doi:10.11887/j.cn.201404032 http://journal.nudt.edu.cn

一种具有遗忘特性的在线学习算法框架*

孙博良,李国辉
(国防科技大学 信息系统与管理学院,湖南 长沙 410073)

摘 要:基于凸优化中的对偶理论,提出了一种具有遗忘特性的在线学习算法框架。其中,Hinge函数的Fenchel对偶变换是将基本学习问题由批量学习转化为在线学习的关键。新的算法过程是通过以不同方式提升含有约束变量的对偶问题实现的:(1)梯度提升;(2)贪婪提升。回顾了以往的相关研究工作,并指出了与之的区别与联系。人造数据集和真实数据集上的实验结果证实了算法框架的有效性。算法可以很好地处理数据流中的分类面漂移问题,为设计和分析新的在线学习算法提供了一个新的思路。

关键词:在线学习;Fenchel对偶;梯度提升;贪婪提升
中图分类号:TP391 **文献标志码**:A **文章编号**:1001-2486(2014)04-0188-07

An online learning algorithmic framework with forgetting property

SUN Boliang, *LI Guohui*
(College of Information System and Management, National University of Defense Technology, Changsha 410073, China)

Abstract: Based on the notion of duality in convex optimization, a novel online learning algorithmic framework with forgetting property is proposed. The Fenchel conjugate of hinge functions is a key to transfer the basic learning problem from batch to online. New online learning algorithms were derived by different dual ascending procedures: (1) gradient ascent; (2) greedy ascent. Earlier researches were reviewed. Detailed experiments on synthetic and real-world datasets verified the effectiveness of the approaches. An important conclusion is that our derived online learning algorithms can handle the settings where the target hypothesis is not fixed but drifts with the sequence of examples, which paves a way to the design and analysis of online learning algorithms.

Key words: online learning; Fenchel conjugate; gradient ascent; greedy ascent

数据收集方法的多样化和存储技术的快速发展使得收集大量数据变得相当容易,如何对这些大量数据进行有效的处理一直是人们所关心的问题。在线学习[1-4,14-15]作为处理大规模实时预测问题的方法在近年来受到研究者的广泛关注。

在线学习过程是在一个序列的学习周期中进行的。在每个学习周期的开始,学习器将获得一个训练样本点,进而对该样本点的标签进行预测。在得到训练样本点的实际标签之后,学习器将对预测模型进行更新,以便可以对以后的样本进行更加准确的预测。在整个学习过程中,学习器不需要使用任何数据的分布信息,因此学习器可以不存储或者仅存储少量训练数据。此外,学习过程分散在不同学习周期内进行,故模型更新较为迅速。这使得对于任意时刻的预测请求,学习器总可以使用最新的模型进行预测。

现有的在线学习算法一般可以视作为基于Zinkevich[5]所提出的在线凸规划。这类算法是在原问题中进行的,一般的算法过程是定义一个即时损失函数进行随机梯度下降,从而避免对原函数的直接优化。在渐进意义下,基于在线凸规划的算法与批量学习算法的性能是相当的。这类算法主要有:感知器[3](Perceptron),在线主动-被动式学习[6](online passive-aggressive algorithms),在线流形正则化[7](online manifold regularization)等。这类算法之间的差别主要是如何利用当前样本构建即时损失函数。

在线凸规划是一种相对保守的更新策略,因为已有样本在学习器中的系数(权重)在以后的学习过程中是不会再次被更新的。本文利用目标函数在对偶问题中的特性[8],进一步分析了对偶问题中在线学习的特点,提出了一种具有遗忘特性的在线学习算法框架。针对学习器中已有样本的更新问题,在不违反对偶变量约束的条件下,在

* **收稿日期**:2013-11-12
作者简介:孙博良(1986—),男,河南洛阳人,博士研究生,E-mail:sunboliang@nudt.edu.cn;
李国辉(通信作者),男,教授,博士,博士生导师,E-mail:guohli@nudt.edu.cn

每个学习周期中加入了一个遗忘变量来实现对对偶问题的更大程度的提升。新的在线学习算法是通过实现不同程度的对偶提升过程得到的。两个人造数据集和一个真实数据集上的实验结果证实了算法框架的有效性。本文还讨论了各种算法处理数据流中的分类面固定和分类面漂移问题的能力。

本文的数学符号表示为:斜体字母表示的是标量,加粗的字母表示向量;Hinge 函数定义为 $[a]_+ = \max(a,0)$,定义 $\langle \boldsymbol{\omega}, \boldsymbol{x} \rangle$ 为两个向量的内积。

