

基于非线性嵌入的自联想神经网络

吴昊东

(复旦大学 计算机科学技术学院, 上海 201203)

摘 要: 传统分类器常依赖于低维度子空间的特征进行分类,但仅在单个子空间下进行分类会因为不同类别的重叠而效果不佳。为此,提出一种基于流形学习的神经网络分类方法,利用非线性嵌入方法获得数据每个类的子空间,再使用非线性嵌入判别准则优化各个径向基函数自联想神经网络的参数。实验结果表明,该方法能有效解决类别重叠问题,分类准确率和鲁棒性高于传统分类方法。

关键词: 非线性嵌入;自联想神经网络;径向基函数;深度学习;模式识别

中文引用格式: 吴昊东. 基于非线性嵌入的自联想神经网络[J]. 计算机工程, 2017, 43(7): 203-210, 216.

英文引用格式: Wu Haodong. Auto-associative Neural Network Based on Nonlinear Embedding [J]. Computer Engineering, 2017, 43(7): 203-210, 216.

Auto-associative Neural Network Based on Nonlinear Embedding

WU Haodong

(School of Computer Science, Fudan University, Shanghai 201203, China)

[Abstract] Most classifiers prefer to classify high-dimensional data based on the characteristics of low-dimensional subspace. However, its performance suffers from overlapping among different classes in the single subspace. To address this issue, a classification method using manifold learning based neural network is proposed. This method uses nonlinear embedding to attain the subspace of each class, and nonlinear embedding criterion to optimize the parameters of each basis function auto-associative neural network. Experimental results show that this method is more robust and has higher accuracy than traditional methods.

[Key words] nonlinear embedding; auto-associative neural network; radial basis function; deep learning; pattern recognition

DOI: 10.3969/j.issn.1000-3428.2017.07.034

0 概述

维数约简常用于避免高维数据,如图像或文本中的维数灾问题。然而,传统的线性方法,如主成分分析(Principal Component Analysis, PCA)和独立成分分析(Independent Component Analysis, ICA),均假设数据由低维空间线性组成。当数据具有非线性分布时,线性维数约简往往不能有效获得其低维嵌入子空间^[1]。

非线性嵌入技术如等距映射(Isometric Mapping, Isomap)算法和局部线性嵌入(Locally Linear Embedding, LLE)^[2]算法等能在数据非线性分布时有效地获得其低维嵌入子空间,但该类算法的不足是:它们常常为所有数据构造一个单独低维子空间,由于不同类别的数据可能会在这个子空间中重叠,使得基于这些非线性嵌入算法的模型分类准确率不高^[3]。

根据现有研究^[4],为每个类别构造一个对应的子空间能有效解决数据重叠问题,从而提升模型分类准确率。因此,本文提出基于非线性嵌入的自联想神经网络(GSCD Nonlinear Auto-associative Modeling, GNAM)学习样本每一类对应的子空间^[5]的方法,并结合集成学习方法,从而实现更有效的分类。

1 自联想神经网络与径向基函数网络

自联想神经网络是一种前馈神经网络,利用 BP (Back Propagation) 算法进行训练。该网络学习的映射关系是输出等于输入。自联想神经网络的结构类似于含一个隐层的多层感知机,包括一个输入层、一个隐层和一个输出层。输入层到隐层的非线性映射可以被看作是一种非线性编码,隐层到输出层的映

基金项目: 教育部高等学校博士学科点专项科研基金(20120071110035)。

作者简介: 吴昊东(1991—),男,硕士研究生,主研方向为机器学习、神经网络。

收稿日期: 2016-03-01 **修回日期:** 2016-04-28 **E-mail:** hdwu@fudan.edu.cn

射则是解码过程。由于自联想神经网络的目的是使得网络输出等于输入,因此通常使用 BP 算法反向传递优化输出与输入的误差作为训练方法。

径向基函数神经网络(Radial Basis Function Neural Network, RBFNN)是 Moody 和 Darken 于 20 世纪 80 年代末提出的一种特殊的单隐层的 3 层前馈网络。径向基函数神经网络采用径向基函数为神经元的激活函数。径向基函数关于 n 维空间的中心点具有径向对称性,而且神经元的输入离中心点越远,神经元的激活程度就越低。这一特性被称为“局部特性”。

若使用 2 个径向基函数网络来替换自联想神经网络的隐层映射和输出层逆映射,则可以得到一个径向基函数自联想神经网络(Radial Basis Function Auto-Associative Neural Network, RBFAANN)。如图 1 所示。径向基函数自联想神经网络被应用在很多机器学习领域,如语音识别^[6]、丢失数据填补、风力机叶片损伤诊断等。本文提出的分类器也基于径向基函数自联想神经网络。

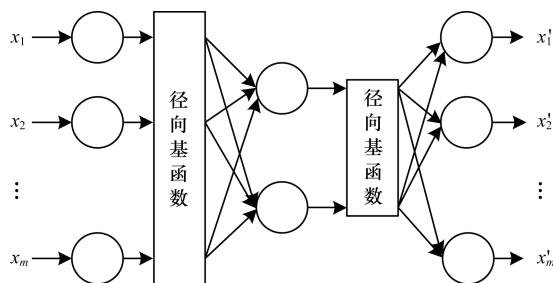


图 1 自联想径向基函数神经网络

2 非线性嵌入与 GSCD 标准

2.1 非线性嵌入

分析高维数据时常面临维数灾问题,一种解决方案是假定这类数据是由内在的低维变量生成的,如人脸图像识别问题。另外,认知科学研究表明人的感知可能是以低维流形方式进行的^[4]。

近年来,许多非线性的嵌入技术涌现。这些技术在分类^[7]、特征提取^[8]、数据可视化^[9]等方面都有较为广泛的应用。这些技术可以粗略地分为 2 类。1) 全局方法,如等距映射算法(Isomap)和最大方差展开(Maximum Variance Unfolding, MVU)。2) 局部方法,如拉普拉斯特征映射(Laplacian Eigenmaps, LE),局部线性嵌入(LLE)^[2],随机近邻嵌入(Stochastic Neighbor Embedding, SNE)等。这些方法能在低维子空间中很好地保留流形的某些性质,如拓扑不变性或保距性等。

