(Machine) Learning Parameter Regions*

José Luis Montiel Olea† and James Nesbit‡

June 11, 2020

Abstract

How many random points from an identified set, a confidence set, or a highest posterior density set suffice to describe them? This paper argues that taking random draws from a *parameter region* in order to approximate its shape is a supervised learning problem (analogous to sampling pixels of an image to recognize it). Misclassification error—a common criterion in machine learning—provides an off-the-shelf tool to assess the quality of a given approximation. We say a parameter region can be *learned* if there is an algorithm that yields a misclassification error of at most ϵ with probability at least $1-\delta$, regardless of the sampling distribution. We show that learning a parameter region is possible if and only if its potential shapes are not too complex. Moreover, the *tightest band that contains a d-dimensional parameter region* is always *learnable from the inside* (in a sense we make precise), with at least $\max\{(1-\epsilon)\ln(1/\delta), (3/16)d\}/\epsilon$ draws, but at most $\min\{2d\ln(2d/\delta), \exp(1)(2d+\ln(1/\delta))\}/\epsilon$. These bounds grow linearly in the dimension of the parameter region, and are uniform with respect to its true shape. We illustrate the usefulness of our results using structural vector autoregressions. We show how many orthogonal matrices are necessary/sufficient to evaluate the impulse responses' identified set and how many 'shotgun plots' to report when conducting joint inference on impulse responses. (JEL-Classification: C1, C32)

*We would like to thank Timothy Armstrong, Denis Chetverikov, Timothy Christensen, Timothy Cogley, Jianqing Fan, Lutz Kilian, Charles Manski, Mikkel Plagborg-Møller, Guillaume Pouliot, Andres Santos, Azeem Shaikh, Elie Tamer, seminar participants at Columbia University, the Federal Reserve Bank of Philadelphia, New York University, Northwestern University, Princeton University, Yale University, the 2018 Conference on Optimisation and Machine Learning in Economics, the 2019 meetings of the American Economic Association (session on Machine Learning and Shrinkage estimation), three anonymous referees and an anonymous Associate Editor at the Journal of Econometrics for helpful comments and suggestions. The usual disclaimer applies.

†Columbia University, Department of Economics, 420 West 118th St., New York, NY 10027. (jm4474@columbia.edu).

‡New York University, Department of Economics, NYU, 19 W.4th St., 6th Floor, New York, NY 10012. (jmn425@nyu.edu).

1 Introduction

Machine learning can be broadly defined as a set of data-driven computational methods used to make informed decisions in different 'learning' tasks, such as prediction, ranking and classification problems (Mohri, Rostamizadeh and Talwalkar (2012)). There is now a large and important body of work showing that machine learning algorithms can be extended and adapted to problems that are of interest for economists; for example estimation of heterogeneous treatment effects (Wager and Athey (2018)); policy evaluation with very many regressors (Belloni, Chernozhukov and Hansen (2014); Belloni et al. (2017)); and the analysis of discretized unobserved heterogeneity (Bonhomme, Lamadon and Manresa (2017)).

This paper aims to contribute to the recent gainful connection between machine learning and econometrics. The paper uses well-known concepts in the supervised learning literature—such as misclassification error, sample complexity, and the definition of learning itself—to study a common approach to describing *parameter regions* in econometric problems: sampling elements from inside of these regions at random.

To fix ideas and introduce notation, consider the problem of reporting the response of prices to a contractionary monetary shock in a sign-restricted structural vector autoregression (SVAR); see Uhlig (2005) and Faust (1998). Theory (the sign restrictions) and data (reduced-form estimators) restrict the model's structural parameters, denoted θ, to belong to some set S. The parameter region of interest, $\lambda(S)$, is the set of d-horizon impulse responses implied by the structural parameters in S; where $\lambda(\cdot)$ is the function that maps θ to the vector of impulse responses.

Describing a parameter region is complicated. Verifying whether some vector of impulse responses belongs to $\lambda(S)$ requires 'inverting' $\lambda(\cdot)$; and this is typically a hard problem. Also, the parameter region of interest is typically of more than one dimension and not much is known about its shape. This means that reporting features of $\lambda(S)$, such as the form of its boundary, is rather difficult.

A common and practical approach to describing parameter regions is *random*

sampling. This means that the econometrician chooses some probability distribution P, takes M i.i.d. draws of θ, computes $\lambda(\theta)$, and then uses this to construct some approximation $\widehat{\lambda}_M$ for the set $\lambda(S)$.

This paper argues that approximating a parameter region as described above can be phrased as a *supervised (machine) learning problem*, where the object of interest is to 'learn' $\lambda(S)$. In our leading examples, parameter regions will be thought of as either: an estimated identified set, a confidence set formed by test inversion, or a highest posterior density credible set. The supervised learning analogy allows us to use some well-known machine learning concepts to achieve two objectives. First, discipline the way we think about the accuracy of a random sampling approximation. Second, provide some guidance on the number of random draws that suffice to guarantee an accurate approximation. To the best of our knowledge, none of these issues have been addressed in the literature before.[1]

Accuracy of Random Sampling Approximations : When can we say that $\widehat{\lambda}_M$ provides a good description/approximation of $\lambda(S)$? The proposal of this paper is to use the *misclassification error* criterion, which is commonly used in the supervised learning literature (Murphy, 2012, p. 205). Imagine there is an omniscient agent (an oracle) who can easily check whether some parameter $\lambda(\theta)$ belongs to the sets $\lambda(S)$ and $\widehat{\lambda}_M$. To judge the quality of the approximation, the oracle computes how often the econometrician's approximation errs on classifying $\lambda(\theta)$ according to some probability measure Q. This is, the oracle computes

$$\mathcal{L}(\widehat{\lambda}_M; \lambda(S), Q) \equiv Q\left(\mathbf{1}\{\lambda(\theta) \in \lambda(S)\} \neq \mathbf{1}\{\lambda(\theta) \in \widehat{\lambda}_M\}\right). \tag{1}$$

The oracle has two concerns. On the one hand, he worries that—due to a possibly insufficient number of draws—the quality of the approximation provided by $\widehat{\lambda}_M$ (which is random as it depends on the sample of M i.i.d draws from P with labels

[1]The closest reference that we are aware of is the work of Bar and Molinari (2013), who propose computational methods for set-identified models via data augmentation and support vector machines. Also, the earliest reference that we found of using random sampling to approximate parameter regions in set identified models is Horowitz et al. (2003), p. 457, and Horowitz and Manski (2006), p. 424.

genreated by $\lambda(S)$) could be poor too often. On the other hand, he also worries about the econometrician's choice of probability distribution P to conduct random sampling. To protect himself against these two issues, the oracle would like the econometrician to guarantee that the number of draws has been large enough to make

$$P\left(\mathcal{L}(\widehat{\lambda}_M; \lambda(S), Q) < \epsilon\right) \geqslant 1 - \delta, \tag{2}$$

for *any* probability distribution P, and for *any* possible shape of the set $\lambda(S)$ (which both the oracle and the econometrician know to belong to some class Λ). In other words, the oracle demands that (2) be satisfied for a sample size large enough, that can only depend on the values of ϵ and δ. These accuracy parameters ensure the probability of observing a misclassification error less than ϵ occurs with probability at least $1 - \delta$, regardless of P and the shape of the parameter region $\lambda(S)$.

The econometrician's problem presented above can be described using supervised learning jargon. There is a sample $(\lambda(\theta_1), \ldots, \lambda(\theta_M))$ of 'inputs' that are i.i.d draws from a distribution P and there are also 'labels' $(l(\theta_1), \ldots, l(\theta_M))$, where $l(\theta) = \mathbf{1}\{\theta \in S\}$. Equation (1) is usually referred to as *generalization error* or simply misclassification probability (see Definition 2.1 in Mohri, Rostamizadeh and Talwalkar (2012)). When P equals Q—that is, when the measure used by the oracle to compute misclassification coincides with one used by the econometrician to generate random samples—the criterion in (2) is the *Probably Approximately Correct* (PAC) *learning* guarantee.[2] Thus, whenever P equals Q (an assumption that we will maintain in the remaining part of the paper), the econometrician's problem of summarizing $\lambda(S)$ is tantamount to using the labeled data to *(machine) learn* $\lambda(S)$.[3]

[2]See Mohri, Rostamizadeh and Talwalkar (2012) p. 13, Definition 2.3 for a textbook treatment. To the best of our knowledge, the definition of learning concepts that are defined by regions in Euclidean n-dimensional spaces was first introduced by Blumer et al. (1989), extending the seminal work of Valiant (1984).

[3]In Appendix B we argue that considering a set-up in which P and Q are different is not very interesting for at least two reasons. First, learning in the sense of (2) is generally impossible if P is allowed to be arbitrary different to Q. Second, and not surprisingly, the criterion in (2) can be satisfied if P is sufficiently close to Q; in which case the arguments and results we can obtain are very similar to the case in which $P = Q$.

An important difference with the typical machine learning setting is that in a standard classification problem P represents the distribution of the data and is thus unknown. It is therefore desirable to control misclassification error *uniformly* over P. When an econometrician tries to (machine) learn a parameter region, P becomes a choice variable. We think that insisting on results that are uniform over P is still appropriate, as it forces the econometrician to provide guarantees that the algorithm will work, regardless of what P is, as long as the number of draws is large enough.

Guidance on the number of draws: The Fundamental Theorem of Statistical Learning (Blumer et al. (1989), Theorem 2.1) allows us to prove that if Λ, the class of sets where the parameter region lives, is too complex—in the sense of having an infinite Vapnik–Chervonenkis dimension (Vapnik (1998))—then it is impossible for the econometrician to satisfy Equation (2). In econometric applications, this result will bind often. For example, some assumptions that are often thought to simplify the analysis of econometric problems (such as the restricting parameter regions to be convex sets), do not simplify the supervised learning problem.[4] Note that the choice of concept class Λ is not only a theoretical concern: it defines the objects that the approximation algorithm can output.

We circumvent this impossibility result by making two modifications to the definition of learning in Equation (2).

First, we assume that both the oracle and the econometrician agree to focus on learning the *tightest band containing the parameter region*. Bands—which are defined as products of intervals in each dimension—are a convenient compromise, for they are often used to summarize uncertainty in the estimation of vector-valued parameters, particularly in the SVAR literature. Moreover, bands are objects of low complexity, regardless of the underlying shape of the parameter region of interest. We relax the definition of learning by assuming the oracle computes misclassification error in Equation (1) with respect to the tightest band containing the parameter

[4] If Λ is the class of convex subsets of $\mathbb{R}^d$ with $d > 1$, there is no algorithm satisfying Equation (2) that can be used to approximate $\lambda(S)$ by means of random sampling. This is because the class of convex subsets of $\mathbb{R}^d$ with $d > 1$ has infinite VC dimension.

region of interest. Throughout the paper we denote such tightest band as $[\lambda(S)]$.[5]

Second, we restrict the class of probability distributions that both the econometrician and the oracle can consider. We show that learning the tightest band continues to be difficult, for the set-difference between $[\lambda(S)]$ and $\lambda(S)$ can be attached an arbitrarily high probability. To avoid this problem, both the econometrician and the oracle agree to consider only probability distributions that sample from *inside* the parameter region of interest. In particular, we focus on distributions for which $P(\theta \in S) = 1$, which implies $P(\lambda(\theta) \in \lambda(S)) = 1$.

Under these two modifications, we show that the tightest band that contains the parameter region can be *learned from the inside*, in a sense made precise but analogous to (2). The algorithm for learning $[\lambda(S)]$ from the inside consists of reporting the largest and smallest values of the random draws inside $\lambda(S)$, along each dimension. The main result of this paper (Theorem 3) shows that the *sample complexity* of this algorithm—that is, the minimal number of draws required to make learning possible—can be bounded from above by $\min\{2d\ln(2d/\delta), \exp(1)(2d+\ln(1/\delta))\}/\epsilon$ and below by $\max\{(1-\epsilon)\ln(1/\delta), (3/16)d\}/\epsilon$.[6] The bounds depend on neither the set S nor the specifics of the function $\lambda(\cdot)$. The only relevant information is the space in which the parameter region $\lambda(S)$ lives. We derive these bounds using proof techniques from the statistical learning literature, in particular Blumer et al. (1989); Ehrenfeucht et al. (1989); Auer, Long and Srinivasan (1998); Shalev-Shwartz and Ben-David (2014).

[5] Another possibility would be to consider other sets to define the output of our algorithm. For example, the *tightest ellipse* containing a parameter region $\lambda(S)$. The VC dimension of d-dimensional ellipsoids is $(d^2+3d)/2$ (Akama and Irie (2011)). Theorem 1 implies that there is an algorithm to learn ellipses. We do not pursue this direction because, in contrast to bands, ellipses are hard to visualize in high dimensions.

[6] In some problems, instead of using random sampling, one can solve for bands by solving constrained maxima/minima problems in each dimension:

$$\min_{\theta \in S} \lambda_j(\theta) \text{ and } \max_{\theta \in S} \lambda_j(\theta),$$

where $\lambda_j(\theta)$ is the j^{th} coordinate of $\lambda(\theta)$. However this approach requires that the optimization problem be sufficiently well-behaved, which may or may not hold depending on the application. The main advantage of random sampling to learn bands is that it requires no special structure in the problem. See Section 3.1 for discussion of the potential difficulties of evaluating the identified set in SVAR's using this maximization approach.

We illustrate our results using two examples motivated by recent research in SVARs (see Kilian and Lütkepohl (2017) for a modern, comprehensive treatment of the topic).

First, we examine the question of how many orthogonal matrices are necessary or sufficient for constructing identified sets of impulse responses in a sign-identified SVAR model. We use random sampling to evaluate a natural estimator of the impulse responses' identified set in a sign-restricted model. We fix the model's reduced-form parameters at their sample estimates and use random draws from the algorithm of Rubio-Ramirez, Waggoner and Zha (2010) (henceforth, RRWZ). With $\epsilon = \delta = 0.1$ (misclassification error of at most 10% with probability at least 90%), the number of draws that suffice to approximate the 16-quarters ahead identified set (of one variable to one shock) is 987. In our empirical application, this translates to approximately $3,000$ iterations of the RRWZ algorithm.

Second, we study the question of how many draws are required when conducting joint inference on structural impulse responses in a point identified SVAR model. We also use random draws to generate 'shotgun plots' (Inoue and Kilian (2013, 2016, 2019)) in a point-identified SVAR model. The objective is to describe both a 68% Wald-ellipse and a 68% highest posterior density set for structural impulse response functions. We take two thousand draws—which for a 68% confidence set implies $1,360$ draws from inside the parameter region—and report an *iso-draw curve*. Namely, all the combinations of (ϵ, δ) that could be supported with this number of draws. Our formulae imply that $2,000$ total draws to summarize a 68% Wald ellipse are sufficient to support the combination $\epsilon = \delta = 0.0732$. In particular, this implies that $2,000$ total draws are sufficient to guarantee that with probability at least 92.68% probability, the misclassification error less than 7.32%.

In these applications the concept class in which $\lambda(S)$ lives is too complicated and thus not learnable. In the three applications we focus on learning the tightest band that contains the parameter region. Consequently, the relevant number of draws for these applications has to come from inside of the parameter region.

OUTLINE: Section 2 presents our main definitions and theoretical results. Section 3 presents our SVAR application. Section 4 concludes. Appendix A contains proofs. Appendix B discusses the learning problem when Q (the measure used by the oracle to compute misclassification error) differs from P (the measure used by the econometrician to generate random draws). Appendix C presents an application to learning the range of posterior means in a set-identified model used in natural language processing.

2 Theory

Let $\Theta \subseteq \mathbb{R}^p$ denote the parameter space for the finite-dimensional component of a parametric or semi-parametric statistical model. Let us assume that due to either theory, or data, or both, the econometrician is able to restrict the values of $\theta \in \Theta$ to belong to some measureable subset $S \subseteq \Theta$. Assume also that the indicator function $l(\theta) \equiv \mathbf{1}\{\theta \in S\}$ can be computed without difficulty, so that each element of θ can be given a binary label of whether it belongs to S (label 1) or not (label 0).

The examples we have in mind are as follows. The set S could be an estimator of an *identified set*; in this case S would contain the parameter values that satisfy some restriction (like a sample moment inequality or a sign restriction). S could also be a *confidence region* obtained by test inversion; in this case S would represent the set of θ values such that, when postulated as a null hypothesis, cannot be rejected. S could also be a *highest posterior density credible set*; in this case S would represent the set of parameter values for which the posterior density is above some threshold.[7]

We allow for the possibility that the parameter of interest is not θ per se, but instead the image of θ under some measurable function $\lambda : \Theta \to \mathbb{R}^d$. This will be relevant in our leading example, a set-identified SVAR, where—as discussed in the introduction—λ represents the impulse response coefficients over different horizons. More generally, λ could report a subvector of θ of dimension $d < p$, or if θ is the

[7]The results in the paper assume that the set S is the object of interest, but in Appendix A.7 we discuss the consequences of observing S with sampling uncertainty.

object of interest, λ could be the identity map. For Theorem 1 it will be important to maintain the assumption that $\lambda(\cdot)$ is injective. We will drop this assumption in Theorems 2 and 3.

As we mentioned in the introduction, the econometrician is interested in describing the set $\lambda(S)$, which mathematically is the image of the set S under λ.[8] We will refer to this set as the *parameter region*. To describe a parameter region, the econometrician chooses a distribution P over Θ, generates a sample of size M and computes $\lambda(\theta_m)$. Each of the elements in the sample has a label $l(\theta_m)$. Note that, under the assumption that $\lambda(\cdot)$ is injective, $l(\theta_m) = 1$ if and only if $\lambda(\theta_m) \in \lambda(S)$, thus the label tells us whether $\lambda(\theta_m)$ belongs to the parameter region $\lambda(S)$ or not.

2.1 Learning $\lambda(S)$

In our set-up, the shape of the parameter region $\lambda(S)$ is not known. To capture this lack of knowledge it will be assumed that $\lambda(S)$ belongs to some class of sets $\Lambda \subseteq 2^{\lambda(\Theta)}$. We will refer to Λ as a *concept class* and we will call each of its elements, $\boldsymbol{\lambda}$, a *concept*.[9] Note that the choice of concept class Λ is not only a theoretical concern: it defines the objects that the algorithm can output.

Our supervised learning problem is formulated as follows. The econometrician (or learning agent) generates a sample of size M, drawn i.i.d. from some distribution P; evaluates these θ-draws under λ, and generates labels that inform whether a draw $\lambda(\theta_m)$ belongs to $\lambda(S)$ or not. Checking whether a draw of $\lambda(\theta_m)$ belongs to $\lambda(S)$ is, in principle, quite difficult unless we make some additional assumptions. One way of achieving this is by assuming $\lambda(\cdot)$ is injective (in which case, we only need to check whether $\theta_m \in S$). We will use this assumption to establish Theorem 1. Another possibility, without imposing restrictions on $\lambda(\cdot)$, is to consider a probability distribution P that places all of its mass on S. This will be the set-up of Theorem 3. The econometrician's task is to use a sample $\{(\lambda(\theta_m), l(\theta_m))\}_{m=1}^{M}$ to select a concept

[8] The image of the set S under a function λ is defined as $\lambda(S) \equiv \{\lambda \mid \exists\, \theta \in S \text{ s.t. } \lambda = \lambda(\theta)\}$.

[9] We use this terminology in order to establish a closer connection to the supervised learning literature.

$\widehat{\lambda}_M \in \Lambda$ that approximates the true concept $\lambda(S)$. A mapping from samples to concepts is called an *algorithm*.

Let $\mathcal{L}$ denote the generalization error defined in equation (1) assuming Q (the measure used by the oracle) equals P (the measure used by the econometrician to generate random draws). Since the shape of $\lambda(S)$ (known to the oracle, but unknown to the econometrician) is allowed to be any set $\boldsymbol{\lambda} \in \Lambda$, we will define misclassification error relative to $\boldsymbol{\lambda}$ as

$$\mathcal{L}(\widehat{\lambda}_M; \boldsymbol{\lambda}, P) \equiv P\left(\mathbf{1}\{\lambda(\theta) \in \boldsymbol{\lambda}\} \neq \mathbf{1}\{\lambda(\theta) \in \widehat{\lambda}_M\}\right).$$

We will say that the concept $\lambda(S)$, in the class Λ, can be *learned* if it satisfies the following definition:

Definition 1 (Learnability of $\lambda(S)$)**.** *The concept $\lambda(S) \in \Lambda$ is said to be* learnable *if there exists an algorithm $\widehat{\lambda}_M$ and a finite function $m(\epsilon, \delta)$ such that for any $0 < \epsilon, \delta < 1$:*

$$P\left(\mathcal{L}(\widehat{\lambda}_M; \boldsymbol{\lambda}, P) < \epsilon\right) \geqslant 1 - \delta,$$

for all distributions P on Θ and for any $\boldsymbol{\lambda} \in \Lambda$, whenever $\boldsymbol{\lambda}$ represents the true set $\lambda(S)$; provided $M \geqslant m(\epsilon, \delta)$.

The concept of learnability in Definition 1 is known in the statistical learning literature as Probably Approximately Correct (PAC) learning. The parameter ϵ determines how far (in terms of generalization error) the concept returned by the algorithm $\widehat{\lambda}_M$ is from the true concept $\lambda(S)$ (this is the 'approximately correct' part). The parameter δ indicates how often the algorithm will yield a misclassification probability larger than ϵ (this is the 'probably' part).

Perhaps without a surprise, our ability to learn will depend on how rich the concept class Λ is. We formalize this argument in the following theorem:

Theorem 1. *Suppose $\lambda(\cdot)$ is injective. $\lambda(S) \in \Lambda \subseteq 2^{\lambda(\Theta)}$ is learnable if and only if*

Λ *has finite Vapnik-Chervonenkis (VC) dimension.*

Proof. See Appendix A.2. □

In a nutshell, Theorem 1 states that a concept class is learnable if and only if it is not too complex. We prove Theorem 1 by invoking the *Fundamental Theorem of Statistical Learning* (FTSL). See Chapter 6.4 in Shalev-Shwartz and Ben-David (2014) for a textbook treatment or Theorem 2.1 in Blumer et al. (1989) p. 935 for the statement of the result as used in the proof of Theorem 1.[10]

The versions of Theorem 1 herein referenced also show that the number of draws required for learning can be bounded from above in terms of ϵ, δ and the VC dimension of the concept class Λ, without reference to P or the specific shape of the parameter region. We will port some of the tools and techniques from the statistical learning literature to derive analogous bounds for the tightest band containing a parameter region. To do so, we will impose some restrictions on the probability distributions P under consideration.

