ERM-Lasso classification algorithm for Multivariate Hawkes Processes paths

Charlotte Dion-Blanc, Christophe Denis, Laure Sansonnet, Romain Edmond Lacoste

▶ **To cite this version:**

Charlotte Dion-Blanc, Christophe Denis, Laure Sansonnet, Romain Edmond Lacoste. ERM-Lasso classification algorithm for Multivariate Hawkes Processes paths. 2024. hal-04646888

HAL Id: hal-04646888
https://hal.science/hal-04646888

Preprint submitted on 15 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ERM-Lasso classification algorithm for Multivariate Hawkes Processes paths

July 2024

Christophe Denis[1,2], Charlotte Dion-Blanc[1], Romain E. Lacoste[2], Laure Sansonnet[1,3]

(1) *LPSM, UMR 8001, Sorbonne Université*
(2) *LAMA, UMR 8050, Université Gustave Eiffel*
(3) *Université Paris-Saclay, AgroParisTech, INRAE, UMR MIA Paris-Saclay*

Abstract

We are interested in the problem of classifying Multivariate Hawkes Processes (MHP) paths coming from several classes. MHP form a versatile family of point processes that models interactions between connected individuals within a network. In this paper, the classes are discriminated by the exogenous intensity vector and the adjacency matrix, which encodes the strength of the interactions. The observed learning data consist of labeled repeated and independent paths on a fixed time interval. Besides, we consider the high-dimensional setting, meaning the dimension of the network may be large *w.r.t.* the number of observations. We consequently require a sparsity assumption on the adjacency matrix. In this context, we propose a novel methodology with an initial interaction recovery step, by class, followed by a refitting step based on a suitable classification criterion. To recover the support of the adjacency matrix, a Lasso-type estimator is proposed, for which we establish rates of convergence. Then, leveraging the estimated support, we build a classification procedure based on the minimization of a L_2-risk. Notably, rates of convergence of our classification procedure are provided. An in-depth testing phase using synthetic data supports both theoretical results.

Keywords Multivariate Hawkes Process · Classification · Empirical Risk Minimization · High Dimension · Lasso

1 Introduction

The supervised classification of complex data has drawn a lot of attention in recent years. This statistical problem covers a broad class of application including classification of multivariate times series (Ismail Fawaz et al., 2019). In particular, cutting-edge methodology for performing classification of time sequences of events is a matter of great interest. In this paper, we tackle the task of supervised classification of sequences of events into K classes with $K > 1$. We therefore consider that each class is characterized by its own underlying occurrence dynamics, and the aim is to discriminate between them. In this work, observations in each class are assumed to come from a multivariate Hawkes process (MHP), denoted N, of size $M \geq 1$, for which the probability distribution of events is given by the vector of intensity process. The shape of the vector of intensity process is assumed to be common to all classes. Therefore, the classes are discriminated

according to the parameters that describe their vector of intensity process: the baselines and the adjacency matrix that governs the relations between the components of the process. For instance, we can consider that the observations come from different groups of people observed over a given time interval whose interactions are modeled through a MHP.

In this work, the observations consist in n independent repeated observations of a mixture of the MHP observed on a fixed time interval $[0, T]$ and their associated label. In particular, we do not assume that data have reached the stationary regime. Hence, the asymptotic is in n the number of repetitions. Furthermore, we consider the high-dimensional framework, where the dimension $M := M_n$ of the MHP may be large with respect to the sample size n. In view of the high-dimensional issue, we consider sparsity assumption on the adjacency matrix of the process. We propose a classification procedure that take advantage of the estimation of the support of the adjacency matrix.

Related works. Hawkes processes (HP) are a family of point processes introduced by Hawkes (1971). Such processes model complex temporal dynamics, where the occurrence of events is impacted by past activity. The multidimensional version of these processes, MHP, is a natural generalization that considerably enriches modeling possibilities. Indeed, in addition to model self-exciting interactions, such a model takes into account positive interactions between connected individuals within a network. These interactions are encoded in the adjacency matrix. Historically applied in seismology (Ogata, 1988), they have since been used in a wide range of applications including genomics (Reynaud-Bouret and Schbath, 2010), neuroscience (Bonnet et al., 2022), finance (Embrechts et al., 2011), urban crime (Mohler et al., 2011), order book in finance (Bacry et al., 2015) and football (Baouan et al., 2022). Another important application is the modeling of social network activity, as in Zhang et al. (2018) and further works. In addition, the Hawkes processes are frequently used as spike-trains models in neurosciences, for example in Reynaud-Bouret et al. (2013); Spaziani et al. (2023); Bonnet et al. (2023). Recently, they have also been used in the ecological field in Nicvert et al. (2024) for interaction between species and in Denis et al. (2024) for bats monitoring. Recently, in the context of repeated observations with T fixed, Lotz (2024) provides a likelihood ratio test for testing presence of interaction.

In the high-dimensional setting, meaning that the number of components M is large, it is classical to impose sparsity assumptions on the adjacency matrix that characterizes the intensity process. Therefore, it appears that reconstructing the support of the adjacency matrix or the connectivity graph, is a matter of great interest, which is related to the Granger causality (see Eichler et al., 2017; Sulem et al., 2024). In particular, this is a crucial issue for connectivity of neurons, see for example Lambert et al. (2018), or in social network (Carstensen et al., 2010).

Let us focus on the Lasso literature in the classical Gaussian framework. The Lasso procedure has been originally introduced in Tibshirani (1996) and Chen et al. (1998). It is a popular statistical method for high-dimensional problems for which efficient implementation procedure have been developed. Besides, the Lasso procedure has been widely studied from the theoretical point of view (see *e.g* Meinshausen and Bühlmann, 2006; Tropp, 2006; Bühlmann and Van De Geer, 2011). In particular, support recovery results has been investigated in Wainwright (2009) as well as multi-class classification methods (see Abramovich and Grinshtein, 2018). Let us emphasize that an induced and undesired effect of ℓ_1 penalization is the shrinkage of large coefficients. To bypass this issue, refitting strategies are commonly used and well covered in the literature, see for example Chzhen et al. (2019).

For Hawkes process, the work of Donnet et al. (2020) is dedicated to a nonparametric Bayesian procedure to tackle high-dimensionality. Let us mention the work Zhou et al. (2013) which proposes an efficient algorithm for a Lasso estimator for high-dimension Hawkes process, implemented in

the Python library `tick` (Bacry et al., 2018). Later, in the work of Bacry et al. (2020), the authors use a least-squares contrast penalized with an ℓ_1-norm together with a trace norm. The resulting estimator benefits from a sparse structure with low rank. Its construction relies only on an observation over a time interval $[0, T]$, with $T \to +\infty$. In particular, theoretical results are obtained for the intensity process estimation under the asymptotic $T \to +\infty$. Under the same observational setup, several sparse support recovery procedure has also been proposed in the literature, for example a likelihood ratio testing procedure in Kim et al. (2011), or a group-Lasso least-squares penalized estimator in Cai et al. (2024).

The present work falls in the supervised classification setting. In particular, we assume that the learning sample of size n consists of i.i.d. labeled data where the features are the jump times of a multivariate Hawkes process observed on the fixed time interval $[0, T]$. In a different observational setup, some works in natural language processing also tackle some similar issues as Lukasik et al. (2016) and later Tondulkar et al. (2022). Closest to ours, the work of Denis et al. (2022) provides a classification procedure for observations coming from a univariate HP where the classes are discriminated by the kernel of the intensity process. In particular, this method is applied in Denis et al. (2024), for modeling echolocation calls of bats recording in several sites throughout France. In the present work, we generalize the approach developed in Denis et al. (2022) in the multivariate setting (MHP) under sparsity assumptions.

Main contributions. In the present paper, we propose a novel classification algorithm, the `ERMLR` algorithm that relies on a two-step procedure. A first step is dedicated to the estimation of the support of the adjacency matrix as well as the weights of the mixture. Then, in a second step, taking advantage of the estimated support, we build a classifier based on the empirical risk minimization principle. We establish rates of convergence for both support estimator and classification procedure. Furthermore, we show through a numerical study that our algorithm exhibits good numerical properties. To sum up our contributions are threefold.

- First, we provide a general device to handle high-dimensional issue for MHP. Following Bacry et al. (2020), the estimation of the parameters of the process as well as the support of the adjacency matrix is based on the minimization of a Lasso-penalized contrast. We establish rates of convergence of the estimated support and the estimated coefficients of the MHP. Notably, our theoretical findings show that the established rates of convergence are comparable to those obtained in the classical Gaussian setting. In particular, we extend the results obtained in Bacry et al. (2020) and Cai et al. (2024) in the context of repeated observations.

- Second, we provide a general classification algorithm dedicated to the supervised classification of MHP. A salient point of our procedure is that we handle high-dimensional issue by leveraging the estimation of the support of the adjacency matrix. Specifically, we consider a classifier that relies on the minimization of a L_2-risk on a set of parameters that depends only on the estimated support of the parameters. We show that the rates of convergence of our classification algorithm is, up to a logarithmic factor, of order the square root of the size of the support over the sample size n. Notably, we extend the results obtained in Denis et al. (2022) to the high-dimensional framework.

- Finally, in view of the numerical complexity of our problem, the implementation of our classification algorithm is a major challenge. The implementation of the overall procedure relies on cutting-edge optimization algorithms. Specifically, the Lasso-penalized contrast is optimized with the `FISTA` algorithm while the calibration of the l_1 penalty is performed using the `EBIC` criterion. Then, for the minimization of the L_2-risk, we consider a parameter-free projected adaptive gradient descent `Free Adagrad` recently introduced in Chzhen et al.

(2023). We evaluate the performance of our procedure on synthetic data and show the good performance of our algorithm. In particular, it reveals that our algorithm succeeds well for both support recovery and classification accuracy.

Outline of the paper. Section 2 describes the model, along with the necessary assumptions. Section 3 proposes the classification algorithm named `ERMLR`. Section 4 provides the main theoretical results on both Lasso estimator of the MHP parameters and the classification procedure. Then, full implementation details about the procedure are given in Section 5 while Section 6 is devoted to numerical results. We also provide a discussion in Section 7. Finally, the proofs are relegated in Appendix.

Notations. For a matrix $A \in \mathbb{R}^{M\times M}$, the Frobenius norm is defined as follows $\|A\|_F = (\mathrm{Tr}(A'A))^{1/2} = (\sum_{j,j'} a_{j,j'}^2)^{1/2}$, where A' denotes the transposition of the matrix A and Tr is the trace operator that returns the sum of diagonal entries of a square matrix. Recall that $\rho(A) \leq \|A\|_2 \leq \|A\|_F$ where $\|A\|_2$ is the subordinate norm and $\rho(A)$ is the spectral radius of A, which is the largest singular value of A. For an integer $L \geq 1$, the set $\{1, \dots, L\}$ is denoted by $[L]$.

2 General framework

Section 2.1 introduces the considered model, some notation, and the considered multiclass classification problem. Section 2.2 is dedicated to the presentation of the main assumptions. Finally, a closed-form expression of the optimal classifier is provided in Section 2.3.

2.1 Formal definitions and notation

Let us first introduce the general linear multivariate Hawkes process and then the considered multiclass classification model.

Multivariate counting process. Consider a M-dimensional counting process N observed on a fixed time interval $[0, T]$, with $M > 1$ the dimension of the network. More specifically, we assume that the counting process $N = (N_1(t), \dots, N_M(t))_{t\in[0,T]}$ is a linear multivariate Hawkes process, where for each $j \in [M]$, $t \in [0, T]$, $N_j(t)$ denotes the number of events that have occurred before time t at location j. The filtration (or history) at time $t \in [0, T]$ associated to the process N is denoted by $\mathcal{F}_t$. Informally, it contains the necessary information for generating the next points of N. Finally, the set of observed jump times of N over $[0, T]$ is denoted by $\mathcal{T}_T = (\mathcal{T}_{T,1}, \dots, \mathcal{T}_{T,M})$, where for each $j \in [M]$, $\mathcal{T}_{T,j}$ is the observed jump times associated to the process N_j. Each process N_j can be characterized by its intensity function. Heuristically, at a given time, the intensity function gives the infinitesimal probability of observing an event in the near future, conditionally on the past of the process. For each $j \in [M]$, the predictable intensity of the process N_j is then defined by

$$\lambda_j^*(t) = \mu_j^* + \sum_{j'=1}^{M} a_{j,j'}^* \int_0^t h(t-s)\,\mathrm{d}N_{j'}(s) = \mu_j^* + \sum_{j'=1}^{M} a_{j,j'}^* \sum_{T_\ell \in \mathcal{T}_{t,j'}} h(t - T_\ell), \tag{1}$$

where $\mu^* = (\mu_j^*)_{j\in[M]}$ is the vector of exogenous intensities, $A^* = (a_{j,j'}^*)_{1\leq j,j'\leq M}$ is the matrix of interactions, and h is the kernel function. For each $j \in [M]$, the coefficient μ_j^* models the arrival of spontaneous events for the j-th component. For each $j, j' \in [M]$, the coefficient $a_{j,j'}^*$ is non-negative and expresses the positive influence of the one-dimensional process $N_{j'}$ on the

one-dimensional process N_j. Finally, the kernel h is a non-negative function supported on $\mathbb{R}_+$ such that $\|h\|_1 = 1$. It dictates how quick these influences vanish over time. In the following, the kernel function h is assumed to be known. Finally, let us define the support of A^*, or the active set, denoted S^*. It corresponds to the positions of the non-zero coefficients $a_{j,j'}$, meaning that component j' has an impact on component j.

Remark 1. *The second equality in the definition of the intensity* (1) *is easily obtained by using that, for any function f, $j \in [M]$, and $t \in [0,T]$, the following stochastic integral is defined as the counting measure*

$$\int_0^t f(s)\, dN_j(s) = \sum_{T_\ell \in \mathcal{T}_{t,j}} f(T_\ell).$$

Multiclass setting. We consider the multiclass classification problem, where each data point is characterized by a couple $(\mathcal{T}_T, Y)$, where $\mathcal{T}_T$ is the set of observed jump times of a counting process N over $[0,T]$ and $Y \in [K]$ is its label. In particular, we assume that $N = (N_1, \dots, N_M)$ is a mixture of a M-dimensional linear HP observed on the time interval $[0,T]$. More precisely, conditional on Y, the counting process N is a M-dimensional linear HP, where for each $j \in [M]$, the predictable intensity of N_j depends on the label Y and is defined at time $t \geq 0$ as follows

$$\lambda^*_{Y,j}(t) = \mu^*_{Y,j} + \sum_{j'=1}^{M} a^*_{Y,j,j'} \int_0^t h(t-s)\, \mathrm{d}N_{j'}(s). \tag{2}$$

The vector $\mu^*_Y = (\mu^*_{Y,j})_{j\in[M]}$ is the vector of baselines associated to the class Y, and the matrix $A^*_Y = (a^*_{Y,j,j'})_{1\leq j,j'\leq M}$ is the $M \times M$ adjacency matrix of the network associated to the label Y. This choice of modeling is motivated by the fact that the classes are characterized by different underlying network behavior, where an edge in the network matches a non-zero $a_{j,j'}$.

We assume in the following that the parameters (μ^*_Y, A^*_Y) are unknown as well as the distribution of Y which is denoted by $p^* = (p^*_k)_{k\in[K]}$. Finally, the kernel function h is assumed to be known and for the sake of simplicity, it does not depend on the classes or on the components of the process. Note that in the numerical section, we consider the standard choice of exponential kernel. However, more general choice of the kernel function may be investigated. For instance, in Bacry et al. (2020) the authors consider the case where h is a sum of exponential functions, which preserves the markovianity of the intensity process.

Objective. In the multiclass setup, the objective is to build a classifier, a measurable function g such that $g(\mathcal{T}_T)$ belongs to $[K]$, and provides an accurate prediction of the label Y. In particular, the misclassification risk assesses the quality of such predictor g. It is defined as

$$\mathcal{R}(g) := \mathbb{P}\left(g(\mathcal{T}_T) \neq Y\right).$$

The set of all classifiers is denoted by $\mathcal{G}$. Naturally, we aim at considering the predictor g^*, namely the Bayes classifier, that achieves the minimum risk over $\mathcal{G}$. In Section 2.3, we provide an explicit formula of the oracle classifier g^*. Nevertheless, since the distribution of the observation $(\mathcal{T}_T, Y)$ is assumed to be unknown, we build a predictor that relies on a training sample of size n which consists of i.i.d. copies of $(\mathcal{T}_T, Y)$. At this step, we draw the reader attention to the fact that the considered asymptotic is as follows. The horizon time T is fixed, while the sample size n goes to infinity. Recall that the size M of the MHP is actually $M = M_n$ and can increase with n. In

the sequel, a predictor built on the training data is denoted by $\hat{g}$. In particular, we require that $\hat{g}$ satisfies the consistency property,

$$\mathbb{E}\left[\mathcal{R}(\hat{g}) - \mathcal{R}(g^*)\right] \to 0,$$

when n tends to infinity. However, in our study, the intrinsic dimension of our problem M^2 may be much larger than the sample size of the learning dataset. In this case, predictor $\hat{g}$ is not consistent. Therefore, as it is usual in this high-dimensional setup, we introduce a sparsity assumption for our model.

2.2 Assumptions

This section is dedicated to the main assumptions that are assumed throughout the paper. In particular, in our multivariate framework, we allow the dimension parameter M to be large, which may induces that M^2 is much larger than the size of the training sample. To alleviate this issue, we introduce a sparsity assumption on the matrices $(A_k^*)_{k\in[K]}$.

Firstly, we introduce an assumption which ensures that each class occurs with non-zero probability.

Assumption 1. *There exists $p_0 > 0$ such that $\min_{k\in[K]}(p_k^*) > p_0$.*

We also assume that the parameters of the process belongs to a compact set.

