Assumptions & Expectations in Semi-Supervised Machine Learning

Mey, Alex

DOI
10.4233/uuid:a9388324-0067-4ab1-86fc-c5d21c882f1e

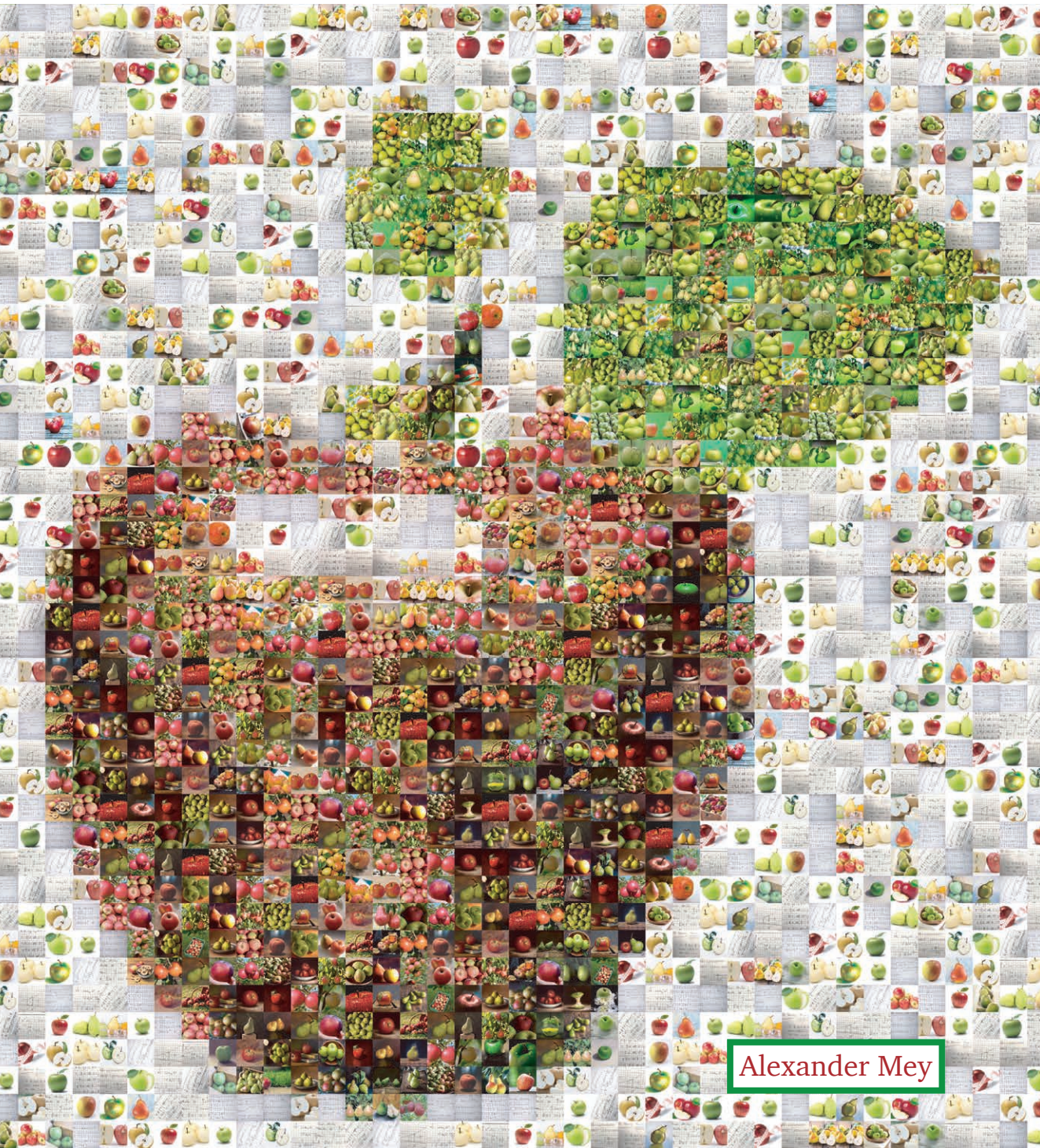
Publication date
2020

Document Version
Final published version

Citation (APA)
Mey, A. (2020). *Assumptions & Expectations in Semi-Supervised Machine Learning*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:a9388324-0067-4ab1-86fc-c5d21c882f1e

Assumptions & Expectations

in Semi-Supervised Machine Learning



Alexander Mey

Assumptions and Expectations in Semi-Supervised Machine Learning

Assumptions and Expectations in Semi-Supervised Machine Learning

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag 21 januari 2020 om 15.00 uur

door

Alexander Mey

Master of Science in Mathematics
Rheinische Friedrich-Wilhelms- Universität Bonn, Duitsland,
geboren te Mechernich, Duitsland.

Dit proefschrift is goedgekeurd door de promoters

prof. dr. ir. M.J.T. Reinders en
prof. dr. M. Loog

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. M.J.T. Reinders,	Technische Universiteit Delft
Prof. dr. M. Loog,	Technische Universiteit Delft, Universiteit Kopenhagen

Onafhankelijke leden:

Prof. dr. ir. G.J.T. Leus	Technische Universiteit Delft
Prof. dr. P.D. Grünwald	Centrum Wiskunde en Informatica, Amsterdam
Prof. dr. M. Biehl	Rijksuniversiteit Groningen
Dr. F.A. Oliehoek	Technische Universiteit Delft
Dr. B. Szörényi	Yahoo! NYC, USA
Prof. G. Jongbloed	Technische Universiteit Delft, reservelid

Keywords: Semi-supervised Learning, Statistical Learning Theory, Class Probability Estimation, Monotonic Learning

Printed by: Gildeprint

Front & Back: Beautiful cover art designed by Franka Rang and Alexander Mey that symbolizes the machine learning process: The essence of the apple is captured by the training examples guided through a theoretical framework.

ISBN 978-94-6366-243-7

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

To all the wonderful people that make each day a happy one.

CONTENTS

Summary xi

Samenvatting xiii

1 Introduction 1
- 1.1 Learning from Data . . . 2
- 1.2 Why Semi-Supervised Learning? . . . 2
- 1.3 How Semi-Supervised Learning? . . . 3
- 1.4 Challenges in Semi-Supervised Learning . . . 3
- 1.5 Organization of this Thesis . . . 4
- References . . . 6

2 A Review of Theoretical Results 7
- 2.1 Introduction and Scope . . . 8
  - 2.1.1 Outline . . . 8
- 2.2 Preliminaries . . . 9
- 2.3 Possibility & Impossibility of Semi-Supervised Learning . . . 9
  - 2.3.1 Impossibility Results . . . 10
  - 2.3.2 Proofs about the Possibility of Semi-Supervised Learning . . . 14
- 2.4 Learning Without Assumptions . . . 17
  - 2.4.1 Reweighing the Labeled Data By the Marginal Distribution . . . 18
  - 2.4.2 Using the Unlabeled Data to Pick the Center of the Version Space . . . 19
  - 2.4.3 Using Unlabeled Data to Combine Multiple Hypothesis Spaces . . . 21
- 2.5 Learning Under Weak Assumptions . . . 21
  - 2.5.1 A General Framework to Encode Weak Assumptions . . . 22
  - 2.5.2 Assuming that the Feature Space can be Split . . . 23
- 2.6 Learning Under Strong Assumptions . . . 24
  - 2.6.1 Assuming that the Model is Identifiable . . . 25
  - 2.6.2 Assuming that Classes are Clustered and Separated . . . 25
  - 2.6.3 Assuming that the Classes are Clustered but Not Necessarily Separated . . . 26
  - 2.6.4 Assuming the Regression Function is Smooth Along A manifold . . . 28
- 2.7 Learning in the Transductive Case . . . 31
  - 2.7.1 Transductive Learning Bounds . . . 31
  - 2.7.2 Safe Transductive Learning . . . 36
- 2.8 Discussion . . . 38
  - 2.8.1 On The Limits of Assumption Free SSL . . . 38
  - 2.8.2 How Good Can Constant Improvement Be? . . . 38
  - 2.8.3 The Amount of Unlabeled Data We Need . . . 39
  - 2.8.4 Using Assumptions in Semi-Supervised Learning . . . 39

2.9 Definitions . . . 40
References . . . 42

3 Manifold Regularization **47**
3.1 Introduction . . . 48
3.2 Related Work . . . 48
3.3 The Semi Supervised Setting . . . 49
3.4 A Framework for Semi-Supervised Learning . . . 49
3.5 Analysis of the Framework . . . 50
3.5.1 Sample Complexity Bounds . . . 51
3.5.2 Comparison to the Supervised Solution . . . 54
3.5.3 The Limits of Manifold Regularization . . . 55
3.6 Rademacher Complexity of Manifold Regularization . . . 55
3.7 Experiment: Concentric circles . . . 56
3.8 Discussion and Conclusion . . . 57
References . . . 59

4 A Soft-Labeled Self-Training Approach **61**
4.1 Introduction . . . 62
4.2 Preliminaries . . . 63
4.3 The Expectation Minimization Framework . . . 63
4.3.1 The Choice of Probability . . . 63
4.3.2 The Semi-Supervised Solution & Related Work . . . 64
4.3.3 An Alternative View . . . 64
4.3.4 A More Flexible Approach . . . 65
4.3.5 Least Squares Classification . . . 66
4.3.6 Nearest Mean Classification . . . 66
4.4 Experiments . . . 66
4.4.1 Controlled Setting . . . 66
4.4.2 Real World Data . . . 67
4.5 Results . . . 67
4.5.1 Controlled Setting . . . 68
4.5.2 Real World Data . . . 68
4.6 Conclusion . . . 69
References . . . 72

5 Posterior Estimation **75**
5.1 Introduction . . . 76
5.2 Related Work . . . 77
5.3 Preliminaries . . . 78
5.3.1 Proper Scoring Rules . . . 78
5.3.2 Link Functions . . . 79
5.3.3 Degenerate Link Functions . . . 79

5.4 Behavior of Proper Composite Losses . . . 79
5.5 Analysis of Loss Functions . . . 82
5.6 Convergence of the Estimator . . . 83
5.6.1 Using the True Risk Minimizer for Estimation . . . 83
5.6.2 Using the Empirical Risk Minimizer for Estimation . . . 84
5.6.3 Misspecification . . . 87
5.6.4 Rate of Convergence . . . 87
5.6.5 Squared Loss vs Squared Hinge Loss . . . 88
5.7 Discussion and Conclusion . . . 89
References . . . 90

6 Open Problem: Monotonicity of Learning **93**
6.1 Introduction. . . . 94
6.2 Preliminaries and Related Work. . . . 94
6.3 The Monotonicity Property . . . 94
6.4 Examples. . . . 95
6.5 Relation to Learnability . . . 97
6.6 Open problem(s) . . . 97
References . . . 99

7 Conclusion **101**
7.1 Further Work Using Causal Knowledge . . . 101
7.2 Implications of Chapter 3 . . . 102
7.3 Extensions of Chapter 3 . . . 103
7.4 Semi-Supervised Learning and Class Probability Estimates . . . 103
7.4.1 Finding Class Probability Estimates via Classification . . . 103
7.4.2 A Simple Idea . . . 103
7.4.3 An Impossibility Result . . . 104
7.4.4 Adding Prior Knowledge . . . 104
7.4.5 Adding Prior Knowledge, But Methodically . . . 105
7.5 Safe Semi-Supervised Learning . . . 105
7.6 Extensions of Chapter 6 . . . 105
7.7 Current Trends in Semi-Supervised Learning . . . 106
7.8 Final Remarks . . . 106
References . . . 107

Acknowledgements **109**

A Appendix **111**
A.1 EM with Generative Models . . . 111
A.2 EM with Discriminative Models . . . 112
A.3 EM Fails with Discriminative Models . . . 112

Curriculum Vitæ **115**

List of Publications **117**

SUMMARY

The goal of this thesis is to investigate theoretical results in the field of semi-supervised learning, while also linking them to problems in related subjects as class probability estimation.

As it is known that semi-supervised methods can decrease the performance compared to supervised methods, the thesis starts by answering the following related questions. What can one guarantee about the performance of semi-supervised learners, and of what kind of type are those guarantees? What assumptions do different methods use and how do they relate? What are the open questions in the field? We answer those questions in Chapter 2 along, and with the help of, an overview of the field. In the discussion of Chapter 2 we elaborate on two open questions that we believe are important to investigate in the future. First, most semi-supervised learning methods are based on assumptions. Can we use those methods effectively in cases where we a priori do not know if the assumption is true or not? Second, some impossibility results show that semi-supervised learners can outperform supervised methods by at most a constant in terms of sample complexity. But, how important can those constants be in practice?

We find a partial answer to the latter question in Chapter 3. The original motivation for the third chapter comes from a different question though: What are the theoretical guarantees of manifold regularization? This question was triggered by the fact that on the one hand manifold regularization is well motivated and widely known in the field, but on the other hand there were no sample complexity bounds for this method prior to this work, to the best of our knowledge. This was in particular surprising, as the method itself is a kernel method and has thus a rich framework to draw from. We discuss two complexity analyses, one based on the notion of pseudo-dimension, which can be seen as an extension of the Vapnik-Chervonenkis dimension to real valued function classes, and the other based on Rademacher complexities. The pseudo-dimension dimension analysis reveals a setting in which manifold regularization can offer, up to logarithmic factors, only a constant improvement over its supervised counterpart, so it essentially obeys an impossibility result that we discuss in Chapter 2. We then present a computationally feasible method to derive an upper bound on the Rademacher complexity for manifold regularization. This potentially also has practical implications, as we speculate that the Rademacher complexity can be useful to choose an adequate hyperparameter for the regularization term in the method, when labeled data is very sparse. Finally we come back to the question of how good constant improvements can be in practice. In the discussion of our review we show, with the help of the findings of Chapter 3, that the constant can be arbitrarily large.

In Chapter 4 we propose a novel method of self-learning. This project took place during the early stages of this work and the results can be seen as preliminary. Nevertheless, we show that in a self-learning setting it can be beneficial to use soft-labels[1] over hard labels.

[1] Soft-labels can be thought in this context of the probability that an object belongs to a certain class.

In the most simple version, self-learning adds the unlabeled data, together with labels that come from the prediction of a previously trained model, to the training set. A new model is then trained with this enriched training set, and the procedure may be iterated. More complex versions only add the unlabeled data on which the model has a high confidence in the label prediction. We propose a version of self-learning, where one adds directly all of the unlabeled data, but takes the confidences, in form of the soft-labels, into account. This leads to a method that can be seen as a generalization of the renowned expectation-maximization algorithm, and we show that this method performs better on many datasets than the standard procedure with hard labels. The work is nevertheless preliminary in the sense that Chapter 5 throws a new light on how to choose the soft-labels, and it is also not yet clear how our method compares to other, more sophisticated versions, of self-learning. In the discussion we elaborate, however, that an extension of our method can lead to a theoretically well motivated version of self-learning. It would be theoretically well motivated in the sense, that we can precisely state what the assumptions of the method are.

We then move in Chapter 5 to the topic of estimating class probabilities $P(Y \mid X)$ with classification methods. As we were working mostly with discriminative binary classification methods, for example support vector machines, we ask the more precise question if one can retrieve class probability estimates with those methods. We answer this question for different loss functions embedded in an empirical risk minimization method. We show that the squared loss, squared hinge loss and the logistic loss are suitable for class probability estimation, while the hinge loss is not. Furthermore, we derive point-wise L1-convergence rates for the estimate. In addition, we point out that the squared loss can be easily used the wrong way, something that we believe many people are not aware of. This chapter of the thesis also opens new possibilities to investigate class probability estimation with asymmetric loss functions.

In Chapter 6 we ask a fundamental question about supervised learning, which was triggered by problems we observe in semi-supervised learning: Semi-supervised learning sometimes degrades performance, so can we come up with methods that guarantee that adding unlabeled data will improve the performance? We decided, however, to take a step back and try to answer the question if we can give those guarantees when we add *labeled* data. We came to the surprising conclusion, that we cannot guarantee monotonic improvement without further assumptions, even not in expectation over the sampling process. In particular, we design in Chapter 6 a simple regression example where adding labeled data degrades the performance.

In Chapter 7 we conclude this thesis and discuss the relations between the chapters. We start by discussing our analysis of manifold regularization from Chapter 3 in view of our review from Chapter 2. We then connect Chapters 4 and 5, and present a potential extension of the method proposed in Chapter 4. Finally we discuss the relation between the open problem presented in Chapter 6 and semi-supervised learning and how one can interpret this thesis in the view of current trends in semi-supervised learning.

Overall, this thesis investigates existing literature on semi-supervised learning, adds new insights to it, unravels a few open problems and formalizes the possibility of class probability estimation which can be used in semi-supervised learning methods and many other applications.

SAMENVATTING

Het doel van dit proefschrift is om theoretische resultaten in het veld van semi-supervised learning te onderzoeken en tegelijkertijd deze resultaten te verbinden aan gerelateerde onderwerpen zoals klasse posterior schatting.

Omdat bekend is dat semi-supervised methodes de prestaties kunnen verminderen ten opzichte van supervised methodes, begint dit proefschrift met het beantwoorden van de volgende vragen: ten eerste, wat kan men garanderen over de prestaties van semi-supervised methodes en van welke aard zijn deze garanties? Ten tweede, welke aannames maken verschillende methodes en hoe verhouden deze zich? Ten derde, wat zijn de open vragen in het vakgebied? Dit proefschrift beantwoordt deze vragen in Hoofdstuk 2 en geeft een overzicht van de belangrijke werken in het vakgebied. We hopen dat het verzamelen en bestuderen van bestaande resultaten, zoals we dat hebben gedaan in dit hoofdstuk, zal leiden tot een strommversnelling en stimulering van onderzoek in dit vakgebied. In Hoofdstuk 2 gaan we dieper in op twee open vragen die belangrijk zijn voor toekomstig onderzoek. Ten eerste: zijn de meeste semi-supervised methodes gebaseerd op aannames? Kunnen we deze methodes effectief gebruiken in gevallen waarin we a priori niet weten of de aanname waar zijn of niet? En ten tweede; sommige onmogelijkheidsresultaten tonen aan dat semi-supervised methodes de supervised methodes kunnen overtreffen met hoogstens een constante in termen van het aantal benodigde objecten voor meen bepaalde error rate. Maar hoe belangrijk kunnen die constantes in de praktijk eigenlijk zijn?

We geven een gedeeltelijk antwoord op de eerste vraag in Hoofdstuk 3. De oorspronkelijke motivatie voor het derde hoofdstuk komt van een andere vraag: wat zijn de theoretische garanties van variëteitsregularisatie? Deze vraag werd kwam op omdat variëteitsregularisatie goed gemotiveerd en algemeen bekend is in het vakgebied, maar er aan de andere kant geen theoretische garanties voor deze methode bestaan wat betreft de fout van de schatters. Dit was verrassend, omdat de methode zelf een kernelmethode is en dus een rijk theoretisch kader heeft om uit te putten. We bespreken twee complexiteitsanalyses, ‘e’en op basis van de notie van de pseudodimensie en de andere op basis van de Rademacher-complexiteit. De analyse van de pseudodimensie laat zien dat variëteitsregularisatie, ten opzichte van begeleide methoden en op logaritmische factoren na, slechts een constante verbetering kan bieden. Daarmee valt deze bevinding feitelijk binnen het onmogelijkheidsresultaat dat we in Hoofdstuk 2 bespreken. Vervolgens presenteren we een voor de computer berekenbare methode om een bovengrens af te leiden voor de Rademacher complexiteit voor variëteitsregularisatie. Dit heeft mogelijk een praktische toepassing, omdat we de Rademacher complexiteit nuttig kan zijn om een geschikte hyperparameter te vinden voor de regularisatieterm in de methode wanneer gelabelde data zeer schaars is. Ten slotte komen we terug op de vraag hoe goed de constante verbeteringen in de praktijk kunnen zijn. In de bespreking van ons overzicht laten we met behulp van de bevindingen van Hoofdstuk 3 zien dat deze constantes willekeurig groot kunnen zijn.

In Hoofdstuk 4 introduceren we een nieuwe self-learning methode voor. Dit project

vond plaats aan het begin van de promotie en de resultaten moeten als voorlopig worden beschouwd. Met de huidige kennis zouden we enkele keuzes anders hebben gemaakt. Desalniettemin laten we zien dat het nuttig kan zijn om soft-labels[2] te gebruiken in plaats van harde labels voor self-learning. In de meest eenvoudige versie neemt self-learning voor niet-gelabelde gegevens de labels over van de voorspellingen van een eerder getraind model. Een nieuw model wordt vervolgens getraind met deze verrijkte trainingsset en de procedure kan meerdere malen worden herhaald. Meer geavanceerde versies voegen alleen de niet-gelabelde gegevens toe waarvan het model een hoog vertrouwen in de voorspelling heeft. We stellen een versie van self-learning voor die rekening houdt met de zekerheid van de voorspelde labels in de vorm van de soft-labels. Dit leidt tot een methode die kan worden gezien als een generalisatie van het expectation-maximization algoritme. We laten zien dat deze methode in veel situaties betere prestaties levert dan de standaardprocedure met harde labels. Het werk is niettemin voorlopig in de zin dat Hoofdstuk 5 een nieuw licht werpt op hoe de soft-labels het beste gekozen kunnen worden. Daarnaast is het nog niet duidelijk hoe onze methode zich verhoudt tot andere, meer geavanceerde versies van self-learning. In de discussie lichten we echter toe dat een uitbreiding van onze methode kan leiden tot een theoretisch goed gemotiveerde versie van self-learning. Het zou theoretisch goed gemotiveerd zijn in de zin dat we precies kunnen aangeven wat de aannames van de methode zijn die nodig zijn voor succesvol leren.

In Hoofdstuk 5 bespreken we het schatten van kansdichtheden, zoals het schatten van de posterior $P(Y \mid X)$. We werken voornamelijk met discriminative binaire classificatiemethoden zoals de support vector machine. Voor zulke modellen onderzoeken we de vraag of deze een schatting kunnen maken van $P(Y \mid X)$. We beantwoorden deze vraag voor verschillende loss-functies voor Empirical Risk Minimization (ERM). We laten zien dat kwadratische loss, kwadraat-hinge-loss en logistic loss geschikt zijn voor het schatten van de posterior, terwijl kwadraat-hinge-loss dat niet is. Voor praktische doeleinden wijzen we erop dat het kwadratische loss gemakkelijk op de verkeerde manier kan worden gebruikt. Waarschijnlijk zijn veel mensen zich hier niet van bewust. Dit hoofdstuk van het proefschrift opent nieuwe mogelijkheden voor het onderzoeken van het schatten van $P(Y \mid X)$ met asymmetrische loss-functies.

In Hoofdstuk 6 stellen we een fundamentele vraag over supervised learning. Deze vraag kwam op omdat in semi-supervised learning soms meer ongelabelde data de prestaties verslechtert. Dat leidde ons tot de vraag of we kunnen garanderen dat het toevoegen van ongelabelde gegevens de prestaties zal verbeteren? Deze vraag bleek erg lastig. We hebben toen besloten een stap terug te doen en we hebben geprobeerd de volgende simpelere vraag te beantwoorden: kunnen we een garantie geven dat de prestaties verbeteren als we *gelabelde* gegevens toevoegen voor een supervised methode? We kwamen tot de conclusie dat we geen monotone verbetering kunnen garanderen zonder verdere aannames, zelfs niet in verwachting over de trainingsdata. In het bijzonder ontwerpen we in Hoofdstuk 6 een eenvoudig regressievoorbeeld waarbij het toevoegen van gelabelde gegevens de prestaties verslechtert.

In Hoofdstuk 7 sluiten we dit proefschrift af en bespreken we de relaties tussen de hoofdstukken. We beginnen met het bespreken van onze analyse van variëteitsregularisatie

[2] Soft-labels kunnen in deze context worden gezien als de waarschijnlijkheid dat een object tot een bepaalde klasse behoort (oftewel de posterior).

uit Hoofdstuk 3 met het oog op ons overzicht van Hoofdstuk 2. We verbinden vervolgens de Hoofdstukken 4 en 5, en presenteren een mogelijke uitbreiding van de methode voorgesteld in Hoofdstuk 4. Dan bespreken we de relatie tussen het open probleem gepresenteerd in Hoofdstuk 6 en semi-supervised learning. We sluiten af met hoe men dit proefschrift kan interpreteren in het licht van de huidige trends in semi-supervised learning.

In het kort; dit proefschrift vat de bestaande literatuur samen, geeft daarin nieuwe inzichten en formaliseert het schatten van de posterior, wat onder andere toepassingen kan vinden in semi-supervised learning en andere toepassingen.

1

INTRODUCTION

This chapter introduces the concept of semi-supervised learning. The introduction will be brief and informal, as the chapter thereafter is a survey of theoretical results in the semi-supervised learning literature. We introduce the basic idea of learning in general, motivate the utility of unlabeled data, identify potential problems and finally give an outline of the rest of this thesis.

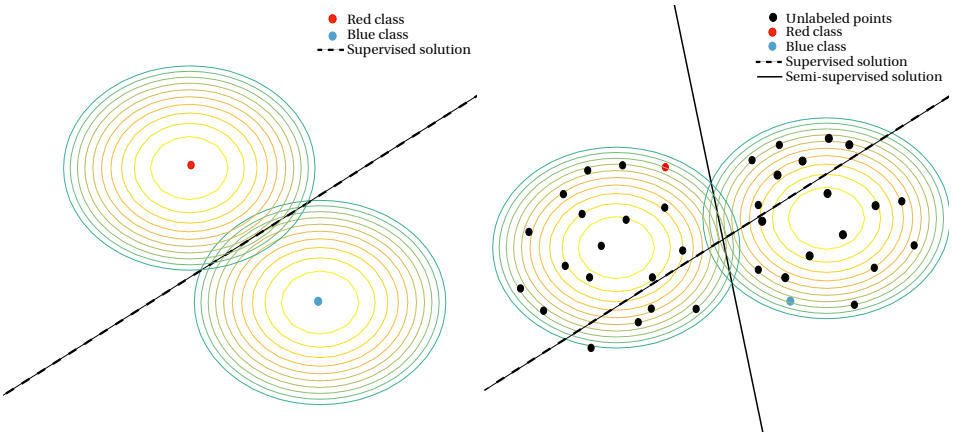
1

1.1. LEARNING FROM DATA

The core question of machine learning and pattern recognition is *how* one can learn from past experience. On a even more fundamental note we start with the question *if* we can learn from the past experience, and thus machine learning is a part of inductive reasoning [1]. The essential idea of this reasoning is that we collect evidential support for a hypothesis. If I walk past a dog every day and it does not bite me, how high is the chance it will bite me tomorrow? If I saw 42 white swans and never any black, how big is the chance that the next swan I see is black? These observations at least seems to support the hypothesis that the dog does not bite, and all swans are white, although those are certainly no guarantees. For this type of reasoning to work, the world cannot be chaotic. A common assumption for machine learning is that our data comes from a certain distribution, which in a way guarantees that a dog does tomorrow not suddenly look like a octopus, and so there is some order. If a dog does tomorrow still look like a dog, then it can be possible to learn from previously collected data, i.e. observations. In this thesis this assumption will be captured in a statistical framework, where one assumes that the data is collected identically and independently from a stationary distribution. Although every actual machine learning method is based on inductive reasoning, most of this thesis is concerned with its counterpart, namely deductive reasoning. Mathematical reasoning is in nature deductive. Instead of gathering evidence for a claim, one starts with (not necessarily proven) premises, and deducts from multiple premises new conclusions. For example: Swans are always black. Klaus is a swan. Klaus is therefore black. This is a valid deduction, even though the actual content of the premises can be wrong. This thesis focuses on theoretical possibilities and impossibilities of various learning scenarios, with speculations on real world impact. While our results are always valid, inductive inference will still be needed to test the premises and to decide if our result is relevant in specific scenarios, or not.

1.2. WHY SEMI-SUPERVISED LEARNING?

Data is at the core of every machine learning method, and for a classification task this data carries a label, like the picture of a dog comes along with the information that there is indeed a dog in the picture. But someone has to annotate the picture and this costs time. In particular the very successful deep neural networks often rely on a large amount of these annotated training examples to come to a good performance [2]. This can be problematic for tasks where labeling data costs a lot of resources as money or time, as for example the annotation of social actions in video data [3]. The problem of not having sufficient training examples does extend to other scenarios, for example when you need an expert to label the data. This is in particular the case in medical settings, as, for instance, in medical image analysis. For that reason one wishes in many settings to reduce the amount of labeled data one needs to train a machine with good performance. Semi-supervised learning offers a possibility to do so, i.e. to reduce the amount of labeled data needed for machine learning. The idea is to guide the learning process with *unlabeled* data. As described above, the bottleneck of labeled data is often the labeling process, while gathering unlabeled data can be easy. As an additional example, think about the task of classifying documents into a finite set of different topics. Gathering documents is no problem, the Internet, for example, offers this in abundance. Reading a document and classifying it, on the other hand, takes



(a) The purely supervised case. We train a Gaussian mixture model based on one point per class.

(b) The semi-supervised cased. To train the Gaussian mixture model we have now also unlabeled data available.

Figure 1.1: The resulting decision boundaries when we train a Gaussian mixture model with two mixture components. Each component is assumed to have equal weight of $\frac{1}{2}$ and we fix the covariance matrix of each component to be uniform.

enormously much more time. From an information theoretical point of view, unlabeled data offers more information about the underlying problem, so why not try to use it?

1.3. How Semi-Supervised Learning?

The first question a reader might ask is: How to use this unlabeled data? One simple answer to this can be given when we think about unlabeled data in the following way. Let us assume that our objects x, for example the documents, come from a set $\mathcal{X}$. The label y, in the previous example the type of the document, belongs to a set $\mathcal{Y}$. It is often assumed that we draw training examples from a distribution P over $\mathcal{X} \times \mathcal{Y}$. A possible way of training a machine is to try to model the distribution P with a family of distributions $p(x, y \mid \theta)$, which is parametrized with a parameter $\theta \in \Theta$. In this setting, gathering labeled data corresponds to gathering information about the full distribution $P(X, Y)$, while we gather information about the marginal distribution $P(X) := P(X, Y \in \mathcal{Y})$ with unlabeled data. We can then try to find a model from $p(x, y \mid \theta)$ that fits the labeled *and* the unlabeled data well. The expectation-maximization method for example is one way to do that [4]. The next section illustrates this method on a simple example. There are of course many more methods, some of which we will cover in the remainder of this thesis.

1.4. Challenges in Semi-Supervised Learning

Consider a scenario where we want to train a two class classification method. More precisely, we assume we observe objects $x \in \mathbb{R}^2$ and for each x we have to decide if it belongs to the red, or the blue class. Figure 1.1 (a) shows two labeled training samples, one from

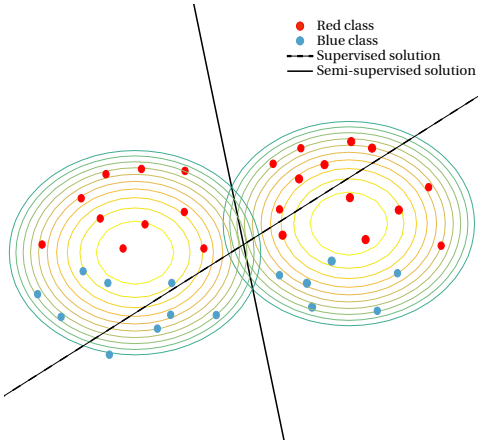


Figure 1.2: The supervised model performs better than the semi-supervised model, because the underlying data does not fit the assumption that each class comes from a Gaussian distribution.

the red and one from the blue class. Figure 1.1 (b) shows in addition unlabeled objects in black, so objects we want to assign to class red or blue. We chose to use a Gaussian mixture model with equal class priors and fixed uniform covariance matrix to do so. As we have labeled and unlabeled data available, we chose to use the expectation-maximization procedure to also make use of the unlabeled data. Figure 1.1 (a) and (b) show the resulting decision boundaries of this model when we use respectively only the labeled data and when we also incorporate the unlabeled data. Assume now further that we reveal the labels of the previously unlabeled data according to Figure 1.2. Comparing the supervised and the semi-supervised solution, we actually observe that the purely supervised found model performs better than the semi-supervised model. This is essentially the case because the semi-supervised model makes wrong assumptions. Informally stated, the model assumes that the data consists out of two clusters, and each cluster belongs to a particular class. The actual data violates this assumption. Now, wrong model assumptions can always happen, but this is in particular a problem in semi-supervised learning. This is because even under model misspecification, we typically assume in supervised learning that if we add more labeled data, we will find a better model nevertheless. In face of Chapter 6, see the next section for details, this might be a bolt statement, but practice shows that it is more often the case than not. One of the big challenges in semi-supervised learning is to try to never be worse than their supervised counterparts. This is because there is a real risk that semi-supervised learning will reduce the performance [5, Chapter 4]. In the beginning of the previous section we asked the question, how to do semi-supervised learning. The above example shows that one of the more fundamental questions might be: Should we use semi-supervised learning at all?

1.5. ORGANIZATION OF THIS THESIS

In the previous sections we pointed out two question about using unlabeled data. How do you use unlabeled data, and should we use it at all? There are a many methods in

semi-supervised learning. Co-training [6], graph based methods [7], EM [4], entropy regularization [8], manifold regularization [9], just to name a few. Theoretical results, on the other hand, are in comparison sparse. We believe, however, that it is in particular in semi-supervised learning of great importance to understand the methods, also from a theoretical point of view, as good as possible. Theoretical results can help to set a right expectation on the method, and understand the underlying assumptions. This can ultimately reduce the risk of using semi-supervised learning in the wrong situation and thus degrading the performance.

We start in Chapter 2 with a review of existing theoretical results in semi-supervised learning. The review mostly collects and relates results from the statistical learning theory. In particular, we use the framework of probabilistic approximately correct learning, in short PAC-learning. This rigorous framework analyses the amount of labeled data one needs to obtain solutions that have a small error with a high probability. We then collect results that study the question how much less labeled data one needs when also unlabeled data is available. Besides that we also look at impossibility results, transductive learning and some asymptotic results.

In Chapter 3 we add our own contribution to the existing literature on theoretical results in semi-supervised learning. We analyse a well known semi-supervised technique, manifold regularization [9, 10]. Our analysis focuses, like our review, on the PAC-learning framework. We use and extend existing literature to derive learning guarantees using two different complexity notions, the pseudo-dimension and Rademacher complexity [11, Chapter 3]. The essential difference between the two complexity notions is that the pseudo-dimension gives learning guarantees that are independent of the domain distribution, while the Rademacher complexity takes the distribution into account. That is in particular useful in semi-supervised learning, as the unlabeled data effectively contains knowledge about the domain distribution. We then speculate and motivate that the Rademacher complexity can be informative for choosing a suitable hyperparameter for manifold regularization.

In Chapter 4 we propose a novel formulation of self-learning that uses class probability estimates to reweigh the unlabeled samples. We compare this to the most simple version of self-learning, where one adds the unlabeled data together with pseudo-labels to the training set, and show that reweighing can increase the performance on many datasets.

In Chapter 5 we investigate the possibility to retrieve class probability estimates within the framework of empirical risk minimization, as for example with support vector machines. We investigate with which loss functions one can retrieve consistent class probability estimates and what the rate of convergence for finite sample sizes is. To some degree one can consider this work as a standalone project, but our motivation for this investigation was still based on understanding how to learn with unlabeled data and was a follow-up project from the work presented in Chapter 4.

In Chapter 6 we go back to the roots of learning and present a, for us surprising, finding, which lead to a fairly general open question. We show that in a simple regression setting adding labeled samples can actually degrade the performance, even in expectation over the sampling process. This leads to the open question under which circumstances one can guarantee that adding more labeled data will improve the performance.

Chapter 7 concludes the thesis. There we discuss how our findings relate, their impact on the field, open questions and possible extensions of our work.

1

REFERENCES

[1] J. Hawthorne, *Inductive logic,* in *The Stanford Encyclopedia of Philosophy* (Metaphysics Research Lab, Stanford University, 2018) spring 2018 ed.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Imagenet classification with deep convolutional neural networks,* in *Advances in Neural Information Processing Systems 25* (Lake Tahoe, Nevada, USA, 2012) pp. 1097–1105.

[3] E. Gedik and H. Hung, *Personalised models for speech detection from body movements using transductive parameter transfer,* Personal and Ubiquitous Computing **21**, 723 (2017).

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood from incomplete data via the EM algorithm,* Journal of the Royal Statistical Society, Series B **39**, 1 (1977).

[5] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning* (The MIT Press, Cambridge, MA, USA, 2010).

[6] A. Blum and T. Mitchell, *Combining labeled and unlabeled data with co-training,* in *Proceedings of the 11th Annual Conference on Computational Learning Theory* (Madison, Wisconsin, USA, 1998) pp. 92–100.

[7] R. Johnson and T. Zhang, *Graph-based semi-supervised learning and spectral kernel design,* IEEE Transactions of Information Theory **54**, 275 (2008).

[8] Y. Grandvalet and Y. Bengio, *Semi-supervised learning by entropy minimization,* in *Advances in Neural Information Processing Systems 17* (The MIT Press, Cambridge, MA, USA, 2005) pp. 529–536.

[9] M. Belkin, P. Niyogi, and V. Sindhwani, *Manifold regularization: A geometric framework for learning from labeled and unlabeled examples,* Journal of Machine Learning Research **7**, 2399 (2006).

[10] V. Sindhwani, P. Niyogi, and M. Belkin, *Beyond the point cloud: From transductive to semi-supervised learning,* in *Proceedings of the 22nd International Conference on Machine Learning* (Bonn, Germany, 2005) pp. 824–831.

[11] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning* (The MIT Press, Cambridge, MA, USA, 2012).

2

A Review of Theoretical Results

Semi-supervised learning is a setting where one has labeled and unlabeled data available. In this chapter we explore different types of theoretical results when one uses unlabeled data in classification and regression tasks. Most methods that use unlabeled data rely on certain assumptions about the data distribution. When those assumptions are not met in reality, including unlabeled data may actually decrease performance. Studying such methods, it therefore is particularly important to have an understanding of the underlying theory. In this review we gather results about the possible gains one can achieve when using semi-supervised learning as well as results about the limits of such methods. More precisely, this review collects the answers to the following questions: What are, in terms of improving supervised methods, the limits of semi-supervised learning? What are the assumptions of different methods? What can we achieve if the assumptions are true? Finally, we also discuss the biggest bottleneck of semi-supervised learning, namely the assumptions they make.

2.1. Introduction and Scope

In various applications gathering unlabeled data is easier, faster and/or cheaper than gathering labeled data. The goal of semi-supervised learning (SSL)[1] is to combine unlabeled and labeled data to design classification or regression rules that outperform schemes that are only based on labeled data. SSL does come, however, with an inherent risk. It is well-known that including unlabeled data can degrade the performance [1, 2]. Studying and understanding SSL from a theoretical point of view allows us to exactly formulate the assumptions we need and the improvements we can expect, as well as the limitations of said methods. With this one can formulate recommendations for using SSL with the aim of avoiding a decrease in performance as good as possible. In this review, we collect and present theoretical results concerning SSL, study the relevant papers in detail, present their main result and point out connections to other works.

This review targets two groups of audience. The first group we target are interested practitioners and researchers working on experimental SSL. While they may not be interested in all the details we present, we believe that the introduction in each of our sections gives a good high level understanding of the types of theoretical results in SSL and the main insights they provoke. The second target audience is everyone working on the theoretical side of SSL. We hope that, especially researchers starting in this field, can find inspiration and connections to their own work in our overview. We mostly present results that describe the performance of semi-supervised learners, often, but not exclusively, in the language of the PAC-learning framework.[2] We interpret the results, draw connections between them and point out what one has to assume for them to be valid. Next to theoretical guarantees of some specific SSL we also present results on the limits of SSL.

2.1.1. Outline

In the next section we introduce the formal learning framework which is also assumed for the majority of the work we present. In Section 2.3 we present results on the limits of SSL, which often arise due to specific assumptions on the model or the data generation process. Opposing to the settings where the improvements of SSL are provably limited, we present in the same section three settings where the improvements of SSL are *unlimited*. With unlimited we mean here that a SSL can PAC-learn the problem, while no supervised learner (SL) can. In Section 2.4 we investigate methods that try to exploit unlabeled data, without having further assumptions on the data distribution. In Section 2.5 we present semi-supervised learners that make *weak* assumptions on the data distribution. Those assumptions are weak in the sense that the resulting learner cannot get a learning rate faster than the standard learning rate of $\frac{1}{\sqrt{n}}$,[3] where n is the number of labeled samples. The improvements are

[1] We overload the abbreviation of SSL to stand either for *semi-supervised learning* or *semi-supervised learner*.

