第60卷 第2期 天 文 学 报 Vol.60 No.2
2019年3月 ACTA ASTRONOMICA SINICA Mar., 2019

doi: 10.15940/j.cnki.0001-5245.2019.02.016

基于XGBoost算法的恒星/星系分类研究*

李 超[1,2] 张文辉[3,4†] 林基明[1‡]

(1 桂林电子科技大学信息与通信工程学院 桂林 541004)
(2 桂林电子科技大学认知无线电与信息处理教育部重点实验室 桂林 541004)
(3 桂林电子科技大学广西云计算与大数据协同创新中心 桂林 541004)
(4 桂林电子科技大学广西高校云计算与复杂系统重点实验室 桂林 541004)

摘要 机器学习在当今的诸多领域已经取得了巨大的成功. 尤其是提升算法. 提升算法适应各种场景的能力较强、准确率较高, 已经在多个领域发挥巨大的作用. 但是提升算法在天文学中的应用却极为少见. 为解决斯隆数字巡天(Sloan Digital Sky Survey, SDSS)数据中恒星/星系暗源集分类正确率低的问题, 引入了机器学习中较新的研究成果–XGBoost (eXtreme Gradient Boosting). 从SDSS-DR7 (SDSS Data Release 7)中获取完整的测光数据集, 并根据星等值划分为亮源集和暗源集. 首先, 分别对亮源集和暗源集使用十折交叉验证法, 同时运用XGBoost算法建立恒星/星系分类模型; 然后, 运用栅格搜索等方法调优XGBoost参数; 最后, 基于星系的分类正确率等指标, 与功能树(Function Tree, FT)、Adaboost (Adaptive boosting)、随机森林(Random Forest, RF)、梯度提升决策树(Gradient Boosting Decision Tree, GBDT)、堆叠降噪自编码(Stacked Denoising AutoEncoders, SDAE)、深度置信网络(Deep Belief Network, DBN)等模型进行对比并分析结果. 实验结果表明: XGBoost在暗源分类中要比功能树算法的星系分类正确率提高了将近10%, 在暗源集的最暗星等中比功能树提高了将近5%. 同其他传统的机器学习算法和深度神经网络相比, XGBoost也有不同程度的提升.

关键词 恒星: 基本参数, 星系: 基本参数, 技术: 测光, 方法: 数据分析
中图分类号: P152; **文献标识码**: A

1 引言

近些年来, 随着各国空间科学技术的进步和大型巡天项目的不断开展, 天文数据量已经呈指数式增长, 数据量也以TB量级, 甚至以PB量级来计量, 天文学显然已经发展到

2019-01-07收到原稿, 2019-01-11收到修改稿

*广西云计算与大数据协同创新中心、广西高校云计算与复杂系统重点实验室项目(编号1716)资助

†zhangwh@guet.edu.cn

‡linjm@guet.edu.cn

了一个前所未有的阶段, 即大数据-巨信息量-全波段时代[1]. 面对如此庞大而又复杂的天文数据, 如何进行高效而且准确的数据分析显得极为重要.

