(Machine) Learning Parameter Regions*

José Luis Montiel Olea † and James Nesbit ‡

October 22, 2018

Abstract

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 it is 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 $(1-\epsilon)/\epsilon \ln(1/\delta)$ draws, but at most $(2d/\epsilon)\ln(2d/\delta)$. 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)

KEYWORDS: Misclassification Error, Sample Complexity, Supervised Learning, Structural Vector Autoregressions.

*We would like to thank Jonás Arias, Timothy Christensen, Timothy Cogley, Thorsten Drautzburg, Lutz Kilian, Mikkel Plagborg-Møller, Guillaume Pouliot, Andres Santos, Azeem Shaikh, seminar participants at Columbia University, the Federal Reserve Bank of Philadelphia, New York University, the 2018 Conference on Optimisation and Machine Learning in Economics, and two anonymous referees 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) 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* ensures 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 the *generalization error* or simply the misclassification probability (see Definition 2.1 in Mohri, Rostamizadeh and Talwalkar (2012)). When P equals Q—that is, when the measure used by the

[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.

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]

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. To enforce the agreement, the oracle computes misclassification error in Equation (1) with respect to the tightest band containing the parameter region of interest. Throughout the paper we denote such tightest band as $[\lambda(S)]$.

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

[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 C 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 $P = Q$.

[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.

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.

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. We show that the *sample complexity* of this algorithm—that is, the minimal number of draws required to achieve learning—can be bounded from above by $(2d/\epsilon)\ln(2d/\delta)$ and below by $((1-\epsilon)/\epsilon)\ln(1/\delta)$.[5]

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 $1,982$. This translates to almost $15,000$ 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, 2018)) 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 $(\epsilon,\ \delta)$ that could be supported with this number of draws. Our formulae imply that $2,000$ total draws to summarize a 68% Wald

[5] 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.

ellipse are sufficient to support the combination $\epsilon = \delta = 0.1377$. In particular, this implies that $2,000$ total draws are sufficient to guarantee that with probability at least 87.23% probability, the misclassification error less than 13.77%.

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 compares the bands for identified sets computed via random sampling in Section 3.1 with those obtained by nonlinear programming using the algorithm in Gafarov, Meier and Montiel Olea (2018). Appendix C 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).

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 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.

We allow for the possibility that the parameter of interest is not θ per se, but instead the image of θ under some 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 object of interest, λ could be the identity map.

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.[6] 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

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

$l(\theta_m) = 1$ 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*.[7] 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. The econometrician's task is to use a sample $\{(\lambda(\theta_m), l(\theta_m))\}_{m=1}^{M}$ to select a concept $\widehat{\lambda}_M \in \Lambda$ 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 (1) assuming Q (the measure used by the oracle) equals P (the measure used by the econometrician to generate random draws). 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 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 $\boldsymbol{\lambda} \in \Lambda$; provided $M \geqslant m(\epsilon, \delta)$.

This concept of learnability 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. *$\lambda(S) \in \Lambda \subseteq 2^{\lambda(\Theta)}$ is learnable if and only if Λ has finite Vapnik-Chervonenkis (VC) dimension.*

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

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. 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).

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.[8]

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.[9] The final message of this section is that learning $\lambda(S)$, in the conventional sense of the word, is difficult and oftentimes impossible.

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 bands 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

[8] See Appendix A.1 for a definition and discussion of VC dimension.
[9] If $d > 1$ then the VC dimension of the class of convex sets in $\mathbb{R}^d$ is infinity.

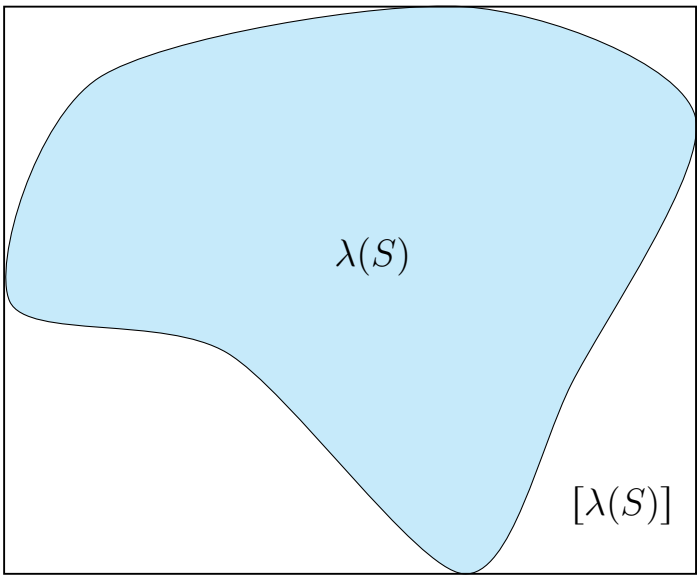

Figure 1: $\lambda(S)$ and $[\lambda(S)]$.