Assumption 2. *(Compactness)*

(i) There exists $0 < \mu_0 < \mu_1$, s.t. for $i \in [M]$ and $k \in [K]$, $\mu_0 \leq \mu_{k,i}^* \leq \mu_1$.

(ii) There exists $C_A > 0$, s.t. $\max_{k\in[K]} \|A_k^*\|_F < C_A$.

Furthermore, we consider the following assumptions, which imply that the process N admits finite exponential moment.

Assumption 3. *(Stability condition)*

(i) The kernel function h belongs to the set $\mathcal{H} := \{h : \mathbb{R}_+ \to \mathbb{R}_+, \int h(t)\, dt = 1\}$ and is bounded.

(ii) $\max_{k\in[K]} \rho(A_k^*) < 1$.

Let us notice here that if $C_A < 1$ it implies that $\rho(A) < 1$.

Assumption 4. *(Exponential moment) There exist $a > 0$, and $C > 0$ that do not depend on M, such that*

$$\sup_{j\in[M]} \mathbb{E}[\exp(aN_j(T))] < C.$$

Remark 2. *Note that Leblanc (2024) proves that the exponential moment of the multivariate Hawkes process is finite, under Assumption 3 and when the intensity process is stationary. Nevertheless, in the general case, the bound of the moment depends on M. Hence, we require a stronger condition, such as Assumption 4 which is more suitable in the high-dimensional framework. For instance, this assumption is satisfied if there exists a positive constant C, that does depend on M, such that $\sum_{j\in[M]} \mu_j \leq C$.*

Finally, we assume that for each $k \in [K]$, the adjacency matrix A_k^* is sparse, meaning that a few coefficients are non-zero. For each $k \in [K]$, let us denote by

$$S_k^* := \left\{(j, j') \in [M]^2, \;\; a_{k,j,j'}^* \neq 0\right\}$$

the active set (or support) of A_k^*, $|S_k^*|$ its cardinality, and S_k^{*c} its complement. Throughout the paper, we consider the following assumption.

Assumption 5. *(Sparsity assumption) There exists a constant $s^* > 0$ such that*

$$\max_{k \in [K]} |S_k^*| \leq s^*.$$

In particular, in our high-dimensional setting, we assume that $s^* << M^2$. Since we do not assume sparsity on the vectors $\mu_k, \; k \in [K]$, we consider the following interplay between parameter M and the sample size of the training dataset. The dimension of the process M may depend on n with $M^2 >> n$. In this case, the sparsity assumption is crucial to overcome the high-dimension issues. However, we assume that M satisfies $M/n \to 0$.

Remark 3. *Note that we only assume sparsity on the adjacency matrix, but not on the vector μ^*. It ensures that all the components of the process are active. However, as in Bacry et al. (2020), it may be possible to consider also sparsity assumption on the vector of exogenous intensities. Nevertheless, this is not the line taken in this work.*

2.3 Bayes rule

In this section, we exhibit a closed-form expression of the Bayes classifier g^* that minimizes the misclassification risk over the set $\mathcal{G}$. The Bayes classifier is characterized by,

$$g^*\left(\mathcal{T}_T\right) \in \underset{k \in \mathcal{Y}}{\operatorname{argmax}} \, \pi_k^*(\mathcal{T}_T),$$

with $\pi_k^*\left(\mathcal{T}_T\right) = \mathbb{P}\left(Y = k | \mathcal{T}_T\right)$. The following result is an extension of the result given in Denis et al. (2022). It gives the expression of the conditional probabilities π_k^* and then provides a closed form of the Bayes classifier.

Proposition 1. *Let $T \geq 0$. For each $k \in [K]$, we define,*

$$F_k^*(\mathcal{T}_T) := -\sum_{j=1}^{M} \int_0^T \lambda_{k,j}^*(s) ds + \sum_{j=1}^{M} \sum_{T_\ell \in \mathcal{T}_{T,j}} \log\left(\lambda_{k,j}^*(T_\ell)\right). \tag{3}$$

Therefore, the sequence of conditional probabilities satisfies

$$\pi_k^*\left(\mathcal{T}_T\right) = \frac{p_k^* \mathrm{e}^{F_k^*(\mathcal{T}_T)}}{\sum_{k'=1}^{K} p_{k'}^* \mathrm{e}^{F_{k'}^*(\mathcal{T}_T)}} \quad \mathbb{P} - a.s.,$$

where $F^* = (F_1^*, \ldots, F_K^*)$.

Proposition 1 exhibits an explicit link between the unknown parameters $(\mu_k^*, A_k^*)_{k \in [K]}$ and the Bayes classifier. In particular, it suggests that a classification rule can be easily obtained by replacing the unknown parameters by estimators in Equation (3). However, the performance of the resulting classifier strongly depends on the quality of the considered estimators. In the present framework, without taking account Assumption 5, the high-dimension of the problem could lead to bad estimates. To overcome this difficulty, we propose a classification algorithm tailored to our setting, which involves Lasso-type estimators.

3 Classification algorithm

In this section, we present the proposed classification algorithm that relies on a *refitting* strategy (see *e.g.* Chzhen et al., 2019). The algorithm is referred as `ERMLR` for *Empirical Risk Minimizer with Lasso Refitting*. Since the construction of the prediction rule goes in several steps and involves a splitting of the training dataset, for the sake of the simplicity, we consider a dataset of size $2n$. More specifically, the learning dataset is denoted denoted $\mathcal{D}_n = \{(\mathcal{T}_T^{(i)}, Y^{(i)}), i = 1, \dots, 2n\}$, which consists of $2n$ independent copies of $(\mathcal{T}_T, Y)$. For the estimation purpose, the data set $\mathcal{D}_n$ is divided into two independent data sets $\mathcal{D}_n^{(1)}$ and $\mathcal{D}_n^{(2)}$ of same size n. For sake of simplicity in the following, we index both sample using $\{1, \dots, n\}$.

To take advantage of Assumption 5, we then consider the following three-stages procedure.

- Based on the first data set $\mathcal{D}_n^{(1)}$, we estimate the distribution $p^* = (p_k^*)_{k\in[K]}$ by its empirical counterpart $\widehat{p}$.
- Based on the second dataset $\mathcal{D}_n^{(2)}$, and for each $k \in [K]$, we estimate by $\widehat{S}_k$ the active set S_k^* with a Lasso-type criterion, described in Section 3.1.
- Based on the second dataset $\mathcal{D}_n^{(2)}$, then, we build a classifier $\widehat{g}$ that minimizes an empirical L_2-risk on a set of predictors that depends on the estimated support $(\widehat{S}_k)_{k\in[K]}$. This construction is detailed in Section 3.2.

3.1 Estimation of the active sets

Our classification procedure relies on the estimation of the active sets S_k^* for all $k \in [K]$. To this aim, we consider the least squares contrast with a Lasso penalty for repeated observations. The considered contrast is an adaptation of the penalized criteria proposed in Bacry et al. (2020) in the context of repeated observations with a fixed horizon time of observation T.

Let $k \in [K]$. Hereafter, we define the estimator of the active set S_k^*. We denote $\left(\mathcal{T}_T^{(1)}, \dots, \mathcal{T}_T^{(n_k)}\right)$ the observations from class k coming from $\mathcal{D}_n^{(2)}$, with $n_k = \sum_{i=1}^n \mathbb{1}_{\{Y^{(i)}=k\}}$ the random number of observations from class k. First, we define the considered contrast. To this end, we introduce the generic parameter $\theta = (\theta_1, \dots, \theta_M)' \in \mathbb{R}^{M(M+1)}$, such that for each $j \in [M]$, $\theta_j = (\theta_{j,\ell})_{0\le\ell\le M}$ writes as

$$\theta_j := (\mu_j, a_{j,1}, \dots, a_{j,M}).$$

The vector of true parameters is also denoted by $\theta_k^* = \left(\theta_{k,1}^*, \dots, \theta_{k,M}^*\right)' \in \mathbb{R}^{M(M+1)}$. For each $j \in [M]$, it expresses as follows

$$\theta_{k,j}^* = \left(\theta_{k,j,\ell}^*\right)_{l\in\{0,\dots,M\}} := \left(\mu_{k,j}^*, a_{k,j,1}^*, \dots, a_{k,j,M}^*\right)' \in \mathbb{R}^{M+1}. \tag{4}$$

Then, for each $\theta \in \mathbb{R}^{M(M+1)}$, and $i \in [n_k]$, we define the corresponding intensity function associated to the observation $\mathcal{T}_T^{(i)}$ that stems from class k for $t \in [0, T]$ as

$$\lambda_{k,j,\theta}^{(i)}(t) = \mu_{k,j} + \sum_{j'=1}^{M} a_{k,j,j'} \sum_{T_\ell^{(i)} \in \mathcal{T}_{t,j'}^{(i)}} h(t - T_\ell^{(i)}).$$

The considered penalized contrast is defined, for each $\theta \in \mathbb{R}^{M(M+1)}$, as follows,

$$R_{T,n_k}(\theta) := \frac{\mathbb{1}_{(n_k\ge1)}}{n_k} \sum_{i=1}^{n_k} \left(\frac{1}{T} \sum_{j=1}^{M} \int_0^T \lambda_{k,j,\theta}^{(i)2}(t)dt - \frac{2}{T} \sum_{j=1}^{M} \sum_{T_\ell^{(i)} \in \mathcal{T}_{T,j}^{(i)}} \lambda_{k,j,\theta}^{(i)}\left(T_\ell^{(i)}\right) \right). \tag{5}$$

Note that if $n_k = 0$, we have $\widehat{\theta} = 0$. The Lasso estimator is then defined as

$$\widehat{\theta}_k \in \underset{\theta \in \mathbb{R}^{M(M+1)}}{\operatorname{argmin}} \left\{ R_{T,n_k}(\theta) + \kappa \sum_{j=1}^{M} \sum_{j'=1}^{M} |\theta_{j,j'}| \right\}. \tag{6}$$

Finally, from the estimator $\widehat{\theta}_k$, we get the estimated support of A_k^*

$$\widehat{S}_k = \{(j, j^{'}) \in [M]^2, \ \widehat{\theta}_{k,j,j'} \neq 0\}.$$

Note that $\widehat{S}_k$ represents the estimated active set of A_k^* since it does not involve the first column of $\widehat{\theta}_k$ that contains the vector of estimated baseline $(\mu_j)_{j\in[M]}$.

3.2 ERM classifier with refitting step

In this section, we present the last step of our estimation procedure, which is dedicated to the construction of the final classifier. In particular, it involves the estimation of parameter $\theta^* = (\theta_k^*)_{k\in[K]}$. We highlight that this step relies on the estimated support $\widehat{S}_k$. To this end, we introduce the constraint set of parameters

$$\Theta_n := \left\{ \theta = (\mu, A) \in \mathbb{R}_+^M \times \mathbb{R}_+^{M^2}, \ \mu_j \in \left[\frac{1}{n}, \log(n) \right], \ j \in [M], \ \|A\|_F \leq \log(n) \right\},$$

and finally the set of interest

$$\widehat{\Theta} := \left\{ \theta = (\theta_1, \ldots, \theta_K) \in \Theta_n^K, \ \operatorname{supp}(A_k) = \widehat{S}_k \right\}. \tag{7}$$

Several comments can be made from the definition of the set of parameters $\widehat{\Theta}$. First we observe that conditional on the event $\{\hat{S}_k = S_k^*\}$, for n large enough, the true parameter θ^* belongs to the set $\widehat{\Theta}$. Indeed, in view of Assumption 2, for n large enough, we may assume that $1/n < \mu_0 < \mu_1 < \log(n)$, and $C_A \leq n$. Furthermore, we emphasize that the choice of the bounds on the coefficients on the parameters of $\widehat{\Theta}$ allows to get rid of the unknown constants defined in Assumption 2. These choices are also driven by technical aspects. In particular, such bounds are required to apply concentration arguments. Additionally, let us mention that contrary to the previous step, the optimization is performed on $\mathbb{R}_+$ for each coefficient.

Let us present the estimation of the parameter θ^* and then the construction of the resulting classifier $\widehat{g}$. This construction follows the strategy provided in Denis et al. (2022) for $M = 1$, and is based on the dataset $\mathcal{D}_n^{(2)}$. It relies on the empirical risk minimization principle. Specifically, for each $\theta \in \widehat{\Theta}$, we introduce an associated score functions $f_\theta = (f_\theta^1, \ldots, f_\theta^K)$ such that for an observed sequence of events $\mathcal{T}_T$

$$f_\theta(\mathcal{T}_T) = 2\pi_{k,\widehat{p},\theta}(\mathcal{T}_T) - 1, \ k \in [K],$$

with

$$\pi_{k,\widehat{p},\theta}(\mathcal{T}_T) = \frac{\widehat{p}_k \mathrm{e}^{F_k(\mathcal{T}_T)}}{\sum_{k'=1}^K \widehat{p}_{k'} \mathrm{e}^{F_{k'}(\mathcal{T}_T)}}$$

and

$$F_{k,\theta}(\mathcal{T}_T) = -\sum_{j=1}^{M} \int_0^T \lambda_{k,j,\theta}(s) ds + \sum_{j=1}^{M} \sum_{T_\ell \in \mathcal{T}_{T,j}} \log\left(\lambda_{k,j,\theta}(T_\ell)\right).$$

Note that the form of the score function f_θ is chosen according to the result provided in Proposition 1. Let $\theta \in \widehat{\Theta}$, and f_θ its associated score function, we define its empirical L_2-risk as

$$\widehat{\mathcal{R}}_2(f_\theta) := \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\left(Z_k^{(i)} - f_\theta^k(\mathcal{T}_T^{(i)})\right)^2, \quad Z_k^{(i)} = 2\mathbb{1}_{\{Y_i=k\}} - 1.$$

Then, we define the estimator of θ^* as the minimizer of the empirical L_2-risk,

$$\widehat{\theta}^{\mathrm{R}} \in \underset{\theta\in\widehat{\Theta}}{\operatorname{argmin}}\, \widehat{\mathcal{R}}_2(f_\theta). \tag{8}$$

From the estimator $\widehat{\theta}^{\mathrm{R}}$ of parameter θ^*, we define the `ERMLR` classifier as follows

$$\widehat{g}(\mathcal{T}_T) \in \underset{k\in\mathcal{Y}}{\operatorname{argmax}}\, \pi_{k,\widehat{p},\widehat{\theta}^{\mathrm{R}}}(\mathcal{T}_T) \tag{9}$$

Note that for computational purpose, as it is usual in classification, the $0-1$ loss is then replaced by the L_2 convex surrogate (see *e.g* Zhang, 2004). In particular, the L_2-loss is classification calibrated and Zhang's lemma Zhang (2004) ensures that

$$\mathbb{E}\left[\mathcal{R}(\widehat{g}) - \mathcal{R}(g^*)\right] \leq \frac{1}{\sqrt{2}}\big(\mathbb{E}\left[\mathcal{R}_2(f_{\widehat{\theta}^{\mathrm{R}}}) - \mathcal{R}_2(f_{\theta^*})\right]\big)^{1/2},$$

with $\mathcal{R}_2$ the oracle counterpart of the considered empirical risk $\widehat{\mathcal{R}}_2$ defined as

$$\mathcal{R}_2(f_\theta) = \mathbb{E}\left[(Z_k - f_\theta(\mathcal{T}_T))^2\right], \text{ with } Z_k = 2\mathbb{1}_{\{Y=k\}} - 1.$$

One of the main appealing property of our classification algorithm is that we take advantage of the estimated support to perform the minimization of the empirical L_2-risk on a set of parameter whose dimension is much smaller than M^2. Besides, rather than using the estimated parameters obtained at the first step (Lasso-step), we consider the estimator of parameter θ^* as the minimizer of loss adapted to our multiclass classification setting.

4 Theoretical results

In this section, we first provide the consistency of the estimator of the active set in Section 4.1. Then, in Section 4.2, we derive the rate of convergence of our classification procedure with respect to the misclassification risk.

4.1 Support recovery for classification

In this section, we present the key result of the Lasso procedure. More precisely, we show that

$$\mathbb{P}\left(\hat{S}_k = S_k^*\right) \to 1, \quad n \to \infty,$$

which implies that for each $k \in [K]$, the Lasso estimator $\hat{\theta}_k$ solution of Equation (6) has nonzero entries at the same positions as the true parameter θ_k^*. In particular, for the multivariate Hawkes process, for $j, j' \in [M]^2$, the Lasso step can be interpreted as interaction selection, where the objective is to select whether a component j is impacted by a component j'.

Before, to give our main result, we introduce some notations for the Lagrangian version of the Lasso criterion given in Equation (6).

Notations. In the rest of the section, we fix a class $k \in [K]$, and for simplicity we drop the dependency on k. Besides, throughout this section, we work conditional on the event $n_k \geq 1$. We also remind the reader that n_k is the random number of observations from class k in the dataset $\mathcal{D}_n^{(1)}$ of size n. Then, we define for each $t \in (0, T]$ the random matrix $\mathbb{H}_t \in \mathbb{R}^{n_k \times (M+1)}$ as follows

$$(\mathbb{H}_t)_{i,j} = H_j^{(i)}(t), \text{ with } H_j^{(i)}(t) := \int_0^t h(t-s)\,\mathrm{d}N_j^{(i)}(s),\ j \neq 0,\ H_0^{(i)} \equiv 1. \tag{10}$$

From the definition of the matrix $\mathbb{H}_t$, we observe that

$$\lambda_{j,\theta}^{(i)}(t) = \sum_{j'=0}^{M} H_{j'}^{(i)}(t)\theta_{j,j'},$$

in other words,

$$\mathbb{H}_t\theta_j = \left(\lambda_{j,\theta}^{(i)}(t)\right)_{i=1,\dots,n_k}.$$

For $j \in [M]$, and $i \in \{1, \dots, n_k\}$, we consider $M_j^{(i)}$ the martingale associated to the counting process $N_j^{(i)}$ through the Doob-Meyer decomposition. We then denote $\mathrm{d}M(t) = \left(\mathrm{d}M_j^{(i)}(t)\right)_{j,i} \in \mathbb{R}^{M \times n_k}$, and define the random martingale matrix Z as

$$Z := \int_0^T (\,\mathrm{d}M(t)\mathbb{H}_t)'.$$

Besides, the j-th column of Z is denoted by Z_j. Therefore, for $j, j^{'} \in [M] \times \{0, \dots, M\}$, the main term of Z is the continuous-time martingale,

$$Z_{j,j'} = \sum_{i=1}^{n_k} \int_0^T H_{j'}^{(i)}(t)\,\mathrm{d}M_j^{(i)}(t). \tag{11}$$

We finally define the random matrix $\mathbb{H}$ of size $(M+1) \times (M+1)$ as

$$\mathbb{H} := \frac{1}{T}\int_0^T \mathbb{H}_t'\mathbb{H}_t\,\mathrm{d}t.$$

In the following, for $S \subset [M]$, we denote $\mathbb{H}_{S,S}$ the matrix where the lines and the columns are restricted to the set S.

