RICE UNIVERSITY

Active Learning and Adaptive Sampling for Non-Parametric Inference

by

Rui M. Castro

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

Approved, Thesis Committee:

Richard G. Baraniuk, Chair,
Victor E. Cameron Professor of
Electrical and Computer Engineering

Michael T. Orchard
Professor of Electrical and
Computer Engineering

Robert D. Nowak,
McFarland-Bascom Professor of Electrical
and Computer Engineering, University of
Wisconsin at Madison

Dennis D. Cox
Professor of Statistics

Houston, Texas

AUGUST 2007

To my family

Abstract

Active Learning and Adaptive Sampling

for Non-Parametric Inference

by

Rui M. Castro

This thesis presents a general discussion of active learning and adaptive sampling. In many practical scenarios it is possible to use information gleaned from previous observations to focus the sampling process, in the spirit of the "twenty-questions" game. As more samples are collected one can learn how to improve the sampling process by deciding *where* to sample next, for example. These sampling feedback techniques are generically known as active learning or adaptive sampling. Although appealing, analysis of such methodologies is difficult, since there are strong dependencies between the observed data. This is especially important in the presence of measurement uncertainty or noise. The main thrust of this thesis is to characterize the potential and fundamental limitations of active learning, particularly in non-parametric settings.

First, we consider the probabilistic classification setting. Using minimax analysis techniques we investigate the achievable rates of classification error convergence for broad classes of distributions characterized by decision boundary regularity and noise conditions (which describe the observation noise near the decision boundary). The results clearly indicate the conditions under which one can expect significant gains through active learning. Furthermore we show that the learning rates derived are tight for "boundary fragment" classes in d-dimensional feature spaces when the feature

marginal density is bounded from above and below.

Second we study the problem of estimating an unknown function from noisy pointwise samples, where the sample locations are adaptively chosen based on previous samples and observations, as described above. We present results characterizing the potential and fundamental limits of active learning for certain classes of nonparametric regression problems, and also present practical algorithms capable of exploiting the sampling adaptivity and provably improving upon non-adaptive techniques. Our active sampling procedure is based on a novel coarse-to-fine strategy, based on and motivated by the success of spatially-adaptive methods such as wavelet analysis in nonparametric function estimation.

Using the ideas developed when solving the function regression problem we present a greedy algorithm for estimating piecewise constant functions with smooth boundaries that is near minimax optimal but is computationally much more efficient than the best dictionary based method (in this case wedgelet approximations). Finally we compare adaptive sampling (where feedback guiding the sampling process is present) with non-adaptive compressive sampling (where non-traditional projection samples are used). It is shown that under mild noise compressive sampling can be competitive with adaptive sampling, but adaptive sampling significantly outperforms compressive sampling in lower signal-to-noise conditions. Furthermore this work also helps the understanding of the different behavior of compressive sampling under noisy and noiseless settings.

Acknowledgements

I offer my gratitude to all who have inspired and challenged me during my graduate school journey. I am deeply grateful to my advisor, Professor Robert Nowak. His contagious enthusiasm and optimism, deep insights (regarding both professional and personal issues), and outstanding dedication to research and teaching have been a great source of inspiration. Thank you Rob for being such great friend and colleague.

I would like to extend my gratitude to Professors Richard Baraniuk, Dennis Cox, Peter Olofsson, Michael Orchard and Richard Tapia. They taught me how to be a better researcher, stimulated my curiosity and have been fantastic role models. A very special thanks to Professors José Leitão and Mário Figueiredo for introducing me to the research world and the field of statistical signal processing. Their continuing support has been very important to me.

I would like to thank many friends and colleagues with whom I shared my path through graduate school. I cannot list everyone of them (my memory does not allow me) but would like to deeply thank José Costa, Michael Lexa, Gabriel Lopes, Jarvis Haupt, Mike Rabbat, Dana Redford, Rajesh Rengarajan, Doreen Rosenstrauch, Clayton Scott, Sinan Sinanovic, Aarti Singh, Paulo Tabuada, Yolanda Tsang, Mike Wakin, Rebecca Willett and Ercan Yıldız.

I would like to issue a very special acknowledgement to my family – my mom, sister and grandfather. Their nurtured my curiosity from an early age, and their unconditional support allowed me to overcome many obstacles of graduate school,

including the Atlantic ocean.

Finally, a very special thanks to Denise – you have been a great companion and friend, and I'm looking forward to enjoy your company for all the journeys ahead.

Contents

1 Introduction **1**
- 1.1 Motivation . . . 3
- 1.2 Thesis Contribution . . . 11
- 1.3 Organization . . . 13

2 Active Learning for Classification **15**
- 2.1 Introduction . . . 15
- 2.2 Framework Description . . . 19
- 2.3 One-Dimensional Threshold Classifiers ($d = 1$) . . . 24
  - 2.3.1 Bounded noise rate ($\kappa = 1$) . . . 28
  - 2.3.2 Unbounded rate noise: $\kappa > 1$ . . . 33
- 2.4 Boundary Fragments ($d > 1$) . . . 39
  - 2.4.1 Minimax Lower Bounds ($d > 1$) . . . 42
  - 2.4.2 Upper Bounds ($d > 1$) . . . 43
- 2.5 Final Remarks and Discussion . . . 47
- 2.6 Proofs . . . 51
  - 2.6.1 Proof of Theorem 1 . . . 51
  - 2.6.2 Proof of Theorem 3 . . . 57
  - 2.6.3 Proof of Theorem 4 . . . 63
  - 2.6.4 Proof of Lemma 3 . . . 67

3 Regression of Piecewise Constant Functions **69**
- 3.1 Introduction . . . 70
- 3.2 Problem Formulation . . . 72
- 3.3 Fundamental Limits - Minimax Lower Bounds . . . 77
  - 3.3.1 Hölder Smooth Functions . . . 78
  - 3.3.2 Piecewise Constant Functions . . . 79
- 3.4 Estimation of Piecewise Constant Functions . . . 83
  - 3.4.1 Passive Learning Algorithm for PC(β, M) . . . 85
  - 3.4.2 Active Learning Algorithms for PC(β, M) . . . 91
- 3.5 Final Remarks and Open Questions . . . 105
- 3.6 Proofs . . . 106
  - 3.6.1 Proof of Theorem 7 . . . 106
  - 3.6.2 Proof of Theorem 8 . . . 112
  - 3.6.3 Proof of Corollary 1 . . . 117
  - 3.6.4 Proof of Theorem 9 . . . 118
  - 3.6.5 Sketch Proof of Theorem 10 . . . 126

3.6.6 Proof of Theorem 11 . . . 129
3.6.7 Sketch Proof of Theorem 12 . . . 141

4 Coarse-to-Fine Learning of Piecewise Constant Functions 144
4.1 Introduction . . . 145
4.2 Problem Formulation . . . 147
4.3 Maximum Penalized Likelihood Estimators . . . 149
4.4 Coarse-to-Fine Estimation . . . 152
4.4.1 Error Analysis . . . 153
4.4.2 Computational Complexity . . . 157
4.5 Simulations . . . 158
4.6 Final Remarks . . . 160

5 Compressive versus Active Sampling 162
5.1 Introduction . . . 163
5.2 Compressive and Active Sampling . . . 165
5.3 Signal Reconstruction Algorithm . . . 170
5.4 Proof of Theorem 14 . . . 172
5.5 Proof of Theorem 15 . . . 174
5.6 Experiments . . . 175
5.7 Final Remarks . . . 178

6 Concluding Remarks 180

A The Burnashev-Zigangirov Algorithm 190

List of Figures

1.1 The role of eye movements in scene perception: (a) example of a natural scene; (b) tracked eye movement during the first seconds of scene perception. (source: MSU Vision Cognition Laboratory - `http://eyelab.msu.edu/visualcognition/`). 2
1.2 Airborne range sensor surveying a terrain. 10
1.3 Illustration of the power of active/adaptive sampling: (a) Original image, with range $[0, 1]$. (b) Estimate through passive learning, using $2^{14} = 128^2$ noisy samples (with noise standard deviation 0.02). (c) Estimate through active learning using also 2^{14} noisy samples (using the algorithm presented in Chapter 3). (d) Sample locations used in the active learning procedure. 11

2.1 Examples of two conditional distributions $\eta(x) = \Pr(Y = 1|X = x)$. (a) In this case $\eta(\cdot)$ satisfies the margin condition with $\kappa = 1$; (b) Here the margin condition is satisfied for $\kappa = 2$. 26
2.2 (a) Example of the conditional distribution $\eta(\cdot)$ of an element of the class $\mathrm{BF}(\alpha, \kappa, L, C, c)$ when $d = 2$ and $\alpha = 2$. (b) The corresponding Bayes classifier. 41
2.3 Illustration of the active classification procedure for boundary fragments when $d = 2$. In this case $\alpha = 2$ therefore we estimate the true Bayes decision boundary with the aid of piecewise linear polynomials. The crosses represent the estimates of g^* obtained by each one of the $M+1$ line searches. The dark solid line segments represent the $M/\lfloor\alpha\rfloor$ interpolation polynomials. 45
2.4 The two conditional distributions used for the proof of Theorem 1. . . 52

3.1 Examples of functions in the classes considered: (a) Hölder smooth function. (b) Piecewise constant function. 71
3.2 Example of Recursive Dyadic Partitions, and the corresponding tree representations. 88
3.3 The two step procedure for $d = 2$ (no shifted partitions): (a) Preview step RDP. Note that the cell with the arrow was pruned shallower than depth J, but it contains a part of the boundary. (b) Additional sampling for the refinement step. (c) Refinement step. 92

3.4 Illustration of the shifted RDP construction for $d = 2$: (a) RDP used in $\widehat{f}_0^p$. The highlighted cell intersects the boundary but it was pruned, since the pruning does not incur in severe error. (b) Shifted RDP, used in $\widehat{f}_1^p$. In this case the problem region is detected, since it would otherwise cause a large error. (c) These are the cells that are going to be refined in the refinement step. 95
3.5 Illustration of the detectability condition. Figure (a) depicts a small region of the boundary, and we are in particular interested on the small region A marked with a square. Figures (b), (c) and (d) depict the cells at depth $J - 1$ in containing A in the various shifted partitions. The cell in (d) is able to "feel" region A, since there is a significant volume of both constant levels in that cell. 101
3.6 Prefix encoding of a Recursive Dyadic Partition. The depicted partition encodes as 100100000 in binary. 120
3.7 Example of RDP tree pruning, for $d = 2$, $J = 4$, and $J' = 2$. The depicted curve is $B(f)$: (a) partition with all leafs at depth J; (b) pruned partition adapted to $B(f)$. 122

4.1 Example boundary estimation problem. (a) Initial RDP used in traditional piecewise linear methods. (b) Initial coarse-resolution RDP used in the preview step. (c) Partition generated in a preview step. Note the example of a Case 3 error. (d) Final partition generated during the refinement step, using shifted partitions. 148
4.2 Shepp-Logan phantom. (a) 256×256 noisy measurements, $\sigma^2 = 0.001$, MSE = 0.0115. (b) preview partition. (c) Shepp-Logan phantom estimate formed by fitting one wedgelet or constant to each of the unpruned squares from the preview step; MSE = 0.000504. (d) Shepp-Logan phantom estimate using standard wedgelet, MSE = 0.00163. . 159

5.1 Example of an 1024×1024 pixel image from the boundary fragment class (image (a)); and a 32×32 pixel noisy image with $\sigma = 0.15$ (image (b)). 166
5.2 Reconstructions of image in Fig. 5.1(a) based on k noisy samples with $\sigma = 0.15$. 176

6.1 Adaptive sampling using ballistic laser imaging: (a) Estimate obtained scanning a turbid medium (with a small circular target) on a regular 128×128 grid (16384 passive samples). (b) Estimate through passive learning, using 32×32 samples (1024 passive samples). (c) Estimate through active learning using 984 samples. (d) Sample locations used in the active learning procedure (see [1] for details). 182

A.1 The Burnashev-Zigangirov (BZ) algorithm. 191

Chapter 1

Introduction

Nenhuma ideia brilhante consegue entrar em circulação

se não agregando a si qualquer elemento de estupidez

No intelligent idea can enter into circulation unless

it's accompanied by an element of stupidity

(in Livro do Desassossego by Bernardo Soares [1]*)*

Sensing and learning the surrounding world is a task anyone is very familiar with, and it is essentially the goal of any learning system. When performing such a task one typically uses some sort of feedback mechanism to guide the sensing process. For example, when looking at a landscape our attention does not usually linger in the clear sky, but instead we look briefly at it and then start focusing on "interesting" features, like people in the scene. However, if there is a bird overhead then one will probably notice that, and maybe redirect attention to it. This sort of adaptive process is illustrated in Figure 1.1 which depicts the way a human visually samples a

[1] *Bernardo Soares is a semi-heteronym of Fernando Pessoa (born in June 13, 1888 in Lisbon, Portugal died in November 30, 1935 in the same city). The* Livro do Desassossego *(Book of Disquietude) is composed of various fragments, assembled after the death of the author.*

(a) (b)

Figure 1.1: The role of eye movements in scene perception: (a) example of a natural scene; (b) tracked eye movement during the first seconds of scene perception. (source: MSU Vision Cognition Laboratory - `http://eyelab.msu.edu/visualcognition/`).

natural scene. Clearly there is some feedback between the learning and the sensing process: one starts by having a rough idea of the entire scene, and then focuses attention on particularly "interesting" details [2, 3]. Although such learning-with-feedback processes are common in nature, most man-made automatic sensing devices and learning strategies do not take advantage of feedback in the sensing process. This is the case with digital cameras, that have a pixel array that *a priori* assigns the same importance to every area in the image. This lack of flexibility is even more critical when considering imaging processes where the collection of all information can be excessively time consuming, costly, or even completely infeasible. For example when making topographic maps one needs to take fewer measurements in areas that are relatively flat (like plains), and many more measurements in areas where significant variations, like cliffs or river beds, exist.

The focus of this thesis is a general framework for the development and analysis

of active learning processes and adaptive sampling, especially under the presence of measurement uncertainty (noise or other kinds of measurement distortions). The key issue is to know *where*, *when* and/or *how* to collect information, and make those decisions in an adaptive or *active* way. We will refer to such processes as *Active Learning* procedures. With this term we encompass many different frameworks that allow for significant flexibility in the information collection process, closing the loop between sensing and learning/processing. Although the general ideas described above are not new a general theoretical framework and methodology does not yet exist. The main reason for this lack in knowledge is the difficulty of the rigorous analysis of such methods, since the feedback mechanism induces strong dependencies between the samples, and therefore conventional statistical tools based on standard laws of large numbers and central limit theorems are no longer applicable.

1.1 Motivation

Next we present some applications that motivate our contribution. Although the list is not all-inclusive, it nevertheless shows the importance and relevance of active learning in a variety of areas.

Active Learning for Classification: The general goal of supervised learning is to estimate a function that maps features to class labels, given a set of labeled training examples. A prototypical example is document classification. Suppose we are given a text document and want to associate a topic/label to it (*e.g.*, finance, sports, nuclear

physics, information theory). Our goal is to devise an algorithm that learns how to perform this task from examples. More precisely, we have access to a number of documents that have been inspected and labeled by an expert (most likely a human), and the learning algorithm uses these instances to construct a general labeling rule for documents. In today's world of electronic media, we have a virtually infinite supply of documents at our fingertips. However, labeling documents for training purposes is expensive and time consuming and ideally we would like our algorithm to learn to correctly label documents based on a modest number of labeled examples. To accomplish this, we would like the algorithm to automatically select unlabeled documents for which it has difficulty labeling itself or whose label is potentially very informative, and then request the correct labels for these documents from an expert (human). The hope is that while the algorithm learns, it makes fewer and fewer requests for labels and in this way the total number of labeled documents required for the learning task may be much smaller than needed if an arbitrary set of labeled documents were used instead. This prototypical example clarifies the sharp increase of interest on active learning in the machine learning community in the last few of years. One can conjecture that this is due to the fact that data sets have become massive and that the cost of labeling all the examples in such data sets is prohibitive or impossible. Although extremely appealing, existing practical active learning methods have many problems that prevent their use. Namely they tend to perform very well when only a very reduced number of labeled examples are provided, but as soon as

that number increases their performance degrades significantly, often becoming worse than passive methods [4]. This is an indication that these learners are flawed, and are often "side-tracked" by the first set of examples.

Various learning paradigms can be considered. One such paradigm is called *query learning*. The learner queries the expert with examples (possibly synthetic) whose a label could be very beneficial. A related active learning paradigm is *pool-based active learning*. Under this framework the learner has access to a large pool of unlabeled examples, and can request labels from any examples in this pool. If the unlabeled data set is so large that we can virtually "pick" any possible example for labeling the two paradigms are essentially the same. We will focus on these two paradigms on this thesis.

The two active learning paradigms described are quite appealing, and very useful to gain a good understanding about the limitations of active learning. Nevertheless, they present several drawbacks in real-world classification settings, namely the synthetic examples might be very unlike the real-world examples (*e.g.*, if one is doing hand-written digit classification then asking an expert what digit corresponds to a checkerboard pattern might not yield any meaningful answer). A paradigm that is more relevant in such settings is called *selective sampling*. In this framework the learner observes unlabeled examples sequentially, and can request a label if it wishes. Therefore if confronted with a particularly challenging and potentially informative example the learner may request the label. On the other hand if the unlabeled ex-

ample presented can be easily and accurately classified based on previously collected information then requesting a label might not be very beneficial. The online learning paradigm is better suited for the analysis of learning in this setting [5], as opposed to statistical learning. We discuss briefly our progress towards an understanding of the fundamental limitations of active learning in such a scenario in the final chapter.

Networking and Sensor Networks: The term *sensor network* can be used in a very broad sense, encompassing many networked sensing structures, including wireless sensor networks and the Internet itself. In general a sensor network consists of distributed sensing devices that are scattered throughout an environment and that can communicate among themselves through a channel.

A prototypical wireless sensor is composed of small devices, with a number of sensors able to measure some environmental features (*e.g.*, temperature, sound, light intensity). These devices have some data processing and communication capability, and are generally powered by batteries or some low-energy source (*e.g.*, solar power). Due to limited energy resources careful management of these units is extremely important in order to optimize the usage of the available assets so that the entire network is feasible and sustainable. It is understood that communications are extremely energy consuming therefore one should only transmit information if it is valuable, in order to save power. Perhaps the most basic approach to energy conservation is simply to limit the number of samples acquired and communicated to a bare minimum. While this may seem rather simple-minded, it turns out that sensor networks allow one

considerable flexibility in what, where and when samples are taken. Feedback during the sampling process can be used in a sequential and adaptive way, in which a fusion center selectively queries nodes in order to rapidly locate important features (*e.g.*, the boundary of a fire, or a moving vehicle). These kinds of ideas have already exploited in wireless sensor networks [6], where significant energy savings can be attained when sensing a piecewise smooth field (modeling for example an oil spill, or a terrain with varying characteristics). As mentioned above, the Internet can also be viewed as a sensor network, since most nodes in the internet are endowed with sensing capabilities, and are able to return information such as queueing delay and packet loss, a list of neighboring nodes, *et cetera* [7]. Limitations on bandwidth, latency and data storage all motivate the use of active learning methods in Internet studies.

Another very compelling reason for active learning procedures is the expense of human-assisted data analysis, as illustrated in the following context: a very important asset when managing computer networks is the ability to detect anomalies/attacks to the network. The most effective attack detection tools employ classifiers that are trained using *post-mortem* examples of anomalies/attacks and normal behavior. The process of labeling examples requires a human expert, and is a very time consuming and expensive task. Therefore one wishes to label only examples that are truly informative, and that can improve the classifier performance given all that was learned up to that point. To do so clearly requires some active learning procedure that automatically decides whether or not a label is needed for a new example, thereby

reducing the number of examples a human expert must inspect.

Active Sampling in Imaging: Consider the following classical regression problem. The goal is to accurately estimate a function from a finite set of noisy point samples of the function. This is the idealization of many imaging problems, for example, surveying a region of interest using a laser range finder (Figure 1.2). This kind of sensor is able to measure range (distance between sensor and observed object) and maybe some properties of the object (*e.g.*, the type of terrain: vegetation, sand, rock, *et cetera*). Suppose we want to use such a setup to construct a topographic map of a region of interest. In general we assume that the underlying function/field is in some large function class. If one has to decide where to collect our samples, before making any observations, then the only reasonable thing to do it to scatter those samples in a uniform way across the function domain, where uniform means the samples are equally concentrated in every region of the domain. Now if we have the possibility of deciding where to sample "on-the-fly", based on information already collected, then our policy might be different. Consider the case where the terrain is relatively flat, except for a ledge. Clearly, to estimate the flat regions a low resolution sampling would suffice, but to accurately locate the ledge area, a higher resolution sampling is needed. If we are pursuing a passive/non-adaptive sampling approach then we need to scan the entire field at the highest possible resolution, otherwise the ledge area might be inaccurately estimated. On the other hand, if an active/adaptive sampling technique is used then we can adapt our resolution to the field (based on

our observations), thereby focusing the sampling procedure in the ledge area.

To illustrate the potential of active sampling consider Figure 1.3, where the algorithm developed in Chapter 3 is used to guide the sampling process. We have a budget of $2^{14} = 16384$ noisy point samples of the scene in Figure 1.3(a). For the result in Figure 1.3(b) we proceeded in a passive way, collecting the samples uniformly at random over the image domain. In Figure 1.3(c) we proceeded in an active way, collecting more samples in regions where these seemed more beneficial (Figure 1.3(d) depicts the samples location). It is visually clear that the active learning procedure improves the overall quality of the estimate: edges are sharper, and further details can be observed in the reconstruction, for example the rightmost leg of the tripod is now fully perceivable. Furthermore the algorithm used is only optimized for learning piecewise constant functions, completely different than the function depicted in Figure 1.3(a). Much better results might be possible with other algorithms.

Greedy and Coarse-to-fine Computation: The problems described above are concerned with decisions about what or where to sample. In those cases, collecting/transmitting information is the critical asset. A different, albeit closely related, class of problems can be considered wherein the critical asset is computation instead. For that class of problems the collection of information is passive, and essentially "free", but processing the information comes at a cost, either because the data sets are massive, or because the processing task is very complex. Under certain scenarios one wants to concentrate the processing power on important tasks, where the benefit

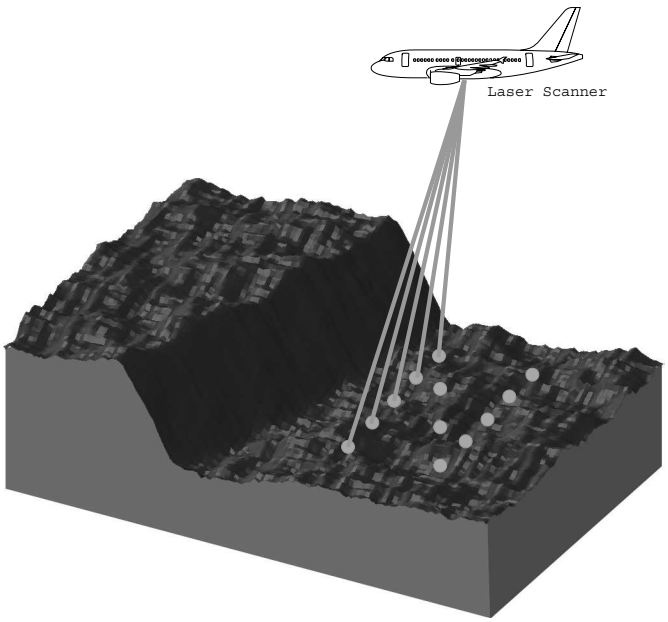
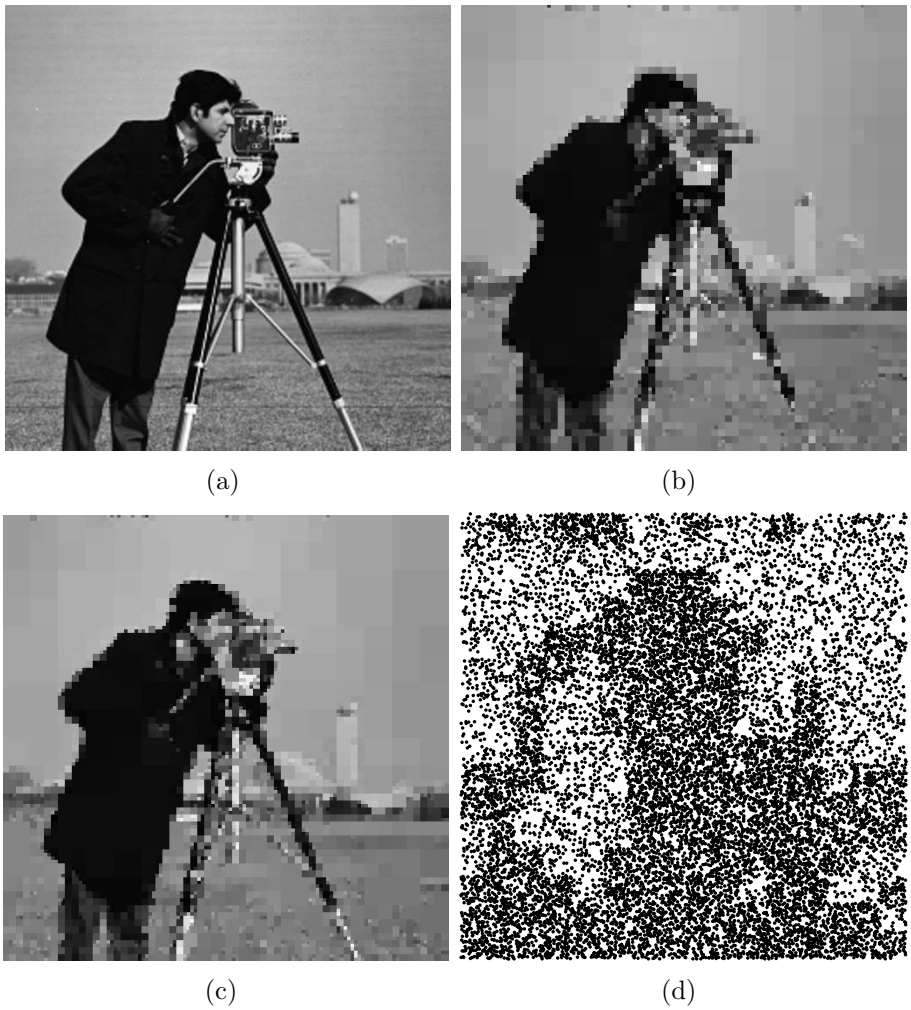


Figure 1.2: Airborne range sensor surveying a terrain.

overcomes the computational cost. Let us consider a concrete example. Suppose we want to estimate a piecewise constant function with smooth boundaries. Accurate detection and localization of the boundary (a manifold) is the key aspect of this problem. In general, algorithms capable of achieving optimal performance require exhaustive searches over large dictionaries that grow exponentially with the dimension of the observation domain. The computational burden of the search often hinders the use of such techniques in practice, and motivates a sequential, coarse-to-fine approach that involves first examining the data on a coarse grid, and then refining the analysis and approximation in regions of interest. When performed wisely these greedy processes allow for a very significant reduction in computation time, without degrading the overall performance.



(a) (b)

(c) (d)

Figure 1.3: Illustration of the power of active/adaptive sampling: (a) Original image, with range $[0, 1]$. (b) Estimate through passive learning, using $2^{14} = 128^2$ noisy samples (with noise standard deviation 0.02). (c) Estimate through active learning using also 2^{14} noisy samples (using the algorithm presented in Chapter 3). (d) Sample locations used in the active learning procedure.

1.2 Thesis Contribution

The work presented in this thesis is focused on having a satisfactory answer to the general question "What are the fundamental limits of active learning, and how can one achieve them?". Surprisingly, such a characterization was lacking even for

very simple scenarios. Most of the work done up to this point considered cases where there is very small or no uncertainty (noise) in the sampling/labeling process. Those scenarios are of little relevance when addressing "real-world" problems, precluding the sound application of active learning methods. By answering the general question above for scenarios where sample uncertainty is present, one can design better active learning algorithms that will perform well in the realm of real-world applications.

The problems addressed in this thesis deal with the sequential adaptation of sampling and decision making in non-parametric statistical inference. In the non-parametric setting, there exists a very limited number of papers analyzing adaptive processes of this nature, and these will be discussed within the body of the dissertation. However, similar techniques have been extensively studied in the context of parametric inference. There is a body of literature known collectively as adaptive sampling (see [8] and references therein) dealing with inference problems where the sampling designs are adjusted during the experiments, based on the observations made, in the spirit of the methods investigated in this thesis. In contrast, however, this adaptive sampling literature addresses problems involving estimation or testing of scalar or parametric quantities (*e.g.*, population sizes), rather than non-parametric function or set estimation, which is the focus of this thesis. Also somewhat related is the field known as sequential analysis [9–11], which deals with the adaptive design of tests or inference procedures based on the outcomes of previous inferences. In these approaches there is no control over the sampling process (*e.g.*, the samples might be

drawn independently from some distribution), only the testing/inference process, and thus they differ significantly from adaptive sampling.

1.3 Organization

The thesis is comprised of four main chapters. Each chapter begins with a detailed summary of the material presented. Although the various chapters are intertwined, they are relatively self-contained in terms of presentation, and can be easily read out of order. The first two chapters are the main component of the thesis.

Chapter 2 is mostly theoretical in nature, and illustrates what are the fundamental limits of active learning in a classification setting, under a minimax framework. Although some active learning algorithms are provided these are mostly of used to prove performance guarantees, and not practical in realistic settings.

In Chapter 3 we develop methods for active learning in non-parametric regression. Besides presenting a solid theoretical framework characterizing the fundamental limits of active learning in these settings, a practical active learning algorithm, with nearly optimal performance, is also constructed.

Chapters 4 and 5 are more modest, and somewhat "lighter". Chapter 4 discusses a twist on the active learning framework, where instead of guiding the sampling process one guides the computational effort. That is, one learns "where" computation is more effective, essentially devising greedy methods. Using the ideas of Chapter 3 a computationally fast method for estimation and denoising of piecewise constant

functions with smooth boundaries is presented and analyzed. Chapter 5 compares adaptive sampling (where feedback guiding the sampling process is present) with non-adaptive compressive sampling (where non-traditional projection samples are used). It is shown that under mild noise compressive sampling can be competitive with adaptive sampling. Furthermore this work also helps the understanding of the different behavior of compressive sampling under noisy and noiseless settings.

Finally Chapter 6 offers some broad closing remarks, as well as speculation on future research directions and open problems.

Chapter 2

Active Learning for Classification

This chapter analyzes the potential advantages and theoretical challenges of "active learning" algorithms in a probabilistic classification setting. Using minimax analysis techniques, we study the achievable rates of classification error convergence for broad classes of distributions characterized by decision boundary regularity and noise conditions. The results clearly indicate the conditions under which one can expect significant gains through active learning. Furthermore we show that the learning rates derived are tight for "boundary fragment" classes in d-dimensional feature spaces when the feature marginal density is bounded from above and below.

2.1 Introduction

Most theory and methods in statistical machine learning focus on inference based on a sample of independent and identically distributed (i.i.d.) observations. We call this typical set-up *passive learning* since the learning algorithms themselves have no influence in the data collection process. As widespread as the passive learning model is, in certain situations it is possible to combine the data collection and learning

processes, using data collected in early stages to guide the selection of new samples. Learning strategies of this nature are called *active learning* procedures. Active learning can offer significant advantages over i.i.d. data collection. In this chapter we focus on classification problems, in which one observes features (*e.g.*, in the document classification scenario described in the introduction these are attributes extracted from documents such as the frequencies of keywords, etc.) and attempts to infer labels (from a finite set of possible labels,*e.g.*, document topics). Although extremely appealing, most practical active learning methods so far are plagued by many problems that prevent their application in realistic settings. In many settings they tend to perform very well when only a few labeled examples are provided, but as soon as that number increases their performance degrades significantly, often becoming worse than passive methods [4]. This is an indication that these learners are often "side-tracked" by the first few labeled examples. This behavior partially motivates the work presented here. By carefully characterizing the fundamental limits of active learning one hopes to be able to design sound practical algorithms not displaying the pitfalls of currently existing techniques.

Interest in active learning has increased greatly in recent years, in part due to the dramatic growth of data sets and the high cost of labeling examples in such sets. There are several empirical and theoretical results suggesting that in certain situations active learning can be significantly more efficient than passive learning [12–15]. Many of these results pertain to the "noiseless" setting, in which the labels are deterministic

functions of the features. In certain noiseless scenarios it has been shown that the number of labeled examples needed to achieve a desired classification error rate is much smaller than what would be needed using passive learning. In fact for some of those scenarios, active learning requires only $O(\log n)$ labeled examples to achieve the same performance that can be achieved through passive learning with n labeled examples [14, 16–18]. This exponential speed-up in learning rates is a tantalizing example of the power of active learning.

Although the noiseless setting is interesting from a theoretical perspective, it is very restrictive, and seldom relevant for practical applications. Some results have been obtained for active learning in the "bounded noise rate" setting. In this setting labels are no longer a deterministic function of the features, rather for a given feature the probability of one label is significantly higher than the probability of any other label. In the case of binary classification this means that if $(\boldsymbol{X}, Y)$ is a feature-label pair, where $Y \in \{0, 1\}$, then $|\Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x}) - 1/2| \geq c$ for all $\boldsymbol{x}$ in the feature space, with $c > 0$. In other words, $\Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x})$ "jumps" at the decision boundary, providing a strong cue that active learning algorithms can use. In fact, this cue is effectively as strong as in the noiseless case. Under the bounded noise rate assumption it can be shown that results similar to the ones for the noiseless scenario can be achieved [15, 19–21]. These results are intimately related to coding with noiseless feedback [22,23] and adaptive sampling techniques in regression analysis [20, 23–26], where related performance gains have been reported. Furthermore, the

active learning algorithm proposed in [19], in addition to providing improvements in certain bounded noise conditions, is shown to perform no worse than passive learning in general conditions.

The main contribution of this chapter is the extension of this results to the case in which the noise is unbounded. To this date this question has not been addressed in the literature and to our knowledge this is the first such characterization of active learning. In the case of binary classification "unbounded noise" means that $\Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x})$ is not bounded away from $1/2$. Notice that in this case there is no strong cue that active learning algorithms can follow, since the labels of features near the decision boundary are almost devoid of information (*i.e.*, $\Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x})$ approaches $1/2$). Since situations like this seem very likely to arise in practice (*e.g.*, simply due to feature measurement or precision errors if nothing else) it is important to identify the potential of active learning in such cases.

Our main result can be summarized as follows. Following Tsybakov's formulation of distributional classes [27], the complexity of classification problems can in many cases be characterized by two key parameters ρ and κ. The parameter $\rho = (d-1)/\alpha$, where d is the dimension of the feature space and α is the Hölder regularity of the Bayes decision boundary, is a measure of the complexity of the boundary. Parameter $\kappa \geq 1$ characterizes the level of "noise", that is, the behavior of $\Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x})$ in the vicinity of the boundary. The value $\kappa = 1$ corresponds to the noiseless or bounded noise situation and $\kappa > 1$ corresponds to unbounded noise conditions. Using this sort

of characterization, we derive lower and upper bounds for active learning performance. In particular, it is shown that the fastest rate of classification error decay (towards the error of the Bayes classifier) using active learning is $n^{-\frac{\kappa}{2\kappa+\rho-2}}$, where n is the number of labeled examples, whereas the fastest decay rate possible using passive learning is $n^{-\frac{\kappa}{2\kappa+\rho-1}}$. Note that the active learning error decay rate is always faster than that of passive learning. Tsybakov has shown that in certain cases, when ($\kappa \to 1$ and $\rho \to 0$) passive learning can achieve "fast" rates approaching n^{-1} (faster than the usual $n^{-1/2}$ rate). In contrast, our results show that in similar situations active learning can achieve much faster rates (in the limit decaying as fast as any negative power of n). Also note that the passive and active rates are essentially the same as $\kappa \to \infty$, which is the case in which $\Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x})$ is very flat near the boundary and consequently there is no cue that can effectively drive an active learning procedure. Furthermore, upper bounds show that the learning rates derived are tight for "boundary fragment" classes in d-dimensional feature spaces when the density of the marginal distribution $P_{\boldsymbol{X}}$ (over features) is bounded from above and below.

2.2 Framework Description

In this chapter we consider the framework close to that of statistical classification theory, described for example in [28]. The main difference pertains the collection of labeled examples, being more general than the classical framework, both under the *passive* and *active* sampling paradigms: instead of requiring labeled examples to be

samples of an unknown joint distribution of features and labels we suppose to learner can *query* the label corresponding to any feature vector. For conciseness we focus on binary classification and a particular kind of feature spaces.

Let $\mathcal{X} \triangleq [0,1]^d$ denote the *feature space* and $\mathcal{Y} \triangleq \{0,1\}$ denote the *label space*. Let $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ be a random vector, with *unknown* distribution $P_{\boldsymbol{X}Y}$. The goal in classification is to construct a "good" classification rule, that is, given a feature vector $\boldsymbol{X} \in \mathcal{X}$ we want to predict the label $Y \in \mathcal{Y}$ as accurately as possible, where the classification rule is a measurable function $f : \mathcal{X} \to \mathcal{Y}$. The performance of the classifier is evaluated in terms of the expected 0/1-loss. With this choice of loss function the risk is simply the probability of classification error,

$$R(f) \triangleq \mathbb{E}[\mathbf{1}\{f(\boldsymbol{X}) \neq Y\}] = \Pr(f(\boldsymbol{X}) \neq Y) \ ,$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function. Since we are considering only binary classification (two classes) there is a one-to-one correspondence between classifiers and sets: Any reasonable deterministic classifier is of the form $f(\boldsymbol{x}) = \mathbf{1}\{\boldsymbol{x} \in G\}$, where G is a measurable subset of $[0,1]^d$. We use the term classifier interchangeably for both f and G. Define the optimal risk as

$$R^* \triangleq \inf_{G \,\text{measurable}} R(G) \ .$$

A classifier attaining the minimal risk R^* is the *Bayes Classifier*

$$G^* \triangleq \{\boldsymbol{x} \in [0,1]^d : \eta(\boldsymbol{x}) \geq 1/2\} \ ,$$

where $\eta(\boldsymbol{x}) = \mathbb{E}[Y|\boldsymbol{X} = \boldsymbol{x}] = \Pr(Y = 1|\boldsymbol{X} = \boldsymbol{x})$, is called the *conditional probability* (we use this term only if it is clear from the context). It is easily shown that $R(G^*) = R^*$. In general $R^* > 0$ unless the labels are a deterministic function of the features, and therefore even the optimal classifier misclassifies sometimes. For that reason the quantity of interest for the performance evaluation of a classifier G is the *excess risk* (or *regret*) $R(G) - R(G^*)$. The excess risk can also be written in a interesting integral form,

$$R(G) - R(G^*) = \int_{G \Delta G^*} |2\eta(\boldsymbol{x}) - 1| \mathrm{d}P_{\boldsymbol{X}}(\boldsymbol{x}) \ , \tag{2.1}$$

where Δ denotes the symmetric difference between two sets[1], and $P_{\boldsymbol{X}}$ is the marginal distribution of $\boldsymbol{X}$.

As mentioned before $P_{\boldsymbol{X}Y}$ is generally unknown, therefore direct construction of the Bayes classifier is impossible. One must construct a classifier based on *training* examples. In the classical frameworks [28] these training examples are simply i.i.d. samples from the distribution $P_{\boldsymbol{X}Y}$. In this work we consider a slightly different, somewhat more flexible setting, where the learner can inquire about particular feature vectors. Under some assumptions on the marginal feature distribution $P_{\boldsymbol{X}}$ this is a

[1]Define $A \Delta B \triangleq (A \cap B^c) \cup (A^c \cap B)$, where A^c and B^c are the complement of A and B respectively.

more powerful sampling paradigm. Formally suppose that we have available a large (infinite) pool of feature examples we can choose from, large enough so that we can choose any feature point $\boldsymbol{X}_i \in [0,1]^d$ and observe its label Y_i. The data collection has a temporal aspect to it, namely we collect the labeled examples one at the time, starting with $(\boldsymbol{X}_1, Y_1)$ and proceeding until $(\boldsymbol{X}_n, Y_n)$ is observed. One can view this process as a query learning procedure, in which one queries the label of a feature vector. Formally we have:

A1 - Given the feature vector $\boldsymbol{X}_i$, $i \in \{1, \ldots, n\}$, the label $Y_i \in \{0, 1\}$ is such that

$$\Pr(Y_i = 1 | \boldsymbol{X}_i) = \eta(\boldsymbol{X}_i) \ .$$

The random variables $\{Y_i\}_{i=1}^n$ are conditionally independent given $\{X_i\}_{i=1}^n$.

A2.1 - Passive Sampling: $\boldsymbol{X}_i$ is independent of $\{Y_j\}_{j \neq i}$.

A2.2 - Active Sampling: $\boldsymbol{X}_i$ depends only on $\{\boldsymbol{X}_j, Y_j\}_{j<i}$. In other words $\boldsymbol{X}_i$ obeys a causal relation of the form

$$\boldsymbol{X}_i = h(\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1}, \boldsymbol{U}_i) \ ,$$

where $h(\cdot)$ is a deterministic function, and $\boldsymbol{U}_i$ accounts for possible randomization of the sampling rule. In other words $\boldsymbol{U}_i$ is a random variable, independent of $(\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1})$.

The function $h(\cdot)$, together with $\boldsymbol{U}_i$, is called the the sampling strategy at time i.

The collection of sampling strategies for all $i \in \{1, \ldots, n\}$ is called the *sampling strategy*, denoted by S_n. It completely defines the sampling procedure. After collecting the n labeled examples, that is after collecting $\{\boldsymbol{X}_i, Y_i\}_{i=1}^n$, we construct a classifier $\widehat{G}_n$ that is desirably "close" to G^* in terms of excess risk. The subscript n denotes dependence on the data set, to ease the notational burden of describing explicitly that dependence.

Under the passive sampling scenario (A2.1) the sample locations do not depend on the labels (except for the trivial dependence between $\boldsymbol{X}_i$ and Y_i), and therefore the collection of sample points $\{\boldsymbol{X}_i\}_{i=1}^n$ can be chosen before any observations (*i.e.*, labels) are collected. On the other hand the active sampling scenario (A2.2) allows for the i^{th} sample location to be chosen using all the information collected up to that point (the previous $i-1$ samples). It is clear that (A2.2) is more general than (A2.1), with the former assumption allowing for much more flexibility. It is important to remark that in the active sampling setting (A2.2) the learning algorithm is adaptively choosing the feature $\boldsymbol{X}_i$. This is often referred to as *adaptive sampling* or *query learning*, where the learner can make queries to an expert, requesting labels of features from synthetic examples [29]. Other active learning paradigms exist, for example *pool-based learning*, where we assume a large (infinite) pool of feature examples for which we can ask for a label. If we assume a nearly infinite pool of unlabeled examples, and that the marginal feature density $p_{\boldsymbol{X}}$ is bounded away from zero, then we can essentially choose any feature point $\boldsymbol{X}_i$ and observe its label Y_i.

To be able to present performance guarantees on the excess risk behavior we need to impose further conditions on the possible distributions $P_{\boldsymbol{X}Y}$. We are particularly interested in the framework proposed by Tsybakov in [27], consisting of a characterization of the regularity of the Bayes decision sets, and the behavior of the conditional probability η in the vicinity of the Bayes decision boundary.

2.3 One-Dimensional Threshold Classifiers ($d = 1$)

In this section we consider a class of problems in which the Bayes classifier is a threshold function. Although this corresponds to a rather simple class of distributions, a complete characterization of achievable performance for active learning in this class was previously unknown. Moreover, the study of this simple class sheds light on the potential advantages and limitations of active learning, and provides crucial understanding to tackle more complicated problems. Throughout this section the feature space is the unit interval $[0, 1]$. Let P_{XY} be the distribution governing $(X, Y) \in [0, 1] \times \{0, 1\}$. Assume that the Bayes classifier for this distribution is of the form $[\theta^*, 1]$, which means that $\eta(x) < 1/2$ for all $x < \theta^*$ and $\eta(x) \geq 1/2$ for all $x \geq \theta^*$. We assume that p_X, the marginal density of X with respect to the Lebesgue measure, is uniform on $[0, 1]$, although the results in this chapter are easily generalized to the case where that marginal density is not uniform, but bounded above and below (in which case one obtains exactly the same rates of excess risk convergence). In order to gain a deeper understanding of the potential of active learning we impose further

conditions on η, characterizing the behavior of the conditional probability around the Bayes decision boundary. For $\kappa \geq 1$ and $c > 0$ we assume that

$$|\eta(x) - 1/2| \geq c|x - \theta^*|^{\kappa - 1} \ , \tag{2.2}$$

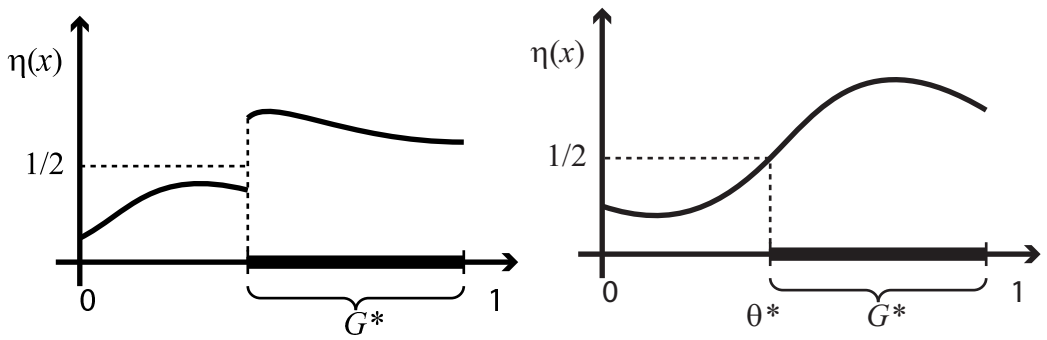
for all x such that $|\eta(x) - 1/2| \leq \delta$, with $\delta > 0$ (with $0^0 \stackrel{\Delta}{=} \lim_{t \to 0^+} t^0 = 1$).

