UNIVERSIDAD AUTÓNOMA DE MADRID

Advanced Kernel Methods for Multi-Task Learning

by

Carlos Ruiz Pastor

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Escuela Politécnica Superior

Computer Science Department

under the supervision of José R. Dorronsoro Ibero

August 2021

What is the essence of life? To serve others and to do good.

Aristotle.

Abstract

.

Resumen

.

Acknowledgements

.

Contents

Abbreviations

ADF	**A**ssumed **D**ensity **F**iltering
AF	**A**cquisition **F**unction
BO	**B**ayesian **O**ptimization
DGP	**D**eep **G**aussian **P**rocess
EI	**E**xpected **I**mprovement
EP	**E**xpectation **P**ropagation
GP	**G**aussian **P**rocess
KL	**K**ullback **L**iebler
MCMC	**M**arkov **C**hain **M**onte **C**arlo
PPESMOC	**P**arallel **P**redictive **E**ntropy **S**earch for **M**ultiobjective **O**ptimization with **C**onstraints
PES	**P**redictive **E**ntropy **S**earch
PESMOC	**P**redictive **E**ntropy **S**earch for **M**ultiobjective **O**ptimization with **C**onstraints
RS	**R**andom **S**earch
UCB	**U**pper **C**onfidence **B**ound

To my family

Chapter 1

Introduction

We begin this manuscript...

1.1 Introduction

1.2 Publications

This section presents, in chronological order, the work published during the doctoral period in which this thesis was written. We also include other research work related to this thesis, but not directly included on it. Finally, this document includes content that has not been published yet and is under revision.

Related Work

Work In Progress

1.3 Summary by Chapters

In this section...

Chapter 3 provides an introduction to GPs and the expectation propagation algorithm. Both are necessary concepts for the BO methods that we will describe in the following chapters. This chapter reviews the fundamentals of GPs and why they are so interesting for BO. More concretely, we review the most popular kernels, the analysis of the posterior and predictive distribution and how to tune the hyper-parameters of GPs: whether by maximizing the marginal likelihood or by generating samples from the hyper-parameter posterior distribution. Other alternative probabilistic surrogate models are also described briefly. Some of the proposed approaches of this thesis are extensions of an acquisition function called predictive entropy search, that is based on the expectation propagation approximate inference technique. That is why we provide in this chapter an explanation of the expectation propagation algorithm.

Chapter 5 introduces the basics of BO and information theory. BO works with probabilistic models such as GPs and with acquisition functions such as predictive entropy search, that uses information theory. Having studied GPs in Chapter 3, BO can be now understood and it is described in detail. This chapter will also

describe the most popular acquisition functions, how information theory can be applied in BO and why BO is useful for the hyper-parameter tuning of machine learning algorithms.

Chapter ?? describes an information-theoretical mechanism that generalizes BO to simultaneously optimize multiple objectives under the presence of several constraints. This algorithm is called predictive entropy search for multi-objective BO with constraints (PESMOC) and it is an extension of the predictive entropy search acquisition function that is described in Chapter 5. The chapter compares the empirical performance of PESMOC with respect to a state-of-the-art approach to constrained multi-objective optimization based on the expected improvement acquisition function. It is also compared with a random search through a set of synthetic, benchmark and real experiments.

Chapter ?? addresses the problem that faces BO when not only one but multiple input points can be evaluated in parallel that has been described in Section **??**. This chapter introduces an extension of PESMOC called parallel PESMOC (PPESMOC) that adapts to the parallel scenario. PPESMOC builds an acquisition function that assigns a value for each batch of points of the input space. The maximum of this acquisition function corresponds to the set of points that maximizes the expected reduction in the entropy of the Pareto set in each evaluation. Naive adaptations of PESMOC and the method based on expected improvement for the parallel scenario are used as a baseline to compare their performance with PPESMOC. Synthetic, benchmark and real experiments show how PPESMOC obtains an advantage in most of the considered scenarios. All the mentioned approaches are described in detail in this chapter.

