McDiarmid-Type Inequalities for Graph-Dependent Variables and Stability Bounds

Rui (Ray) Zhang *
School of Mathematics
Monash University
rui.zhang@monash.edu

Xingwu Liu †
Institute of Computing Technology,
Chinese Academy of Sciences.
University of Chinese Academy of Sciences
liuxingwu@ict.ac.cn

Yuyi Wang
ETH Zurich, Switzerland
X-Order Lab, China
yuyiwang920@gmail.com

Liwei Wang
Key Laboratory of Machine Perception, MOE,
School of EECS, Peking University
Center for Data Science, Peking University
wanglw@cis.pku.edu.cn

Abstract

A crucial assumption in most statistical learning theory is that samples are independently and identically distributed (i.i.d.). However, for many real applications, the i.i.d. assumption does not hold. We consider learning problems in which examples are dependent and their dependency relation is characterized by a graph. To establish algorithm-dependent generalization theory for learning with non-i.i.d. data, we first prove novel McDiarmid-type concentration inequalities for Lipschitz functions of graph-dependent random variables. We show that concentration relies on the forest complexity of the graph, which characterizes the strength of the dependency. We demonstrate that for many types of dependent data, the forest complexity is small and thus implies good concentration. Based on our new inequalities we are able to build stability bounds for learning from graph-dependent data.

1 Introduction

Generalization theory is at the foundation of machine learning. It quantifies how accurate a model would predict on the test data which the learning algorithm is not able to access during training. It usually relies on a crucial assumption: The data are independently and identically distributed (i.i.d.). The i.i.d. assumption allows one to use many powerful tools from probability to prove strong generalization error bounds. However, in real applications, the data are often non-i.i.d. i.e., the data collected can be dependent. There have been extensive discussions on why and how the data are dependent. We refer the readers to [1, 2].

Establishing generalization theory for dependent data has received a lot of attention [3, 4, 5, 6, 7]. A major line of research in this direction models the data dependency by various types of mixing such as α-mixing [8], β-mixing [9], ϕ-mixing [10], η-mixing [11], etc. Mixing models have been used in statistical learning theory to establish generalization error bounds based on Rademacher

*This work was done when this author was a master student at the Institute of Computing Technology, Chinese Academy of Sciences and University of Chinese Academy of Sciences. This research forms part of Rui (Ray) Zhang's master thesis submitted to the University of Chinese Academy of Sciences in May 2019.

†Corresponding author

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

complexity [4, 6, 12] or algorithmic stability [3, 12, 13] via concentration results [14] or independent blocking technique [15]. In these models, the mixing coefficients measure the extent to which the data are dependent to each other. Similar to the mixing models, learning under Dobrushin's condition [16] is also investigated via concentration results [17, 18, 19] using Dobrushin's interaction matrix [20]. Although the results under the various mixing conditions and Dobrushin's condition are fruitful, they are faced with difficulties in application: It is sometimes difficult to determine the quantitative dependency among data points. On the other hand, determining whether two data are dependent or not is often much easier. In this paper, we focus on such qualitative dependency of data. We use simple graphs as a natural tool to describe the dependency among data, and establish generalization theory for such graph-dependent data.

A basic building block of generalization theory is concentration inequality. Different settings and different assumptions require different concentration tools. The less we assume, the more powerful tools we need. In order to establish generalization theory for dependent data, standard concentration for i.i.d. data no longer applies. One must develop concentration inequalities for dependent data, which is a very challenging task.

In his seminal work [21], Janson proved an elegant concentration inequality for graph-dependent data. The inequality is a beautiful extension of Hoeffding inequality. It bounds the probability that the summation of graph-dependent random variables deviates from its expected value, in terms of the fractional coloring number of the dependency graph. Janson's inquality has been extended to any functions that can be decomposed into the summation of some functions of independent random variables [22]. This extension enables to establish generalization error bounds for graph-dependent data via fractional Rademacher complexity.

In [5], PAC-Bayes bounds for classification with non-i.i.d. data are obtained based on fractional colorings of graphs. The results also hold for specific learning settings such as ranking and learning from stationary β-mixing distributions. In [23], Ralaivola and Amini established new concentration inequalities for fractionally sub-additive and fractionally self-bounding functions of dependent variables. Their results are based on the fractional chromatic numbes and the entropy method. In [24], Wang et al. used hypergraphs to model dependent random variables that are generated by independent ones. Leveraging the notion of fractional matching, they also establish concentration inequalities of Hoeffding- or Bernstein-type.

Though fundamental and elegant, the above generalization bounds are algorithm-independent. They considered the complexity of the hypothesis space and data distribution, but does not involve the learning algorithm. To derive better generalization bounds, there are growing interests in developing algorithm-dependent generalization theories. This line of research heavily relies on the algorithmic stability. A key advantage of stability bounds is that they are tailored to specific learning algorithms, exploiting their particular properties.

How can we establish algorithmic stability theory for graph-dependent data? Note that under the assumption of i.i.d. data, Hoeffding-type concentration inequality, which bounds the deviation of sample average from expectation, is not strong enough to prove stability-based generalization. On the contrary, McDiarmids inequality characterizes the concentration of general Lipschitz functions of i.i.d. random variables, hence serving as the key tool for proving the stability theory. Therefore, to build algorithmic stability theory for non-i.i.d. samples, one has to develop McDiarmid-type concentration for graph-dependent random variables.

In this paper, we prove the first McDiarmid-type concentration inequality for graph-dependent random variables in terms of a new notion called forest complexity, which measures the strength of the dependency. It turns out that for various dependency graphs, it is easy to estimate the forest complexity. The proposed concentration inequality enables us to prove stability-based generalization bounds for graph-dependent data. Our results provide basic tools for understanding learning with overparameterized models.

The rest of the paper is organized as follows. In section 2, we briefly introduce the notations and related results. In section 3, we establish McDiarmid-type inequalities for acyclic dependency graphs, and extend the concentration results to the general dependency graphs. In section 4, we apply our concentration results to the learning theory and establish generalization error bounds for learning graph-dependent data via algorithmic stability, we also provide an application of learning m-dependent data. Section 5 concludes the paper and points out the future research directions.

2 Preliminaries

In this section, we present the notations and the basic McDiarmid's inequality for i.i.d. random variables.

Throughout this paper, let n be a positive integer with $[n]$ standing for the set $\{1, 2, \ldots, n\}$. Let Ω_i be a Polish space for any $i \in [n]$, $\mathbf{\Omega} = \prod_{i\in[n]} \Omega_i$ be the product space, $\mathbb{R}$ be the set of real numbers, $\mathbb{R}_+$ be the set of non-negative real numbers, $\mathbb{N}_+$ be the set of non-negative integers.

Concentration inequalities are fundamental tools in statistical learning theory. They are essentially tail probability bounds indicating how much a function of random variables deviates from some value that is usually the expectation. Among the most powerful ones is the McDiarmid's inequality which establishes a sharp, even tight in some cases, bound on the concentration, when the function satisfies $\mathbf{c}$-Lipschitz condition (bounded differences condition), namely, does not depend too much on any individual variable.

Definition 2.1 ($\mathbf{c}$-Lipschitz)**.** *Given a vector $\mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{R}^n_+$, a function $f : \mathbf{\Omega} \to \mathbb{R}$ is said to be $\mathbf{c}$-Lipschitz if for any $\mathbf{x} = (x_1, \ldots, x_n), \mathbf{x}' = (x'_1, \ldots, x'_n) \in \mathbf{\Omega}$, it satisfies*

$$|f(\mathbf{x}) - f(\mathbf{x}')| \leq \sum_{i=1}^{n} c_i \mathbf{1}_{\{x_i \neq x'_i\}},$$

where c_i is called the i-th Lipschitz coefficient of f.

