ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 31,No. 6,Nov. 2016,pp. 1067－1082
DOI:10.16337/j.1004-9037.2016.06.001

http://sjcj.nuaa.edu.cn
E-mail:sjcj@nuaa.edu.cn
Tel/Fax:＋86-025-84892742

在线学习算法综述

潘志松　唐斯琪　邱俊洋　胡谷雨

(解放军理工大学指挥信息系统学院,南京,210007)

摘　要:随着信息技术的迅猛发展,尤其是互联网行业的广泛应用,越来越多的领域出现了对海量、高速到达的数据实时处理需求。如何从浩瀚的"数据海洋"中挖掘有用的知识变得尤为重要。传统批处理模式的机器学习算法在面临大数据时变得力不从心,而在线学习通过流式计算框架,在内存中直接对数据实时运算,为大数据的学习提供了有力的工具,这类在线学习框架有望应对大数据背景下机器学习任务面临的困境与挑战。本文总结了经典和目前主流的在线学习算法,主要包括:(1)在线线性学习算法;(2)基于核的在线学习算法;(3)其他经典的在线学习算法;(4)在线学习算法的优化理论。本文介绍在线学习与深度学习结合方法的研究现状,探讨在线学习算法研究中的关键问题与应用场景,最后展望了在线学习下一步的研究方向。

关键词:在线学习;核;优化理论;概念漂移;深度学习

中图分类号:TP391　　**文献标志码**:A

Survey on Online Learning Algorithms

Pan Zhisong, Tang Siqi, Qiu Junyang, Hu Guyu

(College of Command Information Systems, PLA University of Science and Technology, Nanjing, 210007, China)

Abstract: With the development of information technology, especially the wide application of Internet-involved products, a large number of areas require real-time processing of massive and high velocity data. How to learn informative knowledge from "data ocean" becomes increasingly important. Traditional batched machine learning algorithms come to be pale when dealing with big data. However, the online learning framework employs streaming computing mode and deals with the data directly in the memory, which provides a promising tool for the learning of big data. This online learning framework has a bright prospect in facing difficulties and challenges when learning big data. This paper concludes the traditional and state-of-the-art online learning algorithms, the main contents include: (1) online linear learning algorithms; (2) online kernel learning algorithms; (3) other classical online learning algorithms; (4) optimization methods of online learning algorithms. Additionally, the implementation of online framework on deep learning models is then introduced to inspire interested researchers. Eventually, this paper discusses the key issues and some applications of online learning algorithms, which is followed by the research directions of the research direction.

Key words: online learning; kernel; optimization theory; concept drafting; deep learning

---

基金项目:国家自然科学基金(61473149)资助项目。
收稿日期:2016-11-02;**修订日期**:2016-11-09

引 言

随着自媒体、物联网和云计算等新兴技术的快速发展，产生了类别繁多、形态各异的海量数据，各类应用正全面进入大数据时代[1,2]。例如，全球存在的监控摄像头达到 1 亿个，每天产生的监控视频达到 2.3 ZByte；电商淘宝每分钟产生的订单量达到 8 300 多个。各个行业产生的业务数据大多数情况下可以看作动态达到的流式数据[3]，与传统数据相比，这类数据具有动态性、无序性、无限性、突发性和体积大等特点。首先，大批量的数据源源不断地涌入，将这类数据完全存储下来几乎不可能。其次，数据具有时间属性，带有强烈时间特征；训练样本和测试样本的分布可能不同、样本的特征可能随时间变化（增加或者缺失）、同时可能有新的类别产生，呈现动态变化的特点。这样一类数据分布动态变化的问题给机器学习带来一些深刻的变化和挑战。一方面，就拟合或预测未来数据而言，由于独立同分布假设显然不成立，因此不能像对待传统的学习问题那样，把在历史数据上训练得到的学习机器直接作用于未来的数据，传统的很多理论和方法都需要修正[4]。另一方面，从建模的角度，缺少独立性和同分布性，样本集的概率不能简单地再写成各样本概率的乘积。最后，日益丰富的应用问题中，人们不仅需要学习机器能很好地拟合或预测未来数据，同时也希望它能够揭示出数据的动态演化规律，从而让人们可以更好地理解数据。传统的学习方法归根结底是对某一静态数据分布的学习，没有提供学习数据分布变化规律的办法。这一问题逐渐引起机器学习和数据挖掘领域的重视[5,6]，并将分布随时间变化的数据称为非平稳数据（Non stationary data）或演化数据（Evolutionary data）。从时间效率的角度，提高学习算法对海量数据的处理效率迫在眉睫，将海量训练数据进行批处理的时间耗费往往成为制约实际应用的主要问题。传统基于独立同分布假设条件下的机器学习迎来了来自数据“流”的挑战。

针对此类呈数据流形态时刻到达的数据，传统的批处理式的学习方法一方面存在学习时间长、学习效率低的问题；另一方面难以针对增量数据有效地更新模型，导致难以有效地使模型适应数据中发生的概念迁移和概念演化问题。因此此类数据分布动态变化的问题给机器学习带来一些深刻的变化和挑战。而在线学习算法的流式计算模式则非常契合这类数据的特点，在大规模流式数据的处理中非常有效[7]。一方面，为提高海量数据的学习效率，在线学习假定训练数据是连续到达的，每次训练只利用当前到达的样本来更新模型，从而有效降低了学习复杂度；另一方面，通过读取一次片段数据并在训练完后保留少量的样本、按照时间先后次序利用数据流对模型进行更新，从而保留最新的类别信息。目前在机器学习领域，已经提出大量的在线学习算法及其变种。最早的在线学习算法可以追溯到 20 世纪 50 年代著名的感知器（Perceptron）算法[8]。感知器算法假设样本是线性可分的，当样本线性不可分时，需要采用基于核的感知器算法。近年来，研究人员提出了更加通用的在线核学习算法[9,10]。随着压缩感知技术的发展，利用 l_1 范数获取稀疏解得到了科研人员的关注。最著名的是 Tishirani 于 1997 年提出的最小收缩和选择算子（Least absolute shrinkage and selection operator，Lasso）[11]算法。在线学习获得最优解最简单的方法是截断梯度法（Truncated gradient method）[12]，此外还有前进后退分离法（Forward-backward splitting method）[13]以及正则化对偶平均法（Regularized dual average method）[14]等。上述稀疏在线学习算法的收敛速率为 $O(1/\sqrt{T})$，其中 T 为迭代次数，提高收敛速率是在线学习算法需要解决的关键问题，目前已经出现了很多收敛加速技术[15,16]。基于核的在线学习面临的问题是随着样本个数的增大，与当前样本学习相关的有效集合中支持向量的数目会越来越大。为了解决这个问题，学者们提出了一系列应对方法，比如投影法 Projectron[17]、遗忘法（Forgettron）[18]和随机固定缓冲区的感知器法（Randomized budget perceptron，RBP）[19]。近年来基于多任务的在线学习也成为研究热点，这类算法适应于生物医学等特定场景。与传统的单任务在线学习算法相比，基于多任务的在线学习具有同时学习多个核函数、多个任务的特征联合在线学习的能力。在线学习自提出以来已经发展出许多算法及其变种，根据模型是线性还是非线性模型，将在线学习算法分为两大类：在线线性学习算法和基于核

的在线学习算法。

1 在线线性学习算法

1.1 感知器算法

感知器(Perceptron)[8]是最经典的在线学习二分类算法,算法目标是学习得到一个线性分类面 $f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$。对于新的样本,用 $f(\boldsymbol{x})$的正负来判定它的类别标记。当 $f(\boldsymbol{x})>0$ 时将 $\boldsymbol{x}$ 判定为正类,否则将 $\boldsymbol{x}$ 判定为负类。假设在 t 时刻,获取到新到达的训练样本及其标记$(\boldsymbol{x}_t,y_t)$,这里 $\boldsymbol{x}_t\in\mathbf{R}^n$,$y_t\in\{\pm1\}$。$y_t=1$ 表示 $\boldsymbol{x}_t$ 属于正类,$y_t=-1$ 则表示 $\boldsymbol{x}_t$ 属于负类。当前分类模型记为 f_t,模型参数记作 $\boldsymbol{w}_t$。很明显,当 $y_t\boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{x}_t>0$ 说明模型 f_t 可以正确判定当前样本 $\boldsymbol{x}_t$ 的类别;否则说明 f_t 判定失败。从算法步骤(4)~(7)可以看出,感知器算法只在当前模型判定出现错误的情况下才会更新,每一次更新模型只需利用当前训练样本 $\boldsymbol{x}_t$,因此模型更新的计算复杂度很低。

