Selective Obfuscation for Fairness, Privacy, and Beyond

Hanrui Zhang
Duke University
hrzhang@cs.duke.edu

Vincent Conitzer
Duke University
conitzer@cs.duke.edu

Abstract

We study selective obfuscation in classification problems, where we selectively remove some information from data points to be classified. The goal is to diminish the accuracy of some classifiers, while preserving the accuracy of others. For example, we may consider the former set of classifiers to be undesirable due to fairness or privacy concerns. We prove a uniform convergence bound, which states that after applying any obfuscation, the empirical accuracy of any classifier on a large enough sample set is uniformly close to its actual accuracy, as long as the classes of obfuscations and classifiers considered jointly have bounded complexity. We then provide a way of bounding the joint complexity, by reducing to the individual complexity measures of the obfuscation and classifier classes respectively. Our results provide a general and principled approach for finding nearly optimal obfuscations, which accommodates various notions of fairness and privacy studied in the literature.

1 Introduction

We imagine the following scenario. Several studies rigorously establish that in a certain job market, employers unjustly discriminate in who they hire, based on prejudicial criteria. In response, a new government agency is introduced to change the situation. All job applications in this job market must be submitted to this agency rather than directly to the employer, and employers can only access applications through the agency. The agency plans to pre-process the job applicants' profiles before making them available to employers, in a way that reduces unjust discrimination while continuing to make useful information available to employers.

Historical data suggest that there are two kinds of criteria that are used by employers: fair ones and unfair ones. Ideally, the agency wishes to process – obfuscate, in fact, since processing can only remove information – candidate profiles in such a way that fair criteria can still be accurately assessed and remain informative, while unfair ones become difficult or impossible to assess, and therefore presumably will no longer be used. The question is how to find a way to process profiles that achieves the above goal, ideally with provable guarantees.

This is a nontrivial task, even given perfect knowledge about all job candidates and criteria used by employers. The agency could in principle try to formulate the above as a high-dimensional optimization problem, with a complex objective specified based on all fair and unfair criteria and the distribution of job candidates. Such an approach would likely lead to an overwhelmingly large program, which is not necessarily convex or otherwise tractable, rendering it infeasible for any practical purpose. The agency could alternatively perform a heuristic search, possibly restricted to a small number of job candidates and criteria. While this approach appears more practical, it is not clear that the output solution would perform close to optimally or even at all well. To tackle the problem, one should proceed in a more principled way.

There are many other problems besides job applications that fit the same setup. For example, a city may have placed cameras across the city, and wish to share the data produced by these cameras broadly for other desirable purposes. For example, another entity may wish to use this information to predict traffic and route its users more efficiently. On the other hand, yet another entity may wish to track individual people across the city, which is not the city's intent, and as a result it wishes to obfuscate the data in a way that makes such a use of it impossible, while still enabling the use of it for traffic prediction. In this context, a natural approach to obfuscating would be to blur out people's faces and license plates, though this may not be enough if there are other identifying characteristics.

In this paper, we study the problem of selective obfuscation in classification problems, where the goal, roughly speaking, is to preserve or diminish the accuracy of individual classifiers. We suppose that which classifiers are considered (un)desirable is based on an externally defined measure, whether it is based on fairness, privacy, or other criteria. Thus, the paradigm we introduce aims to provide a black-box approach for finding optimal obfuscations, which can accommodate various notions of fairness and privacy studied in the literature. Our results fit particularly well in scenarios where the desirability measure is implicit ("we know whether it is wrong when we see it"), and users (e.g., employers) lack the motivation, information, or expertise to adapt their classifiers in response to the obfuscation.

1.1 Our Results

In this paper, we consider a rather general notion of obfuscations, and define an obfuscation as a mapping from all data points in the space to their representatives, which are also data points in the space. In order to formally study the problem, we first introduce the notion of *fidelity*, $\mathrm{fid}(o, c)$, of an obfuscation o with respect to a fixed classifier c, which is the probability that the label of a random data point according to c remains the same after applying the obfuscation o (see Definition 2.1). Our first result is the following uniform convergence bound for fidelity.

Theorem 1.1 (Uniform Convergence of Fidelity – informal version)**.** *Fix a class $\mathcal{O}$ of obfuscations and a class $\mathcal{C}$ of classifiers. If the joint complexity (formally defined in Definition 3.3) $d_{\mathrm{J}}(\mathcal{O}, \mathcal{C})$ of $(\mathcal{O}, \mathcal{C})$ is bounded, then with high probability, for all pairs of an obfuscation $o \in \mathcal{O}$ and a classifier $c \in \mathcal{C}$ simultaneously,*

$$\widehat{\mathrm{fid}}_S(o, c) - \mathrm{fid}(o, c) \to 0,$$

uniformly as the number of sample data points increases. Here $\widehat{\mathrm{fid}}_S(o, c)$ is the empirical fidelity on the sample set S, defined as the fraction of points in S whose labels according to c remain after applying the obfuscation o.

The above theorem directly implies a way of finding (nearly) optimal obfuscations: simply find the best obfuscation in $\mathcal{O}$ on a set of sample points, and by uniform convergence one can conclude with confidence that the same obfuscation will perform nearly optimally in general with respect to all possible classifiers in $\mathcal{C}$. Sometimes one may not be able to solve the problem optimally even on a sample set. However, even in such cases the above theorem is meaningful, since one may apply a heuristic method to find an empirically "good enough" obfuscation, and expect it to have similar performance in general. We also remark that to apply the above theorem, one only needs sample access to the distribution $\mathcal{D}$. This is particularly helpful when $\mathcal{D}$ does not admit a succinct representation, or is otherwise inaccessible.

In order to apply the uniform convergence bound, another issue is that one needs to be able to estimate the joint complexity of the obfuscation class $\mathcal{O}$ and the classifier class $\mathcal{C}$. This problem does not seem to admit a general solution. Similar to estimating the VC dimension, one may often have to develop ad hoc arguments for individual classes of classifiers. To this end, we aim to provide a way to bound the joint complexity, by reducing the joint complexity to the individual complexities of the obfuscation class and the classifier class respectively. Based on the observation that an obfuscation can also be seen as a way of clustering data points or a multiclass classifier (because it maps the data points to a generally smaller set of data points), we prove the following relation between complexity measures.