The condition above is very similar to the "noise-condition" introduced by Tsybakov [27]. Condition (2.2) indicates that $\eta(\cdot)$ cannot be arbitrarily "flat" around the decision boundary and plays a critical role on the performance of any classification rule obtained through labeled examples. We also assume a reverse-sided condition on $\eta(\cdot)$, namely

$$|\eta(x) - 1/2| \leq C|x - \theta^*|^{\kappa - 1} \ , \tag{2.3}$$

for all x such that $|\eta(x) - 1/2| \leq \delta$, where $C > c$. This condition, together with (2.2), provides a two-sided characterization of the "noise" around the decision boundary. Similar two-sided conditions have been proposed for other problems [30, 31].

Let $\mathcal{P}(\kappa, c, C)$ be the class of distributions with uniform marginal P_X and satisfying (2.2) and (2.3). If $\kappa = 1$ then the $\eta(\cdot)$ function "jumps" across $1/2$, that is $\eta(\cdot)$ is bounded away from the value $1/2$ (see Figure 2.1(a)). If $\kappa > 1$ then $\eta(\cdot)$ crosses the value $1/2$ at θ^*. An especially interesting case (from a practical perspective) corresponds to $\kappa = 2$ (Figure 2.1(b)). In this case the conditional probability behaves



(a) (b)

Figure 2.1: Examples of two conditional distributions $\eta(x) = \Pr(Y = 1|X = x)$. (a) In this case $\eta(\cdot)$ satisfies the margin condition with $\kappa = 1$; (b) Here the margin condition is satisfied for $\kappa = 2$.

linearly around 1/2, a condition that could arise simply due to errors or inaccuracies in the recorded features. Finally if $\kappa > 2$ then $\eta(\cdot)$ is very "flat" around θ^*.

We begin by presenting lower bounds on the performance of active and passive learning methods for the class of distributions $\mathcal{P}(\kappa, c, C)$. Most of the results that follow involve multiplicative *constant factors*, that is, factors that do not depend on the sample size n. We will denote these by the symbol const, generally without explicitly describing them.

Theorem 1 (Minimax Lower Bound for $d = 1$)**.** *Let $\kappa > 1$. Under the assumptions (A1) and (A2.2) we have*

$$\inf_{\widehat{G}_n, S_n} \sup_{P_{XY} \in \mathcal{P}(\kappa,c,C)} \mathbb{E}\left[R(\widehat{G}_n) - R(G^*)\right] \geq \text{const}(\kappa, c, C)\ n^{-\frac{\kappa}{2\kappa-2}}\ ,$$

for n large enough, where $\text{const}(\kappa, c, C) > 0$ *and the* infimum *is taken over the set of all possible classification rules $\widehat{G}_n$ and active sampling strategies S_n. Note also that*

condition (2.3) *does not play a prominent role in the result, in other words even when* $C = \infty$ *we have* $\text{const}(\kappa, c, C) < \infty$.

The theorem is proved in Section 2.6.1. The proof employs relatively standard techniques, and follows the approach in [32]. The key idea is to reduce the original problem to the problem of deciding among a finite collection of representative distributions. The determination of an appropriate collection of such distributions and careful management of assumption (A2.2) are the key aspects of the proof.

Contrast this result with the one attained for passive learning. Under the passive learning model it is clear that the sample locations $\{X_i\}_{i=1}^n$ must be scattered around the interval $[0, 1]$ in a somewhat uniform manner. These can be deterministically placed, for example over a uniform grid, or simply taken uniformly distributed over $[0, 1]$. Using similar lower bounding techniques, as remarked in the proof of Theorem 1, it can be shown that under assumptions (A1), (A2.1), and $\kappa \geq 1$, we have

$$\inf_{\widehat{G}_n} \sup_{P_{XY} \in \mathcal{P}(\kappa,c,C)} \mathbb{E}\left[R(\widehat{G}_n) - R(G^*)\right] \geq \text{const}(\kappa, c, C)\ n^{-\frac{\kappa}{2\kappa-1}}\ , \tag{2.4}$$

where the samples $\{X_i\}_{i=1}^n$ are independent and identically distributed (i.i.d.) uniformly over $[0, 1]$. Furthermore this bound is tight, in the sense that it is possible to devise classification strategies attaining the same asymptotic excess risk behavior, like the ones described in [27].

We notice that under the passive learning model the excess risk decays at a strictly slower rate than for the active sampling scenario. The difference is dramatic as $\kappa \to 1$.

If $\kappa = 1$ (Figure 2.1(a)) it can actually be shown that an exponential rate of error decay is attainable by active sampling [23]. As $\kappa \to \infty$ the excess risk decay rates become similar, regardless of the sampling paradigm (either (A2.1) or (A2.2)). This indicates that if no assumptions are made on the conditional distribution $\Pr(Y = 1|X = x)$ then no advantage can be gained from the extra complexity of active sampling. As remarked before a most relevant case is $\kappa = 2$. In this case, the rate for active learning is n^{-1}, which is significantly faster than $n^{-2/3}$, the best possible rate for passive learning. Also observe that the difference between active and passive learning rates becomes arbitrarily large as $\kappa \to 1$, with the excess risk of active learning tending to decay faster than n^{-p} for any $p > 0$, and that of passive learning tending to decay like n^{-1}.

Next we describe a methodology showing that the rates of Theorem 1 are nearly achievable. We start by presenting an algorithm proposed by Burnashev and Zigangirov [23], inspired by an idea of Horstein [22]. This algorithm is designed to work in the bounded noise case, that is when $\kappa = 1$, corresponding to a scenario where the conditional probability $\eta(x) = \Pr(Y = 1|X = x)$ is bounded away from $1/2$, that is $|\eta(x) - 1/2| \geq c$ for all $x \in [0, 1]$. This algorithm is then adapted to handle situations in which $\kappa > 1$.

2.3.1 Bounded noise rate ($\kappa = 1$)

First let us suppose that the observations are noiseless (that is $\eta(x) \in \{0, 1\}$). Then it is clear that one can estimate the Bayes decision boundary θ^* very efficiently

using binary bisection: start by taking a sample at $X_1 = 1/2$. Depending on the resulting label $Y_1|X_1$ we know if θ^* is to the left of X_1 (if $Y_1 = 1$) or to the right (if $Y_1 = 0$). Proceeding accordingly we can construct an estimate of θ^* denoted by $\widehat{\theta}_n$ and a corresponding classifier $\widehat{G}_n \stackrel{\Delta}{=} [\widehat{\theta}_n, 1]$ such that

$$R(\widehat{G}_n) - R(G^*) = |\widehat{\theta}_n - \theta^*| \leq 2^{-(n+1)} \ .$$

Now let us assume that some level of noise is present, but that the bounded noise condition, $|\eta(x) - 1/2| \geq c$ for all $x \in [0, 1]$, is met. The learning task is more complicated in this case because deciding where to sample depends on noisy and therefore somewhat unreliable label observations. Nevertheless there is a probabilistic bisection method, proposed in [22], that is suitable for this purpose. The key idea stems from Bayesian estimation. Suppose that we have a prior probability density function $p_0(\cdot)$ on the unknown parameter θ^*, namely that θ^* is uniformly distributed over the interval $[0, 1]$ (that is $p_0(x) = \mathbf{1}\{x \in [0, 1]\}$). To illustrate the process, let us assume the particular scenario that $\theta^* = 1/4$. As in the noiseless case, begin by taking a measurement at $X_1 = 1/2$, that is collect Y_1 given X_1. With probability at least $\eta(X_1) \geq 1/2 + c$ we observe a one, and with probability at most $1 - \eta(X_1) \leq 1/2 - c$ we observe a zero. Therefore it is more likely to observe a one than a zero. Assume for example that $Y_1 = 1$. Given these facts we can compute a "posterior" density $p_1(\cdot)$ simply by applying an approximate Bayes rule (we assume a worst case setting,

where a 1 is observed with probability $1/2 + c$). In this case we would get that

$$p_1(x|X_1 = 1/2, Y_1 = 1) = \begin{cases} 1 + 2c & \text{, if } x \leq 1/2, \\ 1 - 2c & \text{, if } x > 1/2, \end{cases} .$$

The next step is to choose the sample location X_2. We choose X_2 so that is *bisects* the posterior distribution, that is, we take X_2 such that $\Pr_{\theta \sim p_1}(\theta > X_2|X_1, Y_1) = \Pr_{\theta \sim p_1}(\theta < X_2|X_1, Y_1)$. In other words X_2 is just the median of the posterior distribution. If our model is correct, the probability of the event $\{\theta < X_2\}$ is identical to the probability of the event $\{\theta > X_2\}$ and therefore sampling at X_2 seems to be most informative. We continue iterating this procedure until we have collected n samples. The estimate $\widehat{\theta}_n$ is defined as the median of the final posterior distribution. Note that if $c = 1/2$ then probabilistic bisection is simply the binary bisection described above.

The above algorithm seems to work extremely well in practice, but it is difficult to analyze and we are not aware of theoretical guarantees for it, especially pertaining rates of error decay. In [23] an algorithm inspired by that approach was presented. Although its operation is slightly more complicated, it is easier to analyze. The proposed algorithm uses essentially the same ideas but enforces a discrete structure for the posterior, by forcing the samples X_i to lie on a grid, in particular $X_i \in \{0, t, 2t, \ldots, 1\}$ where $m = t^{-1} \in \mathbb{N}$. Furthermore in the application of the Bayes rule we use c' instead of c, where $0 < c' < c < 1/2$. A description of the algorithm can be found in [23], and it is also presented in Appendix A, together with the

performance analysis. We call this method the Burnashev-Zigangirov (BZ) algorithm. This algorithm returns a confidence interval $\widehat{I}_n = [t(i-1), ti]$, $i \in \{1, \ldots, m\}$, such that with high probability $\theta^* \in \widehat{I}_n$. In fact we have the following remarkable result [23]:

$$\Pr(\theta^* \notin \widehat{I}_n) \leq \frac{1-t}{t} \left(\frac{1+2c}{2+4c'} + \frac{1-2c}{2-4c'} \right)^n . \tag{2.5}$$

This bound on the probability of error (the right side of the above expression) can be minimized taking $c' = \frac{1-\sqrt{1-4c^2}}{4c}$, yielding

$$\Pr(\theta^* \notin \widehat{I}_n) \leq \frac{1}{t} \left(\frac{1}{2} + \frac{1}{2}\sqrt{1-4c^2} \right)^n \leq \frac{1}{t} \left(1-c^2\right)^n \leq \frac{1}{t} \exp(-nc^2) , \tag{2.6}$$

where the last two steps follow from the fact that $\sqrt{x} \leq (x+1)/2$ for all $x \geq 0$, and that $(1+s)^x \leq \exp(xs)$ for all $x > 0$ and $s > -1$. An estimate $\widehat{\theta}_n$ can be constructed using, for example, the mid-point of the interval $\widehat{I}_n$, and a candidate classification set/rule is $\widehat{G}_n = [\widehat{\theta}_n, 1]$. To get a bound on the expected excess risk one proceeds

using the standard integration approach.

$$\begin{aligned}
\mathbb{E}[R(\widehat{G}_n)] - R(G^*) &= \mathbb{E}\left[\int_{\widehat{G}_n \Delta G^*} |2\eta(x) - 1| \mathrm{d}x\right] \\
&\leq \mathbb{E}\left[\int_{\widehat{G}_n \Delta G^*} \mathrm{d}x\right] \\
&= \mathbb{E}\left[|\widehat{\theta}_n - \theta^*|\right] \\
&= \int_0^1 \Pr\left(|\widehat{\theta}_n - \theta^*| > z\right) \mathrm{d}z \\
&= \int_0^t \Pr\left(|\widehat{\theta}_n - \theta^*| > z\right) \mathrm{d}z + \int_t^1 \Pr\left(|\widehat{\theta}_n - \theta^*| > z\right) \mathrm{d}z \\
&\leq t + (1-t) \Pr\left(|\widehat{\theta}_n - \theta^*| > t\right) \\
&\leq t + \frac{1}{t} \exp(-nc^2) \ .
\end{aligned}$$

Taking $t = \exp(-nc^2/2)$ yields

$$\mathbb{E}[R(\widehat{G}_n)] - R(G^*) \leq 2 \exp(-nc^2/2) \ .$$

Notice that the excess risk decays exponentially in the number of samples. This is much faster than what is attainable using passive sampling, where the decay rate is $1/n$. This is the same error behavior as in the noiseless scenario, where we have an exponential rate of error decay using binary bisection. The difference is that now the exponent depends on the noise margin c, larger noise margins corresponding to faster error decay rates. In [23] some lower bound guarantees on the exponential decay rate are also given.

2.3.2 Unbounded rate noise: $\kappa > 1$

In this section we consider scenarios where the noise rate is not bounded, that is, observations made closer to the transition point θ^* are noisier than observations made further away. In light of Theorem 1 this degradation of observation quality hinders extremely fast excess risk decay rates.

To study this case, we proceed as in the case $\kappa = 1$. Collect samples over a grid, namely $X_i \in \{0, t, 2t, \dots, 1\}$ where $m = t^{-1} \in \mathbb{N}$. Assume for a brief moment that the grid is not aligned with the transition point θ^*, for example $|\theta^* - kt| \geq t/3$ for all $k \in \{0, \dots, m\}$. This implies that $|\eta(x) - 1/2| \geq c(t/3)^{\kappa-1}$ for all $x \in \{0, t, \dots, 1\}$ (assume that t is small enough so that $\delta \geq c(t/3)^{\kappa-1}$). Of course the non-alignment assumption is quite unrealistic, since θ^* may be arbitrarily close to a grid point, but let us assume this condition for now and remove it later. We can proceed by using the algorithm described in the previous section replacing c by $c(t/3)^{\kappa-1}$ and using (2.5). Notice also that due to (2.3) the behavior of the expected excess risk is related to the behavior of $|\widehat{\theta}_n - \theta^*|$ in an interesting way,

$$\begin{aligned}
\mathbb{E}[R(\widehat{G}_n)] - R(G^*) &= \mathbb{E}\left[\int_{\widehat{G}_n \Delta G^*} |2\eta(x) - 1| \mathrm{d}x\right] \\
&\leq \frac{2C}{\delta}\, \mathbb{E}\left[\int_{\widehat{G}_n \Delta G^*} |x - \theta^*|^{\kappa-1} \mathrm{d}x\right] \\
&= \frac{2C}{\delta}\, \mathbb{E}\left[\int_{\min\{\theta^*, \widehat{\theta}_n\}}^{\max\{\theta^*, \widehat{\theta}_n\}} |x - \theta^*|^{\kappa-1} \mathrm{d}x\right] \\
&= \frac{2C}{\delta\kappa} \mathbb{E}[|\widehat{\theta}_n - \theta^*|^{\kappa}] \,,
\end{aligned}$$

where the factor $1/\delta$ arises because (2.3) is valid only when $|\eta(x) - 1/2| \leq \delta$. We now proceed in a similar fashion as before

$$\begin{aligned}
\mathbb{E}[R(\widehat{G}_n)] - R(G^*) \leq \frac{2C}{\delta\kappa}\mathbb{E}\left[|\widehat{\theta}_n - \theta^*|^\kappa\right] &= \frac{2C}{\delta\kappa}\int_0^1 \Pr\left(|\widehat{\theta}_n - \theta^*|^\kappa > z\right) \mathrm{d}z \\
&= \frac{2C}{\delta\kappa}\int_0^1 \Pr\left(|\widehat{\theta}_n - \theta^*| > z^{1/\kappa}\right) \mathrm{d}z \\
&= \frac{2C}{\delta\kappa}\left[\int_0^{t^\kappa} \Pr\left(|\widehat{\theta}_n - \theta^*| > z^{1/\kappa}\right) \mathrm{d}z \right. \\
&\quad \left. + \int_{t^\kappa}^1 \Pr\left(|\widehat{\theta}_n - \theta^*| > z^{1/\kappa}\right) \mathrm{d}z\right] \\
&\leq \frac{2C}{\delta\kappa}\left[t^\kappa + (1 - t^\kappa)\Pr\left(|\widehat{\theta}_n - \theta^*| > t\right)\right] \\
&\leq \frac{2C}{\delta\kappa}\left[t^\kappa + \frac{1}{t}\exp\left(-nc^2(t/3)^{2\kappa-2}\right)\right] ,
\end{aligned}$$

Finally, let

$$t = 3\left(\frac{\kappa+1}{c^2(2\kappa-2)}\,\frac{\log n}{n}\right)^{\frac{1}{2\kappa-2}} ,$$

to conclude that

$$\mathbb{E}[R(\widehat{G}_n) - R(G^*)] \leq \mathrm{const}(\kappa, c, C) \cdot \left(\frac{\log n}{n}\right)^{\frac{\kappa}{2\kappa-2}} , \tag{2.7}$$

where $\mathrm{const}(\kappa, c, C) > 0$ is a constant factor. The error decay rate of this upper bound corresponds to the rate of lower bound in Theorem 1, apart from logarithmic factors. The result indicates that, in principle, a methodology similar to the BZ algorithm might allow us to achieve the lower bound rates. Note that since t is vanishing with n, the assumption that $\delta \geq c(t/3)^{\kappa-1}$ holds for n large enough.

It is important to emphasize that the above result holds under the assumption that the sampling grid is not aligned with the unknown threshold point θ^*. If this is not the case then we will have $|\eta(x) - 1/2| < c(t/3)^{\kappa-1}$ for one of the sampling points, and the analysis above is no longer valid. This set-back can be avoided in different ways, for example using a direct modification of the BZ sampling strategy, as in [33], or using several sampling grids simultaneously. We describe the latter approach in what follows. Begin by dividing the available measurements into three sets of the same size, and use three sampling grids, each with different set of sampling locations (to be defined shortly). Suppose we have a budget of n samples (without loss of generality assume that n is divisible by 3). We allocate $n/3$ samples and run the BZ algorithm for one sampling grid. Then we use other $n/3$ samples and run the BZ algorithm for another sampling grid, essentially a slightly shifted version of the first sampling grid, and proceed in an analogous fashion with the remaining $n/3$ samples. The rational is that at most one of these sampling grids is going to be closely aligned with the unknown transition point θ^*, and therefore the other two cannot be aligned with θ^*. Therefore for at least two out of these three estimators we have provable performance bounds so it suffices to check for agreement among these three estimates: if at least two of them agree on a possible location for θ^* an overall estimate $\widehat{\theta}_n$ can be generated. Next we describe this procedure formally.

Consider three different sampling grids, $\text{Grid}^{(A)} = \{t, 2t, \ldots, 1-t\}$, $\text{Grid}^{(B)} = \{t/3, 4t/3, \ldots, 1-2t/3\}$, and $\text{Grid}^{(C)} = \{2t/3, 5t/3, \ldots, 1-t/3\}$. Samples can also

be "taken" at $X_i \in \{0,1\}$ but in that case we will impose that $Y_i = X_i$. This does not effect the algorithm performance. For at least two of the estimators we have $|\theta^* - x| \geq t/6$ for all the points in the respective sampling grids. This means that for samples x taken on those grids we have $|\eta(x) - 1/2| \geq c(t/6)^{\kappa-1}$ and therefore we can use the bounded noise rate results in the analysis. We now run the BZ algorithm three times, using $n/3$ samples each time, taken on the grids $\text{Grid}^{(A)}$, $\text{Grid}^{(B)}$ and $\text{Grid}^{(C)}$ respectively. Let $\widehat{I}_n^{(A)}$, $\widehat{I}_n^{(B)}$ and $\widehat{I}_n^{(C)}$ denote the confidence intervals returned by the three instances of the BZ algorithm. Next we aggregate these confidence intervals to construct a final confidence interval $\widehat{I}_n$ as follows:

$$
\begin{array}{l}
\texttt{If } |\widehat{I}_n^{(A)} \cap \widehat{I}_n^{(B)}| = 2t/3 \texttt{ let } \widehat{I}_n = \widehat{I}_n^{(A)} \cup \widehat{I}_n^{(B)} \\
\quad \texttt{else if } |\widehat{I}_n^{(A)} \cap \widehat{I}_n^{(C)}| = 2t/3 \texttt{ then } \widehat{I}_n = \widehat{I}_n^{(A)} \cup \widehat{I}_n^{(C)} \\
\quad\quad \texttt{else if } |\widehat{I}_n^{(B)} \cap \widehat{I}_n^{(C)}| = 2t/3 \texttt{ then } \widehat{I}_n = \widehat{I}_n^{(B)} \cup \widehat{I}_n^{(C)} \\
\quad\quad\quad \texttt{else let } \widehat{I}_n = \{1/2\}
\end{array}
$$

Proposition 1. *Proceeding as described above and assuming* (2.2) *we have*

$$\Pr(\theta^* \notin \widehat{I}_n) \leq \frac{2}{t} \exp\left(-\frac{n}{3} c^2 (t/6)^{2\kappa-2}\right) . \tag{2.8}$$

Proof. We need to consider two separate scenarios: either θ^* is "close" to a point in one of the sampling grids, or θ^* is close to zero or one. Consider the first situation and without loss of generality assume that θ^* is close to a point in $\text{Grid}^{(A)}$, that is there exists $x \in \text{Grid}^{(A)}$ such that $|\theta^* - x| < t/6$. This implies that for all $x \in$

$\text{Grid}^{(B)} \cup \text{Grid}^{(C)}$ we have $|\theta^* - x| \geq t/6$, and therefore

$$\Pr(\theta^* \notin \widehat{I}_n^{(B)}) \leq \frac{1}{t} \exp\left(-\frac{n}{3} c^2 (t/6)^{2\kappa-2}\right) ,$$

and

$$\Pr(\theta^* \notin \widehat{I}_n^{(C)}) \leq \frac{1}{t} \exp\left(-\frac{n}{3} c^2 (t/6)^{2\kappa-2}\right) .$$

Define the event $E = \{\theta^* \in \widehat{I}_n^{(B)} \cup \widehat{I}_n^{(C)}\}$ and notice that

$$\Pr(E) \geq 1 - \frac{2}{t} \exp\left(-\frac{n}{3} c^2 (t/6)^{2\kappa-2}\right) .$$

Assume that E holds. If $|\widehat{I}_n^{(A)} \cap \widehat{I}_n^{(B)}| = 2t/3$ then $\theta^* \in \widehat{I}_n^{(B)} \subseteq \widehat{I}_n$, if $|\widehat{I}_n^{(A)} \cap \widehat{I}_n^{(C)}| = 2t/3$ then $\theta^* \in \widehat{I}_n^{(C)} \subseteq \widehat{I}_n$ and, if $|\widehat{I}_n^{(B)} \cap \widehat{I}_n^{(C)}| = 2t/3$ then $\theta^* \in \widehat{I}_n^{(B)} \subseteq \widehat{I}_n$. Finally since there is a point in $\text{Grid}^{(A)}$ close to θ^* this implies that, under E, we have $|\widehat{I}_n^{(B)} \cap \widehat{I}_n^{(C)}| = 2t/3$ and so $\widehat{I}_n \neq \{1/2\}$. In conclusion, if θ^* is close to a point in $\text{Grid}^{(A)}$ then $\Pr(\theta^* \notin \widehat{I}_n) \leq 1 - \Pr(E)$, implying (2.8).

Now consider the case when θ^* is close to zero or one. Without loss of generality suppose θ^* is close to zero, that is $\theta^* < t/6$. Then

$$\Pr(\theta^* \in \widehat{I}_n^{(A)} \cup \widehat{I}_n^{(C)}) \geq 1 - \frac{2}{t} \exp\left(-\frac{n}{3} c^2 (t/6)^{2\kappa-2}\right) ,$$

and so a similar reasoning as above can be applied yielding the desired result. □

Now we are in a similar situation as before, and taking θ_n as the midpoint of $\widehat{I}_n$

yields the bound

$$\Pr(|\widehat{\theta}_n - \theta^*| \geq t) \leq \frac{2}{t} \exp\left(-\frac{n}{3} c^2 (t/6)^{2\kappa-2}\right) \ , \tag{2.9}$$

without requiring any assumptions on the location of θ^*. This algorithm therefore provably attains the rate in (2.7).

Although the above methodology is satisfying from a theoretical point of view, it is somewhat wasteful (essentially only one third of the samples are effectively used), and does not generalize well to more complicated and realistic scenarios than the one considered in this chapter. It is worth pointing out that, even when the assumption that the sampling grid does not line up with the transition θ^* does not hold, the original BZ algorithm still works extremely well in practice. The difficulties arise solely on the performance analysis. We conclude this section by summarizing the results in the following theorem.

Theorem 2. *Under assumptions* (2.2) *and* (2.3) *the active learning algorithms above satisfies*

$$\sup_{P_{XY} \in \mathcal{P}(\kappa,c,C)} \mathbb{E}[R(\widehat{G}_n) - R(G^*)] \leq \begin{cases} 2\exp(-nc^2/2) & , \text{ if } \kappa = 1 \\ \text{const}(\kappa, c, C) \left(\frac{\log n}{n}\right)^{\frac{\kappa}{2\kappa-2}} & , \text{ if } \kappa > 1 \end{cases} ,$$

for n large enough, where $\text{const}(\kappa, c, C) > 0$ *is a constant factor.*

Note that for $\kappa > 1$ the rate of error decay is almost as good as the rate in

Theorem 1. This means that the methodology developed is nearly minimax optimal, since the rate in Theorem 2 differs only by a logarithmic factor in n. In the next section we generalize these results to a non-parametric setting in multiple dimensions.

2.4 Boundary Fragments ($d > 1$)

In this section we consider a much more general class of distributions, namely scenarios where the Bayes decision set is a boundary fragment. In other words the Bayes decision set is the epigraph of function. We consider Hölder smooth boundary functions. Throughout this section we assume that $d \geq 2$, where d is the dimension of the feature space $[0,1]^d$.

Definition 1. *A function $f : [0,1]^{d-1} \to \mathbb{R}$ is* Hölder smooth*, with smoothness parameter $\alpha \geq 1$, if it has continuous partial derivatives up to order $k = \lfloor \alpha \rfloor$ ($\lfloor \alpha \rfloor$ is the maximal integer such that $\lfloor \alpha \rfloor < \alpha$) and*

$$\forall\ \boldsymbol{z}, \boldsymbol{x} \in [0,1]^{d-1} : \quad |f(\boldsymbol{z}) - TP_{\boldsymbol{x}}(\boldsymbol{z})| \leq L\|\boldsymbol{z} - \boldsymbol{x}\|^{\alpha}\ ,$$

where $L > 0$ and $TP_{\boldsymbol{x}}(\cdot)$ denotes the degree k Taylor polynomial approximation of f expanded around $\boldsymbol{x}$. Denote this class of functions by $\Sigma_{d-1}(L, \alpha)$.

For any $g \in \Sigma_{d-1}(L, \alpha)$ let $\text{epi}(g) \triangleq \left\{(\boldsymbol{x}, y) \in [0,1]^{d-1} \times [0,1] : y \geq g(\boldsymbol{x})\right\}$ be the

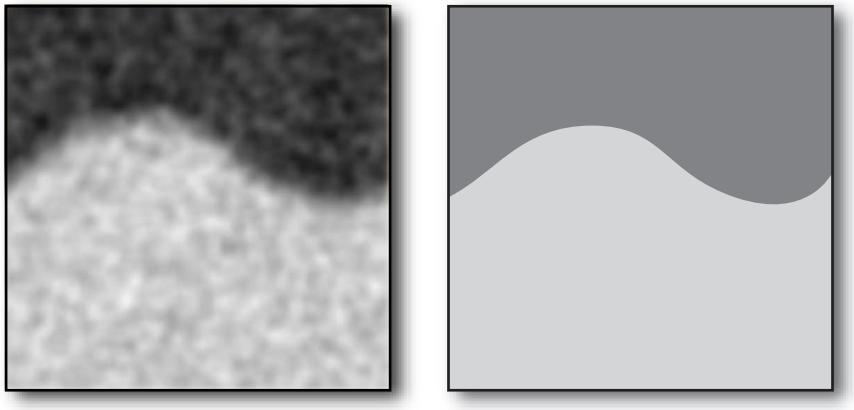
epigraph of g. Define the boundary fragment class of sets

$$\mathcal{G}_{\mathrm{BF}}(L,\alpha) \triangleq \{\mathrm{epi}(g) : g \in \Sigma_{d-1}(L,\alpha)\} \ .$$

In other words $\mathcal{G}_{\mathrm{BF}}(L,\alpha)$ is a collection of sets indexed by Hölder smooth functions of the first $d-1$ coordinates of the feature domain $[0,1]^d$. Therefore the set G^* and the corresponding boundary function g^* are equivalent representations of the Bayes classifier. See Figure 2.2(b) for an example of such a set. Although the boundary fragment classes might seem artificial and unrealistic they are nevertheless useful in order to determine fundamental performance limits and to gain understanding about more complicated model classes with similar characteristics (*e.g.*, similar characterization of the boundary behavior). There are various examples in the literature where boundary fragments have been used for such purposes, see for example [34, 35].

Furthermore, we assume that $p_{\boldsymbol{X}}$, the marginal density of $\boldsymbol{X}$ with respect to the Lebesgue measure, is uniform but, as before, the results in this chapter can be easily generalized to the case there $p_{\boldsymbol{X}}$ is bounded above and below, yielding the same rates of error convergence. As in the previous section we require also $\eta(\cdot)$ to have a certain behavior around the decision boundary. Let $\boldsymbol{x} = (\tilde{\boldsymbol{x}}, x_d)$ where $\tilde{\boldsymbol{x}} = (x_1, \ldots, x_{d-1})$. Let $\kappa \geq 1$ and $c > 0$, then for some $\delta > 0$

$$|\eta(\boldsymbol{x}) - 1/2| \geq c|x_d - g^*(\tilde{\boldsymbol{x}})|^{\kappa-1} \ , \tag{2.10}$$



(a) (b)

Figure 2.2: (a) Example of the conditional distribution $\eta(\cdot)$ of an element of the class $\mathrm{BF}(\alpha, \kappa, L, C, c)$ when $d = 2$ and $\alpha = 2$. (b) The corresponding Bayes classifier.

for all $\boldsymbol{x}$ such that $|\eta(\boldsymbol{x}) - 1/2| \leq \delta$. The condition above is analogous to the one defined in Section 2.3. We assume as well a reverse-sided condition on $\eta(\cdot)$, namely

$$|\eta(\boldsymbol{x}) - 1/2| \leq C|x_d - g^*(\tilde{\boldsymbol{x}})|^{\kappa-1} \,, \tag{2.11}$$

for all $\boldsymbol{x}$ such that $|\eta(\boldsymbol{x}) - 1/2| \leq \delta$, where $C > c$. This condition, together with (2.10), provides a two-sided characterization of the "noise" around the decision boundary. Let $\mathrm{BF}(\alpha, \kappa, L, C, c)$ be the class of distributions satisfying the noise conditions above with parameter κ and whose Bayes classifiers are boundary fragments with smoothness α. An example of such a distribution function, and the corresponding Bayes decision set is presented in Figure 2.2.

2.4.1 Minimax Lower Bounds ($d > 1$)

We begin by presenting lower bounds on the performance of active and passive sampling methods. We start by characterizing active learning for the boundary fragment classes.

Theorem 3. *Let $\rho = (d-1)/\alpha$. Under assumptions (A1) and (A2.2)*

$$\inf_{\widehat{G}_n, S_n} \sup_{P \in BF_{(\alpha,\kappa,L,C,c)}} \mathbb{E}[R(\widehat{G}_n)] - R(G^*) \geq \text{const}(\alpha, \kappa, L, C, c) n^{-\frac{\kappa}{2\kappa+\rho-2}} \ ,$$

for large enough n, where $\inf_{\widehat{G}_n, S_n}$ denotes the infimum over all possible classifiers and active sampling strategies S_n, and $\text{const}(\alpha, \kappa, L, C, c) > 0$ is a constant factor.

The proof of Theorem 3 is presented in Section 2.6.2. An important remark is that, as before, condition (2.11) does not play a prominent role in the lower bound, therefore dropping that assumption (equivalently taking $C = \infty$) does not yield slower rates in the theorem statement.

Contrast this result with the one attained for passive sampling: under the passive sampling scenario it is clear that the sample locations $\{\boldsymbol{X}_i\}_{i=1}^n$ must be scattered around the feature space $[0,1]^d$ in a somewhat uniform manner. These can be simply taken uniformly distributed over $[0,1]^d$. The lower bounds for this passive setting can be easily derived as remarked in the proof of Theorem 3, implying that under (A1)

and (A2.1),

$$\inf_{\widehat{G}_n} \sup_{P \in \mathrm{BF}_{(\alpha,\kappa,L,C,c)}} \mathbb{E}[R(\widehat{G}_n)] - R(G^*) \geq \mathrm{const}(\alpha, \kappa, L, C, c) \cdot n^{-\frac{\kappa}{2\kappa+\rho-1}}, \tag{2.12}$$

for n large enough, where the samples $\{\boldsymbol{X}_i\}_{i=1}^n$ are independent and identically distributed (i.i.d.) uniformly over $[0,1]^d$. Since the upper margin condition (2.11) does not play a prominent role these rates coincide with the ones in [27]. Furthermore this bound is tight, in the sense that it is possible to devise classification strategies attaining the same asymptotic behavior. We notice that under the passive sampling scenario the excess risk decays at a strictly slower rate than the lower bound for the active sampling scenario, and the rate difference can be dramatic, specially for large smoothness α (equivalently low complexity ρ). The active learning lower bound is also tight (in terms of rates, as shown in the next section), which demonstrates that active learning has the potential to improve significantly over passive learning. Finally the result of Theorem 3 is a lower bound, and it therefore applies to the broader classes of distributions introduced in [27], characterized in terms of the metric entropy of the class of Bayes classifiers and a one-sided margin condition, akin to (2.10).

2.4.2 Upper Bounds ($d > 1$)

In this section we present an active learning algorithm for the boundary fragment class and upper bound the corresponding excess risk. The upper bound achieves the rates of Theorem 3 to within a logarithmic factor. This proposed method yields

a classifier $\widehat{G}_n$ that has a boundary fragment structure, although the boundary is no longer a smooth function. It is instead a piecewise polynomial function. The methodology proceeds along the lines of [36,37], extending the one-dimensional active sampling results of Section 2.3 to this higher dimensional setting. To avoid carrying around cumbersome constants we use the 'big-O' [2] notation for simplicity. Also we use a tilde to denote vectors of dimension $d-1$. We focus the description exclusively on the case $\kappa > 1$, but a similar reasoning gives the results for the case $\kappa = 1$.

We begin by constructing a grid over the first $d-1$ dimensions of the feature domain. Let M be an integer and $\tilde{\boldsymbol{l}} \in \{0, \ldots, M\}^{d-1}$. Define the set of line segments $\mathcal{L}_{\tilde{\boldsymbol{l}}} \stackrel{\Delta}{=} \{(M^{-1}\tilde{\boldsymbol{l}}, x_d) : x_d \in [0,1]\}$. We collect N adaptively chosen samples along each line, in order to estimate $g(M^{-1}\tilde{\boldsymbol{l}})$. This yields a total of $N(M+1)^{d-1}$ samples (where $n \geq N(M+1)^{d-1}$). We then interpolate the estimates of g at these points to construct a final estimate of the decision boundary. The adequate choices for M and N will arise from the performance analysis; for now we point out only that both M and N must be growing with the total number of samples n. Figure 2.3 illustrates the procedure.

When restricting ourselves to the line segments in $\mathcal{L}_{\tilde{\boldsymbol{l}}}$, the estimation problem boils down to a one-dimensional change-point detection problem and so we can use the results derived in Section 2.3, in particular (2.9). Since we are using N adaptively

[2] Let u_n and v_n be two real sequences. We say $u_n = O(v_n)$ as $n \to \infty$ if and only if there exists $C > 0$ and $n_0 > 0$ such that $|u_n| \leq C v_n$ for all $n \geq n_0$.

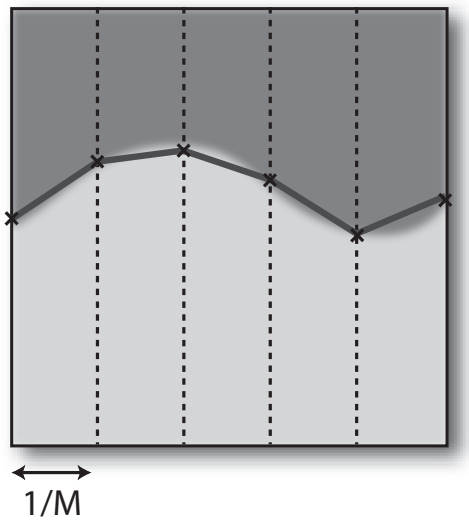


Figure 2.3: Illustration of the active classification procedure for boundary fragments when $d = 2$. In this case $\alpha = 2$ therefore we estimate the true Bayes decision boundary with the aid of piecewise linear polynomials. The crosses represent the estimates of g^* obtained by each one of the $M + 1$ line searches. The dark solid line segments represent the $M/\lfloor\alpha\rfloor$ interpolation polynomials.

chosen samples per line segment, choosing

$$t = t_N \stackrel{\Delta}{=} c_1 \left(\log N/N\right)^{\frac{1}{2\kappa-2}} \tag{2.13}$$

guarantees that $\Pr(|\widehat{g}(M^{-1}\tilde{\boldsymbol{l}}) - g^*(M^{-1}\tilde{\boldsymbol{l}})| > t_N) = O\left(N^{-\gamma}\right)$ as $N \to \infty$, where $\gamma > 0$ can be arbitrarily large provided c_1 is sufficiently large.

Let $\{\widehat{g}(M^{-1}\tilde{\boldsymbol{l}})\}$ be the estimates obtained using this method at each of the points indexed by $\tilde{\boldsymbol{l}}$. We use these estimates to construct a piecewise polynomial fit to approximate g^*. In what follows assume $\alpha > 1$. The case $\alpha = 1$ can be handled in a very similar way (where one would approximate g^* with a stair function). Begin by dividing $[0, 1]^{d-1}$ (that is, the domain of g^*) into cells. Without loss of generally

assume that M is such that $M/\lfloor\alpha\rfloor$ is an integer (this can be enforced with the proper choice of M). Let $\tilde{\boldsymbol{q}} \in \{0, \ldots, M/\lfloor\alpha\rfloor - 1\}^{d-1}$ index the cells

$$I_{\tilde{\boldsymbol{q}}} \stackrel{\Delta}{=} \left[\tilde{\boldsymbol{q}}_1 \lfloor\alpha\rfloor M^{-1}, (\tilde{\boldsymbol{q}}_1 + 1)\lfloor\alpha\rfloor M^{-1}\right] \times \cdots \times \left[\tilde{\boldsymbol{q}}_{d-1} \lfloor\alpha\rfloor M^{-1}, (\tilde{\boldsymbol{q}}_{d-1} + 1)\lfloor\alpha\rfloor M^{-1}\right] \ .$$

Note that these cells partition the domain $[0,1]^{d-1}$ entirely. In each one of the cells we perform a polynomial interpolation using the estimates of g^* at points within the cell. For the cell indexed by $\tilde{\boldsymbol{q}}$ we consider a tensor product polynomial fit $\widehat{L}_{\tilde{\boldsymbol{q}}}$, that can be written as

$$\widehat{L}_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) = \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} \widehat{g}(M^{-1}\tilde{\boldsymbol{l}}) Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \ ,$$

where $\tilde{\boldsymbol{x}} \in [0,1]^{d-1}$. The functions $Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}$ are the tensor-product Lagrange polynomials (see for example [38]),

$$Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \stackrel{\Delta}{=} \prod_{i=1}^{d-1} \prod_{j=0, j \neq \tilde{\boldsymbol{l}}_i - \lfloor\alpha\rfloor \tilde{\boldsymbol{q}}_i}^{\lfloor\alpha\rfloor} \frac{\tilde{\boldsymbol{x}}_i - M^{-1}(\lfloor\alpha\rfloor \tilde{\boldsymbol{q}}_i + j)}{M^{-1}\tilde{\boldsymbol{l}}_i - M^{-1}(\lfloor\alpha\rfloor \tilde{\boldsymbol{q}}_i + j)} \ .$$

We remark that this is a polynomial interpolation and so we have that $\widehat{L}_{\tilde{\boldsymbol{q}}}(M^{-1}\tilde{\boldsymbol{l}}) = \widehat{g}(M^{-1}\tilde{\boldsymbol{l}})$, for all $\tilde{\boldsymbol{l}}$ such that $M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}$. Finally, the estimate $\widehat{g}$ of g^* is given by

$$\widehat{g}(\tilde{\boldsymbol{x}}) = \sum_{\tilde{\boldsymbol{q}} \in \{0,\ldots,M/\lfloor\alpha\rfloor - 1\}^{d-1}} \widehat{L}_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} \ ,$$

which defines a classification rule $\widehat{G}_n$.

Theorem 4. *Let* $M = \lfloor\alpha\rfloor \left\lfloor n^{\frac{1}{\alpha(2\kappa-2)+d-1}} \right\rfloor$ *and* $N = \lfloor n/(M+1)^{d-1} \rfloor$, *and consider*

the active learning algorithm described above. Let $\rho = (d-1)/\alpha$*, then*

$$\sup_{P \in BF(\alpha,\kappa,L,C,c)} \mathbb{E}[R(\widehat{G}_n)] - R(G^*) \leq \text{const}(\alpha, \kappa, L, C, c) \, (\log n/n)^{\frac{\kappa}{2\kappa+\rho-2}} \quad ,$$

for n *large enough, where* $\text{const}(\alpha, \kappa, L, C, c) > 0$*.*

The proof of Theorem 4 is given in Section 2.6.3. One sees that this estimator achieves the rate of Theorem 3 to within a logarithmic factor. It is not clear if the logarithmic factor is an artifact of our construction, or if it is unavoidable. One knows [36] that if $\kappa, \alpha = 1$ the logarithmic factor can be eliminated by using a slightly more sophisticated interpolation scheme.

2.5 Final Remarks and Discussion

We presented upper and lower bounds for active learning algorithms under assumptions on the decision boundary regularity and noise conditions. Since the upper and lower bounds agree up to a logarithmic factor, we may conclude that lower bound is near minimax optimal. That is, for the distributional classes under consideration, no active or passive learning procedure can perform significantly better in terms of excess risk decay rates. The upper bounds were derived constructively, based on active learning algorithms originally developed for one-dimensional change-point detection. In principle, the methodology employed in the upper bound calculation could be applied in practice in the case of boundary fragments and with knowledge of the

key regularity parameters κ and ρ. Unfortunately this is not a scenario one expects to have in realistic settings, and thus a key open problem is the design of active learning algorithms that are adaptive to unknown regularity parameters and capable of handling arbitrary boundaries (not only fragments). It is important to point out that the lower-bounds derived are valid for a broad set of distributional classes, in particular the ones defined in [27]. These classes are characterized in terms of critical attributes for the classification task at hand, and do not impose further conditions that are irrelevant when constructing a classifier. As in our approach the characterization addresses two quantities: (i) the complexity of the corresponding Bayes classifiers; (ii) the behavior of $\eta(x)$ around the level $1/2$. The characterizations below are a generalization of our boundary smoothness and noise margin conditions. Define $d_\Delta(G, G') \stackrel{\Delta}{=} P_X(G \Delta G')$.

Definition 2. *Let $\mathcal{G}$ be a class of subsets of $\mathcal{X}$ and let $\delta > 0$. Let $N_B(\delta, \mathcal{G}, d_\Delta)$ be the smallest value m for which there exists pairs of sets (G_j^L, G_j^U), $j = 1 \ldots, m$, such that $d_\Delta(G_j^L, G_j^U) \leq \delta$ for all $j = 1, \ldots, m$, and for any $G \in \mathcal{G}$ there exists $j(G)$ for which $G_j(G)^L \subseteq G \subseteq G_j(G)^U$. Then $\mathcal{H}_B(\delta, \mathcal{G}, d_\Delta) = \log N_B(\delta, \mathcal{G}, d_\Delta)$ is called the δ-entropy with bracketing of $\mathcal{G}$.*

A class $\mathcal{G}$ of subsets of $\mathcal{X}$ is said to have complexity bound $\rho > 0$ *if there exists a constant $A > 0$ such that*

$$\mathcal{H}_B(\delta, \mathcal{G}, d_\Delta) \leq A\delta^{-\rho}, \quad \forall\ 0 < \delta \leq 1\ .$$

The δ-entropy measures the complexity of a class of sets (below we will choose these to be Bayes classifiers). For example, the class boundary fragment sets $\mathcal{G}_{\mathrm{BF}}(L, \alpha)$ has complexity bound $\rho = (d-1)/\alpha$.

Since the Bayes decision boundary might no longer have a functional description the margin condition (2.10) has to be generalized too.

Definition 3. *Let $\eta(x) = P(Y = 1|X = x)$. We say that η satisfies the* noise condition *with parameter $\kappa \geq 1$ if there exists $c_0 > 0$, $0 < \epsilon_0 \leq 1$ such that*

$$R(G) - R(G^*) \geq c_0 d_\Delta^\kappa(G, G^*) \tag{2.14}$$

for all G such that $d_\Delta(G, G^*) \leq \epsilon_0$.

