

# Semi-supervised Learning for Affective Common-Sense Reasoning

Luca Oneto<sup>1</sup> · Federica Bisio<sup>2</sup> · Erik Cambria<sup>3</sup>  · Davide Anguita<sup>1</sup>

Received: 11 July 2016 / Accepted: 11 October 2016 / Published online: 24 October 2016  
© Springer Science+Business Media New York 2016

## Abstract

**Background** Big social data analysis is the area of research focusing on collecting, examining, and processing large multi-modal and multi-source datasets in order to discover patterns/correlations and extract information from the Social Web. This is usually accomplished through the use of supervised and unsupervised machine learning algorithms that learn from the available data. However, these are usually highly computationally expensive, either in the training or in the prediction phase, as they are often not able to handle current data volumes. Parallel approaches have been proposed in order to boost processing speeds, but this clearly requires technologies that support distributed computations.

**Methods** Extreme learning machines (ELMs) are an emerging learning paradigm, presenting an efficient unified solution to generalized feed-forward neural networks. ELM offers significant advantages such as fast learning speed, ease of implementation, and minimal human intervention.

However, ELM cannot be easily parallelized, due to the presence of a pseudo-inverse calculation. Therefore, this paper aims to find a reliable method to realize a parallel implementation of ELM that can be applied to large datasets typical of Big Data problems with the employment of the most recent technology for parallel in-memory computation, i.e., Spark, designed to efficiently deal with iterative procedures that recursively perform operations over the same data. Moreover, this paper shows how to take advantage of the most recent advances in statistical learning theory (SLT) in order to address the issue of selecting ELM hyperparameters that give the best generalization performance. This involves assessing the performance of such algorithms (i.e., resampling methods and in-sample methods) by exploiting the most recent results in SLT and adapting them to the Big Data framework. The proposed approach has been tested on two affective analogical reasoning datasets. Affective analogical reasoning can be defined as the intrinsically human capacity to interpret the cognitive and affective information associated with natural language. In particular, we employed two benchmarks, each one composed by 21,743 common-sense concepts; each concept is represented according to two models of a semantic network in which common-sense concepts are linked to a hierarchy of affective domain labels.

**Results** The labeled data have been split into two sets: The first 20,000 samples have been used for building the model with the ELM with the different SLT strategies, while the rest of the labeled samples, numbering 1743, have been kept apart as reference set in order to test the performance of the learned model. The splitting process has been repeated 30 times in order to obtain statistically relevant results. We ran the experiments through the use of the Google Cloud Platform, in particular, the Google Compute

---

✉ Erik Cambria  
cambria@ntu.edu.sg

Luca Oneto  
luca.oneto@unige.it

Federica Bisio  
federica.bisio@aizoongroup.com

Davide Anguita  
davide.anguita@unige.it

<sup>1</sup> DIBRIS - University of Genova, Via Opera Pia 13, 16145 Genoa, Italy

<sup>2</sup> aizoOn S.r.l., Strada del Lionetto 6, 10146 Turin, Italy

<sup>3</sup> School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore

Engine. We employed the Google Compute Engine Platform with  $NM = 4$  machines with two cores and 1.8 GB of RAM (machine type n1-highcpu-2) and an HDD of 30 GB equipped with Spark. Results on the affective dataset both show the effectiveness of the proposed parallel approach and underline the most suitable SLT strategies for the specific Big Data problem.

**Conclusion** In this paper we showed how to build an ELM model with a novel scalable approach and to carefully assess the performance, with the use of the most recent results from SLT, for a sentiment analysis problem. Thanks to recent technologies and methods, the computational requirements of these methods have been improved to allow for the scaling to large datasets, which are typical of Big Data applications.

**Keywords** Analogical reasoning · Sentiment analysis · Semi-supervised learning · Classification · Model selection · Extreme learning machines · Vapnik–Chervonenkis theory · Rademacher complexity · Algorithmic stability

## Introduction

The advent of social networks, web communities, blogs, Wikipedia, and other online collaborative media has deeply changed the ways people express their opinions and sentiments. A growing amount of content and ideas are continuously expressed by the millions of people connected to the World Wide Web. As a major consequence, the distillation of knowledge from this huge quantity of unstructured information can be a key tool for marketers who want to create a brand or product image and identity in the minds of their customers. Such a scenario has led to the emerging fields of opinion mining and sentiment analysis [1–4], which deal with information retrieval and knowledge discovery from text using data mining and natural language processing (NLP) techniques [5–8]. However, mining opinions and sentiments from natural language is an extremely difficult task as it involves a deep and broad understanding of the explicit and implicit, regular and irregular, syntactical and semantic rules proper of a language.

Sentic computing [9] tackles these crucial issues by exploiting affective common-sense reasoning, i.e., the intrinsically human capacity to interpret cognitive and affective information associated with natural language, and thus differs from standard statistical approaches to big social data analysis. Common-sense computing techniques are applied in different contexts (including multi-modality [10], handwriting recognition [11], e-health [12], and more) to bridge the semantic gap between word-level

natural language data and the concept-level opinions conveyed by these. To achieve this goal, the sentic computing framework takes advantage of AffectNet [13], a semantic network in which common-sense concepts are linked to a hierarchy of affective domain labels. In particular, the vector space representation of one such semantic network, termed AffectiveSpace [14], enables affective analogical reasoning on natural language concepts.

Current research shows that the emerging field of big social data analysis [15–18] can take advantage of inductive learning systems to support emotion recognition in natural language text. In this context, every common-sense concept is represented according to AffectiveSpace and defined by four affective dimensions [19]: Pleasantness, Attention, Sensitivity, and Aptitude. This representation leads to a further polarity detection task. The current emotion recognition problem is complicated by the fact that labeling all the common-sense concepts of AffectiveSpace is often difficult, expensive, and time-consuming. Therefore, affective dimensions labeling is only available for a set of concepts. The need to properly tackle these issues leads to the use of a semi-supervised classifier. Eventually, a semi-supervised version of the extreme learning machine (ELM) framework [20–23] is adopted.

The interest in semi-supervised learning [4, 24–26] has recently increased, especially because application domains exist (e.g., text mining, natural language processing, image and video retrieval, bioinformatics). In this context, semi-supervised learning can be formalized as a supervised learning problem biased by an unsupervised reference solution. First, a general biased-regularization scheme that encompasses the biased version of ELM is introduced. Then, a semi-supervised learning model based on the biased-regularization [27] scheme follows a two-step procedure. In the first step, an unsupervised clustering of the whole dataset (including both labeled and unlabeled data) obtains a reference solution. As a result, the eventual semi-supervised classification framework can derive a reference function from any clustering algorithm, thus providing remarkable flexibility. In the second step, the clustering outcomes drive the learning process in a biased-regularization ELM to acquire the class information provided by labels. The ultimate result is that the overall learned function exploits both labeled and unlabeled data. The integrated framework applies to both linear and nonlinear data distributions: In the former, one works under a cluster assumption on data; in the latter, one works under a manifold hypothesis [28].

In this paper, we want to address the problem of assessing the performance of a predictive model, i.e., the semi-supervised version of ELM, and quantify its uncertainty. Similar problems have been addressed in the field of

statistical inference since the last century [29]. The now classic approach of parametric statistics identifies a family of models (e.g., linear functions) and a noise assumption (e.g., Gaussian). Given some data, it easily provides an assessment of the performance of the fitted model, along with a quantification of the uncertainty or, in modern terms, an estimation of the generalization error and the related confidence interval<sup>1</sup>. On the contrary, data-driven models exploit nonparametric inference, where it is expected that an effective model would stem out directly from the data, without any assumption on the model family nor any other information that is external to the dataset itself [31, 32].

Statistical learning theory (SLT) tries to find necessary and sufficient conditions for nonparametric inference to build predictive models from data [33–35] or, using the language of SLT, learn an optimal model from data. The main SLT results have been obtained by deriving nonasymptotic bounds on the generalization error of a model or, to be more precise, upper bounds on the excess risk between the optimal predictor and the learned model, as a function of the possibly infinite family of models and the number of available samples [36]. For a long time, SLT has been considered only a theoretical, albeit very sound and deep, statistical framework, without any real applicability to practical problems [37]. It was only in the last decade, after important advances in this field [38], that SLT has been shown to be able to provide practical answers, at least when targeting the inference of data-driven models for classification purposes [39, 40].

This paper shows how to exploit unlabeled samples in the usual semi-supervised learning context so as to tune and assess the performance of a learning algorithm, with particular reference to the ELM applied to an affective analogical reasoning problem. We review all most recent methodologies of model selection (MS) and error estimation (EE) that can be applied to the ELM, as well as how these methodologies can take advantage of unlabeled samples. In brief, among the several methods proposed in the literature for MS and EE, we identify two main categories: out-of-sample and in-sample methods [40]. The first works well in many situations and allows the application of simple statistical techniques in order to estimate quantities of interest by splitting data in different sets, each for a different purpose (training, validation, and test). Some examples of out-of-sample methods are the well-known holdout (HO) and k-fold cross-validation (KCV) [41], leave-one-out (LOO), and bootstrap [42]. In contrast, the in-sample methods exploit the whole set of available data

for training the model, assessing its performance and estimating its generalization error, thanks to the application of rigorous statistical procedures. We describe how in-sample methods can be further divided into two subgroups: the hypothesis space-based methods and the algorithm-based methods [43]. The first subgroup requires knowledge of the hypothesis space from which the algorithm chooses the model. Some examples of these methods are the Vapnik–Chervonenkis (VC) theory [36, 44], (local) Rademacher complexity (RC) theory [38, 45–48], and PAC Bayes theory [39, 49–52]. The second subgroup of methods does not require advance knowledge of the hypothesis space, instead just relying on application of the algorithm on a series of modified training sets. Some examples are the compression bound [53, 54] and algorithmic stability theory (AS) [43, 55, 56]. We also mention the distinctions between the frequentist and Bayesian approaches, although some approaches combine aspects from these two [57].

## Semi-supervised Binary Classification

Before getting into the discussion proper, let us recall some common preliminary definitions [36, 55]. Let us consider a set of labeled samples  $\mathcal{D}_n = \{(x_1, y_1), \dots, (x_n, y_n)\} = \{z_1, \dots, z_n\}$  and another set of unlabeled ones  $\mathcal{D}_{n_u} = \{x_{n+1}, \dots, x_{n+n_u}\}$  drawn i.i.d., according to an unknown probability distribution  $\mu$  over the cartesian product between the input space  $\mathcal{X} \subseteq \mathbb{R}^d$  and the output space  $\mathcal{Y} = \{-1, +1\}$  defined as  $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ . Let us also consider a function  $f \in \mathcal{F}$  where  $f: \mathcal{X} \rightarrow \overline{\mathcal{Y}} = \mathbb{R}$ . The error of  $f$  in approximating  $\mu$  is measured with reference to some  $[0, 1]$ -bounded loss function  $\ell: \mathcal{F} \times \mathcal{Z} \rightarrow [0, 1]$ . The risk of  $f$  can be then defined as such:

$$L(f) = \mathbb{E}_z \{\ell(f, z)\}. \quad (1)$$

Since  $\mu$  is unknown,  $L(f)$  cannot be computed though we can compute its empirical estimators. Before defining them, let us introduce two modified training sets  $\mathcal{D}_n^i$  and  $\mathcal{D}_n^i$ , where the  $i$ th element is respectively removed or replaced by another sample  $z'_i$  sampled i.i.d. from  $\mu$ :

$$\begin{aligned} \mathcal{D}_n^{\setminus i} &: \{z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_n\}, \\ \mathcal{D}_n^i &: \{z_1, \dots, z_{i-1}, z'_i, z_{i+1}, \dots, z_n\}. \end{aligned} \quad (2)$$

If  $\hat{f} = \mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}$ ,  $\hat{f}^{\setminus i} = \mathcal{A}_{(\mathcal{D}_n^{\setminus i} \cup \mathcal{D}_{n_u}, \mathcal{H})}$ , and  $\hat{f}^i = \mathcal{A}_{(\mathcal{D}_n^i \cup \mathcal{D}_{n_u}, \mathcal{H})}$  are chosen form of functions  $\mathcal{F}_{\mathcal{H}}$  according to some criteria, or algorithm,  $\mathcal{A}_{\mathcal{H}}$ , where  $\mathcal{H}$  is a set of hyperparameters that must be tuned, and based respectively on  $\mathcal{D}_n \cup \mathcal{D}_{n_u}$ ,  $\mathcal{D}_n^{\setminus i} \cup \mathcal{D}_{n_u}$ , and  $\mathcal{D}_n^i \cup \mathcal{D}_{n_u}$  ( $\mathcal{D}_{n_u}$  can be exploited or not based on  $\mathcal{A}$ ), we can define the empirical

<sup>1</sup> In this paper, we deal with a frequentist approach, which derives confidence intervals for quantities of interest, but the credible intervals of the Bayesian approach can be addressed equally well in the parametric setting [30].

risk of a function  $f \in \mathcal{F}_{\mathcal{H}}$  and the LOO risk of the algorithm  $\mathcal{A}$  [55] as:

$$\begin{aligned} \widehat{L}_{\text{emp}}(f) &= \frac{1}{n} \sum_{i=1}^n \ell(f, z_i), \\ f \in \mathcal{F}, \widehat{L}_{\text{loo}}(\mathcal{A}_{\mathcal{H}}) &= \frac{1}{n} \sum_{i=1}^n \ell(\mathcal{A}_{(\mathcal{D}_n^i \cup \mathcal{D}_{nu}, \mathcal{H})}, z_i). \end{aligned} \tag{3}$$

If the loss function is not specified, all the results hold for any  $[0, 1]$ -bounded loss function. Instead, some results just hold for particular loss functions; in such cases, we specify which loss function must be adopted as  $L^\ell(f)$ ,  $\widehat{L}_{\text{emp}}^\ell(f)$ ,  $\widehat{L}_{\text{loo}}^\ell(\mathcal{A}_{\mathcal{H}})$ , etc.

### Semi-supervised Extreme Learning Machines

The biased model adopted as a semi-supervised approach exploits both unlabeled and labeled data to learn a classification function empirically. The model is based on the biased-regularization theory, defined as follows: A reference solution (e.g., a hyperplane) is used to bias the solution of a regularization-based learning machine.

### Extreme Learning Machines

The ELM model [20, 58–60] implements a single-hidden layer feed-forward neural network (SLFN) with  $N_h$  mapping neurons. Thus, the hypothesis space can be formalized as follows

$$f(x) = \sum_{j=1}^{N_h} w_j a_j(x, \zeta_j), \tag{4}$$

where  $w_j \in \mathbb{R}$ , and  $a(x, \zeta)$  is a nonlinear piecewise continuous function satisfying ELM universal approximation capability theorems [20]. In general, the activation function is characterized by a set of parameters  $\zeta$ , and the  $j$ th neuron has its  $\zeta_j$ . Sigmoid function, RBF, and polynomial functions represent three popular choices for the activation function.

The peculiar aspect of ELM is that the parameters  $\zeta_j$  are set randomly. Accordingly, the hidden layer implements an explicit mapping of the original input space  $\mathcal{X}$  into a new space  $\mathbb{R}^{N_h}$ . Hence, training ELMs is equivalent to solving a regularized least squares (RLS) problem in a linear space [20]. Let  $H \in \mathbb{R}^{n \times N_h}$  be an activation matrix such that the entry  $H_{ij}$  is the activation value of the  $j$ th hidden neuron for the  $i$ th input pattern. Then, the training problem reduces to the minimization of the convex cost:

$$w^* = \arg \min_w \|Hw - y\|^2. \tag{5}$$

A matrix pseudo-inversion yields the unique  $L_2$  solution, as proven in [20]:

$$w^* = H^+ y. \tag{6}$$

Furthermore, the theory derived in [61] proves that regularization strategies can further improve the generalization performance of the model. As a result, the cost function of Eq. (5) is augmented by a  $L_2$  regularization factor as follows:

$$w^* = \arg \min_w \|Hw - y\|^2 + \lambda \|w\|^2. \tag{7}$$

The vector of weights  $w^*$  is then obtained as follows:

$$w^* = (H^T H + \lambda I)^{-1} H^T y, \tag{8}$$

where  $I \in \mathbb{R}^{N_h \times N_h}$  is an identity matrix.

### A Biased Regularization

The general biased-regularization model works via biasing the solution of a regularization-based learning machine by a reference function (e.g., a hyperplane) [62]. The nature of this reference function is a crucial aspect that concerns the learning theory in general. In the linear domain one can define a generic convex loss function  $\ell$ , and a biased-regularizing term, with the resulting cost function being

$$\ell + \lambda_1 \|w - \lambda_2 w_0\|^2, \tag{9}$$

where  $w_0$  is a reference hyperplane,  $\lambda_1$  is the classical regularization parameter that controls smoothness, and  $\lambda_2$  controls the adherence to the reference solution  $w_0$ . The expression of Eq. (9) is a convex functional and thus admits a global solution. From Eq. (9) one obtains:

$$\begin{aligned} \arg \min_w \|Hw - y\|^2 + \lambda_1 \|w - \lambda_2 w_0\|^2 \\ = \arg \min_w \|Hw - y\|^2 + \frac{\lambda_1}{2} \|w\|^2 - \lambda_1 \lambda_2 w w_0. \end{aligned} \tag{10}$$

The role played by parameter  $\lambda_2$  is indeed critical from both the theoretical and the practical point of view [62]. This parameter allows the cost function (10) to exploit a strong or weak bias on the hypothesis space; i.e., by adjusting  $\lambda_2$  one can modulate the contribution provided by  $w_0$  to the cost function. Hence, one can take advantage of biased regularization even when the reference solution is not optimal. The crucial aspect is the ability to shrink the space to be explored in order to get an optimal solution, which in turn means the ability to reduce the complexity of the hypothesis space [62].