Theorem 1.2 (Subadditivity of Joint Complexity – informal version)**.** *The joint complexity $d_{\mathrm{J}}(\mathcal{O}, \mathcal{C})$ of a k-ary obfuscation class $\mathcal{O}$ and a binary classifier class $\mathcal{C}$ is bounded by*

$$d_{\mathrm{J}}(\mathcal{O}, \mathcal{C}) = \widetilde{O}(d_{\mathrm{N}}(\mathcal{O}) + d_{\mathrm{VC}}(\mathcal{C}) + k),$$

where $d_{\mathrm{N}}(\mathcal{O})$ is the Natarajan dimension of $\mathcal{O}$, $d_{\mathrm{VC}}(\mathcal{C})$ is the VC dimension of $\mathcal{C}$, and $\widetilde{O}$ hides a polylogarithmic factor.

The above theorem enables one to directly transform complexity upper bounds on well-studied classes of obfuscations and classifiers to bounds on the joint complexity. In particular, most commonly used classes of obfuscations and classifiers admit known complexity bounds, which are often nearly optimal. The above theorem, together with the uniform convergence of fidelity, immediately provides provable guarantees for combinations of traditional clustering methods (as obfuscations) and classification methods.

1.2 Related Work

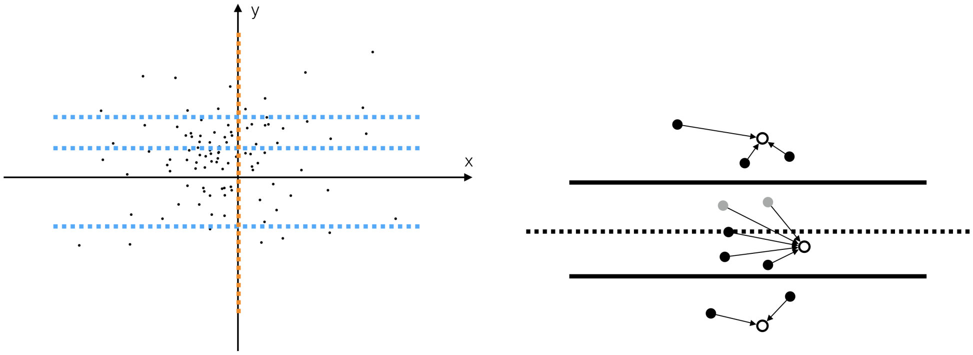
The problems considered in this paper are technically closely related to well-studied topics in machine learning, such as clustering and multiclass classification. For clustering, a fruitful line of research considers clustering generally as an optimization problem, where the goal is to minimize some cost function. Examples include approximate k-means [22, 4, 5] and k-medians [3, 33, 22] clustering. Generalization of clustering methods based on finite samples has been considered, including generalizing from observed features to unobserved ones [29, 30], axiomatic characterization of generalization [27, 12, 39], and uniform convergence of various loss functions [33, 9, 6]. For multiclass classification, generalization bounds have been established based on various measures of complexity [34, 13, 20]. Our results differ from the above in two ways. First, rather than distance-based loss functions or accuracy of classification, we focus on the notion of fidelity, which is not to be minimized (which is the case for loss functions) or maximized (which is the case for accuracy) per se. Second, we consider obfuscations not by themselves, but together with a class of binary classifiers, which models properties of interest in specific problems. Both our motivation and contribution are conceptually different from those of existing results on clustering and multiclass classification.

While our results are conceptually different from most work on fairness or privacy, we discuss below several related results from those areas. A popular notion studied in the fair classification / clustering literature is disparate impact. Research along this line generally requires protected groups to be equally distributed into classes / clusters, e.g., [41, 18, 7, 14]. Blackbox approaches, which transform generic classification / clustering algorithms into fair ones with mild loss, have been developed [2], as well as other notions of fairness [11, 17, 28], and the impact of fairness constraints to efficiency has also been studied [24, 21]. Another line of research considers differentially private clustering and classification, e.g., [16, 38, 40, 37, 10, 25, 31]. The key difference between these results and ours is that we do not consider specific fairness or privacy criteria. Rather, we aim to provide a principled way to find solutions satisfying any externally given criteria, possibly by existing work in the areas of fairness and privacy.

Finally, our results are related to the empirical study of data masking or data obfuscation, where commonly used techniques include shuffling, encrypting, or masking out entries of the data [8, 35]. The focus in this line of research is often empirical evaluation of practical methods, normally with respect to an ad hoc metric of performance. Our results provide a complementary view of data masking, establishing theoretical explanations for some of the popular methods previously studied in relatively empirical ways.

2 A Motivating Example

To better motivate the setup and properly define the problem, consider the following concrete example. Imagine that our data consist of images of pedestrians near the side of the road, taken by cameras on self-driving cars to determine the driving actions they should take. For simplicity of presentation, imagine that these images can be effectively summarized so that all data points lie in a two-dimensional plane X, and are distributed according to $\mathcal{D}$. Each point corresponds to a summarized image, where the x-coordinate primarily represents the pedestrian's skin tone, and the y-coordinate primarily represents the pedestrian's physical posture. The class $\mathcal{C}$ of classifiers considered by users (those writing the software for self-driving cars) is simply all linear separators, i.e., all lines in the plane. It is commonly agreed that all horizontal classifiers (i.e., those parallel to the x-axis, taking into consideration only the physical posture of the pedestrian, which signals intent to enter the road) are fair, and the two classifiers whose decision boundary is precisely the y-axis (classifying pedestrians into those with darker and those with lighter skin) are unfair. We obfuscate



(a)

(b)

Figure 1: 1a: Illustration of the example. Black dots illustrate the distribution of data points. Blue lines are possible decision boundaries of fair classifiers. The orange line is the only possible decision boundary of the two unfair classifiers. 1b: Illustration of a good classifier with low fidelity. The dotted line is the decision boundary of the good classifier with lowest fidelity. Lines and circles illustrate how the obfuscation divides the space into 3 regions, and maps points to their representatives. Black points are those sample points whose labels are preserved by the obfuscation, and grey ones are those whose labels are flipped.

points in the plane by partitioning the plane into k regions, and finding a representative point for each region. Each time a point arrives, it will be mapped to the representative of the region containing the point, which is then fed into the classifiers being used by users. The goal is to confuse unfair classifiers, and keep fair ones accurate, as much as possible. See Figure 1a for an illustration of the example.

