matrix Γ. The particular case $\Gamma = \sqrt{\lambda} I_d$ is known as Ridge Regression [HTF01], and has the practical virtue of reducing noise and leading to smooth solutions. The choice of the parameter λ that trades the smoothness of the solution with fit to the data is

data-dependent. In general, when no prior knowledge on data distribution is available, data-driven calibration procedures (*e.g.* cross-validation) are often used. For further reading on the calibration methods we refer to [AB09]. In 1996, Tibshirani [Tib96] has introduced an ℓ_1-type penalized problem called the Lasso

$$\min_w \|w\|_1 \quad \text{subject to} \quad \|Xw - y\|_2 \leq \epsilon \tag{1.4}$$

which despite not having a closed form solution, has the advantage of leading to vector solutions with few nonzero components. Hence the Lasso selects variables at the same time as it stabilizes the solutions. In fact the ℓ_1-norm, defined as $\|w\|_1 = \sum_{i=1}^d |w_i|$ is the seen as the smallest convex surrogate of ℓ_0. The ℓ_0, sometimes called sparsity index of the vector, is defined by $\|x\|_0 = \sum_{i=1}^d 1\{x_i \neq 0\}$. It simply counts the number of non-zero elements of a vector. Minimizing the ℓ_1-norm has the virtue of leading to sparse solutions as minimizing ℓ_0 would. Compared to the exponential cost of minimizing the ℓ_0 directly through a combinatorial approach (see [Tro04]), minimizing the ℓ_1 norm has the advantage of defining the regularized problem as a convex optimization problem that can be solved in polynomial time.

Analysis of the Lasso

The analysis of the Lasso relies on the *Restricted Isometry Constants* first introduced by Candès and Tao [CT04]. These constants are useful for quantifying the quality of sparse recovery solutions obtained by ℓ_1 norm minimization [CW08]. The design of a low-dimensional feature map and the use of low-rank and sparse criteria enforced by the trace norm and ℓ_1-norm relates our approach to the field of *compressed sensing* (see [CT04]) that specifically focuses on techniques that *under-sample* high-dimensional signals and yet recover them accurately. In simple words, compressed sensing says if a signal is sparse in some basis, then the measurement basis must be as *incoherent* as possible to the reconstruction basis. For instance if a signal is decomposed with few coefficients in the Fourier basis, then sinusoids are not the most suitable measurement basis, but one should rather choose a measurement basis that is the most *de-correlated* with sinusoids. This notion of uncorrelation of basis is formalized by the definitions of incoherence and restricted isometry.

Definition 1 (Restricted isometry constant) *Given a matrix $X \in \mathbb{R}^{N\times d}$ with $N < d$, for any integer $s \geq 1$ we define the constant δ_s as the smalest constant such that for all s-sparse (having s nonzero coefficients) vectors $w \in \mathbb{R}^n$,*

$$(1-\delta_s)\|w\|_2^2 \leq \|Xw\|_2^2 \leq (1+\delta_s)\|w\|_2^2 \ . \tag{1.5}$$

We start by stating two theorems from [CW08]

Theorem 1 (Noiseless recovery) *Suppose $\delta_{2s} < \sqrt{2} - 1$. Let w_s be a copy of w_0 where all but the s-largest entries (in absolute value) of w_0 are set to 0. Let $\hat{w}$ be the minimizer of $\|w\|_1$ subject to $Xw = y$ where $y = Xw_0$. Then there are constants $C_0, C_1 > 0$ such that the two following hold true*

1. *Bound on the ℓ_1 error*
$$\|\hat{w} - w_0\|_1 \leq C_0\|w_s - w_0\|_1$$
2. *Bound on the ℓ_2 error*
$$\|\hat{w} - w_0\|_2 \leq \frac{C_1}{\sqrt{s}}\|w_s - w_0\|_1 \ .$$

Theorem 2 (Robust sparse recovery in presence of noise) *Suppose $\delta_{2s} < \sqrt{2}-1$ and $\|y - Xw_0\|_2 \leq \epsilon$. Let w_s be a copy of w_0 where all but the s-largest entries (in absolute value) of w_0 are set to 0. Let $\hat{w}$ be the minimizer of $\|w\|_1$ subject to $\|Xw - y\| \leq \epsilon$. Then there are constants $C_0, C_1 > 0$ such that*

$$\|\hat{w} - w_0\|_2 \leq \frac{C_0}{\sqrt{s}}\|w_s - w_0\|_1 + C_1\epsilon \ .$$

A similar but less restrictive property called *restricted eigenvalue* has been introduced in [BRT09]. Before stating the definition, we introduce some notations. For J a subset of $\{1, \cdots, d\}$ of cardinality $|J|$, and a vector $w \in \mathbb{R}^d$, we denote by w_J the vector of $\mathbb{R}^d$ that is a copy of w over the set J and the value of coordinates indexed by $k \notin J$ are set to zero. We denote by J^c the complimentary of J in $\{1, \cdots, d\}$, and introduce similarly w_{J^c} that is a copy of w over J^c and vanishes on J. The notation $\|\cdot\|_p$ is used to denote the ℓ_p norm.

Definition 2 (Cone of restriction) *For a positive integer s and a fixed real value $c_0 > 0$, we define the cone of restriction, as the set of vectors which values are concentrated on a subset of size at most s of the entries,*

$$\mathcal{C}(s, c_0) = \{w \in \mathbb{R}^d | \exists J \subset \{1, \cdots, d\}, |J| \leq s, \|w_{J^c}\|_1 \leq c_0\|w_J\|_1\}$$

We now turn to the definition of the restricted eigenvalue of a design matrix X, which imposes a less restrictive condition on the covariates.

Definition 3 (Restricted eigenvalue) *Let*

$$\kappa(s, c_0) \doteq \min_{w \in \mathcal{C}(s,c_0)} \frac{\|Xw\|_2}{\sqrt{N}\|w_{J_0}\|_2} \ .$$

If $\kappa(s, c_0) > 0$, then X verifies RE(s, c_0).

With this definition in mind we can state the following result.

Theorem 3 *Let w^* an s-sparse vector (having $s \geq 1$ nonzero elements), and $y = Xw^* + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2 I_N)$ zero mean gaussian noise with variance $\sigma^2 > 0$, $A > 2\sqrt{2}$ and $\gamma = A\sigma\sqrt{\frac{\log d}{N}}$.*

Define the Lasso estimator using the unconstrained formulation

$$\hat{w}(\gamma) \doteq \arg\min_{w \in \mathbb{R}^d} \frac{1}{N}\|y - Xw\|_2^2 + 2\gamma\|w\|_1 \ . \tag{1.6}$$

Then provided that $\frac{1}{N}X^\top X$ has diagonal entries equal to 1 and verifies RE$(s, 3)$, with probability at least $1 - d^{1-A^2/8}$, by letting $\kappa(s) = \kappa(s, 3)$, we have

1. *Bound on the ℓ_1 error of the solution*

$$\|\hat{w}(\gamma) - w^*\|_1 \leq \frac{16A}{\kappa(s)^2}\sigma s\frac{\log d}{N}$$

2. *Bound on the ℓ_2 error of the covariate times the solution*

$$\|X(\hat{w}(\gamma) - w^*)\|_2^2 \leq \frac{16A}{\kappa(s)^2}\sigma^2 s \log d$$

3. *Bound on the sparsity of the solution*

$$\|\hat{w}(\gamma)\|_0 \leq \frac{64\|\frac{1}{N}X^\top X\|_{op}}{\kappa(s)^2}s \ .$$

We refer the interested reader to [BRT09] for proofs, further reading and also to [LPTvdG11] for extensions to group sparsity and multitask learning.

Practical procedures for solving penalized linear systems

The Tikhonov and ridge regression have closed form solutions

$$\hat{w} = (X^\top X + \Gamma^\top \Gamma)^{-1} X^\top y$$

if $X^\top X + \Gamma^\top \Gamma \in \mathbb{R}^{d \times d}$ is invertible, and this is the case for usual regularizer choices. Therefore, the algorithmic aspects of such estimation procedures rely on the linear system solver.

For minimizing the least squares problem penalized with an ℓ_1 term, note that the objective

$$\mathcal{J}(w) \doteq \frac{1}{N}\|y - Xw\|_2^2 + 2\gamma\|w\|_1$$

is convex, ensuring thus the existence of a minimizer. The nondifferentiable ℓ_1 term makes the practical minimization of $\mathcal{J}$ intractable with standard gradient descent. Different strategies have been considered (see [SFR09] for a survey) for minimizing such objectives. A basic method [FHT10] consists in cyclicly picking coordinates and minimizing the objective with respect to each coordinate. The benefit of this method is that the coordinate-wise updates are exact and the closed-form expression is easy and fast to compute. Despite computational efficiency at each step, when variables are highly correlated, the number of iterations needed to converge to the solution increases dramatically. Other methods start by smoothing the ℓ_1 norm and consider the minimization of regularizers as

$$\sum_{i=1}^{d} \sqrt{w_i^2 + \epsilon}$$

or

$$\sum_{i=1}^{d} \frac{w_i^2}{\alpha_i} + \alpha_i$$

that are respectively smooth approximation and upper bounds of $\|w\|_1$ for $\epsilon, \alpha_i > 0$.

Proximal algorithms [BT09, CP11] have been designed for solving precisely such problems where the objective

$$\mathcal{J} = \ell + \gamma R$$

can be written as the sum of

1. a Lipschitz differentiable and convex term $\ell(w)$
2. a convex term $R(w)$ who is not differentiable, but which proximal is fast to compute.

In the case of the ℓ_1 norm, the proximal operator is given by

$$\mathrm{prox}_{\theta\|\cdot\|_1}(x) \doteq \arg\min_z \theta\|z\|_1 + \frac{1}{2}\|z - x\|_2^2 = \Big(\mathrm{sign}(x_i)\max(|x_i| - \theta, 0)\Big)_i .$$

It has been proven [BT09] that the sequence

$$w_{k+1} = \mathrm{prox}_{\theta\gamma\|\cdot\|_1}(w_k - \theta\nabla f(w_k))$$

converges after $O(\frac{1}{\epsilon})$ steps to a ball of radius ϵ of the minimizer of $\mathcal{J}$. The step size θ is usually taken of the order of magnitude of the inverse of the Lipschitz constant of $\nabla\ell$. An accelerated algorithm (FISTA) that reaches the optimal convergence rate [Nes05] of $O(\frac{1}{\sqrt{\epsilon}})$ can be written using an auxiliary sequence [BT09, Tse08].
The intuition behind the design of the latter algorithms relies on

1. the linear expansion of f around the point w_k
2. the quadratic term $\frac{L}{2}\|w - w_k\|_2^2$ that controls the closeness of the next step point w_{k+1} to the latter w_k:

$$\begin{aligned}
\mathcal{J}(w) \simeq\ & \ell(w_k) + \nabla\ell(w_k)^\top(w - w_k) + \gamma R(w) + \frac{L}{2}\|w - w_k\|_2^2 \\
=\ & L\left\{\frac{1}{2}\|(w - w_k) + \frac{1}{L}\nabla\ell(w_k)\|_2^2 - \frac{1}{2L^2}\|\nabla f(w_k)\|_2^2 + \frac{1}{L}\ell(w_k) + \frac{\gamma}{L}R(w)\right\} \\
=\ & L\left\{\frac{1}{2}\|w - (w_k - \frac{1}{L}\nabla\ell(w_k))\|_2^2 + \frac{\gamma}{L}R(w)\right\} + \text{constant}
\end{aligned}$$

It follows that the point $w_{k+1} = \mathrm{prox}_{\frac{\gamma}{L}R}(w_k - \frac{1}{L}\nabla\ell(w_k))$ is a fair approximation to the minimizer of $\mathcal{J}$ around w_k. The detailed analysis and extensions can be found in [Tse08].

1.3.2 Matrix completion

The problem of matrix completion became popular thanks to challenges suggested by DVD retailers MovieLens and Netflix. It has been extensively studied on both theoretical and algorithmic aspects [SRJ05, CT09, Gro11, RR11, KLT11]. The matrix completion setting assumes that some ratings (1 up to 5 stars) are given by some users on some movies. The goal is to fill in the missing entries. The application of this problem is to recommend to each user a movie he does not know (did not yet rate) and that he is likely to give full score score (high predicted score).
Formally, the problem can be formulated by seeing the rating process as linear measurements over the hidden $users \times products$ rating matrix. Let $\langle A, B\rangle = \mathrm{Tr}(A^\top B)$ denote the inner product for matrices. If $E_{i,j}$ denotes the matrix having all zeros but a one at its (i,j)-th position, the function $A \mapsto \langle A, E_{i,j}\rangle$ is linear and measures the (i,j)-th entry of A. Assume $\Omega \subset \{1, \cdots, \#users\} \times \{1, \cdots, \#movies\}$ denotes the set of indices of the available ratings, *i.e.* if $(i,j) \in \Omega$, then the rating of user i on movie j is available. For simplicity, take

$$n = \max(\#users, \#movies), \quad d = |\Omega| \ ,$$

and denote by $\Omega = \{(i_1, j_1), (i_2, j_2), \cdots (i_d, j_d)\}$ the elements of Ω. Define the linear mapping

$$\begin{aligned}
\omega\ :\ & \mathbb{R}^{n\times n} \to \mathbb{R}^d \\
& X \mapsto \Big(\langle A, E_{i_1,j_1}\rangle\ ,\ \cdots\ ,\ \langle A, E_{i_d,j_d}\rangle\Big)\ .
\end{aligned}$$

Matrix completion aims at recovering a matrix A_0 given the observation $y = \omega(A_0)$. In particular, the theoretical studies explain for which matrices A_0 and which sampling sets Ω one can recover the matrix A. Statistical approaches study the noisy case when the observation is corrupted by an additive noise with mean zero ξ:

$$y = \omega(A_0) + \xi\ .$$

Exact and robust recovery of low-rank matrices from few observations

Theoreticians [Bac08, CT09, Gro11, KLT11] have proved using tools comparable to those used in ℓ_1-recovery literature, that if the rank of A is low and the sampling uniform at random, then

minimizing convex functionals as

$$\frac{1}{d}\|\omega(A) - y\|_2^2 + \tau\|A\|_* \tag{1.7}$$

leads to consistent and optimal solutions for the matrix completion problem. To see the analogy with the sparse recovery, one can think of the set of singular values of a matrix as a vector whose sparsity is equivalent to the low-rank-ness of the matrix. This analogy between vectors and matrices is formalized using convex calculus (subgradient computations) in [Lew95].

Here we state the most recent results for the exact recovery and noisy matrix completion.

Exact recovery We consider the following problem

$$\min_A \|A\|_* \quad \text{subject to} \quad \omega(A) = \omega(A_0) \tag{1.8}$$

and wish to establish under which conditions its solution is unique and equal to A_0.

Let us recall an easy linear lower bound on the number of required observations first. Note that even if the set of rank-r matrices of size $n \times n$ is constituted of matrices having n^2 coefficients, the low-rank constraint reduces the degree of freedom of the set. The number of free parameters of the set of $n \times n$ rank r matrices is bounded by

$$\underbrace{(n-1) + (n-2) + \cdots + (n-r)}_{\text{left singular vectors}} + \underbrace{(n-1) + (n-2) + \cdots + (n-r)}_{\text{right s.vectors}} + \underbrace{r}_{\text{s. values}}$$

$$= r(2n - r) \leq n^2$$

with equality only for full rank $r = n$. Therefore the minimum number of observation required to complete the matrix is at least $r(2n - r)$.

In order to establish sharp lower bounds on the minimum number of observations, note that a matrix with a single constant block has low rank but can hardly be recovered from entry-wise measurements. To formalize this intuition, Candès and Recht have introduced the *incoherence* of a matrix with the canonical basis, that can be extended [Gro11] to any basis, which measures the discrepancy of the two bases and is related to the hardness of completion subject to linear observations. This notion has been defined first by Candès and Recht [CCSA08] and was refined by Gross [Gro11]. Before introducing the definition, let $A = U\Sigma V^\top$ be the SVD of A where $U, V \in \mathbb{R}^{n\times r}$ are orthogonal and $\Sigma = \mathrm{diag}(\sigma_1, \cdots, \sigma_r)$. Let $U^\perp$ and $V^\perp$ matrices of size $n \times (n - r)$ ortho-normally completing the bases of U and V, and define

$$P_{U^\perp} = U^\perp {U^\perp}^\top$$

$$P_{V^\perp} = V^\perp {V^\perp}^\top$$

and also the orthogonal projections

$$\mathcal{P}_A(B) = B - P_{U^\perp} B P_{V^\perp}$$

and

$$\mathcal{P}_A^\perp(B) = P_{U^\perp} B P_{V^\perp}$$

Definition 4 (Coherence of a subspace) *Let U be a subspace of $\mathbb{R}^n$ of dimension r and P_U be the orthogonal projection onto U. Then the coherence of U (vis-à-vis the standard basis (e_i)) is defined to be*

$$\mu(U) \doteq \frac{n}{r} \max_{1 \leq i \leq n} \|P_U e_i\|^2$$

We now state the theorem

Theorem 4 (Exact matrix completion [CCSA08]) *Let $A = \sum_{m=1}^{r} \sigma_m u_m v_m^\top$ be the SVD of A. Let $\widehat{U}$ and $\widehat{V}$ denote respectively spaces respectively spanned by left and right singular vectors. Assume the two following assumptions hold :*

- ***A0*** *The subspace coherences obey $\max(\mu(\widehat{U}), \mu(\widehat{V})) \leq \mu_0$.*
- ***A1*** *The maximum entry of $\sum_{m=1}^{r} u_m v_m^\top$ is bounded in absolute value by $\mu_0 \frac{r}{n}$.*

Then the exact recovery of A given the observation of d entries selected uniformly at random is possible if

$$d > Crn \log^2 n \max(\mu_1^2, \mu_0^{1/2}\mu_1, \mu_0 n^{1/4}) \ . \tag{1.9}$$

The latter results have been improved by David Gross, who generalizes the low-rank matrix completion framework to low-rank matrix recovery from few linear observations in any basis. We first recall the definition

Definition 5 (Coherence of a matrix) *Let $(\epsilon_i)_{i=1}^{n^2}$ be an orthonormal basis of $\mathbb{R}^{n \times n}$, let $A \in \mathbb{R}^{n \times n}$ be a symmetric matrix with eigenvalue decomposition $\sum_{m=1}^{r} \sigma_m u_m u_m^\top$, and let $\mathcal{P}_T$ denote the projection onto the span of its eigenvectors. The matrix A has incoherence μ with respect to the basis $(\epsilon_i)_{i=1}^{n^2}$ if either*

$$\max_i \left(\|\epsilon_i\|_{op}\right)^2 \leq \frac{\mu}{n}$$

or the two estimates

$$\max_i \|P_U \epsilon_i\| \leq 2\mu \frac{r}{n}$$

$$\max_i \langle \epsilon_i, \sum_{m=1}^{r} \operatorname{sign} \sigma_i u_m u_m^\top \rangle \leq \mu \frac{r}{n^2}$$

hold.

A result due to Gross [Gro11] improves the dependency in μ of the bound :

Theorem 5 (Exact matrix recovery in any basis [Gro11]) *Let $\omega : A \mapsto (\langle \epsilon_{i_1}, A \rangle, \cdots, \langle \epsilon_{i_d}, A \rangle)$ where the set $\Omega = (i_1, \cdots i_d)$ is chosen uniformly at random in $\{1, \cdots, n^2\}$, then the solution to the exact recovery problem is unique and equal to A_0 with probability at least $1 - n^{-\beta}$ if*

$$d \geq O(nr\nu(1+\beta)\log^2 n) \ .$$

Trace-norm regularized regression. Consider the linear mapping

$$\omega : A \mapsto (\langle A, X_1\rangle, \cdots, \langle A, X_d\rangle)$$

for a set of random matrices X_i. Assume a corrupted observation $y = \omega(A_0) + \xi$ is available, ξ denoting a zero-mean noise. Let the solution of the unconstrained optimization problem

$$\min_A \frac{1}{d}\left(\sum_{i=1}^{d} \mathbb{E}\langle A, X_i\rangle^2\right) - \frac{2}{d}\sum_{i=1}^{d} y_i\langle A, X_i\rangle + \tau\|A\|_*$$

be denoted by $\hat{S}(\tau)$, where the expectation is taken over the distribution of X_is. The following result is proven in [KLT11]

Theorem 6 (Trace-norm regression) *Assume that there exist a constant $\mu > 0$ such that*

$$\mathbb{E}\|\omega(A)\|_2^2 \geq \mu^{-2}\|A\|_F^2 \ .$$

If $\tau > 2\|\frac{1}{d}\sum_{i=1}^{d}(y_iX_i - \mathbb{E}y_iX_i)\|_\infty$, then

$$\|\hat{A}(\tau) - A_0\|_F^2 \leq \min\left\{2\tau\|A_0\|_*, \left(\frac{1+\sqrt{2}}{2}\right)^2 \tau^2\ \mu^2 \mathrm{rank}(A_0)\right\} \tag{1.10}$$

The expectation being taken over the law of X_is.

In order to refine the last results, we need the two following definitions that are similar to the restricted cone and restricted eigenvalue property defined for the ℓ_1-penalized regression.

Definition 6 (Cone of restriction) *For a matrix $A \in \mathbb{R}^{n\times n}$ of rank r, we define the cone of restriction as the set of matrices that projection onto the singular spaces of A dominate the orthogonal projection, or the set of matrices nearly aligned with the row and column spaces of A. For $c_0 > 0$ let*

$$\mathcal{C}(A, c_0) \doteq \{B \in \mathbb{R}^{n\times n} \ \mid\ \|\mathcal{P}_A^\perp(B)\|_* \leq c_0\|\mathcal{P}_A(B)\|_*\}$$

We can now define the constant $\kappa_{c_0}(A)$ for any matrix A by

Definition 7 (Restricted eigenvalue constant κ_{c_0}) *Let*

$$\kappa(A, c_0) \doteq \inf_{B\in\mathcal{C}(A,c_0)\setminus\{0\}} \frac{\sqrt{\frac{1}{d}\mathbb{E}\|\omega(B)\|_2^2}}{\|\mathcal{P}_A(B)\|_F} \ .$$

Koltchinskii *et al.*[KLT11] have proved the following theorem.

Theorem 7 (Trace-norm regression) *If $\tau > 3\|\frac{1}{d}\sum_{i=1}^{d}(y_iX_i - \mathbb{E}y_iX_i)\|_\infty$, then*

$$\frac{1}{d}\mathbb{E}\|\omega(\hat{A}(\tau) - A_0)\|_F^2 \leq \inf_A \frac{1}{d}\left\{\mathbb{E}\|\omega(A - A_0)\|_F^2 + \frac{\tau^2}{\kappa(A, c_0)}\ \mathrm{rank}(A)\right\} \tag{1.11}$$

Proximal gradient descent for regression with trace norm penalty

Proximal gradient descent algorithms [BT09] can be used for matrix completion. The proximal operator associated with the trace norm, also called the shrinkage operator [CCSA08] is given by:

$$\mathcal{D}_\tau(B) \doteq \arg\min_A \frac{1}{2}\|A - B\|_F^2 + \tau\|A\|_* \ .$$

The practical computation of $\mathcal{D}_\tau$ requires the singular value decomposition of

$$B = U \operatorname{diag}(\sigma_1, \cdots, \sigma_n) V^\top \ ,$$

and is equal to

$$\mathcal{D}_\tau(B) = U \operatorname{diag}\Big(\max(\sigma_i - \tau, 0)\Big)_i V^\top \ .$$

To optimize

$$\ell(A, B) + \tau\|A\|_*$$

where ℓ is a Lipschitz differentiable function, the sequence suggested by the proximal iterative gradient descent follows the update rule

$$S_{k+1} = \mathcal{D}_{\frac{\tau}{L}}\left(S_k - \frac{1}{L}\nabla\ell(S_k)\right)$$

where L is the Lipschitz constant of ℓ. The corresponding accelerated algorithms exist. Although the memory requirement of FISTA may be a limitation in case of large values of n.

1.3.3 Link prediction heuristics

The problem of predicting missing links of a partially observed graph has a particular interest in many applications such as predicting hyperlinks of webpages [TWAK03], finding protein-protein interactions [KKY+09], studying social networks [LNK07a], as well as collaborative filtering and recommendations [HKV08]. The study of the linkage structure of social networks dates back at least to Katz [Kat53]. More recently, Liben-Nowell and Kleinberg [LNK07a] have popularized the problem of link prediction by comparing several classical methods on a scientific collaboration network. More recently, several works have built Bayesian and regularization-based types of algorithms and methods for dealing with the link prediction problem and variants [RBEV10, KX11, SM06, SSG07, SCM10a, MGJ09].

The link prediction problem has been studied in two different perspectives. The *static* version considers that some of the edges of a graph are hidden and one wants to reconstruct the partially observed graph. The *temporal* or *dynamic* version assumes that the edges appear over time on some fixed set of nodes, and the order of arrival of the edges is relevant. Therefore predicting the *future* edges requires to extract some temporal patterns that guide the apparition of the edges over time. Such mechanisms often comprise diffusion-like processes on graphs that are believed to handle specific temporal patterns observed in networks [LKF05].
The static link prediction heuristics have been studied and compared by Liben-Nowell and Kleinberg [LNK07a]. We review some of the heuristics compared by them as well as more recent developments in this direction. Roughly the different algorithms can be grouped using the type of graph features they use:

1. *Degree-based link prediction* only relies on the distribution of node degrees, and predicts creation of links among high degree nodes.

2. *Path-based link prediction* uses the distance among nodes to predict connections among close nodes. Despite nice algorithmic properties, these algorithms are not flexible enough for reaching optimal performances.

3. *Factorization based link prediction* builds over the assumption that a small number of latent features describe the nodes. The nodes that are close in the space of latent factors are likely to be connected.

Degree-based link prediction

The *Preferential Attachment Model* relies on the observation that degrees in social networks follow a power-law distributions. Compared to the Poisson distribution of degrees in Erdos-Renyi random graphs, unbalances power-law distributions give higher importance to high degree nodes. In the context of recommendation, such an approach consists in recommending the most popular items to all users, or to all users of an identified cluster. This algorithm is inspired by the following graph generation process that produces graphs with power law degree distribution having a pareto coefficient $\beta \in [2, 3]$.

Definition 8 (Preferential attachment model [AB02]) *Starts from an initial node, and then follows iteratively two types of steps :*
*1-*Vertex step-*with probability p, add a new node, and connect it to a vertex with probability proportional to degrees*
*2-*Edge step-*with probability $1 - p$, connect two nodes chosen with probability proportional to their degrees*

The preferential attachment model is a recursive random graph generation process. This is why given some observed graph, it suggests edges to add to a the graph. Note that even though this simple model produces graphs with the desired degree distribution, neither the "small world" [AB02] effect, nor the high clustering coefficient property of real world graphs are reproduced.

In terms of recommender systems, applying such a model to an e-commerce market history -seen as a bipartite graph- is too simplistic. The items degrees in the bipartite graph of a market are in fact nothing but the sales volumes of the items. As a consequence, the preferential attachment model used in a recommendation contexts would promote best-selling items (high degree nodes) to all the users, which is very naive. For instance the movie market is composed of dramas, horrors, science fictions *etc.* and recommending a drama best-seller to science-fiction fanatics is a poor recommendation.

The next algorithms presented involve features of higher order than the degree. Nearest-Neighbors, Random-Walks and Katz approaches rely on the distribution of length of paths in the graph and emphasize the probability of establishment of links among nodes that are close to each other but are not connected yet.

Path-based link prediction

"People who bought this book also bought that one" is the sentence used by Amazon to introduce the recommended items to users. Such models link vertices that are reachable through short paths on the graph. Their success is explained by the high frequency of short paths and high clustering coefficients in many social and market networks. Algebraically, computing such paths requires to compute powers of adjacency matrices. Matrix multiplication using a naive algorithm has a complexity rising with n^3, the fastest known algorithm for dense matrices runs in $O(n^{2.38})$. For sparse matrices such as most adjacency matrices, the complexity is independent of the size of matrices and scales linearly with their respective number of non-zero elements, and depends also on their sparsity patterns. Here we review some basic facts about walks on graphs and the corresponding matrices.

Proposition 1 (Powers of the Adjacency Matrix) *If A is the adjacency matrix of a graph $\mathcal{G}$, then for all integer k, and all indices i and j, the (i, j) entry of A^k, $(A^k)_{i,j}$ equals the number of walks of length k joining the vertex i to j .*

The score we refer to as Katz [Kat53], first normalizes the adjacency matrix by the diagonal matrix formed by inverse degrees of nodes

$$W = \mathrm{diag}(1/d_1, \cdots, 1/d_n)A \ ,$$

and then sums powers of this stochastic matrix with decreasing exponential weights, ensured by $0 < \beta < 1$:

$$\sum_{k=1}^{\infty} \beta^k W^k = \beta W (I_n - \beta W)^{-1} \ .$$

Factor based link prediction

Modeling graphs as strong connections among points in the factor space having close latent factors has been considered in building geometric random graph models, but also in link prediction and link analysis [Hof09, SCM10a]. Most of these approaches build on Bayesian models for the factors, especially when dealing with temporal trends in the link formation and factor evolution [SM06, SSG07, YCY+07, FAD+11, VAHS11].

	Link prediction	Matrix completion
Observed entries	no infomation (all equal 1)	**relevant**
Sparsity pattern	**relevant**	no infomation (uniformly random)

Table 1.2: Comparison of the information sources in link prediction and matrix completion problems.

Given a loss function $\ell(A, B)$ measuring the fit of the observed adjacency matrix B to the score matrix $A \in \mathbb{R}^{n\times n}$, the link prediction problem subject to low-dimensional factor assumption can be formulated as the minimization of $\ell(UV^\top, B)$ for $U, V \in \mathbb{R}^{n\times k}$ where k is the number of factors. Regularized algorithms of this type have been developed for collaborative filtering tasks [SRJ05, HKV08]. They give iterative solutions by alternately minimizing the following objective in the two variables U and V:

$$\ell(UV^\top, B) + \lambda(\|U\|_F^2 + \|V\|_F^2)$$

despite efficiency and scalability [RR11], these approaches aim at minimize non-convex objectives and therefore the theoretical study of these algorithms is quite poor. The use of the singular value decomposition in the factorization of the matrix leads to well-posed formulations.
Given a fixed target rank k one can compute the rank k matrix the closest to A in Frobenius norm by first computing A's SVD $A = U \operatorname{diag}(\sigma_i) V^\top$ and then setting to 0 all but the first k singular values of A. That is $A_k = U \operatorname{diag}(\sigma_1, \cdots, \sigma_k, 0, \cdots, 0) V^\top$.
A similar method is to solve the optimization problem

$$\min_A \operatorname{rank}(A) \qquad \text{subject to} \quad \|A - B\|_F \leq \epsilon$$

that has a solution in terms of the SVD of A obtained by hard-thresholding the singular values of A. Using the tightest convex relaxation of the rank, the nuclear norm (sum of the singular values) leads to the following optimization problem:

$$\min_A \|A\|_* \qquad \text{subject to} \quad \|A - B\|_F \leq \epsilon \ .$$

The corresponding lagrangian is $\mathcal{L}(A) = \frac{1}{2}\|A - B\|_F^2 + \tau \|A\|_*$. It is a strictly convex function of A and its minimizer is obtained by soft-thresholding the singular values of B,

$$\mathcal{D}_\tau(B) = U \operatorname{diag}(\max(\sigma_i - \tau, 0)) V^\top$$

and is called the *shrinkage* operator [CCSA08]. The advantage of the formulation using the nuclear norm is that as long as the loss function $\ell(A, B) = \frac{1}{2}\|A - B\|_F^2$ is replaced by other convex loss functions and the problem remains convex. The convex formulation allows to develop efficient algorithms and provide theoretical guarantees about the estimator.

Spectral functions

An interesting property pointed out by Kunegis *et al.* [KL09] is that many of the standard link prediction algorithms can be written as spectral functions of the adjacency matrix or the normalized adjacency matrix. This characterization allows to search for the best scoring rules in a very low dimensional subspace of matrix functions if one affords to pay the cost of computing an SVD and storing possibly dense $n \times n$ matrices. Even though for large values of n such a process becomes computationally challenging, this characterization motivates the use of the trace norm as a regularization term in a link-prediction objective.

The score functions obtained by walks on graphs and by low-rank approximation of adjacency matrices belong to a specific subset of the functions mapping $\mathbb{R}^{n \times n}$ to itself called unitary invariant functions [Lew95]. A characterization of these functions goes as follows, if $A = U \operatorname{diag}(\sigma_i) V^\top$ is the SVD of A, a unitary invariant function of A can be written as $U \operatorname{diag}(f(\sigma_i)) V^\top$ for a real-valued function $f : \mathbb{R}_+ \to \mathbb{R}_+$. The powers of A and consequently their linear combinations, the projection onto the r highest singular vector space and shrinkage are all unitary invariant functions of the adjacency matrix. Note that the unitary invariant functions form a linear space of dimension n that is much lower than the dimension of the set of functions $\mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n}$, that equals $(n^2)^{n^2}$.

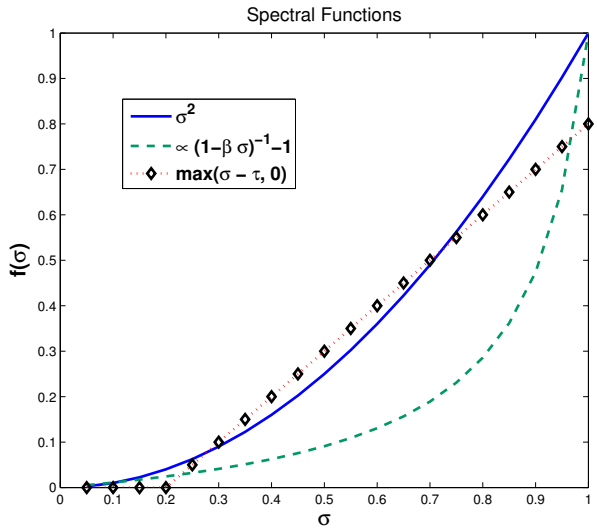


Figure 1.5: Some spectral functions of the adjacency matrix used for link preiction.

Chapter 2

Description and simulation of marketing graph data

2.1 Exploratory analysis of graph data

2.1.1 Static graph descriptors

We borrow the mathematical vocabulary of graphs for representing relations. We denote by $V = \{1, \cdots, n\}$ the set of individual objects called *nodes* or *vertices*, and suppose that the overall number of nodes n is fixed. We denote by E the *edges, links* or *connections* among nodes. E can formally be seen as a subset of $\mathbb{R}^{V \times V}$. The basic case of binary graphs corresponds to a subset of $\{0,1\}^{V \times V}$, the cases where edges have non-negative attributes is called weighted graphs. We recall the definition of a graph and a bipartite graph for the sake of completeness.

Definition 9 (Graph and adjacency matrix) *Given a set of vertices $V = \{1, \cdots, n\}$, we define a graph by $G = (V, E)$ where $E \subset \mathbb{R}^{V \times V}$ is called the edge set. The adjacency matrix $A \in \mathbb{R}^{n \times n}$ of a graph $G = (V, E)$ is defined as follows. Each element of E of the form $e : (i, j) \mapsto a$ is reflected in A through its (i, j)-th element : $A_{i,j} = a$.*

For modeling the preferences of users among a set of items, we use bipartite graphs.

Definition 10 (Bipartite graph and incidence matrix) *Given two disjoint sets of vertices*

$$V_1 = \{1, \cdots, n_1\} \text{ and } V_2 = \{1, \cdots, n_2\} ,$$

we call bipartite graph $G = (V_1, V_2, E)$ where $E \subset \mathbb{R}^{V_1 \times V_2}$ is the edge set. We associated an incidence matrix $M \in \mathbb{R}^{n_1 \times n_2}$ to a bipartite graph $G = (V_1, V_2, E)$. Each element of E of the form $e : (i, j) \mapsto a$ is reflected in M through its (i, j)-th element : $M_{i,j} = a$.

It is straightforward to see that a bipartite graph can be seen as a unipartite one over the vertex set $V_1 \cup V_2$. The adjacency matrix of the unipartite graph is given by

$$A = \begin{pmatrix} 0 & M \\ M^\top & 0 \end{pmatrix} \in \mathbb{R}^{(n_1+n_2)\times(n_1+n_2)} .$$

All along the current document we will use unipartite graphs and adjacency matrices unless the contrary is explicitly mentioned. We highlight the fact that this choice is only to simplify the notations and does not hurt the generality of the methodology.

We will be interested by low-dimensional representations of the graphs. We call *feature map* the mapping of the graph to the vector space $\mathbb{R}^d$ whose coordinates are descriptors of the graph.

Definition 11 (Graph Feature Map) *We refer to $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$ as the graph feature map.*

It is important to keep in mind that by *feature* we refer indifferently to two different types of graph descriptors.

1. **Global descriptors.** These graph features describe the whole graph as an entity. For instance in the parametric graph models, the set of parameters generating the graph can be seen as global graph descriptors. The power-law exponents, the average clustering coefficient or the Gini index of the degrees are other examples of global descriptors.

2. **Local descriptors.** Various quantities, such as the degree, the pagerank, the number of length-k cycles passing by a node, are used to describe the nodes of the graph. We refer to these as local descriptors.

We only quickly review some of the features usually used to describe graph data.

Degree of nodes, power-law distribution, Lorenz curves. The *degree* of a node is the number of connection it establishes with other nodes. In the case of weighted graphs, the degree is defined as the sum of weights of the edges of a node. If the graph is directed the *in-degree* and *out-degree* denote respectively the sum of weights of edges from a node to its neighbors, and from its neighbors to it. If A is the adjacency matrix of a graph, its degree vector is given by $A\mathbf{1}_n$, where $\mathbf{1}_n$ is the all 1 vector of length n. note that the mapping $A \mapsto A\mathbf{1}_n$ is linear.

The distribution of degrees in many graphs describing human activities such as social networks, e-commerce marketplace graphs, information graphs such as wikipedia *etc.* are known to have long tails. It has been discussed and empirically verified ,see [CSN09] or Figure (2.1.3), that many of them follow Pareto laws, sometimes called power-law or scale-free. The benefits generated by different markets also concentrate around lines in log-log plots (see Figure 2.4) when plotted as a function of sales volumes. Several types of behaviors can be distinguished. For instance books that are sold very few times generate more benefits than best-sellers, as opposite to video-games or electronic items. Bottom left: concave regions (books for instance): the more popular, the less revenue-generating the items are. Convex regions (electronic devices and video games) have the opposite behavior.

Definition 12 (Power-law) *The random variable X is said to follow a power-law distribution of parameter $(x_{\min}, \alpha)$ if its density function is of the form*

$$p(x) = Cx^{-\alpha} \quad \textit{for} \;\; x > x_{\min}$$

where $C = (\alpha - 1)x_{\min}^{\alpha-1}$ is a normalization constant, and $p(x) = 0$ if $x \leq x_{\min}$.