恒星/星系分类一直是天文数据分析的基本内容之一, 而且人们对它的研究最早可以追溯到18世纪[2]. 基于形态、启发式分割等原始的恒星/星系分类方法在之前被广泛应用. 随着机器学习的不断发展, 越来越多基于恒星/星系分类算法的研究也随之展开. 如严太生等[3]通过去除离群化数据, 并且使用自动聚类的方法, 对SDSS-DR6 (Sloan Digital Sky Survey Data Release 6)的测光数据进行恒星/星系的分类, 结果表明自动聚类算法具有较高的效率; Vasconcellos等[4]使用了约13种不同的决策树算法对SDSS-DR7的测光数据进行了恒星/星系分类研究, 结果表明功能树决策树算法在恒星/星系分类的这个问题上要优于其他决策树算法; Sevilla-Noarbe等[5]基于SDSS-DR9 (SDSS Data Release 9)测光图像目录中给定的特征数据集, 做了Boosted决策树在恒星/星系分类问题上的应用研究, 实验结果表明Boosted决策树的分类性能要优于SDSS数据集中给定的type测光分类器; Kim等[6]提出了一个深度卷积网络框架, 并将其应用到天文图像数据中进行恒星/星系的分类, 取得了非常好的效果; 李俊峰等[7]通过深度置信网络(Deep Belief Network, DBN)、神经网络(Neural Network, NN)和支持向量机(Support Vector Machine, SVM)等算法对SDSS数据分类的性能对比, 研究并分析了3种自动光谱分类算法是否适用; 刘蓉等[8]提出了一种非参数回归与Adaboost (Adaptive boosting)相结合且对恒星光谱进行MK分类的方法, 将恒星按照其光谱型和光度型进行分类, 同时识别出其光谱型的次型; Xan等[9]在集成学习的背景下探索了天文学中恒星/星系的分类, 并给出了合理的解释. 虽然在天文学领域, 已经研究并使用了很多优秀的算法, 但是这些算法都存在一些问题, 比如泛化能力弱. 即在亮源集有很高的分类正确率, 但在暗源集分类正确率低的问题始终无法得到有效的解决.

到目前为止, 国内外将XGBoost (eXtreme Gradient Boosting)算法应用到天文数据挖掘领域的并不多见, 尤其是用来研究恒星/星系的分类. 基于此, 本文研究了基于XGBoost的恒星/星系分类算法, 首次将XGBoost方法应用到SDSS-DR7测光数据之中, 并将XGBoost与功能树(Function Tree, FT)、Adaboost、随机森林(Random Forest, RF)、梯度提升决策树(Gradient Boosting Decision Tree, GBDT)、堆叠降噪自编码(Stacked Denoising AutoEncoders, SDAE)和DBN等模型的分类效果进行对比, 验证XGBoost方法在天文学研究中的应用价值.

2 斯隆数字巡天

迄今为止, 世界上已经有非常多的巡天项目投入使用, 但是在众多巡天项目中, SDSS被认为是最成功, 也是最有影响力的一个. SDSS的测光系统分别对天体进行u、g、r、i、z 5个波段的测量. 本文使用的测光数据只针对r波段. 在测光数据中, 同时带有光谱证认参数和测光参数的数据集仅占全部测光数据集的极少一部分, 剩下的绝大部分只有测光参数. 这意味着, 本文提出的XGBoost恒星/星系分类模型可能是对那些没有光谱证认参数的天体进行准确分类的一个有效方法.

3　提升算法

提升算法基于这样一种思想: 即对于任何一个复杂的任务来说, 将多个专家的判断进行适当的综合所得出的最终判断, 要比其中任何一个专家单独的判断好. 实际上, 这和“三个臭皮匠顶个诸葛亮”的道理是相似的. 提升算法是一种非常常用的统计学习方法, 其应用非常广泛并且有很好的效果. 在分类问题中, 首先, 它通过更新训练样本的权重, 能够学习到多个分类器. 然后, 再将这些分类器进行线性组合, 以此来提高分类器的分类性能.

3.1　GBDT原理

梯度提升决策树[10](GBDT)算法本质上是一种以决策树作为基函数的提升算法. 梯度提升决策树模型可以表示为决策树的加法模型:

$$f_M(\boldsymbol{x}) = \sum_{m=1}^{M} T(\boldsymbol{x};\theta_m) \ , \tag{1}$$

其中, $\boldsymbol{x}$表示样本数据集, $T(\boldsymbol{x};\theta_m)$表示决策树, θ_m表示决策树的参数, M表示决策树的个数. 梯度提升树使用前向分布算法. 首先, 它需要确定初始的提升树$f_0(\boldsymbol{x}) = 0$. 然后, 根据前向分布算法得出第m步的模型:

$$f_m(\boldsymbol{x}) = f_{m-1}(\boldsymbol{x}) + T(\boldsymbol{x};\theta_m) \ , \tag{2}$$