算法 1 感知器算法

(1) 算法初始化:令 $w_1=0$

(2) for $t=1,2,\cdots,T$ do

(3) 收到当前时刻的训练样本$(\boldsymbol{x}_t,y_t)$

(4) if $y_t\boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{x}_t>0$ then

(5) $\boldsymbol{w}_{t+1}=\boldsymbol{w}_t$

(6) else

(7) $\boldsymbol{w}_{t+1}=\boldsymbol{w}_t+y_t\boldsymbol{x}_t$

(8) end if

(9) end for

(10) 算法输出:$\boldsymbol{w}_{t+1}$

可以通过随机打乱样本序列,计算算法在任意样本序列上判定失败的次数,即 $y_t\boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{x}_t\leqslant0$ 的次数来衡量感知器算法的效果。目前理论上已经证明,对于线性可分的样本集,在该样本集上无限次重复感知器算法,该算法判定失败的次数有限,亦即感知器算法可以收敛[20]。

1.2 稀疏在线学习算法

随着压缩感知(Compressive sensing,CS)技术的发展,基于 l_1 范数的稀疏优化技术引起了研究人员的广泛研究,例如 Lasso 通过利用 l_1 正则化约束,在回归的同时可以进行特征选择,降低了计算复杂度。传统批处理模式使用全体训练样本可以获得稀疏解。对于在线学习经常采用的随机梯度下降算法往往很难保证解的稀疏性,因此需要其他手段来获取稀疏解。截断梯度法[12]是获取稀疏解最简单直接的方法,当待更新的权值小于设定的阈值时则将权值置为 0;否则继续更新权值。截断梯度法的稀疏程度可以灵活控制,通过改变参数可以实现从无稀疏到全稀疏。另一种获得稀疏解的方法是 Duchi 和 Singer 提出的前进后退分离方法[13],该方法前进的步骤是对新到达的样本求其梯度并且更新权值,再利用 l_1 范数最小化的后退步骤获取稀疏解。即该方法获取稀疏解的过程可以分为两个步骤交替执行。每次迭代先运行一个无约束的梯度下降过程,接下来求解 l_1 正则化项最小化的优化问题,从而获得稀疏解,在此过程中保持两个步骤的求解结果尽可能接近。该方法的可拓展性很好,对于第 2 步的正则化项,根据问题解的需要,可以合理选择其他正则化项,比如 l_2 范数最小化、l_2 平方范数最小化以及 l_∞ 范数等。

正则化对偶平均法[14]也可以获得稀疏解。该方法的目标函数由两个凸函数相加组成,凸的损失函数和凸的正则化项,其中正则化项采用 l_1 范数来获得稀疏解。对偶平均法每次迭代更新变量时不仅与损失函数的偏梯度有关,而且与以往所有损失函数偏梯度的均值和整个正则化项都有关。使用 l_1 范数

作为正则化项的对偶平均法在获得稀疏解方面效果明显，其详细求解步骤可以分为 3 步：(1)求解损失函数的偏梯度；(2)计算以往所有损失函数偏梯度的均值；(3)通过优化变量的闭合解更新权值。以上 3 种获取在线学习稀疏解的方法可达到 $O(1/\sqrt{T})$的收敛率，其中 T 为算法迭代次数。算法计算复杂度越低往往收敛速率也比较慢。

2 基于核的在线学习算法

当样本线性不可分时，往往需要将样本特征向量 $\boldsymbol{x}$ 映射到高维再生核希尔伯特空间(Reproducing kernel Hillbert space) H，$\varphi(\boldsymbol{x}):R^{d+1}\rightarrow H$，从而将线性不可分问题转化为线性可分问题，而将样本映射到高维空间的非线性映射函数 φ 往往很难求得，实际算法往往采用核函数($K(\boldsymbol{x},\boldsymbol{y})=\langle\varphi(\boldsymbol{x}),\varphi(\boldsymbol{y})\rangle$，“$\langle\rangle$”表示内积运算)来实现非线性映射，利用核函数可以有效度量两个样本的相似度。

2.1 基于核的感知器算法

1.1 节介绍的感知器算法只能处理线性可分的情形。为了提高算法的分类能力，1999 年 Freund 和 Schapire 提出了基于核函数的非线性感知器算法[21]。与前文所述的感知器算法相比，仅在模型更新时不同，在基于核的感知器算法中，当算法在当前样本$(\boldsymbol{x}_t,y_t)$上判定错误时，更新模型时将 $y_tK(\boldsymbol{x}_t,\cdot)$加到当前模型。从理论上来看，基于核的感知器算法和线性感知器算法有很多相似之处，但在算法实现层面两者差别很大。对于线性感知器算法，算法迭代过程中只需存储和更新 1 个 n 维的向量 $\boldsymbol{w}_t$，而基于核的感知器算法迭代时需要存储和更新 1 个函数 f_t，其可以表示为

$$f_t(\cdot)=\sum_{i\in\varepsilon_t}y_iK(\boldsymbol{x}_i,\cdot) \tag{1}$$

式中：ε_t 为算法在 t 时刻之前所有判定错误的时刻集合。从式(1)可以看出，算法迭代过程中需要存储所有判定错误的样本及其标记信息，也就是说，基于核的感知器算法在求解时的存储复杂度与判定错误的次数呈线性增长趋势。在 t 时刻，获得样本$(\boldsymbol{x}_t,y_t)$需要计算

$$f_t(\boldsymbol{x}_t)=\sum_{i\in\varepsilon_t}y_iK(\boldsymbol{x}_i,\boldsymbol{x}_t) \tag{2}$$

很明显 $f_t(\boldsymbol{x}_t)$的计算复杂度随错误次数线性增长。因此基于核的感知器算法的计算复杂度随算法判定错误的次数线性增长。从以上分析也可以看出，随着时间的推移，样本个数越来越多，集合$\{\boldsymbol{x}_i\mid i\in\varepsilon_t\}$变得越来越大，因此核函数支持向量的个数逐步增长，这一方面会占用更多的存储资源，同时也提高了算法的计算复杂度，随着算法迭代次数的增大，模型更新将变得越来越慢。如果样本的个数趋向于无穷，那么支持向量的个数也将趋于无穷。为了解决这个问题，学者们提出了许多改进的核在线学习算法，例如基于核的在线梯度下降算法，通过截断法在核系数小于一定阈值时设其为 0，此外还有固定缓冲区(Fixed budget)的方法。

2.2 基于核的在线梯度下降算法

根据文献[9]，在希尔伯特空间中使用随机梯度下降法求解基于核的在线学习算法为

$$L(f,\boldsymbol{x}_t,y_t)=\min_{f\in H}\quad \ell(f(\boldsymbol{x}_t),y_t)+\frac{\lambda}{2}\|f\|_H^2 \tag{3}$$

模型更新表达式为 $f_{t+1}=f_t-\eta\partial_f L(f,\boldsymbol{x}_t,y_t)\mid_{f=f_t}$，这里 η 为学习率。由核的再生特性，$\langle f,K(\boldsymbol{x},\cdot)\rangle_H=f(\boldsymbol{x})$可得$\partial_f\ell(f,\boldsymbol{x}_t,y_t)=\ell'(f(\boldsymbol{x}_t),y_t)K(\boldsymbol{x}_t,\cdot)$。因此模型更新规则可以写为 $f_{t+1}=(1-\eta\lambda)\cdot f_t-\eta\ell'(f(\boldsymbol{x}_t),y_t)K(\boldsymbol{x}_t,\cdot)$。

令 $f_1=0$，则 f_t 可以表示为

$$f_t(\boldsymbol{x})=\sum_{i=1}^{t-1}\alpha_iK(\boldsymbol{x}_i,\boldsymbol{x}) \tag{4}$$

系数 α 的更新规则为:当 $i=t$ 时,$\alpha_t=-\eta\ell'(f(\boldsymbol{x}_t),y_t)$;当 $i<t$ 时,$\alpha_i=(1-\eta\lambda)\alpha_i$。由 $f_t(\boldsymbol{x})$ 的表达式可知,当样本个数趋向于无穷大时,那么核的项数也将趋向于无穷大,系数 α 的更新规则很好地解决了这个问题。每一次迭代,系数 $\alpha_i(i\neq t)$ 都会衰减 $(1-\eta\lambda)$ 倍,经过 k 次迭代后,系数 α_i 将衰减为 $(1-\eta\lambda)^k\alpha_i$,因此剔除对应系数 α_i 很小的项 $K(\boldsymbol{x}_i,\boldsymbol{x})$ 不会引起大的误差。

2.3 固定缓冲区的核在线学习算法

为解决基于核的在线学习算法中随样本个数的增加,支持向量的数目也不断增加的问题,核在线梯度下降算法采用截断法在核系数小于一定阈值时置其为 0,与此同时,还有一类固定缓冲区的核在线学习方法,例如投影法 Projectron[17]、遗忘法 Forgettron[18] 和随机固定缓冲区的感知器法 RBP[19]。当前到达的新样本判定错误时,投影法更新判定函数,然后将判定函数投影到原始空间,从而获得新的判定函数。如果投影前后的两个判定函数变化不大,那么更新当前判定函数为投影后的新函数;如果投影前后两个判定函数变化超过设定的阈值,则将当前到达的新样本加入到支持向量的集合中。遗忘法在每次迭代时,核函数的系数以常数值衰减,当固定缓冲区达到上限时,就把对应核函数系数最小值剔除,从而控制支持向量的数目。随机固定缓冲区的感知器法在每次迭代时,将那些判定错误的样本加入到支持向量的集合中,当固定缓冲区达到上限时,用新来的样本随机替换掉缓冲区中的一个旧样本,从而控制支持向量的数目。

