Online-to-PAC generalization bounds under graph-mixing dependencies

Baptiste Abélès **Eugenio Clerico** **Gergely Neu**
Universitat Pompeu Fabra, Barcelona, Spain

Abstract

Traditional generalization results in statistical learning require a training data set made of independently drawn examples. Most of the recent efforts to relax this independence assumption have considered either purely temporal (mixing) dependencies, or graph-dependencies, where non-adjacent vertices correspond to independent random variables. Both approaches have their own limitations, the former requiring a temporal ordered structure, and the latter lacking a way to quantify the strength of inter-dependencies. In this work, we bridge these two lines of work by proposing a framework where dependencies decay with graph distance. We derive generalization bounds leveraging the online-to-PAC framework, by deriving a concentration result and introducing an online learning framework incorporating the graph structure. The resulting high-probability generalization guarantees depend on both the mixing rate and the graph's chromatic number.

1 INTRODUCTION

Consider the problem of predicting house prices based on data collected from a variety of locations. The value does not only depend on factors like home size, age, and amenities, but is also influenced by the neighborhood. In the language of probability theory, this can be modeled with a set of *dependent* random variables, with prices of neighboring houses showing positive correlation that decays with distance. Similar dependencies occur between users' opinions on social networks, where connected members are more likely to share similar views (Montanari and Saberi, 2010). In this paper, we study the generalization ability of learning algorithms trained on such correlated data sets, where dependencies are encoded in graph structures.

In machine learning, a model's gap in accuracy on training data and new, previously unseen, inputs is known as *generalization error*. Controlling this quantity offers theoretical guarantees on the typical performance on future data, reflecting the algorithm's ability to infer patterns (Shalev-Shwartz and Ben-David, 2014). In the past decades, a vast body of literature on this area has emerged, developing tools such as Rademacher complexity, VC dimension, uniform stability, and PAC-Bayesian inequalities (Vapnik, 2000; Bousquet and Elisseeff, 2002; Bousquet et al., 2004; Alquier, 2024). Yet, the great majority of current analyses consider training data sets made of independent and identically distributed (i.i.d.) examples, a strong requirement unrealistic for many applications (e.g., traffic forecasting (Yu et al., 2018), stock price prediction (Ariyo et al., 2014), or the examples above).

Recently, interest in statistical learning frameworks accounting for data correlations has surged. A major research line models these dependencies via mixing assumptions (see Bradley 2005 for several notions of mixing, such as α-, β-, ϕ-, and ψ-mixing), which control how quickly the influence between random variables decays as the (temporal, spacial, etc.) distance between them grows. This setting provide a quantitative measure of the dependencies among the data points, but has the major drawback of requiring data to have a well defined ordered structure. An alternative common framework takes a more qualitative approach, where the dependencies are captured by a *dependency graph* that assigns an edge to any pair of vertices whose associated data are dependent. This approach can encode correlations among non-ordered data, but leads to loose results when the actual dependencies are weak (Janson, 2004). In this work, we propose combining the mixing and graph-based perspectives to tackle situations where the strength of the dependencies is somehow known, yet the data lack an

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

ordered structure. In this *graph-mixing* scenario, the correlations decay as the graph distance increases.

To prove our generalization results, we follow an algorithmic approach that derives guarantees for statistical learning using tools for *regret analysis* in online learning. Online learning is a framework that deals with sequential decision problems, where a *learner* (a.k.a. *player*) interacts with an evolving environment. The learner's goal is to select actions over time to minimize the regret (a quantity comparing the player's actions to the best fixed action in hindsight). We refer to Cesa-Bianchi and Lugosi (2006) and Orabona (2019) for thorough introductions to the subject. A recent line of research (commonly called *algorithmic statistics*) has explored unconventional ways to tackle classical statistical challenges, drawing connections with online learning. This approach has successfully addressed problems such as hypothesis testing (Grünwald et al., 2024), decision making (Foster et al., 2021), mean estimation (Orabona and Jun, 2024), and martingale concentration (Rakhlin and Sridharan, 2017). This strategy has also been applied to study generalization in statistical learning, for instance leading to Rademacher (Kakade et al., 2008) and PAC-Bayesian bounds (Jang et al., 2023; Lugosi and Neu, 2023; Abeles et al., 2024; Chatterjee et al., 2024). Most of these methods split the problem into two parts, a worst-case one that is dealt with online regret analysis, and a probabilistic one. When studying generalization with i.i.d. data (as in Jang et al. 2023 and Lugosi and Neu 2023), the probabilistic part reduces to upper bounding the deviations of a martingale. This martingale structure is lost if dependencies among the training data are present. Abeles et al. (2024) addressed this issue for mixing processes, imposing a delayed-feedback constraint on the player's strategy in the framework of Lugosi and Neu (2023), which allowed them to decompose the overall error into the regret of a delayed online learning strategy and the fluctuations of a stationary mixing process. Here, we follow a similar approach to tackle more complex graph-mixing dependencies, by introducing a novel online learning framework on graphs, and studying the concentration of what we name *graph-mixing processes.*

Several works established generalization guarantees under mixing assumptions. Mohri and Rostamizadeh (2007, 2010) obtained generalization bounds for uniformly stable algorithms under stationary ϕ- and β-mixing, leveraging concentration tools from Yu (1994) and Kontorovich and Ramanan (2008). Fu et al. (2023) tightened these results, adapting stability techniques from Feldman and Vondrak (2019) and Bousquet et al. (2020) to achieve optimal rates under ψ-mixing assumptions. Stability bounds were also proved by He et al. (2016), in the context of ranking, under ϕ-mixing assumptions. Rademacher bounds under β-mixing stationary conditions were first established by Mohri and Rostamizadeh (2008) via the blocking technique from Yu (1994), and later extended to the non-stationary case by Kuznetsov and Mohri (2017). Excess risks bounds, comparing the algorithm's output with the best predictor in some given class, were established by Steinwart and Christmann (2009) under geometrically α-mixing conditions. Later, Alquier and Wintenberger (2012) and Alquier et al. (2013) employed PAC-Bayesian tools to upper bound the excess risk in model selection, when data are coming from a time series generalizing the standard notion of mixing (following ideas by Rio 2000). Excess risk was also studied by Agarwal and Duchi (2013), who extended the online-to-batch conversion of Cesa-Bianchi et al. (2001) to the case of β- and ϕ-mixing data, under the hypothesis of a bounded, convex, and Lipschitz loss function. Finally, we mention the two results that are the closest to our current work. Both Chatterjee et al. (2024) and Abeles et al. (2024) build on the online-to-PAC framework introduced by Lugosi and Neu (2023). The former follow the approach of Agarwal and Duchi (2012) to deal with ϕ- and β-mixing stationary dependencies, and hence need strong regularity for the losses. Conversely, the latter consider a slightly different definition of stationary mixing and (as previously mentioned) leverage the framework of online learning with delayed feedback to perform the online-to-PAC reduction.

For the standard graph-dependence setting (with independent variables for non-adjacent vertices) a classical result is a Hoeffding-flavored concentration inequality by Janson (2004), where the graph's fractional chromatic number (a graph-theoretic combinatorial quantity) appears and re-weights the sample size. The core idea in Janson's proof consists in splitting the graph into sets whose vertices are non-adjacent (and hence independent), an approach that resembles the blocking technique from Yu (1994) for mixing processes. The first generalization bound in this framework was motivated by ranking (whose loss form naturally leads to these graph-dependencies) and obtained by Usunier et al. (2005), who built on Janson (2004) to establish Rademacher-like bounds. Later, Ralaivola et al. (2010) derived PAC-Bayesian bounds, again leveraging the same blocking technique. More recently, Zhang et al. (2019) proved a novel McDiarmid-type inequality for tree-dependent random variables, and extended it to general graphs by decomposing them into forests. Via this concentration result they established generalization bounds for uniformly stable algorithms. We refer to Zhang and Amini (2024) for a recent survey on these and other results of generalization on graphs.