[2] PAC-learning stands for *Probabilistically Approximately Correct*-learning. In this framework one can study how far a trained classifier is off of the best classifier from a given class, given a certain amount of labeled data. The rate at which we approach the best classifier is called learning rate. Nice introductions to this framework can be found in [3] and [4]. We also refer to Definition 1, where we introduce the notion of sample complexity. PAC-learnable means that the sample complexity is always finite.

[3] The learning rate is the rate in which we converge to the best classifier from a given class in number of the labeled samples. That the standard rate is in order of $\frac{1}{\sqrt{n}}$ follows from classic results as shown for example by Vapnik [5].

instead given by a constant. In Section 2.6 we present learners that use *strong* assumptions under which one can converge exponentially fast to the best classifier in a given class, i.e. the learning rate is in order of e^{-n}. In Section 2.7 we present results in the transductive setting, a setting where one is only interested in the labels of the unlabeled data. In the same section we also present a line of research that tries to construct semi-supervised learners that are never worse than their supervised counterparts. In Section 2.8 we discuss the overall results and point out what the current challenges in the field are. In Section 2.8.4 we furthermore explain in more detail what is formally meant by using assumptions in SSL and the problems that occur with that.

2.2. PRELIMINARIES

Unless further specified all results are presented in the standard statistical learning framework. This means that we are given a feature space $\mathcal{X}$ and a label space $\mathcal{Y}$ together with an unknown distribution P on $\mathcal{X} \times \mathcal{Y}$. Overloading the notation we write $P(X)$ and $P(Y)$ for the marginal distributions on $\mathcal{X}$ and $\mathcal{Y}$ and similar for conditional distributions. We observe a labeled n-sample $S_n = ((x_1, y_1), ..., (x_n, y_n))$ and an unlabeled m-sample $U_m = (x_{n+1}, ..., x_{n+m})$, where each (x_i, y_i) for $1 \leq i \leq n$ and each x_j for $n+1 \leq j \leq n+m$ is identically and independently distributed according to P. One then choses a hypothesis class H, where each $h \in H$ is a mapping $h : \mathcal{X} \to \mathcal{Y}$, and a loss function $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. Unless specified otherwise we assume for classification that $\mathcal{Y} = \{-1, +1\}$ and the loss is the 0-1 loss, $l(y, \hat{y}) = I_{\{y \neq \hat{y}\}}$. In the regression task we assume that $\mathcal{Y} = \mathbb{R}$ and $l(y, \hat{y}) = (y - \hat{y})^2$. Based on the n labeled and m unlabeled samples we then try to find a $h \in H$ such that the risk $R(h) := \mathbb{E}_{X,Y}[l(h(X), Y)]$ is small. Finally, whenever we have any quantity A that depends on the distribution P, we write $\hat{A}$ for a empirically estimated version of A. For example, given a labeled sample S_n we write $\hat{R}(h) = \frac{1}{n}\sum_{i=1}^{n} l(h(x_i), y_i)$ for the empirical risk of $h \in H$ measured on S_n. If not clear from context we will clarify on which sample we measure. In Table 2.1 on page 41 we present a complete list of the notation we use.

2.3. POSSIBILITY & IMPOSSIBILITY OF SEMI-SUPERVISED LEARNING

In SSL we want to use information about the distribution on $\mathcal{X}$ to improve learning, but it is not necessarily clear that this information can be useful at all. Some authors formalize this idea and then present situations where unlabeled data can help or where it cannot. This section follows the same division. In Subsection 2.3.1 we present different settings where authors could show that unlabeled data cannot help, while In Subsection 2.3.2 we present three specific settings where unlabeled data can give unlimited improvements. By unlimited we mean that no supervised learner can PAC learn in those settings while a semi-supervised learner can.

The negative results often assert an independence between the posterior probability $P(Y \mid X)$ and the marginal distribution $P(X)$. This does, however, not directly mean that unlabeled data is useless, as we are usually not only interested in $P(Y \mid X)$ but on the complete risk of a classifier h, $\mathbb{E}_{X,Y}[l(h(X), Y)]$, which *does* depend on $P(X)$ [6, 5.1.2]. In Section 2.4.1 and 2.4.2, for example,we present work that show risk improvements even

when $P(Y \mid X)$ and $P(X)$ are independent.

2.3.1. Impossibility Results

Impossibility Because of the Data Generation Process

Seeger [7] looks at a simple data generation model and investigates how prior information about the data distribution changes our posterior belief about the model if the prior information is included in a Bayesian fashion. To use the Bayesian approach, the data is assumed to be generated in the following manner. We assume now that the distribution P comes from a model class with parameters μ and θ. First values $\mu \sim P_\mu$ and $\theta \sim P_\theta$ are sampled independently and then the data is generated by gathering samples $x \sim P(X \mid \mu)$ with corresponding labels $y \sim P(Y \mid X, \theta)$ as shown in Figure 2.1. The goal in this setting is to infer θ from a finite labeled sample $S_n = (x_i, y_i)_{1 \leq i \leq n}$. Using a Bayesian approach it can be easily shown that $P(\theta \mid S_n)$ is independent of any finite unlabeled sample and μ itself. In other words: Unlabeled information does not change the posterior belief about θ given the labeled data S_n. A possible solution presented is to assume a dependency between μ and θ, so drawing an additional arrow between μ and θ in Figure 2.1.

Impossibility Because of The Model Assumptions

Hansen [8] investigates when unlabeled data should change our posterior belief about a model. In comparison to [7] no data generation assumptions are made, but rather assumptions about the model we use. He looks at solutions derived from the expected squared loss between this given model and the true desired label output. Splitting the joint distribution $P(X, Y \mid \theta)$ of our model as $P(X, Y \mid \theta) = P(Y \mid X, \theta_1, \theta_2) P(X \mid \theta_2, \theta_3)$ he concludes that unlabeled data can be discarded if θ_2, the shared parameter between the label and marginal distribution, is empty.

Earlier work by Zhang and Oles [9] distinguishes the same type of models, but the impossibility is about the asymptotic efficiency of semi-supervised classifiers. The paper as well considers two types of joint probability models:

1. Parametric: $P(X, Y \mid \alpha) = P(X \mid \alpha) P(Y \mid X, \alpha)$
2. Semi-Parametric: $P(X, Y \mid \alpha) = P(X) P(Y \mid X, \alpha)$

One can show that the Fisher information $I(\hat{\alpha})_{\text{unlabeled + labeled}}$ of an MLE estimator $\hat{\alpha}$ that takes labeled an unlabeled data into account can be decomposed as $I(\hat{\alpha})_{\text{unlabeled +labeled}} = I(\hat{\alpha})_{\text{unlabeled}} + I(\hat{\alpha})_{\text{labeled}}$. So, as long as unlabeled data is available, the Fisher information of the semi-supervised learner is bigger compared to the supervised learner, which is shown to have a Fisher information given by $I(\hat{\alpha})_{\text{labeled}}$. It follows that the SSL is asymptotically

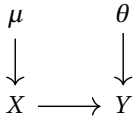


Figure 2.1: The data generation process used in the analysis of Seeger.

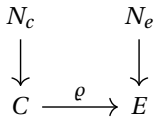


Figure 2.2: Simple functional causal model used by Schölkopf *et al.* [10]. The effect E is caused by C given a deterministic mapping ϱ. Both E and C are influenced by a noise variables N_E and N_C.

more efficient, although not necessarily strictly. In the parametric case we observe that $I(\hat{\alpha})_{\text{unlabeled}} = 0$ and the semi-supervised and supervised estimator have the same asymptotic behavior. In Section 2.4.1 we will present a method that allows for asymptotic efficiency of a SSL even when using a discriminative model $P(Y \mid X, \alpha)$.

IMPOSSIBILITY BECAUSE OF THE CAUSAL DIRECTION

Schölkopf *et al.* [10, Sections 2 and 3] analyze a functional causal model shown as in Figure 2.2. They analyze different learning scenarios under the assumption that the label is the cause C and the feature is the effect E and vise versa. This model introduces an asymmetry in cause and effect, since it leads to the fact that $P(C)$ and $P(E \mid C)$ are independent, while $P(E)$ and $P(C \mid E)$ are not independent. Assuming now that X is the cause of the label Y, we find that the prediction $P(Y \mid X)$ is algorithmically independent of newly gained information about $P(X)$. The situation changes though if we assume that the label Y was caused by X. One problem with this is, that we do not necessarily know if the feature is a cause or an effect. But for example in medical settings this might not be too difficult, as we can identify causal features as those that do actually cause an illness, while effect features are the symptoms of an illness. The work of von Kügelgen *et al.* [11] uses this knowledge to derive a SSL method which only takes the unlabeled data of effect features into account.

IMPOSSIBILITY TO ALWAYS OUTPERFORM A SUPERVISED LEARNER

Inspired by a successful minimax approach for a *generative* linear discriminant model of Loog (see Section 2.7.2), Krijthe and Loog [13] investigate a similar approach to find semi-supervised solutions for *discriminative* models that are never worse than their supervised counterparts. They use a setting where the discriminative models are derived with a monotonously decreasing loss function. The setting is also transductive, so where one is only interested in the performance of our model on the unlabeled data U_m, see also Section 2.7. They essentially show that, under some mild conditions, there is always a labeling of the unseen data U_m such that a semi-supervised learner will perform worse on U_m than the supervised solution. In this sense it is impossible to guarantee that the semi-supervised solution will always outperform the supervised solution.

IMPOSSIBILITY IF WE ONLY KNOW THE MANIFOLD

Lafferty and Wasserman [14, Section 3] show that knowledge of the manifold alone, without additional assumption, is not sufficient to outperform a purely supervised learner. They work in a regression setting and extend work of Bickel and Li [15] to show that there is a supervised learner that can adapt to the dimension of the manifold and thus can achieve minimax rates equivalent to a learner that directly works on the lower dimensional manifold.

We note that Lafferty and Wasserman [14] also show that we can essentially achieve faster rates if we also assume a semi-supervised smoothness assumption. We do not cover more details at this point, but offer a qualitatively very similar analysis in Section 2.6.4.

Impossibility If We Don't Have Additional Assumptions

Ben-David *et al.* [1] started a series of investigations by conjecturing that SSL is, in some sense, generally not possible without any assumptions. In particular we assume that a given domain distribution does not restrict the possible labeling functions, similarly to the data generation process in Figure 2.1. They hypothesize that a semi-supervised learner can't have essentially better sample complexity bounds (see Definitions 1 and 2) than a SL, without any additional assumptions at least. This is different from the previous sections, as there are no further restrictions on the model or the data generation process.

In the following two sections we want to illustrate the precise idea of those conjectures, why they do not hold generally and in which scenarios they are true.

We start with the contributions of Ben-David *et al.* [1]. They hypothesize that the worst-case sample complexity for any semi-supervised learner improves over a supervised learner at most by a constant which only depends on the hypothesis class. The first conjecture states that for the realizable case.

Conjecture 1 (Conjecture 4)**.** [4] *For any hypothesis class H, there exists a constant $c(H)$ such that for any domain distribution D on $\mathcal{X}$*

$$\sup_{h \in H} m(H, D_h, \epsilon, \delta) \leq \sup_{h \in H} c(H) m^{\mathrm{SSL}}(H, D_h, \epsilon, \delta), \tag{2.1}$$

for ϵ and δ small enough, where D_h is the distribution on $\mathcal{X} \times \mathcal{Y}$ with marginal distribution D and conditional distribution $D_h(Y = h(x) \mid X = x) = 1$.

The second conjecture states the same for the agnostic case, so where we replace D_h for any arbitrary distribution P.

Conjecture 2 (Conjecture 5)**.** *For any hypothesis class H, there exists a constant $c(H)$ such that for any domain distribution D*

$$\sup_{P \in \mathrm{ext}(D)} m(H, P, \epsilon, \delta) \leq \sup_{P \in \mathrm{ext}(D)} c(H) m^{\mathrm{SSL}}(H, P, \epsilon, \delta), \tag{2.2}$$

for ϵ and δ small enough and where $\mathrm{ext}(D)$ is the set of all distributions P on $\mathcal{X} \times \mathcal{Y}$ such that the marginal distribution fulfills $P(X) = D$.

In other words: The paper conjectures that if we are given a fixed domain distribution, one can always find a labeling function on it such that for this labeling function the sample complexity gap between SL and SSL can only be a constant. The paper proofs these conjectures for smooth distributions on the real line and threshold functions in the realizable case and for threshold functions and unions of intervals in the agnostic case. The sample complexity comparison is by construction a worst case analysis, in cases where the target hypothesis behaves benign we might still get non-constant improvements. We explore those cases in Section 2.6. On another note, one can also ask the question how good

[4] In brackets we note under which name the statement can be found in the original paper.

a constant improvement by itself can already be. We will elaborate on this in the discussion.

The Conjectures 1 and 2 are essentially true in the realizable case when the hypothesis class has finite VC-dimension. Darnstädt *et al.* [16] showed that Conjecture 1, the realizable case, is true with a small alteration: the supervised learner is allowed to be twice as inaccurate and for the finite VC-dimension case we get an additional term of $\log(\frac{1}{\epsilon})$. In Chapter 3 we take this idea, in a certain way, a step further, and we present a setting in which a manifold regularization scheme obeys the limits stated by the conjecture, again up to logarithmic factors, even though in this case the domain distribution carries information about the labeling function. Darnstädt *et al.* [16] prove the following version of Conjecture 1.

Theorem 1 (Theorem 1)**.** *Let H be a hypothesis class such that it contains the constant zero and constant one function. Then for every domain distribution D and every $h \in H$,*

1. *If H is finite then*

$$m(H, D_h, 2\epsilon, \delta) \leq O(\ln|H|)\, m^{\mathrm{SSL}}(H, D_h, \epsilon, \delta). \tag{2.3}$$

2. *If H has finite VC-dimension then*

$$m(H, D_h, 2\epsilon, \delta) \leq O(\mathrm{VC}(H)) \log(\frac{1}{\epsilon})\, m^{\mathrm{SSL}}(H, D_h, \epsilon, \delta). \tag{2.4}$$

First note that this statement holds for all D_h, so in particular if we take the supremum over all $h \in H$ as in Conjecture 1. Golovnev *et al.* [17] show that if the hypothesis class H is given by the projections over $\{0,1\}^d$, there is a set of domain distributions such that any supervised algorithm needs $\Omega(\mathrm{VC}(H))$ as many samples as the semi-supervised counterpart, which has knowledge of the full domain distribution. So in particular Inequality (2.4) is tight up to logarithmic factors. This actually shows that the constant improvement can be arbitrarily good, as we can increase the VC-dimension by increasing the dimension Golovnev *et al.* [17, Proposition 4]. The agnostic version of Theorem 1 is an open problem.

In the case of a hypothesis class with infinite VC-dimension, however, the conjecture ceases to hold, also for the slightly altered formulations. This is essentially the case because we can start with a class that has infinite VC-dimension, and thus cannot be learned by a supervised learner. A semi-supervised learner, however, can restrict this class in a way such that it has finite VC-dimension. This will become clearer in the next section where we collect three different setups in which a semi-supervised learner can PAC-learn, while a supervised learner cannot.[5]

IMPOSSIBILITY IF WE DON'T RESTRICT THE POSSIBLE LABELING FUNCTIONS

Golovnev *et al.* [17] show that if the domain $\mathcal{X}$ is finite and we allow all deterministic labeling functions on it, no semi-supervised learner can improve in the realizable PAC-learning framework even by a constant over a consistent supervised learner. Consistent means here that the learner achieves 0 training error. The supervised learner is, however, to be allowed twice as inaccurate and twice as unsure.

[5] In this context PAC-learnability means that $m(H, \epsilon, \delta)$ is finite for all $\epsilon, \delta > 0$.

Theorem 2 (Theorem 8)**.** *Let $\mathcal{X}$ be a finite domain, and let $H_{\text{all}} = \{0,1\}^{\mathcal{X}}$ be the set of all deterministic binary labeling functions on $\mathcal{X}$. Let A be any consistent supervised learner, P a distribution over $\mathcal{X}$ and $\epsilon, \delta \in (0,1)$. Then*

$$m(A, H_{\text{all}}, P, 2\epsilon, 2\delta) \leq m^{\text{SSL}}(H_{\text{all}}, P, \epsilon, \delta). \tag{2.5}$$

While the more general Theorem 1 states that a semi-supervised can still be better by a constant depending on the hypothesis class, we find that in the previous setting one even loses this advantage.

A similar result can be found for the agnostic case. Theorem 2 of [18] essentially states that Conjecture 2 (the agnostic case), is true for the finite VC-dimension case, if there are no restrictions on the labeling function. The difference is that they consider in an in-expectation and not a high probability framework and there is a condition on the domain distribution D, while Conjecture 2 is formulated to hold for *all* distributions D. This condition is, however, very mild, the essential assumption of the theorem is that there are no restrictions on the labeling function.

The intuition for both of the previous results is the same: If we allow all labeling functions, there is no label information about the support of $\mathcal{X}$ that we did not observe yet. Finding the labels for this part is equally slow for supervised and semi-supervised learners. In the next section we present hypothesis classes on which semi-supervised learners can be effective. Following the previous result, it is not surprising that those classes are carefully chosen.

2.3.2. Proofs about the Possibility of Semi-Supervised Learning

We consider three specific settings in which it can be shown that a SSL can learn, while a SL cannot. We first present the work of Darnstädt *et al.* [16] and Globerson *et al.* [19], these aim to answer Conjectures 1 and 2 covered in the previous subsection. They show that there is a hypothesis class H^* and a collection of domain distributions $\mathcal{D}^*$ such that no supervised learner can learn H^* under the distributions of $\mathcal{D}^*$. Given, however, any $P \in \mathcal{D}^*$, a semi-supervised learner that has access to a finite, but depending on P arbitrarily large, amount of unlabeled data can learn H^* with the same rate of convergence. Next we present the work of Niyogi [20] as it gives the best example to illustrate how a shift from not learnable to learnable is possible when going from SL and SSL.

Proving the Realizable Case with a Discrete Set

Darnstädt *et al.* [16] give the first example that shows that Conjecture 1 does not generally hold. This is captured in the following theorem, and the other results of this section will be very similar.

Theorem 3 (Theorem 2)**.** *There exists a hypothesis class H^* and a family of domain distributions $\mathcal{D}^*$ such that*

1. *For every $D \in \mathcal{D}^*$,*

$$m^{\text{SSL}}(H^*, D, \epsilon, \delta) \leq O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right).$$

2. *For all* $\epsilon < \frac{1}{2}$ *and* $\delta < 1$,

$$m(H^*, \epsilon, \delta) = \sup_{D \in \mathcal{D}^*} m(H^*, D, \epsilon, \delta) = \infty.$$

In order for the SSL to be able to PAC-learn for all $D \in \mathcal{D}^*$ it needs knowledge of the full distribution D. (Although for each fixed $D \in \mathcal{D}^*$ a finite amount of unlabeled data suffices). Since the supervised learner can only collect labeled samples it will never be able to achieve this knowledge with a finite number of samples, and thus has an infinite sample complexity. The construction of H^* and $\mathcal{D}^*$ can be considered rather artificial. We discuss papers that show similar behavior with a hypothesis class which is loosely based on the manifold assumption in the next two subsections. We nevertheless want to give the intuition for the given example, as it, as well as the other examples, use the same trick.

Darnstädt *et al.* [16] set the example up as follows. The domain $\mathcal{X}$ consists of all sequences $x = (x_1, x_2, ..., x_l)$ of arbitrary finite length and $x_i \in \{0, 1\}$. The distributions $D \in \mathcal{D}^*$ on $\mathcal{X}$ are such that there is a sequence $D(x_{\sigma(1)} = 1) > D(x_{\sigma(2)} = 1) > ...$, which drops sufficiently quick[6], where σ is a random permutation on the length of x. The hypothesis class H^* contains all hypotheses h_i with $h_i(x) = x_i$ and the constant 0 hypothesis. Note that although the class has infinite VC-dimension it still takes some effort to show that no supervised learner can learn it w.r.t to all distributions in $\mathcal{D}^*$. This is because the VC-dimension might not be infinite over $\mathcal{D}^*$. We want to sketch how the SSL can learn it. After fixing a $D \in \mathcal{D}^*$ and $\epsilon, \delta > 0$ we draw enough unlabeled samples to identify all positions $i \in \mathbb{N}$ such that x_i is with a high probability 0. For all those indices i we can remove h_i from H^* as the constant 0 hypothesis will be good enough for predicting accurately. They then show that the remaining hypotheses in H^* can be learned from finitely many samples. Note that it is important that the admissible domain distributions are restricted. If D^* would also include distributions that essentially put equal weight on all positions i, unlabeled data could not help to restrict H^*. In short: this example, and also the following, are essentially set up such that H and D have a certain link, and in those cases knowledge about D can actually give knowledge about H. Note, however, that the knowledge about D did not restrict the set of possible labeling functions from H. It was rather that D helped to identify which hypotheses we can safely ignore.

PROVING THE AGNOSTIC CASE USING ALGEBRAIC VARIETIES

Globerson *et al.* [19] provide a different example using a hypothesis class which loosely follows the manifold assumption. Using the same example one can also show that Conjecture 2, so the impossibility conjecture for the agnostic case, is not true in general.

The theorem is very similar to Darnstädt *et al.* [16], the difference is in the construction of the hypothesis set and the set of distributions.

Theorem 4 (Theorem 5)**.** *There exists a hypothesis class* H_{alg} *and a set of distributions* $\mathcal{D}_{\text{alg}}$ *such that.*

1. *For every* $D \in \mathcal{D}_{\text{alg}}$,

$$m^{\text{SSL}}(H_{\text{alg}}, D, \epsilon, \delta) < \frac{2}{\epsilon} \log \frac{2}{\delta}. \tag{2.6}$$

[6] Note that with $x_{\sigma(i)} = 1$ we mean the subset $V \subset \mathcal{X}$ with $V := \{x = (x_1, x_2, ..., x_l) \in \mathcal{X} \mid x_{\sigma(i)} = 1\}$.

2. *The supervised sample complexity is infinite,*

$$\sup_{D \in \mathcal{D}_{\text{alg}}} m(H_{\text{alg}}, D, \epsilon, \delta) = \infty. \tag{2.7}$$

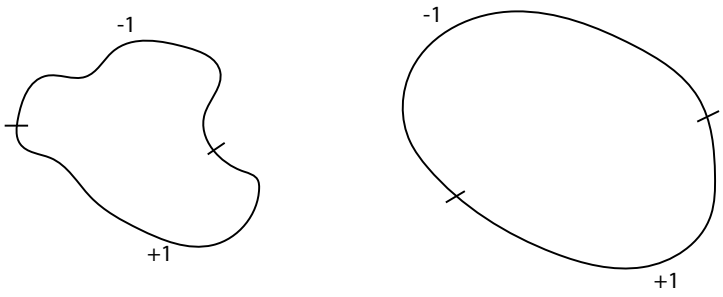
The hypothesis class H_{alg} consists of all hypotheses that have class label 1 on an algebraic set, so essentially a type of manifold, and 0 outside of that algebraic set. This is still a very expressive set with infinite VC dimension. But if we restrict the set of admissible domain distributions $\mathcal{D}_{\text{alg}}$ also to be (a certain type of) algebraic sets, a semi-supervised learner with knowledge of $D \in \mathcal{D}_{\text{alg}}$ can learn efficiently: we can think of $\mathcal{D}_{\text{alg}}$ as the set of distributions that have support on a finite combination of distinguishable algebraic sets $V_1, ..., V_k$. Once we know that the distribution has support on $V_1, ..., V_k$, we only have to figure out which of those algebraic sets have label 1 and which have label 0. A SSL can thus reduce the class H_{alg} by only considering the hypotheses that have class label 1 on combinations from $V_1, ..., V_k$. Since the set of all possible combinations is finite, a SSL can learn them with a sample complexity bounded by Inequality (2.6). Note that although the true labeling function does not have to be part of this restricted set, one can show that it is anyway always optimal to predict with a hypothesis from it. The argument for that is similar to the explanation of the agnostic case below.

The paper also discusses that this argumentation can be extended to the agnostic case, so when the true target function is not in H_{alg}. This extension might appear problematic at first, because the semi-supervised algorithm restricts the hypothesis set H_{alg}, and to guarantee PAC-learnability we need to know that the best predictor from the H_{alg} is still in this restricted set. But this is indeed the case, because the set of domain distributions $\mathcal{D}_{\text{alg}}$ was exactly created for that to hold. To show that, assume that the distribution is supported on an irreducible algebraic set V_0. Our SSL can now chose to label it completely 1 or 0, while both options might lead to non-zero error. But labeling it completely as either 1 or 0 is already ideal, as using any other algebraic set $V_1 \in H_{\text{alg}}$ will lead to one of those two labelings. This is because, by construction, V_1 is either equal to V_0 (which leads to label everything as 1) or has an intersection of zero mass (which leads to labeling almost everything as 0).

This seems to contradict the findings in 2.3.1, as Lafferty and Wasserman [14] show that a supervised learner can also adapt to the underlying manifold. This discrepancy is not easy not analyze as Lafferty and Wasserman [14] work in the regression setting, while Globerson *et al.* [19] analyse classification. The intuition, however, is that Globerson *et al.* [19] present the supervised learner with an impossible, meaning not PAC-learnable, task. Lafferty and Wasserman [14] on the other hand restrict the target functions to be smooth, and thus the supervised learner is presented with a sufficiently easy problem.

Using the Manifold Assumption to Make A Class Learnable

Niyogi [20] provides another setup in which a semi-supervised learner can effectively learn while a supervised learner cannot. The motivation, however, was independent of Ben-David *et al.* [1] and was meant as a general theoretical analysis of the manifold learning framework as introduced in Belkin *et al.* [21]. Also, their results are in-expectation, while the previous papers give PAC bounds, which means that they hold with high probability. Although



(a) (b)

Figure 2.3: The shapes shown in (a) and (b) are two different embeddings of a circle in the Euclidean plane. One half of the circle is labeled as 1, while the other half is labeled as −1, while we assume that everything outside the circle is labeled as 1.

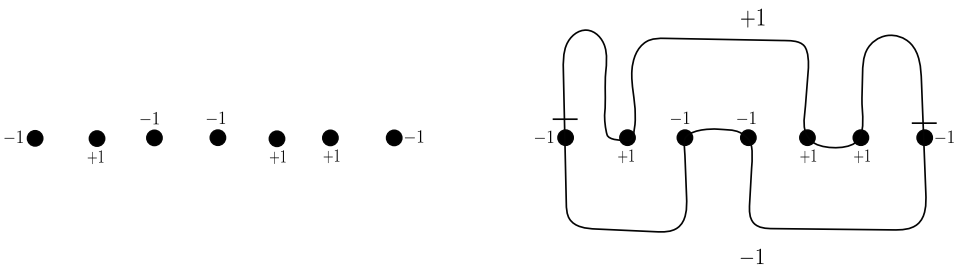
the paper presents the results in an in-expectation framework we slightly alter the setup and present it in the PAC learning framework. We believe this is sufficient to understand the ideas and allows us to draw better connections to the previous papers. Although this work is based on the manifold assumption, so a given domain distribution does limit the possible labeling functions, we believe that it is the most intuitive setting to understand why a supervised learner cannot learn, while a semi-supervised learner can.

The example is built as follows. First it is assumed that the admissible domain distributions are given by the class $\mathcal{P}_c$ which have support on embeddings of a circle in the Euclidean plane, see also Figure 2.3. The hypothesis class H_c consists of all possible binary labelings of half circles, while everything outside the circle is labeled as 1,[7] see also Figure 2.3. The SSL that knows the specific embedding of the circle, only needs to find two thresholds on the given circle, a class with VC-dimension of 2, so the SSL can learn efficiently. In Figure 2.4 we schematically show why H_c has an infinite VC dimension and thus cannot be learned by any supervised learner.

2.4. LEARNING WITHOUT ASSUMPTIONS

As argued in the previous section it can be difficult to use unlabeled data without any additional assumptions, and in some situations one can show that unlabeled data cannot help at all. As already mentioned in the introduction of Section 2.3, this impossibility stems sometimes from the fact that we only consider improvements of the estimate of the conditional probability $P(Y \mid X)$. The work we present in this section looks at the complete risk $\mathbb{E}_{X,Y}[l(h(X), Y)]$, a quantity which is always influenced by the marginal distribution $P(X)$. Furthermore no additional assumptions about the distribution P are made, and the theoretical guarantees are accordingly weak. We first present the work of Sokolovska *et al.* [22] who use the unlabeled data to reweigh the labeled points, and show improvements in terms

[7] The labeling outside of the circle is a formality to ensure that the supervised learner makes predictions for the whole circle, as the learner does not a priori know in which part of the space the circle is embedded.



(a) Assume we are given 7 points that are labeled as depicted above.

(b) The circle above labels the points correctly. The upper half assigns points the label −1, while the lower half labels points as +1.

Figure 2.4: A schematic proof why the hypothesis set H_c has an infinite VC dimension. Given the points in (a) we can label them correctly with the circle given in (b).

of asymptotic efficiency. Interestingly, one needs that the model is misspecified to show this result. Second we present the work of Kääriäinen [23] who uses the unlabeled data to pick the center of the version space. The best possible improvements are bounded by a factor of 2. Finally we present the work of Leskes [24] who uses unlabeled data to combine different hypothesis spaces and shows that the learning rates depend on the highest Rademacher complexity amongst those hypothesis spaces.

2.4.1. Reweighing the Labeled Data By the Marginal Distribution

Sokolovska *et al.* [22] proposed a semi-supervised learner that uses knowledge of the marginal distribution $P(X)$ in a re-weighing scheme. To avoid difficulties for the theoretical analysis they restrict the feature space $\mathcal{X}$ to contain only finitely many points and assume that the SSL has access to the full marginal distribution $P(X)$.[8] They consider models that directly estimate class probabilities $p(y \mid x, \theta)$, while they measure performance by the negative log-likelihood $l(x, y \mid \theta) = -\ln p(y \mid x, \theta)$. They then analyze asymptotic behavior, in particular the asymptotic variance of the model estimation. They compare two models, the classical maximum log-likelihood estimate based on the labeled data only

$$\theta^{\mathrm{SL}} = \operatorname*{argmin}_{\theta \in \Theta} \sum_{(x,y) \in S_n} l(x, y \mid \theta) \tag{2.8}$$

and a semi-supervised learner that also takes the marginal $P(x)$ into account

$$\theta^{\mathrm{SSL}} = \operatorname*{argmin}_{\theta \in \Theta} \sum_{(x,y) \in S_n} \frac{P(x)}{\sum_{z \in X_n} I_{\{x=z\}}} l(x, y \mid \theta). \tag{2.9}$$

Again, note that the semi-supervised learner weighs each feature with the true, instead of the empirical, distribution. Let us first state the results about θ^{SSL} and then discuss them.

[8] The work is continued by Kawakita and Kanamori [25] and extended to non-discrete features spaces.

Theorem 5 (Theorem 1). *Let* $\theta^* \in \arg\min_{\theta\in\Theta} \mathbb{E}[l(x, y \mid \theta)]$ *and define the following matrices*

$$H(\theta^*) = \mathbb{E}_X \left[\mathbb{V}_{Y|X}[\nabla_\theta l(X, Y \mid \theta) \mid X]\right] \tag{2.10}$$

$$I(\theta^*) = \mathbb{E}_{X,Y} \left[\nabla_\theta l(X, Y \mid \theta) \nabla_\theta^T l(X, Y \mid \theta)\right] \tag{2.11}$$

$$J(\theta^*) = \mathbb{E}_{X,Y} \left[\nabla_\theta^T \nabla_\theta l(X, Y \mid \theta)\right], \tag{2.12}$$

where $\mathbb{V}_{Y|X}$ *is the variance over the conditional random variable* $Y \mid X$*. Then* θ^{SL} *and* θ^{SSL} *are consistent and asymptotically normal estimators of* θ^* *with*

$$\sqrt{n}(\theta^{\mathrm{SL}} - \theta^*) \to \mathcal{N}(0, J^{-1}(\theta^*) I(\theta^*) J^{-1}(\theta^*)) \tag{2.13}$$

$$\sqrt{n}(\theta^{\mathrm{SSL}} - \theta^*) \to \mathcal{N}(0, J^{-1}(\theta^*) H(\theta^*) J^{-1}(\theta^*)) \tag{2.14}$$

and θ^{SSL} *is asymptotically efficient, meaning that it achieves asymptotically the smallest variance of any unbiased estimator.*

Asking now when θ^{SSL} asymptotically dominates θ^{SL} we get the somewhat surprising answer that we need the model to be misspecified. From a statistical point of view it is maybe not so surprising, since in the well-specified case (along with some other regularity conditions) the MLE θ^{SL} is already asymptotically efficient itself. Specifically, we have that then $H(\theta^*) = J(\theta^*) = I(\theta^*)$, and we recover the classical result that the MLE is asymptotically normal with a variance of the inverse Fisher information matrix $I(\theta^*)$. The paper then examines, with the logistic regression model, when the difference between $I(\theta^*)$ and $H(\theta^*)$ is particularly big. It is shown that this is the case the more $P(Y \mid X)$ is bounded away from 1/2, so in particular when the Bayes error is small. This is very similar to *Tsybakov's low noise* [26], which is used in statistical learning to show fast learning rates. In Sections 2.6.1 and 2.6.2 similar assumptions are made to show that some semi-supervised learners can converge exponentially fast to the Bayes error.

2.4.2. Using the Unlabeled Data to Pick the Center of the Version Space

Kääriäinen [23] introduces a method for bounding the risk by using unlabeled data to collect information about the agreement of two classifiers. A semi-supervised estimator is then derived as the hypothesis that minimizes this bound. Unfortunately the idea only works really in the realizable case. Although we do not get a new algorithm for the agnostic case, the paper still presents new bounds based on the unlabeled data.

Realizable Case

The idea for the realizable case is to consider the version space, so the space that contains all hypotheses that have no training error. The unlabeled data gives rise to a pseudo-metric on this space by measuring the disagreement of the hypotheses on it. We are going to pick the hypothesis that has the lowest worst-case disagreement to all other hypothesis, of which one must be the true one as we assume realizability. Let us make this more precise. Given two hypotheses $f, g \in H$ we define the disagreement pseudo-metric $d(f, g)$ as

$$d(f, g) = P(f(X) \neq g(X)). \tag{2.15}$$

This metric is specifically useful in the semi-supervised case since is does not depend on labels. We can approximate it with the empirical version by

$$\hat{d}(f,g) = \frac{1}{m}\sum_{i=n}^{n+m} I_{\{f(x_i)=g(x_i)\}}. \tag{2.16}$$

The version space is defined as $H_0 = \{h \in H \mid \hat{R}(h) = 0\}$. Let h_0 be the true hypothesis, then we know that $h_0 \in H_0$ and one can show that $R(h) = d(h, h_0)$ for all $h \in H$. This allows us to bound

$$R(h) = d(h,h_0) = \hat{d}(h,h_0) + (\hat{d}-d)(h,h_0) \leq \sup_{g\in H_0} \hat{d}(h,g) + \sup_{g,g'\in H_0} (\hat{d}-d)(g,g'). \tag{2.17}$$

As Inequality (2.17) bounds the true risk of a hypothesis h, we try to minimize this risk by choosing the hypothesis that minimizes the right-hand side of Inequality (2.17). More precisely, we choose the semi-supervised estimator as the *empirical center of the version space*, so we set

$$h^{\mathrm{SSL}} = \arg\inf_{h\in H_0} \sup_{g\in H_0} \hat{d}(h,g). \tag{2.18}$$

With this we can of course only control the first term on the right-hand side of Inequality (2.17). We can bound the second term, however, with concentration inequalities derived from a Rademacher Complexity for the space $\mathcal{G} = \{x \mapsto I_{\{f(x)=g(x)\}} \mid f, g \in H_0\}$. It is then true that with probability at least $1-\delta$ [23, Theorem 3]

$$R(h^{\mathrm{SSL}}) \leq \inf_{h\in H_0} \sup_{g\in H_0} \hat{d}(h,g) + \mathrm{empRad}(\mathcal{G}) + \frac{3}{\sqrt{2}}\sqrt{\frac{\ln\frac{2}{\delta}}{m}}. \tag{2.19}$$

Note the two terms on the right hand-side of Inequality (2.19) go to 0 for increasing m and note that in this case also $\hat{d}(f,g) \to d(g,g)$. So ignoring for a minute that we only have finitely many unlabeled data we can compare the SSL (2.18) to purely supervised solutions. Note that in the realizable case a purely supervised method would also choose a hypothesis in H_0. As the supervised learner h^{SL} has no further information we can always find a target hypothesis h^* such that $R(h^{\mathrm{SL}}) = \sup_{g\in H_0} d(h^{\mathrm{SL}}, g) = d(h^{\mathrm{SL}}, h^*)$. So the best bound for any supervised learner h^{SL} is given by $R(h^{SL}) \leq \sup_{g\in H_0} d(f,g)$. The SSL bound (2.19) on the other hand allows us to bound $R(h^{\mathrm{SSL}}) \leq \inf_{h\in H_0} \sup_{g\in H_0} d(h,g)$, at least for m going to infinity. From a geometrical viewpoint $\sup_{g\in H_0} d(h^{\mathrm{SL}}, g)$ is the diameter of H_0, while, $\inf_{h\in H_0} \sup_{g\in H_0} d(h,g)$ is the radius. As the difference between the radius and the diameter, with respect to d, is at most 2, we find that the differences in the SSL and SL risk bounds is at most a constant factor of 2.

Bounds for the General Case

In the general case we do not assume that the target hypothesis is part of our hypothesis class. To still make use of the considered metric, the author proposes the following general recipe for bounds in that case. The starting point is the observation that bounds for randomized classifiers are generally tighter when compared to their deterministic counterparts [27, 28]. The idea is now to use such a randomized classifier f_{rand} as an anchor, similarly

to the target hypothesis in the realizable case. To get a bound for a classifier f we then can use the bound for the randomized classifier together with a slack term that includes $\hat{d}(f_{\text{rand}}, f)$. Depending on which kind of randomized classifier we take, we obtain different bounds. This includes for example PAC-Bayesian bounds as well as bounds based on cross-validation and bagging methods. They explicitly derive a cross-validation bound, where the randomized classifier is given by a uniform distribution over the classifiers obtained in the multiple cross-validation rounds.

2.4.3. USING UNLABELED DATA TO COMBINE MULTIPLE HYPOTHESIS SPACES

Leskes [24] presents another scheme that relies on measuring the classification agreement between hypotheses on unlabeled data. The idea here is to use a boosting scheme, so we start with $L \in \mathbb{N}$ different hypothesis classes $H^1, ..., H^L$. We want to find the best fitting hypothesis over all L hypothesis classes $H^1, ..., H^L$. As that would generally lead to an overly increased complexity, the paper reduces the set of possible hypotheses by only considering those that agree sufficiently on the unlabeled data. In this context sufficiently means that we switch to a new hypothesis class H_v for a $v > 0$ that is defined as

$$H_v = \{(h^1, ..., h^L) \in H^1 \times ... \times H^L \mid V(h^1, ..., h^L) \leq v\},$$

where

$$V(h^1, ..., h^L) := \mathbb{E}_X[\frac{1}{L}\sum_i h^i(X)^2 - (\frac{1}{L}\sum_i h^i(X))^2].$$

The term $V(h^1, ..., h^L)$ essentially measures the variance of disagreement within L different hypotheses and is approximated with the unlabeled data. The hypothesis class H_v only keeps those collections of hypotheses that have a sufficiently small variance of disagreement. The paper then presents a generalization bound that holds for all h^l with $1 \leq l \leq L$ simultaneously and the bound depends on the maximum Rademacher complexity of the L base hypothesis classes $H^1, ..., H^L$.

2.5. LEARNING UNDER WEAK ASSUMPTIONS

In the previous two sections we investigated what is possible for semi-supervised learners when we do not have any additional assumptions. Now we investigate what a SSL can achieve under what we call *weak* assumptions. With weak assumptions we mean those that cannot essentially change the learning of $O(\frac{1}{\sqrt{n}})$, but rather gives improvements by a constant which can depend on the hypothesis class. In Section 2.6 we will investigate what we have to assume to escape the $\frac{1}{\sqrt{n}}$ regime. We first cover the work of Balcan and Blum [29], as it is a general framework that allows us to analyze the learning guarantees for multiple semi-supervised learners. They show that semi-supervised learners that fall in this framework learn by a constant faster than supervised learners, where the constant depends on the hypothesis class and the semi-supervised learner we use.