Assumptions. Classical conditions in the ℓ_1 constraint framework are considered as, for instance, in Bühlmann and Van De Geer (2011) and references therein. According to Equation (4), the true parameter is denoted $\theta_j^* = \left(\theta_{j,\ell}^*\right)_{\ell \in \{0,\dots,M\}}$, and $\theta_j^* = \left(\mu_j^*, a_{j,1}^*, \dots, a_{j,M}^*\right)' \in \mathbb{R}^{M+1}$. For each $j \in [M]$, we also denote $S_{\theta_j}^*$ the active set of θ_j^*. Note that, since $\mu_j^* > 0$, it contains at least one element.

The first assumption is the mutual incoherence, which is also referred as irrepresentability condition. Heuristically, this imposes that the correlation between the non-active and active variables must not be higher than the variations within the actives variables, otherwise the lasso estimator would not be able to dissociate them. It involves an incoherence parameter $\gamma \in (0, 1]$ that must not be too small.

Assumption 6 (Mutual incoherence (MI)). *There exists some* $0 < \gamma \leq 1$ *such that, a.s.*

$$\max_{j=1,\ldots,M} \|\mathbb{H}_{S^{*c}_{\theta_j}, S^*_{\theta_j}} \mathbb{H}^{-1}_{S^*_{\theta_j}, S^*_{\theta_j}}\|_\infty \leq 1 - \gamma.$$

The following condition ensures that the submatrix $\mathbb{H}_{S^*_{\theta_j}, S^*_{\theta_j}}$ does not have its columns linearly dependent (in which case it could be impossible to estimate θ^* when the true active set is known). The notation $\Lambda_{\min}$ denotes the minimal eigenvalue.

Assumption 7 (Minimum eigenvalue (ME)). *There exists* $\Lambda_0 > 0$ *such that, a.s.,*

$$\min_{j=1,\ldots,M} \Lambda_{\min}\left(\frac{\mathbb{H}_{S^*_{\theta_j}, S^*_{\theta_j}}}{n_k}\right) \geq \Lambda_0.$$

Finally, the last condition of minimum signal ensures that the non-zero entries of the true coefficients are large enough to be properly estimated. Specifically, it imposes that the minimum value of the true parameter restricted to the support S^* cannot decay to zero faster than the regularization parameter, κ which is specified in Theorem 1.

Assumption 8 (Minimum signal condition (MS)).

$$\min_{j,j' \in S^*} \left|\theta^*_{j,j'}\right| > \Lambda_0 \max_{j=1,\ldots,M} \sqrt{\left|S^*_{\theta_j}\right|} \frac{\log^4(nM^2)}{\sqrt{n}}.$$

Support recovery result. The result provided by Theorem 1 is the main ingredient to derive rate of convergence of our classification procedure. Nevertheless, it is an interesting result *per se*. Under the above assumptions, for each class $k \in [K]$, we establish the uniqueness of the Lasso solution, the consistency of the estimated support, and the uniform consistency of the estimator of θ^*.

Theorem 1. *Assume that* $n > \frac{2}{p_0}$, *and let* $\kappa = \dfrac{\log^4(nM^2)}{\sqrt{n}}$. *Grant Assumptions (MI), (ME), and (MS). There exists an event* Ω_n *with* $\mathbb{P}(\Omega_n) \geq 1 - \dfrac{C}{n}$, *on which* $n_k \geq 1$, *and*

$$\min_{\theta \in \mathbb{R}^{M(M+1)}} \left\{ R_{T,n_k}(\theta) + \kappa \sum_{j=1}^{M} \sum_{j'=1}^{M} |\theta_{j,j'}| \right\},$$

where R_{T,n_k} *is given in Equation* (5), *admits a unique solution* $\widehat{\theta}$ *which satisfies the following*

(i) $\operatorname{supp}(\widehat{\theta}) = \operatorname{supp}(\theta^*)$;

(ii)

$$\left\|\widehat{\theta} - \theta^*\right\|_\infty \leq \frac{\Lambda_0 \max_{j=1,\ldots,M} \sqrt{|S^*_{\theta_j}|} \log^4(nM^2)}{\sqrt{n}}.$$

Several comments can be made from the above result. First, a straightforward consequence of Theorem 1, is that for each $k \in [K]$,

$$\mathbb{P}\left(\hat{S}_k = S^*_k\right) \geq 1 - \frac{C}{n}.$$

Hence, our result provides rate of convergence for the estimator of the support $\hat{S}_k$. Furthermore, in view of Assumption 8, we have that on the event Ω_n, $\hat{\theta}_{j,j'} > 0$. Notably, Theorem 1 extends the result of Bacry et al. (2020) in the context of repeated observations with fix observation time. In particular, the work of Bacry et al. (2020) does not provide support recovery result. However, we emphasize that our result requires stronger assumption than in Bacry et al. (2020). Let us notice that the result holds also for κ larger that $\log^4(nM^2)/\sqrt{n}$ but in this case the rates of convergence is slower.

Second, up to logarithmic factor, the condition on the tuning parameter $\kappa = \kappa_n$ is of the same order as in Wainwright (2009). Besides, up to a logarithmic factor, we obtain a rate of convergence of order $\max_{j=1,\dots,M} |\sqrt{S^*_{\theta_j}}|/\sqrt{n}$ in sup-norm for the estimator $\hat{\theta}$, we can note that this rate is of the same order than the one that would expect in the classical Gaussian framework Bühlmann and Van De Geer (2011). We also highlight that in the logarithmic factor, the power of the log term is in part due to the fact that the number of jump-times of the process is not bounded a.s.

Finally, the proof of this result is based on a preliminary lemma, which gives a control in probability of the maximum of the martingale terms $Z_{j,j'}$ defined in Equation (11). This inequality is obtained using a Bernstein type inequality proven in Bacry et al. (2020). This data-driven inequality and the sub-exponential property of the counting process (see Assumption 4) lead to the concentration result. Then, we follow the primal-dual-witness method of proof (see for instance Tibshirani and Wasserman, 2017).

4.2 Rate of convergence of the `ERMLR` classifier

In this section, we derive theoretical property of the `ERMLR` algorithm $\hat{g}$. To establish our result, we take advantage of the support recovery result provided in Section 4.1. On the set $\{\hat{S}_k = S^*_k\}$, the excess risk of $\hat{g}$ is upper-bounded by applying classical arguments derived from the classification framework. While we use Theorem 1 to bound the excess risk on the event $\{\hat{S}_k \neq S^*_k\}$. Then, we obtain the following result.

Theorem 2. *Grant Assumptions 1, 3 and 2. For n large enough, there exists a constant $C > 0$ such that,*

$$\mathbb{E}\left[\mathcal{R}(\widehat{g}) - \mathcal{R}(g^*)\right] \leq C\left(\frac{(M + s^*)\log(nM)}{n}\right)^{1/2},$$

where C depends on $T, K, \|h\|_\infty, \mu_0, \mu_1, p_0$.

As expected, we highlight that, thanks to the Lasso step, we manage to obtain, up to a logarithmic factor, a rate of order $\sqrt{(M + s^*)/n}$ rather than $\sqrt{M + M^2)/n}$. Notably, we show that the proposed algorithm achieves the usual parametric rate.

5 Implementation

In this section, a comprehensive description of the implementation details is specified. As the `ERMLR` procedure execution involves two minimization problems, these two steps are described separately in Section 5.1 and Section 5.2. In both cases, each choice is discussed in terms of the state of the art and its relevance in the context of its use. Besides, let us highlight that the implementation of the procedure relies on state-of-the art algorithms and `C++` codes wrapped in `Python` which serves the purpose of rapid computation.

5.1 Implementation details for the Lasso step

For the support recovery step, our strategy consists in the minimization of the least squares contrast with Lasso penalty defined in Equation (6). This objective function is written as the sum of two functions. While the least squares contrast is differentiable, convex and smooth (*i.e.* with a Lipschitz continuous gradient), the ℓ_1-norm is non-differentiable at zero. To this extent, to carry out the minimization of such objective function, we use first-order optimization algorithm based on proximal methods with Nesterov's momentum method, namely `FISTA`, see Beck and Teboulle (2009). Compared to the classical proximal algorithm, the construction of a new iterate of the descent is based on a specific linear combination of the previous two points. This makes `FISTA` benefits from a significantly faster rate of convergence. A recommended choice of the descent step is $1/L$ with L the Lipschitz constant of the gradient. We stop the descent after 200 iterations if the stopping criterion, based on relative distance between two successive iterations, is not fulfilled yet.

Another important aspect of the Lasso step concerns the calibration of the penalization constant κ which controls the regularization. As our goal is to recover the true support, κ must be large enough to set all non-active coefficients to zero. To this end, our strategy is the following: different values of κ are explored through a grid of sufficiently fine size, denoted Δ, and the one that minimizes a specific model selection criterion is chosen. The criterion used here is the Extended Bayesian Information Criteria (`EBIC`) introduced by Chen and Chen (2008). For some $\gamma \in [0, 1]$ and $\kappa \in \Delta$, this criterion takes the following form:

$$\texttt{EBIC}_\gamma(\kappa) := -2L_{T,n}\left(\widehat{\theta}(\kappa)\right) + \left|S_{\widehat{\theta}(\kappa)}\right| \log(n) + 2\gamma \log\left(\binom{M^2}{\left|S_{\widehat{\theta}(\kappa)}\right|}\right)$$

where $\widehat{\theta}(\kappa)$ is the Lasso estimated with the tuning parameter κ, $L_{T,n}$ is the log-likelihood of the model, $\left|S_{\widehat{\theta}(\kappa)}\right|$ is the size of this support, namely the number of active coefficients of $\widehat{\theta}(\kappa)$.

Compared to a classical `BIC` criteria (namely $\gamma = 0$), an additional penalization is added to take into account the number of possible active sets of the same size. As this quantity is also increasing with this size, it seems to be very relevant in a high-dimensional setting. In the following, we choose $\gamma = 1$ and $|\Delta| = 40$ as exploration grid size.

Finally, let us highlight that for both the least squares contrast and the log-likelihood functional, computation such as gradient or loss evaluation are optimized and implemented in `C++` which serves the purpose of rapid computation.

5.2 Implementation details for the ERM step

For the classification step, our strategy consists in minimizing the convexified empirical risk defined in Equation (9). According to the definition of the constraint set of parameters defined in Equation (7), each coefficient must be positive. To ensure that each coefficient $a_{k,j,j'}$ remains in $[0, c]$, with $c > 0$, the minimization is done under inequality constraints and we use a projected gradient descent algorithm. Nevertheless, since this objective function is non-smooth and non-convex *w.r.t.* to the coefficients, its minimization requires particular care. In particular, the tuning of the step-size in the descent is very tricky [1]. On the other hand, adaptive gradient methods, such as `AdaGrad` (see Duchi et al., 2011), have been widely used in large-scale optimization due to their ability to adjust the step size for each feature according to the geometry of the problem. In practice, `AdaGrad`

[1] Furthermore, classical method such as backtracking line-search with Armijo-Wolfe condition cannot be used due to the piece-wise constant nature of the projection operator (see Michael W. Ferry and Zhang (2023)).

is known to be an efficient method in non-convex setting (in particular for training deep neural networks optimization, see Gupta et al. (2014)). In addition, some theoretical guarantees for the convergence of `AdaGrad` for non-convex functions have been provided in the literature (see Ward et al., 2020; Wang et al., 2023). With this in mind, we use a parameter-free projected adaptive gradient descent method, in the inspiration of `AdaGrad`, called `Free AdaGrad` and introduced in Chzhen et al. (2023). Compared with the classical algorithm, its main advantage lies in the fact that it is adaptive to the distance between the initialization and the optimum, and to the sum of the square norm of the gradients. The initial starting point is chosen as the estimate given by the Lasso step, the initial guess for the distance between the starting point and the optimum is taken as $\gamma_0 = 0.1$ and we stop the descent after 1000 iterations if the stopping criteria described before is not fulfilled yet.

6 Numerical results

The goal of this section is to investigate the performance of our method from a numerical standpoint using synthetic data. First, in Section 6.1, alternative strategies are proposed for comparison purpose. Then, the simulation and evaluation scheme is thoroughly detailed in Section 6.2 and in Section 6.3. Finally, the obtained results, for support recovery by the Lasso step in Section 6.4 and the classification procedure performance in Section 6.5 are presented.

6.1 Benchmark

Let us detail here the different competitors which are compared with our classifier.

Simple plug-in strategy. A full plug-in strategy consists in use the estimators $\widehat{\theta}$ of the parameters, obtained by minimizing the least-squares contrast with Lasso penalty on the adjacency matrix given in Equation (6). Then, we plug $\hat{\theta}$ into the Bayes classifier formula. Consequently, the resulting classifier for a new observation $\mathcal{T}_T$ is

$$\widehat{g}_{\widehat{p},\widehat{\theta}}(\mathcal{T}_T) = \underset{k\in\mathcal{Y}}{\operatorname{argmax}} \frac{\widehat{p}_k e^{F_{\widehat{\mu}_k,\widehat{A}_k}(\mathcal{T}_T)}}{\sum_{k'=1}^{K} \widehat{p}_{k'} e^{F_{\widehat{\mu}_{k'},\widehat{A}_{k'}}(\mathcal{T}_T)}},$$

where $\widehat{p}$ is the estimated distribution of Y. This classifier is learned on the entire training sample $\mathcal{D}_n$ of size $2n$. This classifier is referred as `PI`.

Oracle on estimated support. We are also interested in another predictor, referred to as `OES` for *The Oracle on Estimated Support*, which is defined as follows

$$\widehat{g}_{\widehat{p},\theta^*_{\widehat{S}}}(\mathcal{T}_T) \in \underset{k\in\mathcal{Y}}{\operatorname{argmax}}\, \pi_{k,\widehat{p},\theta^*_{\widehat{S}}}(\mathcal{T}_T).$$

where

$$(\theta^*_{\widehat{S}})_{k,j,j'} := \begin{cases} \theta^*_{k,j,j'} & \text{if } (j,j') \in \widehat{S}_k \\ 0 & \text{otherwise} \end{cases}.$$

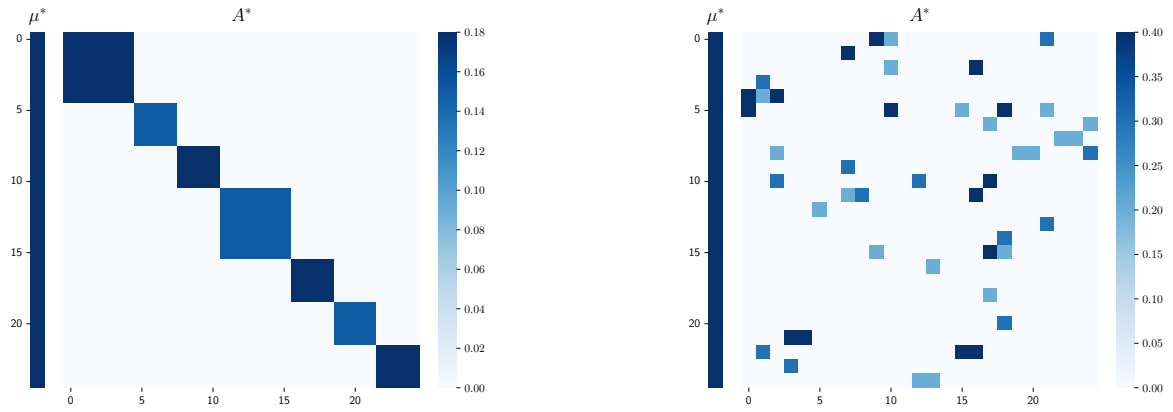
It corresponds to the best possible predictor that relies on the support recovered in the Lasso step. Note that if the true support is recovered by the Lasso step, then it exactly corresponds to the Bayes rule. By taking into account this predictor, we can quantify the effect of poor support recovery in terms of classification error, while evaluating the gain that could be obtained by an `ERM` step.

6.2 Simulation scheme

In this section, we give some details on the panel of scenarios on which our Lasso estimator and our classifier are evaluated.

MHP path generation. Concerning synthetic data generation, each path is simulated using cluster representation algorithm (see Møller and Rasmussen (2005)). This sampling procedure relies on the branching structure of the MHP, that can be viewed as Poisson cluster process. We consider the classical choice of exponential kernel $h(s) = \beta \exp(-\beta s)$ with $\beta = 3$.

Scenarios. We consider two scenarios, referred to as *Scenario 1* and *Scenario 2*. In both scenarios, different structures of the interaction matrix A^* are explored. In *Scenario 1*, A^* is chosen to be a diagonal block matrix. In addition to self-exciting interaction, the block structure models interaction between a group of connected components. Coefficient values, which gives the intensity of influence, are the same within each block, but vary from one to another. For a larger value of M, the blocks are expanded so that the parsimony rate remains the same for each value of M. In *Scenario 2*, the coefficients are chosen randomly with different values. Due to the randomness of the choice of the active set, the diagonal coefficients may be all set to zero. Thus, there may be no self-excitation in this case. In both scenarios, the vector of exogenous intensity μ^* is chosen as constant for each component, meaning that spontaneous events occur in the same way for each individual. In Figure 1, a visual representation of theses scenarios, for $M = 25$ is given in the form of a heat map. In particular, the values of the coefficients of the matrix A^* are given by the color bar. We precise also the sparsity rate, which is the % of zero-coefficients in the matrix, i.e. $|S^{*c}|/M^2$.