Theorem 1 emphasizes that approximating the unknown parameter region $\lambda(S)$ will require the econometrician to take a stand on the complexity of the concept class Λ in which the algorithm takes values (and this class has to be correctly specified). If this class is too complex—in the sense of having infinite VC dimension—then learning is not possible.[11]

The restriction on the complexity of learnable concept classes is relevant in applications. For example, even certain restrictions that seem to simplify the approximation problem (like restricting Λ to be the class of convex sets so that they can be summarized using their support function) are usually not enough.[12] The final message of this section is that learning $\lambda(S)$, in the conventional sense of the word, is difficult and oftentimes impossible.

Finally, in order to prove Theorem 1 we restricted the function of interest, $\lambda(\cdot)$,

[10] An application of VC dimension as a measure of complexity of decision rules in decision making along with an application of the FTSL can be found in Al-Najjar (2009); Al-Najjar and Pai (2014).
[11] See Appendix A.1 for a definition and discussion of VC dimension.
[12] If $d > 1$ then the VC dimension of the class of convex sets in $\mathbb{R}^d$ is infinity.

to be injective. This avoids the problem of having two different points, θ and θ', only one of them belonging to S (hence with different labels $l(\theta) \neq l(\theta')$), but both mapping to the same point under λ, i.e. $\lambda(\theta) = \lambda(\theta')$. We view the injectivity assumption as being sufficient, but not necessary, for Theorem 1 to hold (we do not have an example of a set S where the VC dimension is finite, λ is injective, but $\lambda(S)$ is not learnable). One possibility for obtaining a more general version of the theorem could be to allow for non-deterministic labels as in the *agnostic learning* framework (see Theorem 6.7 in Shalev-Shwartz and Ben-David (2014)). We decided not to pursue this generalization in this paper. In addition, the injectivity assumption will be satisfied in the SVAR illustrative example in Section 3.

2.2 Learning $[\lambda(S)]$

With the impossibility result of Theorem 1 in mind, we introduce the notion of the *tightest band that contains the parameter region* $\lambda(S)$. We want to argue that such a band is *learnable from the inside* in a sense we will make precise.

The tightest band containing the parameter region $\lambda(S)$ is defined as the hyper-rectangle

$$[\lambda(S)] \equiv \bigtimes_{j=1}^{d} \left[\inf_{\theta \in S} \lambda_j(\theta) \, , \, \sup_{\theta \in S} \lambda_j(\theta) \right],$$

where $\lambda_j(\theta)$ denotes the j^{th} coordinate of $\lambda(\theta)$.

Figure 1 displays an example of a parameter region $\lambda(S)$ of strange shape along the band $[\lambda(S)]$.

Bands for vector-valued parameters are versatile tools for visualizing estimation uncertainty in econometric problems (see Horowitz and Lee (2012), Freyberger and Rai (2018), Montiel Olea and Plagborg-Møller (2019)). For example, bands for impulse response functions at different horizons are typically reported in SVAR applications.

In the context of statistical learning theory, bands (usually referred to as *axis-*

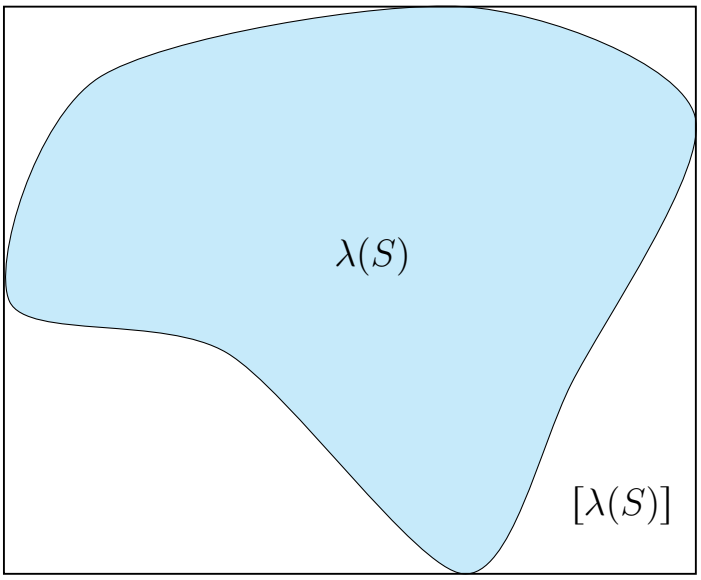

Figure 1: $\lambda(S)$ and $[\lambda(S)]$.

aligned hyperrectangles) are objects of low complexity: the VC dimension of the set of all bands in $\mathbb{R}^d$ is $2d$. Thus, in light of Theorem 1, if the concept class Λ, to which $\lambda(S)$ belongs, consisted only of bands, then $\lambda(S)$ would be learnable.

For a given set $\lambda(S)$, we have defined $[\lambda(S)]$ to be the smallest band containing $\lambda(S)$. Abusing notation, we will now specifically use $[\widehat{\lambda}_M]$ to denote the algorithm that reports the smallest rectangle that contains all of the elements of the sample $\{\lambda(\theta_m)\}_{m=1}^M$ that have positive labels (as opposed to the 'banded' version of an arbitrary algorithm $\widehat{\lambda}_M$). This algorithm keeps track of the maximum and minimum value of the random draws in each dimension (provided those draws are in the set we want to learn), and it is typically used for learning bands.

Definition 2 (Learning algorithm for bands)**.** *Given a sample* $\boldsymbol{\theta}_M \equiv (\theta_1, \ldots, \theta_M)$ *with labels* $\boldsymbol{l}_M \equiv (l(\theta_1), \ldots, l(\theta_M))$*, let* $[\widehat{\lambda}_M]$ *denote the algorithm that reports*

$$[\widehat{\lambda}_M] \equiv \bigtimes_{j=1}^{d} \left[\min_{m|l(\theta_m)=1} \lambda_j(\theta_m), \max_{m|l(\theta_m)=1} \lambda_j(\theta_m) \right],$$

where $\lambda_j(\theta)$ *is the* j^{th} *coordinate of* $\lambda(\theta)$*.*

Note that if there is no draw θ_m for which $l(\theta_m) = 1$, the algorithm above outputs

the empty set.[13]

Can the algorithm $[\widehat{\lambda}_M]$ learn $[\lambda(S)]$ in a sense analogous to Definition 1? To be more precise, we would like to know if there exists a function $m(\epsilon, \delta)$ such that

$$P\left(\mathcal{L}([\widehat{\lambda}_M]; [\boldsymbol{\lambda}], P) < \epsilon\right) \geqslant 1 - \delta, \tag{3}$$

for any distribution P on Θ, and for any $\boldsymbol{\lambda} \in \Lambda$, provided $M \geqslant m(\epsilon, \delta)$ and assuming the true labels are generated according to $\boldsymbol{\lambda}$? This last point is important, because if the true labels were generated by $[\boldsymbol{\lambda}]$, then Theorem 1 would imply the existence of a learning algorithm (and in fact, $[\widehat{\lambda}_M]$ would be one such algorithm).[14] Unfortunately, Theorem 2 below shows that even if we allow ourselves to compute misclassification error relative to $[\boldsymbol{\lambda}]$, learning is still not possible due to two different features of the problem: a) the richness of the class of probability distributions under consideration, and b) the fact that the true labels are generated by $\boldsymbol{\lambda}$ and not $[\boldsymbol{\lambda}]$.

Theorem 2 (Impossibility of learning bands)**.** *Suppose there exists a concept $\boldsymbol{\lambda} \in \Lambda$ that is not a band; that is $[\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda} \neq \varnothing$. Suppose further that there exists a probability distribution that places arbitrarily large mass on the set $[\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda}$. That is, for any $\eta \in (0, 1)$ there exists P_η over Θ such that:*

$$P_\eta\left(\lambda(\theta) \in [\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda}\right) \geqslant \eta.$$

Under the assumptions above, $[\widehat{\lambda}_M]$ cannot learn $[\lambda(S)]$ in the sense of (3). Moreover, there is no algorithm $\widehat{\lambda}_M$ (that outputs bands or any other set) that both i) returns the empty set whenever $l(\theta_m) = 0$ for all $m = 1, \ldots, M$ and ii) satisfies

$$P\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P) < \epsilon\right) \geqslant 1 - \delta,$$

[13] To keep the notation as simple as possible, we have decided not make dependence of the algorithm $[\widehat{\lambda}_M]$ on the sample $\boldsymbol{\theta}_M$ and the labels $\boldsymbol{l}_M$ explicit. If confusion arises, we shall write $[\widehat{\lambda}_M](\boldsymbol{\theta}_M, \boldsymbol{l}_M)$, but we remind the reader that $[\widehat{\lambda}_M]$ is only used to denote the algorithm in Definition 2, and not the banded version of some arbitrary algorithm.

[14] To see this, simply let S be $[\boldsymbol{\lambda}]$, and let $\lambda(\cdot)$ be the identity. Therefore, the concept class Λ becomes the set of all bands, which has finite VC dimension and, by Theorem 1, is learnable.

for any distribution P on Θ, and for any $\boldsymbol{\lambda} \in \Lambda$, provided $M \geqslant m(\epsilon, \delta)$ and assuming that the true labels are generated by $\boldsymbol{\lambda}$.

Proof. See Appendix A.3. □

Theorem 2 assumes that the true labels are being generated by $\boldsymbol{\lambda}$ instead of by S. As we mentioned in the introduction this is typically infeasible, unless we make additional assumptions (for example, assuming injectivity of $\lambda(\cdot)$, in which case we could check whether $\lambda(\theta) \in \boldsymbol{\lambda}$ simply by checking if $\theta \in S$). Theorem 2 shows that *even if* we had access to labels generated by $\boldsymbol{\lambda}$, no algorithm could learn $[\boldsymbol{\lambda}]$.

2.3 Learning $[\lambda(S)]$ from the inside

Theorem 1 gives a necessary and sufficient condition for learnability. Surprisingly, Theorem 2 demonstrates that even when we focus on algorithms that output bands (and thus allow us to ignore the complexity of Λ), learning continues to be difficult, even if misclassification error is defined relative to the tightest band and not the true set. We think the result is surprising because bands have finite VC dimension, and thus should be learnable.

Theorem 2 shows that the richness of the class of probability distributions for which equation (3) must hold is one of the determinants of the impossibility result. It is also crucial that we have introduced a difference between the true set and the tightest band containing it (in particular, misclassification error is computed relative to $[\boldsymbol{\lambda}]$, but the true labels are generated by $\boldsymbol{\lambda}$). If we allow probability distributions that place arbitrarily large mass on the difference between $[\boldsymbol{\lambda}]$ and $\boldsymbol{\lambda}$ then, with high probability, we will get samples with only 0-labels. As we showed above, this would lead to an arbitrarily large misclassification probability.

To further illustrate the difficulties in learning $[\boldsymbol{\lambda}]$ consider the following example. Suppose

$$\Lambda = \{[a, b] | a \leqslant b\} \cup \{[0, 1] \cup \{2\}\}.$$

That is, the concept class is the set of all intervals $[a, b]$ and an additional set formed

by attaching $\{2\}$ to the $[0,1]$ interval. It can be shown that the VC dimension of Λ is finite and thus, by Theorem 1, there exists an algorithm that learns the class.

Consider now the learning requirement in equation (3), which computes misclassification error relative to $[\boldsymbol{\lambda}]$, but assumes that the true labels are generated using $\boldsymbol{\lambda}$. Suppose that the true set generating labels is $\boldsymbol{\lambda} = [0,1] \cup \{2\}$, which differs from the smallest band containing this set; as $[\boldsymbol{\lambda}] = [0,2]$. Consider a probability distribution P that places all of its mass on $[\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda} = (1,2)$. Then, any sample from this distribution will have only zero labels (as the true labels are generated by $\boldsymbol{\lambda} = [0, 1] \cup \{2\}$). This is a problem for $[\widehat{\lambda}_M]$ because this algorithm reports the empty set absent positive labels. This implies that the misclassification error relative to $[0,2]$ is 1. This argument is the essence of Theorem 2.

A natural question to ask is whether an algorithm different to $[\widehat{\lambda}_M]$ can fare better. Theorem 2 answered this question in the negative, provided we focus on algorithms that report the empty set whenever the sample does not contain positive labels. In our simple example it is possible to show that, even if we consider an algorithm that outputs an arbitrary set absent positive labels, learning in the sense of equation (3) is not possible. We do this in Appendix A.6. A key restriction that we continue to impose is that the set reported when there are no positive labels, does not depend on the sampled λ's.

One way to get around this problem, is to restrict the class of distributions that the econometrician can use to conduct random sampling from S. In particular, we define the set

$$\mathcal{P}(S) \equiv \{P \mid P \text{ is a distribution on } \Theta \text{ and } P(S) = 1\}.$$

Note that $\mathcal{P}(S)$ is the collection of all probability distributions that sample from *inside* the set S, and thus from inside the parameter region $\lambda(S)$. This means that for any $P \in \mathcal{P}(S)$, if $\boldsymbol{\lambda}$ is the true set, then $P(\lambda(\theta) \in \boldsymbol{\lambda}) = P(\theta \in S) = 1$ and therefore $P(\lambda(\theta) \in [\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda}) = 0$. We use this class to relax the learning desideratum

presented in Definition 1.[15]

In general, it is non-trivial to obtain draws from inside of the set S. In specific applications, such as SVARs, there are algorithms that address this problem (see Algorithm 2 of Amir-Ahmadi and Drautzburg (2017)). In some other applications, such as the text analysis problem described in the Appendix C, it is still possible to suggest algorithms to draw from inside S.

We also think that a plus of our theory (not a con) is that it is not important to receive any information on how many points in total (from both inside and outside the set S) end up being sampled. The accuracy of the approximation only depends on how many points are obtained from the inside.

In some applications, S can be lower dimensional compared to the parameter space Θ (think, for example, of SVARs with equality restrictions). This raises the question of whether it is possible to sample from inside S. We think that whether or not it is a daunting task to sample from inside a lower dimensional set really depends on the application. For the SVAR application with equality restrictions, the set S is lower dimensional compared to Θ, but one can still generate draws from the set by imposing equality restrictions (see for example how the robust Bayes algorithm of Giacomini and Kitagawa (2018) deals with equality restrictions).

Theorem 3 below provides an explicit formula for the number of draws that suffice to learn the set $[\lambda(S)]$ from the inside. The formula depends on accuracy parameters (ϵ,δ), and on the dimension of the space where $\lambda(\theta)$ lives, which we have assumed to be $\mathbb{R}^d$. The theorem also provides a formula for the number of draws that are necessary to learn $[\lambda(S)]$

Theorem 3. *The algorithm* $[\widehat{\lambda}_M]$ *in Definition 2 learns* $[\lambda(S)]$ *from the inside, whatever shape* $\lambda(S)$ *has. That is there exists a finite function* $m(\epsilon,\delta)$ *such that for*

[15] Another way to get around this problem would be to restrict $\lambda(S)$ to be a subset of a compact set, to have nonempty interior, which cannot be arbitrarily small, and to assume that the density of the random draws is bounded away from 0 on the compact set. We chose to not follow this path, as simply restricting P to be in $\mathcal{P}(S)$ still allows us to conveniently leverage results from the statistical learning literature.

any $0 < \epsilon, \delta < 1$*:*

$$P\left(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) < \epsilon\right) \geqslant 1 - \delta,$$

for all distributions $P \in \mathcal{P}(S)$ *on* Θ*, and whatever the shape of* $\lambda(S)$ *is, provided* $M \geqslant m(\epsilon, \delta)$*. Moreover, the sample complexity of the algorithm—that is, the smallest function* $m^*(\epsilon, \delta)$ *that makes learning from the inside possible—admits the upper bound:*

$$m^*(\epsilon, \delta) \leqslant \min\{2d \ln(2d/\delta)\ ,\ \exp(1)(2d + \ln(1/\delta))\}/\epsilon.$$

Additionally if $\lambda(S)$ *contains at least two different points, then*

$$(1 - \epsilon) \ln(1/\delta)/\epsilon \leqslant m^*(\epsilon, \delta).$$

If in addition, $\lambda(S)$ *satisfies the regularity condition Assumption 3 stated in Appendix A.4, and if* $\epsilon, \delta \leqslant \frac{1}{8}$ *then* $m^*(\epsilon, \delta)$ *then*

$$(3/16)d/\epsilon \leqslant m^*(\epsilon, \delta).$$

Proof. See Appendix A.4. □

The upper bound on the sample complexity, $m^*(\epsilon, \delta)$, provides a very concrete recommendation on the number of draws that suffice to learn the set $[\lambda(S)]$ from the inside. For example, in the context of a sign-restricted SVAR, the upper bound to learn the tightest band that contains any k coefficients of the impulse response function is $\min\{2k \ln(2k/\delta), \exp(1)(2k + \ln(1/\delta))\}/\epsilon$ draws. For $\epsilon = \delta = 0.01$ (misclassification error of at most 1% with probability 99%) and $k = 25$ the recommendation of Theorem 3 is that 14,844 draws of impulse response coefficients that satisfy the sign restrictions are sufficient to learn.[16] The lower bound also reveals the number

[16] Theorem 3 implies that if one has M draws of θ, $m^*(\epsilon, \delta)$ of them must come from inside of S. This means that one should choose M such that $\sum_{i=1}^{M} l(\theta_i)$ equals the upper bound on $m^*(\epsilon, \delta)$.

of draws that are necessary to learn. In the context of the SVAR application the number equals 469 draws.

The necessary and sufficient conditions for the number of draws in Theorem 3 only depend on (ϵ, δ) and the dimension in which the parameter region $\lambda(S)$ lives. There is no specific reference to neither the set S nor the function $\lambda(\cdot)$. This is a feature of the 'learning bounds' provided in the statistical learning literature, and typically reported as part of the Fundamental Theorem of Statistical Learning. The proof of our main theorem is based on the idea that bands, which have a VC dimension of $2d$, are learnable using the algorithm we described in Definition 2. Our proof is not a corollary of existing results in the statistical learning literature, but is based on tools and techniques that are used there. In particular, we used the general construction for a lower bound provided in Blumer et al. (1989) and the specific upper bound they report for rectangles in $\mathbb{R}^2$. We then sharpened the lower bound using arguments in Ehrenfeucht et al. (1989) that refer to general learning problems. In order to obtain an upper bound $O(d)$ we relied on the results of Auer, Long and Srinivasan (1998), which are specific to learning axis-aligned rectangles. Finally, we note that bounds reported in Theorem 3 seem to be consistent with state-of-the-art VC bounds in the statistical learning literature.[17]

We understand that it might be difficult for the researcher to take a stand on his/her desired combination of ϵ and δ. Unfortunately, our theory does not provide a concrete recommendation for choosing these tuning parameters (despite attaching a very specific meaning to them). Our bound can still be of practical use in those cases. For any number of draws the researcher is willing to take, we can associate all possible combinations of (ϵ, δ) that would make our upper bound return such a number. We refer to such mapping as an "iso-draw" curve and we display it in

[17] For example, Theorem 6.8 in Shalev-Shwartz and Ben-David (2014) shows that there exist constants C_1, C_2 such that a class of concepts with VC dimension equal to d is PAC learnable with sample complexity

$$C_1[d + \ln(1/\delta)]/\epsilon \leqslant m^*(\epsilon, \delta) \leqslant C_2[d \ln(1/\epsilon) + \ln(1/\delta)]/\epsilon.$$

Because we focus on learning bands, we can remove the $\ln(1/\epsilon)$ term from the upper bound and characterize the constants C_1, C_2.

Figure 2 for a parameter region of dimension $d = 25$. Our recommendation for practitioners is to report the iso-draw curve associated to a whatever many draws are feasible in their specific application.

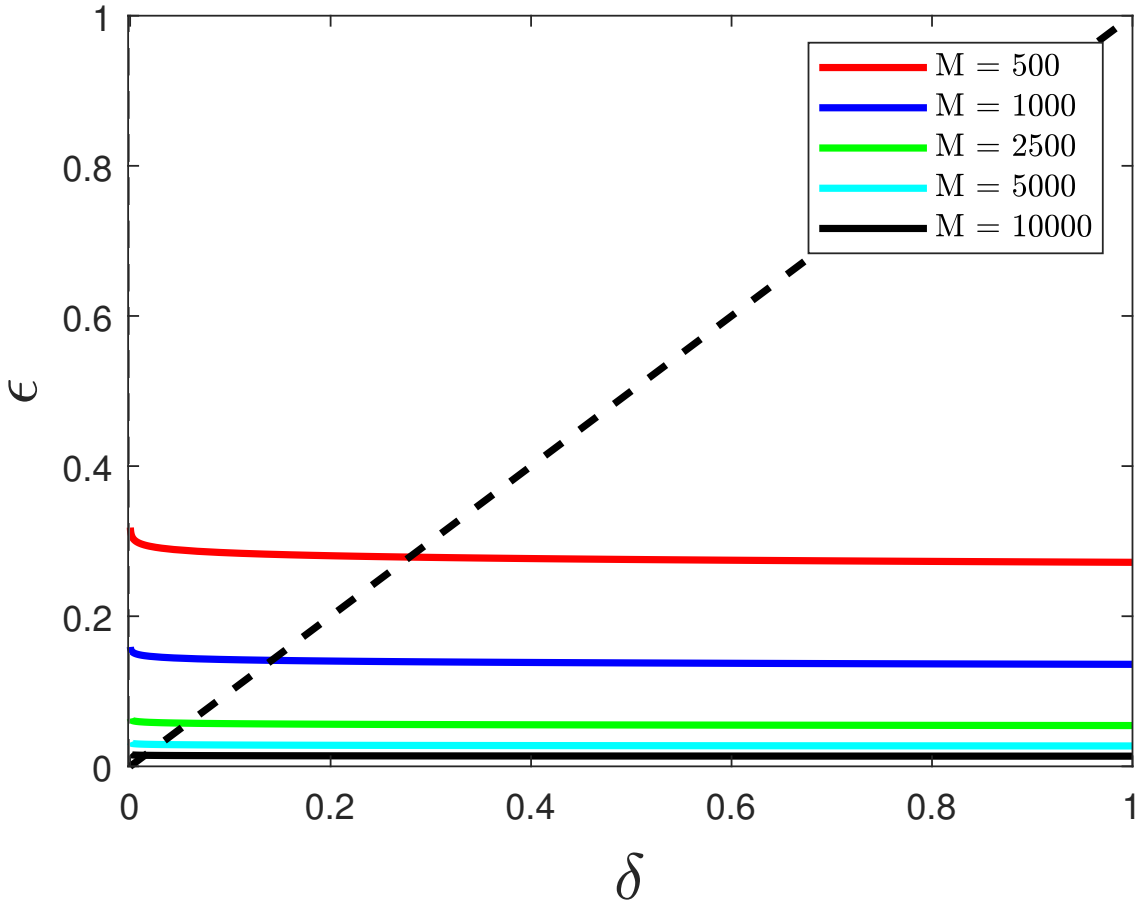

Figure 2: Iso-draw curves. For a fixed M, the combinations of (ϵ, δ) such that $M = \min\{2d\ln(2d/\delta), \exp(1)(2d + \ln(1/\delta))\}/\epsilon$. In this example $d = 25$.