Chapter ?? addresses a transformation that enables standard GPs to deliver better results in problems that contain integer-valued and categorical variables. We can apply BO to problems where we need to optimize functions that contain integer-valued and categorical variables with more guarantees of obtaining a solution with low regret. A critical advantage of this transformation, with respect to other approaches, is that it is compatible with any acquisition function. This transformation makes the uncertainty given by the GPs in certain areas of the space flat. As a consequence, the acquisition function can also be flat in these zones. This phenomenom raises an issue with the optimization of the acquisition function, that must consider the flatness of these areas. We use a one exchange neighbourhood approach to optimize the resultant acquisition function. We test our approach in synthetic and real problems, where we add empirical evidence of the performance of our proposed transformation.

Chapter ?? shows a real problem where BO has been applied with success. In this problem, BO has been used to obtain the optimal parameters of a hybrid Grouping Genetic Algorithm for attribute selection. This genetic algorithm is combined with an Extreme Learning Machine (GGA-ELM) approach for prediction of ocean wave features. Concretely, the significant wave height and the wave energy flux at a goal marine structure facility on the Western Coast of the USA is predicted. This chapter illustrates the experiments where it is shown that BO improves the performance of the GGA-ELM approach. Most importantly, it also outperforms a random search of the hyper-parameter space and the human expert criterion.

Chapter ?? provides a summary of the work done in this thesis. We include the conclusions retrieved by the multiple research lines covered in the chapters. We also illustrate lines for future research.

1.4 Definitions and Notation

Chapter 2

Foundations and Concepts

This chapter presents...

2.1 Introduction

2.2 Kernels

2.2.1 Motivation and Definition

2.2.2 Reproducing Kernel Hilbert Spaces

2.2.3 Examples and Properties

2.3 Risk Functions and Regularization

2.3.1 Empirical and Expected Risk

2.3.2 Regularized Risk Functional

2.3.3 Representer Theorem

2.4 Optimization

2.4.1 Convex Optimization

2.4.2 Unconstrained Problems

2.4.3 Constrained Problems

2.5 Statistical Learning

2.5.1 Uniform Convergence and Consistency

2.5.2 VC dimension and Structural Learning

2.6 Support Vector Machines

2.6.1 Linearly Separable Case

2.6.2 Non-Linearly Separable Case

2.6.3 Kernel Extension

2.6.4 SVM properties

2.6.5 Connection with Structural Learning

2.6.6 SVM Variants

2.7 Conclusions

In this chapter, we covered...

Chapter 3

Multi-Task Learning

This chapter presents...

3.1 Introduction

3.2 Why does Multi-Task Learning work?

3.2.1 Inductive Bias Learning Problem

Tipically in Machine Learning the goal is to find the best hypothesis $h(x, \alpha_0)$ from a space of hypotheses $\mathcal{H} = \{h(x, \alpha), \alpha \in \Lambda\}$, where Λ is any set of parameters. This best candidate can be selected according to different inductive principles, which define a method of approximating a global function $f(x)$ from a training set: $z := \{(x_i, y_i),\ i = 1, \ldots, n\}$ where (x_i, y_i) are sampled from a distribution F. In the classical statistics we find the Maximum Likelihood approach, where the goal is to estimate the density $f(x) = p(y \mid x)$ and the hypotheses space is parametric, i.e. $\mathcal{H} = \{h(x, \alpha), \alpha \in \Lambda \subset \mathbb{R}^m\}$. The learner select the parameter α that maximizes the probability of the data given the hypothesis. Another more direct inductive principle is Empirical Risk Minimization (ERM), which is the most common one. In ERM the densities are ignored and an empirical error $\hat{e}_z$ is minimized with the hope of minimizing the true expected error e_F, which would result in a good generalization. Inductive Bias learning, also known as "Learning to learn" has the goal of finding a good hypotheses space from which statistical learning methods can be applied to

In Baxter (2000) an effort is made to define the concepts needed to construct the theory about inductive bias learning, which can be seen as a generalization of multi-task learning. This is done by extending the work of hypotheses space capacity Vapnik (2013) to a family of spaces of hypotheses capacity, with the goal of learning a good space of hypotheses from which we can obtain a good hypothesis.