Finally, of special interest is the work of Lampert et al. (2018), establishing concentration inequalities for the sum of random variables with graph-encoded correlations. Their way to deal with dependencies shares many similarities with our model, which could actually be seen as an instance of their broader setting. However, this higher generality comes at the price of a rather convoluted technical analysis, introducing more complex notions of interdependence. Their approach is based on an approximation theorem from Bradley (1983) that allows to replace a set of dependent random variables with independent copies, at a price of an additive term involving a suitably defined separation coefficient. Another closely related approach is the work of Féray (2018), proving central limit theorems by encoding dependencies into a weighted graph, whose edges' weights measure the strength of the dependencies. In the present paper, we opted to develop a simpler framework which, while slightly less flexible, allowed us to conduct a more transparent analysis, easier to adapt to practical needs. We defer to future research combining our analysis with the techniques developed in these two works.

2 THE GENERALIZATION PROBLEM

We consider a data set $S_n = (Z_1, ..., Z_n)$, drawn from a probability distribution μ_n over $\mathcal{Z}^n$, where $\mathcal{Z}$ denotes a (measurable) instance space. We assume that each Z_i has the same marginal μ. The simplest situation is when all the element of S_n are i.i.d., in which case $\mu_n = \mu^n$, but we will focus on more general situations. We denote as $\mathcal{W}$ a measurable class of hypotheses, and we let $\ell : \mathcal{W} \times \mathcal{Z} \to [0, \infty)$ be the loss function, with $\ell(w, z)$ measuring the quality of the hypothesis $w \in \mathcal{W}$ on the instance $z \in \mathcal{Z}$. The statistical learner's goal is to find a hypothesis that performs well on average, ideally the $w \in \mathcal{W}$ that minimizes the *population loss* $\mathcal{L}(w) = \mathbb{E}_\mu[\ell(w, Z)]$. Yet, μ is unknown to the learner, whose only knowledge comes from the training dataset S_n. We define the *empirical loss* to be the average of ℓ on the training data set, $\widehat{\mathcal{L}}_n(w) = \frac{1}{n}\sum_{t=1}^n \ell(w, Z_t)$.

A learning algorithm is a procedure to get a hypothesis $w \in \mathcal{W}$ starting from a training data set S_n. More generally, we will consider a *randomized* learning algorithm, that is, a mapping $\mathcal{A} : \mathcal{Z}^n \to \Delta_\mathcal{W}$, where $\Delta_\mathcal{W}$ denotes the set of probabilities over $\mathcal{W}$. Note that deterministic algorithms (mapping S_n to a single w) can be seen as a particular case of the randomized setting, where the output distribution is a Dirac mass. As previously mentioned, the ultimate goal of the statistical learner is to optimize the population loss. In the context of randomized algorithms, we aim to control the expected value of this quantity. For a probability measure $P \in \Delta_\mathcal{W}$, with a slight abuse of notation, we define the *expected population loss* as $\mathcal{L}(P) = \langle P, \mathcal{L}\rangle$, where $\langle P, f\rangle$ denotes the expectation under P of a measurable function f on $\mathcal{W}$. Similarly, we define the *expected empirical loss* as $\widehat{\mathcal{L}}_n(P) = \langle P, \widehat{\mathcal{L}}_n\rangle$.

For convenience, we denote the output of a randomized algorithm $\mathcal{A}$ as $\widehat{\mathrm{P}}_n = \mathcal{A}(S_n) \in \Delta_\mathcal{W}$. We stress here that $\widehat{\mathrm{P}}_n$ is a stochastic quantity, as it depends on the random training data set S_n. Hence, the expected population loss $\mathcal{L}(\widehat{\mathrm{P}}_n)$ is stochastic. We call *generalization bound* a high-probability inequality in the form

$$\mu_n\left(\mathcal{L}(\widehat{\mathrm{P}}_n) \le \mathcal{B}\big(\widehat{\mathcal{L}}_n(\widehat{\mathrm{P}}_n), \delta\big)\right) \ge 1 - \delta\,, \qquad (1)$$

where $\mathcal{B}$ is some function and $\delta \in [0, 1]$ is the confidence level. For the sake of brevity, we introduce the notation $\le_\delta$ for inequalities holding with probability at least $1 - \delta$, and (1) becomes $\mathcal{L}(\widehat{\mathrm{P}}_n) \le_\delta \mathcal{B}(\widehat{\mathcal{L}}_n(\widehat{\mathrm{P}}_n), \delta)$.

2.1 Online-to-PAC reduction

Lugosi and Neu (2023) have recently established a framework, which they named *online-to-PAC* conversion, to obtain generalization bounds for statistical learning (in the i.i.d. setting) by upper bounding the regret of an online learner in the following associated online learning game.

Definition 1 (Generalization game)**.** *Fix an arbitrary data set $S_n = (Z_1, \ldots, Z_n) \in \mathcal{Z}^n$. For n rounds, an online player and an adversary play the following game. At each round $t = 1, 2, \ldots, n$:*

- *the online learner picks a distribution $\pi_t \in \Delta_\mathcal{W}$;*
- *the adversary picks a map $g_t : w \mapsto \mathcal{L}(w) - \ell(w, Z_t)$;*
- *the learner incurs a cost $-\langle \pi_t, g_t\rangle$;*
- *Z_t is revealed to the learner.*

Let $\Pi = (\pi_t)_{t\ge1}$ be an online strategy for the game above. We remark that the learner's choice of π_t has to be done before Z_t is revealed, and so can only depend on the past observations (up to round $t-1$). Fixed an arbitrary $P \in \Delta_\mathcal{W}$, we define the regret of Π against P at round n as $\mathrm{R}_{\Pi,n}(P) = \sum_{t=1}^n(\langle P, g_t\rangle - \langle \pi_t, g_t\rangle)$. The online-to-PAC reduction is the next decomposition.

Theorem 1 (Theorem 1, Lugosi and Neu 2023)**.** *Fix any online strategy Π for the generalization game. Any statistical learning algorithm $\widehat{\mathrm{P}}_n = \mathcal{A}(S_n)$ satisfies*

$$\mathcal{L}(\widehat{\mathrm{P}}_n) \le \widehat{\mathcal{L}}_n(\widehat{\mathrm{P}}_n) + \frac{1}{n}\big(\mathrm{R}_{\Pi,n}(\widehat{\mathrm{P}}_n) + M_{\Pi,n}\big)\,, \qquad (2)$$

where $M_{\Pi,n} = \sum_{t=1}^n \langle \pi_t, g_t\rangle$.

A key remark to make use of this decomposition comes from the fact that, when the training data set S_n is drawn from μ^n (and hence i.i.d.), the negation $M_{\Pi,n}$, of the online learner's cumulative cost, is a martingale under the natural filtration induced by S_n, that is, the σ-fields $\mathcal{F}_t = \sigma(X_1, \dots, X_t)$. This follows from the fact that the online strategy is a predictable sequence of actions, as π_t does not depend on Z_t and is $\mathcal{F}_{t-1}$-measurable.[1] In particular, one can leverage classical martingale concentration results to get high probability generalization bounds in the form of (1). We remark that in practice it is not necessary to actually play the generalization game. Indeed, one can replace the regret $\mathrm{R}_{\Pi,n}$ by an upper bound, whenever this is known. The study and derivation of regret upper bounds is a main topic of interest in the online learning community.

For a concrete application of the above observations, we state a corollary of Theorem 1, which uses the parameter-free online strategy introduced by Orabona and Pál (2016) for learning with expert advice.

Corollary 1 (Corollary 6, Lugosi and Neu 2023)**.** *Assume that ℓ is bounded in $[0, 1]$, fix $\delta \in (0, 1)$ and an arbitrary $P \in \Delta_{\mathcal{W}}$ (whose choice cannot depend on S_n). Then, the following generalization bound holds in high probability, uniformly for all algorithms $\widehat{\mathrm{P}}_n = \mathcal{A}(S_n)$,*

$$\mathcal{L}(\widehat{\mathrm{P}}_n) \leq_\delta \widehat{\mathcal{L}}_n(\widehat{\mathrm{P}}_n) + \sqrt{\frac{3\mathrm{KL}(\widehat{\mathrm{P}}_n|P) + 9}{n}} + \sqrt{\frac{\log\frac{1}{\delta}}{2n}} \,.$$

We notice that the above result is in the typical form of a PAC-Bayes bound (Guedj, 2019; Alquier, 2024), which typically involves a complexity term in the form of the relative entropy, KL, between a data-agnostic *prior* P and the data-dependent *posterior* $\widehat{\mathrm{P}}_n$. Indeed, the framework introduced by Lugosi and Neu (2023) allows to recover several classic PAC-Bayesian results, and provides a range of generalizations thereof.