其中, $f_{m-1}(\boldsymbol{x})$是当前的模型. 最后, 根据经验风险最小化确定下一棵决策树的参数$\widehat{\theta}_m$:

$$\widehat{\theta}_m = \arg\min_{\theta_m} \sum_{i=1}^{N} L(y_i, f_{(m-1)}(\boldsymbol{x}) + T(\boldsymbol{x};\theta_m)) \ , \tag{3}$$

式中, y_i表示第i个样本的真实标签, N表示数据样本的个数. 当L采用平方误差形式的损失函数时:

$$L(\boldsymbol{y}, f(\boldsymbol{x})) = (\boldsymbol{y} - f(\boldsymbol{x}))^2 \ , \tag{4}$$

其中, $\boldsymbol{y}$表示所有样本数据的真实标签. 这时, 损失函数变为:

$$L\left(\boldsymbol{y}, f_{(m-1)}(\boldsymbol{x}) + T(\boldsymbol{x};\theta_m)\right) = \left[\boldsymbol{y} - f_{(m-1)}(\boldsymbol{x}) - T(\boldsymbol{x};\theta_m)\right]^2 \ . \tag{5}$$

如果是对于分类问题, GBDT算法需要将基分类器限制为分类树. 虽然训练数据中的输入和输出之间可能存在着较为复杂的关系, 但是决策树模型本身固有的特点决定了决策树的线性组合可以很好地拟合训练数据, 并得到模型参数.

3.2　XGBoost原理

XGBoost[11]也是提升算法的一种. 与传统的GBDT在优化时使用一阶导数信息不同, XGBoost在优化时做出了很好的改进. 它通过对损失函数进行2阶泰勒展开, 在保留一阶导数信息的同时也加入了2阶导数的信息, 这样可以使得模型在训练集上更快地收敛. 不仅如此, XGBoost为了控制模型的复杂程度, 还在损失函数中添加了一个正则项, 防止模型出现过拟合. XGBoost算法具体推导过程如下. 设$\boldsymbol{D} = \{(x_i, y_i)\}(|\boldsymbol{D}| = n, x_i \in$

$R^d, y_i \in R$)为一个拥有n个样本、每个样本有d个特征的数据集; x_i表示第i个样本数据. 树的集成模型通过K (树的数目)个相加函数来预测最终结果:

$$\widehat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in F \ , \tag{6}$$

其中, $F = \{f(\boldsymbol{x}) = w_{q(x)}\}(q : R^d \to T, w \in R^T)$ (q表示将样本实例R^d映射到相应叶索引的结构, T表示叶子节点的数目, R^T为叶子节点权重w的空间)代表了一个决策树的函数空间, 样本x_i和预测值$\widehat{y}_i$的函数关系记为ϕ; $w_{q(x)}$把每一个节点映射成一个值, 即$f(\boldsymbol{x})$的值; f_k表示第k棵树的模型. 每一个f_k对应着一个独立的树结构q和叶子节点的权值w. 为了学习模型中使用的函数集, 故定义正则化目标函数如下:

$$\begin{cases} L(\phi) = \sum\limits_i l(\widehat{y}_i, y_i) + \sum\limits_k \Omega(f_k) \\ \Omega(f) = \gamma T + \frac{1}{2}\lambda \parallel w\parallel^2 \end{cases} , \tag{7}$$

其中, l是一个用来衡量预测值$\widehat{y}_i$和真实值y_i之间差异的可微凸损失函数, Ω表示模型复杂度的惩罚项, γ表示叶子数目的正则化参数, 用来抑制节点继续向下分裂, λ表示叶子权重的正则化参数. 目标是最小化损失函数

$$L^{(t)} = \sum_{i=1}^{n} l\left(y_i, \widehat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \ , \tag{8}$$