Lorenz curve and Gini index. The characterization of the unbalance of long-tailed data is also done using the Lorenz curves, especially in economics. Lorenz curves allows to obtain the famous 80%-20% rule that says "80% of wealth is on the hand of 20% of the population" or "80% of the sales are generated by 20% of the items". The Lorenz curve represents the cumulative weight of the population as a function of the fraction of the population ordered in the increasing order of weight. If the weight measured is the wealth, then the Lorenz curve is the set of points which x coordinate represents to which fraction of the population an individual belongs in terms of wealth, and the y coordinate represents the percentage of cumulative wealth of the population that is at most as rich as x. If $L : [0; 1] \to [0; 1]$ is the Lorenz function (cumulative weight of the population as a function of the fraction of the population ordered

in the increasing order of weight), and $F \in [0;1]$ represents the fraction of the population considered, then the Gini index is defined by $G = 1 - 2\int_0^1 L(F)dF$. It is straightforward to see that if X follows a $(x_{\min}, \alpha)$ power-law, then $G(X) = \frac{1}{2\alpha-1}$. Another well-knwon fact about Pareto law is its *scale-invariance*: for a constant $c > 0$, the two probabilities $p(x)$ and $p(cx)$ are proportional. In fact

$$p(cx) = c^{-\alpha}p(x)$$

Because of this property sometimes authors refer to social graphs having power-law degree distribution as free-scale graphs, for emphasizing that microscopic phenomena are repeated at larger scales in such graphs.

Adjacency matrix powers. If $A \in \{0,1\}^{n\times n}$ is the adjacency matrix of an unweighted graph, one sees thanks to the formula

$$(A^2)_{i,j} = \sum_{k=1}^{n} A_{i,k}A_{k,j}$$

that A^2 coefficient (i,j) is the number of paths from i to j, which coincides in the undirected case with the common neighbors of i and j. One can see similarly that A^k coefficients are the number of paths of length k joining pairs of nodes, and in particular A^k diagonal elements count the number of self-cycles of length k. In particular the diagonal elements of A^3, counting the number of triangles surrounding a node have been made popular for measuring the *homophily* in a network. We call the clustering coefficient of a node the ratio

$$\frac{\text{number of triangles surrounding a node}}{\text{number of 'V'surrounding a node}} = \frac{(A^3)_{i,i}}{d_i(d_i-1)/2}\,.$$

A smoothed generalization of random walks leads us to PageRank, the famous vertex ranking algorithm suggested by Brin and Page [BP98] that is said to be used in the Google searching engines for evaluating the relevance of each website based on the hyper-link structure of the internet web. See Figure 2.2 for an illustration of the distribution of some basic topological features of graphs. Clustering coefficients and order 1,2,3 degree (ratios) on each side of the bipartite graphs. For each node of a bipartite graph the clustering coefficient is the ratio of number of length 4 loops over the number of length 4 paths starting from that node. The higher values belong to Music and Books markets where homophily seems to play a dominant effect compared to Video Games and Electronic Device markets. Similarly the relatively higher importance of hubs in electronic and video games market can be read in the degree features distribution.

Definition 13 (Personalized PageRank) *Let $\mathcal{G}$ be a graph, and A the corresponding adjacency matrix, and $W = D^{-1}A$ the associated stochastic matrix. The stationary distribution of the random walk with teleport probability $\alpha \in]0;1[$ to $\boldsymbol{s}$, also called personalized PageRank vector is the solution of*

$$\boldsymbol{pr}_\alpha(\boldsymbol{s}) = \alpha\boldsymbol{s} + (1-\alpha)\boldsymbol{pr}_\alpha(\boldsymbol{s})W$$

Proposition 2 *The matrix*

$$R_\alpha = \alpha\sum_{k=0}^{\infty}(1-\alpha)^k W^k$$

satisfies

$$\boldsymbol{pr}_\alpha(\boldsymbol{s}) = \boldsymbol{s}R_\alpha$$

the convergence being ensured by $\alpha \in]0;1[$.

Linear approximation to the feature map. Some graph features are non-linear functions of the adjacency matrix. For instance the number of triangles incident to a node i is given by $(A^3)_{i,i}$, and therefore the clustering coefficient is a non-linear feature of the graph. In the current document, unless the contrary is mentioned explicitly, the feature map ω is linear. This simplifying hypothesis allows to build convex and differentiable objective functions in the Chapter 4. These feature maps ω should be considered as the tangent map to the actual feature map of interest. Further work should treat more general non-linear features.

2.1.2 Dynamics of graph descriptors

When the temporal evolution of the graph data matters for the application of interest, either directly or indirectly, the dynamic behavior of the graph features gets involved in the methodology. Rich information patterns are hidden in the temporal trends of graph descriptors. In subsection 1.2.2 we provided an example where the evolution of a linear feature vector of an adjacency matrix is used for valuation of customers of a C-to-C website. A more basic example (see subsection 1.2.3) is the characterization of product diffusion in a marketplace through the shape of the time series of its sales volume. The sales volume of a product being equal to its degree in the bipartite user-product graph, the diffusion level can be characterized using the time series of that graph feature. In the case of recommender systems discussed in subsection 1.2.4 it is the future value of the graph itself that is of interest. We will see that most heuristic link prediction algorithms rely on the assumption that a set of graph descriptors evolve smoothly. The prediction of edges becomes tractable when the relevant set of features is identified and their evolution well understood. In chapter 4 we explore regularized regression types of methods in order to encode this idea for a times series of graphs.

2.1.3 Marketplace seen as a dynamic bipartite graph

The graph describing a marketplace is a bipartite graph formed by the set of users and the set of products as disjoint node sets. The edges of the graph model the connections observed among the users and the products. The most frequent type of connection we are interested in is the transactions: the purchase of a product by a user is modeled as an edge between the corresponding nodes. In the sequel we provide a concrete idea of the basic characteristics of typical e-commerce data sets. We review properties of the databases that have been studied and illustrated in this thesis. The properties we report are chosen in order to clarify the objects of interest in the following sections, and are not exhaustive for a descriptive study of these databases. For more descriptive studies on information or internet domain graphs one could refer to [Les08] for instance.

Descriptive study of e-commerce markets In order to give a concrete idea of the domain specificities, we provide description of four real-world markets with different characteristics. We refer to them as **Music**, **Books**, **Electronic Devices**,**Video Games**; the market products being categorized by database managers. The data are collected from January 2009 until February 2011 unless the contrary is mentioned, see basic statistics in table 2.1. In the remainder of this work we model the marketplace data as dynamic bipartite graphs. In fact we see each purchase or declaration of affinity through any other channel as a link between a user and an item. The data we deal with can be modeled as a list of pairs of linked objects.

For basic statistics on the size and typical characteristics of the data see table 2.1, where we used the following notations. n_1: number of users, n_2: number of products, $\langle \Delta E \rangle$: daily number of sales, $\hat{\alpha_U}$ and $\hat{\alpha_P}$ estimated power coefficient for users and products. More imbalanced

distributions appear in Electronic and Video Games markets.

Domain	n_1	n_2	$\langle \Delta E \rangle$	$\hat{\alpha_U}$	$\hat{\alpha_P}$
Music	0.4M	60 k	2k	2.8	2.5
Books	1.2M	1.7M	18k	2.8	2.7
Electronic	0.5M	60k	2k	2.9	2.5
Video Games	0.9M	0.2M	9k	3.0	1.6

Table 2.1: Basic statistics on our data sets.

The tables 2.2, 2.3, 2.4 and 2.5 show the contingency table for each market. From these data, we can notice that in all markets very frequent users seem to buy very rare products, occasional buyers seem to be attracted by the popular products.

Buyer \ Item	Very Popular (> 80)	Popular(>25)	Rare (>7)	Very Rare
Very Frequent (>36)	4.77	6.30	6.87	7.05
Frequent (>11)	6.39	6.26	6.15	6.18
Occasional (>4)	7.14	6.29	5.87	5.69
Very Occasional	6.69	6.13	6.09	6.06

Table 2.2: Books contingency table. Sum of each row/column equals 25% of the total amount of purchases (edges of the bipartite graph).

Buyer \ Item	Very Popular (> 90)	Popular(> 25)	Rare (> 7)	Very Rare
Very Frequent (> 43)	3.72	5.97	7.48	7.80
Frequent(> 12)	5.40	6.42	6.55	6.61
Occasional (> 4)	7.34	6.43	5.70	5.50
Very Occasional	8.51	6.16	5.24	5.06

Table 2.3: Music contingency table. Sum of each row/column equals 25%.

The dynamics of the market graph We are interested in a sequence of graphs evolving over time. It is a choice of modelling to keep the vertex set V fixed and let the edges change over time. We formally represent a graph sequence as a series indexed by time $(G_t)_t = (V, E_t)_t$ where $E_t \subset \mathbb{R}^{V \times V}$. We choose to discretize the time domain to the set $\{1, \cdots, T\}$.
Temporal evolution of product sales. Users have different habits and behaviors on different market categories. The sales explode seasonally (before Christmas or in September for books for example) on some markets (see figure 2 and 3). The spikes in sales series drastically increase the amount of input data but, in the same time, they introduce a bias in the collection of input data. We define the penetration[SJT09] of a product over a market as the normalized cumulative volume of sales. The shape of penetration of products in different markets has an impact on the accuracy of the prediction of future popularity. For instance on the Video Games market, the shortness of product cycles make the prediction of popularity of the corresponding products intractable if based only on their purchase history and without injecting exogenous information.

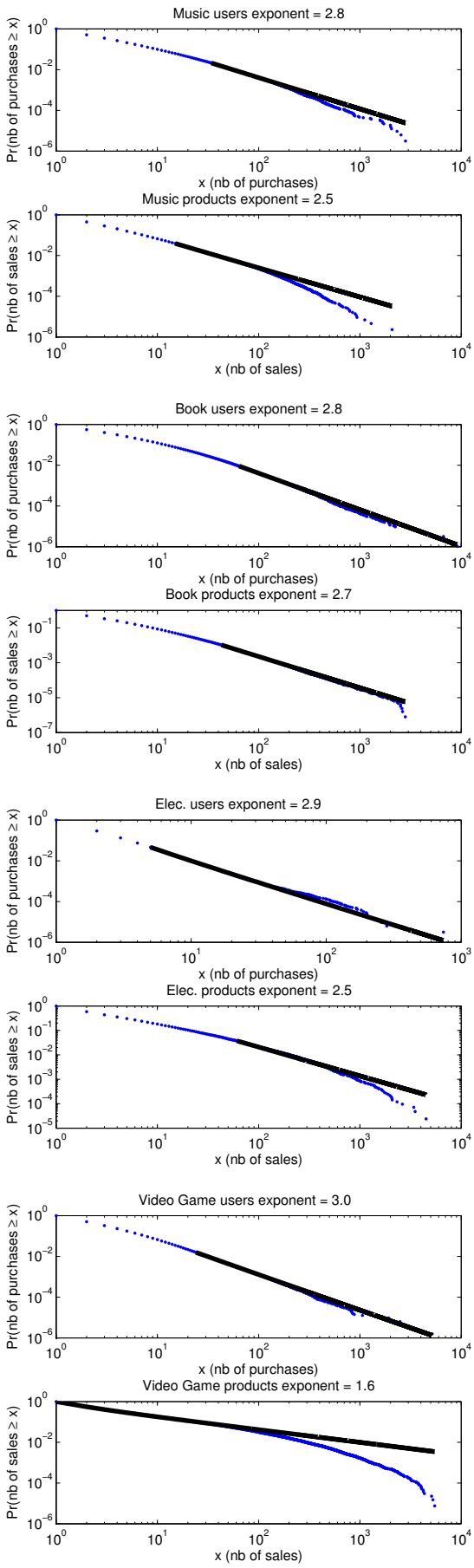


Figure 2.1: Zones of validity of the power-law fit on different markets.

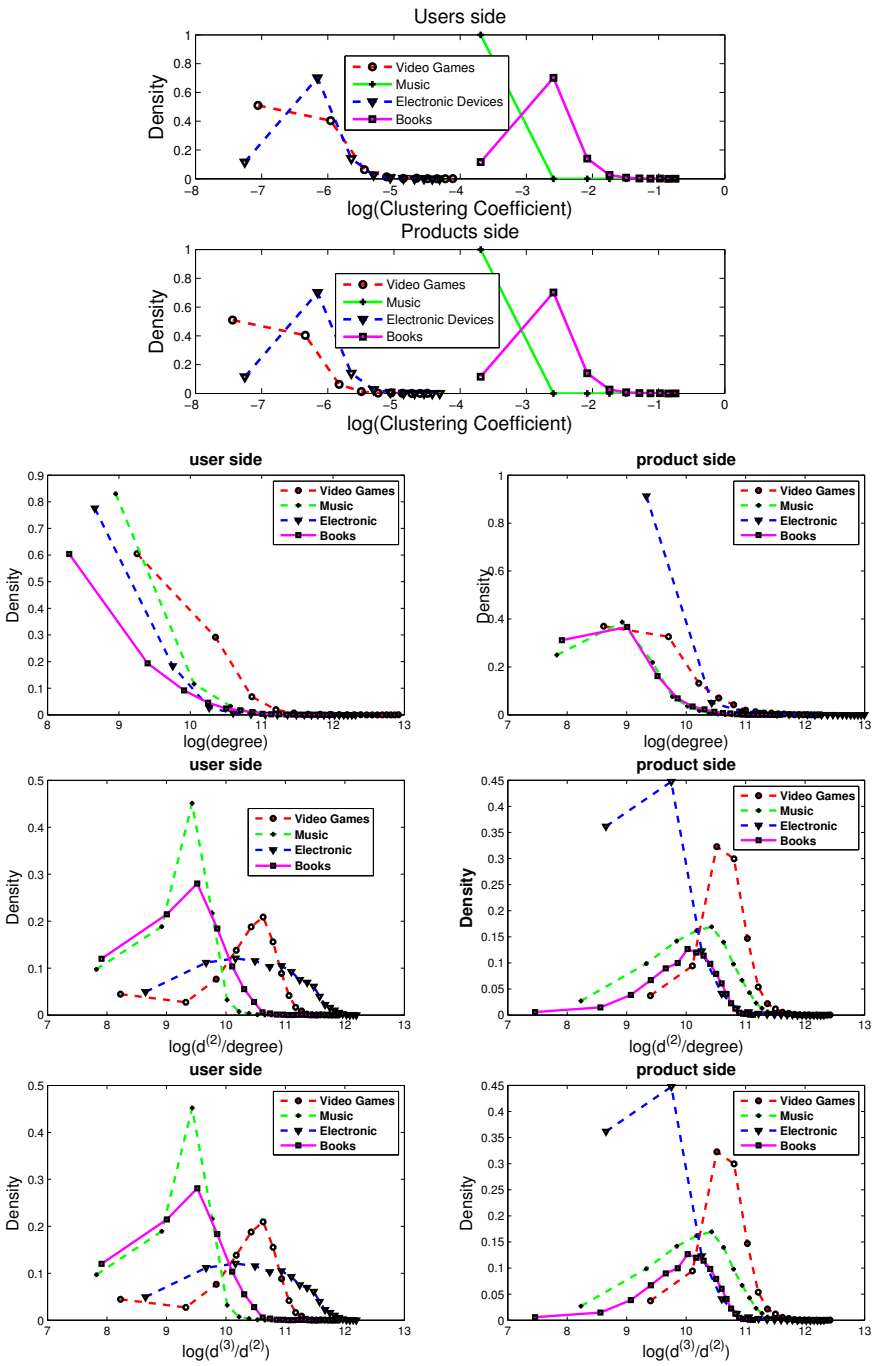


Figure 2.2: Distribution of topological network features.

Buyer \ Item	Very Popular(>1180)	Popular(>355)	Rare (>75)	Very Rare
Very Frequent (>32)	4.28	5.10	6.20	9.41
Frequent (>11)	6.19	6.44	6.45	5.914
Occasional (>4)	6.90	6.81	6.35	4.92
Very Occasional	7.62	6.63	6.00	4.74

Table 2.4: Video Games contingency table. Sum of each row/column equals 25%.

Buyer \ Item	Very Popular (> 397)	Popular (> 73)	Rare (> 16)	Very Rare
Very Frequent (> 11)	5.19	5.98	6.67	7.13
Frequent (> 4)	6.72	6.42	6.01	5.83
Occasional (> 2)	6.93	6.32	6.05	5.69
Very Occasional	6.13	6.27	6.25	6.33

Table 2.5: High-Tech contingency table. Sum of each row/column equals 25%.

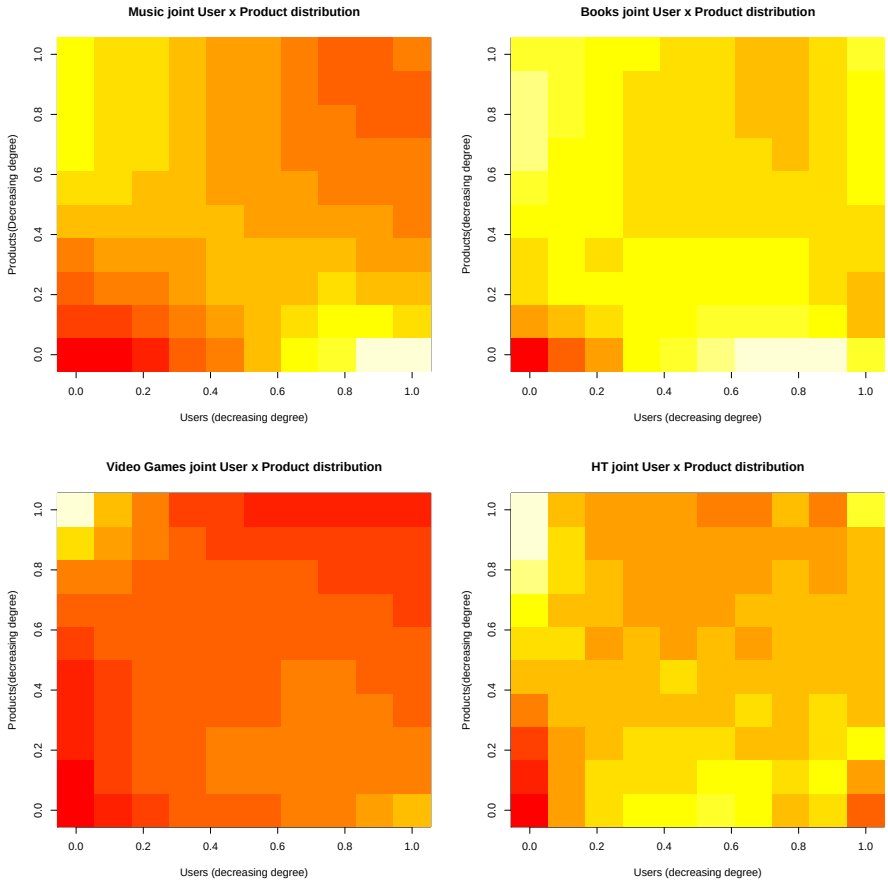


Figure 2.3: Music,Books,Video Games and Computers contingency heat maps. Sum of each row/column equals 10%. The top-left corner (most frequent user and rarest products) is much more dense than in the cultural market.

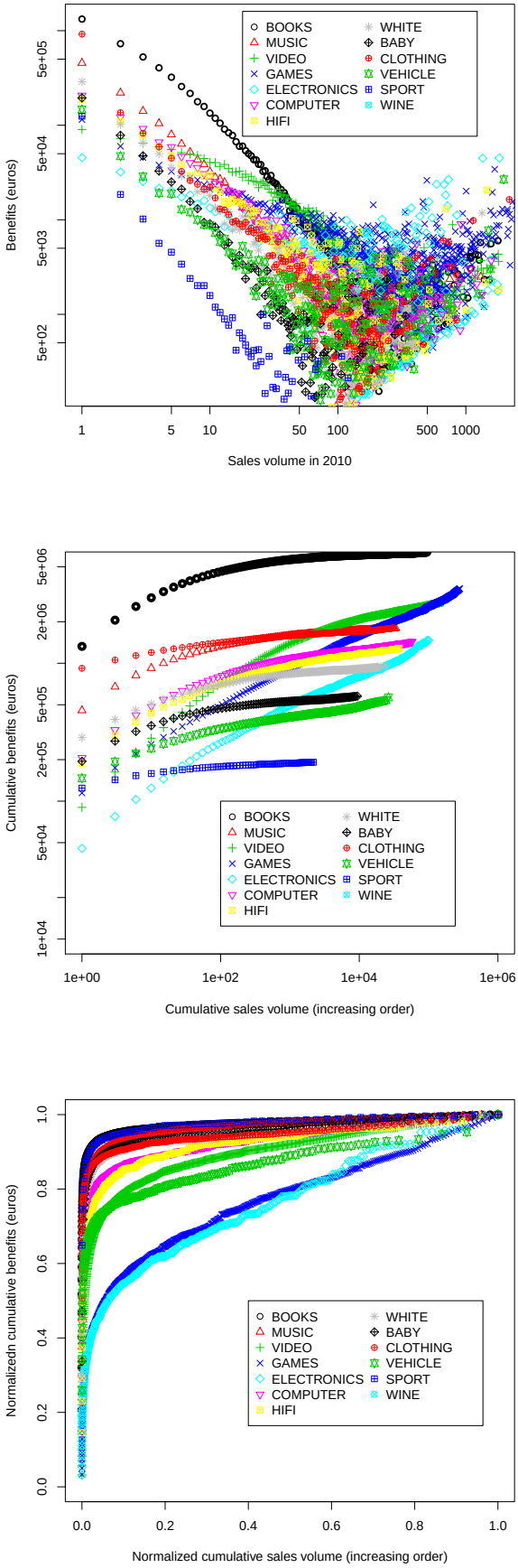


Figure 2.4: The benefits generated by different markets.

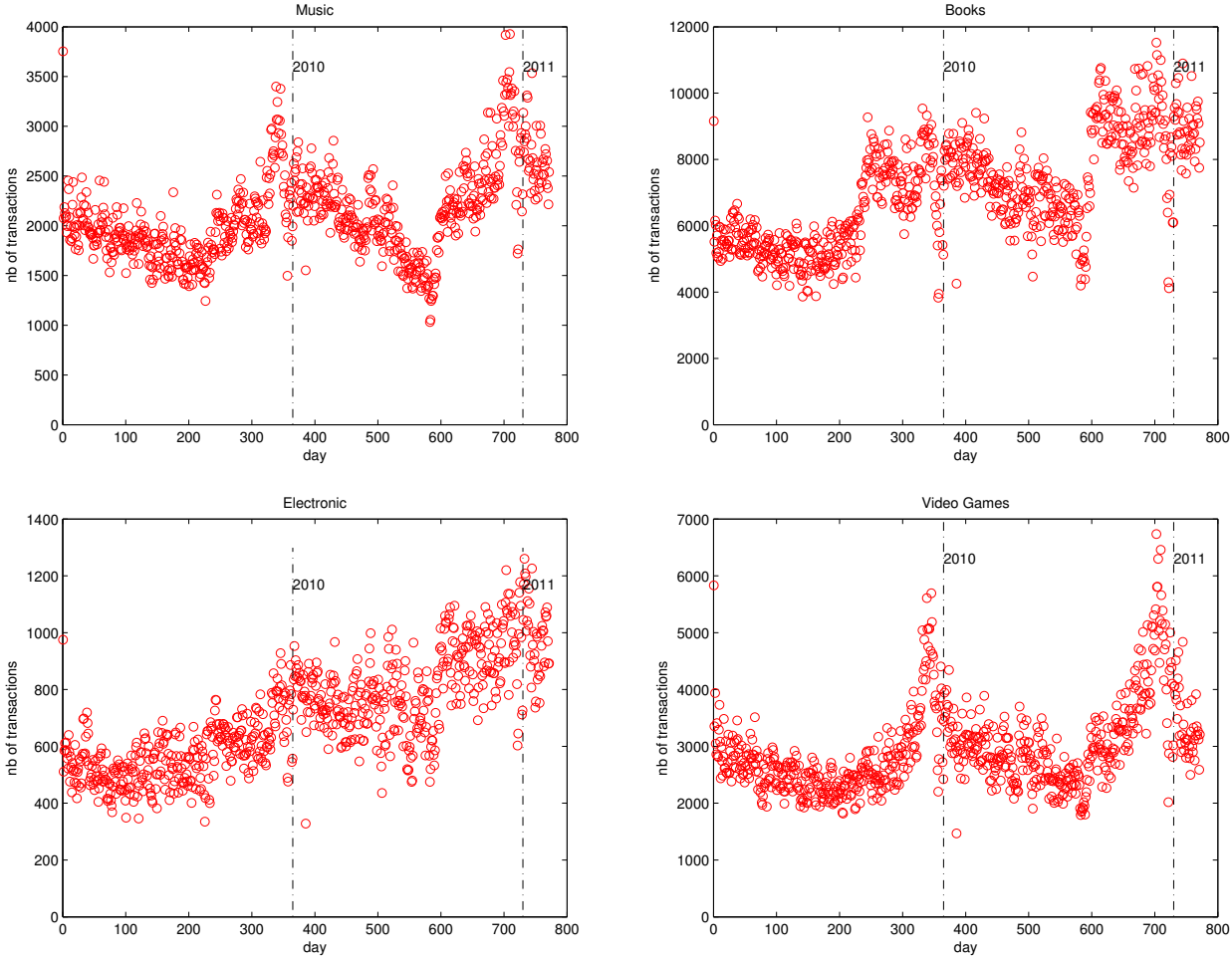


Figure 2.5: Evolution of the daily number of sales on different markets. The seasonality in Electronic device transaction seem less important, peaks in book sales are in September and before Christmas for Music and Video Games items.

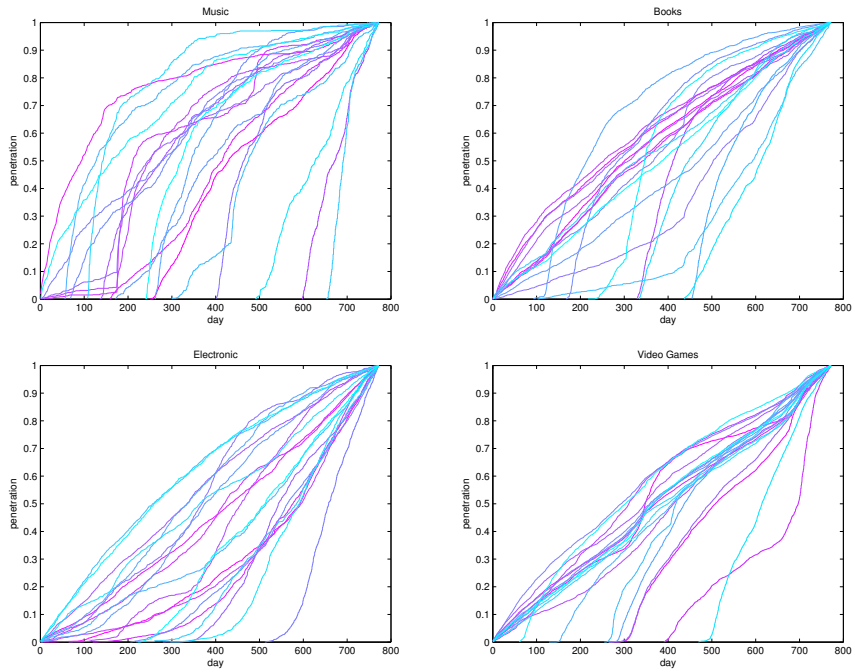


Figure 2.6: Penetrations of a set of 20 products randomly chosen among the 200 best-selling items of each market. The shapes of so-called S-curves have different profiles in various domains. Notice the diversity of diffusion profiles and the fact that for each product family they more or less follow similar properties.

2.1.4 Social networks

We have performed experiments with the Facebook100 data set analyzed by [TMP11]. The data set comprises all friendship relations between students affiliated to a specific university, for a selection of one hundred universities.

2.1.5 Protein interactions

We use data from [HJB+09], in which protein interactions in Escherichia coli bacteria are scored by strength in $[0, 2]$. The data is, by nature, sparse. In addition to this, it is often suggested that interactions between two proteins are governed by a small set of factors, such as surface accessible amino acid side chains [BG01], which motivates the estimation of a low-rank representation.

2.2 Synthetic data sets

In this section we introduce new generative models for random graphs. Further reading about standard generative models of static random graphs can be found in [Kol09a]. The motivation for developing new generative random graph models is to simulate situations where the working assumptions are rigorously verified in order to have an understanding and evaluation of the algorithms performance before testing them on the real data. In the sequel we review standard random graph models and develop new models that incorporate desired properties. In the static case we build sparse low-rank matrices and add a sparse random noise in order to

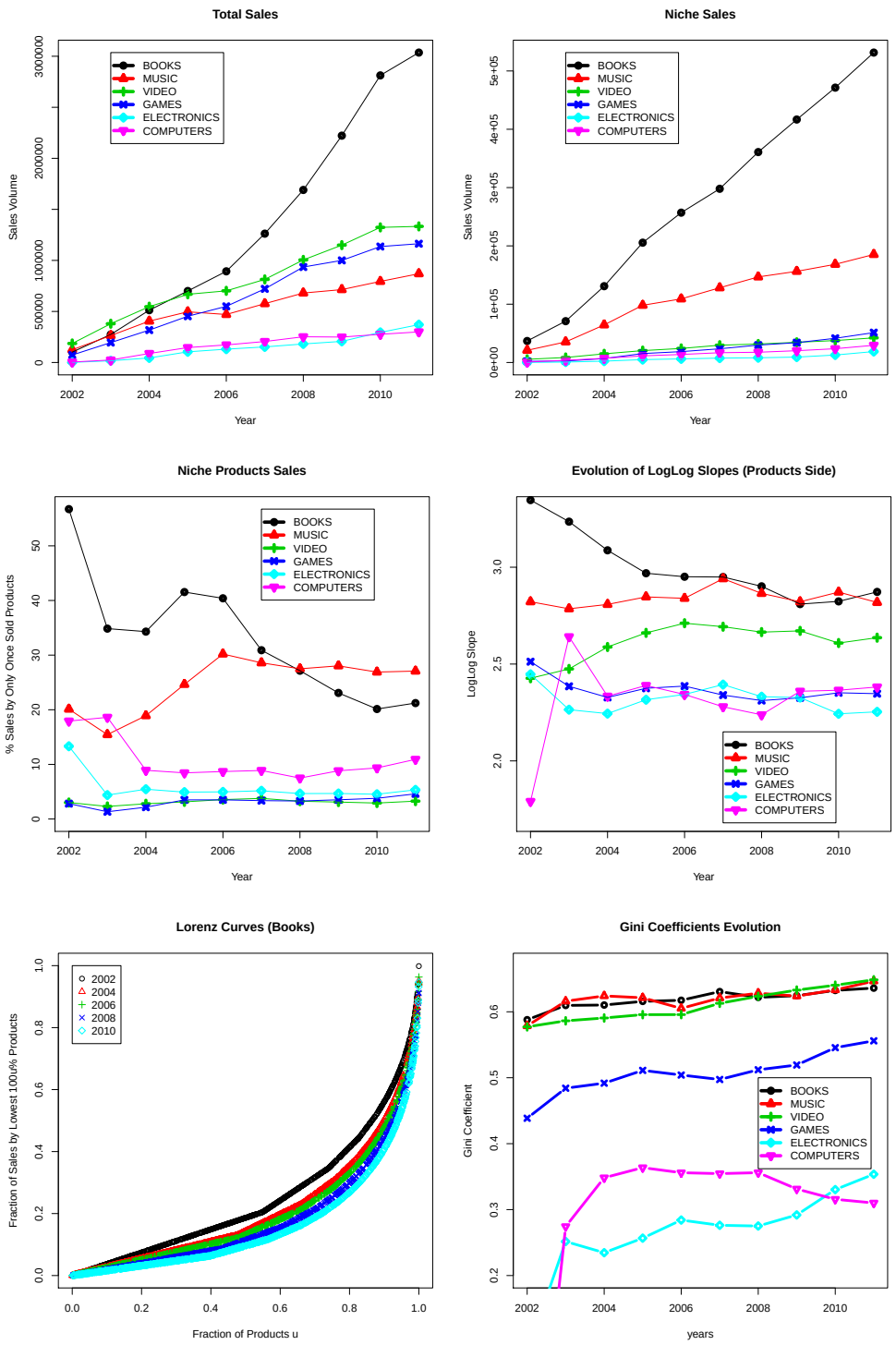


Figure 2.7: The evolution of products diversity over 10 years in several products category: books sold on the website seem to concentrate around popular items.

obtain a sparse noisy version of the signal matrix, and in the dynamic case we build a sequence of sparse low-rank adjacency matrices having linear feature vectors evolving in an autoregressive way. In Table (2.7) page 59 we bring together in a single table both the standard baseline models and the generative models introduced in the current thesis and also the corresponding estimation methodology.

2.2.1 Generative models of static random graphs

Erdos-Renyi graphs. The most basic model of random graphs has been introduced by Erdös and Renyi [ER59]. In this model any pair of nodes have equal probability to be linked by an edge. The only parameter that characterizes such a graph is the probability of presence of a link or equivalently the total number of edges. We denote by $\mathscr{G}(n,p)$ a graph having $|V| = n$ vertices, and each pair of vertices independently connected with probability p. $\mathscr{G}(n,|E|)$ is a graph having $|V| = n$ vertices, and $|E|$ edges are chosen at random among the $\binom{n}{2}$ possibilities.

For $n \to \infty$, provided that $n^2 p \to \infty$, $\mathscr{G}(n,p)$ and $\mathscr{G}(n,|E| = p\binom{n}{2})$ have equivalent properties. The following facts are known about Erdos-Renyi graphs, we refer the interested reader to [Bol01] for more details.

Property 1 (Degree distribution) *The degree distribution of $\mathscr{G}(n,p)$ follows a binomial law*

$$\mathbb{P}(d = k) = p^k (1-p)^{n-k-1} \binom{n-1}{k}$$

provided that $pn \to \lambda$, the asymptotic degree distribution of a random Erdos Renyi graph follows a Poisson law with parameter λ :

$$\mathbb{P}(d = k) \approx \frac{\lambda^k e^{-k}}{k!}$$

Property 2 (Number of cliques) *The expected number of $k-$vertex cliques in $\mathscr{G}(n,p)$ is $p^k \binom{\nu}{k}$*

Property 3 (Clustering coefficient) *The clustering coefficient, or the probability that 3 connected vertices of $\mathscr{G}(n,p)$ belong to a triangle, is $p \approx \frac{2|E|}{n(n-1)}$.*

Preferential attachment. This generative model has been designed to reproduce the popularity distribution observed in real-world networks. Its temporal formation process mimics the *rich-get-richer* phenomenon. This iterative generative model starts with an initial graph $G_0 = \{V_0, E_0\}$ and adds at each iteration a new vertex that connects to the older vertices with probability proportional to their degrees : high degree nodes are preferred to low-degree nodes. For further details, variants and links to some problems in statistical physics refer to [AB02].

Geometric random graphs. This family of random graphs [Pen03] is built as follows:

1. draw points randomly from a distribution over a subset of $\mathbb{R}^r$
2. define the vertices of the graph as the sampled points
3. connect pairs of vertices by an edge if the distance of the sample points is lower than a fixed threshold.

These graphs are often drawn from uniform distribution and over cubes or spheres.

Exponential random graph models (ERGM) [WP96]. These models see the graph z as the realization of a random variable Z. The probability mass distribution of these models is assumed to have the form

$$\mathbb{P}_\theta(A = a) = e^{\theta^\top \omega(a) - \Phi(\theta)} \tag{2.1}$$

where $\theta \in \mathbb{R}^d$ is a vector of parameters, g a function of the graph $a : \omega(a) \in \mathbb{R}^d$ and Ψ a normalization factor. In practice, if a set of parameters of interest θ (such as the number of triangles, cycles or other patterns of interest) is identified as being the sufficient statistics for estimating the graph, this model turns out to fit the right parameters by log-likelihood maximization easily.

Stochastic block model. This model introduced by Nowicki and Snijders [NS01] suggests that k blocks or clusters form the structure of the graph. This assumption is based on the observation of presence of communities in social networks. Some extensions of this work have relaxed the assumption of the existence of rigid blocks to overlapping[LBA09, ABFX08], or time-evolving blocks [HSX11].

Latent factor models. This family of models introduced by Hoff *et al.* [HRH02] assumes that given some node latent factors z_i, z_j and possibly pair-wise covariate descriptors $x_{i,j}$ the edges can be assumed to be independently distributed :

$$\mathbb{P}(A|X, Z, \theta) = \prod_{i \neq j} \mathbb{P}(A_{i,j}|X_{i,j}, Z_i, Z_j, \theta) \tag{2.2}$$

for a parameter θ and the $n \times n$ random variable Y denoting the graph adjacency matrix. The authors mention that a convenient choice of modeling, that relates this model to the exponential random graph model, may simplify the estimation of parameters to a log-likelihood maximization. They take the example of log-likelihood depending linearly on the euclidean distance :

$$\log odd(a_{i,j} = 1|x_{i,j}, z_i, z_j, \alpha, \beta) \propto \alpha - \beta x_{i,j} + \|z_i - z_j\|_2$$

Intuitively, the closer the latent factors z_i and z_j are, the higher the probability of observing a link between nodes i and j is. For further reading and a dynamic extension refer to [KH10].

2.2.2 Linear time series analysis

We call *time series*, or sequence of real or vector valued random variables indexed by time. We need to recall some basic models devoted to the study of time series before introducing generative models of dynamic random graph sequence. Predicting the future value of a time series is a challenge of interest for a wide range of applications as meteorology, economics, finance and supply chain management. Various models have been developed focusing each on time series with particular properties. We recover a family of models and methods used in the current thesis and refer to [Tsa05, BD09] for further reading. We start by a basic definition. For simplicity, we index all time series with non-negative integers $t \in \mathbb{N}$.

Definition 14 (Stationarity) *There are two forms of stationarity:*

- *If the joint distribution of any vector $(\zeta_{t-1}, \ldots, \zeta_{t-k})$, where $k < t$, remains constant over time, the the time series is strongly stationary.*
- *The weak form only assumes that the mean $\mu_t = \mathbb{E}(\zeta_t)$ and the covariance between lagged values $\Sigma_{t-k,t} = Cov(\zeta_{t-k}, \zeta_t)$, where $k < t$, are time-invariant.*

The weak form of stationarity is a more realistic assumption than the strong form. Linear econometric models have thus been developed for weakly stationary time series. We call linear time series a time series that can be written as

$$\zeta_t = \mu + \sum_{i=0}^{\infty} \psi_i \epsilon_{t-i}$$

with $\psi_0 = 1$ and ϵ_{t-i} stand for zero-mean random variables. Among linear models we focus on some particularly simple and interesting families.

Auto-Regressive models.

Definition 15 (AR(p) models) *A time series $(\zeta_t)_t$ is said to be autoregressive of order p if it follows a recursive relation of the form*

$$\zeta_t = \phi_0 + \sum_{i=1}^{p} \phi_i \zeta_{t-i} + \epsilon_t,$$

for real coefficients (ϕ_i), where $(\epsilon_t)_t$ are i.i.d. zeros-mean noise often called shocks, innovations, or innovative residuals.

To see the link with the definition of linear time series, consider the case of AR(1). In this case

$$\zeta_t = \phi_0 + \phi_1 \zeta_{t-1} + \epsilon_t \ .$$

Therefore the average value of the series is $\mu = \phi_0 + \phi_1 \mu$ or $\mu = \frac{\phi_0}{1-\phi_1}$. The model can be written as

$$\zeta_t = \frac{\phi_0}{1-\phi_1} + \sum_{i=0}^{\infty} \phi_1^i \epsilon_t$$

which is the equation of a linear time series.