Theorem 2.2 (McDiarmid's inequality [25])**.** *Suppose $f : \mathbf{\Omega} \to \mathbb{R}$ is $\mathbf{c}$-Lipschitz, and $\mathbf{X} = (X_1, \ldots, X_n)$ is a vector of independent random variables with each X_i taking values in Ω_i. Then for any $t > 0$, the tail probability satisfies*

$$\mathbf{Pr}\left(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t\right) \leq \exp\left(-\frac{2t^2}{\|\mathbf{c}\|_2^2}\right). \tag{1}$$

Notice that the McDiarmid's inequality works for independent random variables. Janson's Hoeffding-type inequality [21] for graph-dependent random variables is a special case of McDiarmid-type inequality when the function is a summation. Specifically, when $f(\mathbf{X}) = \sum_{i=1}^n X_i$ with each X_i ranging over an interval of length c_i,

$$\mathbf{Pr}\left(\sum_{i=1}^{n} X_i - \mathbf{E}\left[\sum_{i=1}^{n} X_i\right] \geq t\right) \leq \exp\left(-\frac{2t^2}{\chi^*(G)\|\mathbf{c}\|_2^2}\right), \tag{2}$$

where $\mathbf{c} = (c_1, \ldots, c_n)$ and $\chi^*(G)$ is the fractional coloring number of a dependency graph G of random variables $\mathbf{X}$.

3 McDiarmid Concentration for Graph-dependent Random Variables

In this section we present our first set of main results, the McDiarmid-type concentration inequalities (i.e., concentration of Lipschitz functions) for graph-dependent random variables. The results in this section will serve as the tools for developing learning theory for dependent data.

We start from the simplest case that the dependency graph is acyclic, i.e., trees or forests. We prove McDiarmid-type concentration bounds for trees and forests with very simple forms. These inequalities are then extended to general graphs. To this end, we introduce the notion of forest complexity, which characterizes to what extent a general graph can be best approximated by a forest. We prove McDiarmid-type concentration inequality for general graph-dependent random variables in terms of the forest complexity. Finally we demonstrate that for many important classes of graphs, forest complexity is easy to estimate.

Below we first define the notion of dependency graphs, which is a widely used model in probability, statistics, and combinatorics, see [26, 27, 28, 29, 30] for examples.

Definition 3.1 (Dependency Graphs)**.** *An undirected graph G is called a dependency graph of a random vector $\mathbf{X} = (X_1, \ldots, X_n)$ if*

1. *$V(G) = [n]$*
2. *if $I, J \subset [n]$ are non-adjacent in G, then $\{X_i\}_{i\in I}$ and $\{X_j\}_{j\in J}$ are independent.*

3.1 McDiarmid Concentration for Acyclic Graph-dependent Variables

Our first result is for the case that the dependency graph is a tree.

Theorem 3.2. *Suppose that $f : \mathbf{\Omega} \to \mathbb{R}$ is a $\mathbf{c}$-Lipschitz function and G is a dependency graph of a random vector $\mathbf{X}$ that takes values in $\mathbf{\Omega}$. If G is a tree, then for any $t > 0$, the following inequality holds:*

$$\mathbf{Pr}(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{\langle i,j\rangle \in E(G)}(c_i + c_j)^2 + c_{\min}^2}\right), \tag{3}$$

where $c_{\min}$ is the minimum entry in $\mathbf{c}$.

The proof of this theorem relies on decomposing $f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})]$ into the summation $\sum_{i=1}^{n} V_i$ with $V_i := \mathbf{E}[f(\mathbf{X})|X_1, \ldots X_i] - \mathbf{E}[f(\mathbf{X})|X_1, \ldots X_{i-1}]$. We show that each V_i ranges in an interval of length at most $c_i + c_j$, where j is the parent of i in the tree (in the proof, we make the tree rooted by choosing the vertex with the minimum Lipschitz coefficient as the root). The theorem is then proved by applying the Chernoff-Cramér technique to $\sum_{i=1}^{n} V_i$. For details, please refer to Subsection A.1 in the supplementary materials.

Like McDiarmid's inequality, Theorem 3.2 also claims a deviation probability bound that decays exponentially. The decay rate is determined by two interplaying factors. One is the Lipschitz coefficient that is inherent to the function. The other is the pattern of the dependency, namely, which random variables are dependent and connected by an edge.

We then generalize the above result to the case where dependency graph G is a forest.

Theorem 3.3. *Suppose that $f : \mathbf{\Omega} \to \mathbb{R}$ is a $\mathbf{c}$-Lipschitz function and G is a dependency graph of a random vector $\mathbf{X}$ that takes values in $\mathbf{\Omega}$. If G is a forest consisting of trees $\{T_i\}_{i\in[k]}$, then for any $t > 0$, the following inequality holds:*

$$\mathbf{Pr}(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{\langle i,j\rangle \in E(G)}(c_i + c_j)^2 + \sum_{i=1}^{k} c_{\min,i}^2}\right), \tag{4}$$

where $c_{\min,i} = \min\{c_j : j \in V(T_i)\}$.

Theorem 3.3 can be proved in a similar way as Theorem 3.2. The detailed proof is presented in Subsection A.2 of the supplementary materials.

We point out that Theorem 3.3 is a strict generalization of the McDiarmid's inequality for i.i.d. random variables. If all the random variables are independent, i.e., there is no edge in the dependency graph, then it is clear that Eq. (4) degenerates exactly to Eq. (1).

Theorem 3.3 also clearly demonstrates how dependency between random variables affects concentration. The decay rate of the probability that $f(\mathbf{X})$ deviates from its expectation is approximately reversely proportional to the number of edges in the dependency graph.

3.2 McDiarmid Concentration for General Graphs

In this subsection, we consider general graphs. Our basic idea for handling general graphs is to use a forest to approximate the graph. Specifically, we partition the variables into groups so that the dependency graph of these groups is a forest. We try to find the optimal forest approximation, which leads to the notion of forest complexity. We then prove McDiarmid-type concentration inequality for general graph-dependent random variables in terms of its forest complexity, which yields a very simple form.

We first define the concept of forest approximation.

Definition 3.4 (Forest Approximation)**.** *Given a graph G, a forest F, and a mapping $\phi : V(G) \to V(F)$, if $\phi(u) = \phi(v)$ or $\langle \phi(u), \phi(v)\rangle \in E(F)$ for any $\langle u, v\rangle \in E(G)$, we say that (ϕ, F) is a forest approximation of G. Let $\Phi(G)$ denote the set of forest approximations of G.*

Intuitively, a forest approximation is transforming a graph into a forest by merging vertices and removing the incurred self-loops and multi-edges. In this way, we rule out the redundant variables that heavily depend on others and thus contribute little to concentration.

Based on forest approximation, we define the notion of forest complexity of a graph, which intuitively measures how much the graph looks like a forest.

Definition 3.5 (Forest Complexity)**.** *Given a graph G and any forest approximation $(\phi, F) \in \Phi(G)$ with F consisting of trees $\{T_i\}_{i\in[k]}$, let*

$$\lambda_{(\phi,F)} = \sum_{\langle u,v\rangle\in E(F)} \left(|\phi^{-1}(u)| + |\phi^{-1}(v)|\right)^2 + \sum_{i=1}^{k} \min_{u\in V(T_i)} |\phi^{-1}(u)|^2.$$

We call

$$\Lambda(G) = \min_{(\phi,F)\in\Phi(G)} \lambda_{(\phi,F)}$$

the forest complexity of the graph G.

Now we are ready to state our McDiarmid-type concentration inequality for general graph-dependent random variables.

Theorem 3.6. *Suppose that $f : \mathbf{\Omega} \to \mathbb{R}$ is a $\mathbf{c}$-Lipschitz function and G is a dependency graph of a random vector $\mathbf{X}$ that takes values in $\mathbf{\Omega}$. For any $t > 0$, the following inequality holds:*

$$\mathbf{Pr}(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t) \leq \exp\left(-\frac{2t^2}{\Lambda(G)\|\mathbf{c}\|_\infty^2}\right).$$

With the tool of forest approximation, we reduce the concentration problem defined on graphs to that defined on forests. Basically, we use a new variable to represent each set of the original variables that are merged together by the forest approximation. The function can be equivalently transformed into a function of the new variables whose dependency graph is the forest. The proof is done by applying Theorem 3.3 to the new function. For details, please refer to Subsection A.3 in the supplementary materials.

Like the above theorems, Theorem 3.6 also establishes an exponentially decaying probability of deviation. The decay rate is totally determined by the Lipschitz coefficient of the function and the forest complexity of the variables' dependency graph. Intuitively, the more the dependency graph looks like a forest, the faster the deviation probability decays. This uncovers how the dependencies among random variables influence concentration.

