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 hypothesis $\mathcal{H} = \{h(x, \alpha), \alpha \in A\}$, where $A$ 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 hypothesis space is parametric, i.e. $\mathcal{H} = \{h(x, \alpha), \alpha \in A \subset \mathbb{R}^m\}$. The learner select the parameter $\alpha$ 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 $\hat{R}_z$ is minimized with the hope of minimizing the true expected error $R_F$, which would result in a good generalization. Several models use the ERM principle to generalize from data such as Neural Networks or Support Vector Machines. These methods are designed to find a good hypothesis $h(x, \alpha)$ from a given space $\mathcal{H}$. The definition of such space $\mathcal{H}$ define the bias for these problems. If $\mathcal{H}$ does not contain any good hypothesis, the learner will not be able to learn.

The best hypothesis space we can provide is the one containing only the optimal hypothesis, but this is the original problem that we want to solve. Therefore, in the single task scenario, there is no difference between bias learning and ordinary learning. Instead, we focus on the situation where we want to solve multiple related tasks. In that case, we can obtain a good space $\mathcal{H}$ that contains good solutions for the different tasks. 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 strict multi-task learning. This is done by defining an environment of tasks and extending the work of Vapnik (2013), which defines the capacity of space of hypothesis, Baxter defines the capacity of a family of spaces of hypothesis.

Before presenting the concepts defined for Bias Learning, and to establish an analogy to those of ordinary learning, we briefly review some statistical learning concepts.

### Ordinary Learning

In the ordinary statistical learning, some theoretical concepts are used:

- an *input space* $\mathcal{X}$ and an *output space* $\mathcal{Y}$,
- a *probability distribution* $F$, which is unknown, defined over $\mathcal{X} \times \mathcal{Y}$,
- a *loss function* $\ell(\cdot, \cdot) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, and
- a *hypothesis space* $\mathcal{H} = \{h(x, \alpha), \alpha \in A \subset \mathbb{R}^m\}$ with hypothesis $h(\cdot, \alpha) : \mathcal{X} \to \mathcal{Y}$.

The goal for the learner is to select a hypothesis $h(x, \alpha) \in \mathcal{H}$, or equivalently $\alpha \in A$, that minimizes the expected risk

$$R_F(\alpha) = \int_{\mathcal{X} \times \mathcal{Y}} \ell(h(x, \alpha), y)\, dF(x, y).$$

The distribution $F$ is unknown, but we have a training set $z = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ of samples drawn from $F$. The approach is then is to apply the ERM inductive principle, that is to minimize the empirical risk

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

Thus, a learner $\mathcal{A}$ maps the set of training samples to a set of hypothesis:

$$\mathcal{A} : \bigcup (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}.$$

Although $\hat{R}_z(\alpha)$ is an unbiased estimator of $R_F(\alpha)$, it has been shown Vapnik (2013) that this approach, despite being the most evident one, is not the best principle that can be followed. This has 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 be said about $R_F$ when $\alpha$ minimizes $\hat{R}_z(\alpha)$, and moreover, his results are valid also for small number of training samples $n$. More specifically, Vapnik sets the sufficient and necessary conditions for the consistency of an inductive learning process, i.e. for $\hat{R}_z(\alpha) \xrightarrow{P} R_F(\alpha)$ uniformly. Vapnik also defines the capacity of a hypothesis space and use it to derive bounds on the rate of this convergence for any $\alpha \in A$ and, more importantly, bounds on the difference $\inf_{\alpha \in A} \hat{R}_z(\alpha) - \inf_{\alpha \in A} R_F(\alpha)$. Under some general conditions, he proves that

$$\inf_{\alpha \in A} \hat{R}_z(\alpha) - \inf_{\alpha \in A} R_F(\alpha) \leq B(n/\mathrm{VCdim}(\mathcal{H})) \tag{3.1}$$

where $B$ is some non-decreasing function and $\mathrm{VCdim}(\mathcal{H})$ is the capacity of the space $\mathcal{H}$, also named the VC-dimension $\mathcal{H}$. This means that the generalization ability of a learning process can be controlled in terms of two factors:

- The number of training samples $n$. A greater number of training samples assures a better generalization of the learning process.This looks intuitive and could be already inferred from the asymptotical properties.

- The VC-dimension VCdim $(\mathcal{H})$ of the hypothesis space $\mathcal{H}$, which is desirable to be small. This term is not intuitive and is the most important term in Vapnik theory.

The VC-dimension measures the capacity of a set of hypothesis $\mathcal{H}$. If the capacity of the set $\mathcal{H}$ is too large, we may find a hypothesis $h\left(x, \alpha^*\right)$ that minimizes $\hat{R}_z$ but does not generalize well and therefore, does not minimize $R_F$. This is the overfitting problem. On the other side, if we use a simple $\mathcal{H}$, with low capacity, we could be in a situation where there is not a good hypothesis $h\left(x, \alpha\right) \in \mathcal{H}$, so the empirical risk $\inf_{\alpha \in A} R_F$ is too large. This is the underfitting problem.

### Bias Learning: Concept and Components

In Baxter (2000) two main concepts are presented: the *family of hypothesis spaces* and an *environment* of related tasks. For simplicity we write $h(x)$ instead of $h(x, \alpha)$, and since $\alpha$ completely defines $h$, we also substitute $\alpha$ by $h$ for an easier notation. Using these concepts, the bias learning problem has the following components:

- an *input space* $\mathcal{X}$ and an *output space* $\mathcal{Y}$,
- an *environment* $(\mathcal{P}, Q)$ where $\mathcal{P}$ is a set of distributions $P$ defined over $\mathcal{X} \times \mathcal{Y}$, and we can sample from $\mathcal{P}$ according to a distribution $Q$,
- a *loss function* $\ell\left(\cdot, \cdot\right) : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$, and
- a *family of hypothesis spaces* $\mathbb{H} = \{\mathcal{H}_\eta, \eta \in \eta\}$, where each element $\mathcal{H}_\eta$ is a set of hypothesis.

Analogous to ordinary learning, the goal is to minimize the expected risk, defined as

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

Again, we do not know $\mathcal{P}$ nor $Q$, but we have a training set samples from the environment $(\mathcal{P}, Q)$ 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 $m$ pairs $z_r = \{(x_1^r, y_1^r), \ldots, (x_m^r, y_m^r)\}$ according to $P_r$ where $(x_i^r, y_i^r) \in X \times Y$ .

We obtain a sample $z = \{(x_i^r, y_i^r), r = 1, \ i = 1, \ldots, \ m = 1, \ldots, T\}$, with $m$ examples from $T$ different learning tasks, and

$$\boldsymbol{z} := \begin{matrix} (x_1^1, y_1^1) & \ldots & (x_m^1, y_m^1) \\ \vdots & \ddots & \vdots \\ (x_1^T, y_1^T) & \ldots & (x_m^T, y_m^T) \end{matrix}$$

is named as a $(T, m)$-sample. Using $\boldsymbol{z}$ we can define the empirical loss as

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

which is an average of the empirical losses of each task. Note, however, that in the case of the bias learner, this estimate is biased, since $R_{P_r}(h)$ does not coincide with $\hat{R}_{z_r}(h)$.

Putting all together, a bias learner $\mathcal{A}$ maps the set of all $(T, m)$-samples to a family of hypothesis spaces:

$$\mathcal{A} : \bigcup (\mathcal{X} \times \mathcal{Y})^{(T,m)} \to \mathbb{H}.$$

To follow an analogous path to that of ordinary learning, the milestones in bias learning theory should include:

- Checking the consistency of the Bias Learning methods, i.e. proving that $\hat{R}_{\boldsymbol{z}}(\eta)$ converges uniformly in probability to $R_Q(\eta)$.
- Defining a notion of capacity of hypothesis space families $\mathbb{H}$.
- Finding a bound of $\hat{R}_{\boldsymbol{z}}(\eta) - R_Q(\eta)$ for any $\eta$ using the capacity of the hypothesis space family. If possible, finding also a bound for $\inf_{\eta\in\eta} \hat{R}_{\boldsymbol{z}}(\eta) - \inf_{\eta\in\eta} R_Q(\eta)$.

To achieve these goals some previous definitions are needed. From this point, since any $\mathcal{H}$ is defined by a $\eta \in \eta$, we omit $\eta$ and write just $\mathcal{H}$ for simplicity.