1　一种新的在线学习算法框架

设样本数据流为 $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_T, y_T)$,其中,$\boldsymbol{x}_t \in \mathbf{R}^n$ 为训练样本点,y_t 为其相应的标签($t \in \{1,2,\cdots,T\}$)。为了描述的准确性和简洁性,本文主要研究线性两分类问题,也就是说分类器的形式为 $f(\boldsymbol{x}_t) = \langle \boldsymbol{\omega}, \boldsymbol{x}_t \rangle$ 且 $y_t \in \{1, -1\}$(这里忽略了一般线性分类器中的偏移量 b 来移除类似求解 SVMs 时的等式约束,并减少计算复杂度[6])。那么,基本的批量学习(batch learning)算法可以描述为最小化:

$$J(\boldsymbol{\omega}) = \frac{1}{2}\boldsymbol{\omega}^2 + C\sum_{t=1}^{T}[1 - y_t\langle \boldsymbol{\omega}, \boldsymbol{x}_t \rangle]_+ \quad (1)$$

式中,右边第一项为正则化项,第二项是对不同样本的 Hinge 损失函数,C 是权重参数。本文采用凸优化中的对偶理论,分析公式(1)在对偶问题中的特性,进而得到新的在线学习算法框架。

1.1　在线学习的对偶视角

在描述对偶问题之前,首先回顾一下 Fenchel 对偶的定义,这是本文的主要理论分析工具。函数 $f: dom\, f \to \mathbf{R}$ 的 Fenchel 对偶定义为:

$$f^*(\boldsymbol{\lambda}) = \sup\{\langle \boldsymbol{\lambda}, \boldsymbol{\omega} \rangle - f(\boldsymbol{\omega}) : \boldsymbol{\omega} \in \mathrm{dom}\, f\} \quad (2)$$

特别地,对 Hinge 函数的 Fenchel 对偶变换是将公式(1)由批量学习转化为在线学习的关键。

定理 1　令 $f(\boldsymbol{\omega}) = [\gamma - y\langle \boldsymbol{\omega}, x \rangle]_+$,其中 $\gamma \geqslant 0$ 且 $x \in \mathbf{R}^n$。那么,$f(\boldsymbol{\omega})$ 的 Fenchel 对偶为

$$f^*(\boldsymbol{\lambda}) = \begin{cases} -\alpha\gamma & \text{if} \quad \boldsymbol{\lambda} \in \{-\alpha y x, \alpha \in [0,1]\} \\ \infty & \text{otherwise} \end{cases} \quad (3)$$

证明　可见 Hinge 函数 $f(\boldsymbol{\omega})$ 的形式可以改写为:

$$f(\boldsymbol{\omega}) = [\gamma - y\langle \boldsymbol{\omega}, \boldsymbol{x} \rangle]_+ = \max_{\alpha \in [0,1]} \alpha(\gamma - y\langle \boldsymbol{\omega}, x \rangle)$$

基于 Fenchel 对偶的定义,可以得到:

$$\begin{aligned} f^*(\boldsymbol{\lambda}) &= \max_{\boldsymbol{\omega}}(\langle \boldsymbol{\lambda}, \boldsymbol{\omega} \rangle - \max_{\alpha \in [0,1]} \alpha(\gamma - y\langle \boldsymbol{\omega}, \boldsymbol{x} \rangle)) \\ &= \max_{\boldsymbol{\omega}} \min_{\alpha \in [0,1]} (\langle \boldsymbol{\lambda}, \boldsymbol{\omega} \rangle - \alpha(\gamma - y\langle \boldsymbol{\omega}, \boldsymbol{x} \rangle)) \\ &= \min_{\alpha \in [0,1]} \max_{\boldsymbol{\omega}} (-\alpha\gamma + [\lambda + \alpha y \boldsymbol{x}, \boldsymbol{\omega}]) \\ &= \min_{\alpha \in [0,1]} (-\alpha\gamma + \max_{\boldsymbol{\omega}} [\lambda + \alpha y \boldsymbol{x}, \boldsymbol{\omega}]) \end{aligned}$$

由上式可见,若 $\lambda + \alpha y \boldsymbol{x} \neq 0$,$\max_{\boldsymbol{\omega}} [\lambda + \alpha y \boldsymbol{x}, \boldsymbol{\omega}]$ 为 ∞;反之,若 $\lambda + \alpha y \boldsymbol{x} = 0$,可以得到 $f^*(\boldsymbol{\lambda}) = -\alpha\gamma$。

回到在线学习问题上来,人们希望得到这样一个分类器序列 $\boldsymbol{\omega}_0, \boldsymbol{\omega}_1, \cdots, \boldsymbol{\omega}_T$,可以使得 $J(\boldsymbol{\omega}_0) \geqslant J(\boldsymbol{\omega}_1) \geqslant \cdots \geqslant J(\boldsymbol{\omega}_T)$。然而在式(1)中,在没有获得全部训练样本的条件下直接减小 $J(\boldsymbol{\omega})$ 的函数值是不可能的。实际上,在学习周期 t 中,可以使用的训练样本仅有 $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_t, y_t)\}$。为了避免上述的矛盾,本文使用 Fenchel 对偶对原问题的形式进行变换,并在对偶问题中重新解释在线学习过程。