3 其他经典的在线学习算法

第 1,2 节介绍的在线学习算法主要针对单任务学习(Single task learning,STL)问题,而现实生活中很多场景下的数据分析非常适合应用多任务学习(Multi task learning,MTL),比如自然语言处理、生物基因序列分析以及图片视频搜索等。多任务学习利用多个任务之间的相关性(例如特征共享、参数共享和子空间共享),学习任务之间的共性来避免模型欠拟合,从而提升算法的泛化能力。近年来出现了很多基于多任务的在线学习算法,如 Group Lasso 在线学习算法等,这类在线学习算法在某些场景下能更好地满足需求,也是在线学习算法的研究热点之一。

3.1 基于多任务的在线学习算法

假设有 Q 个任务,每个任务有 N^q 个样本 $D_q=\{\boldsymbol{z}_i^q=(\boldsymbol{x}_i^q,y_i^q)\}_{i=1}^{N_q}$,每个任务的空间分布各不相同,但是这些任务具有相关性。多任务学习的目标就是学习 Q 个判别函数 f_q,利用 $f_q(\boldsymbol{x}_i^q)$ 来估计 y_i^q。当 $Q=1$ 则退化为单任务学习。在多任务学习中,第 q 个任务的判别函数 f_q 是以权重向量 $\boldsymbol{w}_q$ 为参数的一个超平面,即 $f_q(\boldsymbol{x})=\boldsymbol{w}_q^{\mathrm{T}}\boldsymbol{x}$,$q=1,2,\cdots,Q$。$Q$ 个权重向量构成矩阵 $\boldsymbol{W}=(\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_Q)$。多任务学习的目标就是学习权重矩阵 $\boldsymbol{W}$,传统批处理模式的多任务学习目标表达式可以形式化为

$$\min_{\boldsymbol{w}}\quad \varphi(\boldsymbol{W})=\sum_{q=1}^{Q}\frac{1}{N_q}\sum_{i=1}^{N_q}\ell_q(\boldsymbol{w}_q,\boldsymbol{z}_i^q)+\lambda R(\boldsymbol{W}) \tag{5}$$

式中:λ 为正则化系数,$R(\boldsymbol{W})$ 为正则化项,$\ell_q(\boldsymbol{w}_q,\boldsymbol{z}_i^q)$ 为第 q 个任务第 i 个样本的损失。采用对偶平均在线算法来求解多任务学习优化问题,优化目标可以形式化为

$$\min_{\boldsymbol{w}}\quad \varphi(\boldsymbol{W})=\{\langle\overline{\boldsymbol{G}_t},\boldsymbol{W}\rangle+\lambda R(\boldsymbol{W})+\frac{\beta_t}{t}h(\boldsymbol{W})\} \tag{6}$$

式中:对偶变量 $\boldsymbol{G}_t=\partial\ell_t$ 为第 t 次迭代的损失函数在变量 $\boldsymbol{W}$ 上的偏梯度,$\overline{\boldsymbol{G}_t}$ 为前次迭代偏梯度的均值,$h(\boldsymbol{W})$ 为额外的强凸函数,$\{\beta_t\}_{t\geqslant 1}$ 为非负非递减序列。文献[22]给出了利用对偶平均法求解多任务在线特征选择的算法步骤,算法中正则化项取矩阵 $\boldsymbol{W}$ 的 ℓ_1 范数和 $\ell_{2,1}$ 范数的线性组合。其中 ℓ_1 范数控制稀疏性,用于特征选择,$\ell_{2,1}$ 范数用于学习任务之间的共有特征,算法的收敛率为 $O(1/\sqrt{T})$。

3.2 基于 Group Lasso 的在线学习算法

Group Lasso 是 Lasso 的一个扩展算法，假设样本可以表示为特征组的形式，$\boldsymbol{x}_i=[[\boldsymbol{x}_i^1]^{\mathrm{T}},[\boldsymbol{x}_i^2]^{\mathrm{T}},\cdots,[\boldsymbol{x}_i^M]^{\mathrm{T}}]^{\mathrm{T}}$，优化目标为权重向量 $\boldsymbol{W}=(\boldsymbol{W}_1^{\mathrm{T}},\boldsymbol{W}_2^{\mathrm{T}},\cdots,\boldsymbol{W}_M^{\mathrm{T}})^{\mathrm{T}}$，Group Lasso 的优化目标可以形式化为

$$\min_{\boldsymbol{W}}\left\{\sum_{m=1}^{M}\ell(\boldsymbol{W}_m,\boldsymbol{x}_i^m,y_i)+\frac{\lambda}{2}\|\boldsymbol{W}\|_1^2\right\}=\min_{\boldsymbol{W}}\left\{\sum_{m=1}^{M}\ell(\boldsymbol{W}_m,\boldsymbol{x}_i^m,y_i)+\frac{\lambda}{2}\Big(\sum_{m=1}^{M}\|\boldsymbol{W}_m\|_1\Big)^2\right\}\tag{7}$$

式中：$\ell(\boldsymbol{W}_m,\boldsymbol{x}_i^m,y_i)$ 为样本 $(\boldsymbol{x}_i,y_i)$ 在第 m 组特征上的损失。文献[23]给出了加速 Group Lasso 的在线学习算法，该算法引入加速技术，通过增加查询点 O_t，每次迭代求解的 g_t 是损失函数偏梯度在查询点 O_t 的值，算法不断更新查询点，算法目标表达式中附加了强凸表达式来保证查询点和优化权重向量尽可能靠近，从而保证算法持续加速，这种加速的 Group Lasso 在线算法可以达到 $O(1/T^2)$ 的收敛速率。

4 在线学习算法的优化理论以及相关优化算法

4.1 在线学习的“损失函数＋正则化项”优化框架

机器学习在线学习算法在优化理论中早已有所讨论，人们常常使用的术语是“增量学习”。其主要思路是当目标函数由一些子函数相加组成时，通过对每个子函数依次进行“首尾相接”的传递式梯度优化迭代最终得到原问题的最优解。在机器学习中，正则化损失函数优化问题一般都具有子函数和的形式，且子函数具有明确的含义，即每个子函数都表示某一个样本导致的正则化损失，因此此时的梯度增量算法可以表示为依次对每个样本导致的损失进行逐个优化，这就是在线学习的思想。目前大多数的机器学习算法都遵循正则化经验损失的设计框架。在线学习是机器学习的一种优化方法，也遵循这种理论框架。在具体应用中，一旦一个在线学习问题的损失函数确定以后，所面临的任务主要是如何求解正则化损失函数导致的优化问题。为了方便理解，以二分类问题为例，假设训练数据独立同分布，训练样本集可以表示为 $(\boldsymbol{x}_t,y_t)$，$x_t\in\mathbf{R}^n$，$y_t\in\{-1,+1\}$。机器学习优化问题可以写为

$$\min_{\boldsymbol{w}}F(\boldsymbol{w})=\lambda\|\boldsymbol{w}\|_p^{\eta}+\frac{1}{m}\sum_{(x,y)\in S}l(\boldsymbol{w};\boldsymbol{x},y)\tag{8}$$

式中：$l(\boldsymbol{w};\boldsymbol{x},y)$ 为损失项，用来控制优化模型的训练精度，λ 为正则化项系数，p 为选择的正则化类型。正则化项控制分类器的泛化性能。当前，机器学习领域常用的正则化项主要有

(1) ℓ_0 正则化：$r\|\boldsymbol{w}\|=\|\boldsymbol{w}\|_0=N$，$N$ 为向量 $\boldsymbol{w}$ 中非零元素的总个数。

(2) ℓ_1 正则化：$r(\boldsymbol{w})=\|\boldsymbol{w}\|_1=\sum_{i=1}^{n}|\boldsymbol{w}_i|$。

(3) ℓ_2 正则化：$r(\boldsymbol{w})=\|\boldsymbol{w}\|_2=\Big(\sum_{i=1}^{n}|\boldsymbol{w}_i^2|\Big)^{1/2}$。

(4) $\ell_1+\ell_2$ 混合正则化：$r(\boldsymbol{w})=\|\boldsymbol{w}\|_1+\|\boldsymbol{w}\|_2=\sum_{i=1}^{n}|\boldsymbol{w}_i|+\Big(\sum_{i=1}^{n}|\boldsymbol{w}_i^2|\Big)^{1/2}$。