We then cover in more detail the idea of co-training. Although co-training can also be viewed in the framework of Balcan and Blum [29] we want to present a few more details on it. In particular we present the work of Sridharan and Kakade [30] who formulate the as-

sumption of co-training in an information theoretical framework, which allows to precisely quantify the bias-variance trade-off.

2.5.1. A General Framework to Encode Weak Assumptions

We start with the work done by Balcan and Blum [29], as it offers an elegant way to formalize different assumptions in a general framework. Many existing methods can be cast in this framework; transductive support vector machines [31, 32], Multi-View assumptions [24, 30, 33] and transductive graph-based methods [34]. The idea is to introduce a function χ that measures the compatibility between a hypothesis h and the marginal distribution $P(X)$. Compatibility can mean many different things in this context. As a simple example we could call a hypothesis h compatible with a marginal distribution $P(X)$ if its decision boundary goes through low density regions. As we usually only observe a finite sample size, the function χ needs to be defined for each point in the feature space, so one sets

$$\chi : H \times \mathcal{X} \to [0,1]. \tag{2.20}$$

The compatibility measure χ gives then rise to the function

$$R_{\text{unl}}(h) := 1 - \mathbb{E}_{X \sim P(X)}[\chi(h, X)], \tag{2.21}$$

which we will call the *unsupervised loss*. We will try to optimize it in addition to the loss measured on the labeled sample. The paper states several more theorems in the same flavor as the one presented here. The differences are mostly in the realizability assumptions (regarding the unsupervised and the supervised error) and the bounding technique. They present bounds derived from uniform convergence as well as bounds based on covering numbers. The following theorem is the double agnostic case (neither the labeled nor the unlabeled loss have to be zero).

Theorem 6 (Theorem 10)**.** *Let* $h_t^* = \operatorname{arg\,min}_{h \in H}[R(h) \mid R_{\text{unl}}(h) \leq t]$. *Then, given an unlabeled sample size of at least*

$$\mathcal{O}\left(\frac{\max[VC(H), VC(\chi(H))}{\epsilon_2} \ln \frac{1}{\epsilon_2} + \frac{1}{\epsilon_2^2} \ln \frac{1}{\delta}\right)$$

we have that

$$m(h^{\text{SSL}}, H, \epsilon, \delta) \leq \frac{32}{\epsilon^2}\left[VC(H(t + 2\epsilon_2)) + \ln \frac{2}{\delta}\right], \tag{2.22}$$

where h^{SSL} *is the hypothesis that minimizes* $\hat{R}(h^{\text{SSL}})$ *subject to* $\hat{R}_{\text{unl}}(h^{\text{SSL}}) \leq t + \epsilon$ *and* $H(t) := \{h \in H \mid R_{\text{unl}}(h) \leq t\}$. *Here* $\hat{R}$ *is the empirical risk measured with the sample* S_n *and* $\hat{R}_{\text{unl}}$ *is the empirical unlabeled risk measured on the sample* U_m.

We note that the original paper uses a different measure of complexity, so the term $VC(H(t+2\epsilon_2))$ is different. We use the standard VC-dimension instead to avoid additional notation and to allow for an easier comparison to other results. They use a complexity notion that in Vapnik [5] could be found under (the exponentiated) annealed entropy and has the advantage to be distribution dependent.

We now compare Theorem 6 to the results of the previous section, in particular to Conjecture 1 and the answers to this as found in Theorems 3 and 4. We know that in the purely supervised case we can achieve a similar sample complexity as (2.22) by replacing $VC(H(t+2\epsilon_2))$ with $VC(H)$. As we know that the sample complexity given by (2.22) is tight up to constants (compare Chapter 6 from [3]), we know that the sample complexity between a purely supervised learner and the semi-supervised learner as defined in this paper cannot differ by more than $\mathcal{O}\left(\frac{VC(H)}{VC(H(t+2\epsilon_2)}\right)$. So the gap in the learning rates is indeed given by a constant that only depends on the hypothesis class as postulated by Conjecture 2. This constant can, however, be infinite if $VC(H)$ is infinite but $VC(H(t+2\epsilon_2))$ is finite. This is exactly the type of example that refuted the conjecture and which we presented in Section 2.3.2.

Theorem 6 quantifies to some degree the fundamental bias-variance trade-off in SSL when we use assumptions. Employing a semi-supervised compatibility function we reduce the variance of the training procedure as we effectively restrict the original hypothesis space H. If, however, the compatibility function does not match the underlying problem, we bias the procedure away from good solutions.

2.5.2. ASSUMING THAT THE FEATURE SPACE CAN BE SPLIT

In multi-view learning, also sometimes called co-regularization or co-training, one assumes that the feature space $\mathcal{X}$ can be decomposed as $\mathcal{X} = \mathcal{X}^1 \times \mathcal{X}^2$, and each partial feature space $\mathcal{X}^1, \mathcal{X}^2$ is already enough to learn. In the early work on co-training Blum and Mitchell [33] use the idea in a web page classification set. One part of the features, say $\mathcal{X}^1$, is given by the text on the web page itself, while the other one, $\mathcal{X}^2$, is given by the anchor text of hyperlinks pointing to the web page. The idea is that if both partial features spaces have sufficient information about the correct label, we would expect that a correct classifier predicts the same label given any of the two partial features. We can thus discard classifiers that disagree on the two views.

There are multiple theoretical results about this approach, it can be for example analyzed in the framework of the previous section. Rosenberg and Bartlett [35] and Farquhar *et al.* [36] analyze a Rademacher complexity term under the multi-view assumption. Sindhwani and Rosenberg [37] define a kernel that directly includes the assumption as a regularization term, and thus find a RKHS where co-regularization automatically happens.

Here we detail the work of Sridharan and Kakade [30], as this ties in best with the other results we present. In addition their information theoretic framework allows to also analyze the penalty one suffers if the assumption is not exactly true. We split the random variable X which takes values in $\mathcal{X}$ into $X = (X^1, X^2)$. In their framework the multi-view assumption can be formalized as follows.

Multi-View Assumption Let $I(A;B \mid C)$ be the mutual information between random variables A and B, conditioned on knowing already the random variable C. Then there exists an ϵ_{info} such that

$$I(Y;X^2 \mid X^1) \leq \epsilon_{\text{info}} \tag{2.23}$$

and

$$I(Y;X^1 \mid X^2) \leq \epsilon_{\text{info}}. \tag{2.24}$$

Intuitively this states that once we know one of the features, the other feature will not tell us much more about Y.

Comparing this to co-training we can see it as a relaxation. In co-training one assumes that each view is already sufficient to fully learn, which corresponds here to $\epsilon_{\text{info}} = 0$. If, however, $\epsilon_{\text{info}} > 0$, we cannot learn perfectly from one view. (But this is fine in this framework). We assume then, that we have for each view X^1 and X^2 a corresponding hypothesis set H^1 and H^2. We will do predictions with *pairs* of hypotheses $(f_1, f_2) \in H^1 \times H^2$. The paper uses the notion of compatibility functions (2.20). In particular they define a compatibility function $\chi : H := H^1 \times H^2 \to [0,1]$ as $\chi(h^1, h^2, x) := d(f_1(x^1), f_2(x^2))$, where $d : \mathcal{Y} \times \mathcal{Y} \to [0,1]$ is some sort of distance measure that fulfills a relaxed triangle inequality and $x = (x^1, x^2)$ is a sample. The distance d measures in essence how much f_1 and f_2 agree on a sample x. For a given threshold $t \in \mathbb{R}$ we find now the best *pair* of hypotheses with the constrained empirical risk minimization problem

$$\min_{(h^1,h^2)\in H} \quad \sum_{i=1}^{n} l(h^1(x_i^1), y_i) + l(h^2(x_i^2), y_i) \quad \text{subject to} \quad \hat{R}_{\text{unl}}(h^1, h^2) \le t. \tag{2.25}$$

Recall the definition of $R_{\text{unl}}(h)$ from Equation (2.21). The main theorem, which gives guarantees on the solution found by the procedure above, needs the following notation. Let β_*, β_*^1 and β_*^2 be the Bayes error, measured with the loss l, when learning from $X^1 \times X^2$, X^1 and X^2 respectively. We also set $\epsilon_{\text{bayes}} = \max\{R(f_*^1) - \beta_*^1, R(f_*^2) - \beta_*^2\}$, where f_*^i is the best predictor from H^i. Finally we set $\hat{H}(t) = \{(h^1, h^2) \in H \mid \hat{R}_{\text{unl}}(h^1, h^2) \le t\}$.

Theorem 7. *Assume that the loss l is bounded by 1. There exists an $t \in \mathbb{R}$ (depending among other on ϵ_{info}, ϵ_{bayes} and m), such that under some further regularity conditions on $\chi = d$ and the loss l, and given at least $m(\hat{H}(t), \epsilon, \delta)$ labeled samples, with probability $1 - \delta$*

$$\frac{R(\hat{h}^1) + R(\hat{h}^2)}{2} \le \beta_* + \epsilon + \epsilon_{\text{bayes}} + \sqrt{\epsilon_{\text{info}}}. \tag{2.26}$$

We see now that the information theoretic assumption allows us to explicitly describe the bias introduced when switching from the full hypothesis set H to the restricted one $\hat{H}(t)$. This bias is given by $\sqrt{\epsilon_{\text{info}}}$.

2.6. Learning Under Strong Assumptions

In the previous section we analyzed assumptions that only could give us a constant improvement, and did not allow us to escape the general learning rate of $\frac{1}{\sqrt{n}}$. Now we analyze assumptions which allow us to escape this regime, and can even give exponentially fast convergence. The following example illustrates the basic idea behind that. Assume we are given a set of unlabeled data and we use it to cluster the data. If we assume that the clustering is correct, meaning that each cluster corresponds to a class, we essentially need only enough labeled data to identify which cluster belongs to which class. The work we present in this section extends this idea in various ways and answers the following questions. What if we have class overlap? What if there is noise in the clusters? What about regression?

2.6.1. ASSUMING THAT THE MODEL IS IDENTIFIABLE

One of the classic works in semi-supervised learning, that deals with a topic closely related to sample complexity, was done by Castelli and Cover [38]. The setting is very restricted but can give exponentially fast convergence rates to the Bayes risk in the number of labeled samples n. This is very powerful considering that the results of the previous sections could often not essentially fasten the rate of $\frac{1}{\sqrt{n}}$ (compare for example Inequality (2.22) after solving for ϵ).

The first key assumption to obtain those results lies in the data generation process. First the label is drawn with $P(y=1)=\eta$ and $P(y=0)=\bar{\eta}$ and then a feature is drawn according to a density $f_y(x)$. Unlabeled data is thus drawn from the mixture $\eta f_1 + \bar{\eta} f_2$. The second key assumption is that the class of mixture models is identifiable, i.e. that we can infer the mixture model uniquely given only unlabeled data. After observing enough unlabeled data to identify the mixture we only have to figure out how to label each part of the two mixture components. As we thus have only to decide between two alternatives we can find a classifier h by a simple likelihood ratio test, which converges exponentially fast to the Bayes risk in the number of the labeled samples n:

$$R(h) - \min_{h \in H} R(h) \leq \exp\left(n \ln(2\sqrt{\mu\bar{\mu}} \int \sqrt{f_1(x) f_2(x) dx}) + o(n)\right) \tag{2.27}$$

For the analysis it is necessary to assume that one has an infinite amount of unlabeled data. The work is continued in [39], where the authors consider cases where we already have knowledge about the densities f_y. Sinha and Belkin [40] extend a similar framework to the case where the marginal distribution $P(x)$ is unknown. They assume instead that $P(x)$ can be well estimated with a mixture of two spherical Gaussian distributions with density functions $f_1(x)$ and $f_{-1}(x)$. In particular they assume that $||f_1 - P(\cdot|Y=1)||_S$ and $||f_{-1} - P(\cdot|y=-1)||_S$ can be bounded with a small number, where $||\cdot||_S$ is a Sobolev norm. Finally we want to mention the work of Ratsaby and Venkatesh [41], where exponential decay of excess risk is achieved under the assumptions of well-specification and the model class are mixtures of two spherical Gaussian distributions.

2.6.2. ASSUMING THAT CLASSES ARE CLUSTERED AND SEPARATED

In [42] we are presented explicit bounds on the generalization error using another formulation of the cluster assumption. It closely resembles the work of the previous section and under their assumption we again obtain exponentially fast convergence. Their first and simple setup is that we are given a collection of pairwise disjoint clusters $C_1, C_2, ...$ and we make a *cluster assumption*, i.e we assume that the labeling function $x \mapsto \text{sign}(P(Y=1 \mid X=x) - \frac{1}{2})$ is constant on each cluster C_i. So the clusters have a label-purity of some degree, which we can specify by

$$\delta_i = \int_{C_i} |2P(Y=1|X=x) - 1| dP(x), \tag{2.28}$$

where the cluster C_i is pure iff δ_i is either 1 or 0. Assuming that we know the clusters, we let $h_n^{\text{SSL}}(x)$ be the majority voting classifier per cluster. More formally, given a labeled sample S_n let $X_i^+ := \{(x,y) \in S_n \mid x \in C_i, y=1\}$ and similarly $X_i^- := \{(x,y) \in S_n \mid x \in C_i, y=-1\}$. Then given a new data point $x \in C_i$ we set

$$h^{\mathrm{SSL}}(x) = \begin{cases} 1 & \text{if } |X_i^+| \geq |X_i^-| \\ -1 & \text{if } |X_i^+| < |X_i^-|. \end{cases} \tag{2.29}$$

Note that this defines only a function on the clusters. The paper argues, however, that unlabeled data cannot help where no unlabeled data was observed. Consequently it only analyses the possible gain from unlabeled data on the clusters. Thus the excess risk is now restricted to the set $C := \bigcup C_i$, so we set the excess risk as

$$\mathcal{E}_C(h) = \int_C |2P(Y=1|X=x) - 1| I_{\{h(x) \neq h^*(x)\}} dP(x),$$

where h^* is the Bayes classifier. The following theorem describes the gain one can make with respect to the expected cluster excess risk.

Theorem 8 (Theorem 3.1)**.** *Let $(C_i)_{i \in I}$ be a collection of sets with $C_i \subset \mathcal{X}$ for all $i \in I$ such that this collection fulfills the above defined cluster assumption. Then the majority voting classifier h_n^{SSL} as defined above satisfies*

$$\mathbb{E}_{S_n, U_m}\left[\mathcal{E}_C(h_n^{\mathrm{SSL}})\right] \leq 2 \sum_{i \in I} \delta_i e^{\frac{-n\delta_i^2}{2}}. \tag{2.30}$$

So knowing the clusters we recover the exponential convergence in the labeled sample size as in [38]. The biggest effort of the paper goes, however, in the definition of clusters and the finite sample size estimation of such. The derivations are rather long and here we limit ourselves to describe the underlying intuition. First we assume that the marginal distribution $P(X)$ allows for a density function $p(x)$ with respect to the Lebesgue measure. With that one can define the density level sets of $\mathcal{X}$ w.r.t. a parameter $\lambda > 0$ as $\Gamma(\lambda) := \{x \in \mathcal{X} \mid p(x) \geq \lambda\}$. For a fixed $\lambda > 0$ we think of a clustering essentially as path-connected components of the density level sets $\Gamma(\lambda)$, where it is ensured that pathological cases are excluded. Estimating the set $\Gamma(\lambda)$ with finitely many unlabeled samples adds a slack term to Inequality (2.30) that drops polynomially in the unlabeled sample size. So, to ensure that we still can learn exponentially fast, the number of unlabeled samples has to grow exponentially with the number of labeled samples.

2.6.3. Assuming that the Classes are Clustered but Not Necessarily Separated

Singh *et al.* [43] propose a different formalization of the cluster assumption, one that allows to distinguish cases where SSL does help and where not. This is done by restricting the class of distributions $\mathcal{P}$ and then investigating which of those distributions allow for successful semi-supervised learning. The class $\mathcal{P}$ is constructed such that the marginal distributions are constituted of different clusters that are sometimes easy to distinguish and sometimes not. The marginal densities $p(x)$ from $\mathcal{P}$ are given by mixtures of K densities p_k. So $p(x) = \sum_{i=1}^K a_k p_k(x)$ with $a_k > 0$ and $\sum_{i=1}^K a_k = 1$ and each p_k has support on a set $C_k \subset \mathcal{X}$ which fulfills some regularity conditions. We call the sets C_k clusters, and each one is assumed to have its own smooth label distribution function $p_k(y \mid x)$. So with probability a_k we draw from $p_k(x)$ and then label x according to $p_k(y \mid x)$. We further only consider

distributions that lead to clusters with margin, with our without overlap, of at least γ (see also Figure 2.5), and denote the resulting class of distributions by $\mathcal{P}(\gamma)$. In this formulation the clusters are not of the main interest, but rather what the authors call the *decision sets*.

To define a decision set we denote with C_k^c the complement of C_k and define $C_k^{\neg c} := C_k$. A set $D \subset \mathcal{X}$ is called a decision set if it can be written as

$$D = \bigcap_{k \in K} C_k^{i_k}$$

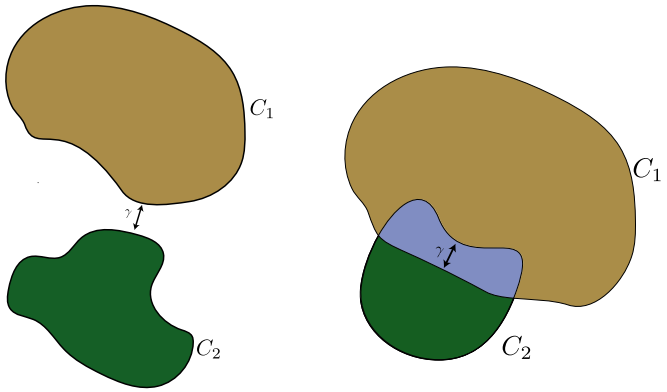
for $i_k \in \{c, \neg c\}$, see Figure 2.5 (b) for an example. The advantage of the decision sets over the clusters are that the full distribution $p(x, y)$ is not necessarily smooth on each cluster, as they might exhibit jumps at the borders. On the decision sets, however, $p(x, y)$ will be smooth, if each $p_k(y \mid x)$ is smooth. Thus, if we would know the decision sets we could use a semi-supervised learner that uses the smoothness assumption.

The main theorem answers the question whether or not one can learn the decision sets from finitely many unlabeled points.

Theorem 9 (Corollary 1)**.** *Let $\mathcal{E}(h) = R(h) - R^*$ be the excess risk with respect to the Bayes classifier R^*. Assume that $\mathcal{E}$ is bounded and that there is a learner h_n^D that has knowledge of all decision sets D and fulfills the following excess risk bound.*

$$\sup_{P \in \mathcal{P}(\gamma)} \mathbb{E}_P[\mathcal{E}(h_n^D)] \leq \epsilon_2(n) \tag{2.31}$$

Assume that $|\gamma| > 6\sqrt{d}\kappa_0(\frac{(\ln m)^2}{m})^{\frac{1}{d}}$, where κ_0 is a constant, then a semi-supervised learner



(a) The clusters C_1 and C_2 are separated with margin γ. The different decision regions are here just the clusters.

(b) The clusters C_1 and C_2 are have an overlap (light blue) with margin γ. The three colors also constitute three different decision sets.

Figure 2.5: Picture (a) shows the concept of a positive γ-margin, while (b) shows a negative γ-margin.

$h^{\text{SSL}}_{n,m}$ *exists such that*

$$\sup_{P \in \mathcal{P}(\gamma)} \mathbb{E}_P[\mathcal{E}(h^{SS}_{n,m})] \leq \epsilon_2(n) + O\left(\frac{1}{m} + n\left(\frac{(\ln m)^2}{m}\right)^{\frac{1}{d}}\right). \tag{2.32}$$

Note the following. If the learner h^D_n that knows the decision sets has a convergence rate of $\epsilon_2(n)$, it follows from Inequality (2.32) that the unlabeled data needs to increase with a rate of $\epsilon_2(\frac{1}{n})$ to ensure that the semi-supervised learner has the same convergence rate as h^D_n. For example, if h^D_n converges exponentially fast, we need a exponentially much more unlabeled then labeled data, which is the same finding as in the previous section.

The intuition here is fairly simple. The bigger γ the less unlabeled samples we need to estimate the decision sets D, and once we know those, we can perform as well as h^D_n. To analyze if a semi-supervised learner that first learns the decision sets empirically has an advantage over all supervised learners, they first find minimax lower bounds for all fully supervised learners. They then give upper bounds for a specific semi-supervised learner and the conclusions are intuitive: For SSL to be useful, the parameter γ and the number of unlabeled samples should be such that the fully supervised learner cannot distinguish the decision sets, while the semi-supervised learner can. So γ should not be too big, as then the supervised learner can also distinguish the decision sets. And, of course, the unlabeled data should not be too little, as then the semi-supervised learner cannot distinguish the decision sets.

To present specific differences between SSL and SL the authors assume that $\mathcal{X} = [0,1]^d$ and that the conditional expectations $\mathbb{E}_{Y \sim p_k(Y|X=x)}[Y|X = x]$ are Hölder-α smooth functions in x. Depending on γ the paper presents a table for cases when SSL can be essentially faster than SL. In those cases the SL has an expected lower bound for the convergence rate of $n^{-\frac{1}{d}}$ while the convergence rate of the SSL is upper bounded by $n^{-\frac{2\alpha}{2\alpha+d}}$.

2.6.4. Assuming the Regression Function is Smooth Along a Manifold

As we will elaborate further in the discussion section, an issue in SSL is that most methods are based on assumptions on the full distribution. The problem is that we usually cannot verify whether the assumptions holds or not. This is crucial to know, since in case the assumption does not hold, it is quite likely that we want to use a supervised learner instead. The work of Azizyan *et al.* [44] is one of the few papers that touches on that topic as they introduce a semi-supervised learner that depends on a parameter α, where $\alpha = 0$ recovers a purely supervised learner. The paper then gives generalization bounds for the semi-supervised learner when we cross-validate α. As this work uses the regression setting, while most other presented papers deal with classification, and gives a clean formalization of the SSL, we present here the details. The authors use a version of the manifold assumption, so we enforce our estimated regression function $h^{\text{SSL}}(x)$ to behave smoothly in high density regions. The density of the marginal distribution $P(X)$ is measured with a smoothed density function $p_\sigma(x)$

$$p_\sigma(x) := \int \frac{1}{\sigma^d} K\left(\frac{||x-u||}{\sigma}\right) dP(u), \tag{2.33}$$

where K is a symmetric kernel on $\mathbb{R}^d$ with compact support and $\sigma > 0$. Let $\Gamma(x_1, x_2)$ be the set of all continuous paths $\gamma : [0, L(\gamma)] \to \mathbb{R}^d$ from $x_1 \in \mathbb{R}$ to $x_2 \in \mathbb{R}$ with unit speed and where $L(\gamma)$ is the length of γ. With this we can define a new metric (the so-called exponential metric) on $\mathbb{R}^d$ that depends on a parameter $\alpha \geq 0$ and the smoothed density $p_\sigma(x)$.

$$D(x_1, x_2) = \inf_{\gamma \in \Gamma} \int_0^{L(\gamma)} e^{-\alpha p_\sigma(\gamma(t))} \, dt \tag{2.34}$$

First note that $\alpha = 0$ corresponds to the Euclidean distance. Second, note that high values of $p_\sigma(x)$ on the path between two points x_1 and x_2 lead to shorter distances between those points in the new metric, and this is emphasized with large α. If we assume that Q is another kernel and we set $Q_\tau(x) := \frac{1}{\tau^d} Q(\frac{x}{\tau})$ we can define the semi-supervised estimator as

$$h^{\mathrm{SSL}}(x) := \frac{\sum_{i=1}^n y_i Q_\tau(\hat{D}(x, x_i))}{\sum_{i=1}^n Q_\tau(\hat{D}(x, x_i))}. \tag{2.35}$$

The estimator is thus a nearest-neighbor regressor, where neighbors are weighted according to their distance in the D-metric. The following theorems gives bounds on the squared risk of h^{SSL} under the assumption that $\sup_{y \in \mathcal{Y}} |y| = M < \infty$.

Theorem 10 (Theorem 4.1)**.** *Let $\mathcal{P}(\alpha, \sigma, L)$ be a class of probability measures that fulfill certain regularities depending on parameters $\alpha, \sigma, L \geq 0$ (more details after the Theorem). Assume that for all $P \in \mathcal{P}$ we have $P(||\hat{p}_\sigma - p_\sigma|| \geq \epsilon_m) \leq \frac{1}{m}$, then*

$$\mathbb{E}_{S_n, U_m}[R(h^{\mathrm{SSL}}] \leq L^2(\tau e^{\alpha \epsilon_m})^2 + \frac{1}{n} M^2 (2 + \frac{1}{e}) \mathcal{N}_{P,\alpha,\sigma}(e^{-\alpha \epsilon_m} \frac{\tau}{2}) + \frac{4M^2}{m}. \tag{2.36}$$

In this notation $\mathcal{N}_{P,\alpha,\sigma}(\epsilon)$ is the *covering number* of P in the D-metric: The minimum number of closed balls in $\mathcal{X}$ of size ϵ w.r.t to the D-metric necessary to cover the support of $P(X)$, see also Shalev-Shwartz and Ben-David [3, Chapter 27]. In the Euclidean case, so when $\alpha = 0$, we can bound $\mathcal{N}_{P,\alpha,\sigma}(\epsilon) \leq (\frac{C}{\epsilon})^d$ with a constant C. The covering number can be much smaller when $\alpha > 0$ and $P(X)$ is concentrated on a manifold with dimension smaller than d. The regularity conditions on $\mathcal{P}(\alpha, \sigma, L)$ are essentially the following. First we assume that $P(X)$ has compact support. Second, all regression functions $f_P(x) = \mathbb{E}P(Y \mid X = x) : \mathbb{R}^d \to \mathbb{R}$ are L-Lipschitz continuous, where the domain $\mathbb{R}^d$ is equipped with the exponential metric D and the co-domain $\mathbb{R}$ is equipped with the Euclidean distance.

As the previous Theorem might be quite difficult to parse, the paper offers a simplified corollary, under some further regularity conditions.

Corollary 1 (Corollary 4.2)**.** *Assume that $\mathcal{N}_{P,\alpha,\sigma}(\delta) \leq (\frac{C}{\delta})^\xi$ for some certain range of δ. Furthermore assume that m is large enough and that $\tau(n, \alpha, \epsilon_m, \xi)$ is well chosen. Then for all $P \in \mathcal{P}(\alpha, \sigma, L)$*

$$\mathbb{E}_{S_n, U_m}[R(h^{\mathrm{SSL}}] \leq \left(\frac{C}{n}\right)^{\frac{2}{2+\xi}}. \tag{2.37}$$

The paper then analyzes the additional penalty we occur in trying to find the best α. This is done by discretizing the parameter space $\Theta = \mathcal{T} \times \mathcal{A} \times \Sigma$ such that $\theta = (\tau, \alpha, \sigma) \in \Theta$ and

$|\Theta| = J < \infty$. Assume that we have in addition to the training sample S_n also a validation set $V = \{(v_1, z_1), ..., (v_n, z_n)\}$, for convenience also of size n. Let h_θ^{SSL} be the semi-supervised hypothesis trained on S_n with the parameters θ. We then choose the final hypothesis h^{SSL} by choosing θ with cross-validation

$$h^{\text{SSL}} := \arg\min_{h_\theta^{\text{SSL}}} \sum_{i=1}^{n} (h_\theta^{\text{SSL}}(v_i) - z_i)^2. \tag{2.38}$$

Theorem 11 (Theorem 6.1)**.** *Let $\mathcal{E}(h) := R(h) - R(h^*)$ be the excess risk, where h^* is the true regression function. There are constants [not universal, depend to some degree on the problem] $0 < a < 1$ and $0 < t < \frac{15}{38(M^2+\sigma^2)}$ such that*

$$\mathbb{E}_{S_n,U_m,V}[\mathcal{E}(h^{\text{SSL}})] \leq \frac{1}{1-a}\left(\min_{\theta\in\Theta} \mathbb{E}_{S_n,U_m}[\mathcal{E}(h_\theta^{\text{SSL}})] + \frac{\ln(nt4M^2) + t(1-a)}{nt}\right), \tag{2.39}$$

This is particularly interesting since we implicitly compare to the supervised solution, as long as we include $\alpha = 0 \in \mathcal{A}$. From Inequality (2.39) we see that the validation process introduces a penalty term of $O(\frac{\ln(n)}{n})$. In the worst case this can be seen as an additional error term if we use the semi-supervised method, but the assumption is actually not true.

Finally the authors identify a case where the semi-supervised learning rate can be strictly better than the supervised learning rate, much like we have seen in Section 2.3.2. In particular, they construct a set of distributions $\mathcal{P}_n$, which depends on the number of labeled samples, such that

1. the estimator $h^{\text{SSL}}(x)_{\tau,\alpha,\sigma}$, as defined in Equation (2.35), fulfills

$$\sup_{P\in\mathcal{P}_n} \mathbb{E}_{S_n}[R(\hat{f}_{\tau,\alpha,\sigma})] \leq \left(\frac{C}{n}\right)^{\frac{2}{2+\xi}},$$

   under the assumption that $m \geq 2^{\frac{2}{2+\xi}}$.

2. for all purely supervised estimators h^{SL} we have that

$$\sup_{P\in\mathcal{P}_n} \mathbb{E}_{S_n}[h^{\text{SL}}] \geq \left(\frac{C}{n}\right)^{\frac{2}{d-1}}.$$

To obtain essentially different learning rates we need that $\xi < d-3$, which is the case if P is concentrated on a set with dimension strictly less than $d-3$ [44, Lemma 1]. It is also worth noting that the construction of $\mathcal{P}_n$ works by concentrating the distributions more for bigger n. If $\mathcal{P}_n$ does not concentrate, and remains smooth for bigger n, the labeled data is already enough to approximate the marginal distribution.

This is similar to the work presented in Section 2.6.3, as they also show that SSL can only work if the marginal distribution $P(X)$ is not too easy to identify. We can also draw parallels to the work presented in Section 2.3.2; if we would restrict the domain distributions such that only smooth circle embeddings would be allowed, a supervised learner could also learn efficiently. This is because then a finite number of labeled samples would be sufficient to learn the domain distribution uniformly, so the semi-supervised learner would loose its benefits.

2.7. LEARNING IN THE TRANSDUCTIVE CASE

While many methods use unlabeled data to find better classification rules, some consider schemes where one only cares about the labels of the unlabeled data. Those methods are often called transductive [5, Chapter 8]. We present the most important theoretical results. A more detailed survey on theoretical and practical transductive learning can be found in Chapter 2 of Pechyony [45]. In Subsection 2.7.1 we present learning bounds in the transductive case. They often arise as direct extension of the inductive case and related concepts. In Subsection 2.7.2, which cannot be found as part of [45], we present two papers that touch on the topic of so-called safe semi-supervised learners. Their aim is to construct semi-supervised learners that are never worse than their supervised counterparts.

One can distinguish two transductive settings, where the essential difference is that in one setting we sample without replacement, so the samples become dependent. The work about transductive learning which we present here deals with Setting 1, mostly because of convenience. We note, however, that one can transform bounds from Setting 1 to bounds from Setting 2 [5, Theorem 8.1].

Setting 1

1. We start with a fixed set of points $X_{n+m} = \{x_1, ..., x_{n+m}\}$.
2. We reveal the labels Y_n of a set $X_n \subset X_{n+m}$ which is uniformly selected at random. For notational convenience we usually assume w.l.o.g that X_n are the first n and X_m are the last m points of X_{n+m}.
3. Based on $S_n = (X_n, Y_n)$ and X_m we try to find a classifier h with good performance on $R_m(h) := \sum_{i=n}^{n+m} l(x_i, y_i)$.

Setting 2

1. We start with a fixed distribution P on $\mathcal{X} \times \mathcal{Y}$.
2. We draw n i.i.d. samples according to P to obtain a training set S_n. We draw m i.i.d. samples according to $P(X)$ to obtain a test set X_m.
3. Based on $S_n = (X_n, Y_n)$ and X_m we try to find a classifier h with good performance on $\mathbb{E}_{S_n, X_m}\left[\frac{1}{m}\sum_{i=n}^{n+m} l(h(x_i), y_i)\right]$.

Note that in this section our test error is denoted by $R_m(h)$ and the training error by $R_n(h)$. This reflects that the test is of size m while the training set of size n. We will not use the hat notation here, as in the transductive setting we do not necessarily have an underlying distribution.

2.7.1. TRANSDUCTIVE LEARNING BOUNDS

VAPNIK'S IMPLICIT TRANSDUCTIVE BOUND

Transductive inference goes back to Vapnik [46]. We present the result found as Equation (8.15) in Theorem 8.2. from Vapnik [5]. Assume that we are given $n+m$ samples and we pick at random n samples on which we can train. We then want to estimate the error we

make on the leftover m samples. Vapnik shows that a hypergeometric distribution describes the probability that the observed error on the train and test set is bigger than ϵ

$$P\left(\frac{|R_m(h) - R_n(h)|}{\sqrt{R_{n+m}(h)}} > \epsilon\right).$$

Let ϵ^* be the smallest $\epsilon > 0$ such that

$$P\left(\frac{|R_m(h) - R_n(h)|}{\sqrt{R_{n+m}(h)}} > \epsilon\}\right) \leq 1 - \delta.$$

Using a uniform bound[9] and substituting $R_{n+m} = \frac{m}{n+m}R_m + \frac{n}{n+m}R_n$ one can derive the following result.

Theorem 12. *For all $h \in \{-1, 1\}^{n+m}$ the following inequality holds with a probability of $1 - \delta$*

$$R_m(h) \leq R(h) + \frac{(\epsilon^*)^2 m}{2(m+n)} + \epsilon^* \sqrt{R(h) + \left(\frac{\epsilon^* m}{2(m+n)}\right)^2} \tag{2.40}$$

The problem of this inequality is that the term ϵ^* is an implicit function of n, m, δ and h and thus it is unclear what the learning rates are that we can actually achieve. This problem is addressed in the paper presented in the next section.

Bounds as a Direct Extension of Inductive Bounds

The transductive bound of Inequality (2.40) is difficult to interpret as it contains a function which can only be implicitly calculated. Derbeko *et al.* [47] find explicit transductive bounds in a PAC-Bayes framework. We present a bound from the paper which is essentially a direct extension of an inductive bound from [48]. To present the result they use a Gibbs classifier. For that, let q be any distribution over the hypothesis set H. The Gibbs classifier G_q classifies a new instance $x \in \mathcal{X}$ with an $h \in H$ drawn accordingly to q. The risk of G_q over the set S_n is then $R_n(G_q) = \mathbb{E}_{h\sim q}[\frac{1}{n}\sum_{i=1}^{n} l(h(x_i), y_i)]$.

Theorem 13 (Theorem 17)**.** *Let p be any (prior) distribution on H, which may depend on S_{n+m}, and let $\delta > 0$. Then for any randomly selected subset $S_n \subset S_{n+m}$ and for any distribution q on H, it holds with probability at least $1 - \delta$ that*

$$R_m(G_p) \leq R_n(G_p) + \frac{m+n}{m}\left(\sqrt{\frac{2R_n(G_p)(\mathrm{KL}(q||p) + \ln\frac{n}{\delta})}{n-1}} + \frac{2(\mathrm{KL}(q||p) + \ln\frac{n}{\delta})}{n-1}\right). \tag{2.41}$$

This theorem is indeed a direct extension of the inductive supervised case as found under Equation (6) in [48], the only difference is that the term $\frac{m+n}{m}$ is missing. Although McAllester [49] showed that under certain conditions one can select the prior p after having seen S_m, this is generally not allowed in inductive PAC-Bayesian theory. In the transductive setting this is allowed, as we only care about the performance on the points from the

[9] Note that in the transductive case we effectively can have only finitely many different hypotheses.

set S_{n+m}. In a way this is the same as learning with a fixed distribution when our fixed distribution has only mass on finitely many points [50] .

Derbeko *et al.* [47] exploit this by choosing a prior p with a cluster method. More precisely, after observing the dataset X_{n+m} one constructs c different clusterings on it. Each clustering leads to multiple classifiers by assigning all points in a cluster to the same class. One then puts essentially a uniform prior p on those classifiers and we select a posterior distribution q over the classifiers by minimizing Inequality (2.41), and obtain the Gibbs classifier G_q.

Comparing this approach to the fully supervised (and thus necessarily inductive) case, we realize that the possible performance improvements have the same flavor as the improvements one can gain in semi-supervised learning with assumptions, as analyzed in Sections 2.5 and 2.6. Using the clustering approach from above will reduce the penalty in Inequality (2.41) which is coming from $\mathrm{KL}(q||p)$. In other words: We reduce the variance of the classifier. On the other hand, using a clustering approach will bias our solution, and we will degrade over a supervised solution if clusterings have a high impurity, meaning that the clusterings don't have clear majority classes.

BOUNDS BASED ON STABILITY

In [51] transductive bounds are explored under the notion of stability, the assumption that the output of a classifier does not change much if we perturb the input a bit. The transductive bounds are an extension of the inductive bounds that use the notion of *uniform stability* [52] and *weak stability* [53, 54]. We present the simpler transductive bound based on uniform stability and explain the difference to weak stability.

Assume that $h^{\mathrm{trans}} \in H$ is a transductive learner, so a hypothesis that we (deterministically) choose based on a labeled set S_n and an unlabeled set X_m. Furthermore define $S_n^{ij} := \big(S_n \setminus \{(x_i, y_i)\}\big) \cup \{(x_j, y_j)\}$ and $X_m^{ij} := \big(X_m \setminus \{x_j\}\big) \cup \{x_i\}$. So S_n^{ij} is the set we obtain when we replace in S_n the i-th example from the training set with the j-th example from the test set. We say that h^{trans} is *β-uniformly stable* if for all choices $S_n \subset S_{n+m}$ and for all $1 \le i, j \le n+m$ such that $(x_i, y_i) \in S_n$ and $x_j \in X_m$ it holds that

$$\max_{1 \le k \le n+m} |h^{\mathrm{trans}}_{(S_n, X_m)}(x_k) - h^{\mathrm{trans}}_{(S_n^{ij}, X_m^{ij})}(x_k)| \le \beta. \tag{2.42}$$

In words: The transductive learner h^{trans} is β-uniformly stable if the output changes less than β if we exchange two points from the train and test set. The bounds are formulated using a γ-margin loss. For $\gamma > 0$ we set

$$l_\gamma(y_1, y_2) = \max(0, \min(1, 1 - \frac{y_1 y_2}{\gamma})). \tag{2.43}$$

Consequently we write $R_\gamma(h)$ for the risk of h when measured with the loss l_γ. Note that for $\gamma \to 0$ the l_γ loss converges to the $0-1$ loss.

Theorem 14 (Theorem 1)**.** *Let h^{trans} be a β-uniformly stable transductive learner and $\gamma, \delta > 0$. Then, with probability of at least $1 - \delta$ over all train and test partitions, we have that*

$$R_m(h^{\mathrm{trans}}) \le R_n^\gamma(h^{\mathrm{trans}}) + \frac{1}{\gamma} O\left(\beta \sqrt{\frac{mn \ln \frac{1}{\delta}}{m+n}}\right) + O\left(\sqrt{(\frac{1}{m} + \frac{1}{n}) \ln \frac{1}{\delta}}\right). \tag{2.44}$$

Note that β will depend on n and m, and we would expect that the bigger our training set is, the less our algorithm changes if we exchange two samples from the train and test set. In the transductive bounds based on Rademacher complexities, in the section further below, one can achieve a convergence rate of $\frac{1}{\sqrt{\min(m,n)}}$. To obtain the same rate with Inequality (2.44) we need that β behaves as $O\left(\sqrt{(\frac{1}{n}+\frac{1}{m})\frac{1}{\min(n,m)}}\right)$. This stability rate can be indeed achieved for regularized RKHS methods as demonstrated by Johnson and Zhang [55].

Bounds Based on Transductive Rademacher Complexities

Rademacher complexities are a well studied and established tool for risk bounds in the inductive case [56]. El-Yaniv and Pechyony [57] introduce a transductive version of these quantities. While in the inductive case we have to chose our hypothesis class before seeing any data, the transductive case allows us to chose the hypothesis class data-dependent. The definition of the transductive Rademacher complexity of a hypothesis class H follows closely the inductive case and will be denoted by $\mathrm{tRad}(H)$. Utilizing the γ-margin loss function (2.43) and the corresponding empirical risk $R^{\gamma}(h)$, the paper shows then that for all $h \in H$

$$R_m(h) \leq R_n^{\gamma}(h) + \frac{\mathrm{tRad}(H)}{\gamma} + O\left(\frac{1}{\sqrt{\min(m,n)}}\right).$$

Examining the inequality on first sight, it seems somewhat surprising that the labeled and unlabeled data play an equivalent role in terms of convergence. While slow convergence for $n \ll m$ is not really surprising one has to realize that in the case where $m \ll n$ the transductive risk has a very high variance and thus we have large intervals for high-confidence estimations. This bound can be used to directly estimate the trandsuctive risk for transductive algorithms.