Statistical Learning

In the classical learning approach, one has the following components:

- an input space X and an output space Y,
- a probability P (which is unknown) defined over $X \times Y$,

- a loss function $l : Y \times Y \to R$, and
- a hypothesis space $\mathcal{H}$ with hypotheses or functions $h : X \to Y$.

The goal for the learner is to select a hypothesis $h \in \mathcal{H}$ that minimizes the expected loss:

$$e_P(h) = \int_{X \times Y} l(h(x), y) dP(x, y).$$

The density P is unknown, but we have a training set $z = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ drawn from P. One alternative, known as Empirical Risk Minimization (ERM), is to minimize:

$$\hat{e}_z(h) = \frac{1}{m} \sum_{i=1}^{m} l(h(x_i), y_i).$$

Although this is the more straight-forward way to minimize $e_P(h)$ ($\hat{e}_z(h)$ is an unbiased estimator of $e_P(h)$), it has been shown Vapnik (2013) that there are smarter ways (with better generalization properties) of minimizing the expected loss. This have relation with two facts: the first one is that the unbiased property is an asymptotical one, the second one has to do with overfitting. Vapnik answers to the question of what can we say about $e_P(h)$ when h minimizes $\hat{e}_z(h)$, and moreover, his results are valid also for small number of training patterns m. Under some general conditions, he proves that:

$$e_P(h) \leq \hat{e}_z(h) + R(d/m) \tag{3.1}$$

where R is some non-decreasing function and d is the VC-dimension of the hypothesis h. This means that the empirical loss $\hat{e}_z(h)$ gets closer to the expected loss $e_P(h)$ if:

- We use a larger number m of training samples (this was already inferred from the asymptotical properties)
- The VC-dimension (or capacity) of the hypotheses we use is small. This is the most important term in Vapnik theory.

The VC-dimension measures the capacity of a function or hypotheses space. In the case of a set of indicator functions, it is the maximum number of vectors $x_1, \ldots, x_d$ that can be shattered (in two classes) by functions of this set. In the case of real functions, it is defined as the VC-dimension of the following set of indicator functions $I(x, h, \beta) = \mathbf{1}_{\{h(x)-\beta\}}$. The VC-dimension gives a notion of the capacity of our set of hypotheses. If the the capacity of our set of hypotheses $\mathcal{H}$ is too large, we may find an hypotheses h^* that minimizes $\hat{e}_z(h)$ but does not generalize well and therefore, does not minimize $e_P(h)$. We also have to notice that using a very simple (low capacity) space of hypotheses $\mathcal{H}$ we could be in a situation where there is not a good hypothesis $h \in \mathcal{H}$, so we cannot minimize $\hat{e}_z(h)$.

The Structural Risk Minimization as an inductive principle proposed by Vapnik (as opposed to the ERM) tries to find a tradeoff between minimizing $\hat{e}_z(h)$ and minimizing d. The idea is to define an admissible structure, that is a sequence of hypotheses spaces:

$$\mathcal{H}_1 \subset \mathcal{H}_2 \subset \ldots \subset \mathcal{H}_k \subset \ldots$$

where their VC-dimensions are ordered:

$$d_1 < d_2 < \ldots < d_k < \ldots$$

where d_i is the VC-dimension of $\mathcal{H}_i$ SRM selects the hypothesis $h^* = h_i^* \in \mathcal{H}_i$ that obtains the best bound for the actual risk $e_P(h)$. This admissible structure can be built in various ways. In Neural Networks in can be the built by increasing the number of hidden layers. In other methods, such as SVM or Ridge Regression, this is done by decreasing the regularization. However, this SRM principle is usually replaced by a cross-validation (CV) procedure.