Rai (2018), Montiel Olea and Plagborg-Møller (2018)). 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-aligned hyperrectangles*) are objects of low complexity: the VC dimension of a band 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. The following algorithm—which 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)—would guarantee learning.

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*

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

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

The algorithm simply keeps track of the largest and smallest '1-labeled' draws of $\lambda_j(\theta_m)$ in each of the $j = 1, \ldots, d$ dimensions. Note that if there is no draw θ_m for which $l(\theta_m) = 1$, the algorithm above outputs the empty set.

Can the algorithm $[\widehat{\lambda}_M]$ learn $[\lambda(S)]$ in the sense of 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 $M \geqslant m(\epsilon, \delta)$? Unfortunately, we answer this question in the negative. Theorem 2 below shows that even if we allow ourselves to compute misclassification error relative to $[\boldsymbol{\lambda}]$, learning is still not possible due the richness of the class of probability distributions under consideration.

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:*

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

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

Proof. See Appendix A.3. □

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

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. In particular, Theorem 2 shows that the richness of the class of probability distributions for which (3) must hold is to blame for the impossibility result. 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.

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 $P \in \mathcal{P}(S)$ we will have that $P(\lambda(\theta) \in [\boldsymbol{\lambda}] \backslash \boldsymbol{\lambda}) = 0$. We use this class to relax the learning desideratum presented in Definition 1.

Note that in practice, sampling from inside of S need not be an issue for the econometrician. For any distribution P' that places positive mass on S, we can

construct a distribution $P \in \mathcal{P}(S)$, by simply discarding all draws that fall outside S. The cost we pay to learn from the inside is that the required number of draws has to come from inside S.

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 the 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 $[\lambda(S)]$, which is obtained under the assumption that the true parameter region $\lambda(S)$ contains at least two different points.

Theorem 3. *The algorithm* $[\widehat{\lambda}_M]$ *in Definition 2 learns* $[\lambda(S)]$ *from the inside. Moreover, the sample complexity of* $[\widehat{\lambda}_M]$*—denoted* $m^*(\epsilon,\delta)$*—admits the upper bound:*

$$m^*(\epsilon,\delta) \leqslant (2d/\epsilon)\ln(2d/\delta).$$

Additionally if $\lambda(S)$ *contains at least two different points, then* $m^*(\epsilon,\delta)$ *admits the lower bound:*

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

Proof. See Appendix A.4. □

The upper bound on the sample complexity 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 $(2k/\epsilon)\ln(2k/\delta)$ 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 $42,586$ draws of impulse response coefficients that satisfy the sign restrictions are sufficient to learn. The number of draws necessary to learn (the lower bound) is 456 draws.

We understand that it might be difficult for the researcher to take a stand on his/her desired combination of ϵ and δ. 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 Figure 2 for a parameter region of dimension $d = 25$.

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 for it. This approximation can be learned, even though we only have labels for $\lambda(S)$ and not $[\lambda(S)]$. The price that