The AR(p) model implies the following linear recurrent equation between the lag-i autocovariance $\rho(i) = \Sigma_{t-i,t}$:

$$\rho_k = \sum_{i=1}^{p} \phi_i \rho_{k-i}.$$

The associated polynomial equation is called the characteristic equation of the model:

$$Q(x) = 1 - \sum_{i=1}^{p} \phi_i x^i = 0.$$

Let $(\zeta_{i,\star}^{-1})_{1 \leq i \leq p}$ denote the inverses of the roots of the polynomial Q. These complex numbers are known as *characteristic roots*. It can be shown that a sufficient and necessary condition ensuring the weak stationarity of the AutoRegressive process (ζ_t) is that its the characteristic roots are smaller than 1 in modulus, i.e. $|\zeta_{i,\star}| > 1$.

Moving Average. It is interesting to note that Moving-Average models can be seen as a class of AutoRegressive models of infinite order:

$$\zeta_t = \phi_0 + \sum_{i=1}^{+\infty} \phi_i \zeta_{t-i} + \epsilon_t,$$

which, under specific parameter constraints, reduce to the expression:

$$\zeta_t = \theta_0 + \sum_{i=1}^{q} \theta_i \epsilon_{t-i},$$

where q is said to be the order of the Moving-Average model MA(q).

Auto-Regressive Moving-Average. AutoRegressive and Moving-Average models can be refined by introducing a combination of both, called AutoRegressive Moving- Average (ARMA) models. An ARMA process of order (p, q) is given by

$$\zeta_t = \phi_0 + \sum_{i=1}^{p} \phi_i \zeta_{t-i} + a_t + \sum_{i=1}^{q} \theta_i a_{t-i}.$$

Usually, this mixed form requires fewer parameters (than pure AR or MA models) to describe dynamic weak stationary data.

Besides, the condition for weak stationarity, computed with the AutoRegressive part of the ARMA model, is the same as it would be for this pure AR model. Therefore, ζ_t is weakly stationary if the characteristic roots computed with the autoregressive part are smaller than 1 in modulus.
As a consequence, the lag-i autocorrelation $\rho(i)$ of an ARMA process converges exponentially to 0 as the lag i increases. Yet, specific time series can display a decay of the autocorrelation at a polynomial rate. These long-memory effects can be reproduced by fractionally differenced processes.

Moreover, ARMA models can be extended by allowing a fixed number of characteristic roots to equal 1. Simple examples are the random walk, defined by

$$\zeta_t = \zeta_{t-1} + a_t,$$

and the random walk with drift

$$\zeta_t = \mu + \zeta_{t-1} + a_t,$$

where μ is a time trend of the series (ζ_t) and is called the drift of the model. More generally, AutoRegressive Integrated Moving-Average (ARIMA) models of order (p, d, q) are obtained by differencing d times the series ζ_t and by modelling the residuals with an ARMA model of order (p, q).

Estimating autoregressive models by ordinary least squares. When dealing with real valued time series, a simple strategy is to fit linear models to the time series. The autoregressive models attempt at establishing a linear relation among successive values of a time series. We call lagged time series, those which values at time t depends on values on times

$t-1, \cdots, T-l+1$. In the framework of autoregressive models, one would model the linear dependencies of such series as

$$\zeta_t = \sum_{k=1}^{l} \zeta_{t-k} a_k + \epsilon_t$$

where ϵ_t is zero mean noise. Note that the vector

$$z_t \doteq \begin{pmatrix} \zeta_t \\ \zeta_{t-1} \\ \vdots \\ \zeta_{t-l+1} \end{pmatrix} \in \mathbb{R}^l$$

follows the autoregressive relation

$$z_t = \begin{pmatrix} a_1 & a_2 & a_3 & \cdots & a_l \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix} z_{t-1} \ .$$

The natural generalization of autoregressive models to lagged series or multivariate series leads to vector autoregressive models. VAR models fit a vector time series $Z_t \in \mathbb{R}^d$ to its past value Z_{t-1} through a linear relation of the form

$$Z_t = W^\top Z_{t-1} + E_t$$

where E_t sands for the noise vector, that we assume having zero mean and independent from its past values. Notice that by introducing the slack variables

$$X = \begin{pmatrix} Z_{T-1}^\top \\ \vdots \\ Z_0^\top \end{pmatrix} \in \mathbb{R}^{T \times d}$$

and

$$Y = \begin{pmatrix} Z_T^\top \\ \vdots \\ Z_1^\top \end{pmatrix} \in \mathbb{R}^{T \times d}$$

the regression problem aims at estimating $W \in \mathbb{R}^{d \times d}$ in the relation

$$Y = XW + E$$

E denoting the noise. One easily gets the ordinary least squares estimate

$$\widehat{W}_{\text{o.l.s.}} = \left(X^\top X\right)^{-1} X^\top Y \ .$$

2.2.3 Generative models of dynamic graphs

In this section we introduce some models of dynamical graph that despite being extremely simplified still incorporate some desired characteristics. They have two main properties:

- A low-dimensional vector, called latent factors vector, evolves smoothly over time and dictates the evolution of the linkage structure of the graph
- Each noiseless version of the graph is *simple* at any time, meaning that a community structure exists in every snapshot of the graph. This structure is reflected onto the adjacency matrix as the low-rank property of the matrix. This prior knowledge will be used later for solving the prediction problem in an inverse problem fashion.

We built the generative models inspired by the standard random graph models, especially geometric random graph models that are related to the literature on low-rank matrices, low-rank matrix completion, and factor models. We will argue why they benefit from several properties of former generative models, making them realistic candidates for time evolving graph models. The construction process of these models is closely related to the hypotheses made for designing our main objective.

Dynamical system of latent factors. This model is close in spirit to the latent factor model in that a set of r independent *latent* factors guide the structure of the network. Let r be an integer and lets refer to $\mathbb{R}^r$ as the space of *latent factors*. Suppose a set of n points $\Psi^{(1)}(t), \cdots, \Psi^{(n)}(t) \in \mathbb{R}^r$ evolve over time following the same differential equation

$$\frac{\partial \Psi^{(i)}(t)}{\partial t} = f^\star(\Psi^{(i)}(t)) \tag{2.3}$$

where $f^\star : \mathbb{R}^r \to \mathbb{R}^r$ is continuous.
The observation of latent factors is by assumption impossible. For instance, in the case of social network data they may correspond to the psychological state of the users that evolves smoothly over time following some unknown but still stable laws.
We assume that we are also given an attraction or similarity function

$$k : \mathbb{R}^r \times \mathbb{R}^r \to \mathbb{R}$$

that is not necessarily symmetric, We observe a link between nodes i and j at time t if

$$k(\Psi^{(i)}(t), \Psi^{(j)}(t)) \geq 1 \ .$$

S-curve latent factors. Take the case of directed graphs and say $\Psi(t) = [U_t, V_t]$ where

$$\forall k \in \{1, \cdots, r\}, \quad U_t^{(i,k)} = \frac{1}{\sqrt{2\pi}\sigma_{i,k}} e^{\frac{-(t-\mu_{i,k})^2}{2\sigma_{i,k}^2}} + \epsilon_{i,k}$$

suppose V_ts have similar behaviors, and take

$$k(\Psi^{(i)}(t), \Psi^{(j)}(t)) = \mathbf{1}\{U^{(i)}(t) > \theta\}\mathbf{1}\{V^{(j)}(t) > \theta\}^\top$$

for some real valued threshold θ and parameters $\mu_{i,k}, \sigma_{i,k} > 0$, additive noise $\epsilon \sim \mathcal{N}(0, \eta_{i,k}^2)$. Hence the data observed can be modeled as a discrete sequence of graphs $\mathcal{G}_1, \cdots, \mathcal{G}_T$ where for all $t \in \{1, \cdots, T\}$, the set of nodes is $V = \{1, \cdots, n\}$ and the set of edges of $\mathcal{G}_t$ is defined by the latter equation.
The main motivation of such a model comes from the connection it establishes with the marketing literature on the cycle of life of products (see for instance [SJT09]) having S-like curves, and the factor model and geometric random graphs.

Discrete time dynamical system of latent factors. In a similar fashion we introduce a discrete-time generative model for the graph sequence and the node-related features without fixing the closed-form expression of the feature values, but rather by discretising a dynamic model and fixing the expression of $f^\star$. In fact notice that if for some smooth function $f^\star$,

$$\frac{\partial}{\partial t} z_t^{(i)} = f^\star(z_t^{(i)}) \tag{2.4}$$

then in the discrete time domain it means that

$$\forall i, \quad z_t^{(i)} - z_{t-1}^{(i)} = \eta f^\star(z_{t-1}^{(i)}) \tag{2.5}$$

where $\eta > 0$ is the time step unit.

$$\begin{cases} X_t = A_t \Omega \\ A_t = U_t V_t^\top + z_t \\ \forall i \begin{cases} U_t^{(i)} = U_{t-1}^{(i)} + \eta f^\star(U_{t-1}^{(i)}) + u_{t,i} \\ V_t^{(i)} = V_{t-1}^{(i)} + \eta f^\star(V_{t-1}^{(i)}) + v_{t,i} \end{cases} \end{cases}$$

where $u_{t,i}, v_{t,i}$ are vectors representing noise in the latent factor space, for instance drawn from $\mathcal{N}(0, \delta^2 I_r)$ in $\mathbb{R}^r$, and the entries of z_t are independently drawn from a centered Gaussian with variance σ^2 and represent noise at edge level. The important assumption on $f^\star : \mathbb{R}^r \to \mathbb{R}^r$ is continuity.

In the synthetic datasets used in our numerical experiments we took

$$f^\star(x) = e^{\frac{-\|x - v_1\|^2}{\sigma_1^2}}(x - v_1) + e^{\frac{-\|x - v_2\|}{\sigma_2}}(x - v_2) \ ,$$

where vectors $v_1, v_2 \in \mathbb{R}_+^r$ are chosen randomly and $\epsilon, \sigma_1, \sigma_2$ are positive constants. We used $n = 100$, $r = 4$, $T = 60$ and the entries of $U_0, V_0 \in \mathbb{R}^{n \times r}$ are drawn according to a uniform distribution in $(0, 1)$.

Low-rank matrix sequence having linear autoregressive features (LR-LinFeat). Let $V_0 \in \mathbb{R}^{n \times r}$ be an orthogonal matrix *i.e.* $V_0^\top V_0 = I_r$. Take $W_0 \in \mathbb{R}^{r \times r}$ be a square matrix, and fix also $U_0 \in \mathbb{R}^{n \times r}$. Define the sequence of matrices $(A_t)_{t \geq 0}$ by for $t = 1, 2, \cdots$

$$U_t = U_{t-1} W_0 + N_t$$

and

$$A_t = U_t V_0^\top$$

for zero-mean i.i.d noise N_t, for example gaussians, $(N_t)_{i,j} \sim \mathcal{N}(0, \sigma^2)$

If we define the linear feature map $\omega(A) = AV_0$, note that

1. The sequence $\Big(\omega(A_t)\Big)_t = \Big(U_t\Big)_t$ follows the linear autoregressive relation

$$\omega(A_t) = \omega(A_{t-1})W_0 + N_t$$

2. For any time index t, the matrix A_t has rank at most r

the requirement not fulfilled by this matrix sequence is sparsity of A_t.

Low-rank and sparse sequence with thresholded linear features. Let $V_0 \in \mathbb{R}^{n\times r}$ be a sparse matrix and $V_0^\dagger$ its pseudo-inverse $V_0^\dagger V_0 = V_0^\top V_0^{\top\dagger} = I_r$. Fix the two matrices $W_0 \in \mathbb{R}^{r\times r}$ and $U_0 \in \mathbb{R}^{n\times r}$. For $t = 1, 2, \cdots$ let

$$U_t = \mathbf{1}\{U_{t-1}W_0 + N_t > \theta\}$$

and

$$A_t = U_t V_0^\top + M_t$$

for white gaussian i.i.d noise $(N_t)_{i,j} \sim \mathcal{N}(0, \sigma^2)$.

Now define the linear feature map $\omega(A) = AV_0^{\top\dagger}$ and note that

1. The sequence $\Big(\omega(A_t)\Big)_t = \Big(U_t\Big)_t$ does not follows linear autoregressive relation. In fact

$$\omega(A_t) - \omega(A_{t-1})W_0 = \mathbf{1}\{U_{t-1}W_0 + N_t > \theta\} - U_{t-1}W_0$$

2. $\forall t$, A_t has rank at most r

3. $\forall t$, A_t is sparse as $V_0^\top$ and U_t are both sparse

The advantage of this model is to come up with sparse and low-rank matrices, but the linear features do not follow a linear autoregressive relation. One may possibly argue that in some particular cases their asymptotic behavior approaches the linear autoregressive one. Notice also that we avoided thresholding a low rank dense matrix for obtaining a sparse and low-rank matrix. In fact a basic example suffices to get convinced that thresholding low-rank matrices does not result in low-rank matrices. Take $u > 1$ and $v = (u, \cdots, u^n)^\top \in \mathbb{R}^n$, and $A = vv^\top = (u^{i+j})_{i,j}$. The matrix $\mathbf{1}\{A > \theta\}$ for $\theta = u^n$ is anti-triangular and has full rank n despite the minimal rank 1 of A.

The following model attempts to fulfill all the requirements.

Sparse low-rank sequence having permutation-autoregressive features. Let $\nu : \mathbb{R}^{n\times r} \times \mathbb{N}$ be a discrete noise function, such that $\nu(U, \sigma)$ is a copy of U where σ pairs of elements have been chosen uniformly at random and permuted. Let

- $V_0 \in \{0, 1\}^{n\times r}$ be a sparse matrix and $V_0^\dagger$ its pseudo-inverse $V_0^\dagger V_0 = V_0^\top V_0^{\top\dagger} = I_r$. $V_0^\dagger$ has no reason to be sparse
- $W_0 \in \mathcal{S}(r)$ a permutation matrix on r elements: $W_0 \in \{0, 1\}^{r\times r}$ is invertible with $W_0^{-1} \in \{0, 1\}^{r\times r}$
- $U_0 \in \{0, 1\}^{n\times r}$

- Fix an integer $\sigma < nr$ and take for $t = 1, 2, \cdots$

$$U_t = \nu(U_{t-1}W_0, \sigma)$$

$$A_t = U_t V_0^\top$$

Now define the linear feature map $\omega(A) = AV_0^{\top\dagger}$ and note that

- The sequence $\Big(\omega(A_t)\Big)_t = \Big(U_t\Big)_t$ follows the linear autoregressive relation

$$\|\omega(A_t) - \omega(A_{t-1})W_0\|_0 = \|U_t - U_{t-1}W_0\|_0 = \|\nu(U_{t-1}W_0, \sigma) - U_{t-1}W_0\|_0 \leq \sigma$$

  and

$$\|\omega(A_t) - \omega(A_{t-1})W_0\|_2 \leq \sqrt{\sigma}$$

- $\forall t$, A_t has rank at most r
- $\forall t$, A_t is sparse as $V_0^\top$ and U_t are both sparse

Low-rank and sparse matrix sequence having linear autoregressive features (SPLR-LinFeat). The difference with the non-sparse low-rank case is that the noise is supposed to be sparse.

The construction goes as follows. Let $V_0 \in \mathbb{R}^{n\times r}$ be a sparse matrix $V_0^\dagger$ its pseudo-inverse $V_0^\dagger V_0 = V_0^\top V_0^{\top\dagger} = I_r$. Fix the two sparse matrices $W_0 \in \mathbb{R}^{r\times r}$ and $U_0 \in \mathbb{R}^{n\times r}$. Now define the sequence of matrices $(A_t)_{t\geq 0}$ by for $t = 1, 2, \cdots$

$$U_t = U_{t-1}W_0 + N_t$$

and

$$A_t = U_t V_0^\top + M_t$$

for i.i.d sparse noise matrices N_t and M_t, which means that for any pair of indices (i, j), with high probability $(N_t)_{i,j} = 0$ and $(M_t)_{i,j} = 0$.

If we define the linear feature map $\omega(A) = AV_0^{\top\dagger}$, note that

1. The sequence $\Big(\omega(A_t)\Big)_t = \Big(U_t + M_t V_0^{\top\dagger}\Big)_t$ follows the linear autoregressive relation

$$\omega(A_t) = \omega(A_{t-1})W_0 + N_t + M_t V_0^{\top\dagger}$$

2. For any time index t, the matrix A_t is close to $U_t V_0$ that has rank at most r
3. A_t is sparse, and furthermore U_t is sparse

The two following models are designed for generating sequences of SPD matrices, used frequently in the study of gaussian graphical models.

Positive semi-definite low rank matrices with continuously evolving features. The motivation for introducing this model is to add PSD condition to the low-rank graphs, which appear when studying the graphical model underlying high-dimensional distributions [ZLW08, KSAX10]. Let $z_t^{(i)} \in \mathbb{R}^r$ denote latent factors of each node i at time t. Suppose we observe the sequence of covariances

$$\left(\Sigma_t = \left(\langle z_t^{(i)}, z_t^{(j)} \rangle\right)_{1 \le i,j \le n}\right)_{1 \le t \le T} .$$

Equivalently if we denote by

$$Z_t = \begin{pmatrix} z_t^{(1)\top} \\ \vdots \\ z_t^{(n)\top} \end{pmatrix} \in \mathbb{R}^{n \times r}$$

the matrix of latent factors, we have

$$\Sigma_t = Z_t Z_t^\top$$

Suppose there exists a function $f^\star : \mathbb{R}^r \to \mathbb{R}^r$ such that in the continuous time domain

$$\frac{\partial}{\partial t} z_t^{(i)} = f^\star(z_t^{(i)})$$

Smoothness assumption. The function $f^\star$ assumed to be smoothly differentiable. In the discrete time domain it means that

$$\forall i, \quad z_t^{(i)} - z_{t-1}^{(i)} = \eta f^\star(z_{t-1}^{(i)})$$

where $\eta > 0$ is the time step unit.

Positive semi-definite low rank matrices with linearly evolving features. Take

$$\Sigma_t = Z_t Z_t^\top + M_t M_t^\top$$

where for some $W_0 \in \mathbb{R}^{d \times d}$

$$Z_t = Z_{t-1} W_0 + N_t$$

Generate gaussian vectors $x_t \sim \mathcal{N}(0, \Sigma_t)$, the dynamical graphical model describing $(x_t)_t$ has linear autoregressive features $(Z_t)_t$.

Name	Generative process	Interesting properties	Prediction / estimation
Erdös-Renyi	Link any pair of vertices with the same probability p	1. Binomial degrees $\sim$ Poisson(np) 2. Clustering coefficient $\frac{2\|E\|}{n(n-1)}$ in expectation	Impossible
Rich-get-richer	Preferential attachment	Power-law degree distribution	Link probability proportional to degrees of extremal nodes
ERGM	Sampling from the distribution $\mathbb{P}_\theta(A = a) \propto e^{\theta^\top \omega(a)}$	Feature vector $\omega(a)$	Log-likelihood maximization
Latent factor model / Geometric random graphs	Draw edges randomly in $\mathbb{R}^r$ and connect close edges	Low-dimensional underlying manifold	Matrix factorization, minimization of $\|\omega(A) - \omega_0\|_2^2 + \tau\|A\|_* + \gamma\|A\|_1$
Noisy low-rank graph	$B = UV^\top + N$ with $U, V \in \mathbb{R}^{n\times r}$ sparse, $N \in \mathbb{R}^{n\times n}$ sparse noise	1. Sparse low-rank signal 2. Sparse noise	Minimization of $\|A-B\|_1+\tau\|A\|_*+\gamma\|A\|_1$

Table 2.6: A summary of different *static* random graph generators, corresponding properties and estimation / prediction procedures.

Name	Generative process	Interesting properties	Prediction / estimation
LR-LinFeat	$\begin{cases} A_t = U_t V_0 + M_t \\ U_t = U_{t-1} W_0 + N_t \\ \omega(A_t) = A_t V_0^\top \end{cases}$ V_0 orthogonal, M_t, N_t zero-mean i.i.d. noise	1. A_t low-rank 2. $\omega(A_t)$ linear autoregressive	Minimizing $\mathcal{L}$ for $\gamma = 0$, **pen** $= \frac{1}{2}\|.\|_F^2$
SPLR-LinFeat	$\begin{cases} A_t = U_t V_0 + M_t \\ U_t = U_{t-1} W_0 + N_t \\ \omega(A_t) = A_t V_0^{\top\dagger} \end{cases}$ V_0, W_0, sparse M_t, N_t zero-mean sparse i.i.d. noise	1. A_t sparse 2. A_t low-rank 3. $\omega(A_t)$ linear autoregressive	Minimizing $\mathcal{L}$ for **pen** $= \|.\|_1$
DynSysLF	$\begin{cases} A_t = U_t V_t^\top + z_t \\ \forall i \begin{cases} U_t^{(i)} = U_{t-1}^{(i)} + \\ \eta f^\star(U_{t-1}^{(i)}) + u_{t,i} \\ V_t^{(i)} = V_{t-1}^{(i)} + \\ \eta f^\star(V_{t-1}^{(i)}) + v_{t,i} \end{cases} \end{cases}$ $f^\star$ smooth	1. A_t low rank 2. f regular over the graph	Minimization of $\mathcal{L}_{\text{Lap}}$

Table 2.7: *Dynamic* random graph generators.

Chapter 3

Estimation of sparse low-rank matrices

3.1 Context

Matrix estimation is at the center of many modern applications and theoretical advances in the field of high dimensional statistics. The key element which differentiates this problem from standard high dimensional vector estimation lies in the structural assumptions which are formulated in this context. Indeed, the notion of sparsity assumption has been transposed into the concept of low-rank matrices and opened the way to numerous achievements (see for instance [Sre04, CCS08]). In this chapter, we argue that the low-rank-ness is not only an equivalent of sparsity for matrices but that being low-rank and sparse can actually be seen as two orthogonal concepts. The underlying structure we have in mind is that of a block diagonal matrix. This situation occurs for instance in covariance matrix estimation in the case of groups of highly correlated variables or when denoising/clustering social graphs.

Efficient procedures developed in the context of sparse model estimation mostly rely on the use of ℓ_1-norm regularization [Tib96]. Natural extensions include cases where subsets of related variables are known to be active simultaneously [YL06]. These methods are readily adapted to matrix valued data and have been applied to covariance estimation [EK09, BT10] and graphical model structure learning [BEGd07, FHT08]. In the low-rank matrix completion problem, the standard relaxation approach leads to the use of the trace norm as the main regularizer within the optimization procedures [SRJ05, KLT11] and their resolution can either be obtained in closed form (loss measured in terms of Frobenius norm) or through iterative proximal solutions [CP11, BT09] (for general classes of losses). However, solutions of low-rank estimation problems are in general not sparse at all, while denoising and variable selection on matrix-valued data are blind to the global structure of the matrix and process each variable independently.

In this chapter, we study the benefits of using the sum of ℓ_1 and trace-norms as regularizer. This sum of penalties on the same object allows to benefit from the virtues of both of them, in the same way as the elastic-net [ZH05] combines the sparsity-inducing property of the ℓ_1 norm with the smoothness of the quadratic regularizer. Trace norm and ℓ_1 penalties have already been combined in a different context. In Robust PCA [CLMW09] and related literature, the signal X is assumed to have an additive decomposition $X = S+L$ where S is sparse and L low-rank. Note that X is not in general sparse nor low-rank and that this decomposition is subject to identifiability issues, as analyzed, e.g., in [CSPW11]. The decomposition is recovered by using ℓ_1-norm regularization over X and trace norm regularization over Y. This technique has been successfully applied to background substraction in image sequences, to graph clustering [JSX11] and covariance estimation [Luo11].

Here, we consider the different situation where the matrix S is sparse and low-rank at

the same time. We demonstrate the applicability of our mixed penalty on different problems. We develop proximal methods to solve these convex optimization problems and we provide numerical evidence as well as theoretical arguments which illustrate the trade-off which can be achieved with the suggested method.

A leading insight for inferring graph structures coming from social activities is that the informative graph is highly structured [GN02] while the noise can be modeled as a parsimonious random pattern, such as an Erdős-Rényi random graph. A variety of approaches have been developed for the *link prediction* problem from a snapshot or a sequence of snapshots of a graph using matrix factorization [Kor08], probabilistic methods [TWAK03] or heuristics [LNK07b, SCM10b]. We treat the temporal case in chapter 4 of this thesis. Most methods assume, however, that the observed links are reliable and try to predict links that are either missing or, in a time-dependent setting, that are likely to appear in the future. As a consequence, these approaches can not remove irrelevant links and can be sensitive to their presence. Another feature of these relational databases is that both the informative component and the noise are sparse. In terms of matrix recovery, the setup corresponds to the case where the observed adjacency matrix is a sum of a *simultaneously* sparse and low-rank matrix (highly structured part) and a sparse matrix (noise) that is potentially full-rank with an almost flat spectrum [VM11].

Recently, Candès *et al.* suggested in [CLMW09] the robust principal component analysis (RPCA) approach for decomposing a matrix as the sum of a sparse matrix and of a low-rank matrix. A closely related method has been applied to graph clustering [JSX11]. In spite of computationally more attractive variants [LGW$^+$09, ZT11], RPCA and its derivatives have high computational and memory requirements which may be unsuitable for the analysis of massive datasets. More important, this approach has severe limitations when the sparse component turns out to be low-rank or when the low-rank component is itself sparse. This intuition has been formalized as the rank-sparsity uncertainty property in [CSPW11]. The setup considered in this chapter corresponds exactly to the case where the RPCA approach fails to perform accurate recovery. We propose a method, explicitly designed to recover matrices that are simultaneously sparse and low-rank when corrupted by sparse noise. Recovering simultaneously sparse and low-rank matrices was addressed recently in the case of the squared loss in [RSV12] for well concentrated and dense noise. In the latter case, the natural loss for the estimation error is the ℓ_1-norm [BJK78, GH10]. We develop two algorithms to solve relaxations of the graph denoising problem and demonstrate their efficiency. The first algorithm that we propose is based on a convex formulation. It enforces both sparsity and low-rank on the estimator under a sparsity inducing loss function. We also present a scalable method based on matrix factorization, where we perform rank-one updates with a flavor similar to sparse principal component analysis (SPCA, see, e.g., [DEGJL07]).

The remainder of the chapter is organized as follows. After motivating the problem in section 3.1, the regularized approach is described in Section 3.2. The oracle bounds for regression with the introduced penalty are provided in Section 3.3. The algorithms for the optimization of both the differentiable and non-differentiable loss are introduced in Section 3.4. Section 3.5 contains numerical results which illustrate the potential of these algorithms.

3.2 Setup and motivations

3.2.1 Notations

We first set some notations. For a matrix $X = (X_{i,j})_{i,j}$, the Frobenius norm is $\|X\|_F^2 \doteq \sum_{i,j} X_{i,j}^2$, the matrix ℓ_1-norm is $\|X\|_1 \doteq \sum_{i,j} |X_{i,j}|$ and the nuclear norm is defined by $\|X\|_* \doteq \sum_{i=1}^{\mathrm{rank}(X)} \sigma_i$,

where σ_i are the singular values of X and rank(X) is the rank of X. The sparsity index of X or ℓ_0 pseudo-norm is $\mathtt{nnz}(X) = \|X\|_0 = |\{X_{i,j} \neq 0\}|$. We shall also use $\|X\|_{\text{op}} \doteq \sup_{w\,:\,\|w\|_2=1} \|Xw\|_2$ and $\|X\|_\infty \doteq \max |X_{i,j}|$. The inner product of two matrices X and Y of the same size is defined by $\langle X, Y\rangle = \mathrm{Tr}(X^\top Y)$. The component wise product of matrices is denoted by $\circ$. The class S_n^+ of matrices is the convex cone of positive semidefinite matrices in $\mathbb{R}^{n\times n}$. We denote the projection of a matrix X onto a convex set $\mathcal{S}$ by $P_{\mathcal{S}}(X)$. The matrix $(X)_+$ is the componentwise positive part of the matrix X, and $\mathrm{sign}(X)$ is the sign matrix associated to X with the convention $\mathrm{sign}(0) = 0$.

3.2.2 Motivations

We consider a matrix regression model with observations of the form

$$y_i = \langle \Omega_i, X_0\rangle + \varepsilon_i, \quad i = 1, \dots, d \tag{3.1}$$

where $\varepsilon = (\varepsilon_i)$ is i.i.d. noise, $X_0 \in \mathbb{R}^{n\times n}$ is an unknown matrix to be estimated and

$$\Omega_1, \dots, \Omega_d \in \mathbb{R}^{n\times n}$$

are fixed (deterministic) design matrices. As a shorthand, define the linear map $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$ as

$$\omega(X) = (\langle \Omega_1, X\rangle, \cdots, \langle \Omega_d, X\rangle) \quad .$$

Shortly, we have $y = \omega(X_0) + \varepsilon$.

For $\mathcal{S} \subset \mathbb{R}^{n\times n}$ some convex admissible set, we consider the estimation procedure

$$\widehat{X} = \underset{X\in\mathcal{S}}{\arg\min} \{\ell(\omega(X), y) + \tau\|X\|_* + \gamma\|X\|_1\} \tag{3.2}$$

and in the following, we study properties of the estimator $\widehat{X}$. The underlying assumption in this work is that the unknown matrix to be recovered is simultaneously sparse and low-rank. This is the case of matrices that are block-diagonal in a certain basis, but they are not the only sparse low-rank matrices.

In the following we provide examples of applications where target matrices to be estimated are sparse low-rank. We write the loss functions desirable for each case and provide basic properties of each. In the next section we study the case of the quadratic loss by providing oracle bounds.

Multivariate regression

The problem (3.1) is a multivariate regression problem (see [BF97]). Assuming that the multiple outputs share a common structure, *i.e.* they can be expressed using the same *small* set of linear combinations of initial features, then the regression matrix X_0 to be estimated is low-rank. Moreover, in a setup where the number of descriptors is potentially high we expect most of them to be irrelevant for the prediction task, hence the same low-rank matrix X_0 is also sparse. A special case of regression involving sparse low-rank matrices arises when predicting linear features of a graph sequence in an autoregressive fashion. Such a setting has been considered in problems such as customer valuation and network effect measurement [ZEPR11] and link prediction in a temporal setting (*e.g.* [RBEV10], [VAHS11]), where the observations consist of a set of linear features of the graph (such as the degrees) and the prior knowledge on the domain is that the graph to be predicted contains highly connected communities. The

community structure of the graph is translated by a joint rank and sparsity constraint, that must be imposed on the linear regression.

In section 3.3 we provide oracle bounds that guarantee the quality of the estimator defined by (3.2) in the case of least-squares regression.

Covariance matrix estimation

Many interesting examples arise from the simplest denoising case, where the design matrices correspond to the canonical basis of $\mathbb{R}^{n\times n}$, or equivalently $\omega(X) = X$. In this setting the matrix A represents a noisy estimate of the true covariance matrix. It can be obtained for instance by estimating the sample covariance with few (less than n) observations which may be noisy themselves. The search space in this case is $\mathcal{S} = X_n^+$, the cone of positive semidefinite matrices. For this application the noise matrix can be a full matrix, theregore the squared norm $\ell(X, A) = \|X - A\|_F^2$ is often used in matrix covariance estimation applications. Corollaries of oracle bounds presented in section 3.3 apply to the current case ensuring the quality of the estimated covariance matrix.

Least absolute deviation and graph denoising

We now turn to the case of sparse noise, in the following sense: for each $i = 1, \dots, d$, ε_i is zero with probability p close to one. Hence, the expected number of nonzero entries of ε is dp. In a regression setup such a situation is often treated by using an ℓ_1 loss $\ell(X, y) = \|\omega(X) - y\|_1$, and is called *least absolute deviation* (see [BJK78]), which has been studied in a penalized framework by [GH10]. In the present work, for shortness and as our principal goal is to study the virtues of the mixed penalty, we study only the simple denoising case $\omega(A) = A$.

A problem which arises in many relational databases is the presence of irrelevant relations between objects, while many relevant links are often missing. Representing these data as graphs, both missing and undesirable links can be modeled as noisy edges. These noisy links can be due to measurement bias or to the specific interpretation on what type of interaction should be encoded through these links. In the case of social networks, a link is supposed to indicate friendship between two members. However, relations may be declared after randomly meeting at a party, while close friends or family may see no reason to establish digital friendship through the social network. A similar example comes from purchase data on e-commerce websites where users may purchase items for their relatives and produce irrelevant links in the bipartite graph of users and products. Recommender systems based on these relational data face the challenge of removing those noisy links, since they are meaningless in terms of affinity between members of the network. Recovering structure in large noisy graphs presents a major interest both in collaborative filtering and exploratory web mining.

Our objective is to explore the following question: "how can one obtain a sparse and low-rank matrix by adding or removing as few edges as possible from a given sparse adjacency matrix? "

In the presence of a sparse unstructured noise, the problem we are interested in is to estimate a matrix simultaneously sparse and low-rank by adding or removing only a few edges from an observed matrix $A = S_0 + \varepsilon$. This can be formulated as follows:

$$\begin{cases} \min\limits_{X} \ \ \|X - A\|_0 \\ \text{subject to } \ \|X\|_0 \le q \ \ \text{and} \ \ \operatorname{rank}(X) \le r \ . \end{cases}$$

This is, however, a hard combinatiorial problem. Instead, we consider the relaxation obtained by replacing $\|.\|_0$ by its tightest convex upper bound, the matrix ℓ_1-norm. Similarly,

we relax the rank constraint using the nuclear norm. This motivates the following penalized estimator, for some parameters $\tau, \gamma > 0$ (set by cross-validation in practice) :

$$\widehat{X} \in \arg\min_{X \in \mathcal{S}} \{\|X - A\|_1 + \tau\|X\|_* + \gamma\|X\|_1\} \ . \tag{3.3}$$

The objective function is convex but neither smooth, nor strictly convex, due to the presence of the ℓ_1-loss. This particular loss has been used in regression under the name of least absolute deviations as an alternative to ordinary least squares [BJK78], or in a high-dimensional setting using ℓ_1-norm penalization [GH10]. The introduction of the ℓ_1-loss is generally motivated by the presence of heavy-tailed noise and outliers, as it is more robust. The combination of ℓ_1-norm and nuclear norm penalty on the same matrix to obtain a simultaneously sparse and low-rank estimator has been recently proposed in [RSV12] using a squared loss. As it turns out, the non differentiable ℓ_1 loss is significantly more challenging to tackle, but also much more suited to the problem of inferring links in graphs in the presence of sparse noise.

This approach encompasses various other methods. In particular, RPCA corresponds to the problem

$$\min_{X,Y \in \mathbb{R}^{n \times n}} \{\|Y\|_1 + \tau\|X\|_*\} \text{ s.t. } X + Y = A$$

which can be rewritten using $X = A - Y$

$$\min_{X \in \mathbb{R}^{n \times n}} \{\|X - A\|_1 + \tau\|X\|_*\} \ .$$

Hence, RPCA corresponds to $\gamma = 0$ in our formulation. Another standard approach for estimating the low-rank structure of the adjacency matrix consists of minimizing the convex objective with squared loss

$$\min_{X \in \mathbb{R}^{n \times n}} \{\|X - A\|_F^2 + \tau\|X\|_*\} \ .$$

3.2.3 Recovering a partially observed graph

Assume the matrix A is the adjacency matrix of a partially observed graph: entries are 0 for both not-existing and undiscovered links. The search space is unrestricted as before and the matrix X contains the scores for link prediction; the ideal loss function is the empirical average of the zero-one loss for each coefficient

$$\ell_E(X, A) = \frac{1}{|E|} \sum_{(i,j) \in E} 1\{(A_{i,j} - 1/2) \cdot X_{i,j} \leq 0\} \ ,$$

where E is the set of edges in A. However, as in classification, practical algorithms should use a convex surrogate (e.g., the hinge loss).

Given a subset E of observed edges from a graph adjacency matrix $A \in \{0,1\}^{n \times n}$, we set out to predict unobserved links by finding a sparse rank r predictor $X \in \mathbb{R}^{n \times n}$ with small zero-one loss

$$\ell(X, A) = \frac{1}{n^2} \sum_{(i,j) \in \{1,\ldots,n\}^2} 1\{(A_{i,j} - 1/2) \cdot X_{i,j} \leq 0\}$$

by minimizing the empirical zero-one loss $\ell_E(X, A)$. The objective of a generalization bound is to relate $\ell(X, A)$ with $\ell_E(X, A)$. In the case of the sole rank constraint, Srebro [Sre04] remarked that all low-rank matrices with the same sign pattern are equivalent in terms of loss and applied a standard argument for generalization in classes of finite cardinality. In the work

of Srebro, an argument is used to upper bound the number of distinct sign configurations for predictors of rank r

$$s_{\text{lr}}(n,r) = |\{\text{sign}(X) \,|\, X \in \mathbb{R}^{n\times n}, \text{rank}(X) \leq r\}|$$

leading to the following generalization performance: for $\delta > 0$, $A \in \{0,1\}^{n\times n}$ and with probability $1-\delta$ over choosing a subset E of entries in $\{1,\ldots,n\}^2$ uniformly among all subsets of $|E|$ entries, we have for any matrix X of rank at most r and $\Delta(n,r) = \left(\frac{8en}{r}\right)^{2nr}$

$$\ell(X,A) < \ell_E(X,A) + \sqrt{\frac{\log\Delta(n,r) - \log\delta}{2|E|}}\,. \tag{3.4}$$

We consider the class of sparse rank r predictors

$$\mathcal{M}(n,r,s) = \{UV^T \,|\, U,V \in \mathbb{R}^{n\times r},\ ||U||_0 + ||V||_0 \leq s\}$$

and let $s_{\text{splr}}(n,r,s)$ be the number of sign configurations for the set $\mathcal{M}(n,r,s)$. By upper bounding the number of sign configurations for a fixed sparsity pattern in (U,V) using an argument similar to [Sre04], a union bound gives

$$s_{\text{splr}}(n,r,s) \leq \Gamma(n,r,s) = \left(\frac{16en^2}{s}\right)^s \binom{2nr}{s}\,.$$

Using the same notations as previously, we deduce from this result the following generalization bound: with probability $1-\delta$ and for all $X \in \mathcal{M}(n,r,s)$,

$$\ell(X,A) < \ell_E(X,A) + \sqrt{\frac{\log\Gamma(n,r,s) - \log\delta}{2|E|}}\,. \tag{3.5}$$

In general, bound (3.5) is tighter than (3.4) for sufficiently large values of n as shown in the next proposition. The two bounds coincide when $s = 2nr$, that is, when (U,V) is dense and there is no sparsity constraint.

Proposition 1 *For $r_n = n\beta$ with $\beta \in]0,1]$ and $s_n = n\alpha$ with $\alpha \leq 2\beta$,*

$$\frac{\Delta(n,r_n)}{\Gamma(n,r_n,s_n)} = \Omega\left(\left[\frac{8en(\beta n - \alpha)}{(\beta n)^2}\right]^{2n^2\beta}\right)\,,$$

where Ω denotes bounded bellow asymptotically.

The result follows from the application of Stirling's formula. The expression on the right hand side of the equation, giving the asymptotic ratio between the two bounds diverges when n goes to infinity. In particular it shows that the bound for sparse low-rank matrices 3.5 is asymptotically tighter than the bound 3.4 for low-rank (potentially full) matrices.