当以上优化问题中的 $p=0$ 时，正则化项即为 ℓ_0 范数，此时的优化问题转化为了最原始的稀疏学习问题，由于该问题是 NP 完全的，目前无法有效求解这种非凸优化问题。而 ℓ_1 正则化是 ℓ_0 正则化的一种很好近似，一方面 ℓ_1 正则化可以得到凸优化问题；另一方面如果问题的解具有稀疏性，在一定条件下，ℓ_1 正则化可以求解特征稀疏问题。目前许多稀疏学习、压缩感知、传感网络和图像处理等问题都可以归结为 ℓ_1 正则化问题，具有广泛的研究和应用价值。当 $p=2$ 时，正则化项是 ℓ_2 范数，支持向量机最大“间隔”理论最初就是建立在 ℓ_2 正则化基础之上的。$\ell_1+\ell_2$ 混合正则化不仅能够保持 ℓ_1 正则化的结构特点，同时还满足强凸条件，目前不少学者在稀疏学习优化问题中加入 ℓ_2 正则化，从而将一般凸优化问题转化为强凸优化问题，极大地方便了相关算法的理论分析。损失函数控制模型的训练精度。直观上，对于分类问题来说，样本分对了则造成的损失为 0，分错了则损失为 1，使总的分类损失最小的解则

为最优解,这即是优化 0-1 损失的过程。但是由于直接优化 0-1 损失函数也是 NP 完全问题,目前无法有效求解。为此,很多学习算法选择了 0-1 损失函数的代理损失函数。目前机器学习领域常用的损失函数主要有:

(1)ℓ_1 损失:$l(\boldsymbol{w},\xi)=\max\{0,1-y\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}\}$。

(2)ℓ_2 损失:$l(\boldsymbol{w},\xi)=\max\{0,1-y\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}\}^2$。

(3)对数损失:$l(\boldsymbol{w},\xi)=\log(1+\exp(-y\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}))$。

(4)最小二乘损失或平方损失(Least-squares loss):$l(\boldsymbol{w},\xi)=(y-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x})^2$。

(5)指数损失(Exponential loss):$l(\boldsymbol{w},\xi)=\mathrm{e}^{-y\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}}$。

其中,Hinge 损失是连续非光滑的,不具有可导性,因此基于导数的优化方法不能直接应用,但可以使用其次梯度代替梯度,在处理分类任务时该损失使用较多。ℓ_2 损失一阶可导而二阶不可导,可以使用一阶梯度方法求解,但不能使用其二阶信息(如牛顿法)求解。值得一提的是,ℓ_2 损失相较于 Hinge 损失具有光滑性质,这使得 ℓ_2 损失应用更为广泛。对数、指数以及平方损失均高阶可导且强凸,基于高阶导数的优化方法可以直接应用。这些差异性使得对于不同的机器学习问题必须采用不同的方法求解。结合机器学习优化问题"损失函数+正则化项"的一般范式,一般的批处理学习算法可表述为如下的最优化问题,则

$$\min_{\boldsymbol{w}} F(\boldsymbol{w})=L(\boldsymbol{w})+\lambda R(\boldsymbol{w}) \qquad \text{s.t.} \quad \boldsymbol{w}\in\Omega \tag{9}$$

式中:$\boldsymbol{w}$ 为特征的系数,即要优化的变量;Ω 为 $\boldsymbol{w}$ 的解空间,由约束条件决定;$L(\boldsymbol{w})$ 刻画损失函数;$R(\boldsymbol{w})$ 为正则化项,用来刻画问题的先验信息对问题的约束,比如实际问题中蕴含的结构性特征、局部保持关系和流形结构等特性;λ 为标量参数,用于平衡损失函数和正则化项之间的关系。在大数据环境下,面对高维海量的情况,目前的计算设备无法支持批处理模式高昂的计算代价,于是学者利用分布并行的思想切分数据,如 Mapreduce 机制,实现分而治之的工程手段。但是分而治之的手段割裂了样本之间的关联性,特别对于数据流中蕴含的时间属性没有办法进行刻画。在线学习的学习原理正好符合数据流式到达的场景,此时的目标函数可以形式化为 $L(\boldsymbol{w})=L(\boldsymbol{w},\boldsymbol{x}_1,\boldsymbol{x}_2\cdots,\boldsymbol{x}_n)=\sum_{i=1}^{n}L(\boldsymbol{w},\boldsymbol{x}_i)$ 和 $R(\boldsymbol{w})=R(\boldsymbol{w},\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n)=\sum_{i=1}^{n}R(\boldsymbol{w},\boldsymbol{x}_i)$ 的形式。一个在线学习的"损失函数+正则化项"的一般理论框架可以描述为

$$\min_{\boldsymbol{w}} F(\boldsymbol{w})=L(\boldsymbol{w},\boldsymbol{x}_i)+\lambda R(\boldsymbol{w},\boldsymbol{x}_i) \qquad \text{s.t.} \quad \boldsymbol{w}\in\Omega \tag{10}$$

对于在线监督学习,如果假设第 i 个样本对应的标签值为 t_i,判别函数 $y_i=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i$,那么损失函数定义为 $L(\boldsymbol{w})=\sum_{i=1}^{n}(t_i-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i)^2$ 为最小均方无误差,模型成为线性回归最小二乘。如果损失函数变为 Hinge Loss:$L(\boldsymbol{w})=\sum_{i=1}^{n}\max\{(1-y_it_i),0\}$,模型称为支持向量机。如果损失函数定义为 $L(\boldsymbol{w})=\sum_{i=1}^{n}\ln(1+\exp(-y_it_i))$,则模型为 Logistic Regression。对于正则化项,如果取不同的函数形式,则与不同的损失函数组合时可以得到不同的模型。如基于 ℓ_1 正则化的分类器设计方法可以完成对特征的选择和学习一体化。简单取正则化项为:$R(\boldsymbol{w})=\|\boldsymbol{w}\|_1$,即表示系数向量中有许多向量为 $\boldsymbol{0}$,只有少数非零。可完成基于稀疏表示和分类器设计一体化的目标,则模型成为 Lasso,也即压缩感知或稀疏编码模型。如果 $R(\boldsymbol{w})$ 为 $\boldsymbol{w}$ 的 ℓ_2 范数,则模型为岭回归(Ridge regression)。如果损失函数为 Hinge loss,正则化项为 $\boldsymbol{w}$ 的 ℓ_1 范数,则导致稀疏 SVM 模型。由于上面提到的几类损失函数和正则化项都是可分的,因此可以表示成按照时间到达的样本和的形式,因此这些模型的在线模型有简单的形式。这些损失函数和正则化项的不同组合模型,在批量学习及在线学习的情况下,会衍生出不同的模型。根据不同的机器学习任务,可以衍生出多种在线的新模型算法。机器学习中,大部分训练数据集具有独立同分布的特点,并且样本

的冗余度通常较高，对于大规模数据库往往仅需要很少的训练样本就能得到所需泛化能力的分类器。同时，在线优化方法考虑样本随时间序列产生，即来一个样本训练一个，每次仅求解与单个样本有关的子优化问题，这使得每次迭代的计算复杂度很低，因此在线优化方法特别适合处理大规模数据库。

4.2 稀疏低秩的在线学习算法

稀疏表示的主要思想是将信号表示成一个字典(通常是超定字典，即字典个数大于维数)的稀疏线性组合。稀疏性是指系数向量中有许多分量为零，只有少数分量非零。稀疏编码可近似为向量的 ℓ_1 范数(或矩阵的 ℓ_1 范数)正则化优化问题。根据在线学习的理论框架，在线稀疏学习模型可以形式化为

$$\min_{\boldsymbol{w}} \frac{1}{2}\|Y-\boldsymbol{A}\boldsymbol{w}\|_{\mathrm{F}}^{2}+\lambda\|\boldsymbol{w}\|_{1} \tag{11}$$

样本的数据矩阵 $\boldsymbol{A}=[a_i]\in\mathbf{R}^{N\times d}$，$N$ 和 d 分别为样本数目和特征数目；Y 为目标值；$\|Y-\boldsymbol{A}\boldsymbol{w}\|_{F}^{2}$ 为重构误差，在此基础上得出基于低秩和稀疏的特征选择模型。根据对特征和高级特征描述，已知 N 和 M 分别为样本数目和高级特征数目，重新定义样本的高级特征标号 $A'=[a_i]\in\mathbf{R}^{N\times M}$ 和映射矩阵 $\boldsymbol{w}\in\mathbf{R}^{M\times M}$。低秩和稀疏的高级特征选择模型为

$$\min_{\boldsymbol{w}}\|\boldsymbol{A}'-\boldsymbol{A}'\boldsymbol{w}\|_{\mathrm{F}}^{2}+\lambda_1\|\boldsymbol{w}\|_{*}+\lambda_2\|\boldsymbol{w}\|_{2,1}+\lambda_3\|\boldsymbol{w}\|_{1,1} \tag{12}$$

式中：$\|\boldsymbol{w}\|_{*}$ 为 $\boldsymbol{w}$ 的奇异值之和，是对 $\boldsymbol{w}$ 的秩进行约束，从而发现相关的属性；$\|\boldsymbol{w}\|_{2,1}=\sum_{i=1}^{M}\sqrt{\sum_{j=1}^{M}\boldsymbol{w}_{ij}^{2}}$ 是对 $\boldsymbol{w}$ 的行进行约束，希望 $\boldsymbol{w}$ 的很多行为 0，从而发现不相关属性；$\|\boldsymbol{w}\|_{1,1}=\sum_{i=1}^{M}\sum_{j=1}^{M}|\boldsymbol{w}_{ij}|$ 对 $\boldsymbol{w}$ 中的元素进行进一步的稀疏约束，即希望矩阵 $\boldsymbol{A}$ 的重建系数尽量稀疏。