As (2.10) this characterizes the behavior of $\eta(x)$ near the Bayes decision boundary. Let $\mathcal{P}(\rho, \kappa)$ be a class of distributions satisfying these two conditions, then

$$\liminf_{n\to\infty} \inf_{\widehat{G}_n, S_n} \sup_{P\in\mathcal{P}(\rho,\kappa)} \mathbb{E}[R(\widehat{G}_n)] - R(G^*) n^{\frac{\kappa}{2\kappa+\rho-2}} \geq c_1 > 0 \ ,$$

for a constant $c_1 > 0$. The proof of this fact is essentially Theorem 3, where we just have to note that the class of boundary fragments with Hölder smoothness α has complexity parameter $\rho = (d-1)/\alpha$), and use condition (2.14) directly in the proof, instead of (2.19).

Although we might have lower bounds that hold for general classes, matching upper bounds with a similar form are not generally possible. One difficulty arises

because in most cases *uniform bounds* are not possible, that is, bounds that hold for the entire class of distributions. This is pointed out in [20] and indirectly in [17]. We illustrate this with an example. Consider a one-dimensional problem, with the feature space $[0, 1]$. Suppose we are in the noiseless case and that the Bayes decision sets are simply intervals $[a, b]$ with $a \geq b$. Denote this class of distributions by $\mathcal{I}$. This class is very similar to the threshold class of Section 2.3, although being a little more general. Nevertheless the δ-entropy characterization above holds with ρ arbitrarily close to zero. In light of the lower bounds we would expect to get an excess risk decay faster than any polynomial rate. But now note that distributions where the Bayes decision set has size $1/n$ (*e.g.*, $[0, 1/n]$) it is not possible to accurately detect this set using n samples: with probability $(1 - 1/n)^n$ no sample is going to land inside the Bayes decision set, so all the n labels collected are 0. This means that using these n samples the probability of finding the Bayes decision set is bounded away from zero (since $(1 - 1/n)^n \to \exp(-1)$ as $n \to \infty$). Therefore the best uniform bound we can expect is of the form

$$\inf_{\widehat{G}_n, S_n} \sup_{P \in \mathcal{I}} \mathbb{E}[R(\widehat{G}_n)] - R(G^*) \geq c_2 \frac{1}{n} \ ,$$

for n is large enough and $c_2 > 0$. This bound coincides with the passive learning bound for that same class, so, in terms of minimax performance, active learning does not help in this case. This does not imply that non-uniform bounds are not possible,

in particular it is easy to show that there is an algorithm such that, for each $P \in \mathcal{I}$

$$\mathbb{E}[R(\widehat{G}_n)] - R(G^*) \leq c(P)2^{-n/2} \ ,$$

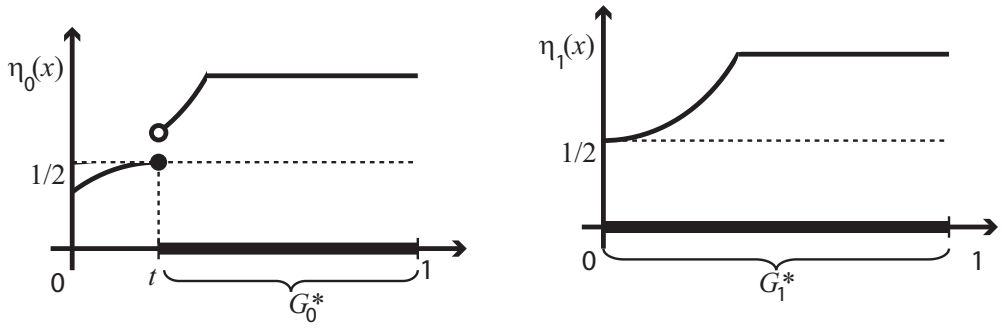
for $n \geq n(P)$, where $c(P) > 0$ and $n(P) > 0$ are functions of P not depending on n. The rates in this bound coincide with the minimax lower bound derived for active learning (in this case for the threshold class, see Section 2.3). It is not known if this kind of matching between minimax lower bounds and non-uniform upper bounds happens for general classes or not, although we expect this to happen for many such classes. The next chapter considers a scenario where one observes such behavior.

The results of this chapter do indicate fundamental limitations of active learning, and thus can provide guidelines for best attainable performance of any method. Moreover, the bounds clarify the situations in which active learning can lead to significant gains over passive learning, and it may be possible to assess the conditions that might hold in a given application in order to gauge the merit of pursuing an active learning approach.

2.6 Proofs

2.6.1 Proof of Theorem 1

The proof strategy follows the basic idea behind standard minimax analysis methods, and consists in reducing the problem of classification in the class $\mathcal{P}(\kappa, c, C)$ to



(a) (b)

Figure 2.4: The two conditional distributions used for the proof of Theorem 1.

a hypothesis testing problem. In this case it suffices to consider two hypothesis and use the following result from [32] (page 76, theorem 2.2).

Theorem 5 (Tsybakov 2004)**.** *Let $\mathcal{F}$ be a class of models. Associated with each model $f \in \mathcal{F}$ we have a probability measure P_f defined on a common probability space. Let $d(\cdot,\cdot) : \mathcal{F}\times\mathcal{F} \to \mathbb{R}$ be a semi-distance. Let $f_0, f_1 \in \mathcal{F}$ be such that $d(f_0, f_1) \geq 2a$, with $a > 0$. Assume also that $\mathrm{KL}(P_{f_1} \| P_{f_0}) \leq \gamma$, where* KL *denotes the Kullback-Leibler divergence*[3]*. The following bound holds.*

$$\begin{aligned} \inf_{\widehat{f}} \sup_{f\in\mathcal{F}} P_f\left(d(\widehat{f}, f) \geq a\right) &\geq \inf_{\widehat{f}} \max_{j\in\{0,1\}} P_{f_j}\left(d(\widehat{f}, f_j) \geq a\right) \\ &\geq \max\left(\frac{1}{4}\exp(-\gamma), \frac{1-\sqrt{\gamma/2}}{2}\right) , \end{aligned}$$

[3] *Let P and Q be two probability measures defined on a common probability space. The Kullback-Leibler divergence is defined as*

$$\mathrm{KL}(P\|Q) = \begin{cases} \int \log \frac{\mathrm{d}P}{\mathrm{d}Q} \mathrm{d}P & \text{, if } P \ll Q, \\ +\infty & \text{, otherwise.} \end{cases} ,$$

where $\mathrm{d}P/\mathrm{d}Q$ is the Radon-Nikodym derivative of measure P with respect to measure Q.

where the infimum is taken with respect to the collection of all possible estimators of f (based on a sample from P_f).

To prove the statement of Theorem 1 we take $\mathcal{F} = \mathcal{P}(\kappa, c, C)$ and are interested in controlling the excess risk

$$R_P(\widehat{G}_n) - R_P(G_P^*) = \int_{\widehat{G}_n \Delta G_P^*} |2\eta_P(x) - 1| \mathrm{d}x \ ,$$

where the subscript P indicates that the excess risk is being measured with respect to the distribution $P \in \mathcal{P}(\kappa, c, C)$. Since the excess risk is not a semi-distance we cannot apply Theorem 5 directly, but we can relate excess risk and the symmetric distance measure, and then use the theorem. The two distributions/hypotheses we consider are completely characterized by the conditional probability η (since the marginal distribution of X is uniform). Let

$$\eta_0(x) = \begin{cases} \min\left(\frac{1}{2} + c\,\mathrm{sign}(x-t)|x-t|^{\kappa-1}, 1\right) & , x \le A \\ \min\left(\frac{1}{2} + c\,x^{\kappa-1}, 1\right) & , x > A \end{cases} , \tag{2.15}$$

$$\eta_1(x) = \min\left(\frac{1}{2} + c\,x^{\kappa-1}, 1\right) \ , \tag{2.16}$$

where $A = t\left(1 + \frac{1}{(C/c)^{1/(\kappa-1)} - 1}\right)$. These are depicted in Figure 2.4 when $C = \infty$. Note that $G_0^* = [t, 1]$ and $G_1^* = [0, 1]$ (provided t is small enough). In what follows we use the subscript 0 or 1 whenever we want to denote explicitly the dependence on the underlying model (respectively characterized by η_0 or η_1). Begin by observing that

the two constructed distribution belong to the class $\mathcal{F} = \mathcal{P}(\kappa, c, C)$ of interest, that is, these two distributions satisfy the margin conditions (2.2) and (2.3). Another key observation is that, for any set $G \subseteq [0,1]$ and $j \in \{0,1\}$ we have

$$R_j(G) - R_j(G_j^*) \geq \frac{4c}{\kappa 2^\kappa} d_\Delta(G, G_j^*)^\kappa \ , \tag{2.17}$$

where $d_\Delta(G, G_j^*) \stackrel{\Delta}{=} \int_{G \Delta G_j^*} \mathrm{d}x$ is the symmetric difference semi-distance. To see this we consider the case $j = 0$ (the case $j = 1$ is analogous). Let G be such that $d_\Delta(G, G_0^*) = \tau$. Then

$$\begin{aligned} R_j(G) - R_j(G_j^*) = \int_{G \Delta G_0^*} |2\eta_0(x) - 1| \mathrm{d}x &\geq \int_{G \Delta G_0^*} 2c|t-x|^{\kappa-1} \mathrm{d}x \\ &\geq \int_{t-\tau/2}^{t+\tau/2} 2c|t-x|^{\kappa-1} \mathrm{d}x \\ &= 2\int_{t}^{t+\tau/2} 2c(x-t)^{\kappa-1} \mathrm{d}x \\ &= \frac{4c}{\kappa 2^\kappa} \tau^\kappa \ . \end{aligned}$$

We now proceed by applying Theorem 5 to the semi-distance d_Δ and posteriorly use (2.17) to control the excess risk. Begin by noting that $d_\Delta(G_0^*, G_1^*) = t$. Define $P_{0,n} \stackrel{\Delta}{=} P^{(0)}_{X_1,\ldots,X_n,Y_1,\ldots,Y_n}$, the probability measure of the random variables $\{X_i, Y_i\}_{i=1}^n$ under hypothesis 0 and define analogously $P_{1,n} \stackrel{\Delta}{=} P^{(1)}_{X_1,\ldots,X_n,Y_1,\ldots,Y_n}$. Define $\boldsymbol{Z}_j^X \stackrel{\Delta}{=}$

$(X_1, \ldots, X_j)$ and $\boldsymbol{Z}_j^Y \triangleq (Y_1, \ldots, Y_j)$. Then

$$\begin{aligned}
\mathrm{KL}(P_{1,n} \| P_{0,n}) &= \mathbb{E}_1 \left[\log \frac{P^{(1)}_{\boldsymbol{Z}_n^X, \boldsymbol{Z}_n^Y}(\boldsymbol{Z}_n^X, \boldsymbol{Z}_n^Y)}{P^{(0)}_{\boldsymbol{Z}_n^X, \boldsymbol{Z}_n^Y}(\boldsymbol{Z}_n^X, \boldsymbol{Z}_n^Y)} \right] \\
&= \mathbb{E}_1 \left[\log \frac{\prod_{j=1}^n P^{(1)}_{Y_j|X_j}(Y_j|X_j) \; P_{X_j|\boldsymbol{Z}_{j-1}^X, \boldsymbol{Z}_{j-1}^Y}(X_j|\boldsymbol{Z}_{j-1}^X, \boldsymbol{Z}_{j-1}^Y)}{\prod_{j=1}^n P^{(0)}_{Y_j|X_j}(Y_j|X_j) \; P_{X_j|\boldsymbol{Z}_{j-1}^X, \boldsymbol{Z}_{j-1}^Y}(X_j|\boldsymbol{Z}_{j-1}^X, \boldsymbol{Z}_{j-1}^Y)} \right] \qquad (2.18) \\
&= \mathbb{E}_1 \left[\log \frac{\prod_{j=1}^n P^{(1)}_{Y_j|X_j}(Y_j|X_j)}{\prod_{j=1}^n P^{(0)}_{Y_j|X_j}(Y_j|X_j)} \right] \\
&= \sum_{j=1}^n \mathbb{E}_1 \left[\log \frac{P^{(1)}_{Y_j|X_j}(Y_j|X_j)}{P^{(0)}_{Y_j|X_j}(Y_j|X_j)} \right] \\
&\leq n \max_{x \in [0,1]} \mathbb{E}_1 \left[\log \frac{P^{(1)}_{Y_1|X_1}(Y_1|X_1)}{P^{(0)}_{Y_1|X_1}(Y_1|X_1)} \middle| X_1 = x \right] ,
\end{aligned}$$

where in the above $\mathbb{E}_1$ denotes the expectation taken with respect to measure $P_{1,n}$. Step (2.18) follows since the distribution of X_j conditional on $\boldsymbol{Z}_{j-1}^X, \boldsymbol{Z}_{j-1}^Y$ depends only on the sampling strategy S_n, and does not change with the underlying distribution, therefore those terms in the numerator and denominator cancel out. The last step follows from the observation that, conditional on the feature vectors, the labels Y_j are independent and identically distributed. The expectation in the last line is the Kullback-Leibler divergence between two Bernoulli random-variables. The following straightforward result provides a bound on that divergence.

Lemma 1. *Let P and Q be Bernoulli random variables with parameters respectively $1/2 - p$ and $1/2 - q$. Let $|p|, |q| \leq 1/4$, then $\mathrm{KL}(P\|Q) \leq 8(p-q)^2$.*

We conclude that

$$\begin{aligned}\mathrm{KL}(P_{1,n}\|P_{0,n}) &\leq 8n(2cA^{\kappa-1})^2 = 32c^2\left(1+\frac{1}{(C/c)^{1/(\kappa-1)}-1}\right)^{2\kappa-2} nt^{2\kappa-2} \\ &= c_0\cdot nt^{2\kappa-2}\ ,\end{aligned}$$

provided A (consequently t) is small enough, so that Lemma 1 is applicable. In the above expression we have $c_0 \stackrel{\Delta}{=} 32c^2\left(1+\frac{1}{(C/c)^{1/(\kappa-1)}-1}\right)^{2\kappa-2}$, a constant factor.

Taking $t = n^{-\frac{1}{2\kappa-2}}$ and using Theorem 5 we conclude that for n large enough (implying t small).

$$\inf_{\widehat{G}_n}\max_{j\in\{0,1\}} P_j\left(d_\Delta(\widehat{G}_n, G_j^*)\geq t/2\right)\geq \frac{1}{4}\exp(-c_0)>0\ ,$$

We can now use (2.17) to conclude that

$$P_j\left(R_j(G)-R_j(G_j^*)\geq\frac{4c}{\kappa 2^\kappa}(t/2)^\kappa\right)\geq P_j(d_\Delta(G,G_j^*)\geq t/2)\ ,$$

and so

$$\begin{aligned}&\inf_{\widehat{G}_n}\sup_{P_{XY}\in\mathcal{P}(\kappa,a)} P_j\left(R(\widehat{G}_n)-R(G^*)\geq\frac{4c}{\kappa 4^k}\cdot n^{-\frac{\kappa}{2\kappa-2}}\right) \\ &\quad\geq \inf_{\widehat{G}_n}\max_{j\in\{0,1\}} P_j\left(R_j(\widehat{G}_n)-R_j(G^*)\geq\frac{4c}{\kappa 4^k}\cdot n^{-\frac{\kappa}{2\kappa-2}}\right) \\ &\quad\geq \inf_{\widehat{G}_n}\max_{j\in\{0,1\}} P_j(d_\Delta(G,G_j^*)\geq t/2)\geq\frac{1}{4}\exp(-c_0)>0\ ,\end{aligned}$$

The statement of the theorem follows from the application of Markov's inequality to the above expression,

$$\mathbb{E}\left[R(\widehat{G}_n) - R(G^*)\right] \;\geq\; \frac{4c}{\kappa 4^k} n^{-\frac{\kappa}{2\kappa-2}} P\left(R(\widehat{G}_n) - R(G^*) \geq \frac{4c}{\kappa 4^k} n^{-\frac{\kappa}{2\kappa-2}}\right) .$$

Remark: Notice that, when bounding the Kullback-Leibler divergence, we considered all the feature examples to be taken at the most beneficial sampling location, in order to maximize the KL divergence. If instead we assume X_i i.i.d. uniformly over $[0,1]$ the Kullback divergence is approximately proportional to $nt^{2\kappa-2} \cdot t = nt^{2\kappa-1}$, since roughly only a fraction $A \sim t$ of the samples are informative (any sample taken in $(A,1]$ is non-informative). Taking $t \sim n^{-1/(2\kappa-1)}$ and proceeding as before yields the passive sampling minimax bound (2.4). □

2.6.2 Proof of Theorem 3

As the proof of Theorem 1, the following proof uses standard techniques for the most part, but to get the bounds desired we need now more than only two hypotheses. The main tool is the following theorem, from [32] (page 85, theorem 2.5).

Theorem 6 (Tsybakov, 2004)**.** *Let $\mathcal{F}$ be a class of models. Associated with each model $f \in \mathcal{F}$ we have a probability measure P_f defined on a common probability space. Let $M \geq 2$ be an integer and let $d(\cdot,\cdot) : \mathcal{F} \times \mathcal{F} \to \mathbb{R}$ be a collection of semi-distances. Suppose we have $\{f_0, \ldots, f_M\} \in \mathcal{F}$ such that*

i) $d(f_j, f_k) \geq 2\,a > 0, \quad \forall_{0 \leq j,k \leq M}$,

ii) $P_{f_0} \ll P_{f_j}, \quad \forall_{j=1,\ldots,M}$, *(see footnote*[4]*)*

iii) $\frac{1}{M}\sum_{j=1}^{M} \mathrm{KL}(P_{f_j} \| P_{f_0}) \leq \gamma \log M$, *where* $0 < \gamma < 1/8$.

The following bound holds.

$$\begin{aligned} \inf_{\widehat{f}} \sup_{f \in \mathcal{F}} P_f \left(d(\widehat{f}, f) \geq a \right) &\geq \inf_{\widehat{f}} \max_{j \in \{0,\ldots,M\}} P_{f_j} \left(d(\widehat{f}, f_j) \geq a \right) \\ &\geq \frac{\sqrt{M}}{1+\sqrt{M}} \left(1 - 2\gamma - 2\sqrt{\frac{\gamma}{\log M}} \right) > 0 \ , \end{aligned}$$

where the infimum is taken with respect to the collection of all possible estimators of f *(based on a sample from* P_f*).*

As in the proof of Theorem 1 we are going to construct a bound on the performance of an estimator measured according to d_Δ, the symmetric difference measure. By later relating this semi-distance with the excess risk we obtain the desired lower bound. To apply the theorem we need to construct a finite subset of distributions in $\mathrm{BF}(\alpha, \kappa, L, C, c)$. The elements of this set are distributions $P_{\boldsymbol{X}Y}$ and therefore uniquely characterized by the conditional probability $\eta(\boldsymbol{x}) = \Pr(Y = 1 | \boldsymbol{X} = \boldsymbol{x})$ (since we are assuming that $P_{\boldsymbol{X}}$ is uniform over $[0,1]^d$). Let $\boldsymbol{x} = (\tilde{\boldsymbol{x}}, x_d)$ with $\tilde{\boldsymbol{x}} \in [0,1]^{d-1}$. As a notational convention we use a tilde to denote a vector of dimension $d-1$. Let $\lceil x \rceil$ denote the minimal integer such that $x > \lceil x \rceil$ and define

$$m = \left\lceil c_0 n^{\frac{1}{\alpha(2\kappa-2)+d-1}} \right\rceil \ , \quad \tilde{\boldsymbol{x}}_{\tilde{\boldsymbol{l}}} = \frac{\tilde{\boldsymbol{l}} - 1/2}{m} \ ,$$

[4] *Let* P *and* Q *be two probability measures defined on a common probability space* $(\Omega, \mathcal{B})$*. Then* $P \ll Q$ *if and only if for all* $B \in \mathcal{B}$*,* $Q(B) = 0 \Rightarrow P(B) = 0$.

where $\tilde{\boldsymbol{l}} \in \{1, \ldots, m\}^{d-1}$ and $c_0 > 0$ is to be determined later. Define also $\varphi_{\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) = Lm^{-\alpha} h(m(\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{x}}_{\tilde{\boldsymbol{l}}}))$, with $h \in \Sigma_{d-1}(1, \alpha)$, $\mathrm{supp}(h) = (-1/2, 1/2)^{d-1}$ and $h \geq 0$. It is easily shown that such a function exists, for example

$$h(\tilde{\boldsymbol{x}}) = a \prod_{i=1}^{d-1} \exp\left(-\frac{1}{1 - 4x_i^2}\right) \mathbf{1}\{|x_i| < 1/2\} \ ,$$

with $a > 0$ sufficiently small. The functions $\varphi_{\tilde{\boldsymbol{l}}}$ are little "bumps" centered at the points $\tilde{\boldsymbol{x}}_{\tilde{\boldsymbol{l}}}$. The collection $\{\tilde{\boldsymbol{x}}_{\tilde{\boldsymbol{l}}}\}$ forms a regular grid over $[0, 1]^{d-1}$.

Let $\Omega = \{\boldsymbol{\omega} = (\omega_1, \ldots, \omega_{m^{d-1}}),\ \omega_i \in \{0, 1\}\} = \{0, 1\}^{m^{d-1}}$, and define

$$g_{\boldsymbol{\omega}}(\cdot) = \sum_{\tilde{\boldsymbol{l}} \in \{1, \ldots, m\}^{d-1}} \omega_{\tilde{\boldsymbol{l}}}\ \varphi_{\tilde{\boldsymbol{l}}}(\cdot),\ \boldsymbol{\omega} \in \Omega \ .$$

The functions $g_{\boldsymbol{\omega}}$ are boundary functions. The binary vector $\boldsymbol{\omega}$ is an indicator vector: if $\omega_{\tilde{\boldsymbol{l}}} = 1$ then "bump" $\tilde{\boldsymbol{l}}$ is present, otherwise that "bump" is absent. Note that $\varphi_{\tilde{\boldsymbol{l}}} \in \Sigma_{d-1}(L, \alpha)$ and these functions have disjoint support, therefore $g_{\boldsymbol{\omega}} \in \Sigma_{d-1}(L, \alpha)$ for all $\boldsymbol{\omega} \in \Omega$. Let $\boldsymbol{\omega} \in \Omega$ and construct the conditional distribution

$$\eta_{\boldsymbol{\omega}}(\boldsymbol{x}) = \begin{cases} \min\left(\frac{1}{2} + c \cdot \mathrm{sign}(x_d - g_{\boldsymbol{\omega}}(\tilde{\boldsymbol{x}}))|x_d - g_{\boldsymbol{\omega}}(\tilde{\boldsymbol{x}})|^{\kappa - 1}, 1\right), & \text{if } x_d \leq A \\ \min\left(\frac{1}{2} + c \cdot x_d^{\kappa - 1}, 1\right), & \text{if } x_d > A \end{cases} \ ,$$

$$A = \max_{\tilde{\boldsymbol{x}}} \varphi(\tilde{\boldsymbol{x}}) \left(1 + \frac{1}{(C/c)^{1/(\kappa - 1)} - 1}\right) = Lm^{-\alpha} h_{\max} \left(1 + \frac{1}{(C/c)^{1/(\kappa - 1)} - 1}\right) \ ,$$

with $h_{\max} = \max_{\tilde{\boldsymbol{x}} \in \mathbb{R}^{d-1}} h(\tilde{\boldsymbol{x}})$. The choice of A is done carefully, in order to ensure

that the functions $\eta_{\boldsymbol{\omega}}$ are similar, but at the same time satisfy the margin conditions. It is easily checked that conditions (2.10) and (2.11) are satisfied for the distributions above. By construction the Bayes decision boundary for each of these distributions is given by $x_d = g_{\boldsymbol{\omega}}(\tilde{\boldsymbol{x}})$ and so these distributions belong to the class $\mathrm{BF}(\alpha, \kappa, L, C, c)$. Note also that these distributions are all identical if $x_d > A$. As n increases m also increases and therefore A decreases, so the conditional distributions described above are becoming more and more similar. This is key to bound the Kullback-Leibler divergence between these distributions.

The above collection of distributions, indexed by $\boldsymbol{\omega} \in \Omega$, is still too large for the application of Theorem 6. Recall the following lemma.

Lemma 2 (Varshamov-Gilbert bound, 1962)**.** *Let $m^{d-1} \geq 8$. There exists a subset $\{\boldsymbol{\omega}^{(0)}, \boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(M)}\}$ of Ω such that $M \geq 2^{m^{d-1}/8}$, $\boldsymbol{\omega}^{(0)} = (0, \ldots, 0)$ and*

$$\rho(\boldsymbol{\omega}^{(j)}, \boldsymbol{\omega}^{(k)}) \geq m^{d-1}/8, \quad \forall\ 0 \leq j < k \leq M\ ,$$

where ρ denotes the Hamming distance.

For a proof of the Lemma 2 see [32](page 89, lemma 2.7). To apply Theorem 6 we use the M distributions $(\eta_{\boldsymbol{\omega}^{(0)}}, \ldots, \eta_{\boldsymbol{\omega}^{(M)}})$ given by the lemma. For each distribution $\eta_{\boldsymbol{\omega}^{(i)}}$ we have the corresponding Bayes classifier G_i^*. As before recall that $d_\Delta(G, G') =$

$\int_{G\Delta G'} \mathrm{d}\boldsymbol{x}$. Let $i \neq j$. By construction we observe that

$$\begin{aligned}
d_\Delta(G_j^*, G_i^*) &= \int_{[0,1]^{d-1}} \int_0^{|g_i^*(\tilde{\boldsymbol{x}}) - g_j^*(\tilde{\boldsymbol{x}})|} 1 \mathrm{d}x_d \mathrm{d}\tilde{\boldsymbol{x}} \\
&= \sum_{\tilde{\boldsymbol{l}} \in \{1,\ldots,m\}^{d-1}} |\boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(i)} - \boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(j)}| \int_{[0,1]^{d-1}} \int_0^{Lm^{-\alpha} h(m(\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{x}}_{\tilde{\boldsymbol{l}}}))} 1 \mathrm{d}x_d \mathrm{d}\tilde{\boldsymbol{x}} \\
&= \sum_{\tilde{\boldsymbol{l}} \in \{1,\ldots,m\}^{d-1}} |\boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(i)} - \boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(j)}| \int_{[0,1]^{d-1}} Lm^{-\alpha} h(m(\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{x}}_{\tilde{\boldsymbol{l}}})) \mathrm{d}\tilde{\boldsymbol{x}} \\
&= \sum_{\tilde{\boldsymbol{l}} \in \{1,\ldots,m\}^{d-1}} |\boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(i)} - \boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(j)}| \int_{[-1/2,1/2]^{d-1}} Lm^{-\alpha-(d-1)} h(\tilde{\boldsymbol{z}}) \mathrm{d}\tilde{\boldsymbol{z}} \\
&= \sum_{\tilde{\boldsymbol{l}} \in \{1,\ldots,m\}^{d-1}} |\boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(i)} - \boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(j)}| Lm^{-\alpha-(d-1)} \|h\|_1 \\
&\geq \rho(\boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(i)}, \boldsymbol{\omega}_{\tilde{\boldsymbol{l}}}^{(j)}) Lm^{-\alpha-(d-1)} \|h\|_1 \\
&\geq \frac{L\|h\|_1}{8} m^{-\alpha} \ ,
\end{aligned}$$

where $\|h\|_1$ denotes the L_1 norm of h. The next step of the proof is to lower-bound $R_i(G) - R_i(G_i^*)$ using $d_\Delta(G, G_i^*)$, where $G \subseteq [0,1]^d$. Suppose $d_\Delta(G, G_i^*) = \tau$. The smallest excess risk $R_i(G) - R_i(G_i^*)$ is attained when the points in set G coincide with the points $\tilde{\boldsymbol{x}}$ such that $\eta_{\boldsymbol{\omega}^{(i)}}$ is closest to $1/2$. Taking this into account we observe that

$$\begin{aligned}
R_i(G) - R_i(G_i^*) &\geq \int_{[0,1]^{d-1}} \int_{g_i^*(\tilde{\boldsymbol{x}}) - \tau/2}^{g_i^*(\tilde{\boldsymbol{x}}) + \tau/2} 2c|x_d - g_i^*(\tilde{\boldsymbol{x}})|^{\kappa-1} \mathrm{d}x_d \mathrm{d}\tilde{\boldsymbol{x}} \\
&= \int_{[0,1]^{d-1}} 2 \int_{g_i^*(\tilde{\boldsymbol{x}})}^{g_i^*(\tilde{\boldsymbol{x}}) + \tau/2} 2c(x_d - g_i^*(\tilde{\boldsymbol{x}}))^{\kappa-1} \mathrm{d}x_d \mathrm{d}\tilde{\boldsymbol{x}} \\
&= 4c \int_{[0,1]^{d-1}} \int_0^{\tau/2} z^{\kappa-1} \mathrm{d}z \mathrm{d}\tilde{\boldsymbol{x}} = \frac{4c}{\kappa 2^\kappa} \tau^\kappa \ .
\end{aligned}$$

Therefore we conclude that for all $G \subseteq [0,1]^d$ we have

$$R_i(G) - R_i(G_i^*) \geq \frac{4c}{\kappa 2^\kappa} d_\Delta^\kappa(G, G_i^*) \ . \tag{2.19}$$

We are ready for the final step of the proof. Now let P_i be the distribution of $(\boldsymbol{X}_1, Y_1, \ldots, \boldsymbol{X}_n, Y_n)$ assuming the underlying conditional distribution is $\eta_{\boldsymbol{\omega}^{(i)}}$. We proceed like in the proof of Theorem 2.6.1.

$$\begin{aligned} \mathrm{KL}(P_i \| P_0) &\leq n \max_{\boldsymbol{x} \in [0,1]^d} \mathbb{E}_i \left[\log \frac{P^{(i)}_{Y_1|\boldsymbol{X}_1}(Y_1|\boldsymbol{X}_1)}{P^{(0)}_{Y_1|\boldsymbol{X}_1}(Y_1|\boldsymbol{X}_1)} \middle| \boldsymbol{X}_1 = \boldsymbol{x} \right] \\ &\leq 8n(2cA^{\kappa-1})^2 = c_1 \cdot n m^{-\alpha(2\kappa-2)} \ , \end{aligned}$$

where $c_1 > 0$ and the last inequality holds provided n is large enough so that A is small enough and Lemma 1 can be applied. Finally

$$\frac{1}{M} \sum_{i=1}^{M} \mathrm{KL}(P_i \| P_0) \leq c_1 \cdot n m^{-\alpha(2\kappa-2)} \leq c_1 \ c_0^{-(\alpha(2\kappa-2)+d-1)} m^{d-1} \ .$$

From Lemma 2 we also have $\frac{\gamma}{8} m^{d-1} \log 2 \leq \gamma \log M$ therefore choosing c_0 large enough in the definition of m guarantees the conditions of Theorem 6 and so

$$\begin{aligned} &\inf_{\widehat{G}_n, S_n} \max_{j \in \{0,\ldots,M\}} P_j \left(d_\Delta(\widehat{G}_n, G_j^*) \geq \frac{L\|h\|_1}{16} m^{-\alpha} \right) \geq \\ &\quad \inf_{\widehat{G}_n, S_n} \max_{j \in \{0,\ldots,M\}} P_j \left(d_\Delta(\widehat{G}_n, G_j^*) \geq \frac{L\|h\|_1 c_0^{-\alpha}}{16} n^{-\frac{1}{2\kappa-2+(d-1)/\alpha}} \right) \geq c_2 \ , \end{aligned}$$

for n large enough, where $c_2 > 0$ comes from Theorem 6. Now using (2.19) similarly

to the proof of Theorem 1 we obtain

$$\inf_{\widehat{G}_n, S_n} \sup_{P \in \mathrm{BF}_{(\alpha,\kappa,L,C,c)}} P\left(R(\widehat{G}_n) - R(G^*) \geq \frac{c}{4\kappa 2^\kappa} L \|h\|_1 c_0^{-\alpha} \cdot n^{-\frac{\kappa}{2\kappa - 2 + (d-1)/\alpha}}\right) \geq c_2 \ ,$$

An application of Markov's inequality yields the original statement of the theorem, concluding the proof.

Remark: As in the proof of Theorem 1 note that if the passive sampling scenario is considered the sample locations $\{\boldsymbol{X}_i\}_{i=1}^n$ have to be selected before any observations are made, therefore they must be somewhat uniformly distributed over $[0,1]^d$. Using a similar reasoning as remarked in Section 2.6.1 we have that $\mathrm{KL}(P_i \| P_0) \leq 8n(cA^{\kappa-1})^2 L h_{\max} \cdot m^{-\alpha} \sim nm^{-2\alpha(\kappa-1)} m^{-\alpha} \sim nm^{-\alpha(2\kappa-1)}$. Therefore choosing $m \sim n^{\frac{1}{\alpha(2\kappa-1)+d-1}}$ and proceeding in analogous fashion as before yields bound (2.12). □

2.6.3 Proof of Theorem 4

The proof methodology aims at controlling the excess risk for an event that happens with high probability. To avoid carrying around cumbersome constants we use the 'big-O' notation (see the footnote on 44). We show the proof only for the case $\kappa > 1$, since the proof when $\kappa = 1$ is almost analogous.

Define the event $\Omega_n = \left\{|\widehat{g}(M^{-1}\tilde{\boldsymbol{l}}) - g^*(M^{-1}\tilde{\boldsymbol{l}})| \leq t_N \quad \forall \tilde{\boldsymbol{l}} \in \{0, \ldots, M\}^{d-1}\right\}$. In words, Ω_n is the event that the M^{d-1} point estimates of g do not deviate very much from the true values. Using a union bound, taking into account (2.9) and the choice t_N in (2.13) one sees that $1 - \Pr(\Omega_n) = O(N^{-\gamma} M^{d-1})$, where γ can be chosen arbitrarily

large. With the choice of M in the theorem and choosing c_1 wisely in the definition of t_N (2.13) we have $1 - \Pr(\Omega_n) = O\left(n^{-\frac{\alpha\kappa}{\alpha(2\kappa-2)+d-1}}\right)$.

The excess risk of our classifier is given by

$$\begin{aligned} R(\widehat{G}_n) - R(G^*) &= \int_{\widehat{G}_n \Delta G^*} |2\eta(\boldsymbol{x}) - 1| \mathrm{d}\boldsymbol{x} \\ &= \int_{[0,1]^{d-1}} \int_{\min(\widehat{g}(\tilde{\boldsymbol{x}}), g^*(\tilde{\boldsymbol{x}}))}^{\max(\widehat{g}(\tilde{\boldsymbol{x}}), g^*(\tilde{\boldsymbol{x}}))} |2\eta\left((\tilde{\boldsymbol{x}}, x_d)\right) - 1| \mathrm{d}x_d \mathrm{d}\tilde{\boldsymbol{x}} \\ &\leq \int_{[0,1]^{d-1}} \int_{\min(\widehat{g}(\tilde{\boldsymbol{x}}), g^*(\tilde{\boldsymbol{x}}))}^{\max(\widehat{g}(\tilde{\boldsymbol{x}}), g^*(\tilde{\boldsymbol{x}}))} \frac{2C}{\delta} |x_d - g(\tilde{\boldsymbol{x}})|^{\kappa-1} \mathrm{d}x_d \mathrm{d}\tilde{\boldsymbol{x}} \\ &= \frac{2C}{\delta} \int_{[0,1]^{d-1}} \int_0^{|\widehat{g}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}})|} z^{\kappa-1} \mathrm{d}z \mathrm{d}\tilde{\boldsymbol{x}} \\ &= \frac{2C}{\delta\kappa} \int_{[0,1]^{d-1}} |\widehat{g}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}})|^{\kappa} \mathrm{d}\tilde{\boldsymbol{x}} = O\left(\|\widehat{g} - g^*\|_\kappa^\kappa\right) \ , \end{aligned}$$

where the inequality follows from condition (2.11), and $\|\cdot\|_\kappa$ denotes the L_κ norm of a function.

Let $L_{\tilde{\boldsymbol{q}}}$, $\tilde{\boldsymbol{q}} \in \{0, \ldots, M/\lfloor\alpha\rfloor - 1\}^{d-1}$ be a clairvoyant version of $\widehat{L}_{\tilde{\boldsymbol{q}}}$, that is,

$$L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) = \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} g^*(M^{-1}\tilde{\boldsymbol{l}}) Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \ .$$

In a sense $L_{\tilde{\boldsymbol{q}}}$ is the "best" classifier is the class of piecewise polynomial classifiers. It is well known that these interpolating polynomials have good local approximation properties for Hölder smooth functions, namely we have that

Lemma 3.

$$\sup_{g \in \Sigma_{d-1}(L,\alpha)} \max_{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}} |L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}})| = O(M^{-\alpha}) \ . \tag{2.20}$$

Lemma 3 is proved in the end of the section. We have almost all the pieces we need to conclude the proof. The last fact needed is a bound on the variation of the tensor-product Lagrange polynomials, namely it is easily shown that

$$\max_{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}} \left| Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \right| \leq \lfloor \alpha \rfloor^{(d-1)\lfloor \alpha \rfloor} \ . \tag{2.21}$$

We are now ready to show the final result. Assume for now that Ω_n holds, therefore $|\widehat{g}(M^{-1}\tilde{\boldsymbol{l}}) - g^*(M^{-1}\tilde{\boldsymbol{l}})| \leq t_N$ for all $\tilde{\boldsymbol{l}}$. Note that t_N is decreasing as n (and consequently N) increase.

$$\begin{aligned}
R(\widehat{G}_n) - R(G^*) &= O\left(\|\widehat{g} - g^*\|_\kappa^\kappa \right) \\
&= O\left(\sum_{\tilde{\boldsymbol{q}} \in \{0,\ldots,M/\lfloor\alpha\rfloor - 1\}^{d-1}} \left\| (\widehat{L}_{\tilde{\boldsymbol{q}}} - g^*) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} \right\|_\kappa^\kappa \right) \\
&= O\left(\sum_{\tilde{\boldsymbol{q}}} \left\| (L_{\tilde{\boldsymbol{q}}} - g^*) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} + (\widehat{L}_{\tilde{\boldsymbol{q}}} - L_{\tilde{\boldsymbol{q}}}) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} \right\|_\kappa^\kappa \right) \\
&= O\left(\sum_{\tilde{\boldsymbol{q}}} \left(\left\| (L_{\tilde{\boldsymbol{q}}} - g^*) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} \right\|_\kappa + \left\| (\widehat{L}_{\tilde{\boldsymbol{q}}} - L_{\tilde{\boldsymbol{q}}}) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} \right\|_\kappa \right)^\kappa \right) \ ,
\end{aligned}$$

Note now that

$$\begin{aligned}
\left\| (L_{\tilde{\boldsymbol{q}}} - g^*) \mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\} \right\|_\kappa &= \left(\int_{I_{\tilde{\boldsymbol{q}}}} \left(L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}}) \right)^\kappa \mathrm{d}\tilde{\boldsymbol{x}} \right)^{1/\kappa} \\
&= O\left(\left(\int_{I_{\tilde{\boldsymbol{q}}}} M^{-\alpha\kappa} \mathrm{d}\tilde{\boldsymbol{x}} \right)^{1/\kappa} \right) = O\left(M^{-\alpha} M^{-\frac{d-1}{\kappa}} \right) \ .
\end{aligned}$$

Where we used Lemma 3. We have also

$$\begin{aligned}
\left\|(\widehat{L}_{\tilde{\boldsymbol{q}}} - L_{\tilde{\boldsymbol{q}}})\mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\}\right\|_\kappa &= \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} \left|\widehat{g}(M^{-1}\tilde{\boldsymbol{l}}) - g^*(M^{-1}\tilde{\boldsymbol{l}})\right| \left\|Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}\right\|_\kappa \\
&\leq \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} t_N \left(\int_{I_{\tilde{\boldsymbol{q}}}} \left|Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}})\right|^\kappa \mathrm{d}\tilde{\boldsymbol{x}}\right)^{1/\kappa} \\
&\leq \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} t_N \left(\int_{I_{\tilde{\boldsymbol{q}}}} \lfloor\alpha\rfloor^{(d-1)\lfloor\alpha\rfloor\kappa} \mathrm{d}\tilde{\boldsymbol{x}}\right)^{1/\kappa} = O\left(t_N M^{-(d-1)/\kappa}\right) .
\end{aligned}$$

Using these two facts we conclude that

$$\begin{aligned}
&R(\widehat{G}_n) - R(G^*) = \\
&\quad O\left(\sum_{\tilde{\boldsymbol{q}}} \left(\left\|(L_{\tilde{\boldsymbol{q}}} - g^*)\mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\}\right\|_\kappa + \left\|(\widehat{L}_{\tilde{\boldsymbol{q}}} - L_{\tilde{\boldsymbol{q}}})\mathbf{1}\{\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}\}\right\|_\kappa\right)^\kappa\right) \\
&= O\left(\sum_{\tilde{\boldsymbol{q}} \in \{0,\ldots,M/\lfloor\alpha\rfloor - 1\}^{d-1}} \left(M^{-\alpha} M^{-\frac{d-1}{\kappa}} + t_N M^{-(d-1)/\kappa}\right)^\kappa\right) \\
&= O\left(M^{d-1}\left(M^{-\alpha} M^{-\frac{d-1}{\kappa}} + t_N M^{-(d-1)/\kappa}\right)^\kappa\right) \\
&= O\left(\left(M^{-\alpha} + t_N\right)^\kappa\right) .
\end{aligned}$$

Plugging in the choices of M and N given in the theorem statement we obtain

$$R(\widehat{G}_n) - R(G^*) = O\left((\log n/n)^{\frac{\alpha\kappa}{\alpha(2\kappa-2)+d-1}}\right) .$$

Finally, noticing that $1 - \Pr(\Omega_n) = O\left(n^{-\frac{\alpha\kappa}{\alpha(2\kappa-2)+d-1}}\right)$ we have

$$\begin{aligned}\mathbb{E}[R(\widehat{G}_n)] - R(G^*) &\leq O\left((\log n/n)^{\frac{\alpha\kappa}{\alpha(2\kappa-2)+d-1}}\right)\Pr(\Omega_n) + 1\cdot(1-\Pr(\Omega_n)) \\ &= O\left((\log n/n)^{\frac{\alpha\kappa}{\alpha(2\kappa-2)+d-1}}\right) ,\end{aligned}$$

concluding the proof. □

2.6.4 Proof of Lemma 3

Let $\tilde{\boldsymbol{x}} \in I_{\tilde{\boldsymbol{q}}}$ and $g \in \Sigma_{d-1}(L,\alpha)$. Taking into account Definition 1 we have

$$\begin{aligned}|L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}})| &= \left|L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}}) + \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}})\right| \\ &\leq \left|L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}})\right| + \left|g^*(\tilde{\boldsymbol{x}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}})\right| \\ &\leq \left|L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}})\right| + L\left\|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}\right\|^{\alpha} \\ &\leq \left|L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}})\right| + O(M^{-\alpha}) .\end{aligned}$$

Note now that the tensor polynomial approximation space contains the space of degree $\lfloor\alpha\rfloor$ polynomials, therefore we can write $L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}})$ as a tensor product polynomial and

so

$$
\begin{aligned}
|L_{\tilde{\boldsymbol{q}}}(\tilde{\boldsymbol{x}}) - g^*(\tilde{\boldsymbol{x}})| &\leq \left| \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} g^*(M^{-1}\tilde{\boldsymbol{l}}) Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(\tilde{\boldsymbol{x}}) \right| + O(M^{-\alpha}) \\
&= \left| \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} \left(g^*(M^{-1}\tilde{\boldsymbol{l}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(M^{-1}\tilde{\boldsymbol{l}}) \right) Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \right| + O(M^{-\alpha}) \\
&\leq \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} \left| g^*(M^{-1}\tilde{\boldsymbol{l}}) - \mathrm{TP}_{\tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1}}(M^{-1}\tilde{\boldsymbol{l}}) \right| \left| Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \right| + O(M^{-\alpha}) \\
&\leq \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} L \left\| \tilde{\boldsymbol{x}} - \tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1} \right\|^{\alpha} \left| Q_{\tilde{\boldsymbol{q}},\tilde{\boldsymbol{l}}}(\tilde{\boldsymbol{x}}) \right| + O(M^{-\alpha}) \qquad (2.22) \\
&\leq \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} L \left\| \tilde{\boldsymbol{x}} - \tilde{\boldsymbol{q}}\lfloor\alpha\rfloor M^{-1} \right\|^{\alpha} \lfloor\alpha\rfloor^{(d-1)\lfloor\alpha\rfloor} + O(M^{-\alpha}) \\
&\leq \sum_{\tilde{\boldsymbol{l}}: M^{-1}\tilde{\boldsymbol{l}} \in I_{\tilde{\boldsymbol{q}}}} O(M^{-\alpha}) + O(M^{-\alpha}) \\
&= \lfloor\alpha\rfloor^{d-1} O(M^{-\alpha}) + O(M^{-\alpha}) = O(M^{-\alpha}) \ ,
\end{aligned}
$$

where we applied Definition 1 again, and in step (2.22) we used (2.21). Finally the last step follows from the observation that the number of terms in the summation is $\lfloor\alpha\rfloor^{d-1}$, which does not depend on M. □

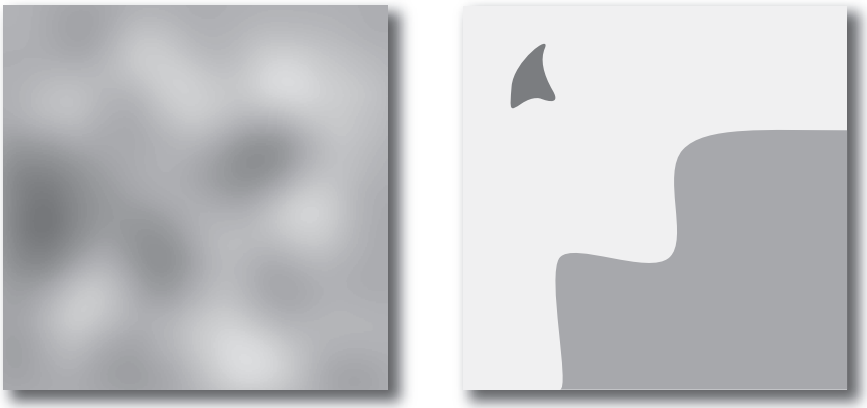
Chapter 3

Regression of Piecewise Constant Functions

In this chapter we consider active learning in a regression setting. Compared to Chapter 2 the results presented here are on one hand more restrictive (we consider solely piecewise constant functions, akin to noise parameter $\kappa = 1$) but also broader since the boundary model considered is much more general than a boundary fragment. The focus of this chapter is to present a practical algorithm for active learning, that works under reasonable practical assumptions, and has provable performance guarantees. The main goal of non-parametric regression is to estimate a function, belonging to a potentially large class, from noisy point-wise samples. In the classical setting the sample locations are chosen *a priori*, that is, the selection of sample locations precedes the gathering of the function observations. In the active setting considered in this chapter, however, the sample locations are chosen in an online fashion: the decision of where to sample next depends on all the observations made up to that point.