(a) *Scenario 1* (b) *Scenario 2*

Figure 1: Visualization of $\theta^* = (\mu^*, A^*)$ in both scenarios for $M = 25$. The exogenous intensity for each component: 0.4. Sparsity rate of A^* in *Scenario 1*: 85%; in *Scenario 2*: 92%.

To illustrate the classification task, we consider the 3-classes classification setting, i.e. $K = 3$. The three classes are created on the basis of the two scenarios described above. For *Scenario 1*, the blocks of different size are interchanged, as well as the values of the coefficients within them. For *Scenario 2*, based on the same support for each class, the values of the coefficient are interchanged. In both cases, the resulting classes are quite balanced and close from each other. Finally, the exogenous intensity is chosen to be the same for each of the three classes. In Table 1, we give for each scenario, the values of the Frobenius norm, the spectral radius and the sparsity

rate as a function of the dimension and of the label. For terminology convenience, we refer to the classification scenario resulting from *Scenario 1* (resp. *Scenario 2*) as *Scenario 1* (resp. *Scenario 2*).

		Scenario 1			Scenario 2		
		$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$
M=10	$\|A_k^*\|_F$	1.37	1.37	1.39	1.44	1.54	1.31
	$\rho(A_k^*)$	0.76	0.76	0.76	0.00	0.00	0.00
	$\|S_k^{*c}\|$	0.86	0.86	0.86	0.89	0.89	0.89
M=25	$\|A_k^*\|_F$	1.63	1.63	1.68	2.07	2.25	2.07
	$\rho(A_k^*)$	0.90	0.90	0.90	0.50	0.55	0.44
	$\|S_k^{*c}\|$	0.85	0.85	0.85	0.92	0.92	0.92
$M=50$	$\|A_k^*\|_F$	1.52	1.52	1.52	2.55	2.77	2.42
	$\rho(A_k^*)$	0.90	0.90	0.90	0.68	0.74	0.63
	$\|S_k^{*c}\|$	0.85	0.85	0.85	0.94	0.94	0.94

Table 1: Presentation of the different scenarios with $K=3$. For each class k, the Frobenius norm, the spectral radius and the sparsity rate of A_k^* are specified.

6.3 Evaluation scheme

Hereafter, we present the evaluation scheme that relies on Monte-Carlo repetitions. We fix $T=5$, and $p^* \sim \mathcal{U}_{[3]}$. For each scenario described, each value of $M \in \{10, 25, 50\}$, and each value of $n \in \{300, 600, 1500\}$, we repeat independently 30 times the following steps.

1. Simulate the data set $\mathcal{D}_{n_{\text{train}}}$ and $\mathcal{D}_{n_{\text{test}}}$;
2. Based on $\mathcal{D}_{n_{\text{train}}}$, for each $k=1,\dots,K$ compute $\widehat{p}_k = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{Y^i=k\}}$;
3. Based on $\mathcal{D}_{n_{\text{train}}}$, Lasso step:
   (a) For each $k \in [K]$, calibrate the penalization constant using $\texttt{EBIC}_1$ criteria by exploring values in the grid Δ. For each $\kappa \in \Delta$ do:
      i. using `FISTA`, compute $\widehat{\theta}_k$ the Lasso estimate with tuning parameter κ;
      ii. based on $\widehat{\theta}_k$, compute $\texttt{EBIC}_1(\kappa)$;

      and choose $\widehat{\kappa}_k \in \underset{\kappa\in\Delta}{\text{argmin}}\ \texttt{EBIC}_1(\kappa)$;
   (b) Given $(\widehat{\kappa}_k)_{k\in\mathcal{Y}}$, for each $k=1,\dots,K$ do:
      i. using `FISTA`, compute the Lasso estimates $\widehat{\theta}_k$ with tuning parameter $\widehat{\kappa}_k$;
      ii. get the estimated support $\widehat{S}_k = \left\{(j,j') \in [M],\ \ \widehat{\theta}_{k,j,j'} \neq 0\right\}$.
   (c) From $(\widehat{S}_k)_{k\in\mathcal{Y}}$ compute the classifier $\widehat{g}_{\texttt{OES}}$, from $(\widehat{\theta}_k)_{k\in\mathcal{Y}}$ compute the classifier $\widehat{g}_{\texttt{PI}}$

4. For one arbitrary class $k \in \mathcal{Y}$, assess the quality of the support recovery using Hamming distance and ℓ_2 distance defined as

$$d_H\left(A_k^*, \widehat{A}_k\right) = \frac{1}{M^2}\sum_{j,j'=1}^{M} \mathbb{1}_{\left\{A^*_{k,j,j'} \neq \widehat{A}_{k,j,j'}\right\}}, \quad \text{and} \quad d_{\ell_2}\left(A_k^*, \widehat{A}_k\right) = \sqrt{\sum_{j,j'=1}^{M} \left|A^*_{k,j,j'} - \widehat{A}_{k,j,j'}\right|^2};$$

5. From $\mathcal{D}_{n_{\text{train}}}$, perform the `ERM` step:

   (a) starting from $(\widehat{\theta}_k)_{k\in\mathcal{Y}}$ as the initial point, we minimize the L_2-risk defined in Equation (8) using `Free AdaGrad` to obtain $(\widehat{\theta}^{\text{R}}_k)_{k\in\mathcal{Y}}$;

   (b) from $\widehat{\theta}^{\text{R}}$ and $\widehat{p}$ we build the classifiers $\widehat{g}_{\texttt{ERMLR}}$.

6. Based on $\mathcal{D}_{n_{\text{test}}} = \{(\mathcal{T}_T^{(i)}, Y^i), i = 1, \ldots, n_{\text{test}}\}$, evaluate the error rate of the classifiers `PI` and `ERMLR` using

$$\text{Err}_{\texttt{PI}} = \frac{1}{n_{\text{test}}}\sum_{i=1}^{n_{\text{test}}} \mathbb{1}_{\{\widehat{g}_{\texttt{PI}}(\mathcal{T}_T^i)\neq Y^i\}}, \quad \text{and} \quad \text{Err}_{\texttt{ERMLR}} = \frac{1}{n_{\text{test}}}\sum_{i=1}^{n_{\text{test}}} \mathbb{1}_{\{\widehat{g}_{\texttt{ERMLR}}(\mathcal{T}_T^i)\neq Y^i\}};$$

6.4 Numerical Results for support recovery

This section is devoted to the discussion of the obtained results of the Lasso procedure. These results are provided in Table 2, in Table 3, in Figure 2 and in Figure 3.

	M	d_H			d_{ℓ_2}		
		$n = 100$	$n = 500$	$n = 1000$	$n = 100$	$n = 500$	$n = 1000$
Scenario 1	10	0.04 (0.03)	0.02 (0.02)	0.02 (0.02)	0.39 (0.07)	0.18 (0.04)	0.13 (0.02)
	25	0.04 (0.01)	0.03 (0.01)	0.03 (0.01)	0.91 (0.07)	0.40 (0.04)	0.29 (0.02)
	50	0.11 (0.01)	0.07 (0.00)	0.07 (0.00)	1.80 (0.12)	1.60 (0.02)	1.64 (0.02)
Scenario 2	10	0.04 (0.02)	0.03 (0.02)	0.03 (0.02)	0.43 (0.07)	0.20 (0.03)	0.14 (0.02)
	25	0.03 (0.01)	0.03 (0.01)	0.03 (0.01)	0.96 (0.11)	0.44 (0.04)	0.32 (0.04)
	50	0.03 (0.00)	0.02 (0.00)	0.01 (0.00)	1.76 (0.09)	0.94 (0.07)	0.68 (0.04)

Table 2: Lasso results over 30 Monte-Carlo repetitions for both scenarios for three value of M. The impact of n is investigated. The standard deviation is provided between parentheses. $T = 5$

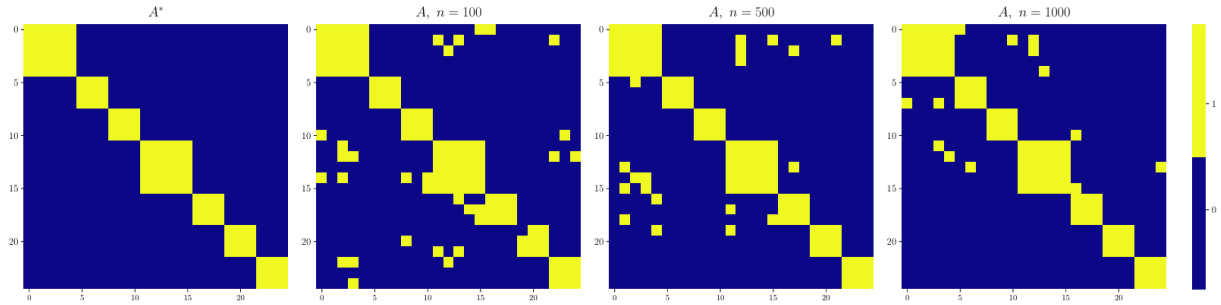


Figure 2: True support $\text{supp}(\theta^*)$ and recovered support $\text{supp}(\hat{\theta})$ in *Scenario 1*.

First, from Table 2, we can see that Hamming distance between the true support and the estimated one is close to zero in all settings, therefore our procedure is able to correctly recover the active set of A^*. This remains true even for small values of n and for high-dimensional networks, for which the Hamming distance is quite small. As expected, the larger n is, the better the support is reconstructed, whether in terms of Hamming distance or ℓ_2 distance. Thus, in addition to reconstructing the support more accurately, a gain is also made in terms of point parameter estimation, illustrating the theoretical result of support consistency and convergence of the associated estimator established in Section 4. In particular, for large value of M, such as $M = 50$, a clear decrease in the Hamming distance is noticeable for increasing values of n. Finally, it is worth emphasizing that, in the case of *Scenario 1*, the Lasso procedure is successful in recovering the underlying block structure of the interaction matrix A^*. This assertion is supported by the Figure 2, which visually shows the convergence of the support to the actual structure as the number of observations increases.

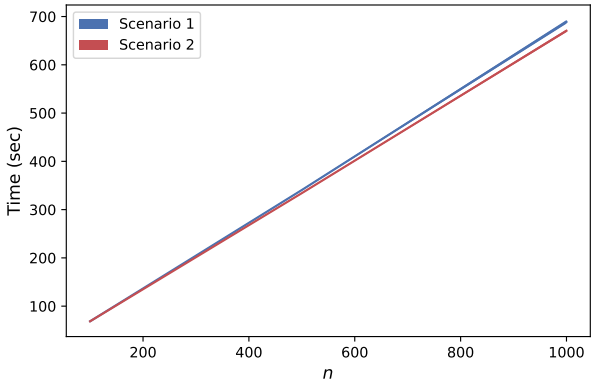


Figure 3: Average execution time over 30 repetitions of the entire Lasso procedure as a function of n for *Scenario 1* with $M = 25$. The standard deviation is shown in shaded fill on either side of the curve.

	M	# events			time (sec)		
		$n = 100$	$n = 500$	$n = 1000$	$n = 100$	$n = 500$	$n = 1000$
Scenario 1	10	6330 (197)	32079 (398)	63905 (666)	8.37 (0.09)	42.12 (0.35)	82.42 (0.53)
	25	14221 (397)	71226 (773)	142073 (1789)	68.67 (0.46)	340.62 (1.36)	688.99 (3.70)
	50	12993 (513)	64743 (839)	129969 (1252)	447.22 (4.21)	2290.80 (22.63)	4519.32 (49.73)
Scenario 2	10	4692 (125)	23367 (250)	46570 (402)	8.17 (0.09)	39.81 (0.34)	80.35 (0.58)
	25	9524 (147)	47651 (354)	95737 (632)	68.64 (0.22)	334.13 (1.30)	670.51 (2.67)
	50	12363 (254)	62228 (575)	124604 (1142)	459.69 (1.36)	2268.71 (7.69)	4522.17 (16.79)

Table 3: Number of observed events, average execution time over 30 Monte-Carlo repetitions for both scenarios. The standard deviation is provided between parentheses. $T = 5$

Now let us discuss the computational cost of our procedure. It can be seen from Figure 3 and Table 3 that the execution time of the entire procedure is quite reasonable, even for a large value of M. It is important noting that the execution time also includes the choice of κ with the `EBIC` criterion, and this with a grid of fine size $|\Delta| = 40$. Thus, we can afford to explore with great precision and still have a relatively short execution time. For comparison purposes, it is worth noting that, as we are dealing with short-time path repetitions, our observations would be

equivalent to a unique path of horizon time of $n \times T$. Finally, as our procedure benefits from fast computational properties, it appears therefore realistic to apply it to large-scale networks. This could be a matter of great interest for real-world applications, which often involve a network of huge dimension.

6.5 Numerical results for classification

This section is devoted to the discussion of the obtained results of the `ERMLR` procedure. These results are provided in Table 4, and Figure 4.

	M	Bayes	`OES`	`PI`	`ERMLR`
Scenario 1	10	0.134 (0.005)	0.135 (0.006)	0.155 (0.007)	**0.152** (0.007)
	25	0.087 (0.004)	0.107 (0.013)	0.143 (0.011)	**0.134** (0.011)
	50	0.092 (0.005)	0.313 (0.05)	**0.218** (0.020)	0.219 (0.019)
Scenario 2	10	0.251 (0.007)	0.255 (0.008)	**0.276** (0.012)	0.278 (0.010)
	25	0.237 (0.008)	0.260 (0.014)	**0.316** (0.016)	0.326 (0.012)
	50	0.246 (0.008)	0.406 (0.031)	**0.391** (0.026)	0.410 (0.032)

(a) $n = 300$

	M	Bayes	`OES`	`PI`	`ERMLR`
Scenario 1	10	0.135 (0.006)	0.135 (0.006)	0.146 (0.007)	**0.144** (0.008)
	25	0.086 (0.004)	0.087 (0.004)	0.118 (0.005)	**0.113** (0.005)
	50	0.091 (0.004)	0.189 (0.021)	0.183 (0.008)	**0.179** (0.009)
Scenario 2	10	0.247 (0.008)	0.248 (0.008)	**0.260** (0.009)	0.262 (0.008)
	25	0.236 (0.007)	0.237 (0.008)	0.276 (0.010)	**0.276** (0.010)
	50	0.245 (0.008)	0.309 (0.014)	**0.333** (0.016)	0.349 (0.020)

(b) $n = 600$

	M	Bayes	`OES`	`PI`	`ERMLR`
Scenario 1	10	0.135 (0.005)	0.136 (0.005)	0.139 (0.005)	**0.139 (0.006)**
	25	0.087 (0.006)	0.087 (0.005)	0.100 (0.006)	**0.098** (0.006)
	50	0.093 (0.005)	0.183 (0.08)	0.179 (0.07)	**0.173** (0.08)
Scenario 2	10	0.253 (0.009)	0.253 (0.009)	**0.257** (0.009)	0.259 (0.010)
	25	0.236 (0.009)	0.236 (0.009)	**0.253** (0.008)	0.254 (0.008)
	50	0.247 (0.008)	0.251 (0.009)	**0.293** (0.012)	0.296 (0.011)

(c) $n = 1500$

Table 4: Empirical error over 30 Monte-Carlo repetitions for each classifier in the three scenarios for three values of M. The impact of n is investigated. The standard deviation is provided between parentheses. The value of $n_{\text{test}} = 3000$ is chosen. $T = 5$

First, from Table 4, we can see that the `ERMLR` is close to the Bayes classifier in terms of error rate, in both scenarios and for each value of M. In particular, note that for $n = 1500$, its error rate is almost equal to that of the Bayes classifier. In fact, as expected the greater the number of

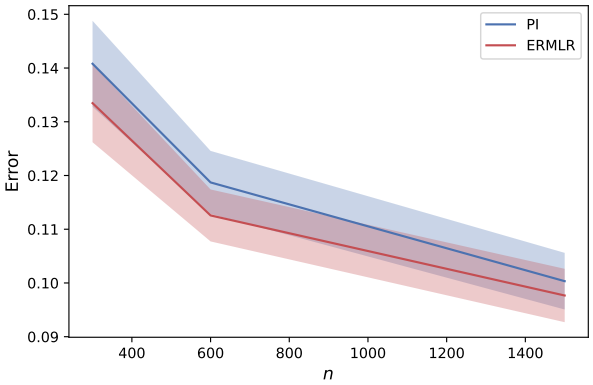


Figure 4: Averaged Error rate over 30 repetitions of the `ERMLR` and `PI` procedure as a function of n for *Scenario 1* with $M = 25$. The standard deviation is shown in shaded fill on either side of the curve.

data, the closer the classifier comes to the Bayes classifier, which illustrate the consistency of the `ERMLR` procedure established in Section 4. This decreasing tendency of the error rate of `ERMLR` is illustrated in *Scenario 1* with $M = 25$ in Figure 4.

Another important point is the comparison with the `PI` classifier as a benchmark. Overall, it can be seen that the `PI` exhibits good performance. This can be explained by the fact that recovering the true support structure is sufficient for accurate class prediction. On the other hand, poor support recovery also impacts the performance of the `ERMLR` predictor. This gap can be quantified with the `OES` oracle classifier, which gives the gain that could be obtained by an `ERM` step. For these reasons, it is not expected to see a big gap between the two. Nevertheless, it is worth noting that, in case of *Scenario 1*, a significant gain by the `ERM` refitting step can be observed. This assertion is supported by Figure 4, where it can be seen that the `ERMLR` classifier is better in terms of error rate. This suggests that, for some particular structures, a refitting step leading to a finer point estimate of the parameters is relevant and leads to better performance.