3.3 Oracle bounds for the estimation of sparse low-rank matrices with quadratic loss

In this section, we present theoretical guarantees for the estimation of sparse low-rank matrices in the presence of noises and using quadratic loss.

In this section we consider the case of linear regression. That is, we consider

$$y = \omega(X_0) + \epsilon \in \mathbb{R}^d$$

with $\epsilon \in \mathbb{R}^d$ having i.i.d zero mean entries and $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$ linear. Define the objective

$$\mathcal{L}(X) = \frac{1}{d}\|\omega(X) - y\|_2^2 + \tau\|X\|_* + \gamma\|X\|_1 \ , \tag{3.6}$$

consider the following estimation procedure

$$\widehat{X} = \arg\min_{X\in\mathcal{S}} \mathcal{L}(X) \ . \tag{3.7}$$

We aim at studying properties of the estimator $\widehat{X}$. Define the matrix $M = \sum_{i=1}^d \epsilon_i \Omega_i$. We begin by stating a simple oracle inequality.

Proposition 3 *Fix a real number $\alpha \in [0,1]$. Under assumption $\tau \geq \frac{2\alpha}{d}\|M\|_{op}$, and $\gamma \geq \frac{2(1-\alpha)}{d}\|M\|_\infty$ if $\widehat{X} = \arg\min_{\mathcal{S}} \mathcal{L}$, we have*

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_2^2 \leq \inf_{X\in\mathcal{S}} \left\{ \frac{1}{d}\|\omega(X - X_0)\|_2^2 + 2\tau\|X\|_* + 2\gamma\|X\|_1 \right\} \tag{3.8}$$

and in particular, when $X_0 \in \mathcal{S}$, taking $X = X_0$ gives

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_2^2 \leq 2\tau\|X_0\|_* + 2\gamma\|X_0\|_1 \ . \tag{3.9}$$

In order to state recovery results on $\widehat{X}$ directly, we introduce an assumption that is similar to the restricted isometry property introduced in [CT04].

Assumption 1 (Restricted Isometry Propery) *There exists $\mu > 0$ such that for any $X_1, X_2 \in \mathcal{S}$*

$$\frac{1}{d}\|\omega(X_1 - X_2)\|_2^2 \geq \mu^{-2}\|X_1 - X_2\|_F^2 \ . \tag{3.10}$$

We can now state the following result, where we set $c_0 = \frac{\sqrt{2}+1}{2}$.

Proposition 4 *Under assumption 1, for any real number $\alpha \in [0,1]$, for constants $\tau \geq \frac{2\alpha}{d}\|M\|_{op}$, and $\gamma \geq \frac{2(1-\alpha)}{d}\|M\|_\infty$, we have*

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_2^2 \leq \inf_{X\in\mathcal{S}} \left\{ \frac{1}{d}\|\omega(X - X_0)\|_2^2 + \frac{\mu^2}{d}\left(c_0\tau\sqrt{\mathrm{rank}(X)} + \gamma\sqrt{\|X\|_0} \right)^2 \right\} \tag{3.11}$$

and in particular if $X_0 \in \mathcal{S}$, taking $X = X_0$ results in

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_2^2 \leq \min\left\{ \frac{\mu^2}{d}\left(c_0\tau\sqrt{\mathrm{rank}(X_0)} + \gamma\sqrt{\|X_0\|_0} \right)^2, 2\tau\|X_0\|_* + 2\gamma\|X_0\|_1 \right\} \tag{3.12}$$

Notice in particular that the bound 3.11 containing the term $c_0\tau\sqrt{\mathrm{rank}(X_0)} + \gamma\sqrt{\|X_0\|_0}$ shows that minimizing the convex surrogate objective $\mathcal{L}$ leads to the optimal solution obtained by minimizing the nonconvex (and exponentially hard to optimize) objective

$$\frac{1}{d}\|\omega(X - X_0)\|_2^2 + \frac{\mu^2}{d}\left(c_0\tau\sqrt{\mathrm{rank}(X)} + \gamma\sqrt{\|X\|_0} \right)^2 \ .$$

An immediate corollary of Propositions 3 and 4 is the following.

Corollary 1 *Under the assumptions of Proposition 4,*

$$\|\widehat{X} - X_0\|_F^2 \leq \mu^2 \min\left\{ \frac{\mu^2}{d}\left(c_0\tau\sqrt{\mathrm{rank}(X_0)} + \gamma\sqrt{\|X_0\|_0}\right)^2, 2\tau\|X_0\|_* + 2\gamma\|X_0\|_1 \right\} . \quad (3.13)$$

In the particular case of denoising $\omega(X) = X$, the constant μ equals $n = \sqrt{d}$, and therefore we have the following result for the minimizer of $\mathcal{J} : S \mapsto \|X - A\|_F^2 + \tau\|X\|_* + \gamma\|X\|_1$ (pay attention to the change of normalization compared to $\mathcal{L}$).

Corollary 2 (Denoising) *Let $\widehat{X}$ be the minimizer of $\mathcal{J}$, and $\alpha \in [0,1]$ a real number, for constants $\tau \geq 2\alpha\|M\|_{op}$, and $\gamma \geq 2(1-\alpha)\|M\|_\infty$, we have*

$$\|\widehat{X} - X_0\|_F^2 \leq \min\left\{ \left(c_0\tau\sqrt{\mathrm{rank}(X_0)} + \gamma\sqrt{\|X_0\|_0}\right)^2, 2\tau\|X_0\|_* + 2\gamma\|X_0\|_1 \right\} \quad (3.14)$$

Assumption 1 can be restrictive. In order to state bounds that hold under less stringent conditions, we introduce the following assumption similar to *restricted eigenvalue conditions* related to those introduced in [BRT09] and [KLT11] that holds in more realistic scenarios than Assumption 1. We first define a cone that contains matrices such that projections onto the singular spaces and onto the sparsity pattern of a matrix S dominate the orthogonal projections, with a trade-off ratio β between the two constraints.

Definition 16 (Cone of restriction) *For a matrix $X \in \mathcal{S} \subset \mathbb{R}^{n\times n}$ and any $\beta, \kappa > 0$, letting $\mathcal{P}_X$ be the projection onto the singular space of X and $\Theta_X = \mathrm{sign}(X)$ the sign pattern of X,*

$$\mathcal{C}(X,\kappa,\beta) = \left\{ B \in \mathcal{S} \;\mid\; \|\mathcal{P}_X^\perp(B)\|_* + \beta\|\Theta_X^\perp \circ B\|_1 \leq \kappa\left(\|\mathcal{P}_X(B)\|_* + \beta\|\Theta_X \circ B\|_1\right) \right\}$$

is the cone of dominant coordinates.

We now relax Assumption 1 by only requiring that it holds over the cone.

Definition 17 (Restricted eigenvalue constant) *For a linear map $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$,*

$$\mu_{\kappa,\beta}(X) = \inf\left\{ \mu' > 0 \;:\; \max(\|\mathcal{P}_X(B)\|_F, \|\Theta_X \circ B\|_F) \leq \frac{\mu'}{d}\|\omega(B)\|_2 \;\; \forall B \in \mathcal{C}(X,\kappa,\beta) \right\} .$$

Assumption 2 (Restricted eigenvalue) *The map $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$ verifies RE assumption κ, β at point X if $\mu_{\kappa,\beta}(X)$ is bounded away from zero.*

We have the following proposition, that holds true under weaker assumptions on ω.

Proposition 5 *Let $\alpha \in [0,1]$. With $\tau > \frac{3\alpha}{d}\|M\|_{op}$ and $\gamma > \frac{3(1-\alpha)}{d}\|M\|_\infty$, we have under Assumption 2*

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_2^2 \leq \inf_{X\in\mathcal{S}}\left\{ \frac{1}{d}\|\omega(X - X_0)\|_2^2 + \frac{\mu_{5,\frac{\gamma}{\tau}}(X)^2}{d}\left(c_1\tau\sqrt{\mathrm{rank}(X)} + c_2\gamma\sqrt{\|X\|_0}\right)^2 \right\} \quad (3.15)$$

and, in particular,

$$\|\omega(\widehat{X} - X_0)\|_2^2 \leq \mu_{5,\frac{\gamma}{\tau}}(X_0)^2\left(c_1\tau\sqrt{\mathrm{rank}(X_0)} + c_2\gamma\sqrt{\|X_0\|_0}\right)^2$$

with $c_1 = \frac{3+2\sqrt{2}}{6}$, $c_2 = \frac{5}{6}$.

The techniques used in the proof (see the Appendix) are very similar to those introduced in [KLT11]. Note that the upper bound interpolates between the results known for trace-norm penalization and the Lasso. In fact, for $\alpha = 0$, τ can be set to zero, and we get a sharp bound for the Lasso, while the trace-norm regression bounds of [KLT11] are obtained for $\alpha = 1$.

In the next Theorem 8, we obtain convergence rates for the procedure 3.7 by combining Proposition 5 with controls on the noise process, by using techniques introduced in [Tro10], see Proposition 8 in appendix. For this, we need to make the assumption that noise entries ϵ_is are independent, zero-mean and subgaussian *i.e.* there exists $\sigma > 0$ such that for any positive real number $\lambda > 0$,

$$\mathbb{E}e^{\lambda\epsilon_i} \leq e^{\sigma^2\lambda^2/2} \ .$$

We introduce two quantities of interest:

$$v^2_{\Omega,\mathrm{op}} = \Big\|\frac{1}{d}\sum_{j=1}^{d}\Omega_j^\top\Omega_j\Big\|_{\mathrm{op}} \vee \Big\|\frac{1}{d}\sum_{j=1}^{d}\Omega_j\Omega_j^\top\Big\|_{\mathrm{op}}, \quad v^2_{\Omega,\infty} = \Big\|\frac{1}{d}\sum_{j=1}^{d}\Omega_j\circ\Omega_j\Big\|_{\infty}$$

which are the (observable) variance terms that naturally appear in the controls of the noise processes.

Theorem 8 *Consider the procedure $\hat{X}$ given by (3.7) with smoothing parameters given by*

$$\tau = 3\alpha\sigma v_{\Omega,\mathrm{op}}\sqrt{\frac{2(t+\log(2n))}{d}}, \quad \gamma = 3(1-\alpha)\sigma v_{\Omega,\infty}\sqrt{\frac{2(t+2\log n)}{d}}$$

for some $\alpha \in (0,1)$ and fix a confidence level $t > 0$. Then, we have

$$\|\omega(\hat{X}-X_0)\|_2^2 \leq \inf_{X\in\mathcal{S}}\Big\{\|\omega(X-X_0)\|_2^2 + 25\mu(X)^2\,\mathrm{rank}(X)\alpha^2\sigma^2 v^2_{\Omega,\mathrm{op}}\frac{2(t+\log(2n))}{d} + 25\mu(X)^2\|X\|_0(1-\alpha)^2\sigma^2 v^2_{\Omega,\infty}\frac{2(t+2\log n)}{d}\Big\}$$

with a probability larger than $1-3e^{-t}$, where $\mu(X) = \mu_{5,\frac{\gamma}{\tau}}(X)$.

3.4 Optimization methods

3.4.1 Preliminary results on proximal operators

We now present how to solve the optimization problem with mixed penalties presented in subsection 2. We consider a loss function $\ell(X, A)$ convex in X. When ℓ is also differentiable in X, we assume that its gradient is Lipschitz with constant L and can be efficiently computed. This is, in particular, the case for the squared Frobenius norm previously mentioned and for other classical choices such as the hinge loss.

We encode the presence of a constraint set $\mathcal{S}$ using the indicator function $1_{\mathcal{S}}(X)$ that is zero when $X \in \mathcal{S}$ and $+\infty$ otherwise, leading to

$$\hat{X} = \underset{X\in\mathbb{R}^{n\times n}}{\arg\min}\ \{\ell(X,A) + \gamma||X||_* + \tau||X||_1 + 1_{\mathcal{S}}(X)\}\ .$$

This formulation involves a sum of a convex differentiable loss and of convex nondifferentiable regularizers which renders the problem non-trivial. A string of algorithms have been

developed for the case where the optimal solution is easy to compute when each regularizer is considered in isolation. Formally, this corresponds to cases where the proximal operator is easy to compute for each regularizer taken separately. We first recall the definition of a proximal operator (Xee [CP11] for a broad overview of proximal methods).

Definition 18 *Let $f : \mathbb{R}^d \to \mathbb{R}$ a convex function, then the proximal operator of f is defined for $x \in \mathbb{R}^d$ by*

$$\mathrm{prox}_f(x) = \underset{z \in \mathbb{R}^d}{\arg\min} \left\{ f(z) + \frac{1}{2}\|z - x\|_2^2 \right\} .$$

The proximal operator of the indicator function is simply the projection onto $\mathcal{S}$, which justifies the alternate denomination of generalized projection operator for prox_R. The proximal operator for the trace norm is given by the shrinkage operation as follows [BT09]. If $Z = U \,\mathrm{diag}(\sigma_1, \cdots, \sigma_n) V^T$ is the singular value decomposition of Z,

$$\mathrm{SHR}_\tau(Z) := \mathrm{prox}_{\tau||\cdot||_*}(Z) = U \,\mathrm{diag}((\sigma_i - \tau)_+)_i V^T .$$

Similarly, the proximal operator for the ℓ_1-norm is the soft thresholding operator

$$\mathrm{ST}_\gamma(Z) := \mathrm{prox}_{\gamma||\cdot||_1} = \mathrm{sgn}(Z) \circ (|Z| - \gamma)_+ .$$

We now generalize this last result to more complex combinations of ℓ_1-norms.

Definition 19 *Let $\alpha_1 < \ldots < \alpha_k$ a sorted sequence of distinct real numbers and $(\beta_1, \ldots, \beta_k)$ nonnegative numbers. We note $\xi_{\alpha,\beta}(x)$ the proximal operator of the function h defined for $x \in \mathbb{R}$ by*

$$h(x, \alpha, \beta) = \sum_{i=1}^{k} \beta_i |x - \alpha_i| .$$

We provide in Lemma 1 below a closed-form expression for $\xi_{\alpha,\beta}$. In particular, $\xi_{\alpha,\beta}$ is continuous and nondecreasing piecewise affine which extends classical results on the proximal operator of the ℓ_1-norm.

Lemma 1 (Proximal operator of h) *Let $\alpha_1 < \ldots < \alpha_k$ a sorted sequence of distinct real numbers, $(\beta_1, \ldots, \beta_k)$ nonnegative numbers, and h defined for $x \in \mathbb{R}$ as in Definition 19 Noting*

$$B_i = \sum_{j=1}^{i} \beta_j - \sum_{j=i+1}^{k} \beta_j ,$$

for $i \in \{0, \ldots, k\}$, the proximal operator of h at $x \in \mathbb{R}$ is

$$\xi_{\alpha,\beta}(x) = \begin{cases} x - B_k & \text{if } x > B_k + \alpha_k \\ x - B_0 & \text{if } x < B_0 + \alpha_1 \\ \alpha_i & \text{if there exists } i \text{ s.t. } x \in [B_{i-1} + \alpha_i; B_i + \alpha_i] \\ x - B_i & \text{if there exists } i \text{ s.t. } x \in [B_i + \alpha_i; B_i + \alpha_{i+1}] . \end{cases} \tag{3.16}$$

Note that since $\alpha_1 + B_0 < \alpha_1 + B_1 < \alpha_2 + B_1 < \alpha_2 + B_2 < \ldots$, there actually exists a single case that applies.

3.4.2 Splitting methods

The family of Forward-Backward splitting methods are iterative algorithms applicable when there is only one nondifferentiable regularizer. These methods alternately take a gradient step over the Lipschitz-differentiable ℓ and and a proximal step over the convex and possibly non-smooth R, leading to updates of the form

$$X_{k+1} = \operatorname{prox}_{\theta R}(X_k - \theta \operatorname{grad}_X \ell(X, A)) \,.$$

In particular, this corresponds to projected gradient descent when R is the indicator function of a convex set.

Douglas-Rachford splitting

We now turn to the question of minimizing

$$\mathcal{L}(X) = \|X - A\|_1 + \tau\|X\|_* + \gamma\|X\|_1 \,.$$

This can be carried out using proximal methods, which are popular methods for nonsmooth convex analysis [CP11].

Douglas-Rachford splitting allows to minimize a sum of two nonsmooth convex functions as long as the proximal operator of each of them can be computed [CP11]. Extensions of the method have been proposed to extend the algorithm to more than two functions but usually at the cost of introducing extra variables [Spi83], which results in a higher computational and memory complexity. Instead of using the proximal of each of the three components of the objective, we propose to apply Douglas-Rachford splitting using the decomposition

$$\mathcal{L}(X) = f(X) + g(X) \text{ with } f(X) = \|X - A\|_1 + \gamma\|X\|_1, \quad g(X) = \tau\|X\|_* \,.$$

Since $f(X)$ is separable with respect to the coefficients $(X_{i,j})$, the proximal operator of f can be computed coordinate-by-coordinate as $\operatorname{prox}_f(X) = (\operatorname{prox}_f(X_{i,j}))_{i,j} \in \mathbb{R}^{n\times n}$. For each coordinate, this reduces to computing the proximal operator of $x \mapsto \gamma|x| + |x - a|$ which can easily be computed using Lemma 1. In particular, this corresponds to the proximal operator of $\xi_{\alpha,\beta}$ with $\alpha = (0, a)$, $\beta = (\gamma, 1)$ when $a \geq 0$, and to $\alpha = (a, 0)$, $\beta = (1, \gamma)$ when $a < 0$.

For convenience in the exposition of the algorithm, we define for $X \in \mathbb{R}^{n\times n}$ the reflexive proximal of a function h as

$$\operatorname{rprox}_h(X) = 2\ \operatorname{prox}_h(X) - X$$

and $P_{\mathcal{S}}$ denotes orthogonal projection onto the convex set $\mathcal{S}$. The pseudocode oh the algorithm based on the Douglas-Rachford scheme is shown in Algorithm 1.

Algorithm 1 Douglas-Rachford Scheme for Graph Denoising

Input $A, \tau, \gamma > 0$, $\theta \in (0, 2)$
Initialization: $Z^{(0)} = A$

for $k = 1, 2, \ldots, K$ **do**
  $Z^{(k)} = \left(1 - \frac{\theta}{2}\right) Z^{(k-1)} + \frac{\theta}{2} \operatorname{rprox}_{\tau\|\cdot\|_*} \circ \operatorname{rprox}_f(Z^{(k-1)})$
end for

$\widehat{X} = \operatorname{prox}_{\tau\|\cdot\|_*}(Z^{(K)})$

Generalized Forward-Backward splitting

A generalization of the two setups (Lipschitz-differentiable + easy proximal) and (easy proximal + easy proximal) has been recently proposed in [RFP11] under the name of Generalized Forward-Backward, which we specialize to our problem in Algorithm 2. The proximal operators are applied in parallel, and the resulting (Z_1, Z_2, Z_3) is projected onto the constraint that $Z_1 = Z_2 = Z_3$ which is given by the mean. The auxiliary variable Z_3 can be simply dropped when $\mathcal{S} = \mathbb{R}^{n \times n}$. The algorithm converges under very mild conditions when the step size θ is smaller than $\frac{2}{L}$.

Algorithm 2 Generalized Forward-Backward

Initialize $X, Z_1, Z_2, Z_3 = A$, q = 3
repeat
  Compute $G = \nabla_X \ell(X, A)$.
  Compute $Z_1 = \mathrm{prox}_{q\theta\tau||.||_*}(2X - Z_1 - \theta G)$
  Compute $Z_2 = \mathrm{prox}_{q\theta\gamma||.||_1}(2X - Z_2 - \theta G)$
  Compute $Z_3 = P_{\mathcal{S}}(2X - Z_3 - \theta G)$
  Set $X = \frac{1}{q}\sum_{k=1}^{q} Z_k$
until convergence
return X

Incremental Proximal Descent

Although Algorithm 2 performs well in practice, the $O(n^2)$ memory footprint with a large leading constant due to the parallel updates can be a drawback in some cases. As a consequence, we mention a matching serial algorithm (Algorithm 3) introduced in [Ber11] that has a flavor similar to multi-pass stochastic gradient descent, and has a convergence rate in $O(\frac{1}{\epsilon}^2)$ to an approximate solution in a radius of $\epsilon > 0$. This convergence rate is slow compared to the standard proximal gradient descent that converges in $O(\frac{1}{\epsilon})$, or $O(\frac{1}{\sqrt{\epsilon}})$ for the accelerated version. We present here a version where updates are performed according to a cyclic order, although random selection of the order of the updates is also possible.

Algorithm 3 Incremental Proximal Descent

Initialize $X = A$
repeat
  Set $X = X - \theta \nabla_X \ell(X, A)$
  Set $X = \mathrm{prox}_{\theta\tau||.||_*}(X)$
  Set $X = \mathrm{prox}_{\theta\gamma||.||_1}(X)$
  Set $X = P_{\mathcal{S}}(X)$
until convergence
return S

PSD constraint

For any positive semidefinite matrix, we have $||Z||_* = \mathrm{Tr}(Z)$. The simple form of the trace norm allows to take into account the positive semidefinite constraint at no additional cost, as the shrinkage operation and the projection onto the convex cone of positive semidefinite matrices can be combined into a single operation.

Lemma 2 *For $\tau \geq 0$ and $X \in \mathbb{R}^{n\times n}$, if $P_{S_n^+}$ denotes the projection onto the PSD cone,*

$$\begin{aligned}\mathrm{prox}_{\tau||\cdot||_*+1_{S_n^+}}(X) &= \arg\min_{Z\succeq 0} \frac{1}{2}||Z-X||_F^2 + \tau||Z||_* \\ &= P_{S_n^+}\left(X - \tau I_n\right) .\end{aligned}$$

The solution of this problem is given by soft-thresholding the singular values of A and is known as the shrinkage operator (e.g., [CP11]). This is similar to the best rank-r approximation of a matrix in Eucliean norm that is obtained through hard-thresholding of the singular values.

3.4.3 Fast methods using factorization-based updates

The method proposed in Algorithm 1 is slow due to full SVD computations at every iteration in order to compute the proximal operator of the nuclear norm. Given the size of real data sets, there is a real need for exploring more scalable algorithms. It is well known [SRJ05] that the nuclear norm can be characterized as

$$\|X\|_* = \frac{1}{2} \min_{X=UV^\top, U,V\in\mathbb{R}^{n\times n}} \left\{\|U\|_F^2 + \|V\|_F^2\right\} .$$

Motivated by this characterization, number of approaches have been developed [SRJ05] for matrix estimation subject to rank constraint without explicitly penalizing the objective with the nuclear norm. Most of these approaches optimize for factorizations of the form $X = UV^\top$ and address the case of the squared loss $\|UV^T - A\|_F^2$ with penalization proportional to $\|U\|_F^2 + \|V\|_F^2$. The motivation for such a factorization rather the convex methods is

1. *Good empirical results.* See for instance [Kor08].
2. *Attracting scalability.* In fact, as clearly argued in [RR11], optimizing factors leads to scalable methods both in terms of memory requirement and computational complexity, and may be parallelized easily.
3. *Interpretability.* The factors U and V can be interpreted as the latent features describing individuals nodes. Nonnegative matrix factorization (see for instance [KP08]) encourages further interpretability by imposing nonnegativity to the entries of the factors.

Nevertheless it should be pointed out that this method leads to nonconvex objectives. Alternating minimization schemes are usually developed, where one alternates minimization over U with minimization over V. This method has no theoretical guarantee to converge to a global optimum, but empirical success has been reported in many experiments.

Sparse factorizations

Sparse matrix factorizations have been proposed in sparse principal component analysis (SPCA) to extend the classical principal component analysis and find sparse directions that maximize the variance of the projection by penalizing the sparsity of the principal components [ZHT04]. Similarly, [KP08] have proposed in the context of nonnegative matrix factorization to encourage sparsity of the factors U and V by penalizing $\|U\|_1$ and $\|V\|_1$ instead of directly enforcing the sparsity of X by penalizing by $\|X\|_1$. A simple baseline for graph denoising is thus the following squared-loss objective for sparse factorization:

$$\mathcal{L}_2(U,V) = \|UV^\top - A\|_F^2 + \frac{\tau}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) + \gamma\left(\|U\|_1 + \|V\|_1\right) . \tag{3.17}$$

As we already outlined, the squared loss is not adequate for sparse noise. We propose to use a factorization method along these lines to find an approximate solution to the optimization problem defined at Equation (3.3). The surrogate objective we consider is hence

$$\mathcal{L}_{UV}(U,V) = \|UV^\top - A\|_1 + \frac{\tau}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) + \gamma\left(\|U\|_1 + \|V\|_1\right) .$$

The penalty on U and V can also be interpreted as the elastic-net penalty [ZH05] over the factors U and V. Alternating minimization of $\mathcal{L}_{UV}$ is more complex than with traditional alternating minimization approaches in the presence of the squared loss, and we will now present a procedure to optimize $\mathcal{L}_{UV}$ in $U, V \in \mathbb{R}^{n\times r}$ for some fixed r.

Alternating minimization with rank one updates

Recently, [DHM12, SSGO11] have suggested to recover low-rank matrices by successively finding rank-one updates minimizing a given loss, while [DEGJL07] introduced a method for SPCA using successive sparse rank-one updates. In a similar fashion, we propose to use successive sparse rank one updates minimizing the following objective in $(u, v) \in \mathbb{R}^n$:

$$\ell(u,v) = \|uv^\top - A\|_1 + \frac{\tau}{2}(\|u\|_2^2 + \|v\|_2^2) + \gamma(\|u\|_1 + \|v\|_1) .$$

We then deflate the original A by uv^T and repeat the procedure r times where r is a fixed target maximum rank.

Note that for $\tau > 0$ the functional ℓ is strictly convex in each variable u or v, the other one being kept fixed. Although ℓ is also jointly convex in (u, v) for $\tau > 2$, the joint minimization is hard to carry out, and we suggest to perform a regularized altering minimization in u and v instead. More precisely, we consider sequences $(u^{(k)}, v^{(k)})_k$ defined by

$$\begin{aligned} u^{(k+1)} &= \arg\min_{u\in\mathbb{R}^n} \left\{ \|uv^{(k)\top} - A\|_1 + \gamma\|u\|_1 + \frac{1}{2\theta}\|u - (1-\theta\tau)u^{(k)}\|_2^2 \right\} \\ v^{(k+1)} &= \arg\min_{v\in\mathbb{R}^n} \left\{ \|u^{(k+1)}v^\top - A\|_1 + \gamma\|v\|_1 + \frac{1}{2\theta}\|v - (1-\theta\tau)v^{(k)}\|_2^2 \right\} . \end{aligned}$$

for some $\theta > 0$. Each update corresponds to one step of proximal gradient descent [CP11] with respect to each variable, while the other one is kept fixed. Notice that for $\theta = 1/\tau$, there is no extra regularization and this is simply a standard alternate minimization of ℓ. In practice, we choose $\theta \in (0, 1/\tau)$ as this exhibited empirical superiority to the standard alternate minimization in numerical experiments, as shown in Table 3.3. In addition to this, we initialize the alternate minimization using the left/right singular vectors corresponding to the highest singular value. The full procedure is detailed in Algorithm 4.

In the following, we sketch the computations for the minimization with respect to u and point out that equivalent formulas can be obtained by swapping u and v and replacing A by

Algorithm 4 Factorization-based Graph Denoising with Rank-1 updates

Input $A, \tau, \gamma > 0$, $\theta < 1/\tau$, $r \leq n, T$: number of iterations
Initialization: $Z^{(0)} = 0$

for $k = 1, 2, \ldots, r$ **do**
  Initialize $u^{(k)}, v^{(k)}$

  for $i = 1, \cdots, T$ **do**
    $u^{(k)} = \arg\min_{u \in \mathbb{R}^n} \ell(u, v^{(k)})$
    $v^{(k)} = \arg\min_{v \in \mathbb{R}^n} \ell(u^{(k)}, v)$
  end for

  $Z^{(k)} = Z^{(k-1)} + u^{(k)} v^{(k)^T}$
  $A = A - u^{(k)} v^{(k)^T}$
end for

$\widehat{X} = Z^{(r)}$

$A^\top$. For computing $u^{(k+1)}$, first notice that

$$\begin{aligned}
&\|uv^{(k)^\top} - A\|_1 + \gamma\|u\|_1 + \frac{1}{2\theta}\|u - (1-\theta\tau)u^{(k)}\|_2^2 \\
&\quad = \sum_{i=1}^{n} \left[\sum_{j=1}^{n} |u_i v_j^{(k)} - A_{i,j}| + \gamma|u_i| + \frac{1}{2\theta}(u_i - (1-\theta\tau)u_i^{(k)})^2 \right] \\
&\quad = \sum_{i=1}^{n} \left[\sum_{j \mid v_j^{(k)} \neq 0} |v_j^{(k)}||u_i - \frac{A_{i,j}}{v_j^{(k)}}| + \gamma|u_i| + \frac{1}{2\theta}(u_i - (1-\theta\tau)u_i^{(k)})^2 \right] + C \\
&\quad = \frac{1}{\theta}\sum_{i=1}^{n} \left[\sum_{j \mid v_j^{(k)} \neq 0} \theta|v_j^{(k)}||u_i - \frac{A_{i,j}}{v_j^{(k)}}| + \theta\gamma|u_i| + \frac{1}{2}(u_i - (1-\theta\tau)u_i^{(k)})^2 \right] + C\,.
\end{aligned}$$

The term

$$C = \sum_{j \mid v_j^{(k)} = 0} |A_{i,j}|$$

does not depend on u. We have a sum (over i) of terms of the form $\sum_k \beta_k |u_i - \alpha_k| + \frac{1}{2\theta}(u_i - \delta)^2$, and we can thus use Lemma 1 to find the minimizer of this expression with respect to u_i. The vector $u^{(k)}$ can hence be updated by applying Lemma 1 for each i at the point

$$x = (1 - \theta\tau)u_i^{(k)}$$

and using

$$\{\alpha_j\} = \{A_{i,j}/v_j^{(k)},\ j \text{ s.t. } v_j^{(k)} \neq 0\} \cup \{0\},\ \{\beta_j\} = \{\theta|v_j^{(k)}|,\ j \text{ s.t. } v_j^{(k)} \neq 0\} \cup \{\theta\gamma\}$$

where we used a set notation as in practice, for each i, the coefficients of α_i and β_i must be sorted by increasing value of $A_{i,j}/v_j^{(k)}$, that is, we must find an injection

$$\pi : \{1, \cdots, q\} \to \{1, \cdots, n\}$$

by sorting the q non-zeros elements of $v^{(k)}$ such that $A_{i,\pi(1)}/v^{(k)}_{\pi(1)} < \cdots < A_{i,\pi(q)}/v^{(k)}_{\pi(q)}$. Each iteration thus has a cost of $O(\|u\|_0\|v\|_0 \log \|v\|_0 + \|v\|_0\|u\|_0 \log \|u\|_0)$, which leads to an efficient procedure for sparse u and v in

$$O\left(r\,T\left\{\|u\|_0\|v\|_0 \log \|v\|_0 + \|v\|_0\|u\|_0 \log \|u\|_0\right\}\right) \quad .$$

The terms $\|.\|_0$ should be read as the average sparsity over the iterations. The runtime of the algorithm in practice highly depends on the sparsity of the initial vectors $u^{(k)}$ and $v^{(k)}$. It can considerably be reduced by initializing for each k, $u^{(k)}$ and $v^{(k)}$ with sparse vectors. In our experiments we initialized them with thresholded first singular vectors or with normalized hard-thresholded degrees of the nodes. Further experiments are needed to understand the dependency of the solution on the initialization.

Remark 1 (Penalizing the product $\|u\|\|v\|$ rather than the sum $\|u\| + \|v\|$) *It is known [Jam87] that the function of X defined by the value*

$$\inf\left\{\sum_{i=1}^{\infty}\|u_i\|_L\|v_i\|_R \;\middle|\; u_i, v_i \in \mathbb{R}^n \;\; s.t. \;\; X = \sum_{i=1}^{\infty} u_i v_i^\top\right\}$$

for any pair of norms $\|.\|_L$ and $\|.\|_R$ is a norm. If both the norms are the ℓ_1 norm, then we obtain the standard element-wise ℓ_1 norm of the matrix:

$$\inf\left\{\sum_{i=1}^{\infty}\|u_i\|_1\|v_i\|_1 \;\middle|\; u_i, v_i \in \mathbb{R}^n \;\; s.t. \;\; X = \sum_{i=1}^{\infty} u_i v_i^\top\right\} = \sum_{i,j}|X_{i,j}| \; .$$

In order to use this observation, rather than minimizing ℓ, one may rather want to minimize the non-convex functional

$$\widetilde{\ell}(u,v) = \|uv^\top - A\|_1 + \frac{\tau}{2}(\|u\|_2^2 + \|v\|_2^2) + \gamma\,\|u\|_1\|v\|_1 \; .$$

Note that the same Algorithm 4 applies by simply replacing the inside loop iterations to

$$u^{(k)} = \arg\min_{u\in\mathbb{R}^n} \widetilde{\ell}(u, v^{(k)})$$

and

$$v^{(k)} = \arg\min_{v\in\mathbb{R}^n} \widetilde{\ell}(u^{(k)}, v) \quad .$$

The minimizer of $\widetilde{\ell}$ can be obtained from the minimizer of ℓ (see Lemma 1) by replacing γ by $\gamma\|v\|_1$. As we are using only $r < \infty$ factors the optimized functional does not have a convex equivalent in the same way as the trace norm is the equivalent of using sum of squares of Frobenius norms. This does not cure the nonconvexity of the function

$$\mu(x) = \inf\left\{\|u\|_1 + \|v\|_1 \;\middle|\; x = uv^\top\right\} \quad .$$

To see why it is nonconvex, notice that for $n = 1$ one gets the square root $\sqrt{|x|}$. In the numerical experiments we have not yet seen a significant difference between the results of the two methods. An issue with these methods, due to nonconvexity, is the existence of non-optimal stationary points such as $(0,0)$.

Remark 2 (Rank r updates) *Instead of the objective $\ell(u, v)$ of two vector variables, one can use a similar function $\mathcal{L}_{UV}(U, V)$ of matrices $U, V \in \mathbb{R}^{n \times r}$. Minimizing following each single coordinate variable $u_{i,j}$ requires the same tools as minimizing $\ell(u, v)$, so one can perform coordinate descent on $\mathcal{L}_{UV}$ as well. The comparative study of several competing methods for sparse matrix factorization is left to future work.*

3.5 Numerical experiments

We present numerical experiments to highlight the benefits of our method. For efficiency reasons, we use the incremental proximal descent algorithm (Algorithm 3).

3.5.1 Covariance estimation and graph denoising with Frobenius norm loss

Synthetic data

Covariance matrix estimation. We draw N vectors $x_i \sim \mathcal{N}(0, \Sigma)$ for a block diagonal covariance matrix $\Sigma \in \mathbb{R}^{n \times n}$. We use r blocks of random sizes and of the form $vv^\top$ where the entries of v are drawn i.i.d. from the uniform distribution on $[-1, 1]$. Finally, we add gaussian noise $\mathcal{N}(0, \sigma^2)$ on each entry. In our experiments $r = 5$, $N = 20$, $n = 100$, $\sigma = 0.6$. We apply our method (SPLR), as well as trace norm regularization (LR) and ℓ_1 norm regularization (SP) to the empirical covariance matrix, and report average results over ten runs. Figure 3.1 shows the RMSE normalized by the norm of Σ for different values of τ and γ. Note that the effect of the mixed penalty is visible as the minimum RMSE is reached inside the (τ, γ) region. We perform, on the same data, separate cross-validations on (τ, γ) for SPLR, on τ for LR and on γ for SP. We show in Figure 3.2 the supports recovered by each algorithm, the output matrix of LR being thresholded in absolute value. The support recovery demonstrates how our approach discovers the underlying patterns despite the noise and the small number of observations.

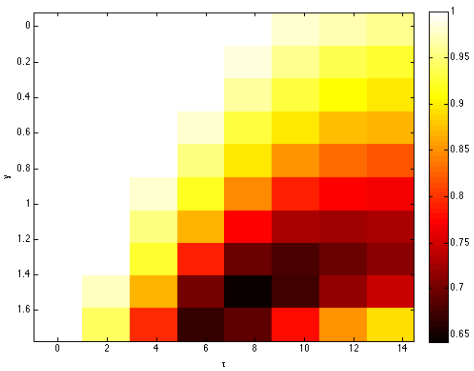


Figure 3.1: Covariance estimation. Cross-validation: normalized RMSE scores (SPLR)

Real data sets

Protein Interactions. We use data from [HJB+09], in which protein interactions in Escherichia coli bacteria are scored by strength in $[0, 2]$. The data is, by nature, sparse. In addition to this, it is often suggested that interactions between two proteins are governed by a small set of factors, such as surface accessible amino acid side chains [BG01], which motivates the estimation of a

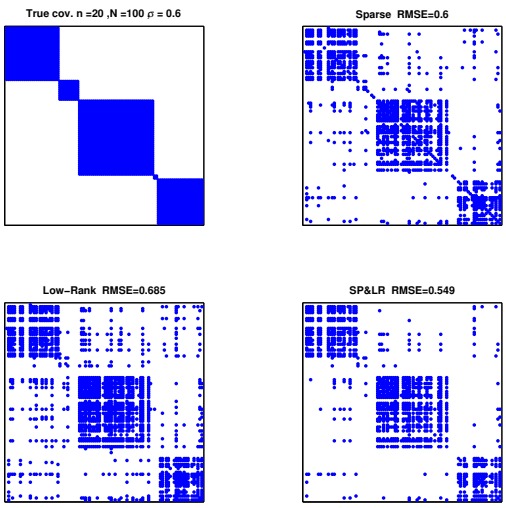


Figure 3.2: Covariance estimation. Support of Σ (top left), and of the estimates given by SP (top right), LR (bottom left), and SPLR (bottom right)

low-rank representation. Representing the data as a weighted graph, we filter to retain only the 10% of all 4394 proteins that exhibit the most interactions as measured by weighted degree. We corrupt 10% of entries of the adjacency matrix selected uniformly at random by uniform noise in $[0, \eta]$. Parameters are selected by cross-validation and algorithms are evaluated using mean RMSE between estimated and original adjacency matrices over 25 runs. RMSE scores are shown in Table 3.1 and show the empirical superiority of our approach (SPLR).

η	SPLR	LR	SP
0.1	**0.0854** $\pm$0.012	0.1487 $\pm$0.02	0.1023 $\pm$0.02
0.2	**0.2073** $\pm$ 0.03	0.2673 $\pm$ 0.3	0.2484 $\pm$ 0.03
0.3	**0.3105** $\pm$ 0.03	0.3728 $\pm$ 0.03	**0.3104** $\pm$ 0.02

Table 3.1: Prediction of interactions in Escherichia coli. Mean normalized RMSE and standard deviations.

Social Networks. We have performed experiments with the Facebook100 data set analyzed by [TMP11]. The data set comprises all friendship relations between students affiliated to a specific university, for a selection of one hundred universities. We select a single university with 41554 users and filter as in the previous case to keep only the 10% users with highest degrees. In this case, entries are corrupted by impulse noise: a fixed fraction σ of randomly chosen edges are flipped, thus introducing noisy friendship relations and masking some existing relations. The task is to discover the noisy relations and recover masked relations. We compare our method to standard baselines in link prediction [LNK07b]. Nearest Neighbors (NN) relies on the number of common friends between each pair of users, which is given by A^2 when A is the noisy graph adjacency matrix. Katz's coefficient connects a pair of nodes according to a score based on the number of paths connecting them, emphasizing short paths. Results are reported in Table 3.2 using the area under the ROC curve (AUC). SPLR outperforms LR but also NN and Katz which do not directly seek a low-rank representation.