其中, $L^{(t)}$表示第t棵树的目标函数; $\widehat{y}_i^{(t-1)}$表示前$t-1$棵树的输出值之和, 构成前$t-1$棵树的预测值; f_t表示第t棵树的模型, $f_t(x_i)$表示第t棵树的输出结果, $\widehat{y}_i^{(t-1)} + f_t(x_i)$相加构成最新的预测值. 定义$g_i$和$h_i$:

$$g_i = \frac{\partial l\left(y_i, \widehat{y}_i^{(t-1)}\right)}{\partial \widehat{y}_i^{(t-1)}} \ , \tag{9}$$

$$h_i = \frac{\partial^2 l\left(y_i, \widehat{y}_i^{(t-1)}\right)}{\partial^2 \widehat{y}_i^{(t-1)}} \ , \tag{10}$$

将损失函数在$\widehat{y}_i^{(t-1)}$处利用泰勒公式展开:

$$L^{(t)} \approx \sum_{i=1}^{n} \left[l(y_i, \widehat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \ . \tag{11}$$

去掉常数项, 第t次迭代后的损失函数变为:

$$L^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \ , \tag{12}$$

定义$I_j = \{i|q(x_i) = j\}$作为叶子节点j的实例集, 根据(12)式得:

$$\begin{aligned} L^{(t)} &= \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i {f_t}^2(x_i) \right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \ , \\ &= \sum_{j=1}^{T} \left[(\sum\nolimits_{i\in I_j} g_i) w_j + \frac{1}{2} (\sum\nolimits_{i\in I_j} h_i + \lambda) w_j^2 \right] + \gamma T \ , \end{aligned} \tag{13}$$

其中, w_j表示叶子节点j的权重. 对于固定的决策树的结构$q(\boldsymbol{x})$, 可以计算得出叶子节点j的最优权重w_j^*:

$$w_j^* = -\frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda} \ , \tag{14}$$

将w_j^*代回目标函数, 得:

$$L^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{(\sum_{i\in I_j} g_i)^2}{\sum_{i\in I_j} h_i + \lambda} + \gamma T \ . \tag{15}$$

(15)式作为衡量树结构质量的指标, 可以用来计算树结构q的得分. 即便如此, 想要列举出所有可能的树结构q几乎不可能. 因此, 需要使用贪心算法迭代地在每一个已有的叶子节点添加分支. 假定$I_{\rm L}$和$I_{\rm R}$是划分后左右子树叶子节点的集合, 即$I = I_{\rm L} \cup I_{\rm R}$, 则划分后的损失函数如下:

$$L_{\rm split} = \frac{1}{2} \left[\frac{(\sum_{i\in I_{\rm L}} g_i)^2}{\sum_{i\in I_{\rm L}} h_i + \lambda} + \frac{(\sum_{i\in I_{\rm R}} g_i)^2}{\sum_{i\in I_{\rm R}} h_i + \lambda} - \frac{(\sum_{i\in I} g_i)^2}{\sum_{i\in I} h_i + \lambda} \right] - \gamma \ . \tag{16}$$

4 实验测试

4.1 数据集介绍

为了能与已有算法进行更好的比较, 本研究采用的恒星/星系数据集是使用SQL (Structured Query Language)指令在SDSS数据库中提取, 并且与文献[4]保持一致. 数据特征如表1所示.

表 1 用于SDSS-DR7恒星/星系分类的特征

Table 1 The features for SDSS-DR7 star/galaxy classification

Variable	Attribute
psfMag	PSF (point-spread function) magnitude
fiberMag	Fiber magnitude
petroMag	Petrosian magnitud
modelMag	Model magnitude
petroRad	Petrosian radius
petroR50	Radius carrying 50% of Petrosian flux
petroR90	Radius carrying 90% of Petrosian flux
lnLStar	Likelihood PSF
lnLExp	Likelihood exponential
lnLDeV	Likelihood deVaucouleurs
mRrCc, mE1, mE2	Adaptive moments
specClass	Spectroscopic classification

4.2 实验分析

4.2.1 特征重要性测试

通过对数据特征仿真, 得知数据特征的重要程度如图1所示, 其中F score是表示特征重要程度的参数.