Maximov *et al.* [58] make different use of Rademacher complexities to derive risk bounds for a specific multi-class algorithm. Their algorithm uses a given clustering based on the full data to find a hypothesis which is in some way compatible with the found clustering. The transductive multi-class Rademacher complexities then make direct use of this clustering. With this algorithm the authors show that if we have K initial classes one can achieve a learning rate in the order of $\tilde{O}(\frac{\sqrt{K}}{\sqrt{n}} + \frac{K^{3/2}}{\sqrt{m}})$ [58, Corollary 4]. Not surprisingly the learning rates are essentially the same as in the binary transductive cases, although we note that this analysis was done with Setting 2.

Bounds Based on Learning a Kernel

As a direct extension of the inductive case [59], Lanckriet *et al.* [60] propose to use the unlabeled data to learn a kernel that is suitable for transductive learning. The idea is to use a kernel method that allows to choose from a certain class of kernels in order to optimize the objective function. The presented PAC-bound shows that good (transductive) performance is achieved with a good trade-off between the complexity of the kernel class and the empirical error. Their exemplary kernel classes are designed as follows. Given an initial set of kernels $\{K_1, ..., K_k\}$, that are defined on the labeled *and* unlabeled data, they define

$$K_c := \{K = \sum_{j=1}^{k} \mu_j K_j \mid K \succcurlyeq 0, \mu_j \in \mathbb{R}, \mathrm{trace}(K) \leq c\}$$

and

$$K_c^+ := \{K = \sum_{j=1}^{k} \mu_j K_j \mid K \succcurlyeq 0, \mu_j \in \mathbb{R}, \mu_j \geq 0, \text{trace}(K) \leq c\}.$$

Every class of kernels $\mathcal{K}$ give rise to the hypothesis set

$$H_{\mathcal{K}} = \{h(x_j) := \sum_{j=1}^{2n} \alpha_i K_{ij} \mid K \in \mathcal{K}, \alpha = (\alpha_1, ..., \alpha_{2n}) \in \mathbb{R}^{2n}, \alpha^t K \alpha \leq \frac{1}{\gamma^2}\}.$$

The error bound found in this paper reads then as follows.

Theorem 15 (Theorem 24)**.** *For every $\gamma > 0$, with probability at of at least $1 - \delta$ over every training and test set of size n (so $m = n$) uniformly chosen from (X, Y), every function $h \in H_{\mathcal{K}}$ has*

$$R_m(h) \leq \hat{R}_n^{\text{hinge}}(h) + \frac{1}{\sqrt{n}} \left(4 + \sqrt{2 \log(\frac{1}{\delta})} + \sqrt{\frac{\text{comp}(\mathcal{K})}{n\gamma^2}}\right),$$

where $\hat{R}^{\text{hinge}}(h)$ is the empirical hinge loss of h and $\text{comp}(\mathcal{K})$ is a complexity measure of $\mathcal{K}$ defined as

$$\text{comp}(\mathcal{K}) = \mathbb{E} \max_{K \in \mathcal{K}} \sigma^t K \sigma$$

with σ being a vector of $2n$ Rademacher variables. The complexity measure for the previously defined kernel classes $\mathcal{K}_c$ and $\mathcal{K}_c^+$ can be computed and bounded by

$$\mathcal{K}_c = c \mathbb{E} \max_{K \in \mathcal{K}} \sigma^t \frac{K}{\text{trace } K} \sigma \leq cn,$$

and

$$\mathcal{K}_c^+ \leq c \min\left(k, n \max_{1 \leq j \leq k} \frac{\lambda_j}{\text{trace}(K_j)}\right),$$

where λ_j is the largest eigenvalue of K_j.

Note that since $m = n$ we find that this bound gives the same learning rate of $O(\frac{1}{\sqrt{m+n}})$ as also found in Sections 2.7.1 and 2.7.1.

The effect the unlabeled data has on this procedure depends on the initial kernel guesses $\{K_1, ..., K_k\}$, but is of no further interested in this paper. We can find extensions in [61](p. 282, bottom), where the K_i are chose in a particular way: If we assume that ψ_i is the i-th eigenvector of the graph Laplacian L we can set $K_i = \psi_i \psi_i{}^t$. As described in [61] (p. 280) we can then enforce classifiers found by this procedure to be smooth along the data manifold, if we enforce that μ_i is small when the eigenvalue of ψ_i is large. Similar results are obtained by Johnson and Zhang [62], where the biggest difference are the kernels that are used. Instead of using an initial set of kernels, Johnson and Zhang [62] use the spectral decomposition of a given kernel and shrinks it, where the shrinkage depends on the unlabeled data.

2.7.2. Safe Transductive Learning

In the semi-supervised learning community it is well known that using a semi-supervised procedure often comes with a risk of performance degradation [61, Chapter 4]. This problem led some authors to ask the question whether it is possible to do semi-supervised learning in a safe way, which means that one can guarantee that the SSL will not be worse than a supervised counterpart. So far we compared mostly SSL and SL risk bounds. But, even if the assumptions of the risk bounds are true, a smaller bound still does not guarantee improvements. We will specifically look at work from Li and Zhou [63] and Loog [12]. The results from both works are based on a minimax formulation and show that, under some assumptions, one can indeed guarantee improvements by doing SSL. The analysis is also done in the transductive Setting 1. This means that we have a training set S_n and a test set X_m.

A Minimax Approach for SVMs

The baseline for the model proposed by Li and Zhou [63] is the $S3VM$ [64], which takes the unlabeled data into account by finding a low-margin solutions. The proposed model $S4VM$ finds a few diverse low-margin solutions, and then picks amongst these within a minimax framework to hedge against possible worst case scenarios. Assume we found a set of a few proposed solutions $H_p = \{h_1, ..., h_T\}$. The idea is to contrast those solutions to the supervised solutions h^{SVM}. Assume for now that we know the true labels $Y_m = (y_n, ..., y_{n+m})$ of X_m. With this we can calculate the gain and loss in performance when comparing the supervised h^{SVM} to any other classifier h.

$$\text{gain}(h, Y_m, h^{SVM}) := \sum_{i=n}^{n+m} I_{\{h(x_i)=y_i\}} I_{\{h^{SVM}(x_i)\neq y_i\}} \tag{2.45}$$

$$\text{loss}(h, Y_m, h^{SVM}) := \sum_{i=n}^{n+m} I_{\{h(x_i)\neq y_i\}} I_{\{h^{SVM}(x_i)=y_i\}} \tag{2.46}$$

If we define our objective as to be the difference of those two

$$J(h, y, h^{SVM}) = \text{gain}(h, Y_m, h^{SVM}) - \text{loss}(h, Y_m, h^{SVM}), \tag{2.47}$$

we can define a semi-supervised model h^{SSL} as the maximizer of this difference. Since we actually don't know the true labeling, we assume a worst-case scenario that leads to the following max-min formulation.

$$h^{\text{SSL}} = \arg\max_{h\in H_p} \min_{Y\in Y_p} J(h, Y, h^{SVM}) \tag{2.48}$$

Here $Y_p = \{(h(u_1), ..., h(u_m)) \mid h \in H_p\}$ is the set of all possible labelings that we can achieve with H_p. To guarantee that our SSL is not worse than the SL it is important to assume that the true labels Y_m are part of the set Y_p, because only then we can guarantee the following.

Theorem 16 (Theorem 1)**.** *If $Y_m \in Y_p$, the accuracy of h^{SSL} is never worse than the accuracy of h^{SVM}, when performance is measured on the unlabeled data X_m.*

Again, the crucial assumption is that $Y_m \in Y_p$, which corresponds in this case exactly to a low-density assumption. This is because the set Y_p contains possible labelings that come from classifiers that fulfill the low density assumption. One can imagine to use the same procedure also for different assumptions as we can encode them by Y_p, the set of all labelings that we consider possible. While this paper still needs some assumptions, Loog [12] shows a case where we get guaranteed improvements assumption-free. This, however, comes at the cost of measuring the improvements in terms of likelihood, and not in terms of accuracy.

A Minimax Approach for Generative Models

The second paper in this line of research is to our knowledge possibly the only paper in semi-supervised learning that considers a completely assumption-free case. This of course comes at a cost, more on that later. The starting point is a family of probability density functions $p(x, y \mid \theta)$ on $\mathcal{X} \times \mathcal{Y}$, where $\theta \in \Theta$ is a parametrization. First we set θ^{SL} to be the supervised maximum likelihood estimator for the model $p(x, y \mid \theta)$, so

$$\theta^{\mathrm{SL}} = \arg\min_{\theta \in \Theta} \left[\sum_{(x,y) \in S_n} \ln p(x, y \mid \theta) \right].$$

Assume for now that we know the true conditional probabilities $p = (p_1, ..., p_{m+n}) \in [0,1]^{m+n}$ with $p_i = P(Y = 1 \mid X = x_i)$ for $x_i \in S_n \cup X_m$. If we would know this we would actually rather optimize the expected log-likelihood of the model $p(x, y \mid \theta)$ evaluated on the complete dataset $X_{n+m} = \{x_1, ..., x_{n+m}\}$,

$$L(\theta \mid X_{n+m}, p) = \mathbb{E}_{Y \sim p} \left[\sum_{x \in X_{n+m}} \ln p(x, Y \mid \theta) \right]. \tag{2.49}$$

To be better than the supervised model θ^{sup} on the complete (transductive) likelihood (2.49) we would like to maximize the likelihood gain over it. So we want to find the θ that maximizes the likelihood gain

$$C(\theta, \theta^{\mathrm{SL}} \mid X_{n+m}, p) = L(\theta \mid X_{n+m}, p) - L(\theta^{\mathrm{SL}} \mid X_{n+m}, p). \tag{2.50}$$

We cannot maximize (2.50) directly, since we do not know the class true probability distribution p. We instead set $p(y_i \mid x_i) = 1$ for all labeled points $(x_i, y_i) \in S_n$ which gives us the vector $p_n = (p(1 \mid x_1), ..., p(1 \mid x_n))$ and for the unlabeled points X_m we consider a worst case, which leads to the following max-min formulation.

$$\theta^{\mathrm{SSL}} = \arg\max_{\theta \in \Theta} \min_{p_m \in [0,1]^m} C(\theta, \theta^{\mathrm{SL}} \mid X_{n+m}, (p_n, p_m)) \tag{2.51}$$

Note that the vector p_m can be the true labels Y_m of the unlabeled data X_m. Note also that $C(\theta^{\mathrm{SSL}}, \theta^{\mathrm{SL}} \mid X_{n+m}, (p_n, p_m)) \geq 0$ for all $p_m \in [0,1]^m$, so in particular if $p_m = Y_m$, as we can always chose $\theta^{SSL} = \theta^{\mathrm{SL}}$. That means that the following theorem holds.

Theorem 17 (Lemma 1)**.** *Let θ^{SSL} be a solution found in Equation (2.51), then*

$$L(\theta^{\mathrm{SL}} \mid X_{n+m}, Y_{n+m}) \leq L(\theta^{\mathrm{SSL}} \mid X_{n+m}, Y_{n+m}), \tag{2.52}$$

and for some specific choices for the model $p(x, y \mid \theta)$ the previous inequality is almost surely strict. So we are guaranteed that the transductive likelihood of our semi-supervised model is larger than of the supervised model.

An important difference between this work and the previous section is that for this paper one employs a generative model $p(x, y)$, while the SVM used by Li and Zhou [63] is a discriminative model that inherently optimizes the class probability $p(y \mid x)$. Krijthe and Loog [13], see also Subsection 2.3.1, show that to some degree it is actually necessary to use a generative model: The semi-supervised estimator of Equation (2.51) will coincide with the supervised estimator for a large class of discriminative models. There are several explanations why a joint model $p(x, y)$ helps out in the situation. The intuitive and obvious one is that the likelihood of this model takes the marginal distribution $P(X)$ into account, a quantity that can be measured from unlabeled data.

2.8. Discussion

We covered the main theoretical ideas and results that have been put forward over the past four decades in the field of semi-supervised learning. Specifically, we focused on results that inform us about its potential and the lack of such potential. We covered the answers to the questions: What are the limits of semi-supervised learning? What are the assumptions of different methods? What can we achieve if the assumptions are true? We like to wrap up our survey and mention a few realizations that, we think, get to the core of it.

2.8.1. On The Limits of Assumption Free SSL

In Section 2.3 we reviewed work that analyzes the limits of semi-supervised learning when no particular assumptions about the distribution are made, which a semi-supervised learner can exploit. The most general formulation of this is captured in Conjecture 1 and 2. They essentially state that a semi-supervised learner can beat all supervised learners by at most a constant. We then presented work that shows that the conjectures do not hold in full generality, but in particular situations. They essentially hold for the realizable case and hypothesis classes of finite VC-dimension, while they do not hold in the realizable or agnostic case for infinite VC-dimension. It remains to investigate the case of agnostic PAC-learning with a finite VC-dimension.

2.8.2. How Good Can Constant Improvement Be?

The question studied in Section 2.3.1 and the previous Subsection is whether a semi-supervised learner can offer more than a constant improvement, in terms of sample complexity. One can, however, also ask the question how good already a constant improvement can be in practice. The answer to that can be seen through a thought experiment. Assume that we have two classes given by two concentric d-dimensional spheres. Assume that we have enough unlabeled data for a manifold regularization scheme to identify the spheres. With this the semi-supervised learner needs only one labeled sample per class to give a perfect classification, while every supervised learner needs for good generalization a labeled sample size which increases in the dimension d. In this case manifold regularization is very effective even though we will see in the next chapter that, depending on the setting, it might only have a constant improvement in terms of sample complexity. This seems con-

tradictory, but recall that the constant can depend on the hypothesis class. If the supervised classifier uses a hypothesis class H, we can interpret manifold regularization as switching to a restricted space $\tilde{H}_\lambda$. This space only contains hypotheses that fulfill a manifold assumption, where the regularization parameter λ indicates to which degree this assumption is enforced. With this we can keep the VC-dimension of the restricted class fixed, while the VC-dimension of H will increase with the dimension. This in turn means that the constant improvement can be arbitrarily high. While this example uses the manifold assumption, Golovnev *et al.* [17] give a example with a semi-supervised learner that has the full knowledge of the domain distribution. We explain the particular example in Section 2.3.1. This shows that the constant improvement can be arbitrarily high if we have further assumptions, like the manifold assumption, or full knowledge of the marginal distribution. It is an open question if one can have arbitrarily high constants without assumptions and with limited unlabeled data.

2.8.3. THE AMOUNT OF UNLABELED DATA WE NEED

In Section 2.3.2 we presented three settings, in which a semi-supervised learner can PAC-learn, while no supervised learner can. For that we need, in principle, an infinite amount of unlabeled data and we also cannot create an example where that is not the case. If a fixed finite amount of unlabeled data would be enough to learn under any given distribution P we could just use the same strategy to learn in a supervised way as we can always chose to ignore the label. The way those examples work is that for each fixed P a finite amount of unlabeled data is sufficient, but this amount can be arbitrarily large. This has the consequence that if we want to learn over all possible distributions we need an arbitrarily large amount ($= \infty$) of unlabeled data. The improvements that semi-supervised learning can offer which we present in Sections 2.4, 2.5 and 2.6 do not necessarily need an infinite amount of unlabeled data, although it sometimes assumed for convenience. The difference is that in those settings supervised learner are also able to PAC-learn, but a semi-supervised learner is able to do this with fewer labeled samples. In Sections 2.6.2 and 2.6.3 we saw two instantiations of a cluster assumption, and the authors showed that the amount of unlabeled data needs to increase exponentially with the amount of labeled data to make use of this assumption. This is because the error in finding the clusters decreases only polynomially in the number of unlabeled points as shown in Inequality 2.32.

2.8.4. USING ASSUMPTIONS IN SEMI-SUPERVISED LEARNING

In Sections 2.5 and 2.6 we investigate what a semi-supervised learner can achieve once assumptions are made. A semi-supervised assumption is a link between the domain distribution and the labeling function. In particular we assume that we can ignore certain labeling functions after we have seen a specific domain distribution. The cluster assumption, for example, would exclude labeling functions that do not assign the same label to points belonging to the same cluster. The obvious, but real problem with this is that we do not know if such assumptions do hold or not. We speculate that testing if such an assumption is true or not consumes as many labeled points as learning directly a good classification rule with a supervised learner. To make this statement precise we define an assumption as a property of the distribution P on $\mathcal{X} \times \mathcal{Y}$. Let $\mathcal{P}^A$ be a set of distributions on $\mathcal{X} \times \mathcal{Y}$. We say that P fulfills assumption A iff $P \in \mathcal{P}^A$. For example $\mathcal{P}^A$ could only contain distributions such

that the marginal distributions $P(X)$ have always support on clusters, and each cluster has a unique label. Then P fulfills this particular cluster assumption A iff $P \in \mathcal{P}^A$. The crucial thing to note is, that the assumption A is a property on P, so we need labeled samples to test whether its true or not. It is thus of interest to compare the consumption of labeled data for reducing the uncertainty about the assumption to the consumption of labeled data for the convergence of the semi-supervised learner. We might of course know a priori that the assumption is true and do not need to test it, but what if not?

One of the few works that analyze this is reviewed in Section 2.6.4. Azizyan *et al.* [44] show that one can get essentially faster rates if the assumption is true, but we pay a penalty of $O(\frac{\ln(n)}{n})$ if it is not true. Balcan *et al.* [65] investigates how one can test for a property in an active way, so when we can choose which samples we want to label. The implications of this testing procedure for semi-supervised learning are, however, not clear. Of course, we may claim that it is not even necessary to test if the assumption is true or not, following Vapnik's principle: Why should we test if the assumption is true or not, when we are ultimately only interested whether the semi-supervised learner performs better or not? We believe that this is an important open question in semi-supervised learning.

2.9. Definitions

Definition 1. ***Supervised Sample Complexity*** *Given a learning problem (P, l, H) and $\epsilon, \delta > 0$ we define the sample complexity $m(B, H, P, \epsilon, \delta) \in \mathbb{N}$ of a supervised learner B as the smallest natural number k such that with probability at least $1 - \delta$ over all possible draws of a labeled sample S_k it holds that*

$$R(B(S_k)) - \inf_{h \in H} R(h) \leq \epsilon.$$

Or in short

$$m(B, H, P, \epsilon, \delta) = \{\min k \in \mathbb{N} \mid P\left(R(B(S_k)) - \inf_{h \in H} R(h) \leq \epsilon\right) \geq 1 - \delta\}.$$

Although not explicitly mentioned in the definition above, if B is semi-supervised it has additional input in form of either $P(X)$, or a random draw from it. Sometimes we drop the learner B from the sample complexity notation $m(B, H, P, \epsilon, \delta)$, and write either $m(H, P, \epsilon, \delta)$ or $m^{\text{SSL}}(H, P, \epsilon, \delta)$ if there exists a supervised or semi-supervised learner respectively that achieves the sample complexity.

Definition 2. ***Semi-Supervised Sample Complexity*** *Given a learning problem (P, l, H) and $\epsilon, \delta > 0$ we define the sample complexity $m^{\text{SSL}}(B, H, P, \epsilon, \delta) \in \mathbb{N}$ of a semi-supervised learner B, which has information about the marginal in the form of $U \in \{U_m, P(X)\}$, as the smallest natural number k such that with probability at least $1 - \delta$ over all possible draws of a labeled sample S_k it holds that*

$$R(B(S_k, U)) - \inf_{h \in H} R(h) \leq \epsilon.$$

Or in short

$$m^{\text{SSL}}(B, H, P, \epsilon, \delta) = \{\min k \in \mathbb{N} \mid P\left(R(B(S_k), U) - \inf_{h \in H} R(h) \leq \epsilon\right) \geq 1 - \delta\}.$$

symbol	explanation
$\mathcal{X}$	Feature space, for example $\mathcal{X} = \mathbb{R}^n$
$\mathcal{Y}$	Label space. Classification: $\mathcal{Y} = \{-1, 1\}$. Regression: $\mathcal{Y} = \mathbb{R}$.
P	Distribution on $\mathcal{X} \times \mathcal{Y}$
$\mathcal{P}$	A set of distributions on $\mathcal{X} \times \mathcal{Y}$
X, Y	Random variables distributed according to P
$P(X)$	Marginal distribution of P w.r.t to $\mathcal{X}$
$P(Y)$	Marginal distribution of P w.r.t to $\mathcal{Y}$
D	Domain distribution on $\mathcal{X}$
$\mathcal{D}$	A set of domain distributions on $\mathcal{X}$
$I_{\{\text{Boolean expression}\}}$	Indicator function (equals 1 if expression is true and 0 else)
$l(\hat{y}, y)$	Loss function, if not specified otherwise $l(\hat{y}, y) = I_{\hat{y}=y}$
H	Hypothesis class, where each $h \in H$ is a map $h : \mathcal{X} \to \mathcal{Y}$
$R(h)$	The risk of $h \in H$. Precisely: $R(h) = \mathbb{E}_{X,Y}[l(h(X), y)]$
(x_i, y_i)	A realization of (X, Y)
S_n	A labeled sample set of size n, $S_n = ((x_1, y_1), ..., (x_n, y_n))$
U_m	A unlabeled sample set of size m, usually $U_m = \{x_{n+1}, ..., x_{n+m}\}$
$\hat{R}_n(h) = \hat{R}(h)$	Empirical risk of h w.r.t S_n, $\hat{R}(h) = \frac{1}{n}\sum_{i=1}^{n} l(h(x_i), y_i)$
h^{SSL}	Model trained on S_n and U_m or $P(X)$, where $h^{\text{SSL}} : \mathcal{X} \to \mathcal{Y}$
h^{SL}	Model trained on S_n, where $h^{\text{SL}} : \mathcal{X} \to \mathcal{Y}$
$m(H, \epsilon, \delta)$	Supervised sample complexity, see Definition 1
$m^{\text{SSL}}(H, \epsilon, \delta)$	Semi-supervised sample complexity, see Definition 2

Table 2.1: Complete list of notations used in this chapter.

We usually drop the learner B from the sample complexity notation $m(B, H, P, \epsilon, \delta)$, and write either $m(H, P, \epsilon, \delta)$ or $m^{\text{SSL}}(H, P, \epsilon, \delta)$ if there exists respectively a supervised or semi-supervised learner that achieves this sample complexity. Similarly we drop the distribution P from the notation and write $m(H, \epsilon, \delta)$ or $m^{\text{SSL}}(H, \epsilon, \delta)$ if we can achieve this sample complexity for all distributions P.

REFERENCES

[1] S. Ben-David, T. Lu, and D. Pál, *Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning,* in *Proceedings of the The 21st Annual Conference on Learning Theory* (Helsinki, Finland, 2008).

[2] F. Cozman and I. Cohen, *Risks of semi-supervised learning: How unlabeled data can degrade performance of generative classifiers,* in *Semi-Supervised Learning* (The MIT Press, Cambridge, MA, USA, 2006) Chap. 4, pp. 57–72.

[3] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, New York, NY, USA, 2014).

[4] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning* (The MIT Press, Cambridge, MA, USA, 2012).

[5] V. N. Vapnik, *Statistical Learning Theory* (Wiley-Interscience, 1998).

[6] J. Peters, D. Janzing, and B. Schölkopf, *Elements of Causal Inference - Foundations and Learning Algorithms* (The MIT Press, Cambridge, MA, USA, 2017).

[7] M. Seeger, *Input-dependent Regularization of Conditional Density Models*, Tech. Rep. (Institute for Adaptive and Neural Computation, 2000).

[8] L. K. Hansen, *On bayesian transduction: Implications for the covariate shift problem,* in *Dataset Shift in Machine Learning* (The MIT Press, Cambridge, MA, USA, 2009) p. 65–72.

[9] T. Zhang and F. J. Oles, *A probability analysis on the value of unlabeled data for classification problems,* in *Proceedings of the 17th International Conference on Machine Learning* (Stanford, CA, USA, 2000).

[10] B. Schölkopf, D. Janzing, J. Peters, E. Sgouritsa, K. Zhang, and J. Mooij, *On causal and anticausal learning,* in *Proceedings of the 29th International Conference on Machine Learning* (Omnipress, New York, NY, USA, 2012) pp. 1255–1262.

[11] J. von Kügelgen, A. Mey, and M. Loog, *Semi-generative modelling: Covariate-shift adaptation with cause and effect features,* in *The 22nd International Conference on Artificial Intelligence and Statistics* (Okinawa, Japan, 2019) pp. 1361–1369.

[12] M. Loog, *Contrastive pessimistic likelihood estimation for semi-supervised classification.* IEEE Transactions on Pattern Analysis and Machine Intelligence **38**, 462 (2016).

[13] J. Krijthe and M. Loog, *The pessimistic limits of margin-based losses in semi-supervised learning,* in *Advances in Neural Information Processing Systems 31* (Montreal, Canada, 2018) pp. 1795–1804.

[14] J. D. Lafferty and L. A. Wasserman, *Statistical analysis of semi-supervised regression.* in *Advances in Neural Information Processing Systems 20* (Curran Associates, Inc., 2007) pp. 801–808.

[15] P. J. Bickel and B. Li, *Local polynomial regression on unknown manifolds,* in *Complex Datasets and Inverse Problems*, Vol. Volume 54 (Institute of Mathematical Statistics, Beachwood, Ohio, USA, 2007) pp. 177–186.

[16] M. Darnstädt, H. U. Simon, and B. Szörényi, *Unlabeled data does provably help.* in *Symposium on Theoretical Aspects of Computer Science*, Vol. 20 (Kiel, Germany, 2013) pp. 185–196.

[17] A. Golovnev, D. Pál, and B. Szörényi, *The information-theoretic value of unlabeled data in semi-supervised learning,* in *Proceedings of the 36th International Conference on Machine Learning* (Long Beach, California, USA, 2019) pp. 2328–2336.

[18] C. Göpfert, S. Ben-David, O. Bousquet, S. Gelly, I. O. Tolstikhin, and R. Urner, *When can unlabeled data improve the learning rate?* in *Conference on Learning Theory 2019* (Phoenix, AZ, USA, 2019) pp. 1500–1518.

[19] A. Globerson, R. Livni, and S. Shalev-Shwartz, *Effective semisupervised learning on manifolds.* in *Conference on Learning Theory 2018* (Amsterdam, The Netherlands, 2017) pp. 978–1003.

[20] P. Niyogi, *Manifold regularization and semi-supervised learning: some theoretical analyses.* Journal of Machine Learning Research **14**, 1229 (2013).

[21] M. Belkin, P. Niyogi, and V. Sindhwani, *Manifold regularization: A geometric framework for learning from labeled and unlabeled examples,* Journal of Machine Learning Research **7**, 2399 (2006).

[22] N. Sokolovska, O. Cappé, and F. Yvon, *The asymptotics of semi-supervised learning in discriminative probabilistic models.* in *Proceedings of the 25th International Conference on Machine Learning*, Vol. 307 (Helsinki, Finland, 2008) pp. 984–991.

[23] M. Kääriäinen, *Generalization error bounds using unlabeled data.* in *18th Annual Conference on Learning Theory* (Springer, Bertinoro, Italy, 2005) pp. 127–142.

[24] B. Leskes, *The value of agreement, a new boosting algorithm,* in *Proceedings of the 18th Conference on Learning Theory* (Bertinoro, Italy, 2005).

[25] M. Kawakita and T. Kanamori, *Semi-supervised learning with density-ratio estimation.* Machine Learning **91**, 189 (2013).

[26] A. B. Tsybakov, *Optimal aggregation of classifiers in statistical learning,* The Annals of Statistics **32**, 135 (2004).

[27] D. A. McAllester, *Pac-bayesian stochastic model selection,* Machine Learning **51**, 5 (2003).

[28] J. Langford and J. Shawe-Taylor, *Pac-bayes & margins,* in *Advances in Neural Information Processing Systems 15* (Vancouver, British Columbia, Canada, 2002) pp. 439–446.

[29] M.-F. Balcan and A. Blum, *A discriminative model for semi-supervised learning.* Journal of the ACM **57**, 19:1 (2010).

[30] K. Sridharan and S. M. Kakade, *An information theoretic framework for multi-view learning.* in *21st Annual Conference on Learning Theory* (Helsinki, Finland, 2008) pp. 403–414.

[31] T. Joachims, *Transductive inference for text classification using support vector machines,* in *Proceedings of the Sixteenth International Conference on Machine Learning* (San Francisco, CA, USA, 1999) pp. 200–209.

[32] S. Boyd and L. Vandenberghe, *Convex Optimization* (Cambridge University Press, New York, NY, USA, 2004).

[33] A. Blum and T. Mitchell, *Combining labeled and unlabeled data with co-training,* in *Proceedings of the 11th Annual Conference on Computational Learning Theory* (Madison, Wisconsin, USA, 1998) pp. 92–100.

[34] A. Blum and S. Chawla, *Learning from labeled and unlabeled data using graph mincuts,* in *Proceedings of the 18th International Conference on Machine Learning* (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001) pp. 19–26.

[35] D. S. Rosenberg and P. L. Bartlett, *The rademacher complexity of co-regularized kernel classes,* in *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics* (San Juan, Puerto Rico, 2007) pp. 396–403.

[36] J. Farquhar, D. Hardoon, H. Meng, J. S. Shawe-taylor, and S. Szedmák, *Two view learning: Svm-2k, theory and practice,* in *Advances in Neural Information Processing Systems 18* (MIT Press, 2006) pp. 355–362.

[37] V. Sindhwani and D. S. Rosenberg, *An rkhs for multi-view learning and manifold co-regularization,* in *Proceedings of the 25th International Conference on Machine Learning* (Helsinki, Finland, 2008) pp. 976–983.

[38] V. Castelli and T. M. Cover, *On the exponential value of labeled samples.* Pattern Recognition Letters **16**, 105 (1995).

[39] V. Castelli and T. M. Cover, *The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter.* IEEE Transactions on Information Theory **42**, 2102 (1996).

[40] K. Sinha and M. Belkin, *The value of labeled and unlabeled examples when the model is imperfect.* in *Advances in Neural Information Processing Systems 20* (Vancouver, British Columbia, Canada, 2007) pp. 1361–1368.

[41] J. Ratsaby and S. S. Venkatesh, *Learning from a mixture of labeled and unlabeled examples with parametric side information.* in *Proceedings of the 8th Annual Conference on Computational Learning Theory* (Santa Cruz, CA, USA, 1995) pp. 412–417.

[42] P. Rigollet, *Generalization error bounds in semi-supervised classification under the cluster assumption.* Journal of Machine Learning Research **8**, 1369 (2007).

[43] A. Singh, R. D. Nowak, and X. Zhu, *Unlabeled data: Now it helps, now it doesn't.* in *Advances in Neural Information Processing Systems 21* (Vancouver, British Columbia, Canada, 2008) pp. 1513–1520.

[44] M. Azizyan, A. Singh, and L. A. Wasserman, *Density-sensitive semisupervised inference,* Computing Research Repository **abs/1204.1685** (2012).

[45] D. Pechyony, *Theory and Practice of Transductive Learning*, Ph.D. thesis, Isreal Institute of Technology (2008).

[46] V. Vapnik, *Estimation of Dependences Based on Empirical Data* (Springer-Verlag, Berlin, Heidelberg, 1982).

[47] P. Derbeko, R. El-Yaniv, and R. Meir, *Explicit learning curves for transduction and application to clustering and compression algorithms,* Journal of Artificial Intelligence Research **22**, 117 (2004).

[48] D. McAllester, *Simplified pac-bayesian margin bounds,* in *Learning Theory and Kernel Machines* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2003) pp. 203–215.

[49] D. A. McAllester, *Pac-bayesian stochastic model selection,* Machine Learning **51**, 5 (2003).

[50] G. M. Benedek and A. Itai, *Learnability with respect to fixed distributions,* Theoretical Computer Science **86**, 377 (1991).

[51] R. El-Yaniv and D. Pechyony, *Stable transductive learning.* in *19th Annual Conference on Learning Theory*, Vol. 4005 (Pittsburgh, PA, USA, 2006) pp. 35–49.

[52] O. Bousquet and A. Elisseeff, *Stability and generalization,* Journal of Machine Learning Research **2**, 499 (2002).

[53] S. Kutin and P. Niyogi, *Almost-everywhere algorithmic stability and generalization error,* in *Proceedings of the 18th Conference in Uncertainty in Artificial Intelligence* (Edmonton, Alberta, Canada, 2002) pp. 275–282.

[54] S. Kutin, *Extensions to McDiarmid's inequality when differences are bounded with high probability*, Tech. Rep. (University of Chicago, 2002).

[55] R. Johnson and T. Zhang, *On the effectiveness of laplacian normalization for graph semi-supervised learning,* Journal of Machine Learning Research **8**, 1489 (2007).

[56] P. L. Bartlett, O. Bousquet, and S. Mendelson, *Local rademacher complexities,* The Annals of Statistics **33**, 1497 (2005).

[57] R. El-Yaniv and D. Pechyony, *Transductive rademacher complexity and its applications,* Journal of Artificial Intelligence Research **35**, 193 (2009).

[58] Y. Maximov, M.-R. Amini, and Z. Harchaoui, *Rademacher complexity bounds for a penalized multiclass semi-supervised algorithm.* Computing Research Repository **abs/1607.00567** (2016).

[59] P. L. Bartlett and S. Mendelson, *Rademacher and gaussian complexities: Risk bounds and structural results,* Journal of Machine Learning Research **3**, 463 (2003).

[60] G. R. G. Lanckriet, N. Cristianini, P. L. Bartlett, L. E. Ghaoui, and M. I. Jordan, *Learning the kernel matrix with semidefinite programming.* Journal of Machine Learning Research **5**, 27 (2004).

[61] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning* (The MIT Press, Cambridge, MA, USA, 2006).

[62] R. Johnson and T. Zhang, *Graph-based semi-supervised learning and spectral kernel design,* IEEE Transactions on Information Theory **54**, 275 (2008).

[63] Y.-F. Li and Z.-H. Zhou, *Towards making unlabeled data never hurt.* in *Proceedings of the 28th International Conference on Machine Learning* (Bellevue, Washington, USA, 2011) pp. 1081–1088.

[64] K. P. Bennett and A. Demiriz, *Semi-supervised support vector machines,* in *Advances in Neural Information Processing Systems 11* (Denver, CO, USA, 1999) pp. 368–374.

[65] M. Balcan, E. Blais, A. Blum, and L. Yang, *Active property testing,* in *53rd Annual IEEE Symposium on Foundations of Computer Science* (New Brunswick, NJ, USA, 2012) pp. 21–30.

3

MANIFOLD REGULARIZATION

Manifold regularization is a commonly used technique in semi-supervised learning. It enforces the classification rule to be smooth with respect to the data-manifold. Here, we derive sample complexity bounds based on pseudo-dimension for models that add a convex data dependent regularization term to a supervised learning process, as is in particular done in Manifold regularization. We then compare the bound for those semi-supervised methods to purely supervised methods, and discuss a setting in which the semi-supervised method can only have a constant improvement, ignoring logarithmic terms. By viewing Manifold regularization as a kernel method we then derive Rademacher bounds which allow for a distribution dependent *analysis. Finally we illustrate that these bounds may be useful for choosing an appropriate manifold regularization parameter in situations with very sparsely labeled data.*

3.1. INTRODUCTION

In many applications, as for example image or text classification, gathering unlabeled data is easier than gathering labeled data. Semi-supervised methods try to extract information from the unlabeled data to get improved classification results over purely supervised methods. A well-known technique to incorporate unlabeled data into a learning process is manifold regularization (MR) [1, 2]. This procedure adds a data-dependent penalty term to the loss function that penalizes classification rules that behave non-smooth with respect to the data distribution. This chapter presents a sample complexity and a Rademacher complexity analysis for this procedure. In addition it illustrates how our Rademacher complexity bounds may be used for choosing a suitable Manifold regularization parameter.

We organize this chapter as follows. In Sections 3.2 and 3.3 we discuss related work and introduce the semi-supervised setting. In Section 3.4 we formalize the idea of adding a distribution-dependent penalty term to a loss function. Algorithms such as manifold, entropy or co-regularization [1, 3, 4] follow this idea. Our formalization of this idea is inspired by Balcan and Blum [5] and allows for a similar sample complexity analysis. Section 3.5 reviews the work from Balcan and Blum [5] and generalizes a bound from their paper. We use this to derive sample complexity bounds for the proposed framework, and thus in particular for MR. For the specific case of regression, we furthermore adapt a sample complexity bound from Anthony and Bartlett [6], which is essentially tighter than the first bound, to the semi-supervised case. In the same section we sketch a setting in which we show that if our hypothesis set has finite pseudo-dimension, and we ignore logarithmic factors, any semi-supervised learner (SSL) that falls in our framework has at most a constant improvement in terms of sample complexity. This and related behavior has been observed and investigated before [7, 8] for assumption free SSL and we relate our results to this previous work. In Section 3.6 we show how one can obtain distribution *dependent* complexity bounds for MR. We review a kernel formulation of MR [9] and show how this can be used to estimate Rademacher complexities for *specific* datasets. In Section 3.7 we illustrate on an artificial dataset how the distribution dependent bounds could be used for choosing the regularization parameter of MR. This is particularly useful as the analysis does not need an additional labeled validation set. The practicality of this approach requires further empirical investigation. In Section 3.8 we discuss our results and speculate about possible extensions.

3.2. RELATED WORK

There are currently two related analyses of MR that show, to some extent, that a SSL can learn efficiently if it knows the true underlying manifold, while a fully supervised learner may not. In [10] we find an investigation of a setting where distributions on the input space $\mathcal{X}$ are restricted to ones that correspond to unions of irreducible algebraic sets of a fixed size $k \in \mathbb{N}$, and each algebraic set is either labeled 0 or 1. A SSL that knows the true distribution on $\mathcal{X}$ can identify the algebraic sets and reduce the hypothesis space to all 2^k possible label combinations on those sets. As we are left with finitely many hypotheses we can learn them efficiently, while they show that every supervised learner is left with a hypothesis space of infinite VC dimension.

The work in [2] considers manifolds that arise as embeddings from a circle, where the labeling over the circle is (up to the decision boundary) smooth. They then show that a

learner that has knowledge of the manifold can learn efficiently while for every fully supervised learner one can find an embedding and a distribution for which this is not possible.

The relation to this chapter is as follows. They provide specific examples where the sample complexity between a semi-supervised and a supervised learner are infinitely large, while we explore general sample complexity bounds of MR and sketch a setting in which MR can not essentially improve over supervised methods.

3.3. THE SEMI SUPERVISED SETTING

We work in the statistical learning framework: we assume we are given a feature domain $\mathcal{X}$ and an output space $\mathcal{Y}$ together with an unknown probability distribution P over $\mathcal{X} \times \mathcal{Y}$. In binary classification we usually have that $\mathcal{Y} = \{-1, 1\}$, while for regression $\mathcal{Y} = \mathbb{R}$. We use a loss function $\phi : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$, which is convex in the first argument and in practice usually a surrogate for the 0-1 loss in classification, and the squared loss in regression tasks. A hypothesis f is a function $f : \mathcal{X} \to \mathbb{R}$. We set (X, Y) to be a random variable distributed according to P, while small x and y are elements of $\mathcal{X}$ and $\mathcal{Y}$ respectively. Our goal is to find a hypothesis f, within a restricted class $\mathcal{F}$, such that the expected loss $Q(f) := \mathbb{E}[\phi(f(X), Y)]$ is small. In the standard supervised setting we choose a hypothesis f based on an i.i.d. sample $S_n = \{(x_i, y_i)\}_{i \in \{1,..,n\}}$ drawn from P. With that we define the empirical risk of a model $f \in \mathcal{F}$ with respect to ϕ and measured on the sample S_n as $\hat{Q}(f, S_n) = \frac{1}{n} \sum_{i=1}^{n} \phi(f(x_i), y_i)$. For ease of notation we sometimes omit S_n and just write $\hat{Q}(f)$. Given a learning problem defined by $(P, \mathcal{F}, \phi)$ and a labeled sample S_n, one way to choose a hypothesis is by the empirical risk minimization principle

$$f_{\text{sup}} = \arg\min_{f \in \mathcal{F}} \hat{Q}(f, S_n). \tag{3.1}$$

We refer to f_{sup} as the *supervised solution*. In SSL we additionally have samples with unknown labels. So we assume to have $n + m$ samples $(x_i, y_i)_{i \in \{1,..,n+m\}}$ independently drawn according to P, where y_i has not been observed for the last m samples. We furthermore set $U = \{x_1, ..., x_{x_{n+m}}\}$, so U is the set that contains all our available information about the feature distribution.

