

DOI: 10.11992/tis.201612012

网络出版地址: <http://kns.cnki.net/kcms/detail/23.1538.TP.20170508.0922.002.html>

# 基于测度学习支持向量机的钢琴乐谱难度等级识别

郭龙伟, 关欣, 李铨

(天津大学 电子信息工程学院, 天津 300072)

**摘要:** 现有钢琴乐谱难度分类主要由人工方式完成, 效率不高, 而自动识别乐谱难度等级的算法对类别的拟合度较低。因此, 与传统将乐谱难度等级识别归结为回归问题不同, 本文直接将其建模为基于支持向量机的分类问题。并结合钢琴乐谱分类主观性强、特征之间普遍存在相关性等特点, 利用测度学习理论有难度等级标签乐谱的先验知识, 依据特征对难度区分的贡献度, 改进高斯径向基核函数, 从而提出一种测度学习支持向量机分类算法——ML-SVM 算法。在 9 类和 4 类难度两个乐谱数据集上, 我们将 ML-SVM 算法与逻辑回归, 基于线性核函数、多项式核函数、高斯径向基核函数的支持向量机算法以及结合主成分分析的各个支持向量机算法进行了对比, 实验结果表明我们提出算法的识别正确率优于现有算法, 分别为 68.74% 和 84.67%。所提算法有效提高了基于高斯径向基核函数支持向量机算法在本应用问题中的分类性能。

**关键词:** 数字钢琴乐谱; 难度等级识别; 分类算法; 支持向量机; 测度学习; 高斯径向基核函数

**中图分类号:** TP391.4    **文献标志码:** A    **文章编号:** 1673-4785(2018)02-0196-06

中文引用格式: 郭龙伟, 关欣, 李铨. 基于测度学习支持向量机的钢琴乐谱难度等级识别[J]. 智能系统学报, 2018, 13(2): 196-201.

英文引用格式: GUO Longwei, GUAN Xin, LI Qiang. Recognition of difficulty level of piano score based on metric learning support vector machine[J]. CAAI transactions on intelligent systems, 2018, 13(2): 196-201.

## Recognition of difficulty level of piano score based on metric learning support vector machine

GUO Longwei, GUAN Xin, LI Qiang

(Department of Electronic Information Engineering, Tianjin University, Tianjin 300072, China)

**Abstract:** The existing classification work about piano score's level is mainly done manually and inefficient, while the algorithm automatically recognizing the difficulty class of music score has a low classification fitting degree. Therefore, different from the traditional method that takes the recognition for the difficulty class of music score as a regression issue, the paper directly modelled it as a classification based on the support vector machine, in addition, in combination with such characteristics of the score classification as intense subjectivity and common dependency among features, the metric learning theory was utilized. The prior knowledge of the score with difficult level tag was sufficiently utilized, according to the contribution of feature in difficulty distinguishment, the Gauss radial basis kernel function was improved, so as to propose a kind of metric learning support vector machine classification algorithm —ML-SVM algorithm. In the score datasets with level 9 and level 4 difficulty, ML-SVM algorithm was compared with logistic regression, the support vector machine algorithm based on linear kernel function, polynomial kernel function, Gauss radical basis (GRB) kernel function, and various support vector machine algorithms combining principal component analysis. The results show that the proposed algorithm is much more accurate than the existing algorithms, reaching the accuracy rate 68.74% and 84.67% respectively. The proposed algorithm effectively improves the classification performance of SVM algorithm based on GRB kernel function in this application.

**Keywords:** digital piano score; recognition of difficulty level; classification algorithm; support vector machine; metric learning; Gauss radial basis kernel function

收稿日期: 2016-12-08. 网络出版日期: 2017-05-08.

基金项目: 国家自然科学基金项目 (60802049, 61471263); 天津市自然科学基金重点项目 (16JCZDJ31100).