σ	SPLR	LR	NN	Katz
5 %	0.9293	0.9291	0.7680	**0.9298**
10 %	**0.9221**	0.9174	0.7620	0.9189
15 %	**0.9117**	0.9024	0.7555	0.9068
20 %	**0.8997**	0.8853	0.7482	0.8941

Table 3.2: Facebook denoising data. Mean AUC over 10 simulation runs. All standard deviations are lower than $3 \cdot 10^{-4}$.

3.5.2 Graph denoising with ℓ_1 norm loss

Regularization paths

The model parameters τ, γ can be tuned using cross-validation in a supervised setting. In an exploratory perspective, one may rather look at the sparsity and rank of the solution as a function of τ and γ. In particular, exploring the solutions of the optimization problem defined by Equation (3.3) for various parameters clearly shows the benefit of our method compared to RPCA. In the left panel of Figure 3.3, we show the regularization path of RPCA for various values of the parameter τ. We insist on the fact that using this unique parameter does not allow to achieve different levels of sparsity and rank independently. The right panel of Figure 3.3 shows on an example that by choosing nonzero values for both τ and γ, we achieve better support recovery: top left shows the support of A_0 in blue and of the noise in red, while bottom left corresponds to RPCA and bottom right to Convex Graph Denoising. We point out that for $\tau = 0$, nonzero values of γ do not lead to any interesting solution (top right) as separating sparse noise from sparse signal without any additional prior is meaningless.

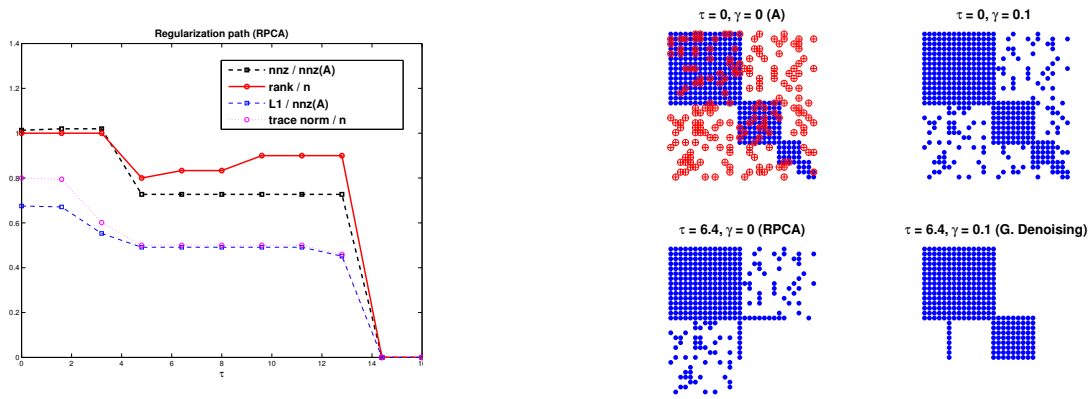


Figure 3.3: Regularization path for RPCA (left), sparsity patterns of solutions for different values of the parameters (right)

Phase transition

We analyzed the influence of the characteristics of the noise and of A_0 on the performance of Factorization-based Graph Denoising. For different values of the sparsity of the noise, of the sparsity of A_0 and of the rank of A_0, we report the mean relative ℓ_1 error after 50 iterations of cross-validation on the parameters of the algorithm.

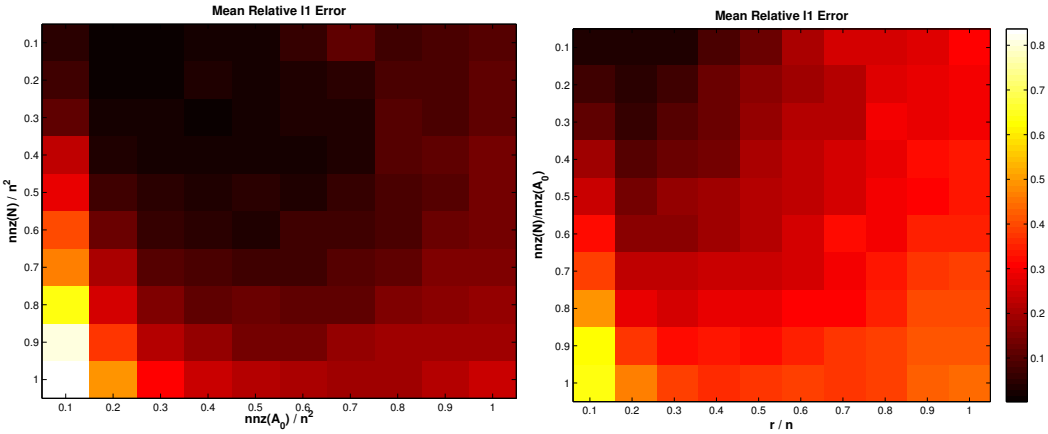


Figure 3.4: Influence of A_0 and N on Factorization-based Graph Denoising

Figure 3.4 (left panel) shows the performance of the algorithm when the sparsities of A_0 and of the noise vary while Figure 3.4 (right panel) shows the performance when the rank r of A_0 and the sparsity of the noise vary. Unsurprisingly, the recovery is all the more accurate than the noise is sparse and A_0 low-rank. Level sets take the form of diagonal lines which indicate the following tradeoff: to maintain an equal recovery error when the density of the noise increases, the rank of A_0 must decrease, or equivalently, A_0 must become more structured.

Empirical evaluation

Synthetic data. We generate synthetic data $A = A_0 + N$, where $A_0 = U_0 V_0^\top$. The two matrices $V_0 \in \mathbb{R}^{n\times r}$ and $N \in \mathbb{R}^{n\times n}$ have sparsity patterns chosen uniformly at random and their entries are drawn i.i.d. from the uniform distribution $\mathcal{U}(-1, 1)$. The matrix U_0 has a one on each row at a randomly chosen position. We point out that using this construction, A_0 is both sparse and low-rank, N is sparse and has high rank, and both have entries of with the same order of magnitude.

In our experiments, the number of nodes is $n = 100,\ r = 6,\ \|N\|_0 = 200,\ \|A_0\|_0 = 350$. We performed 10-fold cross validation for choosing the values of the parameters τ and γ, which not surprisingly turn out to be the very close for all the competitors. We then evaluated the algorithms through 150 test iterations on data with the same characteritics. We report in Table 3.3 the performance of the algorithms in terms of relative ℓ_1 error $\|\widehat{A} - A_0\|_1/\|A_0\|_1$ (mean absolute deviation), relative ℓ_2 error $\|\widehat{A} - A_0\|_F/\|A_0\|_F$ and relative ℓ_0 error $\|\widehat{A} - A_0\|_0/\|A_0\|_0$.

We present results with Convex Graph Denoising (Algorithm 1), Factorization-based Graph Denoising (Algorithm 4) with and without regularization, Robust Principal Component Analysis (RPCA, [CLMW09]) implemented using Douglas-Rachford splitting, a dual implementation of RPCA ([LGW+09]), sparse matrix factorization corresponding to Equation (3.17), and shrinkage.

Convex Graph Denoising achieves the best results both in ℓ_1- and ℓ_2-norm. Factorization-based Graph Denoising achieves slightly inferior results but still performs second to best at a smaller computation cost (see Figure 3.5). In addition to this, Factorization-based Graph Denoising achieves a better recovery of the support. Due to the lack of explicit regularization of the sparsity of X, RPCA and shrinkage perform poorly. The sparse factorization method of Equation ((3.17)) also achieves poor results due to the use of the squared loss.

Figure 3.5 shows the computation time for Convex Graph Denoising and Factorization-

	rel. ℓ_2 error	rel. ℓ_1 error	rel. ℓ_0 error
Convex Graph Denoising	$\mathbf{0.173 \pm 0.012}$	$\mathbf{0.056 \pm 0.008}$	0.903 ± 0.040
Fact. Graph Denoising	0.241 ± 0.017	0.139 ± 0.017	$\mathbf{0.351 \pm 0.022}$
Fact. Graph Denoising, no reg.	0.292 ± 0.021	0.191 ± 0.022	0.359 ± 0.023
RPCA	0.970 ± 0.406	1.029 ± 0.556	2.119 ± 0.121
RPCA (dual)	0.699 ± 0.246	2.405 ± 1.081	123.299 ± 7.801
sparse factorization, eq. (3.17)	1.042 ± 0.417	1.619 ± 0.718	5.051 ± 0.307
shrinkage	0.937 ± 0.007	0.937 ± 0.007	39.279 ± 3.804

Table 3.3: Relative errors and standard deviations for graph denoising (see text for detail)

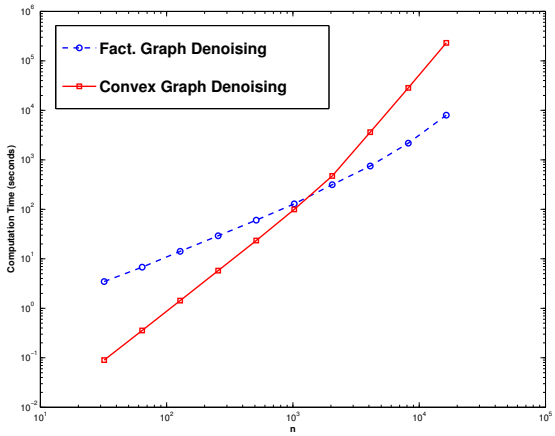


Figure 3.5: Computation time

based Graph Denoising when the size of the graph varies. For small values of n, the overhead due to the alternate minimization makes the factorization approach more expensive than the convex formulation. For larger values of n, however, the factorization approach gradually becomes more and more efficient.

Real data. We performed empirical evaluation of our Douglas-Rachford algorithm for graph denoising over a subset of the e-commerce network data for comparing its predictive performance with baseline link prediction approaches. The data set contains the purchase history of the 1000 most popular music items by the 1000 most active users. Their purchases are tracked over a period of 10 months that constitute the training set and the 11th month purchases where subject to prediction. We compared the output of the algorithm with the link prediction scores computed by nearest neighbors, Katz algorithm and a low-rank approximation of the training adjacency matrix. Figure 3.6 contains the precision and recall scores of the different algorithms and shows empirical superiority of our algorithm on this recommender system task.

3.6 Discussion

- **Optimization.** Other optimization techniques can be considered for future work. A trace norm constraint alone can be taken into account without projection or relaxation into a penalized form by casting the problem as a SDP as proposed in [Jag11]. The special form of this SDP can be leveraged to use the efficient resolution technique from [Haz08].

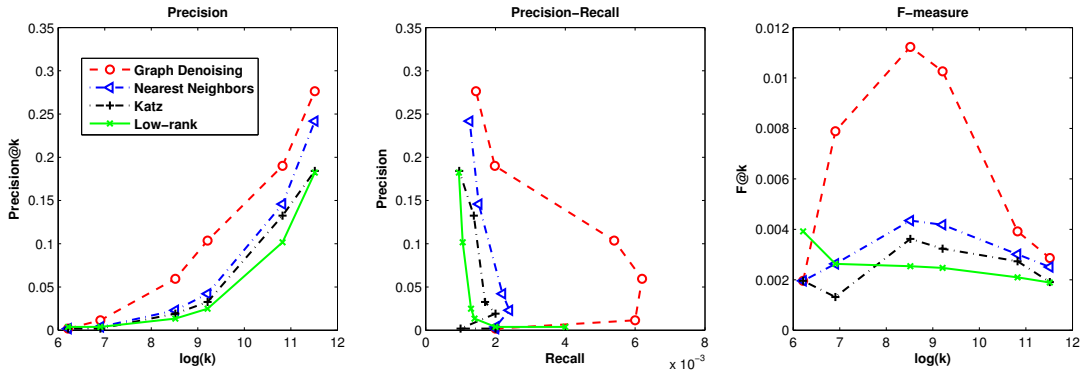


Figure 3.6: Graph denoising for recommender system application: precision-recall curves.

This method applies to a differentiable objective whose curvature determines the performances. Extending these methods with projection onto the ℓ_1 ball or a sparsity-inducing penalty could lead to interesting developments.

- **Generalization of the approach** - The methods presented in this chapter can be seamlessly extended to nonsquare matrices, which can arise, for instance, from adjacency matrices of bipartite graphs. Our work also applies to a wide range of other losses. A useful example that links our work to the matrix completion framework is when linear measurements of the target matrix or graph are available, or can be predicted as in [RBEV10, RSV12]. In this case, the loss can be defined in the feature space, and we treat this in Chapter 4 of this manuscript. Due to the low-rank assumption, our method does not directly apply to the estimation of precision matrices often used for gaussian graphical model structure learning [FHT08], and the applications of conditional independence structures generated by low-rank and possibly sparse models is to be discussed. The suggested approach can be seen as a method to estimate *sparse* matrices whose nonzero coordinates are structured in specific *low-rank* patters. When the groups of simultaneously active or when hierarchy of active variables are known in advance, a family of approaches called structured sparsity [BJMO11, JMOB11] generalize the Lasso by penalizing the sum of ℓ_2 norm of groups. The first approach dedicated to estimate sparse vectors under strong variable correlation assumption was the Elastic Net [ZH05] that uses the hybrid $\ell_1 + \epsilon\ell_2$ penalty. The trace-Lasso [GOB11] builds a more powerful regularizer for sparse regression under highly correlated designs by inserting the covariate matrix X into the regularizer, and penalizes $\|(X^\top X)^{1/2}\operatorname{diag}(w)\|_*$ for the variable w. In some applications we have a general prior knowledge of how the variables will be organized, but do not know in advance any specific labeling / ordering / grouping / hierarchy of the variables. In the applications discussed in this chapter, as community detection or clustering, we saw how the block-dioagonal *in some permuted basis* assumption was translated onto a low-rank assumption. We enforced this structural assumption on the estimation procedure by overlapping a trace-norm over the ℓ_1 of the same matrix variable. It would be very interesting to explore methods for estimating vectors / matrices under other types of prior knowledge. For instance estimating adjacency matrices of graphs containing tree-like hierarchical structures has potential interesting applications.

Chapter 4

Graph prediction in a temporal setting

4.1 Context

Forecasting systems behavior with multiple responses has been a challenging issue in many contexts of applications such as collaborative filtering, financial markets, or bioinformatics, where responses can be, respectively, movie ratings, stock prices, or activity of genes within a cell. Statistical modeling techniques have been widely investigated in the context of multivariate time series either in the multiple linear regression setup [BF97] or with autoregressive models [Tsa05]. More recently, kernel-based regularized methods have been developed for multitask learning [EMP05, APMY07]. These approaches share the use of the correlation structure among input variables to enrich the prediction on every single output. Often, the correlation structure is assumed to be given or it is estimated separately. A discrete encoding of correlations between variables can be modeled as a graph so that learning the dependence structure amounts to performing graph inference through the discovery of uncovered edges on the graph. The latter problem is interesting *per se* and it is known as the problem of link prediction where it is assumed that only a part of the graph is actually observed [LNK07b, KX11]. This situation occurs in various applications such as recommender systems, social networks, or proteomics, and the appropriate tools can be found among matrix completion techniques [SRJ05, CT09, ABEV09]. In the realistic setup of a time-evolving graph, matrix completion was also used and adapted to take into account the dynamics of the features of the graph [RBEV10]. The estimation of a VAR model for node degrees (that are linear graph features) has been considered by [ZEPR11], and successfully applied to customer valuation, and to measure network effect in user generated content market places. In this work, we study the prediction problem where the observation is a sequence of graphs adjacency matrices $(A_t)_{0\leq t\leq T}$ and the goal is to predict A_{T+1}. This type of problem arises in applications such as recommender systems where, given information on purchases made by some users, one would like to predict future purchases. In this context, users and products can be modeled as the nodes of a bipartite graph, while purchases or clicks are modeled as edges. In functional genomics and systems biology, estimating regulatory networks in gene expression can be performed by modeling the data as graphs and fitting predictive models is a natural way for estimating evolving networks in these contexts. A large variety of methods for link prediction only consider predicting from a single static snapshot of the graph - this includes heuristics [LNK07b, SCM10b], matrix factorization [Kor08], diffusion [ML10], or probabilistic methods [TWAK03]. More recently, some works have investigated using sequences of observations of the graph to improve the prediction, such as using regression on features extracted from the graphs [RBEV10], using matrix factorization [Kor10], continuous-time regression [VAHS11] or non-parametric models [SCJ12]. In fact link prediction can also be seen as a special case of matrix completion where

the goal is to estimate the missing entries of the adjacency matrix of the graph where the entries can be only "0s" and "1s". The ratings of matrix completion provide a richer information than the binary case (existence or absence of a link). Consider for instance the issue of link prediction in recommender systems. In that case, we consider a bipartite graph for which the vertices represent products and users, and the edges connect users with the products they have purchased in the past. The setup we consider in the present work corresponds to the binary case where we only observe purchase data, say the presence of a link in the graph, without any score or feedback on the product for a given user. The fundamental different in this case is that no *negative feedback* is observed. By definition, the non-observation of a link can either mean negative label or unknown label. Hence, we will deal here with the situation where the components of snapshots of the adjacency matrix only consist of "1s" and missing values. As opposite to matrix completion where the sparsity pattern of observation is supposed to by chosen at random and contain no information, all the information in the case of link prediction remains in the sparsity pattern and the value of the entries (all equal to 1) are not relevant. Our main assumption is that the network effect is a cause and a symptom at the same time, and therefore, the edges and the graph features should be estimated simultaneously. We propose a regularized approach to predict the uncovered links and the evolution of the graph features simultaneously. We provide oracle bounds under the assumption that the noise sequence has subgaussian tails and we prove that our procedure achieves a trade-off in the calibration of smoothing parameters which adjust with the sparsity and the rank of the unknown adjacency matrix. The rest of this chapter is organized as follows. In Section 2, we describe the general setup of our work with the main assumptions and we formulate a regularized optimization problem which aims at jointly estimating the autoregression parameters and predicting the graph. In Section 3, we provide technical results with oracle inequalities and other theoretical guarantees on the joint estimation-prediction. Section 4 is devoted to the description of the numerical simulations which illustrate our approach. We also provide an efficient algorithm for solving the optimization problem and show empirical results. The proof of the theoretical results are provided as supplementary material in a separate document.

4.2 Estimation of low-rank graphs with autoregressive features

Our approach is based on the asumption that features can explain most of the information contained in the graph, and that these features are evolving with time. We make the following assumptions about the sequence $(A_t)_{t\geq 0}$ of adjacency matrices of the graphs sequence.

Low-Rank. We assume that the matrices A_t have low-rank. This reflects the presence of highly connected groups of nodes such as communities in social networks, or product categories and groups of loyal/fanatic users in a market place data, and is sometimes motivated by the small number of factors that explain nodes interactions.

Autoregressive linear features. We assume to be given a linear map $\omega : \mathbb{R}^{n\times n} \to \mathbb{R}^d$ defined by

$$\omega(A) = \Big(\langle \Omega_1, A\rangle, \cdots, \langle \Omega_d, A\rangle\Big), \tag{4.1}$$

where $(\Omega_i)_{1\leq i\leq d}$ is a set of $n\times n$ matrices. These matrices can be either deterministic or random in our theoretical analysis, but we take them deterministic for the sake of simplicity. The vector time series $(\omega(A_t))_{t\geq 0}$ has autoregressive dynamics, given by a VAR (Vector Auto-Regressive) model:

$$\omega(A_{t+1}) = W_0^\top \omega(A_t) + N_{t+1},$$

where $W_0 \in \mathbb{R}^{d\times d}$ is a unknown sparse matrix and $(N_t)_{t\geq 0}$ is a sequence of noise vectors in $\mathbb{R}^d$. An example of linear features is the degree (*i.e.* number of edges connected to each node, or the sum of their weights if the edges are weighted), which is a measure of popularity in social and commerce networks. Introducing

$$\mathbf{X}_{T-1} = (\omega(A_0), \ldots, \omega(A_{T-1}))^\top \quad \text{and} \quad \mathbf{X}_T = (\omega(A_1), \ldots, \omega(A_T))^\top,$$

which are both $T \times d$ matrices, we can write this model in a matrix form:

$$\mathbf{X}_T = \mathbf{X}_{T-1} W_0 + \mathbf{N}_T, \tag{4.2}$$

where $\mathbf{N}_T = (N_1, \ldots, N_T)^\top$.

This assumes that the noise is driven by time-series dynamics (a martingale increment), where each coordinates are independent (meaning that features are independently corrupted by noise), with a sub-gaussian tail and variance uniformly bounded by a constant σ^2. In particular, no independence assumption between the N_t is required here.

Notations. The notations $\|\cdot\|_F$, $\|\cdot\|_p$, $\|\cdot\|_\infty$, $\|\cdot\|_*$ and $\|\cdot\|_{\text{op}}$ stand, respectively, for the Frobenius norm, entry-wise ℓ_p norm, entry-wise ℓ_∞ norm, trace-norm (or nuclear norm, given by the sum of the singular values) and operator norm (the largest singular value). We denote by $\langle A, B\rangle = \text{tr}(A^\top B)$ the Euclidean matrix product. A vector in $\mathbb{R}^d$ is always understood as a $d \times 1$ matrix. We denote by $\|A\|_0$ the number of non-zero elements of A. The product $A \circ B$ between two matrices with matching dimensions stands for the Hadamard or entry-wise product between A and B. The matrix $|A|$ contains the absolute values of entries of A. The matrix $(M)_+$ is the componentwise positive part of the matrix M, and sign(M) is the sign matrix associated to M with the convention sign$(0) = 0$

If A is a $n \times n$ matrix with rank r, we write its SVD as $A = U\Sigma V^\top = \sum_{j=1}^r \sigma_j u_j v_j^\top$ where $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_r)$ is a $r \times r$ diagonal matrix containing the non-zero singular values of A in decreasing order, and $U = [u_1, \ldots, u_r]$, $V = [v_1, \ldots, v_r]$ are $n \times r$ matrices with columns given by the left and right singular vectors of A. The projection matrix onto the space spanned by the columns (resp. rows) of A is given by $P_U = UU^\top$ (resp. $P_V = VV^\top$). The operator $\mathcal{P}_A : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ given by $\mathcal{P}_A(B) = P_U B + BP_V - P_U B P_V$ is the projector onto the linear space spanned by the matrices $u_k x^\top$ and $y v_k^\top$ for $1 \leq j, k \leq r$ and $x, y \in \mathbb{R}^n$. The projector onto the orthogonal space is given by $\mathcal{P}_A^\perp(B) = (I - P_U)B(I - P_V)$. We also use the notation $a \vee b = \max(a, b)$.

4.2.1 Joint prediction-estimation through penalized optimization

In order to reflect the autoregressive dynamics of the features, we use a least-squares goodness-of-fit criterion that encourages the similarity between two feature vectors at successive time steps. In order to induce sparsity in the estimator of W_0, we penalize this criterion using the ℓ_1 norm. This leads to the following penalized objective function:

$$J_1(W) = \frac{1}{dT}\|\mathbf{X}_T - \mathbf{X}_{T-1}W\|_F^2 + \kappa\|W\|_1,$$

where $\kappa > 0$ is a smoothing parameter.

Now, for the prediction of A_{T+1}, we propose to minimize a least-squares criterion penalized by the combination of an ℓ_1 norm and a trace-norm. This mixture of norms induces sparsity and a low-rank of the adjacency matrix. Such a combination of ℓ_1 and trace-norm was already studied in [GL11] for the matrix regression model, and in [RSV12] for the prediction of an adjacency matrix.

The objective function defined below exploits the fact that if W is close to W_0, then the features of the next graph $\omega(A_{T+1})$ should be close to $W^\top \omega(A_T)$. Therefore, we consider

$$J_2(A, W) = \frac{1}{d}\|\omega(A) - W^\top \omega(A_T)\|_F^2 + \tau\|A\|_* + \gamma\|A\|_1,$$

where $\tau, \gamma > 0$ are smoothing parameters. The overall objective function is the sum of the two partial objectives J_1 and J_2, which is jointly convex with respect to A and W:

$$\mathcal{L}(A, W) \doteq \frac{1}{dT}\|\mathbf{X}_T - \mathbf{X}_{T-1}W\|_F^2 + \kappa\|W\|_1 + \frac{1}{d}\|\omega(A) - W^\top \omega(A_T)\|_2^2 + \tau\|A\|_* + \gamma\|A\|_1, \quad (4.3)$$

If we choose convex cones $\mathcal{A} \subset \mathbb{R}^{n\times n}$ and $\mathcal{W} \subset \mathbb{R}^{d\times d}$, our joint estimation-prediction procedure is defined by

$$(\hat{A}, \hat{W}) \in \underset{(A,W)\in\mathcal{A}\times\mathcal{W}}{\arg\min}\ \mathcal{L}(A, W). \quad (4.4)$$

It is natural to take $\mathcal{W} = \mathbb{R}^{d\times d}$ and $\mathcal{A} = (\mathbb{R}_+)^{n\times n}$ since there is no *a priori* on the values of the feature matrix W_0, while the entries of the matrix A_{T+1} must be positive.

In the next section we propose oracle inequalities which prove that this procedure can estimate W_0 and predict A_{T+1} at the same time.

4.2.2 Main result

The central contribution of our work is to bound the prediction error with high probability under the following natural hypothesis on the noise process.

Assumption 3 *We assume that $(N_t)_{t\geq 0}$ satisfies $\mathbb{E}[N_t|\mathcal{F}_{t-1}] = 0$ for any $t \geq 1$ and that there is $\sigma > 0$ such that for any $\lambda \in \mathbb{R}$ and $j = 1, \ldots, d$ and $t \geq 0$:*

$$\mathbb{E}[e^{\lambda(N_t)_j}|\mathcal{F}_{t-1}] \leq e^{\sigma^2\lambda^2/2}.$$

Moreover, we assume that for each $t \geq 0$, the coordinates $(N_t)_1, \ldots, (N_t)_d$ are independent.

The main result (Theorem 12) can be summarized as follows. The prediction error and the estimation error can be simultaneously bounded by the sum of three terms that involve homogeneously (a) the sparsity, (b) the rank of the adjacency matrix A_{T+1}, and (c) the sparsity of the VAR model matrix W_0. The tight bounds we obtain are similar to the bounds of the Lasso and are upper bounded by:

$$C_1\sqrt{\frac{\log d}{Td^2}}\|W_0\|_0 + C_2\sqrt{\frac{\log n}{d}}\|A_{T+1}\|_0 + C_3\sqrt{\frac{\log n}{d}}\operatorname{rank} A_{T+1}\ .$$

The positive constants C_1, C_2, C_3 are proportional to the noise level σ. The interplay between the rank and sparsity constraints on A_{T+1} are reflected in the observation that the values of C_2 and C_3 can be changed as long as their sum remains constant.

4.3 Oracle inequalities

In this section we give oracle inequalities for the mixed prediction-estimation error which is given, for any $A \in \mathbb{R}^{n\times n}$ and $W \in \mathbb{R}^{d\times d}$, by

$$\mathcal{E}(A, W)^2 \doteq \frac{1}{d}\|(W - W_0)^\top \omega(A_T) - \omega(A - A_{T+1})\|_2^2 + \frac{1}{dT}\|\mathbf{X}_{T-1}(W - W_0)\|_F^2. \quad (4.5)$$

It is important to have in mind that an upper-bound on $\mathcal{E}$ implies upper-bounds on each of its two components. It entails in particular an upper-bound on the feature estimation error $\|\mathbf{X}_{T-1}(\widehat{W} - W_0)\|_F$ that makes $\|(\widehat{W} - W_0)^\top \omega(A_T)\|_2$ smaller and consequently controls the prediction error over the graph edges through $\|\omega(\widehat{A} - A_{T+1})\|_2$.

The upper bounds on $\mathcal{E}$ given below exhibit the dependence of the accuracy of estimation and prediction on the number of features d, the number of edges n and the number T of observed graphs in the sequence.

Let us recall $\mathbf{N}_T = (N_1, \ldots, N_T)^\top$ and introduce the noise processes

$$M = -\sum_{j=1}^{d} (N_{T+1})_j \Omega_j \quad \text{and} \quad \Xi = \sum_{t=1}^{T+1} \omega(A_{t-1}) N_t^\top,$$

which are, respectively, $n \times n$ and $d \times d$ random matrices. The source of randomness comes from the noise sequence $(N_t)_{t \geq 0}$, see Assumption 3. If these noise processes are controlled correctly, we can prove the following oracle inequalities for procedure (4.4). The next result is an oracle inequality of slow type (see for instance [BRT09]), that holds in full generality.

Theorem 9 *Let $(\hat{A}, \hat{W})$ be given by* (4.4) *and suppose that*

$$\tau \geq \frac{2\alpha}{d} \|M\|_{\mathrm{op}}, \quad \gamma \geq \frac{2(1-\alpha)}{d} \|M\|_\infty \quad \textit{and} \quad \kappa \geq \frac{2}{dT} \|\Xi\|_\infty \tag{4.6}$$

for some $\alpha \in (0,1)$. Then, we have

$$\mathcal{E}(\widehat{A}, \widehat{W})^2 \leq \inf_{(A,W) \in \mathcal{A} \times \mathcal{W}} \Big\{ \mathcal{E}(A,W)^2 + 2\tau \|A\|_* + 2\gamma \|A\|_1 + 2\kappa \|W\|_1 \Big\}.$$

For the proof of oracle inequalities of fast type, the *restricted eigenvalue* (RE) condition introduced in [BRT09] and [Kol09b, Kol09c] is of importance. Restricted eigenvalue conditions are implied by, and in general weaker than, the so-called *incoherence* or RIP (Restricted isometry property, [CT05]) assumptions, which excludes, for instance, strong correlations between covariates in a linear regression model. This condition is acknowledged to be one of the weakest to derive fast rates for the Lasso (see [vdGB09] for a comparison of conditions).

Matrix version of these assumptions are introduced in [KLT11]. Below is a version of the RE assumption that fits in our context. First, we need to introduce the two restriction cones.

The first cone is related to the $\|W\|_1$ term used in procedure (4.4). If $W \in \mathbb{R}^{d \times d}$, we denote by $\Theta_W = \mathrm{sign}(W) \in \{0, \pm 1\}^{d \times d}$ the signed sparsity pattern of W and by $\Theta_W^\perp \in \{0,1\}^{d \times d}$ the orthogonal sparsity pattern. For a fixed matrix $W \in \mathbb{R}^{d \times d}$ and $c > 0$, we introduce the cone

$$\mathcal{C}_1(W, c) \doteq \Big\{ W' \in \mathcal{W} : \|\Theta_W^\perp \circ W'\|_1 \leq c \|\Theta_W \circ W'\|_1 \Big\}.$$

This cone contains the matrices W' that have their largest entries in the sparsity pattern of W.

The second cone is related to mixture of the terms $\|A\|_*$ and $\|A\|_1$ in procedure (4.4). Before defining it, we need further notations and definitions.

For a fixed $A \in \mathbb{R}^{n \times n}$ and $c, \beta > 0$, we introduce the cone

$$\mathcal{C}_2(A, c, \beta) \doteq \Big\{ A' \in \mathcal{A} : \|\mathcal{P}_A^\perp(A')\|_* + \beta \|\Theta_A^\perp \circ A'\|_1 \leq c \Big(\|\mathcal{P}_A(A')\|_* + \beta \|\Theta_A \circ A'\|_1 \Big) \Big\}.$$

This cone consist of the matrices A' with large entries close to that of A and that are "almost aligned" with the row and column spaces of A. The parameter β quantifies the interplay between these too notions.

Definition 20 (Restricted Eigenvalue (RE)) *For $W \in \mathcal{W}$ and $c > 0$, we introduce*

$$\mu_1(W, c) = \inf \Big\{ \mu > 0 : \|\Theta_W \circ W'\|_F \leq \frac{\mu}{\sqrt{dT}} \|\mathbf{X}_{T+1} W'\|_F, \ \ \forall W' \in \mathcal{C}_1(W, c) \Big\}.$$

For $A \in \mathcal{A}$ and $c, \beta > 0$, we introduce

$$\mu_2(A, W, c, \beta) = \inf \Big\{ \mu > 0 : \|\mathcal{P}_A(A')\|_F \vee \|\Theta_A \circ A'\|_F \\ \leq \frac{\mu}{\sqrt{d}} \|W'^\top \omega(A_T) - \omega(A')\|_2, \ \ \forall W' \in \mathcal{C}_1(W, c), \forall A' \in \mathcal{C}_2(A, c, \beta) \Big\}.$$

The RE assumption consists of assuming that the constants μ_1 and μ_2 are non-zero. Now we can state the following Theorem that gives a fast oracle inequality for our procedure using RE.

Theorem 10 *Let $(\hat{A}, \hat{W})$ be given by* (4.4) *and suppose that*

$$\tau \geq \frac{3\alpha}{d} \|M\|_{\mathrm{op}}, \quad \gamma \geq \frac{3(1-\alpha)}{d} \|M\|_\infty \quad \textit{and} \quad \kappa \geq \frac{3}{dT} \|\Xi\|_\infty \tag{4.7}$$

for some $\alpha \in (0, 1)$. Then, we have

$$\mathcal{E}(\widehat{A}, \widehat{W})^2 \leq \inf_{(A,W) \in \mathcal{A} \times \mathcal{W}} \Big\{ \mathcal{E}(A, W)^2 + \frac{25}{18} \mu_2(A, W)^2 \big(\mathrm{rank}(A) \tau^2 + \|A\|_0 \gamma^2 \big) \\ + \frac{25}{36} \mu_1(W)^2 \|W\|_0 \kappa^2 \Big\},$$

where $\mu_1(W) = \mu_1(W, 5)$ and $\mu_2(A, W) = \mu_2(A, W, 5, \gamma/\tau)$ (see Definition 20).

The proofs of Theorems 9 and 10 use tools introduced in [KLT11] and [BRT09].

Note that the residual term from this oracle inequality mixes the notions of sparsity of A and W via the terms $\mathrm{rank}(A)$, $\|A\|_0$ and $\|W\|_0$. It says that our mixed penalization procedure provides an optimal trade-off between fitting the data and complexity, measured by both sparsity and low-rank. This is the first result of this nature to be found in literature.

In the next Theorem 11, we obtain convergence rates for the procedure (4.4) by combining Theorem 10 with controls on the noise processes. We introduce

$$v_{\Omega,\mathrm{op}}^2 = \Big\| \frac{1}{d} \sum_{j=1}^d \Omega_j^\top \Omega_j \Big\|_{\mathrm{op}} \vee \Big\| \frac{1}{d} \sum_{j=1}^d \Omega_j \Omega_j^\top \Big\|_{\mathrm{op}}, \quad v_{\Omega,\infty}^2 = \Big\| \frac{1}{d} \sum_{j=1}^d \Omega_j \circ \Omega_j \Big\|_\infty,$$

$$\sigma_\omega^2 = \max_{j=1,\dots,d} \frac{1}{T+1} \sum_{t=1}^{T+1} \omega_j(A_{t-1})^2,$$

which are the (observable) variance terms that naturally appear in the controls of the noise processes. We introduce also

$$\ell_T = 2 \max_{j=1,\dots,d} \log\log \left(\frac{\sum_{t=1}^{T+1} \omega_j(A_{t-1})^2}{T+1} \vee \frac{T+1}{\sum_{t=1}^{T+1} \omega_j(A_{t-1})^2} \vee e \right),$$

which is a small (observable) technical term that comes out of our analysis of the noise process Ξ. This term is a small price to pay for the fact that no independence assumption is required on the noise sequence $(N_t)_{t \geq 0}$, but only a martingale increment structure with sub-gaussian tails.

Theorem 11 *Consider the procedure $(\hat{A}, \hat{W})$ given by* (4.4) *with smoothing parameters given by*

$$\tau = 3\alpha\sigma v_{\Omega,\mathrm{op}}\sqrt{\frac{2(x+\log(2n))}{d}}, \quad \gamma = 3(1-\alpha)\sigma v_{\Omega,\infty}\sqrt{\frac{2(x+2\log n)}{d}},$$
$$\kappa = 6\sigma\sigma_\omega \frac{1}{d}\sqrt{\frac{2e(x+2\log d+\ell_T)}{T+1}}$$

for some $\alpha \in (0,1)$ and fix a confidence level $x > 0$. Then, we have

$$\begin{aligned}\mathcal{E}(\widehat{A},\widehat{W})^2 \leq \inf_{(A,W)\in\mathcal{A}\times\mathcal{W}} \Big\{ &\mathcal{E}(A,W)^2 + 25\mu_2(A)^2 \operatorname{rank}(A)\alpha^2\sigma^2 v_{\Omega,\mathrm{op}}^2 \frac{2(x+\log(2n))}{d} \\ &+ 25\mu_2(A)^2\|A\|_0(1-\alpha)^2\sigma^2 v_{\Omega,\infty}^2 \frac{2(x+2\log n)}{d} \\ &+ 25\mu_1(W)^2\|W\|_0\sigma^2\sigma_\omega^2 \frac{2e(x+2\log d+\ell_T)}{d^2(T+1)}\Big\}\end{aligned}$$

with a probability larger than $1-17e^{-x}$, where μ_1 and μ_2 are the same as in Theorem 10.

The proof of Theorem 11 follows directly from Theorem 10 and basic noise control results, see Proposition 8 in appendix. In the next Theorem, we propose more explicit upper bounds for both the indivivual estimation of W_0 and the prediction of A_{T+1}.

Theorem 12 *Under the same assumptions as in Theorem 11, for any $x > 0$ the following inequalities hold with a probability larger than $1-17e^{-x}$:*

$$\begin{aligned}&\frac{1}{dT}\|\mathbf{X}_T(\hat{W}-W_0)\|_F^2 \\ &\quad\leq \inf_{A\in\mathcal{A}}\Big\{\frac{1}{d}\|\omega(A)-\omega(A_{T+1})\|_F^2 + \frac{25}{18}\mu_2(A,W)^2\big(\operatorname{rank}(A)\tau^2+\|A\|_0\gamma^2\big)\Big\} \\ &\qquad+\frac{25}{36}\mu_1(W_0)^2\|W_0\|_0\kappa^2\end{aligned} \tag{4.8}$$

$$\begin{aligned}&\|\hat{W}-W_0\|_1 \leq 5\mu_1(W_0)^2\|W_0\|_0\kappa \\ &+6\sqrt{\|W_0\|_0}\mu_1(W_0)\inf_{A\in\mathcal{A}}\sqrt{\frac{1}{d}\|\omega(A)-\omega(A_{T+1})\|_F^2+\frac{25}{18}\mu_2(A,W)^2\big(\operatorname{rank}(A)\tau^2+\|A\|_0\gamma^2\big)}\end{aligned} \tag{4.9}$$

$$\begin{aligned}\|\hat{A}-A_{T+1}\|_* \leq\ &5\mu_1(W_0)^2\|W_0\|_0\kappa + (6\sqrt{\operatorname{rank} A_{T+1}}+5\beta\sqrt{\|A_{T+1}\|_0})\mu_2(A_{T+1}) \\ &\times\inf_{A\in\mathcal{A}}\sqrt{\frac{1}{d}\|\omega(A)-\omega(A_{T+1})\|_F^2+\frac{25}{18}\mu_2(A,W)^2\big(\operatorname{rank}(A)\tau^2+\|A\|_0\gamma^2\big)}\,.\end{aligned} \tag{4.10}$$

4.4 Algorithms and numerical experiments

4.4.1 Generalized forward-backward algorithm for minimizing $\mathcal{L}$

We use the algorithm designed in [RFP11] for minimizing our objective function. Note that this algorithm is inherently superior to the method introduced in [RBEV10] as it directly minimizes $\mathcal{L}$ jointly in (S, W) rather than alternately minimize in W and S.