To obtain a good obfuscation, a natural first attempt is to partition the plane into k horizontal strips, which intuitively removes all horizontal information (i.e., the x-coordinate of a point), while approximately preserving vertical information (i.e., the y-coordinate). Yet, it is not straightforward to determine how fat or thin each of these strips should be, or where to place the representative point within each strip. To answer these questions, we first need to clarify the exact meaning of “confusing” a classifier.

2.1 Measuring Confusion by Fidelity

Observe that, according to a fixed classifier, the labels of a point and its region’s representative (i.e., the point to which the obfuscation will map the original point) may or may not be the same. So, in order to confuse a classifier, one would like to make the label of the representative of a random point to be minimally correlated with the true label of the same point. To make the above formal, consider an obfuscation $o : X \to X$ as a mapping, taking each point in X to its repesentative. Moreover, consider a classifier $c : X \to \{0, 1\}$ as a mapping, taking each point to its label. Choosing an obfuscation o to confuse a classifier c is then equivalent to minimizing the *fidelity* of o with respect to c, as defined below.

Definition 2.1 (Fidelity)**.** The fidelity $\mathrm{fid}(o, c)$ of an obfuscation o with respect to a classifier c (under distribution $\mathcal{D}$) is defined to be

$$\mathrm{fid}(o, c) = \Pr_{x \sim \mathcal{D}}[c(x) = c(o(x))].$$

With this definition of fidelity, we may now rephrase the problem in the following way. Let $\mathcal{G} \subseteq \mathcal{C}$ be the set of all good (fair) classifiers (i.e., all horizontal ones), and $\mathcal{B} \subseteq \mathcal{C}$ be the set of all bad (unfair) ones (i.e., the two whose decision boundary is the y-axis). Let $\mathcal{O}$ be the set of obfuscations which partition the plane into k horizontal strips. We wish to find the minimum $\varepsilon > 0$, such that there exists an obfuscation $o \in \mathcal{O}$,

- whose fidelity with respect to good classifiers is above $1 - \varepsilon$, i.e., $\forall c \in \mathcal{G}$, $\mathrm{fid}(o, c) \geq 1 - \varepsilon$, and
- whose fidelity with respect to bad classifiers is below $1/2 + \varepsilon$, i.e., $\forall c \in \mathcal{B}$, $\mathrm{fid}(o, c) \leq \frac{1}{2} + \varepsilon$.

2.2 Optimal Obfuscation Given Uniform Convergence

With the problem properly formulated, we now come back to our motivating example, and try to find an (approximately) optimal obfuscation. We base our discussion on what, for now, is an assumption: uniform convergence of fidelity. That is, we assume that simultaneously for all $o \in \mathcal{O}$ and $c \in \mathcal{C}$, for a sufficiently large number m, the fidelity $\mathrm{fid}(o, c)$ is close to the empirical fidelity $\widehat{\mathrm{fid}}_S(o, c)$ on a random set $S \sim \mathcal{D}^m$ of m iid sample data points, defined as

$$\widehat{\mathrm{fid}}_S(o, c) = 1 - \frac{1}{|S|} \sum_{x \in S} |c(x) - c(o(x))|.$$

We will show later that the assumption in fact holds with high probability. For now, we assume that m is large enough that the fidelity can always be considered (approximately) the same as the empirical fidelity on the sample set S.

For our motivating example, we first show that for any $o \in \mathcal{O}$, there is always a good classifier $c \in \mathcal{G}$ such that

$$\mathrm{fid}(o, c) \le 1 - 1/(2k) + o(1).$$

Consider the region in o containing the most points from S. Let $S' \subseteq S$ be the set of points from S in that region. Clearly $|S'| \ge m/k$. For simplicity, suppose no two points in S share the same y-coordinate. Now let x be the point in S' with the median y-coordinate, and let $c \in \mathcal{G}$ be any horizontal classifier passing through x (there are two possible choices). No matter where the representative of S' is, there are always at least $|S'|/2 - 1$ points not lying on the same side of c as the representative. These

$$|S'|/2 - 1 \ge m/(2k) - 1$$

points are necessarily classified incorrectly by c. As a result,

$$\widehat{\mathrm{fid}}_S(o, c) \le 1 - 1/(2k) + 1/m,$$

which given idealized uniform convergence implies

$$\mathrm{fid}(o, c) \le 1 - 1/(2k) + 1/m.$$

See Figure 1b for an illustration.

Approximately matching this upper bound, we now construct an obfuscation which has fidelity at least $1 - 1/(2k) - o(1)$ with respect to *all* good classifiers, and at most $1/2$ with respect to all bad ones. First, in light of the above upper bound, we partition S uniformly into k horizontal strips, each containing approximately m/k of the points. For any $i \in [k]$, let $S_i \subseteq S$ be the points from S partitioned into the i-th region. We restrict the choice of the representatives, requiring that the representative of the i-th region has the median y-coordinate in S_i. A similar argument as the above then guarantees fidelity $1 - 1/(2k) - o(1)$ with respect to any good classifier. (The good classifier's boundary will lie within some region, within which at least half of the points lie on the same side of the boundary as the representative; for all other regions, all the points, including the representative, lie on the same side.) The only problem left is to decide the x-coordinates of the representatives, aiming to uniformly minimize fidelity with respect to bad classifiers.

The choice of the x-coordinates is again straightforward. For simplicity, suppose no point in S lies on the y-axis. For the i-th region, if at least half of the points in S_i have positive x-coordinate, then let the x-coordinate of the representative be any negative number; otherwise, let it be any positive number. Now for any bad classifier $c \in \mathcal{B}$ and for any region i, at least half of the points in S_i are on the opposite side of the y-axis from the representative of the region, which means these points are classified incorrectly by c. So, overall, the obfuscation constructed above has fidelity at most $1/2$ with respect to any bad classifier.

Finally, we remark that even with uniform convergence, there is still an additive error diminishing as the size m of the sample set grows (see Theorem 3.1). This also contributes to the above fidelity bounds, which we omit in the analysis above for ease of presentation.

3 Uniform Convergence of Fidelity

In the previous section, we have illustrated what uniform convergence allows us to do, by a minimally nontrivial example. In this section, we show that uniform convergence in fact holds whenever the obfuscation class and the classifier class jointly have bounded complexity.

We first introduce a few useful notions, and define the joint complexity, which, as we will show, dictates the rate of convergence.