通信作者: 关欣. E-mail: [guanxin@tju.edu.cn](mailto:guanxin@tju.edu.cn).

虽然互联网中存在海量的钢琴乐谱资源, 但其难易程度不一, 对于业余和初级乐器学习者来说, 由于缺乏专业知识和指导, 难以有效地找到与自身

学习难度匹配的乐谱来学习。专业音乐学习者一般依据固定进阶教材,但缺乏个性化、灵活的学习方案。此外,一些经典但难度高的乐谱,会有诸多简化版本存在。因此,有必要对海量的钢琴乐谱资源的难度等级进行区分。

现在绝大部分乐谱难度等级仍然依赖于专业人士主观判断。对于海量的数字乐谱,人工判断难度等级会是一个耗时耗力的巨大工程。人工主观判断也很难稳定、可靠地把握每个难度等级之间的区别,尤其对于多类别问题。不同的人对于同一首乐谱可能会给出不同的难度等级,甚至对于同一首乐谱,同一个人不同的时间也会给出不同的难度等级。因此,自动识别乐谱难度等级的系统具有重要的理论意义和应用价值。

为网络中存在的数字乐谱提供难度等级标签将会大大提高寻找合适难度等级乐谱的效率,并能提高音乐网站的用户体验。通常研究中将乐谱难度识别归为模式分类问题,从符号乐谱中定义并提取难度相关特征,利用分类思想实现乐谱难度等级的识别。下面将针对不同的技术来回顾钢琴乐谱难度等级识别领域的研究现状。

Shih-Chuan Chiu 等<sup>[1]</sup>最先开始钢琴乐谱难度等级识别研究。由于钢琴乐谱难度识别是一个相对较新的研究问题,现有的符号音乐特征(symbolic music feature)较少直接用于难度等级识别。所以他们首先定义8个与乐谱难度密切相关的特征,将此8个难度相关特征与语义特征一起作为特征空间。后用特征选择算法 ReliefF<sup>[2]</sup>按照特征重要程度(即各个特征对难度等级的辨别能力大小)分配权值,选择权值最大的10个特征作为后续实现难度等级识别的特征空间。最终用3个回归算法:多元线性回归、逐步回归<sup>[3]</sup>、支持向量回归<sup>[4]</sup>实现难度等级识别。实验中,支持向量回归算法得到最好的效果,其 $R^2$ 统计量为39.9%<sup>[3]</sup>。

Véronique Sébastien 等<sup>[5]</sup>也将乐谱难度等级识别看作分类问题,他们利用无监督的聚类算法实现乐谱难度等级识别。首先定义7个难度相关特征,这些特征从 MusicXML 格式<sup>[6]</sup>乐谱文件提取。之后用主成分分析(principal component analysis, PCA)将特征投影到低维空间,以降低特征维数。然后,用分层聚类(hierarchical clustering)<sup>[7]</sup>将乐谱聚成3类,即3个难度类别。

总的来说,无论是多元线性回归还是逐步回归都假设特征与难度等级之间为线性关系,此假设过于简化特征与难度等级之间的实际关系。而支持向量回归虽然可实现非线性拟合,但拟合结果尚不令

人满意。与有监督算法相比,非监督算法,如分层聚类,虽能充分利用特征与难度等级之间的自然分布关系,但无法利用已有的难度等级标签作为先验知识提高识别效果。例如,实验中,原始乐谱数据是4个类别,Véronique Sébastien 经过 PCA 降维,应用聚类算法,最终仅得到3个难度类别。

在现有有监督的回归拟合算法识别钢琴乐谱难度等级的研究中,支持向量回归有最好的拟合效果(其 $R^2$ 统计量值为39.9%)。但支持向量回归,更适合逼近、拟合连续分布数据,对于此离散类别的分类问题能力有限。所以本文利用一种与支持向量回归算法原理相近,但更适用于解决分类问题的支持向量机分类算法实现数字钢琴乐谱难度等级识别。