7 Discussion

In the present work, we propose a novel classification algorithm tailored to classify Multivariate Hawkes Processes paths in high-dimension. For each class, a first step is dedicated to the sparse estimation of the support of the adjacency matrix. Then, in a second step, we build a classifier that takes of advantage of the estimated support. Specifically, the resulting classifier is based on the minimization of a `ERM` criterion. We establish rates of convergence for both estimated support and classification algorithm. Finally, we illustrate the numerical performance of our procedure through a comprehensive simulation study.

A possible guideline for further research is to consider a more challenging model by including inhibition interaction. From a theoretical aspect, it may be tricky since adding inhibition effect induces complication due to the non linearity of the underlying intensity function. In particular, providing a closed form of the compensator is a key aspect to compute the least-square contrast or the likelihood function. The work of Bonnet et al. (2022) and Bonnet et al. (2023) should form a theoretical basis for this future work. From a practical point of view, a procedure which is able to deal with inhibition, may be applied to generalize the work of Denis et al. (2024). Indeed, the use of MHP allows to model simultaneously different species echolocation calls and then the

effects of inter-species cooperation. Furthermore, adding inhibition effects, potentially translates the ecological aspect of inter-species competition.

Another direction could be to investigate a penalized `ERM` classifier. It would allow to deal with the high-dimensional setting without the prior Lasso step. Indeed, this procedure relies on a global penalized criterion dedicated to the classification task. This direction is left for further investigations.

Finally, from a practical standpoint, `sparkle`, a full Python library for Hawkes process inference in high-dimension and classification is in development. It consists in a toolkit for Hawkes process modeling which relies on `C++` codes wrapped in `Python` for fast computation.

Acknowledgements

This work has been supported by the Chaire "Modélisation Mathématique et Biodiversité" of Veolia-École polytechnique-Museum national d'Histoire naturelle-Fondation X, through a Ph.D. scholarship. The project is also part of the 2022 DAE 103 EMERGENCE(S) - PROCECO project supported by Ville de Paris. Finally, the authors thank Vincent Rivoirard for fruitful discussions.

Appendix

This appendix gathers the proofs of the theoretical results of the paper. It is organized as follows. Appendix A provides useful technical results. The proof of the closed-form expression of the Bayes classifier is established in Appendix B. The proof of the support recovery result is given in Appendix C. Finally, the rate of convergence of the ***`ERMLR`*** *algorithm is proved in Appendix D.*

Throughout the proofs, the notation C refers to a generic positive constant, which may differ from line to line. In particular, this generic constant C does not depend on n or on the dimension M. However, it may depend on the other parameters. For the sake of simplicity we denote $\mathcal{T}$ for $\mathcal{T}_T$.

Appendix A Technical results

Proposition A.1. *For any classifier $g \in \mathcal{G}$, we have*

$$\mathcal{R}(g) - \mathcal{R}(g^*) = \mathbb{E}\left[\sum_{1\le i\neq k\le K} |\pi_i^*(\mathcal{T}) - \pi_k^*(\mathcal{T})| \mathbb{1}_{\{g^*(\mathcal{T})=i, g(\mathcal{T})=k\}}\right].$$

Proof. This result is established by Denis et al. (2022). Let $g \in \mathcal{G}$ a classifier. We observe that

$$\mathcal{R}(g) = \mathbb{E}\left[\mathbb{1}_{\{g(\mathcal{T})\neq Y\}}\right] = 1 - \mathbb{E}\left[\mathbb{1}_{\{g(\mathcal{T})=Y\}}\right] = 1 - \mathbb{E}\left[\pi^*_{g(\mathcal{T})}\right].$$

Therefore, from the above equation and the definition of the Bayes classifier g^*, we get

$$\mathcal{R}(g) - \mathcal{R}(g^*) = \mathbb{E}\left[\left|\pi^*_{g^*(\mathcal{T})} - \pi^*_{g(\mathcal{T})}\right|\right].$$

Since for each $g \in \mathcal{G}$, $g(\mathcal{T}) = \sum_{k=1}^K k\mathbb{1}_{\{g(\mathcal{T})=k\}}$, the above equation yields the result. □

Lemma A.1. *Let $A \in \mathbb{R}^{d\times d}$ (symmetric), and $X \in \mathbb{R}^d$. Then,*

$$\|AX\|_\infty \le \sqrt{d}\rho(A)\|X\|_\infty$$

Lemma A.2 (Hoeffding). *Let $B \sim \mathcal{B}(n,p)$, with $p \in (0,1)$. We then have for all $t > 0$ and $n > \frac{t}{p}$,*

$$\mathbb{P}(B \leq t) \leq \exp\left(-2n(p - t/n)^2\right).$$

Appendix B Proof for Bayes classifier

We first denote for all $k \in \mathcal{Y}$

$$\Phi_t^k := \frac{\mathrm{d}\mathbb{P}_k|_{\mathcal{F}_t^N}}{\mathrm{d}\mathbb{P}_0|_{\mathcal{F}_t^N}},$$

with $\mathcal{F}_T^N := \sigma\left(\mathcal{T}_T\right) = \sigma\left(N_t, 0 \leq t \leq T\right)$. We classically obtain:

$$\log(\Phi_t^k) = -\sum_{j=1}^{M} \int_0^t (\lambda_{k,j}^*(s) - 1)\,\mathrm{d}s + \int_0^t \log(\lambda_{k,j}^*(s))\,\mathrm{d}N_j(s),$$

by writing *w.r.t.* a Poisson process measure of intensity 1 (see Chapter 13 of Daley and Vere-Jones, 2003). Thus, for $t \geq 0$, we have the following equation for the mixture measure

$$d\mathbb{P}|_{\mathcal{F}_t^N} = \sum_{k=1}^{K} p_k\,\mathrm{d}\mathbb{P}_k|_{\mathcal{F}_t^N} = \sum_{k=1}^{K} p_k \Phi_t^k\,\mathrm{d}\mathbb{P}_0|_{\mathcal{F}_t^N}$$

and then

$$\frac{\mathrm{d}\mathbb{P}_k|_{\mathcal{F}_t^N}}{\mathrm{d}\mathbb{P}|_{\mathcal{F}_t^N}} = \frac{p_k \Phi_t^k\,\mathrm{d}\mathbb{P}_0|_{\mathcal{F}_t^N}}{\sum_{j=1}^K p_j \Phi_t^j\,\mathrm{d}\mathbb{P}_0|_{\mathcal{F}_t^N}} = \frac{p_k \Phi_t^k}{\sum_{j=1}^K p_j \Phi_t^j}.$$

Finally, by using (3), it comes $\pi_k^*\left(\mathcal{T}_T\right) = \frac{p_k^* \mathrm{e}^{F_k^*}}{\sum_{j=1}^K p_j^* \mathrm{e}^{F_j^*}}$, that concludes the proof.

Appendix C Proofs for support recovery

In this section, we gather the proof of the result provided in Section 4.1. We first recall and introduce the main notations for the proof of the main result in Section C.1. Then, in section C.2 we establish a Bernstein lemma. This lemma is the cornerstone of the proof of the support recovery which is given in Section C.3.

C.1 Notations

We recall that the learning sample is $D_n = \{\left(\mathcal{T}_T^i, Y_i\right), \ldots, (\mathcal{T}_T^n, Y_n)\}$. Let $k \in \mathcal{Y}$ be a fixed integer. Throughout this section, all the results are established for a generic class k. Let us define the random variables

$$n_k = \sum_{i=1}^{n} \mathbb{1}_{Y^{(i)}=k}.$$

Hence $n_k \sim \mathcal{B}(n, p_k^*)$. We also recall that $\min_{k\in[K]} p_k^* \geq p_0 > 0$.

For sake of simplicity, we remove the dependency *w.r.t.* k. To sum up, our parameters of interests are μ, A, and we at our disposal a sample of (random) size n_k. In the rest of this section, we work **conditional on** $\{n_k \geq 1\}$.

C.2 A Bernstein lemma

Lemma C.1 (Bernstein Lemma)**.** *Assume that* $n \geq \dfrac{2}{p_0^*}$*. Let us define the event*

$$\Omega_n := \left\{ \frac{1}{n_k} \max_{j,j'} \left| Z_{j,j'} \right| \leq C \frac{\log^3 \left(nM^2 \right)}{\sqrt{n}} \right\} \bigcap \left\{ n_k \geq \frac{np_k^*}{2} \right\}.$$

There exists $C_{\|h\|_\infty, p_0^*} > 0$*, such that* $\mathbb{P}\left(\Omega_n\right) \geq 1 - \dfrac{C_{\|h\|_\infty, p_0^*}}{n}$.

Proof. Fore clarity of presentation, the proof is divided in two steps.

First step. In this step we work on the event $\{n_k \geq 1\}$ and conditional on $\mathbb{1}_{\{Y_1=k\}}, \ldots, \mathbb{1}_{\{Y_n=k\}}$. For $j, j^{'} \in [M] \times (\{0\} \cup [M])$, we apply Theorem 4 in Bacry et al. (2020) to the real valued random variable $Z_{j,j'}$. For clarity, we consider the same notations as in Bacry et al. (2020).

To this end, for a fixed $(j, j^{'}) \in [M] \times (\{0\} \cup [M])$ and $t \in [0; T]$, we define the tensor (see Bacry et al. (2020) for its definition and related properties) $\mathbb{T}_t$ of shape $1 \times 1 \times M \times n_k$ as follows

$$(\mathbb{T}_t)_{1,1,k,\ell} = \begin{cases} H_{j'}^{(\ell)}(t) \ \text{ if } \ k = j \\ 0 \ \text{ else,} \end{cases} \tag{12}$$

for $k \in [M]$ and $\ell \in [n_k]$. We also recall that the matrix $\mathrm{d}M(t)$ is defined by the main term $\mathrm{d}M(t)_{j,i} = \mathrm{d}M_j^{(i)}(t)$. According to Bacry et al. (2020) we have that $Z_{j,j'} = Z_{\mathbb{T}}(T) \in \mathbb{R}$ defined by

$$Z_{\mathbb{T}_t}(T) = \int_0^T \mathbb{T}_t \circ dM_t$$

satisfies

$$Z_{\mathbb{T}_t}(T) = \sum_{k=1}^{M} \sum_{i=1}^{n_k} \int_0^T (\mathbb{T}_t)_{1,1,k,i} dM_{k,i}(t) = \sum_{i=1}^{n_k} \int_0^T H_{j'}^{(i)}(t) \, \mathrm{d}M_j^{(i)}(t). \tag{13}$$

Furthermore, we observe that since the tensor $\mathbb{T}_t$ is symmetric we have

$$\widehat{V}_{\mathbb{T}}(t) := \int_0^t \mathbb{T}_s^2 \circ \, \mathrm{d}N_s = \sum_{i=1}^{n_k} \int_0^t \left(H_{j'}^{(i)}(s) \right)^2 \mathrm{d}N_j^{(i)}(s),$$

and

$$b_{\mathbb{T}_t} := \sup_{0 \leq s \leq t} \max(\|\mathbb{T}_s\|_{\mathrm{op},\infty} \|\mathbb{T}_s'\|_{\mathrm{op},\infty}) = \sup_{0 \leq s \leq t} \max_{i=1,\ldots,n_k} \left| H_{j'}^{(i)}(s) \right|$$

which both depend on (j, j').

Applying Theorem 4 of Bacry et al. (2020) on the event $\{n_k \geq 1\}$ and conditional on $\mathbb{1}_{\{Y_1=k\}}, \ldots, \mathbb{1}_{\{Y_n=k\}}$, we then obtain that for $x > 0$ with probability at least $1 - C\exp(-x)$ the following holds

$$\left| Z_{j,j'} \right| \leq 2\sqrt{\lambda_{\max}(\widehat{V}_{\mathbb{T}_T})(x + \ell_x(T))} + c(x + \ell_x(T)) \left(1 + b_{\mathbb{T}_T}\right), \tag{14}$$

since for all $(j, j^{'}) \in [M] \times \{0, \ldots M\}$,

$$\lambda_{\max}\left(\widehat{V}_{\mathbb{T}}(T) \right) \leq \widehat{V}_\infty := \max_{i,j'} \left(H_{j'}^{(i)}(T) \right)^2 \max_{i,j} N_j^{(i)}(T), \tag{15}$$

and

$$b_{\mathbb{T}_T} \leq b_\infty := \max_{i,j'} \left| H_{j'}^{(i)}(T) \right|, \tag{16}$$

from Equation (14), setting $x = \log(nM^2)$, with an union bound on j, j' we obtain that the event

$$\mathcal{Z} = \left\{ \max_{j,j'} \left| Z_{j,j'} \right| \leq 2\sqrt{\widehat{V}_\infty \left(\log(nM^2) + \ell_\infty\right)} + c\left(\log(nM^2) + \ell_\infty\right)(1 + b_\infty) \right\},$$

with

$$\ell_\infty = 2\log\log\left(\frac{4\widehat{V}_\infty}{\log(nM^2)} \vee 2\right) + 2\log\log\left(4b_\infty \vee 2\right), \tag{17}$$

satisfies

$$\mathbb{1}_{\{n_k \geq 1\}} \mathbb{P}\left(\mathcal{Z}^c | \mathbb{1}_{\{Y^{(1)}=k\}}, \ldots, \mathbb{1}_{\{Y^{(n)}=k\}}\right) \leq \mathbb{1}_{\{n_k \geq 1\}} \frac{C}{n} \leq \frac{C}{n}.$$

From the above inequality, we deduce that

$$\begin{aligned}
\mathbb{P}\left(\mathcal{Z}^c\right) &= \mathbb{P}\left(\mathcal{Z}^c, n_k \geq 1\right) + \mathbb{P}\left(\mathcal{Z}^c, n_k = 0\right) \\
&\leq \frac{C}{n} + \mathbb{P}\left(n_k = 0\right) \\
&\leq \frac{C}{n} + \exp(n\log(1 - p_k)) \\
&\leq \frac{C}{n} + \exp(n\log(1 - p_0) \leq \frac{C}{n}.
\end{aligned} \tag{18}$$

Second step. In this step, we provide a bound for $\widehat{V}_\infty$, b_∞, and ℓ_∞ respectively defined in Equation (15), (16), and (17). To this end, we introduce the event

$$\Omega = \left\{ n_k \geq \frac{np_k}{2} \right\} \bigcap \left\{ \mathbb{1}_{\{n_k \geq 1\}} \max_{i,j} N_j^{(i)}(T) \leq \log^{5/3}(Mn) \right\}.$$

Note that, in view of the definition of $H_{j'}^{(i)}$, we have that on the event $\{n_k \geq 1\}$, we have

$$\widehat{V}_\infty \leq \max\left(n_k \left\|h\right\|_\infty \max_{i,j} \left(N_j^{(i)}(T) \right)^3, n_k \max_{i,j} (N_j^{(i)}(T) \right) \leq C_{\left\|h\right\|_\infty} n_k \log^5(n).$$

With the same idea, we have that $b_\infty \leq C_{\left\|h\right\|_\infty} \log^{5/3}(n)$. Finally, we observe that $\ell_\infty \leq 2\log(nM^2)$ (as $M \geq 2$). Hence, on the event $\Omega \cap \mathcal{Z}$, it holds that $n_k \geq 1$ (since $n \geq \frac{2}{p_0}$), and,

$$\frac{1}{n_k} \max_{j,j'} \left| Z_{j,j'} \right| \leq C \frac{\log^3\left(nM^2\right)}{\sqrt{n_k}} \leq C \frac{\log^3\left(nM^2\right)}{\sqrt{np_k}} \leq C \frac{\log^3\left(nM^2\right)}{\sqrt{np_0}}.$$

To conclude the proof, since $\mathbb{P}\left(\Omega_n^c\right) \leq \mathbb{P}\left((\mathcal{Z} \cap \Omega)^c\right)$, it remains to control $\mathbb{P}\left((\mathcal{Z} \cap \Omega)^c\right)$.

Conditional on $\mathbb{1}_{\{Y_1=k\}}, \ldots, \mathbb{1}_{\{Y_n=k\}}$, on the event $\{n_k \geq 1\}$, applying the sub-exponential property of $N_j^{(i)}$, and Proposition 2.7.1 in Vershynin (2018), we get

$$\begin{aligned}
\mathbb{P}\left(\max_{i,j} N_j^{(i)}(T) > \log^{5/3}(Mn) \right) &\leq Mn_k \exp\left(-c\log^{5/3}(nM) \right) \\
&\leq \frac{1}{n}.
\end{aligned}$$

Therefore, from Lemma A.2,

$$\begin{aligned}\mathbb{P}\left(\Omega^c\right) &\leq \frac{1}{n} + \mathbb{P}\left(n_k \leq \frac{np_k}{2}\right) \leq \frac{1}{n} + \exp\left(-n\frac{p_0}{2}\right) \\ &\leq \frac{1}{n}.\end{aligned}$$

Finally, combining the last equation with Equation (18), we deduce that,

$$\mathbb{P}\left((\mathcal{Z} \cap \Omega)^c\right) \leq \frac{C}{n},$$

which yields the result.

□

C.3 Proof of the main result 1

Throughout the proof, we work on the event

$$\Omega_n := \left\{ \frac{1}{n_k} \max_{j,j'} \left|Z_{j,j'}\right| \leq C\frac{\log^3\left(nM^2\right)}{\sqrt{n}} \right\} \bigcap \left\{ n_k \geq \frac{np_k}{2} \right\}.$$

Note that on the event Ω_n, since $n \geq \frac{2}{p_0}$, the random variable n_k satisfies $n_k \geq 1$.