Definition 3.1 (Stable Set)**.** The *stable set* $s(o,c)$ under a pair (o,c) of an obfuscation $o : X \to X$ and a classifier $c : X \to \{0,1\}$ is defined to be $s(o,c) = \{x \in X \mid c(x) = c(o(x))\}$.

In other words, the stable set is the set of all points whose labels remain the same after applying the obfuscation.

Definition 3.2 (Joint Shattering)**.** Fix the space X of data points. A subset $A \subseteq X$ of data points is *jointly shattered* by an obfuscation class $\mathcal{O}$ and a classifier class $\mathcal{C}$, if

$$\{s(o,c) \cap A \mid o \in \mathcal{O}, c \in \mathcal{C}\} = 2^A.$$

In other words, a set is shattered if we can make any subset of it (the intersection with) the stable set, by a choice of classifier and obfuscation. With the above notions, we are ready to define the joint complexity.

Definition 3.3 (Joint Complexity)**.** The joint complexity $d_{\mathrm{J}}(\mathcal{O},\mathcal{C})$ of an obfuscation class $\mathcal{O}$ and a classifier class $\mathcal{C}$ over space X is defined to be the cardinality of the largest set $A \subseteq X$ of points that is jointly shattered by $\mathcal{O}$ and $\mathcal{C}$. That is,

$$d_{\mathrm{J}}(\mathcal{O},\mathcal{C}) = \max\{|A| \mid A \subseteq X, \{s(o,c) \cap A \mid o \in \mathcal{O}, c \in \mathcal{C}\} = 2^A\}.$$

One may compare the above definition with the VC dimension (see, e.g., [26]).

Definition 3.4 (VC Dimension)**.** The VC dimension of a binary classifier class $\mathcal{C}$ over space X is defined to be

$$d_{\mathrm{VC}}(\mathcal{C}) = \max\{|A| \mid A \subseteq X, \{c \cap A \mid c \in \mathcal{C}\} = 2^A\}.$$

The two definitions are conceptually similar, where the key difference is that for the joint complexity, we consider shattering by the family of all stable sets induced by pairs in $\mathcal{O} \times \mathcal{C}$, while for the VC dimension, we simply consider shattering by $\mathcal{C}$. In light of this, let

$$\mathcal{S}(\mathcal{O},\mathcal{C}) = \{s(o,c) \mid o \in \mathcal{O}, c \in \mathcal{C}\}.$$

Note that any binary-valued function $f : X \to \{0,1\}$ can be equivalently seen as a subset of X, defined as $\{x \in X \mid f(x) = 1\}$. From now on, we use the function interpretation and the subset interpretation interchangeably. $\mathcal{S}$ (we will omit the parameters when they are clear from the context), similar to any binary classifier class, is a family of subsets of the space X, so the notion of the VC dimension applies to $\mathcal{S}$ as well. Given these observations, one may equivalently define the joint complexity in the following way.

Lemma 3.1 (Alternative Definition of Joint Complexity)**.** *The joint complexity $d_{\mathrm{J}}(\mathcal{O},\mathcal{C})$ of an obfuscation class $\mathcal{O}$ and a classifier class $\mathcal{C}$ satisfies $d_{\mathrm{J}}(\mathcal{O},\mathcal{C}) = d_{\mathrm{VC}}(\mathcal{S}(\mathcal{O},\mathcal{C}))$.*

The proof of the above lemma, as well as all other proofs, is deferred to the appendix. We will find the alternative definition particularly useful in establishing the uniform convergence of fidelity, formally stated below.

Theorem 3.1 (Uniform Convergence of Fidelity)**.** *Fix any space of data points X, distribution $\mathcal{D}$ over X, obfuscation class $\mathcal{O}$, and classifier class $\mathcal{C}$. There exists an absolute constant C, such that for any $\varepsilon > 0$ and $\delta > 0$, with a set of*

$$m \geq \frac{C(d_{\mathrm{J}}(\mathcal{O},\mathcal{C}) \cdot \log(1/\varepsilon) + \log(1/\delta))}{\varepsilon^2}$$

iid sample points $S \sim \mathcal{D}^m$, with probability at least $1 - \delta$, simultaneously for all $o \in \mathcal{O}$ and $c \in \mathcal{C}$,

$$\left|\mathrm{fid}(o,c) - \widehat{\mathrm{fid}}_S(o,c)\right| \leq \varepsilon.$$

4 Bounding the Joint Complexity

Theorem 3.1 guarantees uniform convergence whenever the joint complexity is bounded. However, as discussed above, estimating the joint complexity directly is often a tricky task, since the family of stable sets $\mathcal{S}(\mathcal{O}, \mathcal{C})$ often appears less structured, even if $\mathcal{O}$ and $\mathcal{C}$ themselves have nice properties or low complexity. To effectively estimate the joint complexity, we would like to bound $d_{\mathrm{J}}(\mathcal{O}, \mathcal{C})$ by a function of the complexities of $\mathcal{O}$ and $\mathcal{C}$ respectively. This section is dedicated to such a bound.

In order to develop the bound, first we need to be able to measure the complexity of $\mathcal{O}$. (The VC dimension is a natural measure of complexity for $\mathcal{C}$.) Below we show how an obfuscation class can be transformed to a class of multiclass classifiers, for which there exists well-studied measures of complexity.

First observe that each obfuscation induces a set of multiclass classifiers, in the way defined below.

Definition 4.1 (Induced Multiclass Classifiers)**.** Fix an obfuscation $o : X \to X$ which partitions the space X into k subsets, i.e., which satisfies $|\{o(x) \mid x \in X\}| = k$. A k-class classifier $c : X \to [k]$ is an induced classifier of o (denoted $o \to c$), if for any $x_1, x_2 \in X$,

$$o(x_1) = o(x_2) \iff c(x_1) = c(x_2).$$

Intuitively, an induced classifier numbers the subsets in the partition given by o, such that no two subsets get the same number. To measure the complexity of a k-ary obfuscation class $\mathcal{O}$, we consider its induced class $\mathcal{I}(\mathcal{O})$ of k-class classifiers, $\mathcal{I}(\mathcal{O}) = \{c \mid \exists o \in \mathcal{O}, o \to c\}$.

Recall the following definition of the *Natarajan dimension* of a class of multiclass classifiers.