## 1 支持向量机理论

SVM 是基于统计学习理论中经验风险最小化原则的一种机器学习算法<sup>[8]</sup>。SVM 已广泛应用于纹理分类(texture classification)<sup>[9]</sup>、文本分类<sup>[10]</sup>、人脸识别<sup>[11]</sup>、语音识别<sup>[12]</sup>等各个领域。理论和实践证明,SVM 对噪声和离群点鲁棒性好,泛化能力强,经过扩展可解决多分类问题<sup>[8]</sup>。

SVM 实现分类的关键是核函数,利用核函数<sup>[13]</sup>可以将低维线性不可分的问题转化到高维空间实现线性可分,同时避免因维数增加而导致过大的计算量。常用的核函数有线性核函数(linear kernel function)、多项式核函数(polynomial kernel function)、高斯径向基核函数(Gauss radical basis kernel function, GRB)等<sup>[13]</sup>。由于本研究问题的特征数目远小于样本数目,并且为降低分类模型的参数复杂度,本文考虑采用高斯径向基核函数。

然而传统的基于高斯径向基核函数的 SVM (GRB-SVM) 算法假设特征不相关且权重相同,即不能根据特征对难度等级的贡献度差别对待。而在本应用问题中,特征对难度区分的贡献度不同。为弥补 GRB-SVM 不能根据特征重要程度差别对待的不足,本文结合测度学习(metric learning)理论,充分利用训练乐谱中关于难度等级的先验知识,从带难度等级标签的乐谱特征空间中有监督地学习到能更好区分钢琴难度的投影矩阵,保留特征之间的相关关系,从而利用该矩阵改进高斯径向基核函数,提出一种测度学习支持向量机分类算法——ML-SVM 算法,实现数字钢琴乐谱难度等级识别。

## 2 算法原理与算法描述

本文提出的 ML-SVM 乐谱难度识别算法主要包括以下3部分:基于测度学习从训练数据中有监

督的学习到能增加钢琴乐谱难度区分度的投影矩阵;得到新的距离测度,并利用该测度改进高斯径向基核函数,建立 ML-SVM 算法模型;最后利用网格搜索算法得到核函数参数的最优组合,建立分类模型,实现数字钢琴乐谱难度等级的识别。ML-SVM 乐谱难度识别算法框图,如图 1 所示。

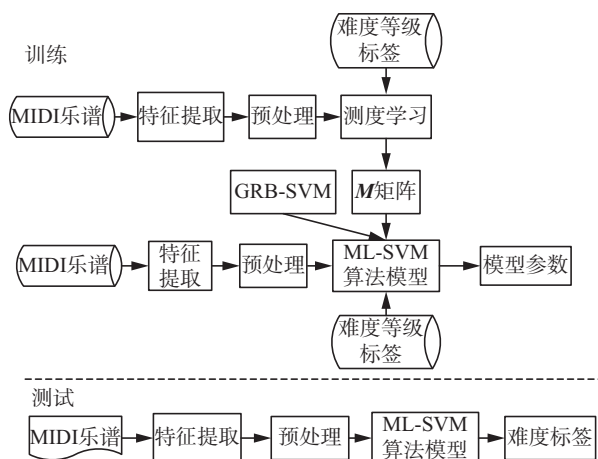


图 1 ML-SVM 算法的框图

Fig. 1 The frame chart of ML-SVM algorithm

## 2.1 特征空间的建立

为获取更多的钢琴乐谱难度相关特征,本文采用文献[1]与文献[5]中的难度相关特征作为本文的特征空间。所以本文中的特征空间包括:文献[1]中除指法复杂度(属于乐谱标注层面信息, MIDI 乐谱中不包含此信息),调号(key signature 是元标签信息)之外的全部难度和语义特征;文献[5]中除去指法特征(同样属于标注层面特征)的全部特征,共 22 个特征,组成一个 22 维的特征向量来表示 MIDI 乐谱。