The proof follows the *primal-dual witness method* as in Hastie et al. (2015) Chapter 11, and goes in several steps. Let us consider the penalized contrast

$$\mathcal{C}(\theta) := R_{T,n_k}(\theta) + \kappa \sum_{j=1}^{M}\sum_{j'=1}^{M} |\theta_{j,j'}|. \tag{19}$$

An element z of the subgradient of $\mathcal{C}$ at some point θ writes as follows

$$\nabla R_{T,n_k}(\theta) + \kappa z,$$

where the concatenated vector z is $z = (z_1, \ldots, z_M)'$ with $z_{j,0} = 0$ and $z_{j,j'} = \text{sign}(\theta_{j,j'})$ for $j' \geq 2$ (with the convention that $\text{sign}(0) \in [-1, 1]$). We say that a pair $(\widehat{\theta}, \widehat{z})$ is optimal if it satisfies the following zero-subgradient equation

$$\nabla R_{T,n_k}(\widehat{\theta}) + \kappa\widehat{z} = 0. \tag{20}$$

First step. We first build an "oracle" pair $(\widehat{\theta}, \widehat{z})$ that satisfies Equation (20) and such that $\widehat{\theta}_{S^{*c}} = 0$. First we define $\widehat{\theta}$, and $\widehat{z}_{S^*}$ as follows.

1. $\widehat{\theta}_{S^{*c}} = 0$,
2. $\widehat{\theta}_{S^*} \in \underset{\theta_{S^*}}{\text{argmin}}\ \widetilde{R}_{T,n_k}(\theta_{S^*}) + \kappa \sum_{j=1}^{M} \sum_{j' \in S^*_{\theta_j}} |\theta_{j,j'}|$, where

$$\widetilde{R}_{T,n_k}(\theta_{S^*}) = \frac{1}{n_k T}\sum_{i=1}^{n_k}\sum_{j=1}^{M}\int_0^T \left(\sum_{j' \in S^*_{\theta_j}} \theta_{j,j'} H^{(i)}_{j'}(t)\right)^2 \mathrm{d}t - 2\int_0^T \left(\sum_{j' \in S^*_{\theta_j}} \theta_{j,j'} H^{(i)}_{j'}(t)\right) \mathrm{d}N^{(i)}_j(t).$$

In view of the above conditions, since $\widehat{\theta}_{S^*}$ is a minimizer, we have for each $j \in [M]$

$$\left(\nabla R_{T,n_k}(\widehat{\theta})\right)_{S^*_{\theta_j}} + \kappa \widehat{z}_{S^*_{\theta_j}} = 0.$$

We then have to build for each $j \in [M]$, $\widehat{z}_{S_j{*c}}$ such that

$$\left(\nabla R_{T,n_k}(\widehat{\theta})\right)_{S^{*c}_{\theta_j}} + \kappa \widehat{z}_{S_j{*c}} = 0.$$

Hence, from the above equations and from the notation given in Equation (10), we deduce that $(\widehat{\theta}, \widehat{z})$ must satisfies

$$\frac{2}{n_k}\mathbb{H}_{S^{*c}_{\theta_j}, S^*_{\theta_j}} \left(\widehat{\theta}_j - \theta^*_j\right)_{S^*_j} - \frac{2}{n_k}(Z_j)_{S^{*c}_{\theta_j}} + \kappa z_{S^{*c}_{\theta_j}} = 0,$$

and

$$\frac{2}{n_k}\mathbb{H}_{S^*_j, S^*_{\theta_j}} \left(\widehat{\theta}_j - \theta^*_j\right)_{S^*_j} - \frac{2}{n_k}(Z_j)_{S^*_j} + \kappa z_{S^*_j} = 0.$$

From the last equation, and as $z_{S^*_{\theta_j}} = \mathrm{sign}((\widehat{\theta}_j)_{S^*_{\theta_j}})$, we observe that

$$\left(\widehat{\theta}_j - \theta^*_j\right)_{S^*_{\theta_j}} = \mathbb{H}^{-1}_{S^*_j, S^*_{\theta_j}}(Z_j)_{S^*_j} - \frac{n_k\kappa}{2}\mathbb{H}^{-1}_{S^*_{\theta_j}, S^*_{\theta_j}} \mathrm{sign}((\widehat{\theta}_j)_{S^*_{\theta_j}}). \tag{21}$$

Therefore, we set for each $j \in [M]$,

$$z_{S^{*c}_{\theta_j}} = -\frac{2}{n_k\kappa}\left(\mathbb{H}_{S^{*c}_{\theta_j}, S^*_{\theta_j}} \mathbb{H}^{-1}_{S^*_j, S^*_{\theta_j}} (Z_j)_{S^*_j} - (Z_j)_{S^{*c}_{\theta_j}}\right) + \mathbb{H}_{S^{*c}_{\theta_j}, S^*_{\theta_j}} \mathbb{H}^{-1}_{S^*_j, S^*_{\theta_j}} \mathrm{sign}((\widehat{\theta}_j)_{S^*_{\theta_j}}). \tag{22}$$

We then have build an optimal solution $(\widehat{\theta}, \widehat{z})$ that satisfies the required condition.

Second step. The goal of the second step is to prove that $\|\widehat{z}_{S^{*c}}\|_\infty < 1$ which implies the following result.

Lemma C.2. *Assume that $\|\widehat{z}_{S^*c}\|_\infty < 1$. Then, any solution $\widetilde{\theta}$ of the minimization problem $\min_\theta \mathcal{C}(\theta)$ satisfies $\widetilde{\theta}_{S^{*c}} = 0$.*

Proof. Let $\widetilde{\theta}$ another solution. Then, it holds that

$$R_{T,n}(\widehat{\theta}) + \kappa\langle \widehat{z}, \widehat{\theta}\rangle = R_{T,n}(\widetilde{\theta}) + \kappa \sum_{j=1}^M \sum_{j'=1}^M |\widetilde{\theta}_{j,j'}|,$$

we deduce that

$$R_{T,n}(\widehat{\theta}) - \kappa\langle \widehat{z}, \widetilde{\theta} - \widehat{\theta}\rangle = R_{T,n}(\widetilde{\theta}) + \kappa\left(\sum_{j=1}^M \sum_{j'=1}^M |\widetilde{\theta}_{j,j'}| - \langle \widehat{z}, \widetilde{\theta}\rangle\right).$$

Since the pair $(\widehat{\theta}, \widehat{z})$ satisfies Equation (20), we have that

$$\kappa\widehat{z} = -\nabla R_{T,n}(\widehat{\theta}),$$

which leads to

$$R_{T,n}(\widehat{\theta}) - R_{T,n}(\widetilde{\theta}) + \langle \nabla R_{T,n}(\widehat{\theta}), \widetilde{\theta} - \widehat{\theta} \rangle = \kappa \left(\sum_{j=1}^{M} \sum_{j'=1}^{M} |\widetilde{\theta}_{j,j'}| - \langle \widehat{z}, \widetilde{\theta} \rangle \right).$$

Hence, from the above equation and the convexity of $R_{T,n}$ we deduce that

$$\kappa \left(\sum_{j=1}^{M} \sum_{j'=1}^{M} |\widetilde{\theta}_{j,j'}| - \langle \widehat{z}, \widetilde{\theta} \rangle \right) \leq 0.$$

Therefore, we obtain that

$$\sum_{j=1}^{M} \sum_{j'=1}^{M} |\widetilde{\theta}_{j,j'}| \leq \langle \widehat{z}, \widetilde{\theta} \rangle = \sum_{j=1}^{M} \sum_{j'=1}^{M} \widehat{z}_{j,j'} \widetilde{\theta}_{j,j'}.$$

Since $\|\widehat{z}_{S^{*c}}\|_\infty < 1$, if there exists $\tilde{\theta}_{j,j'} \neq 0$ for $(j, j') \in S^{*c}$ we get

$$\sum_{j=1}^{M} \sum_{j'=1}^{M} |\widetilde{\theta}_{j,j'}| < \sum_{j=1}^{M} \sum_{j'=1}^{M} |\widetilde{\theta}_{j,j'}|,$$

which leads us to a contradiction. Therefore $\widetilde{\theta}_{S^{*c}} = 0$. □

Now we show that for $\kappa \geq \dfrac{\log^4(nM^2)}{\sqrt{n}}$, we have $\|\widehat{z}_{S^{*c}}\|_\infty < 1$ on the event Ω_n. From Equation (22), we deduce that for each $j \in [M]$

$$\|\widehat{z}_{S^{*c}_{\theta_j}}\|_\infty \leq \|\mathbb{H}_{S^{*c}_{\theta_j}, S^*_{\theta_j}} \mathbb{H}^{-1}_{S^*_{\theta_j}, S^*_{\theta_j}}\|_\infty + \|\mathbb{H}_{S^{*c}_{\theta_j}, S^*_{\theta_j}} \mathbb{H}^{-1}_{S^*_{\theta_j}, S^*_{\theta_j}}\|_\infty \frac{2}{n_k \kappa} \|(Z_j)_{S^*_{\theta_j}}\|_\infty + \frac{2}{n_k \kappa} \|(Z_j)_{S^{*c}_{\theta_j}}\|_\infty.$$

From Assumption (MI), we get for some $\gamma \in (0, 1)$

$$\|\widehat{z}_{S^{*c}}\|_\infty \leq (1 - \gamma) \left(1 + \frac{2}{n_k \kappa} \|(Z_j)_{S^*_{\theta_j}}\|_\infty \right) + \frac{2}{n_k \kappa} \|(Z_j)_{S^{*c}_{\theta_j}}\|_\infty. \tag{23}$$

From Lemma C.1 we have with probability larger than $1 - \dfrac{CM}{n}$ on an event Ω_n that

$$\frac{1}{n_k} \|(Z_j)_{S^*_{\theta_j}}\|_\infty \leq \frac{C \log^3(nM^2)}{\sqrt{n}}, \quad \frac{1}{n_k} \|(Z_j)_{S^{*c}_{\theta_j}}\|_\infty \leq \frac{C \log^3(nM^2)}{\sqrt{n}}.$$

Hence, from Equation (23), for n large enough, we deduce that, with probability larger than on Ω_n,

$$\|\widehat{z}_{S^{*c}}\|_\infty < 1,$$

provided that $\dfrac{C \log^3(nM^2)}{\kappa \sqrt{n}} \to 0$ as $n \to +\infty$. Therefore, the choice $\kappa \geq \dfrac{\log^4(nM^2)}{\sqrt{n}}$ yields the desired result.

Third step. In the second step, we show for n large enough that on Ω_n, any solution of $\min_\theta \mathcal{C}(\theta)$ (with $\mathcal{C}$ given in (19)) is a solution of

$$\min_{\theta_{S^*}} \widetilde{R}_{T,n}(\theta_{S^*}) + \kappa \sum_{j=1}^{M} \sum_{j' \in S^*_{\theta_j}} |\theta_{j,j'}|.$$

In this step, we establish the following result.

Lemma C.3. *Let $\widehat{\theta}_{S^*}$ defined as*

$$\widehat{\theta}_{S^*} \in \operatorname*{argmin}_{\theta_{S^*}} \left\{ \widetilde{R}_{T,n}(\theta_{S^*}) + \kappa \sum_{j=1}^{M} \sum_{j' \in S^*_{\theta_j}} |\theta_{j,j'}| \right\}.$$

Under Assumption (ME), for $\kappa = \dfrac{\log^4(nM^2)}{\sqrt{n}}$, it holds that on Ω_n

$$\left\| \widehat{\theta}_{S^*} - \theta_{S^*} \right\|_\infty \leq \frac{C\Lambda_0 \max_j \sqrt{|S^*_{\theta_j}|} \log^4(nM^2)}{\sqrt{n}}.$$

Proof. From Equation (21), we get for each $j \in \{1, \dots, M\}$

$$\left\| \widehat{\theta}_{S^*_{\theta_j}} - \theta_{S^*_{\theta_j}} \right\|_\infty \leq \left\| \left(\frac{\mathbb{H}_{S^*_j, S^*_{\theta_j}}}{n_k} \right)^{-1} \frac{(Z_j)_{S^*_{\theta_j}}}{n_k} \right\|_\infty + \frac{\kappa}{2} \left\| \left(\frac{\mathbb{H}_{S^*_{\theta_j}, S^*_{\theta_j}}}{n_k} \right)^{-1} \operatorname{sign}((\widehat{\theta}_j)_{S^*_{\theta_j}}) \right\|_\infty.$$

Applying Lemma A.1, and C.1 together with Assumption (ME), we obtain

$$\left\| \widehat{\theta}_{S^*_{\theta_j}} - \theta^*_{S^*_{\theta_j}} \right\|_\infty \leq \Lambda_0 \sqrt{\left| S^*_{\theta_j} \right|} \left(\frac{C \log^3(nM^2)}{\sqrt{n}} + \kappa \right).$$

Therefore, the choice of $\kappa = \dfrac{\log^4(nM^2)}{\sqrt{n}}$ yields the desired result. □

Fourth step. We deduce from Lemma C.3 and Assumption 8 that

$$\operatorname{sign}(\widehat{\theta}_{S^*}) = \operatorname{sign}(\theta^*_{S^*}).$$

Therefore, from Equation (21), we deduce that on Ω_n for $\kappa = \dfrac{\log^4(nM^2)}{\sqrt{n}}$,

$$\theta_{S^*} \mapsto \min_{\theta_{S^*}} \widetilde{R}_{T,n}(\theta_{S^*}) + \kappa \sum_{j=1}^{M} \sum_{j' \in S^*_{\theta_j}} |\theta_{j,j'}|,$$

admits a unique minimizer $\widehat{\theta}_{S^*}$ which satisfies for each $j \in \{1, \dots, M\}$,

$$(\widehat{\theta}_j)_{S^*_{\theta_j}} = (\theta_j)_{S^*_{\theta_j}} + \mathbb{H}^{-1}_{S^*_{\theta_j}, S^*_{\theta_j}} (Z_j)_{S^*_{\theta_j}} - \frac{n_k \kappa}{2} \mathbb{H}^{-1}_{S^*_{\theta_j}, S^*_{\theta_j}} \operatorname{sign}((\theta^*{}_j)_{S^*_{\theta_j}}).$$

Hence, in view of Steps 2, with the choice of $\kappa = \dfrac{\log^4(nM^2)}{\sqrt{n}}$, we then have shown that there is a unique solution $\widehat{\theta}$ of $\min_\theta \mathcal{C}(\theta)$ which satisfies on Ω_n

$$\widehat{\theta}_{S^{*c}} = 0, \text{ and } \operatorname{sign}(\widehat{\theta}_{S^*}) = \operatorname{sign}(\theta^*_{S^*}),$$

and

$$\left\|\widehat{\theta}_{S^*} - \theta_{S^*}\right\|_\infty \leq \frac{C\Lambda_0 \max_j \sqrt{|S^*_{\theta_j}|}\log^4(nM^2)}{\sqrt{n}}.$$

Appendix D Proofs for the rate of convergence of ERMLR algorithm

We first establish a technical result in Section D.1, then rate of convergence of the ERMLR algorithm is given in Section D.2.

D.1 Technical result

We recall that the set Θ_n is defined as follows

$$\Theta_n := \left\{\theta = (\mu, A) \in \mathbb{R}_+^M \times \mathbb{R}_+^{M^2}, \mu_j \in \left[\frac{1}{n}, \log(n)\right], j \in [M], \|A\|_F \leq n\right\}.$$

We also introduce the set Π of conditional probabilities

$$\Pi := \left\{ \pi_{p,\theta} = \left(\frac{p_k \mathrm{e}^{F_{\theta_k}(\cdot)}}{\sum_{k'=1}^K p_{k'}\mathrm{e}^{F_{\theta_{k'}}(\cdot)}}\right)_{k\in[K]} : \theta = (\theta_1, \ldots, \theta_K) \in \Theta_n^K, \sum_{k=1}^K p_k = 1, \min_k p_k > \frac{p_0}{2} \right\}$$

The following result provides a bound on ℓ_1-distance between two elements of the set Π. It shows that this distance can be bounded by the distance between the corresponding parameters of the associated model.

Proposition D.1. *Let $\pi = \pi_{p,\theta}$ and $\pi' = \pi_{p',\theta'}$ two elements of Π. Grant Assumptions 3, 2 and 1, the following holds*

$$\mathbb{E}\left[\left\|\pi - \pi'\right\|_1\right] \leq \frac{K}{p_0}\|p - p'\|_1 + CK^2n^2\log(n)\left(\sqrt{M}\max_{k\in[K]}\|\mu_k - \mu'_k\|_1 + M\max_{k\in[K]}\|A_k - A'_k\|_F\right),$$

where C is a constant depending on T, μ_0, μ_1 and $\|h\|_\infty$.