式(1)可以改写为:

$$\min_{\boldsymbol{\omega}_0, \boldsymbol{\omega}_1, \cdots, \boldsymbol{\omega}_T} \frac{1}{2}\boldsymbol{\omega}_0^2 + C\sum_{t=1}^{T}[1 - y_t\langle \boldsymbol{\omega}_t, \boldsymbol{x}_t \rangle]_+$$
$$\text{s.t.} \quad \forall i \in 1,2,\cdots,T, \quad \boldsymbol{\omega}_i = \boldsymbol{\omega}_0 \quad (4)$$

使用 Lagrange 对偶定理,式(4)可以通过引入一个新的向量组 $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_T$ 来重新描述:

$$\max_{\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_T} \min_{\boldsymbol{\omega}_0, \boldsymbol{\omega}_1, \cdots, \boldsymbol{\omega}_T} \frac{1}{2}\boldsymbol{\omega}_0^2 + C\sum_{t=1}^{T}[1 - y_t(\boldsymbol{\omega}_t, \boldsymbol{x}_t)]_+ + \sum_{t=1}^{T}\langle \boldsymbol{\lambda}_t, \boldsymbol{\omega}_0 - \boldsymbol{\omega}_t \rangle$$

定义 $g_t(\boldsymbol{\omega}_t) = [1 - y_t\langle \boldsymbol{\omega}_t, \boldsymbol{x}_t \rangle]_+$。考虑对偶函数:

$$\begin{aligned} &D(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_T) \\ &= \min_{\boldsymbol{\omega}_0, \boldsymbol{\omega}_1, \cdots, \boldsymbol{\omega}_T} \frac{1}{2}\boldsymbol{\omega}_0^2 + C\sum_{t=1}^{T} g_t(\boldsymbol{\omega}_t) + \sum_{t=1}^{T}\langle \boldsymbol{\lambda}_t, \boldsymbol{\omega}_0 - \boldsymbol{\omega}_t \rangle \\ &= -\frac{1}{2}\Big(-\sum_{t=1}^{T}\boldsymbol{\lambda}_t\Big)^2 - \sum_{t=1}^{T} g_t^*(\boldsymbol{\lambda}_t) \end{aligned} \quad (5)$$

式中 g_t^* 为 g_t 的 Fenchel 对偶。可见,式(1)可以重新描述为最大化对偶问题:

$$\min_{\boldsymbol{\omega}} J(\boldsymbol{\omega}) = \max_{\lambda_1, \lambda_2, \cdots, \lambda_T} D(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_T) \quad (6)$$

根据定理 1 中的结论,可以约束对偶向量 $\lambda_t \in \{-\alpha C y_t \boldsymbol{x}_t, \alpha \in [0,1]\}$。基于上述分析,对偶函数可以由一个新的对偶向量 $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_T]$ 来描述:

$$D(\boldsymbol{\alpha}) = -\frac{1}{2}\Big(C\sum_{t=1}^{T}\alpha_t y_t \boldsymbol{x}_t\Big)^2 + C\sum_{t=1}^{T}\alpha_t \quad (7)$$

这样,可以通过更新 $\boldsymbol{\alpha}$ 来提升对偶函数 $D(\boldsymbol{\alpha})$。由式(7)可见,每一个训练样本 $(\boldsymbol{x}_t, y_t)$ 有一个与其相应的对偶系数变量 $\boldsymbol{\alpha}_t$,并且这些对偶系数之间是相互独立的。不同于原函数的下降问题,对偶函数的提升不需要使用所有的样本(将未获得

样本相应的对偶系数置为 0)。因此,对偶问题能够以在线的方式进行优化。根据文献[13]中的分析,对于任意对偶向量都对应着一个原问题中的边界向量。学习周期 t 中得到对偶向量 $\boldsymbol{\alpha}_t = [(\alpha_1)_t, (\alpha_2)_t, \cdots, (\alpha_T)_t]$ 所对应的边界向量为:

$$\boldsymbol{\omega}_t = C\sum_{i=1}^{T}(\alpha_i)_t y_i \boldsymbol{x}_i \tag{8}$$

基于以上分析,可以得到在线学习问题的一种对偶视角。本文从对偶问题出发,提出一种具有遗忘特性的在线学习算法框架。

1.2 具有遗忘特性的在线学习算法框架

在学习周期 t 中,可以进行更新的对偶系数变量有 $\alpha_1, \alpha_2, \cdots, \alpha_t$。因此,学习周期 t 中的对偶向量更新过程必须服从以下两个约束条件:

(1) 对于 $t < i \leqslant T$, $(\alpha_i)_t = 0$;