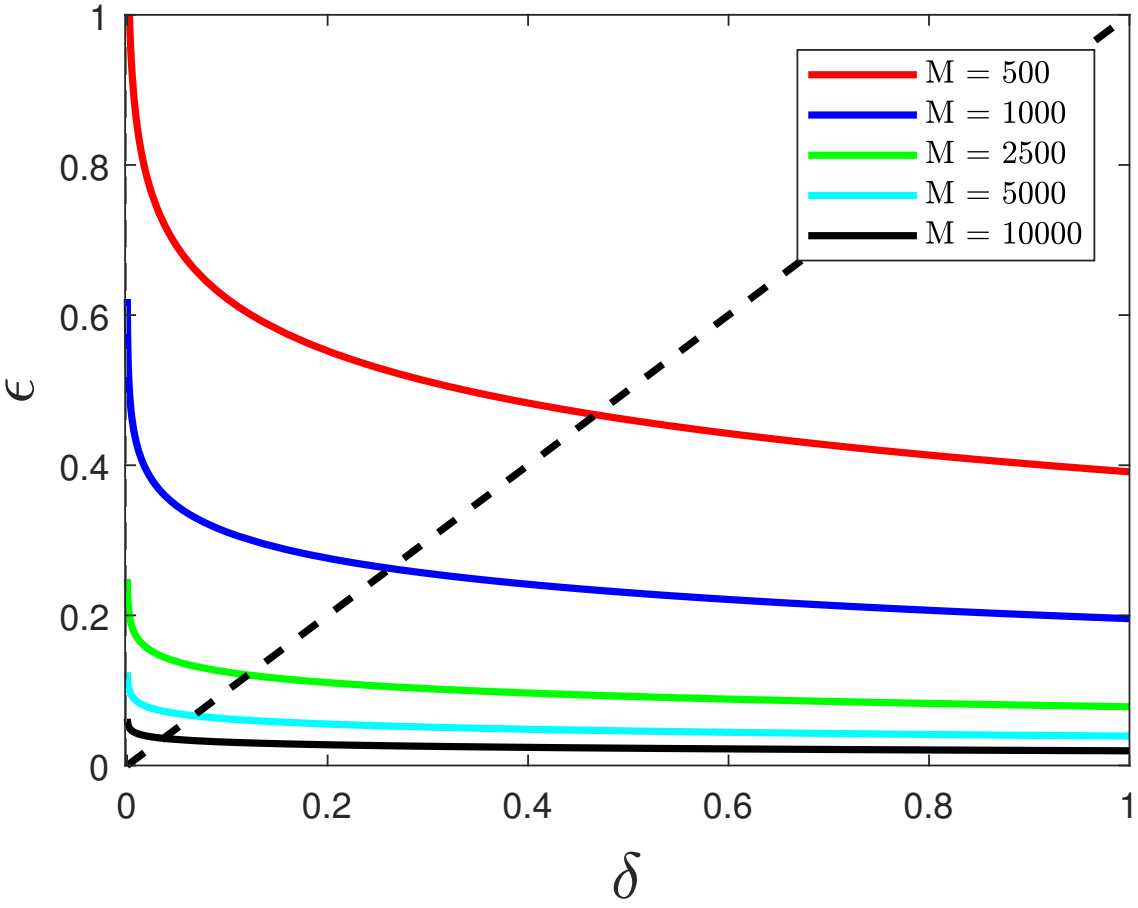

Figure 2: Iso-draw curves. For a fixed M, the combinations of (ϵ, δ) such that $M = (2d/\epsilon)\ln(2d/\delta)$. In this example $d = 25$.

we pay for this, is that we can only guarantee learning for distributions that draw from inside $\lambda(S)$.

Note also that the bound on the sample complexity grows 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.

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 variables have quarterly frequency and the sample period is October 1982 to October 2007.[10] The model is given by

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

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

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

$$\lambda_{k,i,3}(A, B) = e_i' C_k(A) B e_3, \tag{5}$$

where e_i denotes the i^{th} column of $\mathbb{I}_3$ and $A \equiv (A_1, \ldots, A_4)$.[11]

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 (5). If the restrictions allow the econometrician to 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

[11] $C_k(A)$ is defined recursively by the formula $C_0 \equiv \mathbb{I}_n$, 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).

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. The identified set—the set of structural matrices B that satisfies the sign restrictions—is defined as:

$$S \equiv \left\{ B \in \mathbb{R}^{3x3} \middle| BB' = \widehat{\Sigma}, \text{ and } B \text{ satisfies the sign restrictions in Table 1} \right\}.$$

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

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

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.

Algorithm 2 of RRWZ can be used to sample at random from inside the set S to describe $\lambda(S)$. Let M denote the total number of draws. 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 (5). 3) If the impulse responses do not satisfy the sign restrictions, discard the draw and return to step 1. 4) Repeat until one has M draws.

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

$$(2d/\epsilon)\ln(2d/\delta) = 1,982.$$

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 $1,982$ draws of B that satisfy the sign restrictions.[12]

Figure 3 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 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).

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

[12] To the best of our knowledge there are no theory-based suggestions on how many draws are required to stop the RRWZ algorithm. (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. The theorems in this paper complement the existing recommendations.