Proof. Let us consider $\pi, \pi' \in \Pi$ with respective parameters (p, θ), and (p', θ'). We have that

$$\left\|\pi(\mathcal{T}) - \pi'(\mathcal{T})\right\|_1 \leq \left\|\pi(\mathcal{T}) - \pi_{p,\theta'}(\mathcal{T})\right\|_1 + \left\|\pi_{p,\theta'}(\mathcal{T}) - \pi'(\mathcal{T})\right\|_1. \tag{24}$$

Since for any k, j and $(x_1, \ldots, x_K)$,

$$\left|\frac{\partial \phi_k^p(x_1, \ldots, x_K)}{\partial p_j}\right| \leq \frac{1}{p_0},$$

we deduce by mean value inequality

$$\left\|\pi_{p,\theta'}(\mathcal{T}) - \pi'(\mathcal{T})\right\|_1 \leq \frac{K}{p_0}\left\|p - p'\right\|_1.$$

Besides for any k, j and p,

$$\left|\frac{\partial \phi_k^p(x_1,\ldots,x_K)}{\partial x_j}\right| \leq 1,$$

we also deduce

$$\left\|\pi(\mathcal{T}) - \boldsymbol{\pi}_{p,\theta'}(\mathcal{T})\right\|_1 \leq K \sum_{k=1}^{K} \left|F_{\theta_k}(\mathcal{T}) - F_{\theta'_k}(\mathcal{T})\right|.$$

Therefore, from Equation (24), we obtain

$$\mathbb{E}\left[\left\|\pi(\mathcal{T}) - \pi'(\mathcal{T})\right\|_1\right] \leq \frac{K}{p_0}\left\|p - p'\right\|_1 + K\sum_{k=1}^{K}\mathbb{E}\left[\left|F_{\theta_k}(\mathcal{T}) - F_{\theta'_k}(\mathcal{T})\right|\right].$$

Hence, it remains to bound the second term in the *r.h.s.* of the above inequality. Using Cauchy-Schwartz inequality, for each k, we have that

$$\begin{aligned}
&\mathbb{E}\left[\left|F_{\theta_k}(\mathcal{T}) - F_{\theta'_k}(\mathcal{T})\right|\right] \\
&= \mathbb{E}\left[\left|\sum_{j=1}^{M}\left(\int_0^T \log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\mathrm{d}N_j(t) - \int_0^T\left(\lambda_{j,\theta_k}(t) - \lambda_{j,\theta'_k}(t)\right)\mathrm{d}t\right)\right|\right] \\
&\leq \mathbb{E}\left[\left(\sum_{j=1}^{M}\int_0^T\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right|\mathrm{d}N_j(t)\right)^2\right]^{1/2} + \mathbb{E}\left[\sum_{j=1}^{M}\int_0^T\left|\lambda_{j,\theta_k}(t) - \lambda_{j,\theta'_k}(t)\right|\mathrm{d}t\right]. \qquad (25)
\end{aligned}$$

Now, we observe that

$$\left|\lambda_{j,\theta_k}(t) - \lambda_{j,\theta'_k}(t)\right| \leq |\mu_{k,j} - \mu'_{k,j}| + \|h\|_\infty \sum_{j'=1}^{M}|a_{k,j,j'} - a'_{k,j,j'}|N_{j'}(T).$$

Therefore, we deduce

$$\mathbb{E}\left[\sum_{j=1}^{M}\int_0^T\left|\lambda_{j,\theta_k}(t) - \lambda_{j,\theta'_k}(t)\right|\mathrm{d}t\right] \leq T\sum_{j=1}^{M}|\mu_{k,j} - \mu'_{k,j}| + T\|h\|_\infty\sum_{j'=1}^{M}\sum_{j=1}^{M}|a_{k,j,j'} - a'_{k,j,j'}|\mathbb{E}\left[N_{j'}(T)\right]. \qquad (26)$$

Now, we bound the first term in the *r.h.s.* of Equation (25). Using that $x \mapsto \log(1+x)$ is a Lipschitz function, we obtain:

$$\begin{aligned}
\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right| &\leq \left|\log\left(\frac{\mu_{k,j}}{\mu'_{k,j}}\right)\right| + \left|\frac{\lambda_{j,\theta_k}(t)}{\mu'_{k,j}} - \frac{\lambda_{j,\theta'_k}(t)}{\mu_{k,j}}\right| \\
&\leq n\left|\mu_{k,j} - \mu'_{k,j}\right| + n^2\left|\mu_{k,j}\lambda_{j,\theta_k}(t) - \mu'_{k,j}\lambda_{j,\theta'_k}(t)\right| \\
&\leq n\left|\mu_{k,j} - \mu'_{k,j}\right| + n^2\left(|\mu_{k,j} - \mu'_{k,j}|\lambda_{j,\theta'_k}(t) + \mu'_n\left|\lambda_{j,\theta_k}(t) - \lambda_{j,\theta'_k}(t)\right|\right) \\
&\leq n\left|\mu_{k,j} - \mu'_{k,j}\right| + n^2\Bigg(|\mu_{k,j} - \mu'_{k,j}|\lambda_{j,\theta'_k}(t) \\
&\quad + \log(n)\Big(|\mu'_{k,j} - \mu_{k,j}| + \|h\|_\infty\sum_{j'=1}^{M}N_{j'}(T)|a_{k,j,j'} - a'_{k,j,j'}|\Big)\Bigg). \qquad (27)
\end{aligned}$$

Besides, applying the Doob's decomposition for the processes $N_j, j \in [M]$, and the Cauchy-Schwartz's inequality, we get

$$\begin{aligned}\mathbb{E}\left[\left(\sum_{j=1}^{M}\int_0^T\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right|\mathrm{d}N_j(t)\right)^2\right] &\leq M\sum_{j=1}^{M}\mathbb{E}\left[\int_0^T\log^2\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\lambda^*_{Y,j}(t)\,\mathrm{d}t\right] \\ &+ M\sum_{j=1}^{M}\mathbb{E}\left[\left(\int_0^T\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right|\lambda^*_{Y,j}(t)\,\mathrm{d}t\right)^2\right] \quad (28)\end{aligned}$$

From Assumption 4, we have $\mathbb{E}\left[\left(\lambda^*_{Y,j}(t)\right)^2\right] < \infty$. Therefore, the first term in the *r.h.s.* in Equation (28) can be bounded as follows

$$\begin{aligned}\mathbb{E}\left[\int_0^T\log^2\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\lambda^*_{Y,j}(t)\,\mathrm{d}t\right] &\leq \int_0^T\mathbb{E}\left[\log^4\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right]^{1/2}\mathbb{E}\left[\left(\lambda^*_{Y,j}(t)\right)^2\right]^{1/2}\mathrm{d}t \\ &\leq CT\sup_{t\in[0,T]}\mathbb{E}\left[\log^4\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right]^{1/2}.\end{aligned}$$

Similarly, we obtain:

$$\begin{aligned}\mathbb{E}\left[\left(\int_0^T\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right|\lambda^*_{Y,j}(t)\,\mathrm{d}t\right)^2\right] &\leq T\mathbb{E}\left[\int_0^T\log^2\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\left(\lambda^*_{Y,j}(t)\right)^2\mathrm{d}t\right] \\ &\leq CT^2\sup_{t\in[0,T]}\mathbb{E}\left[\log^4\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right]^{1/2}.\end{aligned}$$

Then, by Assumption 3, from Equation (27) and Equation (28), we get

$$\begin{aligned}&\mathbb{E}\left[\left(\sum_{j=1}^{M}\int_0^T\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right|\mathrm{d}N_j(t)\right)^2\right] \\ &\leq CT^2M\sum_{j=1}^{M}\sup_{t\in[0;T]}\mathbb{E}\left[|\mu_{k,j}-\mu'_{k,j}|^4\left(n+n^2\lambda_{j,\theta'_k}(t)+n^2\log(n)\right)^4\right. \\ &\quad\left.+n^8\log(n)^4\|h\|_\infty^4\left(\sum_{j'=1}^{M}N_{j'}(T)|a_{k,j,j'}-a'_{k,j,j'}|\right)^4\right]^{1/2} \\ &\leq CT^2M\sum_{j=1}^{M}\left(\left[n^4\sup_{t\in[0;T]}\mathbb{E}\left[\left(\lambda_{j,\theta'_k}(t)\right)^4\right]^{1/2}+n^2+n^4\log(n)^2\right]|\mu_{k,j}-\mu'_{k,j}|^2\right. \\ &\quad\left.+Cn^4\log(n)^2\mathbb{E}\left[\left(\sum_{j'=1}^{M}N_{j'}(T)|a_{k,j,j'}-a'_{k,j,j'}|\right)^4\right]^{1/2}\right)\end{aligned}$$

where C is a constant depending on μ_0, μ_1 and $\|h\|_\infty$. In view of Assumption 4 $\mathbb{E}\left[(\lambda_{j,\theta'_k}(t))^4\right] \leq C$. Therefore, from the above equation, and Cauchy Schwartz's inequality, we deduce

$$E\left[\left(\sum_{j=1}^{M}\int_0^T\left|\log\left(\frac{\lambda_{j,\theta_k}(t)}{\lambda_{j,\theta'_k}(t)}\right)\right|\,\mathrm{d}N_j(t)\right)^2\right] \leq CT^2M\left(n^4+n^2+n^4\log(n)^2\right)\sum_{j=1}^{M}|\mu_{k,j}-\mu'_{k,j}|^2$$
$$+CT^2Mn^4\log(n)^2\mathbb{E}\left[\left(\sum_{j'=1}^{M}N_{j'}(T)^2\right)^2\right]^{1/2}\sum_{j=1}^{M}\sum_{j'=1}^{M}|a_{k,j,j'}-a'_{k,j,j'}|^2.$$

From Assumption 4 we have that $\mathbb{E}\left[\left(\sum_{j'=1}^{M}N_{j'}(T)^2\right)^2\right] \leq CM^2$. Thus, gathering Equations (25) and (26), it comes

$$\begin{aligned}\mathbb{E}[\|\pi-\pi'\|_1] \;\leq\; & \frac{K}{p_0}\|p-p'\|_1 \\ & +KC\sum_{k=1}^{K}\left(\sum_{j=1}^{M}|\mu_{k,j}-\mu'_{k,j}|+\sum_{j=1}^{M}\sum_{j'=1}^{M}|a_{k,j,j'}-a'_{k,j,j'}|\right) \\ & +KCn^2\log(n)\sum_{k=1}^{K}\left(M\sum_{j=1}^{M}|\mu_{k,j}-\mu'_{k,j}|^2+M^2\sum_{j=1}^{M}\sum_{j'=1}^{M}|a_{k,j,j'}-a'_{k,j,j'}|^2\right)^{1/2}\end{aligned}$$

with C depending on μ_0, μ_1, $\|h\|_\infty$ and T. Finally, using that $\|x\|_2 \leq \|x\|_1 \leq \sqrt{d}\|x\|_2$ for $x \in \mathbb{R}^d$, we obtain

$$\begin{aligned}\mathbb{E}\big[\|\pi-\pi'\|_1\big] \;\leq\; & \frac{K}{p_0}\|p-p'\|_1 \\ & +K^2Cn^2\log(n)^2\max_{k\in[K]}\left(\sum_{j=1}^{M}|\mu_{k,j}-\mu'_{k,j}|+\sum_{j=1}^{M}\sum_{j'=1}^{M}|a_{k,j,j'}-a'_{k,j,j'}|\right) \\ & +K^2Cn^2\log(n)^2\max_{k\in[K]}\left(\sqrt{M}\sum_{j=1}^{M}|\mu_{k,j}-\mu'_{k,j}|+M\|A_k-A'_k\|_F\right)\end{aligned}$$

thus

$$\begin{aligned}\mathbb{E}\big[\|\pi-\pi'\|_1\big] \;\leq\; & \frac{K}{p_0}\|p-p'\|_1+K^2Cn^2\log(n)\sqrt{M}\max_{k\in[K]}\|\mu_k-\mu'_k\|_1 \\ & +K^2CMn^2\log(n)\max_{k\in[K]}\|A_k-A'_k\|_F.\end{aligned}$$

Finally, combining the above equation, Equations (25) and (26) yields the desired result.

□

D.2 Proof of Theorem 2

We begin this section by a lemma that provides a bound on the ε-covering number of the set $\hat{\Theta}$ defined in Equation 7.

Lemma D.1. *Let $\varepsilon > 0$. There exists an ε-net $\mathcal{M}_\varepsilon \subset \hat{\Theta}$ with*

$$|\mathcal{M}_\varepsilon| \leq \left(\frac{(\log(n) - 1/n)M}{\varepsilon}\right)^{MK} \left(\frac{3n}{\varepsilon}\right)^{\sum_k \widehat{S}_k}.$$

In particular, for all $(\mu, A) \in \widehat{\Theta}$ there exists $(\mu_\varepsilon, A_\varepsilon) \in \mathcal{M}_\varepsilon$ s.t. $\max_{k\in[K]} \|\mu_k - \mu_{k,\varepsilon}\|_1 \leq \varepsilon$ and $\|A_k - A_{k,\varepsilon}\|_F \leq \varepsilon$.

Proof of Lemma D.1. First, we observe that the set

$$\left\{\frac{1}{n} + k\frac{(\log(n) - 1/n)}{\lceil \frac{M(\log(n)-1/n)}{\varepsilon} \rceil},\ k \in \left\{1, \ldots, \frac{M(\log(n) - 1/n)}{\varepsilon} - 1\right\}\right\}$$

is and ε/M-cover of the interval $[1/n \log(n)]$. Therefore, we deduce that there exists $\mathcal{M}_{\varepsilon,\mu}$ an ε-cover of $\{\mu \in \mathbb{R}^M,\ s.t.\ \mu \in \Theta_n\}$ for $\|\cdot\|_1$, such that

$$\mathcal{M}_{\varepsilon,\mu} \leq \left(\frac{(\log(n) - 1/n)M}{\varepsilon}\right)^M. \tag{29}$$

Let $k \in [K]$. For $\varepsilon > 0$, the covering number of the Euclidean ball centered in 0 and with radius n in $\mathbb{R}^{\widehat{S}_k}$, satisfies

$$\mathcal{N}(\varepsilon, \bar{\mathcal{B}}(0, n), \|.\|_2) \leq \left(\frac{3n}{\varepsilon}\right)^{\widehat{S}_k}.$$

Hence, we deduce that there exists $\mathcal{M}_{\varepsilon,A,k}$ an ε-cover of $\{A \in \Theta_n,\ s.t.\ \mathrm{supp}(A) = \widehat{S}_k\}$, for $\|\cdot\|_F$, such that

$$\mathcal{M}_{\varepsilon,A,k} \leq \left(\frac{3n}{\varepsilon}\right)^{\widehat{S}_k}. \tag{30}$$

From Equation (29) and (30) we obtain the desired result. □

Proof of Theorem 2. We first recall that the construction of the `ERMLR` algorithm is based on a dataset $\mathcal{D}_n = \{(\mathcal{T}_T^{(i)}, Y^{(i)}), i = 1, \ldots, 2n\}$ of size $2n$ which is split into two independent dataset of same size n that are denoted respectively $\mathcal{D}_n^{(1)}$ and $\mathcal{D}_n^{(2)}$.

Based on the first sample $\mathcal{D}_n^{(1)}$, we estimate the vector of weights p^* by its empirical frequencies $\hat{p}$. Hence for each k, we have

$$\hat{p}_k = \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{\{Y^{(i)}=k\}}$$

Then, based on sample $\mathcal{D}_n^{(2)}$, we build the estimator $\widehat{S} := \left(\widehat{S}_1, \ldots, \widehat{S}_K\right)$ as described in Section 3.1. Besides, we also build the estimator of the vector of score function $\hat{f} = f_{\hat{\theta}^R}$, and $\hat{g}$ its associated classifier. Since $\mathcal{D}_n^{(1)}$ and $\mathcal{D}_n^{(2)}$ are independent, we have that $\hat{p}$ is independent on $\widehat{f}$ and $\widehat{g}$.

Let us introduce the set $\mathcal{A} = \left\{\widehat{p} :\ \min(\widehat{p}) \geq \frac{p_0}{2}\right\}$. Note that on $\mathcal{A}^c$ we have

$$|\min(p^*) - \min(\widehat{p})| \geq \frac{p_0}{2},$$

which implies that there exists $k \in \mathcal{Y}$ s.t. $|p_k^* - \widehat{p}_k| \geq \frac{p_0}{2}$. Thus, using Hoeffding's inequality we get

$$\begin{aligned}\mathbb{P}(\mathcal{A}^c) &\leq \sum_{k=1}^K \mathbb{P}\left(|p_k^* - \widehat{p}_k| \geq \frac{p_0}{2}\right) \\ &\leq 2K\mathrm{e}^{-np_0^2/2}.\end{aligned} \tag{31}$$

Now, let us work on $\Omega = \mathcal{A} \bigcap \{\hat{S} = S^*\}$, and denote

$$\Delta_n := \sum_{k=1}^{K} (\widehat{p}_k - p_k^*)^2, \tag{32}$$

which is a random variable independent from $\mathcal{D}_n^{(2)}$. We also recall that for each $\theta \in \hat{\Theta}$, the score function f_θ is defined as follows

$$f_\theta(\mathcal{T}_T) = 2\pi_{k,\widehat{p},\theta}(\mathcal{T}_T) - 1, \ k \in [K].$$

We introduce

$$\tilde{\theta} = \underset{\theta \in \hat{\Theta}}{\operatorname{argmin}} \, \mathcal{R}_2(f_\theta).$$

The oracle counterpart of $\hat{f}$. Our aim is to control

$$\mathbb{E}\left[\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)\right] = \mathbb{E}\left[\left(\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)\right)\mathbb{1}_{\{\Omega\}}\right] + \mathbb{E}\left[\left(\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)\right)\mathbb{1}_{\{\Omega^c\}}\right]. \tag{33}$$

Since for each $\theta \in \hat{\Theta}$ defined by (7), $\mathcal{R}_2(f_\theta)$ is bounded, from Theorem 1, and Equation (31), we deduce that

$$\mathbb{E}[\left(\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)\right)\mathbb{1}_{\{\Omega^c\}}] \leq C\mathbb{P}\left(\Omega^c\right) \leq C\left(\frac{1}{n} + \exp\left(-np_0^2/2\right)\right). \tag{34}$$

Therefore, it remains to bound the first term in the *r.h.s.* of Equation (33). Hence, we work on the set Ω. We consider the following decomposition

$$\mathcal{R}_2(\hat{f}) - \mathcal{R}_2(f^*) = \left(\mathcal{R}_2(\hat{f}) - \mathcal{R}_2(f_{\tilde{\theta}})\right) + \left(\mathcal{R}_2(f_{\tilde{\theta}}) - \mathcal{R}_2(f^*)\right) \tag{35}$$

In a first step, we control the second term in the *r.h.s.* of the above equation. For n large enough, we observe that on Ω, $\theta^* \in \hat{\Theta}$. Therefore, from the definition of $\tilde{\theta}$, we deduce

$$\begin{aligned}
\mathcal{R}_2(f_{\tilde{\theta}}) - \mathcal{R}_2(f^*) &= \mathcal{R}_2(f_{\tilde{\theta}}) - \mathcal{R}_2(f_{\theta^*}) + \mathcal{R}_2(f_{\theta^*}) - \mathcal{R}_2(f^*) \\
&\leq \mathcal{R}_2(f_{\theta^*}) - \mathcal{R}_2(f^*).
\end{aligned}$$