3.1 Introduction

In the regression setting, significantly faster rates of error decay are expected to be achievable using active sampling in cases involving function classes whose complexity (in the Kolmogorov sense) is dominated by the complexity of lower dimensional objects. This is the case, for example, for functions that are smooth or slowly varying, apart from highly localized abrupt changes such as jumps or edges. We illustrate this behavior by characterizing the fundamental limits of active sampling for two broad nonparametric function classes which map $[0,1]^d$ to the real line: (i) Hölder smooth functions (spatially homogeneous complexity, see Figure 3.1(a)) and (ii) piecewise constant functions that are constant except on a $(d-1)$-dimensional *boundary set* or discontinuity in $[0,1]^d$ (spatially concentrated complexity, see Figure 3.1(b)). We conclude that, when the functions are spatially homogeneous and smooth, passive learning algorithms are minimax optimal over all estimation methods and all (active or passive) sampling schemes, indicating that active learning methods will not lead to faster rates of convergence in this setting. For piecewise constant functions, active sampling techniques can capitalize on the highly localized nature of the boundary by focusing the sampling process in the estimated vicinity of the boundary. These gains were seen in Chapter 2, for boundary fragment classes. Here we consider a much more general boundary model, that does not require the functional description used by boundary fragments. We present an active learning method that provably improves on the best possible performance based on conventional passive sampling and



(a) (b)

Figure 3.1: Examples of functions in the classes considered: (a) Hölder smooth function. (b) Piecewise constant function.

which achieves faster error convergence rates for the piecewise constant class. Furthermore, we show that no other active sampling method can significantly improve upon this performance (in a minimax sense). The proposed methodology is based on a multiscale coarse-to-fine strategy that employs techniques from the theory of spatially-adaptive estimation schemes such as wavelet thresholding. Active sampling brings a similar adaptive and decision-theoretic approach to bear on the process of data collection itself by exploiting the well-known property that singularities tend to "persist-across-scale" [39].

These adaptive sampling theory and methods show promise for a number of practical applications. In particular, in imaging techniques such as laser scanning [1], it is possible to adaptively vary the scanning process. Adaptive sampling in this context can significantly reduce image acquisition times. The techniques developed in this thesis were use in [1] to greatly reduce the number of samples needed for the

reconstruction of images using ballistic photons. Wireless sensor networks constitute another key application area. Because of necessarily limited energy resources, it is desirable to limit the number of measurements collected as much as possible. Incorporating adaptive sampling strategies into such systems can dramatically lengthen the lifetime of the system. In fact, active sampling problems like the one posed in Section 3.4.2 have already found application in fault line detection [24] and boundary estimation in wireless sensor networking [6].

3.2 Problem Formulation

Let $\mathcal{F}$ denote a generic class of functions mapping $[0,1]^d$ to the real line. Later we will consider particular classes $\mathcal{F}$. Let $f : [0,1]^d \to \mathbb{R}$ be a function in that class. Our goal is to estimate this *unknown* function from a finite number of noise-corrupted samples. We consider two different scenarios: (a) *passive sampling*, where the locations of the sample points are chosen statistically independently from the measurement outcomes, and (b) *active sampling*, where the location of the i^{th} sample point can be chosen as a function of the previous samples locations and observations. The statistical model we consider builds on the following assumptions, similar to the ones introduced in Chapter 2:

B1 - The observations $\{Y_i\}_{i=1}^n$ are given by

$$Y_i = f(\boldsymbol{X}_i) + W_i,\ i \in \{1, \ldots, n\}\ ,$$

where the random variables W_i are independent and identically distributed (i.i.d.) and independent of $\{\boldsymbol{X}_i\}_{i=1}^n$.

B2 - The random variables W_i are Gaussian with zero mean and variance σ^2.

B3.1 - Passive Sampling: Each sample location $\boldsymbol{X}_i \in [0,1]^d$ may be either deterministic or random, but is independent of $\{Y_j\}_{j\in\{1,\ldots,i-1,i+1,\ldots,n\}}$. They do not depend in any way on f.

B3.2 - Active Sampling: Each sample location $\boldsymbol{X}_i$ is random, with a distribution that depends only on $\{\boldsymbol{X}_j, Y_j\}_{j=1}^{i-1}$. In other words

$$\boldsymbol{X}_i = h(\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1}, \boldsymbol{U}_i) \ ,$$

where $h(\cdot)$ is a deterministic function, and $\boldsymbol{U}_i$ accounts for possible randomization of the sampling rule (that is $\boldsymbol{U}_i$ is a random variable, independent of $(\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1})$.) Finally given $\{\boldsymbol{X}_j, Y_j\}_{j=1}^{i-1}$ the random variable $\boldsymbol{X}_i$ does not depend in any way on f.

Note that (B3.1) and (B3.2) are identical to (A2.1) and (A2.2) respectively (see page 22). Therefore the passive sampling strategy is a special case of active sampling. For the scenarios addressed in this chapter we assume either (B3.1) or (B3.2) holds.

The performance metric considered is the usual squared L_2 norm,

$$\|f - g\|^2 = \int_{[0,1]^d} |f(\boldsymbol{x}) - g(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \ ,$$

where $f, g : [0,1]^d \to \mathbb{R}$. An estimator is a function $\widehat{f}_{\boldsymbol{X}_1,\ldots,\boldsymbol{X}_n,Y_1\ldots,Y_n} : [0,1]^d \to \mathbb{R}$. Given $\{\boldsymbol{X}_i, Y_i\}_{i=1}^n$, $\widehat{f}_{\boldsymbol{X}_1,\ldots,\boldsymbol{X}_n,Y_1\ldots,Y_n}(\cdot)$ is a function mapping $[0,1]^d$ to the real line. As before we will usually drop the explicit dependence of the estimator on $\{\boldsymbol{X}_i, Y_i\}_{i=1}^n$, and denote the estimator by $\widehat{f}_n(\cdot)$, where the subscript n denotes the dependency on the n data points. When choosing an estimator $\widehat{f}_n$ our main concern is to ensure that $\|\widehat{f}_n - f\|^2$ is small.

Besides the construction of the estimator $\widehat{f}_n$, our framework allows also another degree of freedom: we can choose a *sampling strategy*, that is, a rule indicating where to collect the next sample. To be more formal, when working under assumption (B3.2) we need to specify the distribution of $\boldsymbol{X}_i$ given $\boldsymbol{X}_{i-1}, \ldots, \boldsymbol{X}_1, Y_{i-1}, \ldots, Y_1$. This is called the sampling strategy for sample i, and it is completely specified by $h(\cdot)$ and $\boldsymbol{U}_i$. The collection of the n sampling distributions (one for each sample) is called the *sampling strategy* and denoted by S_n. The pair $(\widehat{f}_n, S_n)$ is called the *estimation strategy*. Note that even under assumption (B3.1) we have to define a sampling strategy (this is the classical experiment design problem), although in this case the strategy cannot depend on $\{Y_j\}$. We use the term *active learning* (resp. passive learning) method to denote estimation strategies that rely on active (resp. passive) sampling strategies.

In the rest of the chapter we study the fundamental performance limits of active learning for certain classes of functions, and describe practical estimation strategies that nearly achieve those fundamental limits. As discussed in the introduction, the

extra degree of spatial adaptivity under (B3.2) can provide some gains when the functions in the class have well localized features. Some classes we consider have this property. In this case as the number of samples increases we can "focus" the sampling on features that are impairing the estimation performance.

Assumption (B2) can be relaxed. Actually for the bulk of our results we only need the variables W_i to be independent, and their distribution needs to satisfy a certain "moment condition" (see the statement of Theorem 8) controlling the tail behavior of the distribution. Many random variables satisfy that condition (*e.g.*, bounded random variables). To avoid cumbersome derivations we consider only the Gaussian assumption throughout, and just remark that these results can be generalized to other noise models.

We consider essentially two different types of functions: functions that are uniformly smooth; and functions that are piecewise constant, in the sense that these are comprised of constant regions separated by boundaries that have upper box-counting dimension at most $d-1$ [40]. The upper box-counting dimension of a set B is defined using a cover of the set by closed balls of diameter r: Let $N(r)$ denote the minimal number of closed balls of diameter r that are a cover of B. The upper box-counting dimension of B is defined as $\limsup_{r\to 0} -\log N(r)/\log r$. The upper box-counting dimension is also known as the entropy dimension.

For the next definition we need the following concept: a function $f : [0,1]^d \to \mathbb{R}$ is locally constant at a point $\boldsymbol{x} \in [0,1]^d$ if

$$\exists \epsilon > 0 \ : \forall \boldsymbol{y} \in [0,1]^d : \quad \|\boldsymbol{x} - \boldsymbol{y}\| < \epsilon \ \Rightarrow \ f(\boldsymbol{y}) = f(\boldsymbol{x}) \ .$$

Definition 4. *A function $f : [0,1]^d \to \mathbb{R}$ is* piecewise constant *if it is locally constant at any point $\boldsymbol{x} \in [0,1]^d \setminus B(f)$, where $B(f) \subseteq [0,1]^d$ is a set with upper box-counting dimension at most $d-1$. Furthermore let f be uniformly bounded on $[0,1]^d$ (that is, $|f(\boldsymbol{x})| \leq M, \ \forall \boldsymbol{x} \in [0,1]^d$) and let $B(f)$ satisfy $N(r) \leq \beta r^{-(d-1)}$ for all $r > 0$, where $\beta > 0$ is a constant and $N(r)$ is the minimal number of closed balls of diameter r that covers $B(f)$. The set of all piecewise constant functions f satisfying the above conditions is denoted by* $\mathrm{PC}(\beta, M)$.

The concept of box-counting dimension is related the concept of topological dimension of a set [40], and these coincide when the set is "well-behaved". Essentially this condition means that the "boundaries" between the various constant regions are $(d-1)$-dimensional non-fractal curves. We will frequently refer to the set $B(f)$ as the *boundary set*. The bound on $N(r)$ in the above definition leads to a bound on the upper box-counting dimension. That same condition is essentially a bound on the $d-1$ dimensional volume of $B(f)$, controlled by parameter β.

The class $\mathrm{PC}(\beta, M)$ has the main ingredients that make active sampling appealing: a function $f \in \mathrm{PC}(\beta, M)$ is "well-behaved" everywhere, except in the small set $B(f)$. We will see that the critical task for any estimator of f is accurately finding the

location of the boundary $B(f)$.

3.3 Fundamental Limits - Minimax Lower Bounds

In this section we study the fundamental limitations of active learning methodologies. Let $\mathcal{F}$ be a generic function class, and let $(\widehat{f}_n, S_n)$ be an estimation strategy. We are interested in bounds on the *maximal risk* $\sup_{f\in\mathcal{F}} \mathbb{E}_{f,S_n}[\|\widehat{f}_n - f\|^2]$, where $\mathbb{E}_{f,S_n}$ is the expectation with respect to the probability measure of $\{\boldsymbol{X}_i, Y_i\}_{i=1}^n$ induced by model f and sampling strategy S_n. In what follows we will drop this explicit dependence whenever clear from the context. The goal of this section is to find tight lower bounds for the maximal risk, over all possible estimation strategies. That is, we present bounds of the form

$$\inf_{(\widehat{f}_n,S_n)\in\Omega} \sup_{f\in\mathcal{F}} \mathbb{E}_{f,S_n}[\|\widehat{f}_n - f\|^2] \geq c\psi_n^2, \ \forall n \geq n_0 \ , \tag{3.1}$$

where $n_0 \in \mathbb{N}$, $c > 0$ is a constant, ψ_n is a positive sequence converging to zero, and Ω is the set of all estimation strategies. The sequence ψ_n^2 is denoted as a *lower rate of convergence* (Clearly ψ_n^2 is defined up to a bounded factor, that might depend on n).

It is also possible to devise upper bounds on the maximal risk. These are usually obtained through explicit estimation strategies (as presented in Section 3.4).

3.3.1 Hölder Smooth Functions

In this section we consider classes of functions whose complexity is homogeneous over the entire domain, as in Figure 3.1(a), so that there are no localized features. It is known from the classical works of Ibragimov and Has'minskii [41, 42] that for density estimation in smooth classes (*e.g.*, Sobolev) the choice of design does not improve the rate of estimation. We show that for regression of Hölder smooth functions $\Sigma_d(L, \alpha)$ a similar behavior holds (recall Definition 1 on page 39, and note that we are now considering d dimensional functions). The passive learning model has been studied extensively, and there is a vast statistical literature on the optimal rates of convergence [34, 43]. It turns out that, for the function class in question, the extra flexibility of active sampling does not provide any substantial benefit over passive sampling strategies, since a simple uniform sampling scheme is naturally matched to the homogeneous "distribution" of the target function's complexity.

Theorem 7 (Minimax Lower Bound for $\Sigma_d(L, \alpha)$)**.** *Under the assumptions (B1), (B2) and (B3.2) we have*

$$\inf_{(\widehat{f}_n, S_n) \in \Theta_{ACTIVE}} \sup_{f \in \Sigma_d(L,\alpha)} \mathbb{E}_{f,S_n}[\|\widehat{f}_n - f\|^2] \geq \text{const}(L, \alpha, \sigma^2) n^{-\frac{2\alpha}{2\alpha+d}} \ , \tag{3.2}$$

for n large enough, where $\text{const}(L, \alpha, \sigma^2) > 0$ *and Θ_{ACTIVE} is the set of all active estimation strategies. The above rate is the optimal error rate: there is an estimator*

$\widehat{f}_n$ and a sampling strategy S_n such that

$$\sup_{f \in \Sigma_d(L,\alpha)} \mathbb{E}_{f,S_n}[\|\widehat{f}_n - f\|^2] \leq \text{const}(L, \alpha, \sigma^2) n^{-\frac{2\alpha}{2\alpha+d}} \ , \tag{3.3}$$

where $\text{const}(L, \alpha, \sigma^2) > 0$ *(note that this constant factor might be different than the one above). Furthermore this bound is achieved for a passive sampling strategy with sample locations* $\{X_i\}_{i=1}^n$ *uniformly distributed over* $[0, 1]^d$.

The proof of Theorem 7 is presented in Appendix 3.6.1. Although this result might seem surprising at a first glance, it supports our intuition: estimation using active sampling can only be advantageous if the target functions have spatially localized features. This is not the case for the class $\Sigma_d(L, \alpha)$; these are uniformly smooth functions. Classical approximation theory results also support this intuition: the best m-term approximation scheme for the Hölder class of functions is a linear scheme, using a piecewise polynomial fit. There are various practical estimators achieving the performance rates predicted by Theorem 7, including some based on kernels, splines or wavelets [32].

3.3.2 Piecewise Constant Functions

We now turn our attention to the class of piecewise constant functions $\text{PC}(\beta, M)$. This is a generalization of boundary fragments, such as considered in Chapter 2, and also considered in [34]. In the context of regression the boundary fragment class consists of piecewise constant functions in which the boundary location in the

d^{th} dimension is a Lipschitz function of the first $d-1$ dimensions. Specifically, let $g : [0,1]^{d-1} \to [0,1]$ be a Lipschitz function, that is

$$|g(\boldsymbol{x}) - g(\boldsymbol{z})| \leq \|\boldsymbol{x} - \boldsymbol{z}\|, \ \forall \ \boldsymbol{x}, \boldsymbol{z} \in [0,1]^{d-1} \ . \tag{3.4}$$

Define

$$G = \{(\boldsymbol{x}, y) : 0 \leq y \leq g(\boldsymbol{x}), \ \boldsymbol{x} \in [0,1]^{d-1}\} \ . \tag{3.5}$$

Finally define $f : [0,1]^d \to \mathbb{R}$ by $f(\boldsymbol{x}) = 2M\mathbf{1}_G(\boldsymbol{x}) - M$. The class of all the functions of this form is called the *boundary fragment* class (usually $M = 1$), denoted in this chapter by $\text{BF}(M)$. It is straightforward to to show that $\text{BF}(M) \subseteq \text{PC}(\beta, M)$, for a suitable constant β. Other types of boundary fragments can be considered replacing (3.4) with other conditions, for example enforcing the boundary is smooth in a Hölder sense (as in Chapter 2), or possibly an analytic function [25].

Under the passive learning model we consider in this chapter, we have the following result [34]. Under (B1), (B2) and (B3.1) we have

$$\inf_{(\widehat{f}_n, S_n) \in \Theta_{\text{PASSIVE}}} \sup_{f \in \text{BF}(M)} \mathbb{E}_{f, S_n}[\|\widehat{f}_n - f\|^2] \geq \text{const}(M, \sigma^2) n^{-\frac{1}{d}} \ , \tag{3.6}$$

for n large enough, where $\text{const}(M, \sigma^2) > 0$. It can be shown that the above bound is tight, in the sense that a corresponding upper-bound (2.7) holds and the rate in the theorem is the optimal rate of convergence. This is done in the same spirit as in Chapter 2, reducing the problem of estimating f to multiple change-point detection

problems. Noticing that $\mathrm{BF}(M) \subseteq \mathrm{PC}(\beta, M)$ we obtain

Proposition 2 (Passive Learning Minimax Lower Bound - $\mathrm{PC}(\beta, M)$)**.** *Under (B1), (B2) and (B3.1) we have*

$$\inf_{(\widehat{f}_n, S_n) \in \Theta_{PASSIVE}} \sup_{f \in \mathrm{PC}(\beta, M)} \mathbb{E}_{f, S_n}[\|\widehat{f}_n - f\|^2] \geq \mathrm{const}(\beta, M, \sigma^2) n^{-\frac{1}{d}} \ , \tag{3.7}$$

for n large enough, where $\mathrm{const}(\beta, M, \sigma^2) > 0$ and $\Theta_{PASSIVE}$ is the set of all passive estimation strategies.

It is possible to construct estimators that nearly achieve the above performance rate for piecewise constant functions, as we will see in Section 3.4. All these methods use sample locations $\{\boldsymbol{X}_i\}_{i=1}^n$ that are distributed in a uniform way over $[0,1]^d$ (either in a random or deterministic fashion).

We now turn our attention to the active learning settings. In [36,37] this problem is addressed for the class of boundary fragments. This work is very similar in spirit to what is presented in Chapter 2 and makes use of the results in Burnashev and Zigangirov [23]. Let $d \geq 2$. In [36] it was shown that under (B1), (B2) and (B3.2) we have

$$\inf_{(\widehat{f}_n, S_n) \in \Theta_{\mathrm{ACTIVE}}} \sup_{f \in \mathrm{BF(M)}} \mathbb{E}_{f, S_n}[\|\widehat{f}_n - f\|^2] \geq \mathrm{const}(M, \sigma^2) n^{-\frac{1}{d-1}} \ , \tag{3.8}$$

for n large enough, where $\mathrm{const}(M, \sigma^2) > 0$.

The above result is restricted to $d \geq 2$. It can be shown that the rate in the above bound is actually the optimal estimation rate. In contrast with Theorem 7

we observe that using active sampling has a potential performance gain over passive sampling, effectively equivalent to a dimensionality reduction: the exponent in (3.8) depends now on the dimension of the boundary set, $d-1$, instead of the dimension of the entire domain, d.

Noticing again that $\mathrm{BF}(M) \subseteq \mathrm{PC}(\beta, M)$ we have the following important result.

Proposition 3 (Active Learning Minimax Lower Bound - $\mathrm{PC}(\beta, M)$)**.** *Let $d \geq 2$. Under the requirements of the active sampling model we have*

$$\inf_{(\widehat{f}_n, S_n) \in \Theta_{ACTIVE}} \sup_{f \in \mathrm{PC}(\beta, M)} \mathbb{E}_{f, S_n}[\|\widehat{f}_n - f\|^2] \geq \mathrm{const}(\beta, M, \sigma^2) n^{-\frac{1}{d-1}} \ , \tag{3.9}$$

for n large enough, where $\mathrm{const}(\beta, M, \sigma^2) > 0$.

In the next section we verify that the bound in Proposition 3 is tight, and present a learning methodology whose performance is arbitrarily close to that bound (in terms of the polynomial rate).

When dealing with boundary fragments, relatively simple estimation algorithms can be constructed by taking advantage of the very special functional form of the boundary set. These algorithms begin by dividing the unit hypercube into "strips" and performing a one-dimensional change-point estimation in each of the strips. Under the passive sampling framework such change-point estimation has a performance limited by the parametric rate $1/n$, but, as seen above, when considering active sampling, the change-point detection can be done extremely accurately, using the Burnashev and Zigangirov method, by actively selecting the samples in each strip. Un-

fortunately, the boundary fragment class is very restrictive and impractical for most applications. Recall that boundary fragments consist of only two regions, separated by a boundary that is a function of the first $d-1$ coordinates. The class $\mathrm{PC}(\beta, M)$ is much larger and more general (*e.g.*, the function depicted in Figure 3.1(b)), so the algorithmic ideas that work for boundary fragments can no longer be used. In particular, the reduction of the problem to one-dimensional change-point detection problems on strips is no longer possible. A completely different approach is required, using radically different tools.

3.4 Estimation of Piecewise Constant Functions

In this section we present various estimation strategies, both for the passive and active learning settings. All the estimation strategies we present in this section hinge on tree structured partitions, that allow for the necessary degree of spatial adaptivity. The design and analysis of the proposed methods are intertwined, since we use various bounding techniques as guidelines in their construction. The following fundamental risk bound is a key tool.

Theorem 8 (Oracle Bound for Penalized Squared Error Estimation)**.** *Assume (B1) and (B3.1), and suppose* $|f(\boldsymbol{x})| \leq M$ *for all* $\boldsymbol{x} \in [0,1]^d$*. Furthermore let* $\{\boldsymbol{X}_i\}_{i=1}^n$ *be i.i.d., uniform over* $[0,1]^d$*. Suppose also that for all* $i \in \{1, \ldots, n\}$ *we have* $\mathbb{E}[W_i] = 0$*,* $\mathrm{Var}(W_i) \leq \sigma^2$*, and*

$$\mathbb{E}\left[|W_i|^k\right] \leq \mathrm{Var}(W_i)\frac{k!}{2}h^{k-2} \ , \tag{3.10}$$

for some $h > 0$ *and all* $k \geq 2$. *Equation* (3.10) *is known as the* Bernstein's moment condition.

Let Γ *be a countable class of functions mapping* $[0,1]^d$ *to the real line such that*

$$|g(x)| \leq M \quad \forall x \in [0,1]^d, \forall g \in \Gamma \ .$$

Let $\text{pen} : \Gamma \to [0, +\infty)$ *be a penalty function satisfying*

$$\sum_{g \in \Gamma} e^{-\text{pen}(g)} \leq 1 \ . \tag{3.11}$$

Finally define the estimator

$$\widehat{f_n} \triangleq \arg\min_{g \in \Gamma} \left\{ \frac{1}{n} \sum_{i=1}^{n} (Y_i - g(\boldsymbol{X}_i))^2 + \frac{\lambda}{n} \text{pen}(g) \right\} \ , \tag{3.12}$$

where $\lambda > 2(\sigma^2 + M^2) + 8(hM + M^2/3)$.

Then

$$\mathbb{E}\left[\|\widehat{f_n} - f\|^2 \right] \leq \min_{g \in \Gamma} \ \frac{1}{1-a} \left\{ (1+a)\|f - g\|^2 + \frac{\lambda}{n} \text{pen}(g) + \frac{4\lambda}{n} \right\} \ , \tag{3.13}$$

with $a = \frac{2(\sigma^2 + M^2)}{\lambda - 8(hM + M^2/3)}$.

This theorem is an oracle bound, that is, the expected error of the estimator is, up to a multiplicative constant, the best possible relative to the penalized criterion among all the models in Γ. This result follows very closely the approach in [44], with

some small modifications due to the noise model considered. For completeness we present the proof in Section 3.6.2. Note that there is some freedom when deciding how important the penalty is (parameter λ), and the condition on λ depends on M. This dependence is due to the bounding techniques used, and it is possible to derive similar bounds for general $\lambda > 0$. Other oracle bounds could be used instead (for example the bounds in [45]) yielding very similar results.

Remark: Equation (3.11) can be interpreted as a Kraft inequality [46]. This means that we can construct the penalty function $\text{pen}(\cdot)$ explicitly describing a prefix code for the elements of Γ (a prefix code is such that a codeword can be decoded as soon as it is entirely received). For each $f \in \Gamma$, $\text{pen}(f)$ is the length (in *nats* [46]) of the codeword associated with f. The coding argument is sometimes convenient when designing a penalty, since a prefix code automatically satisfies (3.11).

3.4.1 Passive Learning Algorithm for $\text{PC}(\beta, M)$

In this section we focus on the passive sampling model (B3.1) and the class of functions $\text{PC}(\beta, M)$. The ideas behind the particular estimation strategy presented here are the key for the construction of an active learning method able to nearly achieve the minimax lower bound of Proposition 3.

Since the location of the boundary is *a priori* unknown, it is natural to distribute the sample points uniformly over the unit hypercube. Various sampling schemes can be used to accomplish this, but we focus on a very simple randomized scheme. Let $\{\boldsymbol{X}_i\}_{i=1}^n$ be i.i.d. uniform over $[0,1]^d$. Under assumption (B2), W_i is Gaussian and so

we obtain the following corollary of Theorem 8.

Corollary 1. *Assume (B1), (B2) and (B3.1). Furthermore let* $\{\boldsymbol{X}_i\}_{i=1}^n$ *be i.i.d., uniform over* $[0,1]^d$*, and independent of* $\{Y_i\}_{i=1}^n$*. Consider a class of models* Γ *satisfying the conditions of Theorem 8 and define the estimator*

$$\widehat{f}_n(\boldsymbol{X},\boldsymbol{Y}) \triangleq \arg\min_{g\in\Gamma}\left\{\frac{1}{n}\sum_{i=1}^{n}(Y_i - g(\boldsymbol{X}_i))^2 + \frac{\lambda}{n}\mathrm{pen}(g)\right\}\ , \tag{3.14}$$

with $\lambda = 6(\sigma^2+M^2) + 8(\frac{2}{3}\sqrt{\frac{2}{\pi}}\sigma M + M^2/3)$*. Then*

$$\mathbb{E}\left[\|\widehat{f}_n - f\|^2\right] \leq \min_{g\in\Gamma}\left\{2\|f-g\|^2 + \frac{3}{2}\frac{\lambda}{n}\mathrm{pen}(g)\right\} + 6\frac{\lambda}{n}\ . \tag{3.15}$$

The proof is presented in Section 3.6.3. All the estimators we consider for the piecewise constant function class are based on *Recursive Dyadic Partitions* (RDPs). The elements of an RDP are quasi-disjoint subintervals of $[0,1]^d$, such that their union is the entire unit hypercube (two sets are quasi-disjoint if and only if their intersection has Lebesgue measure zero). We consider quasi-disjoint sets to avoid technicalities pertaining the boundaries of the partition sets, although it is possible construct proper partitions using this same reasoning. A RDP is any partition that can be constructed using only the following rules:

1. $\{[0,1]^d\}$ is a RDP;

2. Let $\pi = \{A_0, \ldots, A_{k-1}\}$ be a RDP, where $A_i = [a_{i1}, b_{i1}] \times \ldots \times [a_{id}, b_{id}]$. Then $\pi' = \{A_1, \ldots, A_{i-1}, A_i^{(0)}, \ldots, A_i^{(2^d-1)}, A_{i+1}, \ldots, A_k\}$ is a RDP, where $\{A_i^{(0)}, \ldots, A_i^{(2^d-1)}\}$ is obtained by dividing the hypercube A_i into 2^d quasi-disjoint hypercubes of equal size. Formally, let $q \in \{0, \ldots, 2^d - 1\}$ and $q = q_1 q_2 \ldots q_d$ be the binary representation of q. Then

$$A_i^{(q)} = \left[a_{i1} + \frac{b_{i1} - a_{i1}}{2} q_1, b_{i1} + \frac{a_{i1} - b_{i1}}{2}(1 - q_1)\right] \times \cdots \times \left[a_{id} + \frac{b_{id} - a_{id}}{2} q_d, b_{id} + \frac{a_{id} - b_{id}}{2}(1 - q_d)\right] .$$

Whenever a partition π' can be constructed by repeated application of rule (ii) to a partition π we say that the partitions are nested, and that $\pi' \preceq \pi$ (meaning that the partition π' is "finer" than partition π).

Other recursive partition strategies can also be considered, such as "free-split" procedures [47]. Some of these can also be analyzed under our framework, although extra difficulties arise.

It is clear that an RDP π can be described effectively by a rooted tree structure, where each leaf corresponds to a element of the partition, the root node corresponds to the set $[0, 1]^d$, and the internal nodes correspond to the aggregation of elements of π. This idea is illustrated in Figure 3.2 for the two-dimensional case. Denote the set of all RDPs by Π. We define the depth of a leaf in a RDP as the distance (number of edges) from the root to the leaf in the tree representation of the RDP. For example

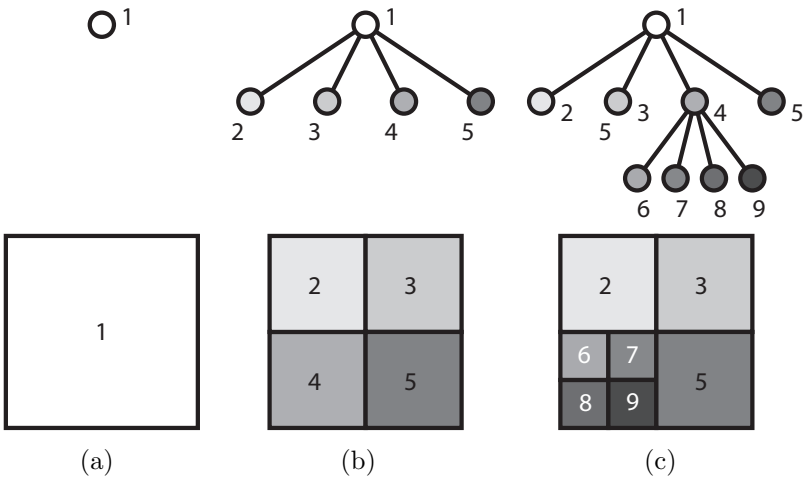


Figure 3.2: Example of Recursive Dyadic Partitions, and the corresponding tree representations.

in Figure 3.2(c) the RDP has four leafs at depth two and three leafs at depth one.

Our class of models is constructed assigning a constant value to each of the elements of a RDP, that is, our estimator is a stair function supported over a RDP. Formally, let π be a RDP and define

$$\Xi(\pi) = \left\{ g(x) : g(x) = \sum_{A \in \pi} c_A \mathbf{1}_A(\boldsymbol{x}),\ c_A \in \mathbb{R} \right\} . \tag{3.16}$$

The estimator $\widehat{f}_n$ we consider is best constructed in a two-stage way: define $\widehat{f}_n^{(\pi)} : [0,1]^d \to \mathbb{R}$ such that

$$\widehat{f}_n^{(\pi)} \ \triangleq \ \underset{g \in \Xi(\pi)}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} (Y_i - g(\boldsymbol{X}_i))^2 , \tag{3.17}$$

that is, for a fixed RDP π the function $\widehat{f}_n^{(\pi)}$ is the least squares fit of the data over

the class $\Xi(\pi)$. Now define

$$\widehat{\pi} \stackrel{\Delta}{=} \arg\min_{\pi \in \Pi} \left\{ \frac{1}{n} \sum_{i=1}^{n} (Y_i - \widehat{f}_n^{(\pi)}(\boldsymbol{X}_i))^2 + \frac{\lambda}{n} \text{pen}\left(\widehat{f}_n^{(\pi)}\right) \right\} , \tag{3.18}$$

where

$$\text{pen}\left(\widehat{f}_n^{(\pi)}\right) = \left(\frac{2^d \log 2}{2^d - 1} + \log(2n+1) \right) |\pi| , \tag{3.19}$$

and $|\pi|$ denotes the number of elements of partition π. Note that for $n \geq 2$ we have $\text{pen}\left(\widehat{f}_n^{(\pi)}\right) \leq c_{\text{pen}} |\pi| \log n$, where $c_{\text{pen}} = \frac{2^d}{2^d - 1} + \frac{5}{2 \log 2}$. This upper-bound can be used as the definition of the penalty, yielding essentially the same performance.

Finally, the estimator $\widehat{f}_n : [0,1]^d \to \mathbb{R}$ is defined as

$$\widehat{f}_n = \widehat{f}_n^{(\widehat{\pi})} . \tag{3.20}$$

The computation of $\widehat{f}_n$ can be done by efficiently using tree pruning algorithms, in the spirit of CART [47]. Although the estimator (3.20) is very appealing and practical, it is difficult to analyze under the scope of Theorem 8 and Corollary 1, since there is an uncountable number of possible models (because $\Xi(\pi)$ is uncountable). Instead we are going to analyze a related estimator, using only a finite subset of $\Xi(\pi)$, obtained by quantizing the constant values decorating each tree leaf. That modified estimator, presented in the Appendix 3.6.4, allows us to prove the main result of this section. It is important to note that the quantization modification of the estimator is important for the analysis of the algorithm, but the original algorithm, as presented in the

current section, still works extremely well in practice. The original algorithm can be studied using more sophisticated techniques, such as the ones described in [48].

Theorem 9 (Passive Learning Minimax Upper Bound - PC(β, M))**.** *Under (B1), (B2) and (B3.1) the algorithm described satisfies*

$$\sup_{f \in \mathrm{PC}(\beta,M)} \mathbb{E}[\|\widehat{f}_n - f\|^2] \;\leq\; \mathrm{const}(\beta, M, \sigma^2) \begin{cases} \frac{\log^2 n}{n} & , \text{ if } \; d = 1, \\ \left(\frac{\log n}{n}\right)^{\frac{1}{d}} & , \text{ if } \; d > 1, \end{cases},$$

where $\mathrm{const}(\beta, M, \sigma^2) > 0$.

The proof of the theorem is presented in Appendix 3.6.4 and consists of the analysis of the proposed estimator. It employs relatively standard techniques, common in the wavelet and regression trees literature. We observe that we get the same rate (up to a logarithmic factor) of Proposition 2, therefore this is the optimal polynomial rate of convergence for the passive sampling scenario.

Remark: *Although the estimator used in the proof of Theorem 9 involved a search of all possible RDPs, this is not at all required. We need only to consider RDPs up to a certain depth (more precisely up to depth* $J = \lceil \frac{1}{d} \log(n/\log(n)) \rceil$*). This fact is clear from the proof, since in the oracle bound we only need to consider such RDPs.*

3.4.2 Active Learning Algorithms for $\mathrm{PC}(\beta, M)$

In this section we present active learning algorithms (that is, estimation algorithms using active sampling) that improve upon the best passive learning performance rates under certain conditions. A very desirable feature of these methods is that they should be provably as good as the passive learning methods described in the previous chapter, but can have a greatly improved performance under many situations, in particular when working with the piecewise constant class.

The proposed scheme is based on a two-step approach. In the first step, called the *preview step*, an estimator of f is constructed using $n/2$ samples (assume without loss of generality that n is even), distributed uniformly over $[0,1]^d$. In the second step, called the *refinement step*, we select $n/2$ samples near the perceived locations of the boundaries (estimated in the preview step) separating constant regions. At the end of this process we will have half the samples concentrated in the vicinity of the perceived boundary set $B(f)$. Since accurately estimating f near the boundary set is key to obtaining faster rates, such a method has the potential to outperform the passive learning methodology described earlier. A graphical depiction of this approach is given in Figure 3.3. The two-steps of the proposed active learning method are described in more detail below. For simplicity assume throughout that n is even.

Preview: The goal of this step is two-fold: it provides an accurate estimate of f "away" from the boundary region, and very importantly provides a coarse estimate of the location of $B(f)$. This step is simply the application of the passive learn-

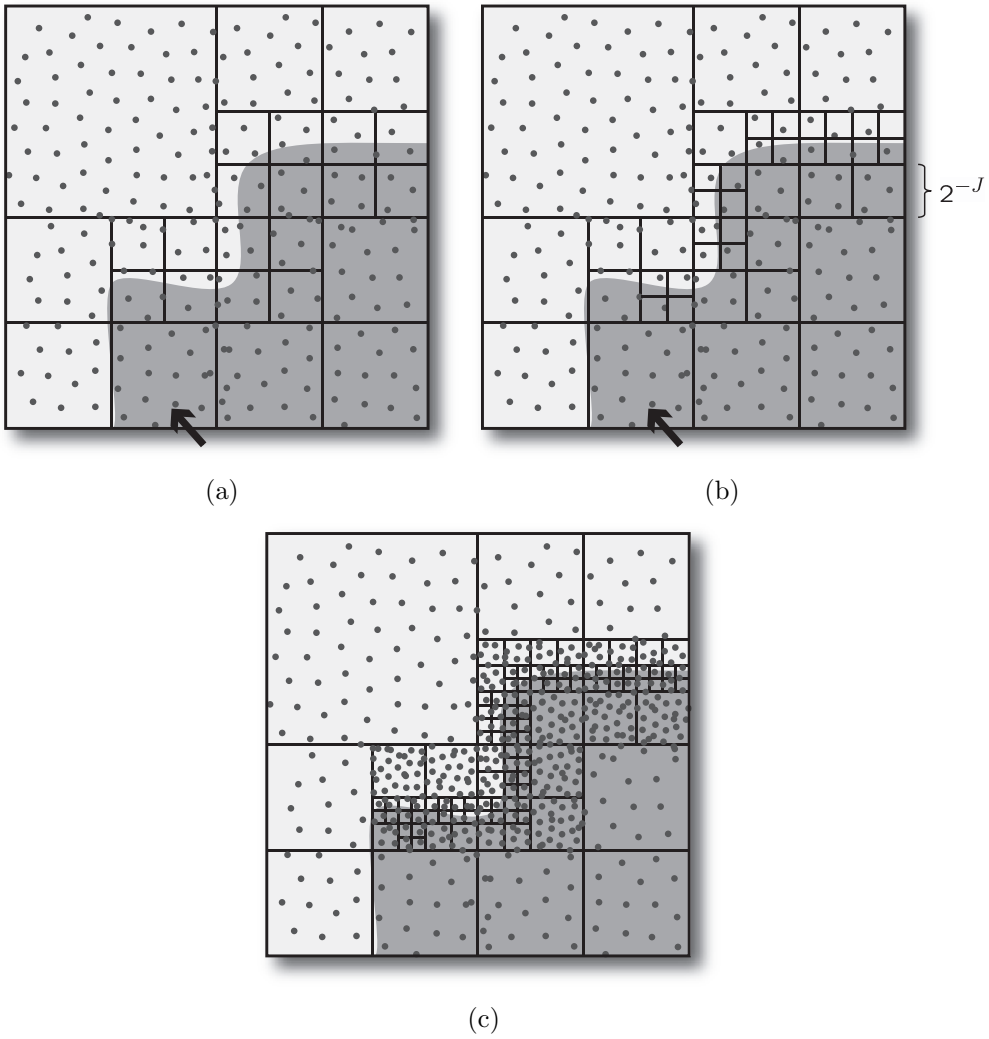


Figure 3.3: The two step procedure for $d = 2$ (no shifted partitions): (a) Preview step RDP. Note that the cell with the arrow was pruned shallower than depth J, but it contains a part of the boundary. (b) Additional sampling for the refinement step. (c) Refinement step.

ing method described in Section 3.4.1 using $n' \stackrel{\Delta}{=} n/2$ samples at points distributed uniformly over $[0,1]^d$. Denote this estimator by $\widehat{f}_0^p$. The RDP tree obtained with this methodology is going to be somewhat adapted to the boundary structure, with smaller (in size) leafs near the boundary, and larger leafs away from it. Therefore the passive learning algorithm provides us with a "boundary detector". Any region

spanned by a leaf that is at a considerable depth is a candidate for resampling and refinement. Specifically we refine any region spanned by leafs at depths greater or equal to

$$j \leq J = \left\lceil \frac{d-1}{(d-1)^2+d} \log(n'/\log(n') \right\rceil \ . \tag{3.21}$$

The reason for this choice of depth will be clear from the analysis of the algorithm, but for now notice that these depths are shallow enough so that a significant amount of data falls inside each cell. This is used to guarantee that we reliably detect the boundary if present. Formally,

$$\widehat{f}_0^p = \widehat{f}_{n'}^{(\widehat{\pi}_0^p)} \ ,$$

where $\widehat{f}_{n'}$ was defined in (3.18) and

$$\widehat{\pi}_0^p \triangleq \underset{\pi \in \Pi}{\arg\min} \left\{ \frac{1}{n'} \sum_{i=1}^{n'} (Y_i - \widehat{f}_{n'}^{(\pi)}(\boldsymbol{X}_i))^2 + \frac{\lambda}{n'} \text{pen}\left(\widehat{f}_{n'}^{(\pi)}\right) \right\} \ .$$

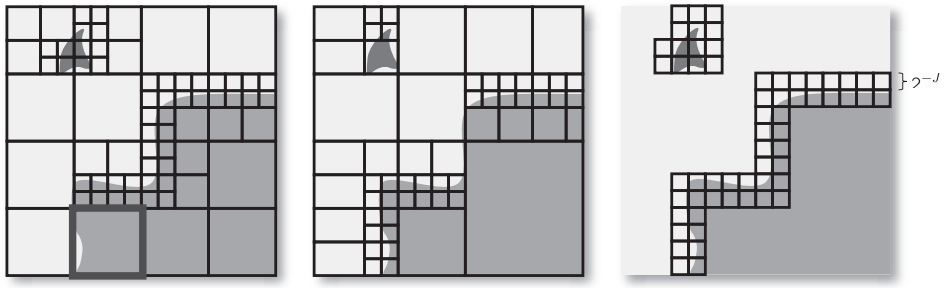
Unfortunately, if the set $B(f)$ is somewhat aligned with the dyadic splits of the RDP, leafs intersecting the boundary can be pruned without incurring a large error, and therefore we would not be able to properly detect the boundary in those situations. This is illustrated in Figure 3.4(a); the highlighted cell was pruned and contains a piece of the boundary. The error incurred by pruning should be small, since f is mostly a constant in that region. However,worst-case analysis reveals that the squared bias induced by these small volumes can add up, leading to relatively large total error.

This problem is related to the fact that our preview estimator is not translation invariant; that is, if instead of f we consider a slightly translated version of f, the RDP-based preview estimate obtained may change considerably. A way of mitigating this issue is to consider multiple RDP-based estimators, each one using a RDP appropriately shifted. We use $d+1$ estimators in the preview step: one on the initial uniform partition, and d over partitions whose dyadic splits have been translated by 2^{-J} in each one of the d coordinates. The main idea is illustrated in Figure 3.4 for the case of a horizontal shift: pruning the cells intersecting the highlighted boundary region would cause a large error, therefore making it easier to detect the boundary.

Formalizing the structure of the shifted partitions is a little cumbersome, but for the sake of completeness we include the rules to construct such partitions below. These are similar the rules presented before for the regular RDPs. A shifted RDP in the l^{th} coordinate satisfies the following.

1. $\{[0,1]^d\}$ is a RDP;

2. Let $\pi = \{A_1, \ldots, A_k\}$ be a RDP, where $A_i = [a_{i1}, b_{i1}] \times \ldots \times [a_{id}, b_{id}]$. Then $\pi' = \{A_1, \ldots, A_{i-1}, A_i^{(1)}, \ldots, A_i^{(2^d)}, A_{i+1}, \ldots, A_k\}$ is a RDP, where $\{A_i^{(1)}, \ldots, A_i^{(2^d)}\}$ is obtained by dividing the hyper-rectangle A_i into 2^d quasi-disjoint hyper-rectangles of equal size (except near the edge of the unit hypercube). Formally, let $q \in \{0, \ldots, 2^{d-1}\}$ and $q = q_1 q_2 \ldots q_d$ be the corresponding binary represen-



(a) (b) (c)

Figure 3.4: Illustration of the shifted RDP construction for $d = 2$: (a) RDP used in $\widehat{f}_0^p$. The highlighted cell intersects the boundary but it was pruned, since the pruning does not incur in severe error. (b) Shifted RDP, used in $\widehat{f}_1^p$. In this case the problem region is detected, since it would otherwise cause a large error. (c) These are the cells that are going to be refined in the refinement step.

tation. Then

$$
\begin{aligned}
A_i^{(q)} = & \\
& \left[a_{i1} + \frac{b_{i1} - a_{i1}}{2} q_1, b_{i1} + \frac{a_{i1} - b_{i1}}{2} (1 - q_1) \right] \times \cdots \\
& \times \left[a_{il} + \frac{b_{il} - a_{il}}{2} q_l + 2^{-J-1} \left(\mathbf{1}\{a_{il} = 0\} + \mathbf{1}\{b_{il} = 1\} \right) q_l, \right. \\
& \qquad \left. b_{il} + \frac{a_{il} - b_{il}}{2} (1 - q_l) + 2^{-J-1} \left(\mathbf{1}\{a_{il} = 0\} + \mathbf{1}\{b_{il} = 1\} \right) (1 - q_l) \right] \times \cdots \\
& \left[a_{id} + \frac{b_{id} - a_{id}}{2} q_d, b_{id} + \frac{a_{id} - b_{id}}{2} (1 - q_d) \right] .
\end{aligned}
$$

The preview estimators built on shifted partitions are defined as

$$
\widehat{f}_l^p \triangleq \widehat{f}_{n'}^{(\widehat{\pi}_l^p)} ,
$$

where

$$\widehat{\pi}_l^p \triangleq \underset{\pi \in \Pi^l_{\text{shift}}}{\arg\min} \left\{ \frac{1}{2\sigma^2} \sum_{i=1}^{n'} (Y_i - \widehat{f}_{n'}^{(\pi)}(\boldsymbol{X}_i))^2 + \lambda \cdot \text{pen}\left(\widehat{f}_{n'}^{(\pi)}\right) \right\} .$$

The analysis of an estimator built on top of this shifted partitions is similar to the one for regular partitions. Therefore the proof of Theorem 9 applies also to this estimator. The only difference is that now the volume of cells in the partition at depth j might be larger than 2^{-j}, although it is at most 2×2^{-j} (therefore the right-hand-side of (3.36) is multiplied by 2). This only affects the constant $\text{const}(\beta, M, \sigma^2)$ in Theorem 9.