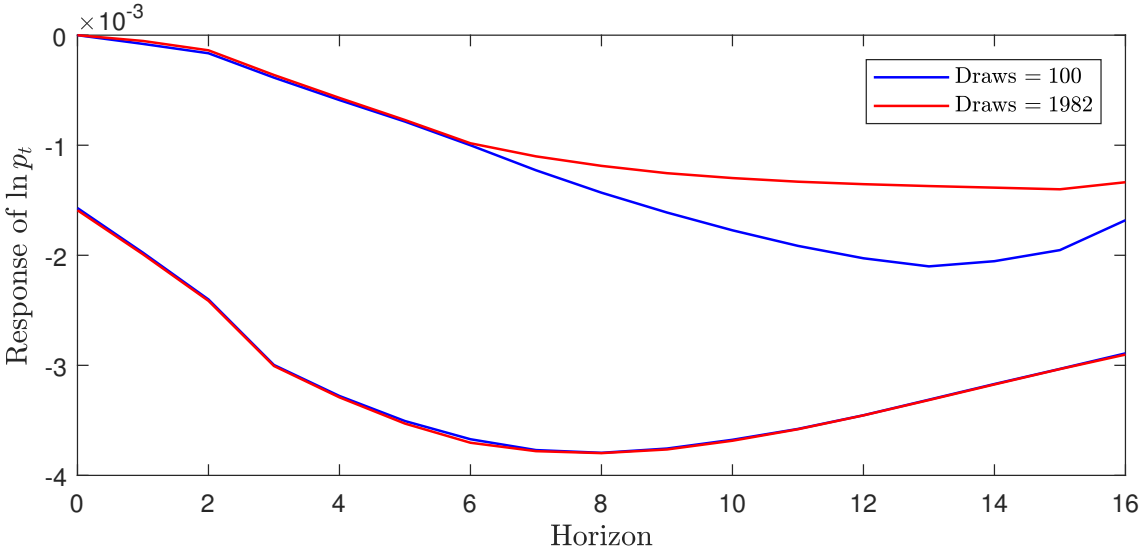

Figure 3: The tightest band that contains the identified set. The parameter region $\lambda(S)$ is defined as the dynamic responses of $\ln p_t$ to a contractionary monetary policy shock on impact and for 16 quarters after impact. The sufficient number of draws from inside the parameter region required to learn, for $\epsilon = \delta = 0.1$ and with $d = 17$, is $1,982$. These bands are plotted in RED. Bands constructed using 100 draws from within $\lambda(S)$ are plotted in BLUE.

identified models, in which all draws of B satisfy the sign restrictions, and fall inside S.

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.[13] Under such an identification scheme, the vector of 17 impulse responses, denoted γ, can be estimated consistently. The goal here is to summarize a Wald ellipse reporting shotgun plots as in Inoue and Kilian (2016) (IK henceforth).

Define the Wald statistic

$$W(\gamma) = (\gamma - \hat{\gamma}_T)'\hat{\Sigma}^{-1}(\gamma - \hat{\gamma}_T), \tag{6}$$

where $\hat{\gamma}_T$ is the least squares estimator of γ and $\hat{\Sigma}$ 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.

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

Note that in this example, λ is the identity 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 4 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

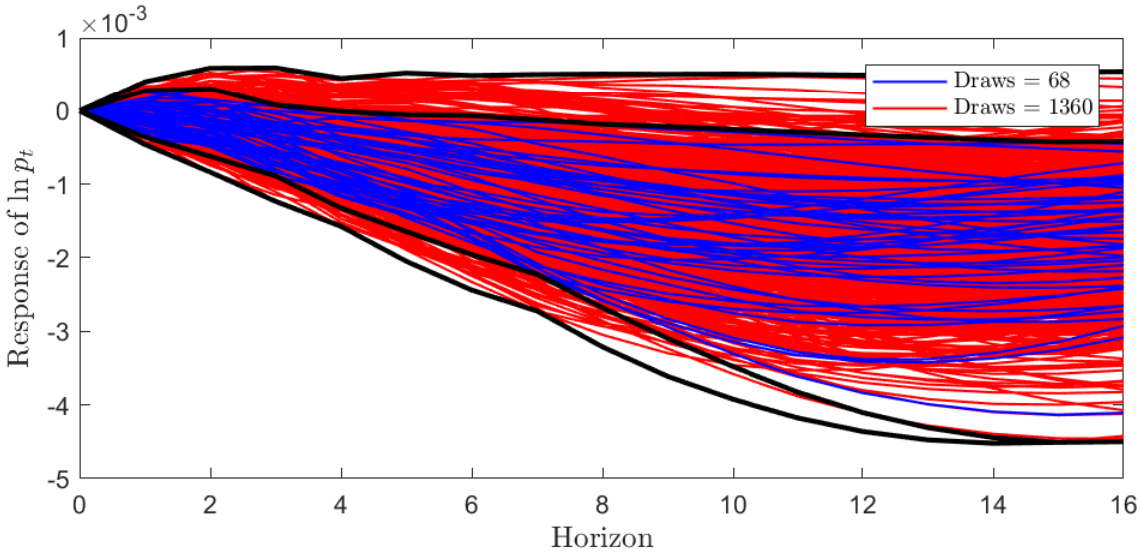

Figure 4: Shotgun plot of the 68% joint confidence region of the dynamic response of $\ln p_t$ to a monetary policy shock. BLUE and RED lines represent 68 and $1,360$ draws from inside S, respectively. BLACK lines represent the minimum and maximum (pointwise) at each horizon.

of choosing specific values of (ϵ, δ), Figure 5 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.