为了更好地研究非线性嵌入,需要用一些方法来比较这些不同的嵌入技术。常见的定量分析非线性嵌入技术的嵌入质量的算法有标准如放大因子

(Magnification Factors, MF) 和主延伸方向(Principal Spread Direction, PSD)^[10]。尽管这 2 个标准能定量的反应嵌入质量,但却因为放大因子和主延伸方向对不同的嵌入技术导致的仿射变换敏感而很难用在一般的评判框架下使用它们。针对此问题,文献[11]提出新的评判标准 GSCD。这个标准能够评估非线性嵌入技术得到的低维流形的总体光滑性和数据点的主延伸方向的联合方向一致性,并且证明了仿射不变性,使得该标准的适用性大大提升。

2.2 GSCD 标准

2.2.1 ALSTD 标准

由于不同的嵌入技术在得到低维表示时使用了不同的缩放比例,为了使标准能够普遍适用,必须设计拥有仿射不变性的标准,ALSTD 就是这样一个标准^[11]:

$$ALSTD = \frac{1}{N} \sum_{i=1}^N STD_i \quad (1)$$

其中,STD 定义如下^[11]:

$$STD_i^2 = \frac{1}{k_i} \sum_{x_j \in N(x_i)} \text{lb}^2 MF(x_j) - \left(\frac{1}{k_i} \sum_{x_j \in N(x_i)} \text{lb} MF(x_j) \right)^2 \quad (2)$$

其中, k_i 表示高维样本 x_i 的近邻 $N(x_i)$ 的数量; N 为样本的数量; MF 表示从原空间到低维空间的拉伸或压缩程度,即前文提到的放大因子,文献[11]详细介绍了利用径向基函数(Radial Basis Function, RBF)来求解 MF 的方法。

2.2.2 ALCD 和 GSCD 标准

ALSTD 的最大缺陷是无法确定总体光滑性改变方向的控制方向,例如 MF 等于 4 时,一个二维空间的拉伸有 2 种可能情况, $1 \times 41 \times 4$ 和 $2 \times 22 \times 2$ 。所以需要设定标准来衡量嵌入算法的方向的一致性^[11]:

$$CD_i = \max_{\|w\|=1} \frac{1}{N_i} \sum_{i=1}^{N_i} \langle w, v_i \rangle^2 \in [0, 1] \quad (3)$$

其中, N_i 是数据点 z_i 的近邻数量; $v_i \in V$ 是 z_i 的主延伸方向 PSD,而向量 w 则是所有的 $v_i, i=1, 2, \dots, N_i$ 中角度最小的向量。

文献[11]详细介绍了 CD 的求解方法。为了在总体联合方向和局部联合方向之间得到平衡,根据 CD 定义标准 ALCD:

$$ALCD = \frac{1}{N} \sum_{i=1}^N CD_i \quad (4)$$

GSCD 可以同时描述联合方向一致性和光滑度,定义如下:

$$GSCD = \frac{1}{N} \sum_{i=1}^N \frac{STD_i}{CD_i} \quad (5)$$

不难发现,GSCD 分数越低,则嵌入算法的嵌入质量更好,即光滑度更好,联合方向一致性更高。GSCD 分数依赖前文提到的 RBF 函数中 σ 的选取,不同的 σ 会造成最后的 GSCD 分数不同。

2.2.3 GSCD 标准总结

文献[11]指出 GSCD 标准采用了简单的几何直觉,易于计算。这 4 个标准从几何角度评估了非线性嵌入算法的嵌入质量,而且通过实验证明了这些标准能一定程度上处理数据集外部样本(out-of-samples)。GSCD 标准除了可以对比不同的非线性嵌入算法,其本身的分数也可以比较同一个嵌入算法的性能优劣,该性质容易令人联想到利用 GSCD 标准来调节 RBF 函数参数 α 的可能性。

3 基于 GSCD 标准的自联想网络分类器

3.1 算法原理

本节以一个例子来介绍整个算法的原理。图 2 假定所有样本存在于三维空间,其中假设一共有 2 个类(A 和 B),每个类有 2 个子类(A1,A2 和 B1,B2),从图的左上方可以看到。其中,A 类样本被标记为三角形;B 类样本则被标记为十字。一个接近 A1 子类的测试样本被标记为圆,并且假设这个样本属于 A 类。

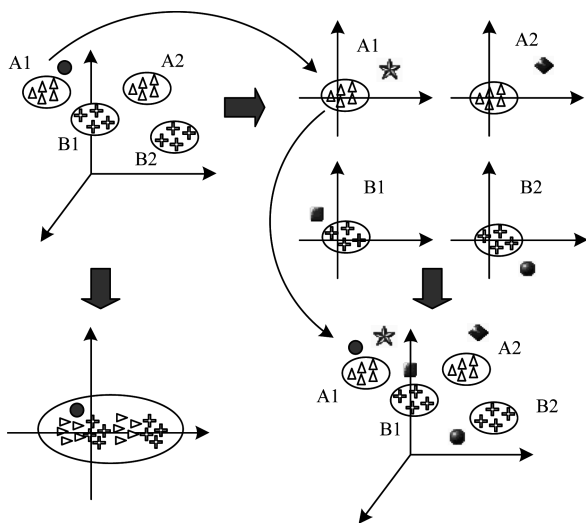


图 2 本文算法原理

如果不同类别的数据被共同投影到同一个二维空间,如图左下方所示,不同类别的数据很容易重叠,从而造成信息的压缩。反之,如果将各个子类通过非线性嵌入算法投射到各自的子空间,如图右下方所示,重叠的问题就可以被较好地解决。

为了解决流形学习算法不适应数据集外部样本问题,选择利用 GSCD 标准来调节一个径向基函数网络的参数以近似非线性嵌入算法的投射函数。

在将各个子类投射到各自子空间之后,在各自子空间下缺乏一个统一的度量方法,不利于对测试样本进行分类,所以需要一个相反的投影将各个样本重构到原来的空间中,如图 2 右下所示。直观上