Note that given a reference hyperplane  $w_0$ , a regularization constant  $\lambda_1$ , and a biasing constant  $\lambda_2$ , the problem:

$$w^* = \arg \min_w \|Hw - y\|^2 + \lambda_1 \|w - \lambda_2 w_0\|^2, \quad (11)$$

has solution:

$$w^* = (H^T H + \lambda_1 I)^{-1} (H^T y + \lambda_1 \lambda_2 w_0). \quad (12)$$

The proof is trivial [62]. Note, also, that thanks to the representer theorem [63, 64] it is possible to write  $w^*$  as:

$$w^* = \sum_{i=1}^n \alpha_i \phi(x_i) = H^T \alpha. \quad (13)$$

### A Semi-supervised Learning Scheme Based on Biased Regularization

The biased version of the ELM can support a semi-supervised framework for the classification task [62]. Let  $H$  denote the activation matrix of the whole dataset  $\mathcal{D}_n \cup \mathcal{D}_{n_u}$ ,  $H_n$  denote the activation matrix of  $\mathcal{D}_n$ ,  $y_n$  denote the corresponding vector of labels, and  $H_{n_u}$  denote the activation matrix of  $\mathcal{D}_{n_u}$ . The semi-supervised learning scheme then requires one to apply the following four-step procedure:

1. Clustering: Use any clustering algorithm to perform an unsupervised partition of the dataset  $\mathcal{D}_n \cup \mathcal{D}_{n_u}$  by discarding the available labels (a bipartition in the simplest case).
2. Calibration: For every cluster, a majority voting scheme is adopted to set the cluster label; this is done by exploiting the labeled samples. Then, for each cluster, assign to each sample the cluster label. Let  $\hat{y}$  denote this new set of labels.
3. Mapping: Given  $\mathcal{D}_n \cup \mathcal{D}_{n_u}$  and  $\hat{y}$ , train the selected learning machine and obtain the solution  $w_0$ .
4. Biasing: Given  $\mathcal{D}_n$ , train the biased version of the learning machine (biased by  $w_0$ ). The solution  $w$  carries information derived from both the labeled data  $\mathcal{D}_n$  and the unlabeled data  $\mathcal{D}_{n_u}$ .

This procedure, Step 4, Biasing, has similarities to that adopted in deep learning architectures [65, 66]. In the latter case, the training algorithm performs a preliminary unsupervised stage and then uses labels only to adjust the network for the specific classification task; the eventual representation still mostly reflects the outcome of the learning process completed in the pre-training phase. Likewise, in the proposed framework, a pre-training phase builds  $w_0$  and a final adjustment derives the final  $w$ .

The semi-supervised learning scheme possesses some interesting features:

- Since the proposed method could be applied to both linear and nonlinear domains, the result is a completely generalizable learning scheme.

- Clustering and biasing can be tackled independently. If one wants to adopt a particular solution for biasing or a new clustering algorithm is designed, then the two actions can be controlled and adjusted separately.
- If the learning machine is a single-layer learning machine whose cost is convex then convexity is preserved and a global solution is granted.
- Every clustering method can be used to build the reference solution.

### Model Selection

The selection of the optimal hyperparameters of a predictive model is the fundamental problem of SLT and is still the target of current research [38, 40, 41, 57, 67, 68]. The approaches can be divided into two large families: out-of-sample methods, like HO, cross-validation, and the bootstrap [40–42, 69], and more recent in-sample methods, like the class of function-based methods [40] (based on the VC dimension [36], RC [45–47, 70], PAC Bayes theory [49, 50]), algorithm-based methods [43] (based on compression bounds [53], and AS theory [55, 56]).

Out-of-sample methods [40, 71] are favored by practitioners because they work well in many situations and allow the application of simple statistical techniques for estimating the quantities of interest. Some examples of out-of-sample methods are the well-known HO, the KCV, the LOO, and the bootstrap (BOO) [41, 42, 72]. All these techniques rely on a similar idea: The original dataset is resampled, with or without replacement, to build two independent datasets called, respectively, the training and validation (or estimation) sets. The first one is used for training a classifier, while the second one is exploited to estimate its generalization error, so that the hyperparameters can be tuned to achieve its minimum value. Note that both error estimates computed through the training and validation sets are, obviously, optimistically biased; therefore, if a generalization estimate of the final model is desired, it is necessary to build a third independent set, called the test set, by nesting two of the resampling procedures mentioned above. Furthermore, the resampling procedure itself can introduce artifacts in the estimation process and so must be carefully designed.

In-sample methods [40, 71], instead, allow the whole set of available data for both training the model and estimating its generalization error to be exploited, thanks to the application of rigorous statistical procedures [38, 50, 55]. Despite their unquestionable advantage with respect to out-of-sample methods, their use is not

widespread: One of the reasons is the common belief that though in-sample methods are very useful for gaining deep theoretical insights on the learning process or for developing new learning algorithms, they are not suitable for practical purposes. However, recent advances and deeper insights on these methods demonstrate that this is no longer true [73].

Note that the conventional out-of-sample and in-sample techniques neither take into account nor take advantages of the unlabeled samples available in the semi-supervised learning framework. For more details about the advantages and disadvantages of the different methods one can refer to [39, 40, 43].

### Out-of-Sample methods

As described earlier, these techniques rely on a similar idea: The original dataset  $\mathcal{D}_n$  is resampled once or many ( $n_r$ ) times, with or without replacement, to build two independent datasets called training and validation sets, respectively  $\mathcal{T}_{n_t}^r$  and  $\mathcal{V}_{n_v}^r$ , with  $r \in \{1, \dots, n_r\}$ . Note that  $\mathcal{T}_{n_t}^r \cap \mathcal{V}_{n_v}^r = \emptyset$  and  $\mathcal{T}_{n_t}^r \cup \mathcal{V}_{n_v}^r = \mathcal{D}_n$ . Then, in order to select the best set of hyperparameters  $\mathcal{H}$  from a set of possible ones  $\mathfrak{H} = \{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_{n_{\mathcal{H}}}\}$  for the algorithm  $\mathcal{A}_{\mathcal{H}}$  or, in other words, to perform the MS, we have to apply the following procedure:

$$\mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathfrak{H}} \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r). \quad (14)$$

Since the data in  $\mathcal{T}_{n_t}^r$  are i.i.d. from the ones in  $\mathcal{V}_{n_v}^r$ , the idea is that  $\mathcal{H}^*$  should be the set of hyperparameters which result in a small error on a dataset that is independent from the training set. This approach is theoretically grounded by the following reasoning: Since the data in  $\mathcal{T}_{n_t}^r$  are i.i.d. from the ones in  $\mathcal{V}_{n_v}^r$ , we can state, thanks to the Hoeffding’s inequality [74], that the generalization error of the function trained using  $\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}$  can be bounded as:

$$L(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}) \leq \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r) + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n_v}}, \quad (15)$$

and the bound holds with probability  $(1 - \delta)$ . Since we are choosing  $\mathcal{H}^* \in \mathfrak{H}$  (i.e., we are choosing over  $n_{\mathcal{H}}$  functions trained with different configurations of the hyperparameters) we have to apply the union bound [36] and we have that, with probability  $(1 - \delta)$ , the generalization error of the function chosen between the  $n_{\mathcal{H}}$  functions is:

$$\begin{aligned} L(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}) &\leq \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r) + \sqrt{\frac{\ln(\frac{n_{\mathcal{H}}}{\delta})}{2n_v}} \\ &\leq \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r) + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n_v}} \\ &\quad + \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n_v}}. \end{aligned} \quad (16)$$

In repeating the training/validation split procedure  $n_r$  times, then, we choose the set of hyperparameters  $\mathcal{H} \in \mathfrak{H}$  and obtain the generalization error of the classifier  $f_{\mathcal{H}}$  which randomly selects one of the functions  $\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H}^*)}$  with  $r \in \{1, \dots, n_r\}$ . Each time a new sample must be classified, it can be bounded with the probability  $(1 - \delta)$ :

$$\begin{aligned} L(f_{\mathcal{H}}) &\leq \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r) + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n_v}} \\ &\quad + \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n_v}}, \mathcal{H} \\ &\in \mathfrak{H}. \end{aligned} \quad (17)$$

Based on the SLT, we have to choose the set of hyperparameters that minimize the estimated generalization error and obtain that:

$$\begin{aligned} \mathcal{H}^* &: \arg \min_{\mathcal{H} \in \mathfrak{H}} L(f_{\mathcal{H}}) \\ &\approx \arg \min_{\mathcal{H} \in \mathfrak{H}} \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r) \\ &\quad + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n_v}} + \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n_v}} \\ &= \arg \min_{\mathcal{H} \in \mathfrak{H}} \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{T}_{n_t}^r \cup \mathcal{D}_{n_u}, \mathcal{H})}, \mathcal{V}_{n_v}^r). \end{aligned} \quad (18)$$

The first approximation is due to the fact that the distribution of the data is unknown and hence the true generalization error of  $L(f_{\mathcal{H}})$  cannot be computed [36]. Since we have its rigorous upper bound, we can use it based on SLT [36]. The last equality holds because the last two terms,  $\sqrt{\frac{\ln(\frac{1}{\delta})}{2n_v}}$  and  $\sqrt{\frac{\ln(n_{\mathcal{H}})}{2n_v}}$ , are constants and do not affect the choice of  $\mathcal{H}^*$ . Consequently, we have retrieved the criteria of Eq. (14). Note that after the best set of hyperparameters is found, one usually replaces the model  $f_{\mathcal{H}^*}$  with the model obtained by training the algorithm with the whole dataset  $\mathcal{A}_{(\mathcal{D}_n, \mathcal{H}^*)}$  [75]. Moreover, note that this is an approximation, since for classifying a new sample we use the function retrained with  $\mathcal{H}^*$  over the whole set of data

and so, basically, we are not directly optimizing the hyperparameters for  $\mathcal{A}(\mathcal{D}_n, \mathcal{H})$  with  $\mathcal{H} \in \mathfrak{H}$ .

If  $n_r = 1$ , if  $t$  and  $v$  are aprioristically set such that  $n = t + v$ , and if the resample procedure is performed without replacement, we obtain the HO method [40]. In order to implement the KCV method, we have to set  $n_r \leq \binom{n}{k}$ ,  $t = (k - 1)\frac{n}{k}$  and  $v = \frac{n}{k}$  and the resampling must be done without replacement [40, 41, 67]. Finally, for implementing the BOO method,  $t = n$  and  $\mathcal{T}_t^r$  must be sampled with replacement from  $\mathcal{D}_n$ , while  $\mathcal{V}_{n_r}^r = \mathcal{D}_n \setminus \mathcal{T}_t^r$  and  $\mathcal{T}_{n_r}^r$  [40, 42]. Note that for the bootstrap procedure  $n_r \leq \binom{2n - 1}{n}$ .

It is worthwhile noting that the only hypothesis needed in order to rigorously apply the out-of-sample technique is the i.i.d. hypothesis on the data in  $\mathcal{D}_n$  and that all these techniques work for any deterministic algorithm.

### In-Sample Methods

For the in-sample methods, there are two subfamilies of techniques: the class of function-based ones and the algorithm-based ones [43]. The difference between the two is that the first requires the knowledge of  $\mathcal{F}_{\mathcal{H}}$  and so cannot be applied to some algorithms (e.g., the  $k$ -nearest neighbor algorithm) while the second can be applied to any deterministic algorithm without additional knowledge. Both subfamilies, like the out-of-sample methods, require the i.i.d. hypothesis.

#### Vapnik–Chervonenkis Theory

The milestone result from the class of function-based methods in SLT is the VC theory [36]. In this case, the analysis holds just for the semi-supervised learning problems where the hard loss function  $\ell_H$  is exploited:

$$\ell_H(f, z) = \begin{cases} 1 & \text{if } y f(x) \leq 0 \\ 0 & \text{otherwise} \end{cases} = \frac{1 - y \operatorname{sign}[f(x)]}{2}. \tag{19}$$

In the VC theory the following quantity is defined:

$$\mathcal{F}_{\mathcal{H}}|_{\mathcal{D}_n} = \left\{ \left\{ \operatorname{sign}[f(x_1)], \dots, \operatorname{sign}[f(x_n)] \right\} \mid f \in \mathcal{F}_{\mathcal{H}} \right\}, \tag{20}$$

which is the set of distinct functions that shatter the dataset. Then the VC entropy  $H_n(\mathcal{F}_{\mathcal{H}})$  and the annealed VC entropy  $A_n(\mathcal{F}_{\mathcal{H}})$ , together with their empirical counterparts  $\widehat{H}_n(\mathcal{F}_{\mathcal{H}})$  and  $\widehat{A}_n(\mathcal{F}_{\mathcal{H}})$  [36], can be recalled:

$$H_n(\mathcal{F}_{\mathcal{H}}) = \mathbb{E}_{x_1, \dots, x_n} \widehat{H}_n(\mathcal{F}_{\mathcal{H}}), \tag{21}$$

$$\widehat{H}_n(\mathcal{F}_{\mathcal{H}}) = \ln(|\mathcal{F}_{\mathcal{H}}|_{\mathcal{D}_n}|),$$

$$A_n(\mathcal{F}_{\mathcal{H}}) = \ln(\mathbb{E}_{x_1, \dots, x_n} |\mathcal{F}_{\mathcal{H}}|_{\mathcal{D}_n}|), \quad \widehat{A}_n(\mathcal{F}_{\mathcal{H}}) = \widehat{H}_n(\mathcal{F}_{\mathcal{H}}), \tag{22}$$

where  $|\cdot|$  is the cardinality of a set. Based on previous definitions, it is possible to define the growth function and the VC dimension, respectively  $G_n(\mathcal{F}_{\mathcal{H}})$  and  $d_{\text{VC}}(\mathcal{F}_{\mathcal{H}})$ , as:

$$G_n(\mathcal{F}_{\mathcal{H}}) = \max_{x_1, \dots, x_n} \ln(|\mathcal{F}_{\mathcal{H}}|_{\mathcal{D}_n}|), \tag{23}$$

$$d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) = \max_n \{n : G_n(\mathcal{F}_{\mathcal{H}}) = 2^n\}. \tag{24}$$

Note that  $G_n(\mathcal{F}_{\mathcal{H}}) \leq d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n)$ . Thanks to the Vapnik results it is possible to prove that [76]:

$$\mathbb{P} \left\{ \sup_{f \in \mathcal{F}_{\mathcal{H}}} |L^{\ell_H}(f) - \widehat{L}_{\text{emp}}^{\ell_H}(f)| \geq t \right\} \leq 4 \exp \left[ \left( \frac{A_{2n}(\mathcal{F}_{\mathcal{H}})}{n} - t^2 \right) n \right] \leq 4 \exp \left[ \left( \frac{d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n} - t^2 \right) n \right]. \tag{25}$$

Consequently, we have that with probability  $(1 - \delta)$ :

$$L^{\ell_H}(f) \leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n) + \ln(\frac{4}{\delta})}{n}} \leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n}} + \sqrt{\frac{\ln(\frac{4}{\delta})}{n}}, \quad \forall f \in \mathcal{F}_{\mathcal{H}}. \tag{26}$$

Moreover, since  $\mathcal{F}_{\mathcal{H}}$  is chosen in a set of possible spaces  $\mathfrak{F} = \{\mathcal{F}_{\mathcal{H}_1}, \dots, \mathcal{F}_{\mathcal{H}_{n_{\mathcal{H}}}}\}$ , we have to apply the union bound [36, 77], so we have that with probability  $(1 - 2\delta)$ :

$$L^{\ell_H}(f) \leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n}} + \sqrt{\frac{\ln(\frac{4n_{\mathcal{H}}}{\delta})}{n}} \leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n}} + \sqrt{\frac{\ln(\frac{4}{\delta})}{n}} + \sqrt{\frac{\ln(n_{\mathcal{H}})}{n}}, \quad \forall f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}. \tag{27}$$

The VC theory is basically a more refined form of union bound that is able to deal with the class of functions which have an infinite number of functions [36]. In particular, the entropies and the growth function measure the number of distinct functions with respect to the distribution of the data, while the  $d_{\text{VC}}$  is a measure of dimensionality for a general nonlinear class of functions [36, 44].