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 $p(\gamma \mid y^T)$ as the posterior density of the dynamic structural

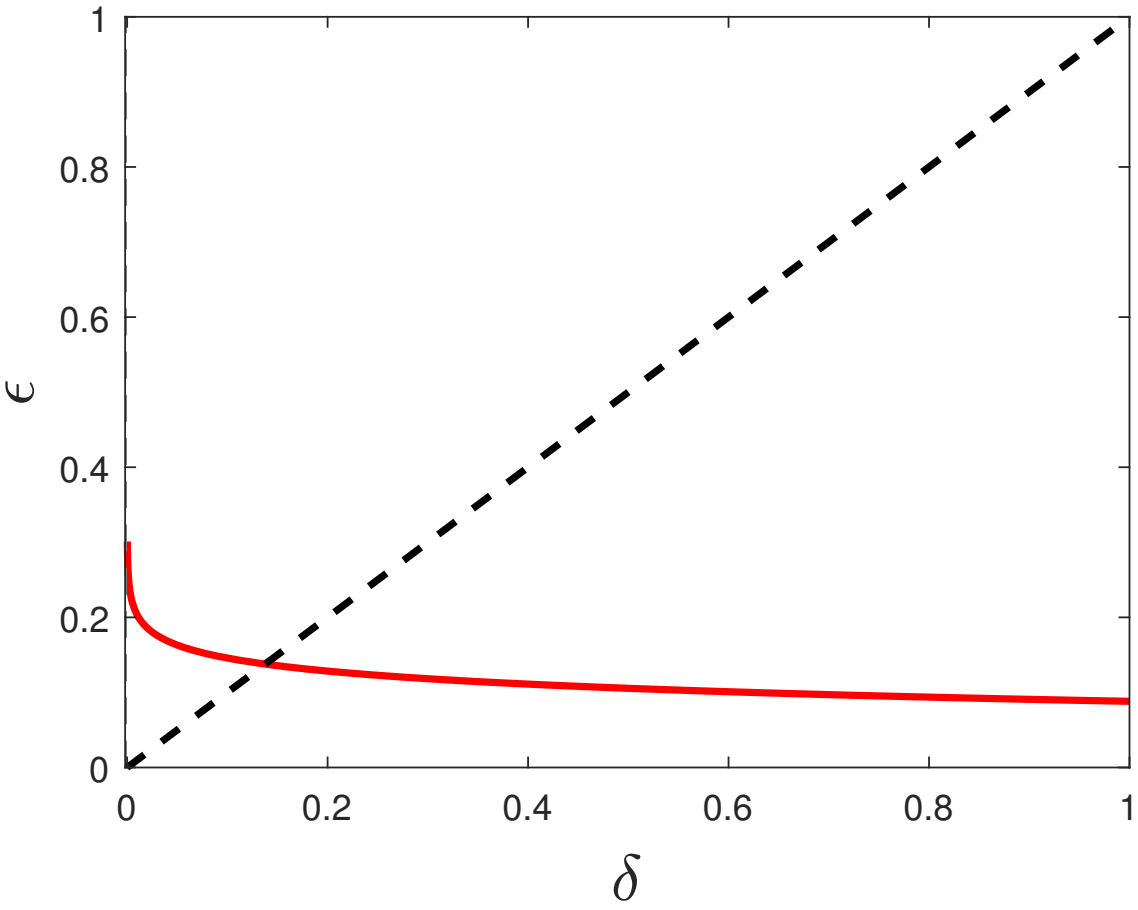

Figure 5: 'Iso-draw' curve for $M = 1,360$ and $d = 17$.

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 $p(S|y^T) \geqslant 1-\alpha$. We construct the HPD credible set as in Inoue and Kilian (2013, 2018). 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 the highest posterior density.

Figure 6 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 5 from the previous subsection corresponds to the iso-draw curve for this application.