## 2.2 测度学习得到距离测度 $D_M$ , 改进 GRB 核函数

由于传统的 SVM 算法无法根据特征对类别的辨别能力差别对待,所以本文考虑利用测度学习从训练乐谱中有监督地学习到一个投影矩阵,充分考虑特征对难度等级的贡献度,为特征分配不同的权重,进而得到新的距离测度  $D_M$ ,改进原始高斯径向基核函数,改进后的高斯径向基核函数为

$$k_M(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2\sigma^2} D_M(\mathbf{x}_i, \mathbf{x}_j)\right) \quad (1)$$

式中:  $\sigma$  是高斯径向基核函数参数,可用网格搜索算法得到,  $\mathbf{x}_i$ 、 $\mathbf{x}_j$  表示特征向量,  $\exp$  表示以自然常数  $e$  为底的指数函数,  $D_M$  为新的距离测度:

$$D_M = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \quad (2)$$

通过矩阵  $\mathbf{M}$ , 依据特征对难度等级的区别能力及特征之间的相关关系,将特征投影到一个类别区分度更高的空间<sup>[14]</sup>。  $D_M$  的构建及投影矩阵  $\mathbf{L}(\mathbf{L}^T \mathbf{L} = \mathbf{M})$  的学习过程如下。

本算法的关键在于找到最佳的特征投影矩阵  $\mathbf{L}$ 。考虑用变换矩阵  $\mathbf{L} \in R^{n \times n}$  ( $n$  表示特征的维数) 实现特征投影:

$$\mathbf{x}'_i = \mathbf{L} \mathbf{x}_i \quad (3)$$

则投影后的特征向量  $\mathbf{x}'_i$  是原始特征向量  $\mathbf{x}_i$  和投影矩阵  $\mathbf{L}$  元素的一个线性组合,则投影空间中距离测度  $d_L$  为

$$d_L(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)\|_2 = \sqrt{[\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)]^T [\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)]} \quad (4)$$

为避免求均方并保证距离为正值,取距离的平方并用矩阵形式表示,则新的距离测度  $D_M$  为

$$D_M = d_L^2 = [\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)]^T [\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j)] = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{L}^T \mathbf{L} (\mathbf{x}_i - \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \quad (5)$$

式中: 矩阵  $\mathbf{M} = \mathbf{L}^T \mathbf{L}$ 。希望投影后,相同难度标签的乐谱之间距离减小,不同难度标签的乐谱之间距离增大。求解  $\mathbf{M}$  的过程归结为求解如下优化问题:

$$\begin{aligned} \min_{\mathbf{M} \geq 0} & \sum_{(i,j) \in S} D_M(\mathbf{x}_i, \mathbf{x}_j) \\ \text{s.t.} & \sum_{(i,j,k) \in R} [D_M(\mathbf{x}_i, \mathbf{x}_j) - D_M(\mathbf{x}_i, \mathbf{x}_k)] \geq 1 \end{aligned} \quad (6)$$

式中:  $S$  表示由相同难度等级的两个乐谱组成的数据对集 ( $\mathbf{x}_i$ 、 $\mathbf{x}_j$  难度等级相同),  $R$  表示在  $S$  的基础上,每个对集加入一个不同难度等级的乐谱组成的三集合 ( $\mathbf{x}_i$ 、 $\mathbf{x}_j$  难度等级相同,  $\mathbf{x}_i$  和  $\mathbf{x}_k$  难度等级不同)。此优化目标与大间隔近邻算法 (large margin nearest neighbor, LMNN)<sup>[15]</sup> 有相似之处。由于  $\mathbf{M}$  是半正定对称矩阵,所以此优化问题是凸的,可以用梯度下降法有效求解<sup>[15]</sup>,并且只有全局最优解。