Finally we denote by $m^L(\epsilon, \delta)$ the sample complexity of an algorithm L. That means that for all $n \geq m^L(\epsilon, \delta)$ and all possible distributions P the following holds. If L outputs a hypothesis f_L after seeing an n-sample, we have with probability of at least $1 - \delta$ over the n-sample S_n that $Q(f_L) - \min_{f \in \mathcal{F}} Q(f) \leq \epsilon$.

3.4. A FRAMEWORK FOR SEMI-SUPERVISED LEARNING

We follow the work of Balcan and Blum [5] and introduce a second convex loss function $\psi : \mathcal{F} \times \mathcal{X} \to \mathbb{R}_+$ that only depends on the input feature and a hypothesis. We refer to ψ as the *unsupervised loss* as it does not depend on any labels. We propose to *add* the unlabeled data through the loss function ψ and add it as a penalty term to the supervised loss to obtain the semi-supervised solution

$$f_{\text{semi}} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \phi(f(x_i), y_i) + \lambda \frac{1}{n+m} \sum_{j=1}^{n+m} \psi(f, x_j), \tag{3.2}$$

where $\lambda > 0$ controls the trade-off between the supervised and the unsupervised loss. This is in contrast to [5], as they use the unsupervised loss to restrict the hypothesis space directly. In the following section we recall the important insight that those two formulations are equivalent in some scenarios and we can use [5] to generate sample complexity bounds for the here presented SSL framework.

For ease of notation we set $\hat{R}(f,U) = \frac{1}{n+m}\sum_{j=1}^{n+m}\psi(f,x_j)$ and $R(f) = \mathbb{E}[\psi(f,X)]$. We do not claim any novelty for the idea of adding an unsupervised loss for regularization. A different framework can be found in Chapelle *et al.* [11, Chapter 10]. We are, however, not aware of a deeper analysis of this particular formulation, as done for example by the sample complexity analysis in this chapter. As we are in particular interested in the class of MR schemes we first show that this method a fits our framework.

Example: Manifold Regularization Overloading the notation we write now $P(X)$ for the distribution P restricted to $\mathcal{X}$. In MR one assumes that the input distribution $P(X)$ has support on a compact manifold $M \subset \mathcal{X}$ and that the predictor $f \in \mathcal{F}$ varies smoothly in the geometry of M [1]. There are several regularization terms that can enforce this smoothness, one of which is $\int_M ||\nabla_M f(x)||^2 dP(x)$, where $\nabla_M f$ is the gradient of f along M. We know that $\int_M ||\nabla_M f(x)||^2 dP(x)$ may be approximated with a finite sample of $\mathcal{X}$ drawn from $P(X)$ [12]. Given such a sample $U = \{x_1, ..., x_{n+m}\}$ one defines first a weight matrix W, where $W_{ij} = e^{-||x_i - x_j||^2/\sigma}$. We set L then as the Laplacian matrix $L = D - W$, where D is a diagonal matrix with $D_{ii} = \sum_{j=1}^{n+m} W_{ij}$. Let furthermore $f_U = (f(x_1), ..., f(x_{n+m}))^t$ be the evaluation vector of f on U. The expression $\frac{1}{(n+m)^2} f_U^t L f_U = \frac{1}{(n+m)^2}\sum_{i,j}(f(x_i) - f(x_j))^2 W_{ij}$ converges to $\int_M ||\nabla_M f||^2 dP(x)$ under certain conditions [12]. This motivates us to set the unsupervised loss as $\psi(f,(x_i,x_j)) = (f(x_i) - f(x_j))^2 W_{ij}$, and this is indeed a convex function in f.

3.5. ANALYSIS OF THE FRAMEWORK

In this section we analyze the properties of the solution f_{semi} found in Equation (3.2). We derive sample complexity bounds for this procedure, using results from [5], and compare them to sample complexities for the supervised case. In [5] the unsupervised loss is used to restrict the hypothesis space directly, while we use it as a regularization term in the empirical risk minimization as usually done in practice. To switch between the views of a constrained optimization formulation and our formulation (3.2) we use the following classical result from convex optimization [13, Theorem 1].

Lemma 1. *Let $\phi(f(x), y)$ and $\psi(f, x)$ be functions convex in f for all x, y. Then the following two optimization problems are equivalent:*

$$\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\phi(f(x_i), y_i) + \lambda\frac{1}{n+m}\sum_{i=1}^{n+m}\psi(f, x_i) \tag{3.3}$$

$$\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\phi(f(x_i), y_i) \quad \textit{subject to} \quad \sum_{i=1}^{n+m}\frac{1}{n+m}\psi(f, x_i) \leq \tau \tag{3.4}$$

Where equivalence means that for each λ we can find a τ such that both problems have the same solution and vice versa.

For our later results we will need the conditions of this lemma are true, which we believe to be not a strong restriction. In our sample complexity analysis we stick as close as possible to the actual formulation and implementation of MR, which is usually a convex optimization problem. We now first turn to our sample complexity bounds.

The next subsection introduces the sample complexity bound and shows how it can be used to give theoretical guarantees for the presented framework.

3.5.1. SAMPLE COMPLEXITY BOUNDS

Sample complexity bounds for supervised learning use typically a notion of complexity of the hypothesis space to bound the worst case difference between the estimated and the true risk. As our hypothesis class allows for real-valued functions, we will use the notion of pseudo-dimension $\text{Pdim}(\mathcal{F}, \phi)$, an extension of the VC-dimension to real valued loss functions ϕ and hypotheses classes $\mathcal{F}$ [14, 15]. Informally speaking, the pseudo-dimension is the VC-dimension of the set of functions that arise when we threshold real-valued functions to define binary functions. Note that sometimes the pseudo-dimension will have as input the loss function, and sometimes not. This is because some results use the concatenation of loss function and hypotheses to determine the capacity, while others only use the hypotheses class. This lets us state our first main result, which is a generalization of [5, Theorem 10] to bounded loss functions and real valued function spaces.

Theorem 18. *Let $\mathcal{F}_\tau^\psi := \{f \in \mathcal{F} \mid \mathbb{E}[\psi(f,x)] \leq \tau\}$. Assume that ϕ, ψ are measurable loss functions such that there exists constants $B_1, B_2 > 0$ with $\psi(f,x) \leq B_1$ and $\phi(f(x),y) \leq B_2$ for all x, y and $f \in \mathcal{F}$ and let P be a distribution. Furthermore let $f_\tau^* = \arg\min_{f \in \mathcal{F}_\tau^\psi} Q(f)$. Then an unlabeled sample U of size*

$$m \geq \frac{8{B_1}^2}{\epsilon^2}\left[\ln\frac{16}{\delta} + 2\,\text{Pdim}(\mathcal{F},\psi)\ln\frac{4B_1}{\epsilon} + 1\right]$$

and a labeled sample S_n of size

$$n \geq \max\left(\frac{8{B_2}^2}{\epsilon^2}\left[\ln\frac{8}{\delta} + 2\,\text{Pdim}(\mathcal{F}^\psi_{\tau+\frac{\epsilon}{2}},\phi)\ln\frac{4B_2}{\epsilon} + 1\right], \frac{h}{4}\right)$$

is sufficient to ensure that with probability at least $1-\delta$ the classifier $g \in \mathcal{F}$ that minimizes $\hat{Q}(\cdot, S_n)$ subject to $\hat{R}(\cdot, U) \leq \tau + \frac{\epsilon}{2}$ satisfies

$$Q(g) \leq Q(f_\tau^*) + \epsilon. \tag{3.5}$$

Proof. The result will be shown by combining three partial results with the union bound. First we show that the unlabeled sample size is big enough to guarantee that with probability at least $1 - \frac{\delta}{4}$ it holds that $\hat{R}(f_\tau^*) \leq \tau + \frac{\epsilon}{2}$. For $h = \text{Pdim}(\mathcal{F}, \psi)$ Theorem 5.1 from [14] states that

$$P\left[\sup_{f\in\mathcal{F}}(\hat{R}(f) - R(f)) > \frac{\epsilon}{2}\right] \leq 4e^{h(\ln\frac{2m}{h}+1) - \frac{m}{{B_1}^2}(\frac{\epsilon}{2}-\frac{1}{m})^2}.$$

Bounding

$$4e^{h(\ln\frac{2m}{h}+1) - \frac{m}{{B_1}^2}(\frac{\epsilon}{2}-\frac{1}{m})^2} \leq \frac{\delta}{4}$$

and rewriting this gives us that

$$m \geq \frac{4{B_1}^2}{\epsilon^2}\left[\ln\frac{16}{\delta} + h\ln\frac{2em}{h}\right] = \frac{4{B_1}^2}{\epsilon^2}\left[\ln\frac{16}{\delta} + h\ln m + h\ln\frac{2e}{h} + 1\right]$$

is sufficient to ensure that $\hat{R}(f) - R(f) < \frac{\epsilon}{2}$ for all $f \in \mathcal{F}$ with probability at least $1 - \frac{\delta}{4}$. Using the inequality $\ln x \leq \alpha x - \ln\alpha - 1$ with $x = m$ and $\alpha = \frac{\epsilon^2}{8h{B_1}^2}$ we can conclude that a sample of size

$$\begin{aligned} m &\geq \frac{4{B_1}^2}{\epsilon^2}\left[\ln\frac{16}{\delta} + h(\frac{\epsilon^2}{8h{B_1}^2}m + \ln\frac{8h{B_1}^2}{\epsilon^2} - 1) + h\ln\frac{2e}{h} + 1\right] \\ &= \frac{m}{2} + \frac{4{B_1}^2}{\epsilon^2}\left[\ln\frac{16}{\delta} + h\ln\frac{16{B_1}^2}{\epsilon^2} + 1\right] \\ &\iff \\ m &\geq \frac{8{B_1}^2}{\epsilon^2}\left[\ln\frac{16}{\delta} + 2h\ln\frac{4B_1}{\epsilon} + 1\right] \end{aligned}$$

is sufficient to guarantee $\hat{R}(f) - R(f) < \frac{\epsilon}{2}$ for all $f \in \mathcal{F}$ with probability at least $1 - \frac{\delta}{4}$. In particular choosing $f = f_\tau^*$ and noting that by definition $R(f_\tau^*) \leq \tau$ we conclude that with the same probability

$$\hat{R}(f_\tau^*) \leq \tau + \frac{\epsilon}{2}. \tag{3.6}$$

For the second part we use the classical Hoeffding inequality with a labeled sample size of n

$$P\left[\hat{Q}(f_\tau^*) - Q(f_\tau^*) \geq \theta\right] \leq e^{\frac{-2\theta^2 n}{{B_2}^2}}.$$

Choosing $\theta = B_2\sqrt{\ln(\frac{4}{\delta})\frac{1}{2n}}$ lets us conclude that with probability at least $1 - \frac{\delta}{4}$ it holds that

$$\hat{Q}(f_\tau^*) \leq Q(f_\tau^*) + B_2\sqrt{\ln(\frac{4}{\delta})\frac{1}{2n}}. \tag{3.7}$$

For the third part we use again Theorem 5.1 from [14] with $h = \text{Pdim}(\mathcal{F}_\tau^\psi, \phi)$, which states that

$$n \geq \frac{4{B_2}^2}{\epsilon^2}\left[\ln\frac{8}{\delta} + h\ln\frac{2en}{h} + 1\right] \tag{3.8}$$

is sufficient to guarantee with probability at least $1 - \frac{\delta}{2}$ that

$$Q(f) - \hat{Q}(f) \leq \frac{\epsilon}{2} \quad \text{for all} \quad f \in \mathcal{F}_{\tau+\frac{\epsilon}{2}}^\psi. \tag{3.9}$$

With the same reasoning as for the first part we obtain the same guarantee with a labeled sample of size

$$n \geq \frac{8{B_2}^2}{\epsilon^2}\left[\ln\frac{8}{\delta} + 2h\ln\frac{4B_2}{\epsilon} + 1\right].$$

Putting everything together with we get, using the union bound, that with probability $1-\delta$ the classifier g that minimizes $\hat{Q}(\cdot, X, Y)$ subject to $\hat{R}(\cdot, U) \leq \tau + \frac{\epsilon}{2}$ satisfies

$$Q(g) \leq \hat{Q}(g) + \frac{\epsilon}{2} \leq \hat{Q}(f_\tau^*) + \frac{\epsilon}{2} \leq Q(f_\tau^*) + \frac{\epsilon}{2} + B_2\sqrt{\frac{\ln(\frac{4}{\delta})}{2n}}.$$

The first inequality follows from Inequality (3.9). The second inequality follows because g is the empirical minimizer. Note that we also need Inequality (3.6), i.e. that $\hat{R}(f_\tau^*) \leq \tau + \frac{\epsilon}{2}$, to make sure that f_τ^* was in the search space. The third inequality follows from Inequality (3.7). To obtain the final inequality we use the labeled sample size to show that

$$\frac{\epsilon}{2} \geq \sqrt{\frac{{B_2}^2}{n}\left[\ln\frac{8}{\delta} + h\ln\frac{2en}{h} + 1\right]} \geq B_2\sqrt{\frac{\ln(\frac{4}{\delta})}{2n}}.$$

The first inequality holds by assumption of the labeled sample size from Inequality (3.8), while the second inequality is shown by reducing it to

$$h\ln\frac{2en}{h} + 1 \geq \frac{1}{2}\ln(\frac{1}{2})$$

which holds as the right-hand side is negative, while the left-hand side is positive as $2en > h$ since by our assumptions $4n > h$.

□

The next subsection uses this theorem to derive sample complexity bounds for MR. First, however, a remark about the assumption that the loss function ϕ is globally bounded. If we assume that $\mathcal{F}$ is a reproducing kernel Hilbert space there exists an $M > 0$ such that for all $f \in \mathcal{F}$ and $x \in \mathcal{X}$ it holds that $|f(x)| \leq M||f||_{\mathcal{F}}$. If we restrict the norm of f by introducing a regularization term with respect to the norm $||.||_{\mathcal{F}}$, we know that the image of $\mathcal{F}$ is globally bounded. If the image is also closed it will be compact, and thus ϕ will be globally bounded in many cases, as most loss functions are continuous. This can also be seen as a justification to also use an intrinsic regularization for the norm of f in addition to the regularization by the unsupervised loss, as only then the guarantees of Theorem 18 apply. Using this bound together with Lemma 1 we can state the following corollary to give a PAC-style guarantee for our proposed framework.

Corollary 2. *Let ϕ and ψ be convex supervised and an unsupervised loss function that fulfill the assumptions of Theorem 18. Then f_{semi} (3.2) satisfies the guarantees given in Theorem 18, when we replace for it g in Inequality (3.5).*

Recall that in the MR setting $\hat{R}(f) = \frac{1}{(n+m)^2}\sum_{i=1}^{n+m} W_{ij}(f(x_i) - f(x_j))^2$. So we gather unlabeled samples from $\mathcal{X} \times \mathcal{X}$ instead of $\mathcal{X}$. Collecting m samples from $\mathcal{X}$ equates $m^2 - 1$ samples from $\mathcal{X} \times \mathcal{X}$ and thus we only need $\sqrt{m}$ instead of m unlabeled samples for the same bound.

3.5.2. COMPARISON TO THE SUPERVISED SOLUTION

In the SSL community it is well-known that using SSL does not come without a risk [11, Chapter 4]. Thus it is of particular interest how those methods compare to purely supervised schemes. There are, however, many potential supervised methods we can think of. In many works this problem is avoided by comparing to all possible supervised schemes [7, 8, 10]. The framework introduced in this chapter allows for a more fine-grained analysis as the semi-supervision happens on top of an already existing supervised methods. Thus, for our framework, it is natural to compare the sample complexities of f_{sup} with the sample complexity of f_{semi}. To compare the supervised and semi-supervised solution we draw from Anthony and Bartlett [6, Chapter 20], where one can find lower and upper sample complexity bounds for the regression setting. To use this we have to restrict to the square loss, so in this section we set $\phi(f(x), y) = (f(x) - y)^2$. The main insight from [6, Chapter 20] is that the sample complexity depends in this setting on whether the hypothesis class is (closure) convex or not. As we anyway need convexity of the space, which is stronger than closure convexity, to use Lemma 1, we can adapt Theorem 20.7 from [6] to our semi-supervised setting.

Theorem 19. *Assume that $\mathcal{F}^{\psi}_{\tau+\epsilon}$ is a closure convex class with functions mapping to $[0,1]$[1], that $\psi(f,x) \leq B_1$ for all $x \in \mathcal{X}$ and $f \in \mathcal{F}$ and that $\phi(f(x),y) = (f(x)-y)^2$. Assume further that there is a $B_2 > 0$ such that $(f(x)-y)^2 < B_2$ almost surely for all $(x,y) \in \mathcal{X} \times \mathcal{Y}$ and $f \in \mathcal{F}^{\psi}_{\tau+\epsilon}$. Then an unlabeled sample size of*

$$m \geq \frac{2{B_1}^2}{\epsilon^2}\left[\ln\frac{8}{\delta} + 2\,\text{Pdim}(\mathcal{F},\psi)\ln\frac{2B_1}{\epsilon} + 2\right]$$

and a labeled sample size of

$$n \geq \mathcal{O}\left(\frac{B^2}{\epsilon}\left(\text{Pdim}(\mathcal{F}^{\psi}_{\tau+\epsilon})\ln\frac{\sqrt{B}}{\epsilon} + \ln\frac{2}{\delta}\right)\right) \tag{3.10}$$

is sufficient *to guarantee that with probability at least $1-\delta$ the classifier g that minimizes $\hat{Q}(\cdot)$ w.r.t $\hat{R}(f) \leq \tau + \epsilon$ satisfies*

$$Q(g) \leq \min_{f \in \mathcal{F}^{\psi}_{\tau}} Q(f) + \epsilon. \tag{3.11}$$

Proof. As in the proof of Theorem 18 the unlabeled sample size is sufficient to guarantee with probability at least $1 - \frac{\delta}{2}$ that $R(f^*_\tau) \leq \tau + \epsilon$. The labeled sample size is big enough to guarantee with at least $1 - \frac{\delta}{2}$ that $Q(g) \leq Q(f^*_{\tau+\epsilon}) + \epsilon$ [6, Theorem 20.7]. Using the union bound we have with probability of at least $1-\delta$ that $Q(g) \leq Q(f^*_{\tau+\epsilon}) + \epsilon \leq Q(f^*_\tau) + \epsilon$. □

Note that the previous theorem of course implies the same learning rate in the supervised case, as the only difference will be the pseudo-dimension term. As in specific scenarios this is also the best possible learning rate, we obtain the following negative result for SSL.

[1] In the remarks after Theorem 18 we argue that in many cases |f(x)| is bounded, and in those cases we can always map to [0,1] by re-scaling.

Corollary 3. *Assume that $\mathcal{F}$ maps to the interval $[0,1]$ and $\mathcal{Y} = [1-B, B]$ for a $B \geq 2$. If $\mathcal{F}$ and $\mathcal{F}_\tau^\psi$ are both closure convex, then for sufficiently small $\epsilon, \delta > 0$ it holds that $m^{sup}(\epsilon,\delta) = \tilde{\mathcal{O}}(m^{semi}(\epsilon,\delta))$, where $\tilde{\mathcal{O}}$ suppresses logarithmic factors, and m^{semi}, m^{sup} denote the sample complexity of the semi-supervised and the supervised learner respectively. In other words, the semi-supervised method can improve the learning rate by at most a constant which may depend on the pseudo-dimensions, ignoring logarithmic factors. Note that this holds in particular for the manifold regularization algorithm.*

Proof. The assumptions made in the theorem allow is to invoke Equation (19.5) from [6] which states that $m^{\text{semi}} = \Omega(\frac{1}{\epsilon} + \text{Pdim}(\mathcal{F}_\tau^\psi))$.[2] Using Inequality (3.10) as an upper bound for the supervised method and comparing this to Eq. (19.5) from [6] we observe that all differences are either constant or logarithmic in ϵ and δ. □

3.5.3. THE LIMITS OF MANIFOLD REGULARIZATION

We now relate our result to the conjectures published in Shalev-Shwartz and Ben-David [16]: A SSL cannot learn faster by more than a constant (which may depend on the hypothesis class $\mathcal{F}$ and the loss ϕ) than the supervised learner. Theorem 1 from [7] showed that this conjecture is true up to a logarithmic factor, much like our result, for classes with finite VC-dimension, and SSL that do *not* make any distributional assumptions. Corollary 3 shows that this statement also holds in some scenarios for all SSL that fall in our proposed framework. This is somewhat surprising, as our result holds explicitly for SSLs that *do* make assumptions about the distribution: MR assumes the labeling function behaves smoothly w.r.t. the underlying manifold.

3.6. RADEMACHER COMPLEXITY OF MANIFOLD REGULARIZATION

In order to find out in which scenarios semi-supervised learning can help it is useful to also look at distribution *dependent* complexity measures. For this we derive computational feasible upper and lower bounds on the Rademacher complexity of MR. We first review the work of Sindhwani *et al.* [9]: they create a kernel such that the inner product in the corresponding kernel Hilbert space contains automatically the regularization term from MR. Having this kernel we can use standard upper and lower bounds of the Rademacher complexity for RKHS, as found for example in [17]. The analysis is thus similar to [4]. They consider a co-regularization setting. In particular [9, p1] show the following, here informally stated, theorem.

Theorem 20 ([9, Propositions 2.1, 2.2])**.** *Let H be a RKHS with inner product $\langle\cdot,\cdot\rangle_H$. As before let $U = \{x_1, ..., x_{n+m}\}$, $f, g \in H$ and $f_U = (f(x_1), ..., f(x_{n+m}))^t$. Furthermore let $\langle\cdot,\cdot\rangle_{\mathbb{R}^n}$ be any inner product in $\mathbb{R}^n$. Let $\tilde{H}$ be the same space of functions as H, but with a newly defined inner product by $\langle f, g\rangle_{\tilde{H}} = \langle f, g\rangle_H + \langle f_U, g_U\rangle_{\mathbb{R}^n}$. Then $\tilde{H}$ is a RKHS.*

Assume now that L is a positive definite n-dimensional matrix and we set the inner product $\langle f_U, g_U\rangle_{\mathbb{R}^n} = f_U^t L g_U$. By setting L as the Laplacian matrix (Section 3.4) we note

[2] Note that the original formulation is in terms of the fat-shattering dimension, but this is always bounded by the pseudo-dimension.

that the norm of $\tilde{H}$ automatically regularizes w.r.t. the data manifold given by $\{x_1, ..., x_{n+m}\}$. We furthermore know the exact form of the kernel of $\tilde{H}$.

Theorem 21 ([9, Proposition 2.2])**.** *Let $k(x, y)$ be the kernel of H, K be the gram matrix given by $K_{ij} = k(x_i, x_j)$ and $k_x = (k(x_1, x), ..., k(x_{n+m}, x))^t$. Finally let I be the $n + m$ dimensional identity matrix. The kernel of $\tilde{H}$ is then given by $\tilde{k}(x, y) = k(x, y) - k_x^t(I + LK)^{-1}Lk_y$.*

This interpretation of MR is useful to derive computationally feasible upper and lower bounds of the empirical Rademacher complexity, giving distribution *dependent* complexity bounds. With $\sigma = (\sigma_1, ..., \sigma_n)$ i.i.d Rademacher random variables (i.e. $P(\sigma_i = 1) = P(\sigma_i = -1) = \frac{1}{2}$.), recall that the empirical Rademacher complexity of the hypothesis class H and measured on the sample labeled input features $\{x_1, ..., x_n\}$ is defined as

$$\mathrm{Rad}_n(H) = \frac{1}{n}\mathbb{E}_\sigma \sup_{f \in H} \sum_{i=1}^{n} \sigma_i f(x_i).$$

Theorem 22 ([17, p. 333])**.** *Let H be a RKHS with kernel k and $H_r = \{f \in H \mid ||f||_H \leq r\}$. Given an n sample $\{x_1, ..., x_n\}$ we can bound the empirical Rademacher complexity of H_r by*

$$\frac{r}{n\sqrt{2}}\sqrt{\sum_{i=1}^{n} k(x_i, x_i)} \leq \mathrm{Rad}_n(H_r) \leq \frac{r}{n}\sqrt{\sum_{i=1}^{n} k(x_i, x_i)}. \tag{3.12}$$

The previous two theorems lead to upper bounds on the complexity of MR, in particular we can bound the maximal reduction over supervised learning.

Corollary 4. *Let H be a RKHS and for $f, g \in H$ define the inner product $\langle f, g\rangle_{\tilde{H}} = \langle f, g\rangle_H + f_U(\mu L)g_U^t$, where L is a positive definite matrix and $\mu \in \mathbb{R}$ is a regularization parameter. Let $\tilde{H}_r$ be defined as before, then*

$$\mathrm{Rad}_n(\tilde{H}_r) \leq \frac{r}{n}\sqrt{\sum_{i=1}^{n} k(x_i, x_i) - k_{x_i}^t(\frac{1}{\mu}I + LK)^{-1}Lk_{x_i}}. \tag{3.13}$$

Similarly we can obtain a lower bound in line with Inequality (3.12).

The corollary allows us to compute upper bounds of the Rademacher complexity for MR and shows in particular that the difference of the Rademacher complexity of the supervised and the semi-supervised method is given by the term $k_{x_i}^t(\frac{1}{\mu}I_{n+m} + LK)^{-1}Lk_{x_i}$. This can be used for example to compute generalization bounds [15, Chapter 3]. We can also use the kernel to compute local Rademacher complexities which may yield tighter generalization bounds [18]. Here we illustrate the use of our bounds for choosing the regularization parameter μ without the need for an additional labeled validation set.

3.7. EXPERIMENT: CONCENTRIC CIRCLES

We illustrate the use of Eq. (3.13) for model selection. In particular, it can be used to get an initial idea of how to choose the regularization parameter μ. The idea is to plot the Rademacher complexity versus the parameter μ as in Figure 3.1. We propose to use an

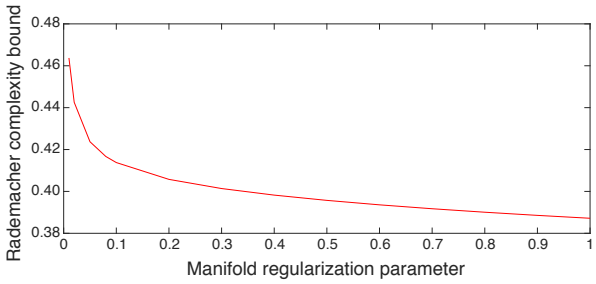


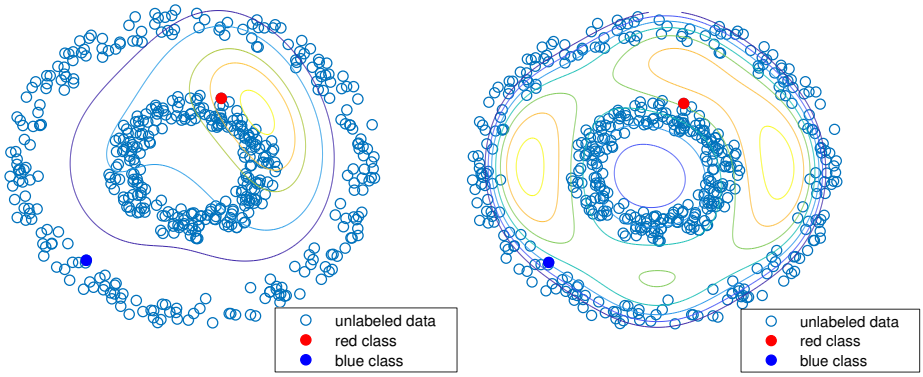
Figure 3.1: The behavior of the Rademacher complexity when using manifold regularization on circle dataset with different regularization values μ.

heuristic which is often used in clustering, the so called elbow criteria [19]. We essentially want to find a μ such that increasing the μ will not result in much reduction of the complexity anymore. We test this idea on a dataset which consists out of two concentric circles with 500 datapoints in $\mathbb{R}^2$, 250 per circle, see also Figure 3.2. We use a Gaussian base kernel with bandwidth set to 0.5. The MR matrix L is the Laplacian matrix, where weights are computed with a Gaussian kernel with bandwidth 0.2. Note that those parameters have to be carefully set in order to capture the structure of the dataset, but this is not the current concern: we assume we already found a reasonable choice for those parameters. We add a small L2-regularization that ensures that the radius r in Inequality (3.13) is finite. The precise value of r plays a secondary role as the behavior of the curve from Figure 3.1 remains the same.

Looking at Figure 3.1 we observe that for μ smaller than 0.1 the curve still drops steeply, while after 0.2 it starts to flatten out. We thus plot the resulting kernels for $\mu = 0.02$ and $\mu = 0.2$ in Figure 3.2. We plot the isolines of the kernel around the point of class one, the red dot in the figure. We indeed observe that for $\mu = 0.02$ we don't capture that much structure yet, while for $\mu = 0.2$ the two concentric circles are almost completely separated by the kernel. If this procedure indeed elevates to a practical method needs further empirical testing.

3.8. DISCUSSION AND CONCLUSION

This chapter analysed improvements in terms of sample or Rademacher complexity for a certain class of SSL. The performance of such methods depends both on how the approximation error of the class $\mathcal{F}$ compares to that of $\mathcal{F}_\tau^\psi$ and on the reduction of complexity by switching from the first to the latter. In our analysis we discussed the second part. The first part depends on a notion the literature often refers to as a *semi-supervised assumption*. This assumption basically states that we can learn with $\mathcal{F}_\tau^\psi$ as good as with $\mathcal{F}$. Regarding our example of the two concentric circles, this would mean that each circle actually corresponds to a class. Without prior knowledge, it is unclear whether one can test efficiently if the assumption is true or not. Or is it possible to treat just this as a model selection problem? The only two works we know that provide some analysis in this direction are [20], which discusses the sample consumption to test the so-called cluster assumption, and [21], which



(a) $\mu = 0.02$ (b) $\mu = 0.2$

Figure 3.2: The resulting kernel when we use manifold regularization with parameter μ set to 0.02 and 0.2.

analyzes the overhead of cross-validating the hyper-parameter coming form their proposed semi-supervised approach.

As some of our settings need restrictions, it is natural to ask whether we can extend the results. First, Lemma 1 restricts us to convex optimization problems. If that assumption would be unnecessary, one may get interesting extensions. Neural networks, for example, are typically not convex in their function space and we cannot guarantee the fast learning rate from Theorem 19. But maybe there are semi-supervised methods that turn this space convex, and thus could achieve fast rates. In Theorem 19 we have to restrict the loss to be the square loss, and [6, Example 21.16] shows that for the absolute loss one cannot achieve such a result. But whether it is possible for the hinge loss, which is a typical choice in classification, is unknown to us. Corollary 3 considers regression and one can wonder if similar results hold for classification, e.g. when we use the hinge loss. We speculate that this is indeed true, as at least the related classification tasks, that use the $0-1$ loss, cannot achieve a rate faster than $\frac{1}{\epsilon}$ [16, Theorem 6.8].

Finally, we sketch a scenario in which sample complexity improvements of MR can be at most a constant over their supervised counterparts, ignoring logarithmic factors. This may sound like a negative result, as we saw in the previous chapter that other methods, that seem to have similar assumptions, can achieve learning rates that are exponential in the number of labeled samples. But constant improvement can still have significant effects, if this constant can be arbitrarily large. For that consider again the example of the two concentric circles. If we set the regularization parameter μ high enough, the only possible classification functions will be the one that classifies each circle uniformly to one class, while the pseudo-dimension of the supervised model can be arbitrarily high, and thus also the constant in Corollary 3. In conclusion, one should realize the significant influence constant factors in finite sample settings can have.

REFERENCES

[1] M. Belkin, P. Niyogi, and V. Sindhwani, *Manifold regularization: A geometric framework for learning from labeled and unlabeled examples,* JMLR **7**, 2399 (2006).

[2] P. Niyogi, *Manifold regularization and semi-supervised learning: Some theoretical analyses,* JMLR **14**, 1229 (2013).

[3] Y. Grandvalet and Y. Bengio, *Semi-supervised learning by entropy minimization,* in *NeuRIPS* (Vancouver, British Columbia, Canada, 2004) pp. 529–536.

[4] V. Sindhwani and D. S. Rosenberg, *An rkhs for multi-view learning and manifold co-regularization,* in *ICML* (Helsinki, Finland, 2008) pp. 976–983.

[5] M.-F. Balcan and A. Blum, *A discriminative model for semi-supervised learning.* Journal of the ACM **57**, 19:1 (2010).

[6] M. Anthony and P. L. Bartlett, *Neural Network Learning: Theoretical Foundations*, 1st ed. (Cambridge University Press, New York, NY, USA, 2009).

[7] M. Darnstädt, H. U. Simon, and B. Szörényi, *Unlabeled data does provably help.* in *STACS*, Vol. 20 (Kiel, Germany, 2013) pp. 185–196.

[8] S. Ben-David, T. Lu, and D. Pál, *Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning,* in *Proceedings of the The 21st Annual Conference on Learning Theory* (Helsinki, Finland, 2008).

[9] V. Sindhwani, P. Niyogi, and M. Belkin, *Beyond the point cloud: From transductive to semi-supervised learning,* in *ICML* (Bonn, Germany, 2005) pp. 824–831.

[10] A. Globerson, R. Livni, and S. Shalev-Shwartz, *Effective semisupervised learning on manifolds.* in *COLT* (Amsterdam, The Netherlands, 2017) pp. 978–1003.

[11] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning* (The MIT Press, Cambridge, MA, USA, 2006).

[12] M. Belkin and P. Niyogi, *Towards a theoretical foundation for laplacian-based manifold methods,* Journal of Computer and System Sciences **74**, 1289 (2008).

[13] M. Kloft, U. Brefeld, P. Laskov, K.-R. Müller, A. Zien, and S. Sonnenburg, *Efficient and accurate lp-norm multiple kernel learning,* in *NeuRIPS* (Vancouver, British Columbia, Canada, 2009) pp. 997–1005.

[14] V. N. Vapnik, *Statistical Learning Theory* (Wiley-Interscience, 1998).

[15] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning* (The MIT Press, Cambridge, MA, USA, 2012).

[16] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, New York, NY, USA, 2014).

[17] S. Boucheron, O. Bousquet, and G. Lugosi, *Theory of classification: A survey of some recent advances,* ESAIM: Probability and Statistics **9**, 323 (2005).

[18] P. L. Bartlett, O. Bousquet, and S. Mendelson, *Local rademacher complexities,* The Annals of Statistics **33**, 1497 (2005).

[19] P. Bholowalia and A. Kumar, *Article: Ebk-means: A clustering technique based on elbow method and k-means in wsn,* International Journal of Computer Applications **105**, 17 (2014).

[20] M. Balcan, E. Blais, A. Blum, and L. Yang, *Active property testing,* in *53rd Annual IEEE Symposium on Foundations of Computer Science* (New Brunswick, NJ, USA, 2012) pp. 21–30.

[21] M. Azizyan, A. Singh, and L. A. Wasserman, *Density-sensitive semisupervised inference,* Computing Research Repository **abs/1204.1685** (2012).

4

A Soft-Labeled Self-Training Approach

In this chapter we propose a self-training method that uses the notion of soft-labels, which can be thought of as a class probability estimate. We show that a self-training approach with soft-labeling is preferable in many cases in terms of expected loss (risk) minimization. The main idea is to use the soft-labeling to minimize the risk on labeled and unlabeled data together, in which the hard-labeled self-training is an extreme case. This method is related to the well-known expectation-maximization method and can be seen as an extension to discriminative models.

Parts of this chapter have been published in the proceedings of the 23rd International Conference on Pattern Recognition [1].

4.1. INTRODUCTION

The challenge of semi-supervised learning (SSL) is to handle situations where obtaining labeled samples is time-consuming or expensive, but unlabeled data is easy to get. Typical examples would be document classification [2], image classification [3] and gene function prediction [4]. A simple approach to SSL is the so-called self-training or self-learning. In this setting one first trains a classifier with the available labeled data, and then labels the unlabeled data using the classifier. The classifier is then retrained with the whole data and this process can be iterated. Triguero *et al.* [5] conduct a survey on self-training where they compare different methods on different classifiers. One of the better known methods, called co-training [6], splits the feature space in two different subspaces and tries to train two distinct classifiers that label new data for each other. This new labeled data is then used to retrain each classifier. Other methods retrain only with the unlabeled data of its own most confident predictions [7].

The aim of this chapter is to show that a soft-labeled self-training approach can improve the overall risk of a classifier compared to its hard-labeled counterpart in most settings. In Section 4.2 we are going to provide some elementary definitions that we use throughout the chapter. Section 4.3.1 will address the problem that most loss based classifiers do not automatically give a posterior class probability given the observation, and we propose soft labels (posterior probabilities) for loss based classifiers. It should be kept in mind that the aim of this chapter is not to define the best soft-labeling. We rather define a reasonable soft-labeling to show that in most cases it is preferred to take a soft-labeling over a hard-labeling for self-training in terms of risk minimization. A different comparison between soft and hard-labeling for the case of the least squares classifier is also done in [8]. The derivation of the soft-labeled method is done by including the unlabeled data together with variables for their soft-labels in the objective function, and then minimizing the objective function in terms of the linear model *and* the soft-labels. An explanation of the hard-labeling method will be found in Section 4.3.2, where we draw a comparison to our own work. One of the results of [8] is that the hard-labeled variant is more prone to get stuck in local optima. The rest of Section 4.3 will focus on the expectation minimization framework that we use and how this translates for the nearest mean classifier (NMC) and the least squares classifier. One can compare the idea to the expectation maximization (EM) algorithm [9] which tries to maximize the likelihood on the complete data, a concept that is closely related to self-training. The similarities and differences between EM and the proposed method will be shown in Section 4.3.2. Another similarity can be drawn to Contrastive Pessimistic Likelihood Estimation [10], where a worst case labeling of unlabeled data is considered to improve in a semi-supervised manner the likelihood in LDA. Our proposed soft labels follow a similar worst case consideration. However the very strong result of [10] gives a guaranteed likelihood improvement, our concept of risk minimization holds only in expectation. As we will show in Section 4.3.3 our solution can be understood as the minimizer of the expected loss over all possible labelings of the unlabeled data, with a prior over the possible labelings found by our supervised classifier. In Section 4.4 we describe the experiments done on artificial and real world data. Section 4.5 will present the results of the experiments. Testing our method on the NMC we find that in most cases the accuracy deteriorates even though we achieve the goal of a better risk minimization. We create and display an artificial example with similar behavior in order to try to understand this

phenomenon. The least squares classifier benefits from the soft-labeling in all measured categories on all real world datasets. In Section 4.6 we will give the conclusion and discuss possible future work.

4.2. PRELIMINARIES

Let $f : \mathcal{X} \to \mathbb{R}$ be a classifier with an input space $\mathcal{X}$ and and output space of numerical values $\mathbb{R}$. In the following we are going to consider classifiers f that are based on optimizing a loss function $L : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ where $\mathcal{Y}$ is the set of possible labels. We will furthermore only consider binary classification problems, i.e. $\mathcal{Y} = \{0, 1\}$. In particular: Given a sample set $(X, Y) := \{(x_i, y_i)\}_{i=1}^N$ we obtain the classifier of choice by the minimization:

$$f^* = \arg\min_f \sum_{i=1}^N L(f(x_i), y_i). \tag{4.1}$$

4.3. THE EXPECTATION MINIMIZATION FRAMEWORK

In this section we provide the framework of our method. Given a supervised trained classifier we are going to derive a conservative posterior probability for unseen data, which will be related to the loss function. Then we use these probabilities to define the objective function which we use to find the semi-supervised solution. After that we give an alternative view of the objective function. We then introduce a more flexible approach that will allow us to smoothly vary the soft-labels to hard-labels. We conclude the section by making the framework explicit for the least squares classifier and the NMC.

4.3.1. THE CHOICE OF PROBABILITY

In order to define the expectation minimization framework we first need to define a soft-labeling, or, in other terms, a probability distribution over the possible labels given an observation. What we want in detail is the following: Let $x \in \mathcal{X}$ be a feature vector and f a fixed classifier. We then want a $p \in [0, 1]$ that serves as an estimate of $P(Y = 1 \mid X = x)$. In some cases the classifier itself will give us a reasonable choice for p, for example in the case of logistic regression or naive Bayes, but for a general loss function this is not the case and thus p has to be found in a different way.