4.3 在线学习算法的优化快速求解

4.3.1 黑箱在线优化问题的求解

2003 年，Zinkevich[24] 提出了一种在线学习算法，其实质就是用优化理论中增量投影次梯度方法处理学习问题，这项研究在机器学习优化算法的发展过程中具有极其重要的影响，它规范了机器学习领域对在线学习的认识，明确了在线算法的理论分析工具——Regret bound，使得机器学习在线算法的研究有了统一的理论框架，彻底摆脱了长期以来一直依赖启发式策略的尴尬。Regret bound 还可以用来描述在线算法的收敛速度，比较不同算法的优劣。在目标函数凸性进一步增强的情形下，Hazan 在 2006 年针对强凸问题得到了比根号形式更好的对数形式 Regret bound[25]。遗憾的是，上述两项研究仅局限于理论分析。或许是由于缺乏必要的实验，在线算法的这些研究并未引起人们的充分关注。2007 年，Shai 在这些研究的基础上，将在线投影次梯度算法进一步发展为随机算法(Pegasos[26])，并对支持向量机问题进行求解。Pegasos 在大规模数据库上获得了很好的实验效果，处理 800 000 个样本的 RCV1 数据库仅需几秒钟的时间，与其他算法相比，在线算法的优势明显。至此，在线算法的研究无论是在理论还是应用上都取得了成功的第一步，也引起了众多学者的强烈关注。Pegasos 在 ℓ_2 正则化问题(SVM)上取得相当的成功后，人们自然想拓广这种方法求解一般形式的正则化损失函数优化问题。然而，在 ℓ_1 正则化问题上已有的在线算法却陷入了严重的困境，即将整个正则化项和损失函数不加区别地作为目标函数使用投影次梯度在线算法时，出现了无法获得稀疏解的现象，这意味着 ℓ_1 正则化项的结构在优化中没有得到保证，即使用 ℓ_1 正则化项的目的没有达到，此时人们意识到在在线优化中保证正则化项结构的重要性和必要性。

4.3.2 在线结构优化求解方法

2008 年 Duchi 提出了高效的 ℓ_1 投影算法[27]，为使用投影次梯度方法解决 ℓ_1 范数约束问题铺平了道路，但其解决的 ℓ_1 范数约束问题毕竟与 ℓ_1 正则化问题存在一定的区别，同时也付出了 $k\log n$ 的计算

代价(k 是解向量非零特征个数,n 是样本个数)。随后,还出现了一些如何在在线算法中保证 ℓ_1 正则化结构的工作。ℓ_1 正则化问题的稀疏性对在线算法的实用性提出了严峻的挑战,正是这种挑战促使了结构优化方法的诞生。2009 年,微软研究院的学者 Xiao Lin 发表了一篇具有重要影响的论文[14],将 Nesterov 的对偶平均优化算法[28,29]推广为正则化情形下的在线算法,很好地解决了 ℓ_1 正则化问题的稀疏性问题。随后,Duchi 等于 2010 年将经典优化方法镜面下降算法推广为在线形式,得到了一种更简洁的保证正则化项结构的在线算法[30],统一了在线算法研究方面很多零散的研究结果,通过 Regret bound 建立了在线优化和批处理优化之间的密切联系。由于机器学习最终关注的是泛化能力,一些研究者还讨论了在线优化算法的泛化能力。至此,在线学习的研究无论是在优化方法还是在统计分析方面均取得了令人瞩目的重要进展。

4.3.3 在线优化问题的 Regret bound 分析

在线优化算法的研究中,人们普遍关注的首要问题是算法是否达到了 Regret bound。通过对经典黑箱和结构的一阶优化方法的分析,知道经典方法都具有了最优 Regret bound,但遗憾的是,最终解大多都采取了加权平均的输出方式,仍然存在着一些弊端。特别是对 ℓ_1 正则化问题,尽管每一步迭代产生的解具有稀疏性,但平均求和的最终输出方式却破坏了这种求解算法最初极力维护的稀疏性。对于标准随机优化算法随机梯度下降(Stochastic gradient descent,SGD),只有当目标函数不仅强凸而且光滑时,才能相对比较容易地获得个体收敛速率界。对于单纯的强凸问题,如果不改变算法本身,也必须对平均的输出方式进行修改,才能获得最优收敛速率,但研究者对这些结果似乎一点也不满意。实际上,人们最为期待的莫过于 SGD 对于强凸问题能否达到最优个体收敛速率,这个问题看似简单,截至目前却始终没有答案。为了维护 SGD 的经典与尊严,Shamir 在 2012 年的机器学习顶级会议上把强凸目标函数下 SGD 的个体最优收敛速率作为开放问题提出[31]。2013 年,Shamir 等提出一种由平均输出方式收敛速率得到个体收敛速率的一般技巧[32],尽管得到了众盼所归的 SGD 个体收敛速率,但获得的收敛速率界与平均输出方式的收敛速率界相差一个对数因子,显然未能达到最优收敛速率界。在随机优化算法的个体收敛速率研究方面,Chen 等提出的 Optimal 正则对偶平均(Regularized dual averaging,RDA)获得了比较全面而又非常理想的结果[33]。其主要的思路是对对偶平均方法进行改进,在每一步迭代中增加一个不同形式子优化问题求解,对研究者通常独立讨论的一般凸、强凸或光滑等类型的问题,均获得了个体最优收敛速率。

2015 年,Nesterov 等在对偶平均方法的迭代中巧妙地嵌入了一种线性插值操作,证明了该方法在一般凸情形下具有最优的个体收敛速率,并且这种个体收敛呈现出与平均方式收敛同样的稳定性[34]。从理论角度来说,这种改动与标准的对偶平均方法区别极小,是对偶平均方法很好的扩展,也是对一阶梯度方法个体最优收敛速率比较接近期待的一种回答。仔细分析可以发现,Optimal RDA[33]和线性插值操作技巧获得个体收敛速率的原理不同[34],以至于获得的学习结果也不相同。一般地说,在处理 ℓ_1 正则化问题时,Optimal RDA 的个体解具有很好的稀疏性,但却不具有收敛的稳定性,这也是子优化问题的解直接作为最终输出的通用弊病;而线性插值技巧嵌入在迭代过程的梯度运算后,使最终的个体解具有很好的收敛稳定性,但却不具有稀疏性,这也是将插值累积作为最终输出的通用弊病。另外,这两种个体收敛速率分析的思路目前仅适用于步长策略灵活的对偶平均方法。值得指出的是,文献[33]对对偶平均算法的改动很大,这实际上与标准 SGD 个体收敛速率 open 问题的本意已经有所偏离,而线性插值操作技巧[34]对于强凸或光滑等目标函数情形能否得到个体最优收敛速率也未讨论。

5 在线学习与深度学习相结合

随着深度学习技术的不断发展,其高度非线性的模型表达能力、逐层抽象的特征提取能力使其在图像处理、自然语言处理和语音分析等领域得到了越来越广泛的应用。传统的深度学习算法其典型训练

方法是 mini-batch 的梯度下降算法，训练过程较为繁杂。深度学习中的模型结构固定，并没有考虑数据的时间序列属性，若数据流随着时间推移发生概念漂移、时间演化特性等问题，模型结构难以随之调整、严重滞后，难以保证模型表达能力。针对此问题，文献[35]采用在线增量学习框架，基于大规模流式数据调整降噪自编码器的特征维度。通过根据数据流调整特征维数应对由于数据分布发生变化而产生的概念漂移问题，通过深度学习与在线学习的结合，有效提高了深度学习对时间序列属性的刻画效果。首先，利用降噪自编码器作为深度学习的模块。降噪自编码器模型中的编码阶段和解码阶段的非线性变换分别为

$$\boldsymbol{h}=f(\boldsymbol{x})=\mathrm{sigm}(\boldsymbol{W}\boldsymbol{x}+b) \tag{13}$$

$$\hat{\boldsymbol{x}}=g(\boldsymbol{h})=\mathrm{sigm}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{h}+c) \tag{14}$$

式中：$\mathrm{sigm}(s)=\dfrac{1}{1+\exp(-s)}$为非线性激活函数，$\boldsymbol{h}$ 为编码过程的神经元输出，$\hat{\boldsymbol{x}}$为解码后的神经元输出。网络采用交叉熵损失作为损失函数，通过最小化损失函数作为优化网络参数的目标，则

$$\min_{\boldsymbol{W}} \quad \varphi(\boldsymbol{W})=\sum_{j}\psi(\boldsymbol{x}^{\langle j\rangle})=\sum_{j}\Big(-\sum_{i=1}^{D}\boldsymbol{x}_i\log\hat{\boldsymbol{x}}_i+(1-\boldsymbol{x}_i)\log(1-\hat{\boldsymbol{x}}_i)\Big) \tag{15}$$

