Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Self-Paced Learning: An Implicit Regularization Perspective

Yanbo Fan,[1,4] **Ran He,**[1,2,3,4*] **Jian Liang,**[1,2,4] **Baogang Hu**[1,4]
[1]National Laboratory of Pattern Recognition, CASIA
[2]Center for Research on Intelligent Perception and Computing, CASIA
[3]Center for Excellence in Brain Science and Intelligence Technology, CAS
[4]University of Chinese Academy of Sciences (UCAS)
{yanbo.fan, rhe, jian.liang, hubg}@nlpr.ia.ac.cn

Abstract

Self-paced learning (SPL) mimics the cognitive mechanism of humans and animals that gradually learns from easy to hard samples. One key issue in SPL is to obtain better weighting strategy that is determined by the minimizer function. Existing methods usually pursue this by artificially designing the explicit form of SPL regularizer. In this paper, we study a group of new regularizer (named self-paced implicit regularizer) that is deduced from robust loss function. Based on the convex conjugacy theory, the minimizer function for self-paced implicit regularizer can be directly learned from the latent loss function, while the analytic form of the regularizer can be even unknown. A general framework (named SPL-IR) for SPL is developed accordingly. We demonstrate that the learning procedure of SPL-IR is associated with latent robust loss functions, thus can provide some theoretical insights for its working mechanism. We further analyze the relation between SPL-IR and half-quadratic optimization and provide a group of self-paced implicit regularizer. Finally, we implement SPL-IR to both supervised and unsupervised tasks, and experimental results corroborate our ideas and demonstrate the correctness and effectiveness of implicit regularizers.

Introduction

Inspired by the learning process and cognitive mechanism of humans and animals, Bengio et al. (2009) propose a new learning strategy called *curriculum learning* (CL), which gradually includes more and more hard samples into training process. A curriculum can be seen as a sequence of training criteria. For example, in the training of a shape recognition system, images that exhibit less variability such as squares and circles are considered first, followed by hard shapes like ellipses. The curriculum in CL is usually determined by some certain priors and thus is problem specific and lacks generalizations. To alleviate this, Kumar, Packer, and Koller (2010) propose a new learning strategy named self-paced learning (SPL) that incorporates curriculum updating in the process of model optimization. General SPL model consists of a problem specific weighted loss term on all samples and a SPL regularizer on sample weights. Alternative search strategy (ASS) is generally used for optimization. By gradually increasing the penalty of the SPL regularizer during the optimization, more samples are included into training from easy to hard by a self-paced manner. Due to its ability of avoiding bad local minima and improving the generalization performance, many works have been developed based on SPL (Li et al. 2016; Liang et al. 2016; Jiang et al. 2015; Zhang et al. 2015; Supancic and Ramanan 2013; Lee and Grauman 2011).

One key issue in SPL is to obtain better weighting strategy that is determined by the minimizer functions, and existing methods usually pursue this by artificially designing the explicit form of SPL regularizers (Zhang et al. 2016; Xu, Tao, and Xu 2015; Zhao et al. 2015; Jiang et al. 2014a; 2014b). Some examples can be found in the supplementary material. Though shown to be effective in many applications experimentally, the underlying working mechanism of SPL is still unclear and is heavily desired for its future development. One attempt in this aspect is (Meng and Zhao 2015), they show that the ASS method used for SPL accords with the *majorization minimization* (Vaida 2005) algorithm implemented on a latent SPL objective, and deduce the latent objective of hard, linear and mixture regularizers.

In this paper we study a group of new regularizer (named self-paced implicit regularizer) for SPL based on the convex conjugacy theory. Comparing with existing SPL regularizers, self-paced implicit regularizer is deduced from robust loss function and its analytic form can be even unknown. Its properties and corresponding minimizer function can be learned from the latent loss function directly. Besides, the proposed self-paced implicit regularizer is independent of the learning objective and thus leads to a general framework (named SPL-IR) for SPL. SPL-IR can be optimized via ASS algorithm. More importantly, we demonstrate that the learning procedure of SPL-IR is indeed associated with latent robust loss functions, thus may provide some theoretical insights for its working mechanism (e.g. its robustness to outliers and heavy noise). We further analyze the relations between SPL-IR and half-quadratic (HQ) optimization and provide a group of self-paced implicit regularizer accordingly. Such relations can be beneficial to both SPL and HQ optimization. Finally, we implement SPL-IR to three classical tasks (i.e. matrix factorization, clustering and classification). Experimental results corroborate our ideas and demonstrate the correctness and effectiveness of SPL-IR.

Our work has three main contributions: (1) We pro-

pose self-paced implicit regularizer for SPL, and develop a general implicit regularization framework (named SPL-IR) based on it. The self-paced implicit regularizers not only enrich the family of regularizers for SPL but also provide some insights on the working mechanism of SPL. (2) We analyze the connections between SPL-IR and HQ optimization, and provide a group of robust loss function induced self-paced implicit regularizer for SPL-IR accordingly. (3) Experimental results on both supervised and unsupervised tasks corroborate our ideas and demonstrate the correctness and effectiveness of SPL-IR.