来看,测试样本在经过 A1 子空间重构后应该具有最小的欧式距离。这样,该分类器就可以根据测试样本和其经过各个子空间得到的重构点的欧式距离来判别测试样本的类别。

3.2 分类器原理和架构

首先,来自每一类的数据被一种覆盖算法(covering algorithm)预聚类为若干个子类,其中子类的个数是预先定义的。相比其他的聚类方法如谱聚类^[12]和基于最大期望聚类法,本文的覆盖算法可以视为一种贪婪算法,能降低计算复杂度。其次,各个子类的维度分别通过上文提到的非线性嵌入算法实现降维。这样每个子类通过非线性嵌入算法降维均可以得到一个表示自身和其对应的低维子空间的映射。

然后,使用双向的径向基函数神经网络为每个子类组成一个自联想网络,其中,降维的 RBF 函数逼近非线性嵌入算法的维数约简函数,升维的 RBF 函数则用来重构高维空间数据。由于降维 RBF 逼近非线性嵌入算法,因此利用 GSCD 标准来训练 RBF 的参数对整个分类算法至关重要。升维的 RBF 函数利用验证集数据计算重构误差来选取合适的参数。

最后,当训练完每个子类的自联想网络分类器后,在假定当同类数据经过自己对应类别的分类器之后,重构误差应当最小的前提下,可以得到分类器的分类标准。

为了提高 GNAM 分类器的性能,本文还集成了一系列子类个数不同的 GSCD-NAM 分类器群,得到集成的 En-GNAM(Ensemble Version of GSCD Nonlinear Auto-associative Modeling)。En-GNAM 算法只需要知道子类的最大数目。

3.3 覆盖算法

虽然每一类的数据可以通过原有的一些聚类分析算法^[3]分割为多个子类,但是很多聚类算法只被设计用来处理特定的分布和形状。还有些聚类算法存在额外的问题,如谱聚类算法^[12]需要求解特征值和特征向量,计算复杂度相对较大;最大期望法无法保证全局最优且需要循环收敛获得最终聚类结果,其计算复杂度也较大。

为了解决以上问题,下面引入一种覆盖算法,该算法只需要子类数目一个先验知识,而不需要其他关于数据分布的知识。以下算法将每类数据预聚类为一组子类。该覆盖算法计算复杂度与样本个数和子类个数成线性关系,相比谱聚类和基于最大期望值聚类算法降低了聚类计算复杂度,另一方面,聚类的子类中心通过随机选取,可以视为贪婪算法。

算法的具体步骤如下:

步骤 1 给定 K 个类别, S 个子类和相关样本集合 (X, Y) , 其中, $X = \{x_i \in \mathbb{R}^n, i = 1, 2, \dots, N\}$ 是数据, $Y = \{y_i, i = 1, 2, \dots, N\}$ 是对应的类标号, 定义每个指标集合 $C_k = \{i | y_i, i = 1, 2, \dots, K\}$ 和最接近 $\left\lceil \frac{|C_k|}{S} \right\rceil$ 的整数 N_k , N_k 表示第 k 类的样本数量, 首先令 $k = 1$ 。

步骤 2 对每个 C_k , 将 C_k 划分为 S 个子集 $C_{k,j}$, $j = 1, 2, \dots, S$, 令 $j = 1$ 。

步骤 3 随机选择 $i \in C_k$ 并根据欧拉距离找到 x_i 的最近邻, 其中 $\mathcal{N}(x_i)$ 的大小应为 $N_k - 1$, 此时令 $C_{k,j} = \{i\} \cup \mathcal{N}(x_i)$, 并且令 $C_k = C_k \setminus C_{k,j}$, “ \setminus ”操作从集合 C_k 除去集合 $C_{k,j}$ 的元素。

步骤 4 如果 $j < S$, 令 $j = j + 1$, 则跳到步骤 3; 否

则结束关于 j 的循环。

步骤 5 如果 $k < K$, 令 $k = k + 1$, 则跳到步骤 2; 否则结束关于 k 的循环并输出结果。

该算法的潜在机制就是不重叠地将各个类划分为一系列的子类。接下来将通过 LLE 算法将每个子类投射到各自的子空间。

3.4 局部线性嵌入算法

在 GNAM 算法中本文使用局部线性嵌入算法得到各子类数据的低维流形。与其他嵌入算法如 Isomap 算法和 SNE 算法相比, LLE 算法的优点是对一个稀疏矩阵而言, 它能有一个唯一的解析解^[2]。图 3 展示了使用覆盖算法和 LLE 算法的整个流程。