3.3 Illustrations and Examples

This subsection consists of two parts. In the first part we review a widely-studied random process that generates dependent data whose dependency graph can be naturally constructed. In the second part, we deal with some dependency graphs to show that in many cases, the forest complexity is small and easy to estimate.

Consider a data generating procedure modeled by the *spatial Poisson point process*, which is a Poisson point process on $\mathbb{R}^2$ (See [31, 32] for discussions of using this process to model data collection in various machine learning applications.) The number of points in each finite region follows a Poisson distribution, and the number of points in disjoint regions are independent. Given a finite set $\mathcal{I} = \{I_i\}_{i=1}^n$ of regions in $\mathbb{R}^2$, let X_i be the number of points in region I_i, $1 \leq i \leq n$. Then the graph $G\left([n], \{\langle i, j\rangle : I_i \cap I_j \neq \emptyset\}\right)$ is a dependency graph of the random variables $\{X_i\}_{i=1}^n$.

We present three examples to demonstrate that estimating the forest complexity $\Lambda(G)$ is usually easy. All the examples can naturally appear in the above process.

Example 3.7 (G is a tree)**.** In this case, the identity map between G and itself is a forest approximation of G. Then $\Lambda(G) \leq |E(G)|(1+1)^2 + 1 = 4n - 3 = O(n)$. We get an upper bound of $\Lambda(G)$ that is linear in the number of variables, which is almost tight compared with Hoeffding's inequality or Janson's result (see (2) with $\chi^*(G) = 2$).

Example 3.8 (G is a cycle C_n)**.** If n is even, a forest approximation is illustrated in Figure 1, where the cycle is approximated by a path F of length $\frac{n}{2}$. The approximation ϕ maps any vertex of G to the vertex of F having the same shape, so each gray belt stands for a preimage set of ϕ. We will keep this convention in the rest of this section. By the illustrated forest approximation,

$\Lambda(G) \leq 2 \times (1+2)^2 + (\frac{n}{2} - 2)(2+2)^2 + 1 = 8n - 13 = O(n)$. When n is odd, according to the forest approximation shown in Figure 2, $\Lambda(G) \leq (1+2)^2 + (\frac{n-1}{2} - 1)(2+2)^2 + 1 = 8n - 14 = O(n)$. Since $\chi^*(G)$ is 2 or 3, our bound is again very tight compared with Jansons result.

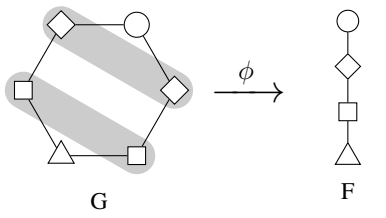

Figure 1: A forest approximation of C_6

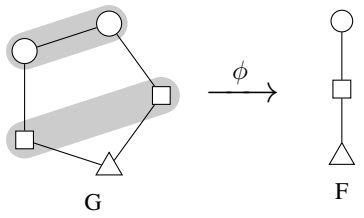

Figure 2: A forest approximation of C_5

Example 3.9 (G is a grid)**.** Suppose G is a two-dimensional $(m \times m)$-grid. Then $n = m^2$. Considering the forest approximation illustrated in Figure 3, $\Lambda(G) \leq 2[3^2 + 5^2 + \ldots + (2m-1)^2] + 1 = \frac{2m(2m+1)(2m-1)-3}{3} = O(m^3) = O(n^{\frac{3}{2}})$

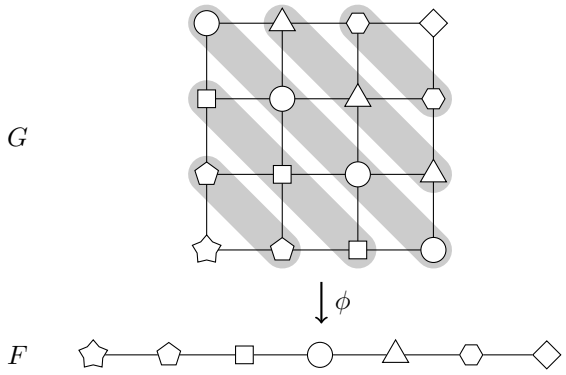

Figure 3: A forest approximation of the (4×4)-gird

4 Generalization Theory for Learning from Graph-Dependent Data

This section establishes stability generalization error bounds for learning from graph-dependent data, using the concentration inequalities derived in the last section.

Consider the supervised learning setting: Let $\mathbf{S} = ((x_1, y_1), \ldots, (x_n, y_n)) \in (\mathcal{X} \times \mathcal{Y})^n$ be a training sample of size n, where $\mathcal{X}$ is the input space and $\mathcal{Y}$ is the output space. Let D be the underlying distribution of data on $\mathcal{X} \times \mathcal{Y}$. Assume that all the training data points (x_i, y_i)'s have the same marginal distribution D and that G is a dependency graph of $\mathbf{S}$.

Throughout this section, fix a non-negative loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. For any hypothesis $f : \mathcal{X} \to \mathcal{Y}$, the empirical error on sample $\mathbf{S}$ is

$$\widehat{R}(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f(x_i)).$$

For learning from dependent data, the generalization error can be defined in various ways. We adopt the following widely-used one [33, 34, 35, 36]

$$R(f) = \mathbf{E}_{(x,y)\sim D}[\ell(y, f(x))], \tag{5}$$

which assumes that the test set is independent of the training set.

4.1 Bounding Generalization Error via Algorithmic Stability

Algorithmic stability has been used in the study of classification and regression to derive generalization bounds [37, 38, 39, 40, 41, 42]. A key advantage of stability bounds is that they are designed for

specific learning algorithms, exploiting particular properties of the algorithms. Introduced 17 years ago, uniform stability [43] is now among the most widely used notions of algorithmic stability.

Given a training sample $\mathbf{S}$ of size n and $i \in [n]$, remove the i-th element from $\mathbf{S}$, resulting in a sample of size $n-1$, which is denoted by $\mathbf{S}^{\setminus i} = ((x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}) \ldots, (x_n, y_n))$. For a learning algorithm $\mathcal{A}$, define $f_{\mathbf{S}}^{\mathcal{A}} : \mathcal{X} \to \mathcal{Y}$ to be the the hypothesis that $\mathcal{A}$ has learned from the sample $\mathbf{S}$.

Definition 4.1 (Uniform Stability [43])**.** *Given integer $n > 0$, the learning algorithm $\mathcal{A}$ is called β_n-uniformly stable with respect to the loss function ℓ, if for any $i \in [n]$, $\mathbf{S} \in (\mathcal{X} \times \mathcal{Y})^n$, and $(x, y) \in \mathcal{X} \times \mathcal{Y}$, it holds that*

$$|\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x))| \leq \beta_n.$$

Intuitively, the stability of a leaning algorithm means that any small perturbation of training samples has little effect on the result of learning.

Now, we begin our analysis with studying the distribution of $\Phi_{\mathcal{A}}(\mathbf{S}) = R(f_{\mathbf{S}}^{\mathcal{A}}) - \widehat{R}(f_{\mathbf{S}}^{\mathcal{A}})$, namely, the difference between the empirical and the generalization errors. The mapping $\Phi_{\mathcal{A}} : (\mathcal{X} \times \mathcal{Y})^n \to \mathbb{R}$ will play a critical role in estimating $R(f_{\mathbf{S}}^{\mathcal{A}})$ via stability. We first show that the deviation of $\Phi_{\mathcal{A}}(\mathbf{S})$ from its expectation can be bounded with high probability (Lemma 4.2), and then upper bound the expected value of $\Phi_{\mathcal{A}}(\mathbf{S})$ in Lemma 4.3.

Lemma 4.2. *Given a sample $\mathbf{S}$ of size n with dependency graph G, assume that the learning algorithm $\mathcal{A}$ is β_n-uniformly stable. Suppose the loss function ℓ is bounded by M. Then for any $t > 0$, it holds that*

$$\mathbf{Pr}(\Phi_{\mathcal{A}}(\mathbf{S}) - \mathbf{E}[\Phi_{\mathcal{A}}(\mathbf{S})] \geq t) \leq \exp\left(-\frac{2n^2t^2}{\Lambda(G)(4n\beta_n + M)^2}\right).$$

Lemma 4.2 is proved in two steps. First, we treat $\Phi_{\mathcal{A}}(\cdot)$ as an n-ary function and show that its Lipschitz coefficients are all bounded by $4\beta_n + M/n$. Second, regarding $\mathbf{S}$ as a random vector, we apply Theorem 3.6 to $\Phi_{\mathcal{A}}(\mathbf{S})$. For detail, see Subsection B.1 of the supplementary materials.