式中：$\boldsymbol{x}$ 为自编码器神经元的输入，$\bar{\boldsymbol{x}}$ 为自编码器神经元的输出，j 为训练样本数目。其次，基于在线学习方法，通过统计交叉熵损失较大的训练数据个数的方式判断是否发生严重感念漂移，需要调整网络结构。若判断需要调整特征数量，则首先将一定数量的特征进行融合，其次添加一定数量的特征节点。随后，固定其他未变节点相关参数，利用交叉熵损失较大的训练数据集合，基于反向传播算法，对发生改变的节点对应参数进行训练，从而在调整网络结构后，高效地训练出收敛的更加针对近期数据的降噪自编码网络。通过上述机制，实现了在数据分布发生漂移的情况下微调网络结构、网络参数和深度特征，得到满足时间序列要求的在线并动态调整的网络，为深度学习的在线化问题提出了解决思路，并在一定程度上实现了深度学习模型中超参数的调整，对未来深度学习技术的在线化有较强指导意义。

6 在线学习算法研究的关键问题

目前随着信息技术的发展，产生了形态各异和类型繁多的数据，作为大数据处理手段的在线学习算法也面临着前所未有的困难和挑战。

6.1 在线学习算法的收敛性

在线学习算法作为大数据的处理手段，对算法的实时性要求很高，这就需要算法既要具有低的时间复杂度，又要有高的收敛速度。目前在线学习算法的收敛速率大部分都是 $O(1/\sqrt{T})$。这一收敛速率显然不能满足算法实时处理的需求。为了解决这个问题，很多学者提出了相关的加速在线学习算法。文献[15]提出了随机优化与在线学习的加速梯度算法，但这个加速算法的加速效果需要参数满足特定的条件，只有在特定情况下算法才能取得最优收敛速率 $O(1/T^2)$。Ji Shuiwang 等[16]提出了基于迹范数最小化的加速梯度在线学习算法，算法的最优收敛速率为 $O(1/T^2)$，但该算法基于批处理模式，还涉及到矩阵奇异值分解，因此不适合在线学习场景。文献[23]采用一种小批量加速技术，基于正则化对偶平均提出了一种加速算法。该加速算法计算复杂度低，收敛速率可以达到 $O(1/T^2)$，一定程度上解决了在线学习算法收敛速率低的问题。

6.2 在线学习算法的可扩展性

目前数据规模迅速膨胀，无法将全部数据存储下来。在线学习算法不需要存储数据，而是在内存中直接对数据进行实时运算。然而，对于非线性的在线学习算法，也就是基于核的在线学习算法，每次样本类别判定错误时就将当前样本加入到支持向量的集合中，理论上这个支持向量的集合大小可以达到

无穷大。为解决这个问题,现有方法大都是通过固定缓冲区的方法,包括前述的投影法、遗忘法和随机固定缓冲区的感知器法等,这类固定缓冲区的方法本质上是对所有样本的一种近似,这种近似难免存在误差,因此对于算法的性能会产生不利的影响,如何研究更加巧妙的方法来解决这个问题值得进一步探索。

7 在线学习算法在数据流上的应用

由于传统机器学习方法都可以在线化,而且在时间复杂度和计算效率上针对数据流等问题更有优势。本文就目前在线学习在一些具体的数据流学习上的应用,对在具体应用上的一些方法进行简述。

7.1 网络数据流的在线学习

网络数据流分类和异常检测任务的核心是以网络数据流为输入,快速准确地判断异常情况的发生,识别异常类型并进行预警。传统方法的基本思路是首先提取网络数据的流特征(Flow feature)将其作为刻画流的根本属性,然后采用贝叶斯方法、决策树、神经网络和支持向量机等算法对其进行分类。由于缺乏对数据流的概念漂移[36]和演化、大规模的高速特性、数据不平衡性以及非独立同分布情况的有效处理机制,传统框架无法客观反映网络流数据的本质特性,且不能满足网络数据流分类的特殊要求。数据流中的特征冗余、概念漂移和时间演化特性等问题在早期研究中已经有所反映。实际应用领域中的数据流概念可能是事例中若干特征组成的集合,也可能是若干特征隐含分布规律等,这种数据潜在概念随时间发生改变的现象称为"概念漂移"(Concept drifting)。而概念演化(Concept evolution)[37]是指随着时间的推移新类别或者新特征出现从而导致整个数据分布发生变化。无论是概念漂移还是概念演化都定义了记忆固定长度样本的时间窗来保证学习到漂移的概念,通常需要时间窗较长以保证学习到足够的样本,在大规模网络数据流的情况下,这样的配置使得空间和时间要求都呈指数增加。大量的数据导致运算困难,与此同时,针对高速网络数据流的学习模型选择也是困难的问题。例如,在骨干流量中的加密虚拟专用网络(Virtual private network,VPN)检测中[38],由于各种加密 VPN 采用的加密手段和加密机制都不相同,因此针对这些加密负载部分需要定义大量的数据以随机性检验特征值。重叠模板按照 8 位计算,产生的密数据特征接近 2 万维,这对于数据流的在线学习无疑是个不小的数字。同时,由于流随时间演化的特性,数据分布已经不能满足独立同分布的假设,因此模型结构或参数也不固定,但是数据流源源不断地到来,不同的流既有共同的特征组又有不同的分布,即为在多任务环境下完成对特征组的在线特征选择。

在线学习应用于数据流学习可以解决上述部分问题。首先,数据以流的形式到达,需要提供高效的在线学习,以方便对数据的实时处理,因此需要对数据以最精简高效的方式表示。针对动态数据流,模型的快速增量学习和演化策略,需要建立多层次语义的特征挖掘模型,进而,为保证最终分类模型的分类精度和泛化能力。其次,数据具有强烈的时间高层特征,每个时间片段所对应的类先验概率和类条件概率都可能变化,Yang[39]等同时学习多个时间片段中的共同特征,通过在线学习模式挖掘其中的短时关联关系,希望可以获得对多个时间片段各自的分类器都最优的特征组。更为重要的是,新类别不断的出现也是数据流的一个新特点。如,在网络异常检测中,网络异常通常包括各种网络故障、流量的异常表现和拥塞等,各种网络攻击层出不穷,数据是原数据中从未出现过的,因此要求新的在线学习方法能够自动地侦测当前要鉴别的流数据是原来数据中存在的还是新生成的。数据样本不断增长,而特征描述也在变化(增加和缺失),基于在线学习的预测和分析也需要对每个特征的重要性进行排序,实现对概念演化的辨识。用户希望通过对底层特征的描述以及多个特征的组合形成对新生成类别语义上的描述,在时间序列演变过程中能够实时根据高层特征的变化作出自适应的调整,获得对新生成类别的鉴别。总的来看,针对数据流分类问题目前主要采用在线学习算法、在线的特征表示和选择技术,研究在

线特征学习的新理论和新方法,针对网络数据流具有极其重要的研究意义和应用价值。

7.2 图像的检索与分类

经典的基于机器学习的图像检索[40,41]和分类[42,43]研究主要是在静态环境下展开的,研究者们通常假设用海量的训练样例来构建模型的标准距离度量。如传统的图像检索主要在图像数据库中进行,今天的图像检索和图像自动标注应用都在互联网上完成,对于训练和测试数据集的样本分布不同、图像的相关性很难获得统一的度量来定义,往往随着数据的不断涌现而实时地调整。Chechik[44]等应用被动-竞争算法(Passive-aggressive algorithms,PA)提出了一种在线可调整的图像相似性学习方法(Online algorithm for scalable image similarity learning, OASIS)。该算法采用"损失函数"+"正则化框架",利用最小化 Hinge 损失函数,可以快速地实现图像相似性的调整,在包含 270 多万个图像的数据集上进行了测试,结果显示具有良好的自适应性。Wang[45]等从在线 Flickr 图像组中学习从而能够在线调整相似性度量。该算法利用 103 个 Fickr 分组图像,利用分类器进行训练,计算分类器输出了 Label 特征向量距离作为图像的相似性度量。训练过程采用随机梯度在线学习算法,论文表明在 Corel 图像集和 Pascal VOC2007 图像集上性能优于其他传统的图像检索和分类方法。