2.2 Going beyond the i.i.d. assumption

As it is the case for Corollary 1, also the other applications of Theorem 1 in Lugosi and Neu (2023) leverage the fact that $M_{\Pi,n}$ is a martingale to derive high-probability generalization bound. However, as previously mentioned, this approach cannot be directly applied when inter-dependencies among the training data are present, as these can prevent $M_{\Pi,n}$ from being a martingale. Two solutions (Chatterjee et al., 2024; Abeles et al., 2024) have been recently proposed to extend the online-to-PAC reduction to situations where the correlations in the training data set can be controlled by stationary mixing assumptions. The analysis of Chatterjee et al. (2024) was inspired by Agarwal and Duchi (2013) and involves controlling the concentration properties of $M_{\Pi,n}$ under strong regularity assumptions for the loss, leaving the online formulation untouched. On the other hand, Abeles et al. (2024) took a perhaps more natural perspective. They introduced a delayed feedback in the online generalization game (a delay of d means that Z_t is only revealed at round $t+d$), ensuring that $M_{\Pi,n}$ becomes a stationary mixing process, whose concentration can be controlled via a standard blocking technique (Yu, 1994). Our current work extends this approach to more general dependencies, encoded by a graph. To do so, we need to introduce a suitable online framework for learning on graphs that generalizes the online learning with delays setting. This will ensure that $M_{\Pi,n}$ is a sum of terms whose correlations can be suitably controlled, allowing to obtain high-probability generalization guarantees. These ideas will be formalized in the next section, after the definitions of several graph-theoretic concepts.

3 TECHNICAL TOOLS

As already mentioned, we will model dependencies between random variables in the language of graphs, and will extend the online-to-PAC conversion framework of Lugosi and Neu (2023) to deal with data points with a graph-dependency structure. This section presents the technical background that is necessary for formulating our assumptions on the data, and formulates an online learning framework defined on a graph structure, which will serve as basis for our reduction.

3.1 Basic definitions

We first introduce here a few basic definitions related to graphs, which will be used throughout our analysis.

Definition 2. *A* graph *G is a pair of sets (V, E). The elements of V are called* vertices*, or* nodes*, and the elements of E are called* edges*. Each edge is an unordered pair of elements of V.*

We will only consider loopless graphs where each edge includes two distinct vertices. Given a graph G, the set of its vertices is denoted as $V(G)$, while $E(G)$ refers to its edges. Two vertices u and v of G are said to be *adjacent* if $\{u, v\}$ is an edge in $E(G)$, otherwise they are called non-adjacent. The number of edges a vertex v belongs to is called the *degree* of v, and the *degree of the graph* is defined as the highest degree among all its vertices. The *order* of a graph is the number of its vertices. A sequence of edges in the form $\{v_0, v_1\}, \{v_1, v_2\} \dots \{v_{t-1}, v_t\}$ is called a *path* of length

[1] In general, one could let π_t depend on other sources of randomness, not encoded in the data. This can be addressed by suitably adapting the filtration, but leaves all the results that we present unchanged.

t, connecting v_0 to v_t. Two vertices are *connected* if there is a path connecting them. We define the *graph distance* $\mathrm{dist}_G(u,v)$ as the length of the shortest path from u to v. If u and v are not connected, then we let $\mathrm{dist}_G(u,v) = +\infty$.

A subset S of $V(G)$ is called a *stable subset* of the graph G if any two vertices u and v in S are non-adjacent.[2] A family $\{S_k\}_k$ of stable subsets of G is a *stable cover* if $\cup_k S_k = V(G)$. Moreover, a stable cover such that all the S_k are disjoint is called a *stable partition* of G. The *chromatic number* χ of a graph G is the cardinality of the smallest stable partition of G, namely the minimum number of stable subsets needed to form a stable partition of G.

More broadly, one can consider weighted families $\{(w_k, S_k)\}_k$ of stable subsets of G, where the w_k are non-negative coefficients. A *stable fractional cover* is a weighted family such that $\sum_k w_k \mathbf{1}_{v\in S_k} \geq 1$, for each vertex $v \in V(G)$. If $\sum_k w_k \mathbf{1}_{v\in S_k} = 1$ for any v, we speak of a *stable fractional partition*. The *fractional chromatic number* χ_f of G is the minimal value of $\sum_k w_k$, among all the stable fractional partitions of G. As any stable partition is a stable fractional partition with all the weights set to 1, we see that $\chi_f \leq \chi$.

The previous definitions can be generalized by replacing the non-adjacency condition with one involving a minimal distance. We give formal definitions for the resulting objects, which play a key role in our analysis.

Definition 3. *A* d-stable subset *S of G is a subset of $V(G)$ such that $\mathrm{dist}_G(u,v) \geq d$ for any two distinct elements u and v in S.*

Note that the 2-stable subsets of G are exactly its stable subsets, while any subset of $V(G)$ is 1-stable.

Definition 4. *A* d-stable fractional partition *of G is a weighted family of d-stable subsets of G, $\{(w_k, S_k)\}_k$, such that $\sum_k w_k \mathbf{1}_{v\in S_k} = 1$ for all $v \in V(G)$.*

Definition 5. *The* fractional d-chromatic number $\chi_f^{(d)}$ *of G is the minimal value of $\sum_k w_k$, among all the d-stable fractional partitions of G.*

Another way of thinking about d-stable sets is in terms of power graphs. The d-th power graph of G is a graph G^d such that $V(G^d) = V(G)$, with an edge for any two vertices whose distance (in G) is at most d. The d-stable subsets of G are exactly the stable subsets of G^{d-1}, and therefore $\chi_f^{(d)} = \chi_f(G^{d-1})$.

3.2 (G,ϕ)-mixing processes

We will consider a dependency structure between the training data $S_n = \{Z_1, \dots, Z_n\}$ specified in terms of a graph $G = (V, E)$, with the set of nodes V associated to the set of data points, and the edges E describing the pairwise dependencies between them. The strength of the dependence between any two points Z_i and Z_j is assumed to decay with the graph distance between the corresponding nodes v_i and v_j in the graph, with the graph distance between any pair (u,v) defined as the length of the shortest path between the two nodes. In order to define the precise dependence structure between the data points (which will be formalised in Assumption 1), we will make use of the concept of a dependence structure that we call a (G,ϕ)*-mixing process*, defined as follows.

Definition 6. *Let $X_G = \{X_v\}_{v\in V(G)}$ be a family of centred random variables, labelled on a graph G. We say that X_G is a* (G,ϕ)-mixing process *if there exists a non-negative non-increasing sequence $\phi = (\phi_d)_{d>0}$ such that, for any $v \in V$,*

$$\mathbb{E}\left[X_v \,|\mathcal{F}_{v,d}\right] \leq \phi_d\,,$$

where $\mathcal{F}_{v,d} = \sigma(\{X_{v'} : \mathrm{dist}_G(v,v') \geq d\})$.

When G is a chain (with nodes indexed by time t, and edges connecting consecutive time indices), the above definition is closely related to standard mixing assumptions, suggesting that the process effectively forgets random variables that are sufficiently far apart in time. The two main differences are that our condition focuses on expectations rather than total variation distance (or alike), and, since we use undirected graphs, it does not account for the direction of time as in typical mixing processes. Furthermore, the graph-dependency structure considered by Janson (2004), Usunier et al. (2005), and Zhang et al. (2019) is recovered by letting ϕ be a threshold sequence, such that $\phi_d = 0$ for all $d > d^\star$, and $\phi_d = +\infty$, for $d \leq d^\star$. In a way, the (G,ϕ)-mixing processes capture both the qualitative aspect of the standard graph-dependence, and the quantitative side of the mixing conditions.