**Bias Learning: Capacities and Uniform Convergence**

In first place, a *sample-driven* pseudo-metric of $(T, 1)$-empirical risks is defined. Consider a sequence of $T$ probabilities $\boldsymbol{P} = (P_1, \ldots, P_T)$ sampled from $\mathcal{P}$ according the the distribution $Q$. Consider also the set of sequences of $T$ hypothesis

$$\mathcal{H}^T := \{\boldsymbol{h} = (h_1, \ldots, h_T), h_1, \ldots, h_T \in \mathcal{H}\}.$$

We can define then the set of $(T, 1)$-empirical risks as

$$\mathcal{H}_\ell^T := \left\{ \boldsymbol{h}_\ell(x_1, y_1, \ldots, x_T, y_T) = \sum_{r=1}^{T} \ell(h(x_i), y_i), h_1, \ldots, h_T \in \mathcal{H} \right\}$$

The family of the set of $T$-risks of hypothesis is then $\mathbb{H}^T = \bigcup_{\mathcal{H}\in\mathbb{H}} \mathcal{H}^T$. Now we can define

$$d_{\boldsymbol{P}}(\boldsymbol{h}_\ell, \boldsymbol{h}'_\ell) = \int_{(\mathcal{X}\times\mathcal{Y})^T} \left| \boldsymbol{h}_\ell(x_1, y_1, \ldots, x_T, y_T) - \boldsymbol{h}'_\ell(x_1, y_1, \ldots, x_T, y_T) \right| dP_1(x_1, y_1) \ldots dP_T(x_T, y_T)$$

for $\boldsymbol{h}_\ell, \boldsymbol{h}'_\ell \in \mathcal{H}_\ell, \mathcal{H}'_\ell$ as a pseudo-metric in $\mathbb{H}^T$.

Then, a *distribution-driven* pseudo-metric is defined. Given a distribution $P$ on $\mathcal{X} \times \mathcal{Y}$. Consider the set of infimum expected risk for each $\mathcal{H}$:

$$\mathcal{H}^* := \inf_{h\in\mathcal{H}} R_P(h).$$

The family of such sets is defined as $\mathbb{H}^* = \{\mathcal{H}^*, \mathcal{H} \in \mathbb{H}\}$. The pseudo-metric in this space is given by $Q$:

$$d_Q = \int_{\mathcal{P}} |\mathcal{H}_1^* - \mathcal{H}_2^*| \, dQ$$

With these two pseudo-metrics, two capacities for families of hypothesis spaces are defined. For that the definition of $\epsilon$-cover is needed. Given a pseudo-metric $d_S$ in a space $\mathcal{S}$, a set of $l$ elements $s_1, \ldots, s_l \in \mathcal{S}$ is an $\epsilon$-cover of $\mathcal{S}$ if $d_S(s, s_i) \leq \epsilon$ for some $i = 1, \ldots, l$. Let $\mathcal{N}(\epsilon, \mathcal{S}, d_S)$ denote the size of the smallest $\epsilon$-cover. Then, we can define the following capacities of a family space $\mathbb{H}$:

- The *sample-driven capacity* $C\left(\epsilon, \mathbb{H}^T\right) := \sup_{\boldsymbol{P}} \mathcal{N}(\epsilon, \mathbb{H}^T, d_{\boldsymbol{P}})$.
- The *distribution-driven capacity* $C\left(\epsilon, \mathbb{H}^*\right) := \sup_{Q} \mathcal{N}(\epsilon, \mathbb{H}^*, d_{Q})$.

Using these capacities, the convergence (uniformly over all $\mathcal{H} \in \mathbb{H}$) of bias learners can be proved (Baxter, 2000, Theorem 2). Moreover, the bias expected risk is bounded

$$\hat{R}_{\boldsymbol{z}}(\mathcal{H}) \leq R_Q(\mathcal{H}) + \epsilon$$

with probability $1 - \eta$, given sufficiently large $T$ and $m$,

$$T \geq \max\left(\frac{256}{T\epsilon^2} \log \frac{8C\left(\frac{\epsilon}{32}, \mathbb{H}^*\right)}{\eta}, \frac{64}{\epsilon^2}\right), \; m \geq \max\left(\frac{256}{T\epsilon^2} \log \frac{8C\left(\frac{\epsilon}{32}, \mathbb{H}^T\right)}{\eta}, \frac{64}{\epsilon^2}\right).$$

It should be noted that the bound for $m$ is inversely proportional to $T$, that is, the more tasks we have, the less samples we need for each task.

### Multi-Task Learning

The previous result is a result for pure Bias Learning, where we have an $(\mathbb{H}^*, Q)$-environment of tasks. In Multi-Task Learning, we have a fixed number of tasks $T$ and a fixed sequence of distributions $\boldsymbol{P} = (P_1, \ldots, P_T)$, where $P_i$ is a distribution over $(\mathcal{X} \times \mathcal{Y})^m$. The goal is not learning a hypothesis space $\mathcal{H}$ but a sequence of hypothesis $\boldsymbol{h} = (h_1, \ldots, h_T), h_1, \ldots, h_T \in \mathcal{H}$. Thus, the Multi-Task expected risk is

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

and the empirical risk is defined as

$$\hat{R}_{\boldsymbol{z}}(\boldsymbol{h}) = \sum_{r=1}^{T} \hat{R}_{z_r}(h_r) = \sum_{r=1}^{T} \sum_{i=1}^{m} l(h_r(x_i^r), y_i^r). \tag{3.5}$$

A similar result to that of Bias Learning is given for Multi-Task Learning (Baxter, 2000, Theorem 4):

$$\hat{R}_{\boldsymbol{z}}(\boldsymbol{h}) \leq R_{\boldsymbol{P}}(\boldsymbol{h}) + \epsilon,$$

with probability $1 - \eta$ given that the number of samples per task

$$m \geq \max\left(\frac{64}{T\epsilon^2} \log \frac{4C\left(\frac{\epsilon}{16}, \mathbb{H}^T\right)}{\eta}, \frac{16}{\epsilon^2}\right).$$

Observe that we do not need the *distribution-driven* capacity in this case, just the *sample-driven* capacity.

### Feature Learning

Feature Learning is a common way to encode bias. The most popular example are Neural Networks, where all the hidden layers can be seen as a Feature Learning engine that learns a mapping from the original space to a space with "strong" features. In general, a set of "strong" feature maps is defined as $\mathcal{F} = \{f, f : \mathcal{X} \to \mathcal{V}\}$. Using these features, functions $g \in \mathcal{G}$ (which are tipically simple) are built: $\mathcal{X} \to_f \mathcal{V} \to_g \mathcal{Y}$. Thus, for each

map $f$, the hypothesis space can be expressed as $\mathcal{H}_f = \{h = \mathcal{G} \circ f, g \in \mathcal{G}\}$ , and the family of hypothesis spaces is $\mathbb{H} = \{\mathcal{H}_f, f \in \mathcal{F}\}$. Now, the Bias Learning problem is the problem of finding a good mapping $f$. It is proved (Baxter, 2000, Theorem 6) that in the Feature Learning case the capacities of $\mathbb{H}$ can be bounded by the capacities of $\mathcal{F}$ and $\mathcal{G}$ as

$$C\left(\epsilon, \mathbb{H}^T\right) \leq C\left(\epsilon_1, \mathcal{G}^T\right)^T C_{\mathcal{G}_\ell}(\epsilon_2, \mathcal{F}),$$
$$C\left(\epsilon, \mathbb{H}^*\right) \leq C_{\mathcal{G}_\ell}(\epsilon, \mathcal{F})$$

with $\epsilon = \epsilon_1 + \epsilon_2$. Here, $C_{\mathcal{G}_\ell}(\epsilon, \mathcal{F})$ is defined as $C_{\mathcal{G}_\ell}(\epsilon, \mathcal{F}) := \sup_P \mathcal{N}(\epsilon, \mathcal{F}, d_{[P,\mathcal{G}_\ell]})$, where

$$d_{[P,\mathcal{G}_\ell]}(f, f') = \int_{\mathcal{X}\times\mathcal{Y}} sup_{g\in\mathcal{G}} \left|\ell\left(g \circ f(x), y\right) - \ell\left(g \circ f'(x), y\right)\right| dP(x, y)$$

is a pseudo-metric. Using these results alongside those presented for Bias Learning is useful to establish bounds for Feature Learning models like Neural Networks.

### Generalized VC-Dimension for Multi-Task Learning

The concepts presented until now rely on the concepts of two capacities of a family of hypothesis spaces $\mathbb{H}$ to establish bounds in the difference $\hat{R}_{\boldsymbol{z}}(\boldsymbol{h}) - R_Q(\boldsymbol{h})$, that is, the probability of large deviations between the empirical and expected risks for a given hypothesis sequence. However, it would be more interesting to establish some bounds between the empirical error and the *best expected error*. To overcome this limitations, a generalized VC-dimension is developed in Baxter (2000) for Multi-Task Learning with Boolean hypothesis.