The proposed choice of p is motivated by the idea to minimize the maximum possible loss we can incur when using the loss function L. That means our p for each $x \in \mathcal{X}$ is found by the following min-max equation:

$$p^* = \arg\min_{p \in [0,1]} \max\{pL(f(x), 1), (1 - p)L(f(x), 0)\} \tag{4.2}$$

One can think about this as a game where we will suffer a loss for x, depending on what the true label is. We do not know which label it will be, but we are allowed to weigh the loss. And we do the weighing in such a way that we reduce the maximum loss. Since $L(f(x), 1)$ and $L(f(x), 0)$ are constant for each x we find that the solution of the equation is given when p equalizes both terms, i.e.

$$pL(f(x), 1) = (1 - p)L(f(x), 0) \tag{4.3}$$

which is solved by

$$p = \frac{L(f(x), 0)}{L(f(x), 0) + L(f(x), 1)}. \tag{4.4}$$

4.3.2. The Semi-Supervised Solution & Related Work

In the following we are going to present the main idea in how to derive the semi-supervised solution. Assume that we have additionally to the labeled data $(X, Y) := \{(x_i, y_i)\}_{i=1}^{N}$ also a set of M unlabeled data points U. In this semi-supervised setting we are trying, similarly to the expectation maximization algorithm in the likelihood setting, to update our classifier by minimizing the expected risk on the labeled and unlabeled data together. The idea is to train a classifier f_{sup} with the labeled data first and use this to find the probability distributions over the labels for each unlabeled data point as defined in the previous subsection. Note that for every labeled sample $x_i \in X$ we set $p(Y = y_i \mid X = x_i) = 1$ since we actually made the observation. With this we can define a *risk* for a classifier f using these probabilities:

$$R(V, f) := \mathbb{E}\left[\sum_{v \in V} L(f(v), k)\right] = \sum_{v \in V} p(0|v) L(f(v), 0) + p(1|v) L(f(v), 1) \tag{4.5}$$

where $V = X \cup U$. Our semi-supervised solution f_{semi} is now simply found by minimizing this risk:

$$f_{semi} = \arg\min_{f} R(V, f) \tag{4.6}$$

The new solution can than be used to get better estimates of the posterior probabilities, and the procedure can be iterated. We want to remark that the same objective function is found in [8]. The difference there is that the posterior probabilities (there named responsibilities), are part of the minimization task, and thus can be seen as an *optimistic* label estimate. For this specific formulation, however, the optimistic approach degrades to hard-label self-learning. In contrast to that Loog [10] introduces a pessimistic approach, which essentially minimizes the objective for the worst case posterior label distribution $p(Y \mid X)$ and can give strong improvement guarantees. This was done, however, with a generative model, and Krijthe and Loog [11] actually show that for a large class of discriminative models such a pessimistic approach is impossible. While [8] and [10] can be seen as an optimistic and a pessimistic approach respectively, this work can be seen as an in-between approach as it deals with an average case with respect to posterior estimates.

At this point the similarity to EM also becomes clear. While the EM makes use of a likelihood function to maximize the expected log-likelihood we make use of a loss function to minimize the expected loss. The biggest difference is that the posterior probabilities are given by the probabilistic model in the case of EM, while we have to create them in a heuristic manner.

4.3.3. An Alternative View

In this subsection we give an alternative description of the risk in Equation (4.5) to get an intuition in what this risk is minimizing. For this we define a probability distribution over all possible labelings $\Theta = \{\theta : V \to \mathcal{Y}\}$. For $\theta \in \Theta$ we set $p(\theta) = \frac{1}{2^{N+M-1}(N+M)} \sum_{v \in V} p(\theta(v) \mid v)$.

This is indeed a probability distribution:

$$\begin{aligned}\sum_{\theta\in\Theta} p(\theta) &= \frac{1}{2^{N+M-1}(N+M)} \sum_{k\in\mathcal{Y}} \sum_{v\in V} 2^{N+M-1} \cdot p(k \mid v) \\ &= \frac{1}{2^{N+M-1}(N+M)} \sum_{v\in V} 2^{N+M-1} \cdot 1 = 1\end{aligned}$$

The first equality holds since a specific label of a single observation appears exactly 2^{N+M-1} times in the sum over all possible labelings, the cardinality of all possible labelings of all other observations. The second equation holds since each $p(\cdot \mid v)$ is a probability distribution itself. We also set for a particular labeling $\theta \in \Theta$ the loss of a classifier f to be

$$L(f,\theta) = \sum_{v\in V} L(f(v),\theta(v)). \tag{4.7}$$

This allows us to formulate the risk (4.5) up to a constant as an expectation over all possible labelings:

$$\begin{aligned}E_\Theta\left[L(f,\theta)\right] &= \sum_{\theta\in\Theta} p(\theta) L(f,\theta) \\ &= \frac{1}{2^{N+M-1}(N+M)} \sum_{v\in V} \sum_{k\in K} 2^{N+M-1} p(k|v) \cdot L(f(v),k) \\ &= \frac{1}{(N+M)} \sum_{v\in V} \mathbb{E}\left[L(f(v),k)\right]\end{aligned}$$

So up to a constant this is equivalent to (4.5) and thus gives the same solution by minimizing. That means that our solution is derived by minimizing the expected loss over all possible labelings, with a probability distribution derived from our initial classifier.

4.3.4. A MORE FLEXIBLE APPROACH

Adding a parameter α to the proposed soft-label as follows gives us a more flexible approach and lets us smoothly move between soft-labeling and hard-labeling. We achieve this by modifying the min-max expression as follows:

$$\arg\min_{p\in[0,1]} \max\{pL(f(x),1)^\alpha, (1-p)L(f(x),0)^\alpha\} \tag{4.8}$$

The solution of this is for the same reasoning as in Section 4.3.1 given by

$$p = \frac{L(f(x),0)^\alpha}{L(f(x),0)^\alpha + L(f(x),1)^\alpha}. \tag{4.9}$$

Choosing for example α big enough corresponds to the decision that a hard-labeled self-training approach is the best. This might be the case when classes are properly separated as we will see in the experiments. The effects of this parameter will be shown in a controlled setting and then tested on real world data.

The proposed method can be used for every classifier which is based on minimizing a loss function. To keep things simple we chose to test the proposed method on the nearest mean (NMC) and the least squares classifier. We will describe in the following how equation Equation (4.5) translates in both of these cases.

4.3.5. Least Squares Classification

The least squares classifier (see for instance [11, section 3.4.3]) tries to optimize the least square criterion $L(f(v), k) = ||f(v) - k||^2$ for linear classifier f. Setting $\mathcal{Y} = \{-1, 1\}$ expression (4.5) becomes

$$\sum_{v \in D} p(k|v)||f(v) - 1||^2 + (1 - p(v, k))||f(v) + 1||^2. \tag{4.10}$$

Setting $\pi = (p(1 \mid v))_{v \in V}$ and $\pi^- = (p(-1 \mid v))_{v \in V}$ as the vectors of probabilities we get similarly to the supervised case the following closed form solution.

$$f_{semi} = (D^T D)^{-1}(D^T \pi - D^T \pi^-). \tag{4.11}$$

4

4.3.6. Nearest Mean Classification

Choosing the loss as $L(f(v), k) = ||v - m_k^f||^2$ for $k \in K$ and minimizing this for the model (m_1^f, m_{-1}^f) will give us the nearest mean classifier, i.e. the vector (m_1^f, m_{-1}^f) will correspond to the two class means. Assigning a new unseen data point to the class of its minimum loss is in this case equivalent to assigning it to the class with the nearest mean. Thus this loss defines the nearest mean classifier [12]. Using expression (4.5) for the semi-supervised case, the solution becomes a weighted mean:

$$m_k = \frac{\sum_{v \in D} p(k|v)\, v}{\sum_{v \in D} p(k|v)} \tag{4.12}$$

4.4. Experiments

This section is devoted to test the proposed method on the nearest mean and the least squares classifier. First, we examine the behavior in a controlled environment and then on 11 real world datasets for the nearest mean classifier and on 8 real world datasets for the least squares classifier. The datasets were taken from UCI Machine Learning Repository [13], all having 2 classes. Specifications can be found in Table 4.1. We did not perform the least square classification on the full 11 datasets since in 3 cases the structure and dimension of the data led to unstable behavior of the matrix $(D^T D)^{-1}$. To keep things simple we furthermore used only one iteration of our algorithm.

4.4.1. Controlled Setting

The first experiments were done on two normally distributed classes of dimension two with same covariance (given by the identity matrix), different means, and with equal class priors. The dataset Gauss 1 has class means $(0, 0)$ and $(0, 1)$, Gauss 2 has class means $(0, 0)$ and $(0, 2)$ and Gauss 5 has class means $(0, 0)$ and $(0, 5)$. We initially created 100,000 points per class on which we then did 1000 test runs. In each run we randomly chose in total 4 labeled samples, where we made sure that at least 1 point per class is included, and 100 unlabeled points to train the classifier, and used the rest to test the classifier. All experiments were done with the parameter α being 0.1 and 1 and we compared this to a hard-labeling, i.e. $\alpha = \infty$, as well as to the supervised trained classifier. The measurements based on which

Table 4.1: Specifications of the datasets

Data	Dimension	Objects	Lowest class prior
ad	1558	2359	0.1615
Haberman	3	306	0.2647
ionosphere	33	351	0.3590
Parkinson	22	195	0.2462
Pima	8	768	0.3490
sonar	60	208	0.4663
spambase	57	4601	0.3940
spect	22	267	0.2060
spectf	44	349	0.2722
transfusion	3	748	0.2380
wdbc	30	569	0.3726

we evaluate the methods are the mean accuracy improvement compared to the supervised solution (MAI), the percentage of in how many initiations the semi-supervised classifier had a worse accuracy than the supervised classifier (%neg) and the risk (the mean loss with respect to the loss we used) on the test set (Risk). We also show the mean accuracy from the supervised solution (acc). Note that in the evaluation one should in particular pay attention to Risk, since this is the value that the proposed method tries to optimize (cf. [14, 15]). This value shows if the soft-labeled approach minimizes expression (4.5) in our experiments better than a hard-labeled approach and whether it will improve it at all compared to the supervised solution.

4.4.2. REAL WORLD DATA

In the case of the nearest mean classifier we took 4 labeled points and only 50 unlabeled points to make the test set as big as possible. To make sure that the matrix $D^T D$ from the closed form solution of the least square classifier is invertible, we chose the number of labeled samples N (also indicated in the tables) for each dataset individually, depending on the dimension and the structure of the data. To have still enough points to test on, we took $2N$ unlabeled points to train the semi-supervised classifier. The evaluation is the same as in the controlled setting.

4.5. RESULTS

The results are presented in Tables 4.3, 4.4, 4.5 and 4.6. Table 4.3 and 4.4 show the results from the controlled settings while Tables 4.5 and 4.6 present the results from the real world data. Each table contains the results for $\alpha = 0.1$, $\alpha = 1$ and the hard-labeled approach. The best performing method is highlighted in bold for each criterion on each dataset.

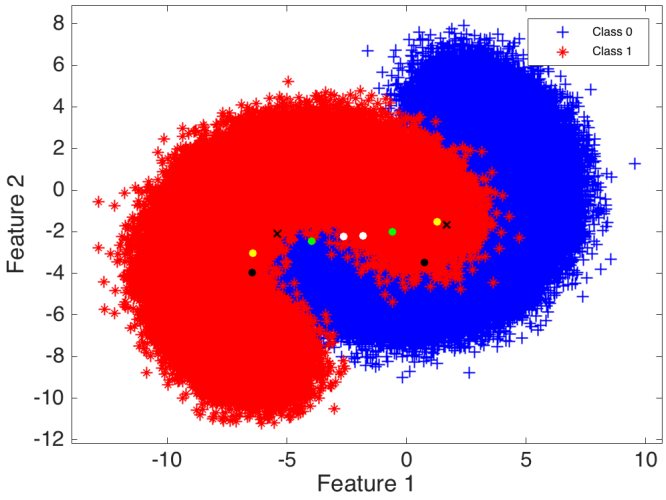


Figure 4.1: An artificial example for the failure of NMC. The black crosses show the class means. The black, yellow, green and white dots show respectively the estimated means from the supervised solution, a hard labeled self-training and a self-training with $\alpha = 1$ and $\alpha = 0.1$. Although the hard labeled self-training (yellow dots) gives a better estimate for the mean, it deteriorates in accuracy

4.5.1. CONTROLLED SETTING

In the controlled setting both classifier show the expected behavior: the harder the problem is (meaning the bigger the Bayes error), the better the soft-labeled approach is. For Gauss 1 we get improvements in MAI and Risk for both choices of α in comparison to the hard-labeling. In the case of Gauss 5 the hard labeling gives the best results. This was to be expected since the classification problem is in this case rather easy and one can assume that most of the predicted labels will be correct. This is given in this case, since the NMC converges in this setting fast to the Bayes classifier. This is supported by the accuracy of the NMC in the supervised solution.

In case of the least squares we see strict improvements for both α on Gauss 1 and 2 compared to the hard-labeling. In Gauss 1 we manage to switch an average deterioration to an average improvement by choosing $\alpha = 0.1$. An interesting behavior can be found for Gauss 5. Although the setting for $\alpha = 0.1$ gives the best MAI we get a worse performance in terms of %neg and Risk. A similar behavior is also seen on three of the real world datasets (Haberman, spect, wdbc). Remarkable is that even on Gauss 5 our method for $\alpha = 1$ gives similar (MAI, Risk) or better results (%neg) compared to hard-labeling.

4.5.2. REAL WORLD DATA

For the NMC the results are mixed in terms of %neg and MAI, but only on one dataset (wdbc) the hard-labeling outperforms our method in terms of the risk. Interestingly this is the dataset where we find the biggest improvement in terms of %neg and MAI of our method compared to the hard-labeling. That suggests that the actual loss we are minimizing might

Table 4.2: NMC on banana-shaped data

Method	%neg	MAI	Risk	Risk SV	Acc
$\alpha = 0.1$	0.599	**0.004**	5.295	5.207	0.762
$\alpha = 1$	0.62	0	4.87	5.207	0.762
hard label	**0.594**	-0.007	**4.695**	5.207	0.761

not be the best choice in this case. The NMC for these datasets seem in general not to be a good choice together with the self-training approach. The hard-labeling manages to give only in two cases a positive MAI, and similar results hold for the soft-labeling. In terms of Risk we find on the other hand that we get good improvements over the supervised solution and the hard-labeled approach. In Figure 4.1 we provide an artificial example that illustrates this behavior, that despite the improved risk minimization we deteriorate in accuracy. We used a two-dimensional banana-shaped dataset where the nearest mean classifier is a clear model misspecification. We trained on eight labeled and 100 unlabeled points and did 1000 test runs to evaluate. The results are noted in Table 4.2. In this setting only the hard labeling gave improvements on Risk, but had the biggest deterioration in terms of accuracy. This is due to the misspecification and the fact that a better class mean estimate does not give a higher accuracy on this dataset. We expect that similar misspecification happens on the real world data, where the dataset wdbc shows the most similar behavior.

The results for the least squared classifier are clearer. The hard-labeled approach is outperformed on *every* dataset in *every* criterion by one of the soft-labeled counterparts. We find in this setting a more direct influence from the risk to the MAI. This can be explained by the fact that the loss of least squares classifier models the actual 0-1 loss. The loss used for the NMC classifier merely measures how good we are estimating the actual class mean, and thus can suffer heavily from misspecification. Note that on some datasets (pima, sonar, spectf, wdbc) the improvement in risk of the semi-supervised methods are fairly big. This can be explained by the fact that the least square loss is strongly affected by outliers. Adding unlabeled data to the training gives a higher chance to catch those outliers, and minimize the expected loss on them.

4.6. CONCLUSION

Our aim was to show that a soft-labeled self-training is in many cases to be be preferred over a hard-labeled self-training approach, at least in terms of risk minimization. In the controlled setting we could show that for Gaussian data, a soft-labeled approach is to be

Table 4.3: NMC on artificial data

	$\alpha = 0.1$			$\alpha = 1$			hard label			Supervised	
Data	%neg	MAI	Risk	%neg	MAI	Risk	%neg	MAI	Risk	Risk	Acc
Gauss 1	0.225	**0.018**	1.3254	**0.223**	0.015	**1.310**	0.272	0.009	1.449	1.566	0.593
Gauss 2	0.089	0.041	1.515	**0.052**	**0.044**	1.392	0.067	0.036	**1.369**	1.574	0.703
Gauss 5	0.192	0.009	2.48	0.116	0.01	1.586	**0.049**	**0.0105**	**1.269**	1.586	0.983

Table 4.4: Least Squares classifier on artificial data

	$\alpha = 0.1$			$\alpha = 1$			hard label			Supervised	
Data	%neg	MAI	Risk	%neg	MAI	Risk	%neg	MAI	Risk	Risk	Acc
Gauss 1	**0.397**	**0.006**	**0.241**	0.526	-0.002	0.295	0.544	-0.002	0.351	2.66	0.573
Gauss 2	**0.316**	**0.018**	0.218	0.388	0.003	**0.205**	0.427	0.000	0.234	2.178	0.703
Gauss 5	0.186	**0.027**	0.173	**0.0460**	0.024	**0.067**	0.061	0.024	0.068	0.295	0.925

Table 4.5: NMC on real world data

	$\alpha = 0.1$			$\alpha = 1$			hard label			Supervised	
Data	%neg	MAI	Risk	%neg	MAI	Risk	%neg	MAI	Risk	Risk	Acc
ad	0.796	-0.071	109.4	0.787	-0.044	**101.6**	**0.410**	0.009	105.4	111.9	0.772
Haberman	0.639	-0.032	**11.7**	0.662	-0.035	11.8	**0.609**	**-0.015**	13.8	14.2	0.563
ionosphere	0.476	-0.026	2.796	**0.410**	-0.027	**2.775**	0.485	**-0.017**	3.064	3.433	0.637
Parkinson	0.718	-0.050	87.6	0.682	-0.04	**87.4**	**0.468**	**-0.012**	102.0	100.9	0.631
pima	0.676	-0.029	**97.4**	0.689	-0.033	99.2	**0.608**	**-0.0289**	123.1	119.3	0.570
sonar	**0.423**	**0.006**	**1.3**	0.558	-0.01	1.30	0.645	-0.02	1.41	1.58	0.542
spambase	**0.478**	**0.009**	**319.5**	0.584	0.000	349.7	0.677	-0.006	464.3	362.1	0.585
spect	**0.673**	-0.053	2.096	0.685	-0.054	**2.092**	0.688	**-0.022**	2.151	2.419	0.631
spectf	**0.536**	**-0.006**	55.96	0.539	-0.006	**55.9**	0.786	-0.037	63.257	66.9	0.583
transfusion	**0.611**	**-0.024**	**22.6**	0.615	-0.027	23.3	0.668	-0.026	28.1	27.1	0.536
wdbc	**0.14**	**0.043**	472.0	0.226	0.030	367.3	0.566	0.003	**363.1**	396.8	0.86

Table 4.6: Least Squares on real world data

	$\alpha = 0.1$			$\alpha = 1$			hard label			Supervised		N
Data	%neg	MAI	Risk	%neg	MAI	Risk	%neg	MAI	Risk	Risk	Acc	
ad	-	-	-	-	-	-	-	-	-	-	-	-
Haberman	0.369	**0.015**	**0.233**	**0.314**	0.01	0.307	0.372	0.009	0.349	0.758	0.637	8
ionosphere	**0.141**	**0.042**	**0.189**	0.18	0.035	0.243	0.176	0.031	0.305	1.70	0.771	51
Parkinson	-	-		-	-		**-**	**-**	-	-	-	-
pima	**0.262**	**0.043**	**0.236**	0.314	0.031	0.314	0.269	0.022	0.388	2.791	0.608	11
sonar	**0.147**	**0.112**	**0.226**	0.189	0.088	0.271	0.190	0.056	0.413	28.2	0.55	62
spambase	**-**	-	-	-	-	**-**	-	**-**	-	-	-	-
spect	0.381	**0.005**	0.177	**0.337**	0.003	**0.178**	0.413	0.001	0.198	0.208	0.748	60
spectf	**0.188**	**0.047**	**0.232**	0.33	0.017	0.328	0.367	0.01	0.443	3.665	0.588	52
transfusion	**0.361**	**0.012**	**0.213**	0.366	0.007	0.263	0.425	0.004	0.297	0.447	0.677	10
wdbc	0.160	**0.062**	**0.173**	**0.075**	0.052	0.227	0.115	0.041	0.295	1.344	0.759	36

preferred if the class overlap gets bigger. The real world data shows that in terms of risk minimization our method is clearly to be preferred over a hard-labeling. In all datasets we report only one where the hard-labeling actually outperforms the soft-labeling in terms of the risk. In the case of the least squares classification we find that our method outperforms the hard-labeling in all presented criteria, meaning that it is also preferred in terms of 0-1 loss in this case.

In Chapter 7 we discuss possible extensions of this work, in particular in view of the other chapters of this thesis.

REFERENCES

[1] A. Mey and M. Loog, *A soft-labeled self-training approach,* in *23rd International Conference on Pattern Recognition* (Cancun, Mexico, 2016) pp. 2604–2609.

[2] J. Su, J. S. Shirab, and S. Matwin, *Large scale text classification using semi-supervised multinomial naive bayes,* in *Proceedings of the 28th International Conference on Machine Learning* (Bellevue, Washington, USA, 2011) pp. 97–104.

[3] D. Dai and L. V. Gool, *Ensemble projection for semi-supervised image classification,* in *IEEE International Conference on Computer Vision* (Sydney, Australia, 2013) pp. 2072–2079.

[4] Z.-H. You, Z. Yin, K. Han, D.-S. Huang, and X. Zhou, *A semi-supervised learning approach to predict synthetic genetic interactions by combining functional and topological properties of functional gene network,* BMC Bioinformatics **11**, 1 (2010).

[5] I. Triguero, S. García, and F. Herrera, *Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study,* Knowledge and Information Systems , 1 (2014).

[6] A. Blum and T. Mitchell, *Combining labeled and unlabeled data with co-training,* in *Proceedings of the 11th Annual Conference on Computational Learning Theory* (Madison, Wisconsin, USA, 1998) pp. 92–100.

[7] M. Li and Z.-H. Zhou, *Setred: Self-training with editing.* in *Advances in Knowledge Discovery and Data Mining* (Hanoi, Vietnam, 2005) pp. 611–621.

[8] J. Krijthe and M. Loog, *Optimistic semi-supervised least squares classification,* in *23rd International Conference on Pattern Recognition* (Cancun, Mexico, 2016) pp. 1677–1682.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood from incomplete data via the em algorithm,* Journal of the Royal Statistical Society, Series B **39**, 1 (1977).

[10] M. Loog, *Contrastive pessimistic likelihood estimation for semi-supervised classification,* IEEE Transactions on Pattern Analysis and Machine Intelligence **38**, 462 (2016).

[11] J. Krijthe and M. Loog, *The pessimistic limits of margin-based losses in semi-supervised learning,* in *Advances in Neural Information Processing Systems 31* (Montreal, Canada, 2018) pp. 1795–1804.

[12] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis* (John Willey & Sons, New York, NY, USA, 1973).

[13] D. N. A. Asuncion, *UCI machine learning repository,* (2007).

[14] M. Loog and A. C. Jensen, *Semi-supervised nearest mean classification through a constrained log-likelihood,* IEEE Transactions on Neural Networks and Learning Systems **26**, 995 (2014).

[15] M. Loog, J. H. Krijthe, and A. C. Jensen, *On measuring and quantifying performance: Error rates, surrogate loss, and an example in SSL,* in *Handbook of Pattern Recognition and Computer Vision* (World Scientific, Singapore, 2016) Chap. 1.3.

5

POSTERIOR ESTIMATION

In this work we investigate to which extent one can recover class probabilities within the empirical risk minimization (ERM) paradigm. The main aim of this chapter is to extend existing results and emphasize the tight relations between empirical risk minimization and class probability estimation. Based on existing literature on excess risk bounds and proper scoring rules, we derive a class probability estimator based on empirical risk minimization. We then derive fairly general conditions under which this estimator will converge, in the L_1-norm and in probability, to the true class probabilities. Our main contribution is to present a way to derive finite sample L_1-convergence rates of this estimator for different surrogate loss functions. We also study in detail which commonly used loss functions are suitable for this estimation problem and finally discuss the setting of model-misspecification.

5.1. INTRODUCTION

In binary classification problems we try to predict a label $y \in \{-1, 1\} = \mathcal{Y}$ based on an input feature vector $x \in \mathcal{X}$. Since optimizing for the classification accuracy is often computationally too complex, one typically measures performance through a surrogate loss function. Such methods are designed to achieve good classification performance, but often we are also interested in the classifier's confidence or a class probability estimate as such. We may, for instance, not only want to classify a tumor as benign or malignant, but also know an estimated probability that the predicted label is wrong. Also various methods in active or semi-supervised learning rely on such class probability estimates. In active learning they are, for instance, used in uncertainty based rules [1, 2] while in semi-supervised learning they can be used for performing entropy regularization [3].

In this chapter we derive necessary and sufficient conditions under which classifiers, obtained through the minimization of an empirical loss function, allow us to estimate the class probability in a consistent way. More precisely, we present a general way to derive finite sample bounds based on those conditions. While the use of class probability estimates, as argued before, finds a broad audience, the necessary tools to understand the behavior, especially the literature on proper scoring rules, is not that broadly known. So next to our contribution on finite sample behavior for class probability estimation we present a condensed introduction to this, in our opinion, under-appreciated field.

A proper scoring rule is essentially a loss function that can measure the class probability *point-wise*. We investigate in which circumstances those loss functions make use of this potential and lift this point-wise property to the complete space. Next to proper scoring rules we use *excess risk bounds* to come to our results. Excess risk bounds are essentially inequalities that quantify how much an empirical risk minimizer is off from the true risk.

Combining those two areas, our main contributions are the following. Based on the existing literature, we define in Section 5.4, Equation (5.8), a probability estimate $\hat{\eta}$ derived from an empirical risk minimizer. Based on this we analyze in Section 5.5 to which extent commonly used loss functions are suitable for the task of class probability estimation. Following this and the analysis thereafter, we argue in Section 5.6.5 that the squared loss is, in view of this chapter, not a particular good choice. In Section 5.6 we derive conditions that ensure that the estimator $\hat{\eta}$ converges in probability towards the true posterior. In the same section we present a general way to analyze the finite sample behavior of the convergence rate for different loss functions. The idea is to bound the L_1-distance between the estimated and the true class probability by the excess risk and then use bounds on the excess risk together with the properties of proper scoring rules to show convergence. In the same section we discuss the behavior of the estimator when it is misspecified. In this case one can in general not recover the true class probabilities, but instead find the best approximation with respect to a Bregman divergence. In Section 5.7 we conclude and discuss our analysis. In particular we discuss how one can extend this work to asymmetric loss functions and analyze their convergence behavior per class label. The following two sections start with related work and some preliminaries.

5.2. RELATED WORK

Many results on posterior estimation in the context of non-parametric regression can be found in [4]. The main differences from our results to those type of results is twofold. First, to obtain meaningful convergence rate guarantees, the results of [4] make usually assumptions on the distribution. We shift this burden from the distribution to the hypothesis set used. The difference is, that while we always have meaningful finite sample guarantees, our estimation procedure is not consistent in the case of model misspecification. The methods used by [4] are always consistent, but may have arbitrarily slow convergence on some distributions. Second, as we assume that the excess risk bounds we use are true with high probability over drawn samples, our convergence results hold also with high probability, while [4] makes those statements in expectation over the sampling process.

The starting point of our analysis follows closely the notation and concepts as described by Buja *et al.* [5] and Reid and Williamson [6, 7]. While Buja *et al.* [5] and Reid and Williamson [6] deal with the inherent structure of proper scoring rules, Reid and Williamson [7] make connections between the expected loss in prediction problems and divergence measures of two distributions. In contrast to that we investigate under which circumstances proper scoring rules can make use of their full potential in order to estimate class probabilities.

Telgarsky *et al.* [8] perform an analysis similar to ours as they also investigate convergence properties of a class probability estimator, their start and end point are very different though. While we start with theory from proper scoring rules, their paper directly starts with the class probability estimator as found in [9]. The problem is that the estimator in [9] only appears as a side remark, and it is unclear to which extent this is the best, only or even the correct choice. This chapter contributes to close this gap and answers those questions. They show that the estimator converges to a unique class probability model. In relation to this one can view this chapter as an investigation of this unique class probability model and we give necessary and sufficient conditions that lead to convergence to the true class probabilities. Note also that their paper uses convex methods, while our work in comparison draws from the theory of proper scoring rules.

Agarwal and Agarwal [10] look at the problem in a more general fashion. They connect different surrogate loss functions to certain statistics of the class probability distribution, e.g. the mean, while we focus on the estimation of the full class probability distribution. This allows us to come to more specific results, such as finite sample behavior.

The probability estimator we use also appears in [11] where it is used to derive excess risk bounds, referred to as surrogate risk bounds, for bipartite ranking. The methods used are very similar in the sense that these are also based on proper scoring rules. The difference is again the focus, and even more so the conditions used. They introduce the notion of strongly proper scoring rules which directly allows one to bound the L_2-norm, and thus the L_1-norm, of the estimator in terms of the excess risk. We show that convergence can be achieved already under milder conditions. We then use the concept of modulus of continuity, of which strongly proper scoring rules are a particular case, to analyze the rate of convergence.

5.3. PRELIMINARIES

We work in the classical statistical learning setup for binary classification. We assume that we observe a finite i.i.d. sample $(x_i, y_i)_{1\leq i\leq n}$ drawn from a distribution P on $\mathcal{X}\times\mathcal{Y}$. Here $\mathcal{X}$ denotes a feature space and $\mathcal{Y}=\{-1,1\}$ denotes a binary response variable. We then decide upon a hypothesis class $\mathcal{F}$ such that every $f\in\mathcal{F}$ is a map $f:\mathcal{X}\to\mathcal{V}$ for some space $\mathcal{V}$. Given the space $\mathcal{V}$ we call any function $l:\{-1,1\}\times\mathcal{V}\to[0,\infty)$ a *loss function*. The interpretation of the loss function is that we incur the penalty $l(y,v)$ when we predicted a value v while we actually observed the label y. Our goal is then to find a predictor $f_n\in\mathcal{F}$ based on the finite sample such that $\mathbb{E}[l(Y,f_n(X)]$ is small, where $X\times Y$ is a random variable distributed according to P. In other words, we want to find an estimator f_n that approximates the true risk minimizer f_0 well in terms of the expected loss, where

$$f_0 := \arg\min_{f\in\mathcal{F}} \mathbb{E}[l(Y,f(X))]. \tag{5.1}$$

The estimator f_n is often chosen to be the empirical risk minimizer

$$f_n = \arg\min_{f\in\mathcal{F}} \sum_{i=1}^{n} l(y_i, f(x_i)).$$

5

As we show in this chapter, finding such an f_n implicitly means to find a good estimate for $p(y\mid x) := P(Y=y\mid X=x)$ in many settings. Since we regularly deal with $p(y\mid x)$ and related quantities we introduce the following notation. To start with, we define $\eta(x) := P(Y=1\mid X=x)$. Depending on the context we drop the feature x and think of $\eta\in[0,1]$ as a scalar. Accepting the small risk of overloading the notation we sometimes also think of η as a Bernoulli distribution with outcomes in $\mathcal{Y}$ and parameter η, as in the following definition. We define the *point-wise conditional risk* as

$$L(\eta,v) := \mathbb{E}_{Y\sim\eta}[l(Y,v)] = \eta l(1,v) + (1-\eta)l(-1,v), \tag{5.2}$$

the *optimal point-wise conditional risk* as

$$L^*(\eta) := \min_{v\in\mathcal{V}} L(\eta,v), \tag{5.3}$$

and we denote by $v^*(\eta)$ the set of values that optimize the point-wise conditional risk

$$v^*(\eta) := \arg\min_{v\in\mathcal{V}} L(\eta,v). \tag{5.4}$$

Finally we define the *conditional excess risk* as

$$\Delta L(\eta,v) := L(\eta,v) - L^*(\eta). \tag{5.5}$$

5.3.1. PROPER SCORING RULES

If we chose $\mathcal{V}=[0,1]$, we say that $l:\{-1,1\}\times\mathcal{V}\to\mathbb{R}$ is a *CPE loss*, where CPE stands for class probability estimation. The name stems from the fact that if $\mathcal{V}=[0,1]$ it is already normalized to a value that can be interpreted as a probability. If l is a CPE loss we call it a *proper scoring rule* or *proper loss* if $\eta\in v^*(\eta)$ and we call it a *strictly proper scoring rule* or *strictly proper loss* if $v^*(\eta)=\{\eta\}$. In other words, l is a proper scoring rule if η is *a* minimizer of $L(\eta,\cdot)$ and this is strict if η is the only minimizer. In case l is strict we drop the set notation of v^*, so that $v^*(\eta)=\eta$.

5.3.2. LINK FUNCTIONS

As we will see later strictly proper CPE losses are well suited for class probability estimation. In general, however, we cannot expect that $\mathcal{V} = [0,1]$, but we may still want to use the corresponding loss function for class probability estimation. To do that we will use the concept of link functions [5, 6]. A *link function* is a map $\psi : [0,1] \to \mathcal{V}$, so a function that indeed links the values from $\mathcal{V}$ to something that can be interpreted as a probability. Combining such a link function with a loss $l : \{-1,1\} \times \mathcal{V} \to [0,\infty)$ one can define a CPE loss l_ψ as follows.

$$l_\psi : \{-1,1\} \times [0,1] \to [0,\infty)$$
$$l_\psi(y,q) := l(y,\psi(q))$$

We call the combination of a loss and a link function (l,ψ) a *(strictly) proper composite loss* if l_ψ is (strictly) proper as a CPE loss.

To distinguish between the losses l and l_ψ we subscript the quantities (5.2)-(5.5) with a ψ if we talk about l_ψ instead of l. For example we define $L_\psi(\eta,q) := L(\eta,\psi(q))$ for $q \in [0,1]$ and in the same way we define $v^*_\psi(\eta)$, $L^*_\psi(\eta)$ and $\Delta L_\psi(\eta,q)$. Note that if (l,ψ) is a strictly proper composite loss, we know that $v^*_\psi(\eta)$ are single element sets, but the same does not need to hold for $v^*(\eta)$.

5

5.3.3. DEGENERATE LINK FUNCTIONS

To ask a composite loss (l,ψ) to be proper is not a strong requirement, one can check that choosing ψ as constant function already fulfills this. This is because a composite loss (l,ψ) is proper, iff the true posterior η is a minimizer of the conditional risk $L_\psi(\eta,\cdot)$, i.e. $\eta \in v^*_\psi(\eta)$. If ψ is constant, then so is the conditional risk $L_\psi(\eta,\cdot)$ and then every value is a minimizer, so in particular η is a minimizer. We want to avoid this degenerate behavior for the task of probability estimation and will ask ψ to cover enough of $\mathcal{V}$ in the following sense. We call a composite loss (l,ψ) *non-degenerate* if for all $\eta \in [0,1]$ we have that $\operatorname{Im}\psi \cap v^*(\eta) \neq \emptyset$, where $\operatorname{Im}\psi \subset \mathcal{V}$ is the image of ψ on $[0,1]$. This does not directly exclude constant link functions for example, but consider the following. If ψ is constant and non-degenerate, then there is a single $v = \operatorname{Im}\psi$ such that $v \in v^*(\eta)$ for all η. Thus v would always minimize the loss, and we would, irrespectively of the input, always predict v. This is of course a property that no reasonable loss function should carry.

5.4. BEHAVIOR OF PROPER COMPOSITE LOSSES

For our convergence results we will need a loss function to be a strictly proper CPE loss. In this section we investigate how to characterize those loss functions.

We start by investigating proper CPE loss functions. Our first lemma states that the link functions that turns the loss l into a proper composite loss is already defined by the behavior of v^*.

Lemma 2. *Let $l : \{-1,1\} \times \mathcal{V} \to [0,\infty)$ be a loss function and ψ be a link function. The composite loss function (l,ψ) is then proper and non-degenerate if and only if $\psi \in v^*$, meaning that $\psi(\eta) \in v^*(\eta)$ for all $\eta \in [0,1]$.*

Proof. First we show that if (l,ψ) is proper and non-degenerate, then $\psi \in v^*$. Let (l,ψ) be a proper composite loss, so $\eta \in v^*_\psi(\eta)$, i.e. η minimizes $L(\eta,\psi(\cdot))$. As (l,ψ) is non-degenerate

there exists at least one η_1 such that $\psi(\eta_1) \in v^*(\eta)$. If $\psi(\eta) \notin v^*(\eta)$ we would find that η can not be a minimizer of $L(\eta, \psi(\cdot))$ as then $L(\eta, \psi(\eta_1)) < L(\eta, \psi(\eta))$.

Now we show that (l, ψ) is a proper non-degenerate composite loss if $\psi \in v^*$. By definition, (l, ψ) is proper if $\eta \in v^*_\psi(\eta)$. This is the case if and only if $L(\eta, \psi(\eta)) = \min_{q \in [0,1]} L(\eta, \psi(q))$. But this is the case if $\psi \in v^*$ since $v^*(\eta)$ is defined as the set of minimizers of $L(\eta, \cdot)$. The non-degenerate follows directly by definition. □

This lemma gives thus necessary and sufficient condition on our link ψ to lead to a proper loss function. The result is very similar to Corollary 12 and 14 found in [6]. Their corollaries state necessary and sufficient conditions on the link function, using the assumption that the loss has differentiable partial losses, which is an assumption we don't require.

In Section 5.6.2 we show that *strictly* proper losses, together with some additional assumptions, lead to consistent class probability estimates. So it is useful to know how to characterize those functions. The following lemma shows that a link function that turns a loss into strictly proper and non-degenerate CPE loss can be characterized again by the behavior of v^*.

5

Lemma 3. *Let $l : \{-1, 1\} \times \mathcal{V} \to [0, \infty)$ be a loss function and ψ a link function. A composite loss function (l, ψ) is then strictly proper and non-degenerate if and only if $\psi \in v^*$ and $v^*(\eta_1) \cap v^*(\eta_2) \cap \operatorname{Im} \psi = \emptyset$ for all pairwise different $\eta_1, \eta_2 \in [0, 1]$.*

Proof. By definition the composite loss is strictly proper if and only if $\eta = v^*_\psi(\eta)$. First we show that (l, ψ) is strictly proper and non-degenerate if $\psi \in v^*$ and $v^*(\eta_1) \cap v^*(\eta_2) = \emptyset$ for all $\eta_1, \eta_2 \in [0, 1]$. From Lemma 2 we know already that $\eta \in v^*_\psi(\eta)$, we only have to show that η is the only element in the set. For that assume that it is not the only element, so that there is a $\gamma \in [0, 1]$ such that $\gamma \in v^*_\psi(\eta)$. As in the proof of Lemma 2 one can conclude that $\psi(\gamma) \in v^*(\eta)$. But we also know, again from Lemma 2, that $\psi(\gamma) \in v^*(\gamma)$. That means that $\psi(\gamma) \in v^*(\eta) \cap v^*(\gamma) \cap \operatorname{Im} \psi \neq \emptyset$, which is a contradiction to our assumption.