Preliminaries

Self-Paced Learning via Explicit Regularizers

Given training dataset $\mathbf{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with n samples, where $\mathbf{x}_i \in R^d$ is the i-th sample, y_i is the optional information according to the learning objective (e.g. y_i can be the label of $\mathbf{x}_i$ in a classification model). Let $f(.\ , \mathbf{w})$ denote the learned model and $\mathbf{w}$ be the model parameter. $L(y_i, f(\mathbf{x}_i, \mathbf{w}))$ is the loss function of i-th sample. The objective of SPL is to jointly optimize the model parameter $\mathbf{w}$ and the latent sample weights $\mathbf{v} = [v_1, v_2, \ldots, v_n]$ via the following minimization problem:

$$\min_{\mathbf{w},\mathbf{v}} \mathbb{E}(\mathbf{w}, \mathbf{v}; \lambda) = \sum_{i=1}^{n} v_i L(y_i, f(\mathbf{x}_i, \mathbf{w})) + g(\lambda, v_i), \quad (1)$$

where $g(\lambda, v)$ is called self-paced regularizer and λ is a penalty parameter that controls the learning pace. ASS algorithm is generally used for (1), which alternatively optimizes $\mathbf{w}$ and $\mathbf{v}$ while keeping the other fixed. Specifically, given sample weights $\mathbf{v}$, the minimization over $\mathbf{w}$ is a weighted loss minimization problem that is independent of regularizer $g(\lambda, v)$; given model parameter $\mathbf{w}$, the optimal weight of i-th sample is determined by

$$\min_{v_i} \quad v_i L(y_i, f(\mathbf{x}_i, \mathbf{w})) + g(\lambda, v_i). \quad (2)$$

Since $\ell_i = L(y_i, f(\mathbf{x}_i, \mathbf{w})$ is constant once $\mathbf{w}$ is given, the optimal value of v_i is uniquely determined by the corresponding minimizer function $\sigma(\lambda, \ell_i)$ that satisfies

$$\sigma(\lambda, \ell_i)\ell_i + g(\lambda, \sigma(\lambda, \ell_i)) \leq v_i \ell_i + g(\lambda, v_i), \forall v_i \in [0, 1]. \quad (3)$$

For example, if $g(\lambda, v_i) = -\lambda v_i$ (Kumar, Packer, and Koller 2010), the optimal v_i^* is calculated by

$$v_i^* = \sigma(\lambda, \ell_i) = \begin{cases} 1, & if \quad \ell_i \leq \lambda \\ 0, & otherwise \end{cases} \quad (4)$$

By gradually increasing the value of λ, more and more hard samples are included into the training process. There are many efforts that have been put into the learning of appropriate minimizer functions (Zhang et al. 2016; Xu, Tao, and Xu 2015; Zhao et al. 2015; Jiang et al. 2014a; 2014b; Supancic and Ramanan 2013), and we categorize them as SPL with explicit regularizers as they usually require the explicit form of regularizer $g(\lambda, v)$. $\sigma(\lambda, \ell)$ is then derived from the form of $g(\lambda, v)$.

Half-Quadratic Optimization

Half-quadratic optimization (Nikolova and Ng 2005; Geman and Yang 1995; Geman and Reynolds 1992) is a commonly used optimization method that based on the convex conjugacy theory. It tries to solve a nonlinear objective function via optimizing a series of half-quadratic reformulation problems iteratively (He, Tan, and Wang 2014; He et al. 2014b; Yuan and Hu 2009). Its multiplicative form is briefly introduced as follows.

Given a differentiable function $\phi(t) : R \to R$, if $\phi(t)$ further satisfies the conditions of the multiplicative form of HQ optimization in (Nikolova and Chan 2007), the following equation holds for any fixed t,

$$\phi(t) = \inf_{p \in R_+} \left\{ \frac{1}{2} p t^2 + \psi(p) \right\}, \quad (5)$$

where $\psi(p)$ is the dual potential function of $\phi(t)$ and $R_+ = \{t | t \geq 0\}$. $\psi(p)$ is convex and reads

$$\psi(p) = \sup_{t \in R_+} \left\{ -\frac{1}{2} p t^2 + \phi(t) \right\}, \quad (6)$$

More analysis about $\phi(t)$ and $\psi(p)$ refers to (Nikolova and Ng 2005). The optimal p^* that minimizes (5) is uniquely determined by the corresponding minimizer function $\delta(t)$, which is derived from the convex conjugacy theory and is only relative to function $\phi(t)$. For each t, $\delta(t)$ is such that

$$\frac{1}{2}\delta(t)t^2 + \psi(\delta(t)) \leq \frac{1}{2} p t^2 + \psi(p), \ \forall p \in R_+. \quad (7)$$

The optimization of $\phi(t)$ can be done via iteratively minimizing t and p in (5). One only needs to focus on $\phi(t)$ and its corresponding minimizer function $\delta(t)$ in HQ optimization, and the analytic form of the dual potential function $\psi(p)$ can be even unknown.

The Proposed Method

In this section, we first give the definition of the proposed self-paced implicit regularizer and derive its minimizer function based on the convex conjugacy theory. Then we develop a general self-paced learning framework (named SPL-IR) based on implicit regularization. Finally, we analyze the relations between SPL-IR and HQ optimization.

Self-Paced Implicit Regularizer

Based on our above analysis of SPL, we define the self-paced implicit regularizer as follows,

Definition 1. ***Self-Paced Implicit Regularizer.*** *A self-paced implicit regularizer $\psi(\lambda, v)$ is defined as the dual potential function of a robust loss function $\phi(\lambda, t)$, and satisfies*

1. $\phi(\lambda, t) = \min_{v \geq 0}\ vt + \psi(\lambda, v)$;

2. $\sigma(\lambda, t)$ is the minimizer function of $\phi(\lambda, t)$ that satisfies $\sigma(\lambda, t)t + \psi(\lambda, \sigma(\lambda, t)) \leq vt + \psi(\lambda, v)$, $\forall\ v \in R_+$;

3. $\sigma(\lambda, t)$ is non-negative and up-bounded, $\forall\ t \in R_+$;

4. $\sigma(\lambda, t)$ is monotonically decreasing w.r.t. t, $\forall\ t \in R_+$;

5. $\sigma(\lambda, t)$ is monotonous w.r.t. $\lambda \in R_+$;

where λ is a hyper-parameter and it is the same in $\phi(\lambda, t)$, $\psi(\lambda, v)$ and $\sigma(\lambda, t)$. λ is considered to be fixed in the first four conditions.