Let $\mathcal{H}$ be a space of boolean functions and $\mathbb{H}$ a boolean hypothesis space family. Denote the set of $T \times m$ matrices in $\mathcal{X}$ as $\mathcal{X}^{T\times m}$ For each $X \in \mathcal{X}^{T\times m}$ and each $\mathcal{H} \in \mathbb{H}$ define the set of binary $T \times m$ matrices

$$\mathcal{H}_{|X} := \left\{ \begin{pmatrix} h\left(x_1^1\right) & \dots & h\left(x_m^1\right) \\ \vdots & \ddots & \vdots \\ h\left(x_1^T\right) & \dots & h\left(x_m^T\right) \end{pmatrix}, h \in \mathcal{H} \right\},$$

and the corresponding family of such sets as

$$\mathbb{H}_{|X} = \bigcup_{\mathcal{H}\in\mathbb{H}} \mathcal{H}_{|X}.$$

For each $T, m \geq 0$ define the number of binary matrices obtainable with $\mathbb{H}$ as

$$\Pi_{\mathbb{H}}(T, m) := \max_{X\in\mathcal{X}^{T\times m}} \left|\mathbb{H}_{|X}\right|.$$

Note that $\Pi_{\mathbb{H}}(T,m) \leq 2^{Tm}$ and if $\Pi_{\mathbb{H}}(T,m) \leq 2^{Tm}$ we say that $\mathbb{H}$ shatters $\mathcal{X}^{T\times m}$. For each $n > 0$ define

$$\begin{aligned} d_{\mathbb{H}}(T) &:= \max_{m:\Pi_{\mathbb{H}}(T,m)=2^{Tm}} m, \\ \overline{d}(\mathbb{H}) &:= \mathrm{VCdim}\left(\mathbb{H}^1\right) = \mathrm{VCdim}\left(\bigcup_{\mathcal{H}\in\mathbb{H}} \mathcal{H}\right), \\ \underline{d}(\mathbb{H}) &:= \max_{\mathcal{H}\in\mathbb{H}} \mathrm{VCdim}\left(\mathcal{H}\right). \end{aligned}$$

Here, $d_{\mathbb{H}}(T)$ is the generalized VC-dimension, and

$$d_{\mathbb{H}}(T) \geq \max\left(\left\lfloor \frac{\overline{d}(\mathbb{H})}{T} \right\rfloor, \underline{d}(\mathbb{H})\right)$$

where it can be observed that

$$\overline{d}(\mathbb{H}) \geq d_{\mathbb{H}}(T) \geq \underline{d}(\mathbb{H}). \tag{3.6}$$

Proof? Now we can present the relevant result expressed in (Baxter, 2000, Corollary 13).

**Theorem 3.1.** *Given a sequence* $\boldsymbol{P} = (P_1, \ldots, P_T)$ *on* $(\mathcal{X} \times \{0,1\})^T$*, and a sample* $\boldsymbol{z}$ *from this distribution. Consider also a sequence* $\boldsymbol{h} = (h_1, \ldots, h_T)$ *of boolean hypothesis* $h_i \in \mathcal{H}$*, then for every* $\epsilon > 0$

$$\left| R_{\boldsymbol{P}}(\boldsymbol{h}) - \hat{R}_{\boldsymbol{z}}(\boldsymbol{h}) \right| \leq \epsilon,$$

*with probability* $1-\eta$ *given that the number of samples per task*

$$m \geq \frac{88}{\epsilon^2}\left[2d_{\mathbb{H}}(T)\log\frac{22}{\epsilon} + \frac{1}{T}\log\frac{4}{\eta}\right]. \tag{3.7}$$

Here, since $d_{\mathbb{H}}(T) \geq d_{\mathbb{H}}(T+1)$, it is easy to see that as the number of task $T$ increases, the number of examples needed per task can decrease. Moreover, as shown in (Baxter, 2000, Theorem 14), if this bound on $m$ is not fulfilled, then we can always find a sequence of distributions $\boldsymbol{P}$ such that

$$\inf_{h\in\mathcal{H}} \hat{R}_{\boldsymbol{z}}(\boldsymbol{h}) > \inf_{h\in\mathcal{H}} R_{\boldsymbol{P}}(\boldsymbol{h}) + \epsilon.$$

With this results we can see that the condition (3.7) has some important properties:

- It is a computable bound, given that we know how to compute $d_{\mathbb{H}}(T)$.
- It provides a sufficient condition for the uniform convergence (in probability) of the empirical risk to the expected risk.
- It provides a necessary condition for the consistency of Multi-Task Learners, i.e. uniform convergence of the best empirical risk to the best expected risk.

### Conclusion

In Baxter (2000) several new concepts are developed. The $(\mathcal{P}, Q)$-environment of tasks is useful to characterize the concept of related tasks. Moreover, using this definition,

Baxter is able to give some important results of uniform convergence in the Bias Learning paradigm. From this general view, Multi-Task Learning is a particular case and the uniform convergence results are also valid. The Feature Learning approach, which can be seen as a more particular method of Multi-Task Learning has some interesting results splitting the analysis into the feature learning process and the construction of models over these features. Finally, the most important result is the definition of a generalized VC-dimension and the uniform convergence of Multi-Task Learning models using this concept. Although this is a result only valid for boolean hypothesis, it helps to shed some light on Multi-Task Learning and the reasons of its effectiveness.

### 3.2.2 Learning with Related Tasks

Using the work of Baxter (2000) as the foundation, several important notions and results are presented in Ben-David and Borbely (2008) for boolean hypothesis functions defined over $\mathcal{X} \times \{0, 1\}$. One of the main contributions of this work is a notion of task relatedness. In Baxter (2000) the tasks are related by sharing a common inductive bias that can be learned. In Ben-David and Borbely (2008) a precise mathematical definition for task relatedness is given. The other important contribution is the focus on the individual risk of each task. In Baxter (2000) all the results are given for the Multi-Task empirical and expected risks, which are an average of the risks of each task. However, bounding this average does not bound the risk of each particular task. This is specially relevant if we are in a Transfer Learning scenario, where there is a target task that we want to solve and the remaining tasks can be seen as an aid to improve the performance in the target.

#### A Notion of Task Relatedness: $\mathcal{F}$-Related Tasks

The main concept for the theory developed in Ben-David and Borbely (2008) is a set of $\mathcal{F}$ of transformations $f : \mathcal{X} \to \mathcal{X}$. Given a probability distribution $F$ over $\mathcal{X} \times \{0, 1\}$, a set of tasks with distributions $P_1, \ldots, P_T$ are $\mathcal{F}$-related if, for each task there exists some $f_i \in \mathcal{F}$ such that $P_i = f_i(F)$.

**Definition 3.2** ($\mathcal{F}$-related task)**.** Consider a measurable space $(\mathcal{X}, \mathcal{A})$ and the corresponding measurable product space $(\mathcal{X} \times \{0, 1\}, \mathcal{A} \times \wp(\{0, 1\}))$. Consider $P$ a probability distribution over this product space and a function $f : \mathcal{X} \to \mathcal{X}$, then we define the distribution $f[P]$ such that for any $S \in \mathcal{A}$,

$$f[P](S) := P(\{(f(x), b), (x, b) \in S\}).$$

Let $\mathcal{F}$ be a set of transformations $f : \mathcal{X} \to \mathcal{X}$, and let $P_1, P_2$ be distributions over $(\mathcal{X} \times \{0, 1\}, \mathcal{A} \times \wp(\{0, 1\}))$, then the distributions $P_1, P_2$ are $\mathcal{F}$-related if $f[P_1] = P_2$ or $f[P_2] = P_1$ for some $f \in \mathcal{F}$.

This notion establishes a clear definition of related tasks but we are interested in how a learner can use this relatedness to improve the learning process. For that, considering that $\mathcal{F}$ is a group under function composition, we regard at the action of the group $\mathcal{F}$ over the set of hypothesis $\mathcal{H}$. This action defines the following equivalence relation in $\mathcal{H}$:

$$h_1 \sim_{\mathcal{F}} h_2 \iff \exists f \in \mathcal{F}, h_1 \circ f = h_2.$$

This equivalence relation defines equivalence classes $[h]$, that is let $h' \in \mathcal{H}$ be an hypothesis, then $h' \in [h]$ iff $h' \sim_{\mathcal{F}} h$. We consider the quotient space

$$\mathcal{H}_{\mathcal{F}} := \mathcal{H}/\sim_{\mathcal{F}} = \{[h], h \in \mathcal{H}\}.$$

It is important to observe that $\mathcal{H}_{\mathcal{F}} = \mathbb{H}'$ is a hypothesis space family, since it is a set of equivalence classes $[h] = \mathcal{H}'$, which are set of hypothesis.

### The Multi-Task Empirical Risk Minimization

This equivalence classes are useful to divide the learning process in two stages, this is called the *Multi-Task ERM*. Consider the samples $z_1, \dots, z_T$ from $T$ different tasks, then

1. Select the best hypothesis class $[h^{\mathcal{F}}] \in \mathcal{H}_{\mathcal{F}}$:

$$[h^{\mathcal{F}}] := \min_{[h] \in \mathcal{H}_{\mathcal{F}}} \inf_{h_1, \dots, h_T \in [h]} \frac{1}{T} \sum_{r=1}^{T} \hat{R}_{z_i}(h_i),$$

2. Select the best hypothesis $h^{\diamond}$ for the target task (without loss of generality, consider the first one):

$$h^{\diamond} = \inf_{h \in [h^{\mathcal{F}}]} \hat{R}_{z_1}(h).$$

For example, consider the handwritten digits recognition problem, we might integrate $T$ different datasets designed in different conditions. Each dataset have been created using certain conditions of light and some specific scanner for getting the images. Even different pens or pencils might be influential in the stroke of the numbers. All these conditions are the $\mathcal{F}$ transformations, and each $f \in \mathcal{F}$ generate a different bias for the dataset. However, there exists a probability for "pure" digits, e.g. the pixels of digit one have higher probability around a line in the middle of the picture than in the sides. This "pure" probability distribution $P_0$ and all the distributions $P_1, \dots, P_T$ from which our datasets have been sampled might be $\mathcal{F}$-related among them and with $P_0$. If we first determine the $\mathcal{F}$-equivalent class of hypothesis $[h]$ suited for digit recognition in the first stage, then it will be easier to select $h_1, \dots, h_T \in [h]$ for each dataset in the second one.