Recently in [78, 79] it has been proved that with probability  $(1 - \delta)$ :

$$\begin{aligned}
 A_{2n}(\mathcal{F}_{\mathcal{H}}) &\leq 4\widehat{A}_n(\mathcal{F}_{\mathcal{H}}) + 8\ln\left(\frac{1}{\delta}\right) \\
 &= 4\widehat{H}_n(\mathcal{F}_{\mathcal{H}}) + 8\ln\left(\frac{1}{\delta}\right). \tag{28}
 \end{aligned}$$

Consequently, we can state that with probability  $(1 - \delta)$ :

$$\begin{aligned}
 &\mathbb{P}\left\{\sup_{f \in \mathcal{F}_{\mathcal{H}}} \left|L^{\ell_H}(f) - \widehat{L}_{\text{emp}}^{\ell_H}(f)\right| \geq t\right\} \\
 &\leq 4 \exp\left[\left(\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}}) + 8\ln\left(\frac{1}{\delta}\right)}{n} - t^2\right)n\right]. \tag{29}
 \end{aligned}$$

Based on this last result, we can state that with probability  $(1 - 2\delta)$  the following inequality holds:

$$\begin{aligned}
 L^{\ell_H}(f) &\leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}}) + 8\ln\left(\frac{1}{\delta}\right) + \ln\left(\frac{4}{\delta}\right)}{n}} \\
 &\leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}}) + 9\ln\left(\frac{4}{\delta}\right)}{n}} \\
 &\leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n}} + \sqrt{\frac{9\ln\left(\frac{4}{\delta}\right)}{n}}, \quad \forall f \in \mathcal{F}_{\mathcal{H}}. \tag{30}
 \end{aligned}$$

Moreover, since  $\mathcal{F}_{\mathcal{H}} \in \mathfrak{F}$  by applying the union bound [36, 77] we have that with probability  $(1 - 2\delta)$ :

$$\begin{aligned}
 L^{\ell_H}(f) &\leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n}} + \sqrt{\frac{9\ln\left(\frac{4n}{\delta}\right)}{n}} \\
 &\leq \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n}} + \sqrt{\frac{9\ln\left(\frac{4}{\delta}\right)}{n}} \\
 &\quad + \sqrt{\frac{9\ln(n)}{n}}, \quad \forall f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}. \tag{31}
 \end{aligned}$$

Based on the results of Eqs. (27) and (31) we can present the two MS procedures based on the VC theory, noting that some terms are constants. In particular, the original approach based on Eq. (27) says that:

$$\begin{aligned}
 f_{\mathcal{H}^*}^*, \mathcal{H}^* &: \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} L^{\ell_H}(f) \\
 &\approx \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{d_{\text{VC}}(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n}}. \tag{32}
 \end{aligned}$$

The second approach, based on Eq. (31), says that:

$$\begin{aligned}
 f_{\mathcal{H}^*}^*, \mathcal{H}^* &: \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} L^{\ell_H}(f) \\
 &\approx \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}^{\ell_H}(f) + \sqrt{\frac{4\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n}}. \tag{33}
 \end{aligned}$$

In order to extend the analysis to a real-valued loss we have to exploit a result of [36] which states that:

$$\begin{aligned}
 &\mathbb{P}\left\{\sup_{f \in \mathcal{F}_{\mathcal{H}}} \left|L(f) - \widehat{L}_{\text{emp}}(f)\right| \geq t\right\} \\
 &= \mathbb{P}\left\{\sup_{f \in \mathcal{F}_{\mathcal{H}}} \left|\mathbb{E}_z \ell(f, z) - \frac{1}{n} \sum_{i=1}^n \ell(f, z_i)\right| \geq t\right\} \\
 &\leq \mathbb{P}\left\{\sup_{f \in \mathcal{F}_{\mathcal{H}}} \sup_{\beta \in [0,1]} |\mathbb{E}_z \{\ell(f, z) > \beta\}| \geq t\right\} \\
 &\quad - \frac{1}{n} \sum_{i=1}^n \{|\ell(f, z_i) > \beta|\} \geq t\}. \tag{34}
 \end{aligned}$$

Then, we define:

$$\mathcal{F}_{\mathcal{H}}^{\beta} |_{\mathcal{D}_n} = \left\{[\ell(f, z_1) > \beta], \dots, [\ell(f, z_n) > \beta]\right\} \mid f \in \mathcal{F}_{\mathcal{H}} \}. \tag{35}$$

Based on these definitions, it is possible to define the VC entropy, the growth function, and the VC dimension for real-valued functions:

$$\begin{aligned}
 A_n^{\beta}(\mathcal{F}_{\mathcal{H}}) &= \ln\left(\mathbb{E}_{x_1, \dots, x_n} \sup_{\beta \in [0,1]} \left|\mathcal{F}_{\mathcal{H}}^{\beta} |_{\mathcal{D}_n}\right|\right), \\
 \widehat{A}_n^{\beta}(\mathcal{F}_{\mathcal{H}}) &= \ln\left(\sup_{\beta \in [0,1]} \left|\mathcal{F}_{\mathcal{H}}^{\beta} |_{\mathcal{D}_n}\right|\right), \tag{36}
 \end{aligned}$$

$$\begin{aligned}
 G_n^{\beta}(\mathcal{F}_{\mathcal{H}}) &= \max_{x_1, \dots, x_n} \sup_{\beta \in [0,1]} \ln\left(\left|\mathcal{F}_{\mathcal{H}}^{\beta} |_{\mathcal{D}_n}\right|\right), \\
 d_{\text{VC}}^{\beta}(\mathcal{F}_{\mathcal{H}}) &= \max_n \{n : G_n^{\beta}(\mathcal{F}_{\mathcal{H}}) = 2^n\},
 \end{aligned}$$

and finally state that:

$$\begin{aligned}
 \mathbb{P}\left\{\sup_{f \in \mathcal{F}_{\mathcal{H}}} \left|L(f) - \widehat{L}_{\text{emp}}(f)\right| \geq t\right\} &\leq 4 \exp\left[\left(\frac{A_{2n}^{\beta}(\mathcal{F}_{\mathcal{H}})}{n} - t^2\right)n\right] \\
 &\leq 4 \exp\left[\left(\frac{d_{\text{VC}}^{\beta}(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n} - t^2\right)n\right]. \tag{37}
 \end{aligned}$$

since  $G_n^{\beta}(\mathcal{F}_{\mathcal{H}}) \leq d_{\text{VC}}^{\beta}(\mathcal{F}_{\mathcal{H}}) \ln(n)$ . By following the same argument presented before we can state that with probability  $(1 - \delta)$ :

$$A_{2n}^{\beta}(\mathcal{F}_{\mathcal{H}}) \leq 4\widehat{A}_n^{\beta}(\mathcal{F}_{\mathcal{H}}) + 8\ln\left(\frac{1}{\delta}\right). \tag{38}$$

Consequently, we can state that with probability  $(1 - \delta)$ :

$$\begin{aligned}
 &\mathbb{P}\left\{\sup_{f \in \mathcal{F}_{\mathcal{H}}} \left|L(f) - \widehat{L}_{\text{emp}}(f)\right| \geq t\right\} \\
 &\leq 4 \exp\left[\left(\frac{4\widehat{A}_n^{\beta}(\mathcal{F}_{\mathcal{H}}) + 8\ln\left(\frac{1}{\delta}\right)}{n} - t^2\right)n\right]. \tag{39}
 \end{aligned}$$

By following the same argument presented before, it is possible to propose the two approaches for MS based on

the VC theory for real-valued loss functions. The first approach states that:

$$f_{\mathcal{H}^*, \mathcal{H}^*}^* : \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}(f) + \sqrt{\frac{d_{\text{VC}}^\beta(\mathcal{F}_{\mathcal{H}}) \ln(n)}{n}}. \tag{40}$$

The second approach, based on Eq. (31), instead says that:

$$f_{\mathcal{H}^*, \mathcal{H}^*}^* : \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}(f) + \sqrt{\frac{4\widehat{A}_n^\beta(\mathcal{F}_{\mathcal{H}})}{n}}. \tag{41}$$

*(Local) Rademacher Complexity*

One of the most powerful classes of function-based methods is based on the RC [40, 70]. In particular, it is possible to prove that the following bound holds with probability  $(1 - \delta)$  [80]:

$$L(f) \leq \widehat{L}_{\text{emp}}(f) + \widehat{R}_n(\mathcal{F}_{\mathcal{H}}) + 3\sqrt{\frac{\ln(\frac{2}{\delta})}{2n}} + 3\sqrt{\frac{\ln(n_{\mathcal{H}})}{2n}}, \tag{42}$$

$\forall f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F},$

where

$$\widehat{R}_n(\mathcal{F}_{\mathcal{H}}) = \mathbb{E}_\sigma \sup_{f \in \mathcal{F}_{\mathcal{H}}} \frac{2}{n} \sum_{i=1}^n \sigma_i \ell(f, z_i),$$

$$\sigma_{i \in \{1, \dots, n\}} \in \{\pm 1\}, \quad \mathbb{P}\{\sigma_i = +1\} = \mathbb{P}\{\sigma_i = -1\} = \frac{1}{2}. \tag{43}$$

RC is essentially a more refined form of union bound that is able to deal with class of functions with infinite number of functions [80]. Another interpretation of the RC is that it measures the ability of the class of functions to fit random labels [38]. More refined interpretations and the advantages and disadvantages with respect to the VC theory can be found in [44].

Therefore, based on the same principles described above, we have that:

$$f_{\mathcal{H}^*, \mathcal{H}^*}^* : \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} L(f) \approx \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}(f) + \widehat{R}_n(\mathcal{F}_{\mathcal{H}}). \tag{44}$$

Let us use the following loss function:

$$\ell_S(f, z) = \begin{cases} 1 & \text{if } yf(x) < 0 \\ 1 - yf(x) & \text{if } 0 \leq yf(x) \leq 1, \\ 0 & \text{if } yf(x) > 1 \end{cases}, \tag{45}$$

also called soft loss function. Let us also suppose that the  $f \in \mathcal{F} \in \mathfrak{F}$  can be expressed as:

$$f(x) = w^T x = \sum_{i=1}^n \alpha_i \phi(x_i)^T \phi(x), \quad \alpha \in \mathbb{R}^n, \tag{46}$$

where  $w$  is constrained such that  $\|w - \lambda_1 w_0\|^2 \leq W^2, \lambda_1 \in [0, \infty)$ . (47)

In such a case, instead of computing  $\widehat{R}_n(\mathcal{F}_{\mathcal{H}})$  we can bound it as [80]:

$$\begin{aligned} \widehat{R}_n^{\ell_S}(\mathcal{F}_{\mathcal{H}}) &= \mathbb{E}_\sigma \sup_{f \in \mathcal{F}_{\mathcal{H}}} \frac{2}{n} \sum_{i=1}^n \sigma_i \ell_S(f, z_i) \\ &\leq 4W \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}}. \end{aligned} \tag{48}$$

Consequently, we obtain that:

$$\begin{aligned} f_{\mathcal{H}^*, \mathcal{H}^*}^* : \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} L^{\ell_S}(f) &\approx \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}^{\ell_S}(f) + \widehat{R}_n(\mathcal{F}_{\mathcal{H}}) \\ &\approx \arg \min_{f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}} \widehat{L}_{\text{emp}}(f) + 4W \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}}. \end{aligned} \tag{49}$$

Recently, a more refined version of the RC, named the local Rademacher complexity (LRC), which is able to discard those functions from the class of functions which are not useful during the learning process, has been proposed in the literature [46, 47]. The result is the following bound which holds with probability  $(1 - 3n_{\mathcal{H}}\delta)$ :

$$\begin{aligned} L(f) &\leq \min_{K \in (1, \infty)} \frac{K}{K-1} \widehat{L}_{\text{emp}}(f) + \frac{r}{K}, \quad \forall f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}. \\ \text{s.t. } \sup_{\alpha \in [0, 1]} \alpha \left[ \text{LR} + \sqrt{\frac{2 \ln(\frac{1}{\delta})}{n}} \right] &\leq \frac{r}{K}, \quad r \geq 0 \\ \text{LR} &\leq \widehat{R} \left( \left\{ f : f \in \mathcal{F}_{\mathcal{H}}, \widehat{L}_{\text{emp}}(f) \leq \frac{r}{\alpha} + \text{LR} + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n}} \right\} \right) \\ &+ \sqrt{\frac{2 \ln(\frac{1}{\delta})}{n}} \end{aligned} \tag{50}$$

Unfortunately, computing the above LRC-based bound is not a trivial task and can be undertaken only when the number of samples is limited [45–47]. Furthermore, when there are a large number of samples, the advantages of using the LRC with respect to RC are not so evident [45].

*PAC Bayes Theory*

The PAC Bayes theory is the last major theory in the class of function-based methods. In the PAC Bayes theory, we do not bound the error of a  $f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F}$  but instead bound the error of the stochastic Gibbs classifier (also called randomized classifier) and the majority vote classifier (also called Bayes

classifier) [49–51, 68, 81, 82]. The Gibbs classifier draws an  $f \in \mathcal{F}_{\mathcal{H}}$  according to a probability distribution  $Q_{\mathcal{H}}$  over  $\mathcal{F}_{\mathcal{H}}$  each time a label for an input  $X \in \mathcal{X}$  is required. One can also choose from different distributions  $Q_{\mathcal{H}} \in \mathfrak{Q} = \{Q_{\mathcal{H}}^1, \dots, Q_{\mathcal{H}}^n\}$  [50]. The Bayes classifier is the majority voting of the different classifiers according to the distribution  $Q_{\mathcal{H}}$  [51]. This framework, despite being really powerful, is not suited for algorithm like the ELM since it is built for ensemble methods [51] like Bagging [83, 84], Boosting [85, 86] or Bayesian approaches [87].

*Algorithmic Stability Theory*

Algorithmic-based methods circumvent the problem of knowing the class of functions by defining the properties that an algorithm should satisfy in order to achieve good generalization performances [43]. The AS theory [43, 55, 88], in particular, states that it is possible to prove that the following bounds hold with probability  $(1 - \delta)$  [55]:

$$L(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}) \leq \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}) + \sqrt{\frac{n_{\mathcal{H}}}{2n\delta} + \frac{3n_{\mathcal{H}}\beta_{\text{emp}}}{\delta}}, \forall \mathcal{H} \in \mathfrak{H}, \tag{51}$$

$$L(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}) \leq \widehat{L}_{\text{loo}}(\mathcal{A}_{\mathcal{H}}) + \sqrt{\frac{n_{\mathcal{H}}}{2n\delta} + \frac{3n_{\mathcal{H}}\beta_{\text{loo}}}{\delta}}, \forall \mathcal{H} \in \mathfrak{H}, \tag{52}$$

where

$$\beta_{\text{emp}}(\mathcal{A}_{\mathcal{H}}, n) = \mathbb{E}_{\mathcal{D}_n, z_i} |\ell(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}, z_i) - \ell(\mathcal{A}_{(\mathcal{D}_n^i \cup \mathcal{D}_{n_u}, \mathcal{H})}, z_i)|, \tag{53}$$

$$\beta_{\text{loo}}(\mathcal{A}_{\mathcal{H}}, n) = \mathbb{E}_{\mathcal{D}_n, z} |\ell(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}, z) - \ell(\mathcal{A}_{(\mathcal{D}_n^i \cup \mathcal{D}_{n_u}, \mathcal{H})}, z)|. \tag{54}$$

Basically, AS states that if an algorithm selects similar functions, the training set being slightly changed results in the algorithm having good generalization performances [56].

It has been proven recently in [43] that  $\beta_{\text{loo}}(\mathcal{A}_{\mathcal{H}}, n)$  can be estimated directly from the data, if  $\beta_{\text{loo}}(\mathcal{A}_{\mathcal{H}}, n)$  decreases with  $n$ . We wish to highlight that this property is a desirable requirement for any learning algorithm, since we need that in order to be able to prove the learnability in the AS framework:

$$\lim_{n \rightarrow \infty} \beta_{\text{emp}}(\mathcal{A}_{\mathcal{H}}, n) = 0, \tag{55}$$

or that, in other words, the impact on the learning procedure of removing or replacing one sample from  $\mathcal{D}_n$  should decrease, on average, as  $n$  grows. Numerous researchers

have already studied this property in the past. In particular, it is related to the consistency concept [89]. However, connections can also be identified with the trend of the learning curves of an algorithm [90]. Such quantity is also strictly linked to the concept of Smart Rule [89]. It is worth underlining that, in many of the above-referenced works, the required property is proved satisfied by many well-known algorithms like least squares, regularized least squares, and kerneled regularized least squares. Consequently, the property is also true for ELM, which is itself represented by a random protection with a regularized least squares.

Therefore in [43] it is proved that with probability  $(1 - \delta)$ :

$$\beta_{\text{loo}}(\mathcal{A}_{\mathcal{H}}, n) \leq \frac{8}{n\sqrt{n}} \sum_{i,j,k=1}^{\sqrt{n}/2} |\ell(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}/2}^k \cup \mathcal{D}_{n_u}, \mathcal{H})}, z_j^k) - \ell(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}/2}^i \cup \mathcal{D}_{n_u}, \mathcal{H})}, z_j^k)| + \sqrt{\frac{\ln(\frac{1}{\delta})}{n}}, \tag{56}$$

where

$$\mathcal{D}_{\sqrt{n}/2}^k : \{z_{(k-1)\sqrt{n}+1}, \dots, z_{(k-1)\sqrt{n}+\sqrt{n}/2}\}, k \in \{1, \dots, \sqrt{n}/2\}, \tag{57}$$

$$z_j^k : z_{(k-1)\sqrt{n}+\sqrt{n}/2+j}, k \in \{1, \dots, \sqrt{n}/2\}. \tag{58}$$

By plugging this last result in the bound of Eq. (52) we obtain the fully empirical-based bound of [43], where every quantity involved in the bound can be computed from the available data. Note also that  $\widehat{H}_{\text{loo}}(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}/2}, \mathcal{H})}, \mathcal{D}_{\sqrt{n}/2})$  can be effectively estimated via a Monte Carlo procedure: This enables computing a subset  $s_{MC}$  of the required steps, i.e.,  $s_{MC} \ll \frac{n\sqrt{n}}{8}$ .