Table 1: Loss function $\phi(\lambda, t)$ and its corresponding minimizer function $\sigma(\lambda, t)$, λ is a hyper-parameter.

	Huber	Cauchy	L1-L2	Welsch
Loss function $\phi(\lambda, t)$	$\begin{cases} t^2/2, & \lvert t\rvert \le \lambda \\ \lambda\lvert t\rvert - \frac{\lambda^2}{2}, & \lvert t\rvert > \lambda \end{cases}$	$\lambda^2 \log(1 + (t/\lambda)^2)$	$\sqrt{\lambda + t^2} - 1$	$\lambda^2(1 - \exp(-\frac{t^2}{\lambda^2}))$
Minimizer function $\sigma(\lambda, t)$	$\begin{cases} 1 & \lvert t\rvert \le \lambda \\ \lambda/\lvert t\rvert, & \lvert t\rvert > \lambda \end{cases}$	$2/(1 + (t/\lambda)^2)$	$1/\sqrt{\lambda + t^2}$	$2\exp(-\frac{t^2}{\lambda^2})$

Proposition 1 *For any fixed λ, if $\phi(\lambda, t)$ further satisfies the conditions referred in (Nikolova and Chan 2007), its minimizer function $\sigma(\lambda, t)$ is uniquely determined by $\phi(\lambda, t)$ and the analytic form of the dual potential function $\psi(\lambda, v)$ can be even unknown during the optimization.*

The proof of Proposition 1 is given in the appendix. According to Definition 1, the self-paced implicit regularizer is derived from robust loss function. Its properties can be learned from both $\psi(\lambda, v)$ and the latent loss function $\phi(\lambda, t)$. The corresponding minimizer function $\sigma(\lambda, t)$ can be learned from $\phi(\lambda, t)$ directly. During the optimization, the optimal v^* is determined by $\sigma(\lambda, t)$ and the analytic form of $\psi(\lambda, v)$ can be even unknown, hence $\psi(\lambda, v)$ is named self-paced implicit regularizer. Besides, the last three conditions in Definition 1 are required for SPL regimes. Specifically, let t denote the sample loss, condition 4 indicates that the model is likely to select easy samples (with smaller losses) in favor of hard samples (with larger losses) for a fixed λ, and condition 5 makes sure that we can incorporate more and more samples through turning parameter λ.

The self-paced implicit regularizer $\psi(\lambda, v)$ defined here is derived from robust loss function $\phi(\lambda, t)$. By establishing the relations between $\phi(\lambda, t)$ and $\psi(\lambda, v)$, we can analyze its working mechanisms as well as develop new SPL regularizers based on the development of robust loss functions. Moreover, the properties of $\psi(\lambda, v)$ and its corresponding minimizer function $\sigma(\lambda, t)$ can be learned from their latent robust loss function $\phi(\lambda, t)$ directly.

Self-Paced Learning via Implicit Regularizers

We can develop an implicit regularization framework for SPL based on the proposed self-paced implicit regularizer. By substituting the regularization term $g(\lambda, v)$ in (1) with a self-paced implicit regularizer $\psi(\lambda, v)$ given in Definition 1, we obtain the following SPL-IR problem,

$$\min_{\mathbf{w},\mathbf{v}} \mathbb{E}(\mathbf{w}, \mathbf{v}; \lambda) = \sum_{i=1}^{n} v_i L(y_i, f(\mathbf{x}_i, \mathbf{w})) + \psi(\lambda, v_i). \quad (8)$$

It can be solved via ASS algorithm, which alternatively optimizes $\mathbf{w}$ and $\mathbf{v}$ while keeping the other fixed. However, different from existing SPL regularizers, the analytic form of $\psi(\lambda, v)$ in (8) can be unknown and the optimal $\mathbf{v}^*$ is determined by the corresponding minimizer function given in Definition 1. The optimization procedure of (8) is described in Algorithm 1. Model (8) is called an implicit regularization framework since it does not require the explicit form of $\psi(\lambda, v)$. The concept of implicit regularization is also used in (Mahoney 2012; Mahoney and Orecchia 2011) as a by-product of approximation algorithms.

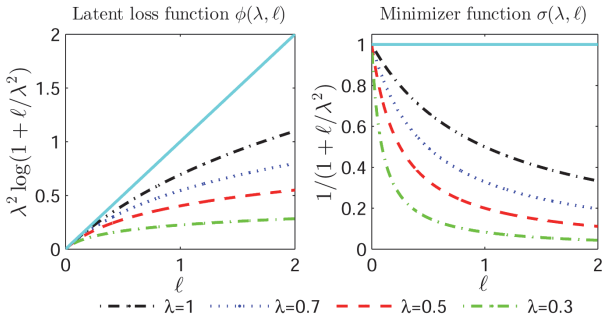
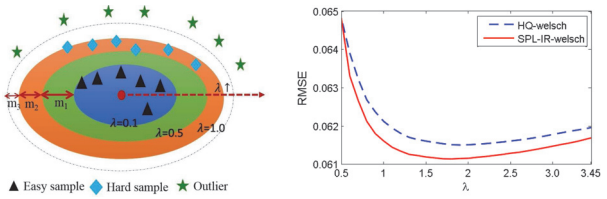


Figure 1: Example of latent loss function and its corresponding minimizer function in Definition 1. The x-axis refers to the original loss ℓ. The solid lines are given for comparison, it is $y = x$ in the left figure, and $y = 1$ in the right one.