Lemma 4.3. *Given a sample $\mathbf{S}$ of size n with dependency graph G, assume that the learning algorithm $\mathcal{A}$ is β_i-uniformly stable for any $i \leq n$. Suppose the maximum degree of G is Δ. Let $\beta_{n,\Delta} = \max_{i \in [0,\Delta]} \beta_{n-i}$. It holds that*

$$\mathbf{E}[\Phi_{\mathcal{A}}(\mathbf{S})] \leq 2\beta_{n,\Delta}(\Delta + 1).$$

The proof of the lemma is based on iterative perturbations on the training sample $\mathbf{S}$. A perturbation is essentially removing a data point from or adding a data point to $\mathbf{S}$. The property of uniform stability of the algorithm guarantees that each perturbation causes a discrepancy up to $\beta_{n,\Delta}$, and in total $2(\Delta + 1)$ perturbations have to be made in order to *eliminate* the dependency between a data point and the others. For detail, please refer to Subsection B.2 of the supplementary materials.

Combining Lemma 4.2 and Lemma 4.3, we immediately have

Theorem 4.4. *Given a sample $\mathbf{S}$ of size n with dependency graph G, assume that the learning algorithm $\mathcal{A}$ is β_i-uniformly stable for any $i \leq n$. Suppose the maximum degree G is Δ, and the loss function ℓ is bounded by M. Let $\beta_{n,\Delta} = \max_{i \in [0,\Delta]} \beta_{n-i}$. For any $\delta \in (0, 1)$, with probability at least $1 - \delta$, it holds that*

$$R(f_{\mathbf{S}}^{\mathcal{A}}) \leq \widehat{R}(f_{\mathbf{S}}^{\mathcal{A}}) + 2\beta_{n,\Delta}(\Delta + 1) + \frac{4n\beta_n + M}{n}\sqrt{\frac{\Lambda(G)\ln(1/\delta)}{2}}.$$

Remark 4.5. It is well known that for many learning algorithms $\beta_n = O(1/n)$ [43]. Thus, we often have $\beta_{n,\Delta}(\Delta + 1) \leq \beta_{n-\Delta}(\Delta + 1) = O(\frac{\Delta}{n-\Delta})$, which vanishes asymptotically if $\Delta = o(n)$. The term $O\left(\sqrt{\Lambda(G)}/n\right)$ also vanishes asymptotically if $\Lambda(G) = o(n^2)$. As a result, in case of *weak* dependence such as the examples in Subsection 3.3, the generalization error is almost upper-bounded by the empirical error. We also observe that if the training data are i.i.d., Theorem 4.4 degenerates to the standard stability bound in [43], by applying $\Delta = 0$, $\beta_{n,\Delta} = \beta_n$, $\Lambda(G) = n$.

4.2 Application: Learning from m-dependent data

We present a practical application in machine learning. Suppose there are linearly aligned locations, for example, real estates along a street. Let y_i be the observation at location i, e.g., the house price, and x_i stand for the random variable modeling geographical effect at location i. Suppose that x's are mutually independent and each y_i is geographically influenced by a neighborhood of size at most $2q+1$. One hope to learn the model of y from a sample $\{((x_{i-q},\ldots,x_i,\ldots,x_{i+q}),y_i)\}_{i\in[n]}$, where n is the size of the sample. This model accounts for the impact of local locations on house prices. Similar scenarios are frequently considered in spatial econometrics, see [44] for more examples.

This application is a special case of m-dependence, which is an important statistical model introduced by Hoeffding in [45]. m-dependence has been studied extensively in probability, statistics, and combinatorics [46, 47, 48].

Definition 4.6 (m-dependence [45])**.** *For some $m,n\in\mathbb{N}_+$, a sequence of random variables $\{X_i\}_{i=1}^n$ is called m-dependent if for any $i\in[n-m-1]$, $\{X_j\}_{j=1}^i$ is independent of $\{X_j\}_{j=i+m+1}^n$.*

The upper part of Figure 4 illustrates a dependency graph of 2-dependent sequence $\{X_i\}_{i=1}^n$.

As illustrated in Figure 4, we divide an m-dependent sequence into blocks of size m, and sequentially map the blocks to vertices of a path of length $\lceil\frac{n}{m}\rceil$. This forest approximation leads to

$$\Lambda(G)\le\left(\left\lceil\frac{n}{m}\right\rceil-1\right)(m+m)^2+m^2\le 4mn=O(mn)$$

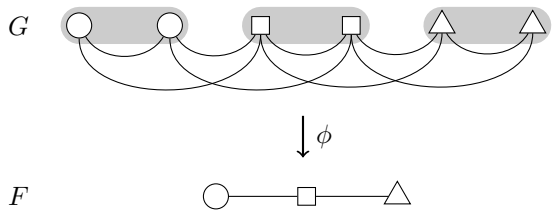

Figure 4: A forest approximation of a 2-dependent sequence. The approximation ϕ maps any vertex of G to the vertex of F having the same shape, so each gray belt stands for a pre-image set of ϕ.

Combining Theorem 4.4 and the estimated forest complexity, we have

Corollary 4.7. *Given an m-dependent sequence $\mathbf{S}$ of length n as training sample, assume that the learning algorithm $\mathcal{A}$ is β_i-uniformly stable for any $i\le n$. Suppose the loss function ℓ is bounded by M. For any $\delta\in(0,1)$, with probability at least $1-\delta$, it holds that*

$$R(f_{\mathbf{S}}^{\mathcal{A}})\le\widehat{R}(f_{\mathbf{S}}^{\mathcal{A}})+2\beta_{n,2m}(2m+1)+(4n\beta_n+M)\sqrt{\frac{2m\ln(1/\delta)}{n}}.$$

Choose any uniformly stable learning algorithm $\mathcal{A}$ in [43] with $\beta_n=O(1/n)$, such as regularization algorithms in RKHS. Apply it to the above mentioned house price prediction problem. Then for any fixed q, with high probability, Corollary 4.7 leads to $R(f_{\mathbf{S}}^{\mathcal{A}})\le\widehat{R}(f_{\mathbf{S}}^{\mathcal{A}})+O\left(\sqrt{\frac{\ln(1/\delta)}{n}}\right)$ for sufficiently large n, matching the stability bound of the i.i.d. case in [43].

5 Conclusion and Future Work

In this paper, we establish McDiarmid-type concentration inequalities for general functions of graph-dependent random variables. We apply our concentration results to obtain a stability-based generalization error bound for learning from graph-dependent samples. There are several possible extensions of this work.

- We provide upper bounds of the forest complexity for several classes of graphs. It is an interesting algorithmic problem to efficiently estimate the forest complexity. One heuristic method to do this on a connected graph is via graph diameter, by merging vertices of the same distances to a peripheral vertex, resulting in a path as long as the diameter. Can the problem be solved approximately?
- If more information of the dependency structure is known, e.g., a dependency hypergraph [24], can we obtain better McDiarmid-type inequalities and tighter generalization bounds?
- In [3, 12, 6], generalization error is defined different from that in this paper. The relationship between these two definitions has been discussed in [3, 12]. It is a natural question whether our results can be adapted to that definition.
- There are some newly introduced dependency graph models such as thresholded dependency graphs [49] and weighted dependency graphs [50, 51]. Can the problem in this paper be solved under these new models?

Acknowledgments

Rui (Ray) Zhang would like to thank Nick Wormald for valuable comments on an early version of this paper. Yuyi Wang would like to thank Dr. Ondřej Kuželka for very helpful discussions. Liwei Wang would like to thank Yunchang Yang for very helpful discussions. Xingwu Liu's work is partially supported by the National Key Research and Development Program of China (Grant No. 2016YFB1000201), the National Natural Science Foundation of China (61420106013), State Key Laboratory of Computer Architecture Open Fund (CARCH3410), and Youth Innovation Promotion Association of Chinese Academy of Sciences.