Support Vector Machines, which are the most representative models of this theory, use the VC-dimension also in other way (apart from the SRM principle or the CV procedure). The goal of finding the optimal hyperplane, that is, that with the maximum margin between the classes, has its motivation in the fact the set of such type of hypotheses have a lower VC-dimension that the set of all hyperplanes do.

Bias Statistical Learning

According to Baxter, in Bias Learning we have the following components:

- an input space X and an output space Y,
- a family of probabilities $\mathcal{P}$ of probabilities P defined over $X \times Y$, and a distribution Q defined over $\mathcal{P}$,
- a loss function $l : Y \times Y \to R$, and
- a family of hypotheses spaces $\mathbb{H}$ of spaces $\mathcal{H}$ with hypotheses or functions $h : X \to Y$.

The goal here is also to minimize the expected risk, which in this case is defined as:

$$e_Q(\mathcal{H}) = \int_{\mathcal{P}} \inf_{h \in \mathcal{H}} e_P(h) dQ(P) = \int_{\mathcal{P}} \inf_{h \in \mathcal{H}} \int_{X \times Y} l(h(x), y) dP(x, y) dQ(P). \tag{3.2}$$

Again, we do not know $\mathcal{P}$ nor Q, but we have a training set obtained in the following way:

1. Sample T times from Q obtaining $P_1, \ldots, P_T \in \mathcal{P}$
2. For $r = 1, \ldots, T$ sample using P_r m pairs $z_r = \{(x_1^r, y_1^r), \ldots, (x_m^r, y_m^r)\}$ where $(x_i^r, y_i^r) \in X \times Y$.

We obtain $z = \{(x_i^r, y_i^r), r = 1,\ i = 1, \ldots,\ m = 1, \ldots, T\}$. We call z a (T, m)-sample, with m examples from T different learning tasks (the constant size m is for simplicity purposes). Now we can define an empirical loss of z as:

$$\hat{e}_z(\mathcal{H}) = \sum_{r=1}^{T} \inf_{h \in \mathcal{H}} \hat{e}_{z_r}(h) = \sum_{r=1}^{T} \inf_{h \in \mathcal{H}} \sum_{i=1}^{m} l(h(x_i^r), y_i^r), \tag{3.3}$$

which is simply an average of the empirical losses of each task. Note, however, that in this case this estimate is biased, since (as we have seen) $e_{P_r}(h)$ does not coincide with $\hat{e}_{z_r}(h)$. As in the classical approach, we want more intelligent ways of minimizing the actual error $e_Q(\mathcal{H})$. By following the same path, we want to:

- Define a notion of capacity of hypotheses space families (VC-dimension like)
- Find a bound of $e_Q(\mathcal{H})$ using the empirical error $\hat{e}_z(\mathcal{H})$ and the capacity of the hypotheses space family.

- Define an inductive principle to learn good hypotheses spaces $\mathcal{H}$?
- Find (bias learning) models that uses low capacity hypotheses space families?

Baxter does not directly address these points. In first place, he defines some concepts that allow to write two concepts of capacity related to $\mathbb{H}$. Using these two concepts, he shows uniform convergence of $e_Q(\mathcal{H})$ to $\hat{e}_z(h)$. Moreover, this result is valid not only asymptotically:

$$e_Q(\mathcal{H}) \leq \hat{e}_z(\mathcal{H}) + \epsilon$$

with m such that

$$m \geq \frac{1}{T\epsilon^2} C(\mathbb{H})$$

where $C(\mathbb{H})$ is a notion of capacity of $\mathbb{H}$ (this bound is very simplified in this work). We can observe that as the number of tasks grow, we need less examples in each task. Also, if we have related tasks (and then $\mathbb{H}$ can have low capacity) the number of necessary examples per task also decreases. However, this bound is not so useful because it is theoretical, that is, we have theoretical notions of the capacity of $\mathbb{H}$ but we cannot compute it.