Intuitively, one can expect the empirical mean of (G,ϕ)-mixing processes to concentrate around their true mean (zero) at a rate that is determined by the overall strength of dependencies: densely connected graphs are expected to yield poor concentration as compared to graphs with fewer connections. The measure of “connectedness” of the graph that we use is the fractional d-chromatic number $\chi_f^{(d)}$. The following proposition (see Appendix A.1 for the proof) provides a bound on the empirical mean of (G,ϕ)-mixing processes with bounded range.

[2] Stable subsets are also known as *independent subsets*. However we preferred the (also commonly used) term ‘stable’ to avoid confusion with probabilistic independence.

Proposition 1. *Let X_G be a (G, ϕ)-mixing process, where G is a graph of order n. Assume all the X_v take values in a bounded interval of length Δ, are centered, and have all the same marginal distribution. Then, for any $\delta > 0$, the following high probability inequality holds:*

$$\frac{1}{n} \sum_{v \in V(G)} X_v \leq_\delta \min_{d=1\ldots n} \left(\phi_d + \sqrt{\frac{\Delta^2 \chi_f^{(d)}}{2n} \log \frac{1}{\delta}} \right) .$$

3.3 Sequential learning on graphs

We aim to generalize the online-to-PAC approach introduced in Section 2 to derive generalization bounds for data with a graph-dependency structure. In order to do this, we need to define a class of online learning games that respects the graph structure that underlies the data. This section presents this class of games, which we call *sequential learning on graphs.*

Let $\mathcal{A}$ and $\mathcal{B}$ be two sets, dubbed the action space and the outcome space. We assume that $\mathcal{A}$ is a vector space. Given a graph G of order n, for each $v \in V(G)$ we define two sets $\mathcal{A}_v \subseteq \mathcal{A}$ and $\mathcal{B}_v \subseteq \mathcal{B}$. We assume that $\mathcal{A}_v$ is a convex subset of $\mathcal{A}$. We also define a cost function $C_v : \mathcal{A} \times \mathcal{B} \to \mathbb{R}$. We consider an arbitrary ordering $\{v_t\}_{t=1}^n$ of G, constituting a permutation of the vertices of G. In each round $t = 1, 2, \ldots, n$, the player moves to node v_t and picks an action $a_t \in \mathcal{A}_{v_t}$. Then, the outcome $b_t \in \mathcal{B}_{v_t}$ is revealed. The player incurs a cost $c_t(a_t, b_t)$, where $c_t = C_{v_t}$. The player can select their actions using past information only, namely at round t the action can depend on $b_1, \ldots, b_{t-1}$ and on the previous actions, but not on the present and future outcomes. For a fixed *comparator* $a \in \mathcal{A}$ and a player's strategy $\Pi = (a_t)_{t \in [n]}$, we define the *regret* of Π against a at round $T \leq n$ as

$$\mathrm{R}_{\Pi,T}(a) = \sum_{t=1}^{T} \big(c_t(a_t, b_t) - c_t(a, b_t) \big) .$$

In the specific game that we consider, the graph structure G is used to pose further constraints on how the player is allowed to select their actions. In particular, we will consider *sheltered* players, who are only allowed to use information from nodes that are "sufficiently far" from the currently selected node v_t. To make this formal, we define the *d-exterior* of a node v (where $d \in [n]$) as $U_{v,d} = \{u \in G : \mathrm{dist}_G(u, v) \geq d\}$.

Definition 7. *In the online game defined above, a* d-sheltered learner *is a player whose action a_t in round t can only depend on outcomes b_s, from rounds $s < t$ such that $v_s \in U_{v_t,d}$.*

The following result shows that an upper bound on the regret of a standard learner often translates into an upper bound for the regret of a d-sheltered learner.

Proposition 2. *Assume that, for all $v \in V(G)$, the cost C_v is convex in a. If there exists a standard online strategy Π achieving regret $\mathrm{R}_{\Pi,T}(a) \leq F(T)$ for any $T \leq n$, where F is a concave function, then, for any $d \in [n]$, there is a d-sheltered learner with strategy Π_d, whose regret is bounded as*

$$\mathrm{R}_{\Pi_d,n}(a) \leq \chi_f^{(d)} F\big(n/\chi_f^{(d)}\big) .$$

We obtain the above result (proof in Appendix A.2) in a constructive way, by explicitly devising a d-sheltered learner's strategy by averaging the actions of several standard players. Notably, the resulting class of games generalizes the well-studied setting of online learning with delayed feedback (Weinberger and Ordentlich, 2002; Joulani et al., 2013). Indeed, this setting is seen as the special case where G is a chain and the player is constrained to be d-sheltered, with d corresponding to the delay in observing the feedback, and $\chi_f^{(d)} = d$. The rates of Weinberger and Ordentlich (2002) and Joulani et al. (2013) are recovered by our result. We defer a discussion of other related online settings to Section 5.

4 GENERALIZATION BOUNDS UNDER GRAPH-MIXING

We are now ready to state our assumptions on the dependence structure of the training data, and provide our main results: the graph-mixing counterparts of the generalization bounds of Section 2.

Our main assumption on the dependencies is that, for any hypothesis $w \in \mathcal{W}$, the losses $\ell(w, Z_t)$ constitute a (G, ϕ)-mixing process. This is formalized as follows.

Assumption 1. *Let $S_n = (Z_1, \ldots, Z_n)$ be a training data set drawn from a distribution μ_n on $\mathcal{Z}^n$, such that each Z_t has the same marginal distribution μ. We assume that there exists a graph G (of order n), a bijection $\iota : G \to [n]$, and a non-negative non-increasing sequence $\phi = (\phi_d)_{d>0}$, such that, for all $w \in \mathcal{W}$, the graph-labelled process $X_G(w) = \big(X_v(w)\big)_{v \in V(G)}$ is a (G, ϕ)-mixing process, where*

$$X_v(w) = \mathcal{L}(w) - \ell(w, Z_{\iota(v)}) .$$

This assumption is essentially an extension to the graph setting of the mixing condition proposed by Abeles et al. (2024). It comes from the intuition that the loss associated with the observations Z_v becomes almost independent with respect to the family of points which are at least d edges away in the associated graph.

We can now state the graph-mixing counterpart of Theorem 1. First, we notice that the generalization game of Definition 1 induces an online problem on G.

Definition 8 (Generalization game on G)**.** *Consider a training data set S_n satisfying Assumption 1 with graph G and bijection ι. Consider the following online game on G. For all $v \in V(G)$, let $\mathcal{A}_v = \mathcal{A} = \Delta_{\mathcal{W}}$, $\mathcal{B}_v = \mathcal{B}$ be the space of measurable functions on $\mathcal{W}$, and $C_v(a, b) = -\langle a, b\rangle$. For n rounds, an online player and an adversary play the following game. At round $t = 1, \ldots, n$:*

- *the online player moves on $v_t = \iota^{-1}(t)$;*
- *the online player picks a distribution $\pi_t \in \Delta_{\mathcal{W}}$;*
- *the adversary picks a map $g_t : w \mapsto \mathcal{L}(w) - \ell(w, Z_t)$;*
- *the learner incurs a cost $-\langle \pi_t, g_t\rangle$;*
- *Z_t is revealed to the learner.*

Combining the results from Sections 3 and 3.3 we obtain the following generalization result.

Theorem 2. *Consider a data set S_n that satisfies Assumption 1. Fix $d \in [n]$ and an arbitrary strategy Π of a d-sheltered player for the game of Definition 8. For $v \in V(G)$, define $\widetilde{X}_v = \langle \pi_{\iota(v)}, g_{\iota(v)}\rangle$. Then, any statistical learning algorithm $\widehat{\mathrm{P}}_n = \mathcal{A}(S_n)$ satisfies*

$$\mathcal{L}(\widehat{\mathrm{P}}_n) \leq \widehat{\mathcal{L}}_n(\widehat{\mathrm{P}}_n) + \frac{\mathrm{R}_{\Pi,n}(\widehat{\mathrm{P}}_n) + M_\Pi}{n}\,, \tag{3}$$

with $M_\Pi = \sum_{v \in V(G)} \widetilde{X}_v$. Moreover, $\widetilde{X}_G$ is a $(G, \widetilde{\phi})$-mixing process, where we let $\widetilde{\phi}_{d'} = \phi_{d'}$ for $d' \geq d$, and $\widetilde{\phi}_{d'} = +\infty$ for $d' < d$.