Definition 4.2 (Natarajan Dimension, Rephrased [34])**.** Let $\mathcal{C}$ be a family of k-class classifiers over space X. A set S is shattered by $\mathcal{C}$ if there exist $c_1, c_2 : S \to [k]$ satisfying for all $x \in S$, $c_1(x) \neq c_2(x)$, and for all $T \subseteq S$, there exists $c \in \mathcal{C}$ such that for all $x \in T$, $c(x) = c_1(x)$, and for all $x \in S \setminus T$, $c(x) = c_2(x)$. The Natarajan dimension $d_{\mathrm{N}}(\mathcal{C})$ of $\mathcal{C}$ is the cardinality of the largest set shattered by $\mathcal{C}$.

The Natarajan dimension is a generalization of the VC dimension, and measures the complexity of classes of multiclass classifiers in various senses. In particular, when $k = 2$, it coincides precisely with the VC dimension. We now extend this notion to obfuscation classes via induced classifiers.

Definition 4.3 (Natarajan Dimension of Obfuscation Classes)**.** The Natarajan dimension $d_{\mathrm{N}}(\mathcal{O})$ of an obfuscation class $\mathcal{O}$ is defined to be the Natarajan dimension of its induced class of classifiers $\mathcal{I}(\mathcal{O})$, i.e., $d_{\mathrm{N}}(\mathcal{O}) = d_{\mathrm{N}}(\mathcal{I}(\mathcal{O}))$.

With measures of complexity properly defined, we are now ready to prove the following result.

Theorem 4.1 (Subadditivity of Joint Complexity)**.** *For any k-ary obfuscation class $\mathcal{O}$ and classifier class $\mathcal{C}$, the joint complexity satisfies*

$$d_{\mathrm{J}}(\mathcal{O}, \mathcal{C}) = O(k + (d_{\mathrm{N}}(\mathcal{O}) + d_{\mathrm{VC}}(\mathcal{C})) \cdot \log(d_{\mathrm{N}}(\mathcal{O}) + d_{\mathrm{VC}}(\mathcal{C}) + k)).$$

5 Further Applications of the Framework

In this section, we discuss several applications of our framework, which illustrate the power and implications of our results in more practical tasks.

5.1 Concluding Remarks on the Introductory Example

In Section 2, we analyzed a minimally nontrivial example in order to illustrate the problem and the power of our results. One assumption left unproved was that uniform convergence in fact holds for the obfuscation class $\mathcal{O}$ and classifier class $\mathcal{C}$ considered in the example. Given Theorems 3.1 and 4.1, we are now ready to give a simple proof.

In order to bound $d_{\mathrm{J}}(\mathcal{O}, \mathcal{C})$, we only need to bound $d_{\mathrm{N}}(\mathcal{O})$ and $d_{\mathrm{VC}}(\mathcal{C})$ respectively. For the latter, it is well known that the family of all linear classifiers in the plane has VC dimension $d_{\mathrm{VC}}(\mathcal{C}) = 3$. As for $\mathcal{O}$, we apply the following lemma.

Lemma 5.1 ([20], rephrased)**.** *Let $\mathcal{C}$ be a multiclass classifier class. Furthermore, suppose each $c \in \mathcal{C}$ can be expressed as a binary decision tree (see Section 5.2 of [20] for a formal definition) with at most k leaves, where each internal node corresponds to a binary classifier from class $\mathcal{C}_0$, and each leaf corresponds to an output label. Then $d_{\mathrm{N}}(\mathcal{C}) = O(d_{\mathrm{VC}}(\mathcal{C}_0) \cdot k \log(d_{\mathrm{VC}}(\mathcal{C}_0) \cdot k))$.*

In words, a binary decision tree may amplify the complexity of a binary classifier class at most roughly linearly in the number of leaves. Observe that the induced class $\mathcal{I}(\mathcal{O})$ satisfies the conditions of Lemma 5.1. More precisely, each classifier in $\mathcal{I}$ can be represented in the following way: if a point is above the topmost boundary, then it belongs to the topmost region; otherwise, if it is above the second boundary from above, then it belongs to the second region; etc. The binary classifier class at each internal node has VC dimension 2, and there are precisely k leaves in the tree. As a result, by Lemma 5.1, $d_{\mathrm{N}}(\mathcal{O}) = d_{\mathrm{N}}(\mathcal{I}(\mathcal{O})) = O(k \log k)$. Now by Theorem 4.1, we may conclude that $d_{\mathrm{J}}(\mathcal{O}, \mathcal{C}) = O(k \log^2 k)$, which through Theorem 3.1 directly implies uniform convergence.

5.2 Generalization Analysis for a Popular Clustering Methodology

A popular paradigm of clustering in metric spaces is to choose k centers in the space according to some carefully engineered criteria, and then associate each point in the space to the closest center. Examples of this paradigm include k-means [32] and k-medians [15]. We show in this subsection that such clustering methods generalize well in Euclidean spaces, in terms of fidelity with respect to any classifier class with bounded complexity.

Let $\mathcal{O}$ be the class of all clustering methods (as obfuscations) in $\mathbb{R}^d$ which choose k centers as representatives and attach each point to the closest center. We argue below $d_{\mathrm{N}}(\mathcal{O}) = O(k^2 d \log(kd))$, which directly implies generalization bounds on fidelity.

To see why this is true, we again apply Lemma 5.1. Fix a way of clustering uniquely determined by the k centers $\{y_i\}_{i \in [k]}$. The procedure which decides the representative of any point x in $\mathbb{R}^d$ again can be written as a decision tree as follows. If y_1 is the closest center to x, then x belongs to cluster 1; otherwise, if y_2 is the closest center, then x belongs to cluster 2; etc. There are again precisely k leaves in the decision tree. As for the class of binary classifiers used at internal nodes, observe that the classifier at any node is the intersection of $k-1$ half-spaces, which is known to have VC dimension $O(kd \log k)$ [19]. Lemma 5.1 then implies the desired bound on $d_{\mathrm{N}}(\mathcal{O})$.

One implication of the above generalization bound is that, as long as one cares about properties that can be modelled by fidelity , it suffices to optimize the way of clustering on a large enough but finite set of data points. This allows one to perform clustering with provable guarantees following the steps below.

1. Draw a sample set S with enough iid data points.
2. Run any clustering algorithm on S, e.g., Lloyd's algorithm for k-means. Note that such methods often work only for finite input sets, so it is impossible to run them on the distribution of data points directly.
3. Check if the output clusters have high empirical fidelity on S with respect to the classifiers of interest, which capture the properties that one hopes to preserve after clustering. (Also, one may check that the clusters have low empirical fidelity on the classifiers that capture the properties that are unfair.) If satisfied, settle with the current clusters; otherwise, go back to Step 2 and try a different clustering method.