The bounds of Eqs. (51) and (52) are polynomial bounds in  $n$  (so not very tight when  $n$  is small), while  $\beta_{\text{emp}}$  and  $\beta_{\text{loo}}$  are two versions of hypothesis stability (HS) which are able to take into account both the properties of the algorithm and the property of the distribution that has generated the data  $\mathcal{D}_n$  [43, 55]. It is possible to improve the bounds of Eqs. (51) and (52) by exploiting a stronger notion of AS, known as the uniform stability (US). In particular, the following bounds hold with probability  $(1 - \delta)$ :

$$L(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}) \leq \widehat{L}_{\text{emp}}(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}) + \beta^i + (4n\beta^i + 1) \sqrt{\frac{\ln(\frac{n_{\mathcal{H}}}{\delta})}{2n}}, \forall \mathcal{H} \in \mathfrak{H}, \tag{59}$$

$$L(\mathcal{A}_{(\mathcal{D}_n \cup \mathcal{D}_{n_u}, \mathcal{H})}) \leq \widehat{L}_{\text{loo}}(\mathcal{A}_{\mathcal{H}}) + \beta^i + (4n\beta^i + 1) \sqrt{\frac{\ln(\frac{n_{\mathcal{H}}}{\delta})}{2n}}, \forall \mathcal{H} \in \mathfrak{H}, \tag{60}$$

where

$$\beta^i = \left| \ell(\mathcal{A}(\mathcal{D}_n \cup \mathcal{D}_{nu}, \mathcal{H}), \cdot) - \ell(\mathcal{A}(\mathcal{D}_n^i \cup \mathcal{D}_{nu}, \mathcal{H}), \cdot) \right|_{\infty}, \quad (61)$$

$$\beta^{\setminus i} = \left| \ell(\mathcal{A}(\mathcal{D}_n \cup \mathcal{D}_{nu}, \mathcal{H}), \cdot) - \ell(\mathcal{A}(\mathcal{D}_n^{\setminus i} \cup \mathcal{D}_{nu}, \mathcal{H}), \cdot) \right|_{\infty}. \quad (62)$$

Note that  $\beta^i \leq 2\beta^{\setminus i}$ .

Unfortunately, the US ( $\beta^i$  or  $\beta^{\setminus i}$ ) is not able to take into account the properties of the distribution that has generated the data  $\mathcal{D}_n$  and sometimes is not even able to capture the properties of the algorithm because it deals with a worst-case learning scenario [43].

All the four AS-based bounds of Eqs. (51), (52), (59), and (60) can be used to select the best set of hyperparameters  $\mathcal{H} \in \mathfrak{H}$  for the algorithm  $\mathcal{A}_{\mathcal{H}}$ . In particular, all the bounds are in the form:  $L(\mathcal{A}(\mathcal{D}_n, \mathcal{H})) \leq \epsilon(\mathcal{A}_{\mathcal{H}}, \mathcal{D}_n, n, \delta, n_{\mathcal{H}})$ . In order to perform the MS procedure we have:

$$\mathcal{A}(\mathcal{D}_n, \mathcal{H}^*) : \arg \min_{\mathcal{H} \in \mathfrak{H}} \epsilon(\mathcal{A}_{\mathcal{H}}, \mathcal{D}_n, n, \delta, n_{\mathcal{H}}). \quad (63)$$

The procedure of Eq. (63) can be exploited with any algorithm for which it is possible to compute one notion of AS.

### Compression Bound

The compression bound is the result of the approximation of the Kolmogorov theory [91] and, in particular, the minimum description length principle [92]. The compression bound [53] states that the less data of  $\mathcal{D}_n$  we use for learning the better generalization performance our model will have. Unfortunately, this approach is not suited for ELM but just for algorithms which produce sparse models like SVM [39].

## The Use of Unlabeled Samples for Extreme Learning Machine Model Selection

As we described before, it is not possible (or it does not make sense) to apply some of the methodologies described above (e.g., the PAC Bayes theory and the compression bound theory). In this section, we show how to apply the out-of-sample methods, the VC theory, the RC theory, and the AS theory to the ELM and how to take advantage of unlabeled samples both for training a more accurate model thanks to the regularization framework depicted in “Semi-supervised Extreme Learning Machines” section, and during the MS process. In particular, we show how to perform the MS effectively with and without exploiting the unlabeled samples for the three version of ELM presented in this paper:

- ELM-NoR: the easier ELM which does not implement any regularization strategy [see Eq. (5)],
- ELM-R: the now-standard ELM which implements the typical Tikhonov regularization schema [93] [see Eq. (7)],

- ELM-SemiR: the ELM which implements the semi-supervised regularization schema presented in Eq. (11).

The ELM-NoR has just one hyperparameter: the number of hidden neurons, so  $\mathcal{H} = \{N_h\}$ . For ELM-R the hyperparameters are the number of hidden neurons and the regularization hyperparameter, so we have that  $\mathcal{H} = \{N_h, \lambda\}$ . Finally, for the ELM-SemiR the hyperparameters are the number of hidden neurons and the two regularization hyperparameters, so we have that  $\mathcal{H} = \{N_h, \lambda_1, \lambda_2\}$ .

## Out-of-Sample Methods

The main problem of the out-of-sample methods, as described in “Out-of-Sample Methods” section, is that instead of tuning the hyperparameters for the classifier we are tuning the performance of an ensemble classifier. Since we are dealing with binary classification a reasonable choice is to use the hard loss function  $\ell_H$  which counts the number of errors of a classifier trained with  $\mathcal{A}_{\mathcal{H}}$  over a dataset. In particular, if we use the ELM we can define:

$$\begin{aligned} w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}} &= \mathcal{A}(\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}, \mathcal{H}), w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{nu}} \\ &= \mathcal{A}(\mathcal{D}_n \cup \mathcal{D}_{nu}, \mathcal{H}), w_{\mathcal{H}}^{\mathcal{D}_n^{\setminus i} \cup \mathcal{D}_{nu}} = \mathcal{A}(\mathcal{D}_n^{\setminus i} \cup \mathcal{D}_{nu}, \mathcal{H}), \end{aligned} \quad (64)$$

where  $\mathcal{A}_{\mathcal{H}}$  can be the solution of ELM-NoR or ELM-R or ELM-SemiR trained over  $\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}$  for  $w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}}$ , trained over  $\mathcal{D}_n \cup \mathcal{D}_{nu}$  for  $w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{nu}}$ , etc. Note that  $\mathcal{D}_{nu}$  can be exploited or not based on  $\mathcal{A}$ , in fact ELM-NoR and ELM-R do not use it. The procedure of Eq. (18) states that:

$$w_{\mathcal{H}^*}^{\mathcal{D}_n \cup \mathcal{D}_{nu}}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathfrak{H}} \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}^{\ell_H}(w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}}, \mathcal{V}_{n_r}^r). \quad (65)$$

We remember that (see “Out-of-Sample Methods” section) we are minimizing the error of the classifier which randomly chooses one of the  $w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}}$  with  $r \in \{1, \dots, n_r\}$  while at the end of the procedure we classify a new sample with  $w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{nu}}$ . This produces a sub-optimal result [40, 57]. In order to fix this bias we can employ the unlabeled samples. In particular, we can estimate the difference between the error of  $w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}}$  with  $r \in \{1, \dots, n_r\}$  and the one of  $w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{nu}}$ . In fact, when the hard loss function is used, the error of  $w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{nu}}$  is bounded by the average error of  $w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}}$  with  $r \in \{1, \dots, n_r\}$  plus the average difference between the prediction of  $w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{nu}}$  and  $w_{\mathcal{H}}^{\mathcal{F}_{n_r}^r \cup \mathcal{D}_{nu}}$  with  $r \in \{1, \dots, n_r\}$ :

$$L^{\ell_H}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) \leq \frac{1}{n_r} \sum_{r=1}^{n_r} L^{\ell_H}(w_{\mathcal{H}}^{\mathcal{F}_r^r \cup \mathcal{D}_{n_u}}) + \frac{1}{n_r} \sum_{r=1}^{n_r} \mathbb{P} \left\{ \text{sign} \left[ \left( w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}} \right)^T x \right] \neq \text{sign} \left[ \left( w_{\mathcal{H}}^{\mathcal{F}_r^r \cup \mathcal{D}_{n_u}} \right)^T x \right] \right\}. \tag{66}$$

The first term can be bounded as we have done in “Out-of-Sample Methods” section, while the second term can be bounded by using the unlabeled patterns. In fact, if we define the following quantity:

$$\widehat{D}_{\mathcal{H}} = \frac{1}{n_r(n+n_u)} \sum_{r=1}^{n_r} \sum_{i=1}^{n+n_u} \left\{ \text{sign} \left[ \left( w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}} \right)^T x \right] \neq \text{sign} \left[ \left( w_{\mathcal{H}}^{\mathcal{F}_r^r \cup \mathcal{D}_{n_u}} \right)^T x \right] \right\}, \tag{67}$$

since the data in  $\mathcal{D}_n \cup \mathcal{D}_{n_u}$  are i.i.d., the quantity of Eq. (67) is an unbiased estimator of

$$D_{\mathcal{H}} = \frac{1}{n_r} \sum_{r=1}^{n_r} \mathbb{P} \left\{ \text{sign} \left[ \left( w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}} \right)^T x \right] \neq \text{sign} \left[ \left( w_{\mathcal{H}}^{\mathcal{F}_r^r \cup \mathcal{D}_{n_u}} \right)^T x \right] \right\}. \tag{68}$$

Consequently by exploiting the Hoeffding’s inequality we can state that with probability  $(1 - \delta)$ :

$$D_{\mathcal{H}} \leq \widehat{D}_{\mathcal{H}} + \sqrt{\frac{\ln(\frac{1}{\delta})}{2(n+n_u)}}. \tag{69}$$

Consequently, we have that with probability  $(1 - 2\delta)$ :

$$L^{\ell_H}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) \leq \frac{1}{n_r} \sum_{r=1}^{n_r} L^{\ell_H}(w_{\mathcal{H}}^{\mathcal{F}_r^r \cup \mathcal{D}_{n_u}}) + D_{\mathcal{H}} \leq \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}^{\ell_H}(\mathcal{A}(\mathcal{F}_r^r \cup \mathcal{D}_{n_u}, \mathcal{H}), \mathcal{V}_{n_r}^r) + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n_v}} + \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n_v}} + \widehat{D}_{\mathcal{H}} + \sqrt{\frac{\ln(\frac{1}{\delta})}{2(n+n_u)}} + \sqrt{\frac{\ln(n_{\mathcal{H}})}{2(n+n_u)}}, \quad \mathcal{H} \in \mathfrak{H}. \tag{70}$$

Note that, if we have just few unlabeled samples the bound is very loose, while if we have a lot of unlabeled samples (in the semi-supervised learning framework usually  $n_u \gg n$ ) the bound fast converges to the conventional bound [see Eq. (17)], plus  $\widehat{D}_{\mathcal{H}}$  which takes into account the bias discussed above. Based on this last result we can derive the out-of-sample MS procedure for ELM which exploits also the unlabeled samples:

$$w_{\mathcal{H}^*}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathfrak{H}} \frac{1}{n_r} \sum_{r=1}^{n_r} \widehat{L}_{\text{emp}}^{\ell_H}(w_{\mathcal{H}}^{\mathcal{F}_r^r \cup \mathcal{D}_{n_u}}, \mathcal{V}_{n_r}^r) + \widehat{D}_{\mathcal{H}}. \tag{71}$$

where  $\widehat{D}_{\mathcal{H}}$  is defined above.

### Vapnik–Chervonenkis Theory

Let us start by considering the ELM-NoR. Let us consider the VC theory when the hard loss function is employed. Since the ELM searches a linear separator in the space defined by the random protection of the original input space  $\mathbb{R}^r$  into the space defined by the  $N_h$  hidden neurons we have that  $d_{\text{VC}} \leq N_h$  [36].

Unfortunately, this is a loose upper bound which does not depend on the distribution of the data [36]. In order to be able to take into account the distribution of the data we have to use the VC entropy  $\widehat{A}_n(\mathcal{F}_{\mathcal{H}})$ . Note that, from Eq. (33), in order to perform the MS procedure with the VC entropy we need to compute  $\widehat{A}_n(\mathcal{F}_{\mathcal{H}})/n \in [0, \ln(2)]$ . The VC entropy is the number of configurations of the labels that can be shattered by  $\mathcal{F}_{\mathcal{H}}$ . Consequently, let  $\sigma_i \in \{-1, 1\}^n$  be one of the possible  $2^n$  configurations of the labels; we have to search how many of them can be shattered by a linear separator in the random projection space, then we have to check for how many of the following problems

$$\min_w \theta^T w, \quad \text{s.t.} \quad Hw = \sigma_i, \quad i \in \{1, \dots, 2^n\}, \tag{72}$$

at least one solution exists. Note that the above problem is a linear programming (LP) problem [94]. Searching for a feasible solution of an LP problem is again an LP problem [94] which can be solved in polynomial time [94]. We have to solve  $2^n$  problems and this represents an NP problem. The issue can be circumvented by noting that we can estimate  $\widehat{A}_n(\mathcal{F}_{\mathcal{H}})$  through a Monte Carlo procedure by checking just a random subset, and in particular  $n_{MC} \leq 2^n$  realizations of the labels [44]. If we indicate with  $\widehat{\widehat{A}}_n(\mathcal{F}_{\mathcal{H}})$  the logarithm number of configurations of the  $n_{MC}$  that can be shattered, thanks to the Serfling’s bound [95] (since we are bounding the expected value of an hypergeometric distribution), we can state that with probability  $(1 - \delta)$ :

$$\mathbb{P} \left\{ \frac{\widehat{\widehat{A}}_n(\mathcal{F}_{\mathcal{H}})}{n} \geq \frac{\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n_{MC}} + t \right\} \leq e^{-\frac{2n_{MC} \frac{t^2}{\ln(2)}}{1 - \frac{n_{MC}-1}{2^n}}} \leq e^{-\frac{n_{MC} t^2}{1 - \frac{n_{MC}-1}{2^n}}}. \tag{73}$$

Note that the quality of the estimation does not depend on  $n$  but just on  $n_{MC}$ . Consequently,  $\widehat{\widehat{A}}_n(\mathcal{F}_{\mathcal{H}})/n_{MC}$  rapidly converges to its mean  $\widehat{A}_n(\mathcal{F}_{\mathcal{H}})/n$ . Consequently, let us suppose that  $w_{\mathcal{H}}$  is the solution of the ELM-NoR for a value of its hyperparameters, thanks to the procedure of Eqs. (32) and (33), we have that, based on the VC theory:

$$w_{\mathcal{H}^*}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathfrak{S}} \widehat{L}_{\text{emp}}^{\ell_H}(w_{\mathcal{H}}) + \sqrt{\frac{N_H}{n}}, \quad (74)$$

$$w_{\mathcal{H}^*}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathfrak{S}} \widehat{L}_{\text{emp}}^{\ell_H}(w_{\mathcal{H}}) + \sqrt{4 \frac{\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n_{MC}}}. \quad (75)$$

We now show that the unlabeled samples can be useful to improve the quality of the estimation of the VC entropy. In particular, let us suppose that  $\mathcal{D}_{n_u}$  contains more samples than  $\mathcal{D}_n$ , in particular  $n_u \geq 2n$ . This is a reasonable hypothesis since usually the number of unlabeled samples exceeds by a large amount the number of labeled ones [96]. In particular, let us define  $m = \lfloor n + n_u/2n \rfloor$  and let us consider the original bound of [33]:

$$\begin{aligned} & \mathbb{P} \left\{ \sup_{f \in \mathcal{F}_{\mathcal{H}}} |L^{\ell_H}(f) - \widehat{L}_{\text{emp}}^{\ell_H}(f)| \geq t \right\} \\ & \leq 4 \exp \left[ \left( \frac{A_{2n}(\mathcal{F}_{\mathcal{H}})}{n} - t^2 \right) n \right]. \end{aligned} \quad (76)$$

We show that  $A_{2n}(\mathcal{F}_{\mathcal{H}})$  can be estimated more effectively with the use of the unlabeled samples. In particular, let us define the following quantity:

$$\begin{aligned} & \widehat{A}_{2n}^m(\mathcal{F}_{\mathcal{H}}) \\ & = \frac{1}{m} \sum_{i=0}^{m-1} \ln \left( \left| \left\{ \left\{ \text{sign}[f(x_{i2n+1})], \dots, \text{sign}[f(x_{i2n+2n})] \right\} \mid f \in \mathcal{F}_{\mathcal{H}} \right\} \right| \right), \end{aligned} \quad (77)$$

which is basically the VC entropy for sample size of  $2n$  averaged over  $m$  different realizations. Thanks to the results of [78, 79] we can state that with probability  $(1 - \delta)$ :

$$A_{2n}(\mathcal{F}_{\mathcal{H}}) \leq \widehat{A}_{2n}^m(\mathcal{F}_{\mathcal{H}}) + \frac{1}{m} \ln \left( \frac{1}{\delta} \right), \quad (78)$$

which is a much tighter estimate with respect to the one presented in Eq. (28). Moreover,  $\widehat{A}_{2n}^m(\mathcal{F}_{\mathcal{H}})$  can be estimated from the data since we do not need the labels of the unlabeled samples; hence, by following the same reasoning presented above we can define a Monte Carlo estimation of  $\widehat{A}_{2n}^m(\mathcal{F}_{\mathcal{H}})$  through  $\widehat{\widehat{A}}_{2n}^m(\mathcal{F}_{\mathcal{H}})$ , which is VC entropy-based MS procedure which exploits also the unlabeled samples [the counterpart of the method of Eq. (75)]:

$$w_{\mathcal{H}^*}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathfrak{S}} \widehat{L}_{\text{emp}}^{\ell_H}(w_{\mathcal{H}}) + \sqrt{\frac{\widehat{\widehat{A}}_n^m(\mathcal{F}_{\mathcal{H}})}{n_{MC}}}. \quad (79)$$

Regarding the ELM-R and the ELM-SemiR the reasoning is more complex. In fact, in this case we cannot use the hard loss function  $\ell_H$  since we would eliminate the effect of the regularization hyperparameter [76, 97]. For this reason we have to employ a smooth loss function like the  $\ell_S$ . In

particular, it is straightforward to see that  $\ell_H \leq \ell_S$ . So, even if we use  $\ell_S$  we still have information about  $\ell_H$  [80]. In order to apply the procedure for real-valued losses [see Eqs. (40) and (41)] we have to compute the VC entropy and the VC dimension for real-valued functions which are  $d_{\text{VC}}^{\beta}(\mathcal{F}_{\mathcal{H}})$  and  $\widehat{A}_n^{\beta}(\mathcal{F}_{\mathcal{H}})$ . Unfortunately, upper bounding the  $d_{\text{VC}}^{\beta}(\mathcal{F}_{\mathcal{H}})$  is a rather complex phase while estimating  $\widehat{A}_n^{\beta}(\mathcal{F}_{\mathcal{H}})$  cannot be transformed to a polynomial problem as we have done for the  $\widehat{A}_n(\mathcal{F}_{\mathcal{H}})$ . Moreover,  $\widehat{A}_n^{\beta}(\mathcal{F}_{\mathcal{H}})$  requires the knowledge of the labels so the unlabeled samples cannot be exploited for improving the MS strategy. Other extensions to real-valued functions of the VC theory have been proposed in [98, 99], but their applications in real world are not feasible.