Proof. (3) is equivalent to (2), so we will only need to show that $\widetilde{X}_G$ is $(G, \widetilde{\phi})$-mixing. Clearly, when $d' < d$, we have $\mathbb{E}[X_v|\mathcal{F}_{v,d'}] \leq \widetilde{\phi}_{d'} = +\infty$. For $d' \geq d$, $\mathcal{F}_{v,d'} \supseteq \mathcal{F}_{v,d}$, and so $\pi_{\iota(v)}$ is $\mathcal{F}_{v,d'}$-measurable, by definition of d-sheltered learner. Hence, $\mathbb{E}[\widetilde{X}_v|\mathcal{F}_{v,d'}] = \langle \pi_{\iota(v)}, \mathbb{E}[g_{\iota(v)}|\mathcal{F}_{v,d'}]\rangle$, and $\mathbb{E}[g_{\iota(v)}(w)|\mathcal{F}_{v,d'}] \leq \phi_{d'} = \widetilde{\phi}_{d'}$ for all $w \in \mathcal{W}$ by Assumption 1. □

The usefulness of the above result comes from the fact that we know how to upper bound (in high probability) (G, ϕ)-mixing processes. Hence, we can derive a graph-mixing counterpart of Corollary 1.

Corollary 2. *Consider a data set S_n that satisfies Assumption 1, assume that ℓ is bounded in $[0, 1]$, fix $\delta \in (0, 1)$, $d \in [n]$, and an arbitrary $P \in \Delta_{\mathcal{W}}$ (whose choice cannot depend on S_n). Then, the following generalization bound holds in high probability, uniformly for all algorithms $\widehat{\mathrm{P}}_n = \mathcal{A}(S_n)$,*

$$\begin{aligned}\mathcal{L}(\widehat{\mathrm{P}}_n) &\leq_\delta \widehat{\mathcal{L}}_n(\widehat{\mathrm{P}}_n) + \phi_d \\ &\quad + \left(\sqrt{3\mathrm{KL}(\widehat{\mathrm{P}}_n|P) + 9} + \sqrt{\frac{1}{2}\log\frac{1}{\delta}}\right)\sqrt{\frac{\chi_f^{(d)}}{n}}\,.\end{aligned}$$

Proof. The proof combines Theorem 2, Proposition 1, and Proposition 2. Fix $P \in \Delta_{\mathcal{W}}$ and $d \in [n]$. By a slight generalization of Corollary 6 in Orabona and Pál (2016) (see the proof of Corollary 6 in Lugosi and Neu, 2023), we know that for any d there is a (standard) online strategy for the game of Definition 8, whose regret (for any comparator P') is upper bounded by $\sqrt{(3 + \mathrm{KL}(P'|P))3n}$. By Proposition 2, there is a d-sheltered online strategy Π_d whose regret is upper bounded by $\sqrt{(3{+}\mathrm{KL}(P'|P))3n\chi_f^{(d)}}$. Apply Theorem 2. Since $\widetilde{\phi}_d = \phi_d$, by Proposition 1 $\frac{1}{n}M_{\Pi_n^{(d)}} \leq_\delta \phi_d + \sqrt{\frac{1}{2n}\chi_f^{(d)}\log\frac{1}{\delta}}$, and so we conclude. □

We stress that this is only one of the many possible bounds that can be derived from our framework, given that different online learning algorithms may lead to different regret bounds. We refer to Section 3 of Lugosi and Neu (2023) for further examples, including generalized PAC-Bayesian bounds where the relative entropy, KL, appearing in the above bound is replaced by other strongly convex functionals of $\widehat{\mathrm{P}}_n$.

The tightness of these bounds relies on the chromatic number of the power graph, and the coefficients ϕ characterizing the strength of dependencies. Typical assumptions regarding the latter include functions of the form $\phi_d = Ce^{-d/\tau}$, for some $C, \tau > 0$ (called *geometric mixing*), or $\phi_d = Cd^{-r}$ for some $C, r > 0$ (called *algebraic mixing*). As for the chromatic indices, it is known that they can always be bounded as $\chi_f^{(d)} = \mathcal{O}(\Delta^d)$ where Δ is the degree of the original graph. It is often possible, though, to show tighter bounds for graphs that arise naturally in practical applications. We demonstrate a few concrete examples below, and refer to Alon and Mohar (2002) for a more exhaustive treatment.

Temporal processes. The simplest non-trivial example is the class of mixing processes in time, which we have already mentioned extensively. These processes can be modeled by a graph G, whose nodes correspond to the time indices $1, 2, \ldots, n$, and edges connect neighboring indices, namely $E(G) = (\{(t, t+1\})_{t \in [n]}$. This can model a variety temporally-dependent data sequences, such as stock prices, energy consumption, or sensor data from physical environments (see, e.g., Ariyo et al., 2014; Takeda et al., 2016). In this case, one can easily see that $\chi_f^{(d)} = d$. Thus, in this setting our guarantees almost exactly recover the recent results of Abeles et al. (2024). We refer the reader to their work for details.

Processes on a spatial grid. A direct generalization of the previous case is where the graph is a 2-dimensional grid of size $n = I \times J$, for some integers I and J. Such graphs can model spatially organized data like the house-price example mentioned in the introduction. A straightforward calculation shows that $\chi_f^{(d)}$ is of order d^2 for this class of graphs[3]. For the sake of concreteness, let us suppose that the mixing is geometric. Then, the guarantee of Corollary 2 implies a generalization bound of order $\mathcal{O}\Big(Ce^{-d/\tau}+\sqrt{\frac{d^2}{n}\big(\mathrm{KL}(\widehat{\mathrm{P}}_n|P)+\log\frac{1}{\delta}\big)}\Big)$. Setting $d = \tau\log(Cn)$, neglecting log factors this becomes $\widetilde{\mathcal{O}}\Big(\sqrt{\tau^2\big(\mathrm{KL}(\widehat{\mathrm{P}}_n|P)+\log\frac{1}{\delta}\big)/n}\Big)$, thus demonstrating a linear dependence with the mixing-time parameter τ. This argument can be easily extended to other planar graphs of similar regularity, and generalized to k-dimensional grids where the chromatic indices scale as d^k, eventually yielding a dependence of order $\tau^{k/2}$ on the mixing time.

5 CONCLUSION

We have introduced a new model for statistical learning with dependent data, and provided a general framework for developing generalization bounds for learning algorithms. A key tool in our analysis was a reduction to a family of online learning games. We conclude by discussing some further related work and highlighting some interesting open problems.

The tightness of our bounds. Our upper bounds on the generalization error depend on variations of the chromatic number of the dependency graph. While it is easy to construct hard examples where this dependence is tight (e.g., when G is composed of several disconnected cliques), it is not clear if our bounds can be further improved to scale with more fine-grained graph properties. On a related note, it is also easy to construct examples where our bounds are vacuous, yet it still should be possible to estimate the test error with good rates. To this end, consider a graph of size n, with $n/2$ isolated vertices and the remaining $n/2$ vertices forming a clique. The chromatic number of this graph is $n/2$, which makes our bounds trivial. However, in such a case it is clearly a bad idea to measure the training error on all samples: the heavy dependence of the second half of the data points leads to a massive bias. This bias, however, can be completely removed by simply discarding the second half of observations and only using the i.i.d. samples. This pathological case suggests that the empirical mean can be an arbitrarily poor estimator of the mean, and much more efficient estimators can be constructed by taking the graph structure into account. Our analysis suggests an obvious way to do so: find the largest d-stable subset and then use only data points from this set. Our techniques can be used to show the same generalization bound for this method as for the empirical mean, but the example above indicates that its actual performance could be much better. The downside, of course, is that this approach requires full knowledge of the graph and requires additional computation. In contrast, our bounds need only high-level information about the graph, as they only assume knowledge of the chromatic numbers, which might be easier to estimate than finding stable sets. We leave a detailed investigation of this interesting question open for future work.