## 2.3 ML-SVM 算法

对于给定数据样本  $\{(\mathbf{x}_i, y_i), i = 1, 2, \dots, p\}$ , 其中  $\mathbf{x}_i \in R^n$  为  $n$  维向量,  $y_i \in \{-1, +1\}$  为样本标签,表示两个不同的类别,  $p$  为样本个数。则基于测度学习得到新的距离测度  $D_M$ ,改进高斯径向基核函数后的 ML-SVM 算法模型求解最优超平面的问题即为求解下列目标函数:

$$\begin{aligned} \max_{\alpha} f(\alpha) &= \sum_{i=1}^p \alpha_i - \frac{1}{2} \sum_{i,j=1}^p \alpha_i \alpha_j y_i y_j k_M \\ \text{s.t.} & \sum_{i=1}^p \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C, i = 1, 2, \dots, p \end{aligned} \quad (7)$$

式中  $k_M = \exp\left(-\frac{1}{2\sigma^2} D_M(\mathbf{x}_i, \mathbf{x}_j)\right)$ 。

得到最优解  $\alpha^* = [\alpha_1^* \alpha_2^* \dots \alpha_n^*]^T$ 。选择  $\alpha^*$  的一个小于  $C$  的正分量  $\alpha_j^*$ ,并据此计算:

$$b^* = y_j - \sum_{i=1}^p y_i \alpha_i^* k(\mathbf{x}_i, \mathbf{x}_j) \quad (8)$$

构造决策函数:



$$f(\mathbf{x}) = \text{sgn}\left(\sum_{i=1}^p y_i \alpha_i^* k(\mathbf{x}, \mathbf{x}_i) + b^*\right) \quad (9)$$

依据决策函数可以得到待分类乐谱  $\mathbf{x}$  所属类别<sup>[8]</sup>。其中  $\alpha$  是拉格朗日系数,  $C$  是错误分类的惩罚参数,  $k_M$  是改进后的高斯径向基核函数, 核函数表达式中的  $D_M(\mathbf{x}_i, \mathbf{x}_j)$  是测度学习得到的距离测度。

另外, 原始 SVM 是二分类的, 本文采用一对余 (one versus rest) 方法<sup>[8,16]</sup>将 ML-SVM 扩展到多分类。主要思想是: 对于一个  $m$  类分类问题, 通过建立  $m$  个二分类 ML-SVM 模型, 每一个 ML-SVM 模型需要训练所有的训练样本, 但训练样本中只将一个类别记为正, 其余所有类别记为负。对于待识别的样本, 依次调用训练好的  $m$  个 ML-SVM 模型, 计算它在各个 ML-SVM 模型中决策函数的值, 选择决策函数值最大的 ML-SVM 模型给出的类别, 作为待识别乐谱难度类别。

### 3 实验

本文算法全部用 MATLAB 软件实现, 所用的计算机环境为 32 位 Windows 7 操作系统, 内置 Intel I5-4200M 处理器和 4 GB 的内存。

在实验中, 为更好地评估本文提出的 ML-SVM 算法的分类性能和泛化能力, 在 9 类和 4 类难度两个乐谱数据集中, 将 ML-SVM 算法与逻辑回归、基于线性核函数的 SVM、基于多项式核函数的 SVM、基于原始高斯径向基核函数的 SVM (均用 one versus rest 扩展到多分类) 算法以及结合主成分分析的各个 SVM 算法进行对比试验, 以识别准确率作为算法性能的评价指标。每个实验独立重复 5 次, 并用五折交叉验证, 取平均准确率作为最终识别准确率。同时为更全面评估识别算法性能, 也给出各个算法结果的 90% 置信区间。

#### 3.1 实验数据集

钢琴乐谱数据集采用包含音高、节拍、时间、和弦、速度和信道等钢琴乐谱信息的 MIDI 格式数字乐谱文件<sup>[17]</sup>。MIDI 文件小, 易于获得。为和现有的研究作对比, 我们采用了文献<sup>[1]</sup>中包含 9 个难度等级, 共 176 个 MIDI 文件的数据集。