Any leaf that is at depth greater or equal to J in any of the $d+1$ RDPs obtained in the preview step indicates the highly probable presence of a boundary, and will be estimated after resampling in the refinement step, described below.

Refinement: With high probability, the boundary is contained in deeper leafs, that is, leafs that are at depths greater or equal to J. In the refinement step we collect additional n' samples in the corresponding partition sets, and obtain a refined estimate of the function f. To facilitate the formal description it is useful to introduce a pruned version of the RDPs obtained in the preview step. Let $\pi \in \Pi$ be a RDP and define $\text{PRUNED}_J(\pi)$ to be the RDP such that the corresponding tree has the same structure of the tree describing π for all the leafs at depths shallower or equal to J, and no leafs at depths greater than J. In other words, if π has a partition cell whose corresponding depth is greater than J then in $\text{PRUNED}_J(\pi)$ this cell is replaced by the

parent cell at depth J were the smaller cell is contained. Let

$$\widehat{\mathcal{R}} = \bigcup_{l=0}^{d} \{A \in \text{PRUNED}_J(\widehat{\pi}_l^p) : \ A \text{ corresponds to a leaf at depth } J\} \ . \tag{3.22}$$

Note that $\widehat{\mathcal{R}}$ is a collection of sets (a set of sets), and that according to Definition (3.22) there might be repetitions in the elements of $\widehat{\mathcal{R}}$. In the following assume that those repetitions are removed, that is $\widehat{\mathcal{R}}$ is a collection of disjoint hypercubes of sidelength 2^{-J}. Assume that $\widehat{\mathcal{R}}$ is not empty (a comment regarding this is issued below). For each set $A \in \widehat{\mathcal{R}}$ we collect $n'/|\widehat{\mathcal{R}}|$ samples, distributed uniformly over each A (recall that $|\widehat{\mathcal{R}}|$ denotes the number of elements of $\widehat{\mathcal{R}}$). Therefore we collect a total of n' samples in this step. For each set $A \in \widehat{\mathcal{R}}$ we repeat the tree pruning process described in Section 3.4.1 (but now instead of defining the possible RDPs over the unit hypercube, we define them over A, see Figure 3.3(c)). This produces a higher resolution estimate in the vicinity of the perceived boundary set $B(f)$, yielding a better performance than the passive learning technique. Denote the estimator obtained for the set $A \in \widehat{\mathcal{R}}$ by $\widehat{f}_A^r$. The overall estimator of f, after the preview and refinement steps is denoted by $\widehat{f}_{\text{ACTIVE}}$ and it is defined as

$$\widehat{f}_{\text{ACTIVE}}(\boldsymbol{x}) = \begin{cases} \widehat{f}_A^r(\boldsymbol{x}) & \text{, if } \boldsymbol{x} \in A : A \in \widehat{\mathcal{R}}, \\ \widehat{f}_0^p(\boldsymbol{x}) & \text{, otherwise} \end{cases} \ . \tag{3.23}$$

Some simple modifications can be done to this estimator to improve the performance in practice. For example instead of using solely $\widehat{f}_0^p$ in the regions that are not

refined one can consider the arithmetic average of all the shifted preview estimators. It is easy to show that the expected performance of this average estimator is as good as the performance of any of the preview estimators. In practice though these averaged estimates are a little more desirable since some of the artifacts created by the dyadic structure of the RDPs are mitigated.

It is important at this point to mention that the performance of this algorithm is as good as the performance of the passive algorithm (up to a constant multiplicative factor). This is extremely desirable since it guarantees that we are always going to perform as well as the best passive learning method, but in many cases we can attain a much better performance by taking advantage of active sampling.

Theorem 10. *Let $d \geq 2$. Under (B1), (B2) and (B3.1) we have*

$$\begin{aligned} \sup_{f \in \mathrm{PC}(\beta, M)} \mathbb{E}[\|\widehat{f}_{ACTIVE} - f\|^2] &\leq \sup_{f \in \mathrm{PC}(\beta, M)} \mathbb{E}[\|\widehat{f}_0^p - f\|^2] \\ &\leq \mathrm{const}(\beta, M, \sigma^2) \left(\frac{\log n}{n}\right)^{\frac{1}{d}} , \end{aligned}$$

where $\mathrm{const}(\beta, M, \sigma^2) > 0$.

The proof of the theorem is presented in Section 3.6.5 and proceeds by noticing that the extra sampling in the refinement step will only improve the expected performance of $\widehat{f}_0^p$ and never degrade it.

To guarantee that the proposed algorithm has improved performance over passive methods we need to restrict the piecewise constant class a little further. In particular

we consider the following definition.

Definition 5 (Boundary Detectability)**.** *Let $f \in PC(\beta, M)$ and consider the partition of $[0,1]^d$ into 2^{dJ} identical hypercubes (cells), with J as defined above in* (3.21)*. Denote this partition by $\mathcal{C}_J$. This partition corresponds to an RDP with all the leafs at depth J. Let $f_J : [0,1]^d \to \mathbb{R}$ be a coarse approximation of f. Formally, for all partition cells $A \in \mathcal{C}_J$ we have*

$$f_J(\boldsymbol{x}) = \sum_{A \in \mathcal{C}_J} \frac{1}{\mathrm{Vol}(A)} \left(\int_A f(\boldsymbol{t}) \mathrm{d}\boldsymbol{t} \right) \mathbf{1}_A(\boldsymbol{x}) \ ,$$

where $\mathrm{Vol}(\cdot)$ *denotes the volume of a set (in the above expression* $\mathrm{Vol}(A) = 2^{-dJ}$*).*

Let $\mathcal{C}_J^B = \{A \in \mathcal{C}_J : \ A \cap B(f) \neq \emptyset\}$ be the set of cells in $\mathcal{C}_J$ that intersects the boundary. Let $A \in \mathcal{C}_J^B$ and consider the l-coordinate shifted RDPs. Let $\mathcal{A}(A, l)$ denote the parent cell of A at depth $J-1$, that is, the cell corresponding to the parent node of A in the shifted RDP (recall that $l = 0$ corresponds to the usual non-shifted RDPs). Function f has a detectable boundary *if, for all $A \in \mathcal{C}_J^B$ we have, for at least one $l \in \{0, \ldots, d\}$,*

$$\int_{\mathcal{A}(A,l)} \left(f_J(\boldsymbol{x}) - \frac{1}{\mathrm{Vol}\left(\mathcal{A}(A,l)\right)} \int_{\mathcal{A}(A,l)} f(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} \right)^2 \mathrm{d}\boldsymbol{x} \geq C^* 2^{-dJ} \ , \tag{3.24}$$

where $C^* > 0$, and $J \geq J^*$ (in other words we are considering a high enough resolution).

We denote by $\mathrm{PC}^*(\beta, M, C^*, J^*)$ *the class of piecewise constant functions with*

detectable boundaries.

The "detectability" condition above is required to ensure that boundaries can be reliably detected in the preview stage. This is influenced by the magnitude of the transition between constant regions of f, larger "jumps" yield larger C^* values. But this condition restricts further the shape of the boundary sets, although it is still quite general encompassing many interesting cases. The condition restricts the existence of arbitrarily small boundary features (for example, very small constant regions, or ribbon like features). It further enforces the boundary set $B(f)$ to be "cusp-free". A cusp-free boundary cannot have the behavior you observe in the graph of $|x|^{1/2}$ at the origin. Less "aggressive" kinks are allowed, such as in the graph of $|x|$. A cusp-like structure is difficult to detect with the preview step, since it is very "thin", but might still have enough volume to prevent the algorithm from achieving the correct rate.

It might be possible for the proposed algorithm to display provable gains (with respect to passive learning) requiring only a weaker condition. Nevertheless a condition restricting the size of small constant regions is required for uniform bounds. Figure 3.5 illustrates the detectability condition: the piece of the boundary in the central cell at depth J is "sensed" by the three different RDP elements at depths $J-1$ (one for each possible partition shift). Clearly the partition in Figure 3.5(d) is able to "feel" the small boundary piece.

The detectability condition holds in particular for Lipschitz boundary fragments. Figure 3.5 also illustrates this. For boundary fragments two partitions suffice: the

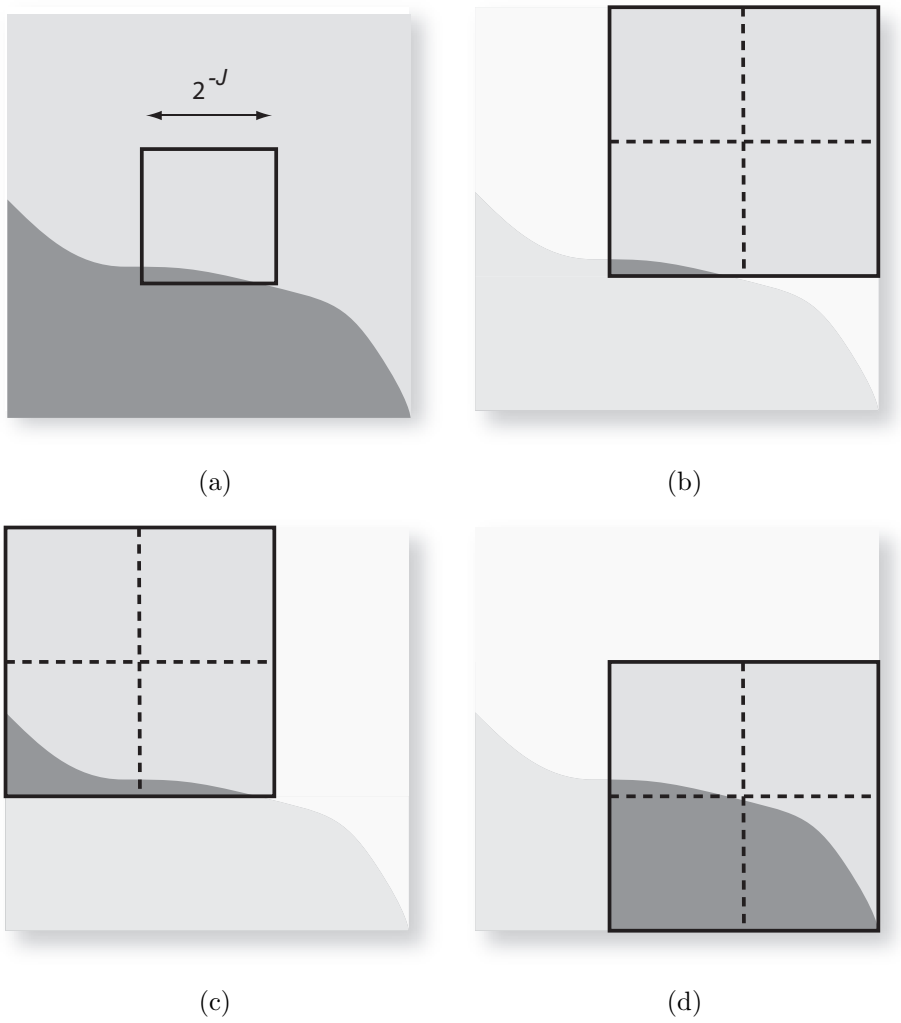


Figure 3.5: Illustration of the detectability condition. Figure (a) depicts a small region of the boundary, and we are in particular interested on the small region A marked with a square. Figures (b), (c) and (d) depict the cells at depth $J-1$ in containing A in the various shifted partitions. The cell in (d) is able to "feel" region A, since there is a significant volume of both constant levels in that cell.

original partition and a partition shifting the d^{th} coordinate. Finally, a loose interpretation of the detectability condition is that most of the boundary set must locally be similar to a boundary fragment with respect to some orientation of the coordinate axis.

Theorem 11 (Active Learning Upper Bound - $\mathrm{PC}^*(\beta, M)$)**.** *Let $d \geq 2$. Under assumptions (B1), (B2), and (B3.2), we have for the algorithm presented above*

$$\sup_{f \in \mathrm{PC}^*(\beta,M,C^*)} \mathbb{E}\left[\|\widehat{f}_{ACTIVE} - f\|^2\right] \leq \mathrm{const}(\beta, M, \sigma^2, C^*) \left(\frac{n}{\log n}\right)^{-\frac{1}{d-1+1/d}}, \quad (3.25)$$

where $\mathrm{const}(\beta, M, \sigma^2, C^*) > 0$ *and n is large enough.*

The proof of Theorem 11 is presented in Section 3.6.6. Note that we improve on the passive learning rates using this learning method, but do not achieve the lower bound of Proposition 3. By iterating this methodology (generalizing to a multi-step approach) it is possible get arbitrarily close to the rate of Proposition 3, as we see below. Note that the bound only holds for a number of samples n large enough. Essentially we need to guarantee that we are performing the boundary detection in the preview step at a fine enough resolution, so that the detectability condition holds. Since that resolution grows with n it follows that we just need the number of samples n be such that $J \equiv J(n) \geq J^*$.

The main idea behind the proof of Theorem 11 is to decompose the error of the estimator for three different cases: (i) the error incurred during the preview step in regions "away" from the boundary; (ii) the error incurred by not detecting a piece of the boundary (and therefore not performing the refinement step in that area); (iii) the error remaining in the refinement region after the refinement step. The type-(i) error is controlled by considering the refinement of any region that was assigned a small partition set (corresponding to a leaf deeper than J) in the preview step. This ensures

that in regions that are not refined the quality of the preview estimate is very good, not exceeding the error rate in (3.25). Type-(ii) error corresponds to the situations when a part of the boundary was not detected in the preview step. This can happen because of the inherent randomness of the noise and sampling distribution, or because the boundary is somewhat aligned with the dyadic splits, like in Figure 3.3(a). The latter can be a problem and this is why one needs to perform $d+1$ preview estimates over shifted partitions. If the boundary satisfies the detectability condition then it is guaranteed that one of those preview estimators is going to "feel" the boundary since it is not aligned with the corresponding partition. A piece of the boundary region is not refined if it is not detected in *all* the shifted partition estimators. The worst-case error can be shown not to exceed the value in (3.25), therefore failure to detect the boundary has the same contribution for the total error as the type-(i) error. Finally, analysis of type-(iii) error is relatively easy. Nonetheless one needs to make sure that the size of the region that needs to be refined is not too large, since in that case the density of samples in the refinement step might be not be sufficient to improve on passive methods. In other words, one needs to make sure that in the preview step not many regions are wrongly characterized as boundary.

As said before, one can reiterate the two-step procedure: For example, to obtain a three step procedure we can start with a similar preview step, using $n'' \stackrel{\Delta}{=} n/3$ samples, and a different value of J, adjusted accordingly. In the refinement step apply the two-step procedure as described above, instead of the passive strategy. With this three

step approach we attain the error decay rate $\sim n^{\frac{1}{d-1+\epsilon_2}}$, (ignoring logarithmic factors) where $\epsilon_2 = 1/((d-1)^2+d)$. To obtain this result we use in the first step

$$J = \left\lceil \frac{(d-1)^2}{d(d^2-2d+2)} \log(n''/\log(n'')) \right\rceil \ .$$

Notice that the error decay rate improved with respect to Theorem 11. This procedure can be repeated, and we get the following result

Theorem 12 (Upper Bound for K-step Method on $\mathrm{PC}^*(\beta, M)$)**.** *Let $\widehat{f}_n^{(K)}$ be the estimator obtained using a K-step approach as described above (K is arbitrary, but fixed with respect to n). Let $d \geq 2$ and assume (B1),(B2), and (B3.2). Then*

$$\mathbb{E}\left[\|\widehat{f}_n^{(K)} - f\|^2\right] \leq \mathrm{const}(\beta, M, \sigma^2, C^*, K) \left(\frac{n}{\log n}\right)^{-\frac{1}{d-1+\epsilon_K}} \ ,$$

where $\mathrm{const}(\beta, M, \sigma^2, C^*, K) > 0$ *and* $\epsilon_K > 0$ *is given by*

$$\epsilon_K = \begin{cases} 1/K & , \text{ if } d = 2 \\ \frac{d-2}{(d-1)^K - 1} & , \text{ if } d > 2 \end{cases} \ .$$

Therefore ϵ_K can be made arbitrarily small as the number of steps increases.

The proof of Theorem 12 is presented in Section 3.6.7, and proceeds by carefully choosing the maximum resolution J at each step, balancing the error between each subsequent steps (as in Theorem 11). Notice that the rate in this theorem can be made arbitrarily close to the lower bound rate of Proposition 3, meaning that this

is the optimal polynomial error rate of active learning for the piecewise constant function class.

3.5 Final Remarks and Open Questions

The results presented in this chapter show that in certain scenarios active learning attains provable gains over the classical passive approaches. Methodologies based on active sampling are intuitively appealing, and can be applied in many practical problems. Despite these draws, the analysis of such active methods is quite challenging due to the loss of statistical independence in the observations. The function classes presented here are non-trivial canonical examples illustrating under what conditions one might expect active learning to improve rates of convergence. The algorithms presented for actively learning members of the piecewise constant class demonstrates the possibilities of these ideas, and in fact, this algorithm has already been applied in the context of field estimation using wireless sensor networks [6] and ballistic laser imaging [1]. As pointed out earlier, the overall methodology is spatially-adaptive in two ways: in the estimation/model-selection procedure *and* in the data collection process. In fact, the active sampling procedure leverages the spatial-adaptivity of multiscale estimators in order to guide the data collection process.

The algorithmic ideas presented in this chapter are relatively simple and intuitive, but the formal analysis demonstrates the difficulties and challenges inherent in the study of procedures based on adaptive sampling. For example, the algorithm

developed is rather "aggressive" or greedy: In the preview step we cannot miss any part of the boundary set, since we have no further chance of detecting it (in the refinement step). This is the reason we need the boundary detectability condition (3.24). Although less aggressive algorithmic techniques can be devised, their analysis becomes extremely difficult, particularly when dealing with non-parametric settings. In principle, the proposed method can be extended to the piecewise smooth class of functions, as long as condition (3.24) is satisfied. This enforces that such a function f is discontinuous in the boundary set. Although intuitively this algorithmic extension should work, we do not have yet a formal analysis. The main difficulty is proving a corresponding version of Lemma 6. Another possibility for generalization is consideration of different boundary models: (i) smoother boundaries, for example locally smooth boundaries, and (ii) consider discontinuities in a derivative of the regression function across a $d-1$-dimensional boundary (instead of a discontinuity of f itself). In any of these cases an active sampling procedure may yield performance gains over passive methodologies. We are currently exploring these possibilities.

3.6 Proofs

3.6.1 Proof of Theorem 7

The proof of Theorem 7 follows closely the construction used to prove Theorem 3. The key idea of the proof is to reduce the problem of estimating a function in $\Sigma_d(L, \alpha)$ to the problem of deciding among a finite number of such functions. In other words,

instead of an estimation problem we consider an hypotheses testing problem. The proof methodology for the passive setting is adequate for the active scenario because we can choose an adequate set of hypothesis without knowledge of the sampling strategy. There are also other modifications needed, due to the extra flexibility of the sampling strategy.

The proof is essentially an application of Theorem 6 on page 58. When applying the theorem the probability measures we consider are $P_{\boldsymbol{X}_1,\ldots,\boldsymbol{X}_n,Y_1,\ldots,Y_n}$, and the distance metric used is simply the L_2 norm $\|\cdot\|_2$.

Consider a fixed sample size n. The first step is the construction of a suitable collection of hypothesis $f_j(\cdot) \in \Sigma_d(L,\alpha)$, $j = 0,\ldots,M$. Recall that $\lceil x \rceil$ denotes the minimal integer such that $x > \lceil x \rceil$. Let $c_0 > 0$ and define

$$m = \left\lceil c_o n^{\frac{1}{2\alpha+d}} \right\rceil, \quad h = \frac{1}{m}, \quad \boldsymbol{x}_k = \frac{\boldsymbol{k} - 1/2}{m} ,$$

and

$$\varphi_{\boldsymbol{k}}(\boldsymbol{x}) = Lh^{\alpha} K\left(\frac{\boldsymbol{x} - \boldsymbol{x}_k}{h}\right) ,$$

where $\boldsymbol{k} \in \{1,\ldots,m\}^d$, $\boldsymbol{x} \in [0,1]^d$ and $K : \mathbb{R}^d \to [0,+\infty)$ satisfies $K \in \Sigma_d(1,\alpha)$ and supp $K = (-1/2,1/2)^d$. It is easily shown that such a function K exists, for example

$$K(\boldsymbol{x}) = a\tilde{K}(2\boldsymbol{x}), \text{ with } \tilde{K}(\boldsymbol{x}) = \prod_{i=1}^{d} \exp\left(-\frac{1}{1-x_i^2}\right) \mathbf{1}\{|x_i| < 1\} ,$$

where $\boldsymbol{x} = (x_1,\ldots,x_i)$ and $a > 0$ is sufficiently small.

Let $\Omega = \{\boldsymbol{\omega} = (\omega_1, \ldots, \omega_{m^d}),\ \omega_i \in \{0,1\}\} = \{0,1\}^{m^d}$, and define

$$\xi = \left\{ f_{\boldsymbol{\omega}}(\cdot) : f_{\boldsymbol{\omega}}(\cdot) = \sum_{\boldsymbol{k} \in \{1,\ldots,m\}^d} \omega_{\boldsymbol{k}} \varphi_{\boldsymbol{k}}(\cdot),\ \boldsymbol{\omega} \in \Omega \right\} .$$

Note that $\varphi_{\boldsymbol{k}} \in \Sigma_d(L,\alpha)$ and these functions have disjoint support, therefore $\xi \subseteq \Sigma_d(L,\alpha)$. For $\boldsymbol{\omega}, \boldsymbol{\omega'} \in \Omega$

$$\begin{aligned}
\|f_{\boldsymbol{\omega}} - f_{\boldsymbol{\omega'}}\| &= \left[\int_{[0,1]^d} (f_{\boldsymbol{\omega}}(\boldsymbol{x}) - f_{\boldsymbol{\omega'}}(\boldsymbol{x}))^2 \mathrm{d}\boldsymbol{x} \right]^{1/2} \\
&= \left[\sum_{\boldsymbol{k} \in \{1,\ldots,m\}^d} (\omega_{\boldsymbol{k}} - \omega'_{\boldsymbol{k}})^2 \int_{[0,1]^d} \varphi_{\boldsymbol{k}}^2(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right]^{1/2} \\
&= Lh^{\alpha+d/2} \|K\| \left[\sum_{\boldsymbol{k} \in \{1,\ldots,m\}^d} |\omega_{\boldsymbol{k}} - \omega'_{\boldsymbol{k}}| \right]^{1/2} \\
&= Lh^{\alpha+d/2} \|K\| \sqrt{\rho(\boldsymbol{\omega}, \boldsymbol{\omega'})} ,
\end{aligned}$$

where ρ is the Hamming distance between $\boldsymbol{\omega}$ and $\boldsymbol{\omega'}$, and $\|K\| = \sqrt{\int_{[0,1]^d} K^2(\boldsymbol{x}) \mathrm{d}x}$. We will choose our hypotheses set from ξ, but we do not need the entire set. We will use Lemma 2 on page 60 to yield a suitable subset of ξ. Since the lemma was previously stated in a different way we restate the result in a convenient form: Let $m^d \geq 8$. There exists a subset $\{\boldsymbol{\omega}^{(0)}, \boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(M)}\}$ of Ω such that $\boldsymbol{\omega}^{(0)} = (0, \ldots, 0)$ and

$$\rho(\boldsymbol{\omega}^{(j)}, \boldsymbol{\omega}^{(k)}) \geq m^d/8, \quad \forall\ 0 \leq j < k \leq M$$

and $M \geq 2^{m^d/8}$.

Define $f_j(\cdot) \stackrel{\Delta}{=} f_{\boldsymbol{\omega}^{(j)}}(\cdot)$, with $j = 0, \ldots, M$. This is the collection of hypotheses we use with Theorem 6. We need to verify the three conditions in the theorem. As already pointed out, notice that $f_j \in \Sigma_d(L, \alpha)$.

i)

$$\begin{aligned} \|f_j - f_k\| = Lh^{\alpha+d/2}\|K\|\sqrt{\rho(\boldsymbol{\omega}^{(j)}, \boldsymbol{\omega}^{(k)})} &\geq Lh^{\alpha+d/2}\|K\|\sqrt{m^d/8} \\ &= Lm^{-\alpha}\|K\|/\sqrt{8}\ , \end{aligned}$$

as long as $m^d > 8$. This is the case if $n \geq n^*$ with $n^* = (8^{1/d}/c_0)^{2\alpha+d}$ (since then $c_0(n^*)^{1/(2\alpha+d)} \geq 8^{1/d}$). Taking this into account let $n \geq n^*$. Then $m \leq c_0 n^{\frac{1}{2\alpha+d}} + 1 \leq c_0 n^{\frac{1}{2\alpha+d}}(1 + 8^{-1/d})$, and therefore

$$\begin{aligned} \|f_j - f_k\| &\geq L\frac{1}{\sqrt{8}}m^{-\alpha}\|K\| \\ &\geq L\frac{1}{\sqrt{8}}(1 + 8^{-1/d})^{-\alpha}c_0^{-\alpha}\|K\|n^{-\frac{\alpha}{2\alpha+d}} \\ &\geq L/3\ c_0^{-\alpha}\|K\|n^{-\frac{\alpha}{2\alpha+d}} \\ &= A\psi_n\ , \end{aligned}$$

where $\psi_n = n^{-\frac{\alpha}{2\alpha+d}}$ and $A = Lc_0^{-\alpha}\|K\|/3$.

ii) Under the modeling assumptions we the probability measure of $\{\boldsymbol{X}_j, Y_j\}_{j=1}^n$ is completely defined by the noise model and the sampling strategy S_n. Therefore it is clear that, without loss of generality, the conditional random variable

$\boldsymbol{X}_i|\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1}$ has a density $p_{\boldsymbol{X}_i|\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1}}$ with respect to a suitable dominating measure, therefore the condition $P_{f_j} \ll P_{f_0}$ holds trivially.

iii) From our active sampling modeling assumptions we see that the probability measure of $(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n, Y_1, \ldots, Y_n)$ has a nice factorization. As mentioned in (ii) the conditional random variable $\boldsymbol{X}_i|\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1}$ has a density $p_{\boldsymbol{X}_i|\boldsymbol{X}_1 \ldots \boldsymbol{X}_{i-1}, Y_1 \ldots Y_{i-1}}$ with respect to a suitable dominating measure. For a function $f \in \Sigma_d(L, \alpha)$ the joint probability measure of the sample points and observations has a density (with respect to a suitable dominating measure) of the form

$$\begin{aligned} & p_{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n, Y_1, \ldots, Y_n}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, y_1, \ldots, y_n) \\ &= p_{\boldsymbol{Z}_n^X, \boldsymbol{Z}_n^Y}(\boldsymbol{z}_n^X, \boldsymbol{z}_n^Y) \\ &= \prod_{i=1}^{n} p_{Y_i|\boldsymbol{X}_i;f}(y_i|\boldsymbol{x}_i)\ p_{\boldsymbol{X}_i|\boldsymbol{Z}_{i-1}^X, \boldsymbol{Z}_{i-1}^Y}(\boldsymbol{x}_i|\boldsymbol{z}_{i-1}^X, \boldsymbol{z}_{i-1}^Y)\ , \end{aligned} \tag{3.26}$$

where $\boldsymbol{Z}_i^X \triangleq (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_i)$, $\boldsymbol{z}_i^X \triangleq (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i)$, $\boldsymbol{Z}_i^Y \triangleq (Y_1, \ldots, Y_i)$ and $\boldsymbol{z}_i^Y \triangleq (y_1, \ldots, y_i)$. Notice that (B1) yields $p_{Y_i|\boldsymbol{X}_i;f}(y_i|\boldsymbol{x}_i) = p_W(y_i - f(\boldsymbol{x}_i))$, where $p_W(x) \triangleq 1/(2\pi\sigma^2) \exp(-x^2/(2\sigma^2))$ is the density of Gaussian random variable with zero mean and variance σ^2. Taking into account the factorization in (3.26)

and recalling that $f_0(\cdot) = 0$ we have

$$\begin{aligned}
\mathrm{KL}(P_j \| P_0) &= \mathbb{E}_{f_j}\left[\log \frac{\prod_{i=1}^n p_{Y_i|\boldsymbol{X}_i;f_j}(Y_i|\boldsymbol{X}_i)\; p_{\boldsymbol{X}_i|\boldsymbol{Z}_{i-1}^X,\boldsymbol{Z}_{i-1}^Y}(\boldsymbol{X}_i|\boldsymbol{Z}_{i-1}^X,\boldsymbol{Z}_{i-1}^Y)}{\prod_{i=1}^n p_{Y_i|\boldsymbol{X}_i;f_0}(Y_i|\boldsymbol{X}_i)\; p_{\boldsymbol{X}_i|\boldsymbol{Z}_{i-1}^X,\boldsymbol{Z}_{i-1}^Y}(\boldsymbol{X}_i|\boldsymbol{Z}_{i-1}^X,\boldsymbol{Z}_{i-1}^Y)}\right] \\
&= \mathbb{E}_{f_j}\left[\log \frac{\prod_{i=1}^n p_{Y_i|\boldsymbol{X}_i;f_j}(Y_i|\boldsymbol{X}_i)}{\prod_{i=1}^n p_{Y_i|\boldsymbol{X}_i;f_0}(Y_i|\boldsymbol{X}_i)}\right] \\
&= \sum_{i=1}^n \mathbb{E}_{f_j}\left[\log \frac{p_W(Y_i - f_j(\boldsymbol{X}_i))}{p_W(Y_i - f_0(\boldsymbol{X}_i))}\right] \\
&\leq n \max_{\boldsymbol{x}\in[0,1]^d} \mathbb{E}_{f_j}\left[\log \frac{p_W(Y_1 - f_j(\boldsymbol{X}_1))}{p_W(Y_1 - f_0(\boldsymbol{X}_1))}\middle| \boldsymbol{X}_1 = \boldsymbol{x}\right] \\
&\leq n \max_{\boldsymbol{x}\in[0,1]^d} \frac{1}{2\sigma^2}(f_j(\boldsymbol{x}) - f_0(\boldsymbol{x}))^2 \\
&\leq n \max_{\boldsymbol{x}\in[0,1]^d} \frac{1}{2\sigma^2}(f_j(\boldsymbol{x}))^2 \\
&\leq \frac{1}{2\sigma^2} L^2 K_{\max}^2 n c_0^{-2\alpha} n^{-\frac{2\alpha}{2\alpha+d}} \\
&= \frac{1}{2\sigma^2} L^2 K_{\max}^2 c_0^{-(2\alpha+d)} c_0^d n^{\frac{d}{2\alpha+d}} \leq \frac{1}{2\sigma^2} L^2 K_{\max}^2 c_0^{-(2\alpha+d)} m^d \ .
\end{aligned}$$

From Lemma 2 we have $m^d \leq 8 \log M / \log 2$ therefore choosing

$$c_0 = \left(\frac{4L^2 K_{\max}^2}{\sigma^2 \gamma \log 2}\right)^{\frac{1}{2\alpha+d}} \ ,$$

with $0 < \gamma < 1/8$ fulfills all the conditions of Theorem 6 yielding

$$\inf_{(\widehat{f}_n, S_n)\in\Theta_{\mathrm{ACTIVE}}} \sup_{f\in\Sigma_d(L,\alpha)} \mathcal{P}_{f,S_n}\left(\|\widehat{f}_n - f\| \geq A n^{-\frac{\alpha}{2\alpha+d}}\right) \geq c_1 > 0 \ ,$$

where $c_1 = \frac{\sqrt{M}}{1+\sqrt{M}}\left(1 - 2\gamma - 2\sqrt{\frac{\gamma}{\log M}}\right)$.

The end result follows now from a straightforward application of Markov's in-

equality and so, for $n > n^*$,

$$\inf_{(\widehat{f}_n, S_n) \in \Theta_{\text{ACTIVE}}} \sup_{f \in \Sigma_d(L,\alpha)} \mathbb{E}_{f,S_n}[\|\widehat{f}_n - f\|^2] \geq c_1 A^2 n^{-\frac{2\alpha}{2\alpha+d}} \ . \tag{3.27}$$

□

3.6.2 Proof of Theorem 8

The proof follows closely the strategy in [44], with changes pertaining the different noise model considered. For the sake of completeness we include the full derivation here. The proof hinges on a concentration inequality due to Craig [49].

Theorem 13 (Craig, 1933)**.** *Let $\{U_i\}_{i=1}^n$ be independent random variables, satisfying the Bernstein moment condition*

$$\mathbb{E}\left[|U_i - E[U_i]|^k\right] = \text{Var}(U_i)\frac{k!}{2}h^{k-2} \ ,$$

for some $h > 0$ and all $k \geq 2$. Let $\overline{U} = (1/n)\sum_{i=1}^n U_i$. Then

$$\Pr\left(\overline{U} - \mathbb{E}[\overline{U}] \geq \frac{\tau}{n\epsilon} + \frac{n\epsilon\text{Var}(\overline{U})}{2(1-c)}\right) \leq \exp(-\tau) \ ,$$

for $0 < \epsilon h \leq c < 1$ and $\tau > 0$.

Start by defining

$$r(g, f) = \mathbb{E}\left[(Y - g(\boldsymbol{X}))^2\right] - \mathbb{E}\left[(Y - f(\boldsymbol{X}))^2\right] \ .$$

Note that

$$r(g,f) = \mathbb{E}\left[(g(\boldsymbol{X}) - f(\boldsymbol{X}))^2\right] \ ,$$

since $\mathbb{E}[Y|\boldsymbol{X}] = f(\boldsymbol{X})$. Define now the empirical version of $r(g,f)$, that is

$$\begin{aligned} \widehat{r}_n(g,f) &\triangleq \frac{1}{n}\sum_{i=1}^{n}(Y_i - g(\boldsymbol{X}_i))^2 - \frac{1}{n}\sum_{i=1}^{n}(Y_i - f(\boldsymbol{X}_i))^2 \\ &= -\frac{1}{n}\sum_{i=1}^{n} U_i \ , \end{aligned}$$

where $U_i = -(Y_i - g(\boldsymbol{X}_i))^2 + (Y_i - f(\boldsymbol{X}_i))^2$. Notice that the estimator in (3.12) can be written as

$$\widehat{f}_n(\boldsymbol{X},\boldsymbol{Y}) = \arg\min_{g\in\Gamma}\left\{\widehat{r}(g,f^*) + \frac{\lambda}{n}\mathrm{pen}(g)\right\} \ .$$

At this point we are going to apply Theorem 13 to $\{U_i\}_{i=1}^n$. For this we need to verify the moment condition in the Theorem. Begin by noticing that

$$\begin{aligned} U_i &= 2(Y_i - f(\boldsymbol{X}_i))(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i)) - (f(\boldsymbol{X}_i) - g(\boldsymbol{X}_i))^2 \\ &= 2W_i(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i)) - (f(\boldsymbol{X}_i) - g(\boldsymbol{X}_i))^2 \ . \end{aligned} \tag{3.28}$$

The variance of U_i can be easily upper bounded noticing that the U_i is the sum of two uncorrelated terms. The variance of the first term is

$$\begin{aligned} \mathrm{Var}\left(2W_i\left(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i)\right)\right) &= 4\mathrm{Var}(W_i)\mathbb{E}\left[(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2\right] \\ &= 4\mathrm{Var}(W_i)r(g,f) \ . \end{aligned}$$

The variance of the second term is easily bounded by

$$\begin{aligned}
\mathrm{Var}\left((g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2\right) &\leq \mathbb{E}\left[(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^4\right] \\
&\leq 4M^2\mathbb{E}\left[(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2\right] \\
&\leq 4M^2 r(g, f) \ ,
\end{aligned}$$

therefore we conclude that $\mathrm{Var}(U_i) \leq 4(\sigma^2 + M^2)r(g, f)$.

To determine the moment condition constant h we will use a result presented in [50]. Let A and B be two uncorrelated random variables satisfying the moment condition with constants h_A and h_B respectively. Then $A + B$ satisfies the moment condition with constant $2(h_A+h_B)$. We now proceed by considering the decomposition of U_i into two terms, as in (3.28), and checking the moment condition for each one of the these. For the first term we have

$$\begin{aligned}
&\mathbb{E}\left[\left|2W_i(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))\right|^k\right] \\
&\quad = \mathbb{E}\left[|2W_i|^k\right]\mathbb{E}\left[|g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i)|^k\right] \\
&\quad \leq \mathrm{Var}(2W_i)\frac{k!}{2}(2h)^{k-2}\mathbb{E}\left[|g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i)|^2\right](2M)^{k-2} \\
&\quad \leq \mathrm{Var}\left(2W_i\left(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i)\right)\right)\frac{k!}{2}(4hM)^{k-2} \ ,
\end{aligned}$$

for $k \geq 2$. The second term is bounded, and so we have simply

$$\begin{aligned}
&\mathbb{E}\left[\left|(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2 - \mathbb{E}\left[(g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2\right]\right|^k\right] \\
&\leq \quad \mathrm{Var}\left((g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2\right)(4M^2)^{k-2} \\
&\leq \quad \mathrm{Var}\left((g(\boldsymbol{X}_i) - f(\boldsymbol{X}_i))^2\right)\frac{k!}{2}(4M^2/3)^{k-2} \ ,
\end{aligned}$$

for $k \geq 2$. Finally, using the result in [50] we conclude that U_i satisfies Bernstein's moment condition with

$$h_{U_i} = 2(4hM + 4M^2/3) \ .$$

Applying Theorem 13 to $\{U_i\}_{i=1}^n$, with $\tau = \mathrm{pen}(g) + \log(1/\delta)$ and $\epsilon = 1/\lambda$ we get

$$r(g,f) - \widehat{r}_n(g,f) \geq \lambda\frac{\mathrm{pen}(g) + \log(1/\delta)}{n} + \frac{2(\sigma^2 + M^2)r(g,f)}{\lambda(1-c)} \ ,$$

with probability not greater than $\delta e^{-\mathrm{pen}(g)}$. Using the union of events bound we conclude that

$$r(g,f) - \widehat{r}_n(g,f) < \lambda\frac{\mathrm{pen}(g) + \log(1/\delta)}{n} + \frac{2(\sigma^2 + M^2)r(g,f)}{\lambda(1-c)} \ , \tag{3.29}$$

for all $g \in \Gamma$, with probability at least $1 - \delta$. We need to choose c and λ so that the conditions in Theorem 13 hold, therefore take $c = \epsilon h_{U_i} = 8(hM + M^2/3)/\lambda$ and

$\lambda > h_{U_i}$ (so that $c < 1$). Rearranging the terms in (3.29) we get

$$(1-a)r(g,f) < \widehat{r}_n(g,f) + \frac{\lambda}{n}\text{pen}(g) + \frac{\lambda}{n}\log(1/\delta) \ , \tag{3.30}$$

with probability at least $1-\delta$, where $a = \frac{2(\sigma^2+M^2)}{\lambda(1-c)}$. For our purposes it is desirable that $a < 1$. This can be ensured by taking $\lambda > 2(\sigma^2 + M^2) + 8(hM + M^2/3)$.

Taking into account the definition of $\widehat{f}_n$ we have in particular that

$$\begin{aligned}(1-a)r(\widehat{f}_n,f) &< \widehat{r}_n(\widehat{f}_n,f) + \frac{\lambda}{n}\text{pen}(\widehat{f}_n) + \frac{\lambda}{n}\log(1/\delta), \\ &\leq \widehat{r}_n(g,f) + \frac{\lambda}{n}\text{pen}(g) + \frac{\lambda}{n}\log(1/\delta) \ , \end{aligned} \tag{3.31}$$

with probability at least $1-\delta$, for all $g \in \Gamma$. Applying Craig's Theorem once more, but this time to $\{-U_i\}_{i=1}^n$, using $\tau = \log(1/\delta)$, we get

$$\widehat{r}_n(g,f) - r(g,f) < ar(g,f) + \frac{\lambda}{n}\log(1/\delta) \ ,$$

with probability at least $1-\delta$, therefore putting this together with (3.31) conclude that

$$(1-a)r(\widehat{f}_n,f) < (1+a)r(g,f) + \frac{\lambda}{n}\text{pen}(g) + \frac{2\lambda}{n}\log(1/\delta) \ , \tag{3.32}$$

with probability at least $1-2\delta$. Or rearranging the various terms

$$r(\widehat{f}_n,f) < \frac{1+a}{1-a}r(g,f) + \frac{\lambda}{n(1-a)}\text{pen}(g) + \frac{2\lambda}{n(1-a)}\log(1/\delta) \ , \tag{3.33}$$

with probability at least $1 - 2\delta$ for every $g \in \Gamma$. Equation (3.33) is a *Probably Approximately Correct* (PAC) bound. It can easily be converted to an expected risk bound by a standard integration argument, using the fact that $\mathbb{E}[Z] = \int_0^\infty \Pr(Z > t)\mathrm{d}t$, for an arbitrary non-negative random variable Z. To simplify the presentation let

$$\Upsilon(g, f) \triangleq \frac{1+a}{1-a} r(g, f) + \frac{\lambda}{n(1-a)} \mathrm{pen}(g) \ ,$$

and set $\delta = e^{-\frac{n(1-a)}{2\lambda}t}$. Then

$$\begin{aligned} \mathbb{E}\left[r(\widehat{f}_n, f) - \Upsilon(g, f)\right] &\leq \int_0^\infty \Pr\left(r(\widehat{f}_n, f) - \Upsilon(g, f) \geq t\right) \\ &\leq \int_0^\infty 2e^{-\frac{n(1-a)}{2\lambda}t} \\ &= \frac{4\lambda}{n(1-a)} \ , \end{aligned}$$

for every $g \in \Gamma$, yielding the final result. □

3.6.3 Proof of Corollary 1

Check the moment condition for a Gaussian random variable. The moments of W_i are given by

$$\mathbb{E}[|W_i|^k] = \sigma^k \begin{cases} \prod_{i=1}^{k/2} (2i-1) & \text{, if } k \text{ is even,} \\ \\ \sqrt{\frac{2}{\pi}} \prod_{i=1}^{(k-1)/2} (2i) & \text{, if } k \text{ is odd,} \end{cases} .$$

Using this fact one concludes that the moment condition is satisfied with $h = \frac{2}{3}\sqrt{\frac{2}{\pi}}\sigma$. Now by choosing the particular value of λ in the corollary statement we obtain the final result. $\square$

3.6.4 Proof of Theorem 9

As mentioned before we are going to analyze a modification of the estimator described in (3.20). The modification entails to construction of a discrete analogue of $\Xi(\pi)$ (see equation (3.16)). We do this by restricting the decorating constant to lie on the set

$$\mathcal{Q}_n \stackrel{\Delta}{=} \{-M, -M\frac{n-1}{n}, \ldots, M\frac{n-1}{n}, M\} \ .$$

Define

$$\Xi_{\mathcal{Q}_n}(\pi) = \left\{ \sum_{A\in\pi} c_A \mathbf{1}_A(\boldsymbol{x}) : \ C_A \in \mathcal{Q}_n \ \forall A, i \right\} \ .$$

The class of possible estimators we consider is

$$\Gamma = \bigcup_{\pi\in\Pi} \Xi_{\mathcal{Q}_n}(\pi) \ . \tag{3.34}$$

This is clearly a countable set (although not finite), and is the set of models we will use to apply Corollary 1.