Then on Ω, we deduce from the mean value theorem that

$$\mathcal{R}_2(f_{\tilde{\theta}}) - \mathcal{R}_2(f^*) \leq \mathcal{R}_2(f_{\theta^*}) - \mathcal{R}_2(f^*) \leq C\Delta_n, \tag{36}$$

with Δ_n given in Equation (32). Since, $\mathbb{E}\left[\Delta_n\right] \leq \dfrac{C}{n}$, from Equation (35), we deduce that

$$\mathbb{E}\left[\left(\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)\right)\mathbb{1}_{\{\Omega\}}\right] \leq \mathbb{E}\left[\mathcal{R}_2(\hat{f}) - \mathcal{R}_2(f_{\tilde{\theta}})\mathbb{1}_{\{\Omega\}}\right] + \frac{C}{n}. \tag{37}$$

Now, we focus on the first term in the r.h.s. of Equation (35). We denote

$$D_f := \mathcal{R}_2(f) - \mathcal{R}_2(f_{\tilde{\theta}}), \quad \text{and} \quad \widehat{D}_f := \widehat{\mathcal{R}}_2(f) - \widehat{\mathcal{R}}_2(f_{\tilde{\theta}}).$$

And we want to control $\mathbb{E}[D_{\hat{f}}]$. By Lemma D.1, there exists a subset $\mathcal{M}_\varepsilon \subset \hat{\Theta}$ such that for $\hat{\theta}^R = \left(\hat{\mu}, \hat{A}\right)$, there exists $\theta_\varepsilon = (\mu_\varepsilon, A_\varepsilon) \in \mathcal{M}_\varepsilon$ satisfying

$$\max_{k \in [K]} \|\mu_{k,\varepsilon} - \hat{\mu}_k\|_1 \leq \varepsilon \quad \text{and} \quad \max_{k \in [K]} \|A_{k,\varepsilon} - \hat{A}_k\|_F \leq \varepsilon.$$

Then, the following decomposition holds

$$\begin{aligned} D_{\widehat{f}} &\leq D_{\widehat{f}} - 2\widehat{D}_{\widehat{f}} \\ &= (D_{\widehat{f}} - D_{f_{\theta_\varepsilon}}) + (2\widehat{D}_{f_{\theta_\varepsilon}} - 2\widehat{D}_{\widehat{f}}) + (D_{f_{\theta_\varepsilon}} - 2\widehat{D}_{f_{\theta_\varepsilon}}) \\ &=: T_1 + T_2 + T_3. \end{aligned}$$

Applying Proposition D.1 with $\varepsilon = 1/(n^3 M \log(n))$ we get

$$\mathbb{E}\left[T_i\right] \leq \frac{C}{n}, \quad \text{for } i = 1, 2.$$

Besides,

$$T_3 \leq \max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}).$$

Therefore, gathering Equation (33), (34), (36), and (37), we deduce that

$$\mathbb{E}[\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)] \leq \mathbb{E}\left[\max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}) \mathbb{1}_{\{\Omega\}}\right] + \frac{C}{n}. \tag{38}$$

To finish the proof, it remains to control the first term in the *r.h.s.* of Inequality (38). Conditional on $\mathcal{D}_n^{(1)}$, we have that

$$\mathbb{E}\left[\max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}) \mathbb{1}_{\{\Omega\}} | \mathcal{D}_n^{(1)}\right] = \mathbb{1}_{\{\mathcal{A}\}} \mathbb{E}\left[\max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}) \mathbb{1}_{\{\hat{S} = S^*\}} | \mathcal{D}_n^{(1)}\right].$$

Note that On the set $\{\hat{S} = S^*\}$, the set $\mathcal{M}_\varepsilon$ is an ε-net of the *deterministic* set

$$\tilde{\Theta} = \left\{\theta = (\theta_1, \ldots, \theta_K) \in \Theta_n^K, \ \operatorname{supp}(A_k) = S_k^*\right\},$$

and then is also deterministic. Besides, from Lemma D.1, we deduce that for $\varepsilon = \dfrac{1}{n^3 M \log(n)}$

$$\log\left(|\mathcal{M}_\varepsilon|\right) \leq CK\left(M + s^*\right) \frac{\log(nM)}{n}.$$

Furthermore, for $u > 0$ conditional on $\mathcal{D}_n^{(1)}$, it holds that

$$\begin{aligned} \mathbb{E}\left[\max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}) \mathbb{1}_{\{\hat{S} = S^*\}}\right] &\leq u + \int_u^\infty \mathbb{P}\left(\max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}) \mathbb{1}_{\{\hat{S} = S^*\}} \geq t\right) \mathrm{d}t \\ &\leq u + \int_u^\infty \mathbb{P}\left(\max_{\theta \in \mathcal{M}_\varepsilon} (D_{f_\theta} - 2\widehat{D}_{f_\theta}) \geq t\right) \mathrm{d}t. \end{aligned} \tag{39}$$

Now, we have to bound the last term in the above equation. Let $\theta \in \mathcal{M}_\varepsilon$, and $f := f_\theta$. Let us introduce the least squares function

$$\ell_f(Z, \mathcal{T}) := \sum_{k=1}^K (Z_k - f^k(\mathcal{T}))^2.$$

Since for each $\theta \in \tilde{\Theta}$, f_θ is uniformly bounded by 1, we get from Bernstein's inequality that, conditionally on $\mathcal{D}_n^{(1)}$, for $t \geq 0$

$$\begin{aligned} \mathbb{P}\left(D_f - 2\widehat{D}_f \geq t\right) &\leq \mathbb{P}\left(2(D_f - \widehat{D}_f) \geq t + D_f\right) \\ &\leq \exp\left(\frac{-n(t + D_f)^2/8}{B_f + (t + D_f)4K/3}\right), \end{aligned} \tag{40}$$

with

$$B_f := \mathbb{E}\left[\left(\ell_f(Z,\mathcal{T}) - \ell_{f_{\tilde{\theta}}}(Z,\mathcal{T})\right)^2\right].$$

From the Cauchy-Schwartz inequality, we observe that conditionally on $\mathcal{D}_n^{(1)}$

$$\begin{aligned}\mathbb{E}\left[(\ell_f(Z,\mathcal{T}) - \ell_{f^*}(Z,\mathcal{T}))^2\right] &\leq C_K \sum_{k=1}^{K} \mathbb{E}\left[(f^k(\mathcal{T}) - f^{*k}(\mathcal{T}))^2\right] \\ &= C_K\left(\mathcal{R}_2(f) - \mathcal{R}_2(f^*)\right).\end{aligned}$$

Thus, since

$$B_f \leq 2\mathbb{E}\left[\left(\ell_f(Z,\mathcal{T}) - \ell_{f^*}(Z,\mathcal{T})\right)^2\right] + 2\mathbb{E}\left[\left(\ell_{f_{\tilde{\theta}}}(Z,\mathcal{T}) - \ell_{f^*}(Z,\mathcal{T})\right)^2\right],$$

we deduce that

$$B_f \leq C_K\left(\mathcal{R}_2(f) - \mathcal{R}_2(f^*) + \mathcal{R}_2(f_{\tilde{\theta}}) - \mathcal{R}_2(f^*)\right).$$

Then, as $\mathcal{R}_2(f) - \mathcal{R}_2(f^*) = \mathcal{R}_2(f) - \mathcal{R}_2(f_{\tilde{\theta}}) + \mathcal{R}_2(f_{\tilde{\theta}}) - \mathcal{R}_2(f^*)$, on the event $\mathcal{A}$ and conditionally on $\mathcal{D}_n^{(1)}$, we deduce from the above inequality and Equation (36) that

$$B_f \leq C_K\left(D_f + \Delta_n\right).$$

Hence, from Inequality (40), we get for $t \geq \Delta_n$,

$$\mathbb{P}\left(D_f - 2\widehat{D}_f \geq t\right) \leq \exp\left(-C_K n t\right),$$

which leads to

$$\mathbb{P}\left(\max_{\theta\in\mathcal{M}_\varepsilon}(D_f - 2\widehat{D}_f) \geq t\right) \leq |\mathcal{M}_\varepsilon| \exp\left(-C_K n t\right).$$

In view of Equation (39), we then obtain that, conditionally on $\mathcal{D}_n^{(1)}$,

$$\mathbb{E}\left[\max_{\theta\in\mathcal{M}_\varepsilon}(D_{f_\theta} - 2\widehat{D}_{f_\theta})\mathbb{1}_{\{\Omega\}}|\mathcal{D}_n^{(1)}\right] \leq \max\left(\Delta_n, \frac{C\log(|\mathcal{M}_\varepsilon|)}{n}\right) + \int_{C\log(\mathcal{M}_\varepsilon)/n}^{+\infty} |\mathcal{M}_\varepsilon| \exp(-Cnt)\,\mathrm{d}t.$$

As before, we use that $\mathbb{E}\left[\Delta_n\right] \leq C/n$, and we deduce from the above inequality by integrating over $\mathcal{D}_n^{(1)}$ that

$$\mathbb{E}\left[\max_{\theta\in\mathcal{M}_\varepsilon}(D_{f_\theta} - 2\widehat{D}_{f_\theta})\mathbb{1}_{\{\Omega\}}\right] \leq \frac{C\log(|\mathcal{M}_\varepsilon|)}{n}.$$

Since for $\varepsilon = 1/(\log(n)n^3M)$ we have that $\log(|\mathcal{M}_\varepsilon|) \leq C(M+s^*)\log(nM)$, we obtain from the above inequality and Equation (38) that

$$\mathbb{E}[\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)] \leq C\frac{(M+s^*)\log(nM)}{n}.$$

From the above inequality, we get the desired by applying the Zhang's lemma

$$\mathbb{E}[\mathcal{R}(\hat{g}) - \mathcal{R}(g^*)] \leq \frac{1}{\sqrt{2}}\left(\mathbb{E}[\mathcal{R}_2(\widehat{f}) - \mathcal{R}_2(f^*)]\right)^{1/2}.$$

□

References

Abramovich, F. and Grinshtein, V. (2018). High-dimensional classification by sparse logistic regression. *IEEE Transactions on Information Theory*, 65(5):3068–3079.

Bacry, E., Bompaire, M., Deegan, P., Gaïffas, S., and Poulsen, S. V. (2018). tick: a python library for statistical learning, with an emphasis on hawkes processes and time-dependent models. *Journal of Machine Learning Research*, 18(214):1–5.

Bacry, E., Bompaire, M., Gaïffas, S., and Muzy, J.-F. (2020). Sparse and low-rank multivariate hawkes processes. *Journal of Machine Learning Research*, 21(50):1–32.

Bacry, E., Mastromatteo, I., and Muzy, J.-F. (2015). Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005.

Baouan, A., Bismuth, E., Bohbot, A., Coustou, S., Lacome, M., and Rosenbaum, M. (2022). What should clubs monitor to predict future value of football players. *arXiv preprint arXiv:2212.11041*.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.

Bonnet, A., Dion-Blanc, C., Gindraud, F., and Lemler, S. (2022). Neuronal network inference and membrane potential model using multivariate hawkes processes. *Journal of Neuroscience Methods*, 372:109550.

Bonnet, A., Martinez Herrera, M., and Sangnier, M. (2023). Inference of multivariate exponential hawkes processes with inhibition and application to neuronal activity. *Statistics and Computing*, 33(4):91.

Bühlmann, P. and Van De Geer, S. (2011). *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media.

Cai, B., Zhang, J., and Guan, Y. (2024). Latent network structure learning from high-dimensional multivariate point processes. *Journal of the American Statistical Association*, 119(545):95–108.

Carstensen, L., Sandelin, A., Winther, O., and Hansen, N. (2010). Multivariate hawkes process models of the occurrence of regulatory elements. *BMC bioinformatics*, 11:1–19.

Chen, J. and Chen, Z. (2008). Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771.

Chen, S. S., Donoho, D. L., and Saunders, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61.

Chzhen, E., Giraud, C., and Stoltz, G. (2023). Parameter-free projected gradient descent. *arXiv preprint arXiv:2305.19605*.

Chzhen, E., Hebiri, M., and Salmon, J. (2019). On Lasso refitting strategies. *Bernoulli*, 25(4A):3175–3200.

Daley, D. and Vere-Jones, D. (2003). Basic properties of the poisson process. *An Introduction to the Theory of Point Processes: Volume I: Elementary Theory and Methods*, pages 19–40.

Denis, C., Dion-Blanc, C., Lacoste, R. E., Sansonnet, L., and Bas, Y. (2024). Bats monitoring: A classification procedure of bats behaviors based on hawkes processes. *Journal of the Royal Statistical Society Series C: Applied Statistics*.

Denis, C., Dion-Blanc, C., and Sansonnet, L. (2022). Multiclass classification for hawkes processes. In *Uncertainty in Artificial Intelligence*, pages 539–547. PMLR.

Donnet, S., Rivoirard, V., and Rousseau, J. (2020). Nonparametric bayesian estimation for multivariate hawkes processes. *The Annals of statistics*, 48(5):2698–2727.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Eichler, M., Dahlhaus, R., and Dueck, J. (2017). Graphical modeling for multivariate hawkes processes with nonparametric link functions. *Journal of Time Series Analysis*, 38(2):225–242.

Embrechts, P., Liniger, T., and Lin, L. (2011). Multivariate hawkes processes: an application to financial data. *Journal of Applied Probability*, 48(A):367–378.

Gupta, M. R., Bengio, S., and Weston, J. (2014). Training highly multiclass classifiers. *The Journal of Machine Learning Research*, 15(1):1461–1492.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical Learning with Sparsity The Lasso and Generalizations*. Chapman & Hall/CRC.

Hawkes, A. (1971). Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90.

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., and Muller, P.-A. (2019). Deep learning for time series classification: a review. *Data mining and knowledge discovery*, 33(4):917–963.

Kim, S., Putrino, D., Ghosh, ., and Brown, E. (2011). A granger causality measure for point process models of ensemble neural spiking activity. *PLoS computational biology*, 7(3):e1001110.

Lambert, R., Tuleau-Malot, C., Bessaih, T., Rivoirard, V., Bouret, Y., Leresche, N., and Reynaud-Bouret, P. (2018). Reconstructing the functional connectivity of multiple spike trains using Hawkes models. *Journal of Neuroscience Methods*, 297:9–21.

Leblanc, T. (2024). Exponential moments for hawkes processes under minimal assumptions. *hal-04527359*.

Lotz, A. (2024). A sparsity test for multivariate hawkes processes.

Lukasik, M., Srijith, P., Vu, D., Bontcheva, K., Zubiaga, A., and Cohn, T. (2016). Hawkes processes for continuous time sequence classification: an application to rumour stance classification in twitter. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 393–398.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the Lasso. *The Annals of Statistics*, 34(3):1436–1462.

Michael W. Ferry, Philip E. Gill, E. W. and Zhang, M. (2023). A class of projected-search methods for bound-constrained optimization. *Optimization Methods and Software*, 0(0):1–30.

Mohler, G., Short, M., Brantingham, P., Schoenberg, F., and Tita, G. (2011). Self-exciting point process modeling of crime. *Journal of the american statistical association*, 106:100–108.

Møller, J. and Rasmussen, J. G. (2005). Perfect simulation of hawkes processes. *Advances in applied probability*, 37(3):629–646.

Nicvert, L., Donnet, S., Keith, M., Peel, M., Somers, M., Swanepoel, L., Venter, J., Fritz, H., and Dray, S. (2024). Using the multivariate h awkes process to study interactions between multiple species from camera trap data. *Ecology*, page e4237.

Ogata, Y. (1988). Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, 83(401):9–27.

Reynaud-Bouret, P. and Schbath, S. (2010). Adaptive estimation for hawkes processes; application to genome analysis. *The Annals of Statistics*, 38(5):2781–2822.

Reynaud-Bouret, P., Tuleau-Malot, C., Rivoirard, V., and Grammont, F. (2013). Spike trains as (in) homogeneous poisson processes or hawkes processes: non-parametric adaptive estimation and goodness-of-fit tests. *Journal of Mathematical Neuroscience*.

Spaziani, S., Girardeau, G., Bethus, I., and Reynaud-Bouret, P. (2023). Heterogeneous multiscale multivariate autoregressive model: Existence, sparse estimation and application to functional connectivity in neuroscience. *Annals of Statistics*.

Sulem, D., Rivoirard, V., and Rousseau, J. (2024). Bayesian estimation of nonlinear Hawkes processes. *Bernoulli*, 30(2):1257–1286.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

Tibshirani, R. and Wasserman, L. (2017). Sparsity, the Lasso, and Friends. Technical report, Carnegie Mellon University.

Tondulkar, R., Dubey, M., Srijith, P., and Lukasik, M. (2022). Hawkes process classification through discriminative modeling of text. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Tropp, J. A. (2006). Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE transactions on information theory*, 52(3):1030–1051.

Vershynin, R. (2018). *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Wainwright, M. J. (2009). Sharp thresholds for high-dimensional and noisy sparsity recovery using l1 constrained quadratic programming (lasso). *IEEE transactions on information theory*, 55(5):2183–2202.

Wang, B., Zhang, H., Ma, Z., and Chen, W. (2023). Convergence of adagrad for non-convex objectives: Simple proofs and relaxed assumptions. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 161–190. PMLR.

Ward, R., Wu, X., and Bottou, L. (2020). Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *Journal of Machine Learning Research*, 21(219):1–30.

Zhang, R., Walder, C., Rizoiu, M.-A., and Xie, L. (2018). Efficient non-parametric bayesian hawkes processes. *arXiv preprint arXiv:1810.03730*.

Zhang, T. (2004). Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32(1):56–85.

Zhou, K., Zha, H., and Song, L. (2013). Learning triggering kernels for multi-dimensional hawkes processes. In *International Conference on Machine Learning*, pages 1301–1309.