(2) 新的对偶向量满足 $D(\boldsymbol{\alpha}_t) \geqslant D(\boldsymbol{\alpha}_{t-1})$。

可见,第一个约束说明了未获得的样本不会对当前对偶函数的函数值产生影响,第二个约束说明了在线学习过程实际上是对偶函数的提升过程。

本文提出一种具有遗忘特性的对偶更新策略,在学习周期 t 中,引入一个新的变量 η_t,使得若 $i \neq t$,令

$$(\alpha_i)_t = (1-\eta_t) \times (\alpha_i)_{t-1} \tag{9}$$

由于 $\alpha \in [0,1]$,同样约束 $\eta_t \in [0,1]$。根据式(7)和式(8),学习周期 t 中需要提升的对偶函数形式可以改写为:

$$\begin{aligned} D_t(\alpha_t, \eta_t) &= -\frac{1}{2}\Big(C\sum_{i=1}^{t}(\alpha_i)_t y_i \boldsymbol{x}_i\Big)^2 + C\sum_{i=1}^{t}(\alpha_i)_t \\ &= -\frac{1}{2}\big((1-\eta_t)\boldsymbol{\omega}_{t-1} + C\alpha_t y_t \boldsymbol{x}_t\big)^2 \\ &\quad + C(1-\eta_t)\sum_{i=1}^{t-1}(\alpha_i)_{t-1} + C\alpha_t \end{aligned} \tag{10}$$

η_t 的初始值为 0。可见,η_t 的引入并没有违反对偶系数变量的取值范围约束。这种更新策略下的边界向量可以写为:

$$\boldsymbol{\omega}_t = (1-\eta_t) \times \boldsymbol{\omega}_{t-1} + C(\alpha_t)_t y_t \boldsymbol{x}_t \tag{11}$$

明显地,若 $\eta_t > 0$,以往样本在边界向量中的系数绝对值将会被减小,从而降低其对分类器的影响。所以,本文称 η_t 为遗忘变量,称这种在线学习算法框架带有遗忘特性。作为一个小结,图 1 中描述了具有遗忘特性的在线学习算法框架。

特别地,若使得遗忘变量等于一个固定常数,即 $\eta_t = f \in [0,1]$ $(t \in \{1,2,\cdots,T\})$,那么遗忘变量类似于 Levy 和 Lindenbaum[11] 在其工作中使用的遗忘因子。

INPUT: positive scalar C.

INITIALIZE: a coefficient vector $\boldsymbol{\alpha}_0$ and is associated decision boundary vector $\boldsymbol{\omega}_0$.

PROCESS:

For $t = 1, 2, \cdots, T$

Receive a new training point $\boldsymbol{x}_t$,

Predict $\hat{y}_t = sign(\langle \boldsymbol{\omega}_{t-1}, \boldsymbol{x}_t \rangle)$,

Receive the true label y_t,

Choose a parameter group (α_t, η_t) that satisfies

$$D_t(\alpha_t, \eta_t) \geqslant D_t(0,0).$$

Return a new coefficient vector $\boldsymbol{\alpha}_t$ and the boundary vector $\boldsymbol{\omega}_t$ using Eq. (11).

End for.

图 1 一种具有遗忘特性的在线学习算法框架

Fig. 1 An online learning algorithmic framework with forgetting property

2 基于不同对偶提升过程的衍生算法

上节提出了一种基于对偶提升过程的具有遗忘特性的在线学习算法框架。可见,利用不同对偶提升过程可以从中衍生出不同的在线学习算法。

2.1 梯度提升

式(10)中的对偶函数可以简单地在梯度方向上进行提升。由于 $\alpha_t, \eta_t \in [0,1]$,对偶变量的梯度提升过程可以描述为:

$$(\alpha_t)_t = \rho_t C[1 - y_t\langle \boldsymbol{\omega}_{t-1}, \boldsymbol{x}_t \rangle]_+ \tag{12}$$

$$\eta_t = \rho_t\Big[\langle \boldsymbol{\omega}_{t-1}, \boldsymbol{\omega}_{t-1} \rangle - C\sum_{i=1}^{t-1}(\alpha_i)_{t-1}\Big]_+ \tag{13}$$

其中,ρ_t 为步长。可见,若使得 $\eta_t = 0$,梯度提升过程对边界向量的更新与感知器算法是一致的。

记 I_t 为在学习周期 t 中将被更新的对偶变量集合,可以有 $I_t \in \{\alpha_t, \eta_t\}$。图 2 中展示了 I_t 的选择方法。

PROCESS: $I_t = \varnothing$,

If $1 - y_t\langle \boldsymbol{\omega}_{t-1}, \boldsymbol{x}_t \rangle > 0$, $I_t = I_t \cup \{\alpha_t\}$;

If $\langle \boldsymbol{\omega}_{t-1}, \boldsymbol{\omega}_{t-1} \rangle - \sum_{i=1}^{t-1}(\alpha_i)_{t-1} > 0$, $I_t = I_t \cup \{\eta_t\}$;

Return I_t.