An insightful phenomenon is that the learning procedure of SPL-IR is actually associated with certain latent loss functions. For example, for a certain implicit regularizer and its corresponding minimizer function $v_i^* = \sigma(\lambda, \ell_i) = 1/(1 + \ell_i/\lambda^2)$ in Algorithm 1 (where $\ell_i = L(y_i, f(\mathbf{x}_i, \mathbf{w}^*))$), one can be considered to minimize a latent robust function $\sum_{i=1}^{n} \lambda^2 \log(1 + \ell_i/\lambda^2)$ during each round. Figure 1 gives a graphical illustration. The latent loss function $\phi(\lambda, \ell)$ can be considered to carry out a meaningful transformation on the original loss ℓ. When ℓ is larger than a certain threshold, $\phi(\lambda, \ell)$ becomes a constant and its corresponding minimizer function $\sigma(\lambda, \ell)$ becomes zero, hence the related sample is not considered for optimization. Through this, it can suppress the influence of hard samples (refer to larger ℓ) while retaining that of easy samples (refer to smaller ℓ). This may also provide some insights on the robustness of SPL-IR to outliers and heavy noise as they can usually cause larger losses. More specifically, starting with a small λ (e.g. 0.3), only a small part of samples with very small losses are involved (they are considered to contain reliable information). As λ increases, the suppressing effect of $\phi(\lambda, \ell)$ on larger losses becomes weaker and their corresponding weights increase, consequently more and more hard samples with larger losses (may also contain more knowledge) are involved into the training process. While gradually incorporating these knowledge, the model becomes stronger and stronger. The learning procedure of some existing regularizers like hard and linear (Meng and Zhao 2015) can also be explained under the framework of SPL-IR.

SPL-IR in (8) is considered as a general SPL framework from two aspects: firstly, $\psi(\lambda, v)$ represents a spectrum of



(a) Toy Example (b) HQ and SPL-IR

Figure 2: In (a), training samples are roughly divided into three types: easy samples ▲, hard samples ◆ and outliers ★. λ is usually fixed in HQ methods (e.g. $\lambda = 0.5$), hence some samples may be discarded incorrectly. In contrast, SPL-IR can gradually incorporate more samples from easy to hard (i.e. λ grows iteratively). (b) demonstrates the performances of HQ and SPL-IR methods on a synthetic matrix factorization dataset, Welsch minimizer function is adopted for both methods. For HQ-welsch, standard HQ algorithm (Nikolova and Ng 2005) is implemented with each λ independently. More details refer to Section 3.3 and 4.1.

self-paced implicit regularizer that is developed based on robust loss function and the convex conjugacy theory; secondly, $\psi(\lambda, v)$ is independent of specific model objective $L(y_i, f(\mathbf{x}_i, \mathbf{w}))$ and thus can be used in various applications. Besides, standard ASS strategy is used for both SPL with explicit regularizer (model (1)) and SPL-IR (model (8)). It includes a weighted loss minimization step and a weight updating step at each iteration. Hence for a specific loss function $L(y_i, f(\mathbf{x}_i, \mathbf{w}))$ and a fixed number of iteration, the time complexities of SPL with explicit regularizer and SPL-IR is in the same order of magnitude.

SPL-IR and Half-Quadratic Optimization

We can develop new self-paced implicit regularizers based on the development of robust loss functions. Specifically, we analyze the relations between SPL-IR and HQ optmization and provide several self-paced implicit regularizers accordingly. For better demonstration, we first give an equivalent quadratic form definition of self-paced implicit regularizer,

Definition 2 (Quadratic Form). ***Self-Paced Implicit Regularizer.*** *A self-paced implicit regularizer $\psi(\lambda, v)$ is defined as the dual potential function of a robust loss function $\phi(\lambda, t)$, and satisfies*

1. $\phi(\lambda, t) = \min_{v \geq 0} \frac{1}{2} vt^2 + \psi(\lambda, v)$;

2. $\sigma(\lambda, t)$ is the minimizer function of $\phi(\lambda, t)$ and satisfies $\frac{1}{2}\sigma(\lambda, t)t^2 + \psi(\lambda, \sigma(\lambda, t)) \leq \frac{1}{2}vt^2 + \psi(\lambda, v)$, $\forall\, v \in R_+$.

3. $\sigma(\lambda, t)$ is non-negative and up-bounded, $\forall\, t \in R_+$;

4. $\sigma(\lambda, t)$ is monotonically decreasing w.r.t. t, $\forall\, t \in R_+$;

5. $\sigma(\lambda, t)$ is monotonous w.r.t. $\lambda \in R_+$;

where λ is a hyper-parameter and it is the same in $\phi(\lambda, t)$, $\psi(\lambda, v)$ and $\sigma(\lambda, t)$. λ is considered to be fixed in the first four conditions.