He then focuses on feature learning, that is, we define a specific type of learners where we have two steps:

- Learning the space of features (or learning $\mathcal{H}_f$)
- Learning the best linear model g over the features given by f (or learning the best hypothesis $g \circ f$ in $\mathcal{H}_f$)

For these type of models, he also shows how a multi-output Neural Network (which can be seen an MTL NN) is a model that embodies this feature learning approach. The NN can be seen as a linear model g over a transformation f of the original data. That is, the set of functions $\mathcal{H}_f = \{g \circ f\}$ used by the neural network is learned trying to minimize the empirical loss when we update f. Baxter derives specific bound for the NN using that the capacity of the hypotheses space family $\mathbb{H}$ can be computed for this case.

Finally, he show some results for Multi-Task Learning. Multi-Task Learning is a particular case of Bias Learning where we are not interested in finding an optimal hypotheses space $\mathcal{H}$ where we can find good solutions for a great variety of problems, but our goal is to find good solutions for a fixed number of tasks. That is, instead of having a family of probabilities $\mathcal{P}$ with a probability measure $\mathcal{Q}$, we have a fixed vector $\boldsymbol{P} = (P_1, \ldots, P_T)$ of probability measures over a fixed set of tasks and the learner minimizes:

$$e_{\boldsymbol{P}}(\mathcal{H}) = \sum_{r=1}^{T} e_{P_r}(h_r) = \sum_{r=1}^{T} \int_{X \times Y} l(h_r(x), y) dP_r(x, y). \tag{3.4}$$

Then, we define the empirical multi-task error as

$$\hat{e}_z(\mathcal{H}) = \sum_{r=1}^{T} \hat{e}_{z_r}(h_r) = \sum_{r=1}^{T} \sum_{i=1}^{m} l(h_r(x_i^r), y_i^r). \tag{3.5}$$

The difference is subtle but we have a different objective. In the multi-task setting, the hypotheses space $\mathcal{H}$ found will not necessarily be good for generalizing to new tasks. Also, we have got rid of the inf term, which was difficult to deal with.

Restricting to the Boolean functions, Baxter defines an extension of the VC-dimension $d_{\mathbb{H}}(T)$ that can be computed (for these Boolean map functions). Using that, he extends the previous theorems obtaining "computable" bounds and he derives a lower bound for m, if m is below that lower bound then we can find tasks such that the error made by our estimator is larger then certain ϵ.

3.2.2 A notion of Task relatedness

3.2.3 Learning Under Privileged Information

3.3 Multi-Task Learning Methods: An Overview

3.3.1 Bias Learning Approach

3.3.2 Low-Rank Approach

3.3.3 Learning Task Relations

3.3.4 Decomposition Approach

3.4 Deep Multi-Task Learning

3.4.1 Hard Parameter Sharing

3.4.2 Soft Parameter Sharing

3.5 Multi-Task Learning with Kernel Methods

3.6 Conclusions

In this chapter, we covered...

Chapter 4

A Convex Formulation for Regularized Multi-Task Learning

4.1 Introduction

4.2 Convex Multi-Task Learning Support Vector Machines

4.2.1 Convex Formulation

4.2.2 L1 Support Vector Machine

4.2.3 L2 Support Vector Machine

4.2.4 LS Support Vector Machine

4.3 Optimal Convex Combination of trained models

4.4 Experiments

4.5 Conclusions

In this chapter, we have...

Chapter 5

Adaptive Graph Laplacian Multi-Task Support Vector Machine

5.1 Introduction

5.2 Graph Laplacian Multi-Task Support Vector Machine

5.3 Adaptive Graph Laplacian Algorithm

5.4 Experiments

5.5 Conclusions

In this chapter, we have...

Bibliography

Baxter, J. (2000). A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198.

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media.