References

[1] Herold Dehling and Walter Philipp. Empirical process techniques for dependent data. In *Empirical process techniques for dependent data*, pages 3–113. Springer, 2002.

[2] Massih-Reza Amini and Nicolas Usunier. *Learning with Partially Labeled and Interdependent Data*. Springer, 2015.

[3] Mehryar Mohri and Afshin Rostamizadeh. Stability bounds for non-iid processes. In *Advances in Neural Information Processing Systems*, pages 1025–1032, 2008.

[4] Mehryar Mohri and Afshin Rostamizadeh. Rademacher complexity bounds for non-iid processes. In *Advances in Neural Information Processing Systems*, pages 1097–1104, 2009.

[5] Liva Ralaivola, Marie Szafranski, and Guillaume Stempfel. Chromatic pac-bayes bounds for non-iid data: Applications to ranking and stationary β-mixing processes. *Journal of Machine Learning Research*, 11(Jul):1927–1956, 2010.

[6] Vitaly Kuznetsov and Mehryar Mohri. Generalization bounds for non-stationary mixing processes. *Machine Learning*, 106(1):93–117, 2017.

[7] Hao Yi, Alon Orlitsky, and Venkatadheeraj Pichapati. On learning markov chains. In *Advances in Neural Information Processing Systems*, pages 646–655, 2018.

[8] Murray Rosenblatt. A central limit theorem and a strong mixing condition. *Proceedings of the National Academy of Sciences of the United States of America*, 42(1):43, 1956.

[9] VA Volkonskii and Yu A Rozanov. Some limit theorems for random functions. i. *Theory of Probability & Its Applications*, 4(2):178–197, 1959.

[10] Ildar A Ibragimov. Some limit theorems for stationary processes. *Theory of Probability & Its Applications*, 7(4):349–382, 1962.

[11] Leonid Kontorovich. *Measure concentration of strongly mixing processes with applications*. Carnegie Mellon University, 2007.

[12] Mehryar Mohri and Afshin Rostamizadeh. Stability bounds for stationary φ-mixing and β-mixing processes. *Journal of Machine Learning Research*, 11(Feb):789–814, 2010.

[13] Fangchao He, Ling Zuo, and Hong Chen. Stability analysis for ranking with stationary φ-mixing samples. *Neurocomputing*, 171:1556–1562, 2016.

[14] Leonid Aryeh Kontorovich, Kavita Ramanan, et al. Concentration inequalities for dependent random variables via the martingale method. *The Annals of Probability*, 36(6):2126–2158, 2008.

[15] Bin Yu. Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, pages 94–116, 1994.

[16] Yuval Dagan, Constantinos Daskalakis, Nishanth Dikkala, and Siddhartha Jayanti. Learning from weakly dependent data under dobrushins condition. In *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 914–928, Phoenix, USA, 25–28 Jun 2019. PMLR.

[17] Christof Külske. Concentration inequalities for functions of gibbs fields with application to diffraction and random gibbs measures. *Communications in mathematical physics*, 239(1-2):29–51, 2003.

[18] Sourav Chatterjee. Concentration inequalities with exchangeable pairs (Ph. D. thesis). *arXiv preprint math/0507526*, 2005.

[19] Aryeh Kontorovich and Maxim Raginsky. Concentration of measure without independence: a unified approach via the martingale method. In *Convexity and Concentration*, pages 183–210. Springer, 2017.

[20] PL Dobruschin. The description of a random field by means of conditional probabilities and conditions of its regularity. *Theory of Probability & Its Applications*, 13(2):197–224, 1968.

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

[22] Nicolas Usunier, Massih-Reza Amini, and Patrick Gallinari. Generalization error bounds for classifiers trained with interdependent data. In *Advances in neural information processing systems*, pages 1369–1376, 2006.

[23] Liva Ralaivola and Massih-Reza Amini. Entropy-based concentration inequalities for dependent variables. In *International Conference on Machine Learning*, pages 2436–2444, 2015.

[24] Yuyi Wang, Zheng-Chu Guo, and Jan Ramon. Learning from networked examples. In *International Conference on Algorithmic Learning Theory, ALT 2017, 15-17 October 2017, Kyoto University, Kyoto, Japan*, pages 641–666, 2017.

[25] Colin McDiarmid. On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188, 1989.

[26] Paul Erdos and László Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. *Infinite and finite sets*, 10(2):609–627, 1975.

[27] Svante Janson, Tomasz Luczak, and Andrzej Rucinski. *An exponential bound for the probability of nonexistence of a specified subgraph in a random graph*. Institute for Mathematics and its Applications (USA), 1988.

[28] Louis HY Chen. Two central limit problems for dependent random variables. *Probability Theory and Related Fields*, 43(3):223–243, 1978.

[29] Pierre Baldi, Yosef Rinott, et al. On normal approximations of distributions in terms of dependency graphs. *The Annals of Probability*, 17(4):1646–1650, 1989.

[30] Svante Janson, Tomasz Luczak, and Andrzej Rucinski. *Random graphs*, volume 45. John Wiley & Sons, 2011.

[31] Scott Linderman and Ryan Adams. Discovering latent network structure in point process data. In *International Conference on Machine Learning*, pages 1413–1421, 2014.

[32] Alisa Kirichenko and Harry Van Zanten. Optimality of poisson processes intensity learning with gaussian processes. *The Journal of Machine Learning Research*, 16(1):2909–2919, 2015.

[33] Ron Meir. Nonparametric time series prediction through adaptive model selection. *Machine learning*, 39(1):5–34, 2000.

[34] Aurélie C Lozano, Sanjeev R Kulkarni, and Robert E Schapire. Convergence and consistency of regularized boosting algorithms with stationary b-mixing observations. In *Advances in neural information processing systems*, pages 819–826, 2006.

[35] Ingo Steinwart and Andreas Christmann. Fast learning from non-iid observations. In *Advances in neural information processing systems*, pages 1768–1776, 2009.

[36] Hanyuan Hang and Ingo Steinwart. Fast learning from α-mixing observations. *Journal of Multivariate Analysis*, 127:184–199, 2014.

[37] William H Rogers and Terry J Wagner. A finite sample distribution-free performance bound for local discrimination rules. *The Annals of Statistics*, pages 506–514, 1978.

[38] Luc Devroye and Terry Wagner. Distribution-free performance bounds for potential function rules. *IEEE Transactions on Information Theory*, 25(5):601–604, 1979.

[39] Michael Kearns and Dana Ron. Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural computation*, 11(6):1427–1453, 1999.

[40] Samuel Kutin and Partha Niyogi. Almost-everywhere algorithmic stability and generalization error. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 275–282. Morgan Kaufmann Publishers Inc., 2002.

[41] Wenlong Mou, Yuchen Zhou, Jun Gao, and Liwei Wang. Dropout training, data-dependent regularization, and generalization bounds. In *International Conference on Machine Learning*, pages 3642–3650, 2018.

[42] Wenlong Mou, Liwei Wang, Xiyu Zhai, and Kai Zheng. Generalization bounds of sgld for non-convex learning: Two theoretical viewpoints. *arXiv preprint arXiv:1707.05947*, 2017.

[43] Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of machine learning research*, 2(Mar):499–526, 2002.

[44] Luc Anselin. *Spatial econometrics: methods and models*, volume 4. Springer Science & Business Media, 2013.

[45] Wassily Hoeffding, Herbert Robbins, et al. The central limit theorem for dependent random variables. *Duke Mathematical Journal*, 15(3):773–780, 1948.

[46] PH Diananda and MS Bartlett. Some probability limit theorems with statistical applications. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 49, pages 239–246. Cambridge University Press, 1953.

[47] Pranab Kumar Sen. Asymptotic normality of sample quantiles for m-dependent processes. *The annals of mathematical statistics*, pages 1724–1730, 1968.

[48] Louis HY Chen and Qi-Man Shao. Steins method for normal approximation. *An introduction to Steins method*, 4:1–59, 2005.

[49] Christoph H Lampert, Liva Ralaivola, and Alexander Zimin. Dependency-dependent bounds for sums of dependent random variables. *arXiv preprint arXiv:1811.01404*, 2018.