### Rademacher Complexity Theory

By exploiting the same notation adopted in “**Out-of-Sample Methods**” section and by noting again that  $\ell_H \leq \ell_S$  we can state, thanks to the result of “**(Local) Rademacher Complexity**” section, that with probability  $(1 - \delta)$ :

$$\begin{aligned} L^{\ell_H}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) & \leq \widehat{L}_{\text{emp}}^{\ell_H}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) + \widehat{R}_n^{\ell_H}(\mathcal{F}_{\mathcal{H}}) \\ & + 3 \sqrt{\frac{\ln(\frac{2}{\delta})}{2n}} + 3 \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n}}, \forall \mathcal{H} \in \mathfrak{S}. \end{aligned}$$

and

$$\begin{aligned} L^{\ell_H}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) & \leq L^{\ell_S}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) \leq \widehat{L}_{\text{emp}}^{\ell_S}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) + \widehat{R}_n^{\ell_S}(\mathcal{F}_{\mathcal{H}}) \\ & + 3 \sqrt{\frac{\ln(\frac{2}{\delta})}{2n}} + 3 \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n}}, \forall \mathcal{H} \\ & \in \mathfrak{S}. \end{aligned} \quad (81)$$

When the ELM-NoR is exploited one should use the bound of Eq. (80) in order to control the generalization performance of the ELM, since no regularization is applied, while for ELM-R and ELM-SemiR the bound of Eq. (81) should be used. Unfortunately, computing the RC when the hard loss function is exploited results in an NP-hard problem [44, 80]. For this reason we can retrieve the Massart’s Lemma [100] which states that:

$$\widehat{R}_n^{\ell_H}(\mathcal{F}_{\mathcal{H}}) \leq \sqrt{\frac{2\widehat{A}_n(\mathcal{F}_{\mathcal{H}})}{n}} \leq \sqrt{\frac{2d_{\text{VC}} \ln(n)}{n}}. \quad (82)$$

By exploiting this result we retrieve the one reported in “**Vapnik–Chervonenkis Theory**” section for the VC theory.

For ELM-R and ELM-SemiR, instead, we use the  $\ell_S$ . In this case, we exploit the property of Eq. (48) which is proven in [80].

For ELM-R we can state that:

$$\begin{aligned} \widehat{R}_n^{\ell_S}(\mathcal{F}_{\mathcal{H}}) &\leq 4W \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}} \\ &= 4 \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\| \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}}. \end{aligned} \tag{83}$$

The Tikhonov regularization problem of Eq. (7):

$$w^* = \arg \min_w \|Hw - y\|^2 + \lambda \|w\|^2, \tag{84}$$

is equivalent to an Ivanov-based one [69, 101, 102]

$$w^* = \arg \min_w \|Hw - y\|^2, \quad \text{s.t. } \|w\|^2 \leq W^2, \tag{85}$$

for a suitable value of  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\|$ . Note that this bound can be used also for ELM-NoR by exploiting the soft loss function instead of the hard one, but since no regularization is applied in ELM-NoR  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\|$  can assume any value. In fact, for ELM-R from  $\lambda = \infty$  we have that  $W = 0$  while for  $\lambda = 0$  we retrieve the ELM-NoR.

For ELM-SemiR we can state that:

$$\begin{aligned} \widehat{R}_n^{\ell_S}(\mathcal{F}_{\mathcal{H}}) &\leq 4W \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}} \\ &= 4 \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} - \lambda_1 w_0 \right\| \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}}. \end{aligned} \tag{86}$$

The Tikhonov regularization problem of Eq. (11):

$$w^* = \arg \min_w \|Hw - y\|^2 + \lambda_1 \|w - \lambda_2 w_0\|^2, \tag{87}$$

is equivalent to an Ivanov-based one:

$$w^* = \arg \min_w \|Hw - y\|^2, \quad \text{s.t. } \|w - \lambda_2 w_0\|^2 \leq W, \tag{88}$$

for a suitable value of  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} - \lambda_1 w_0 \right\|$ .

Based on these results we can propose the RC-based MS for ELM-R and ELM-SemiR:

$w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}}, \mathcal{H}^*$  :

$$\arg \min_{\mathcal{H} \in \mathcal{S}} \widehat{L}_{\text{emp}}^{\ell_S}(w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}}) + 4W \sqrt{\frac{\sum_{i=1}^n \phi(x_i)^T \phi(x_i)}{n}}. \tag{89}$$

where for ELM-R  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\|$  while for ELM-SemiR, which exploits also the unlabeled samples,  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} - \lambda_1 w_0 \right\|$ .

In order to exploit the unlabeled samples also for the MS process, we have to exploit a recent result reported in [103, 104] which states that:

$$\begin{aligned} L(f) &\leq \widehat{L}_{\text{emp}}(f) + \frac{1}{m} \sum_{i=j}^m \widehat{R}_n^j(\mathcal{F}_{\mathcal{H}}) + 3 \sqrt{\frac{\ln(\frac{2}{\delta})}{2n}} \\ &\quad + \frac{2 + \sqrt{m}}{\sqrt{m}} \sqrt{\frac{\ln(n_{\mathcal{H}})}{2n}}, \end{aligned} \tag{90}$$

$$\forall f \in \mathcal{F}_{\mathcal{H}} \in \mathfrak{F},$$

where  $m = \lfloor n + n_u/n \rfloor$ ,  $\mathcal{D}_n \cup \mathcal{D}_{n_u} = \{x_1, \dots, x_{n+n_u}\}$  and

$$\frac{1}{m} \sum_{j=1}^m \widehat{R}_n^j(\mathcal{F}_{\mathcal{H}}) = \frac{1}{m} \sum_{j=1}^m \mathbb{E}_{\sigma} \sup_{f \in \mathcal{F}_{\mathcal{H}}} \frac{2}{n} \sum_{i=1}^n \sigma_i \ell(f, z_{(j-1)m+i}). \tag{91}$$

Note that the bound of Eq. (90) is tighter than the one of Eq. (42) since we have a better estimation of the RC thanks to the unlabeled samples. Basically, the unlabeled samples give us the ability of computing the average over  $m$  different realizations of the RC. Based on the previous results we can state that if we use the soft loss function, for the ELM-R we have that:

$$\begin{aligned} &\frac{1}{m} \sum_{j=1}^m \left( \widehat{R}_n^j(\mathcal{F}_{\mathcal{H}}) \right)^{\ell_S} \\ &\leq \frac{1}{m} \sum_{j=1}^m 4 \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\| \sqrt{\frac{\sum_{i=1}^n \phi(x_{(j-1)m+i})^T \phi(x_{(j-1)m+i})}{n}}, \end{aligned} \tag{92}$$

while for the ELM-SemiR

$$\begin{aligned} &\frac{1}{m} \sum_{j=1}^m \left( \widehat{R}_n^j(\mathcal{F}_{\mathcal{H}}) \right)^{\ell_S} \\ &\leq \frac{1}{m} \sum_{j=1}^m 4 \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\| \\ &\quad - \lambda_1 w_0 \left\| \sqrt{\frac{\sum_{i=1}^n \phi(x_{(j-1)m+i})^T \phi(x_{(j-1)m+i})}{n}}. \end{aligned} \tag{93}$$

Based on this result we can propose the RC-based MS for ELM-R and ELM-SemiR which takes into account also the unlabeled samples:

$w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}}, \mathcal{H}^*$  :

$$\begin{aligned} &\arg \min_{\mathcal{H} \in \mathcal{S}} \widehat{L}_{\text{emp}}^{\ell_S}(w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}}) \\ &\quad + \frac{1}{m} \sum_{j=1}^m 4W \sqrt{\frac{\sum_{i=1}^n \phi(x_{(j-1)m+i})^T \phi(x_{(j-1)m+i})}{n}}, \end{aligned} \tag{94}$$

where  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} \right\|$  for ELM-R and  $W = \left\| w_{\mathcal{H}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}} - \lambda_1 w_0 \right\|$  for ELM-SemiR.

Also for the LRC it has been proved that the unlabeled samples can improve the reliability of the estimation of the

generalization error of the model [47] but, unfortunately, the open question remains how to effectively compute the LRC in practice.

### Algorithmic Stability Theory

In order to apply the AS to ELM we have to exploit some known results and prove some new ones. We use the same notation of “Out-of-Sample Methods” section.

Let us start with the US. If we use the hard loss function, it is straightforward to prove that for any ELM we have [43, 46, 56]:

$$(\beta^i)^{\ell_H} = (\beta^i)^{\ell_S} = 1, \tag{95}$$

which is the only possible, and trivial, result. If instead we use the soft loss function in [55] it is proved that for a Tikhonov regularization problem like the ELM-R we have that:

$$(\beta^i)^{\ell_S} \leq 8 \frac{\max\{\phi(x_1)^T \phi(x_1), \dots, \phi(x_n)^T \phi(x_n)\}}{n\lambda}. \tag{96}$$

Note that ELM-R is equal to ELM-NoR if  $\lambda \rightarrow 0$ , which results in  $(\beta^i)^{\ell_S} \rightarrow +\infty$ . This means that without regularization the ELM is not stable. For the ELM-SemiR we have that:

$$(\beta^i)^{\ell_S} \leq 8 \frac{\max\{\phi(x_1)^T \phi(x_1), \dots, \phi(x_n)^T \phi(x_n)\}}{n\lambda_1}. \tag{97}$$

Consequently, we can use the US just for ELM-R and ELM-SemiR and by exploiting the results of “Algorithmic Stability Theory” section we can state that:

$$w_{\mathcal{H}^*}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathcal{S}} \widehat{L}_{\text{emp}}^{\ell_S}(w_{\mathcal{H}}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}) + 2(\beta^i)^{\ell_S}, \tag{98}$$

$$w_{\mathcal{H}^*}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathcal{S}} \widehat{L}_{\text{loo}}^{\ell_S}(\mathcal{A}_{\mathcal{H}}) + (\beta^i)^{\ell_S}. \tag{99}$$

Note that  $\beta^i$  must be computed based on Eqs. (96) and (97), respectively, for ELM-R and ELM-SemiR. Note that there is no advantage in having unlabeled samples.

With the HS the approach is quite different. In this case, we can exploit the hard loss function as described also in [43]. Since  $\beta_{\text{emp}}(\mathcal{A}_{\mathcal{H}}, n)$  cannot be estimated from the data [43] we can just use the bound which takes into the LOO error [see Eq. (52)]. In order to compute the  $\beta_{\text{loo}}(\mathcal{A}_{\mathcal{H}}, n)$  of Eq. (56) for ELM with the hard loss function we have to take Eq. (56) and note that with probability  $(1 - \delta)$ :

$$\begin{aligned} \beta_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, n) &\leq \widehat{\beta}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}/2) \\ &= \frac{8}{n\sqrt{n}} \sum_{i,j,k=1}^{\sqrt{n}/2} |\ell^H(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}/2}^k \cup \mathcal{D}_{n_u}, \mathcal{H})}, \check{z}_j^k) \\ &\quad - \ell^H(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}/2}^{k_i} \cup \mathcal{D}_{n_u}, \mathcal{H})}, \check{z}_j^k)| + \sqrt{\frac{\ln(\frac{1}{\delta})}{\sqrt{n}}} \\ &= \frac{8}{n\sqrt{n}} \sum_{i,j,k=1}^{\sqrt{n}/2} \left[ \text{sign} \left[ \left[ w_{\mathcal{H}}^{\mathcal{D}_{\sqrt{n}/2}^k \cup \mathcal{D}_{n_u}} \right]^T \check{x}_j^k \right] \right. \\ &\quad \left. \neq \text{sign} \left[ \left[ w_{\mathcal{H}}^{\mathcal{D}_{\sqrt{n}/2}^{k_i} \cup \mathcal{D}_{n_u}} \right]^T \check{x}_j^k \right] \right] + \sqrt{\frac{\ln(\frac{1}{\delta})}{\sqrt{n}}}. \end{aligned} \tag{100}$$

Note that  $\widehat{\beta}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}/2)$  can be computed from the data and so by applying the bound of Eq. (52) we have the HS MS strategy<sup>2</sup>:

$$w_{\mathcal{H}^*}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathcal{S}} \widehat{L}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}) + \widehat{\beta}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}/2). \tag{101}$$

This approach can be applied to ELM-NoR, ELM-R, or ELM-SemiR.

It can also be shown that the bound of Eq. (56), as well as the MS strategy, can be improved, if some unlabeled data are available. In particular, from Eq. (100) it is possible to note that, if the hard loss function is exploited,  $\widehat{\beta}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}/2)$  does not require the knowledge of the labels. In particular, let us suppose to have at least  $n_u = n$  unlabeled data, since for the ELM  $\beta_{\text{loo}}(\mathcal{A}_{\mathcal{H}}, n)$  decreases with  $n$  we have that:

$$\beta_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, n) \leq \beta_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}). \tag{102}$$

Let us define now the following quantity:

$$\begin{aligned} \widehat{\beta}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}) &= \frac{1}{n\sqrt{n}} \sum_{i,j,k=1}^{\sqrt{n}} |\ell^H(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}}^k \cup \mathcal{D}_{n_u}, \mathcal{H})}, \check{z}_j^k) \\ &\quad - \ell^H(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}}^{k_i} \cup \mathcal{D}_{n_u}, \mathcal{H})}, \check{z}_j^k)| + \sqrt{\frac{\ln(\frac{1}{\delta})}{n}}, \end{aligned} \tag{103}$$

where

$$\begin{aligned} \check{\mathcal{D}}_{\sqrt{n}}^k &: \{z_{(k-1)\sqrt{n}+1}, \dots, z_{(k-1)\sqrt{n}+\sqrt{n}}\}, k \in \{1, \dots, \sqrt{n}\}, z \\ &\in \mathcal{D}_n \end{aligned} \tag{104}$$

<sup>2</sup> We have exploited the property  $\sqrt{a^2b} \leq \frac{a}{2} + b$  in order to remove all the constant terms which do not depend on  $\widehat{\beta}_{\text{loo}}^{\ell_H}(\mathcal{A}_{\mathcal{H}}, \sqrt{n}/2)$ .

$$z_j^k : z_{(k-1)\sqrt{n}+j}, \quad k \in \{1, \dots, \sqrt{n}\}, z \in \mathcal{D}_{n_u}. \tag{105}$$

Note that the label in  $\mathcal{D}_{n_u}$  are unknown but, if the hard loss function is used, we have that:

$$\begin{aligned} & \widehat{\beta}_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}, \sqrt{n}) \\ &= \frac{1}{n\sqrt{n}} \sum_{i,j,k=1}^{\sqrt{n}} |\ell^H(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}}^k \cup \mathcal{D}_{n_u}, \mathcal{H})}, z_j^k) \\ & - \ell^H(\mathcal{A}_{(\mathcal{D}_{\sqrt{n}}^{k_i} \cup \mathcal{D}_{n_u}, \mathcal{H})}, z_j^k)| + \sqrt{\frac{\ln(\frac{1}{\delta})}{n}} \\ &= \frac{1}{n\sqrt{n}} \sum_{i,j,k=1}^{\sqrt{n}} \left[ \text{sign} \left[ \left[ w_{\mathcal{H}^{\sqrt{n} \cup \mathcal{D}_{n_u}}}^k \right]^T x_j^k \right] \right. \\ & \neq \text{sign} \left[ \left[ w_{\mathcal{H}^{\sqrt{n} \cup \mathcal{D}_{n_u}}}^{k_i} \right]^T x_j^k \right] \left. \right], \end{aligned} \tag{106}$$

which does not require the knowledge of the labels. Moreover,  $\widehat{\beta}_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}, \sqrt{n})$  is an empirical unbiased estimator of  $\beta_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}, \sqrt{n})$  (based on the same reasoning proposed in [43]) and therefore thanks to the Hoeffding’s inequality we can state that:

$$\beta_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}, \sqrt{n}) \leq \widehat{\beta}_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}, \sqrt{n}) + \sqrt{\frac{\ln(\frac{1}{\delta})}{2\sqrt{n}}}. \tag{107}$$

By plugging these results into the bound of Eq. (52) and by following the procedure adopted for deriving Eq. (101), we can derive the HS MS strategy which takes advantage also of the unlabeled samples:

$$w_{\mathcal{H}^*}^{\mathcal{D}_n \cup \mathcal{D}_{n_u}}, \mathcal{H}^* : \arg \min_{\mathcal{H} \in \mathcal{S}} \widehat{L}_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}) + \widehat{\beta}_{100}^{\ell_H}(\mathcal{A}, \mathcal{H}, \sqrt{n}). \tag{108}$$

This approach can be applied to ELM-NoR, ELM-R, or ELM-SemiR.