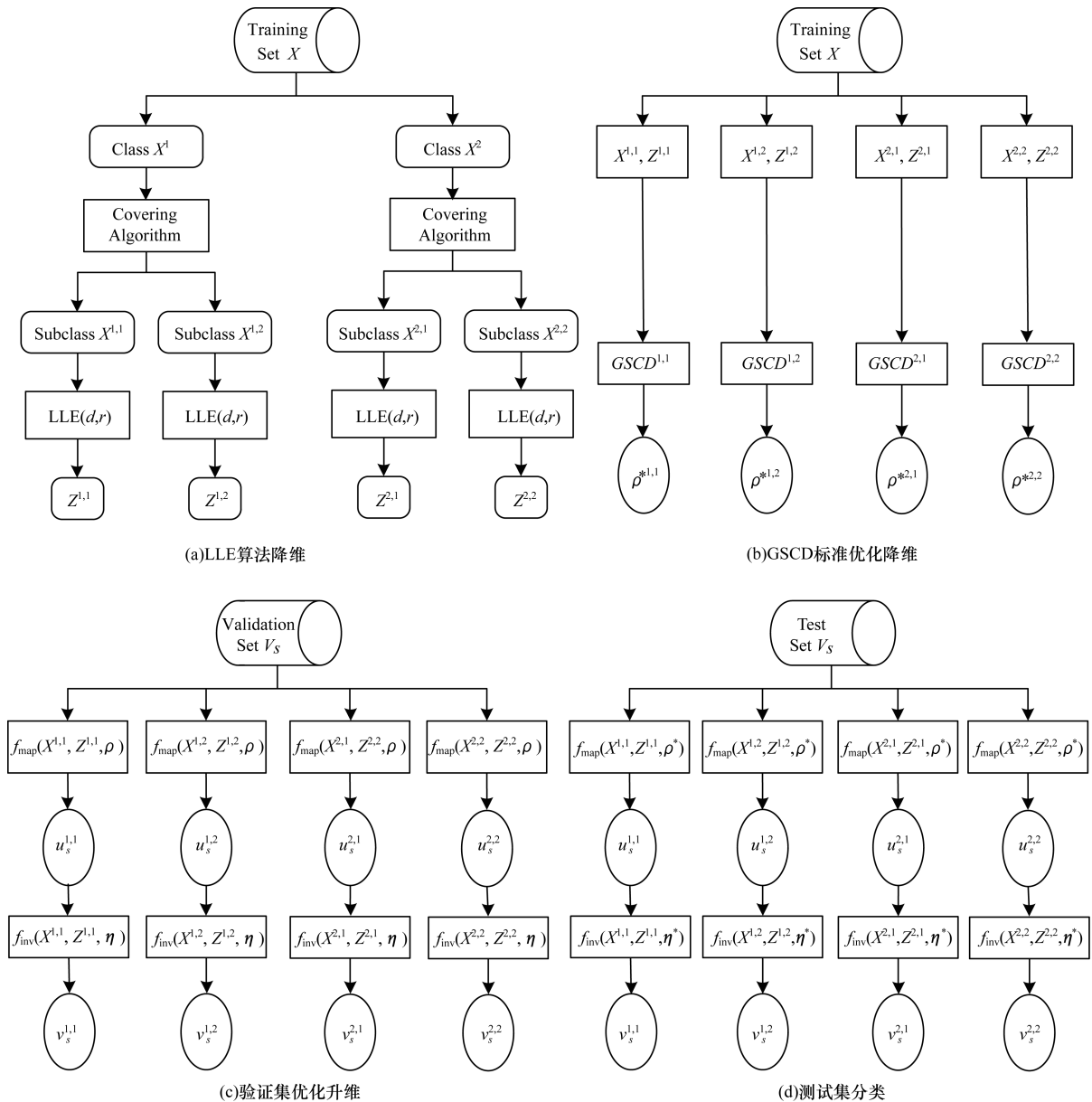


图 3 GNAM 算法流程

3.5 径向基函数自联想网络

虽然通过上述的覆盖算法和 LLE 算法可以将各个子类投影到了各自的低维流形中,但缺少一个通用的标准来衡量数据集外部数据的类别,文献[2]提到,LLE 算法主要用于数据的可视化和挖掘固有空间,其对数据集外部样本的预测准确率并不理想。

考虑到 LLE 算法的缺陷,本文利用了一个自联想形式的径向基函数网络来映射外部样本到低维子空间再反向映射回高维原空间得到重构的样本,以此来解决判定标准的问题。

3.5.1 初始化

降维的 RBF 函数有以下形式:

$$f_{\text{map}}(\mathbf{x}) = \sum_{i=1}^N \mathbf{p}_i k(\mathbf{x}, \mathbf{x}_i; \rho) \quad (6)$$

同样,升维 RBF 函数的形式如下:

$$f_{\text{inv}}(\mathbf{z}) = \sum_{i=1}^N \mathbf{q}_i k(\mathbf{z}, \mathbf{z}_i; \eta) \quad (7)$$

其中, ρ 和 η 是 RBF 函数的参数,本算法中为每个子类别分别确定 ρ , 而 η 则统一设定, $\mathbf{p}_i \in \mathbb{R}^d$, $\mathbf{q}_i \in \mathbb{R}^n$, $i=1, 2, \dots, N$ 是 2 个 RBF 模型的权值,接下来首先介绍这 2 个权值的算法。

3.5.2 权值形式

首先定义以下矩阵:

$$\mathbf{K}_{\text{map}} = \begin{pmatrix} k(x_1, x_1; \rho) & k(x_1, x_2; \rho) & \cdots & k(x_1, x_N; \rho) \\ k(x_2, x_1; \rho) & k(x_2, x_2; \rho) & \cdots & k(x_2, x_N; \rho) \\ \vdots & \vdots & & \vdots \\ k(x_N, x_1; \rho) & k(x_N, x_2; \rho) & \cdots & k(x_N, x_N; \rho) \end{pmatrix} \quad (8)$$

得到:

$$\mathbf{p}_i = \mathbf{K}_{\text{map}}^\dagger \mathbf{z}_i \quad (9)$$

同理:

$$\mathbf{K}_{\text{inv}} = \begin{pmatrix} k(z_1, z_1; \eta) & k(z_1, z_2; \eta) & \cdots & k(z_1, z_N; \eta) \\ k(z_2, z_1; \eta) & k(z_2, z_2; \eta) & \cdots & k(z_2, z_N; \eta) \\ \vdots & \vdots & & \vdots \\ k(z_N, z_1; \eta) & k(z_N, z_2; \eta) & \cdots & k(z_N, z_N; \eta) \end{pmatrix} \quad (10)$$

$$\mathbf{q}_i = \mathbf{K}_{\text{inv}}^\dagger \mathbf{x}_i \quad (11)$$

接下来需要确定 RBF 的参数 ρ 和 η 。

3.5.3 GSCD 标准求解最优参数 ρ 的方法

前文提到,在计算非线性嵌入技术的嵌入质量时,文献[11]利用 RBF 去逼近了嵌入算法的映射函数,最终算得嵌入算法的 GSCD 分数(如 LLE 算法),当时 RBF 参数是通过交叉验证(cross-validation)得到的。由于 GSCD 能很好地评价嵌入算法的得到的低维流形的总体光滑度和联合方向一致性,在几何角度有直观的解释性,不妨利用 GSCD

标准来调节 GNAM 分类器中低维映射的 RBF 函数的参数。

假定当前需要学习的参数 ρ 所对应的 RBF 函数正是在求解 GSCD 分数时用来逼近非线性嵌入算法的 RBF 函数,此时如果对参数 ρ 优化 GSCD 评分使其最低,这样参数 ρ 对应的 RBF 函数应该和它所逼近的非线性嵌入算法有一样好的总体光滑度和联合方向一致性,所以,这样寻找最优参数 ρ 的方法是有直观几何意义的。