Online learning on graphs. To our knowledge, the sequential learning framework we introduce in Section 3.3 has not appeared in the previous literature. That said, several similar models have been studied. The works of Guillory and Bilmes (2009) andCesa Bianchi et al. (2010) consider learning labelings on graphs via actively querying a subset of the labels, and provide mistake bounds that depend on a joint notion of complexity of the labeling and the graph. Obtaining guarantees in terms of such problem-dependent notions of complexity would be desirable in our setting as well, but unfortunately their model is rather different from ours. A more relevant setting is the one studied by Cesa-Bianchi et al. (2020), who study an online learning protocol defined on a network of agents. In each round, one agent wakes up, needs to make a prediction, suffers a loss, and shares the observation with its neighbors. In a certain sense, this problem is the dual of ours: in our setting, a sheltered online learner is not allowed to use information from the neighbors of the currently active node, whereas their setting only allows using information from neighboring nodes. The two settings can be transformed into each other by taking the complements of the underlying graphs. Applying their algorithm to our setting in the most straightforward way yields guarantees that can be recovered by Proposition 2. We find it plausible that approaching our problem from this alternative direction may lead to improved data-dependent guarantees (as suggested by existing follow-up work like that of Achddou et al. 2024), but so far we do not see sufficient evidence to prefer this rather roundabout route over our rather simple formulation that addresses our overall problem in more natural terms. We remain optimistic nevertheless that further progress on online learning with graph structures will enable improvements in the statistical learning setting we study in the present paper. As a final remark, the online framework that we introduced, and the way we developed to couple it with graph-mixing processes' concentration, are

[3] To see this, note that the set of nodes reachable through a path of length d roughly corresponds to the nodes falling into a square of diagonal $2d$ on the two-dimensional plane.

likely to be useful to adapt other algorithmic statistics approaches to graph-mixing dependent settings.

Acknowledgements

The authors would like to thank Rui-Ray Zhang and Gábor Lugosi for the insightful discussions that inspired this work. This project has received funding from the European Research Council (ERC), under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 950180).

References

Abeles, B., Clerico, E., and Neu, G. (2024). Generalization bounds for mixing processes via delayed online-to-pac conversions. *arXiv:2406.12600*.

Achddou, J., Cesa-Bianchi, N., and Laforgue, P. (2024). Multitask online learning: Listen to the neighborhood buzz. *AISTATS*.

Agarwal, A. and Duchi, J. C. (2012). The generalization ability of online algorithms for dependent data. *IEEE Transactions on Information Theory*, 59(1).

Agarwal, A. and Duchi, J. C. (2013). The generalization ability of online algorithms for dependent data. *IEEE Transactions on Information Theory*, 59(1):573–587.

Alon, N. and Mohar, B. (2002). The chromatic number of graph powers. *Combinatorics, Probability and Computing*, 11(1):1–10.

Alquier, P. (2024). User-friendly introduction to pac-bayes bounds. *Foundations and Trends in Machine Learning*, 17(2):174–303.

Alquier, P., Li, X., and Wintenberger, O. (2013). Prediction of time series by statistical learning: General losses and fast rates. *Dependence Modeling*, 1:65–93.

Alquier, P. and Wintenberger, O. (2012). Model selection for weakly dependent time series forecasting. *Electronic Journal of Statistics*, 6:1447–1464.

Ariyo, A. A., Adewumi, A. O., and Ayo, C. K. (2014). Stock price prediction using the arima model. *UKSim-AMSS International Conference on Computer Modelling and Simulation*.

Bousquet, O., Boucheron, S., and Lugosi, G. (2004). *Introduction to Statistical Learning Theory*. Springer.

Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. *Journal of Machine Learning Research*, 2.

Bousquet, O., Klochkov, Y., and Zhivotovskiy, N. (2020). Sharper bounds for uniformly stable algorithms. *COLT*.

Bradley, R. C. (1983). Approximation theorems for strongly mixing random variables. *Michigan Mathematical Journal*, 30(1):69–81.

Bradley, R. C. (2005). Basic properties of strong mixing conditions: A survey and some open questions. *Probability Surveys*, 2:107–144.

Cesa-Bianchi, N., Cesari, T., and Monteleoni, C. (2020). Cooperative online learning: Keeping your neighbors updated. *ALT*.

Cesa-Bianchi, N., Conconi, A., and Gentile, C. (2001). On the generalization ability of on-line learning algorithms. *NeurIPS*.

Cesa Bianchi, N., Gentile, C., Vitale, F., and Zappella, G. (2010). Active learning on trees and graphs. *COLT*.

Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge university press.

Chatterjee, S., Mukherjee, M., and Sethi, A. (2024). Generalization bounds for dependent data using online-to-batch conversion. *NeurIPS*.

Feldman, V. and Vondrak, J. (2019). High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. *COLT*.

Foster, D. J., Kakade, S. M., Qian, J., and Rakhlin, A. (2021). The statistical complexity of interactive decision making. *arXiv:2112.13487*.

Fu, S., Lei, Y., Cao, Q., Tian, X., and Tao, D. (2023). Sharper bounds for uniformly stable algorithms with stationary mixing process. *ICLR*.

Féray, V. (2018). Weighted dependency graphs. *Electronic Journal of Probability*, 23:1–65.

Grünwald, P., de Heide, R., and Koolen, W. (2024). Safe testing. *Journal of the Royal Statistical Society Series B: Statistical Methodology*.

Guedj, B. (2019). A primer on pac-bayesian learning. *arXiv:1901.05353*.

Guillory, A. and Bilmes, J. A. (2009). Label selection on graphs. *NeurIPS*.

He, F., Zuo, L., and Chen, H. (2016). Stability analysis for ranking with stationary ϕ-mixing samples. *Neurocomputing*, 171:1556–1562.

Jang, K., Jun, K.-S., Kuzborskij, I., and Orabona, F. (2023). Tighter PAC-Bayes bounds through coin-betting. *COLT*.

Janson, S. (2004). Large deviations for sums of partly dependent random variables. *Random Structures & Algorithms*, 24(3):234–248.

Joulani, P., György, A., and Szepesvári, C. (2013). Online learning under delayed feedback. *ICML*.

Kakade, S. M., Sridharan, K., and Tewari, A. (2008). On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. *NeurIPS*.

Kontorovich, L. and Ramanan, K. (2008). Concentration inequalities for dependent random variables via the martingale method. *The Annals of Probability*, 36(6):2126–2158.

Kuznetsov, V. and Mohri, M. (2017). Generalization bounds for non-stationary mixing processes. *Machine Learning*, 106(1):93–117.

Lampert, C. H., Ralaivola, L., and Zimin, A. (2018). Dependency-dependent bounds for sums of dependent random variables. *arXiv:1811.01404*.

Lugosi, G. and Neu, G. (2023). Online-to-PAC conversions: Generalization bounds via regret analysis. *arXiv:2305.19674*.

Mohri, M. and Rostamizadeh, A. (2007). Stability bounds for non-i.i.d. processes. *NeurIPS*.

Mohri, M. and Rostamizadeh, A. (2008). Rademacher complexity bounds for non-i.i.d. processes. *NeurIPS*.

Mohri, M. and Rostamizadeh, A. (2010). Stability bounds for stationary ϕ-mixing and β-mixing processes. *Journal of Machine Learning Research*, 11(Feb):798–814.

Montanari, A. and Saberi, A. (2010). The spread of innovations in social networks. *Proceedings of the National Academy of Sciences*, 107(47):20196–20201.

Orabona, F. (2019). A modern introduction to online learning. *arXiv:1912.13213*.

Orabona, F. and Jun, K.-S. (2024). Tight concentrations and confidence sequences from the regret of universal portfolio. *IEEE Transactions on Information Theory*, 70(1):436–455.

Orabona, F. and Pál, D. (2016). Coin betting and parameter-free online learning. *NeurIPS*.