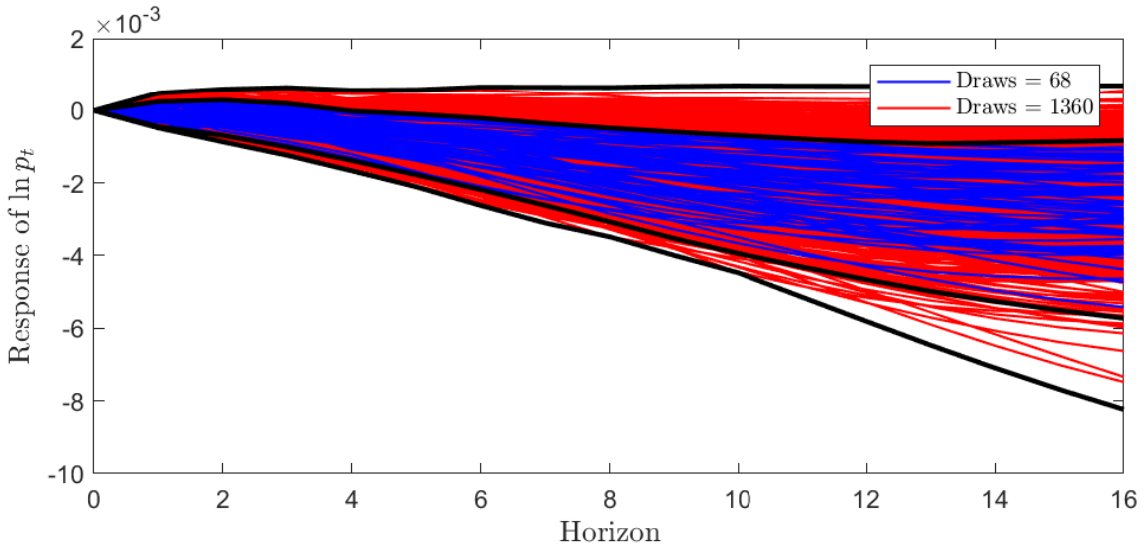

Figure 6: 68% highest posterior density credible set for the dynamic response of $\ln p_t$ to a monetary policy shock. BLUE and RED lines represent 68 and 1,360 draws from inside S, respectively. BLACK lines represent the minimum and maximum (pointwise) at each horizon.

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—and 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.

Second, we restrict that 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 simply keeps track of the largest and minimum value of the parameter of interest in each of its dimensions. We show that learning from the inside requires at least $(1-\epsilon)/\epsilon \ln(1/\delta)$ draws and at most $(2d/\epsilon)\ln(2d/\delta)$ 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.

A Appendix A

A.1 VC dimension

Given a nonempty class $\Lambda \in 2^{\mathbb{R}^d}$ 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.[14]

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 $l(x)$. 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

[14] 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)).

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 $\mathcal{P}_\lambda(\mathcal{P}_\Theta) \subseteq \mathcal{P}(\mathbb{R}^d)$, 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. □

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:

$$\lambda(\theta_m) \in A, \ \ \forall m = 1, \ldots, M \implies \mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P) \geqslant P(\lambda(\theta) \in A). \tag{7}$$

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

$$\begin{aligned} P_\eta\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_\eta) \geqslant \eta\right) &\geqslant P_\eta\left(\mathcal{L}(\widehat{\lambda}_M; [\boldsymbol{\lambda}], P_\eta) \geqslant P_\eta(\lambda(\theta) \in A)\right) \\ &\quad \text{(as } P_\eta(\lambda(\theta) \in A) \geqslant \eta\text{)}, \\ &\quad \geqslant P_\eta(\lambda(\theta_m) \in A \quad \forall m = 1, \ldots, M) \\ &\quad \text{(by (7))}, \\ &\quad \geqslant \eta^M. \end{aligned}$$

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

To prove 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:

$$r \equiv \bigtimes_{j=1}^{d} [\underline{r}_j, \overline{r}_j], \tag{8}$$

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-dimensional hyperrectangle r in the form of (8) such that $P(r) > \epsilon$, let*

$$\overline{h}_j \equiv \inf\{\, h' \in [\underline{r}_j, \overline{r}_j] \mid P(r_{-j}([\underline{r}_j, h'])) \geqslant \epsilon\}. \tag{9}$$

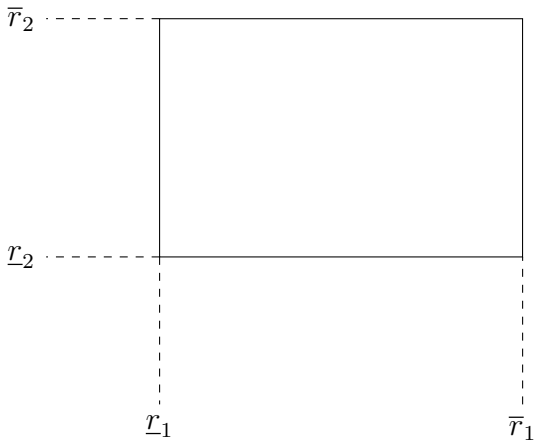

Figure 7: Hyperrectangle r when $d = 2$.

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 (9) 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 Theorem 3. We will prove the upper and lower bounds separately.

Proof of upper bound

Proof. The target concept is $[\lambda(S)]$; which we have defined as the smallest hyperrectangle 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

$$\begin{aligned}
\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) &= P(\mathbf{1}\{\lambda(\theta) \in [\widehat{\lambda}_M]\} \neq \mathbf{1}\{\lambda(\theta) \in [\lambda(S)]\}) \\
&= P\left(\lambda(\theta) \in [\widehat{\lambda}_M] \quad \text{and} \quad \lambda(\theta) \notin [\lambda(S)]\right) \\
&+ P\left(\lambda(\theta) \notin [\widehat{\lambda}_M] \quad \text{and} \quad \lambda(\theta) \in [\lambda(S)]\right). \qquad (10)
\end{aligned}$$

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 (10) is 0 and:

$$\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) = P(\lambda(\theta) \in [\lambda(S)] \backslash [\widehat{\lambda}_M]). \qquad (11)$$

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 $r_j := r_{-j}([\underline{h}_j, \overline{r}_j])$. Figure 8 displays the construction of our special hyperrectangles in the case when $d = 2$.