In fact, similar steps are often implemented in practice in order to tackle real-world problems. Our results provide theoretical explanations for this methodology used by practitioners.

6 Conclusion and Future Research

Entities with valuable data often struggle with the decision of whether to make it more broadly available. On one hand they see many valuable uses that others may have for it, but on the other hand the data may be used in other ways that are best avoided. Similar concerns motivate using differential privacy for census data [1], where what is not desired is the ability to learn about specific individuals. However, often there are different concerns that are not addressed by differential privacy: we would like to make individual data points (applicants, video frames, etc.) available, but we want to avoid specific uses. Given that these uses can vary from domain to domain, general abstract frameworks are needed to reason about these issues. In this paper, we have introduced such a framework, as well as basic results that allow us to leverage existing insights in the theory of machine learning. Future research can improve on these basic results by focusing on specific cases. More importantly, it can apply this methodology to practice, resulting in well-motivated, near-optimal obfuscations of data that allow desirable uses of it while preventing undesirable ones.

Broader Impact

As discussed above, our results can and should be used to improve fairness and to preserve privacy in classification. Potential impact in terms of fairness and privacy has already been discussed. Nevertheless, we note that our results do not impose a particular measure of fairness or privacy. So, implemented with a malicious objective, our results could, for example, exacerbate discrimination.

References

[1] John M Abowd. Stepping-up: The census bureau tries to be a good data steward in the 21st century. Talk at International Conference on Machine Learning (ICML), 2019.

[2] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453*, 2018.

[3] Sanjeev Arora, Prabhakar Raghavan, and Satish Rao. Approximation schemes for Euclidean k-medians and related problems. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 106–113, 1998.

[4] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006.

[5] Olivier Bachem, Mario Lucic, S Hamed Hassani, and Andreas Krause. Approximate k-means++ in sublinear time. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[6] Olivier Bachem, Mario Lucic, S Hamed Hassani, and Andreas Krause. Uniform deviation bounds for k-means clustering. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 283–291. JMLR. org, 2017.

[7] Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. *arXiv preprint arXiv:1902.03519*, 2019.

[8] David E Bakken, R Rarameswaran, Douglas M Blough, Andy A Franz, and Ty J Palmer. Data obfuscation: Anonymity and desensitization of usable data sets. *IEEE Security & Privacy*, 2(6): 34–41, 2004.

[9] Maria Florina Balcan, Heiko Röglin, and Shang-Hua Teng. Agnostic clustering. In *International Conference on Algorithmic Learning Theory*, pages 384–398. Springer, 2009.

[10] Maria-Florina Balcan, Travis Dick, Yingyu Liang, Wenlong Mou, and Hongyang Zhang. Differentially private clustering in high-dimensional Euclidean spaces. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 322–331. JMLR. org, 2017.

[11] Maria-Florina F Balcan, Travis Dick, Ritesh Noothigattu, and Ariel D Procaccia. Envy-free classification. In *Advances in Neural Information Processing Systems*, pages 1238–1248, 2019.

[12] Shai Ben-David and Margareta Ackerman. Measures of clustering quality: A working set of axioms for clustering. In *Advances in neural information processing systems*, pages 121–128, 2009.

[13] Shai Ben-David, Nicolo Cesabianchi, David Haussler, and Philip M Long. Characterizations of learnability for classes of $\{0, \dots, n\}$-valued functions. *Journal of Computer and System Sciences*, 50(1):74–86, 1995.

[14] Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. In *Advances in Neural Information Processing Systems*, pages 4955–4966, 2019.

[15] Paul S Bradley, Olvi L Mangasarian, and W Nick Street. Clustering via concave minimization. In *Advances in neural information processing systems*, pages 368–374, 1997.

[16] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[17] Xingyu Chen, Brandon Fain, Charles Lyu, and Kamesh Munagala. Proportionally fair clustering. *arXiv preprint arXiv:1905.03674*, 2019.

[18] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. In *Advances in Neural Information Processing Systems*, pages 5029–5037, 2017.

[19] Monika Csikos, Andrey Kupavskii, and Nabil H Mustafa. Optimal bounds on the VC-dimension. *arXiv preprint arXiv:1807.07924*, 2018.

[20] Amit Daniely, Sivan Sabato, Shai Ben-David, and Shai Shalev-Shwartz. Multiclass learnability and the ERM principle. *The Journal of Machine Learning Research*, 16(1):2377–2404, 2015.

[21] Paul Gölz, Anson Kahng, and Ariel D Procaccia. Paradoxes in fair machine learning. In *Advances in Neural Information Processing Systems*, pages 8340–8350, 2019.

[22] Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 291–300, 2004.

[23] David Haussler and Philip M Long. *A generalization of Sauer's lemma*. University of California, Santa Cruz, Computer Research Laboratory, 1990.

[24] Lily Hu and Yiling Chen. Welfare and distributional impacts of fair classification. *arXiv preprint arXiv:1807.01134*, 2018.

[25] Zhiyi Huang and Jinyan Liu. Optimal differentially private algorithms for k-means clustering. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 395–408, 2018.

[26] Michael J Kearns and Umesh Virkumar Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

[27] Jon M Kleinberg. An impossibility theorem for clustering. In *Advances in neural information processing systems*, pages 463–470, 2003.

[28] Matthäus Kleindessner, Pranjal Awasthi, and Jamie Morgenstern. Fair k-center clustering for data summarization. *arXiv preprint arXiv:1901.08628*, 2019.

[29] Eyal Krupka and Naftali Tishby. Generalization in clustering with unobserved features. In *Advances in Neural Information Processing Systems*, pages 683–690, 2006.

[30] Eyal Krupka and Naftali Tishby. Generalization from observed to unobserved features by clustering. *Journal of Machine Learning Research*, 9(Mar):339–370, 2008.

[31] Xiaoqian Liu, Qianmu Li, Tao Li, and Dong Chen. Differentially private classification with decision tree ensemble. *Applied Soft Computing*, 62:807–816, 2018.

[32] Stuart Lloyd. Least squares quantization in PCM. *IEEE transactions on information theory*, 28 (2):129–137, 1982.

[33] Nina Mishra, Daniel Oblinger, and Leonard Pitt. Sublinear time approximate clustering. In *SODA*, volume 1, pages 439–447, 2001.