图 2 梯度提升过程中对偶变量的选择方法

Fig. 2 The approach of choosing dual variables in gradient ascending procedure

进一步讨论步长的选择问题。记 $\rho_t^{\max}$ 为梯度

提升的最大步长。根据对偶变量的约束条件可以得到:

$$\rho_t^{\max} = \min\left\{\frac{1}{C[1-y_t\langle\boldsymbol{\omega}_{t-1},\boldsymbol{x}_t\rangle]_+},\frac{1}{[\langle\boldsymbol{\omega}_{t-1},\boldsymbol{\omega}_{t-1}\rangle - C\sum_{i=1}^{t-1}(\alpha_i)_{t-1}]_+}\right\} \tag{14}$$

而梯度提升中的最优步长 ρ_t^* 可以由以下公式得到:

$$\rho_t^* = \frac{\frac{\partial D_t}{\partial I_t},\frac{\partial D_t}{\partial I_t}}{\frac{\partial D_t}{\partial I_t},H(I_t)\frac{\partial D_t}{\partial I_t}} \tag{15}$$

式中,$H(I_t)$是在集合 I_t 上的 Hessian 矩阵。基于上述分析,若选择 $\rho_t \in [0,\ \min\{\rho_t^{\max},\rho_t^*\}]$,必然可以使得 $D_t((\alpha_t)_t,\eta_t) \geqslant D_t(0,0)$。

特别地,可以选择一个较小的固定步长 ρ 来更新对偶变量,这种方式称为 ρ - GA(ρ - Gradient Ascent)。若选择 $\rho_t = \min\{\rho_t^{\max},\rho_t^*\}$,在每个学习周期会有一个最大程度的对偶梯度提升,这种方式称为 AGA(Aggressive Gradient Ascent)。

2.2　贪婪提升

基于梯度提升的在线学习算法实际上采取的是一种保守的对偶提升策略。根据对在线学习的对偶分析,更大的对偶提升可以更快地接近原函数的最小值,从而更快地接近最优分类面,并减小在线学习过程中的错误率。实际上,在图 1 提出的算法框架下,可以采取更加贪婪的方法进行对偶提升。

作为一种比梯度提升更加贪婪的更新方式,可以通过求解以下的二次规划(QP)问题来更新对偶变量 α_t 和 η_t:

$$\begin{aligned}&\max \quad D_t(\alpha_t,\eta_t)\\ &\text{s.t.} \quad \alpha_t,\eta_t \in [0,1]\end{aligned} \tag{16}$$

3　与以往相关工作的联系与区别

3.1　与在线凸规划之间的关系

过去的在线学习算法往往可以归纳到在线凸规划[5](online convex programming)框架下。在线凸规划的基本思想是:“定义一个即时损失函数进行随机梯度下降,从而避免对原函数的直接优化”。在线凸规划中的步长往往被定义为一个退化的步长,例如 $\rho_t = 1/\sqrt{t}$。一个常用的即时损失函数是:

$$J_t(\boldsymbol{\omega}) = C[1-y_t\langle\boldsymbol{\omega},\boldsymbol{x}_t\rangle]_+ \tag{17}$$

基于对 $J_t(\boldsymbol{\omega})$ 梯度下降的更新过程可以写为:

$$\boldsymbol{\omega}_t = \boldsymbol{\omega}_{t-1} + \rho_t C y_t x_t \times \text{sign}([1-y_t\langle\boldsymbol{\omega}_{t-1},\boldsymbol{x}_t\rangle]_+) \tag{18}$$

根据本文提出的在线学习算法框架,将式(18)对应到对偶问题中,其更新方向为:

$$d' = (C \times \text{sign}([1-y_t\langle\boldsymbol{\omega}_{t-1},\boldsymbol{x}_t\rangle]_+),0)^{\mathrm{T}} \tag{19}$$

而本文定义的对偶问题的梯度提升更新方向为:

$$d = (C[1-y_t\langle\boldsymbol{\omega}_{t-1},\boldsymbol{x}_t\rangle]_+,\ [\langle\omega_{t-1},\omega_{t-1}\rangle - C\sum_{i=1}^{t-1}(\alpha_i)_{t-1}]_+)^{\mathrm{T}} \tag{20}$$

明显地,可以得到 $[d,d'] \geqslant 0$,因此式(18)实际上也定义了一个对偶问题的提升方向,也就是说这种在线学习方法同样可以由本文提出的在线学习算法框架衍生得到。不同的是,本文选择的对偶问题提升方向对已经存在于边界向量中的样本具有一种“遗忘”特性。