The theorem in this section differs quite substantially from those that one would usually see in the statistical learning literature. Instead of trying to learn the true set, we are trying to learn a crude approximation of it. This approximation can be learned, even though we only have labels for $\lambda(S)$ and not $[\lambda(S)]$. The price that we pay for this, is that we can only guarantee learning for distributions that draw from inside $\lambda(S)$.

Note that the bounds on the sample complexity grow linearly with the dimension of the set we are trying to learn (d), not the set in which the labels are generated (p). Clearly when λ lives in a lower dimensional space, this can substantially reduce

the number of draws required to learn. Note also that the bounds are tight in the sense that they are both of order $O((d + \ln(1/\delta))/\epsilon)$.[18]

2.4 Hausdorff distance between $[\widehat{\lambda}_M]$ and $[\lambda(S)]$

An important concern about Theorem 3 is the usefulness/interpretation of the concept of learnability.

One important drawback is that our result is silent about the choice of P, and unfortunately, certain choices of P in combination with certain shapes of $\lambda(S)$ can lead to a large Hausdorff distance between the sets $[\lambda(S)]$ and $[\widehat{\lambda}_M]$, even when misclassification error is low. In this section we illustrate this point by means of two simple examples. We describe the first example analytically, and then we explain the second with a figure.[19]

In this section we also try to present a slightly more constructive result. We argue that if the Hausdorff distance between sets replaces misclassification error as the loss function, it is theoretically possible to deem some choices of P better than others. Specifically, we show that for certain distributions, a random sampling approximation that yields a low misclassification error also yields a low *worst-case relative* Hausdorff distance (in a sense made precise). In fact, under some assumptions, we show that the relative Hausdorff distance is bounded above by the misclassification error.

2.4.1 Certain choices of P and some sets $\lambda(S)$ lead to a large Hausdorff distance

In order to introduce our examples we start by focusing on the Hausdorff distance between $[\lambda(S)]$ and $[\widehat{\lambda}_M]$, but relative to the 'worst-case' distance that can be attained over the different shapes $[\widehat{\lambda}_M]$. The *worst-case relative Hausdorff* distance

[18] To see this, note that the lower bound can be bounded below by $(3/36)(\ln(1/\delta) + 2d)/\epsilon$, as $\max x, y \leqslant 1/2(x + y)$.

[19] We would like to thank an anonymous referee for suggesting these examples.

between $[\lambda(S)]$ and $[\hat{\lambda}_M]$ is

$$\tilde{d}_H\left([\lambda(S)],[\hat{\lambda}_M]\right) \equiv \frac{d_H\left([\lambda(S)],[\hat{\lambda}_M]\right)}{\sup_{b\subseteq[\lambda(S)]} d_H\left([\lambda(S)],b\right)}, \tag{4}$$

where $d_H(A,B)$ is the Hausdorff distance between sets A, B, and b denotes a *band* (i.e., a set of the form $\times_{j=1}^{d}[\underline{r}_j,\overline{r}_j]$) contained in $[\lambda(S)]$.[20] We use this normalization to measure misclassification error and Hausdorff distance on the same scale (the $[0,1]$ interval).

Suppose that set S contains two points λ_1 and λ_2. Set $\lambda(\cdot)$ to be the identity so that $\lambda(S)$ contains only two real numbers; that is $\lambda(S)=\{\lambda_1,\lambda_2\}$, with $\lambda_1,\lambda_2\in\mathbb{R}$. Suppose further that $\lambda_1<\lambda_2$.

In this case, the smallest band containing $\lambda(S)$ is given by $[\lambda(S)]=[\lambda_1,\lambda_2]$. The measures that we are interested in, place all of their mass on $\lambda(S)$. Thus, distributions of the form $P(\lambda_1)=p$ and $P(\lambda_2)=1-p$, exhaust the list of distributions that can be used to learn from inside.

Given M i.i.d. draws from these distributions, there are only three possible shapes that the algorithm $[\widehat{\lambda}_M]$ can output: i) the interval $[\lambda_1,\lambda_2]$, ii) the point λ_1, or iii) the point λ_2. A direct calculation shows that the Hausdorff distance between these sets and true set $[\lambda(S)]$ is i) 0, ii) $\lambda_2-\lambda_1$, and iii) $\lambda_2-\lambda_1$, respectively.

Since $\lambda_1<\lambda_2$ by assumption, the worst-case Hausdorff is $\lambda_2-\lambda_1$. Therefore, the relative Hausdorff distance, $\tilde{d}_H\left([\lambda(S)],[\hat{\lambda}_M]\right)$, is i) 0, ii) 1, or iii) 1. On the other hand, misclassfication error between $[\lambda(S)]$ and $[\widehat{\lambda}_M]$ is i) 0, ii) $1-p$, or iii) p.

Suppose now that we choose a probability distribution with $p=0$ (this type of distribution is allowed in Theorem 3), so that we only draw λ_2. In this case, the only possible output the algorithm will give is $[\widehat{\lambda}_M]=\lambda_2$, regardless of the number of draws.

[20] For two sets A, B in $\mathbb{R}^d$ and a norm d in $\mathbb{R}^p$, we define their Hausdorff distance as

$$d_H(A,B)\equiv\max\left\{\sup_{a\in A}\inf_{b\in B} d(a,b),\sup_{b\in B}\inf_{a\in A} d(a,b)\right\}.$$

(Rockafellar and Wets, 1998, p. 117). Throughout the paper we take $d(a,b)$ to be the sup-norm in $\mathbb{R}^d$.

The probability of misclassifying a point drawn at random from a measure with $p = 0$ is 0; as the only point that will ever be drawn is λ_2 and such point is always contained in the set. Thus, misclassification error will be less than ϵ, for any ϵ, with probability 1 for any number of draws. The relative Hausdorff distance, however, will be 1 for every sample of points.

2.4.2 Certain choices of P can lead to a small Hausdorff distance

The example above showed that certain choices of P can lead to a large distance between the sets $[\lambda(S)]$ and $[\widehat{\lambda}_M]$ even when misclassification error is low. One could argue that the large Hausdorff distance between the sets is not an intrinsic feature of the example we presented, but rather a consequence of a poor choice of distribution to generate the random sampling approximation. In this subsection we will present a more general theoretical result showing there are some probability measures for which a low misclassification error yields a low relative Hausdorff distance.

We do want to argue, however, that choosing a good probability measure to generate random samples might be infeasible for some sets $\lambda(S)$. For instance suppose that the true set $\lambda(S)$ is given by the shaded area in Figure 3. Suppose that we generate random samples by using a continuous distribution on $\mathbb{R}^2$ and we censor the distribution to obtain draws from the shaded region (by discarding all draws that are not in the shaded area). In this case, our suggested approximation will always miss the thin strip. Since this strip has probability zero under any continuous distribution, it will not be relevant from the perspective of misclassification error, but, unfortunately, it will be quite important in computing the Hausdorff distance between $[\widehat{\lambda}_M]$ and $[\lambda(S)]$.

It is perhaps possible to rule out examples like the one above by restricting $\lambda(S)$ to have topological properties such as being a connected set with non-empty interior. We think that a result showing that the Hausdorff distance between $[\lambda(S)]$ and $[\widehat{\lambda}_M]$ for a large class of probability measures would definitely be of practical interest. We were not able to derive this type of result for Hausdorff distance, but we can at least

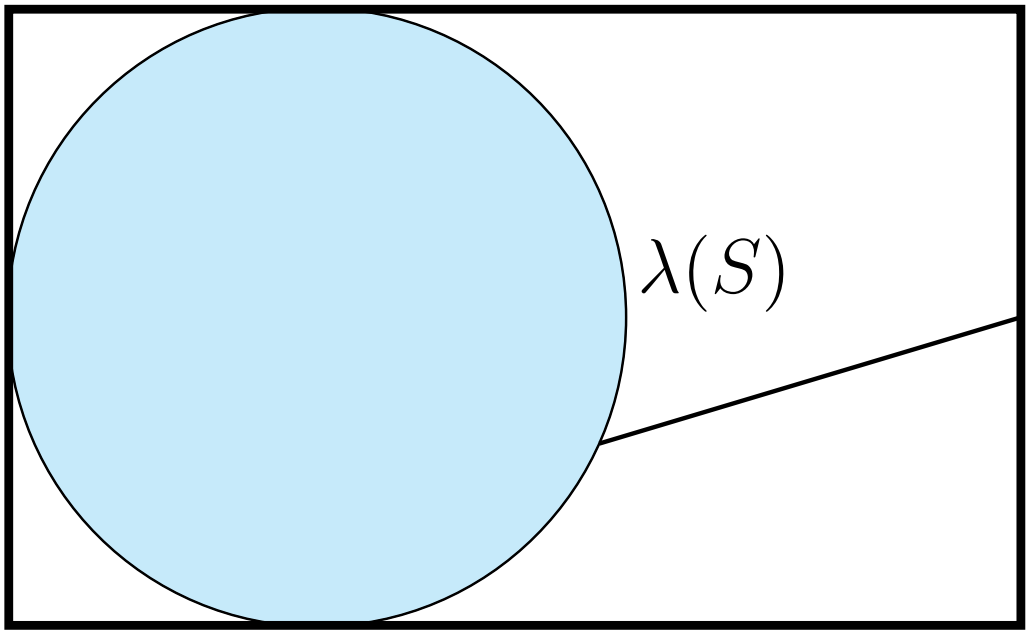

Figure 3: A true set $\lambda(S)$ for which it is not possible to control Hausdorff distance using a continuous probability distribution.

say something constructive for the *relative* Hausdorff distance in Equation (19).

In order to derive this result, we first make an assumption on the shape of the set $\lambda(S)$:

Assumption 1. *The projection of $\lambda(S)$ into its i-th coordinate*

$$p_i(\lambda(S)) \equiv \{x \in \mathbb{R} \mid x = \lambda_i, \lambda \in \lambda(S)\},$$

(where λ_i denotes the i^{th} coordinate of λ) is a bounded interval $[\underline{r}_i, \overline{r}_i]$.

Assumption 1 holds, for example, when $\lambda(S)$ is an (axis-aligned) rectangle; but it is also true for other shapes (for instance, $\lambda(S)$ could be a circle or some convex set).

We now make an assumption on the distribution P (which, as required by Theorem 3 places all of its mass on the set S). Define the tightest band containing $\lambda(S)$ as $[\lambda(S)] \equiv \times_{i=1}^{d} [\underline{r}_i, \overline{r}_i]$.

Assumption 2. *The distribution P on S has the following property: for any set*

$[a_i, b_i] \subseteq p_i(\lambda(S))$ *and any coordinate* i*:*

$$P(\lambda_i(S) \in [a_i, b_i]) = \frac{b_i - a_i}{\overline{r}_i - \underline{r}_i}.$$

That is, we focus on the distributions P that induce a uniform distribution over the projection $p_i(\lambda(S))$.

Assumption 2 holds, for example, when $\lambda(S)$ is an (axis-aligned) rectangle and P is the product of independent uniform marginals.[21] In general, however, there is no guarantee that such a distribution exists.

Proposition 1. *Let Assumption 1 and Assumption 2 hold. Then for any sample of size* M

$$\tilde{d}_H\left([\lambda(S)], [\hat{\lambda}_M]\right) \leqslant \mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P),$$

where $\tilde{d}_H$ *is the relative Hausdorff distance defined in Equation (19).*

Proof. See Appendix A.5. □

Proposition 1 shows that the worst-case relative Hausdorff distance (19) is bounded above by the misclassification error between $[\lambda(S)]$ and $[\hat{\lambda}_M]$. That means that if the misclassification error is smaller than ϵ, then the worst-case Hausdorff distance is also smaller than ϵ. A direct corollary of Theorem 3 is that whenever the number of draws is above our upper bound, then the probability of theworst-case relative Hausdorff distance being below ϵ is at least $1 - \delta$.

3 Applications to SVARs

As an illustrative example, we consider a simple 3-variable monetary SVAR that includes the GDP Deflator (p_t), GDP (gdp_t), and the Federal Funds rate (i_t). The

[21] Assumption 2 may also hold in other cases. For example, Perlman and Wellner (2011) show the existence of such measures on the unit circle of dimension $d \leqslant 3$ (check), and discusses complications with finding distributions that satisfy Assumption 2 and additional symmetry properties in higher dimensions.

variables have quarterly frequency and the sample period is October 1982 to October 2007.[22] The model is given by

$$y_t = \mu + A_1 y_{t-1} + \cdots A_4 y_{t-4} + B\epsilon_t, \tag{5}$$

where ϵ_t are the structural innovations, distributed i.i.d. according to some unknown distribution F, with $\mathbb{E}_F[\epsilon_t] = 0_{3\times 1}$, $\mathbb{E}_F[\epsilon_t \epsilon_t'] = \mathbb{I}_3$ for all $t = 1 \cdots, T$. B is an unknown 3×3 matrix and

$$y_t = (\ln p_t, \ln gdp_t, i_t).$$

The object of interest is the vector of dynamic impulse response coefficients of the natural logarithm of the GDP deflator to a monetary policy shock. The k^{th} period ahead structural impulse response function of variable i to shock 3 (which we assume to be the monetary policy shock) is defined as

$$\lambda_{k,i,3}(A, b) = e_i' C_k(A) b, \tag{6}$$

where e_i denotes the i^{th} column of $/bbI_3$, $A \equiv (A_1, \ldots, A_4)$, and b is the third column of B.[23]

Without further restrictions, time series data on y_t allow the econometrician to consistently estimate A and $\Sigma = BB'$, but not B. There are, in principle, many matrices B such that $BB' = \Sigma$, and thus many structural impulse response functions that can be rationalized by the data. Consequently, it is common in the applied macroeconomics literature to use equality and sign restrictions in an attempt to identify the structural IRF's in (6). If the restrictions allow the econometrician to

[22] The FRED codes are: GDPDEF, GDP, and DFF.

[23] $C_k(A)$ is defined recursively by the formula $C_0 \equiv \mathbb{I}_3$, and

$$C_k(A) \equiv \sum_{m=1}^{k} C_{k-m} A_m,\ k \in \mathbb{N}$$

with $A_m = 0$ if $m > 4$; see (Lütkepohl, 1990, p. 116).

map (A, Σ) into only one matrix B, the SVAR is said to be point-identified. If the map is one-to-many, the SVAR is said to be set-identified.

3.1 Summarizing the identified set in set-identified SVARs

Consider first an SVAR set-identified by means of the sign restrictions on the contemporaneous impulse response coefficients, displayed in Table 1 below.

Series	Contractionary MP Shock
$\ln p_t$	-
$\ln gdp_t$	-
i_t	+

Table 1: Restrictions on contemporaneous responses to a contractionary monetary policy shock. '-' stands for a negative sign restriction and '+' for a positive sign restriction.

Given the least-squares or Maximum Likelihood estimators $(\widehat{A}, \widehat{\Sigma})$ we would like to describe the set of all dynamic responses of $\ln p_t$ to a contractionary monetary policy shock that are consistent with the parameter estimates.

In this example, we are imposing restrictions on only the third column of B, denoted b, and we are only interested in impulse responses with respect to the third shock. Therefore, we define the identified set—the set of vectors b that satisfy the sign restrictions—as:

$$S \equiv \left\{ b \in \mathbb{R}^3 \,\middle|\, b'\widehat{\Sigma}^{-1}b = 1, \text{ and } b \text{ satisfies the sign restrictions in Table 1} \right\}.$$

The parameter region of interest is the impulse responses' identified set for horizons $h = 0, 1, \ldots, 16$, defined as:

$$\lambda(S) \equiv \left\{ (\lambda_0, \lambda_1, \lambda_2, \ldots, \lambda_{16}) \in \mathbb{R}^{17} \,\middle|\, \lambda_k = \lambda_{k,i,3}(\widehat{A}, b),\ b \in S \right\}.$$

Whilst $\lambda(S)$ is typically thought of as a frequentist object, Moon and Schorfheide (2012)(p. 757) recommend reporting the impulse responses' identified set even in Bayesian applications.

We argue that in this example $\lambda(S)$ is injective under mild assumptions. Each impulse response coefficient, $\lambda_{k,i,3}$ is a linear combination of the vector b—the linear combination given by the vector $C_k(\widehat{A})'e_i$. It then suffices to have at least 3 of these vectors that are linearly independent. In our example, the first three periods provide such linearly independent vectors.[24]

Algorithm 2 of RRWZ can be used to sample at random from inside the set S to describe $\lambda(S)$. Let M the desired number of draws from the inside. Set $M' = 0$. 1) Draw a standard normal 3×3 matrix N and let $N = QR$ be the QR decomposition of N with the diagonal on R normalized to be positive. 2) Let $B = \text{chol}(\widehat{\Sigma})Q$, and generate the impulse responses using (6). 3) If the impulse responses do not satisfy the sign restrictions, discard the draw. Otherwise increment M' by 1. Return to step 1. 4) Repeat until $M' = M$.

Setting $\epsilon = \delta = 0.1$ and $d = 17$; evaluating the upper bound in Theorem 3 the number of draws, M, that we would require from inside the identified set is

$$\min\{2d\ln(2d/\delta), \exp(1)(2d + \ln(1/\delta))\}/\epsilon = 987.$$

Thus, in order to ensure a misclassification error of less than 10% with probability 90%, our result suggests to stop the algorithm once we have obtained 987 draws of B that satisfy the sign restrictions.[25]

Figure 4 displays the bands on the identified set for the response of inflation to a contractionary monetary policy shock. For each horizon, we report the minimum

[24] Suppose there are three linearly independent vectors $w = [w_1, w_2, w_3]$. Then, the corresponding impulse response coefficients will be $w'b$ where $w' \in \mathbb{R}^3$. Then, there exists only one vector $\tilde{b}$ that satisfies the equation $w'\tilde{b} = w'b$.

[25] To the best of our knowledge there are no theory-based suggestions on how many draws are required to stop the RRWZ algorithm, which is perhaps the most popular approach to generate draws from inside the identified set in SVARs. Canova and Paustian (2011), p. 351, recommend a fixed number of 15,000 draws from inside the identified set. Kilian and Lütkepohl (2017), p. 432, recommend re-estimating the identified set with different seeds of a Gaussian random number generator, and increasing the number of draws if different seeds lead to qualitatively different results. There are no existing results on the literature that allow us to quantify whether 15,000 total iterations of the RRWZ algorithm are too many or too few draws. The point of our paper is that the theory herein presented allows us to connect these number of draws with a tolerance for misclassification error (ϵ) and for the likelihood such misclassification error being below the tolerance (δ). In addition, we have showed in Proposition 1 that certain choices of P will also control the (worst-case relative) Hausdorff distance.

and the maximum value of the response of $\ln p_t$ over the draws of B that satisfy the sign restrictions (this is exactly the algorithm $[\widehat{\lambda}_M]$ we described in Definition 2).

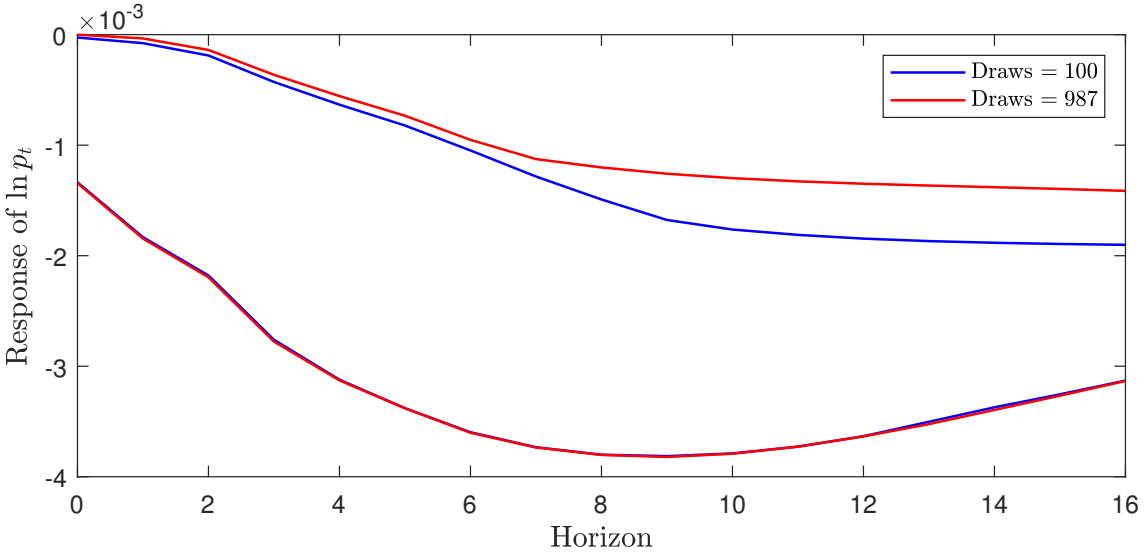

Figure 4: The tightest band that contains the identified set. The parameter region $\lambda(S)$ is defined as the dynamic responses of $\ln p_t$ to a contractionary monetary policy shock on impact and for 16 quarters after impact. The sufficient number of draws from inside the parameter region required to learn, for $\epsilon = \delta = 0.1$ and with $d = 17$, is 987. These bands are plotted in RED. Bands constructed using 100 draws from within $\lambda(S)$ are plotted in BLUE.

In models with tight restrictions, using the RRWZ algorithm to generate draws of B that satisfy the sign restrictions and fall inside the identified set can be challenging. Amir-Ahmadi and Drautzburg (2017) propose an alternative algorithm for partially identified models, in which all draws of B satisfy the sign restrictions, and fall inside S.

As an alternative to random sampling, one can solve for the tightest bands containing the identified set by solving constrained maxima/minima problems at each horizon

$$\min_{\theta \in S} \lambda_h(\theta) \text{ and } \max_{\theta \in S} \lambda_h(\theta).$$

This approach, however, can be difficult to implement. The bounds are defined by nonlinear programs with choice sets that need not be convex.

In our specific SVAR application, we can use the algorithm suggested by Gafarov, Meier and Montiel Olea (2018) to evaluate the bounds of the estimated identified set and compare our random sampling approximation with the 'true' bounds.[26] In principle, the discussion in Section 2.4 suggests that there is no guarantee whatsoever that the Hausdorff distance between these sets will be small, as the probability distribution we have used to generate the random samples does not satisfy Assumption 2 (and there is not an obvious way of how to enforce this property).