### Bounds for $\mathcal{F}$-Related Tasks

The results of Theorem 3.1 can be applied to the hypothesis quotient space of equivalent classes $\mathcal{H}_{\mathcal{F}}$. However the following results is needed first. Let $P_1, P_2$ be $\mathcal{F}$-related distributions, then this statement can be proved (Ben-David and Borbely, 2008, Lemma 2):

$$\inf_{h \in \mathcal{H}} R_{P_1}(h) = \inf_{h \in \mathcal{H}} R_{P_2}(h). \tag{3.8}$$

This indicates that the the expected risk is invariant under transformations of $\mathcal{F}$. Now, one of the main results (Baxter, 2000, Theorem 2) can be given.

**Theorem 3.3.** *Let $\mathcal{F}$ be a set of transformations $f : \mathcal{X} \to \mathcal{X}$ that is a group under function composition. Let $\mathcal{H}$ be a hypothesis space so that $\mathcal{F}$ acts as a group over $\mathcal{H}$, and consider the quotient space $\mathcal{H}_{\mathcal{F}} = \{[h], h \in \mathcal{H}\}$. Consider $\boldsymbol{P} = (P_1, \dots, P_T)$ a sequence of $\mathcal{F}$-related distributions over $\mathcal{X} \times \{0, 1\}$, and $\boldsymbol{z} = (z_1, \dots, z_T)$ the corresponding sequence*

*of samples where $z_i$ is sampled using $P_i$, then for every $[h] \in \mathcal{H}_{\mathcal{F}}$ and $\epsilon > 0$*

$$\left| \inf_{h_1,\dots,h_T \in [h]} \frac{1}{T} \sum_{r=1}^{T} \hat{R}_{z_r}(h_r) - \inf_{h' \in [h]} R_{P_1}(h') \right| \leq \epsilon$$

*with probability greater than $\eta$ if the number of samples from each distribution satisfies*

$$|z_i| \geq \frac{88}{\epsilon^2} \left[ 2d_{\mathcal{H}_{\mathcal{F}}}(T) \log \frac{22}{\epsilon} + \frac{1}{T} \log \frac{4}{\eta} \right]. \tag{3.9}$$

Note that, in contrast to Theorem 3.1, this result bounds the expected risk of a single task, not the average risk. This is the consequence of applying Theorem 3.1 and substituting the average empirical error using the result from (3.8). Also observe that here the hypothesis space family used is the quotient space $\mathcal{H}_{\mathcal{F}}$, and the VC-dimension of such family is used. Using this result, a bound for learners using the Multi-Task ERM principle is given (Ben-David and Borbely, 2008, Theorem 3)

**Theorem 3.4.** *Consider $\mathcal{F}$ and $\mathcal{H}$ as in the previous theorem. Consider also the previous sequences of distributions $(P_1, \dots, P_T)$ and corresponding samples $(z_1, \dots, z_T)$. Consider $\underline{d}(\mathcal{H}_{\mathcal{F}}) = \max_{h \in \mathcal{H}} VCdim([h])$. Let $h^\diamond$ be the hypothesis selected using the Multi-Task ERM principle, then for every $\epsilon_1, \epsilon_2 > 0$*

$$\hat{R}_{z_1}(h^\diamond) - \inf_{h' \in \mathcal{H}} R_{P_1}(h') \leq 2(\epsilon_1 + \epsilon_2)$$

*with probability greater than $\eta$ if*

$$|z_1| \geq \frac{64}{\epsilon^2} \left[ 2\underline{d}(\mathcal{H}_{\mathcal{F}}) \log \frac{12}{\epsilon} + \frac{1}{T} \log \frac{8}{\eta} \right], \tag{3.10}$$

*and for $i \neq 1$*

$$|z_i| \geq \frac{88}{\epsilon^2} \left[ 2d_{\mathcal{H}_{\mathcal{F}}}(T) \log \frac{22}{\epsilon} + \frac{1}{T} \log \frac{8}{\eta} \right]. \tag{3.11}$$

The idea of the proof of this theorem helps to understand how using different tasks can help to improve the performance in the target task. Consider $h^* = \inf_{h \in \mathcal{H}} R_{P_1}(h)$ the best hypothesis for the $P_1$ distribution. According to Theorem 3.3, for $[h^*]$ we have that

$$\inf_{h_1,\dots,h_T \in [h^*]} \frac{1}{T} \sum_{r=1}^{T} \hat{R}_{z_r}(h_r) \leq \inf_{h' \in [h^*]} R_{P_1}(h') + \epsilon_1.$$

Also, in the first stage of Multi-Task ERM principle, we select the hypothesis class $[h^{\mathcal{F}}]$ that minimizes $\inf_{\boldsymbol{h} \in [h]} R_{\boldsymbol{P}}(\boldsymbol{h})$ where $\boldsymbol{h}$ is a sequence of hypothesis of $\mathcal{H}_{\mathcal{F}}$. According to Theorem 3.3, for $[h^{\mathcal{F}}]$ we have that

$$\inf_{h' \in [h^{\mathcal{F}}]} R_{P_1}(h') \leq \inf_{h_1,\dots,h_T \in [h^{\mathcal{F}}]} \frac{1}{T} \sum_{r=1}^{T} \hat{R}_{z_r}(h_r) + \epsilon_1.$$

Using these two inequalities we get

$$\inf_{h' \in [h^{\mathcal{F}}]} R_{P_1}(h') \leq \inf_{h' \in [h^*]} R_{P_1}(h') + 2\epsilon_1$$

under the condition (3.9). This bounds the risk of the hypothesis space given by the equivalence class of $h^{\mathcal{F}}$ and establishes the inequality (3.11).

Once we select $[h^{\mathcal{F}}]$, the second stage is just ERM using this hypothesis space. According to the reference?

$$\inf_{h\in\mathcal{H}} R_{z_1}(h) - \inf_{h\in\mathcal{H}} R_{P_1}(h) \leq \epsilon_2$$

if

$$|z_1| \geq \frac{64}{\epsilon^2}\left[2\text{VCdim}\left(\mathcal{H}\right)\log\frac{12}{\epsilon} + \frac{1}{T}\log\frac{8}{\eta}\right].$$

Since the ERM will not use the whole space $\mathcal{H}$ but the subset $[h^{\mathcal{F}}] \subset \mathcal{H}$, and

$$\text{VCdim}\left([h^{\mathcal{F}}]\right) \leq \max_{[h]\in\mathcal{H}_{\mathcal{F}}} \text{VCdim}\left([h]\right) = \underline{d}(\mathcal{H}_{\mathcal{F}}).$$

then we can write the inequality (3.10) of the theorem. The advantage of using multiple tasks is then illustrated in this bound and it will be defined by the gap between VCdim $(\mathcal{H})$ and $\underline{d}(\mathcal{H}_{\mathcal{F}})$. If $\underline{d}(\mathcal{H}_{\mathcal{F}})$ is smaller than VCdim $(\mathcal{H})$, the number of samples needed to solve the target task will also be smaller. Also, the sample complexity of the rest of tasks is given by $d_{\mathcal{H}_{\mathcal{F}}}(T)$.

That is, Multi-Task Learning allows to select a subset of hypothesis from which a learner can use the ERM principle. In this stage, the sample complexity is controlled by the generalized VC-dimension of the set of equivalent classes of hypothesis. Once the best equivalent class has been selected, the VC-dimension of this subset, compared to the VC-dimension of the whole set of hypothesis, is what marks the difference between Single Task and Multi-Task Learning.