本文提出的算法与在线凸规划对在线学习问题采用了的不同的视角(对偶视角与原视角),可以视作在对偶视角下对在线凸规划的一种扩展。

3.2　与序列优化之间的关系

序列优化首先由 Platt[12] 应用于训练支持向量机(SVMs)。序列优化同样是在对偶问题中进行的,它的核心思想是将对偶问题中求解一个大型的 QP 问题的过程转化为求解一个序列的较小的 QP 问题。与序列优化相似,本文的工作希望通过基于时间序列的样本来优化对偶函数,并且在使用贪婪提升策略的情况下,同样是求解一些较小的 QP 问题。不同之处在于,本文将在线学习问题转化成了一个序列的对偶提升过程。在本文的算法框架下,对偶问题中的变量相互之间是相互独立的,因此能够以在线的方式进行优化。

4　实验与分析

本文通过在两个人造数据集和一个真实数据集上的实验来进一步证明算法的有效性。由算法描述可知,本文的算法仅包含向量之间内积的操作,因此可以使用核方法来寻找在线学习问题的线性分类面。使用 RBF 核函数,其形式为:

$$K(x_i,x_j) = \mathrm{e}^{-(x_i-x_j)^2/2\sigma_K^2} \tag{21}$$

式中 σ_K 为带宽参数。

在实验过程中,有以下几个约定:

(1)训练所使用的数据流是由数据集随机生成的(除了旋转螺旋数据集);

(2)每个样本在在线学习过程中仅被训练

一次;

(3)权重参数 C 使用栅格法在一个有限的范围中进行选择,本文选择的范围是 $C \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\}$;

(4)在 ρ - GA 中,选择的固定步长 $\rho = 0.1$;

(5)用于进行对比的批量学习算法为标准 SVMs。

为了避免不同的数据流对实验结果产生的影响,所有实验结果都是 5 个随机数据序列所产生结果的平均值[9-10]。本文中所有的实验都是在 MATLAB 平台下进行的。

4.1 双月型数据集

双月型数据集是一个用于测试两分类算法的经典数据集。本文中双月型数据集的生成方法来自于 http://manifold.cs.uchicago.edu/manifold_regularization/manifold.html 的实验中,设置生成模型的半径为 4,宽度为 2,每类数据的样本个数为 500。具体分布如图 3 所示。

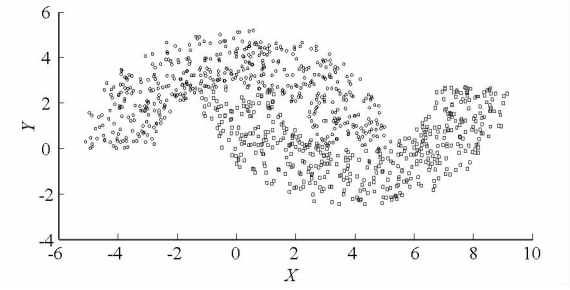

图 3 双月型数据集的分布

Fig. 3 Distribution of two moons dataset

在双月型数据集上,选择的带宽参数为 $\sigma_K = 0.35$[9]。标准 SVMs 在该数据集上的错误率为 0.4%。表 1 列出了所提出的几种在线学习方法的错误率。

表 1 在双月型数据集上的错误率

Tab. 1 Error rates on the two moons dataset

算法	ρ - GA	AGA	贪婪提升
错误率(%)	3.1(3.1)	2.2(2.6)	2.0(2.2)

表 1 中括号内的错误率为在不使用遗忘变量 η_t 时在线学习算法的错误率,即 $\eta_t = 0$。实验结果表明,贪婪的提升策略优于梯度提升策略,且遗忘变量的引入对在线学习的准确率具有一定的提升。

进一步给出实验过程中一些其他有意义的结果。图 4 说明了不同的在线学习算法对对偶函数的提升过程。明显地,对偶函数值在在线学习过程中是不断增加的,且贪婪算法可以更快地提升对偶函数的函数值。

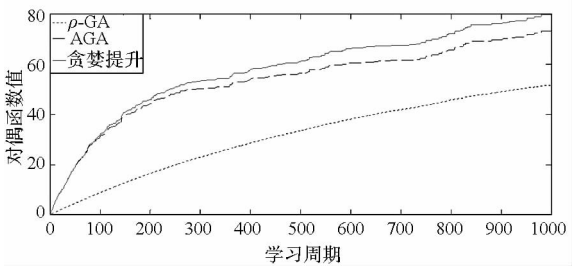

图 4 不同的在线学习过程中对偶函数的变化

Fig. 4 The value of the dual function during the different online learning processes

图 5 中比较了基于贪婪提升的在线学习过程中对偶函数与原函数的函数值变化曲线。对偶函数的函数值在在线学习过程中是不断上升的,相对地,原函数的函数值在学习过程中具有一个下降的趋势(可能存在一些小的波动)。两个曲线在在线学习过程中是相互靠近。这与第 2 节中理论分析所获得的结论是相符合的:(1)对偶函数值始终小于或等于原函数值;(2)可以通过提升对偶函数的方法来获得原函数中一个较优的边界向量。