Figure 4 plots both the analytical bands and the bands generated by 100 draws (chosen ad-hoc) and our recommendation of 987 draws from inside, which are sufficient to learn when $\epsilon = \delta = 0.1$ and $d = 17$, according to Theorem 3. The figure suggests that the Hausdorff distance between the sets is actually small. The relative Hausdorff distance is 0.0284, well under the misclassification error of 0.1.

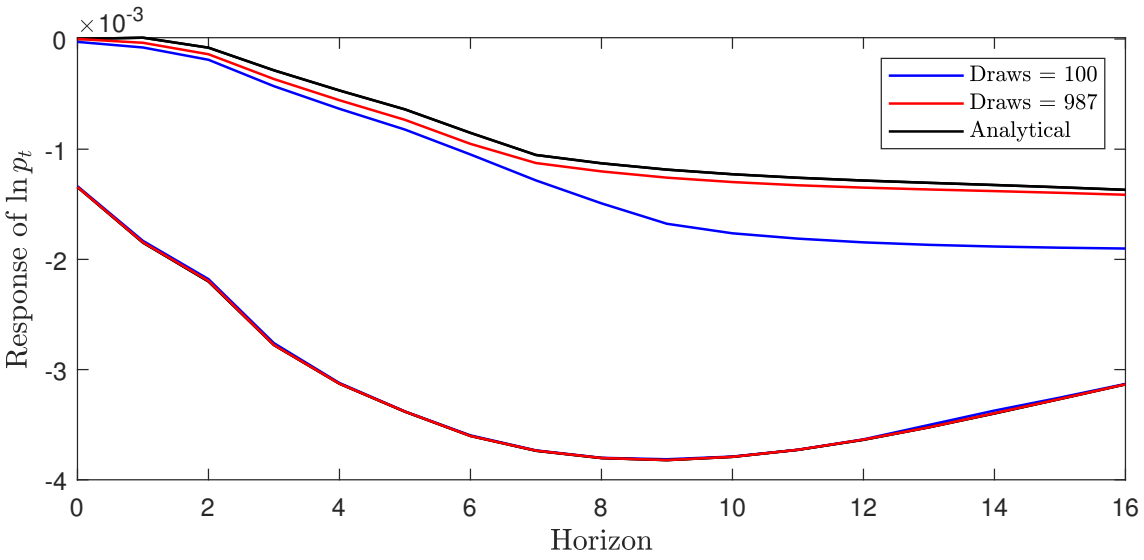

Figure 4: Analytical bands of Gafarov, Meier and Montiel Olea (2018) are plotted in BLACK. Bands generated using $[\widehat{\lambda_M}]$ using 100 draws (BLUE) and 987 (RED) that satisfy the sign restrictions. 987 draws is sufficient to support $\epsilon = \delta = 0.1$, with $d = 17$.

In order to establish a connection between Figure 4 and Proposition 1, Figure 5 reports the empirical c.d.f. of the marginal distributions associated with the measure we used to create the random samples. The support of the distributions has been

[26] We remind the reader, however, that their algorithm is only defined for SVAR models with restrictions on one structural shock.

normalized to the unit interval, and the figure also plots the c.d.f of a uniform random variable for comparison. Proposition 1 shows that whenever these marginals are uniform, then the (*worst-case relative*) Hausdorff distance will be at most equal to the misclassification error. The probability distribution we used to sample from inside do not have uniform marginals, but the difference does not seem to be that large. This might explain why the approximation (in terms of Hausdorff distance) performs reasonably well.

To close this subsection, we discuss another econometric procedure that could have been used to estimate the bounds of the identified set. Giacomini and Kitagawa (2018) develop a robust Bayes algorithm to compute bounds on the identified set that uses a maximization problem. Their algorithm is more general than that in Gafarov, Meier and Montiel Olea (2018), as it covers problems with restrictions on multiple shocks. However the maximization problem they define is non-convex and may be difficult to implement numerically (as the authors acknowledge). In fact, their paper suggests an alternative implementation (Algorithm 2 p. 30) using K random draws. This is done to alleviate concerns about the convergence properties of the numerical optimization in their baseline robust Bayes algorithm.

Our results can help practitioners choosing the number of draws to evaluate the bounds of the identified set. In particular, our theory provides a specific recommendation of K for each specific choice of (ϵ, δ). Conversely, for each choice of K (for example, $K = 1000$), an iso-draw curve provides a combination of (ϵ, δ) that are compatible with the number of draws specified by the user.

3.2 Summarizing a Wald Ellipse in point-identified SVARs

Consider now an SVAR where the dynamic responses to a monetary shock are point identified using two exclusion restrictions: namely, neither output nor prices are affected by a monetary policy shock upon impact.[27] Under such an identification scheme, the vector of 17 impulse responses, denoted γ, can be estimated consistently.

[27] This recursive identification scheme is implemented by setting $B = \text{chol}(\hat{\Sigma})$.

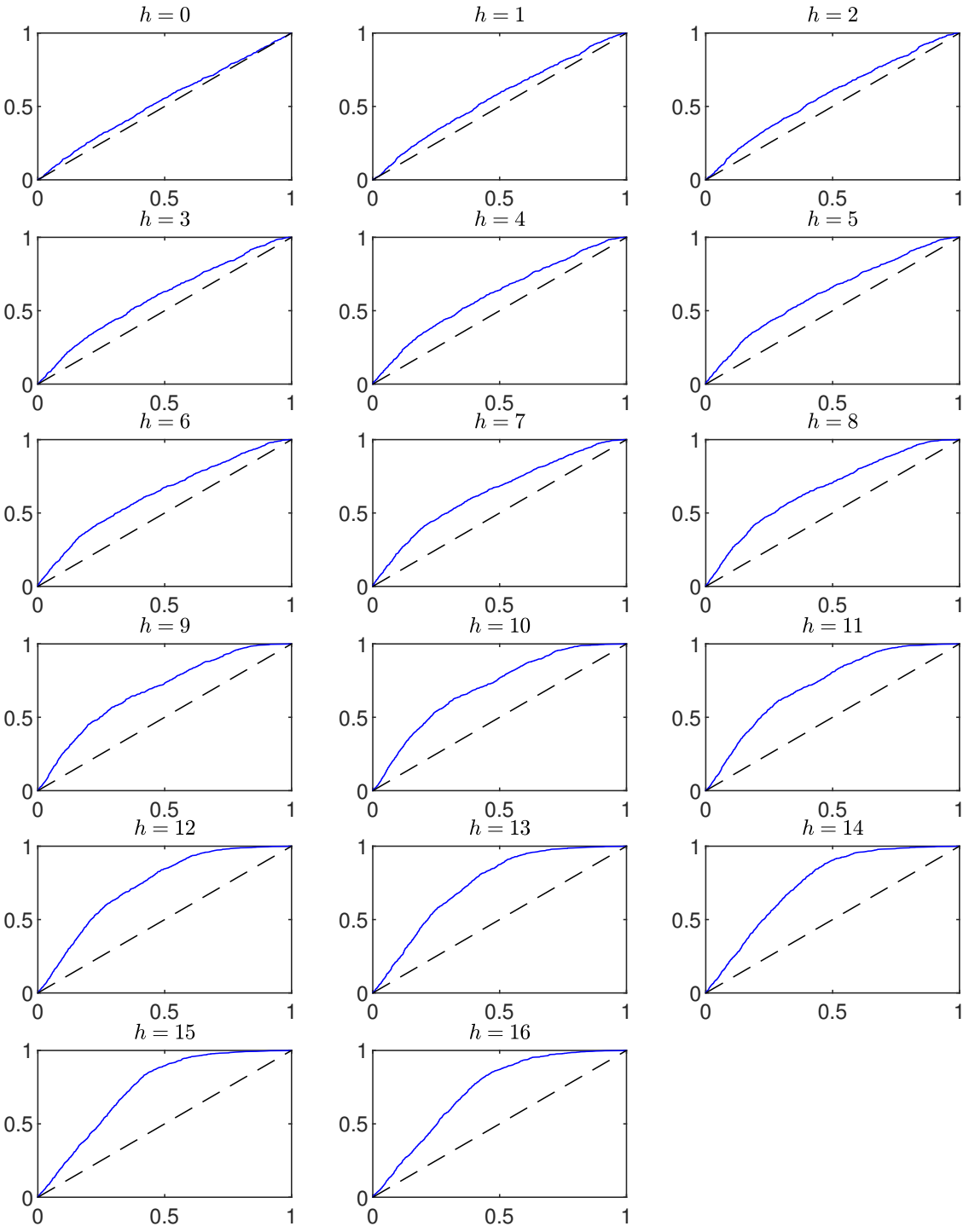

Figure 5: Empirical c.d.f. (SOLID, BLUE) of the marginal distributions at $h = 0, \ldots, 16$. The support has been normalized to the unit interval. The c.d.f. of the uniform distribution is plotted in DOTTED, BLACK.

The goal here is to summarize a Wald ellipse reporting shotgun plots as in Inoue and Kilian (2016) (IK henceforth).

Define the Wald statistic

$$W(\gamma) = (\gamma - \hat{\gamma}_T)' \widehat{\Sigma}^{-1} (\gamma - \hat{\gamma}_T), \tag{7}$$

where $\hat{\gamma}_T$ is the least squares estimator of γ and $\widehat{\Sigma}$ is the estimator for Σ suggested by IK based on bootstrap draws of $\hat{\gamma}_T$. Consider the Wald ellipse

$$S \equiv \{\gamma \in \mathbb{R}^{17} \mid W(\gamma) \leqslant c_\alpha\},$$

where the critical value c_α is computed using the procedure outline in p. 425 of IK. Note that in this example, λ is the identity (which is injective) and $[S]$ is simply the projection of the Wald ellipse onto each of its coordinates.

The algorithm to report shotgun plots suggested by IK can be thought of as a particular implementation of the algorithm in Definition 2: a value of γ is drawn at random (using the residual bootstrap) and plotted only if it belongs to S. We can suggest a number of γ-draws by pretending that the goal of the shotgun plots is to learn the parameter region $[S]$.

Figure 6 displays shotgun plots for the response of inflation to a contractionary monetary policy shock, where 100, and $2,000$ total draws are used, which for a 68% confidence interval corresponds to 68 and $1,360$ draws from inside S respectively. IK rely on $2,000$ total γ draws, corresponding to $1,360$ draws from inside S. Instead of choosing specific values of (ϵ, δ), Figure 7 displays the iso-draw curve for $M = 1$, 360, all possible combinations of accuracy parameters that could be supported using $1,360$ γ draws from inside the parameter region.

In situations where it may be difficult to target a certain number of draws, one can report an iso-draw curve to demonstrate the accuracy of the approximation.

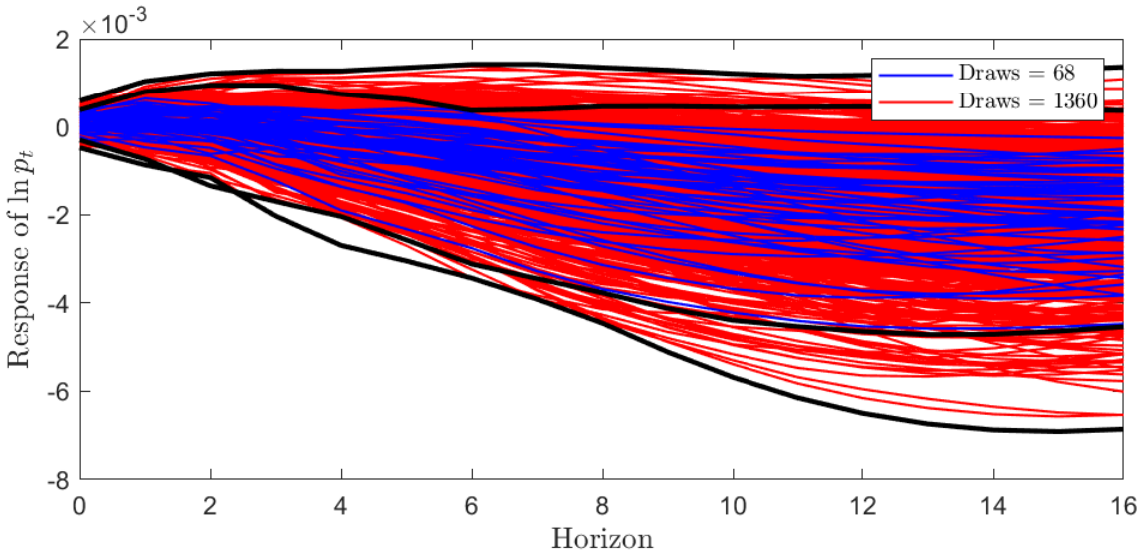

Figure 6: Shotgun plot of the 68% joint confidence region of the dynamic response of $\ln p_t$ to a monetary policy shock. BLUE and RED lines represent 68 and 1,360 draws from inside S, respectively. BLACK lines represent the minimum and maximum (pointwise) at each horizon.

3.3 Highest posterior density credible set in SVARs

Consider again the point-identified model described in the previous subsection. Suppose now that we are interested in constructing the highest posterior density (HPD) credible set for the dynamic structural impulse responses of $\ln p_t$ to a monetary shock. Denote $p(\gamma \,|\, y^T)$ as the posterior density of the dynamic structural impulse responses given data y^T. The $100(1-\alpha)\%$ HPD credible set is

$$S = \{\gamma \in \mathbb{R}^{17} \,|\, p(\gamma|y^T) \geqslant c_\alpha\},$$

where c_α is defined as the largest constant such that $p(S|y^T) \geqslant 1-\alpha$. In this example $\lambda(\cdot)$ is again the identity which is injective. We construct the HPD credible set as in Inoue and Kilian (2013, 2019). We assume a diffuse Gaussian-inverse Wishart prior for the reduced-form VAR parameters θ, which leads a conjugate posterior which can be easily drawn from. We take N draws of reduced form parameters, and compute the impulse responses and their posterior density. The $100(1-\alpha)$ highest posterior density credible set is then the $M = 100(1-\alpha)N$ impulse responses with

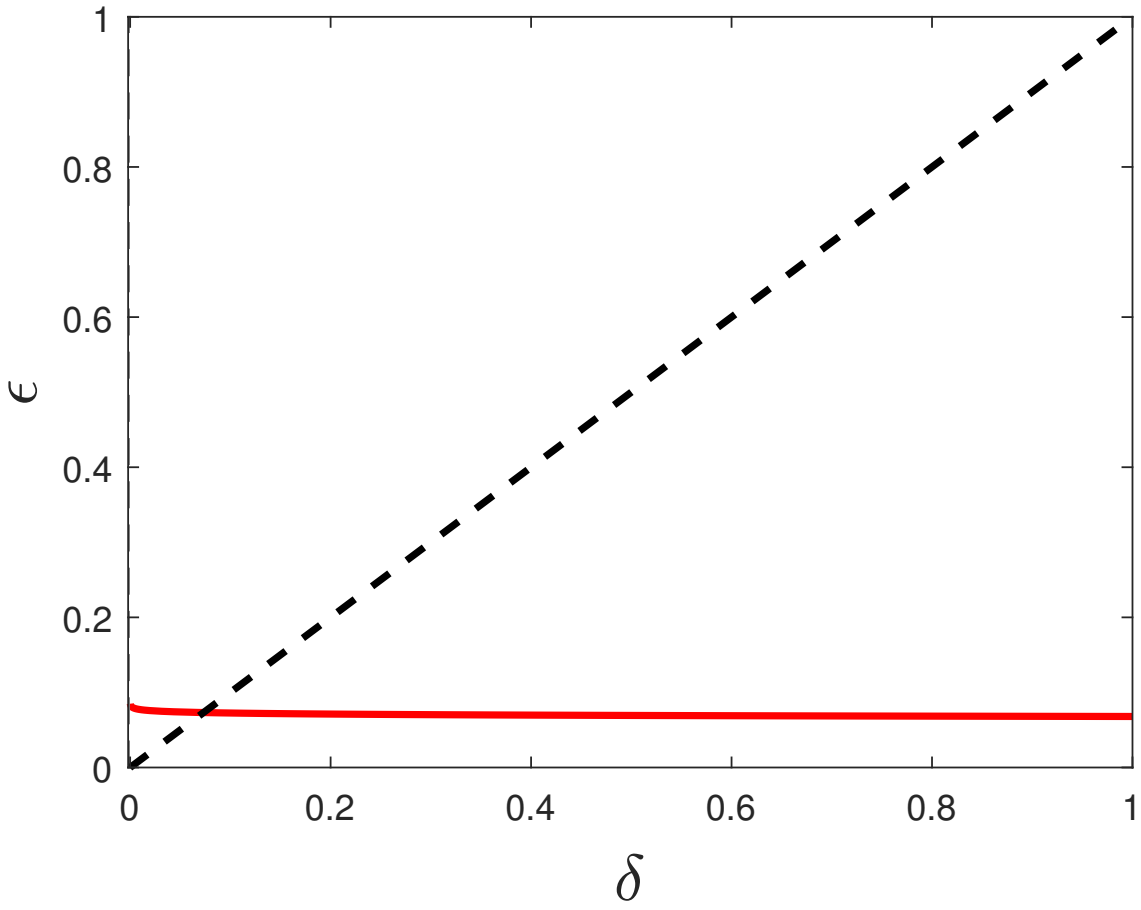

Figure 7: 'Iso-draw' curve for $M = 1,360$ and $d = 17$.

the highest posterior density.

Figure 8 displays the HPD credible set for $\alpha = 0.32$, and N equal to 100 and $2,000$, corresponding to M equal to 68 and $1,360$ respectively. With $1,360$ draws from inside $\lambda(S)$, Figure 7 from the previous subsection corresponds to the iso-draw curve for this application.

Remark: The procedure described above for approximating parameter regions that arise from highest-posterior density sets can be readily extended to parameter regions defined by *level sets of criterion functions*; such as likelihoods, quasi-likelihoods, or profiled versions of them.

The work of Chen, Christensen and Tamer (2018) (henceforth CCT)—who provide computationally attractive procedures to construct confidence sets for identified sets in a general class of models—is a notable example where these types of parameter regions arise. Let θ be the full set of structural parameters of a model and let

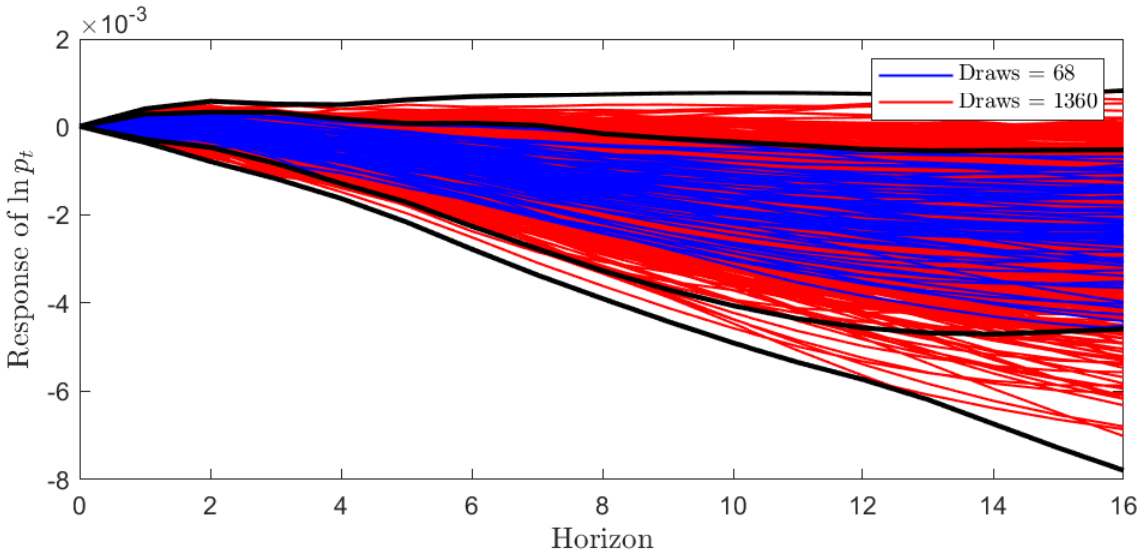

Figure 8: 68% highest posterior density credible set for the dynamic response of $\ln p_t$ to a monetary policy shock. BLUE and RED lines represent 68 and 1,360 draws from inside S, respectively. BLACK lines represent the minimum and maximum (pointwise) at each horizon.

Θ be the parameter space. To allow for subvector inference write $\theta = (\mu, \eta)$ and suppose that the object of interest is constructing a confidence set for the identified set of the parameter μ. The procedure recommended by CCT takes the form

$$CS_{1-\alpha} = \{\, \mu | (\mu, \eta) \in \Theta \text{ and } g(\mu) \geqslant \zeta_{\alpha}^{mc} \,\}$$

(see Procedure 2 in p. 1973 of CCT and Equation (13)), where $g(\mu)$ is a profiled criterion at a point μ (for example, the value of the criterion function at μ after η has been profiled out) and ζ_{α}^{mc} is the $1-\alpha$ quantile of a profiled criterion function based on posterior draws of θ.

Now, suppose that $b = 1, \ldots, B$ posterior draws of θ had been used to compute ζ_{α}^{mc}. By definition, $(1-\alpha)B$ of these draws will be inside $CS_{1-\alpha}$, the parameter region of interest. If μ has dimension d, the smallest band containing $CS_{1-\alpha}$ is simply

$$[CS_{1-\alpha}] = \bigtimes_{i=1}^{d} \left[\min_{\mu_b | g(\mu_b) \geqslant \zeta_{\alpha}^{mc}} \mu_{i,b} \,,\; \max_{\mu_b | g(\mu_b) \geqslant \zeta_{\alpha}^{mc}} \mu_{i,b} \right],$$

where $\mu_{i,b}$ denotes the i^{th} coordinate of the b^{th} draw of μ. Our Theorem 3 immediately gives either i) a combination of (ϵ, δ) that can be supported by the resulting $(1-\alpha)B$ draws or ii) a recommendation of how many more posterior draws are necessary to attain a particular combination of (ϵ, δ) to control misclassification error. There is still the question of how many draws, B, should be used to compute ζ_{α}^{mc}.[28]

4 Conclusion

We showed that sampling at random from a *parameter region* in order to describe it, can be framed as a supervised (machine) learning problem. We used concepts from the supervised learning literature—misclassification error, sample complexity, and the definition of learning itself—to provide some practical guidance on two issues. First, how to think about the accuracy of a random sampling approximation to a parameter region. Second, how many random draws are necessary/sufficient to learn it.