### Analysis of generalized VC-dimension with $\mathcal{F}$-related tasks

As we have seen in Theorem 3.4, the VC-dimensions VCdim $(\mathcal{H})$, $\underline{d}(\mathcal{H}_{\mathcal{F}})$ and $d_{\mathcal{H}_{\mathcal{F}}}(T)$ are crucial for stating the advantage of Multi-Task over Single Task Learning. To understand better how these concepts interact, Ben-David et al. give some theoretical results. Recall that, given a hypothesis space $\mathcal{H}$, $\mathcal{H}_{\mathcal{F}}$ is a family of hypothesis spaces composed by the hypothesis spaces $[h], h \in \mathcal{H}$, then

$$\begin{aligned}
d_{\mathcal{H}_{\mathcal{F}}}(T) &= \max_{\left\{m,\Pi_{\mathcal{H}_{\mathcal{F}}}=2^{Tm}\right\}} m,\\
\underline{d}(\mathcal{H}_{\mathcal{F}}) &= max_{h\in\mathcal{H}}\text{VCdim}\left([h]\right),\\
\overline{d}(\mathcal{H}_{\mathcal{F}}) &= \text{VCdim}\left(\bigcup_{[h]\in\mathcal{H}_{\mathcal{F}}}[h]\right) = \text{VCdim}\left(\mathcal{H}\right).
\end{aligned}$$

Using the result from (3.6) we observe that

$$\underline{d}(\mathcal{H}_{\mathcal{F}}) \leq d_{\mathcal{H}_{\mathcal{F}}}(T) \leq \text{VCdim}\left(\mathcal{H}\right).$$

That is, the best we can hope when bounding the sample complexity in Theorem 3.4 is $\underline{d}(\mathcal{H}_{\mathcal{F}}) = d_{\mathcal{H}_{\mathcal{F}}}(T)$. Ben-David et al. give evidence that, with some restrictions on $\mathcal{H}$, this lower bound can be achieved (Ben-David and Borbely, 2008, Theorem 4).

**Theorem 3.5.** *If the support of $h$ is bounded, i.e. $|\{x \in \mathcal{X}, h(x) = 1\}| < M$, for all $h \in \mathcal{H}$, then there exists $T_0$ such that for all $T > T_0$*

$$d_{\mathcal{H}_\mathcal{F}}(T) = \underline{d}(\mathcal{H}_\mathcal{F}).$$

Thus, a sufficient condition on the hypothesis space $\mathcal{H}$ to achieve the lowest $d_{\mathcal{H}_\mathcal{F}}(T)$ is a bounded support of any hypothesis. Although this condition may be too restricting, it can also be proved that the upper limit of $d_{\mathcal{H}_\mathcal{F}}(T)$, that is, VCdim $(\mathcal{H})$, under some conditions on $\mathcal{F}$ is not achieved.

The following result (Ben-David and Borbely, 2008, Theorem 6) shows this.

**Theorem 3.6.** *If $\mathcal{F}$ is finite and $\frac{T}{\log(T)} \geq VCdim\,(\mathcal{H})$, then*

$$d_{\mathcal{H}_\mathcal{F}}(T) \leq 2\log(|\mathcal{F}|)$$

This inequality indicates that, given a finite set of transformation $\mathcal{F}$, there are scenarios when VCdim $(\mathcal{H})$ is arbitrarily large but $d_{\mathcal{H}_\mathcal{F}}(T)$ is bounded, and therefore, the right-hand side of inequality (3.11) is also bounded. That is, the Multi-Task bound, which substitutes VCdim $(\mathcal{H})$ by $d_{\mathcal{H}_\mathcal{F}}(T)$ is a better one in this cases.

#### Conclusion

TODO?

### 3.2.3 Learning Under Privileged Information

The standard machine learning paradigm tries to find the hypothesis $h$ from a set of hypothesis $\mathcal{H}$ that minimizes the expected risk $\hat{R}_z$ given a set of training samples. Vapnik Vapnik (2013) has developed the theory of statistical learning. In this theory several important results are provided: necessary and sufficient conditions for the consistency of learning processes and bounds for the rate of convergence, which uses the notion of VC-dimension. A new inductive principle, Structural Risk Minimization (SRM), and an algorithm, Support Vector Machine (SVM), that makes use of this notion to improve the learning process.

Nowadays learning approaches based on Deep Neural Networks, which are not focused on controlling the capacity of the set of hypothesis, outperform the SVM approaches in many problems. However, these popular Deep Learning approaches require large amounts of data to learn good hypothesis. It is commonly believed that machines need much more samples to learn than humans do. Vapnik Vapnik and Izmailov (2015); Vapnik and Vashist (2009) reflects on this belief and states that humans typically learn under the supervision of an Intelligent Teacher. This Teacher shares important knowledge by providing metaphors, examples or clarifications that are helpful for the students.

#### LUPI Paradigm

These additional knowledge provided by the Teacher is the Privileged Information that is available only during the training stage. To incorporate the concept of Intelligent Teacher in the Machine Learning framework, Vapnik introduces the paradigm of Learning Under Privileged Information (LUPI). In the LUPI paradigm describes the following model. Given a set of i.i.d. triplets

$$z = \{(x_1, x_1^*, y_1), \ldots, (x_n, x_n^*, y_n)\}\,,\ x \in \mathcal{X}, x^* \in \mathcal{X}^*, y \in \mathcal{Y}$$

generated according to an unknown distribution $P(x, x^*, y)$, the goal is to find the hypothesis $h(x, \alpha^*)$ from a set of hypothesis $\mathcal{H} = \{h(x, \alpha), \alpha \in A\}$ that minimizes some expected risk

$$R_F = \int \ell(h(x, \alpha), y)\, dF(x, y).$$

Note that the goal is the same that in the standard paradigm, however with the LUPI approach we are provided additional information, which is available only during the training stage. This additional information is encoded in the elements $x^*$ of a space $\mathcal{X}^*$, which is different from $\mathcal{X}$. The goal of the Teacher is, given a pair $(x_i, y_i)$, to provide a useful information $x^* \in \mathcal{X}^*$ given some probability $P(x^* \mid x)$. That is, the "intelligence" of the Teacher is defined by the choice of the space $\mathcal{X}^*$ and the conditional probability $P(x^* \mid x)$. To understand better this paradigm consider the following examples.

**Example 1.** Consider that the goal is to find a decision rule that classifies biopsy images into cancer or non-cancer. Here, $\mathcal{X}$ is the space of images, i.e. the matrix of pixels, for example $[0, 1]^{64\times 64}$. The label space is $\mathcal{Y} = \{0, 1\}$. An Intelligent Teacher might provide a student of medicine with commentaries about the images, for example: "There is an area of unusual concentration of cells of Type A." or "There is an aggresive proliferation of cells of Type B". These commentaries are the elements $x^*$ of certain space $X^*$ and the Teacher also chooses the probability $P(x^* \mid x)$.

**Example 2.** Consider the TODO

### Analysis of convergence rates

To get a better insight of how the Privileged Information can help in the Learning process, Vapnik provides a theoretical analysis of its influence on the learning rates. In the standard learning paradigm, how well the expected risk $R_F$ can be bounded is controlled by two factors: the empirical risk $\hat{R}_z$ and the VC-dimension of the set of hypothesis $\mathcal{H}$. In the case of classification, where $\mathcal{Y} = \{-1, 1\}$ and the loss $\ell(h(x, \alpha), y) = \mathbf{1}_{h(x,\alpha)y\leq 0}$, the risks can be expressed as

$$R_F(\alpha) = \int \mathbf{1}_{yh(x,\alpha)\leq 0} dF(x, y) = P(h(x, \alpha)\, y \leq 0),$$
$$\hat{R}_z(\alpha) = \sum_{i=1}^{n} \mathbf{1}_{y_i h(x_i,\alpha)\leq 0} = \nu(\alpha).$$

In (Vapnik, 1982, Theorem 6.8) the following bound for the rate of convergence is given with probability $1 - \eta$:

$$P(h(x, \alpha_n)\, y \leq 0) \leq \nu(\alpha_n) + O\left(\frac{d \log\left(\frac{2n}{d}\right) - \log \eta}{n} \sqrt{\nu(\alpha_n) \frac{n}{d \log\left(\frac{2n}{d}\right) - \log \eta}}\right).$$

That is, the bound is controlled by the ratio $d/n$, where $d$ is the VCdim$(\mathcal{H})$. If this $VC$-dimension is finite, the bound goes to zero as $n$ grows. However, two different cases can be considered.

**Separable case:** the training data can be classified in two groups without errors. That is, there exists $\alpha_n \in \eta$ such that $y_i h(x_i, \alpha_n) > 0$ for $i = 1, \ldots, n$, and thus $\nu(\alpha_n) = 0$.

In this case, the following bound for the rate of converge holds

$$P(h(x, \alpha_n) y \leq 0) \leq O\left(\frac{d \log\left(\frac{2n}{d}\right) - \log \eta}{n}\right).$$

**Non-Separable case:** the training data cannot be classified in two groups without errors. That is, for all $\alpha_n \in A$, there exists $i = 1, \ldots, n$, such that $y_i h(x_i, \alpha_n) \leq 0$, and thus $\nu(\alpha_n) > 0$. In this case, the following bound for the rate of converge holds

$$P(h(x, \alpha_n) y \leq 0) \leq \nu(\alpha_n) + O\left(\sqrt{\frac{d \log\left(\frac{2n}{d}\right) - \log \eta}{n}}\right).$$

Note that there is an important difference here in the rate of convergence. The separable case has a convergence rate of $d/n$, while the non-separable case has a rate of $\sqrt{d/n}$. Vapnik tries to address the question of why there exists such difference.