7.3 无人机对地视频的目标跟踪和识别

由于目标跟踪和在线学习的特点十分吻合,因此对于目标的跟踪和识别问题采用在线学习的方法解决。Ross[46]等提出了使用增量学习实现鲁棒视觉跟踪,其基本思想是通过在线学习学习出一个低维子空间,然后定义一个样本投影到子空间的距离度量,用于度量样本点属于前景目标的似然,从而实现目标跟踪。检测跟踪以在线方式训练一个分类器把目标和背景分开,可较好地实现在线跟踪。但如果在产生跟踪训练数据时有误差,将导致不正确的训练样本标记,引起概念漂移。Babenko[47]等应用在线多实例学习,实现了一种鲁棒目标跟踪方法。为了克服稀疏表示跟踪计算量大的不足,Li[48]等提出了基于非稀疏线性表示的视觉跟踪。该方案有闭式解,并且不损失精度。为了捕捉不同特征维的相关信息,以在线方式学习出一个马氏距离度量,将其整合到优化问题中得到线性表示。在线度量学习显著改进了当目标外观有较大改变时跟踪的鲁棒性。针对多目标跟踪,Yang[49]等提出了一个学习非线性运动模式和鲁棒外观模型的在线方法。该方法构建一个在线非线性运动图以更好地解释方向变化,获取更好的运动仿射参数,设计了一个增量多实例学习方法生成强外观模型用于跟踪。实验表明其跟踪性能良好。Hare[50]等提出了一个在线结构输出学习的基于关键点的目标跟踪方法,将学习目标模型近似为学习一些二值基函数,这些基函数可在运行时有效计算,在具有挑战性的视频数据集上的实验表明该方法可显著改进现有的描述子匹配技术。文献[51,52]利用在线学习对视频帧的特征进行学习,提高了目标跟踪的精确度。视频处理[53,54]与人口密度估计[55-57]是计算机视觉近年来非常热门的应用领域。而随着无人机的广泛应用,无人机获取的视频越来越体现出海量信息的特征,同时对海量视频进行人口密度处理又有实时性的要求。像计算机视觉中的其他应用问题一样,该问题面临的主要困难是视频中巨大的数据量与视频语义之间的鸿沟。在线学习处理数据的方式恰恰直面这一挑战,因此很适合应用于基于无人机视频的人口密度估计问题。本文利用提出的在线稀疏学习和在线低秩学习方法进行密度估计。将采用在线监督学习(如在线 SVM 等)与在线非监督学习(如在线 K 均值聚类、在线字典学习与更新等)等方法来搭建基于无人机视频的人口密度估计系统。传统的批学习方法中,通常对于视频采用逐帧处理的方式,将视频简单地直接分为图片,然后对全部图像抽取特征进行训练以得到字典。此方法优点在于简单直接地将数字图像处理的算法应用于视频中。但同时也有如下缺点:(1)所有视频帧都需进行特征提取与人口数量估计,时间复杂度高,难以达到实时性要求;(2)在无人机的视频采集过程中,由于视频背景、光照条件都随着时间推移发生变化,因此存在一定程度的概念漂移,导致人口估计误差;(3)得到的人口估计结果不够平滑,出现锯齿现象。

在线学习方法进行视频的人口密度估计框架如下:给定一段视频,用时空滑动窗口把视频分成不同的窗口,然后在每个窗口中检测出时空兴趣点,随后提取一些特征描述这些时空兴趣点,如尺度不变特征转换(Scale-invariant feature transform,SIFT)特征、梯度直方图特征(Histogram of gray, HOG)、光流直方图(Histogram of flow, HOF)和颜色直方图等。用这些特征向量组成数据矩阵。随后进行字典学习和编码。不像传统的批学习方法,需要一些特定行为的视频段进行训练以得到字典。本文使用在线学习方法,直接从数据中在线学习并更新字典。稀疏与低秩学习在建模视频和图像时各有优缺点,而且现实世界的视频和图像有时适于用稀疏表示,有时适于用低秩表示,特别在含有较大噪声的情况下,低秩学习具有更好性能。因此本文提出的基于在线学习的无人机视频人口密度估计应比现有算法在实时性和稳定性上有所提升。

7.4 传统的机器学习任务

分类是在线学习的主要应用之一,Dagher[58]等提出了一个增量主非高斯方向分析方法用于人脸识别。该算法增量计算图像序列的主分量,无需估计协方差矩阵。同时把主分量转化为最大化信号源的非高斯独立方向,给出了两个在线算法计算人脸图像数据库的独立分量,人脸识别效果良好。Kapoor[59]等研究了内存受限的人脸识别,提出了一种有限的存储资源分配方法最大化判别能力以适用于流数据的分类,主要解决在线人脸识别的最近邻分类器期望值计算问题。在实际数据集上的实验结果说明了提出方法的有效性。由于缺少足够的训练信息、正常和异常定义的模糊性以及时间约束等,视频流中人的异常行为实时检测面临巨大的挑战,Bin Zhao[60]等提出了一个非监督动态稀疏编码算法用于检测视频中的异常事件。该算法从学习出的事件字典中在线重构查询信号的稀疏编码,基于这样一个直觉:即正常行为更可能从事件字典中重构,而异常行为则不是,提出了一个同时更新稀疏编码和在线字典的凸优化算法。字典在线更新避免了概念漂移,实验效果良好。Mallapragada[61]等研究了从给定图像对相似(Must-link)或不相似(Cannot-link)的成对约束中进行在线视觉字典裁减的问题,即如何选择视觉单词的一个子集使其能更好地解释图像对之间的关系,提出了一个基于对偶梯度下降的高效在线特征选择算法,将成对约束表示的边信息结合进特征选择步骤中,并增加一个组套索正则化项导出尽量多的零特征。用 PASCAL VOC 进行图像聚类实验说明了提出算法的良好效果。

8 结 论

海量数据的大规模、高速、动态性、多样性和数据不平衡性是数据流对现代模式识别和机器学习的挑战。现有的模型和方法不能直接用于数据流,尚有许多问题亟待解决。从数据流的特性来看,机器学习必须要解决以下几个问题:(1) 数据以流的形式到达,需要提供高效的在线学习,以方便对数据实时处理;因此需要对数据以最精简高效的方式表示。针对动态数据流,模型的快速增量学习和演化策略,需要建立多层次语义的特征挖掘模型,进而保证最终模型的精度和泛化能力。(2) 数据具有强烈的时间序列属性,每个时间片段所对应的类先验概率和类条件概率都可能变化,同时又具有一定共通性。因此在在线学习中除了通过增量学习使模型随流数据动态调整,还应学习多个时间片段中的共同特征,挖掘其中的关联关系,以获得对多个时间片段具有较好泛化能力的模型。(3) 新类别不断的出现也是数据流的一个新特点。如在网络异常检测中,网络异常通常包括各种网络故障、流量的异常表现、拥塞等,各种网络攻击层出不穷,数据是原数据中从未出现过的,因此要求新的在线学习方法能够通过流数据判断是否有新类生成,使分类能够根据高层特征的变化自适应地调整,获得对新生成类别的鉴别。(4) 数据样本不断增长,而特征描述也在变化(增加和缺失),因而面对规模庞大的数据流和不断变化的特征,在线学习更需要与特征学习结合,对特征组内部的相关性进行深入的挖掘和学习。

总体来看,在线学习算法由于具有实现简单、可拓展性强和算法性能优越等特点,海量数据处理方

面具有不可替代的作用。本文综述了经典的在线学习算法,并对每一种算法进行了简要的分析,也针对在线学习算法当前面临的关键问题进行了分析。目前来看,结合当前机器学习的最新发展趋势,在线学习算法还存在许多值得研究和有趣的研究方向,主要包括:(1) 在线学习与当前研究热点深度学习有待更加深入有效地融合。(2) 在线学习的分布式实现有待进一步探索和研究。(3) 在线学习能否与强化学习结合,有待进一步探索。

参考文献:

[1] 吴启晖,邱俊飞,丁国如. 面向频谱大数据处理的机器学习方法[J]. 数据采集与处理, 2015,30(4):703-713.
Wu Qihui, Qiu Junfei, Ding Guoru. Machine learning methods for big spectrum data processing[J]. Journal of Data Acquisition and Processing, 2015,30(4):703-713.

[2] Meng X F, Ci X. Big data management: Concepts, techniques and challenges[J]. Journal of Computer Research & Development, 2013,50(1):146-169.

[3] Dawei S, Guangyan Z, Weimin Z. Big data stream computing: Technologies and instances[J]. Journal of Software, 2014, 25(4): 839-862.

[4] Zhao Bin, Li Feifei, Eric P X. Online detection of unusual events in videos via dynamic sparse coding[C]// Computer Vision and Pattern Recognition. Colorado Springs: IEEE, 2010: 3313-3320.

[5] Pavan K. Mallapragada R J, Anil K J. Online visual vocabulary pruning using pairwise constraints[C]// Computer Vision and Pattern Recognition. San Francisco: IEEE, 2013:3073-3080.

[6] Koby C, Yoram S. Ultraconservative online algorithms for multiclass problems[J]. Journal of Machine Learning Research, 2003, 3:951-991.

[7] Hoi S C H, Wang J, Zhao P, et al. Online feature selection for mining big data[C]// Proceedings of the 1st International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications. [S. l.]: ACM, 2012: 93-100.

[8] Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain[J]. Psychological Review, 1958, 65(6):386-408.

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

[10] Crammer K, Dekel O, Keshet J, et al. Online passive-aggressive algorithms[J]. Journal of Machine Learning Research, 2006, 7(3):551-585.

[11] Tibshirani R. Regression shrinkage and selection via the Lasso: A retrospective[J]. Journal of the Royal Statistical Society, 2011, 73(3):273-282.

[12] Langford J, Li L, Zhang T. Sparse online learning via truncated gradient[J]. Journal of Machine Learning Research, 2008, 10(2):777-801.

[13] Duchi J, Singer Y. Efficient online and batch learning using forward backward splitting[J]. Journal of Machine Learning Research, 2009, 10(18):2899-2934.