另外考虑到实际钢琴学习与教学中很多情况会将乐谱分为 4 个难度等级, 大量音乐网站也普遍提供 4 个难度等级 (easy, beginner, intermediate, advanced) 的数字乐谱, 所以为更好地评估本文算法的可拓展性, 切合实际应用情况, 我们还从大型音乐网站 8notes<sup>[18]</sup>收集到 400 首 MIDI 乐谱组成有 4 个难度等级的数据集, 每一个难度等级有 100 个 MIDI

乐谱。为书写及引用方便, 9 个难度等级的数据集简称为 NineS 数据集, 4 个难度等级的数据集简称为 FourS 数据集。

#### 3.2 数据预处理

特征提取后, 一些特征对应的值较大, 而另一些特征对应的值较小, 甚至相差超过两个数量级, 为避免数值较大的特征对整体分类的影响, 利用 Min-Max 归一化方法:

$$\mathbf{x}^* = \frac{\mathbf{x} - \min}{\max - \min} \quad (10)$$

将特征向量归一化到  $[0, 1]$ 。其中  $\min$  和  $\max$  分别表示特征  $\mathbf{x}$  的最小和最大值,  $\mathbf{x}^*$  表示特征  $\mathbf{x}$  经过归一化处理后的特征。

#### 3.3 实验与结果分析

仿真试验中, 将 ML-SVM 算法与逻辑回归 (logistic regression, LR)<sup>[19]</sup>, 基于线性核函数的 SVM (记为 L-SVM), 基于多项式核函数的 SVM (记为 P-SVM) 和基于高斯径向基核函数的 SVM 算法 (记为 GRB-SVM)<sup>[16,20-21]</sup>进行对比。每个实验独立重复 5 次, 每次用 5 折交叉验证, 取平均准确率作为分类性能指标, 同时计算出结果的 90% 置信区间。

各个 SVM 算法中核函数参数利用网格搜索算法<sup>[22]</sup>, 在  $2^{-10} \sim 2^{10}$  内, 以步长 0.5, 5 折交叉验证寻找最优的参数设置。其中 L-SVM, P-SVM (多项式阶数  $d=3$ ) 只需优化惩罚因子  $C$ , GRB-SVM 与 ML-SVM 需要优化惩罚因子  $C$  与核函数参数  $g$  ( $g = \frac{1}{\sigma^2}$ ) 的最优组合。最终的最优参数组合如表 1 所示。

表 1 各个算法的最优参数组合

Table 1 Optimal parameter combination of each algorithm

算法	NineS 数据集		FourS 数据集	
	惩罚因子 $C$	核函数参数	惩罚因子 $C$	核函数参数
L-SVM	16	—	181.019 3	—
P-SVM	22.627 4	—	128	—
GRB-SVM	8	1.414 2	8	0.707 1
ML-SVM	22.627 4	0.176 8	45.254 8	0.088 39

表 2 给出了本文提出的 ML-SVM 算法与 LR、L-SVM、P-SVM、GRB-SVM 算法在 NineS 数据集和 FourS 数据集中的识别准确率及结果的 90% 置信区间。从表 2 中可以看出, 在两个数据集上, 本文提出的 ML-SVM 算法, 识别准确率最高, 分别达到 68.74% 和 84.67%。两个数据集中, 本文提出的算法最终识别准确率均高于 GRB-SVM 算法, 尤其在 FourS 数据集中, 本文所提算法得到 84.67% 的识别准确率, 相对 75.63% 的 GRB-SVM, 识别准确

率有较大提高。且更窄的置信区间表明结果更稳定、显著。基于线性核函数的 SVM(L-SVM) 一直表现欠佳, 在两个数据集中识别效果不如逻辑回归 (LR), 基于多项式核函数的 SVM(P-SVM) 表现良好, 在 FourS 数据集中识别效果仅次于本文算法。