We started by formalizing an obvious observation: parameter regions can be learned if and only if they are not too complex. This result binds often, as some assumptions that are typically imposed to simplify the analysis of econometric problems, do not simplify the supervised learning problem.

We circumvent the impossibility result by introducing two modifications to the standard definition of learning.

First–in order to avoid making assumptions about the shape of the parameter region of interest—we focus on learning the *tightest band* that contains it. This is done by computing misclassification error relative to such tightest band that contains the parameter region, instead of the true set. Bands are convenient, for they are already used to summarize uncertainty in the econometric models used as our main illustrative example.

[28] We think, however, that the results in Belloni, Chernozhukov et al. (2009) could potentially be used to address this point. In particular, their Theorem 3, p. 2031 has a recommendation on the length of the of the burn-in sample and post burn-in samples as a function of some functions of the target distribution (i.e., *global conductance*) and the starting distribution for the chain (i.e., the variance of this distribution).

Second, we restrict the class of probability distributions that both the econometrician and the oracle can consider. In particular, we restrict the econometrician to sample from *inside* the parameter region of interest.

Under these two modifications—which simplify the learning desiderata—we show that the tightest band containing the parameter region of interest can be *learned from the inside*. Our learning algorithm keeps track of the largest and smallest values of the parameter of interest in each of its dimensions. We show that learning from the inside requires at least $\max\{(1-\epsilon)\ln(1/\delta), (3/16)d\}/\epsilon$ draws, but at most $\min\{2d\ln(2d/\delta), \exp(1)(2d+\ln(1/\delta))\}/\epsilon$ draws. In both cases, the random draws have to come from inside the parameter region. We also note that d is the dimension of $\lambda(\Theta)$ not of Θ (which in our examples has a higher dimension).

We used SVARs to showcase the application of our bounds. We considered the problem of describing the identified set in a set-identified SVAR and also the problem of reporting shotgun plots for both frequentist and Bayesian simultaneous inference on impulse responses. We used the bounds directly and indirectly. Directly, to provide a concrete recommendation of the number of draws required for a given ϵ and δ. Indirectly, by constructing *iso-draw curves*; given a number of draws M, the iso-draw curve collects all combinations of (ϵ, δ) that yield M as recommendation. Our recommendation for practitioners is to report the iso-draw curve associated to however many draws are feasible in their specific application.

References

Adams, Terrence M., and Andrew B. Nobel. 2012. "Uniform Approximation of Vapnik–Chervonenkis Classes." *Bernoulli*, 18(4): 1310–1319.

Akama, Yohji, and Kei Irie. 2011. "VC Dimension of Ellipsoids." *arXiv preprint arXiv:1109.4347*.

Al-Najjar, Nabil I. 2009. "Decision Makers as Statisticians: Diversity, Ambiguity, and Learning." *Econometrica*, 77(5): 1371–1401.

Al-Najjar, Nabil I, and Mallesh M Pai. 2014. "Coarse Decision Making and Overfitting." *Journal of Economic Theory*, 150: 467–486.

Amir-Ahmadi, Pooyan, and Thorsten Drautzburg. 2017. "Identification through Heterogeneity." *Unpublished Manuscript*.

Auer, Peter, Philip M Long, and Aravind Srinivasan. 1998. "Approximating hyper-rectangles: learning and pseudorandom sets." *Journal of Computer and System Sciences*, 57(3): 376–388.

Bar, H, and F Molinari. 2013. "Computational Methods for Partially Identified Models via Data Augmentation and Support Vector Machines." Cornell University Working Paper.

Belloni, Alexandre, Victor Chernozhukov, and Christian Hansen. 2014. "Inference on Treatment Effects After Selection Among High-Dimensional Controls." *The Review of Economic Studies*, 81(2): 608–650.

Belloni, Alexandre, Victor Chernozhukov, Iván Fernández-Val, and Chris Hansen. 2017. "Program Evaluation with High-Dimensional Data." *Econometrica*, 85(1): 233–298.

Belloni, Alexandre, Victor Chernozhukov, et al. 2009. "On the Computational Complexity of MCMC-Based Estimators in Large Samples." *The Annals of Statistics*, 37(4): 2011–2055.

Blei, David M, Andrew Y Ng, and Michael I Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning Research*, 3: 993–1022.

Blumer, Anselm, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. 1989. "Learnability and the Vapnik-Chervonenkis Dimension." *Journal of the ACM (JACM)*, 36(4): 929–965.

Bonhomme, Stéphane, Thibaut Lamadon, and Elena Manresa. 2017. "Discretizing Unobserved Heterogeneity." *Unpublished Manuscript.*

Canova, Fabio, and Matthias Paustian. 2011. "Business Cycle Measurement with Some Theory." *Journal of Monetary Economics*, 58(4): 345–361.

Chen, Xiaohong, Timothy M Christensen, and Elie Tamer. 2018. "Monte Carlo Confidence Sets for Identified Sets." *Econometrica*, 86(6): 1965–2018.

Dudley, Richard M. 1999. *Uniform Central Limit Theorems.* Vol. 23, Cambridge Univ Press.

Ehrenfeucht, Andrzej, David Haussler, Michael Kearns, and Leslie Valiant. 1989. "A general lower bound on the number of examples needed for learning." *Information and Computation*, 82(3): 247–261.

Faust, Jon. 1998. "The Robustness of Identified VAR Conclusions about Money." *Carnegie-Rochester Conference Series on Public Policy*, 49: 207–244.

Freyberger, Joachim, and Yoshiyasu Rai. 2018. "Uniform Confidence Bands: Characterization and Optimality." *Journal of Econometrics*, 204(1): 119–130.

Gafarov, Bulat, Matthias Meier, and José Luis Montiel Olea. 2018. "Delta-method Inference for a Class of Set-Identified SVARs." *Journal of Econometrics*, 203(2): 316–327.

Giacomini, Raffaella, and Toru Kitagawa. 2018. "Robust Bayesian Inference for Set-Identified Models." *Cemmap Working Paper 61/18.*

Hansen, Stephen, and Michael McMahon. 2016. "Shocking Language: Understanding the Macroeconomic Effects of Central Bank Communication." *Journal of International Economics*, 99: S114–S133.

Haussler, D., N. Littlestone, and M.K. Warmuth. 1994. "Predicting 0, 1-Functions on Randomly Drawn Points." *Information and Computation*, 115(2): 248–292.

Horowitz, Joel L, and Charles F Manski. 2006. "Identification and Estimation of Statistical Functionals Using Incomplete Data." *Journal of Econometrics*, 132(2): 445–459.

Horowitz, Joel L., and Sokbae Lee. 2012. "Uniform Confidence Bands for Functions Estimated Nonparametrically with Instrumental Variables." *Journal of Econometrics*, 168(2): 175–188.

Horowitz, Joel L, Charles F Manski, Maria Ponomareva, and Jörg Stoye. 2003. "Computation of Bounds on Population Parameters when the Data are Incomplete." *Reliable Computing*, 9(6): 419–440.

Inoue, Atsushi, and Lutz Kilian. 2013. "Inference on Impulse Response Functions in Structural VAR Models." *Journal of Econometrics*, 177(1): 1–13.

Inoue, Atsushi, and Lutz Kilian. 2016. "Joint Confidence Sets for Structural Impulse Responses." *Journal of Econometrics*, 192(2): 421–432.

Inoue, Atsushi, and Lutz Kilian. 2019. "Corrigendum to "Inference on Impulse Response Functions in Structural VAR Models" [J. Econometrics 177 (2013) 1–13]." *Journal of Econometrics*, 209(1): 139–143.

Ke, Shikun, José Luis Montiel Olea, and James Nesbit. 2019. "A Robust Machine Learning Algorithm for Text Analysis." *Working Paper*.

Kilian, Lutz, and Helmut Lütkepohl. 2017. *Structural Vector Autoregressive Analysis*. Cambridge University Press.

Lütkepohl, Helmut. 1990. "Asymptotic Distributions of Impulse Response Functions and Forecast Error Variance Decompositions of Vector Autoregressive Models." *The Review of Economics and Statistics*, 72(1): 116–125.

Mohri, Mehryar, Afshin Rostamizadeh, and Ameet Talwalkar. 2012. *Foundations of Machine Learning.* MIT Press.

Montiel Olea, José Luis, and Mikkel Plagborg-Møller. 2019. "Simultaneous Confidence Bands: Theory, Implementation, and an Application to SVARs." *Journal of Applied Econometrics*, 34(1): 1–17.

Moon, Hyungsik Roger, and Frank Schorfheide. 2012. "Bayesian and Frequentist Inference in Partially Identified Models." *Econometrica*, 80(2): 755–782.

Murphy, Kevin P. 2012. *Machine Learning: A Probabilistic Perspective.* The MIT Press.

Perlman, Michael D, and Jon A Wellner. 2011. "Squaring the Circle and Cubing the Sphere: Circular and Spherical Copulas." *Symmetry*, 3(3): 574–599.

Rockafellar, R Tyrrell, and Roger J-B Wets. 1998. *Variational Analysis.* Vol. 317, Springer Science & Business Media.

Rubio-Ramirez, Juan F, Daniel F Waggoner, and Tao Zha. 2010. "Structural Vector Autoregressions: Theory of Identification and Algorithms for Inference." *The Review of Economic Studies*, 77(2): 665–696.

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

Uhlig, Harald. 2005. "What are the Effects of Monetary Policy on Output? Results from an Agnostic Identification Procedure." *Journal of Monetary Economics*, 52(2): 381–419.

Valiant, Leslie G. 1984. "A Theory of the Learnable." *Communications of the ACM*, 27(11): 1134–1142.

Vapnik, Vladimir Naumovich. 1998. *Statistical Learning Theory.* Vol. 1, New York:Wiley.

Wager, Stefan, and Susan Athey. 2018. "Estimation and Inference of Heterogeneous Treatment Effects using Random Forests." *Journal of the American Statistical Association*, 113(523): 1228–1242.

A Appendix A

A.1 VC dimension

Given a nonempty class $\Lambda \subseteq 2^{\mathbb{R}^d}$ and a finite set of points $\lambda(S) \subseteq \mathbb{R}^d$, let $\Pi_\Lambda(\lambda(S))$ denote the set of all subsets of $\lambda(S)$ that can be obtained by intersecting $\lambda(S)$ with a concept $\boldsymbol{\lambda} \in \Lambda$, that is:

$$\Pi_\Lambda(\lambda(S)) = \{\lambda(S) \cap \boldsymbol{\lambda} \mid \ \boldsymbol{\lambda} \in \Lambda\}.$$

If $\Pi_\Lambda(\lambda(S)) = 2^{\lambda(S)}$, then we say that $\lambda(S)$ is shattered by Λ.

Definition 3 (Vapnik–Chervonenkis dimension)**.** *The* Vapnik–Chervonenkis (VC) dimension *of a concept class* Λ*, denoted* $VCdim(\Lambda)$*, is the cardinality of the largest finite set of points* $\lambda(S)$ *that can be shattered by* Λ*.*

If arbitrarily large finite sets are shattered, the VC dimension of Λ is infinite. Our presentation of shattering and VC dimension follow Blumer et al. (1989) p. 934. An alternative reference is Dudley (1999), p. 134.[29]

A.2 Proof of Theorem 1

Proof. First we will show that if Λ is *trivial*— in the sense of either containing only one concept or two disjoint concepts that partition $\lambda(\Theta)$—we always have learning in the sense of Definition 1.

Suppose that Λ contains only one concept. An algorithm that reports only this concept will always have a misclassification error of zero and thus satisfies Definition 1, for any $M \geqslant 0$.

Suppose Λ contains only two disjoint concepts $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$, such that $\boldsymbol{\lambda}_1 \cup \boldsymbol{\lambda}_2 = \lambda(\Theta)$. Suppose we observe a sample that contains a single observation x and a label

[29] A class with finite VC dimension has finite bracketing numbers, and satisfies uniform laws of large numbers for every ergodic process (Adams and Nobel (2012)).

$l(x)$. The algorithm

$$\widehat{\lambda} = \begin{cases} \boldsymbol{\lambda}_1 & \text{if } (x \in \boldsymbol{\lambda}_1 \text{ and } l(x) = 1) \text{ or } (x \in \boldsymbol{\lambda}_2 \text{ and } l(x) = 0); \\ \boldsymbol{\lambda}_2 & \text{if } (x \in \boldsymbol{\lambda}_2 \text{ and } l(x) = 1) \text{ or } (x \in \boldsymbol{\lambda}_1 \text{ and } l(x) = 0). \end{cases}$$

will achieve zero misclassification error. Hence Definition 1 is satisfied for any $M \geqslant 1$, using an algorithm that throws away all the data points but the first one.

So now we will focus on non-trivial concept classes. We show first that if Λ has finite VC dimension, then $\lambda(S) \in \Lambda$ is learnable in the sense of Definition 1.

To see this, let $\mathcal{P}_\Theta$ denote the set of all probability distributions over $\Theta \subseteq \mathbb{R}^p$ and let $\mathcal{P}(\mathbb{R}^d)$ denote the set of all probability distributions over $\mathbb{R}^d$ (the space in which λ takes its values). Note that each $P \in \mathcal{P}_\Theta$ induces a probability distribution $\tilde{P}$ over $\mathbb{R}^d$ in the obvious way: for any measurable $A \in 2^{\mathbb{R}^d}$, $\tilde{P}(A) \equiv P(\lambda^{-1}(A))$. Let $\mathcal{P}_\lambda(\mathcal{P}_\Theta)$ denote the set of all probability measures induced by the elements of $\mathcal{P}_\Theta$ through the mapping λ. Evidently $\mathcal{P}_\lambda(\mathcal{P}_\Theta) \subseteq \mathcal{P}(\mathbb{R}^d)$.

The Fundamental Theorem of Statistical Learning in Blumer et al. (1989) Theorem 2.1 part i) implies that if $\Lambda \subseteq 2^{\lambda(\Theta)} \subseteq 2^{\mathbb{R}^d}$ has finite VC dimension, then there exists an algorithm $\widehat{\lambda}_M$ such that for any $0 < \epsilon, \delta < 1$ and any $\boldsymbol{\lambda} \in \Lambda$:

$$\sup_{P \in \mathcal{P}(\mathbb{R}^d)} P\left(\mathcal{L}(\widehat{\lambda}_M; \boldsymbol{\lambda}, P) \geqslant \epsilon\right) \leqslant \delta,$$

provided $M \geqslant m(\epsilon, \delta)$. Since $\mathcal{P}_\lambda(\mathcal{P}_\Theta) \subseteq \mathcal{P}(\mathbb{R}^d)$ and $\lambda(\cdot)$ is injective, it then follows that:

$$\sup_{P \in \mathcal{P}_\lambda(\mathcal{P}_\Theta)} P\left(\mathcal{L}(\widehat{\lambda}_M; \boldsymbol{\lambda}, P) \geqslant \epsilon\right) \leqslant \sup_{P \in \mathcal{P}(\mathbb{R}^d)} P\left(\mathcal{L}(\widehat{\lambda}_M; \boldsymbol{\lambda}, P) \geqslant \epsilon\right) \leqslant \delta,$$

provided $M \geqslant m(\epsilon, \delta)$. Thus, $\lambda(S) \in \Lambda$ is learnable in the sense of Definition 1.

Now we show that $\lambda(S) \in \Lambda$ is learnable only if Λ has VC finite dimension. Suppose to the contrary that $\Lambda \subseteq 2^{\lambda(\Theta)}$ has infinite VC dimension. Then for any $d^* \in \mathbb{N}$ there exist d^* distinct points $\{x_1, x_2, \ldots, x_{d^*}\}$ that are shattered by Λ. Since $\Lambda \subseteq 2^{\lambda(\Theta)}$, this implies the existence of at least d^* points $\theta_1, \theta_2, \ldots, \theta_{d^*} \in \Theta$ such that

$\lambda(\theta) = x_m$. Since $\mathcal{P}_\Theta$ contains all possible distributions on Θ, it contains the uniform distribution over $\{\theta_1, \theta_2, \ldots, \theta_{d^*}\}$ which induces a uniform distribution over $\{x_1, x_2, \ldots, x_{d^*}\}$. The proof of part (ii)(b) Case 2 of Theorem 2.1 in (Blumer et al., 1989, pp. 936-937) then implies that any learning algorithm should use at least $\mathcal{O}(d^*)$ draws. We supposed that Λ has infinite VC dimension, so this must hold for any $d^* \in \mathbb{N}$. Therefore, learning is not possible. if Λ has an infinite VC dimension. $\square$

A.3 Proof of Theorem 2

Proof. Suppose that there is an algorithm $\widehat{\lambda}_M$ that satisfies *i*) and *ii*). Take any concept $\boldsymbol{\lambda} \in \Lambda$ that is not a band. This means that $\boldsymbol{\lambda}$ is such that $A \equiv [\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda} \neq \varnothing$. Suppose that we observe an i.i.d. sample of size M, $\boldsymbol{\theta}_M = (\theta_1, \theta_2, \ldots, \theta_M)$ such that $\lambda(\theta_m) \in A$ for all $m = 1, \ldots, M$.

For any such sample, an algorithm that satisfies *i*) outputs the empty set (this happens because for every m, we must have $\lambda(\theta_m) \notin \boldsymbol{\lambda}$, and consequently θ_i cannot be in S). Thus, a sample with $\lambda(\theta_m) \in A$ for all m has only 0-labels and any algorithm satisfying *i*) will, at best, misclassify all $\lambda(\theta_m) \in A$. So for any probability distribution P:

$$\lambda(\theta_m) \in A, \ \ \forall m = 1, \ldots, M \implies \mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P) \geqslant P(\lambda(\theta) \in A). \tag{8}$$

By assumption, for every $0 < \eta < 1$ there exists a probability distribution P_η such that $P_\eta(\lambda(\theta) \in A) \geqslant \eta$. This means that for every η we have that

$$\begin{aligned} P_\eta\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_\eta) \geqslant \eta\right) &\geqslant P_\eta\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_\eta) \geqslant P_\eta(\lambda(\theta) \in A)\right) \\ &\quad \text{(as } P_\eta(\lambda(\theta) \in A) \geqslant \eta\text{)}, \\ &\geqslant P_\eta(\lambda(\theta_m) \in A \quad \forall m = 1, \ldots, M) \\ &\quad \text{(by (8))}, \\ &\geqslant \eta^M. \end{aligned}$$

If $ii)$ is satisfied, then there must exist a function $m(\epsilon, \delta)$—that depends on the algorithm $\widehat{\lambda}_M$—such that for any $M \geqslant m(\epsilon, \delta)$ we have that for any η, $P_\eta(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_\eta) \geqslant \epsilon) \leqslant \delta$. However, note that for any $\eta' \geqslant \epsilon$ it follows that

$$P_{\eta'}\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_{\eta'}) \geqslant \eta'\right) \leqslant P_{\eta'}\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_{\eta'}) \geqslant \epsilon\right) \leqslant \delta.$$

But then this implies that for any $\eta' \geqslant \epsilon$, we have a fortiori that $(\eta')^M \leqslant \delta$. Rearranging for M yields, $M \geqslant \ln(\delta)/\ln(\eta')$ for any $M \geqslant m(\epsilon, \delta)$. In particular, if we let $\bar{m}(\epsilon, \delta)$ denote the smallest integer larger than or equal to $m(\epsilon, \delta)$ we have that $\bar{m}(\epsilon, \delta) \geqslant \ln(\delta)/\ln(\eta')$ for all $\eta' \in (\epsilon, 1)$. This implies that $m(\epsilon, \delta)$ has to be infinity for every ϵ, δ pair as η' can be arbitrarily close to 1. This contradicts $ii)$. □

A.4 Proof of Theorem 3

Proof of upper bound

First term: $2d/\epsilon \ln(2d/\delta)$

To prove the first term of the upper bound in Theorem 3 we first need a lemma. Define a *d-dimensional hyperrectangle* as the Cartesian product of d intervals in the real line; that is:

$$r \equiv \bigtimes_{j=1}^{d} [\underline{r}_j, \widehat{\overline{r}}], \tag{9}$$

where $\underline{r}_j < \overline{r_j}$ for $j = 1, \cdots, d$. For any d-dimensional rectangle r and any $A \subseteq \mathbb{R}$ we will also define $r_{-j}(A)$ as the subset of $\mathbb{R}^d$ generated by replacing the j^{th} interval $[\underline{r}_j, \overline{r}_j]$ in the hyperrectangle R by the set A. That is:

$$r_{-j}(A) = [\underline{r}_1, \overline{r}_1] \times \ldots [\underline{r}_{j-1}, \overline{r}_{j-1}] \times A \times [\underline{r}_{j+1}, \overline{r}_{j+1}] \ldots [\underline{r}_d, \overline{r}_d].$$

Lemma 1. *For any $\epsilon \in (0, 1)$, any probability measure P on $\mathbb{R}^d$, and any d-*

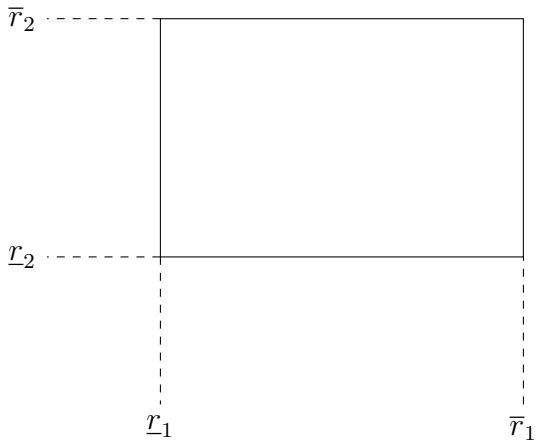

Figure 9: Hyperrectangle r when $d = 2$.

dimensional hyperrectangle r in the form of (9) such that $P(r) > \epsilon$, let

$$\overline{h}_j \equiv \inf\{\, h' \in [\underline{r}_j, \overline{r}_j] \mid P(r_{-j}([\underline{r}_j, h'])) \geqslant \epsilon\}. \tag{10}$$

Then $P(\mathring{r}_{\overline{h}_j}) \leqslant \epsilon$, where $\mathring{r}_{\overline{h}_j} \equiv r_{-j}([\underline{r}_j, \overline{h}_j))$.

Proof. Fix any $k \in [\underline{r}_j, \overline{r}_j]$. Let $r_k \equiv r_{-j}([\underline{r}_j, k])$ and $\mathring{r}_k \equiv r_{-j}([\underline{r}_j, k))$. Note that $\overline{h}_j$ in (10) is well defined as the set

$$\{\, h' \in [\underline{r}_j, \overline{r}_j] \mid P(r_{-j}([\underline{r}_j, h'])) \geqslant \epsilon\}$$

is nonempty by the assumption $P(r) > \epsilon$. Note also that

1. $\mathring{r}_k \subset r_k$ (by definition of r_k and $\mathring{r}_k$).
2. If $k < \overline{h}_j$, then $P(r_k) < \epsilon$ (by definition of $\overline{h}_j$).
3. If $k_n \uparrow \overline{h}_j$, then $\bigcup_{n=1}^{\infty} \mathring{r}_{k_n} = \mathring{r}_h$.