### Affective Analogical Reasoning Dataset

#### The AffectiveSpace model

AffectNet is a semantic network in which common-sense concepts (e.g., ‘read book,’ ‘payment,’ ‘play music’) are linked to a hierarchy of affective domain labels (e.g., ‘joy,’ ‘amazement,’ ‘fear,’ ‘admiration’). In order to enable affective analogical reasoning on natural language concepts, AffectiveSpace [13] is obtained as the vector space representation of such a semantic network. Therefore, in AffectiveSpace, concepts conveying similar semantic and affective information, e.g., ‘enjoy conversation’ and ‘chat

with friends,’ tend to fall near each other in the multi-dimensional space.

Both AffectNet and AffectiveSpace are publicly available at <http://sentic.net>. AffectiveSpace has been obtained applying principal component analysis (PCA) on the matrix representation of AffectNet. Due to computational cost issues, truncated singular value decomposition (TSVD) has been preferred to other dimensionality reduction techniques. TSVD uses an orthogonal transformation to convert the set of common-sense features associated with each concept into a set of uncorrelated variables (the principal components of the SVD).

Indicating AffectNet as  $A$ , a low-rank approximation of it is obtained:  $\tilde{A} = U_M \Sigma_M V_M^T$ . This approximation is based on minimizing the Frobenius norm of the difference between  $A$  and  $\tilde{A}$ , under the constraint  $rank(\tilde{A}) = M$ ; according to the Eckart–Young theorem [105], this represents the best approximation of  $A$  in the least-square sense:

$$\begin{aligned} \min_{\tilde{A} | rank(\tilde{A})=M} |A - \tilde{A}| &= \min_{\tilde{A} | rank(\tilde{A})=M} |\Sigma - U^T \tilde{A} V| \\ &= \min_{\tilde{A} | rank(\tilde{A})=M} |\Sigma - S|, \end{aligned} \tag{109}$$

assuming that  $\tilde{A} = USV^T$ , where  $S$  is diagonal and has  $M$  nonzero diagonal entries for the rank constraint. The minimum of the above equation may be obtained as follows:

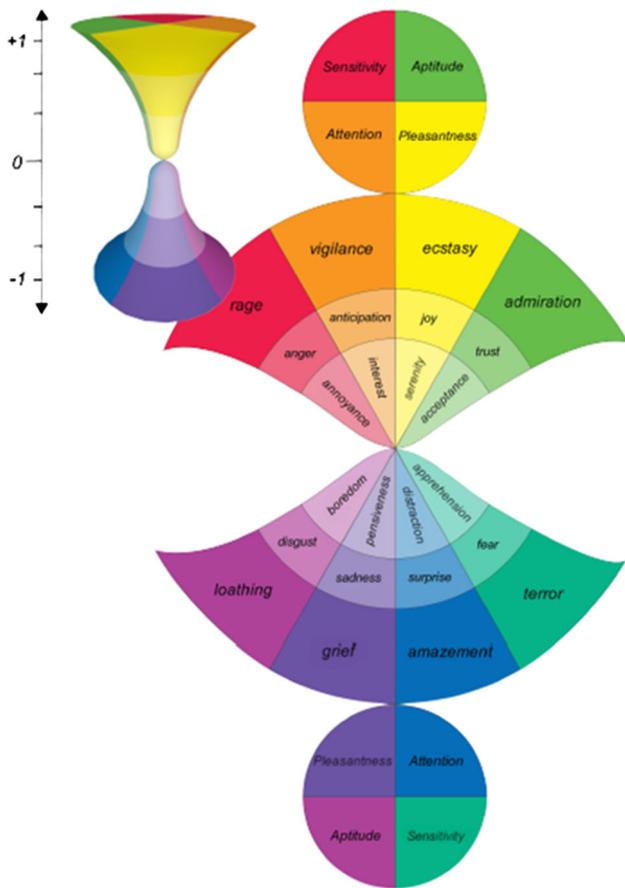
$$\begin{aligned} \min_{\tilde{A} | rank(\tilde{A})=M} |\Sigma - S| &= \min_{s_i} \sqrt{\sum_{i=1}^n (\sigma_i - s_i)^2} \\ &= \min_{s_i} \sqrt{\sum_{i=1}^M (\sigma_i - s_i)^2 + \sum_{i=M+1}^n \sigma_i^2} \\ &= \sqrt{\sum_{i=M+1}^n \sigma_i^2}. \end{aligned} \tag{110}$$

In fact, in the Frobenius norm sense the minimum is obtained when  $\sigma_i = s_i$  ( $i = 1, \dots, M$ ) and the corresponding singular vectors are the same as those of  $A$ . Thus, if only the first  $M$  principal components are kept, common-sense concepts are represented by vectors of  $M$  coordinates.

As already said, concepts with the same affective orientation are likely to have similar features; i.e., concepts conveying the same emotion tend to fall near each other in AffectiveSpace. Concept similarity does not depend on their absolute positions in the vector space, but rather on the angle they make with the origin, as it can be seen in Fig. 1.

The number of singular values  $M$ , which indicates the dimensionality of the AffectiveSpace, represents the trade-off between efficiency and precision: The bigger is  $M$ , the





**Fig. 2** The 3D model of the hourglass of emotions. Since affective states go from strongly positive to null to strongly negative, the model assumes a hourglass shape

### Dataset Description

The proposed MS framework was tested on a benchmark of 23,244 common-sense concepts. Each concept is represented according to the AffectiveSpace model discussed in “The AffectiveSpace Model” section, with dimension M equal to 50 and 100.

The publicly available Sentic API (on <http://sentic.net/api>) was used to obtain for each concept the level of activation for each affective dimension. According to the hourglass model presented in “The AffectiveSpace Model” section, the Sentic API expresses the levels of activation as an analog number in the range  $[-1, 1]$ , which are eventually mapped into the associated polarity according to equation Eq. (111). Only 6813 concepts of the benchmark are labeled, while the others are left unlabeled.

### Experimental Results

In this section, we compare the performance of different ELMs (ELM-noR, ELM-R, and ELM-SemiR) over the dataset described in “Dataset Description” section, tuned with the different MS strategies described in “Model Selection” section. In particular, for the ELMs we have that:

- ELM-noR: The set of possible configurations of hyperparameters is every possible combination of the hyperparameters such that  $\mathfrak{S} = \{N_h : N_h \in \{100; 250; 500; 750; 1000\}\}$

**Table 1** MS methods and ELM: the ‘×’ indicates if the MS method can be applied to the particular ELM

MS method	Sections	Equations	Suitable for		
			ELM-noR	ELM-R	ELM-SemiR
HO	Out-of-Sample Methods	(65)	×	×	×
HO-SEMI	Out-of-Sample Methods	(71)	×	×	×
KCV	Out-of-Sample Methods	(65)	×	×	×
KCV-SEMI	Out-of-Sample Methods	(71)	×	×	×
BOO	Out-of-Sample Methods	(65)	×	×	×
BOO-SEMI	Out-of-Sample Methods	(71)	×	×	×
VC-DIMENSION	Vapnik–Chervonenkis Theory	(74)	×	–	–
VC-ENTROPY	Vapnik–Chervonenkis Theory	(75)	×	–	–
VC-ENTROPY-SEMI	Vapnik–Chervonenkis Theory	(79)	×	–	–
RC	Rademacher Complexity Theory	(89)	–	×	×
RC-SEMI	Rademacher Complexity Theory	(94)	–	×	×
US <sub>EMP</sub>	Algorithmic Stability Theory	(98)	–	×	×
US <sub>LOO</sub>	Algorithmic Stability Theory	(99)	–	×	×
HS	Algorithmic Stability Theory	(101)	×	×	×
HS-SEMI	Algorithmic Stability Theory	(108)	×	×	×

**Table 2** Dimensionality M=50

Task	Polarity			Pleasantness			Attention			Sensitivity			Aptitude		
	no-R	R	SemiR	no-R	R	SemiR	no-R	R	SemiR	no-R	R	SemiR	no-R	R	SemiR
MS	ELM														
<i>Error on the reference set (percentage)</i>															
HO	16.2 ± 0.2	14.6 ± 0.1	14.3 ± 0.1	18.7 ± 0.2	17.1 ± 0.1	16.5 ± 0.1	13.5 ± 0.2	12.7 ± 0.1	12.1 ± 0.1	16.8 ± 0.2	13.3 ± 0.1	13.0 ± 0.1	12.0 ± 0.2	11.1 ± 0.1	11.0 ± 0.1
HO-SEMI	16.0 ± 0.2	14.3 ± 0.1	14.1 ± 0.1	18.5 ± 0.2	17.0 ± 0.1	16.3 ± 0.1	13.2 ± 0.2	12.4 ± 0.1	11.8 ± 0.1	16.3 ± 0.2	12.8 ± 0.1	12.5 ± 0.1	11.6 ± 0.2	11.0 ± 0.1	10.5 ± 0.1
KCV	15.9 ± 0.1	13.5 ± 0.1	13.1 ± 0.1	17.8 ± 0.1	16.8 ± 0.1	16.2 ± 0.1	13.0 ± 0.1	12.4 ± 0.1	11.9 ± 0.1	16.1 ± 0.1	13.0 ± 0.1	12.7 ± 0.1	11.8 ± 0.1	11.1 ± 0.1	10.4 ± 0.1
KCV-SEMI	15.7 ± 0.1	13.2 ± 0.1	12.8 ± 0.1	17.3 ± 0.1	16.2 ± 0.1	15.8 ± 0.1	12.2 ± 0.1	12.0 ± 0.1	11.3 ± 0.1	16.0 ± 0.1	12.8 ± 0.1	12.1 ± 0.1	11.1 ± 0.1	10.8 ± 0.1	10.1 ± 0.1
BOO	15.5 ± 0.1	13.1 ± 0.1	12.8 ± 0.1	17.1 ± 0.1	16.1 ± 0.1	15.8 ± 0.1	12.7 ± 0.1	12.0 ± 0.1	11.7 ± 0.1	15.8 ± 0.1	12.7 ± 0.1	12.3 ± 0.1	11.6 ± 0.1	10.7 ± 0.1	10.1 ± 0.1
BOO-SEMI	15.1 ± 0.1	12.8 ± 0.1	12.1 ± 0.1	17.0 ± 0.1	15.7 ± 0.1	15.2 ± 0.1	12.0 ± 0.1	11.7 ± 0.1	11.1 ± 0.1	15.4 ± 0.1	12.2 ± 0.1	11.7 ± 0.1	10.9 ± 0.1	10.4 ± 0.1	9.8 ± 0.1
VC-	18.1 ± 0.2	-	-	19.7 ± 0.2	-	-	15.0 ± 0.2	-	-	17.1 ± 0.2	-	-	12.9 ± 0.2	-	-
DIMENSION															
VC-ENTROPY	16.2 ± 0.2	-	-	18.5 ± 0.2	-	-	13.0 ± 0.2	-	-	16.3 ± 0.2	-	-	11.8 ± 0.2	-	-
VC-	16.0 ± 0.1	-	-	17.2 ± 0.1	-	-	12.1 ± 0.1	-	-	16.1 ± 0.1	-	-	11.2 ± 0.1	-	-
ENTROPY-SEMI															
RC	-	13.0 ± 0.1	12.9 ± 0.1	-	15.9 ± 0.1	15.4 ± 0.1	-	11.7 ± 0.1	11.3 ± 0.1	-	12.2 ± 0.1	11.9 ± 0.1	-	10.2 ± 0.1	9.9 ± 0.1
RC-SEMI	-	12.7 ± 0.1	12.0 ± 0.1	-	15.7 ± 0.1	15.0 ± 0.1	-	11.5 ± 0.1	11.0 ± 0.1	-	11.9 ± 0.1	11.6 ± 0.1	-	10.0 ± 0.1	9.6 ± 0.1
US <sub>EMP</sub>	-	14.5 ± 0.1	13.9 ± 0.1	-	17.4 ± 0.1	16.8 ± 0.1	-	12.8 ± 0.1	11.4 ± 0.1	-	13.5 ± 0.1	13.3 ± 0.1	-	11.3 ± 0.1	11.0 ± 0.1
US <sub>Lo0</sub>	-	14.4 ± 0.1	13.8 ± 0.1	-	17.5 ± 0.1	16.9 ± 0.1	-	12.4 ± 0.1	11.5 ± 0.1	-	13.3 ± 0.1	13.2 ± 0.1	-	11.2 ± 0.1	11.1 ± 0.1
HS	15.0 ± 0.1	13.0 ± 0.1	12.1 ± 0.1	16.2 ± 0.1	15.8 ± 0.1	14.2 ± 0.1	12.3 ± 0.1	11.7 ± 0.1	10.8 ± 0.1	15.2 ± 0.1	12.2 ± 0.1	11.8 ± 0.1	10.9 ± 0.1	10.2 ± 0.1	9.8 ± 0.1
HS-SEMI	14.5 ± 0.1	12.1 ± 0.1	11.1 ± 0.1	16.1 ± 0.1	15.3 ± 0.1	14.0 ± 0.1	11.8 ± 0.1	11.1 ± 0.1	10.6 ± 0.1	14.9 ± 0.1	11.0 ± 0.1	10.6 ± 0.1	10.4 ± 0.1	9.9 ± 0.1	9.5 ± 0.1
<i>Time needed to build the model (s)</i>															
HO	1 ± 0	11 ± 1	797 ± 103	1 ± 0	11 ± 1	832 ± 133	1 ± 0	11 ± 1	851 ± 97	1 ± 0	10 ± 1	858 ± 102	1 ± 0	11 ± 2	851 ± 97
HO-SEMI	1 ± 0	11 ± 1	852 ± 108	1 ± 0	11 ± 1	786 ± 64	1 ± 0	11 ± 0	827 ± 94	1 ± 0	11 ± 2	821 ± 66	1 ± 0	11 ± 1	846 ± 89
KCV	6 ± 0	119 ± 14	1062 ± 149	6 ± 0	117 ± 13	1030 ± 103	6 ± 1	126 ± 15	1044 ± 141	6 ± 0	115 ± 15	1084 ± 114	6 ± 1	121 ± 20	1033 ± 119
KCV-SEMI	6 ± 1	122 ± 17	1017 ± 115	6 ± 1	121 ± 16	1094 ± 93	6 ± 0	127 ± 6	1071 ± 88	6 ± 1	111 ± 6	1033 ± 59	6 ± 0	114 ± 14	987 ± 55
BOO	12 ± 1	192 ± 17	1494 ± 184	12 ± 1	199 ± 22	1506 ± 158	12 ± 1	204 ± 19	1563 ± 148	12 ± 1	207 ± 16	1535 ± 201	12 ± 1	199 ± 29	1541 ± 124
BOO-SEMI	12 ± 1	201 ± 24	1528 ± 133	12 ± 1	205 ± 20	1587 ± 161	12 ± 1	206 ± 12	1599 ± 179	12 ± 2	200 ± 13	1490 ± 234	12 ± 2	203 ± 19	1471 ± 156
VC-	1 ± 0	-	-	1 ± 0	-	-	1 ± 0	-	-	1 ± 0	-	-	1 ± 0	-	-
DIMENSION															
VC-ENTROPY	12 ± 1	-	-	12 ± 1	-	-	12 ± 1	-	-	12 ± 1	-	-	11 ± 1	-	-
VC-	12 ± 2	-	-	12 ± 2	-	-	12 ± 1	-	-	12 ± 1	-	-	12 ± 1	-	-
ENTROPY-SEMI															
RC	-	11 ± 1	867 ± 75	-	11 ± 1	823 ± 37	-	11 ± 2	826 ± 60	-	11 ± 2	829 ± 51	-	11 ± 1	871 ± 93
RC-SEMI	-	11 ± 1	813 ± 84	-	11 ± 0	851 ± 92	-	11 ± 1	835 ± 52	-	12 ± 1	860 ± 53	-	11 ± 1	861 ± 93
US <sub>EMP</sub>	-	11 ± 1	850 ± 112	-	11 ± 1	823 ± 105	-	11 ± 1	813 ± 44	-	11 ± 1	838 ± 62	-	12 ± 1	880 ± 67
US <sub>Lo0</sub>	-	122 ± 8	1070 ± 158	-	116 ± 10	997 ± 112	-	122 ± 9	1114 ± 131	-	117 ± 19	999 ± 92	-	124 ± 14	1029 ± 78
HS	2 ± 0	23 ± 3	1013 ± 99	2 ± 0	23 ± 2	975 ± 73	2 ± 0	24 ± 2	1000 ± 87	2 ± 0	23 ± 2	960 ± 56	2 ± 0	23 ± 2	934 ± 78
HS-SEMI	2 ± 0	22 ± 2	1006 ± 75	2 ± 0	24 ± 2	952 ± 110	2 ± 0	22 ± 3	1035 ± 71	2 ± 0	22 ± 2	964 ± 185	2 ± 0	23 ± 1	1025 ± 105