需要注意的是,参数 ρ 过大时,高斯函数 $k(\mathbf{x}, \mathbf{c}; \rho)$ 会趋向于 1,过小时则会趋向于 0。过大或过小的 ρ 会使 GSCD 标准的评分没有意义,因此,必须给 GSCD 分数加上一个约束项,使得参数过大或过小时 GSCD 的没有意义导致训练失败。

本文使用如下准则训练参数 ρ , 在区间 $[\text{minValue}, \text{maxValue}]$ 中以 $\delta\rho$ 为步长选取 ρ 使得:

$$\rho^* = \arg \min_{\substack{\rho = \text{minValue} + k\delta\rho \\ \rho \leq \text{maxValue}, k \in N}} \text{GSCD}(\rho) + \lambda \frac{1}{\text{Variance}(\{1 - k(x_i, x_j; \rho) \mid \forall i, j = 1, 2, \dots, N\})} \quad (12)$$

其中, λ 是 GSCD 和约束项之间的平衡参数,本文设定为 0.01。

利用 GSCD 标准训练分类器是本文的一个创新点,既挖掘了 GSCD 标准在分类方面的新功能,又加强了模型的几何直观性和可解释性。

3.5.4 验证集最低误差确定最优参数 η 的方法

令验证集数据 v_i 的低维表示为 u_i , 可以由映射函数得到原数据的重构 v'_i :

$$v'_i = f_{\text{inv}}(u_i) \quad (13)$$

因为每个子类对应的降维模型能生成一个关于 v_i 的低维表示 u_i , 所以对每一个子类,通过径向基函数自联想网络可以分别得到 v_i 重构,记作 $v_i^{k,j}$, 上标 k 和 j 表示 $v_i^{k,j}$ 经过第 j 类的第 k 个子类的自联想网络得到的重构值。

由下式可以得到验证集数据 v_i 类别标号 $\mathcal{L}(v_i)$:

$$\mathcal{L}(v_i) = \arg \min_k \min_j \|v_i - v_i^{k,j}\| \quad (14)$$

令 $E(\eta)$ 表示验证集的错误率,这个函数与参数 η 的取值有关,在区间 $[\text{minValue}, \text{maxValue}]$ 中以 $\delta\eta$ 为步长选取 η 使得 $E(\eta)$ 最小,这样参数 η 就可以确定:

$$\eta^* = \arg \min_{\substack{\eta = \text{minValue} + k\delta\eta \\ \eta \leq \text{maxValue}, k \in N}} E(\eta) \quad (15)$$

3.6 分类

在给定了无类标记的测试样本之后,可以通过以下的分类标准进行分类:

$$\mathcal{L}(t_i) = \operatorname{argmin}_k \min_j \|t_i - g_i^{k,j}(t_i)\| \quad (16)$$

其中, t_i 表示为标记数据; $g_i^{k,j}(\cdot) = f_{\text{inv}}^{k,j}(f_{\text{map}}^{k,j}(\cdot))$, 上式表示 t_i 在经过分类器的映射后得到重建值 $g_i^{k,j}(t_i)$; 与 t_i 欧拉距离最小的 $g_i^{k,j}(t_i)$ 表示 t_i 最有可能属于这个类, 标记 $\mathcal{L}(t_i)$ 为此类。假定通过自身类的 GNAM 分类器时, 得到的重建值误差最小。

对比文献[13]的自联想网络, 本文提出的 GNAM 分类器有以下优点:

1) 本文算法利用覆盖算法将每类数据聚类为多个子类, 并为每个子类分别训练自联想神经网络分类器。解决高维非线性数据在低维子空间的重叠问题, 且在每类数据存在多子类情况下依然适用。

2) GNAM 使用有唯一解析解的 LLE 算法, 部分解决了收敛性问题, 保证了算法复杂度和稳定性。

3) GSCD 标准能很好的评价嵌入算法的得到的低维流形的总体光滑度和联合方向一致性, 在几何角度有直观的解释性, 易于计算, 而且能一定程度上处理数据集外部样本。本文算法利用 GSCD 标准调节神经网络参数, 几何上有直观解释, 保证数据在低维流形上的总体光滑度和联合方向一致性, 对数据集外部样本有更好的处理性能。

3.7 GNAM 的可扩展性

通过改变优化参数的方法, 可以使 GNAM 算法具有更好的可扩展性。具体来说, 如果有新的类加入, 模型不再需要重新训练。原始的 GNAM 算法使用错误率来优化模型的自联想网络的参数, 现在改用 F-measure 来优化, F-measure 定义如下:

$$F = 2 \times (\text{precision} \times \text{recall}) / (\text{precision} + \text{recall}) \quad (17)$$

$$\text{precision} = \frac{tp}{tp + fp} \quad (18)$$

$$\text{recall} = \frac{tp}{tp + fn} \quad (19)$$

其中, tp , fp 和 fn 分别是真阳性, 假阳性和假阴性。另外, 将式(15)中的最小化 $E(\cdot)$ 替换为最大化 $F(\cdot)$ 。通过寻找最大的 F 值, 可以找到最优的参数是得模型的分类错误降低并且确定每个子类的空间维度。GNAM 的可扩展算法在分类任务加入了新类时依然有效, 将其称之为 Scala-GNAM 算法。

3.8 En-GNAM 分类器

为了更好地提升 GNAM 分类器的预测准确率, 本文使用了集成学习技术去集成拥有不同子类数的 GNAM 分类器来决定最终分类结果。En-GNAM 分类器解决了 GNAM 的一个潜在问题, 那就是在预测前 GNAM 分类器必须预先设定子类数量, 但子类数量是个未知数, 显然, 选择一个错误的子类数量会降低分类器的准确率。

采用以下方法构建 En-GNAM 分类器:

1) 选择多个子类数量 S , 通过覆盖算法分别将训练数据分成若干份子类, 这样增加了模型的多样性

2) 随着 S 改变, GNAMs 的结构也随之受到影响。例如如果一共有 26 个类, 当 $S=1$ 时, GNAM 的数量为 26, 当 $S=6$ 时, GNAMs 的数量增加到 $26 \times 6 = 156$ 。