[14] Xiao L. Dual averaging method for regularized stochastic learning and online optimization[C]// Conference on Neural Information Processing Systems 2009. Vancouver, British Columbia, Canada:[s. n.], 2009:2543-2596.

[15] Hu C, Kwok J T, Pan W. Accelerated gradient methods for stochastic optimization and online learning[C]// Conference on Neural Information Processing Systems 2009. Vancouver, British Columbia, Canada:[s. n.], 2009:781-789.

[16] Ji S, Ye J. An accelerated gradient method for trace norm minimization[C]// International Conference on Machine Learning. Montreal, Quebec, Canada:[s. n.], 2009:457-464.

[17] Orabona F, Keshet J, Caputo B. Bounded kernel-based online learning[J]. Journal of Machine Learning Research, 2009, 10 (6):2643-2666.

[18] Dekel O, Shalev-Shwartz S, Singer Y. The forgetron: A kernel-based perceptron on a budget[J]. Advances in Neural Information Processing Systems, 2008, 37(5):1342-1372.

[19] Cavallanti G, Cesa-Bianchi N, Gentile C. Tracking the best hyperplane with a simple budget perceptron[J]. Machine Learning, 2007, 69(2/3):143-167.

[20] Duda R O, Hart P E, Stork D G. Pattern classification[M]. USA: John Wiley & Sons, 2012.

[21] Freund Y, Schapire R E. Large margin classification using the perceptron algorithm[J]. Machine Learning, 1999, 37(3):

277-296.
[22] Yang H, Lyu M R, King I. Efficient online learning for multitask feature selection[J]. ACM Transactions on Knowledge Discovery from Data, 2013, 7(2):1693-1696.
[23] Li Z J, Li Y X, Wang F, et al. Efficient and accelerated online learning for sparse group lasso[C]// IEEE International Conference on Data Mining Workshop. [S. l.]: IEEE, 2014:1171-1177.
[24] Zinkevich M. Online convex programming and generalized infinitesimal gradient ascent[C]// International Conference on Machine Learing. Washington, DC: [s. n.], 2003:928-936.
[25] Hazan E, Kalai A, Kale S, et al. Logarithmic regret algorithms for online convex optimization[J]. Machine Learning, 2007, 69(2/3):169-192.
[26] Shalev-Shwartz S, Singer Y, Srebro N, et al. Pegasos: Primal estimated sub-gradient solver for SVM[J]. Mathematical Programming, 2011, 127(1):3-30.
[27] Duchi J, Shalev-Shwartz S, Singer Y, et al. Efficient projections onto the 1-ball for learning in high dimensions[C]// International Conference on Machine Learning. [S. l.]: ACM, 2008:272-279.
[28] Nesterov Y. A method of solving a convex programming problem with convergence rate $O(1/k2)$[J]. Soviet Mathematics Doklady, 1983, 27(2): 372-376.
[29] Nesterov Y. Primal-dual subgradient methods for convex problems[J]. Mathematical Programming, 2009, 120(1):221-259.
[30] Duchi J C, Shalev-Shwartz S, Singer Y, et al. Composite objective mirror descent[C]// Conference on Learning Theory. Haifa, Israel: [s. n.], 2010: 14-26.
[31] Shamir O. Open Problem: Is averaging needed for strongly convex stochastic gradient descent[C]// Conference on Learning Theory. Edinburgh, Scotland: [s. n.], 2012: 1-3.
[32] Shamir O, Zhang T. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes[C]// International Conference on Machine Learning. Atlanta, USA: [s. n.], 2013: 71-79.
[33] Chen X, Lin Q, Pena J. Optimal regularized dual averaging methods for stochastic optimization[C]// Advances in Neural Information Processing Systems. Lake Tahoe: [s. n.], 2012: 395-403.
[34] Nesterov Y, Shikhman V. Quasi-monotone subgradient methods for nonsmooth convex minimization[J]. Journal of Optimization Theory & Applications, 2015, 165(3):917-940.
[35] Zhou G, Sohn K, Lee H. Online incremental feature learning with denoising autoencoders[J]. Ann Arbor, 2012, 1001: 48109.
[36] Widmer G, Kubat M. Learning in the presence of concept drift and hidden contexts[J]. Machine Learning, 1996, 23(1):69-101.
[37] 张长水, 张见闻. 演化数据的学习[J]. 计算机学报, 2013, 36(2): 310-316.
Zhang Changshui, Zhang Jianwen. Learning on time-evolving data[J]. Chinese Journal of Computers, 2013, 36(2):310-316.
[38] Meng Juan, Yang Longqi, Zhou Yuhuan, et al. Encrypted traffic identification based on sparse logistical regression and extreme learning machine[C]// The 5th International Conference on Extreme Learning Machines. Singapore: [s. n.], 2014.
[39] Yang L, Hu G, Li D, et al. Anomaly detection based on efficient euclidean projection[J]. Security and Communication Networks, 2015, 8(17):3229-3237.
[40] Rui Y, Huang T S, Chang S F. Image retrieval: Current techniques, promising directions, and open issues[J]. Journal of Visual Communication and Image Representation, 1999, 10(1): 39-62.
[41] Gong Y, Lazebnik S, Gordo A, et al. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(12): 2916-2929.
[42] Ciregan D, Meier U, Schmidhuber J. Multi-column deep neural networks for image classification[C]// Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. [S. l.]: IEEE, 2012: 3642-3649.
[43] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]// Advances in Neural Information Processing Systems. Lake Tahoe: [s. n.], 2012: 1097-1105.
[44] Crammer K, Dekel O, Keshet J, et al. Online passive-aggressive algorithms[J]. Journal of Machine Learning Research, 2006, 7: 551-585.
[45] Wang G, Hoiem D, Forsyth D. Learning image similarity from flickr groups using fast kernel machines[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(11): 2177-2188.
[46] Ross D A, Lim J, Lin R S, et al. Incremental learning for robust visual tracking[J]. International Journal of Computer Vision, 2008, 77(1/3): 125-141.
[47] Babenko B, Yang M H, Belongie S. Robust object tracking with online multiple instance learning[J]. IEEE Transactions on

Pattern Analysis and Machine Intelligence, 2011, 33(8): 1619-1632.
[48] Li X, Shen C, Shi Q, et al. Non-sparse linear representations for visual tracking with online reservoir metric learning[C]// Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. [S. l.]: IEEE, 2012:1760-1767.
[49] Yang B, Nevatia R. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models[C]// Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. [S. l.]: IEEE, 2012: 1918-1925.
[50] Hare S, Saffari A, Torr P H S. Efficient online structured output learning for keypoint-based object tracking[C]// Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. [S. l.]: IEEE, 2012:1894-1901.
[51] Liu F, Shen C, Reid I, et al. Online unsupervised feature learning for visual tracking[J]. Image and Vision Computing, 2016, 51: 84-94.
[52] Li X, Shen C, Dick A, et al. Online metric-weighted linear representations for robust visual tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(5): 931-950.
[53] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos[C]// Advances in Neural Information Processing Systems. Montreal, Ganada: [s. n.], 2014: 568-576.
[54] Lv G, Xu T, Chen E, et al. Reading the videos: Temporal labeling for crowdsourced time-sync videos based on semantic embedding[C]// Thirtieth AAAI Conference on Artificial Intelligence. Phoenix, Arizona: [s. n.], 2016:221-231.
[55] Chan A B, Liang Z S J, Vasconcelos N. Privacy preserving crowd monitoring: Counting people without people models or tracking[C]// Computer Vision and Pattern Recognition. [S. l.]: IEEE, 2008: 1-7.
[56] Zhang C, Li H, Wang X, et al. Cross-scene crowd counting via deep convolutional neural networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S. l.]: IEEE, 2015: 833-841.
[57] Zhang Y, Zhou D, Chen S, et al. Single-image crowd counting via multi-column convolutional neural network[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S. l.]: IEEE, 2016: 589-597.
[58] Dagher I, Nachar R. Face recognition using IPCA-ICA algorithm[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(6): 996-1000.
[59] Kapoor A, Baker S, Basu S, et al. Memory constrained face recognition[C]// Computer Vision and Pattern Recognition (CVPR). [S. l.]: IEEE, 2012: 2539-2546.
[60] Zhao B, Fei-Fei L, Xing E P. Online detection of unusual events in videos via dynamic sparse coding[C]// Computer Vision and Pattern Recognition (CVPR). [S. l.]: IEEE, 2011: 3313-3320.
[61] Mallapragada P K, Jin R, Jain A K. Online visual vocabulary pruning using pairwise constraints[C]// Computer Vision and Pattern Recognition (CVPR). [S. l.]: IEEE, 2010: 3073-3080.

作者简介：

潘志松(1973-)，男，教授，博士生导师，研究方向：模式识别，E-mail：panzs@nuaa.edu.cn。

唐斯琪(1993-)，女，硕士研究生，研究方向：模式识别、图像处理。

邱俊洋(1989-)，男，博士研究生，研究方向：在线学习。

胡谷雨(1963-)，男，教授、博士生导师，研究方向：计算机网络、通信网络管理和网络智能化技术。