To construct a penalty function that satisfies the Kraft inequality (3.11) we use an explicit description of a prefix encoding of the elements of Γ, therefore automatically satisfying (3.11). Let $\gamma_\pi \in \Xi_{\mathcal{Q}_n}(\pi) \subseteq \Gamma$. The encoding of an element of γ_π is done

in two steps: (i) encoding the underlying RDP π, (ii) encoding the decorating leaf constant values. To encode the underlying RDP we resort to its tree representation (refer to Figure 3.6), and assign a zero or one value to each node of the tree: zero if that node is a leaf node, and one otherwise. Now collect all those values in a lexicographical order, that is, left-to-right breadth-first order (see example in Figure 3.6). This forms a binary prefix code for that space of RDP trees. Note that each node in a RDP tree has either zero or 2^d descendants, therefore the tree has $1 + 2^d k$ nodes, for some $k \in \mathbb{N}_0$, and it has $1 + (2^d - 1)k$ leaf nodes. The number of leaf nodes is the size of the RDP and so, we can describe a RDP π using

$$\lambda_1(\pi) \stackrel{\Delta}{=} 1 + \frac{2^d}{2^d - 1}(|\pi| - 1)$$

bits. Notice that since this is a binary prefix code it satisfies the Kraft inequality (for binary codes) $\sum_{\pi \in \Pi} 2^{-\lambda_1(\pi)} \leq 1$. For each element of the RDP we consider a constant value in the set $\mathcal{Q}_n$, therefore therefore $\Xi_{\mathcal{Q}_n}$ has $(2n+1)^{|\pi|}$ elements. With this at hand we consider the penalty

$$\text{pen}(\gamma_\pi) = \left(\frac{2^d \log 2}{2^d - 1} + \log(2n+1) \right) |\pi| \ ,$$

identical to the penalty defined in Equation (3.19). We have the following result

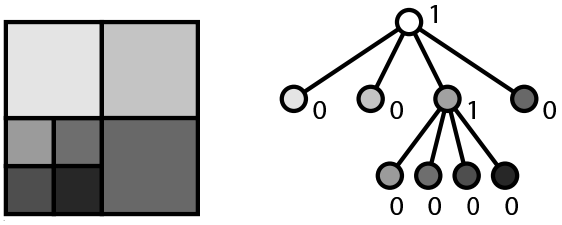


Figure 3.6: Prefix encoding of a Recursive Dyadic Partition. The depicted partition encodes as 100100000 in binary.

Lemma 4. *The penalty above (defined in* (3.19)*) satisfies* (3.11)*, that is*

$$\sum_{\gamma\in\Gamma}\exp(-\mathrm{pen}(\gamma)) \le 1$$

for Γ *defined in* (3.34)*.*

Proof.

$$\begin{aligned}
\sum_{\gamma\in\Gamma}\exp(-\mathrm{pen}(\gamma)) &= \sum_{\bigcup_{\pi\in\Pi}\Xi_{\mathcal{Q}_n}(\pi)} \exp(-\mathrm{pen}(\gamma)) \\
&= \sum_{\pi\in\Pi}\sum_{\gamma\in\Xi_{\mathcal{Q}_n}(\pi)} \exp\left(-\frac{2^d\log 2}{2^d-1}|\pi| - \log(2n+1)|\pi|\right) \\
&= \sum_{\pi\in\Pi}\exp\left(-\frac{2^d\log 2}{2^d-1}|\pi|\right)\sum_{\gamma\in\Xi_{\mathcal{Q}_n}(\pi)}\frac{1}{(2n+1)^{|\pi|}} \\
&= \sum_{\pi\in\Pi} 2^{-\frac{2^d}{2^d-1}|\pi|} \\
&\le \sum_{\pi\in\Pi} 2^{-\left(1+\frac{2^d}{2^d-1}(|\pi|-1)\right)} = \sum_{\pi\in\Pi} 2^{-\lambda_1(\pi)} \le 1\ .
\end{aligned}$$

□

Recall the comment make after (3.19): it was noted that $\mathrm{pen}(\gamma_\pi) \le c_{\mathrm{pen}}|\pi|\log n$ for any $n \ge 2$ and a suitable constant c_{pen}. We use this bound on the penalty, to

avoid dealing with cumbersome constants.

We are now ready to apply Corollary 1. Let $f \in \mathrm{PC}(\beta, M)$ be fixed, but arbitrary. Our strategy is to construct a partition that is well adapted to the boundary set $B(f)$, in the sense that the partition elements that intersect $B(f)$ are small. This is desirable because the discontinuity of $B(f)$ cannot be well approximated with a constant. Away from the boundary we can use larger partition elements.

Let $J \in \mathbb{N}_0$. Consider the RDP tree with all the leafs at depth J. The corresponding RDP has 2^{dJ} elements. Now prune this tree so that leafs intersecting $B(f)$ are at depth J and all the other leafs are possibly a a shalower depth. This process is illustrated in Figure 3.7. We have the following result:

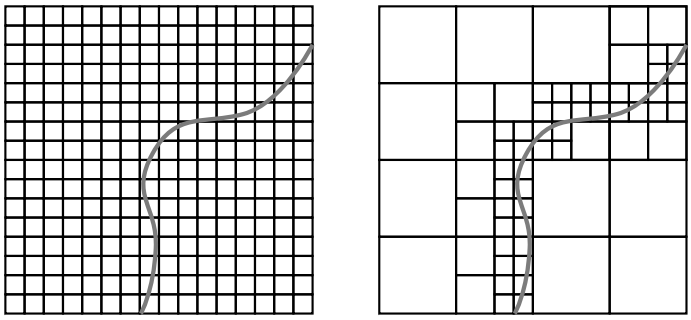
Lemma 5. *There is a RDP such that leafs intersecting $B(f)$ are at depth J and all the other leafs are depths no greater than J. Denote the smallest such RDP by π_J^*. This RDP has at most $2^{2d}\beta 2^{(d-1)J}$ leafs intersecting $B(f)$ and*

$$|\pi_J^*| \leq \begin{cases} \beta' J & , \text{ if } d = 1 \\ \beta' 2^{(d-1)J} & , \text{ if } d > 1 \end{cases},$$

where

$$\beta' = \begin{cases} 2^{2d}\beta & , \text{ if } d = 1 \\ \frac{2^{3d-1}}{2^{d-1}-1}\beta & , \text{ if } d > 1 \end{cases}.$$

Proof. The number of leafs in a tree is trivially bounded by the number of nodes in the tree, and so the proof strategy entails by bounding from above the number of



(a) (b)

Figure 3.7: Example of RDP tree pruning, for $d = 2$, $J = 4$, and $J' = 2$. The depicted curve is $B(f)$: (a) partition with all leafs at depth J; (b) pruned partition adapted to $B(f)$.

nodes π_J^* might have. Let $j \in \mathbb{N}_0$. Begin by noticing that any closed ball of diameter 2^{-j} is contained in at most 2^d nodes at depth j, thus at depth j there are at most $2^d \beta 2^{(d-1)j}$ nodes of π_J that intersect $B(f)$ (recall Definition 4). Due to the diadic structure every leaf has $2^d - 1$ siblings, thus π_J^* has less than $2^{2d}\beta 2^{(d-1)j}$ nodes at depth j, (since $2^{2d} \leq 2^d - 1$). Taking this into account it is clear that π_J^* has at most $2^{2d}\beta 2^{(d-1)J}$ leafs. Finally, the total number of nodes of π_J is bounded from above by the sum of the number of nodes at each depth, that is

$$\begin{aligned} |\pi_J^*| &\leq 2^{2d}\beta \sum_{j=J'+1}^{J} 2^{(d-1)j} \\ &\leq 2^{2d}\beta \sum_{j=0}^{J} 2^{(d-1)j} \\ &\leq \begin{cases} 2^{2d}\beta J & , \text{if } d = 1 \\ \frac{2^{3d-1}}{2^{d-1}-1}\beta 2^{(d-1)J} & , \text{if } d > 1 \end{cases} . \end{aligned}$$

□

The key point of Lemma 5 is that the total number of leafs in the described tree has same order of magnitude as the number of leafs intersecting the boundary set (for $d > 1$).

Let π_J^* be the partition of Lemma 5 and define

$$\bar{f} = \sum_{A \in \pi_J^*} \bar{c}_A \mathbf{1}_A(\boldsymbol{x}) \ ,$$

where

$$\bar{c}_A = \frac{1}{\mathrm{Vol}(A)} \int_A f(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \ .$$

It is easy to verify that $\bar{f}$ minimizes $\|f - \bar{f}\|^2$ over $\Xi(\pi_J^*)$, and it is clear that $|\bar{c}_A| \leq M, \ \forall A \in \pi$. Finally, define

$$f' = \sum_{A \in \pi_J^*} c_A \mathbf{1}_A(\boldsymbol{x}) \ ,$$

where $c_A = M \cdot \mathrm{round}(n\bar{c}_A/M)/n$. Clearly $c_A \in \mathcal{Q}_n$, and so $f' \in \Gamma$. This is the estimate we are going to use in the oracle bound (5.1).

We need to bound $\|f - f'\|^2$. We consider first the case $d > 1$. For the ease of notation define let $\pi^*(B(f)) \stackrel{\Delta}{=} \{A \in \pi_J^* : A \bigcap B(f) \neq \emptyset\}$, the set of elements of π_J^*

that intersect $B(f)$. Then

$$\begin{aligned}
\|f-\bar{f}\|^2 &= \int_{[0,1]^d} |f(\boldsymbol{x})-\bar{f}(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \\
&= \sum_{A\in\pi_J^*} \int_A |f(\boldsymbol{x})-\bar{f}(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \\
&= \sum_{A\in\pi^*(B(f))} \int_A |f(\boldsymbol{x})-\bar{f}(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} + \sum_{A\in\pi_J^*\setminus\pi^*(B(f))} \int_A |f(\boldsymbol{x})-\bar{f}(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \\
&\leq \sum_{A\in\pi^*(B(f))} M^2 \mathrm{Vol}(A) + \sum_{A\in\pi_J^*\setminus\pi^*(B(f))} \int_A |f(\boldsymbol{x})-\bar{c}_A|^2 \mathrm{d}\boldsymbol{x} \qquad (3.35) \\
&\leq \beta' 2^{(d-1)J} M^2 2^{-dJ} \\
&\leq \beta' M^2 2^{-J} \ , \qquad (3.36)
\end{aligned}$$

where step (3.35) follows from Lemma 5 and the fact that f and $\bar{f}$ are equal "away" from the boundary (*i.e.*, are equal within the elements of $\pi_J^* \setminus \pi^*(B(f))$). We have also

$$\begin{aligned}
\|\bar{f}-f'\|^2 &= \int_{[0,1]^d} |\bar{f}(\boldsymbol{x})-f'(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \\
&= \sum_{A\in\pi_J^*} \int_A |\bar{f}(\boldsymbol{x})-f'(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \\
&= \sum_{A\in\pi_J^*} \int_A |\bar{c}_A - c_A|^2 \mathrm{d}\boldsymbol{x} \\
&= \sum_{A\in\pi_J^*} (\bar{c}_A - c_A)^2 \mathrm{Vol}(A) \\
&\leq \frac{M^2}{4n^2} \ .
\end{aligned}$$

Finally

$$\begin{aligned} \|f - f'\|^2 &\leq \|f - \bar{f}\|^2 + \|\bar{f} - f'\|^2 + 2\|f - \bar{f}\|\|\bar{f} - f'\| \\ &\leq \text{const}(\beta, M) \max\left\{2^{-J}, \frac{1}{n}\right\} , \end{aligned}$$

where $\text{const}(\beta, M) > 0$ is a suitable constant. Corollary 1 yields

$$\mathbb{E}[\|f - \widehat{f}_n\|^2] \leq \text{const}(\beta, M, \sigma^2) \max\left\{2^{-J}, \frac{2^{(d-1)J} \log n}{n}, \frac{1}{n}\right\} ,$$

for a suitable $\text{const}(\beta, M) > 0$ and any $n \geq 2$, where the middle argument in the maximum function follows from Lemma 5 and the choice of penalty function. Choosing $J = \lceil \frac{1}{d} \log(n/\log(n)) \rceil$ yields the desired result

$$\mathbb{E}[\|f - \widehat{f}_n\|^2] \leq \text{const}(\beta, M, \sigma^2) \left(\frac{n}{\log n}\right)^{-\frac{1}{d}} ,$$

for some $\text{const}(\beta, M, \sigma^2) > 0$ and all $n \geq 2$.

When $d = 1$ the argument suffers a slight modification. Instead of (3.36) we have

$$\|f - \bar{f}\|^2 \leq \beta' M^2 J 2^{-J} ,$$

which follows by the same reasoning as before but noting that Lemma 5 gives a

different expression for $|\pi^*(B(f))|$. From Corollary 1 we get

$$\mathbb{E}[\|f - \widehat{f}_n\|^2] \leq \text{const}(\beta, M, \sigma^2) \max\left\{ J2^{-J}, \frac{J \log n}{n}, \frac{1}{n} \right\} .$$

With $J = \lceil \log n \rceil$ we obtain

$$\mathbb{E}[\|f - \widehat{f}_n\|^2] \;\; \leq \;\; \text{const}(\beta, M, \sigma^2) \frac{\log^2 n}{n} ,$$

for some $\text{const}(\beta, M, \sigma^2) > 0$ and all $n \geq 2$, concluding the proof. □

3.6.5 Sketch Proof of Theorem 10

The proof of the theorem hinges on the comparison of the active learning algorithm with a different/alternative passive learning algorithm. The alternative passive algorithm is such that it has the same performance of the regular passive learning algorithm presented in Section 3.4.1. Begin by constructing an alternate set of sample points: divide the unit hypercube in 2^{dJ} equal hypercubes (this corresponds to a RDP with all the leafs at depth J), and inside each hypercube take $n'/2^{dJ}$ uniformly distributed samples. By the end of this process one has a total of n' samples $\{\boldsymbol{X'}_i\}_{i=1}^{n'}$ distributed in a somewhat uniform manner over the unit hypercube. This sampling design is completely non-adaptive and it is called the *jittered regular grid design* [34]. Furthermore assume these sample points are independent of $\{\boldsymbol{X}_i, Y_i\}_{i=1}^{n}$. Let $\{Y'_i\}_{i=1}^{n'}$

be the corresponding observations. Recall (3.17) and define

$$\widehat{f}_{\dagger}^{(\pi)} \triangleq \arg\min_{g \in \Xi(\pi)} \frac{1}{n'} \sum_{i=1}^{n'} (Y_i' - g(\boldsymbol{X'}_i))^2 \ . \tag{3.37}$$

Now define

$$\widehat{\pi}_{\dagger} \stackrel{\Delta}{=} \arg\min_{\pi \in \Pi} \left\{ \frac{1}{n} \sum_{i=1}^{n'} (Y_i - \widehat{f}_{\dagger}^{(\pi)}(\boldsymbol{X}_i))^2 + \frac{\lambda}{n} \mathrm{pen}\left(\widehat{f}_{\dagger}^{(\pi)}\right) \right\} \ ,$$

where the penalty is defined as before. Finally let $\widehat{f}_{\dagger} \stackrel{\Delta}{=} \widehat{f}_{\dagger}^{(\widehat{\pi}_{\dagger})}$. It can be shown that this algorithm achieves the error rate of Theorem 9, using for example the techniques in [45]. Namely we have

$$\sup_{f \in \mathrm{PC}(\beta, M)} \mathbb{E}[\|\widehat{f}_{\dagger} - f\|^2] \ \leq \ \mathrm{const}(\beta, M, \sigma^2) \left(\frac{\log n'}{n'}\right)^{\frac{1}{d}} \ , \tag{3.38}$$

where $\mathrm{const}(\beta, M, \sigma^2) > 0$. Then

We now compare the performance of the algorithm just described and the active learning algorithm proposed. Let $\mathrm{NR} \subset [0,1]^d$ be the the region of the domain that

is not going to be resampled, that is $\mathrm{NR} \triangleq \left\{ \boldsymbol{x} \in [0,1]^d : \boldsymbol{x} \notin \bigcup \widehat{\mathcal{R}} \right\}$.

$$
\begin{aligned}
\mathbb{E}\left[\|\widehat{f}_{\mathrm{ACTIVE}} - f\|^2 \right] &= \mathbb{E}\left[\mathbb{E}\left[\|\widehat{f}_{\mathrm{ACTIVE}} - f\|^2 \middle| \widehat{\mathcal{R}} \right] \right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\sum_{A \in \widehat{\mathcal{R}}} \int_A |\widehat{f}_{\mathrm{ACTIVE}}(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} + \right. \right. \\
&\qquad \left. \left. \int_{\mathrm{NR}} |\widehat{f}_{\mathrm{ACTIVE}}(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \middle| \widehat{\mathcal{R}} \right] \right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\sum_{A \in \widehat{\mathcal{R}}} \int_A |\widehat{f}^r_A(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} + \int_{\mathrm{NR}} |\widehat{f}^p_0(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \;\middle|\; \widehat{\mathcal{R}} \right] \right] \\
&\leq \mathbb{E}\left[\sum_{A \in \widehat{\mathcal{R}}} \mathbb{E}\left[\int_A |\widehat{f}^r_A(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \;\middle|\; \widehat{\mathcal{R}} \right] \right] + \\
&\quad \mathbb{E}\left[\mathbb{E}\left[\int_{[0,1]^d} |\widehat{f}^p_0(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \;\middle|\; \widehat{\mathcal{R}} \right] \right] .
\end{aligned}
$$

The second term can be immediately bounded by the performance of the preview stage estimator, which is given by Theorem 9. For the first term notice that when we do a refinement the number of samples in the refined cell is going to be larger than $n'/2^{dJ}$. Therefore

$$
\mathbb{E}\left[\mathbb{E}\left[\int_A |\widehat{f}^r_A(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \middle| \widehat{\mathcal{R}} \right] \right] \leq \mathbb{E}\left[\mathbb{E}\left[\int_A |\widehat{f}_\dagger(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \middle| \widehat{\mathcal{R}} \right] \right] .
$$

In conclusion we have that

$$
\begin{aligned}
\mathbb{E}\left[\mathbb{E}\left[\int_A |\widehat{f}^r_A(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \middle| \widehat{\mathcal{R}} \right] \right] &\leq \mathbb{E}\left[\mathbb{E}\left[\int_A |\widehat{f}_\dagger(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \middle| \widehat{\mathcal{R}} \right] \right] \\
&\leq \mathbb{E}\left[\int_{[0,1]^d} |\widehat{f}_\dagger(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x} \right] .
\end{aligned}
$$

This term is bounded by (3.38), concluding the proof. □

3.6.6 Proof of Theorem 11

We begin by characterizing the behavior of the preview estimators at resolution J. In order to do that we need to introduce some concepts. Consider the partition of $[0,1]^d$ into 2^{dJ} identical hypercubes, with $J \in \mathbb{N}$, like in the definition of boundary detectability. This partition, denoted by $\mathcal{C}_J$, can also be constructed with a RDP where all the leaves are at depth J. Using this partition we define $f_J : [0,1]^d \to \mathbb{R}$, a coarse approximation of f up to resolution J. Formally, we have

$$f_J(\boldsymbol{x}) = 2^{dJ} \sum_{A \in \pi_J} \left(\int_A f(\boldsymbol{t}) \mathrm{d}\boldsymbol{t} \right) \mathbf{1}_A(\boldsymbol{x}) \ .$$

Note that f_J is identical to f "away" from the boundary (since f is piecewise constant), but in the vicinity of the boundary there is some averaging. We need to define also a similar J^{th} resolution version of the preview estimators. Recall the definition of $\textsc{pruned}_J$ on page 96 and define

$$\widehat{f}^p_{l,J} \stackrel{\Delta}{=} \widehat{f}_{n'}(\textsc{pruned}_J(\widehat{\pi}^p_l)) \ ,$$

where $\widehat{f}_{n'}$ was defined in (3.17) on page 88. We have the following important result.

Lemma 6.

$$\mathbb{E}\left[\|\widehat{f}^p_{l,J} - f_J\|^2 \right] \leq \mathrm{const}(\beta, M, \sigma^2) \frac{2^{(d-1)J} \log n'}{n'} \ ,$$

for a suitable $\text{const}(\beta, M, \sigma^2) > 0$*, and all* $n' \geq 2$*.*

Proof. The key idea used in the proof is the construction of a modified observation setup, that is, instead of using $\{\boldsymbol{X}_i, Y_i\}_{i=1}^{n'}$ to determine the estimator $\widehat{f}_l^P$, we use a different observation model, yielding observations $\{\boldsymbol{X}'_i, Y'_i\}_{i=1}^{n'}$. This new observation model is carefully chosen so that the outcome of $\widehat{f}_{l,J}^p$ when using the data set $\{\boldsymbol{X}_i, Y_i\}_{i=1}^{n'}$ or $\{\boldsymbol{X}'_i, Y'_i\}_{i=1}^{n'}$ is statistically indistinguishable.

Take $\boldsymbol{X}'_i$ uniformly over $[0,1]^d$. The new observation model is of the form

$$Y'_i = f_J(\boldsymbol{X}'_i) + W'_i \ ,$$

where $\{W'_i\}$ are all independent but not identically distributed. Namely let $A_{\boldsymbol{X}'_i}$ denote the partition set where $\boldsymbol{X}'_i$ is contained, that is,

$$A_{\boldsymbol{X}'_i} \stackrel{\Delta}{=} A \text{ such that } A \in \mathcal{C}_J \text{ and } \boldsymbol{X}'_i \in A \ .$$

Define

$$W'_i \stackrel{\Delta}{=} f(\boldsymbol{U}_i) - f_J(\boldsymbol{U}_i) + W_i \ ,$$

where $\{\boldsymbol{U}_i\}_{i=1}^{n'}$ are all independent of $\{W_i\}$ and $\boldsymbol{U}_i | \boldsymbol{X}_i \ \sim \text{Unif}(A_{\boldsymbol{X}_i})$, and $\{W_i\}_{i=1}^{n'}$ are (i.i.d.) Gaussian with zero mean and variance σ^2.

Notice that the estimators $\widehat{f}_{l,J}^p$ average the data within each partition cell $A \in \pi_J$, completely ignoring the sample location $\boldsymbol{X}'_i$ within the cell. This ensures that the

above observation model is statistically indistinguishable from the original observation model, when used by the estimation procedure. Note that under the new observation model the regression function is $\mathbb{E}[Y_i'|\boldsymbol{X}_i' = \boldsymbol{x}]$ is $f_J(\boldsymbol{x})$, instead of $f(\boldsymbol{x})$ for the original observation model. This is the key to obtain the desired result, following from the application of Theorem 8, since now we can evaluate the error performance with respect to f_J. We just need to check that W_i' satisfies the moment condition. Since W_i' is the sum of two independent random variables, namely W_i and $f(\boldsymbol{U}_i) - f_J(\boldsymbol{U}_i)$ we can again use the result in [50] for the sum of random variables satisfying the moment condition (as in the proof of Theorem 8). Therefore W_i' satisfies the moment condition (3.10) with constant $h = 2(\frac{2}{3}\sqrt{\frac{2}{\pi}} + \frac{4M^2}{3})$. From here we proceed as in the proof of Theorem 9, by noting that there is a model in Γ, built over a partition with $2^{(d-1)J}$ elements, that approximates f_J extremely well. We conclude that

$$\begin{aligned}\mathbb{E}\left[\|\widehat{f}_l^p - f_J\|^2\right] &\leq \text{const}(\beta, M, \sigma^2)\max\left\{\frac{2^{(d-1)J}\log n'}{n'}, \frac{1}{n'}\right\} \\ &= \text{const}(\beta, M, \sigma^2)\frac{2^{(d-1)J}\log n'}{n'}\ ,\end{aligned}$$

for a suitable $\text{const}(\beta, M, \sigma^2) > 0$, and all $n' \geq 2$. □

To bound the risk of the active learning procedure we are going to consider the error incurred in three different situations: (i) the error incurred during the preview step in regions away from the boundary that are not refined; (ii) the error incurred by not detecting a piece of the boundary (and therefore not performing the refinement stage on that area); (iii) the error incurred during the refinement step. Recall that $\mathcal{C}_J$

it the partition of $[0,1]^d$ into 2^{dJ} identical hypercubes. Let $\mathcal{C}_J^B$ denote the partition sets of $\mathcal{C}_J$ intersecting the boundary, and $\mathcal{C}_J^{\overline{B}}$ denote the remaining partition sets, so that $\mathcal{C}_J = \mathcal{C}_J^B \cup \mathcal{C}_J^{\overline{B}}$. Errors (i), (ii) and (iii) correspond respectively to the errors in $\left(\bigcup \mathcal{C}_J^{\overline{B}}\right) \setminus \widehat{\mathcal{R}}$, $\bigcup \left(\mathcal{C}_J^B\right) \setminus \widehat{\mathcal{R}}$, and $\bigcup \widehat{\mathcal{R}}$.

(i) - Recall that f_J is identical to f "away" from the boundary set $B(f)$. That is, for a fixed set $A \in \mathcal{C}_J$ that does not intersect the boundary we have $f(\boldsymbol{x}) = f_J(\boldsymbol{x})$ for all $\boldsymbol{x} \in A$. Note also that $\widehat{f}_{l,J}^p(\boldsymbol{x})$ is identical to $\widehat{f}_l^p(\boldsymbol{x})$ for any point $\boldsymbol{x} \notin \bigcup \widehat{\mathcal{R}}$, that is, any region that is not going to be refined. Taking this into consideration we have, using Lemma 6

$$
\begin{aligned}
&\mathbb{E}\left[\int_{(\bigcup \mathcal{C}_J^{\overline{B}}) \setminus \widehat{\mathcal{R}}} |\widehat{f}_{\text{ACTIVE}}(\boldsymbol{x}) - f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x}\right] \\
&= \mathbb{E}\left[\int_{(\bigcup \mathcal{C}_J^{\overline{B}}) \setminus \widehat{\mathcal{R}}} |\widehat{f}_0^p(\boldsymbol{x}) - f_J(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x}\right] \\
&= \mathbb{E}\left[\int_{(\bigcup \mathcal{C}_J^{\overline{B}}) \setminus \widehat{\mathcal{R}}} |\widehat{f}_{0,J}^p(\boldsymbol{x}) - f_J(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x}\right] \\
&\leq \mathbb{E}\left[\|f_{0,J}^p - f_J\|^2\right] \\
&\leq \text{const}(\beta, M, \sigma^2) \frac{2^{(d-1)J} \log n'}{n'} \ ,
\end{aligned}
$$

for a suitable $\text{const}(\beta, M, \sigma^2) > 0$ and all $n' \geq 2$.

(ii) - The set of cells in $\mathcal{C}_J$ intersecting the boundary that are not going to be re-sampled (*i.e.*, refined) in the refinement step is given by $\mathcal{C}_J^B \setminus \widehat{\mathcal{R}}$. In other words

$$
\mathcal{C}_J^B \setminus \widehat{\mathcal{R}} = \{A \in \mathcal{C}_J : A \cap B(f) \neq \emptyset,\ A \notin \textsc{pruned}_J(\widehat{\pi}_l^p)\ \forall l \in \{0, \ldots, d\}\} \ .
$$

Assume now that J is large enough, so that the detectability condition holds. We know that for each cell in $\mathcal{C}_J^B$ equation (3.24) holds for at least one of the shifted RDPs. Therefore we can construct a decomposition

$$\mathcal{C}_J^B = \mathcal{C}_{J,0}^B \cup \mathcal{C}_{J,1}^B \cup \cdots \cup \mathcal{C}_{J,d}^B \ ,$$

where $\mathcal{C}_{J,l}^B$ are disjoint and (3.24) holds for all the cells of $\mathcal{C}_{J,l}^B$, with respect to shifted RDPs in the l^{th} coordinate (Notice that many such decompositions might exist, but for our purposes we just need to consider one of them). A simple constructive way of building this decomposition is the following: Define $\mathcal{C}_{J,0}^B$ as all the cells in $\mathcal{C}_J^B$ for which (3.24) holds with $l = 0$. Then define $\mathcal{C}_{J,1}^B$ as all the cells in $\mathcal{C}_J^B \setminus \mathcal{C}_{J,0}^B$ for which (3.24) holds with $l = 1$, and so on. Using such

a decomposition we have the following result

$$
\begin{aligned}
\mathbb{E}[\|\widehat{f}^{p}_{l,J} - f_J\|^2] &\geq \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}}\mathcal{A}(A,l)} \left(\widehat{f}^{p}_{l,J}(\boldsymbol{x}) - f_J(\boldsymbol{x})\right)^2 \mathrm{d}\boldsymbol{x}\right] \\
&= \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}}\mathcal{A}(A,l)} \left(\widehat{f}^{p}_{l}(\boldsymbol{x}) - f_J(\boldsymbol{x})\right)^2 \mathrm{d}\boldsymbol{x}\right] \\
&= \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}}\mathcal{A}(A,l)} \left(f_J(\boldsymbol{x}) - \mathbb{E}[\widehat{f}^{p}_{l}(\boldsymbol{x})|\widehat{\pi}^{p}_{l}]\right)^2 \mathrm{d}\boldsymbol{x}\right] + \\
&\quad \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}}\mathcal{A}(A,l)} \left(\widehat{f}^{p}_{l}(\boldsymbol{x}) - \mathbb{E}[\widehat{f}^{p}_{l}(\boldsymbol{x})|\widehat{\pi}^{p}_{l}]\right)^2 \mathrm{d}\boldsymbol{x}\right] \\
&\geq \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}}\mathcal{A}(A,l)} \left(f_J(\boldsymbol{x}) - \mathbb{E}[\widehat{f}^{p}_{l}(\boldsymbol{x})|\widehat{\pi}^{p}_{l}]\right)^2 \mathrm{d}\boldsymbol{x}\right] \\
&\geq \mathbb{E}\left[\left|\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}\right|\right] C^*\ 2^{-dJ}\ .
\end{aligned}
$$

Using Lemma 6 we conclude that $\mathbb{E}\left[\left|\mathcal{C}^B_{J,l}\setminus\widehat{\mathcal{R}}\right|\right] \leq \frac{2^{(2d-1)J}\log n'}{n'}\frac{1}{C^*}$, and therefore $\mathbb{E}\left[\left|\mathcal{C}^B_{J}\setminus\widehat{\mathcal{R}}\right|\right] \leq (d+1)\frac{2^{(2d-1)J}\log n'}{n'}\frac{1}{C_b(f)}$. The maximum error we incur in our final estimate by erroneously not detecting certain pieces of the boundary is

bounded above by

$$
\begin{aligned}
&\mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}_J^B\setminus\widehat{\mathcal{R}}} A}\left(\widehat{f}_{\text{ACTIVE}}(\boldsymbol{x})-f(\boldsymbol{x})\right)^2\mathrm{d}\boldsymbol{x}\right]\\
&= \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}_J^B\setminus\widehat{\mathcal{R}}} A}\left(\widehat{f}_0^p(\boldsymbol{x})-f(\boldsymbol{x})\right)^2\mathrm{d}\boldsymbol{x}\right]\\
&= \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}_J^B\setminus\widehat{\mathcal{R}}} A}\left(f(\boldsymbol{x})-\mathbb{E}[\widehat{f}_0^p(\boldsymbol{x})]\right)^2\mathrm{d}\boldsymbol{x}\right]+\\
&\quad\ \mathbb{E}\left[\int_{\bigcup_{A\in\mathcal{C}_J^B\setminus\widehat{\mathcal{R}}} A}\left(\widehat{f}_0^p(\boldsymbol{x})-\mathbb{E}[\widehat{f}_0^p(\boldsymbol{x})]\right)^2\mathrm{d}\boldsymbol{x}\right]\\
&\leq \mathbb{E}\left[\left|\mathcal{C}_J^B\setminus\widehat{\mathcal{R}}\right|\right]M^2 2^{-dJ}+\text{const}(\beta,M,\sigma^2)\frac{2^{(d-1)J}\log n'}{n'}\\
&\leq \text{const}(\beta,M,\sigma^2,C^*)\frac{2^{(d-1)J}\log n'}{n'}\ ,
\end{aligned}
$$

where $\text{const}(\beta, M, \sigma^2, C^*) > 0$, and $\text{const}(\beta, M, \sigma^2) > 0$ comes from Lemma 6. The first equality comes simply from (3.23), the definition of the active estimator.

We conclude that the error incurred by failing to detect the boundary has the same contribution for the total error of the estimator as the error away from the boundary, analyzed in (i).

(iii) - In the regions that are going to be refined, that is, the regions in $\widehat{\mathcal{R}}$, we collect further samples and apply the estimator described in Section 3.4.1. Assume for now that we have $O(2^{(d-1)J})$ elements in $\widehat{\mathcal{R}}$. This is proved in Lemma 7 below.

We collect a total of $L \triangleq n'/|\widehat{\mathcal{R}}|$ samples in each element of $\widehat{\mathcal{R}}$. The error incurred by $\widehat{f}^r_A$, the refinement estimator for set $A \in \widehat{\mathcal{R}}$, is upper-bounded by

$$\text{const}(\beta, M, \sigma^2) \left(\frac{\log L}{L}\right)^{1/d} 2^{-dJ} ,$$

where $const(\beta, M, \sigma^2) > 0$ comes from Theorem 9. Therefore the error of the estimator over $\cup_{A\in\widehat{\mathcal{R}}} A$ is upper-bounded by

$$\text{const} \left(\frac{\log L}{L}\right)^{1/d} 2^{-dJ} |\widehat{\mathcal{R}}| .$$

To compute the total error incurred by $\widehat{f}_{\text{ACTIVE}}$ we just have to sum the contributions of (i), (ii) and (iii), and therefore we get

$$\begin{aligned} &\mathbb{E}\left[\|\widehat{f}_{\text{ACTIVE}} - f\|^2\right] \\ &\quad \leq \quad \text{const}(\beta, M, \sigma^2, C^*) \left(\left(\frac{\log L}{L}\right)^{1/d} 2^{-dJ} |\widehat{\mathcal{R}}| + \frac{2^{(d-1)J} \log n'}{n'}\right) , \end{aligned} \tag{3.39}$$

with $\text{const}(\beta, M, \sigma^2, C^*) > 0$. Assuming now that $|\widehat{\mathcal{R}}| = O(2^{(d-1)J})$ we can balance the two terms in the above expression by choosing

$$J = \left\lceil \frac{d-1}{(d-1)^2 + d} \log(n'/\log(n')) \right\rceil ,$$

yielding the desired result.

As mentioned before, we need to show that the number of partition sets requiring

refinement is not very large. Namely, with high probability $|\widehat{\mathcal{R}}| = O(2^{(d-1)J})$. This ensures that there are enough samples in each one of these partition sets to properly perform the refinement step. Without loss of generality, it suffices to analyze the number of elements in $\textsc{pruned}_J(\widehat{\pi}_0^p)$. We will denote this estimator by $\widehat{\pi}$ to ease the notation.

Lemma 7. *Let $\widehat{\pi}$ be the partition estimated according to* (3.20)*. Let π_J^* be the partition adapted to the boundary, according to Lemma 5 (recall that this cannot computed from the data). Then with high probability the number of elements of $\widehat{\pi}$ is comparable with the number of elements of π_J^*, namely*

$$\Pr(|\widehat{\pi}| > 2|\pi_J^*|) \leq 1/n \ ,$$

for n sufficiently large.

From this lemma we conclude that, with high probability, the number of cells to be refined is actually $O(2^{(d-1)J})$ and so all the analysis done before holds, with probability $1 - 1/n$, concluding the proof of Theorem 11. □

Proof of Lemma 7: First note that the number of elements in $\widehat{\pi}$ is equal to $(2^d-1)k+1$ for $k \in \mathbb{N}_0$, due to the dyadic structure of the partitions. We begin by bounding the

probability that $\widehat{\pi}$ has certain number of elements.

$$\begin{aligned}\Pr\left(|\widehat{\pi}| = (2^d-1)k+1\right) &= \Pr\left(\bigcup_{\pi:|\pi|=(2^d-1)k+1} \{\widehat{\pi}=\pi\}\right) \\ &\leq \sum_{\pi:|\pi|=(2^d-1)k+1} \Pr\left(\widehat{\pi}=\pi\right) \\ &\leq \#(k)\cdot \max_{\pi:|\pi|=(2^d-1)k+1} \Pr\left(\widehat{\pi}=\pi\right) \ , \end{aligned} \tag{3.40}$$

where $\#(k)$ is the number of partitions with $(2^d-1)k+1$ elements. A very crude upper-bound on $\#(k)$ is $\binom{2^{dJ}}{k}$. This is obtained noticing that an RDP with $(2^d-1)k+1$ elements is constructed by doing k splits of the trivial RDP (as in the formal rules for the construction of RDPs).

To bound $\Pr(\widehat{\pi}=\pi)$ recall Lemma 5. Let π be an arbitrary RDP (with maximum depth J) such that $|\pi| = (2^d-1)k+1$. There is another partition π' that can be constructed from π by aggregation, adapted to the boundary and such that

$$|\pi'| \leq \min\left(|\pi|, (2^d-1)C2^{(d-1)J}+1\right) \ ,$$

where $C>0$ comes from Lemma 5. If $k \leq C2^{(d-1)J}$ we upper bound $\Pr(\widehat{\pi}=\pi)$ trivially by one. If $k > C2^{(d-1)J}$ notice that π and π' are nested and $\pi \preceq \pi'$. To bound $\Pr(\widehat{\pi}=\pi)$ we will bound the probability that the estimation strategy chooses π against π'. For a fixed partition the choice of model corresponds simply to a projection onto a linear space (recall (3.17)). We choose π against π' if the difference between the squared errors of the model fits for π and π' is greater then the difference of the

respective penalty terms (recall (3.18), (3.19), and (3.20)). Noting again that $\pi \preceq \pi'$ (that is π is nested inside π') the difference between the squared errors is a χ^2 random variable, with $|\pi| - |\pi'|$ degrees of freedom, and so

$$\Pr(\widehat{\pi} = \pi) \leq$$
$$\Pr\left(U_{(2^d-1)(k-C2^{(d-1)J})} > \lambda \left(\frac{2^d \log 2}{2^d - 1} + \log(2n' + 1)\right)(2^d - 1)(k - C2^{(d-1)J})\right) ,$$

where $U_{(2^d-1)(k-C2^{(d-1)J})}$ is a χ^2 random variable with $(2^d - 1)(k - C2^{(d-1)J})$ degrees of freedom. Noting that $\left(\frac{2^d \log 2}{2^d-1} + \log(2n' + 1)\right) \geq \log n'$ we can write the following simpler bound

$$\Pr(\widehat{\pi} = \pi) \leq \Pr\left(U_{(2^d-1)(k-C2^{(d-1)J})} > \lambda \log n'(2^d - 1)(k - C2^{(d-1)J})\right) .$$

In [51] Laurent and Massart state the following lemma (Lemma 1): If U_q is χ^2 distributed with q degrees of freedom then, for $s > 0$

$$\Pr(U_q \geq q + s\sqrt{2q} + s^2) \leq e^{-s^2/2} .$$

Take $q = (2^d - 1)(k - C2^{(d-1)J})$ and $q + s\sqrt{2q} + s^2 = \lambda \log n'\ q$, and assume n is large

enough so that $\lambda \log n' \geq 1/2$. After some manipulation we conclude that

$$\begin{aligned}\Pr(\widehat{\pi} = \pi) &\leq \exp\left(-\frac{q}{2}\left(\lambda \log n' - \sqrt{2\lambda \log n' - 1}\right)\right) \\ &= \exp\left(-(2^d - 1)(k - C2^{(d-1)J})\frac{1}{2}\left(\lambda \log n' - \sqrt{2\lambda \log n' - 1}\right)\right) .\end{aligned}$$

We can now ask for a bound on the probability that the number of elements of $\widehat{\pi}$ exceeds some value. In particular we are going to bound the probability that the chosen partition has approximately twice more leafs than the optimal partition, adapted clairvoyantly to the boundary set. Concretely, we are going to bound

$$\zeta \stackrel{\Delta}{=} \Pr(|\widehat{\pi}| \geq (2^d - 1)2C2^{(d-1)J} + 1) .$$

Using (3.40) we have

$$\begin{aligned}\zeta \leq \sum_{k=2C2^{(d-1)J}}^{\infty} \Bigg\{ &\binom{2^{dJ}}{k} \\ &\exp\left(-(2^d - 1)(k - C2^{(d-1)J})\frac{1}{2}\left(\lambda \log n' - \sqrt{2\lambda \log n' - 1}\right)\right)\Bigg\} .\end{aligned}$$

Let $N \stackrel{\Delta}{=} 2^{dJ}$ and recall the definition of J in (3.21). It is clear that $\lambda \log n' \geq c_0 \log N$, for a suitable constant $c_0 > 0$. Then

$$\begin{aligned}\zeta \leq \sum_{k=2CN^{\frac{d-1}{d}}}^{\infty} \Bigg\{ &\binom{N}{k} \\ &\exp\left(-(2^d - 1)(k - CN^{\frac{d-1}{d}})\frac{1}{2}\left(c_0 \log N - \sqrt{2c_0 \log N - 1}\right)\right)\Bigg\} .\end{aligned}$$

For N large the $\log N$ term dominates the $\sqrt{\log N}$ term, and so, for $\epsilon > 0$ and N sufficiently large we have

$$\begin{aligned}\zeta &\leq \sum_{k=2CN^{\frac{d-1}{d}}}^{\infty} \binom{N}{k} \exp\left(-(2^d-1)(k-CN^{\frac{d-1}{d}})\frac{c_0}{2}\log N(1-\epsilon)\right) \\ &\leq \sum_{k=2CN^{\frac{d-1}{d}}}^{\infty} \frac{N^k}{k!} N^{-(2^d-1)(k-CN^{\frac{d-1}{d}})\frac{c_0}{2}(1-\epsilon)} .\end{aligned}$$

Now we use the fact that $k!$ grows much faster than an exponential, namely, for N sufficiently large we have $k! > N^{\alpha k}$ for some $\alpha > 0$. Take α such that $1 - \alpha - (2^d - 1)\frac{c_0}{2}(1-\epsilon) < 0$. Then, for N sufficiently large

$$\begin{aligned}\zeta &\leq \sum_{k=2CN^{\frac{d-1}{d}}}^{\infty} N^k N^{-\alpha k} N^{-(2^d-1)(k-CN^{\frac{d-1}{d}})\frac{c_0}{2}(1-\epsilon)} \\ &\leq \sum_{k=2CN^{\frac{d-1}{d}}}^{\infty} N^{(1-\alpha-(2^d-1)c_0(1-\epsilon))k} N^{(2^d-1)CN^{\frac{d-1}{d}}\frac{c_0}{2}(1-\epsilon)} \\ &= \frac{N^{(1-\alpha-(2^d-1)c_0(1-\epsilon))2CN^{\frac{d-1}{d}}}}{1-N^{-1}} N^{(2^d-1)CN^{\frac{d-1}{d}}\frac{c_0}{2}(1-\epsilon)} \\ &= \frac{N}{N-1} N^{(1-\alpha-\frac{1}{2}(2^d-1)\frac{c_0}{2}(1-\epsilon))2CN^{\frac{d-1}{d}}} \leq N^{-\gamma} ,\end{aligned}$$

where γ is arbitrarily large, provided α is chosen appropriately, and so $\zeta < 1/n$ for large enough n. □

3.6.7 Sketch Proof of Theorem 12

The main idea of the proof is to construct the multi-step approach iteratively, composing a single step with a multiple step estimator. Let α_K denote the rate

exponent of a K-step algorithm, that is

$$\mathbb{E}[\|\widehat{f}_n^{(K)} - f\|^2] \leq O\left(\left(\frac{n}{\log n}\right)^{\alpha_K}\right) .$$

We construct a $(K+1)$-step algorithm by using a first step in all identical with the one in Theorem 11, but restricting the maximum RDP depth to J_K. The second "step" of this algorithm is going to be the K-step methodology already developed. Therefore we are going to obtain an equation similar to (3.39), but now we can take into account the performance of the K-stage method.

$$\mathbb{E}\left[\|\widehat{f}_n^{(K+1)} - f\|^2\right] \leq O\left(\left(\frac{L}{\log L}\right)^{\alpha_K} 2^{-dJ_K}|\widehat{\mathcal{R}}| + \frac{2^{(d-1)J_K}\log n'}{n'}\right) , \qquad (3.41)$$

where $L \sim n/|\widehat{\mathcal{R}}|$ and $O(|\widehat{\mathcal{R}}|) = 2^{(d-1)J_K}$. Balancing the terms in the above expression yields

$$J_K = \left\lceil \frac{1+\alpha_K}{(d-1)(1+\alpha_K)+1} \log(n/\log n) \right\rceil .$$

Plugging this back into equation (3.41) yields the recursion

$$\alpha_{K+1} = -\frac{1}{(d-1)(1+\alpha_K)+1} ,$$

where $\alpha_1 = -1/d$. The above recursion has a closed-form solution given by

$$\alpha_K = \begin{cases} -\frac{1}{d-1+1/K} & \text{, if } d = 2 \\ -\frac{1}{d-1+\frac{d-2}{(d-1)^K-1}} & \text{, if } d > 2 \end{cases},$$

concluding the proof. □

Chapter 4

Coarse-to-Fine Learning of Piecewise Constant Functions

In this chapter we address the problem of estimation of piecewise constant functions with smooth boundaries. Although we work in a passive learning setting we use essentially the same ideas as in Chapter 3 to construct a greedy algorithm for sequential, coarse-to-fine estimation, with significant computational savings over traditional methods. Accurate detection and localization of the boundary is the key aspect of these kinds of problems. In general, algorithms capable of achieving optimal performance require exhaustive searches over large dictionaries that grow exponentially with the dimension of the observation domain. The computational burden of the search hinders the use of such techniques in practice and is the main motivation for the development of faster methodologies. We consider a sequential, coarse-to-fine approach that involves first examining the data on a coarse grid, and then refining the analysis and approximation in regions of interest. The estimators involve an almost linear-time (in two dimensions) sequential search over the dictionary, and converge at the same near-optimal rate as estimators based on exhaustive searches. Specifically,

for two dimensions, the proposed algorithm requires $O(n^{7/6})$ operations for an n-pixel image, much less than the traditional wedgelet approaches, which require $O(n^{11/6})$ operations.