The equivalency of Definition 1 and Definition 2 is shown in the supplementary material. Seen from Definition 2, there is a close relationship between self-paced implicit regularizer and the dual potential function defined in HQ reformulation (5). Apparently, the dual potential function in (5) and the minimizer function in (7) satisfy the first two conditions in Definition 2, and self-paced implicit regularizer imposes further constraints on the minimizer function $\sigma(\lambda, t)$ for the regimes of SPL. Many robust loss functions and their corresponding minimizer functions in multiplicative form of HQ have been developed (some of them are tabulated in Table 1). It is easy to verify that the functions in Table 1 satisfy all the conditions in Definition 2, hence they can be adapted for self-paced implicit regularizers. The loss functions in Table 1 are well defined and have proven to be effective in many areas (Chen et al. 2016a; 2016b; He et al. 2014a; 2014b). Meanwhile, though self-paced implicit regularizer can be developed from HQ optimization, their optimization procedures are quite different. In HQ, one mainly focuses on the minimization of loss function $\phi(\lambda, t)$ and hyper-parameter λ is predetermined and fixed during the optimization. While aiming to gradually optimize from easy to hard samples, SPL-IR uses the right-hand side $vt^2/2 + \psi(\lambda, v)$ to model problems and one key concern is the weighting strategy that determined by the minimizer function $\sigma(\lambda, t)$. Besides, in order to gradually increase samples, λ is updated stage by stage in SPL-IR.

Figure 2 gives an intuitive interpretation. If we set $t_i = \sqrt{L(y_i, f(\mathbf{x}_i, \mathbf{w}^*))}$ and use the minimizer function of Welsch given in Table 1 for weight updating in Algorithm 1, model (8) can be considered to sequential optimize a group of Welsch loss functions with monotonically increasing λ. Hence SPL-IR is able to gradually optimize from easy to hard samples while incorporating the good properties of robust Welsch functions. On the other hand, for HQ optimization, λ is predefined and fixed during the whole optimization. Hence its performance may be largely influenced by the selection of λ. For example, when λ is somehow small (e.g. $\lambda < 1$ in Figure 2(b)), some hard samples will be simply considered as outliers and discarded. From the comparisons in Figure 2(b), we can find that SPL-IR can always outperform HQ for every λ.

Algorithm 1 : Self-Paced Learning via Implicit Regularizers

Input: Input dataset $\mathbf{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, step size $\mu > 1$.
Output: Model parameter $\mathbf{w}$.

1: Initialize sample weights $\mathbf{v}^*$ and parameter λ;
2: **repeat**
3: Update $(\mathbf{w}^*, \mathbf{v}^*) = \arg\min_{\mathbf{w},\mathbf{v}} \mathbb{E}(\mathbf{w}, \mathbf{v}; \lambda)$ by using ASS algorithms, $\mathbf{v}$ is iteratively optimized by the corresponding minimizer function σ;
4: Monotone increase (or decrease) λ by step-size μ;
5: **until** convergence.
6: **return** $\mathbf{w}^*$

Experiments

To illustrate the correctness and effectiveness of the developed SPL-IR model, we apply it to three classical tasks: matrix factorization, clustering and classification. There are two hyper-parameter (λ, μ) that need to be tuned in Algorithm 1. We follow a standard setting in SPL (Kumar, Packer, and Koller 2010). That is, λ is initialized to obtain about

Table 2: Numerical results of L_1-norm MF problem with L_2-norm regularization. The best results are highlighted in bold.

Method	PRMF	SPL-hard	SPL-mixture	SPL-IR-huber	SPL-IR-L1-L2	SPL-IR-cauchy	SPL-IR-welsch
RMSE	0.1528	0.0949	0.0625	0.0627	0.0650	0.0620	**0.0596**
MAE	0.0994	0.0672	0.0475	0.0476	0.0493	0.0472	**0.0455**

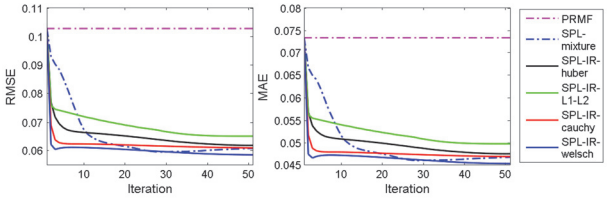


Figure 3: Tendency curves of RMSE and MAE w.r.t. the iterations.

half samples, then it is iteratively updated to involve more and more samples gradually. The practical updating direction depends on the specific minimizer function. For functions given in Table 1, $\lambda_{T+1} = \lambda_T/\mu$ for L1-L2 while $\lambda_{T+1} = \lambda_T * \mu$ for Huber, Cauchy and Welsch, where $\mu > 1$ is a step factor and T is an iteration number. Similar settings are adjusted for the competing SPL regularizers, including SPL-hard (Kumar, Packer, and Koller 2010) and SPL-mixture (Zhao et al. 2015).

Matrix Factorization

Matrix factorization (MF) is one of the fundamental problems in machine learning and data mining. It aims to factorize an $m \times n$ data matrix $\mathbf{Y}$ into two smaller factors $\mathbf{U} \in R^{m\times r}$ and $\mathbf{V} \in R^{n\times r}$, where $r \ll min(m,n)$, such that $\mathbf{U}\mathbf{V}^T$ is possibly close to $\mathbf{Y}$. MF has been successfully implemented in many applications, such as collaborative filtering (Salakhutdinov and Mnih 2008).