每个对应某个 SS 的 GNAM 能够对测试数据作出一个分类预测, 将这些 GNAM 集合到一起, 通过多数表决法确定测试数据的最终分类结果。

4 实验结果与分析

在本节中, 将比较 GNAM 算法和 En-GNAM 算法和其他 4 个算法的性能。这 4 个算法分别是 k-NN 算法和其集成版本 En-KNN, 多项式核和径向基函数核的支持向量机(SVM)算法。比较基于 UCI 数据库的 5 个数据集^[14], 分别是 UCI letter^[15], OCR, OPTDigits, PENDigits 以及 Image Segmentation。

4.1 数据集和参数选择

UCI letter 数据集有 20 000 个标记样本, 26 类, 其中每类约 770 个样本。图 4 是一些样本的例子。OCR 数据集包括了 16 280 个手写字母。平均每类约 600 个样本, 每个样本的维度是 30。第 3 个数据集(OPTDigits)用于手写数字识别, 其中, 有来自 43 个人的手写数字, 30 个人的作为训练集而剩余 13 个人的用来作为测试集。PENDigits 数据集集中的手写数字来自不同的笔, 输入向量的维度统一且输入数值范围是 $[0, 100]$ 。整个数据集收集了 44 个人的手写数字, 一共 10 992 个样本。文献[14]更详细地介绍了 UCI 数据库的所有数据集。随机地对划分每个数据集为三部分, 训练集, 验证集和测试集, 具体划分细节参考表 1。



图 4 UCI letter 数据集样例

表 1 实验数据集和数据划分

数据集	样本数量	原始维度	类别数	训练集	验证集	测试集	约简维度
UCI	20 000	16	26	180 × 26	180 × 26	14 020	10
OCR	16 280	30	26	180 × 26	180 × 26	10 300	16
OPTDigits	5 620	64	10	180 × 10	180 × 10	1 797	20
PENDigits	10 992	16	10	360 × 10	360 × 10	3 498	14
ImaSeg	2 100	18	7	120 × 7	120 × 7	1 106	14

训练集使用覆盖算法和 LLE 算法来构造多个低维子空间,验证集优化 GNAM 的参数,测试集则用来评估 GNAM 和 En-GNAM 算法的分类性能。子类范围设置为 $[1,6]$,步长为 1。

根据文献[4],覆盖个数 r 被设置为 50 (当某些子类的样本个数少于 50 时,设置 r 小于子类样本个数的数)。不同的数据库设定了不同的约简维度,具体见表 1。考虑到计算复杂度和参数有效性,参数 ρ 的范围设为 $[10^1, 10^5]$,步长为 $10^{i \times 0.2}$ ($i = 1, 2, \dots$),而参数 η 的范围设为 $[10^{-10}, 10^{10}]$,步长为 $10^{i \times 0.5}$ ($i = 1, 2, \dots$)。在进行分类之前将每个数据集都进行了标准化。k-NN 算法的最优 k 值根据验证集的准

确率间隔选取得到。另外,En-KNN 算法是集成版本的 k-NN 算法 (其中, k 取 $1 \sim 30$),同样使用多数表决法,这样 En-KNN 算法能和 En-GNAM 算法进行比较。多项式核的支持向量机算法的参数选择范围是 $[0, 10]$,步长为 0.5。径向基函数核支持向量机方法的参数选择范围是 $[10^{-10}, 10^{10}]$,步长 $10^{i \times 0.5}$ ($i = 1, 2, \dots$)。2 种支持向量机方法的参数都靠优化验证集准确率来确定。最后提交的错误率均是在最优参数下的错误率。实验在相同大小的训练集,验证集和测试集下重复 10 次,每次随机选取样本。表 2 展示了 5 个数据集的平均错误率,其中,粗体是最优结果。

表 2 平均错误率比较结果 1

%

数据集	k-NN	SVM-POLY	SVM-RBF	En-KNN	GNAM	En-GNAM
UCI	9.65 ± 0.27	8.13 ± 0.26	7.91 ± 0.30	15.13 ± 0.46	8.61 ± 1.02	7.56 ± 0.92
OCR	11.60 ± 0.21	11.03 ± 0.30	9.25 ± 0.29	14.64 ± 0.23	11.55 ± 0.74	9.27 ± 0.37
OPTDigits	4.08 ± 0.38	2.80 ± 0.25	9.09 ± 0.8	5.23 ± 0.49	3.57 ± 1.48	3.25 ± 0.57
PENDigits	2.86 ± 0.18	2.60 ± 0.37	2.31 ± 0.30	4.80 ± 0.39	2.54 ± 0.43	2.48 ± 0.08
ImaSeg	6.86 ± 0.65	6.74 ± 0.92	6.12 ± 0.90	10.37 ± 0.97	6.56 ± 1.58	5.71 ± 1.16

4.2 分类结果分析

通过观察表 2 可以发现,在每个数据集中,GNAM 和 En-GNAM 算法的错误率均小于 k-NN 和 En-KNN 算法,并且与 2 种支持向量机算法不相上下。其中在 OCR 数据集、OPTDigits 数据集和 PENDigits 上 En-GNAM 都仅以微弱优势落后支持向量机方法,排名第 2,在剩下的 2 个数据集中排名第 1。观察发现集成版本的 En-GNAM 在所有数据集中错误率均低于 GNAM 最好表现 (选取了错误率最低的子类个数),这表明了 GNAM 算法需要关于子类个数的先验知识才能得到较好的准确率,这个缺陷被使用集成学习的 En-GNAM 算法很好地克服了,因为它能自动选择不同的

子类个数并得到集成的结果。

另外,本文还在 OPTDigits 数据集和 PENDigits 数据集上做了关于训练样本大小对错误率影响的实验。除了每个类的训练样本数设置不同外,其他划分参数均与表 1 中相同。GNAM 和 En-GNAM 算法的分类结果显示在图 5 中。上方的每条曲线代表了 S 不同取值的 GNAM 分类器的错误率,下方的柱状图则是 En-GNAM 分类器的错误率和标准差。观察发现,随着训练样本数的增加,错误率明显降低。另外,在不同训练样本数的情况下,En-GNAM 分类器的错误率始终低于 GNAM 分类器,这证明了集成版本的 En-GNAM 分类器比 GNAM 分类器效果更好。