Moreover we use the novel joint penalty from [RSV12] that is more suited for estimating graphs. The proximal operator for the trace norm is given by the shrinkage operation, if $Z = U\operatorname{diag}(\sigma_1,\cdots,\sigma_n)V^T$ is the singular value decomposition of Z,

$$\operatorname{prox}_{\tau\|\cdot\|_*}(Z) = U\operatorname{diag}((\sigma_i-\tau)_+)_i V^T.$$

Method	AUC
Convex	0.9094 ± 0.0176
Factorization	$\mathbf{0.9454 \pm 0.0087}$

Table 4.1: Comparison of the convex and factorization approaches over graph prediction task.

Similarly, the proximal operator for the ℓ_1-norm is the soft thresholding operator defined by using the entry-wise product of matrices denoted by $\circ$:

$$\mathrm{prox}_{\gamma||\cdot||_1} = \mathrm{sgn}(Z) \circ (|Z| - \gamma)_+ \,.$$

The algorithm converges under very mild conditions when the step size θ is smaller than $\frac{2}{L}$, where L is the operator norm of the joint quadratic loss:

$$\Phi : (A, W) \mapsto \frac{1}{dT}\|\mathbf{X}_T - \mathbf{X}_{T-1}W\|_F^2 + \frac{1}{d}\|\omega(A) - W^\top \omega(A_T)\|_F^2 \ .$$

Algorithm 5 Generalized Forward-Backward to Minimize $\mathcal{L}$

Initialize $A, Z_1, Z_2, W, q = 2$
repeat
  Compute $(G_A, G_W) = \nabla_{A,W}\Phi(A, W)$.
  Compute $Z_1 = \mathrm{prox}_{q\theta\tau||\cdot||_*}(2A - Z_1 - \theta G_A)$
  Compute $Z_2 = \mathrm{prox}_{q\theta\gamma||\cdot||_1}(2A - Z_2 - \theta G_A)$
  Set $A = \frac{1}{q}\sum_{k=1}^{q} Z_k$
  Set $W = \mathrm{prox}_{\theta\kappa||\cdot||_1}(W - \theta G_W)$
until convergence
return (A, W) minimizing $\mathcal{L}$

4.4.2 Nonconvex factorization method

An alternative method to the estimation of low-rank and sparse matrices by penalizing a mixed penalty of the form $\tau\|A\|_* + \gamma\|A\|_1$ as in [RSV12] is to factorize $A = UV^\top$ where $U, V \in \mathbb{R}^{n\times r}$ are sparse matrices, and penalize $\gamma(\|U\|_1 + \|V\|_1)$. The objective function to be minimized is

$$\begin{aligned}\mathcal{J}(U, V, W) \doteq \frac{1}{d}\|\mathbf{X}_{T-1}W - \mathbf{X}_T\|_F^2 + \kappa\,\|W\|_1 \\ + \frac{1}{d}\|\omega(A_T)^\top W - \omega(UV^\top)^\top\|_2^2 + \gamma(\|U\|_1 + \|V\|_1)\end{aligned}$$

which is a non-convex function of the joint variable (U, V, W), making the theoretical analysis more difficult. Given that the objective is convex in a neighborhood of the solution, by initializing the variables adequately, we can write an algorithm inspired by proximal gradient descent for minimizing it. The main motivation for directly optimizing the factors U and V of the matrix A rather than penalizing its nuclear norm come from the parallelization and stochastic optimization perspective such an approach offers. We leave further analysis for future work and settle for reporting empirical success over convex methods over simulated data in table 4.1.

Algorithm 6 Minimize $\mathcal{J}$

Initialize U, V, W
repeat
  Compute $(G_U, G_V, G_W) = \nabla_{U,V,W}\Phi(UV^\top, W)$.
  Set $U = \text{prox}_{\theta\gamma||.||_1}(U - \theta G_U)$
  Set $V = \text{prox}_{\theta\gamma||.||_1}(V - \theta G_V)$
  Set $W = \text{prox}_{\theta\kappa||.||_1}(W - \theta G_W)$
until convergence
return (U, V, W) minimizing $\mathcal{J}$

4.4.3 Laplacian regularizer for node features attracted following graph edges

Consider the case where the extracted features and descriptors are node features and node descriptors. In this case ω is matrix valued, $\omega(A) \in \mathbb{R}^{n\times q}$, each row of it coding the features of the corresponding node. We denote by $f^{(i)}$ the i-th component of f. The n prediction functions $f^{(i)}$ are assumed to belong to the same Hilbert space $\mathcal{H}$, equipped with the norm $\|\cdot\|_{\mathcal{H}}$. The norm of $f \in \mathcal{H}^n$ is defined by $\|f\|_{\mathcal{H}^n} := \sqrt{\sum_{i=1}^n \|f^{(i)}\|^2_{\mathcal{H}}}$. We develop here a methodology to relate the graph and the predictors through an additional term that acts as a common regularizer. Before introducing this term we need to make the hypotheses

- $A_t \in \mathbf{S}^n_{\geq 0}, t \in \{0, 1, 2, ..., T\}$, where $\mathbf{S}^n_{\geq 0}$ denotes the set of $n \times n$ symmetric matrices with nonnegative entries.
- We assume also that, the weights of the edges are nondecreasing functions of time, which in the unweighted case means edges do not disappear but new ones may appear over time.

For simplicity of presentation, we consider the case of $\mathcal{H}$ being an RKHS corresponding to a linear kernel. In addition, each $f^{(i)}$ is a linear function represented by a $q \times q$ matrix $W^{(i)}$, so that $f^{(i)}(\omega(A_t)^{(i)}) = \omega(A_t)^{(i)}W^{(i)}$. Then we may assume $\|f^{(i)}\|_{\mathcal{H}} = \|W^{(i)}\|_F$ and the graph regularity term can be expressed in terms of the graph Laplacian of A. Recall that the Laplacian is defined as the operator Λ such that $\Lambda(A) = D - S$ where $D = \text{diag}(d_1, \ldots, d_n)$ and $d_i = \sum_{j=1}^n A_{i,j}$ are the degrees of the graph. A standard computation gives the following expression for J_4:

$$\text{Tr}\big(A^\top \Delta(f)\big) = \sum_{1\leq i,j\leq n} A_{i,j}\|W^{(i)} - W^{(j)}\|^2_F$$

where $W = (W^{(1)}, \ldots, W^{(n)}) \in \mathbb{R}^{n\times q\times q}$. We use the standard extension of the Frobenius norm to 3-tensors, $\|W\|^2_F = \sqrt{\sum_{i=1}^n \|W^{(i)}\|^2_F}$ and the notations:

$$Q(W, \Lambda, V) := \sum_{1\leq i,j\leq n} \Lambda_{ij}\,\text{Tr}(W^{(i)\top}V^{(j)})\ ,$$
$$\Delta(W) := \Big(\|W^{(i)} - W^{(j)}\|^2_F\Big)^n_{i,j=1}\ .$$

We also assume in this work that the node features are linear functions of adjacency matrices : $\omega(A_t) = A_t\Omega$ for some $\Omega \in \mathbb{R}^{n\times q}$. Note that degree, inter/intra cluster degrees and projection onto specific linear subspaces are such features, but not clustering coefficients or statistics on path lengths. We explain later how we chose Ω in our experiments.

The Laplacian-like term that ensures joint regularity of graph and predictors can be written as follows

$$\mathrm{Tr}\big(A^\top \Delta(f)\big) = \frac{1}{2}\sum_{i,j} A_{i,j}\|f^{(i)} - f^{(j)}\|_{\mathcal{H}}^2$$

where we have used

$$\Delta(f) = \Big(\|f^{(i)} - f^{(j)}\|_2^2\Big)_{i,j} \in \mathbb{R}^{n\times n}$$

This term relates the contributions of f and A enforcing regularity of the predictors over the estimated graph through A.
The objective to minimize in this case contains the additional Laplacian-like term enforcing predictors regularity over A:

$$\mathcal{L}_{\text{Lap}}(A, f) \doteq \frac{1}{d}\|y - f(X)\|_F^2 + \kappa\, \mathbf{pen}(f)$$
$$+\frac{1}{2}\|\omega(A) - f(\omega(A_T))\|_2^2 + \tau\|A\|_* + \gamma\|A\|_1 + \frac{\lambda}{2}\sum_{i,j} A_{i,j}\|f^{(i)} - f^{(j)}\|_{\mathcal{H}}^2$$

We only point out that the objective function $\mathcal{L}_{\text{Lap}}(A, f)$ can be optimized inside the region it is convex in using projected gradient algorithms, future investigation is required on both empirical and theoretical properties of the resulting estimator.

In order to determine the convexity region of $\mathcal{L}_{\text{Lap}}$ we use intuition from what happens with a similar optimization problem in dimension 2. Indeed, consider the minimization of $(w, a) \to aw^2 + \alpha a^2 + \beta w^2$ which is the simplified functional in the degenerate case where $n = 1$. The eigenvectors of the Hessian are positive iff $w^2 \leq \alpha\beta$, so this condition defines the convexity region. By adding the quadratic terms in A and W to $Q(W, \Lambda(A), W)$, let us define

$$Q(W, \Lambda, V) := \sum_{1\leq i,j\leq n} \Lambda_{ij}\, \mathrm{Tr}(W^{(i)\top} V^{(j)})$$

and

$$\Psi(A, W) := \frac{\kappa}{2}\|W\|_F^2 + \frac{\nu}{2}\|A - A_T\|_F^2 + \lambda Q(W, \Lambda(A), W).$$

If we suppose that the entries of $Z = A - A_T$ are nonnegative, then the minimizer of Ψ is the trivial solution $A = A_T, W = 0$. In fact,

$$\Psi(A, W) = \frac{\kappa}{2}\|W\|_F^2 + \frac{\nu}{2}\|Z\|_F^2 + \lambda\Big(Q(W, \Lambda(A_T), W) + Q(W, \Lambda(Z), W)\Big) \geq 0 \tag{4.11}$$

and for $Z = 0, W = 0$, $\Psi(A, W) = 0$.

We prove that $\mathcal{L}_{\text{Lap}}$ is convex over a set $\mathcal{E}$ around the minimizer of Ψ and we will ensure henceforth that the descent algorithm takes place inside this convex domain.

Proposition 2 *The function Ψ is convex in the interior of the set:*

$$\mathcal{E} = \left\{ A \in \mathbf{S}_{\geq 0}^n, W \in \mathbb{R}^{n\times d\times q} \;\middle|\; \|W\|_F \leq \frac{\sqrt{\nu\kappa}}{2\lambda\big(\sqrt{n}+1\big)} \right\}.$$

Proof. We introduce the slack variable $Z = A - A_T$ and isolate the quadratic part of $\Psi(Z_0+Z, W_0+W)$ for some (Z_0, W_0) :

$$R(Z,W) := \frac{\nu}{2}\|Z\|_F^2 + \frac{\kappa}{2}\|W\|_F^2 + \lambda\Big(2Q(W,\Lambda(Z),W_0) + Q(W,\Lambda(A_T+Z_0),W)\Big).$$

Thanks to Cauchy-Schwarz and the basic norm property $\|AB\|_F \leq \|A\|_F\|B\|_F$,

$$Q(W,\Lambda(Z),W_0) \geq -\|W\|_F\|\Lambda(Z)\|_F\|W_0\|_F.$$

We have $\Lambda(Z) = D - Z$. We get, again by Cauchy-Schwarz

$$\|D\|_F^2 = \sum_{i=1}^{n}(\sum_{j=1}^{n} Z_{i,j})^2 \leq \sum_{i=1}^{n} n \sum_{j=1}^{n} Z_{i,j}^2 = n\|Z\|_F^2$$

and therefore

$$\|\Lambda(Z)\|_F = \|D - Z\|_F \leq \|D\|_F + \|Z\|_F \leq (\sqrt{n}+1)\|Z\|_F.$$

On the other hand $Q(W,\Lambda(A_T+Z_0),W) \geq 0$, so

$$R(Z,W) \geq \frac{\nu}{2}\|Z\|_F^2 + \frac{\kappa}{2}\|W\|_F^2 - 2\lambda(\sqrt{n}+1)\|W\|_F\|Z\|_F\|W_0\|_F.$$

Letting $z = \|Z\|_F, w = \|W\|_F, w_0 = \|W_0\|_F$, we have

$$R(Z,W) \geq \frac{\nu}{2}z^2 + \frac{\kappa}{2}w^2 - 2\lambda(\sqrt{n}+1)wzw_0 \ .$$

$\psi_{w_0} : (z,w) \mapsto \frac{\nu}{2}z^2 + \frac{\kappa}{2}w^2 - 2\lambda(\sqrt{n}+1)wzw_0$ is a quadratic form. Therefore $R(Z,W)$ is always nonnegative if ψ_{w_0} is positive semidefinite positive. That is, if $\nu + \kappa \geq 0$ (always true) and $\nu\kappa - 4\lambda^2\big(\sqrt{n}+1\big)^2 w_0^2 \geq 0$, and this completes the proof. □

4.4.4 A generative model for graphs having linearly autoregressive features

Let $V_0 \in \mathbb{R}^{n\times r}$ be a sparse matrix, $V_0^\dagger$ its pseudo-inverse such, that $V_0^\dagger V_0 = V_0^\top V_0^{\top\dagger} = I_r$. Fix two sparse matrices $W_0 \in \mathbb{R}^{r\times r}$ and $U_0 \in \mathbb{R}^{n\times r}$. Now define the sequence of matrices $(A_t)_{t\geq 0}$ for $t = 1, 2, \cdots$ by

$$U_t = U_{t-1}W_0 + N_t$$

and

$$A_t = U_t V_0^\top + M_t$$

for i.i.d sparse noise matrices N_t and M_t, which means that for any pair of indices (i,j), with high probability $(N_t)_{i,j} = 0$ and $(M_t)_{i,j} = 0$. We define the linear feature map $\omega(A) = AV_0^{\top\dagger}$, and point out that

1. The sequence $\Big(\omega(A_t)^\top\Big)_t = \Big(U_t + M_t V_0^{\top\dagger}\Big)_t$ follows the linear autoregressive relation

$$\omega(A_t)^\top = \omega(A_{t-1})^\top W_0 + N_t + M_t V_0^{\top\dagger} \ .$$

2. For any time index t, the matrix A_t is close to $U_t V_0$ that has rank at most r

3. The matrices A_t and U_t are both sparse by construction.

4.4.5 Evaluation metrics

Various prediction tasks could be studied in our setup with specific criteria for optimization and evaluation. We focus here on regression (for feature prediction) and graph completion (for link prediction), but classification of vertices may also be a useful task. Here we denote by $\widehat{A}$ the prediction we made for A_{T+1} given the past observations.

1. *Regression.* When the effective value of an asset in the future is of interest, a squared error metric is appropriate, leading to the use of $\|\omega(A_{T+1}) - \widehat{A}\|_F$ for evalution. The reported values are the relative errors $\frac{\|\omega(A_{T+1})-\widehat{A}\|_F}{\|\omega(A_{T+1})\|_F}$.

2. *Graph Completion.* Our method aims simultaneously at the prediction of $\omega(A_{T+1})$ and A_{T+1}. We measure the quality of prediction of A_{T+1} by $\|\widehat{A} - A_{T+1}\|_F$, and the relative values reported are $\frac{\|A_{T+1}-\widehat{A}\|_F}{\|A_{T+1}\|_F}$.

3. *Classification.* Specific patterns may appear in the time series ω_t which we may wish to predict. For instance, predicting top-selling items in a given market is definitely of interest. This problem can be formulated as follows: the matrix ω_t represents the sales volumes over a market (each component corresponds to a product), and we assign a ± 1 label depending on the order of magnitude of the increase of sales volumes over a specific time window.

4. *Ranking* In cases where the ranking of all the instances are of interest for the application, the standard tool to measure the quality of a scoring rule is the use of *Receiver O Characteristics* (ROC) curves. The ROC curves is the plot of the True Positive Rate as a function of the False Positive Rate. The curves that are situated above others correspond to the best scoring rules. The area under the ROC curve is sometimes used to evaluate the quality of a scoring rule.

5. *Top-k ranking* In the recommender system applications for each user, a list of $k = 10$ recommendations are computed. We evaluate the performance with two different measures commonly used in Information Retrieval:

   - the Hit Rate - evaluates if there is a correct hit in the list:
   $$\text{Hit Rate@}N = \frac{\text{\# correct hits}}{n} ,$$
   - the *Normalized Discounted Cumulative Gain* [KK02] - sensitive to the ranking of the item in the list:
   $$\text{NDCG@}N = \frac{\text{DCG@}N}{\sum_{i=1}^{N} \frac{1}{\log(i+1)}} ,$$
   where
   $$\text{DCG@}N = \sum_{i=1}^{N} \frac{2^{y_i} - 1}{\log(i+1)}$$
   and $y_i = 1$ if the i-th recommendation was effectively purchased and $y_i = 0$ otherwise.

Note that even though these two metrics seem very similar, they measure slightly different things. In fact NDCG is sensitive to the order in which the N recommendations are presented, while Hit Rate -that is more easily interpretable- considers this list as an unordered set.

4.4.6 Bias of the measurement methodology

The objective of this thesis was to develop estimation methods for prediction purposes in temporal sequences of graphs. A natural way of testing the validity of the working hypotheses and the suggested algorithms is *backtesting*. It consists of using past data for both training and testing. The stability of the estimators can be measured also by taking several training and test periods in the past. One of the major criticism to such a methodology is that given the unbalanced degree distributions in real-world data, the outputs of the algorithms fitting to real world data are often over-represent the popular items. Figure 4.1 shows for illustrating this statement, the linear dependency of magnitude of standard scores on the sales volumes illustrates the bias due to the heavy-tailed distributions that hurts these scoring rules. Therefore they can suffer from poor recommendation diversity, by either recommending the same best-selling items to all the users or by fitting to the users tastes and not letting them discover new items.

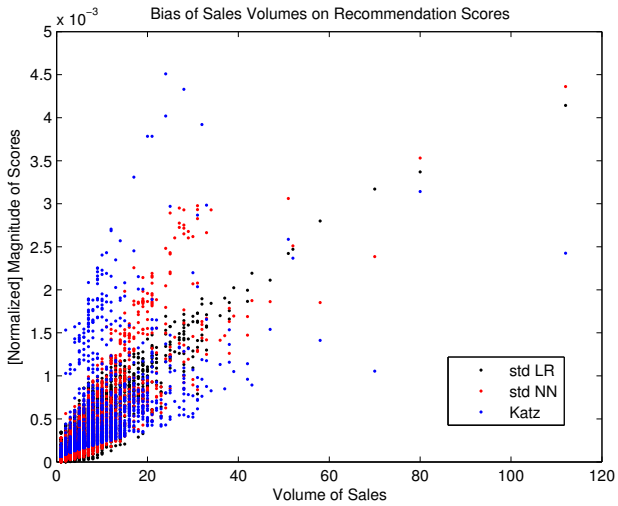


Figure 4.1: Dependency of magnitude of standard scores on the sales volumes.

4.4.7 Empirical evaluation of heuristics on real data sets

It is a clear industrial requirement that the algorithms developed for recommender systems should scale to millions of users and millions of products database size. However, the storage and manipulation of such data would require tera-bytes of cash and RAM memory. Given the fast growing volumes of data that need to be processed, one could clearly not handle quadratic or cubic algorithms as those studied in the current thesis and wait for computers to have sufficient power to deal with these data.

It turned obvious after a short while of studying recommender systems that local algorithms should be preferred to global algorithms. In fact in all the studied approaches, the trace-norm term, or the low-rank criterion was the key to handle the collaborative aspect of the learning process by generalizing the characteristics / tastes of a node / user to the others. Somehow it is a fact that a minor change in one entry of the input matrix will affect the spectrum of it and consequently affect the trace-norm. This is why a first conclusive remark would be to point out the limitation of the approach due to the presence of this term as the only generalizer.

As a case study for our industrial partner, and also by the desire of better understanding the

nature of the data-mining problem, whatever statistical relevance this had, we also developed some very basic algorithms using an item-to-item approach. The idea is to compute the similarity among pairs of products and store only the top-K most similar corresponding to each. The Figure 4.2 shows how to use a cross-validation procedure for tuning the length of lists to merge for books market: $\hat{K} = 7$ maximizes the average Hit Rate in this case. The suggested list of items to a user would be a list obtained by mixing the corresponding lists of his last purchases. Our numerical studies on several databases have proven empirically that such an approach is outperforming very easily its full-matrix scoring rivals by de-biasing the scores to the popularity of the recommended items. The second asset of these approaches was that they were able to provide recommendations lists in sub-linear time, as the heavy computation is done in advance and the products are stored in tables. The Figure 4.3 shows that except for Electronic products market where the notion of similarity/vicinity of products seems meaningless we see the benefits of both steps of our approach on the predictions. The reason why we did not push the study of these algorithms further is that we wanted to build statistical models for better understanding and capturing evolution patterns of the data. The immediate applicability and use of the methods was not the major concern.

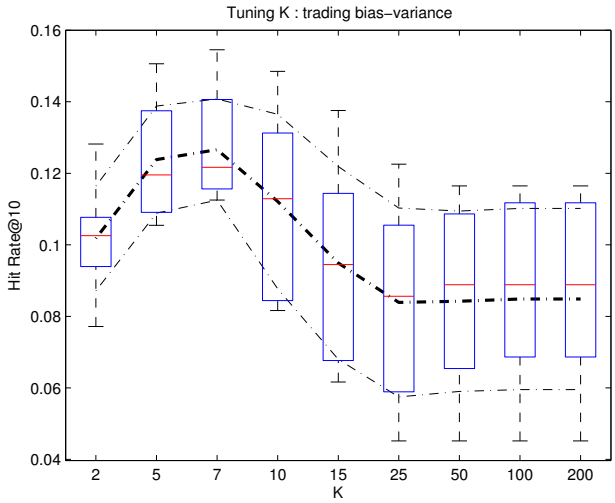


Figure 4.2: Cross-validation for the parameter K, the length of intermediary lists.

Given the directions shown by the mixed regularizers design we think there is clearly a room for improvement by designing and implementing local versions of these regularized algorithms. The deep question remains in the interpretation one makes from the term *local*. In fact the metric of the spaces in which our work was developed is a euclidean metric that can be easily generalized to Hilbert spaces as Reproducing Kernel Hilbert Spaces by using a PSD kernel. Somehow recent research has suggested [CFHW12] that the hyperbolic geometry may be more adapted to the study of graphs of human activity. Even though at the first glance such an assertion may sound more as a non-pragmatic intellectual-game idea, a closer look at the question reveals deep insights which may give a clue to demystifying the nature of these random graphs.

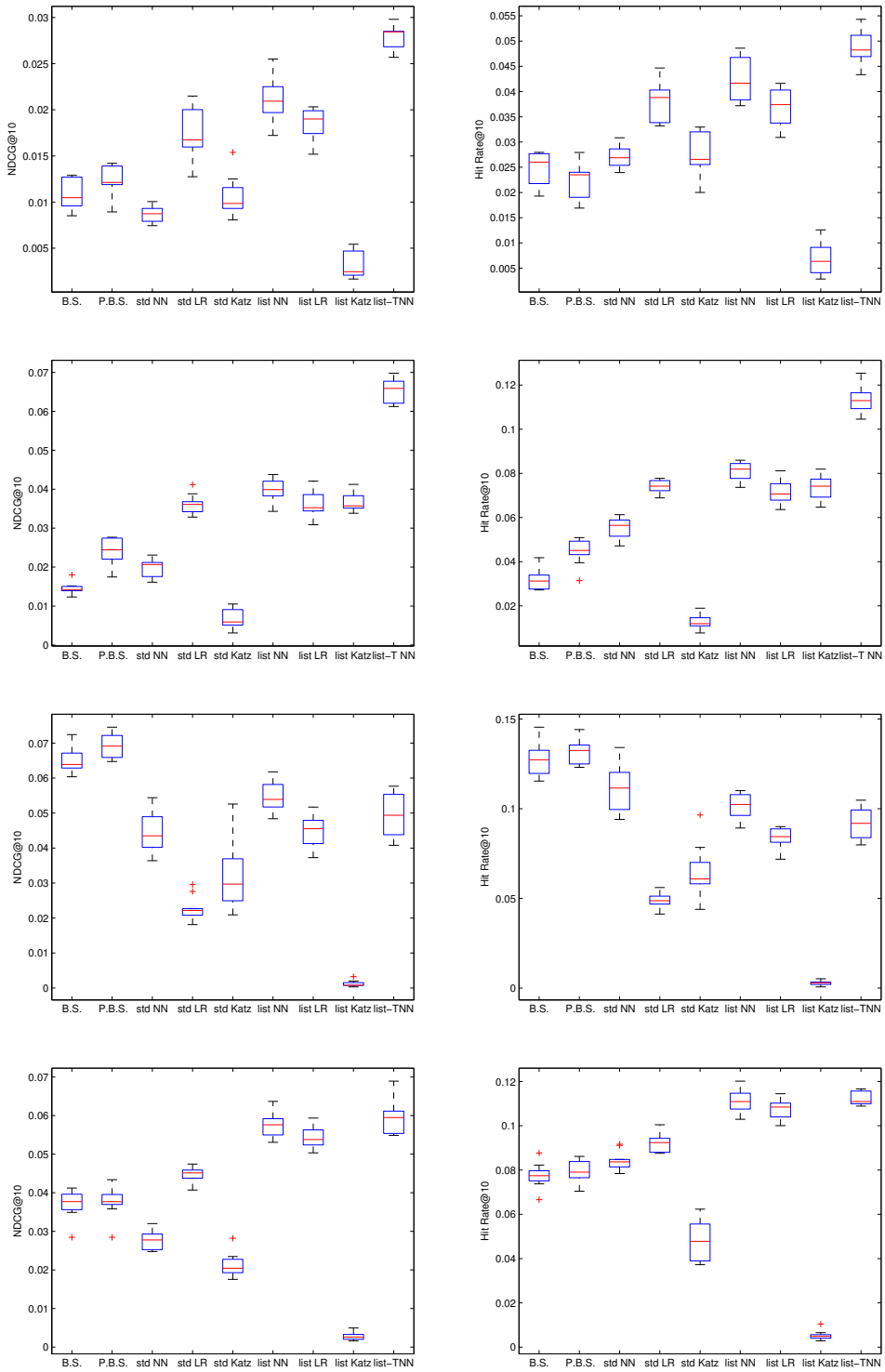


Figure 4.3: NDCG@N in the left column and Hit Rate@N at right. Row 1 : Music; Row 2 : Books; Row 3 : Electronic Devices; Row 4 : Video Games.

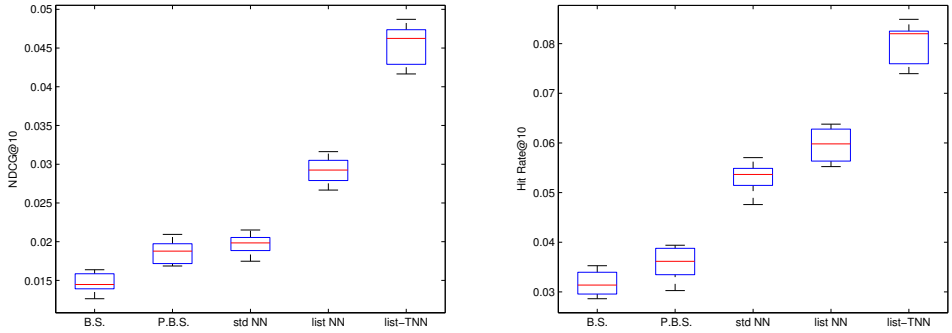


Figure 4.4: Prediction accuracy on 8 samples of 20,000 users and 20,000 products each. Products are randomly chosen from the Books market.

4.4.8 Empirical evaluation of the regularization method on synthetic data sets

We tested the presented methods on synthetic data generated as in section (4.4.4). In our experiments the noise matrices M_t and N_t where built by soft-thresholding *i.i.d.* noise $\mathcal{N}(0, \sigma^2)$. We took as input $T = 10$ successive graph snapshots on $n = 50$ nodes graphs of rank $r = 5$. We used $d = 10$ linear features, and finally the noise level was set to $\sigma = .5$. We compare our methods to standard baselines in link prediction. We use the area under the ROC curve as the measure of performance and report empirical results averaged over 50 runs with the corresponding confidence intervals in Figure 4.5. The competitor methods are the *nearest neighbors* (NN) and static sparse and low-rank estimation, that is the link prediction algorithm suggested in [RSV12]. The algorithm NN scores pairs of nodes with the number of common friends between them, which is given by A^2 when A is the cumulative graph adjacency matrix $\widetilde{A_T} = \sum_{t=0}^{T} A_t$ and the static sparse and low-rank estimation is obtained by minimizing the objective $\|X - \widetilde{A_T}\|_F^2 + \tau\|X\|_* + \gamma\|X\|_1$, and can be seen as the closest *static* version of our method. The two methods *autoregressive low-rank* and *static low-rank* are regularized using only the trace-norm, (corresponding to forcing $\gamma = 0$) and are slightly inferior to their sparse and low-rank rivals. Since the matrix V_0 defining the linear map ω is unknown we consider the feature map $\omega(A) = SV$ where $\widetilde{A_T} = U\Sigma V^\top$ is the SVD of $\widetilde{A_T}$. The parameters τ and γ are chosen by 10-fold cross validation for each of the methods separately.

4.5 Discussion

1. *Comparison with the baselines.* This experiment sharply shows the benefit of using a temporal approach when one can handle the feature extraction task. The left-hand plot shows that if few snapshots are available ($T \leq 4$ in these experiments), then static approaches are to be preferred, whereas feature autoregressive approaches outperform as soon as *sufficient number* T graph snapshots are available (see phase transition). The decreasing performance of static algorithms can be explained by the fact that they use as input a mixture of graphs observed at different time steps. Knowing that at each time step the nodes have specific latent factors, despite the slow evolution of the factors, adding the resulting graphs leads to confuse the factors.

2. *Phase transition.* The right-hand figure is a phase transition diagram showing in which part of rank and time domain the estimation is accurate and illustrates the interplay

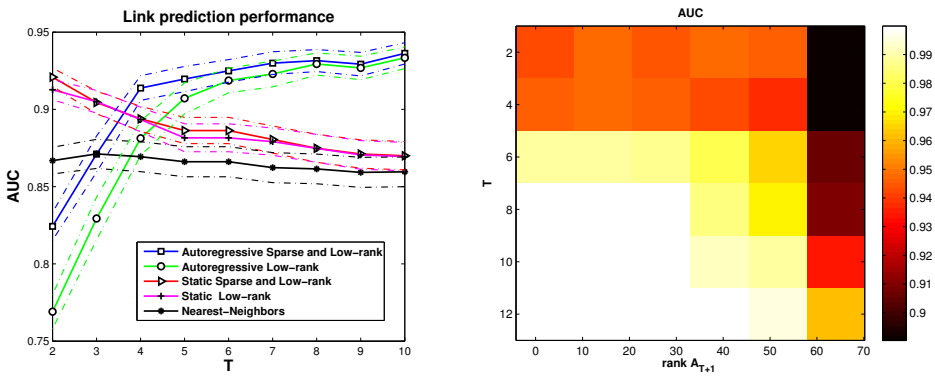


Figure 4.5: Left: performance of algorithms in terms of Area Under the ROC Curve, average and confidence intervals over 50 runs. Right: Phase transition diagram.

between these two domain parameters.

3. *Choice of the feature map ω.* In the current work we used the projection onto the vector space of the top-r singular vectors of the cumulative adjacency matrix as the linear map ω, and this choice has shown empirical superiority to other choices. The question of choosing the best measurement to summarize graph information as in compress sensing seems to have both theoretical and application potential. Moreover, a deeper understanding of the connections of our problem with compressed sensing, for the construction and theoretical validation of the features mapping, is an important point that needs several developments. One possible approach is based on multi-kernel learning, that should be considered in a future work. An extension to nonlinear graph features such as the distribution of triangles or other nonlinear patterns of interest is also to be considered.

4. *Generalization of the method.* In this chapter we consider only an autoregressive process of order 1. For better prediction accuracy, one could consider mode general models, such as vector ARMA models, and use model-selection techniques for the choice of the orders of the model. A general modelling based on state-space model could be developed as well. We presented a procedure for predicting graphs having linear autoregressive features. Our approach can easily be generalized to non-linear prediction through kernel-based methods.

5. *A matrix covariate multi-armed bandits problem* As highlighted several times in the current thesis, seeing the recommendation problem as a prediction problem quickly reaches its limits. In fact recommending very likely items, even though practitioners argue that it avoids customers outflow to competitor retailers, is not necessarily the best business decision for a long-term contact with the customer [Bod08]. The trade-off risen by this problem is an exploitation-exploration trade-off that can easily be modeled using the multi-armed bandit formalism. We were eager to formulate this problem, but not able to solve it so far, a closely related problem has been solved meanwhile by Hazan and coauthors [HKSS12]. The formulation of the problem goes as follows.

   The Bandit Completion problem is an interesting problem that arised when trying to optimally allocate advertisement ressources in typical CRM databases containing $N \simeq 10^7$ users and used by around $K \geq 10^3$ advertisers. The concrete application requires to max-

imize the click-rate (reward) for a limited number of advertisements T. In the following we refer to the matrix containing the expected click rate of a user i on an advertisment sent by advertiser j as the expected reward matrix.

Take N bandits each bandit having K arms to play. We want to study the following scenari :

(a) **Clusters of Bandits** : suppose the bandits can be partitionned in r clusters, $r \ll K, N$. The reward laws being identical for the bandits belonging to the same cluster. We aim at performing simultaneousely the clustering of the bandit population and also finding the best arm for each cluster.

(b) **Low rank rewards matrix** : if $A \in \mathbb{R}^{N \times K}$ is the rewards matrix, *i.e.* $A_{i,j}$ represents the expected reward of arm j for bandit i, we assume A has rank $r \ll K, N$.

At round t the playing bandit and the arm $(n_t, k_t) \in \{1, \cdots, N\} \times \{1, \cdots, K\}$ are chosen and reward $\mathcal{R}(n_t, k_t)$ is obtained. We want to find the optimal strategy for choosing (n_t, k_t) in order to maximize the cumulated reward after T rounds $\sum_{t=1}^{T} \mathcal{R}(n_t, k_t)$, or equivalently if $\mathcal{R}^* = \max_{(n,k) \in \{1,\cdots,N\} \times \{1,\cdots,K\}} \mathbb{E}\mathcal{R}(n, k)$ is the maximal expected reward, we want to minimize the regret

$$R_T = T\mathcal{R}^* - \sum_{t=1}^{T} \mathcal{R}(n_t, k_t)$$

6. *Convexity.* This work has been devoted to develop methods for estimating interactions among objects. The data domain having motivated this work is e-commerce data, while the methods introduced can be used for measuring interactions in other domains such as in biology and drug discovery. We met several nonconvex objective functions. Examples of nonconvex terms are

   - the laplacian inner product that pushes the node features to be close when the covariate is high and vice versa:

   $$(X, w) \mapsto \sum_{i,j} X_{i,j} \|w_i - w_j\|_2^2$$

   - by replacing the distance $\|w_i - w_j\|_2^2$ by an inner product we obtain

   $$(X, w) \mapsto \sum_{i,j} X_{i,j} \langle w_i, w_j \rangle$$

   - in the context of matrix factorization, the standard loss function takes the form

   $$(U, V) \mapsto \|UV^\top - A\|$$

   - the squared error term in matrix feature estimation may be nonconvex if the feature map is not linear

   $$X \mapsto \|\omega(X) - \omega(A_0)\| \text{ where } \omega \text{ is not linear}$$

   and we saw that very basic graph characteristics such as the number of triangles surrounding a node, or the clustering coefficient are nonlinear and nonconvex.

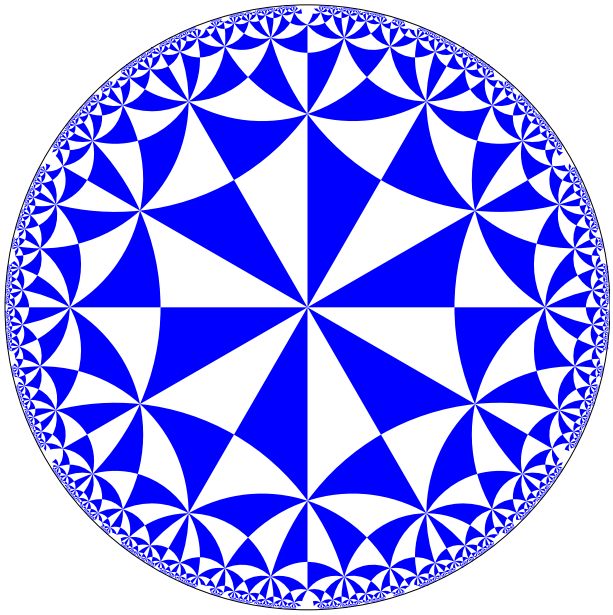


Figure 4.6: Poincaré hyperbolic disk.

As the purpose of using regularization methods is to come up with fast algorithms leading to accurate estimators/predictors, dealing with nonconvex objects is *hopeless*. Nevertheless these terms are natural candidates for describing real-world effects. The classical issue for dealing with these terms is either to ignore their nonconvexity by restricting the search space to the region of convexity (see Proposition 2 for instance) or to replace the problem with a convex one [AEP06] which has the same solution, or to reformulate the problem with use SVD-related tools that ensure the uniqueness of the solution for the high computational cost of spectral computation. A future direction of work is to explore the information relying in such nonconvex terms, possibly for other goals than estimating accurately a vector of parameters.

7. **Detecting hierarchical structures.** The use of a mixed ℓ_1-trace norm penalty has been motivated by the presence of highly connected groups of nodes in real world graphs. In fact such a penalty allows to estimate matrices containing such structures. Another very important characteristic of real-worl networks is the presence of hierarchical structures. A promising direction of work is to develop a methodology to estimate such structures. A series of previous work (see [CFHW12] and references therein), relying on the properties of hyperbolic spaces have suggested to embed the nodes onto a hyperbolic space rather than onto euclidean space. The Figure 4.6 illustrates the Poincaré hyperbolic disk where the multiscale, hierarchical structure of the space is visible. The challenge will be to develop efficient algorithms in such a framework where the expression of the basic objects such as inner products and distances is more complicated than in euclidean case.

Appendix A

Appendices

A.1 Mathematical tools and notations

A.1.1 Basic notions in graph theory

We present in this section usual notions and statistics used to capture topology and characteristics of a graph.

Standard graph features

Definition 21 (Degree) *The degree of a node is the sum of the weights of its adjacent edges. If the graph is directed, we define the in degree and out degree for the sum of weights of edges directed to and from a vertex. If the graph is not weighted, we apply the definition by taking all the weights equal to* 1.