Here we consider the MF problem on a synthetic dataset. Specifically, the data is generated as follows: two matrices $\mathbf{U}$ and $\mathbf{V}$, both of which are of size 100×4, are first randomly generated with each entry drawn from the Gaussian distribution $\mathcal{N}(0,1)$, leading to a ground truth rank-4 matrix $\mathbf{Y}_0 = \mathbf{U}\mathbf{V}^T$. Then we randomly choose 40% of the entries and treat them as missing data. Another 20% of the entries are randomly selected and added to uniform noise on $[-20, 20]$, and the rest are perturbed with Gaussian noise drawn from $\mathcal{N}(0, 0.1^2)$. Similar to (Zhao et al. 2015), we consider L_1-norm MF problem with L_2-norm regularization. The baseline algorithm is PRMF (Wang et al. 2012), and we modify it with different SPL regularizers for comparison. Two commonly used metrics are adopted here: (1) *root mean square error* (RMSE): $\frac{1}{\sqrt{mn}}||\mathbf{Y}_0 - \hat{\mathbf{U}}\hat{\mathbf{V}}^T||_F$, and (2) *mean absolute error* (MAE): $\frac{1}{mn}||\mathbf{Y}_0 - \hat{\mathbf{U}}\hat{\mathbf{V}}^T||_1$, where $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ denote the outputs of MF algorithms. All the algorithms are implemented with 50 realizations and their mean values are reported.

Table 2 tabulates their numerical results. All SPL-IR algorithms obtain performance improvements over baseline algorithm PRMF, which shows the benefits of SPL regimes. Comparing among different SPL regularizers, the results of the proposed self-paced implicit regularizers are comparable to or even better than that of mixture and hard schemes, especially for SPL-IR with welsch regularizer. These demonstrate the correctness and effectiveness of the proposed self-paced implicit regularizer. Figure 3 further plots the tendency curves of RMSE and MAE with different self-paced implicit regularizers and mixture regularizer for better understanding, the results of PRMF are also reported as a baseline. The performances of all implicit regularizers improve rapidly for the first few iterations as more and more easy samples are likely to be involved in these phases. As the number of iterations increases, the improvements become steady as some hard instances or outliers are included.

Table 3: Clustering performance on the Handwritten Digit dataset. The best results are highlighted in bold.

Method	ACC	NMI	AR
BSV	73.37(6.24)	73.04(3.03)	63.36(5.51)
Con-MC	77.48(7.81)	77.32(3.65)	69.00(6.56)
SPL-hard	82.06(5.90)	75.84(2.93)	70.94(4.97)
SPL-mixture	84.48(6.83)	81.19(2.99)	76.26(5.72)
MSPL	83.97(6.99)	80.64(3.51)	75.10(6.40)
SPL-IR-huber	84.25(7.03)	81.04(3.48)	75.64(6.37)
SPL-IR-L1-L2	83.50(6.77)	80.06(3.37)	74.30(6.08)
SPL-IR-cauchy	84.50(7.07)	81.43(3.45)	76.17(6.36)
SPL-IR-welsch	**86.24(7.05)**	**83.26(3.49)**	**79.05(6.41)**

Multi-view Clustering

Multi-view clustering aims to group data with multiple views into their underlying classes (Xu, Tao, and Xu 2013). Most existing multi-view clustering algorithms fit a non-convex model and may be stuck in bad local minima. To alleviate this, Xu, Tao, and Xu (2015) propose a multi-view self-paced learning algorithm (MSPL) that considers the learnability of both samples and views and achieves promising results. Here we simply modified their MSPL model with different SPL regularizers for comparison. The UCI Handwritten Digit dataset [1] is used in this experiment. It consists of 2,000 handwritten digits classified into ten categories (0-9). Each instance is represented in terms of six kinds of features (or views). Here we make use of all the six views for all the comparing algorithms. The baseline algorithms are standard k-means on each single view's representation (the best single view result is reported as BSV), and Con-MC (the features are concatenated on all views firstly, and then standard k-means is applied).

Three commonly used metrics are adopted: clustering ac-

[1] https://archive.ics.uci.edu/ml/datasets

Table 4: Statistical Information of Databases.

Dataset	#.Category	#.Instance	#.Feature
Breast	2	569	30
Spambase	2	4601	57
Svmguide1	2	7089	4

Table 5: Classification accuracy (%). The best results are highlighted in bold, respectively.

Without Label Noise			
Method	Breast	Spambase	Svmguide1
LR	97.36(2.22)	92.35(1.47)	95.39(0.95)
SPL-hard	97.54(2.22)	92.63(1.08)	95.39(0.95)
SPL-mixture	98.25(1.65)	92.83(1.44)	95.51(1.04)
SPL-IR-huber	**98.77(1.19)**	93.05(1.25)	95.57(0.95)
SPL-IR-L1-L2	97.90(1.79)	93.00(1.36)	95.57(1.10)
SPL-IR-cauchy	98.42(1.54)	93.09(1.41)	95.65(1.01)
SPL-IR-welsch	98.25(1.65)	**93.13(1.34)**	**95.68(0.90)**
With 20% Random Label Noise			
Method	Breast	Spambase	Svmguide1
LR	92.08(2.96)	89.28(1.66)	91.52(0.65)
SPL-hard	96.13(2.15)	89.81(1.61)	92.72(1.12)
SPL-mixture	96.66(2.12)	90.76(1.82)	93.81(0.79)
SPL-IR-huber	96.84(2.33)	90.92(1.65)	93.54(0.75)
SPL-IR-L1-L2	94.72(2.89)	90.09(1.65)	92.83(0.71)
SPL-IR-cauchy	97.54(1.90)	90.85(1.55)	93.88(1.05)
SPL-IR-welsch	**97.89(1.63)**	**91.37(1.37)**	**94.37(0.90)**

curacy (ACC), normalized mutual information (NMI) and adjusted rand index (AR) (Hubert and Arabie 1985). Higher value indicates better performance for all the metrics. All algorithms are implemented 20 times and both mean values and standard derivations are reported. Table 3 tabulates their numerical results. It can be seen that all the multi-view algorithms obtain significant improvements over single-view ones, which demonstrates the benefits of integrating information from different views. More importantly, comparing to Con-MC, the SPL-IR algorithms can further improve the performance by gradually optimizing from easy to hard samples and avoiding bad local minima. The proposed self-paced implicit regularizers are comparable to or even better than the compared SPL regularizers.