4.1 Introduction

The dimensionality of signals is often lower than the ambient observation space. For example, a pure sinusoidal process occupies a one-dimensional linear subspace. Linear subspace models and subspace identification techniques have played a major role in modern signal processing. However, in many cases the signal space may be a subset of the observation space that is not a linear subspace, and most likely can be modeled as a lower-dimensional manifold embedded in the observation space. A simple example of this phenomenon occurs in the analysis of images. For example, consider a binary image composed of a "white" region and a "black" region separated by a smooth boundary. This image is simply a one-dimensional curve (the boundary) embedded in the two-dimensional image space. Note that the collection of such images is not a linear subspace of the observation space (*e.g.*, the average of two such images is not a binary image). Estimating or coding these images involves identifying or *learning* the boundary. Several investigators have proposed new basis functions or dictionaries for describing $(d-1)$-dimensional manifolds embedded in d-dimensional spaces (for d=2,3), including wedgelet/beamlet dictionaries and curvelet frames [35, 52, 53]. While promising, the dictionaries are overcomplete and

learning procedures based on those can be quite computationally demanding to implement. This computational hurdle motivates the methodologies presented here. We consider sequential, coarse-to-fine learning strategies. The basic idea is to first examine the data on a coarse grid, and then refine the analysis and approximation in regions predicted near the boundary. By carefully examining the approximation and estimation error tradeoffs in each step, we show that piecewise constant images with smooth boundaries can be optimally recovered through a sequential process in almost linear-time (in two dimensions), yielding significant computational savings.

Note that the main idea exploited here is very similar to the one of the previous chapter: use a stepwise approach, where in the first step we determine a perceived location of the boundary, and in the second step we refine our estimate near that perceived location. The main difference is that we do not collect any extra samples in that second step; instead we are choosing where to perform more exhaustive computation (more beneficial near the boundary). The analysis of proposed methods is quite similar to the one presented in detail in Chapter 3, therefore here we present solely "light-weight" proofs and proof sketches. It is important to note that the observation model and error metrics are significantly different than the ones in the previous chapter.

4.2 Problem Formulation

Consider a function f defined on the d-dimensional hypercube $[0,1]^d \subseteq \mathbb{R}^d$ (assume $d \geq 2$). The function consists of constant regions separated by $(d-1)$-dimensional boundaries, that is, these are functions satisfying Definition 4 on page 76. We further assume these boundaries are locally Hölder smooth with parameter 2 (for example, twice continuously differentiable curves). In other words these boundaries are locally well-approximated by a $d-1$ dimensional hyperplane. In two dimensions this corresponds to the class of functions studied in [35], and an example is shown in Figure 4.1(a). We do not observe the function f directly, but only samples which have been corrupted by noise. Consider a partition of the unit hypercube into n sub-hypercubes of sidelength $n^{-1/d}$ (assume without loss of generality that n is a power of d). Denote each hypercube by $V(i)$, $i \in \{1, \ldots, n\}$. In Figure 4.1(a) this procedure is shown for $d = 2$. Each one of these "small" sub-hypercubes corresponds to a voxel, and these determine the finest resolution we consider. This initial partition can be generated by a recursive dyadic partition (recall the acronym RDP). First divide the domain into 2^d sub-hypercubes of equal size. Repeat this process again on each sub-hypercube. Proceeding in this fashion $1/d \log_2 n$ times yields the initial partition. This gives rise to a complete RDP with $1/n$ leafs (the original domain is divided into n cells). The RDP process can be represented with a rooted tree structure: the root node corresponds to the entire domain (*i.e.*, the unit hypercube), their children nodes correspond to the 2^d sub-hypercubes, and so on.

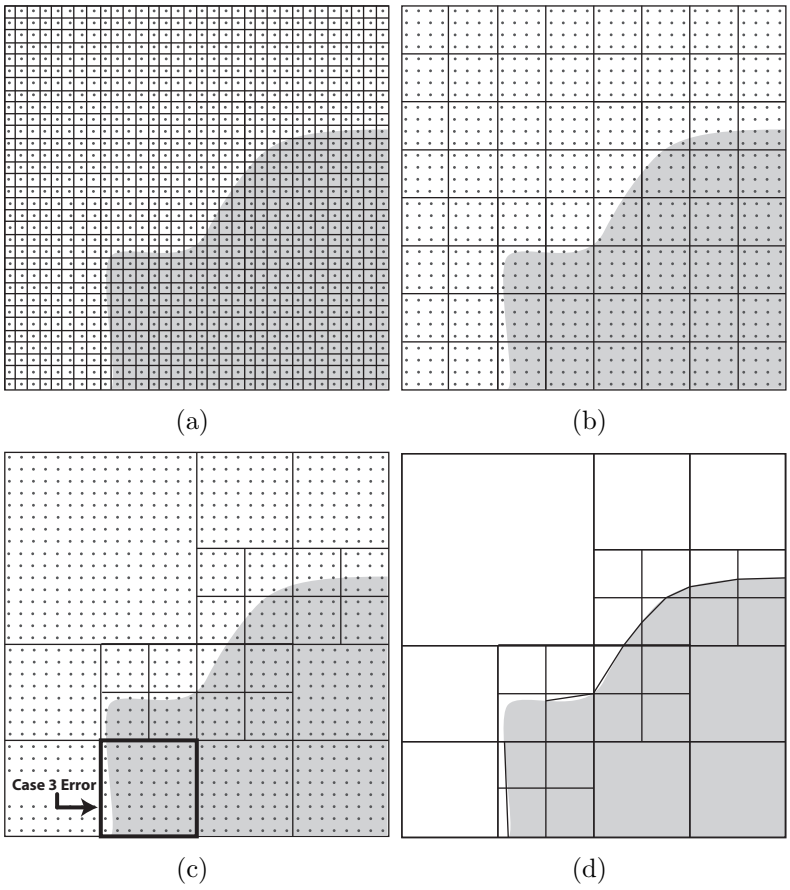


Figure 4.1: Example boundary estimation problem. (a) Initial RDP used in traditional piecewise linear methods. (b) Initial coarse-resolution RDP used in the preview step. (c) Partition generated in a preview step. Note the example of a Case 3 error. (d) Final partition generated during the refinement step, using shifted partitions.

For each voxel we associate the value $\theta(i)$,

$$\theta(i) = \frac{1}{\mathrm{Vol}(V(i))} \int_{V(i)} f \ ,$$

the average of f over the voxel $V(i)$, where $i \in \{1, \ldots, n\}$ and $\mathrm{Vol}(V(i))$ denotes the volume of $V(i)$. We do not observe the value of each voxel directly, but instead only a noisy corrupted version. Our measurements, $x(i)$, are samples of the field

$\theta(i)$ corrupted by additive white Gaussian noise with variance σ^2, that is, $x(i) \sim \mathcal{N}(\theta(i), \sigma^2)$, where we assume that all measurements are statistically independent. Given these measurements we want to estimate the voxel values $\theta(i)$ accurately.

Let $\Theta = \{\theta(i)\}_i$ and $\boldsymbol{x} = \{x(i)\}_i$. Let $\widehat{\theta}_{\boldsymbol{x}}(i)$ be the estimate for the value of voxel i (in what follows we drop the dependence on $\boldsymbol{x}$ for the ease of notation). Define $\widehat{\Theta} = \{\widehat{\theta}(i)\}_i$. The measure of performance considered is the Mean Square Error (MSE), defined as

$$\mathrm{MSE}(\widehat{\Theta}, \Theta) \triangleq \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Big(\theta(i) - \widehat{\theta}(i)\Big)^2\right] . \tag{4.1}$$

4.3 Maximum Penalized Likelihood Estimators

Under the assumption that f is a piecewise constant function it is known that, for any reasonable estimator $\widehat{\Theta}$, as the number of voxels n increases the MSE (4.1) decreases. For piecewise constant functions on $[0,1]^d$ separated by $(d-1)$-dimensional Hölder-2 boundaries, the MSE decays no faster than $O(n^{-2/(d+1)})$, the minimax lower bound [34]. This lower bound is shown to be optimal for a class of boundary fragments, as the ones considered in Chapter 2.

It turns out that it is possible to nearly achieve the optimal rates above using maximum penalized likelihood techniques. This is accomplished using the same techniques as described in Chapter 3. Let Π_n be the class of all possible RDPs (with leafs at depths no larger than $1/d \log_2 n$). Recall that for each $\pi \in \Pi_n$, there is an associ-

ated tree structure (generally of non-uniform depth corresponding to the non-uniform resolution of most RDPs). The leafs of each tree represent dyadic (side-length equal to a power of 2) hypercube regions of the associated partition. Consider a certain RDP π, and define the estimator of the field on each leaf of the partition to be the least-squares fit of a wedgelet to the measurements in the corresponding hypercube. In d dimensions, a wedgelet fit consists of a $(d-1)$-dimensional hyperplane separating a hypercube into two regions and a constant fit to the data in each region. Let $\widehat{\Theta}^{(\pi)}$ denote a model of the field (based on the least-squares model fits on each cell of π). The empirical measure of performance is the sum-of-squared errors between $\widehat{\Theta}^{(\pi)}$ and the data:

$$R\left(\widehat{\Theta}^{(\pi)}, \boldsymbol{x}\right) = \sum \left(\widehat{\theta}^{(\pi)}(i) - x(i)\right)^2 \ .$$

For fixed partition π, the choice of $\widehat{\Theta}^{(\pi)}$ that minimizes $R(\widehat{\Theta}^{(\pi)}, \boldsymbol{x})$ is simply given by the least-squares fits on each square, as discussed above. Now define the complexity penalized estimator as

$$\widehat{\Theta} = \underset{\tilde{\Theta}_\pi : \pi \in \Pi_n}{\arg\min}\, R\left(\tilde{\Theta}_\pi, \boldsymbol{x}\right) + 2\sigma^2 \lambda |\pi| \log n \ , \tag{4.2}$$

where $|\pi|$ denotes the number of hypercubes in the partition π and λ is constant which will be defined later. This optimization can be solved using a bottom-up pruning algorithm [47, 52]. It has the further advantage that upper bounds on the estimation error can be established using several recent information-theoretic results,

most notably the Li-Barron bound [54] and its generalizations [55]. These are oracle bounds, similar to the ones used in the previews chapter, although more adequate to the deterministic sampling framework used here. Specifically, if λ is chosen so that the dictionary of estimators satisfies the Kraft inequality,

$$\sum_{\tilde{\Theta}_\pi : \pi \in \Pi_n} \exp(-\lambda |\pi| \log n) \leq 1 \ ,$$

then

$$\mathrm{MSE}(\widehat{\Theta}, \Theta) \leq \min_{\tilde{\Theta}_\pi : \pi \in \Pi_n} \frac{2}{n} R\left(\tilde{\Theta}, \Theta\right) + \frac{8\sigma^2 \lambda |\pi| \log n}{n} \ . \tag{4.3}$$

For the remainder of the chapter assume that f has Hölder-2 $(d-1)$-dimensional smooth boundaries. For this class of functions it is known that the minimax MSE rate is bounded below by $O(n^{-2/(d+1)})$. It can be shown that solving the optimization in (4.2) yields a partition which balances the approximation error and estimation error terms in the bound on the MSE in (4.3), resulting in a final bound of $O\left((\log n/n)^{-2/(d+1)}\right)$.

The main challenge associated with such methods involves the optimization in (4.2). In general, the algorithms needed to achieve the performance rates described above must exhaustively examine all models in each of the candidate partitions $\pi \in \Pi_n$. For example, in the two-dimensional wedgelet case, the number of wedgelet models required depends on the δ-resolution of the wedgelet analysis, which determines the approximation accuracy [35]. Specifically, for a hypercube of sidelength ℓ

the number of wedgelet models that must be evaluated is $O(\ell/\delta)^2$ (and each evaluation requires $O(n\ell^2)$ operations). Wedgelet analysis nearly achieves the minimax performance rate for the class of images under consideration when $\delta \sim n^{-2/3}$. Since wedgelet approximations must be calculated for each candidate partition $\pi \in \Pi_n$, a total of $O(n^{4/3})$ wedgelet fits must be computed [35], resulting in an overall computational complexity of $O(n^{7/3})$. This can be improved slightly by noting that the evaluation of the wedgelet models can be done in a incremental fashion, where each wedgelet model fit is evaluated using previously evaluated models; this means that for a given hypercube of sidelength ℓ, after the first wedgelet fit is calculated, each successive wedgelet fit can be calculated in $O(\sqrt{n}\ell)$ operations as opposed to $O(n\ell^2)$ operations. Taking this into account yields an overall computational complexity of $O(n^{11/6})$, which is prohibitive in practice.

4.4 Coarse-to-Fine Estimation

The heavy computational complexity of the above techniques motivates the approach presented here. In the constant regions of function f, the piecewise constant approximation estimates perform well. The task that limits the rates of convergence of the MSE is the estimation of f in the vicinity of the boundaries. This a similar behavior we observed in Chapter 3, and so we can proceed in an analogous fashion, where the main idea is to perform the estimation task in a sequential, coarse-to-fine fashion: In the first step (termed the *preview* step), a coarse estimation of the function

is performed, using only piecewise constant approximations, as an attempt to identify the approximate location of the boundary. In a second step (termed the *refinement* step) we perform a wedgelet boundary fit in areas that were identified as possible boundary regions. If the preview step is effective then we will perform wedgelet fits (which are computationally demanding) only in the regions were they are needed, instead of applying and testing them throughout the entire domain.

In the remainder of this section, we show that this method nearly achieves the minimax performance rate of $O(n^{-2/(d+1)})$ and requires significantly fewer computational resources than wedgelet methods based on exhaustive dictionary searches. In the following we are going to omit the logarithmic factors to make the presentation lighter.

4.4.1 Error Analysis

In the first step, the preview step, we start with an uniform RDP with n^γ voxels, $\gamma < 1$, as shown in Figure 4.1(b). We generate an estimator by pruning this RDP, as shown in Figure 4.1(c), and decorating each leaf with a constant model. Let $V_c(i)$ denote the voxels corresponding to this uniform RDP with n^γ leafs (we will refer to this as the *coarse resolution*, as opposed to the *fine resolution* uniform RDP, which has a total of n voxels). Note that each coarse resolution voxel contains $n^{1-\gamma}$ measurements. For each of these coarse resolution voxels let $x_c(i)$ be the average of

the measurements falling into $V_c(i)$, therefore

$$x_c(i) \sim \mathcal{N}(\theta_c(i), n^{-(1-\gamma)}\sigma^2) \ ,$$

where $\{x_c(i)\}_i$ are statistically independent and

$$\theta_c(i) = \frac{1}{n^{1-\gamma}} \sum_{j:V(j)\subseteq V_c(i)} \theta(i) \ .$$

We can evaluate the mean squared error at the coarse resolution, and it is given by

$$\begin{aligned} \mathrm{MSE}_c &\triangleq \mathbb{E}\left[\frac{1}{n^\gamma}\sum_{i=1}^{n}(\theta_c(i) - \widehat{\theta}_c(i))^2\right] && (4.4) \\ &= O(n^{-(1-\gamma)}(n^\gamma)^{-1/d}) \\ &= O(n^{-1+\gamma\frac{d-1}{d}}) \ . && (4.5) \end{aligned}$$

This can be derived by noting that the variance of each $x_c(i)$ is $n^{-(1-\gamma)}\sigma^2$ and the MSE of the piecewise constant estimator at the coarse resolution decays like $O((n^\gamma)^{-1/d})$, for unit variance noise.

Denote the pruned RDP at coarse resolution by RDP_c. In the refinement step we consider a piecewise linear fit on the leafs of RDP_c that were not pruned (*i.e.*, the leafs of RDP_c that are at the deepest level), keeping all the other leafs unaltered, as shown in Figure 4.1(d). The main reasoning is that with very high probability (this is made precise below) most voxels at the coarse resolution that do not intersect

the boundary are going to be pruned, so we can use the unpruned voxels as a good indication for the presence of a boundary.

In the following we evaluate the asymptotic behavior of the MSE at the fine resolution for the two step procedure. Our analysis makes repeated use of the fact that the number of coarse voxels intersecting the boundary is $O(n^{\gamma\frac{d-1}{d}})$ (this follows from the assumption that the boundary as box-counting dimension $d-1$). For each leaf in the pruned RDP_c we consider three situations, and analyze the impact on the overall MSE:

Case 1: *Leafs of RDP_c that do not intersect the boundary:* In this case, averaging the observations in contiguous regions is optimal and so the MSE at the fine resolution behaves like (4.5). Therefore these leafs contribute $O(n^{-1+\gamma\frac{d-1}{d}})$ to the fine resolution MSE. This dictates our choice of γ; by choosing $\gamma = \frac{d}{d+1}$ we obtain the desired MSE rate of $O(n^{-2/(d+1)})$.

Case 2: *Leafs of RDP_c that were not pruned:* For the regions corresponding the these leafs we perform a wedgelet fit (the refinement step) and the MSE decays exactly as if we were doing wedgelet fits everywhere. Therefore these leafs contribute $O(n^{-2/(d+1)})$ to the fine resolution MSE.

Case 3: *Leafs of RDP_c that were pruned, but intersect the boundary:* This scenario corresponds to a case were a voxel intersecting the boundary was somehow "erroneously" pruned to a larger leaf. Therefore this leaf is approximated with a constant, but contains a fragment of the boundary. For a large enough resolution n the

function f in the hypercube corresponding to this leaf is composed of two constant regions. Because the voxel containing the boundary was pruned, we know that the volume of one of the two constant regions is small with respect to the total volume of the leaf hypercube, and behaves like $O(n^{-1/2+\gamma/2})$ (this follows from the fact that the squared bias at the coarse resolution in bounded by (4.5)). This yields an average bias squared at the fine resolution of order $O(n^{-1/2+\gamma(1/2-1/d)})$. Setting $\gamma = \frac{d}{d+1}$ (which is necessary to bound the Case 1 error) does not result in the desired fine resolution MSE rate of $O(n^{-2/(d+1)})$.

We propose a technique that overcomes this difficulty while incurring only minimal extra computational cost. Recall that, for high enough resolution, the function f in the hypercube corresponding to a pruned leaf is composed of two constant regions. Case 3 errors occur when the boundary of the two regions is closely aligned with one of the dyadic splits of the RDP. An example Case 3 error is depicted in Figure 4.1(c). The basic idea is then to perform various preview stages, just like in the active learning algorithm of Chapter 3. We compute $d+1$ preview estimators, one as described above, and the remaining d using shifted partitions (by one coarse voxel) in each of the d coordinates. This ensures that the boundary is "felt" in one of the preview steps, and therefore detected with high probability. It is highly probable that a voxel erroneously pruned in one of the preview stages is not pruned in another preview stage. In the refinement step we perform a wedgelet fit to any coarse resolution voxel that was left unpruned in at least one of the preview estimators.

4.4.2 Computational Complexity

We will describe here the computational complexity associated with the two-dimensional case, which is also studied in the simulation section; the extension to d dimensions is straightforward. First, recall that wedgelets are fit in all leafs of RDP_c which were not pruned, and since $\gamma = 2/3$, each of these hypercubes has side-length $\ell = n^{-1/3}$ and contains $n^{1/3}$ pixels. Because $\delta \sim n^{-2/3}$, $(\ell/\delta)^2 \sim n^{2/3}$ wedgelet fits are evaluated at each of these leafs. Since there are $O(n^{1/3})$ such leafs, a total of $O(n)$ wedgelet fits must be calculated, resulting in an overall computational complexity of $O(n^{4/3})$. Note that the complexity can be reduced to $O(n^{7/6})$ operations by calculating wedgelet fits incrementally, as described above. Thus this coarse-to-fine approach is significantly faster that the traditional wedgelet analysis method.

Remarks: It is important to mention that the above analysis is solely a sketch, but all the results shown here can be made entirely formal. In particular using the a concept similar to Definition 5 on page 100 one can formally justify that the multiple preview estimators suffice to control the Case 3 error. Also one can show that only regions near the boundary are going to be refined, that is, we are going to perform only $O(n^{1/3})$ wedgelet fits. This is shown using essentially Lemma 7 on page 137. Finally we note that slightly better performance can be attained by modifying the algorithm: Run the preview estimators using RDPs supported on n voxel (instead of n^γ voxels), and refine regions corresponding to leafs of the preview RDPs that are at depths not shallower than $\gamma/d \log_2 n$. This will guarantee that the boundary can

be detected more accurately, without increasing significantly the number of regions falsely detected as boundary.

4.5 Simulations

We demonstrate the effectiveness of the proposed method using the Shepp-Logan phantom brain image, commonly used in medical imaging simulations. The n noisy measurements, arranged on a 256×256 grid, are displayed in Figure 4.2(a); in this example, $\sigma^2 = 0.001$ and the mean pixel value is 0.15. For this simulation, the penalization weights are chosen according to the theory in 4.3. Under this scenario, $n^\gamma = 256^{4/3}$, and so the preview step is initialized with an RDP of 4096 4×4 coarse resolution squares. The preview partition in Figure 4.2(b) demonstrates how the initial Haar estimate does not prune the initial coarse resolution squares in regions near the boundaries; after the preview step pruning, the initial coarse resolution RDP of 4096 squares has been pruned back to a nonuniform RDP with only 1199 4×4 squares remaining after using the procedure described in Section 4.4.

The final estimate after the refinement step is displayed in Figure 4.2(c); recall that this requires $O(n^{4/3})$ operations to compute. This estimate can be compared to a standard wedgelet decomposition, as seen in Figure 4.2(d). This requires $O(n^{7/3})$ operations; *i.e.*, a factor of $O(n)$ more operations than the proposed method. These estimates were calculated on a 667 MHz PowerPC G4 with 768 MB of memory running Mac OS 10.2.8; on this machine, the standard wedgelet estimate was computed in

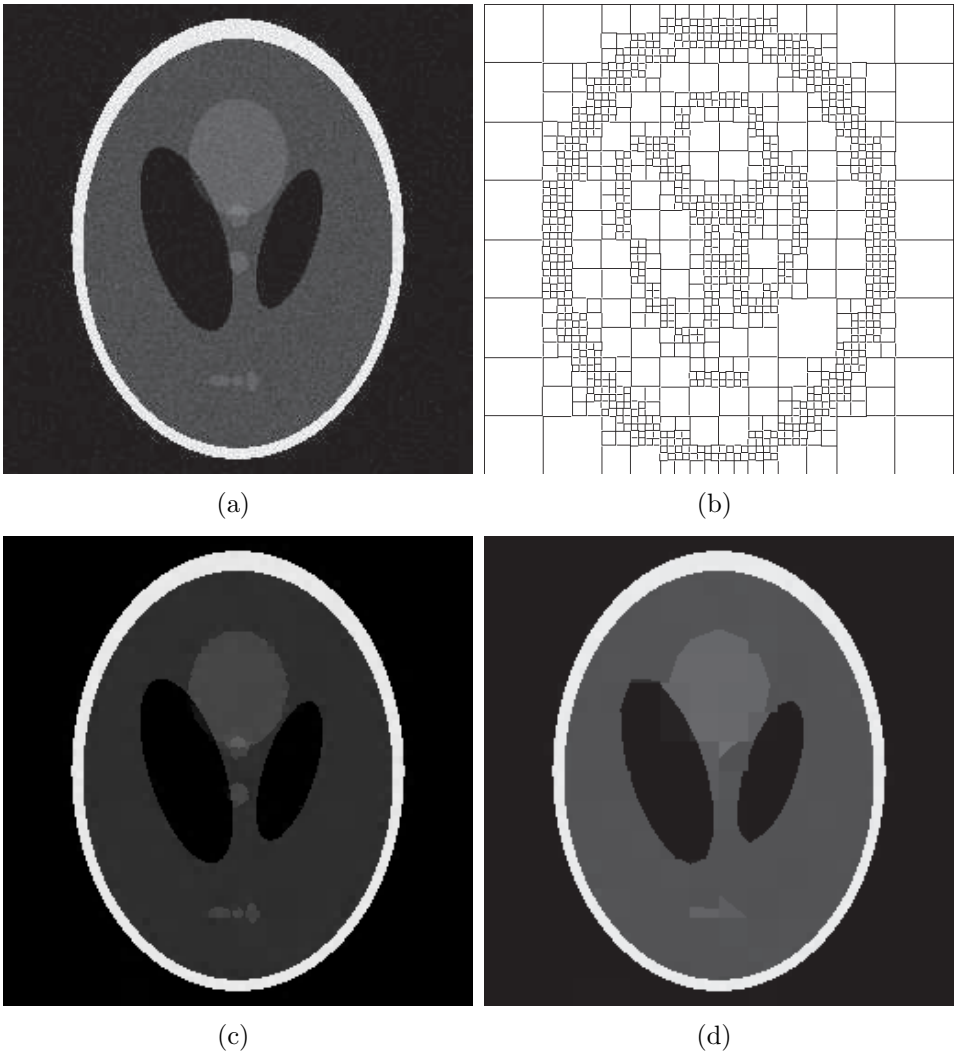


Figure 4.2: Shepp-Logan phantom. (a) 256×256 noisy measurements, $\sigma^2 = 0.001$, MSE = 0.0115. (b) preview partition. (c) Shepp-Logan phantom estimate formed by fitting one wedgelet or constant to each of the unpruned squares from the preview step; MSE = 0.000504. (d) Shepp-Logan phantom estimate using standard wedgelet, MSE = 0.00163.

665 seconds and the coarse-to-fine estimate was computed in 37.4 seconds. This is an excellent example of how the proposed method performs as well as a standard wedgelet estimate in terms of both MSE and visual quality with significant computational savings.

4.6 Final Remarks

This chapter showed that ideas similar to the ones of active learning can be used to construct sound greedy algorithms with provable performance. We study in particular the estimation of piecewise constant functions, where the different constant regions are separated by Hölder-2 smooth boundaries. Techniques for estimation in this class were previously developed by Donoho [35]. In this work we build on the class of models described in [35], but instead of performing a computationally demanding search over a large dictionary of models, we proceed in a sequential fashion, using a two step process, where in the first step we select a subset of image models, and in the second step we make a final model selection. While this is a greedy procedure, it has desirable features, such as low computational cost, and provable asymptotical optimality. Furthermore these gains are evident in simulation studies, where the proposed algorithm leads to a significant reduction in computation time.

The ideas developed in this chapter can possibly be generalized to different contexts, where choosing *when* and *where* to compute can lead to significant savings. For example, in problems of fluid dynamics it would be extremely relevant to unevenly allocate computational resources depending on the region (a region where cavitation occurs requires significantly more computational effort to ensure accurate results). Another interesting, albeit very hard, question pertains to lower bound computational complexity: given a statistical problem for which there is an optimal solution (*e.g.*, an algorithm that achieves the optimal performance rate) can one characterize

the computational complexity of any algorithm achieving that optimal performance, in particular characterize the computational complexity of the "fastest" algorithm? These questions are beyond the scope of this thesis, but nevertheless interesting to consider in the future.

Chapter 5

Compressive versus Active Sampling

Compressive Sampling (CS), or *Compressed Sensing*, has generated a tremendous amount of excitement in the signal processing community. Compressive sampling, which involves non-traditional samples in the form of randomized projections, can capture most of the salient information in a signal with a relatively small number of samples, often far fewer samples than required using traditional sampling schemes. *Active sampling* (AS), also referred to as adaptive sampling, uses information gleaned from previous observations (*e.g.*, feedback) to focus the sampling process. In this chapter we compare the theoretical performance of compressive and active sampling in noisy conditions, and we show that for certain classes of piecewise constant signals and high SNR regimes both CS and AS are near-optimal. This result is remarkable since it shows that compressive sampling, which is non-adaptive/passive sampling, cannot be significantly outperformed by other methods (including active sampling procedures), even in the presence of modest noise.

5.1 Introduction

Compressive Sampling (CS), also called *Compressed Sensing*, has generated a tremendous amount of excitement in the signal processing community. CS involves taking non-traditional samples in the form of randomized projections, such as random binary, Gaussian, or Fourier projection vectors. Specifically, the samples of a signal vector $\boldsymbol{f} \in \mathbb{R}^n$ are inner products of the form

$$y_j = \boldsymbol{\phi}^T(j)\boldsymbol{f}, \quad j = 1, \ldots, k \ ,$$

where $\{\boldsymbol{\phi}(j)\}$ are random vectors (*e.g.*, normalized n-vectors comprised of i.i.d. binary or Gaussian random variables). Recent theoretical results indicate that extremely accurate signal reconstructions are possible from a relatively small number of noiseless random projections [56, 57] when the signal f has some structure, for example f has a sparse representation in some basis. These results have been extended to show that many signals can be very accurately recovered from random projections contaminated with noise [50], in many cases much more accurately than possible using conventional sampling methods. More recently, similar results were confirmed using alternative analysis techniques [58]. Despite these encouraging results, there is a significant gap between the performance bounds for the noiseless and noisy scenarios. This yields pessimistic bounds in regimes where the SNR is high.

As described in Chapters 2 and 3 *Active Sampling* (AS), also known as adaptive

sampling, involves sequential sampling schemes that use information gleaned from previous observations to guide the sampling process. It was shown in those chapters and references therein that adaptively selecting samples in order to learn a target function can outperform conventional sampling schemes. In particular, it was shown that active sampling can be used to recover certain classes of one-dimensional piecewise constant functions in noise with an error that decays exponentially fast in the number of samples taken (see also [23]). This is significantly faster than conventional (uniform) sampling schemes whose errors converge at a much slower polynomial rate, with or without noise present. Similarly encouraging results have been obtained for the recovery of multidimensional piecewise constant functions, in which case AS achieves the optimal minimax-rate among all possible sampling schemes.

The optimality of active sampling for recovering piecewise constant functions from noisy samples suggests an intriguing question. Can non-adaptive/passive CS perform comparatively as well as AS in such situations? This chapter provides an affirmative answer to this question. This is remarkable since it provides evidence that compressive sampling, which is non-adaptive, cannot be significantly outperformed by other methods (including every possible adaptive point sampling procedure) for high SNR regimes. The results hold only for certain classes of piecewise constant functions, but this is a quite rich family of signals that has many interesting potential applications, particularly in image processing. These results provide some understanding about the gap between existing error bounds for CS in the noiseless [56, 57] and noisy sce-

narios [50, 58]. Our results may also serve as a starting point for investigations of the optimality of CS in more general signal spaces.

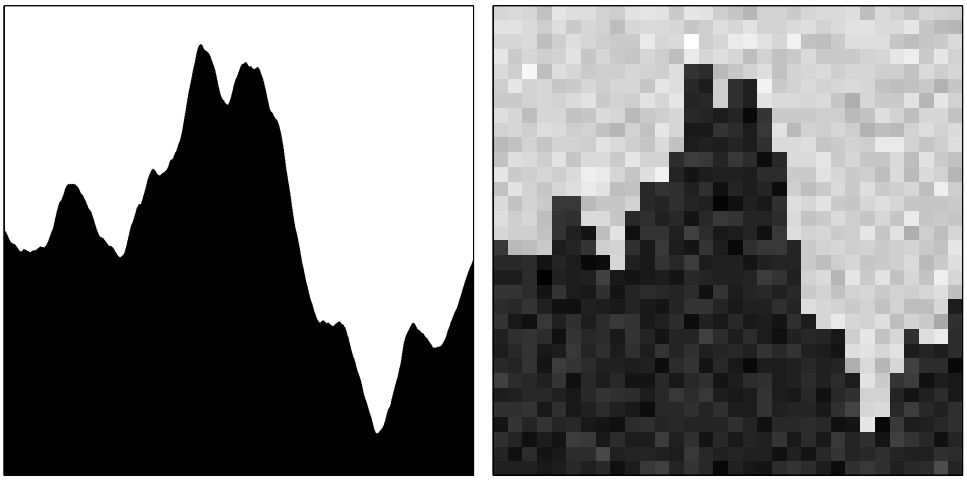
5.2 Compressive and Active Sampling

We focus our attention on classes of piecewise constant functions in one or more dimensions. For illustration, consider the piecewise constant image depicted in Figure 5.1 on the next page. This image belongs to the so-called "boundary fragment" class [34], also called the "horizon" image class [35]. It consists of two constant regions (valued $+1$ and -1 for our purposes) separated by a one-dimensional curve with *functional* form $y = g(x)$; *i.e.*, the vertical coordinate of the boundary, y, is determined by a smooth function, g, of the horizontal coordinate, x.

Our primary concern is how well one can recover the original image in Figure 5.1(a) from noisy samples, such as the noisy pixel samples depicted in Fig. 5.1(b). We assume that the boundary function g is Lipschitz smooth with parameter $L > 0$, that is

$$|g(x_1) - g(x_2)| \leq L|x_1 - x_2| \ ,$$

for all $x_1, x_2 \in [0, 1]$. In this case it is known that standard wavelet denoising methods can reconstruct the image from k uniformly spaced and noisy pixel samples with a mean square error of $O(k^{-1/2})$, and that no estimation procedure based on these samples can perform significantly better (that is $k^{-1/2}$ is the minimax rate) [34]. However, if one allows the possibility of taking pixel samples in an adaptive fashion, sequen-



(a) (b)

Figure 5.1: Example of an 1024×1024 pixel image from the boundary fragment class (image (a)); and a 32×32 pixel noisy image with $\sigma = 0.15$ (image (b)).

tially monitoring the sample values and carefully "focusing-in" on the boundary region which dominates the overall error, then it is possible to achieve a rate of $O(k^{-1})$, which is the best possible error rate among all adaptive and non-adaptive sampling schemes and estimation procedures [36]. In other words, active pixel sampling can produce vastly superior image reconstructions with far fewer samples than conventional uniform sampling schemes, simply because most of the samples are *wasteful*; in conventional schemes most samples are far away from the boundary.

The main result of this chapter, stated formally in the theorem below, shows that non-adaptive CS can have a performance that is similar to the one of AS. Before stating the theorem we define the following piecewise constant function classes. First consider a space of one-dimensional n-point signals $\boldsymbol{f} = (f_1, \ldots, f_n)$,

$$\mathcal{F}_1 \;\; = \;\; \{\boldsymbol{f} : f_i = -\mathbf{1}\{i \leq \theta\} + \mathbf{1}\{i > \theta\},\ \theta \in \{0, \ldots, n\}\} \;\; ,$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function. The vectors in $\mathcal{F}_1$ correspond to step functions. For two-dimensional images, consider the following class of $n \times n$ arrays $\boldsymbol{f}$

$$\mathcal{F}_2 \;\; = \;\; \{\boldsymbol{f} : f_{i,j} = 2 \cdot \mathbf{1}\{j/n \leq g(i/n)\} - 1, \; g \in \mathrm{Lip}(L)\}$$

where $\mathrm{Lip}(L)$ denotes the space of one-dimensional Lipschitz (boundary) functions on $[0,1]$ (i.e., functions satisfying $|g(x) - g(y)| \leq L|x - y|$ for all $x, y \in [0,1]$). Higher dimensional analogs of $\mathcal{F}_2$ are constructed in an analogous fashion from $d-1$ dimensional Lipschitz boundary functions (*e.g.*, see Chapter 2). We begin by stating the result for $d = 1$.

Theorem 14. *Suppose that $\boldsymbol{f} \in \mathcal{F}_1$ and assume that we take $k \leq n$ samples of the form*

$$y_j = \boldsymbol{\phi}^T(j)\boldsymbol{f} + w_j, \quad j = 1, \ldots, k \;,$$

where $\{\boldsymbol{\phi}(j)\}$ are normalized Rademacher random vectors (n-vectors with entries that are i.i.d. random variables taking values $\pm 1/\sqrt{n}$ with equal probability), and $\{w_j\}$ are i.i.d. Gaussian random variables with zero mean and variance σ^2, and independent of $\{\boldsymbol{\phi}(j)\}$. A function estimate $\widehat{\boldsymbol{f}}_k$ can be derived from $\{y_j, \boldsymbol{\phi}(j)\}$ satisfying the following mean square error bounds.

$$\mathbb{E}\left[n^{-1}\|\boldsymbol{f} - \widehat{\boldsymbol{f}}_k\|^2\right] \leq 4n[\alpha(n, \sigma^2)]^k \;,$$

where $0 < \alpha(n, \sigma^2) < 1$, *namely*

$$\alpha(n, \sigma^2) = \max\left\{ \mathrm{e}^{-\frac{1}{2n\sigma^2}}, \frac{1}{2} + \frac{1}{2}\mathrm{e}^{-\frac{2}{n\sigma^2}} \right\} .$$

Note that the error bound decays exponentially fast with the number of samples, far faster than the k^{-1} rate one usually encounters in parametric estimation (which typically considers only non-adaptive sampling). As far as the optimality of the error decay rate is concerned, first consider the one-dimensional case. We know from [23] that the exponential decay of the expected error in the number of samples k is the best one can hope for. This claim comes from results in information theory: estimation of θ, the step location, can be viewed as a communication problem where we transmit θ through an additive white Gaussian noise channel. Due to the noise in the channel the probability of error can only decay exponentially with the number of channel transmissions (equivalent to the number of samples or random projections). However, it may be possible to improve the value of α governing the exponential decay in our bound, even in the passive scenario.

For the multi-dimensional setting we do not consider "pure" projection samples, but instead Rademacher projection taken for various "columns" of the image. This methodology is still non-adaptive, and allows us to use the results obtained for the one-dimensional setting. We state the observation model for the two dimensional case, for simplicity.

Theorem 15. *Suppose that* $\boldsymbol{f} \in \mathcal{F}_2$. *Let* $\tilde{\boldsymbol{f}}_l \stackrel{\Delta}{=} \{f_{i,l},\ i \in \{1:n\}\}$ *be the* l^{th} *column of* f. *Assume that we take* $k \leq n^d$ *samples of the form*

$$y_{j,l_j} = \boldsymbol{\phi}^T(j)\tilde{\boldsymbol{f}}_{l_j} + w_j, \quad j = 1, \ldots, k \ ,$$

where $\{\boldsymbol{\phi}(j)\}$ *are Rademacher random vectors, and* $\{w_j\}$ *are i.i.d. Gaussian random variables with zero mean and variance* σ^2, *and independent of* $\{\boldsymbol{\phi}(j)\}$. *In the above* $l_j \in \{1, \ldots, n\}$ *are going to be carefully chosen to yield the result below (see proof for details). A function estimate* $\widehat{\boldsymbol{f}}_k$ *can be derived from* $\{y_j, \boldsymbol{\phi}(j), l_j\}$ *satisfying the following mean square error bound.*

$$\mathbb{E}\left[n^{-d}\|\boldsymbol{f} - \widehat{\boldsymbol{f}}_k\|^2\right] \leq C(n, L, \sigma^2)\, k^{-1/(d-1)} \ ,$$

where $0 < C(n, L, \sigma^2) \leq 6(L+1)(1+n\sigma^2)\log n$, *for* $n \geq 4$.

For the multi-dimensional setting the bound of the theorem is dramatically different than the bounds obtained using estimation-theoretic techniques, like in [50], where the mean squared error is bounded by a constant (independent of k and n) times $k^{-1/d}\log n$. The new bound is equal to a constant times $k^{-1/(d-1)}\log n$ for small values of σ^2 (i.e., $\sigma^2 \sim 1/n$). The above theorem, together with the results in [23,36], indicates that in the high SNR regime the performance of CS is comparable with the performance of the best active pixel sampling technique. Moreover, it is known that the $k^{-1/(d-1)}$ decay rate is the minimax optimal rate [20,36], implying that no other

active or passive sampling scheme and estimation procedure can significantly improve on AS or CS under these conditions. Furthermore, in the next section we describe a relatively simple Bayesian procedure for constructing $\widehat{\boldsymbol{f}}_k$ from the noisy compressive samples.

5.3 Signal Reconstruction Algorithm

First we consider the reconstruction problem for the one-dimensional class $\mathcal{F}_1$. Each element $\boldsymbol{f} \in \mathcal{F}_1$ is parameterized by $\theta \in \{0, \ldots, n\}$, that is $\boldsymbol{f} \equiv \boldsymbol{f}(\theta)$. The basic reconstruction algorithm used is the maximum likelihood estimator of θ. For analysis purposes it is convenient to formulate the algorithm in a Bayesian way: Let $\boldsymbol{p}(j) \stackrel{\Delta}{=} \{p_0(j), \ldots, p_n(j)\}$ parameterize the posterior after j measurements, that is

$$\Pr(\theta = l | y_1, \ldots, y_j, \boldsymbol{\phi}(1), \ldots \boldsymbol{\phi}(j)) \stackrel{\Delta}{=} p_l(j) \ .$$

We start with a uniform prior on θ, that is, $p_l(0) = 1/(n+1)$ for all $l \in \{0, \ldots, n\}$. Whenever we get a new measurement we update the posterior using Bayes rule. This amounts simply to multiplication by the likelihood of the measurement (because $\{w_i\}_{i=1}^j$ are all independent) followed by a normalization, therefore

$$p_l(j+1) = \frac{p_l(j) \exp\left(-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(l)\right)\right)^2}{\sum_{m=0}^n p_m(j) \exp\left(-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(m)\right)\right)^2} \ ,$$

where $\sigma_u^2 = 2\sigma^2$ for reasons stated in the next section. We consider the maximum *a posteriori* (MAP) estimator

$$\widehat{\theta}_k \stackrel{\Delta}{=} \arg\max_l p_l(k) \ .$$

Note that the outcome of the estimator does not depend on σ_u^2 as long as $\sigma_u^2 > 0$. Finally our estimate of $\boldsymbol{f}$ is simply $\widehat{\boldsymbol{f}}_k \stackrel{\Delta}{=} \boldsymbol{f}(\widehat{\theta}_k)$.

For the multidimensional classes $\mathcal{F}_d$, $d > 1$, it suffices to note that the multidimensional signals of interest can be interpreted as a collection of one-dimensional step function signals from the class $\mathcal{F}_1$. For example, in the two-dimensional case, such as that depicted in Fig. 5.1, each column of the image matrix is a one-dimensional signal in $\mathcal{F}_1$. Thus, we can apply the one-dimensional CS and reconstruction process on image columns separately (although this procedure might not be entirely adequate if the total number of samples is very small, *i.e.*, $k \ll n$). Details of the method are provided in the proof below. Conversion of the multi-dimensional problem into a series of one-dimensional problems is a standard technique in the analysis of signal models in this class, and it is carefully detailed in Chapter 2. Note that the samples are still completely non-adaptive, however this CS scheme differs slightly from other CS proposals in multiple dimensions in which the random projections are taken over the entire array [50, 56, 57], rather than column by column.

5.4 Proof of Theorem 14

The proof of Theorem 14 employs an analysis technique similar in spirit to one used by Burnashev and Zigangirov [23], and detailed in Appendix A. First define

$$M_\theta(j) = \frac{1 - p_\theta(j)}{p_\theta(j)}, \text{ and } N_\theta(j+1) = \frac{M_\theta(j+1)}{M_\theta(j)} \ .$$

Noticing that $\sum_{l=0}^{n} p_l(j) = 1$ we have

$$\begin{aligned} \Pr(\hat{\theta}(k) \neq \theta) &\leq \Pr\left(p_\theta(k) < \frac{1}{2}\right) = \Pr(M_\theta(k) > 1) \\ &\leq \mathbb{E}[M_\theta(k)] \ , \end{aligned}$$

where the last inequality follows from Markov inequality. The definition of $M_\theta(j)$ is chosen to get more leverage out of Markov's inequality (akin to Chernoff bounding techniques). Now we proceed by conditioning

$$\begin{aligned} \mathbb{E}[M_\theta(k)] &= \mathbb{E}[M_\theta(k-1)N_\theta(k)] \\ &= \mathbb{E}\left[M_\theta(k-1)\mathbb{E}[N_\theta(k)|\boldsymbol{p}(k-1)]\right] \\ &\vdots \\ &= M_\theta(0)\mathbb{E}\big[\mathbb{E}[N_\theta(1)|\boldsymbol{p}(0)] \times \cdots \times \mathbb{E}[N_\theta(k)|\boldsymbol{p}(k-1)]\big] \\ &\leq M_\theta(0)\left\{\max_{j\in\{0,\ldots,k-1\}} \max_{\boldsymbol{p}(j)} \mathbb{E}[N_\theta(j+1)|\boldsymbol{p}(j)]\right\}^k \ . \end{aligned}$$

The remainder of the proof entails upper bounding $\mathbb{E}[N_\theta(j+1)|\boldsymbol{p}(j)]$. Plugging

in the definitions we get

$$\mathbb{E}[N_\theta(j+1)|\boldsymbol{p}(j)] = \frac{1}{1-p_\theta(j)} \sum_{m \neq \theta} p_m(j) \mathbb{E}\left[\frac{\mathrm{e}^{-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(m)\right)^2}}{\mathrm{e}^{-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \phi^T(j+1)\boldsymbol{f}(\theta)\right)^2}} \right] .$$

To evaluate the above summation we consider two separate cases: (i) $m < \theta$; (ii) $m > \theta$. After some tedious but straightforward algebra we conclude that

$$\mathbb{E}\left[\frac{\mathrm{e}^{-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(m)\right)^2}}{\mathrm{e}^{-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(\theta)\right)^2}} \right] = \mathbb{E}\left[\exp\left(-2\left(\frac{1}{\sigma_u^2} - \frac{\sigma^2}{\sigma_u^4} \right) \sum_{t=\min(m,\theta)+1}^{\max(m,\theta)} \boldsymbol{\phi}_t(j+1)^2 \right) \right] .$$

The above expression is minimized when $\sigma_u^2 = 2\sigma^2$, justifying our choice for σ_u^2. Although it is not easy to compute the above expectations for general values of m and θ, it is relatively easy to conclude that those are largest when $|m - \theta| = 1$ or $|m - \theta| = 2$, therefore

$$\mathbb{E}\left[\frac{\mathrm{e}^{-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(m)\right)^2}}{\mathrm{e}^{-\frac{1}{2\sigma_u^2}\left(y_{j+1} - \boldsymbol{\phi}^T(j+1)\boldsymbol{f}(\theta)\right)^2}} \right] \leq \max\left\{ \mathrm{e}^{-\frac{1}{2n\sigma^2}}, \frac{1}{2} + \frac{1}{2}\mathrm{e}^{-\frac{2}{n\sigma^2}} \right\} \triangleq \alpha(n, \sigma^2) .$$

Consequently $\mathbb{E}[N_\theta(j+1)|\boldsymbol{p}(j)] \leq \alpha(n, \sigma^2)$ and therefore

$$\Pr(\hat{\theta}(k) \neq \theta) \leq n \left[\alpha(n, \sigma^2)\right]^k .$$

A bound on the expected error then follows trivially, by considering a worst case

scenario when $\widehat{\theta} \neq \theta$,

$$\mathbb{E}\left[n^{-1}\|\hat{f}_k - f\|^2\right] \;\leq\; 4n\;[\alpha(n,\sigma^2)]^k\;.$$

If instead of compressive samples we used carefully chosen adaptive point samples (using ideas similar to the ones in [23]) then we would get bounds with the same structure, but instead of $\alpha(n,\sigma^2)$ the exponent would be $1/2 + 1/2\ \mathrm{e}^{-1/(2\sigma^2)}$. □

5.5 Proof of Theorem 15

For the multidimensional classes $\mathcal{F}_d$, $d > 1$, again note that the multidimensional signals of interest can be interpreted as a collection of one-dimensional step function signals from the class $\mathcal{F}_1$. Furthermore, we know from standard approximation theory that any Lipschitz function can be reasonably approximated by a piecewise constant function. These two observations along with the results for the one-dimensional case suffice to prove the general results for $d > 1$.