[50] Jehanne Dousse and Valentin Féray. Weighted dependency graphs and the ising model. *arXiv preprint arXiv:1610.05082*, 2016.

[51] Valentin Féray et al. Weighted dependency graphs. *Electronic Journal of Probability*, 23, 2018.

A Omitted Proofs in Section 3

A.1 Proof of Theorem 3.2

Given a random vector $\mathbf{X} = (X_1, \ldots, X_n)$ taking values in a product space $\mathbf{\Omega} = \prod_{i\in[n]} \Omega_i$. For any set $S \subseteq [n]$, we denote $\mathbf{X}_S = \{X_i\}_{i\in S}$, and $\mathbf{\Omega}_S = \prod_{i\in S} \Omega_i$ for convenience. The proof of Theorem 3.2 will rest on Lemma A.1, which intuitively means that the small deviation of

$$\mathbf{E}[f(\mathbf{X})|X_1 = x_1, \ldots, X_{i-1} = x_{i-1}, X_i = x_i]$$

with respect to x_i for all i leads to a high concentration of $f(\mathbf{X})$ around its expectation. Our task is thus reduced to show that when $x_1, \ldots, x_{i-1}$ is fixed,

$$\mathbf{E}[f(\mathbf{X})|X_1 = x_1, \ldots, X_{i-1} = x_{i-1}, X_i = x_i] - \mathbf{E}[f(\mathbf{X})|X_1 = x_1, \ldots, X_{i-1} = x_{i-1}, X_i = x_i']$$

is small for any x_i and x_i'. This will be true due to Lemma A.4, if there is a good coupling, namely, jointly distributed variables $(\mathbf{Y}, \mathbf{Z})$ whose Hamming distance is small and whose marginals are the distributions of $\mathbf{X}$ conditional on $\{X_1 = x_1, \ldots, X_{i-1} = x_{i-1}, X_i = x_i\}$ and on $\{X_1 = x_1, \ldots, X_{i-1} = x_{i-1}, X_i = x_i'\}$, respectively. Hence, the main part of the proof is to construct such a coupling (Lemma A.3) whose feasibility relies on the strong independence among $\mathbf{X}$ (Lemma A.2). First of all, recall a lemma in literature.

Lemma A.1 ([25]). *If for any $j \in [n]$ and $\mathbf{y} \in \mathbf{\Omega}_{[j-1]}$, there is $b_j \geq 0$ such that*

$$\sup_{\xi\in\Omega_j} \mathbf{E}[f(\mathbf{X})|\mathbf{X}_{[j-1]} = \mathbf{y}, X_j = \xi] - \inf_{\xi\in\Omega_j} \mathbf{E}[f(\mathbf{X})|\mathbf{X}_{[j-1]} = \mathbf{y}, X_j = \xi] \leq b_j \tag{6}$$

then for any $t > 0$,

$$\mathbf{Pr}(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{j=1}^n b_j^2}\right).$$

By this lemma, it suffice to show that the small deviation of $\mathbf{E}[f(\mathbf{X})|X_1 = x_1, \ldots, X_{i-1} = x_{i-1}, X_i = x_i]$ with respect to x_i for all i is small to prove Theorem 3.2. Before continuing the proof, we assume that

Well-rooted: G is rooted at the vertex n and $c_n = c_{\min}$.

Well-sorted: For any $i, j \in V(G)$, j is a descendent of i only if $j < i$.

These assumptions will not lose generality, since we can relabel the variables $X_1, ..., X_n$ to meet the requirements.

For any non-root vertex $i \in V(G)$, let $p(i)$ be the parent vertex of i. For the rest of the section, arbitrarily fix $i \in [n]$ and define $S = [i+1, n] \setminus \{p(i)\}$, where $[j, k]$ stands for the set $\{j, \ldots, k\}$ of integers. Arbitrarily choose a vector $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbf{\Omega}$ and an element $x_i' \in \Omega_i$. Let $\mathbf{x}' = (x_1, \ldots, x_{i-1}, x_i', x_{i+1}, \ldots, x_n)$. We have the following technical lemma, indicating that $\mathbf{X}_S$ is independent of X_i if $\mathbf{X}_{[i-1]}$ is given.

Lemma A.2. *For any vector $\mathbf{y} \in \mathbf{\Omega}_S$,*

$$\mathbf{Pr}(\mathbf{X}_S = \mathbf{y}|\mathbf{X}_{[i]} = \mathbf{x}_{[i]}) = \mathbf{Pr}(\mathbf{X}_S = \mathbf{y}|\mathbf{X}_{[i]} = \mathbf{x}'_{[i]}).$$

Proof. Let T_i be the subtree of G that is rooted at i. Our basic idea is to prove the stronger property that $\mathbf{X}_S$ is independent of the other parts of $\mathbf{X}_{[i]}$ if $\mathbf{X}_{[i-1]\setminus V(T_i)}$ is given. Since $[i] = V(T_i) \bigcup([i-1] \setminus V(T_i))$, it suffices to show that $\mathbf{X}_{V(T_i)}$ is independent of $\{\mathbf{X}_S, \mathbf{X}_{[i-1]\setminus V(T_i)}\}$, which in turn is reduced to prove the following two claims due to the definition of the dependency graphs.

Claim 1 : $N_G^+(T_i) \bigcap([i-1] \setminus V(T_i)) = \emptyset$, where $N_G^+(T_i) = \bigcup_{k\in V(T_i)} N_G^+(k)$.

Proof of Claim 1: Arbitrarily choose $j \in [i-1] \setminus V(T_i)$. Suppose for contradiction that $j \in N_G(k)$ for some $k \in V(T_i)$, namely, j is either a child or the parent of k. Since $j \notin V(T_i)$, we must have $j = p(k)$ and $k = i$, which implies $j > i$ due to the Assumption Well-rooted. A contradiction is reached, so $N_G(T_i) \bigcap([i-1] \setminus V(T_i)) = \emptyset$. Because $N_G^+(T_i) \bigcap([i-1] \setminus V(T_i)) = N_G(T_i) \bigcap([i-1] \setminus V(T_i))$, Claim 1 holds.

Claim 2 : $N_G^+(T_i) \bigcap S = \emptyset$.

Proof of Claim 2: Arbitrarily choose $j \in S = \{i+1, \ldots, n\} \setminus N_G(i)$. One immediately has $j \notin N_G^+(i)$. Suppose for contradiction that $j \in N_G^+(T_i)$. Then $j \in N_G^+(k)$ for some descendent k of i, which means that either $j = i$ or j is a descendent of i. This in turn means that $j \leq i$ due to the Assumption Well-sorted. A contradiction is reached, so Claim 2 holds.

Since G is a dependency graph of $\mathbf{X}$, Claims 1 and 2 indicate that $\mathbf{X}_{V(T_i)}$ is independent of $\{\mathbf{X}_S, \mathbf{X}_{[i-1]\setminus V(T_i)}\}$. Then

$$\begin{aligned} & \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | X_j = x_j, j \in [i-1] \setminus V(T_i)) \\ = \; & \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | X_j = x_j, j \in ([i-1] \setminus V(T_i)) \bigcup V(T_i)) \\ = \; & \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | X_j = x_j, j \in [i]) \\ = \; & \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}_{[i]}). \end{aligned}$$

Likewise, we also have

$$\mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | X_j = x'_j, j \in [i-1] \setminus V(T_i)) = \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]})$$

Since $\mathbf{x}$ and $\mathbf{x}'$ differ only in the i-th entry,

$$\mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | X_j = x_j, j \in [i-1] \setminus V(T_i)) = \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | X_j = x'_j, j \in [i-1] \setminus V(T_i)).$$

As a result, $\mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}_{[i]}) = \mathbf{Pr}(\mathbf{X}_S = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]})$, this completes the proof of Lemma A.2. □

Then we construct the jointly-distributed random vectors $(\mathbf{Y}, \mathbf{Z}) \in \mathbf{\Omega}^2$ with respect to the fixed $i, \mathbf{x}$, and $\mathbf{x}'$. Specifically, $\mathbf{Y} = (Y_1, \ldots, Y_n)$ and $\mathbf{Z} = (Z_1, \ldots, Z_n)$ are defined as below.