**Table 3** Dimensionality M=100

Task	Polarity			Pleasantness			Attention			Sensitivity			Aptitude		
	no-R	R	SemiR	no-R	R	SemiR	no-R	R	SemiR	no-R	R	SemiR	no-R	R	SemiR
MS	ELM														
<i>Error on the reference set (percentage)</i>															
HO	15.5 ± 0.2	14.2 ± 0.1	14.0 ± 0.1	18.7 ± 0.2	17.1 ± 0.1	16.8 ± 0.1	13.2 ± 0.2	12.5 ± 0.1	11.7 ± 0.1	15.7 ± 0.2	14.3 ± 0.1	13.9 ± 0.1	11.6 ± 0.2	10.9 ± 0.1	10.5 ± 0.1
HO-SEMI	15.0 ± 0.2	14.1 ± 0.1	13.7 ± 0.1	18.3 ± 0.2	16.8 ± 0.1	16.3 ± 0.1	13.0 ± 0.2	12.2 ± 0.1	11.4 ± 0.1	15.2 ± 0.2	14.1 ± 0.1	13.4 ± 0.1	11.5 ± 0.2	10.1 ± 0.1	9.5 ± 0.1
KCV	15.1 ± 0.1	13.1 ± 0.1	13.0 ± 0.1	17.6 ± 0.1	16.4 ± 0.1	16.1 ± 0.1	13.0 ± 0.1	12.1 ± 0.1	11.5 ± 0.1	14.4 ± 0.1	14.1 ± 0.1	13.5 ± 0.1	11.3 ± 0.1	10.2 ± 0.1	9.3 ± 0.1
KCV-SEMI	14.8 ± 0.1	12.8 ± 0.1	12.3 ± 0.1	17.2 ± 0.1	16.1 ± 0.1	15.2 ± 0.1	12.7 ± 0.1	12.0 ± 0.1	11.1 ± 0.1	14.1 ± 0.1	13.7 ± 0.1	13.1 ± 0.1	11.0 ± 0.1	9.8 ± 0.1	9.1 ± 0.1
BOO	14.9 ± 0.1	12.9 ± 0.1	12.6 ± 0.1	16.9 ± 0.1	15.8 ± 0.1	15.1 ± 0.1	12.5 ± 0.1	11.7 ± 0.1	10.9 ± 0.1	14.0 ± 0.1	13.7 ± 0.1	13.2 ± 0.1	11.0 ± 0.1	9.8 ± 0.1	9.1 ± 0.1
BOO-SEMI	14.2 ± 0.1	12.2 ± 0.1	11.6 ± 0.1	16.1 ± 0.1	15.2 ± 0.1	14.2 ± 0.1	12.1 ± 0.1	11.2 ± 0.1	10.1 ± 0.1	13.9 ± 0.1	13.2 ± 0.1	12.2 ± 0.1	10.7 ± 0.1	9.1 ± 0.1	8.9 ± 0.1
VC-	15.3 ± 0.2	-	-	19.1 ± 0.2	-	-	13.2 ± 0.2	-	-	15.9 ± 0.2	-	-	13.1 ± 0.2	-	-
DIMENSION															
VC-ENTROPY	15.1 ± 0.2	-	-	17.6 ± 0.2	-	-	13.0 ± 0.2	-	-	15.2 ± 0.2	-	-	11.3 ± 0.2	-	-
VC-	14.4 ± 0.1	-	-	17.1 ± 0.1	-	-	12.9 ± 0.1	-	-	14.5 ± 0.1	-	-	10.9 ± 0.1	-	-
ENTROPY-SEMI															
RC	-	12.4 ± 0.1	12.0 ± 0.1	-	14.8 ± 0.1	14.1 ± 0.1	-	11.4 ± 0.1	10.5 ± 0.1	-	13.3 ± 0.1	12.2 ± 0.1	-	9.5 ± 0.1	8.9 ± 0.1
RC-SEMI	-	12.1 ± 0.1	10.9 ± 0.1	-	14.2 ± 0.1	13.6 ± 0.1	-	11.1 ± 0.1	10.0 ± 0.1	-	13.1 ± 0.1	11.8 ± 0.1	-	9.0 ± 0.1	8.6 ± 0.1
US <sub>EMP</sub>	-	14.3 ± 0.1	14.0 ± 0.1	-	17.3 ± 0.1	17.2 ± 0.1	-	13.1 ± 0.1	12.9 ± 0.1	-	14.2 ± 0.1	14.0 ± 0.1	-	10.8 ± 0.1	10.7 ± 0.1
US <sub>LOO</sub>	-	14.2 ± 0.1	14.2 ± 0.1	-	17.2 ± 0.1	17.1 ± 0.1	-	13.0 ± 0.1	12.8 ± 0.1	-	14.1 ± 0.1	14.0 ± 0.1	-	10.8 ± 0.1	10.6 ± 0.1
HS	14.1 ± 0.1	12.1 ± 0.1	11.4 ± 0.1	14.9 ± 0.1	14.1 ± 0.1	13.9 ± 0.1	11.6 ± 0.1	11.1 ± 0.1	10.2 ± 0.1	13.3 ± 0.1	13.0 ± 0.1	12.8 ± 0.1	10.1 ± 0.1	9.2 ± 0.1	8.9 ± 0.1
HS-SEMI	13.2 ± 0.1	12.1 ± 0.1	10.7 ± 0.1	15.0 ± 0.1	13.7 ± 0.1	13.2 ± 0.1	10.9 ± 0.1	10.5 ± 0.1	9.9 ± 0.1	12.8 ± 0.1	12.6 ± 0.1	11.8 ± 0.1	9.8 ± 0.1	9.0 ± 0.1	8.5 ± 0.1
<i>Time needed to build the model (s)</i>															
HO	1 ± 0	14 ± 1	999 ± 105	1 ± 0	14 ± 1	1023 ± 104	1 ± 0	13 ± 1	1028 ± 87	1 ± 0	13 ± 1	1005 ± 135	1 ± 0	13 ± 1	990 ± 107
HO-SEMI	1 ± 0	13 ± 1	1016 ± 76	1 ± 0	13 ± 2	1051 ± 91	1 ± 0	13 ± 2	1034 ± 59	1 ± 0	13 ± 1	991 ± 84	1 ± 0	13 ± 1	1021 ± 124
KCV	7 ± 1	147 ± 17	1178 ± 118	7 ± 1	143 ± 21	1270 ± 165	7 ± 0	146 ± 15	1277 ± 110	7 ± 1	141 ± 12	1323 ± 64	7 ± 1	141 ± 15	1303 ± 172
KCV-SEMI	7 ± 1	154 ± 18	1237 ± 126	8 ± 1	142 ± 19	1263 ± 186	7 ± 1	145 ± 9	1195 ± 170	7 ± 1	148 ± 17	1246 ± 115	7 ± 1	141 ± 13	1241 ± 83
BOO	15 ± 1	236 ± 20	1881 ± 194	15 ± 1	237 ± 37	1874 ± 214	15 ± 2	237 ± 23	1873 ± 203	14 ± 1	236 ± 16	1871 ± 135	14 ± 1	246 ± 28	1922 ± 178
BOO-SEMI	14 ± 2	241 ± 18	1722 ± 221	14 ± 2	240 ± 12	1854 ± 119	14 ± 2	243 ± 38	1740 ± 140	14 ± 1	227 ± 33	1873 ± 138	15 ± 2	247 ± 13	1769 ± 77
VC-	1 ± 0	-	-	1 ± 0	-	-	1 ± 0	-	-	1 ± 0	-	-	1 ± 0	-	-
DIMENSION															
VC-ENTROPY	14 ± 1	-	-	15 ± 1	-	-	15 ± 1	-	-	15 ± 2	-	-	14 ± 1	-	-
VC-	14 ± 1	-	-	14 ± 1	-	-	15 ± 1	-	-	15 ± 2	-	-	15 ± 1	-	-
ENTROPY-SEMI															
RC	-	13 ± 2	1017 ± 32	-	13 ± 2	993 ± 91	-	13 ± 1	993 ± 100	-	14 ± 1	1028 ± 85	-	13 ± 2	1002 ± 114
RC-SEMI	-	14 ± 2	957 ± 107	-	13 ± 1	1003 ± 119	-	13 ± 2	966 ± 102	-	13 ± 2	977 ± 102	-	14 ± 1	948 ± 123
US <sub>EMP</sub>	-	13 ± 1	993 ± 57	-	13 ± 2	1022 ± 124	-	14 ± 2	988 ± 104	-	13 ± 1	973 ± 57	-	12 ± 1	1020 ± 143
US <sub>LOO</sub>	-	137 ± 16	1274 ± 136	-	141 ± 10	1235 ± 49	-	145 ± 14	1246 ± 120	-	143 ± 16	1271 ± 161	-	148 ± 19	1296 ± 187
HS	2 ± 0	28 ± 1	1201 ± 176	2 ± 0	26 ± 3	1150 ± 106	2 ± 0	27 ± 3	1131 ± 57	3 ± 0	28 ± 4	1209 ± 25	2 ± 0	27 ± 2	1190 ± 48
HS-SEMI	2 ± 0	27 ± 4	1187 ± 58	2 ± 0	27 ± 3	1196 ± 149	2 ± 0	27 ± 2	1195 ± 168	2 ± 0	27 ± 1	1206 ± 107	2 ± 0	28 ± 2	1178 ± 119

- ELM-R: The set of possible configurations of hyperparameters is every possible combination of the hyperparameters such that  $\mathfrak{S} = \{N_h : N_h \in \{100; 250; 500; 750; 1000\}, \lambda \in 10^{\{-6, -5.5, \dots, 2.5, 3\}}\}$
- ELM-SemiR: The set of possible configurations of hyperparameters is every possible combination of the hyperparameters such that  $\mathfrak{S} = \{N_h : N_h \in \{100; 250; 500; 750; 1000\}, \lambda_1 \in 10^{\{-6, -5.5, \dots, 2.5, 3\}}, \lambda_2 \in 10^{\{-6, -5.5, \dots, 2.5, 3\}}\}$

For the MS strategies, the possible options are several but some of them cannot be applied to every version of ELM exploited in this paper. Therefore, Table 1 reports on the match between MS methods and the type of ELM in which the method can be adopted. In Table 1 we refer to the methods with the following acronyms:

- HO: indicates the usual HO procedure where no unlabeled samples are exploited (see Eq. (65) in “Out-of-Sample Methods” section). Note that  $r = 1$ ,  $v = \lfloor 0.1n \rfloor$  and the resample procedure is performed without replacement;
- HO-SEMI: indicates the usual HO procedure where also the unlabeled samples are exploited (see Eq. (71) in “Out-of-Sample Methods” section). Note that we employed the same parameters of HO;
- KCV: indicates the  $k$ -fold cross-validation procedure where no unlabeled samples are exploited (see Eq. (65) in “Out-of-Sample Methods” section). Note that  $n_r = 10$ ,  $v = \lfloor 0.1n \rfloor$  ( $k = 10$ ) and the resample procedure is performed without replacement;
- KCV-SEMI: indicates the  $k$ -fold cross-validation procedure where also the unlabeled samples are exploited (see Eq. (71) in “Out-of-Sample Methods” section). Note that we employed the same parameters of KCV;
- BOO: indicates the bootstrap procedure where no unlabeled samples are exploited (see Eq. (65) in “Out-of-Sample Methods” section). Note that  $n_r = 30$ ,  $t = n$  ( $k = 10$ ) and the resample procedure is performed with replacement;
- BOO-SEMI: indicates the bootstrap procedure where also the unlabeled samples are exploited (see Eq. (71) in “Out-of-Sample Methods” section). Note that we employed the same parameters of BOO;
- VC-DIMENSION: exploits the VC dimension without employing the unlabeled samples (see Eq. (74) in “Vapnik–Chervonenkis Theory” section)
- VC-ENTROPY: exploits the VC entropy without employing the unlabeled samples (see Eq. (75) in “Vapnik–Chervonenkis Theory” section)
- VC-ENTROPY-SEMI: exploits the VC entropy with the employment of the unlabeled samples (see Eq. (79) in “Vapnik–Chervonenkis Theory” section)

- RC: exploits the Rademacher complexity without employing the unlabeled samples (see Eq. (89) in “Rademacher Complexity Theory” section)
- RC-SEMI: exploits the Rademacher complexity with the employment of the unlabeled samples (see Eq. (94) in “Rademacher Complexity Theory” section)
- US<sub>EMP</sub>: exploits the US and the empirical error without employing the unlabeled samples (see Eq. (98) in “Algorithmic Stability Theory” section)
- US<sub>LOO</sub>: exploits the US and the LOO error without employing the unlabeled samples (see Eq. (99) in “Algorithmic Stability Theory” section)
- HS: exploits the hypothesis stability without employing the unlabeled samples (see Eq. (101) in “Algorithmic Stability Theory” section)
- HS-SEMI: exploits the hypothesis stability with the employment of the unlabeled samples (see Eq. (108) in “Algorithmic Stability Theory” section)

The labeled data have been split into two sets: The first 5000 samples have been used for building the model with the different ELMs (ELM-noR, ELM-R, and ELM-SemiR) and with the different MS strategies (HO, HO-SEMI, KCV, KCV-SEMI, BOO, BOO-SEMI, VC-DIMENSION, VC-ENTROPY, VC-ENTROPY-SEMI, RC, RC-SEMI, US<sub>EMP</sub>, US<sub>LOO</sub>, HS, HS-SEMI) as reported in Table 1, while the rest of the labeled samples, which are 1813, have been kept apart as reference set in order to test the performance of the learned model. The splitting process has been repeated 30 times in order to obtain statistically relevant results.

The experiments have been performed on a Workstation equipped with one Solid State Drive disk of 100 GB, one Hard Disk Drive of 1 TB, 128 GB of RAM, 4 Intel Xeon CPU E5-4620 @2.20 GHz, and Windows Server 2012 R2. The code has been written in Fortran 90 and compiled with the Intel Parallel Studio XE 2016 Composer Edition.

In Tables 2 and 3 we reported the error on the reference set and the time needed to build the model for the different combination of ELM and MS strategies (see Table 1), for  $M = 50$  in Table 2 and  $M = 100$  in Table 3, for the five binary classification tasks (Polarity, Pleasantness, Attention, Sensitivity, and Aptitude).

From the results reported in Tables 2 and 3 it is possible to derive three main conclusions:

- With  $M = 100$  we retrieve models with generally higher accuracy with respect to  $M = 50$ . This is reasonable since the more information we feed the learning machine the more accurate results the final model.
- ELM-SemiR produces models with higher accuracy with respect to ELM-noR and ELM-R. This means that the algorithm is able to exploit and take advantage of

the hidden information given by the unlabeled samples. Anyway, note that ELM-SemiR requires more time to build the model because of the unsupervised pre-training phase.

- The MS strategies which exploit also the unlabeled samples (HO-SEMI, KCV-SEMI, BOO-SEMI, VC-ENTROPY-SEMI, RC-SEMI, HS-SEMI) select models with higher accuracy with respect to their counterparts where the unlabeled samples are not taken into account (HO, KCV, BOO, VC-DIMENSION, VC-ENTROPY, RC, US<sub>EMP</sub>, US<sub>LOO</sub>, HS). As expected from theory, the information hidden in the unlabeled samples helps to improve the performance of the MS strategy. Generally, the difference in terms of time to build the model between the MS methods which exploit the unlabeled samples and the ones which do not is not noticeable.

Besides these general considerations it is possible to derive some other interesting insights from the results of Tables 2 and 3 about the characteristics of each ELM and MS strategy.

- The in-sample methods (VC-DIMENSION, VC-ENTROPY, RC, US<sub>EMP</sub>, US<sub>LOO</sub>) usually perform worse, in terms of accuracy of the selected models, with respect to the out-of-sample ones (HO, KCV, BOO), when the unlabeled samples are not exploited. Anyway, the in-sample methods require less computational effort. The only exception is the HS method which generally produces models with higher accuracy than the in-sample methods.
- When the unlabeled data are exploited for MS purposes, the in-sample methods (RC-SEMI, HS-SEMI) produce models with higher accuracy compared to the out-of-sample ones (HO-SEMI, KCV-SEMI, BOO-SEMI), even if the models selected by the latter methods possess higher accuracy with respect to their counterparts when the unlabeled samples are not exploited (HO, KCV, BOO). The only exception is the VC-ENTROPY-SEMI, which selects more accurate models compared to the VC-ENTROPY but less accurate models than the ones selected with the out-of-sample methods.
- The out-of-sample method which selects the most accurate models is the bootstrap (BOO and BOO-SEMI), while the less accurate models are selected by the HO method (HO and HO-SEMI). This is due to the fact that the bootstrap represents the statistical method which extracts more information from data (as described in “Out-of-Sample Methods” section). In fact, the bootstrap is also the out-of-sample method which requires more time to build the model, while the

HO method is the most computational inexpensive out-of-sample method.

- The in-sample method which selects the most accurate models is the hypothesis stability (HS and HS-SEMI), while the less accurate models are selected by the US-based methods (US<sub>EMP</sub> and US<sub>LOO</sub>) together with the VC-based methods (VC-ENTROPY and VC-ENTROPY-SEMI). This is due to the fact that the VC dimension and the US techniques are not able to properly take into account the properties of the algorithms and the probability distributions that have generated the data (as described in “In-Sample Methods” section). Again, the in-sample method which selects the most accurate models is the also the one with the higher computational requirements.
- Overall, the HS-SEMI method is the one which selects the most accurate model while requiring less computational effort with respect to the out-of-sample methods.

Finally, we would like to stress that the proposed approach is quite general and can be applied in other applications and other learning algorithms.