表 2 各算法的识别准确率及 90% 置信区间

Table 2 Recognition accuracy and 90% confidence interval of each algorithm %

算法	NineS 数据集		FourS 数据集	
	识别准确率	90% 置信区间	识别准确率	90% 置信区间
LR	64	53.7~74.3	70.28	64.6~80.8
L-SVM	61	55.3~66.7	71.30	66.7~75.9
P-SVM	65.42	56.22~74.62	78.37	72.37~84.37
GRB-SVM	66	59.8~72.2	75.63	70.21~81.05
ML-SVM	68.74	60.94~78.54	84.67	80.42~88.92

为进一步验证本文所提算法的有效性, 我们将 ML-SVM 算法与另一种主要用来实现特征降维的投影算法——主成分分析结合 SVM 的分类准确率进行对比。在对原始特征数据进行 PCA 处理时, 保留原始特征 95% 信息量的投影特征。PCA 投影处理后, 最终 NineS 数据集中的特征降到 13 维, 而 FourS 数据集的特征降到 8 维。之后再基于各个核函数的 SVM 算法对 PCA 降维后的数据进行分类。实验结果如表 3 所示。从表中可以看出, 原始特征数据经过 PCA 处理后, 最终各个 SVM 的分类准确率都有所提高。这是因为原始特征数据经过 PCA 投影、降维之后, 可以有效减少混叠以及冗余信息, 进而提高最终分类的准确率。

表 3 PCA 处理后, 各算法的识别准确率

Table 3 After features are processed by PCA, each SVM algorithm's recognition accuracy in two data sets %

算法	NineS 数据集中	FourS 数据集中
	识别准确率	识别准确率
L-SVM	61.58	72.56
P-SVM	66.12	80.15
GRB-SVM	67.2	78.42

虽然原始特征数据经过 PCA 投影、降维处理后, 利用 GRB-SVM 分类的准确率较之前有所提高, 但分类准确率与本文提出的 ML-SVM 算法仍有差距, 尤其在 FourS 数据集中, ML-SVM 仍有较大优势, 这也再次验证了本文提出的 ML-SVM 算法是有效的。

## 4 结束语

针对现有钢琴乐谱难度分类主要由人工方式完成, 效率不高, 区别于传统将乐谱难度等级识别归结为回归问题, 本文直接将其建模为基于支持向量机的分类问题。并结合钢琴乐谱分类主观性强、特征之间普遍存在相关性等特点, 利用测度学习改进高斯径向基核函数, 从而提出一种测度学习支持向量机分类算法——ML-SVM 算法。经过在 9 类和 4 类难度两个乐谱数据集上的对比实验, 结果表明本文所提算法的识别准确率优于现有算法, 且有效提高了基于高斯径向基核函数 SVM 的分类性能。实验结果说明, 利用测度学习理论, 保留特征之间的相关关系, 更适合乐谱难度识别数据与分布特点, 并能够有效识别算法的性能。未来的工作可以考虑应用半监督算法, 以充分利用大量无难度标签数据, 预期将会大大提高分类器的训练效果, 进而提高算法的识别准确率。