The definition of $\overline{h}_j$ implies that for every strictly increasing sequence $k_n \uparrow \overline{h}_j$, we have

$$P(\mathring{r}_{k_n}) \overset{\text{by } 1}{\leqslant} P(r_{k_n}) \overset{\text{by } 2}{\leqslant} \epsilon.$$

By 3 in the list above and continuity from below of probability measures, it follows that $P(\mathring{r}_{\overline{h}_j}) = \lim_{n\to\infty} P(\mathring{r}_{k_n}) \leqslant \epsilon$. A similar proof can be constructed for sets $\mathring{r}_{\underline{h}_j} \equiv r_{-j}((h', \overline{r}_j])$ where

$$\underline{h}_j \equiv \sup\{\, h' \in [\underline{r}_j, \overline{r}_j] \mid P(r_{-j}([h', \overline{r}_j])) \geqslant \epsilon\}.$$

□

Remark on Lemma 1: In the proof of the main theorem we will need to construct rectangles that have probability greater than or equal to $\epsilon/2d$, but ensure that the the interior has probability strictly less than $\epsilon/2d$. This lemma establishes such result without the need to assume absolute continuity of the probability measure. We can think of constructing these rectangles by slowly increasing the maximum (or minimum), h, in the j^{th} dimension, until the probability is greater than or equal to $\epsilon/2d$. Then, a rectangle that doesn't contain this endpoint will have probability less than or equal to $\epsilon/2d$. Clearly this relies on only the continuity from above of all probability measures, as opposed to assuming absolute continuity. Note also that for absolutely continuous probability distributions, our construction gives rectangles of mass exactly equal to $\epsilon/2d$.

We can now move onto the proof of the first term of the upper bound $2d/\epsilon \ln(2d/\delta)$.

Proof. The target concept is $[\lambda(S)]$; which we have defined as the smallest hyper-rectangle containing the set $\lambda(S)$. In $\mathbb{R}^d$, we define $[\lambda(S)]$ as

$$[\lambda(S)] = \bigtimes_{j=1}^{d} [\underline{r}_j, \overline{r}_j],$$

Let $\boldsymbol{\theta}_M = (\theta_1, \ldots, \theta_M)$ be a sample of size M drawn from the distribution P, which need not be absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^p$. Fix $\epsilon > 0$ and consider a hypothesis $[\widehat{\lambda}_M]$ as the proposed d-dimensional hyperrectangle generated by the learning algorithm at an arbitrary—albeit fixed—data realization.

Note that

$$\begin{aligned}\mathcal{L}([\widehat{\lambda}_M];[\lambda(S)],P) &= P(\mathbf{1}\{\lambda(\theta)\in[\widehat{\lambda}_M]\}\neq\mathbf{1}\{\lambda(\theta)\in[\lambda(S)]\})\\ &= P\left(\lambda(\theta)\in[\widehat{\lambda}_M]\quad\text{and}\quad\lambda(\theta)\notin[\lambda(S)]\right)\\ &+ P\left(\lambda(\theta)\notin[\widehat{\lambda}_M]\quad\text{and}\quad\lambda(\theta)\in[\lambda(S)]\right).\end{aligned}\tag{11}$$

Note that the definition of $[\widehat{\lambda}_M]$ implies that if $\lambda(\theta)\in[\widehat{\lambda}_M]$ then $\lambda(\theta)\in[\lambda(S)]$ as $[\widehat{\lambda}_M]\subseteq[\lambda(S)]$. Therefore the second term in (16) is 0 and:

$$\mathcal{L}([\widehat{\lambda}_M];[\lambda(S)],P) = P(\lambda(\theta)\in[\lambda(S)]\backslash[\widehat{\lambda}_M]).\tag{12}$$

Our argument to show that our algorithm learns will rely on the construction of $2d$ d-dimensional 'special' hyperrectangles $(r_1, r_2, \cdots, r_{2d})$. These hyperrectangles will be used to bound the misclassification error of our learning algorithm. The construction is based on Lemma 1 and it goes as follows.

SPECIAL HYPERRECTANGLES: For any odd j in the set $\{1, 2, \ldots, 2d\}$ define

$$\overline{h}_j \equiv \inf\{\, h' \in [\underline{r}_j, \overline{r}_j] \mid P(r_{-j}([\underline{r}_j, h'])) \geqslant \epsilon/2d\}$$

and consider the 'special' hyperrectangle $r_j := r_{-j}([\underline{r}_j, \overline{h}_j])$. Note that $\overline{h}_j$ is well-defined as by assumption $P(\lambda(S)) = 1$, which implies $P([\lambda(S)]) = 1$.

Likewise, for any even index j in the set $\{1, 2, \ldots, 2d\}$, let:

$$\underline{h}_j \equiv \sup\{\, h' \in [\underline{r}_j, \overline{r}_j] \mid P(r_{-j}([h', \overline{r}_j])) \geqslant \epsilon/2d\}$$

and let $r_j := r_{-j}([\underline{h}_j, \overline{r}_j])$.

The constructed hyperrectangles are 'special' because of two reasons. First note that, by construction, the probability of the special hyperrectangles is lower

bounded:

$$P(r_j) \geqslant \epsilon/2d.$$

Second, note that:

$$P\left(\bigcup_{j=1}^{2d} \mathring{r}_j\right) \leqslant \sum_{j=1}^{2d} P(\mathring{r}_j) \leqslant \sum_{j=1}^{2d} \epsilon/2d \leqslant \epsilon, \tag{13}$$

where $\mathring{r}_j \equiv r_{-j}([\underline{r}_j, \overline{h}_j))$ for j odd and $\mathring{r}_j \equiv r_{-j}((\underline{h}_j, \overline{r}_j])$ for j even, and the last inequality follows from Lemma 1, which implies that $P(\mathring{r}_j) \leqslant \epsilon/2d$, for all $j = 1, \ldots, 2d$.

Bound on the misclassification error: Now we use the special hyperrectangles to bound the misclassification error. For each $j \in \{1, 2, \ldots, 2d\}$ consider the event:

$$E_j \equiv \left\{(\theta_1, \ldots, \theta_M) \mid [\widehat{\lambda}_M] \cap r_j \neq \varnothing\right\}.$$

This event contains the samples in which our algorithm intersects the j^{th} special hyperrectangle. We claim that:

$$(\theta_1, \theta_2, \ldots, \theta_M) \in \bigcap_{j=1}^{2d} E_j \implies [\lambda(S)] \backslash [\widehat{\lambda}_M] \subseteq \bigcup_{j=1}^{2d} \mathring{r}_j,$$

and, consequently, $\mathcal{L}(\widehat{\lambda}_M; [\lambda(S)], P) \leqslant \epsilon$. To verify such a claim, take any point $\lambda \in [\lambda(S)] \backslash [\widehat{\lambda}_M]$. Since $[\widehat{\lambda}_M]$ is a rectangle, we can write it as:

$$[\widehat{\lambda}_M] = [\widehat{\underline{r}}_1, \widehat{\overline{r}}_1] \times \ldots \times [\widehat{\underline{r}}_d, \widehat{\overline{r}}_d].$$

Since $\lambda \notin [\widehat{\lambda}_M]$, there must exist a coordinate—denote it λ_j—such either $\lambda_j > \widehat{\overline{r}}_j$ or $\lambda_j < \widehat{\underline{r}}_j$. Without loss of generality, assume that $\lambda_j < \widehat{\underline{r}}_j$. Since $[\widehat{\lambda}_M]$ intersects every special rectangle, in particular it intersects $r_{2(j-1)}$, which implies that $\lambda_j \leqslant$

$\widehat{\underline{r}}_j \leqslant \underline{h}_{2(j-1)}$. Consequently, $\lambda \in \mathring{r}_{2(j-1)}$.

From (13) and (17):

$$(\theta_1, \theta_2, \ldots, \theta_M) \in \bigcap_{j=1}^{2d} E_j \implies \mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) \leqslant \epsilon. \tag{14}$$

Learning guarantee: Our goal is now to find the required number of samples M such that the probability of the event in which $\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) > \epsilon$ is less than δ. We have shown that the event $\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) > \epsilon$ implies that

$$(\theta_1, \theta_2, \ldots, \theta_M) \notin \bigcap_{j=1}^{2d} E_j,$$

or equivalently, that $[\widehat{\lambda}_M] \cap r_j = \varnothing$ for some j. Therefore, it will suffice to show that we can find a sample size large enough such that the events E_j^c have arbitrarily small probability. Note that by definition of $[\widehat{\lambda}_M]$, the event E_j happens if and only if $\exists\ m(j) \in \{1, 2, \ldots, M\}$ such that:

$$\lambda(\theta_{m(j)}) \in r_j \quad \text{and} \quad \theta_{m(j)} \in S. \tag{15}$$

This means that E_j^c happens if there is no M such that (15) happens. Note that:

$$\begin{aligned}
P\left(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) > \epsilon\right) &\leqslant P\left((\theta_1, \theta_2, \ldots, \theta_M) \in \bigcup_{j=1}^{2d} E_j^c\right) \\
&\quad \text{(by (14))}, \\
&\leqslant \sum_{j=1}^{2d} P([\widehat{\lambda}_M] \cap r_j = \varnothing) \\
&\quad \text{(by Boole's inequality)}, \\
&\leqslant \sum_{j=1}^{2d} P(\nexists\ \theta_m \text{ s.t. both } \lambda(\theta_m) \in r_j \text{ and } \theta_m \in S) \\
&\quad \text{(by definition of } [\widehat{\lambda}_M] \text{, as explained in (15))}, \\
&= \sum_{j=1}^{2d} P(\forall\ \theta_m \text{ either } (\lambda(\theta_m) \notin r_j) \text{ or } (\theta_m \notin S))
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{j=1}^{2d} P(\lambda(\theta_m) \notin r_j \text{ or } \theta_m \notin S)^M \\
&\text{(as } \theta_m \text{ are i.i.d.)}, \\
&\leqslant \sum_{j=1}^{2d} \left(P(\lambda(\theta_m) \notin r_j) + P(\theta_m \notin S)\right)^M \\
&\text{(by Boole's inequality)}, \\
&= \sum_{j=1}^{2d} P((\lambda(\theta_m) \notin r_j)^m \\
&\text{(as } P(S) = 1), \\
&\leqslant 2d(1 - \epsilon/2d)^M \\
&\text{(as } P(r_j) \geqslant \epsilon/2d), \\
&\leqslant 2d \exp\left(\frac{-M\epsilon}{2d}\right) \\
&\text{(as } 1 - x \leqslant \exp(-x) \text{ for all } x \in \mathbb{R}).
\end{aligned}
$$

Thus for any $\delta > 0$, to ensure $P\left(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) > \epsilon\right) \leqslant \delta$, we require $2d \exp\left(\frac{-M\epsilon}{2d}\right) \leqslant \delta$. Rearranging for M yields $M \geqslant \frac{2d}{\epsilon} \ln\left(\frac{2d}{\delta}\right)$. □

Second term: $\exp(1)(2d + \ln(1/\delta))\}/\epsilon$

Proof. The target concept is $[\lambda(S)]$; which we have defined as the smallest hyper-rectangle containing the set $\lambda(S)$. In $\mathbb{R}^d$, we define $[\lambda(S)]$ as

$$
[\lambda(S)] \equiv \bigtimes_{j=1}^{d} \left[\inf_{\theta \in S} \lambda_j(\theta) \ , \ \sup_{\theta \in S} \lambda_j(\theta)\right],
$$

where $\lambda_j(\theta)$ denotes the j^{th} coordinate of $\lambda(\theta)$. Note that

$$
\begin{aligned}
\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) &= P(\mathbf{1}\{\lambda(\theta) \in [\widehat{\lambda}_M]\} \neq \mathbf{1}\{\lambda(\theta) \in [\lambda(S)]\}) \\
&= P\left(\lambda(\theta) \in [\widehat{\lambda}_M] \quad \text{and} \quad \lambda(\theta) \notin [\lambda(S)]\right) \\
&+ P\left(\lambda(\theta) \notin [\widehat{\lambda}_M] \quad \text{and} \quad \lambda(\theta) \in [\lambda(S)]\right). \qquad (16)
\end{aligned}
$$

Note that the definition of $[\widehat{\lambda}_M]$—and the fact that P has all of its mass on S—imply

that if $\lambda(\theta) \in [\widehat{\lambda}_M]$ then $\lambda(\theta) \in [\lambda(S)]$. Therefore the second term in (16) is 0 and:

$$\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) = P(\lambda(\theta) \in [\lambda(S)] \backslash [\widehat{\lambda}_M]). \tag{17}$$

Let $\boldsymbol{\theta}_M$ denote the row vector $(\theta_1, \ldots, \theta_M)$. For any $p > 0$ we can bound the probability of misclassification error greater than ϵ:

$$\begin{aligned}
&P_M(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P) > \epsilon) = P_M(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P)^p > \epsilon^p) \\
&\quad \leqslant \frac{1}{\epsilon^p} \mathbb{E}_P[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P)^p] \\
&\quad \text{(where the last line follows by Markov's inequality),} \\
&\quad = \frac{1}{\epsilon^p} \int P(\lambda(\theta) \in [\lambda(S)] \backslash [\widehat{\lambda}_M])^p dP^m(\boldsymbol{\theta}_M) \\
&\quad = \frac{1}{\epsilon^p} \int \left[\prod_{k=M+1}^{M+p} P(\lambda(\theta_k) \in [\lambda(S)] \backslash [\widehat{\lambda}_M]) \right] dP^m(\boldsymbol{\theta}_M) \\
&\quad \text{(assuming } \theta_k \sim P \text{ i.i.d, independently from } \boldsymbol{\theta}_M\text{),}
\end{aligned}$$

Let $\boldsymbol{\theta}_p \equiv (\theta_{M+1}, \ldots, \theta_{M+P})$, and $\boldsymbol{\theta}_{M+p} \equiv (\boldsymbol{\theta}_M, \boldsymbol{\theta}_p)$.

$$\begin{aligned}
&\quad = \frac{1}{\epsilon^p} \int \left[P(\lambda(\theta_k) \in [\lambda(S)] \backslash [\widehat{\lambda}_M], \forall k = M+1, \ldots, M+p) dP^p(\boldsymbol{\theta}_p) \right] dP^M(\boldsymbol{\theta}_M) \\
&\quad \text{(by independence),} \\
&\quad = \frac{1}{\epsilon^p} \int \left[\int \mathbf{1}\{\lambda(\theta_k) \in [\lambda(S)] \backslash [\widehat{\lambda}_M], \forall k = M+1, \ldots, M+p\} dP^p(\boldsymbol{\theta}_p) \right] dP^M(\boldsymbol{\theta}_M) \\
&\quad = \frac{1}{\epsilon^p} \int \mathbf{1}\{\lambda(\theta^*_k) \in [\lambda(S)] \backslash [\widehat{\lambda}_M], \forall k = 1, \ldots, p\} dP^{M+p}(\boldsymbol{\theta}_{M+p}) \\
&\quad \text{(by Fubini's theorem),} \\
&\quad \leqslant \frac{1}{\epsilon^p} \sup_{\boldsymbol{\theta}_{M+p}} \int \mathbf{1}\{\lambda(\theta_{\sigma(k)}) \in [\lambda(S)] \backslash [\hat{\lambda}_M](\theta_{\sigma(1)}, \ldots, \theta_{\sigma(M)}), \forall k = M+1, \ldots, M+p\} dU(\sigma) \\
&\quad \text{(Haussler, Littlestone and Warmuth (1994) p. 258 Corollary 2.1),}
\end{aligned}$$

where σ is a permutation of the elements of $\boldsymbol{\theta}_{M+p}$, $U(\sigma)$ is the uniform distribution on these permutations, and—in a slight abuse of notation—we have made explicit the dependence of $[\widehat{\lambda}_M]$ on the values of the first M elements of the permuted sample,

which we denote as $(\theta_{\sigma(1)}, \ldots, \theta_{\sigma(M)})$.

Define now $\boldsymbol{\theta}_{\sigma(M)} \equiv \{\theta_{\sigma(1)}, \ldots, \theta_{\sigma(M)}\}$, and similarly $\boldsymbol{\theta}_{\sigma(p)} \equiv \{\theta_{\sigma(M+1)}, \ldots, \theta_{\sigma(M+p)}\}$ to be the sets that contain, respectively, the first M and the last p elements of the σ permutation of the row vector $\boldsymbol{\theta}_{M+p}$. We can majorize the probability of a misclassification error greater than ϵ by the supremum of the probability that p points are misclassified, with respect to a random permutation, σ, of the $M+p$ points, $\boldsymbol{\theta}_{M+p}$. Note that the order in the M points used to build $[\hat{\lambda}_M](\boldsymbol{\theta}_{\sigma(M)})$ do not matter, and neither do the p elements, $\boldsymbol{\theta}_{\sigma(p)}$ used to to assess misclassification (this is why we have denoted $\boldsymbol{\theta}_{\sigma(M)}$ and $\boldsymbol{\theta}_{\sigma(p)}$ as sets and not vectors). Because the order of the elements doesn't matter, it suffices to consider how many combinations (i.e. unordered sets) of size M out of the $M+p$ elements in $\boldsymbol{\theta}_{M+p}$ we will have that when given to the algorithm misclassify all the remaining p points.

The total number of combinations of M elements from $M+p$ is $\binom{M+p}{M}$ which equals $\binom{M+p}{p}$. Denote by $\mathcal{B}(\boldsymbol{\theta}_{M+p})$ the collection of all subsets of size M (which we have chosen to denote as $\boldsymbol{\theta}_{\sigma(M)}$) that generate p misclassifications; that is $\theta \notin [\widehat{\lambda}_M](\boldsymbol{\theta}_{\sigma(M)})$ for every $\theta \in \boldsymbol{\theta}_{\sigma(p)}$. We will show that $|\mathcal{B}(\boldsymbol{\theta}_{M+p})| \leqslant \binom{2d+p-1}{p}$, and that this is true for any sample of points $\boldsymbol{\theta}_{M+p}$. Thus, continuing our inequality from above yields

$$\begin{aligned} P_M(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P) > \epsilon) &\leqslant \frac{1}{\epsilon^p} \frac{\binom{2d+p-1}{p}}{\binom{M+p}{p}} \\ &= \frac{1}{\epsilon^p} \frac{(2d+p-1)\ldots 2d}{(m+p)\ldots(m+1)} \\ &< \left(\frac{2d+p}{\epsilon m}\right)^p. \end{aligned}$$

If we choose $p = \log \frac{1}{\delta}$ and $m \geqslant \frac{\exp(1)(2d+p)}{\epsilon}$, then

$$\left(\frac{2d+p}{\epsilon m}\right)^p \leqslant \left(\frac{1}{\exp(1)}\right)^{\lceil \log \frac{1}{\delta} \rceil} \leqslant \delta,$$

and thus $m = \frac{\exp(1)}{\epsilon}\left(2d + \log \frac{1}{\delta}\right)$ will suffice.

Now let's prove the claim $|\mathcal{B}(\boldsymbol{\theta}_{M+p})| \leqslant \binom{2d+p-1}{p}$. We will do so by constructing—

for an arbitrary sample of $M+p$ points $\boldsymbol{\theta}_{M+p}$ that leads to p misclassifications—an injective map φ from $\mathcal{B}(\boldsymbol{\theta}_{M+p})$ to the set of sequences of $2d$ nonnegative integers summing to p. It is known that there are at most $\binom{2d+p-1}{p}$ possible sequences of $2d$ nonnegative integers that sum to p.[30] If φ is injective, this will prove the upper bound for the cardinality of $\mathcal{B}(\boldsymbol{\theta}_{M+p})$. For the rest of the proof, we will write $\mathcal{B}$ without reference to $\boldsymbol{\theta}_{M+p}$. We will also use B to make reference to an element of $\mathcal{B}$, and we will use B^c to denote its complement relative to $\boldsymbol{\theta}_{M+p}$ that is $B^c \equiv \{\theta_1, \ldots, \theta_{M+p}\} \backslash B$.

For a given $B \in \mathcal{B}$ our algorithm for estimating the tightest bands containing S is:

$$[\widehat{\lambda}_M](B) = \bigtimes_{j=1}^{d} [\underline{\widehat{r}}_j, \overline{\widehat{r}}_j],$$

where $\underline{\widehat{r}}_j = \min_{m|l(\theta_m)=1} \lambda_j(\theta_m)$ and $\overline{\widehat{r}}_j = \max_{m|l(\theta_m)=1} \lambda_j(\theta_m)$. Since P puts probability one on S all the elements in a sample, we consider that all labels equal to 1. This means that the min/max in each dimension are taken over all the elements in B.

If $\lambda(\theta) \in [\lambda(S)] \backslash [\widehat{\lambda}_M]$, there must exist at least one dimension—denote it $\lambda_j(\theta)$—such either $\lambda_j(\theta) > \overline{\widehat{r}}_j$ or $\lambda_j(\theta) < \underline{\widehat{r}}_j$. For each element $B \in \mathcal{B}$ we wish to construct a partition of the elements in B^c that allocates the misclassification of a point to a particular dimension, and direction (either min or max).

We will construct this partition in the following way:

$$\begin{aligned}
C_1(B) &\equiv \{\theta_m \in B^c \,|\, \lambda_1(\theta_m) < \underline{\widehat{r}}_1\}, \\
C_2(B) &\equiv \{\theta_m \in B^c \,|\, \lambda_1(\theta_m) > \overline{\widehat{r}}_1\}, \\
C_3(B) &\equiv \{\theta_m \in B^c \,|\, \lambda_2(\theta_m) < \underline{\widehat{r}}_2\} \backslash (C_1(B) \cup C_2(B)), \\
C_4(B) &\equiv \{\theta_m \in B^c \,|\, \lambda_2(\theta_m) > \overline{\widehat{r}}_2\} \backslash (C_1(B) \cup C_2(B)),
\end{aligned}$$

[30] To see this, note that the set of sequences of $2d$ nonnegative integers summing up to p is the support of the a multinomial distribution with $2d$ categories and p trials. The total number of support points is $\binom{2d+p-1}{2d-1}$ which, by definition, equals $\binom{2d+p-1}{p}$.

$$\vdots$$

$$C_{2d-1}(B) \equiv \{\theta_m \in B^c \,|\, \lambda_d(\theta_m) < \underline{\widehat{r}}_d\} \backslash \bigcup_{j=1}^{2d-2} C_j(B),$$

$$C_{2d}(B) \equiv \{\theta_m \in B^c \,|\, \lambda_d(\theta_m) > \widehat{\overline{r}}_d\} \backslash \bigcup_{j=1}^{2d-2} C_j(B).$$

Suppose j is odd. $C_j(B)$ contains all the points of misclassified p which can be 'blamed for being too little' on dimension $(j+1)/2$ (and haven't been blamed for being too little or too big on a smaller dimension). $C_{j+1}(B)$ are the points in dimension $(j+1)/2$ that were too big. From this point onward, we will call $j = 1, \ldots, 2d$ 'dimensions', and wlog will focus on odd j.