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:

$$P\left(\bigcup_{j=1}^{2d} \mathring{r}_j\right) \leqslant \sum_{j=1}^{2d} P(\mathring{r}_j) \leqslant \sum_{j=1}^{2d} \epsilon/2d \leqslant \epsilon, \tag{12}$$

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

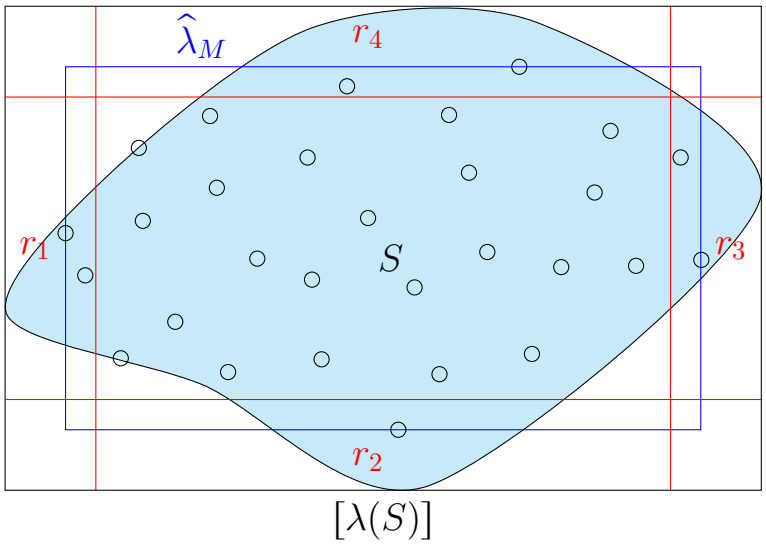

Figure 8: $d = 2$. $[\lambda(S)]$ (BLACK) is the true smallest rectangle that contains S (CYAN). $[\widehat{\lambda}_M]$ (BLUE) is the smallest rectangle that contains all positive labels (BLACK, CIRCLES). 4 'special' rectangles r_1, r_2, r_3, r_4 (RED) parallel to each side of $[\lambda(S)]$. Note that we have $[\widehat{\lambda}_M] \cap r_j \neq \varnothing$, $\forall j = 1, 2, 3, 4$.

$\lambda \in [\lambda(S)] \backslash [\widehat{\lambda}_M]$. Since $[\widehat{\lambda}_M]$ is a rectangle, we can write it as:

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

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

From (12) and (11):

$$(\theta_1, \theta_2, \ldots, \theta_M) \in \bigcap_{j=1}^{2d} E_j \implies \mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) \leqslant \epsilon. \tag{13}$$

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:

$$\lambda(\theta_{m(j)}) \in r_j \quad \text{and} \quad \theta_{m(j)} \in S. \tag{14}$$

This means that E_j^c happens if there is no M such that (14) happens. Note that:

$$\begin{aligned}
P\left(\mathcal{L}([\widehat{\lambda}_M]; [\lambda(S)], P) > \epsilon\right) &\leqslant P\left((\theta_1, \theta_2, \ldots, \theta_M) \in \bigcup_{j=1}^{2d} E_j^c\right) \\
&\text{(by (13))}, \\
&\leqslant \sum_{j=1}^{2d} P([\widehat{\lambda}_M] \cap r_j = \varnothing) \\
&\text{(by Boole's inequality)}, \\
&\leqslant \sum_{j=1}^{2d} P(\nexists\, \theta_m \text{ s.t. both } \lambda(\theta_m) \in r_j \text{ and } \theta_m \in S) \\
&\text{(by definition of } [\widehat{\lambda}_M] \text{, as explained in (14))}, \\
&= \sum_{j=1}^{2d} P(\forall\, \theta_m \text{ either } (\lambda(\theta_m) \notin r_j) \text{ or } (\theta_m \notin S)) \\
&= \sum_{j=1}^{2d} P(\lambda(\theta_m) \notin r_j \text{ or } \theta_m \notin S)^M \\
&\text{(as } \theta_m \text{ are i.i.d.)}, \\
&\leqslant \sum_{j=1}^{2d} \left(P(\lambda(\theta_m) \notin r_j) + P(\theta_m \notin S)\right)^M \\
&\text{(by Boole's inequality)}, \\
&= \sum_{j=1}^{2d} P((\lambda(\theta_m) \notin r_j)^m \\
&\text{(as } P(S) = 1), \\
&\leqslant 2d(1 - \epsilon/2d)^M \\
&\text{(as } P(r_j) \geqslant \epsilon/2d), \\
&\leqslant 2d \exp\left(\frac{-M\epsilon}{2d}\right) \\
&\text{(as } 1 - x \leqslant \exp(-x) \text{ for all } x \in \mathbb{R}).
\end{aligned}$$

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 $M \geqslant \frac{2d}{\epsilon} \ln\left(\frac{2d}{\delta}\right)$. □

Proof of lower bound

Proof. 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.