Classification

The proposed self-paced implicit regularizers can be flexible implemented to supervised tasks. Here we conduct a binary classification task. Specifically, we utilize the L2-regularized Logistic Regression (LR) model as our baseline, and incorporate it with different SPL regularizers for comparison. Liblinear (Fan et al. 2008) is used as the solver of LR. Three real-world databases are considered: Breast [2], Spambase [2] and Svmguide1 (Chang and Lin 2011). Their statistical information is summarized in Table 4. For each dataset, we consider it without additional noise and with 20% random label noise, respectively. The 20% random label noise means we randomly select 20% samples from training data and reversal their labels (change positive to negative, and vice-versa). We use 10-fold cross validation for all the databases, and report both their mean values and their standard derivations.

Classification accuracy is used for performance measure. Table 5 reports their numerical results. For both situations, SPL-IR algorithms can get performance improvements over original LR method to some extent. Moreover, when adding random label noise, the performance of original LR degenerates a lot, while the SPL algorithms can still obtain relatively high performance, especially for SPL-IR with welsch regularizer. This corroborates our analysis about the robustness of SPL-IR to outliers and heavy noise.

[2]https://archive.ics.uci.edu/ml/datasets

Conclusions

In this paper, we study a group of new regularizer, named self-paced implicit regularizer for SPL based on the convex conjugate theory. The self-paced implicit regularizer is derived from robust loss function and its analytic form can be even unknown. Its properties and corresponding minimizer function can be learned from the latent loss function directly. We then develop a general SPL framework (SPL-IR) based on it. We later analyze the relations between SPL-IR and HQ optimization and develop a group of self-paced implicit regularizer accordingly. It is nontrivial to analyze and compare different SPL models theoretically. The proposed SPL-IR models can be seen as novel SPL models induced by robust loss functions. This also can provide us with a perspective to learn their properties based on theoretical and experimental results of robust loss functions (e.g. robust Welsch loss function induced SPL regularizer may be appropriate for outliers and heavy noises). Experimental results on both supervised and unsupervised tasks demonstrate the correctness and effectiveness the proposed self-paced implicit regularizer.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No.61622310, 61273196), State Key Development Program (Grant No. 2016YFB1001001) and Youth Innovation Promotion Association CAS(2015190). We would like to thank Prof. Deyu Meng from XJTU for the valuable comments.

Appendix

Proof (of Proposition 1). The proof sketch is similar to that in (Nikolova and Chan 2007). For ease of representation, we omit λ and use $\phi(t)$, $\psi(v)$ and $\sigma(t)$ for short. Some fundamental assumptions about $\phi(t)$ are: **H1:** $\phi : R_+ \to R$ is increasing with $\phi \not\equiv 0$ and $\phi(0) = 0$; **H2:** $\phi(t)$ is C^1 and concave; **H3:** $\lim_{t\to\infty} \phi(t)/t = 0$.

Put $\theta(t) = -\phi(t)$, then θ is convex by H2. Its convex conjugate is $\theta^*(v) = \sup_{t\geq 0} \{vt - \theta(t)\}$. By the Fenchel-Moreau theorem (Rockafellar 2015), the convex conjugate of θ^* is θ, that is $\theta(t) = (\theta^*)^*(t) = \sup_{v\leq 0} \{vt - \theta^*(v)\} = -\inf_{v\geq 0} \{vt + \theta^*(-v)\}$. Thus we have

$$\psi(v) = \theta^*(-v) = \sup_{t\geq 0} \{-vt - \theta(t)\} = \sup_{t\geq 0} \{-vt + \phi(t)\}. \quad (9)$$

$$\phi(t) = -\theta(t) = \inf_{v\geq 0} \{vt + \theta^*(-v)\} = \inf_{v\geq 0} \{vt + \psi(v)\}. \quad (10)$$

Then the problem becomes how to achieve the supremum in (9) jointly with the infimum in (10). For any $\hat{v} > 0$, define $f_{\hat{v}} : R_+ \to R$ by $f_{\hat{v}}(t) = \hat{v}t + \theta(t)$, then we have $\psi(\hat{v}) = -\inf_{t\geq 0} f_{\hat{v}}(t)$ from (9). According to H1-H3, $f_{\hat{v}}$ is convex with $f_{\hat{v}}(0) = 0$ and $\lim_{t\to+\infty} f_{\hat{v}}(t) = +\infty$. Thus $f_{\hat{v}}$ can reach its unique minimum at a $\hat{t} \geq 0$, and $\psi(\hat{v}) = -\hat{v}\hat{t} + \phi(\hat{t})$ from (9). Hence equivalently the infimum in (10) is reached at $\hat{v}$ as $\phi(\hat{t}) = \hat{v}\hat{t} + \psi(\hat{v})$. Then we have $\hat{v} = \sigma(t) = -\theta'(t) = \phi'(t)$. Thus the optimal v is uniquely determined by the minimizer function $\sigma(t)$ that is derived from $\phi(t)$. The analytic form of the dual potential function $\psi(v)$ could be unknown during the optimization. The proof is then completed.