Note that $\bigcup_{j=1}^{2d} C_j(B) = B^c$, and that by construction $C_j(B)$ are disjoint. Given that B^c contains p points, this implies $\sum_{j=1}^{2d} |C_j(B)| = p$. Hence $\{C_j(B)|\}_{j=1}^{2d}$ is a sequence of nonnegative integers summing to p.

Therefore we can define our map φ as

$$\varphi(B) = (|C_1(B)|, \ldots |C_{2d}(B)|).$$

It remains to be shown that φ is an injective map. By showing that this map is injective we will be able to bound the cardinality of $\mathcal{B}$ by the cardinality of the image of φ. Thus, we need to show that for every $B_1, B_2 \in \mathcal{B}$, $\varphi(B_1) = \varphi(B_2)$ implies $B_1 = B_2$. We will show the contrapositive $B_1, B_2 \in \mathcal{B}$, $B_1 \neq B_2$ implies $\varphi(B_1) \neq \varphi(B_2)$.

Take an arbitrary $B_1, B_2 \in \mathcal{B}$, such that $B_1 \neq B_2$. First we want to show that $R_1 \equiv [\widehat{\lambda}](B_1) \neq [\widehat{\lambda}](B_2) \equiv R_2$, which is to say that the sample of M points B_1 generates a different rectangle than the sample B_2.

Suppose by contradiction that $R_1 = R_2$. As $B_1 \neq B_2$, there exists $\theta^* \in B_1$, such that $\theta^* \in B_2^c$. Because $\theta^* \in B_1$, it is correctly classified by R_1, and because $R_1 = R_2$, it is also correctly classified by R_2. So we have $\theta^* \in B_2^c$ and θ^* is correctly classified by R_2, which contradicts the definition of B_2^c (which only contains misclassified

points).

Now we want to show $\varphi(B_1) \neq \varphi(B_2)$ whenever $B_1 \neq B_2$. As $R_1 \neq R_2$, this implies there exists a dimension, such the bounds for the rectangles in that dimension do not agree. Call the smallest dimension in which this occurs j^* and suppose wlog that $\underline{\widehat{r}}^1_{j^*} < \underline{\widehat{r}}^2_{j^*}$.

There exists $\theta^* \in B_1$, such that $\lambda_{j^*}(\theta^*) = \underline{\widehat{r}}^1_{j^*} < \underline{\widehat{r}}^2_{j^*} \leqslant \lambda_{j^*}(\theta)$ for all $\theta \in B_2$. Because j^* is the smallest dimension in which the rectangles don't agree, $\theta^* \notin C_{j'}(B_2)$ for $j' < j^*$. Take any $\theta_m \in B_1^c$ for which $\lambda_{j^*}(\theta_m) < \underline{\widehat{r}}^1_{j^*}$. Since $\underline{\widehat{r}}^1_{j^*} < \underline{\widehat{r}}^2_{j^*}$, we can conclude that θ_m cannot be in B_2. Therefore $\{\theta_m \in B_1^c \,|\, \lambda_{j^*}(\theta_m) < \underline{\widehat{r}}^1_{j^*}\} \subseteq \{\theta_m \in B_2^c \,|\, \lambda_{j^*}(\theta_m) < \underline{\widehat{r}}^2_{j^*}\}$. Moreover, since $\theta^* \notin C_{j'}(B_2)$ for $j' < j^*$ the set inclusion must be strict. Also, because j^* is the smallest dimension in which R_1 and R_2 differ, we must have $C_{j'}(B_1) = C_{j'}(B_2)$ for all $j' < j^*$.

This implies $C_{j^*}(B_1) \subset C_{j^*}(B_2)$, and $|C_{j^*}(B_1)| < |C_{j^*}(B_2)|$, which in turn implies $\varphi(B_1) \neq \varphi(B_2)$. □

Proof of lower bound

The proof of the upper bound was for any $P \in \mathcal{P}(S)$. In order to construct a lower bound on the sample complexity we construct a specific probability distribution in $\mathcal{P}(S)$, and find the required number of draws to learn from the inside.

We will prove separately the two terms in the maximum.

First term: $(1-\epsilon)/\epsilon \ln(1/\delta)$

Proof. By assumption there exists a concept $\lambda(S) \in \Lambda$ that has two different points. This means that there exists at least two different points in S, denoted θ_1 and θ_2. Consider the probability distribution

$$P(\theta_1) = 1 - \epsilon, \quad P(\theta_2) = \epsilon.$$

Note that this probability distribution belongs $\mathcal{P}(S)$, as $P(S) = 1$.

Suppose that we observe a sample of size M that contains only the value θ_1. The probability of observing such a sample is

$$P((\underbrace{\theta_1, \theta_1, \ldots, \theta_1}_{m \text{ times}})) = (1-\epsilon)^M.$$

On this sample, our algorithm reports the set $\{\lambda(\theta_1)\}$, but misclassifies $\lambda(\theta_2)$. Hence the when we observe this sample, the loss is

$$\mathcal{L}([\widehat{\lambda}_M], [\lambda(S)], P) = P(\theta_2) = \epsilon.$$

Hence

$$\begin{aligned} P\left(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) \geqslant \epsilon\right) &\geqslant P\left(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) = \epsilon\right) \\ &= P((\theta_1, \theta_1, \ldots, \theta_1)) \\ &= (1-\epsilon)^M. \end{aligned}$$

Learning from the inside, implies that $P(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) \geqslant \epsilon) \leqslant \delta$, and hence learning from the inside implies that $(1-\epsilon)^M \leqslant \delta$. Re-arranging for M yields

$$M \geqslant \frac{\ln(1/\delta)}{-\ln(1-\epsilon)}.$$

Therefore in order to learn from the inside, we require $M \geqslant \frac{\ln(1/\delta)}{-\ln(1-\epsilon)}$. In particular as $\frac{1}{-\ln(1-\epsilon)} \geqslant \frac{1-\epsilon}{\epsilon}$ for all $\epsilon \in (0,1)$, learning from the inside with $[\widehat{\lambda}_M]$ implies that $M \geqslant \frac{1-\epsilon}{\epsilon} \ln\left(\frac{1}{\delta}\right)$. Thus, the smallest $m(\epsilon, \delta)$ required to learn from the inside has to be at least $\frac{1-\epsilon}{\epsilon} \ln\left(\frac{1}{\delta}\right)$.

□

Second term: $\frac{3d}{16\epsilon}$

Assumption 3. *For each dimension $j \in \{1, \ldots, d\}$ define*

$$\underline{L}_j \equiv \arg\min_{\theta \in S} \lambda_j(\theta), \quad \overline{U}_j \equiv \arg\max_{\theta \in S} \lambda_j(\theta).$$

These are the θ's in S that give the smallest and largest values in each dimension. Suppose that S is such that there exists a collection of $2d$ different points

$$\boldsymbol{\theta}^* = \{\underline{\theta}_1, \overline{\theta_1}, \ldots, \underline{\theta}_d, \overline{\theta_d}\},$$

such that

1. $\underline{\theta}_j \in \underline{L}_j$ *and* $\overline{\theta}_j \in \overline{U}_j$ *for all* $j = 1, \ldots, 2d$
2. $\underline{\theta}_j, \overline{\theta}_j \notin \underline{L}_{j'}, \overline{U}_{j'}$ *for all* $j' \neq j$,
3. $\underline{\theta}_j < \overline{\theta}_j$ *for all* j.

Proof. Fix the set S and the function λ. Define the set $\lambda(S) \in \mathbb{R}^d$ as we have previously done.

The construction in Assumption 3 implies that each of the $2d$ points in $\boldsymbol{\theta}^*$ is an extreme point (minimum or maximum) of $\lambda(S)$ in one and only one dimension. This implies that given a sample that contains a subset of these $2d$ points, $B \subset \boldsymbol{\theta}^*$, our algorithm $[\widehat{\lambda}_M]$ will misclassify all the points in $\boldsymbol{\theta}^* \backslash B$. Let θ_0 be an element of S satisfying $\theta_0 \notin \underline{L}_j, \overline{U}_j$ for all j. Figure 10, below, illustrates our construction in the case in which $S \subseteq \mathbb{R}^2$ (depicted in blue), $\lambda(\cdot)$ is the identity, and the rectangle is the smallest band containing S.

Consider the following probability distribution over $\boldsymbol{\theta}^* \cup \theta_0$,

$$P^*(\theta_0) = 1 - 8\epsilon, \quad P^*(\theta_i) = \frac{8\epsilon}{2d}, \text{ for } i = 1, \ldots, 2d.$$

This distribution is well-defined as all points are inside S and $\epsilon \leqslant \frac{1}{8}$.

Denote $\boldsymbol{\theta}_M$ as a i.i.d. sample of size M from P^*. Our construction implies that any sample $\boldsymbol{\theta}_M$ from P^* has the property

$$\mathbf{1}\{\theta \notin \boldsymbol{\theta}_M\} = \mathbf{1}\{\lambda(\theta) \notin [\widehat{\lambda}_M]\}, \tag{18}$$

since a sample that contains a subset of the $2d$ points in θ^* (say $B \subset \boldsymbol{\theta}^*$) will imply

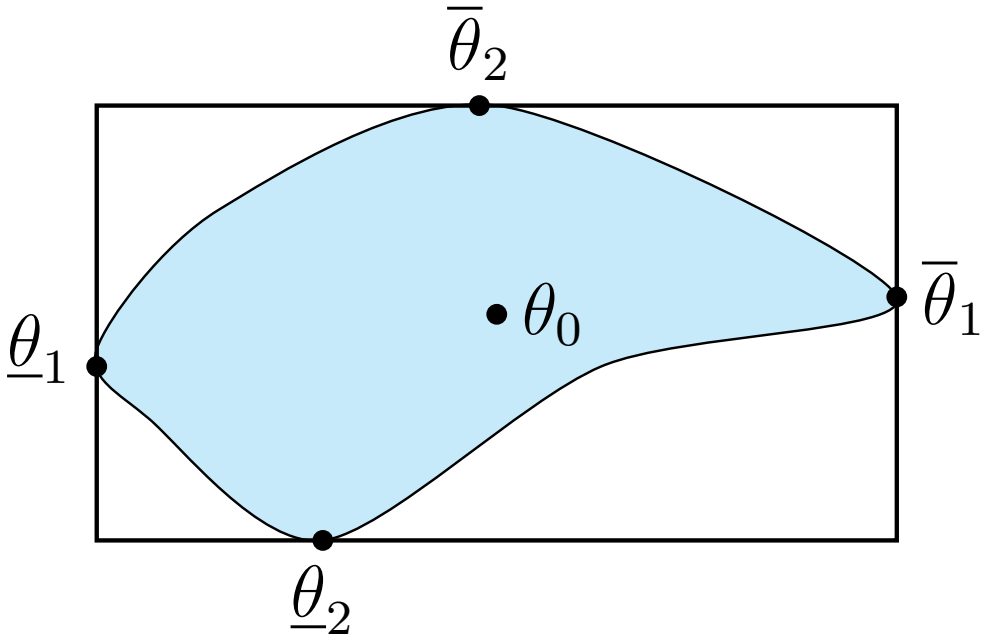

Figure 10: The construction of $\boldsymbol{\theta}^* \cup \theta_0$.

that, for such sample, our algorithm $[\widehat{\lambda}_M]$ will misclassify all the points in $\boldsymbol{\theta}^* \backslash B$.

We will show that for any sample size $M \leqslant \frac{3d}{16\epsilon}$, we have

$$\mathbb{E}[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] \geqslant 2\epsilon.$$

We can write the misclassification error as

$$\begin{aligned}
\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) &= P^*(\lambda(\theta) \notin [\widehat{\lambda}_M]) \\
&= \sum_{\theta \in \boldsymbol{\theta_0} \cup \boldsymbol{\theta}^*} P^*(\theta) \mathbf{1}\{\lambda(\theta) \notin [\widehat{\lambda}_M]\} \\
&\geqslant \sum_{\theta \in \boldsymbol{\theta}^*} P^*(\theta) \mathbf{1}\{\lambda(\theta) \notin [\widehat{\lambda}_M]\} \\
&= \sum_{\theta \in \boldsymbol{\theta}^*} P^*(\theta) \mathbf{1}\{\theta \notin \boldsymbol{\theta}_M\}
\end{aligned}$$

where we have used the fact that—by construction of P^*—any sample $\boldsymbol{\theta}_M$ has the property in equation (18). Consequently

$$\mathbb{E}[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] \geqslant \sum_{\theta \in \boldsymbol{\theta}^*} P^*(\theta) P^*(\theta \notin \boldsymbol{\theta}_M).$$

Note that for any $\theta \in \boldsymbol{\theta}^*$, the probability that it is not in the sample of size M, $\boldsymbol{\theta}_M$ is

$$\begin{aligned} P^*(\theta \notin \boldsymbol{\theta}_M) &= \prod_{m=1}^{M} P^*(\theta_m \neq \theta) \\ &= P^*(\theta_m \neq \theta)^M \\ &= \left(1 - \frac{8\epsilon}{2d}\right)^M. \end{aligned}$$

Therefore,

$$\begin{aligned} \mathbb{E}[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] &= \sum_{\theta \in \boldsymbol{\theta}^*} P^*(\theta) \left(1 - \frac{8\epsilon}{2d}\right)^M \\ &= 2d \frac{8\epsilon}{2d} \left(1 - \frac{8\epsilon}{2d}\right)^M \\ &\geqslant 8\epsilon \left(1 - \frac{M 8\epsilon}{2d}\right) \\ &\text{(by Bernoulli's inequality)} \\ &\geqslant 8\epsilon \left(1 - \frac{3}{4}\right) \\ &\text{(as } M \leqslant (3d)/(16\epsilon)) \\ &= 2\epsilon. \end{aligned}$$

Note that $\mathbb{E}[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] \leqslant \sum_{\theta \in \boldsymbol{\theta}^*} P^*(\theta) = 8\epsilon$. Therefore,

$$\begin{aligned} 2\epsilon \leqslant & \mathbb{E}\left[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)\right] \\ \leqslant & P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) > \epsilon\right) \\ & \times \mathbb{E}\left[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) \,\middle|\, \mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) > \epsilon\right] \\ & + \left(1 - P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] > \epsilon\right)\right) \\ & \times \mathbb{E}\left[\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) \,\middle|\, \mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) \leqslant \epsilon\right] \\ \leqslant & P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) > \epsilon\right) 8\epsilon \end{aligned}$$

$$
\begin{aligned}
&+ \left(1 - P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] > \epsilon\right)\right)\epsilon \\
\leqslant & P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] > \epsilon\right) 7\epsilon + \epsilon.
\end{aligned}
$$

Collecting terms we get

$$
P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] > \epsilon\right) \geqslant \frac{1}{7\epsilon}(2\epsilon - \epsilon) = \frac{1}{7}.
$$

Therefore for $\epsilon \leqslant \frac{1}{8}$ and $\delta \leqslant \frac{1}{8} < \frac{1}{7}$, we have

$$
P\left(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*)] > \epsilon\right) > \delta,
$$

whenever $M \leqslant \frac{3d}{16\epsilon}$. □

A.5 Proof of Proposition 1

Proof. The tightest band containing $\lambda(S)$ is defined as before:

$$
[\lambda(S)] \equiv \bigtimes_{i=1}^{d} [\underline{r}_i, \overline{r}_i].
$$

Also let $[\widehat{\lambda}_M]$ be defined as,

$$
[\widehat{\lambda}_M] \equiv \bigtimes_{i=1}^{d} [\hat{\underline{r}}_i, \hat{\overline{r}}_i],
$$

where, by construction of our algorithm, $[\widehat{\lambda}_M]$ is a subset of $[\lambda(S)]$.

Define a vertex v of $[\lambda(S)]$ to be a point in $\mathbb{R}^d$, where the i^{th} coordinate is either $\underline{r}_i$ or $\overline{r}_i$ for $i = 1, \ldots, d$. Define $\widehat{v}$ to be sample analogue of v (replacing $\underline{r}_i$ and $\overline{r}_i$ by $\widehat{\underline{r}}_i$ and $\widehat{\overline{r}}_i$).

The *worst-case relative Hausdorff* distance between $[\lambda(S)]$ and $[\widehat{\lambda}_M]$ was defined as

$$
\tilde{d}_H([\lambda(S)], [\widehat{\lambda}_M]) = \frac{d_H([\lambda(S)], [\widehat{\lambda}_M])}{\sup_{[\hat{\lambda}_M] \subseteq [\lambda(S)]} d_H([\lambda(S)], [\widehat{\lambda}_M])}. \tag{19}
$$

The numerator can be shown to equal:[31]

$$d_H([\lambda(S)],[\widehat{\lambda}_M]) = \sup_{i=1,\ldots,d}\{\hat{\underline{r}}_i - \underline{r}_i, \overline{r}_i - \hat{\overline{r}}_i\}.$$

Consequently,

$$\tilde{d}_H([\lambda(S)],[\widehat{\lambda}_M]) = \frac{\sup_{i=1,\ldots,d}\{\hat{\underline{r}}_i - \underline{r}_i, \overline{r}_i - \hat{\overline{r}}_i\}}{\sup_{i=1,\ldots,d}\{\overline{r}_i - \underline{r}_i\}}.$$

We would like to show that the worst-case (19) is bounded above by the misclassification error between $[\lambda(S)]$ and $[\widehat{\lambda}_M]$. That means that if the misclassification error is smaller than ϵ, then the worst-case Hausdorff distance is also smaller than ϵ.

The argument goes as follows. First, Assumption 1 implies that $\underline{r}_i$ and $\overline{r}_i$ are finite for every i. Therefore, the worst-case Hausdorff is finite and equals

$$\begin{aligned}
\tilde{d}_H([\lambda(S)],[\widehat{\lambda}_M]) &= \frac{\sup_{i=1,\ldots,d}\{\hat{\underline{r}}_i - \underline{r}_i + (\overline{r}_i - \overline{r}_i), \overline{r}_i - \hat{\overline{r}}_i + +(\underline{r}_i - \underline{r}_i)\}}{\sup_{i=1,\ldots,d}\{\overline{r}_i - \underline{r}_i\}} \\
&= \frac{\sup_{i=1,\ldots,d}\{(\overline{r}_i - \underline{r}_i) - (\overline{r}_i - \hat{\underline{r}}_i), (\overline{r}_i - \underline{r}_i) - (\hat{\overline{r}}_i - \underline{r}_i)\}}{\sup_{i=1,\ldots,d}\{\overline{r}_i - \underline{r}_i\}} \\
&\leqslant \frac{\sup_{i=1,\ldots,d}\{(\overline{r}_i - \underline{r}_i) - (\hat{\overline{r}}_i - \hat{\underline{r}}_i)\}}{\sup_{i=1,\ldots,d}\{\overline{r}_i - \underline{r}_i\}} \\
&\text{(as } \hat{\overline{r}}_i \leqslant \overline{r}_i \text{ and } \hat{\underline{r}}_i \geqslant \underline{r}_i\text{)}, \\
&\leqslant \sup_{i=1,\ldots,d}\left\{\frac{(\overline{r}_i - \underline{r}_i) - (\hat{\overline{r}}_i - \hat{\underline{r}}_i)}{\overline{r}_i - \underline{r}_i}\right\} \\
&= 1 - \frac{\hat{\overline{r}}_{i*} - \hat{\underline{r}}_{i*}}{\overline{r}_{i*} - \underline{r}_{i*}}
\end{aligned}$$

[31] Given that vertices are the extreme points in both $[\lambda(S)]$ and $[\widehat{\lambda}_M]$, we can focus our attention of the differences between the two sets of vertices. We will now show that the distance between the two sets of vertices is the difference between the true vertex and its sample analogue.

Take a vertex of $[\lambda(S)]$, denoted v and w.l.o.g. assume $v \equiv [\overline{r}_1, \ldots, \overline{r}_d]$. Take the sample analogue of that vertex: $\hat{v} \equiv [\hat{\overline{r}}_1, \ldots, \hat{\overline{r}}_d]$ and a different vertex $\widehat{v}'$ of $[\widehat{\lambda}_M]$, with at least one different coordinate, in the j'th dimension $\hat{v}' \equiv [\hat{\overline{r}}_1, \ldots, \hat{\underline{r}}_j, \ldots, \hat{\overline{r}}_d]$.

The claim is that $d_\infty(v, \hat{v}) \leqslant d_\infty(v, \hat{v}')$. Suppose not. Then

$$\begin{aligned}
& d_\infty(v, \hat{v}) > d_\infty(v, \hat{v}') \\
\iff & \max\{\overline{r}_1 - \hat{\overline{r}}_1, \ldots, \overline{r}_j - \hat{\overline{r}}_j, \ldots, \overline{r}_d - \hat{\overline{r}}_d\} > \max\{\overline{r}_1 - \hat{\overline{r}}_1, \ldots, \overline{r}_j - \hat{\underline{r}}_j, \ldots, \overline{r}_d - \hat{\overline{r}}_d\}
\end{aligned}$$

Clearly this can only be true if the max is obtained over the j'th dimension, which implies $\overline{r}_j - \hat{\overline{r}}_j > \overline{r}_j - \hat{\underline{r}}_j \implies \hat{\underline{r}}_j > \hat{\overline{r}}_j$, our desired contradiction.

(where i^* is the dimension over which the sup is attained)

$= 1 - P(\lambda_{i*}(S) \in [\hat{\underline{r}}_{i*}, \hat{\overline{r}}_{i*}])$

(as $\lambda_{i*}(S) \sim U[\underline{r}_{i*}, \overline{r}_{i*})]$ for any dimension, by Assumption 2),

$\leqslant 1 - P\left(\bigcap_{i=1}^{d} \lambda_i(S) \in [\hat{\underline{r}}_i, \hat{\overline{r}}_i]\right)$

$= P(\lambda \in [\lambda(S)] \backslash [\widehat{\lambda}_M])$

(as $P(S) = 1$)

$= \mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P).$

Thus, we have shown that the worst-case Hausdorff distance is at most the misclassification error.