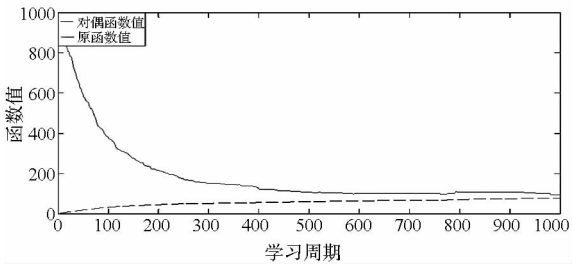

图 5 在线学习(贪婪提升)过程中对偶函数与原函数的函数值变化曲线对比

Fig. 5 The primal objective function vs. the dual function during the online learning process (greedy ascending approach)

图 6 中展示了基于贪婪提升的在线学习过程中所获得的分类器在整个数据集上的错误率。与图 5 中的实验结论相似,在线学习过程中的分类

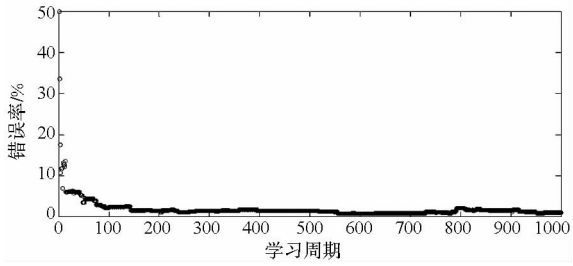

图 6 在线学习(贪婪提升)过程中获得的分类器在整个数据集上的错误率

Fig. 6 The error rates of the boundary vectors during the online learning process (greedy ascending approach)

器在整个数据集中错误率有一个下降趋势,同时在过程中会有一些较小的波动。这个实验同样也证明本文算法框架的有效性。

4.2　Isolet 数据集

Isolet 是 UCI 中的字母表语音数据集。本文使用其中的 Isolet1 和 Isolet5 组成算法的测试数据集,样本个数为 3119,样本维数为 617。本文的任务是将字母表中前 13 个字母的发音与后 13 个字母的发音进行两类区分。实验中,核函数的带宽参数为 10[9]。标准 SVMs 在 Isolet 数据集上的错误率为 5.95%。表 2 列出了本文提出的几种在线学习方法的在 Isolet 数据集上错误率。

表 2　在 Isolet 数据集上的错误率

Tab. 2　Error rates on the Isolet dataset

算法	ρ - GA	AGA	贪婪提升
错误率(%)	20.90(21.22)	10.21(10.83)	8.65(9.13)

在 Isolet 数据集上的实验结果同样证明了具有遗忘特性的在线学习算法框架的有效性。与双月型数据集上的结论相似:贪婪的提升策略优于梯度提升策略,且具有遗忘变量的在线学习算法具有较低的错误率。

4.3　旋转双螺旋形数据流

以上的两个实验都是在分类面相对固定的数据集上的进行的,而在线学习算法往往被希望能够处理分类面漂移(drifting)问题。为了进一步说明具有遗忘特性的在线学习算法的特点,本文在具有分类面漂移特性的旋转双螺旋形数据流上进行了测试。图 7 展示了旋转双螺旋形数据流中的数据流特点,基本的双螺旋形数据集在数据流中旋转了 360°。可见,样本的标签在数据流中会产生变化,而任何一个固定的分类面对旋转双螺旋形数据流的分类错误率都会达到 50%。