References

Bengio, Y.; Louradour, J.; Collobert, R.; and Weston, J. 2009. Curriculum learning. In *ICML*, 41–48.

Chang, C.-C., and Lin, C.-J. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2(3):27.

Chen, B.; Liu, X.; Zhao, H.; and Príncipe, J. C. 2016a. Maximum correntropy kalman filter. *Automatica, forthcoming*.

Chen, B.; Xing, L.; Zhao, H.; Zheng, N.; and Príncipe, J. C. 2016b. Generalized correntropy for robust adaptive filtering. *TSP* 64(13):3376–3387.

Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; and Lin, C.-J. 2008. Liblinear: A library for large linear classification. *JMLR* 9:1871–1874.

Geman, D., and Reynolds, G. 1992. Constrained restoration and the recovery of discontinuities. *TPAMI* 14(3):367–383.

Geman, D., and Yang, C. 1995. Nonlinear image recovery with half-quadratic regularization. *TIP* 4(7):932–946.

He, R.; Hu, B.; Yuan, X.; and Wang, L. 2014a. *Robust recognition via information theoretic learning*. Springer.

He, R.; Zheng, W.-S.; Tan, T.; and Sun, Z. 2014b. Half-quadratic-based iterative minimization for robust sparse representation. *TPAMI* 36(2):261–275.

He, R.; Tan, T.; and Wang, L. 2014. Robust recovery of corrupted low-rank matrix by implicit regularizers. *TPAMI* 36(4):770–783.

Hubert, L., and Arabie, P. 1985. Comparing partitions. *Journal of Classification* 2(1):193–218.

Jiang, L.; Meng, D.; Mitamura, T.; and Hauptmann, A. G. 2014a. Easy samples first: Self-paced reranking for zero-example multimedia search. In *ACM MM*, 547–556.

Jiang, L.; Meng, D.; Yu, S.-I.; Lan, Z.; Shan, S.; and Hauptmann, A. 2014b. Self-paced learning with diversity. In *NIPS*, 2078–2086.

Jiang, L.; Meng, D.; Zhao, Q.; Shan, S.; and Hauptmann, A. G. 2015. Self-paced curriculum learning. In *AAAI*, 2694–2700.

Kumar, M. P.; Packer, B.; and Koller, D. 2010. Self-paced learning for latent variable models. In *NIPS*, 1189–1197.

Lee, Y. J., and Grauman, K. 2011. Learning the easy things first: Self-paced visual category discovery. In *CVPR*, 1721–1728.

Li, H.; Gong, M.; Meng, D.; and Miao, Q. 2016. Multi-objective self-paced learning. In *AAAI*, 1802–1808.

Liang, J.; Li, Z.; Cao, D.; He, R.; and Wang, J. 2016. Self-paced cross-modal subspace matching. In *SIGIR*, 569–578.

Mahoney, M. W., and Orecchia, L. 2011. Implementing regularization implicitly via approximate eigenvector computation. In *ICML*, 121–128.

Mahoney, M. W. 2012. Approximate computation and implicit regularization for very large-scale data analysis. In *PODS*, 143–154.

Meng, D., and Zhao, Q. 2015. What objective does self-paced learning indeed optimize? *arXiv preprint arXiv:1511.06049*.

Nikolova, M., and Chan, R. H. 2007. The equivalence of half-quadratic minimization and the gradient linearization iteration. *TIP* 16(6):1623–1627.

Nikolova, M., and Ng, M. K. 2005. Analysis of half-quadratic minimization methods for signal and image recovery. *SIAM Journal on Scientific Computing* 27(3):937–966.

Rockafellar, R. T. 2015. *Convex analysis*. Princeton university press.

Salakhutdinov, R., and Mnih, A. 2008. Probabilistic matrix factorization. In *NIPS*, 1–8.

Supancic, J. S., and Ramanan, D. 2013. Self-paced learning for long-term tracking. In *CVPR*, 2379–2386.

Vaida, F. 2005. Parameter convergence for em and mm algorithms. *Statistica Sinica* 831–840.

Wang, N.; Yao, T.; Wang, J.; and Yeung, D.-Y. 2012. A probabilistic approach to robust matrix factorization. In *ECCV*, 126–139.

Xu, C.; Tao, D.; and Xu, C. 2013. A survey on multi-view learning. *arXiv preprint arXiv:1304.5634*.

Xu, C.; Tao, D.; and Xu, C. 2015. Multi-view self-paced learning for clustering. In *IJCAI*, 3974–3980.

Yuan, X.-T., and Hu, B.-G. 2009. Robust feature extraction via information theoretic learning. In *ICML*, 1193–1200.

Zhang, D.; Meng, D.; Li, C.; Jiang, L.; Zhao, Q.; and Han, J. 2015. A self-paced multiple-instance learning framework for co-saliency detection. In *ICCV*, 594–602.

Zhang, D.; Meng, D.; Zhao, L.; and Han, J. 2016. Bridging saliency detection to weakly supervised object detection based on self-paced curriculum learning. In *IJCAI*, 3538–3544.

Zhao, Q.; Meng, D.; Jiang, L.; Xie, Q.; Xu, Z.; and Hauptmann, A. G. 2015. Self-paced learning for matrix factorization. In *AAAI*, 3196–3202.