Now we show that $\psi \in v^*$ and $v^*(\eta_1) \cap v^*(\eta_2) \cap \operatorname{Im} \psi = \emptyset$ for all $\eta_1, \eta_2 \in [0, 1]$ if (l, ψ) is strictly proper and non-degenerate. The relation $\psi \in v^*$ follows again from Lemma 2. We prove the second claim by contradiction and assume that there exist $\eta_1, \eta_2, \eta_3 \in [0, 1]$, all pairwise different, such that $\psi(\eta_3) \in v^*(\eta_1) \cap v^*(\eta_2)$. With this choice and using that ψ is strictly proper it follows that $\eta_3 = v^*_\psi(\eta_1)$ and $\eta_3 = v^*_\psi(\eta_2)$. That means that $\eta_1 = \eta_3 = \eta_2$ which is a contradiction. □

So if (l, ψ) is a strictly proper composite loss it will fulfill some sort of injectivity condition on the sets $v^*(\eta)$. With this we will be able to define an inverse ψ^{-1} on those sets, and this will be essentially our class probability estimator. With Lemma 3 we can connect every $v \in \mathcal{V}$ to a unique η_v by the unique relation $v \in v^*(\eta_v)$ if we assume that v^* *disjointly covers* $\mathcal{V}$ in the sense that

$$\bigcup_{\eta \in [0,1]} v^*(\eta) = \mathcal{V} \quad \text{and} \tag{5.6}$$

$$v^*(\eta_1) \cap v^*(\eta_2) = \emptyset \quad \forall\, \eta_1, \eta_2 \in [0, 1],\ \ \eta_1 \neq \eta_2. \tag{5.7}$$

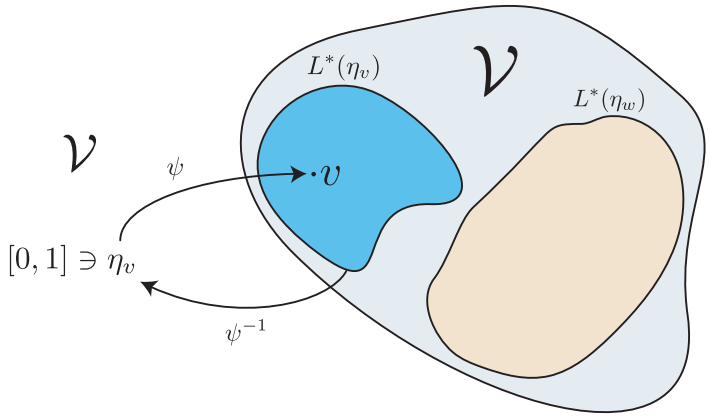


Figure 5.1: The way we generally think of the mapping ψ, ψ^{-1} and the sets v^* if (l,ψ) is non-degenerate and strictly proper. In those cases we can extend ψ^{-1} to the sets v^*. This is well defined as the sets $v^*(\eta_v)$ and $v^*(\eta_w)$ have empty intersection for different $\eta_v, \eta_w \in [0,1]$. Note that Lemma 3 guarantees that $\psi(\eta_v) \in v^*(\eta_v)$.

Note that we know from Lemma 3 that for strict properness it is sufficient for (l,ψ) that the disjoint property (5.7) only holds on $\operatorname{Im}\psi$, the image of ψ. This is merely a technicality and we will assume from now on that every strictly proper composite loss will satisfy (5.7). The covering property (5.6) on the other hand can be violated. This happens for example if we use the squared loss together with $\mathcal{V} = \mathbb{R}$. For the squared loss $v^*(\eta) = 2\eta - 1$, so it only covers the space $[-1,1]$.

If we assume, however, that the regularity properties (5.6) and (5.7) hold for a strictly proper non-degenerate composite loss (l,ψ) we can extend the domain of ψ^{-1} from $\operatorname{Im}\psi$ to the whole of $\mathcal{V}$, see also Figure 5.1 and the examples in Table 5.2.

Definition 3. *Let (l,ψ) be a strictly proper, non-degenerate composite loss and assume that v^* disjointly covers $\mathcal{V}$. We define, by abuse of notation, the inverse link function $\psi^{-1} : \mathcal{V} \to [0,1]$ by $\psi^{-1}(v) = \eta_v$, where η_v is the unique element in $[0,1]$ such that $v \in v^*(\eta_v)$.*

The requirements from the previous definition is what we consider the archetype of a composite loss that is suitable for probability estimation, although not all of the requirements are necessary. This motivates the following definition.

Definition 4. *We call a composite loss (l,ψ) a natural CPE loss if ψ is non-degenerate, v^* fulfills the disjoint cover property (5.6) and (5.7) and (l,ψ) is strictly proper.*

We now have all the necessary work done to make the following observation.

Corollary 5. *If (l,ψ) is a natural CPE loss, then $\psi^{-1} = v^{*-1}$.*

Proof. Let $v \in \mathcal{V}$ and $\eta \in [0,1]$ such that $v \in v^*(\eta)$. Then $v^{*-1}(v) = \eta$. As by the previous lemmas we know that $\psi(\eta) \in v^*(\eta)$ we have by Definition 3 that $\psi^{-1}(v) = \eta$. □

The corollary tells us that we can optimize our loss function over $\mathcal{V}$ to get $v^*(\eta)$ and then map this back with the inverse link ψ^{-1} to restore the class probability η. For this we once more refer to Figure 5.1. Remember that the set $v^*(\eta_v)$ is the set of all $v \in \mathcal{V}$ that minimize the loss if the true posterior probability was η_v. If we use a natural CPE loss (l, ψ) we know then that ψ^{-1} maps all those points back to η_v.

Given a predictor $f : \mathcal{X} \to \mathcal{V}$ this motivates to define an estimator of $\eta(x)$ as

$$\hat{\eta} = \hat{\eta}(x) = \psi^{-1}(f(x)). \tag{5.8}$$

In Section 5.6 we give conditions under which $\hat{\eta}(x)$ converges in probability towards $\eta(x)$ when using an empirical risk minimizer f_n as a prediction rule. More formally; Given any $\epsilon > 0$ we show that under certain conditions $\hat{\eta}_n(x) := \psi^{-1}(f_n(x))$ satisfies

$$P(|\hat{\eta}_n(X) - \eta(X)| > \epsilon) \xrightarrow{n \to \infty} 0, \tag{5.9}$$

where the probability is measured with respect to P. In the next section, however, we want to investigate first v^* and v^{*-1} for some commonly used loss functions.

5

5.5. ANALYSIS OF LOSS FUNCTIONS

We now give examples of some commonly used loss functions and analyze whether they are strictly proper or not, with the aid of Lemma 3. In Table 5.1 we summarize the loss functions we consider and a link function that turns the loss function into a strictly proper composite loss, if possible. Table 5.2 shows the corresponding functions v^* and v^{*-1}. That the link functions indeed fulfills the requirements can be checked with Lemma 3 . The behavior of the squared and squared hinge loss seems to be very similar. In Section 5.6.5, however, we point out an important difference.

Table 5.1: The different loss functions we consider in this chapter together with their link functions that turn them into CPE losses (if possible).

Loss Function	$l(y, v)$	$\psi(\eta)$
Squared	$(1 - yv)^2$	$2\eta - 1$
Logistic	$\ln(1 + e^{-vy})$	$\ln \frac{\eta}{1-\eta}$
Squared Hinge	$\max(0, 1 - vy)^2$	$2\eta - 1$
Hinge	$\max(0, 1 - vy)$	-
0-1	$I_{\{\text{sign}(vy) \neq 1\}}$	-

As already noted by Buja *et al.* [5], also Table 5.2 shows that the hinge loss is not suitable for class probability estimation. We observe that the intersections of $v^*(\eta)$ for

different $\eta \in [0,1]$ are not disjoint. By Lemma 3 we can conclude that there is no link ψ such that (l,ψ) is strictly proper. One way to fix this, proposed by [12] and similar by Platt [13], is to fit a logistic regressor on top of the support vector machine. Bartlett and Tewari [14] investigate the behavior of the hinge loss deeper by connecting the class probability estimation task to the sparseness of the predictor. The hinge loss is of course classification calibrated (essentially meaning that we find point-wise the correct label with it), so between our considered surrogate losses it is the only one that really directly solves the classification problem without implicitly estimating the class probability.

Table 5.2: The functions v^* and v^{*-1} for different loss functions, as well as the rate of convergence (5.14) when calculated with Corollary 7.

Loss Function	$v^{*-1}(v)$	$v^*(\eta)$	$\delta(\epsilon)$
Squared	$\frac{v+1}{2}$	$2\eta - 1$	ϵ^2
Logistic	$\frac{1}{1+e^{-v}}$	$\ln \frac{\eta}{1-\eta}$	$2\epsilon^2$
Squared Hinge	$T(\frac{v+1}{2})$	$\begin{cases} 2\eta-1, & \eta \in (0,1) \\ [1,\infty), & \eta = 1 \\ (-\infty,-1], & \eta = -1 \end{cases}$	ϵ^2
Hinge	$\begin{cases} \frac{1}{2} & v \in (-1,1) \\ (0,\frac{1}{2}) & v = -1 \\ (\frac{1}{2},1) & v = 1 \\ 1, & v > 1 \\ 0, & v < -1 \end{cases}$	$\begin{cases} \text{sign}(2\eta-1), & \eta \in (0,1) \setminus \frac{1}{2} \\ [-1,1] & \eta = \frac{1}{2} \\ [1,\infty), & \eta = 1 \\ (-\infty,-1], & \eta = -1 \end{cases}$	-
0-1	$\begin{cases} [\frac{1}{2},1] & \text{if } v \in (0,\infty) \\ [0,\frac{1}{2}], & v \in (-\infty,0) \\ \frac{1}{2}, & v = 0 \end{cases}$	$\begin{cases} (0,\infty), & \eta \in (\frac{1}{2},1] \\ (-\infty,0), & \eta \in [0,\frac{1}{2}) \\ \mathbb{R}, & \eta = \frac{1}{2} \end{cases}$	-

5.6. CONVERGENCE OF THE ESTIMATOR

We now prove that the estimator $\hat{\eta}(x)$ as defined in Equation (5.8) converges in probability and in the L_1-norm to the true class probability η whenever we use an empirical risk minimizer, for which we have excess risk bounds, as a prediction rule.

5.6.1. USING THE TRUE RISK MINIMIZER FOR ESTIMATION

Before we can investigate under which conditions an empirical risk minimizer can (asymptotically) retrieve $\eta(x)$ we need to investigate under which conditions the true risk minimizer

can retrieve it. In this subsection we formulate a theorem that gives necessary and sufficient conditions for that. Not surprisingly we basically require that our hypothesis class is rich enough so as to contain the class probability distribution already. Bartlett *et al.* [15] and similar works often avoid problems caused by restricted classes by assuming from the beginning that the hypothesis class consists of all measurable functions. This theorem relaxes this assumption for the purpose of class probability estimation.

In this setting we assume that we use a hypothesis class $\mathcal{F}$ where $f \in \mathcal{F}$ are functions $f : \mathcal{X} \to \mathcal{V}$. If we want to do class probability estimation we rescale those functions by composing them with the inverse link $\psi^{-1} : \mathcal{V} \to [0,1]$ so that we effectively use the hypothesis class $\psi^{-1}(\mathcal{F}) := \{\psi^{-1} \circ f \mid f \in \mathcal{F}\}$. We then get the following theorem about the possibility of retrieving the posterior with risk minimization.

Theorem 23. *Assume that (l, ψ) is a natural CPE loss function. Let*

$$f_0 = \operatorname*{arg\,min}_{f \in \mathcal{F}} \mathbb{E}[l(Y, f(X)].$$

Then $\psi^{-1}(f_0(x)) = \eta(x)$ almost surely if and only if $\eta \in \psi^{-1}(\mathcal{F})$.

Proof. If $\psi^{-1}(f_0(x)) = \eta(x)$ then $\eta \in \psi^{-1}(\mathcal{F})$ by the definition of that space.

For the other direction assume that $\eta \in \psi^{-1}(\mathcal{F})$. First observe that

$$\mathbb{E}_X[L(\eta(X), \psi(f(X))] = \mathbb{E}_{X,Y}[l(Y, \psi(f(X)))].$$

Since (l, ψ) is a natural CPE loss we know that $\eta(x)$ is the unique minimizer of $L(\eta(x), \psi(\cdot))$. Since $f_0(X)$ is a minimizer of $\mathbb{E}_{X,Y}[l(Y, \cdot)] = \mathbb{E}[L(\eta(X), \cdot)]$ it follows that $f_0 = \psi(\eta)$ almost surely. As (l, ψ) is regular, the inverse ψ^{-1} is well-defined and thus $\psi^{-1}(f_0) = \eta$. □

Following Theorem 23 we need to assume that our hypothesis class is flexible enough for consistent class probability estimation. We formulate this assumption as follows.

Assumption A Given a natural CPE loss (l, ψ) we assume that $\eta \in \psi^{-1}(\mathcal{F}) = \{\psi^{-1} \circ f \mid f \in \mathcal{F}\}$. In Subsection 5.6.3 we will deal with the case of misspecification, i.e. when $\eta \notin \psi^{-1}(\mathcal{F})$.

5.6.2. USING THE EMPIRICAL RISK MINIMIZER FOR ESTIMATION

In the previous section we considered the possibility of retrieving class probability estimates with the true risk minimizer. To move on to empirical risk minimizers we need the notion of excess risk bounds.

Definition 5. *Let $f_n : \mathcal{X} \to \mathbb{R}$ be any estimator of $f_0 \in \mathcal{F}$, which may depend on a sample of size n. We call*

$$B^{\mathcal{F}}(n, \gamma) : \mathbb{N} \to [0, \infty) \tag{5.10}$$

an excess risk bound for f_n if for all $\gamma > 0$ we have $B^{\mathcal{F}}(n, \gamma) \to 0$ for $n \to \infty$ and with probability of at least $1 - \gamma$ over the $n - sample$ we have

$$\mathbb{E}_X[\Delta L(\eta(X), f_n(X))] = \mathbb{E}_{X,Y}[l(Y, f_n(X)) - l(Y, f_0)] \leq B^{\mathcal{F}}(n, \gamma). \tag{5.11}$$

Excess risk bounds are typically in the order of $\left(\frac{\text{comp}(\mathcal{F})}{n}\right)^{\beta}$, where $\beta \in [0.5, 1]$ and $\text{comp}(\mathcal{F})$ is a notion of model complexity. Common measures for the model complexity are the VC dimension [16], Rademacher complexity [17] or ϵ-cover [18]. The existence of excess risk bounds is tied to the finiteness of any of those complexity notions. A lot of efforts in this line of research are made to find relations between the exponent β and the statistical learning problem given by $\mathcal{F}$, the loss l and the underlying distribution P. Conditions that ensure $\beta > \frac{1}{2}$ are often called easiness conditions, such as the Tsybakov condition [19] or the Bernstein condition [20]. Intuitively those conditions often state that the variance of our estimator gets smaller the closer we are to the optimal solution. For a in-depth discussion and some recent results we refer to the work of Grünwald and Mehta [21].

Excess risk bounds allow us to bound the expected value of $\Delta L(\eta(x), f_n(x))$ for a loss l, so in particular we can bound $\Delta L_\psi(\eta(x), \hat{\eta}(x))$ for a composite loss (l, ψ). We will show L_1-convergence by connecting the behavior of $\Delta L_\psi(\eta(x), \hat{\eta}(x))$ to $|\eta(x) - \hat{\eta}(x)|$. The following lemma introduces a condition that allows us to draw this connection.

Lemma 4. *Let (l, ψ) be a natural CPE loss. Assume that for all $\eta \in [0,1]$ the maps*

$$L_\psi^0(\eta, \cdot) := L_\psi(\eta, \cdot) \restriction_{[0,\eta]} : [0, \eta] \to \mathbb{R}$$

and

$$L_\psi^1(\eta, \cdot) := L_\psi(\eta, \cdot) \restriction_{[\eta,1]} : [\eta, 1] \to \mathbb{R}$$

are strictly monotonic, where $L_\psi(\eta, \cdot) \restriction_I$ refers to the restriction of the mapping $L_\psi(\eta, \cdot)$ to an interval I. This is the case iff $L_\psi(\eta, \cdot)$ is strictly convex with η as its minimizer. Then there exists for all $\epsilon > 0$ a $\delta = \delta(\epsilon) > 0$ such that for all $\eta, \hat{\eta} \in [0,1]$

$$|\Delta L_\psi(\eta, \hat{\eta})| < \delta \Rightarrow |\eta - \hat{\eta}| < \epsilon. \tag{5.12}$$

Proof. With the assumptions on $L_\psi^0(\eta, \cdot)$ and $L_\psi^1(\eta, \cdot)$ those maps have a well defined inverse mapping with their image as the domain and those inverse mappings are continuous [22]. That means in particular that for every $l, \hat{l} \in \operatorname{Im} L_\psi^0(\eta, \cdot)$ and for all $\epsilon > 0$ there exists a $\delta > 0$ such that

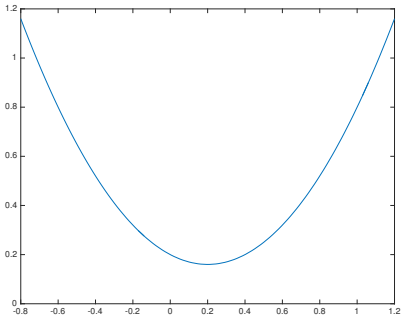
$$|\hat{l} - l| < \delta \Rightarrow |{L_\psi^0}^{-1}(\eta, \hat{l}) - {L_\psi^0}^{-1}(\eta, l)| < \epsilon \tag{5.13}$$

and similar for $L_\psi^1(\eta, \cdot)$. W.l.o.g assume now that $\hat{\eta} < \eta$ so that $\hat{\eta} \in [0, \eta]$. Then we set $l = L_\psi^0(\eta, \eta)$ and $\hat{l} = L_\psi^0(\eta, \hat{\eta})$. Plugging this into (5.13) we get the following relation.

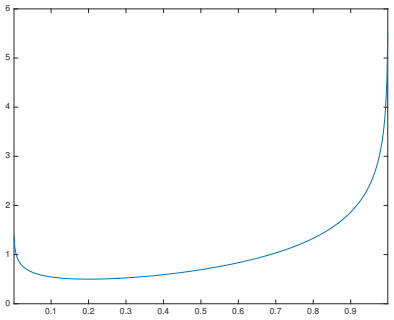
$$\begin{aligned} &|\Delta L_\psi(\eta, \hat{\eta})| = |L_\psi^0(\eta, \hat{\eta}) - L_\psi^0(\eta, \eta)| < \delta \\ \Rightarrow &|\hat{\eta} - \eta| = |{L_\psi^0}^{-1}(\eta, \hat{l}) - {L_\psi^0}^{-1}(\eta, l)| < \epsilon \end{aligned}$$

□

The map $L_\psi^0(\eta, \cdot)$ captures the behavior of the loss when η is the true class probability and we predict a class probability less than η. Similarly $L_\psi^1(\eta, \cdot)$ captures the behavior when we predict a class probability bigger than η, see also Figure 5.2. In Corollary 7, further below, we draw a connection between $\delta(\epsilon)$ and the modulus of continuity of the inverse functions of $L_\psi^1(\eta, \cdot)$ and $L_\psi^0(\eta, \cdot)$. The function $\delta(\epsilon)$ plays an important role in the convergence rate of the estimator $\hat{\eta}(x)$ as described in the next theorem.



(a) The map $L_\psi(\eta,\cdot)$ for $\eta = 0.2$ and l being the squared loss.



(b) The map $L_\psi(\eta,\cdot)$ for $\eta = 0.2$ and l being the logistic loss.

Figure 5.2: The map $L_\psi(\eta,\cdot)$ for the squared and the logistic loss. The two maps $L^0_\psi(\eta,\cdot)$ and $L^1_\psi(\eta,\cdot)$ split it into the parts left and right of η.

5

Theorem 24. *Let (l,ψ) be a natural CPE loss and assume Assumption A holds. Furthermore let $B^{\mathcal{F}}(n,\gamma)$ be an excess risk bound for f_n and assume that $L_\psi(\eta,\cdot)$ is strictly convex with η as its minimizer. Then there exists a mapping $\delta(\epsilon) : [0,1] \to \mathbb{R}$ such that for $\hat{\eta}_n(x) := \psi^{-1}(f_n(x))$ we have with probability of at least $1-\gamma$ that*

$$P(|\eta(X) - \hat{\eta}_n(X)| > \epsilon) \leq \frac{B^{\mathcal{F}}(n,\gamma)}{\delta(\epsilon)}. \tag{5.14}$$

Proof. Using Lemma 4 for the first inequality, Markov's Inequality for the second and the excess risk bound for the third inequality it follows that

$$P(|\eta(X) - \hat{\eta}_n(X)| > \epsilon) \leq P(\Delta L_\psi(\eta(X), \hat{\eta}_n(X)) > \delta)$$

$$= P(\Delta L(\eta(X), f_n(X)) > \delta) \leq \frac{\mathbb{E}[\Delta L(\eta(X), f_n(X))]}{\delta(\epsilon)} \leq \frac{B^{\mathcal{F}}(n,\gamma)}{\delta(\epsilon)}.$$

□

This theorem gives us directly the earlier claimed asymptotic convergence result.

Corollary 6. *Under the assumptions of Theorem 24 we have that $\hat{\eta}_n(x) = \psi^{-1}(f_n(x))$ converges in probability and L_1-norm to $\eta(x)$ with probability 1.*

We do not have to restrict ourselves to asymptotic results though. Theorem 24 can also be used to derive rate of convergences as we will see in Subsection 5.6.4. But before that we briefly want to address the case of misspecification, i.e. the case when Assumption A does not hold.

5.6.3. MISSPECIFICATION

The case of misspecification can be dealt with once we assume that L_ψ^* has a gradient. If this holds then Reid and Williamson [6] show the identity

$$\Delta L_\psi(\eta, \hat{\eta}) = D_{-L_\psi^*}(\eta, \hat{\eta}) \tag{5.15}$$

where $D_{-L_\psi^*}(\eta, \hat{\eta})$ is the with $-L_\psi^*$ associated Bregman divergence between η and $\hat{\eta}$. Excess risk bounds on $\Delta L_\psi(\eta, \hat{\eta})$ translate then into bounds on the Bregman divergence between η and $\hat{\eta}$ and asymptotically we approach the best class probability estimate in terms of this divergence.

5.6.4. RATE OF CONVERGENCE

For the rate of convergence it is crucial to investigate the function $\delta(\epsilon)$ from Inequality (5.14). One way to analyze this is to study the modulus of continuity of the inverse functions of $L_\psi^0(\eta, \cdot)$ and $L_\psi^1(\eta, \cdot)$:

Definition 6. *Let $\omega : [0, \infty] \to [0, \infty]$ be a monotonically increasing function. Let $I \subset \mathbb{R}$ be an interval. A function $g : I \to \mathbb{R}$ admits ω as a modulus of continuity at $x \in I$ if and only if*

$$|g(x) - g(y)| \leq \omega(|x - y|)$$

for all $y \in I$.

For example Hölder and Lipschitz continuity are particular moduli of continuity. This notion allows us a to draw the following connection between ϵ and $\delta(\epsilon)$.

Corollary 7. *Let (l, ψ) be a natural CPE loss and let $\omega : [0, \infty] \to [0, \infty]$ be a monotonically increasing function. Assume that for all $\eta \in [0, 1]$ the mappings ${L_\psi^0}^{-1}(\eta, \cdot)$ and ${L_\psi^1}^{-1}(\eta, \cdot)$ admit ω as a modulus of continuity at η. Then $\delta(\epsilon) := \omega^{-1}(\epsilon)$ is a mapping such that Implication (5.12) holds.*

Proof. W.l.o.g. assume that $\hat{\eta} \in [0, \eta]$. Let $\hat{l} = L_\psi^0(\eta, \hat{\eta})$ and $l = L_\psi^0(\eta, \eta)$. By using that ${L_\psi^0}^{-1}(\eta, \cdot)$ admits ω as a modulus of continuity we have

$$|{L_\psi^0}^{-1}(\eta, l) - {L_\psi^0}^{-1}(\eta, \hat{l})| \leq \omega(|l - \hat{l}|).$$

Plugging in the definition of $\hat{l}$ and l this means that

$$|\hat{\eta} - \eta| \leq \omega(\Delta L_\psi(\eta, \hat{\eta})).$$

Using the monotonicity of ω it follows that if $\Delta L_\psi(\eta, \hat{\eta}) \leq \delta(\epsilon) = \omega^{-1}(\epsilon)$, then

$$|\eta - \hat{\eta}| \leq \omega(\Delta L_\psi(\eta, \hat{\eta})) \leq \omega(\omega^{-1}(\epsilon)) = \epsilon.$$

This is exactly the Implication (5.12). □

Note that it follows from the proof that finding a modulus of continuity ω for ${L_\psi^0}^{-1}(\eta,\cdot)$ and ${L_\psi^1}^{-1}(\eta,\cdot)$ can be done by showing the bound $|\hat{\eta}-\eta| \leq \omega(\Delta L_\psi(\eta,\hat{\eta}))$. We will use that in the following examples, where we analyze $\delta(\epsilon)$ for the squared (hinge) loss and the logistic loss. We show that those loss functions lead to a modulus of continuity given by the square root times a constant. [11] calls loss functions that admit this modulus of continuity *strongly-proper* loss functions. The following analysis can thus be found there in more detail and for a few more examples. We will use for simplicity versions of the losses that do not need a link function, and are already CPE losses, the results are summarized in Table 5.2.

Example: Squared Loss and Squared Hinge Loss Let $l(y,\hat{\eta})$ be given by the partial loss functions $l(1,\hat{\eta}) = (1-\hat{\eta})^2$ and $l(-1,\hat{\eta}) = \hat{\eta}^2$. We can derive that $\Delta L(\eta,\hat{\eta}) = (\eta-\hat{\eta})^2$. With this we can directly bound

$$|\hat{\eta}-\eta| \leq \sqrt{\Delta L(\eta,\hat{\eta})}$$

and thus choose $\delta(\epsilon)$ as the inverse of the square-root function, so that $\delta(\epsilon) = \epsilon^2$. The analysis for the squared hinge loss is the same as this version of the squared loss is already a CPE loss.

Example: Logistic Loss Let $l(y,\hat{\eta})$ be given by the partial loss functions $l(1,\hat{\eta}) = -\ln(\hat{\eta})$ and $l(-1,\hat{\eta}) = -\ln(1-\hat{\eta})$. One can derive that $\Delta L(\eta,\hat{\eta}) = -\eta\ln(\frac{\hat{\eta}}{\eta}) - (1-\eta)\ln(\frac{1-\hat{\eta}}{\eta})$. One can show the bound $|\eta-\hat{\eta}| \leq \sqrt{\frac{1}{2}\Delta L(\eta,\hat{\eta})}$, so that we can choose $\delta(\epsilon) = 2\epsilon^2$.

5.6.5. SQUARED LOSS VS SQUARED HINGE LOSS

In this section we will subscript previously defined entities with S and SH for the squared and square hinge loss respectively. When using squared loss vs the squared hinge loss for class probability estimation there is one big difference in the inverse of the link function, namely its domain. The inverse link function is a map $\psi_S^{-1}: \mathcal{V} \to [0,1]$. If we use the square loss we implicitly chose $\mathcal{V} = [-1,1]$ since this is the range of ψ_S. The range of ψ_{SH} on the other hand is $\mathcal{V} = \mathbb{R}$. That means that if we want to use the squared loss for class probability estimation we really have to parametrize our prediction functions $f: \mathcal{X} \to [-1,1]$, a simple linear model for example would usually not fit this assumption as the range of those models can be outside of $[-1,1]$. For the squared hinge loss on the other hand we can allow for functions $f: \mathcal{X} \to \mathbb{R}$.

[23] proposes to just truncate the inverse link for the squared loss, so using the same inverse link as for the squared hinge loss. This is fine as long as our hypothesis class is flexible enough, but leads to problems if that is not the case as the following example shows.

Assume we are given three one-dimensional data points $x_1 = -1, x_2 = 0, x_3 = 3$ together with their true class probabilities $\eta(x_1) = 0, \eta(x_2) = 1/3, \eta(x_3) = 1$. We want to learn this classification with linear models, which are two-dimensional after including a bias term. That means that $\mathcal{F} = \{f: \mathcal{X} \to \mathbb{R} \mid \exists w_1, w_2 \in \mathbb{R}: f(x) = w_1 x + w_2\}$. One can check that in case of the squared hinge loss function we can recover the true class probabilities with the

linear function given by $(w_1, w_2) = (2, -\frac{1}{4})$. By Theorem 23 we know then that an optimal solution f_0 is also able to recover the true class probabilities.

The squared loss has after truncating the following problem. Although the linear function $(w_1, w_2) = (2, -\frac{1}{4})$ is part of $\psi_S^{-1}(\mathcal{F})$, after truncating, it will not be found back as an optimal solution f_0. One can instead check that for the given example the true risk minimizer is given by $f_0 = (\frac{19}{39}, -\frac{17}{39})$. And this hypothesis does not recover the true class probabilities. This might appear as a contradiction to Theorem 23. But the problem arises because we use a different link function than the one associated to the square loss.

5.7. DISCUSSION AND CONCLUSION

The starting point of this chapter is the question if one can retrieve consistently a class probability estimate based on ERM in a consistent way. To answer this question we draw from earlier work on proper scoring rules and excess risk bounds. Lemmas 2 and 3, our first results, characterize strictly proper composite loss functions in terms of their link function. Based on those lemmas, we subsequently derive necessary and sufficient conditions for retrieving the true class probability with ERM as formulated in Theorem 23. Next to some regularity conditions on the loss function, we show that to retrieve the true probabilities we essentially need that they are already part of our hypothesis class $\mathcal{F}$, which, in a way, is not surprising.

In Section 5.6 we use the results from the previous sections and theory about excess risk bounds to state our main consistency and finite sample size results. We show that consistency arises whenever we use strictly proper (composite) loss functions, our hypothesis class is flexible enough, and we have excess risk bounds. This is the case, for example, whenever one of the complexity notions mentioned in Section 5.6 is finite. We then discuss the relation between the finite sample size behavior of the excess risk bound and the probability estimate and examine this relation for two example loss functions.

In Lemma 4 we introduce fairly general conditions under which a composite loss function (l, ψ) leads to a consistent class probability estimator. In particular we have a condition on the conditional risk $L_\psi(\eta, \cdot)$, see also Figure 5.2. Based on that we derive in Corollary 7 conditions which allow us to analyze the convergence rate for different loss functions. In the corollary we don't distinguish between $L_\psi^0(\eta, \cdot)$ and $L_\psi^1(\eta, \cdot)$, which leads to the same convergence rate for predicting values left and right from η. But the modulus of continuity for those two functions can be really different, especially when using asymmetric proper scoring rules [24]. We believe that by analyzing $L_\psi^0(\eta, \cdot)$ and $L_\psi^1(\eta, \cdot)$ individually one can extend our work to analyze the convergence behavior of asymmetric scoring rules in more detail.

As stated from the outset, one of our main goals is to emphasize the tight relationships between empirical risk minimization and class probability estimation in a distilled and compact version. The concepts of link functions and the relation between them and empirical risk minimization do not get the attention they deserve and are thus reinvented from time to time. Many of those concepts appear for example in the great analysis of Zhang [9] without any explicit reference to proper scoring rules.

REFERENCES

[1] D. D. Lewis and J. Catlett, *Heterogeneous uncertainty sampling for supervised learning,* in *In Proceedings of the Eleventh International Conference on Machine Learning* (New Brunswick, NJ, USA, 1994) pp. 148–156.

[2] N. Roy and A. McCallum, *Toward optimal active learning through sampling estimation of error reduction,* in *Proceedings of the Eighteenth International Conference on Machine Learning* (Williams College, Williamstown, MA, USA, 2001) pp. 441–448.

[3] Y. Grandvalet and Y. Bengio, *Semi-supervised learning by entropy minimization,* in *Advances in Neural Information Processing Systems 17* (Vancouver, British Columbia, Canada, 2004) pp. 529–536.

[4] L. Györfi, M. Kohler, A. Krzyzak, and H. Walk, *A Distribution-Free Theory of Nonparametric Regression.*, Springer series in statistics (Springer, 2002) pp. I–XVI, 1–647.

[5] A. Buja, W. Stuetzle, and Y. Shen, *Loss Functions for Binary Class Probability Estimation and Classification: Structure and Applications*, Tech. Rep. (University Washington, 2005).

[6] M. D. Reid and R. C. Williamson, *Composite binary losses,* Journal of Machine Learning Research **11**, 2387 (2010).

[7] M. D. Reid and R. C. Williamson, *Information, divergence and risk for binary experiments,* Journal of Machine Learning Research **12**, 731 (2011).

[8] M. Telgarsky, M. Dudík, and R. Schapire, *Convex risk minimization and conditional probability estimation,* in *Proceedings of The 28th Conference on Learning Theory* (Paris, France, 2015) pp. 1629–1682.

[9] T. Zhang, *Statistical behavior and consistency of classification methods based on convex risk minimization,* The Annals of Statistics **32**, 56 (2004).

[10] A. Agarwal and S. Agarwal, *On consistent surrogate risk minimization and property elicitation,* in *Proceedings of The 28th Conference on Learning Theory* (Paris, France, 2015) pp. 4–22.

[11] S. Agarwal, *Surrogate regret bounds for bipartite ranking via strongly proper losses,* Journal of Machine Learning Research **15**, 1653 (2014).

[12] R. P. Duin and D. M. Tax, *Classifier conditional posterior probabilities,* in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)* (Sydney, NSW, Australia, 1998) pp. 611–619.

[13] J. C. Platt, *Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods,* in *Advances in Large Margin Classifiers* (The MIT Press, Cambridge, MA, USA, 1999) pp. 61–74.

[14] P. L. Bartlett and A. Tewari, *Sparseness versus estimating conditional probabilities: Some asymptotic results,* in *17th Annual Conference on Learning Theory*, edited by J. Shawe-Taylor and Y. Singer (Banff, Canada, 2004) pp. 564–578.

[15] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, *Convexity, classification, and risk bounds,* Journal of the American Statistical Association **101**, 138 (2006).

[16] V. N. Vapnik, *Statistical Learning Theory* (Wiley-Interscience, 1998).

[17] P. L. Bartlett, O. Bousquet, and S. Mendelson, *Local rademacher complexities,* The Annals of Statistics **33**, 1497 (2005).

[18] G. M. Benedek and A. Itai, *Learnability with respect to fixed distributions,* Theory of Computer Science **86**, 377 (1991).

[19] A. B. Tsybakov, *Optimal aggregation of classifiers in statistical learning,* The Annals of Statistics **32**, 135 (2004).

[20] J.-Y. Audibert, *Une approche PAC-bayésienne de la théorie statistique de l'apprentissage*, Ph.D. thesis, Université Paris 6 (2004).

[21] P. D. Grünwald and N. A. Mehta, *Fast rates with unbounded losses,* The Computing Research Repository **abs/1605.00252** (2016).

[22] H. Hoffmann, *On the continuity of the inverses of strictly monotonic functions.* Bulletin of the Irish Mathematical Society **75**, 45 (2015).

[23] M. Sugiyama, *Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting,* IEICE Transactions **93-D**, 2690 (2010).

[24] R. L. Winkler, *Evaluating probabilities: Asymmetric scoring rules,* Management Science **40**, 1395 (1994).

6

Open Problem: Monotonicity of Learning

This chapter poses the question to what extent a learning algorithm behaves monotonically in the following sense: does it perform better, in expectation, when adding one instance to the training set? We focus on empirical risk minimization and illustrate this property with several examples, two where it does hold and two where it does not. We also relate it to the notion of PAC-learnability.

Parts of this chapter have been published in the proceedings of the 32nd Annual Conference on Learning Theory [1].

6.1. INTRODUCTION.

Recently, there has been an increasing amount of attention on machine learning algorithms that are presently referred to as robust or safe, meaning that even when assumptions are violated, performance will not degrade significantly [2]. The focus is mostly on settings that are slightly different from supervised learning such as online learning [3], domain adaptation [4] and semi-supervised learning [5]. The open problem presented here makes the point that such robustness and safety properties are not even fully understood for standard supervised learning and density estimation.

We focus on what we will refer to as the *monotonicity* of a learner's performance: given one additional training instance, to what extent can we expect a learner to improve? Or, equivalently, when is the so-called learning curve monotone [6]? While this property is undoubtedly desirable, and most of us expect such behavior, there are surprising counterexamples. This open problem asks to unravel this behavior.

6.2. PRELIMINARIES AND RELATED WORK.

Let $S_n = (z_1, \ldots, z_n)$ be a training set of size n, sampled i.i.d. from an (unknown) distribution D over a domain $\mathcal{Z}$. The learner A we consider performs *empirical risk minimization* (ERM). Its output is $A(S_n)$, i.e., a hypothesis h from a prespecified set $\mathcal{H}$ that minimizes the empirical risk over S_n based on a loss function $\mathcal{L} : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}$. In statistical learning, performance is measured through this loss and the aim is to minimize the true risk

$$L_D(h) = \mathop{\mathbb{E}}_{z \sim D} \mathcal{L}(h, z).$$

One can define classification problems, regression, and density estimation in such terms.

Before we formally introduce the concept of monotonicity, we mention related works that already report on non-monotone learning behavior. Duin [7] and Opper and Kinzel [8] describe the so-called peaking phenomenon for classification: when the dimensionality is approximately equal to the size of the training set, the risk in terms of the zero-one loss and mean squared error has a maximum (it peaks). This happens for models that require estimates of the (pseudo-)inverse of the covariance matrix [9], such as linear regression.

Loog and Duin [10] describe what they call dipping: the evaluation risk attains a global minimum for some finite n. Even for $n \to \infty$ the risk never recovers. This phenomenon can occur when there is a mismatch between target (e.g. zero-one) and surrogate loss (e.g. hinge). Ben-David *et al.* [11] analyze this mismatch between surrogate and zero-one loss in more detail.

We focus on the setting where the loss the learner optimizes matches the loss it is evaluated with. Thus the observed behavior in our examples cannot be explained through the dipping phenomenon. This makes our findings more unexpected and the open problem more appealing. Note, indeed, that our learner A (performing ERM) is implicitly associated with a specific loss $\mathcal{L}$ and set $\mathcal{H}$.

6.3. THE MONOTONICITY PROPERTY

The idea is that with an additional instance a learner should improve its performance in expectation over the training set. We need the following building block.

Definition 7 (local monotonicity)**.** *A learner A is locally or (D, n)-monotone with respect to a distribution D and an $n \in \mathbb{N}$ if*

$$\mathop{\mathbb{E}}_{S_{n+1}\sim D^{n+1}} L_D(A(S_{n+1})) \leq \mathop{\mathbb{E}}_{S_n\sim D^n} L_D(A(S_n)).$$

Now we can construct stronger desired properties. We generally want monotonicity for all n. Since the distribution D is unknown, we want local monotonicity to hold for any D on the domain $\mathcal{Z}$.

Definition 8 ($\mathcal{Z}$-monotonicity)**.** *A learner A is $\mathcal{Z}$-monotone if, for all $n \in \mathbb{N}$ and distributions D on $\mathcal{Z}$, it is (D, n)-monotone.*

6.4. EXAMPLES

We now turn to some illustrations and consider to what extent they are $\mathcal{Z}$-monotone. In the remainder, we refer to $\mathcal{Z}$-monotone as monotone. It will be clear from the context what $\mathcal{Z}$ is.

Example I: mean estimation of a normal distribution (monotone). We perform density estimation with a normal distribution with *fixed variance* $\sigma^2 > 0$ and *unknown mean*. The hypothesis class is $\mathcal{H}_\sigma = \left\{h : z \mapsto \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(z-\mu)^2}{2\sigma^2}\right) \mid \mu \in \mathbb{R}\right\}$. We choose the domain $\mathcal{Z} \subset [-1, 1]$. This choice ensures that any distribution D has a finite mean and finite variance. We use negative log-likelihood as loss. Thus ERM is equivalent to maximum likelihood (ML) estimation for this setting. The optimum that ERM finds is $\mu = \frac{1}{n}\sum_i z_i$. The expected risk equals

$$\mathop{\mathbb{E}}_{S_n\sim D^n} L_D(A(S)) = \frac{1}{2}\log(2\pi\sigma^2) + \frac{\sigma_D^2}{2\sigma^2}\left(1 + \frac{1}{n}\right),$$

where σ_D^2 is the true variance of D. So the expected risk decreases monotonically in n.

Proof. For brevity the expectations are now only indicated with the random variable, but not its distribution. We use negative log-likelihood as the loss and with this the expected risk can be computed to

$$\begin{aligned}
&\mathbb{E}_{S_n}\mathbb{E}_Z\left(-log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \frac{(Z-\mu(S_n))^2}{2\sigma^2}\right)\\
=&-log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \frac{1}{2\sigma^2}\mathbb{E}_{S_n}\mathbb{E}_Z\left(Z-\mu(S_n)\right)^2\\
=&-log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \frac{1}{2\sigma^2}\mathbb{E}_{S_n}\mathbb{E}_Z\left(Z^2+\mu(S_n)^2-2Z\mu(S_n)\right).
\end{aligned}$$

Here Z is D-distributed random variable. We solve the double expectation on the right hand side term by term. The first term can be computed to

$$\mathbb{E}_{S_n}\mathbb{E}_Z Z^2 = \mathbb{E}_Z Z^2 = \mathbb{V}_Z Z + (\mathbb{E}_Z Z)^2 = \sigma_D^2 + \mu^2,$$

where $\mathbb{V}$ indicates the variance. Note that for the last step we use that $\mu(S_n)$ is unbiased.

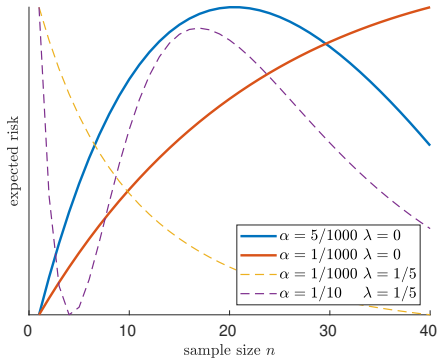


Figure 6.1: Non-monotone behavior as observed in Example III.