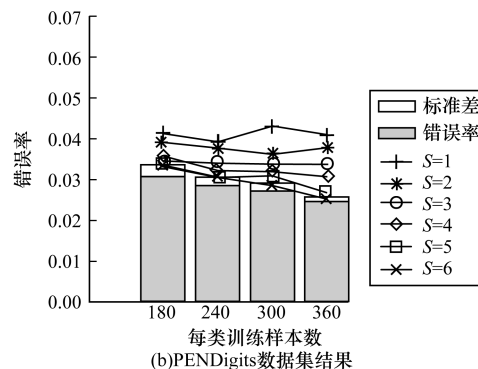
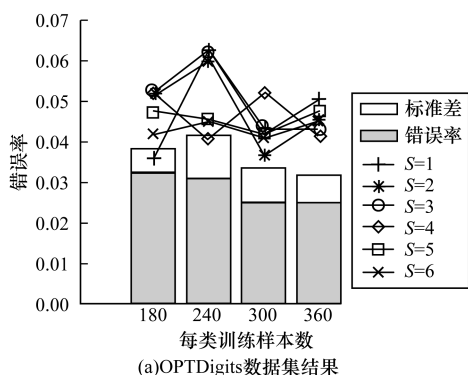


图 5 样本数量实验结果

4.3 GSCD 有效性实验

为了检验利用 GSCD 标准调整参数的算法是否有效,本文将不使用 GSCD 标准随机生成参数的算法与使用 GSCD 标准调整参数的算法 (NAM, En-

NAM) 的分类结果进行了对比,表 3 展示了对比结果。

从结果中可以发现,在每个数据集中,没有使用 GSCD 标准的 NAM 和 En-NAM 算法的平均错

误率都分别比 GNAM 和 En-GNAM 算法高,并且使用 GSCD 标准后错误率降低百分比最少达到了 17.45%,这说明本文使用 GSCD 标准调节参数

后降低了随机选取参数的 NAM 和 En-NAM 算法的错误率,引入 GSCD 标准对提高分类正确率具有重要作用。

表 3 平均错误率比较结果 2

%

数据库	NAM	GNAM	GNAM 相对于 HAM 的下降比例	En-NAM	En-GNAM	En-GNAM 相对于 En-NAM 的下降比例
UCI	15.74	8.61	45.30	10.20	7.56	25.88
OCR	15.83	11.55	27.04	11.23	9.27	17.45
OPTDigits	5.18	3.57	31.08	4.06	3.20	21.18
PENDigits	7.95	2.54	68.05	5.33	2.40	54.97
ImaSeg	11.48	6.56	42.86	7.20	5.71	20.69

4.4 可扩展性实验

为了验证本文提到的 Scala-GNAM 方法的可扩展性,考虑在 UCI Letter 数据集和 Image Segmentation 数据集上进行实验。首先以 F 的最大值为优化目标调节 GNAM 分类器 (Scala-GNAM-F), 计算在不同类别数量的情况下其分类错误率。然后作为比较, 同样地不以 F 的最大值为优化目标调节 GNAM 分

类器 (Scala-GNAM-NF), 计算不同类别数量的情况下其分类错误率。图 6 展示了重复运行 10 次的平均结果。

实验结果表明,随着类别的数量增加,错误率单调递增。而且,Scala-GNAM-F 的错误率比 Scala-GNAM-NF 低。这表明使用不同的评判标准,对 GNAM 的可扩展性有所影响。

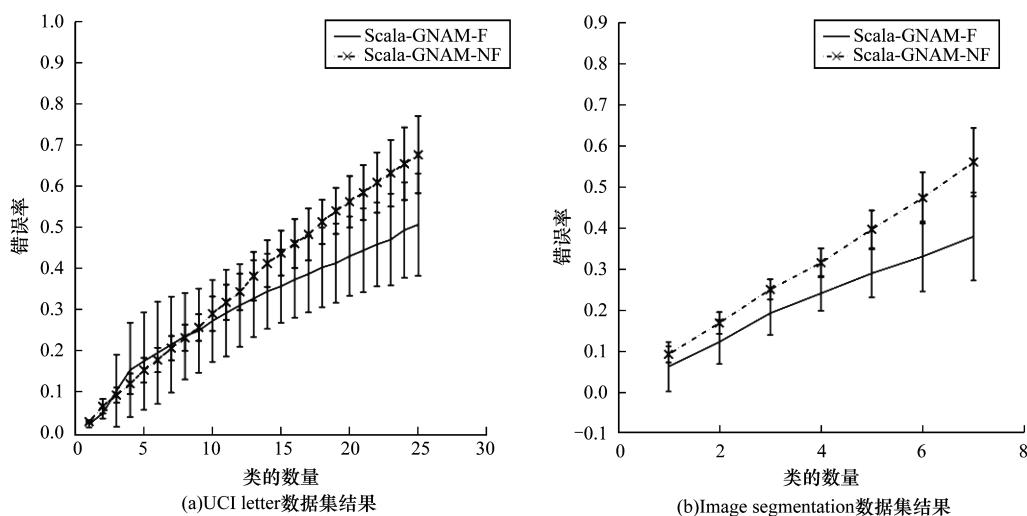


图 6 可扩展性实验结果

5 结束语

本文提出一种基于非线性嵌入的自联想神经网络分类器处理复杂的高维数据分类任务,并引入 GSCD 标准训练网络参数,提升了模型的几何直观性和可解释性,有效解决数据在同一低维流形空间中重叠的问题。在 UCI 数据集上的实验结果表明,本文提出的算法相比于对比算法有效提高分类准确率,并且验证了 GSCD 标准训练有效性和算法的可扩展性。

参考文献

[1] Seung H S, Daniel D L. The Manifold Ways of Perception[J]. Science, 2000, 290(5500): 2268-2269.