Rakhlin, A. and Sridharan, K. (2017). On equivalence of martingale tail bounds and deterministic regret inequalities. *COLT*.

Ralaivola, L., Szafranski, M., and Stempfel, G. (2010). Chromatic PAC-Bayes bounds for non-iid data: Applications to ranking and stationary β-mixing processes. *Journal of Machine Learning Research*, 11(Sep):1927–1956.

Rio, E. (2000). Inégalités de Hoeffding pour les fonctions lipschitziennes de suites dépendantes. *Comptes Rendus de l'Académie des Sciences, Série I, Mathématiques*, 330:905–908.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press.

Steinwart, I. and Christmann, A. (2009). Fast learning from non-i.i.d. observations. *NeurIPS*.

Takeda, H., Tamura, Y., and Sato, S. (2016). Using the ensemble Kalman filter for electricity load forecasting and analysis. *Energy*, 104.

Usunier, N., Amini, M. R., and Gallinari, P. (2005). Generalization error bounds for classifiers trained with interdependent data. *NeurIPS*.

Vapnik, V. (2000). *The nature of statistical learning theory*. Springer-Verlag.

Weinberger, M. and Ordentlich, E. (2002). On delayed prediction of individual sequences. *IEEE Transactions on Information Theory*, 48(7).

Yu, B. (1994). Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, 22(1):94–116.

Yu, B., Yin, H., and Zhu, Z. (2018). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *IJCAI*.

Zhang, R. R. and Amini, M. R. (2024). Generalization bounds for learning under graph-dependence: A survey. *Machine Learning*, 113(7):3929–3959.

Zhang, R. R., Liu, X., Wang, Y., and Wang, L. (2019). Mcdiarmid-type inequalities for graph-dependent variables and stability bounds. *NeurIPS*.

Checklist

The checklist follows the references. For each question, choose your answer from the three possible options: Yes, No, Not Applicable. You are encouraged to include a justification to your answer, either by referencing the appropriate section of your paper or providing a brief inline description (1-2 sentences). Please do not modify the questions. Note that the Checklist section does not count towards the page limit. Not including the checklist in the first submission won't result in desk rejection, although in such case we will ask you to upload it during the author response period and include it in camera ready (if accepted).

In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes, see section 2 and section 3]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Not Applicable]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes, see section 3 and section 4]
   (b) Complete proofs of all theoretical results. [Yes either in the main text or in the appendix.]
   (c) Clear explanations of any assumptions. [Yes, the main assumption we consider in this paper is assumption 1. We provide a concrete intuition in section 3 and gives concrete example in section 4]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Not Applicable]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Not Applicable]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Not Applicable]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary material

A Omitted proofs

A.1 Proof of Proposition 1

The proof leverages the approach introduced by Janson (2004). Fix d and consider a d-stable fractional partition $\{(w_k, S_k)\}_k$ of G. We can write

$$\sum_{v\in V(G)} X_v = \sum_{v\in V(G)} X_v \sum_{k:v\in S_k} w_k = \sum_k w_k \sum_{v\in S_k} X_v \,.$$

In particular, for any $\lambda > 0$, we have

$$\log \mathbb{E}\left[e^{\frac{\lambda}{n}\sum_{v\in V(G)} X_v}\right] = \log \mathbb{E}\left[e^{\frac{\lambda}{n}\sum_k w_k \sum_{v\in S_k} X_v}\right] \leq \sum_k p_k \log \mathbb{E}\left[e^{\frac{\lambda}{n}\frac{w_k}{p_k}\sum_{v\in S_k} X_v}\right] \,, \tag{4}$$

where p is a probability vector ($\sum_k p_k = 1$ with $p_k > 0$ for all k), and in the last step we have applied Jensen's inequality, since $f \mapsto \log \mathbb{E}[e^f]$ is a convex mapping.

Now, for any k we can label arbitrarily the elements in S_k as $v_1^{(k)}, \dots, v_{n_k}^{(k)}$, where n_k is the cardinality of S_k. Let us denote as $\mathcal{F}_i^{(k)}$ the sigma algebra $\sigma(\{X_{v_j^{(k)}} \,:\, j \leq i\})$. Since S_k is a d-stable subset of G, recalling the notation introduced in Definition 6, we have that $\mathcal{F}_{i-1}^{(k)} \subseteq \mathcal{F}_{v_i^{(k)},d}$. In particular, the fact that X_G is a (G,ϕ)-mixing process implies that

$$\mathbb{E}[X_{v_i^{(k)}} | \mathcal{F}_{i-1}^{(k)}] = \mathbb{E}\big[\mathbb{E}[X_{v_i^{(k)}} | \mathcal{F}_{v_i^{(k)},d}] \big| \mathcal{F}_{i-1}^{(k)}\big] \leq \phi_d$$

by the tower property of conditional expectation. Now, this implies that for any $i \leq n_k$ we have

$$\mathbb{E}\left[e^{\frac{\lambda}{n}\frac{w_k}{p_k}\sum_{j=1}^{i} X_{v_j^{(k)}}} \middle| \mathcal{F}_{i-1}^{(k)}\right] \leq \mathbb{E}\left[e^{\frac{\lambda}{n}\frac{w_k}{p_k}\sum_{j=1}^{i-1} X_{v_j^{(k)}}}\right] \mathbb{E}\left[e^{\frac{\lambda}{n}\frac{w_k}{p_k}(X_{v_i^{(k)}} - \mathbb{E}[X_{v_i^{(k)}}|\mathcal{F}_{i-1}])} \middle| \mathcal{F}_{i-1}^{(k)}\right] \exp\left(\frac{\lambda\phi_d}{n}\frac{w_k}{p_k}\right) \,.$$

Moreover, the fact that each X_v is bounded in an interval I of length Δ implies that it is $\Delta^2/4$-subgaussian with respect to any measure, and hence

$$\mathbb{E}\left[e^{\frac{\lambda}{n}\frac{w_k}{p_k}\left(X_{v_i^{(k)}} - \mathbb{E}[X_{v_i^{(k)}}|\mathcal{F}_{i-1}]\right)} \middle| \mathcal{F}_{i-1}^{(k)}\right] \leq \exp\left(\frac{\lambda^2\Delta^2}{8n^2}\frac{w_k^2}{p_k^2}\right) \,.$$

Applying these arguments recursively n_k times we obtain

$$\log \mathbb{E}\left[e^{\frac{\lambda}{n}\frac{w_k}{p_k}\sum_{v\in S_k} X_v}\right] \leq n_k \left(\frac{\lambda^2\Delta^2}{8n^2}\frac{w_k^2}{p_k^2} + \frac{\lambda\phi_d}{n}\frac{w_k}{p_k}\right) \,.$$

We can hence rewrite (4) as

$$\log \mathbb{E}\left[e^{\frac{\lambda}{n}\sum_{v\in V(G)} X_v}\right] \leq \sum_k n_k \left(\frac{\lambda^2\Delta^2}{8n^2}\frac{w_k^2}{p_k} + \frac{\lambda\phi_d}{n} w_k\right) = \sum_k \frac{\lambda^2\Delta^2}{8n^2}\frac{n_k w_k^2}{p_k} + \lambda\phi_d \,,$$

where in the last equality we used that

$$\sum_k w_k n_k = \sum_k w_k \sum_{v\in S_k} 1 = \sum_k \sum_{v\in V(G)} w_k \mathbf{1}_{v\in S_k} = \sum_{v\in V(G)} \sum_{k:v\in S_k} w_k = \sum_{v\in V(G)} 1 = n \,. \tag{5}$$

We can now optimize the choice of p, by setting $p_k = \frac{w_k\sqrt{n_k}}{\sum_{k'} w_{k'}\sqrt{N_{k'}}}$. With this choice we have

$$\log \mathbb{E}\left[e^{\frac{\lambda}{n}\sum_{v\in V(G)} X_v}\right] \leq \frac{\lambda^2}{2n^2}\left(\sum_k w_k\sqrt{n_k}\right)^2 + \lambda\phi_d \,.$$