### Oracle SVM

Vapnik tries to answer these question by looking at Support Vector Machines. In the separable case, one has to minimize the functional

$$J(w) = \|w\|^2$$

subject to the constraints

$$y_i (w x_i + b) \geq 1.$$

However, in the non-separable case the functional to minimize is

$$J(w, \xi_1, \ldots, \xi_n) = \|w\|^2 + C \sum_{i=1}^{n} \xi_i$$

subject to the constraints

$$y_i (w x_i + b) \geq 1 - \xi_i.$$

That is, in the separable case $m$ parameters (of $w$) have to be estimated using $n$ examples, while in the non-separable case $m + n$ parameters (considering $w$ and the slack variables $\xi_1, \ldots, \xi_n$) have to be estimated with $n$ examples.

What would happen if the parameters $\xi_1, \ldots, \xi_n$ were known? In Vapnik and Izmailov (2015) an *Oracle* SVM is considered. Here, the learner (Student) is supplied with a set of triplets

$$(x_1, \xi_1^0, y_1), \ldots, (x_n, \xi_n^0, y_n)$$

where $\xi_1^0, \ldots, \xi_n^0$ are the slack variables for the best decision rule $h(x, \alpha_0) = \inf_{\alpha \in A} R_F(\alpha)$:

$$\xi_i^0 = \max(0, 1 - h(x, \alpha_0)), \ \forall i = 1, \ldots, n.$$

An *Oracle* SVM has to minimize the functional

$$J(w) = \|w\|^2$$

subject to the constraints

$$y_i \left( w x_i + b \right) \geq 1 - \xi_i^0.$$

Since the slack variables $\xi_i^0$ are known in advance, it can be shown Vapnik and Vashist (2009) that for the *Oracle* SVM the following bound holds

$$P(h\left(x, \alpha_n\right) y \leq 0) \leq P(1 - \xi^0 \leq 0) + O\left( \frac{d \log\left(\frac{2n}{d}\right) - \log \eta}{n} \right),$$

where $P(1 - \xi^0 \leq 0)$ is the probability error of the hypothesis $h\left(x, \alpha_0\right)$. That is, we recover the rate $d/n$.

**From Oracle to Intelligent Teacher**

The *Oracle* SVM is a theoretical construct, but we can approximate it by modelling the slack variables with the information provided by the Teacher in the LUPI paradigm. That is, the Teacher defines a space $\mathcal{X}^*$ and a set of functions $\{f^*(x, \alpha^*), \alpha^* \in A^*\}$. Then model the slack variables can be approximated as

$$\xi^* = f^*(x^*, \alpha^*).$$

From the pairs generated by some random generator in nature, the Teacher also defines the probability $P\left(x^* \mid x\right)$ to provide the triplets

$$(x_1, x_1^*, y_1), \ldots, (x_n, x_n^*, y_n).$$

Then, we can consider the problem where the goal is to minimize

$$J(\alpha, \alpha^*) = \sum_{i=1}^{n} \max(0, f^*(x_i^*, \alpha^*))$$

subject to the constraints

$$h\left(x_i, \alpha\right) \geq 1 - f^*(x_i^*, \alpha^*).$$

Let $f(x, \alpha_n), h(x.\alpha_n)$ that minimize this problem. Then, in (Vapnik and Vashist, 2009, Proposition 2) the following results for the bound of convergence is given

$$P(h\left(x, \alpha_n\right) y \leq 0) \leq P(1 - f^*(x^*, \alpha_n^*) \leq 0) + O\left( \frac{(d + d^*) \log\left(\frac{2n}{(d+d^*)}\right) - \log \eta}{n} \right),$$

where $d^*$ is the VC-dimension of the space of hypothesis $\{f(x, \alpha^*) \in A^*\}$. This result shows that, to maintain the best convergence rate $d/n$, we need to estimate the $P(1 - f^*(x^*, \alpha_n^*) \leq 0)$. Although this probability is unknown, we can control it. Considering

$$\alpha_0^* = \inf_{\alpha^* \in A^*} \int_{\mathcal{X}^*} \max(0, f^*(x^*, \alpha^*) - 1) dP(x^*)$$

and

$$\alpha_n^* = \inf_{\alpha^* \in A^*} \sum_{i=1}^{n} \max(0, f^*(x_i^*, \alpha^*) - 1).$$

Consider $\{f^*(x^*, \alpha^*), \alpha^* \in A^*\}$ such that $f^*(x^*, \alpha^*) < B, \alpha^* \in A^*$, then

$$\{\max(0, f^*(x^*, \alpha^*) - 1), \alpha^* \in A^*\}$$

is a set of totally bounded non-negative functions, then we have the standard bound Vapnik (2013)

$$P(1 - f^*(x^*, \alpha_0^*) \leq 0) \leq P(1 - f^*(x^*, \alpha_n^*) \leq 0) + O\left(\sqrt{\frac{d \log\left(\frac{2n}{d}\right) - \log \eta}{n}}\right).$$

with probability $1 - 2\eta$. That is, to have a rate of $d/n$ for $\alpha_n$, we need to estimate $\alpha_n^*$, which has a rate of $\sqrt{d^*/n}$. However, observe that $\mathcal{X}^*$ is the space suggested by the Teacher, which hopefully has a much lower capacity, and thus, the convergence will be faster in this space.

### SVM+

Vapnik describes an extension of the SVM that embodies the LUPI paradigm Vapnik and Izmailov (2015); Vapnik and Vashist (2009). Given a set of triplets

$$(x_1, x_1^*, y_1), \ldots, (x_n, x_n^*, y_n),$$

the idea is to model the slack variables of the standard SVM using the elements $x^* \in \mathcal{X}^*$ as

$$\xi(x^*, y) = [y(w^* \phi^*(x^*) + b^*)]_+ = \max\left(y(w^* \phi^*(x^*) + b^*), 0\right).$$

The minimization problem is the following:

$$\begin{aligned} \underset{w, w^*, b, b^*}{\arg\min} \quad & C \sum_{i=1}^{n} [y_i(\langle w^*, \phi^*(x_i^*)\rangle + b^*)]_+ + \frac{1}{2}\langle w, w\rangle + \frac{\mu}{2}\langle w^*, w^*\rangle \\ \text{s.t.} \quad & y_i(\langle w, \phi(x_i)\rangle + b) \geq 1 - [y_i(\langle w^*, \phi^*(x_i^*)\rangle + b^*)]_+ . \end{aligned} \tag{3.12}$$

Here $\phi$ and $\phi^*$ are two transformations that can be different. However, note that problem (3.12) is not convex due to the positive part term in the objective function. Vapnik et al. propose a relaxation of this problem to obtain a convex one. The idea is to model the slack variables $\xi$ as

$$\xi(x^*, y) = [y(w^* \phi^*(x^*) + b^*)] + \zeta(x^*, y),$$

where $\zeta(x^*, y) \geq 0$. The minimization problem is then

$$\begin{aligned} \underset{w, w^*, b, b^*, \zeta_i}{\arg\min} \quad & C \sum_{i=1}^{n} ([y_i(\langle w^*, \phi^*(x_i^*)\rangle + b^*)] + \zeta_i) + C\Delta \sum_{i=1}^{n} \zeta_i \\ & \quad + \frac{1}{2}\langle w, w\rangle + \frac{\mu}{2}\langle w^*, w^*\rangle \\ \text{s.t.} \quad & y_i(\langle w, \phi(x_i)\rangle + b) \geq 1 - [y_i(\langle w^*, \phi^*(x_i^*)\rangle + b^*) + \zeta_i], \\ & y_i(\langle w^*, \phi^*(x_i^*)\rangle + b^*) + \zeta_i \geq 0, \\ & \zeta_i \geq 0, \\ \text{for} \quad & i = 1, \ldots, n. \end{aligned} \tag{3.13}$$

Problem (3.13) is convex and the corresponding dual problem is

$$\begin{array}{lll} \underset{\alpha_i,\delta_i}{\arg\min} & \frac{1}{2}\sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j k(x_i,x_j) + \frac{1}{2\mu}\sum_{i,j=1}^{n} y_i y_j (\alpha_i-\delta_i)(\alpha_j-\delta_i) k^*(x_i^*,x_j^*) - \sum_{i=1}^{n}\alpha_i \\ \text{s.t.} & 0 \le \delta_i \le C \\ & 0 \le \alpha_i \le C + \delta_i, \\ & \sum_{i=1}^{n}\delta_i y_i = 0, \ \sum_{i=1}^{n}\alpha_i y_i = 0, \\ \text{for} & i = 1,\ldots,n. \end{array} \tag{3.14}$$

where we use the kernel functions

$$k(x_i,x_j) = \langle \phi(x_i), \phi(x_j)\rangle, \ k^*(x_i^*,x_j^*) = \langle \phi^*(x_i^*), \phi^*(x_j^*)\rangle.$$

We can observe in Problem (3.14) that the LUPI paradigm exerts a similarity control, correcting the similarity in space $\mathcal{X}$ with the similarity in the privileged space $\mathcal{X}^*$. For that reason, $\mathcal{X}$ and $\mathcal{X}^*$ are named Decision Space and Correction Space, respectively.