Definition 22 (Path) *A path (or walk) is a sequence of vertices* $(v_1 \dots v_n), \{v_i\} \in V$ *such that two consecutive vertices are linked by an edge :* $\forall k \in \{1, n-1\}, (v_k, v_{k+1}) \in E$. *The length of a path is the number of edges* $n-1$.

Definition 23 (Neighborhood) *The neighborhood of a vertex* v, *noted* $N(v)$, *is the set of neighbors of* v *that is vertices adjacent to* v *not including* v *itself :* $N(x) = \{y|(x,y) \in E\}$.

Definition 24 (Geodesic distance) *The geodesic distance between two nodes is the shortest path between the two nodes.*

Definition 25 (Clique) *A clique of a graph* $G = (V, E)$ *is a subset* C *of the vertex set such that for every two vertices in* C *there exists an edge connecting the two :* $\forall (x,y) \in C^2 (x,y) \in E$. *A* $k-$*clique is a clique containing exactly* k *edges.*

Definition 26 (Clustering coefficient) *The clustering coefficient of a graph is the probability that 3 nodes belong to a triangle.*

$$C_i = \frac{|\textit{neighbor triangles}|}{\binom{d_i}{2}} = \frac{|\textit{3-vertex cliques}|}{\binom{|\textit{distance 1 elements}|}{2}}$$

A "high" value of C_i translates a disposition to establish direct links with the elements of of distance 2. By the same kind of consideration, one can quantify the inclination to attach to distance k elements.

Matrices related to a graph

Another way of normalizing the adjacency matrix is

Remark 3 (Symmetric normalized adjacency) *The symmetric normalized adjacency matrix is*

$$\mathcal{A} = D^{-1/2} A_{\mathcal{G}} D^{-1/2}$$

Remark 4 (Graph laplacian) *The unnormalized laplacian of a graph is the matrix*

$$L = D - A_{\mathcal{G}}$$

.

The normalized laplacian of a graph is the matrix

$$\mathcal{L} = I_n - D^{-1/2} A_{\mathcal{G}} D^{-1/2}$$

The following definition and the related remarks are specific to the bipartite case.

Definition 27 (Incidence matrix of a bipartite graph) *Let $\mathcal{G} = (V_1, V_2, E)$ be a bipartite graph, $n = |V_1|$ and $m = |V_2|$. The **Incidence Matrix** $M_{\mathcal{G}}$ of $\mathcal{G}$ is a $n \times m$ real matrix which elements are $M_{\mathcal{G}}(i,j) = w_{i,j}$ the weights of the edges or 0 if $ij \notin E$.*

Remark 5 (Adjacency matrix of a bipartite graph) *The adjacency matrix of a bipartite graph is*

$$A_{\mathcal{G}} = \begin{pmatrix} 0 & M_{\mathcal{G}} \\ M_{\mathcal{G}}^T & 0 \end{pmatrix}$$

Remark 6 (Projected graph adjacency matrices) *The adjacency matrices of the projected graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are respectively*

$$M_{\mathcal{G}} M_{\mathcal{G}}^T$$

and

$$M_{\mathcal{G}}^T M_{\mathcal{G}}$$

Remark 7 (Symmetric normalized incidence) *The symmetric normalized incidence matrix is*

$$N = D_1^{-1/2} M_{\mathcal{G}} D_2^{-1/2}$$

Think of a particle walking on the graph and define a probability map on the transitions (edges) as follows : given that the particle is located at the node i at time t, it has a uniform probability of moving to each of its neighbors at time $t+1$. It is easy to check that the stochastic matrix of such a random walk is given by the following

Remark 8 (Random walk stochastic adjacency) *The random walk stochastic matrix of a graph is $D^{-1}A$ where D is an $n \times n$ diagonal matrix containing the degrees of the vertices.*

Remark 9 (Random walk stochastic incidences) *The random walk stochastic matrix of the whole graph (related to the adjacency matrix A), is $D^{-1}A$ where D is an $(n+m) \times (n+m)$ diagonal matrix containing the degrees of the vertices. The corresponding random walk incidence matrices are respectively $W_L = D_1^{-1} M_{\mathcal{G}}$ and $W_R = D_2^{-1} M_{\mathcal{G}}^T$ where D_1 and D_2 are respectively $n \times n$ and $m \times m$ diagonal matrices containing degrees of vertices on their diagonal.*

Remark 10 (Random walk and symmetric normalized incidence)

$$NN^T = D_1^{1/2} W_L W_R D_2^{-1/2}$$

and more generally

$$(NN^T)^k = D_1^{1/2} (W_L W_R)^k D_1^{-1/2}$$

A.1.2 Singular Value Decomposition

The singular value decomposition of a matrix is a golden mine of information on its structure that has many powerful properties both from a mathematical and data-mining viewpoint. As we use it frequently in analysis and algorithms it is more than useful to remind some basic theoretical and computational aspects of it for a better understanding.

Theory

Theorem 13 (Singular Value Decomposition) *For any $M \in \mathbb{K}^{n\times m}$, with $\mathbb{K} = \mathbb{C}$ or $\mathbb{R}$, there exists a unique decomposition of the form*

$$M = U\Sigma V^T$$

where U is unitary $n \times n$, V is unitary $m \times m$ and Σ is diagonal $n \times m$ with decreasing diagonal $\Sigma_{1,1} \geq \Sigma_{2,2} \geq ... \geq 0$ elements.

We refer to $\Sigma_{i,i} = \sigma_i$'s as singular values and to columns of U and V as singular vectors u_i and v_i. These notations allow to write the singular value decomposition using vector notations :

$$M = \sum_{i=1}^{\min(n,m)} \sigma_i u_i v_i^*$$

Proposition 6 (Rank r approximation) *The closest rank $r < n, m$ matrix to M in the sense of Frobenius norm,* i.e. *the solution to*

$$\min_{\mathbb{K}^{n\times m}} \|X - M\|_F \text{ subject to } rank(X) = r$$

is $U\tilde{\Sigma}V^*$ where $\tilde{\Sigma}$ is a copy of Σ where the singular values are set to 0 for $i > r$: $\tilde{\Sigma}_{i,i} = 0$ for $i > r$.

The following shrinkage operator and the minimization problem it solves are of particular importance in our applications.

Definition 28 (Shrinkage Operator)

$$D_\tau(U \operatorname{diag}(\sigma_i)V^T) = U \operatorname{diag}(\max(\sigma_i - \tau, 0))V^T$$

Proposition 7 *In [CT09], the authors prove that*

$$D_\tau(M) = arg \min_X \tau\|X\|_* + \frac{1}{2}\|X - M\|_F^2 \ .$$

Computational issues

There are two main ways to simplify the SVD computation to a symmetric eigen-decomposition problem.

1. One can also obtain singular values of M by taking the square roots of eigen values of MM^T or M^TM. The singular vectors correspond to eigenvectors of the symmetric matrices MM^T and M^TM. Due to numerical error, matrix multiplication may hurt the smallest singular values precision. Hence for some applications, despite the higher storage requirement, one might prefer the following transformation.

2. Computing the SVD of a matrix $M \in \mathbb{R}^{n\times m}$ is equivalent to finding the eigenvalue decomposition of the symmetric matrix $A = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}$.

For obtaining the eigen-decomposition of symmetric matrices most numerical solvers first tridiagonalize the matrix using a Householder transformation for a cost of $O(mn^2)$ flops (where without loss of generality $m \geq n$). The next step consists of computing the eigenvectors using a power method. Each iteration of the power method costs $O(n)$.

Remark 11 (Computational Cost of SVD on Dense Matrices) *If $m > n$, the computational cost of SVD is $O(mn^2)$.*

Lanczos If the matrix A is very large and sparse (as adjacency matrices of real-world graphs are often), then a Lanczos algorithm can be applied to more efficiently use the storage memory. Before further details we need the following

Definition 29 (Krylov spaces) *Given a vector v and a square matrix A, the Krylov subspace of order $r \in \mathbb{N}$ is the subspace*

$$\mathcal{K}_r(A, v) = span\{v, Av, A^2v, \cdots, A^{r-1}v\}$$

A key idea is that singular values of the restriction of a matrix to a Kyrlov space are accurate approximations to its singular vectors. Orthogonal bases of Kyrlov space allow to restrict the heavy part of the computations to tridiagonal matrices, and thereby reduce the computational cost of the decomposition.

There is a family of algorithms named Arnoli algorithms used for building an orthogonal basis of the Krylov subspace $\mathcal{K}_r$. Among Arnoldi algorithms, Lanczos algorithms are of special interest when A is symmetric. These algorithm calculates two matrices T_r and V_r which are respectively symmetric-tridiagonal and orthogonal and $T_r = V_m^T A V_m$. The non-zero coefficients of T_r ($\alpha_j = t_{j,j}$ and $\beta_j = t_{j-1,j} = t_{j,j-1}$) are obtained during the following

Algorithm 7 Lanczos

$v_0 \leftarrow 0$
$v_1 \leftarrow$ unit norm random vector
$\beta_0 \leftarrow 0$
for $j = 1, 2, \ldots, r$ **do**
  $w_j \leftarrow Av_j - \beta_j v_{j-1}$
  $\alpha_j \leftarrow \langle w_j, v_j \rangle$
  $w_j \leftarrow w_j - \alpha_j v_j$
  $\beta_{j+1} \leftarrow \|w_j\|$
  $v_{j+1} \leftarrow w_j / \beta_{j+1}$
end for

The families of matrices T_r and V_r are then used to compute an approximate singular value decomposition of A.

Randomization Recently randomized methods have been introduced for making the SVD computation possible in very large scale cases where the singular values decay is fast and only a few number k of singular elements are required. We quickly review the main sketch of these algorithms, further details can be found in [HMT10]

Algorithm 8 Single-vector Lanczos recursion [LASVD]

Use a Lanczos algorithm to generate a family of real symmetric tridiagonal T_r $(r = 1, 2, \cdots, q)$.

for some $k \leq r$ compute the k relevant eigenvalues of T_k that are approximations to the eigenvalues of A

for each λ, compute a corresponding unit vector z such that $T_k z = \lambda z$, and compute the corresponding approximate eigenvector of $A : y = V_r z$.

Algorithm 9 Randomized SVD approximation

Input : a target rank k and a matrix A of size $m \times n$ where $k \ll m \leq n$

Draw $\Omega \in \mathbb{R}^{n \times k}$ at random

Compute $Y = A\Omega$

Using Gramm-Schmidt compute an orthonormal matrix Q such that $Y \approx QQ^T Y$

Let $B = Q^T A$

Compute the SVD of the small matrix $B = \hat{U}\Sigma V^T$

Form $U = Q\hat{U}$

A.2 Proof of propositions: oracle bounds for regression with sparse-low-rank matrices

We recall that P_U denotes the projection associated with the orthogonal matrix U, $P_U = UU^\top$. The projection $\mathcal{P}_X$ is defined for a matrix X having a singular value decomposition $X = U\Sigma V^\top$ by

$$\mathcal{P}_X(B) = P_U B + BP_V - P_U B P_V \ .$$

It is the projector onto the linear space spanned by the matrices $u_k x^\top$ and $y v_k^\top$ for $1 \leq j, k \leq r$ and $x, y \in \mathbb{R}^n$. The projector onto the orthogonal space is given by $\mathcal{P}_A^\perp(B) = (I - P_U)B(I - P_V)$. The notation Θ_X is used for the sign pattern of the matrix X:

$$\Theta_X = \Big(\operatorname{sign}(X_{i,j}) \Big)_{i,j}$$

and $\Theta_X^\perp$ is the matrix having 1s where $X_{i,j} = 0$ and 0s elsewhere.

A.2.1 Proof of proposition 3

We have for any $X \in \mathcal{S}$, by optimality of $\widehat{X}$:

$$\begin{aligned}
\frac{1}{d}\Big(\|\omega(\widehat{X} - X_0)\|_F^2 - \|\omega(X - X_0)\|_F^2 \Big) &= \frac{1}{d}\Big(\|\omega(\widehat{X})\|_F^2 - \|\omega(X)\|_F^2 - 2\langle \omega(\widehat{X} - X), \omega(X_0)\rangle \Big) \\
&\leq \frac{2}{d}\langle \omega(\widehat{X} - X), y - \omega(X_0)\rangle + \tau\left(\|X\|_* - \|\widehat{X}\|_* \right) + \gamma\left(\|X\|_1 - \|\widehat{X}\|_1 \right) \\
&= \frac{2}{d}\langle \widehat{X} - X, M\rangle + \tau\left(\|X\|_* - \|\widehat{X}\|_* \right) + \gamma\left(\|X\|_1 - \|\widehat{X}\|_1 \right) \ .
\end{aligned}$$

Thanks to trace-duality and ℓ_1-duality we have $\langle M, X\rangle \leq \|M\|_\infty \|X\|_1$ and $\langle M, X\rangle \leq \|M\|_{op}\|X\|_*$ for any X, so for any $\alpha \in [0,1]$:

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_F^2 \leq \frac{1}{d}\|\omega(X - X_0)\|_F^2 + \tau\|X\|_* - \tau\|\widehat{X}\|_* + 2\alpha\|\widehat{X} - X\|_*\|M\|_{op} \\ + \gamma\|X\|_1 - \gamma\|\widehat{X}\|_1 + 2(1-\alpha)\|\widehat{X} - X\|_1\|M\|_\infty \ .$$

now using assumptions $\tau \geq \frac{2\alpha}{d}\|M\|_{op}$, and $\gamma \geq \frac{2(1-\alpha)}{d}\|M\|_\infty$, and then triangle inequality

$$\frac{1}{d}\|\omega(\widehat{X} - X_0)\|_F^2 \leq \frac{1}{d}\|\omega(X - X_0)\|_F^2 + 2\tau\|X\|_* + 2\gamma\|X\|_1 \ . \square$$

A.2.2 Proof of proposition 4.

We introduce the the convex function $g : X \mapsto \tau\|X\|_* + \gamma\|X\|_1$, and let

$$Z = \tau Z_* + \gamma Z_1 = \tau\left(\sum_{j=1}^r u_j v_j^\top + P_{U\perp} W_* P_{V\perp}\right) + \gamma\left(\Theta_X + W_1 \circ \Theta_X^\perp\right)$$

denote an element of the subgradient of g, where $\|W_*\|_{op} \leq 1$ and $\|W_1\|_\infty \leq 1$. There exist two matrices W_1 and W_* such that

$$\langle Z, \widehat{X} - X\rangle = \tau\langle\sum_{j=1}^r u_j v_j^\top, \widehat{X} - X\rangle + \tau\|P_{U\perp}\widehat{X}P_{V\perp}\|_* + \gamma\langle\Theta_X, \widehat{X} - X\rangle + \gamma\|\Theta_X^\perp \circ \widehat{X}\|_1 \ .$$

By optimality, an element of the subgradient of $\mathcal{L}$ at $\widehat{X}$ belongs to the normal cone of $\mathcal{S}$ at $\widehat{X}$, we have $\langle\partial\mathcal{L}(\widehat{X}), \widehat{X} - X\rangle \leq 0$ and on the other hand, by the monotonicity of the subgradient of the convex function g we have $\langle\widehat{X} - X, \widehat{Z} - Z\rangle \geq 0$. Therefore we can deduce

$$\langle\partial\mathcal{L}(\widehat{X}), \widehat{X} - X\rangle - \langle\widehat{Z} - Z, \widehat{X} - X\rangle \leq 0 \ . \tag{A.1}$$

Therefore, given that $\nabla\|\omega(\widehat{X}) - y\|_2^2 = 2\sum_{i=1}^d \Omega_i\langle\Omega_i, \widehat{X}\rangle - Y_i\Omega_i$, we obtain

$$\langle\nabla\|\omega(\widehat{X}) - Y\|_2^2, \widehat{X} - X\rangle = 2\langle\epsilon, \omega(\widehat{X} - X)\rangle - 2\langle\omega(\widehat{X} - X_0), \omega(\widehat{X} - X)\rangle \ .$$

The inequality (A.1) can be written as

$$\frac{2}{d}\langle\omega(\widehat{X} - X_0), \omega(\widehat{X} - X)\rangle \leq \\ \frac{2}{d}\langle\epsilon, \omega(\widehat{X} - X)\rangle - \tau\langle\sum_{j=1}^r u_j v_j^\top, \widehat{X} - X\rangle - \tau\|\mathcal{P}_X^\perp(\widehat{X})\|_* - \gamma\langle\Theta_X, \widehat{X} - X\rangle - \gamma\|\Theta_X^\perp \circ \widehat{X}\|_1 \ . \tag{A.2}$$

Thanks to Cauchy-Schwarz

$$|\langle\sum_{j=1}^r u_j v_j^\top, \widehat{X} - X\rangle| \leq \sqrt{\operatorname{rank} X}\|P_U(\widehat{X} - X)P_V\|_F$$

and

$$|\langle\Theta_X, \widehat{X} - X\rangle| \leq \sqrt{\|X\|_0}\|\Theta_X \circ (\widehat{X} - X)\|_F$$

so we have

$$\frac{2}{d}\langle\omega(\widehat{X}-X_0),\omega(\widehat{X}-X)\rangle \leq$$
$$\frac{2}{d}\langle\epsilon,\omega(\widehat{X}-X)\rangle+\tau\sqrt{\operatorname{rank} X}\|P_U(\widehat{X}-X)P_V\|_F-\tau\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_*+\gamma\sqrt{\|X\|_0}\|\Theta_X\circ(\widehat{X}-X)\|_F-\gamma\|\Theta_X^\perp\circ\widehat{X}\|_1 \tag{A.3}$$

We need to bound $\langle\epsilon,\omega(\widehat{X}-X)\rangle$. For this, note that by definition of M,

$$\langle\epsilon,\omega(\widehat{X}-X)\rangle = \langle M,\widehat{X}-X\rangle \ .$$

We can write the following decomposition, that holds true for any real number $\alpha\in[0,1]$,

$$M=\alpha\Big(\mathcal{P}_X(M)+P_{U^\perp}MP_{V^\perp}\Big)+(1-\alpha)\Big(\Theta_X\circ M+\Theta_X^\perp\circ M\Big) \ .$$

We get by applying triangle inequality, Cauchy-Schwarz, Hölder inequality written for the trace-norm and ℓ_1-norm

$$\langle M,\widehat{X}-X\rangle \leq \alpha\Big(\|\mathcal{P}_X(M)\|_F\|\mathcal{P}_X(\widehat{X}-X)\|_F+\|P_{U^\perp}MP_{V^\perp}\|_{op}\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_*\Big)$$
$$+(1-\alpha)\Big(\|\Theta_X\circ M\|_F\|\Theta_X\circ(\widehat{X}-X)\|_F+\|\Theta_X^\perp\circ M\|_\infty\|\Theta_X^\perp\circ\widehat{X}\|_1\Big) \ .$$

By using rank and support inequalities obtained by applying Cauchy-Schwarz inequality to $\|\mathcal{P}_X(M)\|_F$ and $\|\Theta_X\circ M\|_F$ respectively, we get

$$\langle M,\widehat{X}-X\rangle \leq \alpha\Big(\sqrt{2\operatorname{rank} X}\|M\|_{op}\|\mathcal{P}_X(\widehat{X}-X)\|_F+\|M\|_{op}\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_*\Big)$$
$$+(1-\alpha)\Big(\sqrt{\|X\|_0}\|M\|_\infty\|\Theta_X\circ(\widehat{X}-X)\|_F+\|M\|_\infty\|\Theta_X^\perp\circ\widehat{X}\|_1\Big) \ . \tag{A.4}$$

Now by using

$$2\langle\omega(\widehat{X}-X_0),\omega(\widehat{X}-X)\rangle = \|\omega(\widehat{X}-X_0)\|_2^2+\|\omega(\widehat{X}-X)\|_2^2-\|\omega(X-X_0)\|_2^2$$

we can rewrite the inequality (A.3) as follows:

$$\frac{1}{d}\Big(\|\omega(\widehat{X}-X_0)\|_2^2+\|\omega(\widehat{X}-X)\|_2^2-\|\omega(X-X_0)\|_2^2\Big)$$
$$\leq \frac{2\alpha}{d}\Big(\sqrt{2\operatorname{rank} X}\|M\|_{op}\|\mathcal{P}_X(\widehat{X}-X)\|_F+\|M\|_{op}\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_*\Big)$$
$$+\frac{2(1-\alpha)}{d}\Big(\sqrt{\|X\|_0}\|M\|_\infty\|\widehat{X}-X\|_F+\|M\|_\infty\|\Theta_X^\perp\circ\widehat{X}\|_1\Big)$$
$$+\tau\sqrt{\operatorname{rank} X}\|P_U(\widehat{X}-X)P_V\|_F-\tau\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_*+\gamma\sqrt{\|X\|_0}\|\Theta_X\circ(\widehat{X}-X)\|_F-\gamma\|\Theta_X^\perp\circ\widehat{X}\|_1$$
$$\leq \sqrt{\operatorname{rank} X}\Big(\frac{2\sqrt{2}}{d}\alpha\|M\|_{op}+\tau\Big)\|\widehat{X}-X\|_F+\sqrt{\|X\|_0}\Big(\frac{2(1-\alpha)}{d}\|M\|_\infty+\gamma\Big)\|\widehat{X}-X\|_F \tag{A.5}$$

the last inequality being due to the assumption $\tau \geq \frac{2\alpha}{d}\|M\|_{op}$, and $\gamma \geq \frac{2(1-\alpha)}{d}\|M\|_\infty$. So finally, and by using again this assumption,

$$\frac{1}{d}\Big(\|\omega(\widehat{X}-X_0)\|_2^2+\|\omega(\widehat{X}-X)\|_2^2\Big) \leq \frac{1}{d}\|\omega(X-X_0)\|_2^2+\Big(\sqrt{\operatorname{rank} X}\tau(\sqrt{2}+1)+2\sqrt{\|X\|_0}\gamma\Big)\|\widehat{X}-X\|_F$$
$$\leq \|\omega(X-X_0)\|_2^2 + \frac{\mu}{\sqrt{d}}\Big(\sqrt{\operatorname{rank} X}\tau(\sqrt{2}+1)+2\sqrt{\|X\|_0}\gamma\Big)\|\omega(\widehat{X}-X)\|_F \quad \text{(A.6)}$$

and $bx - x^2 \leq \left(\frac{b}{2}\right)^2$ gives

$$\frac{1}{d}\|\omega(\widehat{X}-X_0)\|_2^2 \leq \frac{1}{d}\|\omega(X-X_0)\|_2^2 + \frac{\mu^2}{4d}\Big(\tau\sqrt{\operatorname{rank} X}(\sqrt{2}+1)+2\sqrt{\|X\|_0}\gamma\Big)^2 .$$

The special case result and corollary results follows by using $\frac{1}{d}\|\omega(X_1-X_2)\|_2^2 \geq \mu^{-2}\|X_1-X_2\|_F^2$ and setting $X = X_0$:

$$\|\widehat{X}-X_0\|_F^2 \leq \frac{\mu^4}{d}\Big(\sqrt{\operatorname{rank} X}\tau(\sqrt{2}+1)+2\sqrt{\|X\|_0}\gamma\Big)^2 . \quad \square$$

A.2.3 Proof of proposition 5

We recall the identity

$$2\langle\omega(\widehat{X}-X_0),\omega(\widehat{X}-X)\rangle = \|\omega(\widehat{X}-X_0)\|_2^2 + \|\omega(\widehat{X}-X)\|_2^2 - \|\omega(X-X_0)\|_2^2 .$$

It shows that if $\langle\omega(\widehat{X}-X_0),\omega(\widehat{X}-X)\rangle \leq 0$, then the bound trivially holds. Therefore lets assume $\langle\omega(\widehat{X}-X_0),\omega(\widehat{X}-X)\rangle > 0$. In this case the inequality (A.3) implies

$$\tau\|P_{U\perp}\widehat{X}P_{V\perp}\|_* + \gamma\|\Theta_X^\perp\circ\widehat{X}\|_1 \leq \frac{2}{d}\langle\epsilon,\omega(\widehat{X}-X)\rangle + \tau\|\mathcal{P}_X(\widehat{X}-X)\|_* + \gamma\|\Theta_X\circ(\widehat{X}-X)\|_1 . \quad \text{(A.7)}$$

On the other hand for any $\alpha \in [0,1]$

$$\langle M,\widehat{X}-X\rangle \leq \alpha\|M\|_{op}\Big(\|\mathcal{P}_X(\widehat{X}-X)\|_* + \|\mathcal{P}_X^\perp(\widehat{X}-X)\|_*\Big)$$
$$+ (1-\alpha)\|M\|_\infty\Big(\|\Theta_X\circ(\widehat{X}-X)\|_1 + \|\Theta_X^\perp\circ(\widehat{X}-X)\|_*\Big) . \quad \text{(A.8)}$$

By (A.7) and (A.8) we get

$$(\tau - \frac{2\alpha}{d}\|M\|_{op})\|\mathcal{P}_X^\perp(\widehat{X}-X)\|_* + (\gamma - \frac{2(1-\alpha)}{d}\|M\|_\infty)\|\Theta_X^\perp\circ(\widehat{X}-X)\|_1$$
$$\leq (\tau + \frac{2\alpha}{d}\|M\|_{op})\|\mathcal{P}_X(\widehat{X}-X)\|_* + (\gamma + \frac{2(1-\alpha)}{d}\|M\|_\infty)\|\Theta_X\circ(\widehat{X}-X)\|_1 . \quad \text{(A.9)}$$

For $\tau \geq \frac{3\alpha}{d}\|M\|_{op}$ and $\gamma \geq \frac{3(1-\alpha)}{d}\|M\|_\infty$, by using the fact that for $x \geq 3$, $\frac{x+2}{x-2} \geq 5$, the following holds true

$$\|\mathcal{P}_X^\perp(\widehat{X}-X)\|_* + \beta\|\Theta_X^\perp\circ(\widehat{X}-X)\|_1 \leq 5\|\mathcal{P}_X(\widehat{X}-X)\|_* + 5\beta\|\Theta_X\circ(\widehat{X}-X)\|_1$$

where $\beta = \frac{\gamma}{\tau}$. This proves that $\widehat{X}-X \in \mathcal{C}_{X,5,\beta}$, and therefore by definition of μ,

$$\|\mathcal{P}_X(\widehat{X}-X)\|_F \leq \frac{\mu_{5,\beta}(X)}{d} \text{ and } \|\Theta_X \circ (\widehat{X}-X)\|_F \leq \frac{\mu_{5,\beta}(X)}{d}\|\omega(\widehat{X}-X)\|_2 \,.$$

By using the inequality (A.4) we have

$$\begin{aligned}
&\frac{1}{d}\Big(\|\omega(\widehat{X}-X_0)\|_2^2 + \|\omega(\widehat{X}-X)\|_2^2 - \|\omega(X-X_0)\|_2^2\Big) \\
&\qquad \leq 2\alpha\Big(\sqrt{2\,\mathrm{rank}\,X}\|M\|_{op}\|\mathcal{P}_X(\widehat{X}-X)\|_F + \|M\|_{op}\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_*\Big) \\
&\qquad + 2(1-\alpha)\Big(\sqrt{\|X\|_0}\|M\|_\infty\|\Theta_X \circ \widehat{X} - X\|_F + \|M\|_\infty\|\Theta_X^\perp \circ \widehat{X}\|_1\Big) \\
&+\tau\sqrt{\mathrm{rank}\,X}\|P_U(\widehat{X}-X)P_V\|_F - \tau\|P_{U^\perp}\widehat{X}P_{V^\perp}\|_* + \gamma\sqrt{\|X\|_0}\|\Theta_X \circ (\widehat{X}-X)\|_F - \gamma\|\Theta_X^\perp \circ \widehat{X}\|_1 \,.
\end{aligned} \tag{A.10}$$

After noticing that by writing $\mu_{5,\beta}(X) = \mu(X)$,

$$\|P_U(\widehat{X}-X)P_V\|_F \leq \|\mathcal{P}_X(\widehat{X}-X)\|_F \leq \frac{\mu(X)}{\sqrt{d}}\|\omega(X-X_0)\|_2$$

and similarly for the ℓ_1 norm $\|\Theta \circ (\widehat{X}-X)\|_F \leq \frac{\mu(X)}{\sqrt{d}}\|\omega(X-X_0)\|_2$, we can write

$$\begin{aligned}
&\frac{1}{d}\Big(\|\omega(\widehat{X}-X_0)\|_2^2 + \|\omega(\widehat{X}-X)\|_2^2 - \|\omega(X-X_0)\|_2^2\Big) \\
&\quad \leq \frac{\mu(X)}{\sqrt{d}}\tau\sqrt{\mathrm{rank}\,X}\Big(1+\frac{2\sqrt{2}}{3}\Big)\|\omega(\widehat{X}-X)\|_F + \frac{\mu(X)}{\sqrt{d}}\gamma\sqrt{\|X\|_0}\Big(1+\frac{2}{3}\Big)\|\omega(\widehat{X}-X)\|_F \,.
\end{aligned}$$

So by $bx - x^2 \leq \left(\frac{b}{2}\right)^2$ we finally get

$$\frac{1}{d}\|\omega(\widehat{X}-X_0)\|_2^2 \leq \frac{1}{d}\|\omega(X-X_0)\|_2^2 + \frac{\mu(X)^2}{4d}\Big(\tau(1+\frac{2\sqrt{2}}{3})\sqrt{\mathrm{rank}\,X} + \gamma\frac{5}{3}\sqrt{\|X\|_0}\Big)^2 \,. \square$$

A.3 Proof of propositions: oracle bounds for prediction of feature autoregressive graphs

From now on, we use the notation $\|(A,a)\|_F^2 = \|A\|_F^2 + \|a\|_2^2$ and $\langle (A,a),(B,b)\rangle = \langle A,B\rangle + \langle a,b\rangle$ for any $A,B \in \mathbb{R}^{T\times d}$ and $a,b \in \mathbb{R}^d$.

Let us introduce the linear mapping $\Phi : \mathbb{R}^{n\times n} \times \mathbb{R}^{d\times d} \to \mathbb{R}^{T\times d} \times \mathbb{R}^d$ given by

$$\Phi(A,W) = \Big(\frac{1}{\sqrt{T}}\mathbf{X}_{T-1}W, \omega(A) - W^\top\omega(A_T)\Big).$$

Using this mapping, the objective (4.3) can be written in the following reduced way:

$$\mathcal{L}(A,W) = \frac{1}{d}\Big\|\Big(\frac{1}{\sqrt{T}}\mathbf{X}_T, 0\Big) - \Phi(A,W)\Big\|_F^2 + \gamma\|A\|_1 + \tau\|A\|_* + \kappa\|W\|_1.$$

Recalling that the error writes, for any A and W:

$$\mathcal{E}(A,W)^2 = \frac{1}{d}\|(W-W_0)^\top\omega(A_T) - \omega(A-A_{T+1})\|_F^2 + \frac{1}{dT}\|\mathbf{X}_{T-1}(W-W_0)\|_F^2,$$

we have

$$\mathcal{E}(A,W)^2 = \frac{1}{d}\big\|\Phi(A - A_{T+1}, W - W_0)\big\|_F^2.$$

Let us introduce also the empirical risk

$$R_n(A,W) = \frac{1}{d}\Big\|\Big(\frac{1}{\sqrt{T}}\mathbf{X}_T, 0\Big) - \Phi(A,W)\Big\|_F^2.$$

The proofs of Theorem 9 and 10 are based on tools developped in [KLT11] and [BRT09]. However, the context considered here is very different from the setting considered in these papers, so our proofs require a different scheme.

A.3.1 Proof of Theorem 9

First, note that

$$\begin{aligned} &R_n(\hat{A},\hat{W}) - R_n(A,W) \\ &\qquad = \frac{1}{d}\Big(\|\Phi(\hat{A},\hat{W})\|_F^2 - \|\Phi(A,W)\|_F^2 - 2\langle(\frac{1}{\sqrt{T}}\mathbf{X}_T, 0), \Phi(\hat{A}-A, \hat{W}-W)\rangle\Big). \end{aligned}$$

Since

$$\begin{aligned} &\frac{1}{d}\Big(\|\Phi(\hat{A},\hat{W})\|_F^2 - \|\Phi(A,W)\|_F^2\Big) \\ &\qquad = \mathcal{E}(\hat{A},\hat{W})^2 - \mathcal{E}(A,W)^2 + \frac{2}{d}\langle\Phi(\hat{A}-A,\hat{W}-W), \Phi(A_{T+1},W_0)\rangle, \end{aligned}$$

we have

$$\begin{aligned} &R_n(\hat{A},\hat{W}) - R_n(A,W) \\ &\quad = \mathcal{E}(\hat{A},\hat{W})^2 - \mathcal{E}(A,W)^2 + \frac{2}{d}\langle\Phi(\hat{A}-A,\hat{W}-W), \Phi(A_{T+1},W_0) - (\frac{1}{\sqrt{T}}\mathbf{X}_T, 0)\rangle \\ &\quad = \mathcal{E}(\hat{A},\hat{W})^2 - \mathcal{E}(A,W)^2 + \frac{2}{d}\langle\Phi(\hat{A}-A,\hat{W}-W), (-\frac{1}{\sqrt{T}}\mathbf{N}_T, N_{T+1})\rangle. \end{aligned}$$

The next Lemma will come in handy several times in the proofs.

Lemma 3 *For any $A \in \mathbb{R}^{n\times n}$ and $W \in \mathbb{R}^{d\times d}$ we have*

$$\langle(\frac{1}{\sqrt{T}}\mathbf{N}_T, -N_{T+1}), \Phi(A,W)\rangle = \langle(M, \frac{1}{T}\Xi), (A,W)\rangle = \frac{1}{T}\langle W, \Xi\rangle + \langle A, M\rangle.$$

This Lemma follows from a direct computation, and the proof is thus omitted. This Lemma entails, together with (4.4), that

$$\begin{aligned} \mathcal{E}(\hat{A},\hat{W})^2 \le \mathcal{E}(A,W)^2 &+ \frac{2}{dT}\langle\hat{W}-W, \Xi\rangle + \frac{2}{d}\langle\hat{A}-A, M\rangle \\ &+ \tau(\|A\|_* - \|\widehat{A}\|_*) + \gamma(\|A\|_1 - \|\widehat{A}\|_1) + \kappa(\|W\|_1 - \|\widehat{W}\|_1). \end{aligned}$$

Now, using Hölder's inequality and the triangle inequality, and introducing $\alpha \in (0,1)$, we obtain

$$\begin{aligned} \mathcal{E}(\hat{A},\hat{W})^2 \le \mathcal{E}(A,W)^2 &+ \Big(\frac{2\alpha}{d}\|M\|_{\mathrm{op}} - \tau\Big)\|\hat{A}\|_* + \Big(\frac{2\alpha}{d}\|M\|_{\mathrm{op}} + \tau\Big)\|A\|_* \\ &+ \Big(\frac{2(1-\alpha)}{d}\|M\|_\infty - \gamma\Big)\|\hat{A}\|_1 + \Big(\frac{2(1-\alpha)}{d}\|M\|_\infty + \gamma\Big)\|A\|_1 \\ &+ \Big(\frac{2}{dT}\|\Xi\|_\infty - \kappa\Big)\|\hat{W}\|_1 + \Big(\frac{2}{dT}\|\Xi\|_\infty + \kappa\Big)\|W\|_1, \end{aligned}$$

which concludes the proof of Theorem 9, using (4.6). □

A.3.2 Proof of Theorem 10

Let $A \in \mathbb{R}^{n\times n}$ and $W \in \mathbb{R}^{d\times d}$ be fixed, and let $A = U\,\mathrm{diag}(\sigma_1,\dots,\sigma_r)V^\top$ be the SVD of A. Recalling that $\circ$ is the entry-wise product, we have $A = \Theta_A \circ |A| + \Theta_A^\perp \circ A$, where $\Theta_A \in \{0,\pm 1\}^{n\times n}$ is the entry-wise sign matrix of A and $\Theta_A^\perp \in \{0,1\}^{n\times n}$ is the orthogonal sparsity pattern of A.