A.6 Learning smallest bands in the real line when λ contains one element that is not a band

Assume the set reported when there are no positive labels, does not depend on the sampled λ's. Consider three cases.

Case 1: suppose that we have an algorithm that, absent positive labels, reports a nonempty set $\boldsymbol{\lambda}' \in \Lambda$ that does not intersect $[0, 2]$. In this case, any probability measure P that places all of its mass on $\boldsymbol{\lambda}'$ will have, with probability 1, misclassification error of 1 relative to $[0, 2]$ if the true set is $[0, 1] \cup \{2\}$.

Case 2: Absent positive labels, the algorithm reports a set $\boldsymbol{\lambda}'$ such that $[0, 2] \cap \boldsymbol{\lambda}' = [0, 2]$, or equivalently $[0, 2] \subseteq \boldsymbol{\lambda}'$. Take any other set $\boldsymbol{\lambda}'' \in \Lambda$ in the complement of $\boldsymbol{\lambda}'$ to be the true set generating the labels. Consider a probability measure P that places all of its mass on $[0, 2]$. Since $\boldsymbol{\lambda}''$ is the true set, any sample will lack positive labels which implies that $\boldsymbol{\lambda}'$ will be reported. Thus, with probability 1, misclassification error will be 1, as $[0, 2] \cap \boldsymbol{\lambda}'' = \varnothing$.

CASE 3: Suppose that the set $\boldsymbol{\lambda}'$ reported by the algorithm is such that $[0, 2] \cap \boldsymbol{\lambda}' \neq [0, 2]$ and $[0, 2] \cap \boldsymbol{\lambda}' \neq \varnothing$.

CASE 3.1: $\boldsymbol{\lambda}' = [0, 1] \cup \{2\}$. Take any set $\boldsymbol{\lambda}''$ in the complement of $[0, 2]$ to be the true set and consider a probability measure that puts all of its mass in $[0, 1] \cup \{2\}$. Any sample from P will lack positive labels, thus $\boldsymbol{\lambda}'$ will be reported with probability 1. Since $\boldsymbol{\lambda}''$ was chosen to satisfy $\boldsymbol{\lambda}' \cap \boldsymbol{\lambda}'' = \varnothing$, misclassification error is 1 with probability 1.

CASE 3.2: $\boldsymbol{\lambda}'$ is an interval $[a, b]$ that intersects $[0, 2]$. Suppose the true set is $\boldsymbol{\lambda}'' \equiv ([a, b] \cup [0, 2])\backslash[a, b]$, which is an interval. Take P to be any probability distribution that places all of its mass on $\boldsymbol{\lambda}' = [a, b]$. This means that any sample will lack positive labels. Since $\boldsymbol{\lambda}' \cap \boldsymbol{\lambda}'' = \varnothing$, misclassification error is again 1 with probability 1.

It would seem that this result holds more generally, as long as the concept class contains at least 2 sets. The argument would go as follows:

Suppse that absent positive labels, the algorithm reports $\boldsymbol{\lambda}'$. Suppose the true set is $\boldsymbol{\lambda}''$, such that $\boldsymbol{\lambda}' \nsubseteq [\boldsymbol{\lambda}'']$, and suppose a measure P that puts mass 1 on $\boldsymbol{\lambda}'\backslash[\boldsymbol{\lambda}'']$. Because P puts no mass on $[\boldsymbol{\lambda}'']$, we will always receive no positive labels, and hence output $\boldsymbol{\lambda}'$. We will misclassify all points in $\boldsymbol{\lambda}'\backslash[\boldsymbol{\lambda}'']$, which occur w.p.1, and hence we will have a misclassification error of 1 with probability 1.

A.7 Learning with sampling error

Let S be our set of interest, and $\hat{S}_n$ an estimator of it.

Assumption 4.

$$d_H([\lambda(S)], [\lambda(\hat{S}_n)]) \xrightarrow{p} 0, \tag{20}$$

where d_H is defined in equation Footnote 20.

Define

$$[\lambda(S)] \equiv \times_{j=1}^{d}[\underline{r}_j^*, \overline{r}_j^*],$$
$$[\lambda(\hat{S}_n)] \equiv \times_{j=1}^{d}[\hat{\underline{r}}_j, \hat{\overline{r}}_j],$$
$$[\lambda(\hat{S}_n^\eta)] \equiv \times_{j=1}^{d}[\hat{\underline{r}}_j^\eta, \hat{\overline{r}}_j^\eta],$$

where $\hat{S}_n^\eta$ is an arbitrary subset of $\hat{S}_n$.

Fix $\eta > 0$, and pick a set $\hat{S}_n^\eta$ for which

$$\inf_{j=1,\ldots,d} \min\{|\hat{\underline{r}}_j - \hat{\underline{r}}_j^\eta|, |\hat{\overline{r}}_j - \hat{\overline{r}}_j^\eta|\} \geqslant \eta. \tag{21}$$

Define $P_\eta(\hat{S}_n)$ to be the class of probability measures that are supported on a set $\hat{S}_n^\eta$ selected above.

Pick (ϵ, δ), and let $M \equiv M(\epsilon, \delta)$ denote the number of draws recommended by Theorem 3. Suppose that the estimator $\hat{S}_n$ satisfies

$$d_H([\lambda(S)], [\lambda(\hat{S}_n)]) < \eta/2$$

(which we know happens with high probability for n sufficiently large). The misclassification error between $[\hat{\lambda}_M]$ and $[\lambda(S)]$ is

$$\begin{aligned}\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) = {} & P^*(\lambda(\theta) \in [\hat{\lambda}_M], \lambda(\theta) \notin [\lambda(S)]) \\ & + P^*(\lambda(\theta) \notin [\hat{\lambda}_M], \lambda(\theta) \in [\lambda(S)]).\end{aligned} \tag{22}$$

Take any $P^* \in P_\eta(\hat{S}_n)$. Note that the second term in (22) equal to

$$P^*(\lambda(\theta) \notin [\hat{\lambda}_M], \lambda(\theta) \in [\lambda(S)], \lambda(\theta) \in [\lambda(\hat{S}_n^\eta)]),$$

as P^* is supported on $\hat{S}_n^\eta$, so any draw from P^* belongs to $[\lambda(\hat{S}_n^\eta)]$. This term is no

greater than

$$P^*(\lambda(\theta) \notin [\hat{\lambda}_M], \lambda(\theta) \in [\lambda(\hat{S}_n^\eta)]) = \mathcal{L}([\hat{\lambda}_M]; [\lambda(\hat{S}_n^\eta)], P^*) < \epsilon,$$

with probability $1 - \delta$ by Theorem 3.

For any $P^* \in P_\eta(\hat{S}_n)$, the first term in (22) is

$$P^*(\lambda(\theta) \in [\hat{\lambda}_{M*}], \lambda(\theta) \notin [\lambda(S)], \lambda(\theta) \in [\lambda(\hat{S}_n^\eta)]) \leqslant$$
$$P^*(\lambda(\theta) \notin [\lambda(S)], \lambda(\theta) \in [\lambda(\hat{S}_n^\eta)]).$$

Consider the event $\{\lambda(\theta) \notin [\lambda(S)]\} \cup \{\lambda(\theta) \in [\lambda(\hat{S}_n^\eta)]\}$. $\lambda(\theta) \notin [\lambda(S)]$ implies that $\lambda_j(\theta) \notin [\underline{r}_j^*, \overline{r}_j^*]$ for some dimension j. Suppose w.l.o.g. that $\lambda_j(\theta) < \underline{r}_j^*$. Since

$$d_H([\lambda(\hat{S}_n)], [\lambda(S)]) < \eta/2, \tag{23}$$

which is justified by (20). By equation (23), we have

$$\lambda(\theta) < \underline{r}_j^* < \hat{\underline{r}}_j + \eta/2 < \hat{\underline{r}}_j + \eta.$$

By (21), we have that $\hat{\underline{r}}_j + \eta < \hat{\underline{r}}_j^\eta$. Finally, given that our draws come from P^*, we must have $\hat{\underline{r}}_j^\eta \leqslant \lambda(\theta)$. Combining these inequalities, yields $\lambda(\theta) < \lambda(\theta)$, which occurs with probability zero. Hence the first term in (22) is equal to 0.

Therefore, for any estimator $\widehat{S}_n$ that satisfies 23, we have

$$P^*(\mathcal{L}([\hat{\lambda}_M]; [\lambda(S)], P^*) < \epsilon) \geqslant 1 - \delta.$$

Pepe and James

□

B Appendix B

In this appendix we consider the possibility of computing the misclassification error when the probability measure Q (the measure used by the oracle to compute misclassification error) differs from P (the measure used by the econometrician to generate random draws). Given an algorithm $\widehat{\lambda}_M$, the misclassification error of learning a concept λ thus becomes $\mathcal{L}(\widehat{\lambda}_M, \lambda, Q)$.

A concept $\lambda \in \Lambda$ is (Q, P) learnable if there exists an algorithm $\widehat{\lambda}_M$ and a function $m(\epsilon, \delta)$ such that for any $0 < \epsilon$ and $\delta < 1$:

$$P\left(\mathcal{L}(\widehat{\lambda}_M; \lambda, Q) < \epsilon\right) \geqslant 1 - \delta, \tag{24}$$

for all distributions P on Θ and for any $\lambda \in \Lambda$; provided $M \geqslant m(\epsilon, \delta)$.

We establish the following results

1. We provide a simple example, where θ has dimension $d = 1$, that shows that learning in the sense of (24) is impossible even if Λ has finite VC dimension. The example shows that when Q and P are different, learning becomes complicated because there is a lot of flexibility in the choice of P.

2. We also show that, not surprisingly, if we restrict P to belong to a class P_Q such that

$$\sup_{A \in \text{ continuity sets of } Q} |P(A) - Q(A)| \leqslant \eta,$$

for sufficiently small η, then learning is possible (for a fixed ϵ and δ) and the sufficient number of draws becomes

$$\ln\left(\frac{1}{\delta}\right)\frac{1}{2\epsilon - \eta}, \quad \eta < 2\epsilon.$$

which is larger than $\ln\left(\frac{1}{\delta}\right)\frac{1}{2\epsilon}$; the number of draws that would be required if P were equal to Q.

The example suggests that allowing P and Q to differ does not add much to our previous results.

EXAMPLE:

Suppose that the parameter of interest lives in the real line, so that $d = 1$. Suppose that the concept class contains elements of the form $[a, \infty)$. The class has VC dimension 1.[32]

For notational simplicity, we identify sets of the form $[\lambda, \infty)$, $[\widehat{\lambda}, \infty)$ by the scalars λ, $\widehat{\lambda}$. Algebra shows that

$$P\left(\mathcal{L}(\widehat{\lambda}; \lambda, Q)\right) = |Q(\lambda) - Q(\widehat{\lambda})|.$$

Assume that Q is absolutely continuous with respect to the Lebesgue measure.

We show that in this example, learning is not possible. It is sufficient to show that for **any** algorithm $\widehat{\lambda}_M$, there exists ϵ, δ and λ such that for some P

$$P\left(\mathcal{L}(\widehat{\lambda}_M; \lambda, Q) \geqslant \epsilon\right) \geqslant \delta.$$

regardless of the sample size.

Fix $\lambda \in \mathbb{R}$ and let $\widehat{\lambda}_M$ be an arbitrary algorithm. Let M be an arbitrary sample size.

Without loss of generality,[33] consider algorithms $\widehat{\lambda}_M : (x_1, \ldots, x_m) \to \mathbb{R}$ such that for any set $(a, b) \subset \mathbb{R}$,

$$\widehat{\lambda}_M^{-1}(a, b) \neq \varnothing. \tag{25}$$

[32] Suppose we have 1 point, then λ can label it either 0 or 1, implying one point can be shattered. Suppose there are 2 points. We can generate labels $(0, 0)$, $(1, 1)$ and $(0, 1)$, but can't generate $(1, 0)$ labels. 2 points cannot be shattered, and thus the VC dimension (the largest number of points that can be shattered) of Λ is 1.

[33] If this were not the case, consider any (a, b) for which $\widehat{\lambda}_M^{-1}(a, b) = \varnothing$. Then we could pick $\lambda \in (a, b)$ and set $\epsilon^* = \min\{Q(a) - Q(\lambda), Q(\lambda) - Q(a)\}$. In this case, we have that for any P:

$$\begin{aligned} P\left(\mathcal{L}(\widehat{\lambda}_M, \lambda, Q) \geqslant \epsilon^*\right) &\geqslant P(\widehat{\lambda}_M \geqslant b) + P(\widehat{\lambda}_M \leqslant a) \\ &= 1 - P(\widehat{\lambda}_M \in (a, b)) = 1 - P(\varnothing) = 1. \end{aligned}$$

Take an arbitrary value λ^*, and an arbitrary set $(\underline{\lambda}^*, \overline{\lambda}^*)$, such that $\lambda^* \in (\underline{\lambda}^*, \overline{\lambda}^*)$. $\epsilon^* = \min\{Q(\overline{\lambda}^*) - Q(\lambda^*), Q(\lambda^*) - Q(\underline{\lambda}^*)\} > 0$. Such a set exists as Q is absolutely continuous w.r.t. to the Lebesgue measure. For any algorithm satisfying (25) we have

$$P\left(\mathcal{L}(\widehat{\lambda}_M; \lambda, Q) \geqslant \epsilon^*\right) \geqslant 1 - P(\widehat{\lambda}_M \in (\underline{\lambda}^*, \overline{\lambda}^*)).$$

For any sample size—and given that P is unrestricted—there is a P such that $P(\widehat{\lambda}_M \in (\underline{\lambda}^*, \overline{\lambda}^*))$ can be made arbitrary small. The example shows learning is impossible, even if the concept has finite VC dimension.

Now we show that if we allow for probability distributions P that are close to Q, learning is still possible. The result is not surprising at all, and all we need is to use the right definition of "closeness". Let

$$P_Q^\eta \equiv \left\{ P \mid \sup_{A \in \text{ cont sets of } Q} |P(A) - Q(A)| \leqslant \eta \right\}.$$

We argue that the algorithm that sets $\widehat{\lambda}_M = \min\{x_i | x_i = 1\}$ or $\widehat{\lambda}_M = \max\{x_i | x_i = 0\}$ learns uniformly, for a fixed pair (ϵ, δ), where $\epsilon \geqslant \eta/2$.

The proof goes as follows. Fix $\lambda \in \mathbb{R}$. Find $\underline{\lambda}(\lambda) < \overline{\lambda}(\lambda)$ such that $Q(\overline{\lambda}(\lambda)) - Q(\lambda) = \epsilon = Q(\lambda) - Q(\underline{\lambda}(\lambda))$. Define the set $A(\lambda) = [\underline{\lambda}(\lambda), \overline{\lambda}(\lambda)]$. Then

$$\begin{aligned} P\left(\mathcal{L}(\widehat{\lambda}_M; \lambda, Q) \geqslant \epsilon\right) &= P(x_i \notin [\underline{\lambda}(\lambda), \overline{\lambda}(\lambda)], \quad \forall i) \\ &= (1 - P(A(\lambda)))^M. \end{aligned}$$

Note by definition $Q(A(\lambda)) = 2\epsilon$, which makes the line above equal to

$$(1 - [P(A(\lambda)) - Q(A(\lambda))] - 2\epsilon)^M,$$

implying

$$P(\mathcal{L}(\widehat{\lambda}_M; \lambda, Q) \geqslant \epsilon) \leqslant (1 - (2\epsilon - \eta))^M,$$

as for any $P \in P_Q^\eta$, we have $-\eta \leqslant P(A) - Q(A) \leqslant \eta$. Therefore for a fixed (ϵ, δ)

$$M \geqslant \ln\left(\frac{1}{\delta}\right) \frac{1}{2\epsilon - \eta}$$

suffices to learn the concept class. This requires more draws than when $Q = P$, which would be exactly

$$\ln\left(\frac{1}{\delta}\right) \frac{1}{2\epsilon}.$$

This formalizes the result that, if P is required to be sufficiently close to Q, then learning is indeed possible (but the number of draws required to learn is practically the same as when $P = Q$).

C Appendix D

In this section we describe how to (machine) learn parameter regions that arise in the Latent Dirichlet Allocation (LDA) model of Blei, Ng and Jordan (2003). For more details, see Ke, Montiel Olea and Nesbit (2019) (henceforth, KMN).

The LDA is a popular machine learning algorithm for text analysis. The LDA model assumes that there are K latent topics; a topic is a distribution over the V terms in the vocabulary, $\beta_k \in \Delta^{V-1}$. Each document d is characterized by a document-specific distribution over the K topics, $\theta_d \in \Delta^{K-1}$. The topic distributions $B \equiv (\beta_1, \ldots, \beta_K)$ and the topic compositions $\Theta \equiv [\theta_1, \ldots \theta_D]$ determine the mixture model for each word in document d.

Let $\mathbb{P}_d(t|B, \theta_d)$ denote the probability that a term $t \in \{1, \ldots, V\}$ appears in document d. The model assumes that

$$\mathbb{P}_d(t|B, \theta_d) = \sum_{k=1}^{K} \beta_{t,k}\theta_{k,d}.$$

The likelihood of corpus C is thus parameterized by (B, Θ) and given by

$$\begin{aligned}\mathbb{P}(C|B,\Theta) &= \prod_{d=1}^{D}\prod_{t=1}^{V}(\mathbb{P}_d(t|B,\Theta))^{n_{t,d}} \\ &= \prod_{d=1}^{D}\prod_{t=1}^{V}(B\Theta)_{t,d}^{n_{t,d}} \end{aligned} \tag{26}$$

where $n_{t,d}$ is count of the number of times term t appears in document d. We can collect the terms $\mathbb{P}_d(t)$ in the $V \times D$ matrix P and use (26) to write

$$P = B\Theta. \tag{27}$$

Theorem 1 of KMN shows that the parameters of the likelihood, B and Θ in (27) are set identified and thus the choice of prior matters. They show that the range of posterior means can be described using solutions to the rank K *Non-negative Matrix Factorization* (NMF) of the term-document frequency matrix with weight $W_{t,d}$, $\widehat{P}$—the non-negative matrices (B, Θ) that solve:

$$\min_{B\in\mathbb{R}_+^{V\times K},\Theta\in\mathbb{R}_+^{K\times D}} \sum_{i=1}^{D}\sum_{t=1}^{V} W_{t,d}\left[\hat{P}_{t,d}\log\left(\frac{\hat{P}_{t,d}}{(B\Theta)_{t,d}}\right) - \hat{P}_{t,d} + (B\Theta)_{t,d}\right]. \tag{28}$$

Let $\text{NMF}(\widehat{P}, W_{t,d})$, denote the set rank K NMF's. The set S is:

$$S \equiv \left\{(B,\Theta) \mid (B,\Theta) \in \text{NMF}(\widehat{P}, W_{t,d})\right\},$$

where (B, Θ) are assumed to be matrices whose columns are probability distributions.

The algorithm to solve for a solution of (28) is initialized randomly, and thus the algorithm induces a distribution over S. Thus we can compute the tightest bands that contain the set $\lambda(S)$ for some functional using random sampling.

KMN revisit the work of Hansen and McMahon (2016), studying the effects of increased transparency on the discussions of Federal Open Market Committee (FOMC). Let $\theta_{i,t}$ be the weight of i^{th} topic in meeting at time t, the Herfindahl

index for the topic distribution is given by

$$H_t \equiv \sum_{i=1}^{K} \theta_{i,t}^2.$$

The specific functional of interest is the 'transparency coefficient' (λ) in the regression of the concentration measure on a dummy for the date in which the Federal Reserve changed its transparency policy (October 1993) and controls

$$H_t = \alpha + \lambda D(Trans)_t + \gamma X_t + \epsilon_t.$$

Note that λ is one dimensional, hence $d = 1$. The data are the FOMC transcripts from August 1987–January 2006 which have been extensively preprocessed. The meetings are broken into two sections FOMC1 and FOMC2 and the regression is run separately on each. The resulting dimensions of the term-document matrices are 9000×148 and 6000×148 for FOMC1 and FOMC2 respectively. The number of topics is set to $K = 40$. The resulting number of parameters estimated is large—B is 9000×40 and Θ is 40×148, a total of $365,920$.

KMN take $M = 120$ draws from $\lambda(S)$, corresponding to a misclassification error of at most 5.91% with probability at least 94.09% ($\epsilon = \delta = 0.0591$) and the iso-draw curve presented in Figure 11.

The tightest band containing $\lambda(S)$ for FOMC1 is $[-0.0380, 0.0466]$, and for FOMC2 is $[-0.0615, 0.0350]$.

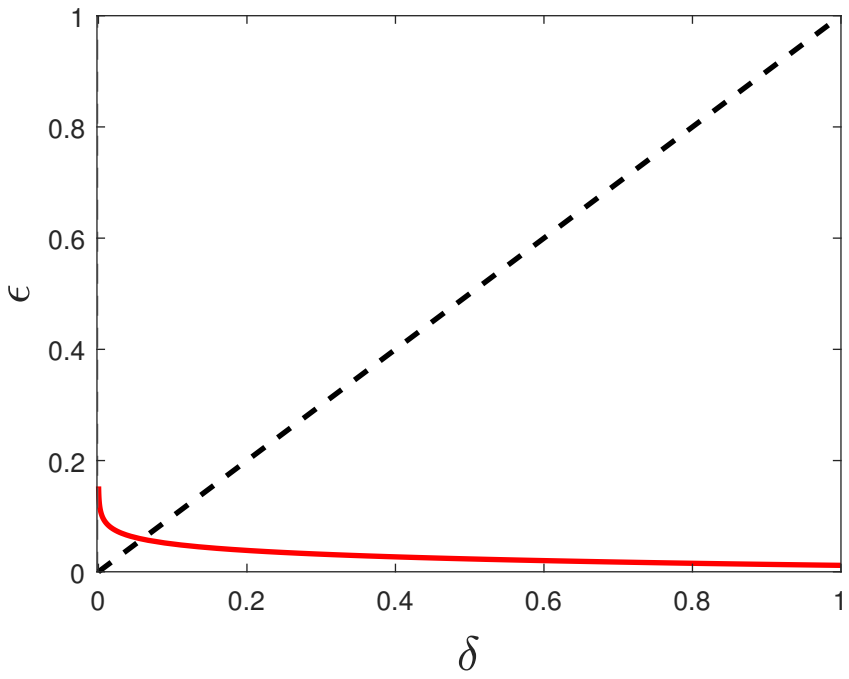

Figure 11: 'Iso-draw' curve for $M = 120$: the values of ϵ and δ that can be supported with 120 draws.