### Connection between SVM+ and MTLSVM

In Liang and Cherkassky (2008) the connection between SVM+ and Multi-Task Learning SVM (MTLSVM) is discussed. The MTLSVM proposed in Liang and Cherkassky (2008) is a Multi-Task Learning model based on the SVM. It solves the primal problem

$$\begin{array}{lll} \underset{w,b,v_r,b_r,\xi_i^r}{\arg\min} & C\sum_{r=1}^{T}\sum_{i=1}^{m_r}\xi_i^r + \frac{1}{2}\langle w,w\rangle + \sum_{r=1}^{T}\frac{\mu}{2}\langle v_r,v_r\rangle \\ \text{s.t.} & y_i^r(\langle w,\phi(x_i^r)\rangle + b + \langle v_r,\phi_r(x_i^r)\rangle + b_r) \ge 1-\xi_i^r, \\ & \xi_i^r \ge 0, \\ \text{for} & r = 1,\ldots,T; \ i = 1,\ldots,m_r. \end{array} \tag{3.15}$$

Here, a combination of a common model for all tasks

$$\langle w,\phi(x_i)\rangle + b$$

and a task-specific model

$$\langle v_r,\phi_r(x_i^r)\rangle + b_r$$

is used. Here, the common transformation $\phi$ and the task-independent ones $\phi_r$ can be different The dual problem corresponding to (3.15) is

$$\begin{aligned}
\arg\min_{\alpha_i} \quad & \frac{1}{2}\sum_{r,s=1}^{T}\sum_{i,j=1}^{m_r} y_i^r y_j^s \alpha_i^r \alpha_j^s k(x_i^r, x_j^s) + \frac{1}{2\mu}\sum_{r,s=1}^{T}\sum_{i,j=1}^{m_r} y_i^r y_j^s \alpha_i^r \alpha_j^s \delta_{rs} k_r(x_i^r, x_j^s) \\
& - \sum_{r=1}^{T}\sum_{i=1}^{m_r} \alpha_i^r \\
\text{s.t.} \quad & 0 \leq \alpha_i^r \leq C \\
& \sum_{i=1}^{m_r} \alpha_i^r y_i^r = 0, \\
\text{for} \quad & r = 1, \ldots, T;\ i = 1, \ldots, m_r.
\end{aligned} \tag{3.16}$$

In Liang and Cherkassky (2008) some similarities between MTLSVM and SVM+ are pointed out. Problem (3.15) can be regarded as an adaptation of (3.13) to solve MTL problems, where the $\mathcal{X} = \mathcal{X}^*$ but different tasks are incorporated, which is reflected on the different transformations $\phi_r$. If we consider the problem (3.15) with a single task, it is a modification of SVM+ where the slack variables are modeled as

$$\xi(x^*, y) = y(w^* \phi^*(x^*) + b^*)$$

and $x^* = x$. That is, it is a relaxation of the original problem (3.12) where the positive part is ignored. This relaxation gives place to some important differences between both models. Since the auxiliary primal variables $\zeta_i$ are no longer required, this is reflected in a simpler dual form (3.16), where only $n$ dual variables have to be estimated, instead of the $2n$ dual variables of (3.14). The Multi-Task element, is reflected on (3.16) through the $\delta_{rs}$ function, which makes the correction of similarity only possible between elements of the same task. A major remark can be make about the differences between MTLSVM and SVM+. The results for the improved rate of convergence with an Intelligent Teacher may not be valid with MTLSVM, since we are no modelling the slack variables $\xi$ adequately. It is still a work in progress to study the rate of convergence of MTLSVM and to establish more clear links with SVM+.

## 3.3 Multi-Task Learning Methods: An Overview

### 3.3.1 Feature Learning

Feature Learning or Feature Transformation is one of the most popular approaches to MTL.

### 3.3.2 Joint Learning

The Joint Learning methods for MTL can be divided into the frequentist and the Bayesian approaches. The frequentist approaches uses a combination of task-specific models and models that are common to all tasks. These two models are learned simultaneously with the goal of leveraging the common and specific information. In the Bayesian approaches common prior is shared for all the tasks models, and the diverse sources of data definetele different posterior distributions for each task.

The first proposal of the frequentist approach, which uses the SVM as the foundation, is found in Evgeniou and Pontil (2004) where the *regularized MTL* SVM is presented. The goal is to find a decision function for each task, each being defined by a vector

$$w_r = w + v_r,$$

where $w$ is common to all tasks and $v_r$ is task-specific. The primal problem of *regularized MTL* SVM, using the unified formulation, is

$$\begin{aligned} \underset{w,v_r,\xi_i^r}{\arg\min} \quad & C\sum_{r=1}^{T}\sum_{i=1}^{m_r}\xi_i^r + \frac{1}{2}\langle w, w\rangle + \sum_{r=1}^{T}\frac{\mu}{2}\langle v_r, v_r\rangle \\ \text{s.t.} \quad & y_i^r(\langle w, x_i^r\rangle + \langle v_r, x_i^r\rangle) \geq p_i^r - \xi_i^r, \\ & \xi_i^r \geq 0, \\ \text{for} \quad & r = 1,\ldots,T;\ i = 1,\ldots,m_r. \end{aligned} \tag{3.17}$$

Note that $\mu$ is a parameter that controls the tradeoff between the relevance of common and specific models. That is, when $\mu$ tends to infinite, the resulting model approaches a common-task standard SVM; when $\mu$ tends to zero, a independent task approach is taken, with one standard SVM problem for each task. This is also reflected in the corresponding dual problem

$$\begin{aligned} \underset{\alpha_i}{\arg\min} \quad & \frac{1}{2}\sum_{r,s=1}^{T}\sum_{i,j=1}^{m_r} y_i^r y_j^s \alpha_i^r \alpha_j^s \langle x_i^r, x_j^s\rangle + \frac{1}{2\mu}\sum_{r,s=1}^{T}\sum_{i,j=1}^{m_r} y_i^r y_j^s \alpha_i^r \alpha_j^s \delta_{rs}\langle x_i^r, x_j^s\rangle \\ & - \sum_{r=1}^{T}\sum_{i=1}^{m_r} p_i^r \alpha_i^r \\ \text{s.t.} \quad & 0 \leq \alpha_i^r \leq C \\ \text{for} \quad & r = 1,\ldots,T;\ i = 1,\ldots,m_r. \end{aligned} \tag{3.18}$$

In this dual form, as $\mu$ grows, the task-specific part goes to zero, and the most important term is the first one, corresponding to the common part. The opposite effect is obtained when $\mu$ shrinks. Moreover, in Evgeniou and Pontil (2004) it is shown that solving (3.17) is equivalent to solving the problem

$$\begin{aligned} \underset{ww_r,\xi_i^r}{\arg\min} \quad & C\sum_{r=1}^{T}\sum_{i=1}^{m_r}\xi_i^r + \frac{1}{2}\sum_{r=1}^{T}\|w_r\|^2 + \frac{\mu}{2}\sum_{r=1}^{T}\left\|w_r - \sum_{s=1}^{T} w_s\right\|^2 \\ \text{s.t.} \quad & y_i^r(\langle w_r, x_i^r\rangle) \geq p_i^r - \xi_i^r, \\ & \xi_i^r \geq 0, \\ \text{for} \quad & r = 1,\ldots,T;\ i = 1,\ldots,m_r. \end{aligned}$$

Now, only the $w_r$ variables are included, and it is clearer that $\mu$ penalizes the variance of the $w_r$ vectors, so all models $w_r$ will tend to a common model as $\mu$ grows.

Multiple extensions of the work of Evgeniou and Pontil (2004) have been presented: in Li et al. (2015); Xu et al. (2014) the method is extended to the Proximal SVM Fung and Mangasarian (2001) and Least Squares SVM Suykens and Vandewalle (1999), respectively. Also, in Parameswaran and Weinberger (2010) the idea is adapted for the Large Margin Nearest Neighbor model Weinberger and Saul (2009). However, in this work we are

interested mainly in two extensions: one is the work of Evgeniou et al. (2005), pivotal for this thesis, which will be described in SectionTODO; the other relevant extension is developed in Liang and Cherkassky (2008), which has already been discussed in the previous section. The multi-task problem described in Liang and Cherkassky (2008) for classification, is also adapted for regression problems in Cai and Cherkassky (2009). Using the unified notation we can express the primal problem as

$$\begin{aligned} \underset{w,b,v_r,b_r,\xi_i^r}{\arg\min} \quad & C\sum_{r=1}^{T}\sum_{i=1}^{m_r}\xi_i^r + \frac{1}{2}\langle w,w\rangle + \sum_{r=1}^{T}\frac{\mu}{2}\langle v_r,v_r\rangle \\ \text{s.t.} \quad & y_i(\langle w,\phi(x_i^r)\rangle + b + \langle v_r,\phi_r(x_i^r)\rangle + b_r) \geq p_i^r - \xi_i^r, \\ & \xi_i^r \geq 0, \\ \text{for} \quad & r = 1,\ldots,T;\ i = 1,\ldots,m_r. \end{aligned} \tag{3.19}$$