The definition (4.4) of $(\hat{A},\hat{W})$ is equivalent to the fact that one can find $\hat{G} \in \partial\mathcal{L}(\hat{A},\hat{W})$ (an element of the subgradient of $\mathcal{L}$ at $(\hat{A},\hat{W})$) that belongs to the normal cone of $\mathcal{A}\times\mathcal{W}$ at $(\hat{A},\hat{W})$. This means that for such a $\hat{G}$, and any $A \in \mathcal{A}$ and $W \in \mathcal{W}$, we have

$$\langle \hat{G}, (\hat{A}-A, \hat{W}-W)\rangle \le 0. \tag{A.11}$$

Any subgradient of the function $g(A) = \tau\|A\|_* + \gamma\|A\|_1$ writes

$$Z = \tau Z_* + \gamma Z_1 = \tau\Big(UV^\top + \mathcal{P}_A^\perp(G_*)\Big) + \gamma\Big(\Theta_A + G_1 \circ \Theta_A^\perp\Big)$$

for some $\|G_*\|_{\mathrm{op}} \le 1$ and $\|G_1\|_\infty \le 1$ (see for instance [**?**]). So, if $\hat{Z} \in \partial g(\hat{A})$, we have, by monotonicity of the sub-differential, that for any $Z \in \partial g(A)$

$$\langle \hat{Z}, \hat{A}-A\rangle = \langle \hat{Z}-Z, \hat{A}-A\rangle + \langle Z, \hat{A}-A\rangle \ge \langle Z, \hat{A}-A\rangle,$$

and, by duality, we can find Z such that

$$\langle Z, \widehat{A}-A\rangle = \tau\langle UV^\top, \widehat{A}-A\rangle + \tau\|\mathcal{P}_A^\perp(\widehat{A})\|_* + \gamma\langle \Theta_A, \widehat{A}-A\rangle + \gamma\|\Theta_A^\perp \circ \widehat{A}\|_1.$$

By using the same argument with the function $W \mapsto \|W\|_1$ and by computing the gradient of the empirical risk $(A,W) \mapsto R_n(A,W)$, Equation (A.11) entails that

$$\begin{aligned}
&\frac{2}{d}\langle \Phi(\widehat{A}-A_{T+1}, \widehat{W}-W_0), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle \\
&\quad\le \frac{2}{d}\langle (\frac{1}{\sqrt{T}}\mathbf{N}_T, -N_{T+1}), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle - \tau\langle UV^\top, \widehat{A}-A\rangle - \tau\|\mathcal{P}_A^\perp(\widehat{A})\|_* \\
&\qquad - \gamma\langle \Theta_A, \widehat{A}-A\rangle - \gamma\|\Theta_A^\perp \circ \widehat{A}\|_1 - \kappa\langle \Theta_W, \widehat{W}-W\rangle - \kappa\|\Theta_W^\perp \circ \widehat{W}\|_1.
\end{aligned} \tag{A.12}$$

Using Pythagora's theorem, we have

$$\begin{aligned}
&2\langle \Phi(\widehat{A}-A_{T+1}, \hat{W}-W_0), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle \\
&\quad= \|\Phi(\widehat{A}-A_{T+1}, \hat{W}-W_0)\|_2^2 + \|\Phi(\widehat{A}-A, \widehat{W}-W)\|_2^2 - \|\Phi(A-A_{T+1}, W-W_0)\|_2^2.
\end{aligned} \tag{A.13}$$

It shows that if $\langle \Phi(\widehat{A}-A_{T+1}, W-W_0), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle \le 0$, then Theorem 10 trivially holds. Let us assume that

$$\langle \Phi(\widehat{A}-A_{T+1}, W-W_0), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle > 0. \tag{A.14}$$

Using Hölder's inequality, we obtain

$$\begin{aligned}
|\langle UV^\top, \hat{A}-A\rangle| &= |\langle UV^\top, \mathcal{P}_A(\hat{A}-A)\rangle| \le \|UV^\top\|_{\mathrm{op}}\|\mathcal{P}_A(\hat{A}-A)\|_* = \|\mathcal{P}_A(\hat{A}-A)\|_*, \\
|\langle \Theta_A, \hat{A}-A\rangle| &= |\langle \Theta_A, \Theta_A \circ (\hat{A}-A)\rangle| \le \|\Theta_A\|_\infty\|\Theta_A \circ (\hat{A}-A)\|_1 = \|\Theta_A \circ (\hat{A}-A)\|_1,
\end{aligned}$$

and the same is done for $|\langle \Theta_W, \hat{W}-W\rangle| \le \|\Theta_W \circ (\hat{W}-W)\|_1$. So, when (A.14) holds, we obtain by rearranging the terms of (A.12):

$$\begin{aligned}
&\tau\|\mathcal{P}_A^\perp(\widehat{A}-A)\|_* + \gamma\|\Theta_A^\perp \circ (\widehat{A}-A)\|_1 + \kappa\|\Theta_W^\perp \circ (\widehat{W}-W)\|_1 \\
&\qquad\le \tau\|\mathcal{P}_A(\hat{A}-A)\|_* + \gamma\|\Theta_A \circ (\hat{A}-A)\|_1 + \kappa\|\Theta_W \circ (\hat{W}-W)\|_1 \\
&\qquad\quad + \frac{2}{d}\langle (\frac{1}{\sqrt{T}}\mathbf{N}_T, -N_{T+1}), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle.
\end{aligned} \tag{A.15}$$

Using Lemma 3, together with Hölder's inequality, we have for any $\alpha \in (0,1)$:

$$\begin{aligned}\langle(\frac{1}{\sqrt{T}}\mathbf{N}_T, -N_{T+1}), \Phi(\widehat{A}-A, \widehat{W}-W)\rangle &= \langle M, \hat{A}-A\rangle + \frac{1}{T}\langle \Xi, \hat{W}-W\rangle \\ &\leq \alpha\|M\|_{\mathrm{op}}\|\mathcal{P}_A(\hat{A}-A)\|_* + \alpha\|M\|_{\mathrm{op}}\|\mathcal{P}_A^{\perp}(\hat{A}-A)\|_* \\ &\quad + (1-\alpha)\|M\|_\infty\|\Theta_A\circ(\hat{A}-A)\|_1 + (1-\alpha)\|M\|_\infty\|\Theta_A^{\perp}\circ(\hat{A}-A)\|_1 \\ &\quad + \frac{1}{T}\|\Xi\|_\infty(\|\Theta_W\circ(\hat{W}-W)\|_1 + \|\Theta_W^{\perp}\circ(\hat{W}-W)\|_1)\,.\end{aligned} \tag{A.16}$$

Now, using (A.15) together with (A.16), we obtain

$$\begin{aligned}&\Big(\tau - \frac{2\alpha}{d}\|M\|_{\mathrm{op}}\Big)\|\mathcal{P}_A^{\perp}(\hat{A}-A)\|_* + \Big(\gamma - \frac{2(1-\alpha)}{d}\|M\|_\infty\Big)\|\Theta_A^{\perp}\circ(\hat{A}-A)\|_1 \\ &\quad + \Big(\kappa - \frac{2}{dT}\|\Xi\|_\infty\Big)\|\Theta_W^{\perp}\circ(\hat{W}-W)\|_1 \\ &\leq \Big(\tau + \frac{2\alpha}{d}\|M\|_{\mathrm{op}}\Big)\|\mathcal{P}_A(\hat{A}-A)\|_* + \Big(\gamma + \frac{2(1-\alpha)}{d}\|M\|_\infty\Big)\|\Theta_A\circ(\hat{A}-A)\|_1 \\ &\quad + \Big(\kappa + \frac{2}{dT}\|\Xi\|_\infty\Big)\|\Theta_W\circ(\hat{W}-W)\|_1\end{aligned}$$

which proves, using (4.7), that

$$\tau\|\mathcal{P}_A^{\perp}(\hat{A}-A)\|_* + \gamma\|\Theta_A^{\perp}\circ(\hat{A}-A)\|_1 \leq 5\tau\|\mathcal{P}_A(\hat{A}-A)\|_* + 5\gamma\|\Theta_A\circ(\hat{A}-A)\|_1.$$

This proves that $\hat{A}-A \in \mathcal{C}_2(A,5,\gamma/\tau)$. In the same way, using (A.15) with $A=\hat{A}$ together with (A.16), we obtain that $\hat{W}-W \in \mathcal{C}_1(W,5)$.

Now, using together (A.12), (A.13) and (A.16) , and the fact that the Cauchy-Schwarz inequality entails

$$\begin{aligned}\|\mathcal{P}_A(\hat{A}-A)\|_* &\leq \sqrt{\operatorname{rank} A}\|\mathcal{P}_A(\hat{A}-A)\|_F, \quad |\langle UV^\top, \hat{A}-A\rangle| \leq \sqrt{\operatorname{rank} A}\|\mathcal{P}_A(\hat{A}-A)\|_F, \\ \|\Theta_A\circ(\hat{A}-A)\|_1 &\leq \sqrt{\|A\|_0}\|\Theta_A\circ(\hat{A}-A)\|_F, \quad |\langle \Theta_A, \hat{A}-A\rangle| \leq \sqrt{\|A\|_0}\|\Theta_A\circ(\hat{A}-A)\|_F\ .\end{aligned}$$

and similarly for $\hat{W}-W$, we arrive at

$$\begin{aligned}&\|\Phi(\widehat{A}-A_{T+1}, \hat{W}-W_0)\|_2^2 + \|\Phi(\widehat{A}-A, \widehat{W}-W)\|_2^2 - \|\Phi(A-A_{T+1}, W-W_0)\|_2^2 \\ &\quad\leq \Big(\frac{2\alpha}{d}\|M\|_{\mathrm{op}} + \tau\Big)\sqrt{\operatorname{rank} A}\|\mathcal{P}_A(\hat{A}-A)\|_F + \Big(\frac{2\alpha}{d}\|M\|_{\mathrm{op}} - \tau\Big)\|\mathcal{P}_A^{\perp}(\hat{A}-A)\|_* \\ &\qquad + \Big(\frac{2\alpha}{d}\|M\|_\infty + \gamma\Big)\sqrt{\|A\|_0}\|\Theta_A\circ(\hat{A}-A)\|_F + \Big(\frac{2\alpha}{d}\|M\|_\infty - \gamma\Big)\|\Theta_A^{\perp}\circ(\hat{A}-A)\|_1 \\ &\qquad + \Big(\frac{2\alpha}{dT}\|\Xi\|_\infty + \kappa\Big)\sqrt{\|W\|_0}\|\Theta_W\circ(\hat{W}-W)\|_F + \Big(\frac{2\alpha}{dT}\|\Xi\|_\infty - \kappa\Big)\|\Theta_W^{\perp}\circ(\hat{W}-W)\|_1,\end{aligned}$$

which leads, using (4.7), to

$$\begin{aligned}&\frac{1}{d}\|\Phi(\widehat{A}-A_{T+1}, \hat{W}-W_0)\|_2^2 + \frac{1}{d}\|\Phi(\widehat{A}-A, \widehat{W}-W)\|_2^2 - \frac{1}{d}\|\Phi(A-A_{T+1}, W-W_0)\|_2^2 \\ &\quad\leq \frac{5\tau}{3}\sqrt{\operatorname{rank} A}\|\mathcal{P}_A(\hat{A}-A)\|_F + \frac{5\gamma}{3}\sqrt{\|A\|_0}\|\Theta_A\circ(\hat{A}-A)\|_F + \frac{5\kappa}{3}\sqrt{\|W\|_0}\|\Theta_W\circ(\hat{W}-W)\|_F.\end{aligned}$$

Since $\hat{A}-A \in \mathcal{C}_2(A,5,\gamma/\tau)$ and $\hat{W}-W \in \mathcal{C}_1(W,5)$, we obtain using Assumption 20 and $ab \leq (a^2+b^2)/2$:

$$\begin{aligned}&\frac{1}{d}\|\Phi(\widehat{A}-A_{T+1}, \hat{W}-W_0)\|_2^2 + \frac{1}{d}\|\Phi(\widehat{A}-A, \widehat{W}-W)\|_2^2 \\ &\quad\leq \frac{1}{d}\|\Phi(A-A_{T+1}, W-W_0)\|_2^2 + \frac{25}{18}\mu_2(A,W)^2\big(\operatorname{rank}(A)\tau^2 + \|A\|_0\gamma^2\big) \\ &\qquad + \frac{25}{36}\mu_1(W)^2\|W\|_0\kappa^2 + \frac{1}{d}\|\Phi(\widehat{A}-A, \widehat{W}-W)\|_2^2,\end{aligned}$$

which concludes the proof of Theorem 10. □

A.3.3 Proof of Theorem 12

For the proof of (4.8), we simply use the fact that $\frac{1}{dT}\|\mathbf{X}_{T-1}(\hat{W}-W_0)\|_F^2 \leq \mathcal{E}(\hat{A},\hat{W})^2$ and use Theorem 11. Then we take $W = W_0$ in the infimum over A, W.

For (4.9), we use the fact that since $\hat{W} - W_0 \in \mathcal{C}_1(W_0, 5)$, we have (see the Proof of Theorem 10),

$$\begin{aligned}\|\hat{W}-W_0\|_1 &\leq 6\sqrt{\|W_0\|_0}\|\Theta_W \circ (\hat{W}-W_0)\|_F \\ &\leq 6\sqrt{\|W_0\|_0}\|\mathbf{X}_{T-1}(\hat{W}-W_0)\|_F/\sqrt{dT} \\ &\leq 6\sqrt{\|W_0\|_0}\mathcal{E}(\hat{A},\hat{W}),\end{aligned}$$

and then use again Theorem 11. The proof of (4.10) follows exactly the same scheme. □

A.3.4 Concentration inequalities for the noise processes

The control of the noise terms M and Ξ is based on recent developments on concentration inequalities for random matrices, see for instance [Tro10]. Moreover, the assumption on the dynamics of the features's noise vector $(N_t)_{t\geq 0}$ is quite general, since we only assumed that this process is a martingale increment. Therefore, our control of the noise Ξ rely in particular on martingale theory.

Proposition 8 *Under Assumption 3, the following inequalities hold for any $x > 0$. We have*

$$\Big\|\frac{1}{d}\sum_{j=1}^{d}(N_{T+1})_j\Omega_j\Big\|_{\mathrm{op}} \leq \sigma v_{\Omega,\mathrm{op}}\sqrt{\frac{2(x+\log(2n))}{d}} \tag{A.17}$$

with a probability larger than $1-e^{-x}$. We have

$$\Big\|\frac{1}{d}\sum_{j=1}^{d}(N_{T+1})_j\Omega_j\Big\|_{\infty} \leq \sigma v_{\Omega,\infty}\sqrt{\frac{2(x+2\log n)}{d}} \tag{A.18}$$

with a probability larger than $1-2e^{-x}$, and finally

$$\Big\|\frac{1}{T+1}\sum_{t=1}^{T+1}\omega(A_{t-1})N_t^\top\Big\|_{\infty} \leq \sigma\sigma_\omega\sqrt{\frac{2e(x+2\log d+\ell_T)}{T+1}} \tag{A.19}$$

with a probability larger than $1-14e^{-x}$, where

$$\ell_T = 2\max_{j=1,\dots,d}\log\log\left(\frac{\sum_{t=1}^{T+1}\omega_j(A_{t-1})^2}{T+1} \vee \frac{T+1}{\sum_{t=1}^{T+1}\omega_j(A_{t-1})^2} \vee e\right).$$

Proof.
For the proofs of Inequalities (A.17) and (A.18), we use the fact that $(N_{T+1})_1, \dots, (N_{T+1})_d$ are independent (scalar) subgaussian random variables.

From Assumption 3, we have for any $n \times n$ deterministic self-adjoint matrices X_j that

$$\mathbb{E}[\exp(\lambda(N_{T+1})_j X_j)] \preceq \exp(\sigma^2\lambda^2 X_j^2/2)\ ,$$

where $\preceq$ stands for the semidefinite order on self-adjoint matrices. Using Corollary 3.7 from [Tro10], this leads for any $x > 0$ to

$$\mathbb{P}\Big[\lambda_{\max}\Big(\sum_{j=1}^{d}(N_{T+1})_j X_j\Big) \geq x\Big] \leq n\exp\Big(-\frac{x^2}{2v^2}\Big), \quad \text{where } v^2 = \sigma^2\Big\|\sum_{j=1}^{d} X_j^2\Big\|_{\mathrm{op}}. \tag{A.20}$$

Then, following [Tro10], we consider the dilation operator $\mathcal{L} : \mathbb{R}^{n\times n} \to \mathbb{R}^{2n\times 2n}$ given by

$$\mathcal{L}(\Omega) = \begin{pmatrix} 0 & \Omega \\ \Omega^* & 0 \end{pmatrix}.$$

We have

$$\Big\|\sum_{j=1}^{d}(N_{T+1})_j\Omega_j\Big\|_{\mathrm{op}} = \lambda_{\max}\Big(\mathcal{L}\Big(\sum_{j=1}^{d}(N_{T+1})_j\Omega_j\Big)\Big) = \lambda_{\max}\Big(\sum_{j=1}^{d}(N_{T+1})_j\mathcal{L}(\Omega_j)\Big)$$

and an easy computation gives

$$\Big\|\sum_{j=1}^{d}\mathcal{L}(\Omega_j)^2\Big\|_{\mathrm{op}} = \Big\|\sum_{j=1}^{d}\Omega_j^\top\Omega_j\Big\|_{\mathrm{op}} \vee \Big\|\sum_{j=1}^{d}\Omega_j\Omega_j^\top\Big\|_{\mathrm{op}}.$$

So, using (A.20) with the self-adjoint $X_j = \mathcal{L}(\Omega_j)$ gives

$$\mathbb{P}\Big[\Big\|\sum_{j=1}^{d}(N_{T+1})_j\Omega_j\Big\|_{\mathrm{op}} \geq x\Big] \leq 2n\exp\Big(-\frac{x^2}{2v^2}\Big) \text{ where } v^2 = \sigma^2\Big\|\sum_{j=1}^{d}\Omega_j^\top\Omega_j\Big\|_{\mathrm{op}} \vee \Big\|\sum_{j=1}^{d}\Omega_j\Omega_j^\top\Big\|_{\mathrm{op}},$$

which leads easily to (A.17).

Inequality (A.18) comes from the following standard bound on the sum of independent sub-gaussian random variables:

$$\mathbb{P}\Big[\Big|\frac{1}{d}\sum_{j=1}^{d}(N_{T+1})_j(\Omega_j)_{k,l}\Big| \geq x\Big] \leq 2\exp\Big(-\frac{x^2}{2\sigma^2(\Omega_j)_{k,l}^2}\Big)$$

together with an union bound on $1 \leq k, l \leq n$.

Inequality (A.19) is based on a classical martingale exponential argument together with a peeling argument. We denote by $\omega_j(A_t)$ the coordinates of $\omega(A_t) \in \mathbb{R}^d$ and by $N_{t,k}$ those of N_t, so that

$$\Big(\sum_{t=1}^{T+1}\omega(A_{t-1})N_t^\top\Big)_{j,k} = \sum_{t=1}^{T+1}\omega_j(A_{t-1})N_{t,k}.$$

We fix j, k and denote for short $\varepsilon_t = N_{t,k}$ and $x_t = \omega_j(A_t)$. Since $\mathbb{E}[\exp(\lambda\varepsilon_t)|\mathcal{F}_{t-1}] \leq e^{\sigma^2\lambda^2/2}$ for any $\lambda \in \mathbb{R}$, we obtain by a recursive conditioning with respect to $\mathcal{F}_{T-1}, \mathcal{F}_{T-2}, \ldots, \mathcal{F}_0$, that

$$\mathbb{E}\Big[\exp\Big(\theta\sum_{t=1}^{T+1}\varepsilon_t x_{t-1} - \frac{\sigma^2\theta^2}{2}\sum_{t=1}^{T+1}x_{t-1}^2\Big)\Big] \leq 1.$$

Hence, using Markov's inequality, we obtain for any $v > 0$:

$$\mathbb{P}\Big[\sum_{t=1}^{T+1}\varepsilon_t x_{t-1} \geq x, \sum_{t=1}^{T+1}x_{t-1}^2 \leq v\Big] \leq \inf_{\theta>0}\exp(-\theta x + \sigma^2\theta^2 v/2) = \exp\Big(-\frac{x^2}{2\sigma^2 v}\Big),$$

that we rewrite in the following way:

$$\mathbb{P}\Big[\sum_{t=1}^{T+1}\varepsilon_t x_{t-1} \geq \sigma\sqrt{2vx}, \sum_{t=1}^{T+1} x_{t-1}^2 \leq v\Big] \leq e^{-x}.$$

Let us denote for short $V_T = \sum_{t=1}^{T+1} x_{t-1}^2$ and $S_T = \sum_{t=1}^{T+1}\varepsilon_t x_{t-1}$. We want to replace v by V_T from the previous deviation inequality, and to remove the event $\{V_T \leq v\}$. To do so, we use a peeling argument. We take $v = T+1$ and introduce $v_k = ve^k$ so that the event $\{V_T > v\}$ is decomposed into the union of the disjoint sets $\{v_k < V_T \leq v_{k+1}\}$. We introduce also $\ell_T = 2\log\log\Big(\frac{\sum_{t=1}^{T+1} x_{t-1}^2}{T+1} \vee \frac{T+1}{\sum_{t=1}^{T+1} x_{t-1}^2} \vee e\Big)$.

This leads to

$$\begin{aligned}
\mathbb{P}\Big[S_T \geq \sigma\sqrt{2eV_T(x+\ell_T)}, V_T > v\Big] &= \sum_{k\geq 0}\mathbb{P}\big[S_T \geq \sigma\sqrt{2eV_T(x+\ell_T)}, v_k < V_T \leq v_{k+1}\Big] \\
&= \sum_{k\geq 0}\mathbb{P}\Big[S_T \geq \sigma\sqrt{2v_{k+1}(x+2\log\log(e^k\vee e))}, v_k < V_T \leq v_{k+1}\Big] \\
&\leq e^{-x}(1+\sum_{k\geq 1}k^{-2}) \leq 3.47e^{-x}.
\end{aligned}$$

On $\{V_T \leq v\}$ the proof is the same: we decompose onto the disjoint sets $\{v_{k+1} < V_T \leq v_k\}$ where this time $v_k = ve^{-k}$, and we arrive at

$$\mathbb{P}\Big[S_T \geq \sigma\sqrt{2eV_T(x+\ell_T)}, V_T \leq v\Big] \leq 3.47e^{-x}.$$

This leads to

$$\mathbb{P}\Big[\sum_{t=1}^{T+1}\omega_j(A_{t-1})N_{t,k} \geq \sigma\Big(2e\sum_{t=1}^{T+1}\omega_j(A_{t-1})^2(x+\ell_{T,j})\Big)^{1/2}\Big] \leq 7e^{-x}$$

for any $1 \leq j,k \leq d$, where we introduced

$$\ell_{T,j} = 2\log\log\Big(\frac{\sum_{t=1}^{T+1}\omega_j(A_{t-1})^2}{T+1} \vee \frac{T+1}{\sum_{t=1}^{T+1}\omega_j(A_{t-1})^2} \vee e\Big).$$

The conclusion follows from an union bound on $1 \leq j,k \leq d$. This concludes the proof of Proposition 8.

Bibliography

[AB02] R. Albert and A.-L. Barabàsi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74, 2002.

[AB09] S. Arlot and F. Bach. Data-driven calibration of linear estimators with minimal penalties, 2009.

[ABEV09] J. Abernethy, F. Bach, Th. Evgeniou, and J.-Ph. Vert. A new approach to collaborative filtering: operator estimation with spectral regularization. *JMLR*, 10:803–826, 2009.

[ABFX08] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 2008.

[AEP06] A. Argyriou, Th. Evgeniou, and M. Pontil. Multi-task feature learning. *Proceedings of Neural Information Processing Systems (NIPS)*, 2006.

[APMY07] A. Argyriou, M. Pontil, Ch. Micchelli, and Y. Ying. A spectral regularization framework for multi-task structure learning. *Proceedings of Neural Information Processing Systems (NIPS)*, 2007.

[Arg06] A. Argyriou. *Learning to Integrate Data from Different Sources and Tasks*. PhD thesis, University of London, 2006.

[Bac08] F. Bach. Consistency of trace norm minimization. *Journal of Machine Learning Research*, 9:1019–1048, 2008.

[Bas63] F. M. Bass. A dynamic model of market share and sales behavio. In *Winter Conference American Marketing Association*, pages 269–275, 1963.

[BD09] P.J. Brockwell and R.A. Davis. *Time Series: Theory and Methods*. Springer Series in Statistics. Springer, 2009.

[BEGd07] O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation. *Machine Learning Research 101*, 2007.

[Ber11] D.P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: a survey. *Optimization for Machine Learning*, page 85, 2011.

[BF97] L. Breiman and J. H. Friedman. Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society (JRSS): Series B (Statistical Methodology)*, 59:3–54, 1997.

[BG01] J.R. Bock and D.A. Gough. Predicting protein–protein interactions from primary structure. *Bioinformatics*, 17(5):455–460, 2001.

[BJK78] G. Bassett Jr and R. Koenker. Asymptotic theory of least absolute error regression. *Journal of the American Statistical Association*, pages 618–622, 1978.

[BJMO11] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Structured sparsity through convex optimization. *CoRR*, abs/1109.2397, 2011.

[BMAP12] L. Baldassarre, J. Morales, A. Argyriou, and M. Pontil. A general framework for structured sparsity via proximal optimization. volume 22, 2012.

[BO04] A.L. Barabasi and Z.N. Oltvai. Network biology: Understanding the cell's functional organization. *Nature Review Genetics*, 2004.

[Bod08] A.V. Bodapati. Recommendation systems with purchase data. *Journal of Marketing Research*, 2008.

[Bol01] B. Bollobas. *Random graphs, vol. 73 of Cambridge Studies in Advanced Mathematics. 2nd ed.* Cambridge University Press, Cambridge, 2001.

[BP98] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the seventh international conference on World Wide Web 7*, WWW7, pages 107–117, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.

[BRT09] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of lasso and dantzig selector. *Annals of Statistics*, 37, 2009.

[BT09] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal of Imaging Sciences*, 2(1):183–202, 2009.

[BT10] J. Bien and R. Tibshirani. Sparse estimation of a covariance matrix. *Biometrika*, 2010.

[CCS08] J.F. Cai, E.J. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *Arxiv preprint Arxiv:0810.3286*, 2008.

[CCSA08] J-F Cai, aE. J. Candès, and Z. Shen. A. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2008.

[CFHW12] W. Chen, W. Fang, G. Hu, and Mahoney M. W. On the hyperbolicity of small-world networks and tree-like graphs. *submitted*, 2012.

[CLMW09] E. J. Candès, X. Li, Y. Ma, and John W. Robust principal component analysis? *Journal of ACM*, 8:1–37, 2009.

[CP11] P.L. Combettes and J.C. Pesquet. Proximal splitting methods in signal processing. *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212, 2011.

[CSN09] A. Clauset, C. R. Shalizi, and M. E.J. Newman. Power-law distributions in empirical data. *SIAM Review 51*, 2009.

[CSPW11] V. Chandrasekaran, S. Sanghavi, P.A. Parrilo, and A.S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM J. Opt.*, 21:572–596, 2011.

[CT04] E. J. Candès and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 12(51):4203–4215, 2004.

[CT05] E. Candès and T. Tao. Decoding by linear programming. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2005.

[CT09] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5), 2009.

[CW08] E. J. Candès and M. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 12(51):21–30, 2008.

[DEGJL07] A. D'Aspremont, L. El Ghaoui, M.I. Jordan, and G.R.G. Lanckriet. A direct formulation for sparse pca using semidefinite programming. *SIAM review*, 49(3):434–448, 2007.

[DHM12] M. Dudik, Z. Harchaoui, and J. Malick. Lifted coordinate descent for learning with trace-norm regularization. In *AISTATS*, 2012.

[EK08] N. El Karoui. Operator norm consistent estimation of large-dimensional sparse covariance matrices. *The Annals of Statistics*, 36(6):2717–2756, 2008.

[EK09] N. El Karoui. Operator norm consistent estimation of large-dimensional sparse covariance matrices. *Annals of Statistics*, 2009.

[EMP05] Th. Evgeniou, Ch. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6:615–637, 2005.

[ER59] P. Erdos and A. Renyi. On the evolution of random graphs. In *PUBLICATION OF THE MATHEMATICAL INSTITUTE OF THE HUNGARIAN ACADEMY OF SCIENCES*, pages 17–61, 1959.

[FAD+11] J. Foulds, A. Asuncion, C. DuBois, C. T. Butts, and P. Smyth. A dynamic relational infinite feature model for longitudinal social networks, 2011.

[FH09] D. Fleder and K. Hosanagar. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management Science*, 55, 2009.

[FHT08] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432, 2008.

[FHT10] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 2010.

[GH10] X. Gao and J. Huang. Asymptotic analysis of high-dimensional lad regression with lasso. *Statistica Sinica*, 20(4), 2010.

[GL11] S. Gaiffas and G. Lecue. Sharp oracle inequalities for high-dimensional matrix prediction. *Information Theory, IEEE Transactions on*, 57(10):6942 –6957, oct. 2011.

[GN02] M. Girvan and M.E.J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821, 2002.

[GOB11] E. Grave, G. Obozinski, and F. Bach. Trace lasso: a trace norm regularization for correlated designs. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2187–2195, 2011.

[Gro11] D. Gross. Recovering low-rank matrices from few coefficients in any basis. *IEEE Transactions on Information Theory*, 57, 2011.

[Haz08] E. Hazan. Sparse approximate solutions to semidefinite programs. In *Proceedings of the 8th Latin American conference on Theoretical informatics*, pages 306–316. Springer-Verlag, 2008.

[HJB+09] P. Hu, S.C. Janga, M. Babu, J.J. Díaz-Mejía, G. Butland, W. Yang, O. Pogoutse, X. Guo, S. Phanse, P. Wong, et al. Global functional atlas of escherichia coli encompassing previously uncharacterized proteins. *PLoS biology*, 7(4):e1000096, 2009.

[HKSS12] E. Hazan, S. Kale, and Sh. Shalev-Shwartz. Near-optimal algorithms for online matrix prediction. *CoRR*, 2012.

[HKV08] Y. Hu, Y. Koren, and CH. Volinsky. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008)*, pages 263–272, 2008.

[HMT10] N. Halko, P. G. Martinsson, and J. Tropp. Finding structure with randomness: Stochastic algorithms for constructing approximate matrix decompositions. *In review*, 2010.

[Hof09] P.D. Hoff. Multiplicative latent factor models for description and prediction of social networks. *Computational & Mathematical Organization Theory*, 15(4):261–272, 2009.

[HRH02] P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *Journal of the Royal Statistical Society*, 97, 2002.

[HRVSN10] M. Hue, M. Riffle, J.-P. Vert, and W. Stafford Noble. Large-scale prediction of protein-protein interactions from structures. *BMC Bioinformatics*, 11:144, 2010.

[HSX11] Q. Ho, L. Song, and E. P. Xing. Evolving cluster mixed-membership blockmodel for time-varying networks. *AISTATS*, 2011.

[HTF01] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[Jag11] M. Jaggi. Convex optimization without projection steps. *Arxiv preprint arXiv:1108.1170*, 2011.

[Jam87] G.J.O. Jameson. *Summing and Nuclear Norms in Banach Space Theory*. London Mathematical Society Student Texts. Cambridge University Press, 1987.

[Jen02] F. V. Jensen. *Bayesian Networks and Decision Graphs*. Springer Publishing Company, Incorporated, 2002.

[JMOB11] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for hierarchical sparse coding. *Journal of Machine Learning Research*, 12:2297–2334, 2011.

[JSX11] Y. Jalali, A.and Chen, S. Sanghavi, and H. Xu. Clustering partially observed graphs via convex optimization. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 1001–1008, New York, NY, USA, June 2011. ACM.

[JV08] L. Jacob and J.-P. Vert. Protein-ligand interaction prediction: an improved chemogenomics approach. *Bioinformatics*, 24(19):2149–2156, 2008.

[Kat53] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[KH10] P. N. Krivitsky and M. S. Handcock. A Separable Model for Dynamic Networks. *ArXiv e-prints*, November 2010.

[KK02] J. Kalervo and J. Kekalainen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on information Sustems*, 2002.

[KKY+09] H. Kashima, T. Kato, Y. Yamanishi, M. Sugiyama, and K. Tsuda. Link propagation: A fast semi-supervised learning algorithm for link prediction. In *Proceedings of the SIAM International Conference on Data Mining, SDM 2009*, pages 1099–1110, 2009.

[KL09] J. Kunegis and A. Lommatzsch. Learning spectral graph transformations for link prediction. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 561–568, New York, NY, USA, 2009. ACM.

[KLT11] V. Koltchinskii, K. Lounici, and A. Tsybakov. Nuclear norm penalization and optimal rates for noisy matrix completion. *Annals of Statistics*, 2011.

[KNG11] N. Komodakis, Paragios N., and Tziritas G. Mrf energy minimization and beyond via dual decomposition. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, pages 531–552, 2011.

[Kol09a] E. D. Kolaczyk. *Statistical Analysis of Network Data*. Springer, 2009.

[Kol09b] V. Koltchinskii. The Dantzig selector and sparsity oracle inequalities. *Bernoulli*, 15(3):799–828, 2009.

[Kol09c] V. Koltchinskii. Sparsity in penalized empirical risk minimization. *Ann. Inst. Henri Poincaré Probab. Stat.*, 45(1):7–57, 2009.

[Kor08] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

[Kor10] Y. Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.

[KP08] J. Kim and H. Park. Sparse nonnegative matrix factorization for clustering. Technical report, Georgia Institute of Technology, 2008.

[KSAX10] M. Kolar, L. Song, A. Ahmed, and E. P. Xing. Estimating time-varying networks. *Annals of Applied Statistics*, 2010.

[KX11] M. Kolar and E. P. Xing. On time varying undirected graphs. *in Proceedings of the 14th International Conference on Artifical Intelligence and Statistics AISTATS*, 2011.

[LAB07] J. Leskovec, L. Adamic, and Huberman B. Dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, 1, 2007.

[LBA09] P. Latouche, E. Birmelé, and C. Ambroise. Overlapping stochastic block models with application to the French political blogosphere. *ArXiv e-prints*, October 2009.

[Les08] J. Leskovec. *Dynamics of large networks*. PhD thesis, Machine Learning Department, Carnegie Mellon University, 2008.

[Lew95] A.S. Lewis. The convex analysis of unitarily invariant matrix norms. *Journal of Convex Analysis*, 2:173–183, 1995.

[LGW+09] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2009.

[LKF05] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 2005.

[LNK07a] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58(7):1019–1031, 2007.

[LNK07b] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[LPTvdG11] K. Lounici, M. Pontil, A. B. Tsybakov, and S. van de Geer. Oracle inequalities and optimal inference under group sparsity. *Annals of statistics*, 2011.

[LRS+10] J. Lee, B. Recht, R. Salakhutdinov, N. Srebro, and J. Tropp. Practical large-scale optimization for max-norm regularization. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1297–1305. 2010.

[LSY03] G. Linden, B. Smith, and J. York. Amazon.com recommendations : Item-to-item collaborative filtering. *IEEE Internet Computing*, 2003.

[Luo11] X. Luo. High dimensional low rank and sparse covariance matrix estimation via convex minimization. *Arxiv preprint arXiv:1111.1133*, 2011.

[Mar08] B. Marlin. *Missing Data Problems in Machine Learning*. PhD thesis, University of British Columbia, 2008.

[MGJ09] K. Miller, Th. Griffiths, and M. Jordan. Nonparametric latent feature models for link prediction. In *Advances in Neural Information Processing System 22*, 2009.

[ML10] S.A. Myers and Jure Leskovec. On the convexity of latent social network inference. In *NIPS*, 2010.

[MMF90] V. Mahajan, E. Muller, and Bass F.M. New product diffusion models in marketing: A review and directions for research. *Journal of Marketing*, 54:1–26, 1990.

[Nes05] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005.

[NS01] K. Nowicki and T. Snijders. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96:1077– 1087, 2001.

[Par11] E. Pariser. *The filter bubble : what the Internet is hiding from you*. Viking, 2011.

[Pen03] M. Penrose. *Random geometric graphs*. Oxford studies in probability. Oxford University Press, 2003.

[RBEV10] E. Richard, N. Baskiotis, Th. Evgeniou, and N. Vayatis. Link discovery using graph feature tracking. *Proceedings of Neural Information Processing Systems (NIPS)*, 2010.

[RFP11] H. Raguet, J. Fadili, and G. Peyré. Generalized forward-backward splitting. *Arxiv preprint arXiv:1108.4404*, 2011.

[RGV12] E. Richard, S. Gaiffas, and N. Vayatis. Link prediction in graphs with autoregressive features. *Proceedings of Neural Information Processing Systems (NIPS)*, 2012.

[Rog62] E.M. Rogers. *Diffusion of innovations*. London: The Free Press, 1962.

[RR11] B. Recht and Ch. Re. Parallel stochastic gradient algorithms for large-scale matrix completion. *Submitted for publication*, 2011.

[RSV12] E. Richard, P.-A. Savalle, and N. Vayatis. Estimation of simultaneously sparse and low-rank matrices. In *Proceeding of 29th Annual International Conference on Machine Learning*, 2012.

[SCJ12] P. Sarkar, D. Chakrabarti, and M.I. Jordan. Nonparametric link prediction in dynamic networks. In *Proceeding of 29th Annual International Conference on Machine Learning*, 2012.

[SCM10a] P. Sarkar, D. Chakrabarti, and A. W. Moore. Theoretical justifiation of popular link prediction heuristics. *In Proceedings of COLT*, 2010.

[SCM10b] P. Sarkar, D. Chakrabarti, and A.W. Moore. Theoretical justification of popular link prediction heuristics. In *International Conference on Learning Theory (COLT)*, pages 295–307, 2010.

[SFR09] M. Schmidt, G. Fung, and R. Rosales. Optimization methods for ℓ_1-regularization. Technical report, University of British Columbia, 2009.

[SJT09] A. Sood, G. M. James, and G. J. Tellis. Functional regression: A new model for predicting market penetration of new products. *Marketing Science*, 28(1):36–51, 2009.

[SKX09] L. Song, M. Kolar, and E.P. Xing. Time-varying dynamic bayesian networks, 2009.

[SM06] P. Sarkar and A. Moore. Dynamic social network analysis using latent space models. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Proceedings of Neural Information Processing Systems 18*, pages 1145–1152. MIT Press, Cambridge, MA, 2006.

[SM10] A. Shojaie and G. Michailidis. Discovering graphical granger causality using a truncating lasso penalty. *Bioinformatics*, 2010.

[Spi83] J.E. Spingarn. Partial inverse of a monotone operator. *Applied mathematics & optimization*, 10(1):247–265, 1983.

[Sre04] N. Srebro. *Learning with matrix factorizations*. PhD thesis, MIT, 2004.

[SRJ05] N. Srebro, J. Rennie, and T. Jaakkola. Maximum-margin matrix factorization. In L. K. Saul, Yair Weiss, and Léon Bottou, editors, *in Proceedings of Neural Information Processing Systems 17*, pages 1329–1336. MIT Press, Cambridge, MA, 2005.

[SSG07] P. Sarkar, S. Siddiqi, and G.J. Gordon. A latent space approach to dynamic embedding of cooccurrence data. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AI-STATS)*, 2007.

[SSGO11] S. Shalev-Shwartz, A. Gonen, and Shamir O. Large-scale convex minimization with a low-rank constraint. In *ICML*, 2011.

[The10] The data deluge. *The Economist*, February 2010.

[Tib96] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 58:267–288, 1996.

[TMP11] A. L. Traud, P. J. Mucha, and M. A. Porter. Social structure of facebook networks. arXiv:1102.2166, 2011.

[Top98] D. Topkis. *Supermodularity and complementarity*. Princeton University Press, 1998.

[Tro04] J. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. Signal Processing*, 50, 2004.

[Tro10] J. A. Tropp. User-friendly tail bounds for sums of random matrices. *ArXiv e-prints*, April 2010.

[Tsa05] R. S. Tsay. *Analysis of Financial Time Series*. Wiley-Interscience; 3rd edition, 2005.

[Tse08] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. Preprint, 2008.

[TWAK03] B. Taskar, M.F. Wong, P. Abbeel, and D. Koller. Link prediction in relational data. In *Neural Information Processing Systems*, volume 15, 2003.

[VAHS11] D.Q. Vu, A. Asuncion, D. Hunter, and P. Smyth. Continuous-time regression models for longitudinal networks. In *Advances in Neural Information Processing Systems*. MIT Press, 2011.

[vdGB09] S. A. van de Geer and P. Bühlmann. On the conditions used to prove oracle results for the Lasso. *Electron. J. Stat.*, 3:1360–1392, 2009.

[VFK04] D. Vakratsas, F. Feinberg, F.and Bass, and G. Kalyanaram. The Shape of Advertising Response Functions Revisited: A Model of Dynamic Probabilistic Thresholds. *Marketing Science*, 23(1):109–119, 2004.

[VLEM01] F. Varela, J-P. Lachaux, Rodriguez E., and J. Martinerie. The brainweb: Phase synchronization and large-scale integration. *Nature Reviews Neuroscience*, 2001.

[VM11] P. Van Mieghem. *Graph spectra for complex networks*. Cambridge University Press, 2011.

[VYH08] J. Villanueva, S. Yoo, and D. M. Hanssens. The impact of marketing-induced versus word-of-mouth customer acquisition on customer equity growth. *Journal of Marketing Research*, 45:48–59, 2008.

[WP96] S. Wasserman and Ph. Pattison. Logit models and logistic regressions for social networks: I. an introduction to markov graphs and p^*. *Psychometrika*, 61(3):401–425, September 1996.

[YCY+07] K. Yu, W. Chu, Sh. Yu, V. Tresp, and Z. Xu. Stochastic relational models for discriminative link prediction. In *Advances in Neural Information Processing Systems*, pages 333–340. MIT Press, 2007.

[YL06] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[ZEPR11] K. Zhang, Th. Evgeniou, V. Padmanabhan, and E. Richard. Content contributor management and network effects in a ugc environment. *Marketing Science*, 2011.

[ZH05] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.

[ZHT04] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, pages 1–30, 2004.

[ZLW08] S. Zhou, J. D. Lafferty, and L. A. Wasserman. Time varying undirected graphs. In *COLT*, pages 455–466, 2008.

[ZT11] T. Zhou and D. Tao. Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *International Conference on Machine Learning*, 2011.