1. $\mathbf{Y}_{[i]} = \mathbf{x}_{[i]}$
2. For any vector $\mathbf{y} \in \mathbf{\Omega}_{[i+1,n]}$,
$$\mathbf{Pr}(\mathbf{Y}_{[i+1,n]} = \mathbf{y}) = \mathbf{Pr}(\mathbf{X}_{[i+1,n]} = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}_{[i]})$$
3. $\mathbf{Z}_{[i]} = \mathbf{x'}_{[i]}, \mathbf{Z}_S = \mathbf{Y}_S$.
4. For any vector $\mathbf{z} \in \mathbf{\Omega}_S$ and element $z \in \Omega_{p(i)}$,
$$\mathbf{Pr}(Z_{p(i)} = z | \mathbf{Z}_S = \mathbf{z}) = \mathbf{Pr}(X_{p(i)} = z | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]}, \mathbf{X}_S = \mathbf{z})$$

The next lemma states that $(\mathbf{Y}, \mathbf{Z})$ has the desired marginal distribution.

Lemma A.3. *For any vector* $\mathbf{y} \in \mathbf{\Omega}_{[i+1,n]}$*, we have*

1. $\mathbf{Pr}(\mathbf{Y}_{[i+1,n]} = \mathbf{y}) = \mathbf{Pr}(\mathbf{X}_{[i+1,n]} = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}_{[i]})$,
2. $\mathbf{Pr}(\mathbf{Z}_{[i+1,n]} = \mathbf{y}) = \mathbf{Pr}(\mathbf{X}_{[i+1,n]} = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]})$.

Proof. (1) holds by the definition of $\mathbf{Y}$. To prove (2), arbitrarily choose $\mathbf{y} = (y_{i+1}, \ldots, y_n) \in \mathbf{\Omega}_{[i+1,n]}$. Then we have

$$\begin{aligned} \mathbf{Pr}(\mathbf{Z}_{[i+1,n]} = \mathbf{y}) = \; & \mathbf{Pr}(\mathbf{Z}_S = \mathbf{y}_S) \mathbf{Pr}(Z_{p(i)} = y_{p(i)} | \mathbf{Z}_S = \mathbf{y}_S) \\ = \; & \mathbf{Pr}(\mathbf{Y}_S = \mathbf{y}_S) \mathbf{Pr}(Z_{p(i)} = y_{p(i)} | \mathbf{Z}_S = \mathbf{y}_S) \\ = \; & \mathbf{Pr}(\mathbf{X}_S = \mathbf{y}_S | \mathbf{X}_{[i]} = \mathbf{x}_{[i]}) \mathbf{Pr}(X_{p(i)} = y_{p(i)} | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]}, \mathbf{X}_S = \mathbf{y}_S) \\ = \; & \mathbf{Pr}(\mathbf{X}_S = \mathbf{y}_S | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]}) \mathbf{Pr}(X_{p(i)} = y_{p(i)} | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]}, \mathbf{X}_S = \mathbf{y}_S) \\ = \; & \mathbf{Pr}(\mathbf{X}_{[i+1,n]} = \mathbf{y} | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]}). \end{aligned}$$

where the fourth equality is due to Lemma A.2. □

Lemma A.4. $\mathbf{E}[f(\mathbf{X}) | \mathbf{X}_{[i]} = \mathbf{x}_{[i]}] - \mathbf{E}[f(\mathbf{X}) | \mathbf{X}_{[i]} = \mathbf{x}'_{[i]}] \leq c_i + c_{p(i)}$.

Proof. By the definition of random vectors $\mathbf{Y}, \mathbf{Z}$ and Lemma A.3,

$$\begin{aligned}\mathbf{E}[f(\mathbf{X})|\mathbf{X}_{[i]} = \mathbf{x}_{[i]}] - \mathbf{E}[f(\mathbf{X})|\mathbf{X}_{[i]} = \mathbf{x}'_{[i]}] = &\ \mathbf{E}[f(\mathbf{Y})] - \mathbf{E}[f(\mathbf{Z})] \\ = &\ \mathbf{E}[f(\mathbf{Y}) - f(\mathbf{Z})] \\ \leq &\ \mathbf{E}\left[\sum_{j=1}^{n} c_j \mathbf{1}_{Y_j \neq Z_j}\right] \\ \leq &\ c_i + c_{p(i)}.\end{aligned}$$

the first equality is due to the coupling constructed before, and the first inequality is by triangle inequality and $\mathbf{c}$-Lipschitz properties of f. □

We are now ready to prove Theorem 3.2.

Proof of Theorem 3.2. By Lemma A.1 and Lemma A.4

$$\begin{aligned}\mathbf{Pr}(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t) \leq &\ \exp\left(-\frac{2t^2}{\sum_{j \in V(G)\setminus\{n\}}(c_j + c_{p(j)})^2 + c_n^2}\right) \\ = &\ \exp\left(-\frac{2t^2}{\sum_{\langle j,k\rangle \in E(G)}(c_j + c_k)^2 + c_{\min}^2}\right)\end{aligned}$$

the last equality is because the root n has no parent and the **Well-rooted** assumption. □

A.2 Proof of Theorem 3.3

Proof of Theorem 3.3. The proof is similar to that of Theorem 3.2. Without loss of generality, we assume that each component of the forest G are well-rooted and well-sorted. Then the proofs of Lemma A.2-A.4 remain valid, since variables in different components are independent. As a result, the theorem holds due to Lemma A.1.

□

A.3 Proof of Theorem 3.6

Lemma A.5. *Suppose that $f : \mathbf{\Omega} \to \mathbb{R}$ is a $\mathbf{c}$-Lipschitz function and G is a dependency graph of a random vector $\mathbf{X}$ that takes values in $\mathbf{\Omega}$. For any $t > 0$ and any $(\phi, F) \in \Phi(G)$ with F consisting of trees $\{T_i\}_{i\in[k]}$, the following inequality holds:*

$$\mathbf{Pr}(f(\mathbf{X}) - \mathbf{E}[f(\mathbf{X})] \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{\langle u,v\rangle \in E(F)}(\widetilde{c}_u + \widetilde{c}_v)^2 + \sum_{i=1}^{k} \widetilde{c}_{\min,i}^2}\right)$$

where $\widetilde{c}_u = \sum_{i\in\phi^{-1}(u)} c_i$ and $\widetilde{c}_{\min,i} = \min_{u\in V(T_i)} \widetilde{c}_u$. Here, $\phi^{-1}(u)$ is the set of pre-images of u.

Proof. For any $u \in V(F)$, define a random vector $\mathbf{Y}_u = \{X_i\}_{i\in\phi^{-1}(u)}$. Treat each $\mathbf{Y}_u$ as a random variable. Define a new random vector $\mathbf{Y} = (\mathbf{Y}_u)_{u\in V(F)}$, and let $g(\mathbf{Y}) = f(\mathbf{X})$. It is easy to check that g is $\widetilde{\mathbf{c}}$-Lipschitz, where $\widetilde{\mathbf{c}} = (\widetilde{c}_u)_{u\in V(F)}$. The theorem immediately follows from Theorem 3.3. □

Lemma A.5 immediately implies Theorem 3.6 by the definition of forest complexity.

B Omitted Proofs in Section 4

B.1 Proof of Lemma 4.2

The following technical lemma is needed.

Lemma B.1 ([43])**.** *Given a β_n-uniformly stable learning algorithm $\mathcal{A}$, for any $\mathbf{S}, \mathbf{S}' \in (\mathcal{X} \times \mathcal{Y})^n$ that differ only in one entry, it holds that*

$$|\Phi_{\mathcal{A}}(\mathbf{S}) - \Phi_{\mathcal{A}}(\mathbf{S}')| \leq 4\beta_n + \frac{M}{n}$$

Proof. In the literature, Lemma B.1 was proved for i.i.d. data, actually, the proof remains valid in our setting. Assume $\mathbf{S}, \mathbf{S}'$ differ only in i-th entry, and denote $\mathbf{S}'$ as $\mathbf{S}^i$

$$\mathbf{S}^i = ((x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_i', y_i'), (x_{i+1}, y_{i+1}) \ldots, (x_m, y_m))$$

and the marginal distribution of (x_i', y_i') is also D.