By Cauchy-Schwarz inequality we have

$$\sum_k w_k \sqrt{n_k} = \sum_k \sqrt{w_k}\sqrt{w_k n_k} \leq \sqrt{\sum_k w_k}\sqrt{\sum_k w_k n_k} = \sqrt{n \sum_k w_k}\,,$$

where again we used (5). Since the choice of the d-stable fractional partition is arbitrary, we can chose an optimal one, such that $\sum_k w_k = \chi_f^{(d)}$. In particular, we get

$$\log \mathbb{E}\left[e^{\frac{\lambda}{n}\sum_{v\in V(G)} X_v}\right] \leq \frac{\lambda^2 \Delta^2 \chi_f^{(d)}}{8n} + \lambda \phi_d\,.$$

By Markov's inequality, we have that for any $t > \phi_d$

$$\mathbb{P}\left(\frac{1}{n}\sum_{v\in V(G)} X_v \geq t\right) \leq \inf_{\lambda>0} \frac{\mathbb{E}\left[e^{\frac{\lambda}{n}\sum_{v\in V(G)} X_v}\right]}{e^{\lambda t}} \leq \inf_{\lambda>0} \exp\left(\frac{\lambda^2\Delta^2\chi_f^{(d)}}{8n} - \lambda(t-\phi_d)\right) = \exp\left(-\frac{2n}{\chi_f^{(d)}}\frac{(t-\phi_d)^2}{\Delta^2}\right)\,.$$

The conclusion now follows by setting the RHS above equal to δ and solving for t. □

A.2 Proof of Proposition 2

First, let us fix an arbitrary d-stable fractional partition $\{(w_k, S_k)\}_k$ of G. The idea is that we will run an independent player on each S_k. Each of them will also be a d-sheltered learner, as, by definition, in a d-stable fractional partition any two distinct vertices are distant at least d from each other. We will see that, by carefully averaging the actions of these players, it is possible to obtain a d-sheltered learner on the full graph G, whose regret can be upper bounded as desired.

First, note that the ordering of $V(G)$ induces an ordering on S_k, and we will write $S_k = (v_1^{(k)}, \dots, v_{n_k}^{(k)})$, where n_k is the cardinality of S_k. We now introduce some notation which will be helpful for what follows. Any vertex $v \in V(G)$ corresponds to an element in the ordered sequence $v_1, \dots, v_n$. We denote as $\iota(v)$ the index of this element (so that $\iota(v_t) = t$ for all t). Similarly, given an element $v \in S_k$, we denote as ι_k its index in the sequence $(v_1^{(k)}, \dots, v_{n_k}^{(k)})$.

For each S_k, we let run an independent copy of the standard player, and we denote their strategy as $\Pi_k = (a_1^{(k)}, \dots, a_{n_k}^{(k)})$. We assume that although these players' choices of the action are independent from each other, for each vertex each player who passes through it receives the same outcome,[4] which corresponds to the outcome that the d-sheltered learner running on the full graph sees. By assumption, we can choose the strategy of the standard player so that we can upper bound the regret of each Π_k as

$$\mathrm{R}_{\Pi_k, n_k}(a) = \sum_{t=1}^{n_k} \left(c_t^{(k)}(a_t^{(k)}, b_t^{(k)}) - c_t^{(k)}(a, b_t^{(k)})\right) \leq F(n_k)\,,$$

where $c_t^{(k)} = C_{v_t^{(k)}}$ and $b_t^{(k)} = b_\iota(v_t^{(k)})$ is the outcome on $v_t^{(k)}$ (which, as previously stated, only depends on the vertex and not on which player is observing it, as it corresponds to the outcome that the d-sheltered learner sees at round $\iota(v_t^{(k)})$).

We will now define the d-sheltered learner's strategy $\Pi = (a_1, \dots, a_n)$ on the full graph. For any $v \in V(G)$, let $\kappa(v) = \{k : v \in S_k\}$. We set $a_t = A(v_t)$, with

$$A(v) = \sum_{k\in\kappa(v)} w_k a_{\iota_k(v)}^{(k)}\,.$$

First, we notice that $A(v) \in \mathcal{A}_v$, since $\mathcal{A}_v$ is assumed to be convex and $A(v)$ is a convex mixture of elements in $\mathcal{A}_v$ (note that $\sum_{k\in\kappa(v)} w_k = 1$ by definition of d-stable fractional partition).

[4] This is somehow limiting the power of a potential adversary for each of these games, but this does not affect the regret bounds that hold for any possible outcome sequence. Also, notice that for the d-sheltered game a potential adversary is allowed to choose freely for any vertex v, and indeed it is this chosen outcome that each of the players of the d-stable subsets will see when passing on v.

We now show that the above definition of a_t define an admissible strategy for a d-sheltered learner. First, we see that a_t only depends on what has happened on the set $H_t = \{v_s^{(k)} : k \in \kappa(v_t) \text{ and } s \leq \iota_k(v_t) - 1\}$. Clearly, $H_t \subseteq \cup_{k\in\kappa(v_t)} S_k$. Since for any $k \in \kappa(v_t)$ all the element in S_k (excluded v_t itself) are distant at least d from v_t, the d-sheltered property of the learner is ensured. Moreover, one can easily check that, for any v and any $k \in \kappa(v)$, it holds that $\iota_k(v) \leq \iota(v)$. Thus, $H_t \subseteq \{v_s : s < t\}$, which means that the learner is only allowed to access past information, as required. We have hence proven that the strategy that we defined is admissible for a d-sheltered learner.

We now study the regret of this d-sheltered learner. For any v, let $r(v) = c_{\iota(v)}(a_{\iota(v)}, b_{\iota(v)}) - c_{\iota(v)}(a, b_{\iota(v)})$, and, for $v \in S_k$, define $r_k(v) = c_{\iota(v)}(a^{(k)}_{\iota_k(v)}, b_{\iota(v)}) - c_{\iota(v)}(a, b_{\iota(v)})$, where we used that, for any $v \in S_k$, $c^{(k)}_{\iota_k(v)} = c_\iota(v)$ and $b^{(k)}_{\iota_k(v)} = b_{\iota(v)}$. With these definitions in mind we can rewrite

$$\mathrm{R}_{\Pi,n}(a) = \sum_{v\in V(G)} r(v) \qquad \text{and} \qquad \mathrm{R}_{\Pi_k,n_k}(a) = \sum_{v\in S_k} r_k(v)\,.$$

Now, notice that thanks to the convexity of the cost we have that

$$c_{\iota(v)}(a_{\iota(v)}, b_{\iota(v)}) \leq \sum_{k\in\kappa(v)} w_k c_{\iota(v)}(a^{(k)}_{\iota_k(v)}, b_{\iota(v)})\,,$$

by Jensen's inequality, as $\sum_{k\in\kappa(v)} w_k = 1$. In particular, $r(v) \leq \sum_{k\in\kappa(v)} w_k r_k(v) = \sum_k \mathbf{1}_{v\in S_k} w_k r_k(v)$, and hence

$$\mathrm{R}_{\Pi,n}(a) = \sum_{v\in V(G)} r(v) \leq \sum_{v\in V(G)} \sum_k \mathbf{1}_{v\in S_k} w_k r_k(v) = \sum_k w_k \sum_{v\in S_k} r_k(v) = \sum_k w_k \mathrm{R}_{\Pi_k,n_k}(a)\,.$$

The fact that $\mathrm{R}_{\Pi_k,n_k}(a) \leq F(n_k)$ yields

$$\mathrm{R}_{\Pi,n}(a) \leq \sum_k w_k F(n_k) = \left(\sum_k w_k\right) \sum_k \frac{w_k}{\sum_{k'} w_{k'}} F(n_k) \leq \left(\sum_k w_k\right) F\left(\frac{n}{\sum_k w_k}\right)\,,$$

where the last inequality follows from Jensen's inequality (since F is concave) and (5). Finally, notice that what we have proven so far holds for any arbitrary d-stable fractional partition. In particular, we can select the partition such that $\sum_k w_k = \chi_f^{(d)}$, and hence obtain

$$\mathrm{R}_{\Pi,n}(a) \leq \chi_f^{(d)} F\big(n/\chi_f^{(d)}\big)\,,$$

which is the regret upper bound that we wanted to prove. □