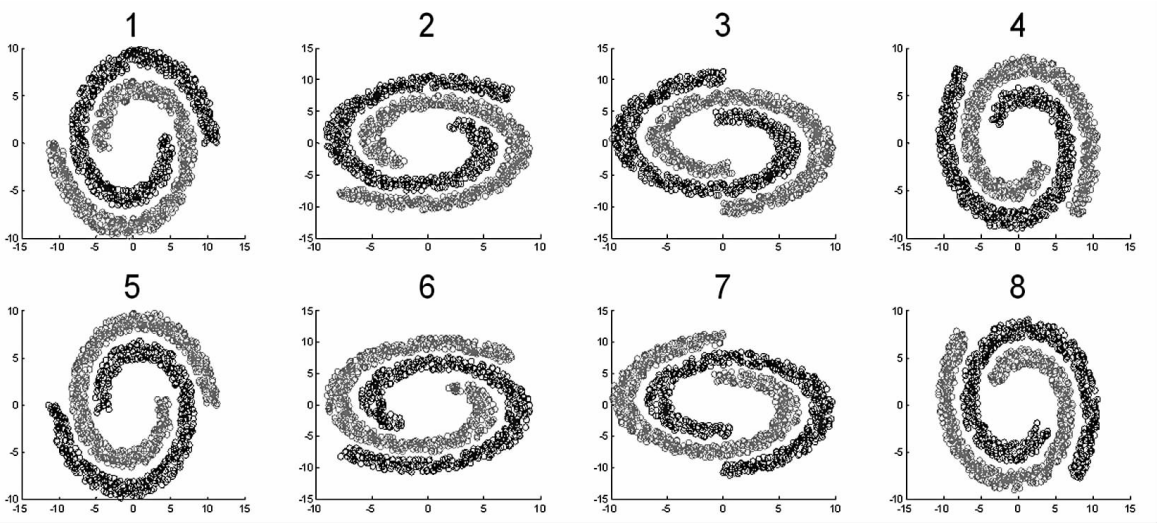

图 7　旋转双螺旋形数据流

Fig. 7　The two rotating spirals data sequence

在旋转双螺旋形数据集上,本文选择的带宽参数为 1[7]。表 3 列出了本文提出的几种在线学习方法在处理旋转双螺旋形数据流时的错误率。

表 3　在旋转双螺旋形数据流上的错误率

Tab. 3　Error rates on the two rotating spirals data sequence

算法	ρ - GA	AGA	贪婪提升
错误率(%)	5.90(6.25)	5.20(5.85)	2.65(3.65)

可见,具有遗忘变量的在线学习算法在旋转双螺旋形数据流中具有很低的分类错误率,即对在线学习的效果有很大的提升。明显地,对于旋转双螺旋形数据流,最近的样本更能够体现当前分类面的特性。而本文提出的具有遗忘特性的在线学习算法框架,对过去的样本具有一种“遗忘”特性,也就是降低了过去的样本在当前分类器中的权重,这种算法特性更加符合旋转双螺旋形数据流对分类器的要求,因而能取得较好的效果。

5　结论

本文提出的具有遗忘特性的在线学习算法框架是解决在线学习过程中已有样本权重更新问题的一种新方法。对偶问题中遗忘变量的引入可以在不违反约束条件的前提下,对已有样本的权重进行更新,更大程度地提升对偶函数的函数值。一个重要的结论是具有遗忘特性的在线学习算法在分类面固定和分类面漂移两种情况下都具有较

高的准确率。下一步的工作将集中在以下两个方面:(1)在线学习过程中的稀疏化方法;(2)半监督条件下的在线学习算法。

参考文献(References)

[1] Blum A. On-line algorithms in machine learning[M]. Springer Berlin Heidelberg, 1998.

[2] Kivinen J, Smola A J, Williamson R C. Online learning with kernels[J]. IEEE Transactions on Signal Processing, 2004, 52(8): 2165 - 2176.

[3] Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain [J]. Psychological review, 1958, 65(6): 386.

[4] Crammer K, Singer Y. Ultraconservative online algorithms for multiclass problems [J]. The Journal of Machine Learning Research, 2003, 3: 951 - 991.

[5] Zinkevich M. Online convex programming and generalized infinitesimal gradient ascent [C]//Proceedings of the 20th International Conference on Machine Learning. Menlo Park, CA: AAAI Press , 2003.

[6] Crammer K, Dekel O, Keshet J, et al. Online passive-aggressive algorithms [J]. The Journal of Machine Learning Research, 2006, 7: 551 - 585.

[7] Goldberg A B, Li M, Zhu X. Online manifold regularization: A new learning setting and empirical study [M]//Machine Learning and Knowledge Discovery in Databases. Springer Berlin Heidelberg, 2008: 393 - 407.

[8] Shalev-Shwartz S, Singer Y. A primal-dual perspective of online learning algorithms[J]. Machine Learning, 2007, 69(2 - 3): 115 - 142.

[9] Sun B, Li G, Jia L, et al. Online manifold regularization by dual ascending procedure [J]. Mathematical Problems in Engineering, 2013(6).

[10] Sun B, Li G, Jia L, et al. Online coregularization for multiview semisupervised learning[J]. The Scientific World Journal, 2013(2).

[11] Levy A, Lindenbaum M. Sequential karhunen-loeve basis extraction and its application to images [C]//International Conference on Image Processing. IEEE, 1998:456 - 460.

[12] Platt J C. Sequential minimal optimization: A fast algorithm for training support vector machine[J]. Advances in Kernel Methods, 1998.

[13] Rockafellar R T. Convex analysis[M]. Princeton University Press, 1997.

[14] Shalev- Shwartz S. Online learning and online convex optimization [J]. Foundations and Trends in Machine Learning, 2011, 4(2): 107 - 194.

[15] Mahdavi M, Yang T, Jin R. Stochastic convex optimization with multiple objectives[C]//Advances in Neural Information Processing Systems,2013: 1115 - 1123.