Notice that we do not require the data to be i.i.d., samples are dependent with the same marginal probability distribution D. First, we bound $R(f_{\mathbf{S}}^{\mathcal{A}}) - R(f_{\mathbf{S}^i}^{\mathcal{A}})$

$$\begin{aligned}
&|R(f_{\mathbf{S}}^{\mathcal{A}}) - R(f_{\mathbf{S}^i}^{\mathcal{A}})| && (7)\\
\leq &|R(f_{\mathbf{S}}^{\mathcal{A}}) - R(f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}})| + |R(f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}) - R(f_{\mathbf{S}^i}^{\mathcal{A}})| && (8)\\
= &|\mathbf{E}_D[\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x))] - \mathbf{E}_D[\ell(y, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x))]| + |\mathbf{E}_D[\ell(y, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x))] - \mathbf{E}_D[\ell(y, f_{\mathbf{S}^i}^{\mathcal{A}}(x))]| && (9)\\
= &|\mathbf{E}_D[\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x))]| + |\mathbf{E}_D[\ell(y, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^i}^{\mathcal{A}}(x))]| && (10)\\
\leq &2\beta_n && (11)
\end{aligned}$$

then, we bound $\widehat{R}(f_{\mathbf{S}}^{\mathcal{A}}) - \widehat{R}_{\mathbf{S}^i}(f_{\mathbf{S}^i}^{\mathcal{A}})$

$$\begin{aligned}
&n|\widehat{R}(f_{\mathbf{S}}^{\mathcal{A}}) - \widehat{R}_{\mathbf{S}^i}(f_{\mathbf{S}^i}^{\mathcal{A}})| && (12)\\
= &\left| \sum_{(x_j, y_j) \in \mathbf{S}} \ell(y_j, f_{\mathbf{S}}^{\mathcal{A}}(x_j)) - \sum_{(x_j, y_j) \in \mathbf{S}^i} \ell(y_j, f_{\mathbf{S}^i}^{\mathcal{A}}(x_j)) \right| && (13)\\
\leq &\sum_{j \neq i} |\ell(y_j, f_{\mathbf{S}}^{\mathcal{A}}(x_j)) - \ell(y_j, f_{\mathbf{S}^i}^{\mathcal{A}}(x_j))| + |\ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i)) - \ell(y_i', f_{\mathbf{S}^i}^{\mathcal{A}}(x_i'))| && (14)\\
\leq &\sum_{j \neq i} |\ell(y_j, f_{\mathbf{S}}^{\mathcal{A}}(x_j)) - \ell(y_j, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x_j))| + \sum_{j \neq i} |\ell(y_j, f_{\mathbf{S}^{\setminus i}}^{\mathcal{A}}(x_j)) - \ell(y_j, f_{\mathbf{S}^i}^{\mathcal{A}}(x_j))| \\
&+ |\ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i)) - \ell(y_i', f_{\mathbf{S}^i}^{\mathcal{A}}(x_i'))| && (15)\\
\leq &2n\beta_n + M && (16)
\end{aligned}$$

combining above bounds, we have

$$\begin{aligned}
|\Phi_{\mathcal{A}}(\mathbf{S}) - \Phi_{\mathcal{A}}(\mathbf{S}^i)| = &\ |(R(f_{\mathbf{S}}^{\mathcal{A}}) - \widehat{R}(f_{\mathbf{S}}^{\mathcal{A}})) - (R(f_{\mathbf{S}^i}^{\mathcal{A}}) - \widehat{R}_{\mathbf{S}^i}(f_{\mathbf{S}^i}^{\mathcal{A}}))| \\
\leq &\ |R(f_{\mathbf{S}}^{\mathcal{A}}) - R(f_{\mathbf{S}^i}^{\mathcal{A}})| + |\widehat{R}(f_{\mathbf{S}}^{\mathcal{A}}) - \widehat{R}_{\mathbf{S}^i}(f_{\mathbf{S}^i}^{\mathcal{A}})| \\
\leq &\ 4\beta_n + \tfrac{M}{n}
\end{aligned}$$

□

combining Lemma B.1 and Theorem 3.6 leads to Lemma 4.2.

B.2 Proof of Lemma 4.3

We introduce a technical lemma before the proof of Lemma 4.3.

Lemma B.2. *Given a sample $\mathbf{S}$ of size n with dependency graph G, assume that the learning algorithm $\mathcal{A}$ is β_i-uniformly stable for any $i \leq n$. Suppose the maximum degree of G is Δ. Let $\beta_{n,\Delta} = \max_{i \in [0,\Delta]} \beta_{n-i}$. It holds that*

$$\max_{(x_i, y_i) \in \mathbf{S}} \mathbf{E}_{\mathbf{S},(x,y)}[\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i))] \leq 2\beta_{n,\Delta}(\Delta + 1).$$

Proof. For any $i \in [n]$, suppose $N_G^+(i) = \{j_1, \ldots, j_{n_i}\}$ with $j_{k-1} > j_k$. Define $\mathbf{S}^{(i,0)} = \mathbf{S}$ and for $k \in [n_i]$, $\mathbf{S}^{(i,k)}$ is obtained from $\mathbf{S}^{(i,k-1)}$ by removing the j_k-th entry. By uniform stability of $\mathcal{A}$, for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and $k \in [n_i]$,

$$|\ell(y, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x))| \leq \beta_{n,\Delta}$$

we have the decomposition via telescoping

$$\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) = \sum_{k=1}^{n_i} (\ell(y, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x)) + \ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x))$$

similarly

$$\ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i)) = \sum_{k=1}^{n_i} (\ell(y_i, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x_i)) - \ell(y_i, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x_i)) + \ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i))$$

Thus, we have

$$\begin{aligned}
& \ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i)) \\
= \; & \textstyle\sum_{k=1}^{n_i} \left((\ell(y, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x))) - (\ell(y_i, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x_i)) - \ell(y_i, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x_i))) \right) \\
& + \ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i)) \\
\leq \; & \textstyle\sum_{k=1}^{n_i} |\ell(y, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x)) - \ell(y, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x))| \\
& + \textstyle\sum_{k=1}^{n_i} |\ell(y_i, f_{\mathbf{S}^{(i,k)}}^{\mathcal{A}}(x_i)) - \ell(y_i, f_{\mathbf{S}^{(i,k-1)}}^{\mathcal{A}}(x_i))| + \ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i)) \\
\leq \; & 2n_i\beta_{n,\Delta} + \ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i))
\end{aligned}$$

As a result,

$$\begin{aligned}
& \mathbf{E}_{\mathbf{S},(x,y)}[\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i))] \\
= \; & \mathbf{E}_{\mathbf{S},(x,y)}[\ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i))] + 2n_i\beta_{n,\Delta} \\
\leq \; & \mathbf{E}_{\mathbf{S},(x,y)}[\ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i))] + 2\beta_{n,\Delta}(\Delta+1) \\
= \; & \mathbf{E}_{\mathbf{S},(x,y)}[\ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x))] - \mathbf{E}_{\mathbf{S}}[\ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i))] + 2\beta_{n,\Delta}(\Delta+1) \\
= \; & \mathbf{E}_{\mathbf{S}^{(i,n_i)},(x,y)}[\ell(y, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x))] - \mathbf{E}_{\mathbf{S}^{(i,n_i)},(x_i,y_i)}[\ell(y_i, f_{\mathbf{S}^{(i,n_i)}}^{\mathcal{A}}(x_i))] + 2\beta_{n,\Delta}(\Delta+1) \\
= \; & 2\beta_{n,\Delta}(\Delta+1)
\end{aligned}$$

The last equality is because (x_i, y_i) and (x, y) are independent of $\mathbf{S}^{(i,n_i)}$ and have the same distribution. □

Proof of Lemma 4.3.

$$\begin{aligned}
\mathbf{E}_{\mathbf{S}}[\Phi_{\mathcal{A}}(\mathbf{S})] = \; & \mathbf{E}_{\mathbf{S}}[\mathbf{E}_{(x,y)}[\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x))] - \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i))] \\
= \; & \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \mathbf{E}_{\mathbf{S},(x,y)}[\ell(y, f_{\mathbf{S}}^{\mathcal{A}}(x)) - \ell(y_i, f_{\mathbf{S}}^{\mathcal{A}}(x_i))] \\
\leq \; & 2\beta_{n,\Delta}(\Delta+1)
\end{aligned}$$

□