Comparing (3.17) and (3.19) we observe that the subyacent idea is the same, but there exists some differences. In first place, note that (3.17) is described as a linear model, while in (3.19) not only non-linear transformations of the data are used, but different transformations can be selected $\phi, \phi_r$ for the common part and for each task-specific term, respectively. Moreover, it is relevant to note the incorporation of the bias terms in (3.19). The dual form of (3.19) is

$$\begin{aligned} \underset{\alpha_i}{\arg\min} \quad & \frac{1}{2}\sum_{r,s=1}^{T}\sum_{i,j=1}^{m_r} y_i^r y_j^s \alpha_i^r \alpha_j^s \left[k(x_i^r,x_j^s) + \delta_{rs}k_r(x_i^r,x_j^s)\right] - \sum_{r=1}^{T}\sum_{i=1}^{m_r} p_i^r \alpha_i^r \\ \text{s.t.} \quad & 0 \leq \alpha_i^r \leq C \\ & \sum_{i=1}^{m_r} \alpha_i^r y_i^r = 0, \\ \text{for} \quad & r = 1,\ldots,T;\ i = 1,\ldots,m_r. \end{aligned} \tag{3.20}$$

In (3.20) an equality constrain, not present in (3.18) is added. This a direct consequence of the incorporation of bias terms in the primal form. Also, it is important to observe the use of different kernel spaces through the functions $k$ and $k_r$. This has connections with the LUPI paradigm Vapnik and Izmailov (2015), as described in the previous section. The kernel space for the common part is named the decision space, and the spaces corresponding to the kernel functions $k_r$ are the correcting spaces. That is, each task can correct the similarity defined by the decision space independently.

In the Bayesian approaches, the first work of Lawrence and Platt (2004) presents a GP model where all the tasks share a common prior. That is, given $T$ tasks, a noise model with latent variables $\boldsymbol{f}^r$ is considered, i.e. $y_i^r = f_i^r + \epsilon$, and $\boldsymbol{f}^r$ follows a GP prior

$$p(\boldsymbol{f}^r|X,\boldsymbol{\theta}) = N(0, K_{\boldsymbol{\theta}})$$

where $K$ is a kernel matrix parametrized by $\boldsymbol{\theta}$ and evaluated at the points $X$. Note that a single $\boldsymbol{\theta}$ parameter is used to model a prior shared for all tasks. The posterior probability can be expressed as

$$p(\boldsymbol{y}^1,\ldots,\boldsymbol{y}^T|\boldsymbol{f}^1,\ldots,\boldsymbol{f}^T,\boldsymbol{\theta},X^1,\ldots,X^T) \propto p(\boldsymbol{f}^1,\ldots,\boldsymbol{f}^T,\boldsymbol{\theta}|X^1,\ldots,X^T)\prod_{r=1}^{T} p(\boldsymbol{y}^r|F^r,\boldsymbol{\theta})$$

where the distribution for the latent parameters is

$$p(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^T, \boldsymbol{\theta}|X^1, \ldots, X^T) \propto \prod_{r=1}^{T} p(\boldsymbol{f}^r|X^r, \boldsymbol{\theta}).$$

Although this idea is interesting for MTL, it presents a rigid framework since we use a fixed model for the prior $p(\boldsymbol{f}^r|X, \boldsymbol{\theta})$. To use Bayesian induction over the prior too, the hierarchical Bayesian model is considered. That is, we consider a different prior $p(\boldsymbol{f}^r|X, \boldsymbol{\theta}^r)$ for each task and a hyperprior for $\boldsymbol{\theta}^r$, $p(\boldsymbol{\theta}^r|\boldsymbol{\phi})$. Then, the distribution for latent parameters is expressed as

$$p(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^T, \boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^T, \boldsymbol{\phi}|X^1, \ldots, X^T) \propto \prod_{r=1}^{T} p(\boldsymbol{f}^r|X^r, \boldsymbol{\theta}^r, \boldsymbol{\phi}) p(\boldsymbol{\theta}^r|\boldsymbol{\phi}).$$

In Yu et al. (2005), a Gaussian hyperprior $p(\boldsymbol{\theta}^r|\boldsymbol{\phi})$ is considered. Note that in this formulation, each task parameter $\boldsymbol{\theta}^r$ is an independent from the rest of parameters $\boldsymbol{\theta}^s, s \neq r$ given $\boldsymbol{phi}$ . That is,

$$p(\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^T|\boldsymbol{\phi}) = \prod_{r=1}^{T} p(\boldsymbol{\theta}^r|\boldsymbol{\phi}).$$

Then, in Xue et al. (2007) a Dirichlet Process Prior is considered for modelling the task parameters. A explicit dependence is then defined over the task parameters

$$p(\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^T|\boldsymbol{\phi}) = \prod_{r=1}^{T} p(\boldsymbol{\theta}^r|\boldsymbol{\theta}^{-r}, \boldsymbol{\phi}).$$

where $\boldsymbol{\theta}^{-r} = \{\boldsymbol{\theta}^s, s \neq r\}$. This formulation converts this model in a task-clustering approach, where the clusters are learned jointly with the task parameters $\boldsymbol{\theta}$. Following this approach of hierarchical Bayes, III (2009) uses a prior for the task parameters $\boldsymbol{\theta}^r$ that learns backwards a genealogy tree. That is, beginning at the leafs, which are the task parameters $\boldsymbol{\theta}^r$, the branches merge until a common root to all the tasks. Thus, by selecting different thresholds or levels of this tree, we can obtain different clusters.

### 3.3.3 Low-Rank

### 3.3.4 Tasks Relations Learning

### 3.3.5 Decomposition

## 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.

Ben-David, S. and Borbely, R. S. (2008). A notion of task relatedness yielding provable multiple-task learning guarantees. *Mach. Learn.*, 73(3):273–287.

Cai, F. and Cherkassky, V. (2009). SVM+ regression and multi-task learning. In *International Joint Conference on Neural Networks, IJCNN 2009, Atlanta, Georgia, USA, 14-19 June 2009*, pages 418–424. IEEE Computer Society.

Evgeniou, T., Micchelli, C. A., and Pontil, M. (2005). Learning multiple tasks with kernel methods. *J. Mach. Learn. Res.*, 6:615–637.

Evgeniou, T. and Pontil, M. (2004). Regularized multi–task learning. In Kim, W., Kohavi, R., Gehrke, J., and DuMouchel, W., editors, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004*, pages 109–117. ACM.

Fung, G. and Mangasarian, O. L. (2001). Proximal support vector machine classifiers. In Lee, D., Schkolnick, M., Provost, F. J., and Srikant, R., editors, *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, CA, USA, August 26-29, 2001*, pages 77–86. ACM.

III, H. D. (2009). Bayesian multitask learning with latent hierarchies. In Bilmes, J. A. and Ng, A. Y., editors, *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*, pages 135–142. AUAI Press.

Lawrence, N. D. and Platt, J. C. (2004). Learning to learn with the informative vector machine. In Brodley, C. E., editor, *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, volume 69 of *ACM International Conference Proceeding Series*. ACM.

Li, Y., Tian, X., Song, M., and Tao, D. (2015). Multi-task proximal support vector machine. *Pattern Recognit.*, 48(10):3249–3257.

Liang, L. and Cherkassky, V. (2008). Connection between SVM+ and multi-task learning. In *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2008, part of the IEEE World Congress on Computational Intelligence, WCCI 2008, Hong Kong, China, June 1-6, 2008*, pages 2048–2054. IEEE.

Parameswaran, S. and Weinberger, K. Q. (2010). Large margin multi-task metric learning. In Lafferty, J. D., Williams, C. K. I., Shawe-Taylor, J., Zemel, R. S., and Culotta, A., editors, *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, pages 1867–1875. Curran Associates, Inc.

Suykens, J. A. K. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Process. Lett.*, 9(3):293–300.

Vapnik, V. (1982). *Estimation of dependences based on empirical data*. Springer Science & Business Media.

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

Vapnik, V. and Izmailov, R. (2015). Learning using privileged information: similarity control and knowledge transfer. *J. Mach. Learn. Res.*, 16:2023–2049.

Vapnik, V. and Vashist, A. (2009). A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5-6):544–557.

Weinberger, K. Q. and Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, 10:207–244.

Xu, S., An, X., Qiao, X., and Zhu, L. (2014). Multi-task least-squares support vector machines. *Multim. Tools Appl.*, 71(2):699–715.

Xue, Y., Liao, X., Carin, L., and Krishnapuram, B. (2007). Multi-task learning for classification with dirichlet process priors. *J. Mach. Learn. Res.*, 8:35–63.

Yu, K., Tresp, V., and Schwaighofer, A. (2005). Learning gaussian processes from multiple tasks. In Raedt, L. D. and Wrobel, S., editors, *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, volume 119 of *ACM International Conference Proceeding Series*, pages 1012–1019. ACM.