## 参考文献:

- [1] CHIU S C, CHEN M S. A study on difficulty level recognition of piano sheet music[C]//IEEE International Symposium on Multimedia. Irvine, CA, USA: IEEE, 2012: 17–23.
- [2] ROBNIK-ŠIKONJA M, KONONENKO I. Theoretical and empirical analysis of Relief[J]. Machine learning, 2003, 53(1/2): 23–69.
- [3] JAMES G, WITTEN D, HASTIE T, et al. An introduction to statistical learning with applications in R[M]. New York: Springer, 2013: 59–102.
- [4] SMOLA A J, SCHÖLKOPF B. A tutorial on support vector regression[J]. Statistics and computing, 2003, 14(3): 199–222.
- [5] SÉBASTIEN V, RALAMBONDRAINNY H, SÉBASTIEN O, et al. Score analyzer: automatically determining scores difficulty level for instrumental e-learning[C]//Proceedings of the 13th International Society for Music Information Retrieval Conference. Porto, Portugal: ISMIR, 2012: 571–576.
- [6] CASTAN G, GOOD M, ROLAND P. Extensible markup language (XML) for music applications: an introduction, the virtual score: representation, retrieval, restoration[M]. Cambridge: MIT Press, 2001: 95–102.
- [7] WARD JR J H. Hierarchical grouping to optimize an objective function[J]. Journal of the American statistical association, 1963, 58(301): 236–244.
- [8] 丁世飞, 齐丙娟, 谭红艳. 支持向量机理论与算法研究综述[J]. 电子科技大学学报, 2011, 40(1): 2–10.  
DING Shifei, QI Bingjuan, TAN Hongyan. An overview on theory and algorithm of support vector machines[J]. Journal

- of university of electronic science and technology of China, 2011, 40(1): 2–10.
- [9] LI Shutao, KWOK J T, ZHU Hailong, et al. Texture classification using the support vector machines[J]. Pattern recognition, 2003, 36(12): 2883–2893.
- [10] SIMON T, KOLLER D. Support vector machine active learning with applications to text classification[J]. The journal of machine learning research, 2002, 2: 45–66.
- [11] OSUNA E, FREUND R, GIROSIT F. Training support vector machines: an application to face detection[C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Juan, Puerto Rico, USA: IEEE, 1997: 130–136.
- [12] WAN V, CAMPBELL W M. Support vector machines for speaker verification and identification[C]//Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop. Sydney, NSW, Australia: IEEE, 2000, 2: 775–784.
- [13] SCHÖLKOPF B, SMOLA, A J. Learning with kernels[M]. GMD-For Schungszentrum Information Stechnik, 1998: 5–93.
- [14] KULIS B. Metric learning: a survey[J]. Foundations and trends in machine learning, 2012, 5(4): 287–364.
- [15] WEINBERGER K Q, SAUL L K. Distance metric learning for large margin nearest neighbor classification[J]. Journal of machine learning research, 2009, 10: 207–244.
- [16] HSU C W, LIN C J. A comparison of methods for multi-class support vector machines[J]. IEEE transactions on neural networks, 2002, 13(2): 415–425.
- [17] MIDI Manufacturers Association. An introduction to MIDI[M]. California: MIDI Manufacturers Association, 2009: 1–16.
- [18] Fours set data sources[EB/OL]. [2015-07-24]. <http://www.8notes.com>.
- [19] HOSMER D W, LEMESHOW S. Applied logistic regression[M]. New York: Wiley, 2000: 31–46.
- [20] WESTON J, WATKINS C. Multi-class support vector machines, CSD-TR-98-04[R/OL]. Egham: Royal Holloway University of London, 1998: 1–10.
- [21] CHANG C C, LIN C J. LIBSVM—a library for support vector machines[J/OL]. ACM transactions on intelligent systems and technology, 2011, 2(3): 27.
- [22] 徐晓明. SVM 参数寻优及其在分类中的应用[D]. 大连: 大连海事大学, 2014: 6–58.
- XU Xiaoming. SVM parameter optimization and its application in the classification[D]. Dalian: Dalian Maritime University, 2014: 6–58.

### 作者简介:



郭龙伟,男,1990年生,硕士研究生,主要研究方向为音乐信息检索。



关欣,女,1977年生,研究员,主要研究方向为音乐信息检索、统计学习、凸优化理论和音乐信号处理。



李镛,男,1974年生,教授,博士生导师,主要研究方向为医学图像处理、智能信息处理、滤波器设计、数字系统和微系统设计。发表学术论文30余篇,出版专著和教材8部。