The second term can be computed to

$$\mathbb{E}_{S_n}\mathbb{E}_Z\mu(S_n)^2 = \mathbb{E}_{S_n}\mu(S_n)^2 = \mathbb{V}_{S_n}\mu(S_n) + (\mathbb{E}_{S_n}\mu(S_n))^2 = \frac{\sigma_D^2}{n} + \mu^2. \tag{6.1}$$

The third term finally is then computed as

$$-\mathbb{E}_{S_n}\mathbb{E}_Z 2Z\mu(S_n) = -(\mathbb{E}_{S_n}\mu(S_n))(\mathbb{E}_Z 2Z) = -\mu(2\mu) = -2\mu^2. \tag{6.2}$$

Combining all the above we obtain:

$$\begin{aligned}
&\mathbb{E}_{S_n}\mathbb{E}_Z\left(-log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \frac{(x-\mu(S_n))^2}{2\sigma^2}\right) \\
&= -log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \frac{1}{2\sigma^2}\left(\frac{\sigma_D^2}{n} + \mu^2 + \mathbb{V}_Z Z + \mu^2 - 2\mu^2\right) \\
&= -log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \frac{\sigma_D^2}{2\sigma^2}\left(\frac{1}{n} + 1\right)
\end{aligned}$$

□

Example II: variance estimation of a normal distribution (not monotone). We take the same domain and loss function as in Example I, but now estimate the variance, while keeping the mean fixed to 0. The hypothesis set is $\mathcal{H}_{\mu=0} = \left\{h : z \mapsto \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{z^2}{2\sigma^2}\right) \mid \sigma > 0\right\}$ and the ML estimate equals $\sigma = \frac{1}{n}\sum_i z_i^2$. This example does not obey the monotone principle. Consider a distribution D that only has support on $\{1, \frac{1}{10}\}$. Let D be given by the probability mass function $p(1) = \alpha$ and $p(\frac{1}{10}) = 1 - \alpha$. For $0 < \alpha < 0.0235$ one can then compute that $L_D(A(S_1)) < L_D(A(S_2))$, demonstrating that the monotonicity property does not hold.

Example III: linear regression (not monotone). Take $\mathcal{H} = \{h \mapsto wx | w \in \mathbb{R}\}$ as hypothesis set and use the mean squared error as loss function. We choose the domain $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, with $\mathcal{X} \subset [-1, 1]$ and $\mathcal{Y} \subset [0, 1]$. We define D through a probability mass function $p(x, y)$. Take $p(\frac{1}{10}, 1) = 1 - \alpha$ and $p(1, 1) = \alpha$, and $p(x, y) = 0$ otherwise. An exact numerical calculation shows that $\mathbb{E}_{S_1} L_D(A(S_1)) < \mathbb{E}_{S_2} L_D(A(S_2))$ for $0 < \alpha < 0.0047$. This shows this learner is not monotone.

Figure 6.1 plots a rescaled version of the expected risk against the sample size n for several settings. The thick lines correspond to ERM. First of all, observe that by changing α, we can shift the peak. This shows that the behavior is unrelated to the peaking behavior [7], since peaking would occur at $n \approx d = 1$. Second, if we add λI to the empirical covariance matrix, which corresponds to L_2-regularization of w, we still observe non-monotone behavior, now even for larger values of α (see the dashed lines in Figure 6.1).

Example IV: the memorize algorithm (monotone). This binary classifier was introduced by Ben-David *et al.* [12]. This learner, when evaluated on a test input object x that is also present in the training set, returns the label of said training object. In case multiple training examples share the same x, the majority voted label is returned. In case the test object is not present in the training set, a default label is returned. This learner is monotone for any distribution under the zero-one loss as it only updates its decision on points that it observes.

6.5. RELATION TO LEARNABILITY

Definition 9 (Agnostic PAC Learnability [6])**.** *$\mathcal{H}$ is agnostic PAC learnable if there exist a function $m_{\mathcal{H}} : (0, 1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: for every $\epsilon, \delta \in (0, 1)$ and for every distribution D over Z, when running A on $n \geq n_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. samples, with probability of at least $1 - \delta$ (over the choice of S_n),*

$$L_D(A(S_n)) - \min_{h^* \in \mathcal{H}} L_D(h^*) \leq \epsilon.$$

From learning theory we know that if the hypothesis class has finite VC-dimension (or other appropriate complexity), the excess risk of ERM is bounded. This bound will be tighter given a larger training set size n. PAC bounds hold with a particular probability, while we are concerned with the risk in expectation over the sample. However, even bounds that hold in expectation over the training sample will not rule out non-monotone behavior. The expected risk can go up as long as the expected risk stays below the upper bound. Thus high probability or expected risk bounds are insufficient to guarantee monotonicity.

This is illustrated by our examples: Example VI is monotone but is not learnable [6]. Example III is learnable if a regularizer is added to the objective of ERM or if the hypothesis space $\mathcal{H}$ is restricted such that the norm of w is bounded. However, as we have seen in Figure 6.1, we still can observe non-monotone behavior in that case.

6.6. OPEN PROBLEM(S)

First and foremost, we are interested to identify, especially for commonly employed learners, on which domains $\mathcal{Z}$ they will or may not act monotonically. In view of the peaking

behavior, $\mathcal{Z}$-monotonicity for all n may be too strong for some settings. Perhaps monotonicity is only possible if n is larger than some N that may depend on $\mathcal{Z}$ and A. For Examples II and III it is an open problem whether they satisfy this weaker notion, and for which (smallest) N this notion is satisfied. Other related notions of monotonicity may also be of interest. For example, instead of demanding a lower loss, we may require that the loss does not degrade too much. Or we can demand the property to hold with high probability with respect to both samples.

More generally, we may ask: why and how does this behavior occur? And maybe more importantly: how can we provably avoid non-monotone behavior? What conditions does a learner need to satisfy to be monotone? Perhaps particular loss functions lead to monotone learners? What if we allow for learning under regularization or other strategies deviating from strict ERM, for example improper learners or randomized decision rules?

Perhaps it is always possible to find a D for a given $\mathcal{Z}$ on which learners are non-monotone. In that case, is it possible to avoid non-monotone behavior under some assumptions on D? Realizability or well-specification could be good candidate-assumptions on D. In fact, this raises the issue to what extent well-specified statistical models can actually be proven to behave monotonically. For instance, is Example II monotone if the problem is well-specified?

All in all, we believe the question of monotonicity of learning offers various tantalizing questions to study, some of which may yet have to be formulated.

REFERENCES

[1] T. Viering, A. Mey, and M. Loog, *Open problem: Monotonicity of learning,* in *Proceedings of the Thirty-Second Conference on Learning Theory* (Phoenix, USA, 2019) pp. 3198–3201.

[2] M. Loog, *Constrained parameter estimation for semi-supervised learning: the case of the nearest mean classifier,* in *ECML PKDD 2010* (Barcelona, Spain, 2010) pp. 291–304.

[3] W. M. Koolen, P. Grünwald, and T. van Erven, *Combining adversarial guarantees and stochastic fast rates in online learning,* in *Advances in Neural Information Processing Systems 29* (Barcelona, Spain, 2016) pp. 4457–4465.

[4] A. Liu, L. Reyzin, and B. D. Ziebart, *Shift-pessimistic Active Learning Using Robust Bias-aware Prediction,* in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (Austin, Texas, USA, 2015) pp. 2764–2770.

[5] J. H. Krijthe and M. Loog, *Projected estimators for robust semi-supervised classification,* Machine Learning **106**, 993 (2017).

[6] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, New York, NY, USA, 2014).

[7] R. P. W. Duin, *Small sample size generalization,* in *Proceedings of the 9th Scandinavian Conference on Image Analysis* (Uppsala, Sweden, 1995) pp. 957–964.

[8] M. Opper and W. Kinzel, *Statistical mechanics of generalization,* in *Models of neural networks III* (Springer, 1996) pp. 151–209.

[9] S. Raudys and R. P. W. Duin, *Expected classification error of the Fisher linear classifier with pseudo-inverse covariance matrix,* Pattern recognition letters **19**, 385 (1998).

[10] M. Loog and R. P. W. Duin, *The dipping phenomenon,* in *Proceedings of the IAPR S+SSPR* (Hiroshima, Japan, 2012) pp. 310–317.

[11] S. Ben-David, D. Loker, N. Srebro, and K. Sridharan, *Minimizing The Misclassification Error Rate Using a Surrogate Convex Loss,* in *Proceedings of the 29th International Conference on Machine Learning* (Edinburgh, Scotland, UK, 2012) pp. 1863–1870.

[12] S. Ben-David, N. Srebro, and R. Urner, *Universal learning vs. no free lunch results,* in *Philosophy and Machine Learning Workshop NIPS* (Granada, Spain, 2011).

7

CONCLUSION

Concluding the thesis we discuss possible extensions of, and relations between, our work as well as its implications for the field. We will start with a work in progress that tries to include knowledge of a causal structure for enhanced semi-supervised learning. We then discuss the implications of our complexity analysis of manifold-regularization from Chapter 3, in view of our survey from Chapter 2, as well as some open problems left by the analysis. After that we connect the findings of Chapters 4 and 5. We then relate the open problem of Chapter 6 to the field of semi-supervised learning and discuss extensions of this work. Finally we discuss the current trends in semi-supervised learning and how they relate to this thesis.

7.1. FURTHER WORK USING CAUSAL KNOWLEDGE

In our review in Section 2.3.1 we briefly discuss the impossibility of semi-supervised learning in case that the labels Y are caused by the features X_C within the framework of a simple functional causal model, see also Figure 2.2. As this is not the case for features X_E that are *caused* by the label, we made use of this observation to construct a model in cases where one can split the feature space into causal features X_C and effect features X_E [1]. The idea is to use a model that jointly models X_E, Y but only conditionally models $Y|X_C$. This reflects that want to take $P(X_E)$ into account for predictions, but not $P(X_C)$. In a current research direction we are relating this type of modeling to semi-supervised assumptions [2], as for example the cluster assumption, which in this context roughly means that two points that are close in the feature space should carry the same label. While in the naive case we assume that this property holds for the joint observation X_E, X_C, we assume in our extension [2] that it holds for the conditional distribution $X_E|X_C$. For that we assume that there are functional relationships f_0, f_1 between X_C and X_E, with this relationship depending on the labels 0 and 1. Given this relationship one can define a new notion of proximity between points from X_E, see also Figure 7.1. The difficulty in practice is to find a good representation for the functional relationships f_0 and f_1.

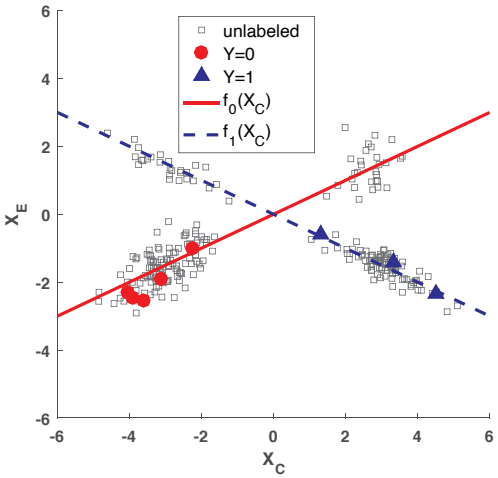


Figure 7.1: In this scenario we consider two points close to each other if both are close to the same regression model. If we would consider the joint space X_E, X_C we would consider the left two clusters close to each other. But with the depicted model the cluster on the bottom left is closer to the one on the top right than to the one on the top left. This is because in this model we consider points close to each other, if they are close to the same regression function. The figure is taken from [2].

7.2. IMPLICATIONS OF CHAPTER 3

First we recall that our review starts with a few impossibility results, and we discuss, in particular in Section 2.3, the hypothesis that a semi-supervised learner can improve the learning rate by at most a constant, unless we have some specific distributional assumptions. The first implication of Chapter 3 is that this hypothesis also holds in some settings for manifold regularization, which operates under, the arguably strong, assumption that the labeling function behaves smooth with respect to the data distribution. This was surprising to some degree, considering that in Section 2.6 we show that, under the seemingly similar cluster assumption, one can achieve exponential fast learning rates. If, for example, the data manifold consists out of two clusters, those two assumptions are just reformulations of each other.

One way to possibly explain why manifold regularization has only constant improvement, while the cluster assumption can lead to exponential learning rates, is that the constant can grow arbitrarily big as we show and discuss in Section 2.8.2. This means that at least in practice, and for small sample sizes, a constant improvement could be as much or more impactful than an exponential learning rate. To analyze this in more detail, one would need to study the precise relation of $\text{Pdim}(H)$ and $\text{Pdim}(\tilde{H}_\lambda)$, where H is an initial hypothesis space, and $\tilde{H}_\lambda$ is the resulting hypothesis space when we use hypotheses from H, and add a manifold regularization method with parameter λ. It would be in particular interesting to study the behavior of $\text{Pdim}(\tilde{H}_\lambda)$ wrt λ. With this we could for example find out how much we have to regularize to obtain a certain sample complexity improvement.

7.3. EXTENSIONS OF CHAPTER 3

In Chapter 3 we theoretically analyze manifold regularization via the pseudo-dimension and the Rademacher complexity. While the pseudo-dimension analysis gives a fairly complete picture of the worst case difference between semi-supervised and supervised learning, one can consider the Rademacher complexity analysis rather as a stepping stone for further theoretical investigations. We give instructions how to compute the Rademacher complexity, and this might already be useful in practice, but this does not answer the question how big the difference in Rademacher complexity compared to models without manifold regularization can be. Can they be essentially different, or could one also get negative results for that, and show that also this can only differ by a constant? To answer these questions one would have to find distributions on which one can analyze the actual Rademacher terms, at least to a degree that makes them comparable as functions in the sample size. From a practical point of view we would certainly expect that the Rademacher complexity terms can be essentially smaller on benign distributions, as manifold regularization is very effective whenever its assumption holds.

7.4. SEMI-SUPERVISED LEARNING AND CLASS PROBABILITY ESTIMATES

We shortly summarize our findings of Chapters 4 and 5 and then discuss their connections and possible extensions.

7.4.1. FINDING CLASS PROBABILITY ESTIMATES VIA CLASSIFICATION

In Chapter 5 we explore the possibility to retrieve a class probability estimate from methods that are designed for binary classification. Assume for example that a trained linear support vector machine results in a prediction function $f : \mathcal{X} \to \mathbb{R}$. Typically we then say that we predict that $x \in \mathcal{X}$ belongs to class $1 \in \mathcal{Y} = \{-1, 1\}$ iff $f(x) \geq 0$. But what if we also want to know a confidence of this prediction, as for example given by an estimate of the class probability $P(Y = 1 \mid X = x)$. In Chapter 5 we explored conditions under which such an estimate is consistently possible and how fast one can point-wise converge to the true class probability.

7.4.2. A SIMPLE IDEA

The motivation to investigate consistent class probability estimates stems from the algorithm proposed in Chapter 4. As a reminder, the essential idea of the method is to note that one can decompose the true risk of a classifier $f : \mathcal{X} \to \mathcal{Y} = \{-1, 1\}$ as

$$\mathbb{E}_{X,Y}\left[l(f(X), Y)\right] = \mathbb{E}_X\left[P(Y = -1 \mid X)l(f(X), -1) + P(Y = 1 \mid X)l(f(X), 1)\right], \tag{7.1}$$

where $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is a loss function. Note that the expectation on the right-hand side of the equation only depends on X, so can be estimated with the unlabeled data. The bottleneck is then the inner part, in particular we don't know the true class probabilities $P(Y \mid X)$. If we assume, however, that we can obtain decent estimates of the class probabilities we can try to use them in Equation (7.1) for a, hopefully, better approximation of the true risk. We can also think of this idea as extending the EM algorithm [3], which is as such only

defined for generative models, to discriminative models. In Chapter 4 we used a heuristic to define those class probabilities, but Chapter 5 offers a principled way to do so. In the next subsection we briefly summarize the heuristic we proposed in Chapter 4 and subsequently discuss what happens when one replaces the heuristic class probability estimate with the consistent one from Chapter 5.

7.4.3. AN IMPOSSIBILITY RESULT

We first summarize the algorithm of Chapter 4.

Algorithm 1

1. Gather some labeled and unlabeled data.
2. Train a discriminative model based on the labeled data.
3. Obtain estimates of $P(Y \mid X)$.
4. Estimate the right-hand side of Equation (7.1) with the labeled data, the unlabeled data and the class probability estimates.
5. Train a new model based on the previous estimate of the true risk, Equation (7.1).

In Chapter 4 we use a heuristic for step 3, but what if we use the consistent estimate of Chapter 5 there? One can show that in this case, most models will actually not change the solution found in step 2, i.e. the supervised solution. A proof of that is presented in Appendix A. This result is to some degree actually not surprising and ties in with the impossibility results we collected in the review, see Subsection 2.3. Those impossibility results sketch certain scenarios in which unlabeled data cannot help. Some of those results rely on the fact that discriminative models do not inherently carry any information about the marginal distribution, and collecting information about the marginal distribution in form of unlabeled data can thus not help to update the model. We observe the same behavior for our proposed Algorithm 1.

7.4.4. ADDING PRIOR KNOWLEDGE

The previous two subsections seem paradoxical. In Chapter 4 we showed that one can use Algorithm 1 to improve supervised classification, when using a certain heuristic to estimate class probabilities in step 3. On the other hand we also argued, that using in step 3 the consistent class probability estimates from Chapter 5 will leave the supervised solution unchanged. This discrepancy is explained by noting that the heuristic class probability estimate used in Chapter 4 has a hyperparameter which allowed us to push the class probability estimate to either 0 or 1 or to $\frac{1}{2}$. The result of our investigation from Chapter 4 is then straightforward: By pushing the estimate for example to 0 or 1, we effectively assume that the complete data distribution is well separated and thus we add prior knowledge to the method. Consequently, if the data is indeed well separated, pushing the estimate to 0 or 1 will improve the performance.

7.4.5. ADDING PRIOR KNOWLEDGE, BUT METHODICALLY

We believe that the previous findings can be combined into a well motivated new semi-supervised learning method. The idea is to use Algorithm 1 with a modified version of the consistent class probability estimates $\hat{\eta}(x)$. We modify them, such that all class probability estimates that are too close to $\frac{1}{2}$ are pushed towards a threshold which corresponds to our expected noise level. A simple example: Assume we have data with two classes, and we know that the noise level is bounded by $P(Y = 1 \mid X = x) \vee P(Y = -1 \mid X = x) \leq 0.2$. We then use Algorithm 1, but with the following modified version of $\hat{\eta}(x)$. Whenever $0.2 < \hat{\eta}(x) < 0.5$ we set $\hat{\eta}(x) = 0.2$ and whenever $0.5 < \hat{\eta}(x) < 0.8$ we set $\hat{\eta}(x) = 0.8$. We believe that using Algorithm 1 with those probability estimates one can show, under certain assumptions, that the class probability estimates are still consistent and we converge to a solution such that $\hat{\eta}(x) < 0.2 \vee \hat{\eta}(x) > 0.8$ for all $x \in \mathcal{X}$ in our training set. In other words, we believe that with this method one could encode an assumption similar to Tsybakov's low noise condition [4] through the unlabeled data, a condition that also found a connection to the work we presented in Section 2.4.1.

7.5. SAFE SEMI-SUPERVISED LEARNING

In Section 2.7.2 we discussed a line of research that tries to identify semi-supervised methods which can guarantee to be better than their supervised counterparts. While this always seemed to be an ambitious goal, it is more so in the light of the findings of Chapter 6. We showed that in a simple regression setting, adding labeled data can decrease the performance for finite sample sizes, even in expectation over the training samples. Considering this, it seems much harder to guarantee that adding *unlabeled* samples will increase the performance. We thus expect that any method that *can* guarantee improvements must be very conservative. Subsequently we expect that to achieve practically relevant improvements, one has to take the risk that comes with many semi-supervised methods.

7.6. EXTENSIONS OF CHAPTER 6

In [5] we extend the non-monotonicity results found in Chapter 6. Most of the non-monotonic behavior shown in Chapter 6 was a result of (exact) computations on a specific data-distribution. In [5] we introduce a technical lemma that let us also formally prove the observed behavior. This lemma specifies a sufficient condition on the loss function which will lead to non-monotonic behavior. Our main theorem then proceeds to show that this condition holds for the squared, absolute and the hinge loss. Furthermore this work also formally shows that for any given sample size n one can construct a distribution such that the risk increases when we use $n+1$ instead of n samples with an ERM algorithm.

But the technical lemma itself is also of interest. As we elaborate in the discussion of [5], the lemma seems to indicate that the learning rate of an algorithm would have to be linear or faster to avoid non-monotonic behavior in our setting. As the lemma is actually independent of the specific learning algorithm chosen and we know that in many settings, as for example agnostic learning, there are no learners that learn with a linear rate, one could imagine to extend the result of Example III from Chapter 6 to any learning algorithm.

7.7. CURRENT TRENDS IN SEMI-SUPERVISED LEARNING

This thesis is fairly independent of current trends in semi-supervised learning. As already mentioned in the introduction, the currently successful deep learning models need a large amount of labeled data, and therefore it is natural to try to mitigate this by replacing part of the labeled data with unlabeled data. The two paradigms that have been adopted for deep learning models are entropy and consistency regularization. The main idea of entropy regularization [6] is that we try to enforce low entropy predictions on the unlabeled data, which means that our decision boundary should be in a low density region. In the deep learning community this idea became known under the term *pseudo-labeling* [7] and is effectively a reinvention of self-learning.

Consistency regularization [8–10] has the underlying idea, that if we transform an unlabeled data point u in a meaningful way into $\hat{u}$, then the predictions $f(u)$ and $f(\hat{u})$, if f is a classifier, should be similar. The idea is thus to add a regularizer of the form $d(f(u), f(\hat{u}))$ to the loss term, where d is some sort of distance function. Regarding pseudo-labeling, that there are few theoretical analyses, and the situation is not improved by the fact that the method is embedded in a deep model. As for deep learning models themself, there are only a few recent advances towards a theoretical understanding [11, 12].

Consistency regularization on the other hand is related to manifold regularization. The difference is that in manifold regularization, as defined and analyzed in Chapter 3, the data manifold is solely defined by the unlabeled data and the distance measure between them. Consistency regularization adds prior knowledge to that, by additionally altering the unlabeled data in meaningful ways[1], and we thus create our own data manifold. With this reasoning we believe that the analysis of Chapter 3 can be extended to consistency regularization. The bottleneck, however, could still be that the underlying models are deep learning models and for those we cannot draw from a rich literature of theoretical results, as opposed to the kernel models which we used in Chapter 3.

7.8. FINAL REMARKS

The aim of this thesis has been to investigate the theoretical foundations of semi-supervised learning. We started this with an extensive review of existing results, and added our own complexity analysis of manifold regularization. We then investigated the possibility to obtain class probability estimates with classification methods, and discussed how this investigation can lead in future work to a new well-motivated semi-supervised learner. We hope that anyone starting in this field can find inspiration for their own work from this thesis.

[1] If we for example have images of a digit, we know that a certain amount of rotation will still result in the same digit.

REFERENCES

[1] J. von Kügelgen, A. Mey, and M. Loog, *Semi-generative modelling: Covariate-shift adaptation with cause and effect features,* in *The 22nd International Conference on Artificial Intelligence and Statistics* (Okinawa, Japan, 2019) pp. 1361–1369.

[2] J. von Kügelgen, M. Loog, A. Mey, and B. Schölkopf, *Semi-supervised learning, causality and the conditional cluster assumption,* Computing Research Repository **abs/1905.12081** (2019).

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood from incomplete data via the em algorithm,* Journal of the Royal Statistical Society, Series B **39**, 1 (1977).

[4] A. B. Tsybakov, *Optimal aggregation of classifiers in statistical learning,* The Annals of Statistics **32**, 135 (2004).

[5] M. Loog, T. Viering, and A. Mey, *Minimizers of the empirical risk and risk monotonicity,* in *Advances in Neural Information Processing Systems 32* (Vancouver, Cananda, 2019) pp. 7476–7485.

[6] Y. Grandvalet and Y. Bengio, *Semi-supervised learning by entropy minimization,* in *Advances in Neural Information Processing Systems 17* (Vancouver, British Columbia, Canada, 2004) pp. 529–536.

[7] D.-H. Lee, *Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks,* ICML Workshop : Challenges in Representation Learning (2013).

[8] P. Bachman, O. Alsharif, and D. Precup, *Learning with pseudo-ensembles,* in *Advances in Neural Information Processing Systems 27* (Montreal, Canada, 2014) pp. 3365–3373.

[9] M. Sajjadi, M. Javanmardi, and T. Tasdizen, *Regularization with stochastic transformations and perturbations for deep semi-supervised learning,* in *Advances in Neural Information Processing Systems 30* (Barcelona, Spain, 2016) pp. 1171–1179.

[10] S. Laine and T. Aila, *Temporal ensembling for semi-supervised learning,* in *5th International Conference on Learning Representations* (Toulon, France, 2017).

[11] G. K. Dziugaite and D. M. Roy, *Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data,* in *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence* (Sydney, Australia, 2017).

[12] M. Belkin, S. Ma, and S. Mandal, *To understand deep learning we need to understand kernel learning,* in *Proceedings of the 35th International Conference on Machine Learning* (Stockholm Sweden, 2018) pp. 541–549.

ACKNOWLEDGEMENTS

There is a number of people I would like to thank for making my PhD journey the best possible it could be. I almost cannot imagine any better place to have done my PhD. All the social events, retreats, borrels or just the chats over a coffee made it such a great time. But more than that, all of you made Delft such an enjoyable work place, that I decided to stay for another two years.

All the people I am grateful for; Alexey, Jaoana and Amim, thanks for showing great hospitality in my first weeks, while I had no own office space yet. Christine and Tom, thanks for being a staple during our movie nights, you endured the good and the bad times. Stavros, Arlin, Tamim, thanks for explaining me at every poster session what a chromosome is, next time I will remember, promise! Ramin, thanks for introducing me to some of the great food from your country. Thanks Christian for being the most German on the floor, especially your particular sense of humor! Thies, thanks for being such a great story creator. Ahmed and Amogh, my favorite sometimes office mates and Soufiane my favorite french. Thomas, thanks for keeping that wild crew in order!

Thanks Jan for leading such a wonderful troop of people, and giving me in many topics an interesting and different point of view. Talking about your people, Yunqiang, Yancong and Osman, talking to you always gave me a good mood, keep your positive attitude! Talking about attitude, Silvia's is truly inspiring, you are an asset for the whole group, thanks for that, peace! Ziqi, make sure that Tom does not derail too much from his plans, thanks in advance for that! Seyran, thanks for being such a nice example of how to balance family and work, I am sure that will come in handy at some point. Nergis and Jin, although we just met, I already know that you will be a great addition to the group! Hamdi and Yanxia, thanks for shaping my journey and being such nice colleagues.

Many thanks to Hayley, you and your group gave so many times the initial spark for marvelous discussions on all the interesting topics surrounding AI, I truly enjoyed those. Yeshwant, Stephanie, Bernd, Jose and Chirag, thanks for further feeding this spark, I hope you keep your enthusiasm for investigating the role of humans in AI.

David, you are an amazing pattern recognizer, thanks for bringing so many insights to the most diverse discussions. The same holds for Taygun, but I guess the apple does not fall far from the tree. Bob, as the group wisest I consider myself lucky that I had the opportunity to listen to many of your talks. The many years you worked in that field truly resulted in some deep insights, thanks for that. Yazhou and Wenjie, I hope you are doing great in China, thanks for being amazing colleagues.

Also thanks to Saskia, Bart and Ruud, your efforts keep the machinery of our group running, many thanks for that!

Thanks to Peter for inviting me to CWI for a few month, and introducing me to a different side of machine learning and many thanks to you and also Balazs for the feedback on this thesis. Also thanks to Muriel, it was really nice to work with you, and I hope we can still finish our project!

Now to some special thanks. First there is to mention my promoter Marcel and supervisor Marco. I think you made a great team in coaching. Marcel, thank you especially for your different and realistic point of view on many topics, you helped me a lot to set, follow and keep track of my goals and ultimately helped me to manage my planning in finishing this thesis. Thanks! Marco, you made most of this possible of course. I could not have wished for a better supervisor. You left me all the room I wanted to explore new directions, while being a great role model for ethics, attitude and honest scientific curiosity. While I hope that this thesis will also help the field to move forward, the greatest achievement of it is in my opinion that I became an independent researcher through your guidance. Thanks for that!

Thanks and credits for shaping me as a scientist goes as well to Jesse. That we shared common research interests was not the only reason that we had many fruitful discussions. Ultimately my thesis would look different without your influence, thanks for opening my eyes to many new directions along the way.

Some of my fondest memories include Laura, Ekin, Sally, Wouter and Tom, thanks for all the great moments! Ekin, you initiated many interesting discussions on ethics and related topics, and I hope we can keep our post coffeetalk coffee as a habit. Laura, our good soul, you always made sure that everybody arrived home after a night out, well done! Sally, your Mexican BBQ is amazing, thanks for all the good laughs and memories. Wouter, your amazing talent to push any conversation into any direction made for so many interesting and sometimes weird conversations, thanks for all the good times. And then there is of course Tom, my office mate. Next to Marco you helped shaping my research process the most, you often have a different angle of looking at things and that really helped in many situations. But the most important, you are just a fun guy to be around, thanks for many great moments!

Franka, I really wonder what I would be doing now if we would have never met, but I would for sure have missed out on this amazing PhD journey. You made me consider doing a PhD in the first place, and I think there is a good chance I would not have moved to the Netherlands if we would not have met. You are my daily support and you helped me keep going in times where I thought I was stuck. Thank you for always listening, always supporting and always being there for me! There is nothing better than coming back to our nice little flat in Gouda after a day of work.

Zum Schluss geht natürlich noch ein riesen Dank an meine Freunde und Familie, insbesondere meine Eltern und Ferdie und Elke. Ihr habt mich bei allen Unternehmungen ausnahmslos unterstützt, egal wie abwegig sie anderen erscheinen könnten. Sei es nun das Vorhaben Mathematik zu studieren, für eine Zeit in England oder Brasilien zu leben, eine Promotion in den Niederlanden anzustreben oder der Wunsch zum 30. Geburtstag einen Lego Todesstern zu bekommen. Diese Unterstützung kam von euch so selbstverständlich, dass ich mich ofter daran erinnern muss, dass sowas nicht selbstverständlich ist, und meines Erachtens das größte Geschenk ist, dass ein Kind von seinen Eltern bekommen kann. Danke auch an meinen Bruder, der als großer Bruder immer ein Vorbild war und noch immer ist. Wie ihr beiden Beruf und Familie meistert ist wortwörtlich Vorbildhaft, und es gibt keine bessere Ablenkung vom Alltag als eure beiden kleinen zu besuchen. Willi, Erich und Daniel, neben meiner Familie und Franka seid ihr die wichtigsten Konstanten in meinem Leben. Ich weiß, dass ich immer und jederzeit auf euch zählen kann. Danke euch allen!

A

APPENDIX

In the following we want to investigate the impossibility of semi-supervised learner from another view. We show that it is impossible to use Expectation-Minimization (EM) as a tool to integrate unlabeled data in a discriminative model. With impossible we mean here, that the found solution will be the same as the supervised solution. As this is a result of our analysis from Chapter 5 we adopt the same notation.

A.1. EM WITH GENERATIVE MODELS

To define an EM approach with discriminative models, we first present the rough idea of SSL with the EM algorithm for generative models. We start with a probability model $p(x, y \mid f)$ parametrized in some $f \in \mathcal{F}$. In the supervised case this model is typically fitted to the observed samples $(X_n, Y_n) = \{x_i, y_i\}_{1 \le i \le n}$ with a maximum (log-)likelihood method, i.e.

$$f_{\text{sup}} = \arg\max_{f \in \mathcal{F}} \ln p(X_l, Y_l \mid f).$$

With some additional unlabeled data $U_m = \{u_1, ..., u_m\} \in \mathcal{X}^m$ we can get an improved maximum likelihood estimate of the complete model. With Z_m we denote the random vector of unknown labels of U_m.

$$f_{\text{semi}} = \arg\max_{f \in \mathcal{F}} \ln p(X_n, Y_n, U_m \mid f)$$

Using the independence assumption of the sampling process we can rewrite this as:

$$\ln p(X_n, Y_n, U_m \mid f) = \ln p(X_n, Y_n \mid f) + \ln p(U_m \mid f) \tag{A.1}$$

This in turn can be rewritten as:

$$\ln p(X_n, Y_n, U_m \mid f) = \ln p(X_n, Y_n \mid f) + \ln p(U_m \mid f) \tag{A.2}$$

$$= \ln p(X_n, Y_n \mid f) + \int_{Z_m \in \mathcal{Y}^m} \ln p(U_m, Z_m \mid f) \, dZ_m \tag{A.3}$$

A

The integral over all possible labelings of the unlabeled data on the right hand side of Equation (A.2) can be complicated to calculate. The EM algorithm avoids the exact calculation by doing it only in expectation over the labels, with an estimated label distribution from the model of the last iteration. We then find a new model by maximizing this expectation in the model parameters. More formally EM iterates the following two steps.

1. Compute $G(f_i, f) = \mathbb{E}_{Z_m \sim p(Z_m|U_m, f_i)}[\ln p(X_n, Y_n, U_m, Z_m \mid f)]$

2. Set $f_{i+1} = \arg\max_{f \in \mathcal{F}} G(f_i, f)$

The EM algorithm can be shown to find a local maximum of the complete log-likelihood (A.2). In Section A.2 we derive a formulation of this algorithm for discriminative models and show that this approach will leave the supervised solution unchanged in many situations.

A.2. EM WITH DISCRIMINATIVE MODELS

We recreate the EM algorithm in the discriminative setting as follows. The log-likelihood used in Equation (A.2) can be viewed as a negative loss function. Generalizing the negative log-likelihood to an arbitrary loss function $l(f(x), y)$ the EM algorithm becomes:

1. Compute $J(f_i, f) = \mathbb{E}_{Z \sim p(Y_u|X_u, f_i)}[\sum_{i=1}^{n} l(f(x_i), y_i) + \sum_{i=1}^{m} l(f(u_i), z_i]$
2. Set $f_{i+1} = \arg\min_{f \in \mathcal{F}} J(f_i, f)$

Note that by switching from a likelihood formulation in the generative case to a loss function formulation in the discriminative case, we also switch from an expectation maximization to an expectation minimization.

The main problem to address in this formulation are the posteriors $p(Z_m \mid U_m, f)$ that are used to update the current model. While in generative models posteriors are defined through the joint probability distribution, discriminative models do not generally define a posterior probability directly. From now on we assume that our classifiers $f : \mathcal{X} \to \mathbb{R}$ map the input to the real numbers. From Chapter 5 we know that the to the loss associated function $v^{*-1} : \mathbb{R} \to [0, 1]$ (5.4), is the only one that makes (l, v^{*-1}) a natural class probability estimation loss [1], and thus leads to consistent posterior estimates[2]. So, in particular, we will set in the later stage $p(Y = 1 \mid X = x, f) := v^{*-1}(f(x))$.

A.3. EM FAILS WITH DISCRIMINATIVE MODELS

In this section we analyze the resulting algorithm. To do so we define the type of solution that EM will find.

Definition 10. *We call a hypothesis $f_0 \in \mathcal{F}$ faithful w.r.t to the data $(X_n, Y_n), U_m$ if the following inequality holds for all $f \in \mathcal{F}$.*

$$\mathbb{E}_{Z_m \sim p(Z_m|U_m, f_0)}[\hat{Q}(f_0)] \leq \mathbb{E}_{Z_m \sim p(Z_m|U_m, f_0)}[\hat{Q}(f)]$$

[1] See Corollary 5
[2] See Theorem 23

The interpretation of faithful solutions is the following. A solution is not faithful if even under that assumption that it is true (as we use the class probability estimates from the solution), the seen data disagrees with it so much that we prefer to change the solution. It is easy to see that the EM algorithm makes sure that we end up with a faithful solution and stops as soon as it found one. The next theorem reveals the problem of the procedure by showing that the purely supervised solution is already a faithful solution.

Theorem 25. *Assume v^{*-1} is defined on the whole of $\mathbb{R}$ (possibly a one to many mapping). Then the supervised solution f_{sup} (3.1) is faithful.*

Proof. First set $f_1 = f_{\text{sup}}$. The posterior estimate of the model f_1 is given by

$$p(z \mid u, f_1) = v^{*-1}(f_1(u)). \tag{A.4}$$

To show that f_1 is faithful we need to show that it minimizes

$$\mathbb{E}_{Z_m \sim p(Z_m \mid U_m, f_1)}[\hat{Q}(f, U_m \cup X_n, Z_m \cup Y_n)]. \tag{A.5}$$

Since f_1 is by definition minimizing the risk of the labeled samples in $\mathcal{F}$ it is enough to show that f_1 also minimizes the sum coming from the unlabeled samples X_u. We will do that by showing that f_1 minimizes each term of the sum individually. So for each unlabeled point $u \in U_m$ we look at the value in $\mathbb{R}$ that minimizes the term in the sum, and show that f_1 is already a valid solution for that. Given the posterior estimates (A.4), the value that minimizes the expected loss on u is given by

$$f_2(u) = \arg\min_{v \in \mathbb{R}} \; p(Y = 1 \mid X = u, f_1) l(f(u), 1) + (1 - p(Y = -1 \mid X = u, f_1)) l(f(u), -1).$$

With the definition of v^* we can rewrite this as

$$f_2(u) = v^*(p(Y = 1 \mid X = u, f_1)) = v^*(v^{*-1}(f_1(u))).$$

Note that either v^{*-1} or v^* might be set function, but we know that $f_1(u) \in v^*(v^{*-1}(f_1(u)))$. So f_1 is a minimizer for each u. With the previous argumentation this suffices to show that $f_1 = f_{\text{sup}}$ is a minimizer of (A.5) and thus a faithful solution. □

One assumption of the theorem is that the domain of v^{*-1} is $\mathbb{R}$. This holds for a lot of typical loss functions like hinge loss and logistic loss. In the case of logistic loss v^{*-1} coincides with the posterior probability defined in logistic regression $\frac{1}{1+e^{-x}}$.

Curriculum Vitæ

Alexander Mey

15-08-1988 Born in Mechernich, Germany.

Education

1999–2008	High School
	Gymnasium am Turmhof, Mechernich
2008–2011	Bachelor in Mathematics
	Rheinische Friedrich-Wilhelms-Universität Bonn
2011–2012	Erasmus Exchange Year and Additional Studies in Mathematics
	University of Warwick
2012–2014	Master in Mathematics
	Rheinische Friedrich-Wilhelms-Universität Bonn
2015–2019	PhD. Computer Science
	Delft University of Technology
2019–2021	Postdoctoral Research
	Delft University of Technology
	PhD thesis: Assumptions and Expectations in Semi-Supervised Machine Learning

List of Publications

8. M. Loog, T. Viering, A. Mey *Minimizers of the Empirical Risk and Risk Monotonicity*, Advances in Neural Information Processing Systems 32, pages 7476-7485, Vancouver, Canada, 2019.

7. T. Viering, A. Mey, M. Loog, *Open Problem: Monotonicity of Learning*, Conference on Learning Theory, pages 3198-3201, Phoenix, Arizona, 2019.

6. A. Mey, T. Viering, M. Loog, *A Distribution Dependent and Independent Complexity Analysis of Manifold Regularization*, ArXiv: 1906.06100, 2019.

5. J. von Kügelgen, A. Mey, M. Loog, B. Schölkopf, *Semi-Supervised Learning, Causality and the Conditional Cluster Assumption*, ArXiv: 1905.12081, 2019.

4. A. Mey, M. Loog, *Consistency and Finite Sample Behavior of Binary Class Probability Estimation*, ArXiv: 1908.11823, 2019.

3. A. Mey, M. Loog, *Improvability Through Semi-Supervised Learning: A Survey of Theoretical Results*, ArXiv: 1908.09574, 2019.

2. J. von Kügelgen, A. Mey, M. Loog, *Semi-Generative Modelling: Covariate-Shift Adaptation with Cause and Effect Features*, The 22nd International Conference on Artificial Intelligence and Statistics, pages 1361-1369, Okinawa, Japan, 2019.

1. A. Mey, M. Loog, *A soft-labeled self-training approach*, 23rd International Conference on Pattern Recognition, pages 2604-2609, Cancun, Mexico, 2016.