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

□

B Appendix B

Comparison with Gafarov, Meier and Montiel Olea (2018)

In this appendix we compare our bands from Section 3.1 with the analytical bands from Gafarov, Meier and Montiel Olea (2018). In a set-identified model with restrictions in only one shock, the tightest bands that contains the identified set are the solution to a nonlinear programming problem. We compare the results of our algorithm with these analytical bands. Figure 9 plots both the analytical bands and the bands generated by 100 draws (chosen ad-hoc) and $1,982$ draws, which are sufficient to learn when $\epsilon = \delta = 0.1$ and $d = 17$.

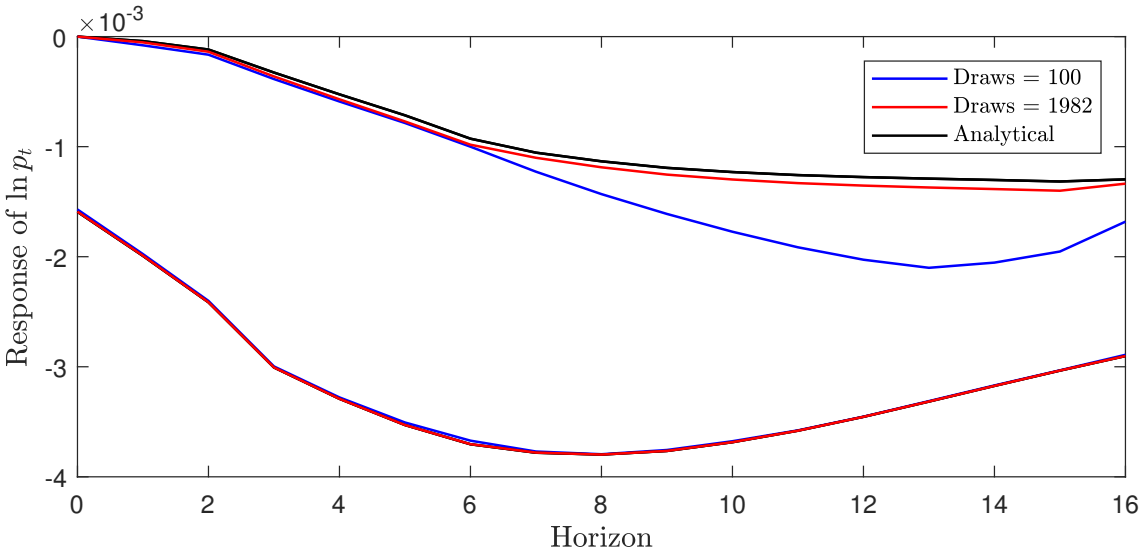

Figure 9: Analytical bands of Gafarov, Meier and Montiel Olea (2018) are plotted in BLACK. Bands generated using $[\widehat{\lambda_M}]$ using 100 draws (BLUE) and $1,982$ (RED) that satisfy the sign restrictions. $1,982$ draws is sufficient to support $\epsilon = \delta = 0.1$, with $d = 17$.

C Appendix C

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 $\delta < 1$:

$$P\left(\mathcal{L}(\widehat{\lambda}_M; \lambda, Q) < \epsilon\right) \geqslant 1 - \delta, \tag{15}$$

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 (15) 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.[15]

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

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

[15] 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.

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,[16] consider algorithms $\widehat{\lambda}_M : (x_1, \ldots, x_m) \to \mathbb{R}$ such that for any set $(a, b) \subset \mathbb{R}$,

$$\widehat{\lambda}_M^{-1}(a, b) \neq \varnothing. \tag{16}$$

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 (16) 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

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

[16] 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:

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

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 $P = Q$).

References

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

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*.

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.

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.

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

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.

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

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. 2018. "Corrigendum to 'Inference on Impulse Response Functions in Structural VAR Models,' *Journal of Econometrics*, 177 (2013), 1–13." *Journal of Econometrics (Forthcoming)*.

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. 2018. "Simultaneous Confidence Bands: Theory, Implementation, and an Application to SVARs." *Journal of Applied Econometrics (Forthcoming)*.

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.

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.