[2] Roweis S, Saul L. Nonlinear Dimensionality Reduction by Locally Linear Embedding[J]. Science, 2000, 290(5500): 2323-2326.

[3] de la Torre F, Kanade T. Multimodal Oriented discriminant Analysis[C]//Proceedings of the 22nd International Conference on Machine Learning. New York, USA: ACM Press, 2005: 177-184.

[4] Zhang Junping, Stan Z L, Wang Jue. Manifold Learning and Applications in Recognition[M]//Tan Y P. Intelligent Multimedia Processing with Soft Computing. Berlin, Germany: Springer, 2005: 281-300.

[5] Ito M, Ohyama W, Wakabayashi T, et al. Rotated Face Recognition by Manifold Learning with Auto-associative Neural Network[C]//Proceedings of Frontiers of Computer Vision. Washington D. C., USA: IEEE Press, 2015: 1-4.

(下转第 216 页)

5 结束语

本文针对传统隐马尔可夫模型在解决词性标注问题上的不足,提出改进的二阶隐马尔可夫模型。该模型能更多地联系上下文,使得在中医诊断文本的标注更为精确。在二阶 HMM 模型参数进行训练的过程中会出现数组下溢的问题,本文引入比例因子加以修正。同时,还给出未登录词的解决方案。在相同的训练集和测试集下,二阶 HMM 模型在中医诊断古文的词性标注准确率明显提高。由于构建的中医诊断语料库规模有限,因此词性标记集和标注模型有待进一步研究。

参考文献

- [1] 王 敏. 基于改进的隐马尔可夫模型汉语词性标注[D]. 太原:山西大学,2007.
- [2] 王凤娥,谭红叶. 基于最大熵的句内时间关系识别[J]. 计算机工程,2012,38(4):241-243.
- [3] 古丽拉·阿东别克,侯呈凤,古丽拉·阿东别克. 改进的 HMM 应用于哈萨克语词性标注[J]. 计算机工程与应用,2010,46(36):147-149.
- [4] 袁里驰. 基于改进的隐马尔可夫模型的词性标注方法[J]. 中南大学学报(自然科学版),2012,43(8):3053-3057.
- [5] 姜尚仆,陈群秀. 基于规则和统计的日语分词和词性标注的研究[J]. 中文信息学报,2010,24(1):117-122.
- [6] Lafferty J, McCallum A, Pereira F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data [C]//Proceedings of ICML '01. San Francisco, USA: Morgan Kaufmann, 2001:282-289.
- [7] McCallum A, Li W. Early Results for Named Entity Recognition with Conditional Randomfields, Feature Induction and Web-enhanced Lexicons [C]//Proceedings of CoNLL'03. Edmonton, Canada: Morgan Kaufmann, 2003:188-191.
- [8] Rabiner L E. A tutorial on Hidden Markov Models and Selected Application in Speech Recognition [J]. Proceedings of the IEEE, 1989, 77(2):257-286.
- [9] 王国龙,杜建强,郝竹林,等. 中医诊断古文的词性标注与特征重组[J]. 计算机工程与设计,2015,36(3):835-841.
- [10] 韩 普,姜 杰. HMM 在自然语言处理领域中的应用研究[J]. 计算机技术与发展,2010,20(2):245-248.
- [11] 史笑兴,王太君,何振亚. 二阶隐马尔可夫模型的学习算法及其与一阶隐马尔可夫模型的关系[J]. 应用科学学报,2001,19(1):29-32.
- [12] 唐亚平,姜瑞雪,樊欣荣. 证素及证素辨证的研究状况[J]. 时珍国医药,2008,19(10):27-29.
- [13] 周顺先,林亚平,王耀南,等. 基于二阶隐马尔可夫模型的文本信息抽取[J]. 电子学报,2007,35(11):2226-2231.
- [14] 杜世平,陈 涛. 与观测信息相关的二阶隐马尔可夫模型的参数估计[J]. 西南师范大学学报(自然科学版),2006,31(3):24-27.
- [15] 方 浩,许鸿文,蔡益宇. 一种基于语义关系改进的隐马尔可夫模型研究[J]. 通信技术,2008,41(5):157-159.
- [16] 张孝飞,陈肇雄,黄河燕. 词性标注中生词处理算法研究[J]. 中文信息学报,2003,17(5):157-159.

编辑 索书志

(上接第 210 页)

- [6] 王 朋,陈树中. 基于混合模型 HMM/RBF 的数字语音识别[J]. 计算机工程,2002,28(12):136-138.
- [7] Li Ma, Crawford M M, Yang Xiaoquan, et al. Local-manifold-learning-based Graph Construction for Semisupervised Hyperspectral Image Classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2015, 53(5):2832-2844.
- [8] Alain G, Bengio Y. What Regularized Auto-encoders Learn from the Datagenerating Distribution [J]. The Journal of Machine Learning Research, 2014, 15(1):3563-3593.
- [9] Gisbrecht A, Hammer B. Data Visualization by Nonlinear Dimensionality Reduction [J]. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2015, 5(2):51-73.
- [10] 何 力,张军平,周志华. 基于放大因子和主延伸方向研究流形学习算法[J]. 计算机学报,2005,28(12):2000-2009.
- [11] Zhang Junping, Wang Qi, Zhou Zhihua. Quantitative Analysis of Nonlinear Embedding [J]. IEEE Transactions on Neural Networks, 2011, 22(12):1987-1998.
- [12] Jain A, Murty M, Flynn P. Data Clustering: A Review [J]. ACM Computing Surveys, 1999, 31(3):264-323.
- [13] Borlard H, Kamp Y. Auto-association by Multilayer Perceptrons and Singular Value Decomposition [J]. Biological Cybernetics, 1988, 59:291-294.
- [14] Blake C L, Merz C J. UCI Repository of Machine Learning Databases [EB/OL]. (2010-11-21). <http://www.ics.uci.edu/mllearn/MLRepository.html>.
- [15] Frey P W, Slate D J. Letter Recognition Using Hollandstyle Adaptive Classifiers [J]. Machine Learning, 1991, 6(2):161-182.

编辑 刘 冰 陆燕菲