Let us first consider the two-dimensional case. Let $\bar{g}_m$ be the best piecewise constant fit to g on m equal-width intervals. Then $|g - \bar{g}_m| \leq Lm^{-1}$ by the Lipschitz assumption. We can estimate the levels of $\bar{g}_m$ using the one-dimensional CS method described previously, considering projections over image columns. We will consider m columns of the image, therefore using k/m samples per column. Putting all these

facts together yields the bound

$$\mathbb{E}\left[n^{-2}\|\boldsymbol{f}-\widehat{\boldsymbol{f}}_k\|^2\right] \leq L\frac{1}{m}+4n\alpha^{k/m}(n,\sigma^2)\ . \tag{5.1}$$

To minimize this bound we simply have to choose

$$\begin{aligned} m &= k\log(\alpha(n,\sigma^2))/\log(L/(-4nk\log(\alpha(n,\sigma^2)))) \\ &\sim k\log(\alpha^{-1}(n,\sigma^2))/\log(nk)\ , \end{aligned}$$

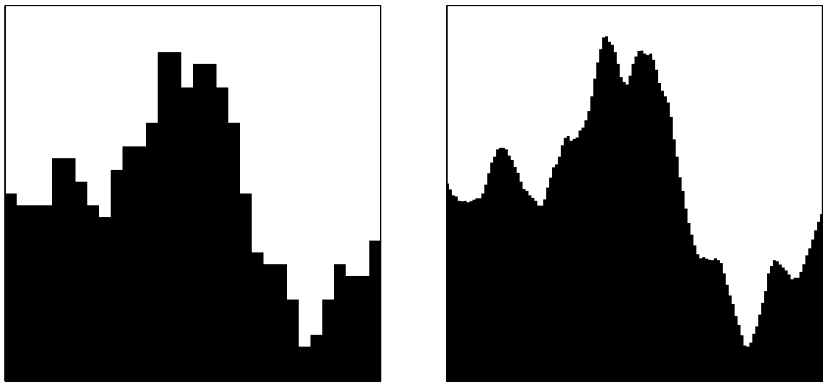
and therefore $\mathbb{E}\left[n^{-2}\|\boldsymbol{f}-\widehat{\boldsymbol{f}}_k\|^2\right] \leq C(n,L,\sigma^2)k^{-1}$, where $C(n,L,\sigma^2)$ can be computed using the value of m given above into the bound (5.1). The analysis and reasoning in the higher dimensional cases is analogous, and one can easily verify that taking k samples leads to a bound on the reconstruction error of $C(n,L,\sigma^2)k^{-1/(d-1)}$, where

$$C(n,L,\sigma^2) = L\frac{\log(4nk^{1/(d-1)})}{-\log\alpha(n,\sigma^2)}+1 \leq 6(L+1)(1+n\sigma^2)\log n\ .$$

□

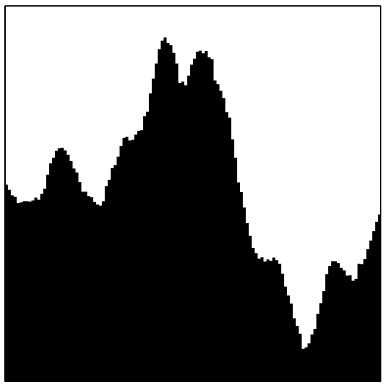
5.6 Experiments

To illustrate the theory and method developed here, we consider the problem of reconstructing the 1024×1024 boundary fragment image $\boldsymbol{f}$ depicted in Figure 5.1(a)

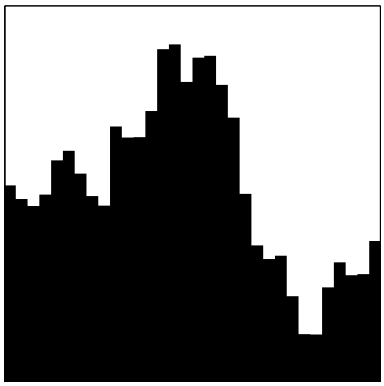


(a) Conventional pixel sampling k=1024, MSE=0.1471

(b) Adaptive pixel sampling k=1024, MSE = 0.0336



(c) Compressive sampling k=1024, MSE = 0.0359



(d) Compressive sampling k=256, MSE = 0.1431

Figure 5.2: *Reconstructions of image in Fig. 5.1(a) based on k noisy samples with* $\sigma = 0.15$.

from a limited number of noisy samples. We compare conventional (non-adaptive) pixel sampling, adaptive pixel sampling, and compressive sampling, with a Gaussian noise of standard deviation $\sigma = 0.15$ added to the samples in each case (equivalent to the noise level depicted in Figure 5.1(b)). In the experimental results depicted in Figure 5.2(a–c) we compare the methods using $k = 1024$ samples in each case. For the conventional pixel sampling case we subsample the original image on a 32×32 pixel lattice and add noise, resulting in the data depicted in Fig. 5.1(b). To reconstruct the image from the noisy pixel samples, we simply compute the maximum likelihood

estimate in each column, using the fact that it is known that the noiseless column is one of 32 possible step functions. The resulting reconstruction is shown in Fig. 5.2(a). In the adaptive sampling case, 128 uniformly spaced columns are selected and 8 adaptive pixel samples are taken in each column based on the method in [23]. The resulting reconstruction is shown in Fig. 5.2(b). Similarly, the compressive sampling is carried out by selecting 128 columns and taking 8 random projection samples in each column. The resulting reconstruction is shown in Fig. 5.2(c). As expected from our theory, compressive sampling and adaptive pixel sampling perform similarly, and both significantly outperform conventional pixel sampling. In Fig. 5.2(d) we present the results of compressed sensing using even fewer samples, namely only $k = 256$ random projections, split among 32 uniformly spaced columns and 8 random projections per column. The result is quite impressive since we get about the same performance as conventional pixel sampling, but with four times fewer samples.

The results depicted in Fig. 5.2 are representative of the performance comparison of the three methods at different sampling rates k and similar noise levels. Notice that the noise level in the simulation is relatively high, much larger than one might expect based on the upper bounds given by our theoretical analysis. This shows that our bounds (for CS in particular) are somewhat loose, and that even better performance than they predict can be expected in practice. The reason for this may be that our error bounds were derived from bounds on the probability of deciding on the wrong changepoint in each column, not the expected squared error directly. In practice

mistakes in the decision process often identify changepoints in the near vicinity of the true changepoint, leading to relatively small square errors.

5.7 Final Remarks

The theory and method presented here demonstrate that for certain classes of piecewise constant signals, compressive sampling is as effective as active sampling, provided the SNR is sufficiently high. This is a significant step forward in our understanding of compressive sampling, since previous results only demonstrated the optimality of compressive sampling in noiseless conditions. The method of reconstruction employed in our work differs markedly from the usual reconstruction strategies employed in compressive sampling (based on l_1 minimization techniques). Although this chapter only addresses signals in special classes it nevertheless indicates promising avenues for compressed sampling. There are several natural questions that arise: can active and compressive sampling be combined to improve further the potential of each one of them? Can the results of this chapter be extended for larger and more complex signal classes? While these remain unanswered we have pursued some ideas of adaptive compressive sampling, where instead of fully-random projections we use projections vectors that change shape as the estimation process goes along, focussing their "attention" at the relevant parts of the signal. Unfortunately analysis and design of such methodologies is quite challenging and relatively little is known at this point. It is interesting to point out that the analysis techniques used here can

also be applied in different contexts, in particular hypotheses testing using random projection samples [50].

Chapter 6

Concluding Remarks

In this thesis it is shown that active learning can dramatically improve performance, compared to commonly used passive learning techniques. This particularly true when the target of the learning process are concepts consisting mainly of spatially concentrated features. Although the potential gains of active learning seem very intuitive, there was a lack of understanding of its limitations. This thesis contributes to that understanding by clarifying in a general way when does active learning help, and how much does it help.

There are still many open questions regarding active learning, in particular good "recipes" for the construction of realistic and general active learning algorithms are still unknown. The work presented here is extremely relevant to assess the "goodness" of proposed algorithms: an algorithm under scrutiny can be applied to the settings described in Chapter 2 and its performance be compared with the performance of a minimax optimal algorithm (which we characterized). The outcomes of such experiments provide some guidance for the algorithm designer. This thesis opens up many avenues for future research, some of which are listed below.

Adaptive Sampling using Multi-Scale Methods: In Chapter 3 we saw how multiscale methods can be leveraged into simple active learning algorithms, with provable performance. These have actually been used is a practical setting for imaging using ballistic laser techniques [1], where very promising results were observed (see Figure 6.1). The key idea of the methods presented in Chapter 3 is to use spatially adaptive estimation procedures to effectively detect the regions where estimation is hard, and therefore extra sampling can be helpful. Other active learning methods, similar to the one described, can be devised in such a way that they perform as well as passive methods, but can lead into significant performance gains under most practical scenarios. From a theoretical analysis standpoint it is relatively easy to design and guarantee that these methods perform as well as passive learning. It is significantly harder to show that they perform better then passive methods, as it is patent from the extent of the proofs in Chapter 3. One possible two-step algorithm is based on wavelet thresholding ideas: in a first step we distribute samples in a uniform way over the domain, collect the corresponding observations and fit a wavelet-based model to this data. Large coefficients at finer resolutions indicate the presence of boundaries or complex features, and the regions corresponding to the support of the those wavelet basis functions are good candidates for resampling in a second step. In the second step we collect further samples in the perceived regions of interest and using all the data collected fit a wavelet-based model. If the procedure is carefully constructed one can guarantee that the extra sampling is not going to "confuse" the estimate, and

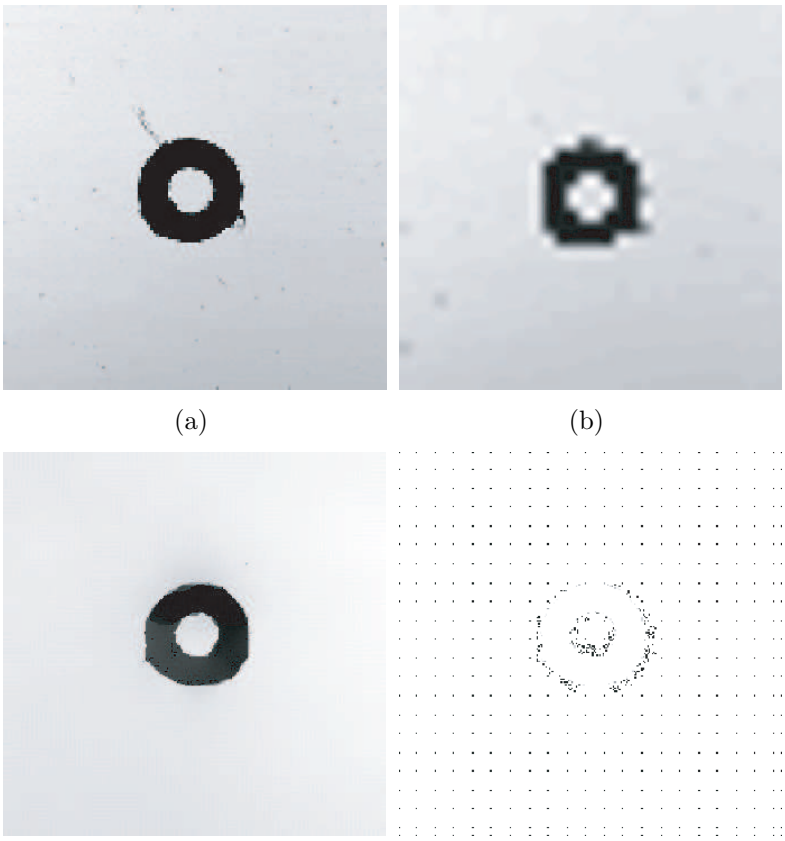


Figure 6.1: Adaptive sampling using ballistic laser imaging: (a) Estimate obtained scanning a turbid medium (with a small circular target) on a regular 128×128 grid (16384 passive samples). (b) Estimate through passive learning, using 32×32 samples (1024 passive samples). (c) Estimate through active learning using 984 samples. (d) Sample locations used in the active learning procedure (see [1] for details).

can only improve it.

Adaptive Compressive Sampling: In the concluding remarks of Chapter 5 it is pointed out that adaptive compressive sampling procedures can possibly be devised. This can be of interest for imaging methods where one can do projective measurements, but not necessarily in the form of randomized projections. A recent project that might benefit from these ideas is the "single-pixel" camera (see

http://www.dsp.ece.rice.edu/cs/cscamera/). The main idea there is to have a imaging device the uses a single-pixel sensing element. The observations are made using a micro-mirror device projecting the image under observation into that single sensing element. Such an architecture is very promising for hyper-spectral imaging, where it is hard to construct and accommodate multiple sensing elements able to span a large part of the electromagnetic spectrum. So far the imaging methods used hinged on compressive sensing ideas, therefore using non-adaptive randomized projections. If instead one adapts the projections based on previews observations it might be possible to improve significantly the performance of these special imaging systems. We have done some progress towards this goal, using some insights put forward in Chapter 5. If talking about estimation of a sparse vector one can "shape" the projection vectors using the posterior probability information. If done carefully this ensures that as more measurements are collected the measurement projection is going to focus on the significant coefficients, and ignore the other entries (that contribute mostly with noise). Although the procedure is very intuitive, it is hard to guarantee even consistency of the estimate.

Classification and Selective Sampling: In the introductory chapter we made reference to alternative sampling paradigms, in particular a paradigm called selective sampling, where the learner is provided with a sequence of unlabeled examples and for each example he can decide whether or not to collect the corresponding label. Although it is possible to study this paradigm under the classical statistical learning

framework, described in Chapter 2, the online learning setting might be more adequate. In the classical statistical learning framework we consider a training data set, use it to learn a inference rule, and assess the performance against the Bayes classifier, that is, the best classification rule given the knowledge of the underlying model. In the online learning setting on the other hand one uses the same training set to assess the performance: the goal is to minimize the cumulative training error in the training data: When confronted with a new training example (a pair feature vector/label) one evaluates how the current inference rule performs on the new example, before using the corresponding label to update the rule. The advantage of this paradigm is that even with very few assumptions made about the relationship between features and labels it is possible to provide meaningful performance guarantees.

We describe here an idea being currently pursued, cast in the sequence prediction setting, namely prediction with expert advise. The ideas described here can be immediately applied to classification problems simply by incorporating side information. Assume you are trying to predict a binary sequence $\{y_t\}_{t=1}^n$ (*e.g.*, rain or shine today) and have access to N experts (*e.g.*, forecasts from different news sources). Formally we denote the sequence of outcomes to be predicted by $y_1, \ldots, y_n$ and the expert advice by $f_{i,t}$, where this denotes the advise expert i provides at time t. For each round $t = 1, \ldots, n$

1. The environment chooses outcome y_t and expert advice $\{f_{i,t}\}_{i=1}^N$. The expert advice is revealed to the forecaster.

2. The forecaster makes a prediction $\widehat{p}_t$

3. The forecaster decides whether or not to have y_t revealed. Let Z_t be the indicator of that decision (1 if y_t is revealed and 0 otherwise).

4. A loss $\ell(\widehat{p}_t, y_t)$ is incurred by the forecaster and a loss $\ell(f_{i,t}, y_t)$ is incurred by expert i.

The main goal is to analyze the performance of the forecaster, namely upperbound

$$\mathbb{E}[\widehat{L}_n] - \min_{i=1,\ldots,N} L_{i,n} \ ,$$

where $\widehat{L}_n = \sum_{t=1}^n \ell(\widehat{p}_t, y_t)$ and $L_{i,n} = \sum_{t=1}^n \ell(f_{i,t}, y_t)$. In the above the expectation pertains the random choices made by the algorithm. The question we are interested is: is it possible to construct an algorithm that is adaptive, in the sense that it can take advantage of extra structure of the sequence if it is present, collecting few outcomes, but do almost as well as a methodology that collects all the outcomes? Assume we are using the zero-one loss (that is, $\ell(y_1, y_2) = \mathbf{1}\{y_1 \neq y_2\}$). We propose an algorithm that is a slight variation of the exponential weighted average method (a excellent overview of these methods is available in [5]). Namely at time t expert i has a weight $w_{i,t}$ associated with it (with $w_{i,0} = 1$ for all i). The weight update rule has the form

$$w_{i,t} = \exp\left(-\eta \sum_{j=1}^{t} \ell(f_{i,j}, y_j) Z_j\right) \ .$$

Let $W_t = \sum_{i=1}^N w_{i,t}$ and define the prediction rule

$$\widehat{p}_t = \begin{cases} f_{1,t} & \text{w.p. } w_{1,t-1}/W_{t-1} \\ f_{2,t} & \text{w.p. } w_{2,t-1}/W_{t-1} \\ \vdots & \\ f_{N,t} & \text{w.p. } w_{N,t-1}/W_{t-1} \end{cases} . \tag{6.1}$$

In words, we choose the i^{th} expert prediction with probability $w_{i,t-1}/W_{t-1}$. Finally we need to define the sampling decision variable Z_t. This is a Bernoulli random variable with parameter depending on the weights $\{w_{i,t-1}\}_{i=1}^N$ and the expert decision at time t $\{f_{i,t}\}_{i=1}^N$. Based on the analysis of a simple scenario and empirical evidence we propose the following sampling strategy. Define

$$A_{0,t} = \frac{\sum_{i:f_{i,t}=0} w_{i,t-1}}{W_{t-1}} ,$$

and $A_{1,t} = 1 - A_{0,t}$. Now take $Z_t = \text{Ber}(p(A_{0,t}))$. A choice that seems to work well is

$$p(A_{0,t}) = -\frac{1}{\log_2(\min(A_{0,t}, 1 - A_{0,t}))} = \frac{1}{1 - \log_2(1 - |2A_{0,t} - 1|)} . \tag{6.2}$$

There are several questions that beg for an answer:

1. Do such method performs "as well" as a method that collects all the outcomes. For such a method $\mathbb{E}[\widehat{L}_n] - \min_{i=1,\ldots,N} L_{i,n} \leq \log(N)/\eta + \eta n/8$, by using essentially the result in theorem 2.2 of [5]. Taking $\eta = 2\sqrt{2(\log N)/n}$ yields

$\mathbb{E}[\widehat{L}_n] - \min_{i=1,\ldots,N} L_{i,n} \leq \sqrt{2n \log N}$, which is essentially the best possible bound without further assumptions.

2. Is it possible to devise a method that can perform almost as well as a "passive" method (that collects all outcomes), or on the contrary can we come up with a sequence that "defeats" such approaches?

Empirically it seems that the answer to both questions is affirmative. When using this algorithm in the settings of Chapter 2 we observe empirically that (with a careful choice of experts) it achieves the minimax lower bounds predicted, and even more desirably it gives us an adaptive procedure (recall the final remarks of Chapter 2), therefore we do not need to know the noise margin in advance. Furthermore this methodology can be made practical and realistic in many settings.

There are several reasons behind the choice in (6.2), most of them empirical, but it is possible to build a justification from the analysis of a simple scenario. Suppose there is an expert that predicts the outcome sequence without errors. In such a scenario it makes sense to use the above algorithm with $\eta = \infty$, since if an expert makes a mistake we can discard it without much concern (with $\eta = \infty$ the weight vector $\boldsymbol{w}_t$ is binary, and indicates which experts agree with the sequence up to time t). We will consider a modification of the algorithm, namely consider the majority vote prediction rule

$$\widehat{p}_t = \mathbf{1}\{A_{1,t} > 1/2\} \ . \tag{6.3}$$

This prediction rule is not randomized and can be shown to be worthless if no prefect expert exists (any deterministic prediction rule will make at least $n/2$ errors in a worst case scenario). It can easily be shown that if all the samples are collected, that is $Z_t = 1$ for all t then

$$\mathbb{E}[\widehat{L}_n^{(N)}] \leq \lceil \log_2 N \rceil \ ,$$

where the superscript (N) explicitly indicates the dependence on the number of experts (this result is valid under both prediction rules (6.1) and (6.3)). We showed that

Theorem 16. *Let* $\widehat{L}_n^{(N)} = \sum_{i=1}^n \ell(\widehat{p}_t, y_t)$*, where* $\widehat{p}_t$ *is given by* (6.3)*. Then*

$$\sum_{i=1}^n \mathbb{E}[\ell(\widehat{p}_t, y_t) | \{y_j, f_{i,j}\}_{i \in \{1,\ldots,N\}, j \in \{1,\ldots,t\}}] \leq \log_2 N \ .$$

Consequently

$$\mathbb{E}[\widehat{L}_n^{(N)}] \leq \log_2 N \ .$$

This indicates that the proposed algorithm is doing as well as one collecting all the outcomes. Furthermore the above bound is tight and one cannot be improved. The proof of the theorem goes by finite induction, and we do not know yet how to generalize it when no perfect experts exist. Finally we would like to point out that the ideas behind exponentially weighted prediction with experts are intimately related to boosting, which can also be used to devise active learning algorithms, such as in [4]. We are currently investigating ways of making these boosting algorithms theoretically

sound, as well as practical and realistic.

Appendix A

The Burnashev-Zigangirov Algorithm

In this appendix we present formally the Burnashev-Zigangirov algorithm and characterize its performance. In what follows we use the symbol $\oplus$ to denote the sum *modulo* two, that is, the exclusive o*r* operation. For convenience some of the notation is slightly different then in the rest of the thesis, in particular the noise variables U_j are Bernoulli with parameter $p(j)$, with $0 \leq p(j) \leq p$ (c was used in Chapter 2 instead of p). We also use the term "posterior" probability in a rather loose way, since this is somehow an approximate posterior probability. The algorithm and observation details are shown in Figure A.1. We have the following remarkable result.

Theorem 17 (Burnashev-Ziganzirov 1973)**.**

$$\sup_{\theta^* \in [0,1]} \Pr(\theta^* \notin \widehat{I}_n) \leq \frac{1-t}{t} \left(\frac{1-p}{2(1-\alpha)} + \frac{p}{2\alpha} \right)^n .$$

Remarks: The above theorem shows that, even under noisy assumptions, there is a dramatic improvement in performance if one allows adaptive sampling strategies. Although the bounds display the exponential error decay behavior, also present in the

Initialization: Let $t > 0$ be such that $m \triangleq t^{-1} \in \mathbb{N}$. Define the posterior after j measurements as $p_j : [0,1] \to \mathbb{R}$,

$$p_j(x) = \sum_{i=1}^{m} a_i(j) \mathbf{1}_{I_i}(x) \ ,$$

where $I_1 = [0,t]$ and $I_i = (t(i-1), ti]$, for $i \in \{2, \ldots, t^{-1}\}$. The collection $\{I_i\}$ is a partition of the interval $[0,1]$. Initialize this posterior taking $a_i(0) = t$. The posterior is completely characterized by $\boldsymbol{a(j)} = \{a_1(j), \ldots, a_m(j)\}$, and that $\sum_{i=1}^{m} a_i(j) = 1$.

1 - **Sample Selection:** To preserve the parametric structure of the posterior we take samples at the interval subdivision points. Define $k(j)$ such that

$$\sum_{i=1}^{k(j)-1} a_i(j) \leq 1/2, \quad \sum_{i=1}^{k(j)} a_i(j) > 1/2 \ .$$

Note that $k(j) \in \{1, \ldots, m\}$. Select X_{j+1} among $\{t(k(j)-1), tk(j)\}$ by flipping a coin, choosing the first point with probability $\pi_1(j)$ and the second point with probability $\pi_2(j) = 1 - \pi_1(j)$, where $\pi_1(j) \triangleq \tau_2(j)/(\tau_1(j) + \tau_2(j))$ and

$$\tau_1(j) \triangleq \sum_{i=k(j)}^{m} a_i(j) - \sum_{i=1}^{k(j)-1} a_i(j) \ , \quad \tau_2(j) \triangleq \sum_{i=1}^{k(j)} a_i(j) - \sum_{i=k(j)+1}^{m} a_i(j) \ .$$

2 - **Noisy Observation:** Observe $Y_{j+1} = \mathbf{1}\{X_{j+1} \geq \theta^*\} \oplus U_{j+1}$, where the random variables U_j are independent Bernoulli random variables with parameters $p(j)$.

3 - **Update posterior:** Update the posterior function after collecting the measurement Y_{j+1}, through the application of Bayes rule (for the application of Bayes rule we "assume" the Bernoulli random variable U_{j+1} to have parameter $0 \leq \alpha < 1/2$ instead of $p(j)$). Let $\beta = 1 - \alpha$. Note that $X_{j+1} = tk$, $k \in \mathbb{N}$ and define $\tau = \sum_{i=1}^{k} a_i(j) - \sum_{i=k+1}^{m} a_i(j)$. For $i \leq k$ we have

$$a_i(j+1) = a_i(j) \begin{cases} \frac{2\alpha}{1-\tau(\beta-\alpha)} & \text{if } Y_{j+1} = 0 \\ \frac{2\beta}{1+\tau(\beta-\alpha)} & \text{if } Y_{j+1} = 1 \end{cases} \ ,$$

and for $i > k$

$$a_i(j+1) = a_i(j) \begin{cases} \frac{2\beta}{1-\tau(\beta-\alpha)} & \text{if } Y_{j+1} = 0 \\ \frac{2\alpha}{1+\tau(\beta-\alpha)} & \text{if } Y_{j+1} = 1 \end{cases} \ ,$$

4 - **Final estimate:** Repeat steps 1,2 and 3 until n samples are collected. The final confidence interval $\widehat{I}_n$ is given by the bin of the posterior that has the largest value. Define $\widehat{i} \triangleq \arg\max_{i \in \{1,\ldots,m\}} a_i(n)$. Then

$$\widehat{I}_n \triangleq I_{\widehat{i}} \ .$$

Figure A.1: The Burnashev-Zigangirov (BZ) algorithm.

noiseless scenario, the exponent depends on the noise parameter p. Finally, we note that although this result was derived for a particular noise model the result is applicable to other noise models. This can be done either by processing the observations, using a thresholding operator, or by modifying the likelihood structure (according to the noise model) in the proof of the Theorem.

The proof of Theorem 17 is extremely elegant and is presented below. The ideas in the proof can be used in various other contexts where feedback is present.

Proof of Theorem 17. For the proof we rely on the notation in the algorithm given in Figure A.1. In particular recall that the unit interval is divided into subintervals of width t, $a_i(j)$ denotes the posterior probability that the changepoint θ^* is located in the i-th subinterval after the j-th sample.

Our first step is to construct an upper bound for the probability $\Pr(\theta^* \notin \widehat{I}_n)$. Let θ^* be fixed, but arbitrary, and define k^* to be the index of the bin containing θ^*, that is $\theta^* \in I_{k^*}$. Define

$$M^*(j) = \frac{1 - a_{k^*}(j)}{a_{k^*}(j)} \ ,$$

and

$$N^*(j+1) = \frac{M^*(j+1)}{M^*(j)} = \frac{a_{k^*}(j)(1 - a_{k^*}(j+1))}{a_{k^*}(j+1)(1 - a_{k^*}(j))} \ .$$

The reasoning behind these definitions is made clear later. For now, notice that $M^*(j)$

is a decreasing function of $a_{k^*}(j)$. We have

$$\begin{aligned}
\Pr(\theta^* \notin \widehat{I}_n) &\leq \Pr(a_{k^*}(j) < 1/2) \\
&= \Pr(M^*(n) > 1) \\
&\leq \mathbb{E}[M^*(n)] \ ,
\end{aligned}$$

where the last step follows from Markov's inequality. The definition of $M^*(j)$ above is meant to get more leverage out of Markov's inequality, in a similar spirit of Chernoff bounding techniques. Using the definition of $N^*(j)$ and some conditioning we get

$$\begin{aligned}
\mathbb{E}[M^*(n)] &= \mathbb{E}[M^*(n-1)N^*(n)] \\
&= \mathbb{E}\left[\mathbb{E}[M^*(n-1)N^*(n)|\boldsymbol{a}(n-1)]\right] \\
&= \mathbb{E}\left[M^*(n-1)E[N^*(n)|\boldsymbol{a}(n-1)]\right] \\
&\vdots \\
&= M^*(0)E\left[E[N^*(1)|\boldsymbol{a}(0)]\cdots E[N^*(n)|\boldsymbol{a}(n-1)]\right] \\
&\leq M^*(0)\left\{\max_{j\in\{0,\ldots,n-1\}} \max_{\boldsymbol{a}(j)} E[N^*(j+1)|\boldsymbol{a}(j)]\right\}^n . \qquad \text{(A.1)}
\end{aligned}$$

The rest of the proof consists of showing that $E[N^*(j+1)|\boldsymbol{a_j}] \leq 1-\epsilon$, for some $\epsilon > 0$. Before proceeding we make some remarks about the above technique. Note that $M^*(j)$ measures how much mass is on the bin containing θ^* (if $M^*(j) = 0$ all the mass in our posterior is in the bin containing θ, the least error scenario). The

ratio $N^*(j)$ is a measure of the improvement (in terms of concentrating the posterior around the bin containing θ^*) by sampling at X_j and observing Y_j. This is strictly less than one when an improvement is made. The bound (A.1) above is therefore only useful if, no matter what happened in the past, a measurement made with the proposed algorithm always leads on average to a performance improvement. This is the case with a variety of other useful myopic algorithms.

To study $E[N^*(j+1)|\boldsymbol{a}(j)]$ we are going to consider three particular cases: (i) $k(j) = k^*$; (ii) $k(j) > k^*$; and (iii) $k(j) < k^*$. Let $\beta = 1-\alpha$, $q = 1-p$, and $q(j) = 1-p(j)$. After tedious but straightforward algebra we conclude that

$$N^*(j+1) = \begin{cases} \frac{1+(\beta-\alpha)x}{2\beta} & \text{, with probability } q(j) \\ \frac{1-(\beta-\alpha)x}{2\alpha} & \text{, with probability } p(j) \end{cases},$$

where we have for the three different cases

(i)

$$x = \begin{cases} \frac{\tau_1(j)-a_{k^*}(j)}{1-a_{k^*}(j)} & \text{, if } X_{j+1} = t(k(j)-1) \\ \frac{\tau_2(j)-a_{k^*}(j)}{1-a_{k^*}(j)} & \text{, if } X_{j+1} = tk(j) \end{cases}$$

(ii)

$$x = \begin{cases} -\frac{\tau_1(j)+a_{k^*}(j)}{1-a_{k^*}(j)} & \text{, if } X_{j+1} = t(k(j)-1) \\ \frac{\tau_2(j)-a_{k^*}(j)}{1-a_{k^*}(j)} & \text{, if } X_{j+1} = tk(j) \end{cases}$$

(iii)

$$x = \begin{cases} \frac{\tau_1(j) - a_{k^*}(j)}{1 - a_{k^*}(j)} & \text{, if } X_{j+1} = t(k(j)-1) \\ -\frac{\tau_2(j) + a_{k^*}(j)}{1 - a_{k^*}(j)} & \text{, if } X_{j+1} = tk(j) \end{cases}$$

Note that $0 \leq \tau_1(j) \leq 1$ and $0 < \tau_2(j) \leq 1$, therefore $|x| \leq 1$. To ease the notation define

$$\begin{aligned} g(x,j) &= \frac{q(j)}{2\beta}(1 + (\beta - \alpha)x) + \frac{p(j)}{2\alpha}(1 - (\beta - \alpha)x) \\ &= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} + \left(\frac{q(j)}{2\beta} - \frac{p(j)}{2\alpha}\right)(\beta - \alpha)x \ . \end{aligned}$$

It can be easily checked that $g(x,j)$ is an increasing function. Using this definition we have

(i)

$$\begin{aligned} & E[N^*(j+1)|\boldsymbol{a}(j)] \\ &= \pi_1(j) g\left(\frac{\tau_1(j) - a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) + \pi_2(j) g\left(\frac{\tau_2(j) - a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) \end{aligned}$$

(ii)

$$\begin{aligned} & E[N^*(j+1)|\boldsymbol{a}(j)] \\ &= \pi_1(j) g\left(-\frac{\tau_1(j) + a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) + \pi_2(j) g\left(\frac{\tau_2(j) - a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) \end{aligned}$$

(iii)

$$
\begin{aligned}
& E[N^*(j+1)|\boldsymbol{a}(j)] \\
&= \pi_1(j) g\left(\frac{\tau_1(j) - a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) + \pi_2(j) g\left(-\frac{\tau_2(j) + a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right)
\end{aligned}
$$

Consider first cases (ii) and (iii). Note that $(\tau - a)/(1-a) \leq \tau$ and $-(\tau+a)/(1-a) < -\tau$ for all $0 < a < 1$. Therefore, for case (ii) we have

$$
\begin{aligned}
& E[N^*(j+1)|\boldsymbol{a}(j)] \leq \pi_1(j) g(-\tau_1(j), j) + \pi_2(j) g(\tau_2(j), j) \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} + \left(\frac{q(j)}{2\beta} - \frac{p(j)}{2\alpha}\right)(\beta - \alpha)(-\pi_1(j)\tau_1 + \pi_2(j)\tau_2) \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} \leq \frac{q}{2\beta} + \frac{p}{2\alpha} \, .
\end{aligned}
$$

Analogously, for case (iii)

$$
\begin{aligned}
& E[N^*(j+1)|\boldsymbol{a}(j)] \leq \pi_1(j) g(\tau_1(j), j) + \pi_2(j) g(-\tau_2(j), j) \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} + \left(\frac{q(j)}{2\beta} - \frac{p(j)}{2\alpha}\right)(\beta - \alpha)(\pi_1(j)\tau_1 - \pi_2(j)\tau_2) \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} \leq \frac{q}{2\beta} + \frac{p}{2\alpha} \, .
\end{aligned}
$$

Finally, for case (i) a we need to proceed in a slightly different way. Begin by noticing

that $\tau_1(j) + \tau_2(j) = 2a_{k(j)}(j) = 2a_{k^*}(j)$. Then

$$
\begin{aligned}
& E[N^*(j+1)|\boldsymbol{a}(j)] \\
&= \pi_1(j) g\left(\frac{\tau_1(j) - a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) + \pi_2(j) g\left(-\frac{\tau_1(j) - a_{k^*}(j)}{1 - a_{k^*}(j)}, j\right) \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} + \left(\frac{q(j)}{2\beta} - \frac{p(j)}{2\alpha}\right)(\beta - \alpha)\frac{\tau_1 - a_{k^*}(j)}{1 - a_{k^*}(j)}\left(\pi_1(j) - \pi_2(j)\right) \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} + \left(\frac{q(j)}{2\beta} - \frac{p(j)}{2\alpha}\right)(\beta - \alpha)\frac{\tau_1 - a_{k^*}(j)}{1 - a_{k^*}(j)}\ \frac{\tau_2(j) + \tau_1(j)}{\tau_1(j) + \tau_2(j)} \\
&= \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} + \left(\frac{q(j)}{2\beta} - \frac{p(j)}{2\alpha}\right)(\beta - \alpha)\frac{\tau_1 - a_{k^*}(j)}{1 - a_{k^*}(j)}\ \frac{2a_{k^*}(j) - 2\tau_1(j)}{\tau_1(j) + \tau_2(j)} \\
&\leq \frac{q(j)}{2\beta} + \frac{p(j)}{2\alpha} \leq \frac{q}{2\beta} + \frac{p}{2\alpha}\ .
\end{aligned}
$$

Plugging in the above results into (A.1) yields

$$
\Pr(|\widehat{\theta}_n - \theta| > t) \leq \frac{1-t}{t}\left(\frac{q}{2\beta} + \frac{p}{2\alpha}\right)^n\ ,
$$

since $M^*(0) = (1-t)/t$. □

Bibliography

[1] S. Farsiu, J. Christofferson, B. Eriksson, P. Milanfar, B. Friedlander, A. Shakouri, and R. Nowak, "Statistical detection and imaging of objects hidden in turbid media using ballistic photons," Submitted to Applied Optics, February 2007.

[2] R. H. S. Carpenter, *Movements of the Eyes*, Pion, London, 2 edition, 1988.

[3] S. Martinez-Conde, S. L. Macknik, and D. H. Hubel, "The role of fixational eye movements in visual perception," *Nature Reviews Neuroscience*, vol. 5, no. 3, pp. 229–240, 2004.

[4] G. Tur, Dilek Hakkani-Tür, and R. E. Schapire, "Combining active and semi-supervised learning for spoken language understanding," *Speech Communication*, vol. 45, pp. 171–186, 2005.

[5] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning and Games*, Cambridge University Press, 2006.

[6] R. Willett, A. Martin, and R. Nowak, "Backcasting: Adaptive sampling for sensor networks," in *Proc. Information Processing in Sensor Networks*, 26-27 April, Berkeley, CA, USA, 2004.

[7] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, "Network tomography: Recent developments," *Statistical Science*, vol. 19, no. 3, pp. 499–517, 2004.

[8] S. K. Thompson and G. A. F. Seber, *Adaptive Sampling*, John Wiley & Sons, Inc., New York, 1996.

[9] A. Wald, *Sequential Analysis*, John Wiley & Sons, Inc., 1947.

[10] M. Ghosh, N. Mukhopadhyay, and P. Sen, *Sequential Estimation*, John Wiley & Sons, Inc., 1997.

[11] P. K. Sen, *Sequential Nonparametrics*, John Wiley & Sons, Inc., 1981.

[12] D. J. C. Mackay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, pp. 698–714, 1991.

[13] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," *Journal of Artificial Intelligence Research*, pp. 129–145, 1996.

[14] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine Learning*, vol. 28, no. 2-3, pp. 133–168, August 1997.

[15] N. Cesa-Bianchi, A. Conconi, and C. Gentile, "Learning probabilistic linear-threshold classifiers via selective sampling," in *The Sixteenth Annual Conference on Learning Theory. LNAI 2777, Springer*, 2003.

[16] S. Dasgupta, A. Kalai, and C. Monteleoni, "Analysis of perceptron-based active learning," in *Eighteen Annual Conference on Learning Theory (COLT)*, 2005.

[17] S. Dasgupta, "Coarse sample complexity bounds for active learning," in *Advances in Neural Information Processing (NIPS)*, 2005.

[18] S. Dasgupta, "Analysis of a greedy active learning strategy," in *Advances in Neural Information Processing (NIPS)*, 2004.

[19] N. Balcan, A. Beygelzimer, and J. Langford, "Agostic active learning," in *23rd International Conference on Machine Learning*, Pittsburgh, PA, USA, 2006.

[20] R. Castro, R. Willett, and R. Nowak, "Faster rates in regression via active learning," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2005, extended version available at `http://homepages.cae.wisc.edu/~rcastro/ECE-05-3.pdf`.

[21] M. Kääriäinen, "On active learning in the non-realizable case," NIPS Workshop on Foundations of Active Learning, 2005.

[22] M. Horstein, "Sequential decoding using noiseless feedback," *IEEE Trans. Info. Theory*, vol. 9, no. 3, pp. 136–143, 1963.

[23] M. V. Burnashev and K. Sh. Zigangirov, "An interval estimation problem for controlled observations," *Problems in Information Transmission*, vol. 10, pp. 223–231, 1974, (Translated from *Problemy Peredachi Informatsii,* 10(3):51–61, July-September, 1974. Original article submitted June 25, 1973).

[24] P. Hall and I. Molchanov, "Sequential methods for design-adaptive estimation of discontinuities in regression curves and surfaces," *The Annals of Statistics*, vol. 31, no. 3, pp. 921–941, 2003.

[25] G. Golubev and B. Levit, "Sequential recovery of analytic periodic edges in the binary image models," *Mathematical Methods of Statistics*, vol. 12, pp. 95–115, 2003.

[26] B. Bryan, J. Schneider, R. C. Nichol, Christopher J. Miller, C. R. Genovese, and L. Wasserman, "Active learning for identifying function threshold boundaries," in *Advances in Neural Information Processing (NIPS)*, 2005.

[27] A. Tsybakov, "Optimal aggregation of classifiers in statistical learning," *The Annals of Statistics*, vol. 32, no. 1, pp. 135–166, 2004.

[28] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*, Springer-Verlag, 1996.

[29] E. Baum, "Neural net algorithms that learn is polynomial time from examples and queries," *IEEE Transaction on Neural Networks*, vol. 2, pp. 5–19, 1991.

[30] L. Cavalier, "Nonparametric estimation of regression level sets," *Statistics*, vol. 29, pp. 131–160, 1997.

[31] A. B. Tsybakov, "On nonparametric estimation of density level sets," *Annals of Statistics*, vol. 25, pp. 948–969, 1997.

[32] A. B. Tsybakov, *Introduction à l'estimation non-paramétrique*, Mathématiques et Applications, 41. Springer, 2004.

[33] R. Castro and R. Nowak, "Upper and lower bounds for active learning," in *44th Annual Allerton Conference on Communication, Control and Computing*, 2006.

[34] A.P. Korostelev and A.B. Tsybakov, *Minimax Theory of Image Reconstruction*, Springer Lecture Notes in Statistics, 1993.

[35] D. Donoho, "Wedgelets: Nearly minimax estimation of edges," *The Annals of Statistics*, vol. 27, pp. 859–897, 1999.

[36] A. P. Korostelev, "On minimax rates of convergence in image models under sequential design," *Statistics & Probability Letters*, vol. 43, pp. 369–375, 1999.

[37] Alexander Korostelev and Jae-Chun Kim, "Rates of convergence for the sup-norm risk in image models under sequential designs," *Statistics & probability Letters*, vol. 46, pp. 391–399, 2000.

[38] C. de Boor, "The error in polynomial tensor-product and chung-yao, interpolation," in *Surface Fitting and Multiresolution Methods*, A LeMéhauté, C. Rabut, and L. Schumaker, Eds. 1997, pp. 35–50, Vanderbilt University Press.

[39] M. Crouse, R. Nowak, and R. Baraniuk, "Wavelet-based statistical signal processing using markov models," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 886–902, 1998.

[40] K. Falconer, *Fractal Geometry: Mathematical Foundations and Applications*, John Wiley & Sons, 1st edition, 1990.

[41] I. A. Ibragimov and R. Z. Has'minskii, *Statistical Estimation - Asymptotic Theory*, Springer-Verlag, 1981.

[42] I. A. Ibragimov and R. Z. Khas'minskii, "On sequential estimation of the location parameter for families of distributions with discontinuous densities," *Theory of Probability and its Applications*, vol. XIX, no. 4, pp. 669–682, 1974.

[43] Charles J. Stone, "Optimal rates of convergence for nonparametric estimators," *The Annals of Statistics*, vol. 8, no. 6, pp. 1348–1360, 1980.

[44] A. R. Barron, "Complexity regularization with application to artificial neural networks," in *Nonparametric Functional Estimation and Related Topics*. 1991, pp. 561–576, Kluwer Academic Publishers.

[45] M. Wegkamp, "Model selection in nonparametric regression," *The Annals of Statistics*, vol. 31, no. 1, pp. 252–273, 2003.

[46] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc., New York, 1991.

[47] L. Breiman, J. Friedman, R. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth, Belmont, CA, 1983.

[48] A. R. Barron, "Information theory and the risk of bayes procedures," in *Bayesian Statistics 6: Information-Theorectic Characterization of Bayes Performance and the Choice of Priors in Parametric and Non-parametric Problems*. 1999, Oxford University Press.

[49] C. Craig, "On the tchebychef inequality of bernstein," *The Annals of Statistics*, vol. 4, no. 2, pp. 94–102, 1933.

[50] J. Haupt and R. Nowak, "Signal reconstruction from noisy random projections," *IEEE Transactions on Information Theory*, vol. 52, no. 9, pp. 4036–4048, 2006.

[51] B. Laurent and P. Massard, "Adaptive estimation of a quadratic functional by model selection," *The Annals of Statistics*, vol. 28, no. 5, pp. 1302–1338, 2000.

[52] E. Candès and D. Donoho, "Curvelets: A surprisingly effective nonadaptive representation for objects with edges," in Curve and Surface Desing, L. L. Schumaker et al. (eds), Vanderbilt University Press, Nashville, TN., 2000.

[53] D. Donoho and X. Huo, "Beamlets and multiscale image analysis," Tech. Rep., Stanford University, 2001, Available at `http://www-stat.stanford.edu/~donoho/reports.html`.

[54] Q. Li and A. Barron, *Advances in Neural Information Processing Systems 12*, chapter Mixture Density Estimation, MIT Press, 2000.

[55] Eric D. Kolaczyk and Robert D. Nowak, "Multiscale likelihood analysis and complexity penalized estimation," *Annals of Statistics*, vol. 32, no. 2, pp. 500–527, 2004.

[56] E. J. Candès and Terence Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[57] D. Donoho, "Compressed sensing," Tech. Rep., Stanford, 2004.

[58] E. J. Candès and T. Tao, "The dantzig selector: statistical estimation when p is much larger than n.," *to appear in The Annals of Statistics*, 2007.