## Conclusion

In this work, we have addressed the problem of exploiting unlabeled samples to perform an emotion recognition task. In particular, we have shown that the unlabeled samples can be exploited during the formulation of the learning algorithm with particular reference to the ELM. More in details, we proposed a different regularization procedure which is able to encapsulate an unsupervised pre-training hint in a form of a reference hyperplane into the ELM formulation. Moreover, we have shown that unlabeled samples can be extremely useful during another key phase of the learning process, the model selection phase, where the hyperparameters which influence the generalization performances of the learned model must be tuned based on the available data. In particular, we have reviewed all the most important state-of-the-art theoretical approaches to model selection and we have shown how to modify the theoretical framework in order to explicitly take advantage of the available information hidden in the unlabeled samples. The results performed on an affective analogical reasoning problem show that our method is indeed able to exploit the information given by unlabeled samples in order to build models with higher generalization performances with respect to the models built without exploiting them.

## Compliance with Ethical Standards

**Conflict of Interest** Luca Oneto, Federica Bisio, Erik Cambria, and Davide Anguita declare that they have no conflict of interest.

**Informed Consent** Informed consent was not required as no human or animals were involved.

**Human and Animal Rights** This article does not contain any studies with human or animal subjects performed by any of the authors.

## References

- Cambria E. Affective computing and sentiment analysis. *IEEE Intell Syst.* 2016;31(2):102–7.
- Saif H, He Y, Fernandez M, Alani H. Contextual semantics for sentiment analysis of twitter. *Inf Process Manag.* 2016;52(1):5–19.
- Xia R, Xu F, Yu J, Qi Y, Cambria E. Polarity shift detection, elimination and ensemble: a three-stage model for document-level sentiment analysis. *Inf Process Manag.* 2016;52(1):36–45.
- Balahur A, Jacquet G. Sentiment analysis meets social media-challenges and solutions of the field in view of the current information sharing context. *Inf Process Manag.* 2015;51(4):428–32.
- Google. Announcing syntaxnet: the world's most accurate parser goes open source. <http://googleresearch.blogspot.it/2016/05/announcing-syntaxnet-worlds-most.html>. 2016.
- Roy RS, Agarwal S, Ganguly N, Choudhury M. Syntactic complexity of web search queries through the lenses of language models, networks and users. *Inf Process Manag.* 2016;52(5):923–48.
- Abainia K, Ouamour S, Sayoud H. Effective language identification of forum texts based on statistical approaches. *Inf Process Manag.* 2016;52(4):491–512.
- Sun J, Wang G, Cheng X, Fu Y. Mining affective text to improve social media item recommendation. *Inf Process Manag.* 2015;51(4):444–57.
- Cambria E, Hussain A. Sentic computing: a common-sense-based framework for concept-level sentiment analysis. Switzerland: Cham; 2015.
- Poria S, Cambria E, Howard N, Huang G-B, Hussain A. Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing.* 2016;174:50–9.
- Wang Q, Cambria E, Liu C, Hussain A. Common sense knowledge for handwritten chinese recognition. *Cogn Comput.* 2013;5(2):234–42.
- Cambria E, Hussain A, Durrani T, Havasi C, Eckl C, Munro J. Sentic computing for patient centered application. In: *IEEE ICSP, Beijing*; 2010. p. 1279–82.
- Cambria E, Gastaldo P, Bisio F, Zunino R. An ELM-based model for affective analogical reasoning. *Neurocomputing.* 2015;149:443–55.
- Cambria E, Fu J, Bisio F, Poria S. AffectiveSpace 2: enabling affective intuition for concept-level sentiment analysis. In: *AAAI, Austin*; 2015. p. 508–14.
- Cambria E, Wang H, White B. Guest editorial: big social data analysis. *Knowl Based Syst.* 2014;69:1–2.
- Chakraborty M, Pal S, Pramanik R, Chowdary CR. Recent developments in social spam detection and combating techniques: a survey. *Inf Process Manag.* 2016;52(6):1053–73.
- Kranjc J, Smailović J, Podpečan V, Grčar M, Žnidaršič M, Lavrač N. Active learning for sentiment analysis on data streams: methodology and workflow implementation in the clowdfloows platform. *Inf Process Manag.* 2015;51(2):187–203.
- Fersini E, Messina E, Pozzi FA. Expressive signals in social media languages to improve polarity detection. *Inf Process Manag.* 2016;52(1):20–35.
- Cambria E, Livingstone A, Hussain A. The hourglass of emotions. In: Esposito A, Esposito AM, Vinciarelli A, Hoffmann R, Müller CC, editors. *Cognitive behavioural systems*. Berlin Heidelberg: Springer; 2012. p. 144–57.
- Huang G-B, Wang DH, Lan Y. Extreme learning machines: a survey. *Int J Mach Learn Cybern.* 2011;2(2):107–22.
- Huang G, Song S, Gupta JND, Wu C. Semi-supervised and unsupervised extreme learning machines. *IEEE Trans Cybern.* 2014;44(12):2405–17.
- Cambria E, Huang G-B, et al. Extreme learning machines. *IEEE Intell Syst.* 2013;28(6):30–59.
- Huang G-B, Cambria E, Toh K-A, Widrow B, Xu Z. New trends of learning in computational intelligence. *IEEE Comput Intell Mag.* 2015;10(2):16–7.
- Chapelle O, Schölkopf B, Zien A, et al. *Semi-supervised learning*. Cambridge: MIT Press; 2006.
- Zhu X. *Semi-supervised learning literature survey*. Madison: University of Wisconsin; 2005.
- Habernal I, Ptáček T, Steinberger J. Supervised sentiment analysis in Czech social media. *Inf Process Manag.* 2014;50(5):693–707.
- Guo Z, Zhang ZM, Xing EP, Faloutsos C. Semi-supervised learning based on semiparametric regularization, vol. 8. In: *SDM, SIAM*; 2008. p. 132–42.
- Belkin M, Niyogi P, Sindhvani V. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *J Mach Learn Res.* 2006;7:2399–434.
- Draper NR, Smith H, Pownell E. *Applied regression analysis*. New York: Wiley; 1966.
- MacKay DJC. Bayesian interpolation. *Neural Comput.* 1992;4(3):415–47.
- Breiman L. Statistical modeling: the two cultures (with comments and a rejoinder by the author). *Stat Sci.* 2001;16(3):199–231.
- Dhar V. Data science and prediction. *Commun ACM.* 2013;56(12):64–73.
- Vapnik VN. An overview of statistical learning theory. *IEEE Trans Neural Netw.* 1999;10(5):988–99.
- Wolpert DH. The lack of a priori distinctions between learning algorithms. *Neural Comput.* 1996;8(7):1341–90.
- Magdon-Ismail M. No free lunch for noise prediction. *Neural Comput.* 2000;12(3):547–64.
- Vapnik VN. *Statistical learning theory*. New York: Wiley-Interscience; 1998.
- Valiant LG. A theory of the learnable. *Commun ACM.* 1984;27(11):1134–42.
- Bartlett PL, Boucheron S, Lugosi G. Model selection and error estimation. *Mach Learn.* 2002;48(1–3):85–113.
- Langford J. Tutorial on practical prediction theory for classification. *J Mach Learn Res.* 2006;6(1):273.
- Anguita D, Ghio A, Oneto L, Ridella S. In-sample and out-of-sample model selection and error estimation for support vector machines. *IEEE Trans Neural Netw Learn Syst.* 2012;23(9):1390–406.
- Kohavi R, et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *International joint conference on artificial intelligence*; 1995.
- Efron B, Tibshirani RJ. *An introduction to the bootstrap*. London: Chapman & Hall; 1993.
- Oneto L, Ghio A, Ridella S, Anguita D. Fully empirical and data-dependent stability-based bounds. *IEEE Trans Cybern.* 2015;45(9):1913–1926.
- Anguita D, Ghio A, Oneto L, Ridella S. A deep connection between the Vapnik–Chervonenkis entropy and the Rademacher complexity. *IEEE Trans Neural Netw Learn Syst.* 2014;25(12):2202–11.
- Oneto, Ghio A, Ridella S, Anguita D. Global Rademacher complexity bounds: from slow to fast convergence rates. *Neural Process Lett.* 2016;43(2):567–602.
- Bartlett PL, Bousquet O, Mendelson S. Local Rademacher complexities. *Ann Stat.* 2005;33(4):1497–1537.

47. Oneto L, Ghio A, Ridella S, Anguita D. Local Rademacher complexity: sharper risk bounds with and without unlabeled samples. *Neural Netw.* 2015 (in press).
48. Lei Y, Binder A, ün Dogan U, Kloft M. Theory and algorithms for the localized setting of learning kernels. *Neural Inf Process Syst.* 2015;173–95. <http://www.jmlr.org/proceedings/papers/v44/LeiBinDogKlo15.pdf>.
49. McAllester DA. Some PAC-Bayesian theorems. In *Proceedings of the eleventh annual conference on Computational learning theory.* ACM; 1998. p. 230–234.
50. Lever G, Laviolette F, Shawe-Taylor J. Tighter PAC-Bayes bounds through distribution-dependent priors. *Theoret Comput Sci.* 2013;473:4–28.
51. Germain P, Lacasse A, Laviolette F, Marchand M, Roy JF. Risk bounds for the majority vote: from a PAC-Bayesian analysis to a learning algorithm. *J Mach Learn Res.* 2015;16(4):787–860.
52. Bégin L, Germain P, Laviolette F, Roy JF. PAC-Bayesian bounds based on the rényi divergence. In: *International conference on artificial intelligence and statistics*; 2016.
53. Floyd S, Warmuth M. Sample compression, learnability, and the Vapnik–Chervonenkis dimension. *Mach Learn.* 1995;21(3):269–304.
54. Langford J, McAllester DA. Computable shell decomposition bounds. In: *Proceedings of the eleventh annual conference on Computational learning theory*; 2000. p. 25–34.
55. Bousquet O, Elisseeff A. Stability and generalization. *J Mach Learn Res.* 2002;2:499–526.
56. Poggio T, Rifkin R, Mukherjee S, Niyogi P. General conditions for predictivity in learning theory. *Nature.* 2004;428(6981):419–22.
57. Guyon I, Saffari A, Dror G, Cawley G. Model selection: beyond the Bayesian/frequentist divide. *J Mach Learn Res.* 2010;11:61–87.
58. Huang GB. What are extreme learning machines? Filling the gap between Frank Rosenblatt’s dream and John von Neumann’s puzzle. *Cogn Comput.* 2015;7(3):263–78.
59. Huang Z, Yu Y, Gu J, Liu H. An efficient method for traffic sign recognition based on extreme learning machine. *IEEE Trans Cybern.* doi:10.1109/TCYB.2016.2533424.
60. Huang GB, Bai Z, Kasun LLC, Vong CM. Local receptive fields based extreme learning machine. *IEEE Comput Intell Mag.* 2015;10(2):18–29.
61. Huang G-B, Zhou H, Ding X, Zhang R. Extreme learning machine for regression and multiclass classification. *IEEE Trans Syst Man Cybern B Cybern.* 2012;42(2):513–29.
62. Bisio F, Decherchi S, Galstado P, Zunino R. Inductive bias for semi-supervised extreme learning machine, vol. 1. In: *Proceedings of ELM-2014*; 2015.
63. Dinuzzo F, Schölkopf B. The representer theorem for hilbert spaces: a necessary and sufficient condition. In: *Advances in neural information processing systems*; 2012. p. 189–196.
64. Schölkopf B, Herbrich R, Smola AJ. A generalized representer theorem. In: *International Conference on Computational Learning Theory.* Springer Berlin Heidelberg; 2001. p. 416–426.
65. Erhan D, Bengio Y, Courville A, Manzagol P-A, Vincent P, Bengio S. Why does unsupervised pre-training help deep learning? *J Mach Learn Res.* 2010;11:625–60.
66. Salakhutdinov R, Hinton G. An efficient learning procedure for deep boltzmann machines. *Neural Comput.* 2012;24(8):1967–2006.
67. Arlot S, Celisse A. A survey of cross-validation procedures for model selection. *Stat Surv.* 2010;4:40–79.
68. McAllester DA. PAC-Bayesian stochastic model selection. *Mach Learn.* 2003;51(1):5–21.
69. Anguita D, Ghio A, Oneto L, Ridella S. In-sample model selection for support vector machines. In: *International joint conference on neural networks*; 2011.
70. Koltchinskii V. Rademacher penalties and structural risk minimization. *IEEE Trans Inf Theory.* 2001;47(5):1902–14.
71. Inoue A, Kilian L. In-sample or out-of-sample tests of predictability: which one should we use? *Econom Rev.* 2005;23(4):371–402.
72. Cheng F, Yu J, Xiong H. Facial expression recognition in Jaffe dataset based on Gaussian process classification. *IEEE Trans Neural Netw.* 2010;21(10):1685–90.
73. Shalev-Shwartz S, Ben-David S. *Understanding machine learning: from theory to algorithms.* Cambridge: Cambridge University Press; 2014.
74. Hoeffding W. Probability inequalities for sums of bounded random variables. *J Am Stat Assoc.* 1963;58(301):13–30.
75. Anguita D, Ghio A, Ridella S, Sterpi D. K-fold cross validation for error rate estimate in support vector machines. In: *International conference on data mining*; 2009.
76. Vapnik VN, Kotz S. *Estimation of dependences based on empirical data*, vol. 41. New York: Springer; 1982.
77. Shawe-Taylor J, Bartlett PL, Williamson RC, Anthony M. Structural risk minimization over data-dependent hierarchies. *IEEE Trans Inf Theory.* 1998;44(5):1926–40.
78. Boucheron S, Lugosi G, Massart P. A sharp concentration inequality with applications. *Random Struct Algorithms.* 2000;16(3):277–92.
79. Boucheron S, Lugosi G, Massart P. *Concentration inequalities: a nonasymptotic theory of independence.* Oxford: Oxford University Press; 2013.
80. Bartlett PL, Mendelson S. Rademacher and Gaussian complexities: risk bounds and structural results. *J Mach Learn Res.* 2003;3:463–82.
81. Laviolette F, Marchand M. PAC-Bayes risk bounds for stochastic averages and majority votes of sample-compressed classifiers. *J Mach Learn Res.* 2007;8(7):1461–87.
82. Lacasse A, Laviolette F, Marchand M, Germain P, Usunier N. PAC-Bayes bounds for the risk of the majority vote and the variance of the Gibbs classifier. In: *Advances in Neural information processing systems*; 2006. p. 769–776.
83. Breiman L. Bagging predictors. *Mach Learn.* 1996;24(2):123–40.
84. Breiman L. Random forests. *Mach Learn.* 2001;45(1):5–32.
85. Schapire RE, Freund Y, Bartlett P, Lee WS. Boosting the margin: a new explanation for the effectiveness of voting methods. *Ann Stat.* 1998;26(5):1651–86.
86. Schapire RE, Singer Y. Improved boosting algorithms using confidence-rated predictions. *Mach Learn.* 1999;37(3):297–336.
87. Gelman A, Carlin JB, Stern HS, Rubin DB. *Bayesian data analysis*, vol. 2. London: Taylor & Francis; 2014.
88. Rakhlin A, Mukherjee S, Poggio T. Stability results in learning theory. *Anal Appl.* 2005;3(04):397–417.
89. Devroye L, Györfi L, Lugosi G. *A probabilistic theory of pattern recognition.* Berlin: Springer; 1996.
90. Dietrich R, Opper M, Sompolinsky H. Statistical mechanics of support vector networks. *Phys Rev Lett.* 1999;82(14):2975.
91. Li M, Vitányi P. *An introduction to Kolmogorov complexity and its applications.* Springer-Verlag, New York: Springer Science & Business Media; 2013.
92. Grünwald PD. *The minimum description length principle.* Cambridge: MIT Press; 2007.
93. Tikhonov AN, Arsenin VI. *Solutions of ill-posed problems.* New York: Vh Winston; 1977.
94. Boyd S, Vandenberghe L. *Convex optimization.* Cambridge: Cambridge University Press; 2004.
95. Serfling RJ. Probability inequalities for the sum in sampling without replacement. *Ann Stat.* 1974;2(1):39–48.
96. Zhu X, Goldberg AB. *Introduction to semi-supervised learning.* Synth Lect Artif Intell Mach Learn. 2009;3(1):1–130.

97. Anguita D, Ghio A, Oneto L, Ridella S. In-sample model selection for trimmed hinge loss support vector machine. *Neural Process Lett.* 2012;36(3):275–83.
98. Bartlett PL, Long PM, Williamson RC. Fat-shattering and the learnability of real-valued functions. In: *Proceedings of the seventh annual conference on Computational learning theory.* ACM; 1994. p. 299–310.
99. Zhou D-X. The covering number in learning theory. *J Complex.* 2002;18(3):739–67.
100. Massart P. Some applications of concentration inequalities to statistics. *Ann Fac Sci Toulouse Math.* 2000;9(2):245–303.
101. Ivanov VV. *The theory of approximate methods and their applications to the numerical solution of singular integral equations.* US: Springer Science & Business Media; 1976.
102. Pelckmans K, Suykens JA, De Moor B, Morozov. Ivanov and Tikhonov regularization based LS-SVMS. In: *International Conference on Neural information processing, Springer Berlin Heidelberg;* 2004. p. 1216–1222.
103. Oneto L, Anguita D, Ghio A, Ridella S. The impact of unlabeled patterns in rademacher complexity theory for kernel classifiers. In: *Advances in Neural information processing systems;* 2011. p. 585–593.
104. Anguita D, Ghio A, Oneto L, Ridella S. Unlabeled patterns to tighten rademacher complexity error bounds for kernel classifiers. *Pattern Recognit Lett.* 2014;37:210–9.
105. Eckart C, Young G. The approximation of one matrix by another of lower rank. *Psychometrika.* 1936;1(3):211–8.