[34] Balas K Natarajan. On learning sets and functions. *Machine Learning*, 4(1):67–97, 1989.

[35] GK Ravikumar, TN Manjunath, Ravindra S Hegadi, and IM Umesh. A survey on recent trends, process and development in data masking for testing. *International Journal of Computer Science Issues (IJCSI)*, 8(2):535, 2011.

[36] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[37] Dong Su, Jianneng Cao, Ninghui Li, Elisa Bertino, and Hongxia Jin. Differentially private k-means clustering. In *Proceedings of the sixth ACM conference on data and application security and privacy*, pages 26–37, 2016.

[38] Jaideep Vaidya, Basit Shafiq, Anirban Basu, and Yuan Hong. Differentially private naive Bayes classification. In *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 1, pages 571–576. IEEE, 2013.

[39] Yule Vaz, Rodrigo Fernandes de Mello, and Carlos Henrique Grossi. Coarse-refinement dilemma: On generalization bounds for data clustering. *arXiv preprint arXiv:1911.05806*, 2019.

[40] Yining Wang, Yu-Xiang Wang, and Aarti Singh. Differentially private subspace clustering. In *Advances in Neural Information Processing Systems*, pages 1000–1008, 2015.

[41] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259*, 2015.

A Omitted Proofs

Proof of Lemma 3.1.

$$\begin{aligned} d_{\mathrm{VC}}(\mathcal{S}) &= \max\{|A| \mid \{s \cap A \mid s \in \mathcal{S}\} = 2^A\} && \text{(for brevity we omit } A \subseteq X) \\ &= \max\{|A| \mid \{s(o,c) \cap A \mid o \in \mathcal{O}, c \in \mathcal{C}\} = 2^A\} \\ &= d_{\mathrm{J}}(\mathcal{O}, \mathcal{C}), \end{aligned}$$

as desired. □

Proof of Theorem 3.1. We prove the theorem by reducing to a special case of the classical VC theorem, stated below (see, e.g., [36]).

Lemma A.1 (VC Theorem, Special Case)**.** *Fix any space of data points X, distribution $\mathcal{D}$ over X, and classifier class $\mathcal{C}'$. There exists an absolute constant C, such that for any $\varepsilon > 0$ and $\delta > 0$, with a set of*

$$m \geq \frac{C(d_{\mathrm{VC}}(\mathcal{C}) \cdot \log(1/\varepsilon) + \log(1/\delta))}{\varepsilon^2}$$

iid sample points $S \sim \mathcal{D}^m$, with probability at least $1 - \delta$, simultaneously for all $c \in \mathcal{C}'$,

$$\left| \Pr_{x \sim \mathcal{D}}[c(x) = 1] - \frac{1}{m} \sum_{x \in S} c(x) \right| \leq \varepsilon.$$

To apply the above lemma, we construct a new class of classifiers, which is precisely the family of stable sets $\mathcal{S} = \mathcal{S}(\mathcal{O}, \mathcal{C})$, and feed that into the lemma as parameter $\mathcal{C}'$. Observe that for any stable set $s = s(o, c)$ of $o \in \mathcal{O}$ and $c \in \mathcal{C}$,

$$\Pr_{x \sim \mathcal{D}}[s(x) = 1] = \Pr_{x \sim \mathcal{D}}[c(x) = c(o(x))] = \mathrm{fid}(o, c).$$

Moreover,

$$\frac{1}{m} \sum_{x \in S} s(x) = \frac{1}{m} \sum_{x \in S} (1 - |c(x) - c(o(x))|) = \widehat{\mathrm{fid}}_S(o, c).$$

Moreover, for any $o \in \mathcal{O}$ and $c \in \mathcal{C}$, there exists some $s \in \mathcal{S}$ such that $s = s(o, c)$. So plugging $\mathcal{C}' = \mathcal{S}$ into Lemma A.1, we immediately obtain that whenever

$$\begin{aligned} m &\geq \frac{C(d_{\mathrm{VC}}(\mathcal{S}) \cdot \log(1/\varepsilon) + \log(1/\delta))}{\varepsilon^2} \\ &= \frac{C(d_{\mathrm{J}}(\mathcal{O}, \mathcal{C}) \cdot \log(1/\varepsilon) + \log(1/\delta))}{\varepsilon^2}, && \text{(Lemma 3.1)} \end{aligned}$$

with probability at least $1 - \delta$, simultaneously for all $o \in \mathcal{O}$ and $c \in \mathcal{C}$,

$$\left| \mathrm{fid}(o, c) - \widehat{\mathrm{fid}}_S(o, c) \right| \leq \varepsilon,$$

which is precisely the desired theorem. □

Proof of Theorem 4.1. In light of Lemma 3.1, we consider equivalently the VC dimension of the family of stable sets $\mathcal{S}(\mathcal{O}, \mathcal{C})$. The high-level plan is to pick an arbitrary set $A \subseteq X$ of a certain size, construct a random subset $r \subseteq A$, and argue that with positive probability, such a set r is not in the projection of $\mathcal{S}$ to A, i.e.,

$$\Pr_r [r \notin \{A \cap s \mid s \in \mathcal{S}\}] > 0.$$

This implies that there exists some subset $r \subseteq A$ not in the projection of $\mathcal{S}$ to A, which means $\mathcal{S}$ cannot shatter A. Because the above argument works for any $A \subseteq X$ of a fixed size, this gives an upper bound on $d_{\mathrm{VC}}(\mathcal{S})$.

We now instantiate the above plan. Let A be any subset of X of size d, where d is a parameter whose value is to be chosen later. We first bound the number of (numbered) partitions induced by the projection of $\mathcal{S}$ to A. We proceed by bounding the size of the projection of $\mathcal{I} = \mathcal{I}(\mathcal{O})$ to A, and that of the projection of $\mathcal{C}$ to A, respectively. Bounds for both of the above can be derived directly from the following generalization of Sauer's lemma [23].

Lemma A.2 (Generalized Sauer's Lemma)**.** *Let $\mathcal{C}'$ be any class of k-class classifiers over X, with Natarajan dimension ${d_\mathrm{N}}' = d_\mathrm{N}(\mathcal{C}')$. For any set $A \subset X$ of size $|A| = n$, the size of the projection of $\mathcal{C}$ to A is bounded by*

$$|\{c|_A \mid c \in \mathcal{C}'\}| = O(k^{2{d_\mathrm{N}}'} \cdot n^{{d_\mathrm{N}}'}),$$

where $c|_A : A \to [k]$ is the unique mapping such that for all $x \in A$, $c|_A(x) = c(x)$.

Let $\mathcal{I}|_A$ and $\mathcal{C}|_A$ denote the projections of $\mathcal{I}$ and $\mathcal{C}$ to A respectively, i.e.,

$$\mathcal{I}|_A = \{c|_A \mid c \in \mathcal{I}\},\ \mathcal{C}|_A = \{c|_A \mid c \in \mathcal{C}\}.$$

Let $d_\mathrm{N} = d_\mathrm{N}(\mathcal{O}) = d_\mathrm{N}(\mathcal{I}(\mathcal{O}))$ and $d_\mathrm{VC} = d_\mathrm{VC}(\mathcal{C})$. Applied to $\mathcal{I}(\mathcal{O})$ and $\mathcal{C}$ respectively, Lemma A.2 immediately gives

$$|\mathcal{I}|_A| = O(k^{2d_\mathrm{N}} \cdot d^{d_\mathrm{N}}),\ |\mathcal{C}|_A| = O(2^{2d_\mathrm{VC}} \cdot d^{d_\mathrm{VC}}).$$

Now consider the product family $\mathcal{P}|_A$ of $\mathcal{I}|_A$ and $\mathcal{C}|_A$, defined by

$$\mathcal{P}|_A = \{p(c_1, c_2) \mid c_1 \in \mathcal{I}|_A, c_2 \in \mathcal{C}|_A\},$$

where $p(c_1, c_2) : X \to [k] \times \{0, 1\}$ is defined such that for $x \in A$,

$$p(c_1, c_2)(x) = (c_1(x), c_2(x)).$$

We have

$$|\mathcal{P}|_A| = |\mathcal{I}|_A| \cdot |\mathcal{C}|_A| = O(k^{2d_\mathrm{N}} \cdot 2^{2d_\mathrm{VC}} \cdot d^{d_\mathrm{N}+d_\mathrm{VC}}).$$

Let

$$\mathcal{S}|_A = \{s|_A \mid s \in \mathcal{S}\}.$$

Observe the following relation between $\mathcal{P}|_A$ and $\mathcal{S}|_A$.

Lemma A.3. *Suppose $s \in \mathcal{S}|_A$. Then there exists $p = p(c_1, c_2) \in \mathcal{P}|_A$ which satisfies for any $x_1, x_2 \in A$, if $c_1(x_1) = c_1(x_2)$, then*

$$s(x_1) = s(x_2) \iff c_2(x_1) = c_2(x_2).$$

Proof. Suppose $s \in \mathcal{S}|_A$ is induced by $s = s(o, c)$, where $o \in \mathcal{O}$ and $c \in \mathcal{C}$. Let c_1 be any k-class classifier induced by o (i.e., $o \to c_1$), and $c_2 = c$. We now argue that for any $x_1, x_2 \in A$, if $c_1(x_1) = c_1(x_2)$, then

$$s(x_1) = s(x_2) \iff c_2(x_1) = c_2(x_2).$$

In fact, since $o \to c_1$, $c_1(x_1) = c_1(x_2)$ implies $o(x_1) = o(x_2)$. We then have

$$\begin{aligned} & s(x_1) = s(x_2) \\ \iff & |c_2(x_1) - c_2(o(x_1))| = |c_2(x_2) - c_2(o(x_2))| \\ \iff & |c_2(x_1) - c_2(o(x_1))| = |c_2(x_2) - c_2(o(x_1))| && (o(x_1) = o(x_2)) \\ \iff & c_2(x_1) = c_2(x_2), \end{aligned}$$

as desired. □

So, if for some set $B \subseteq A$, no $p \in \mathcal{P}|_A$ satisfies the conditions in Lemma A.3, one may conclude that $B \notin \mathcal{S}|_A$, or in other words, $\mathcal{S}$ does not shatter A. We now show this is in fact the case for any $A \subseteq X$ where $|A| = d$ is large enough, using the probabilistic method.

Let r be a subset of A chosen uniformly at random, or equivalently, formed by adding each element of A to r independently with probability 1/2. Fix any $p = p(c_1, c_2) \in \mathcal{P}|_A$. We show p does not satisfy the conditions of Lemma A.3 for r (which takes the role of s in the lemma) with probability at least $1 - 2^{k-d}$. To see this, observe that c_1 induces a partition of A into k parts. Within each part, there are precisely 2 ways of labeling that satisfy the conditions of Lemma A.3, i.e., fixing $i \in [k]$, the lemma requires either $r(x) = c_2(x)$ simultaneously for all $x \in A$ where $c_1(x) = i$, or $r(x) = 1 - c_2(x)$ simultaneously for all $x \in A$ where $c_1(x) = i$. Overall, there are 2^k possible choices of r such that p satisfies the conditions of Lemma A.3. The probability that r is one of these choices is 2^{k-d}.

Now we take a union bound over all $p \in \mathcal{P}|_A$. Recall that

$$|\mathcal{P}|_A| = O(k^{2d_\mathrm{N}} \cdot 2^{2d_\mathrm{VC}} \cdot d^{d_\mathrm{N}+d_\mathrm{VC}}).$$

So with probability at least

$$1 - O(k^{2d_{\mathrm{N}}} \cdot 2^{2d_{\mathrm{VC}}} \cdot d^{d_{\mathrm{N}}+d_{\mathrm{VC}}}) \cdot 2^{k-d},$$

there does not exist $p \in \mathcal{P}|_A$ satisfying the conditions of Lemma A.3. One may conclude $\mathcal{S}$ does not shatter A whenever this probability is positive, which requires setting

$$d = \Theta(k + (d_{\mathrm{N}} + d_{\mathrm{VC}}) \cdot \log(d_{\mathrm{N}} + d_{\mathrm{VC}} + k)).$$

Finally, observe that the above argument works for any $A \subseteq X$ with size $|A| = d$. This implies

$$\begin{aligned} d_{\mathrm{J}}(\mathcal{O}, \mathcal{C}) &= d_{\mathrm{VC}}(\mathcal{S}) < d \\ &= \Theta(k + (d_{\mathrm{N}} + d_{\mathrm{VC}}) \cdot \log(d_{\mathrm{N}} + d_{\mathrm{VC}} + k)), \end{aligned}$$

which concludes the proof. □