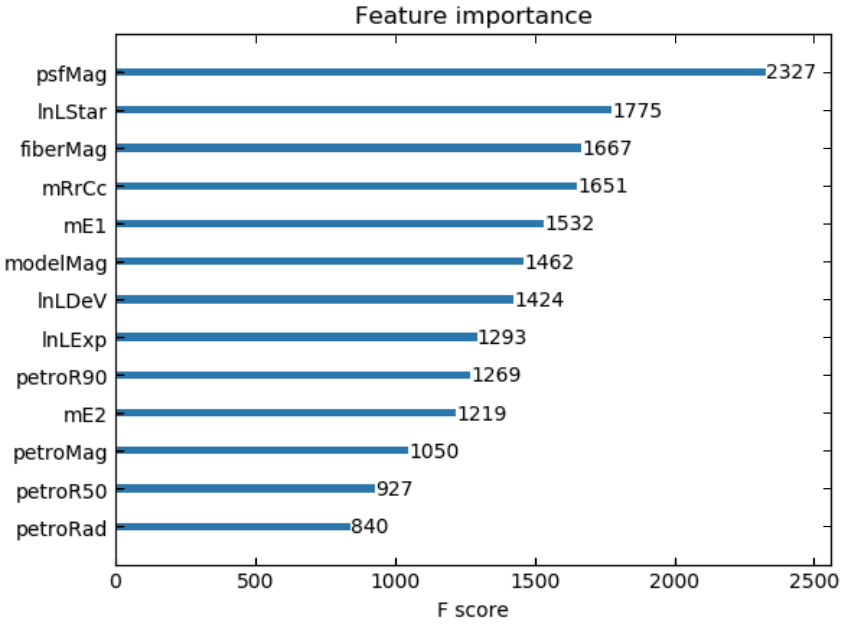

图 1 特征的重要程度

Fig. 1 Feature importance

4.2.2 XGBoost模型优化

XGBoost使用贪心算法, 其具体算法流程如下. 使用栅格搜索对XGBoost算法进行参数调优, 树的深度为6, 学习率为0.01, 在710次迭代下模型收敛, 达到最优值, 利用训练好的模型进行实验.

Algorithm 1: Exact Greedy Algorithm for Split Finding

Input: I, instance set of current node
Input: s, feature dimension
gain $\leftarrow 0$
$G \leftarrow \sum_{i\in I} g_i, H \leftarrow \sum_{i\in I} h_i$
for $k = 1$ to d **do**
  $G_{\mathrm{L}} \leftarrow 0, H_{\mathrm{L}} \leftarrow 0$
  for j in sorted (I, by x_{jk}) **do**
    $G_{\mathrm{L}} \leftarrow G_{\mathrm{L}} + g_j, H_{\mathrm{L}} \leftarrow H_{\mathrm{L}} + h_j$
    $G_{\mathrm{R}} \leftarrow G - G_{\mathrm{L}}, H_{\mathrm{R}} \leftarrow H - H_{\mathrm{L}}$
    score $\leftarrow \max\left(\text{score}, \frac{{G_{\mathrm{L}}}^2}{H_{\mathrm{L}}+\lambda} + \frac{{G_{\mathrm{R}}}^2}{H_{\mathrm{R}}+\lambda} - \frac{G^2}{H+\lambda}\right)$
  end
end
Output: Split with max score

4.2.3 实验方法及模型对比

为了能更好地评估XGBoost模型在恒星/星系分类上的性能, 使用了十折交叉验证的方法(将完整的数据集分为10等份, 其中1份作为测试集, 剩余9份作为训练集), 并且与文献[4]中的FT (分类性能优于其他传统决策树算法)、RF、GBDT、Adaboost和目前已经使用的新的算法如DBN、SDAE等作对比, 详细的对比结果如表2. 同样, 为了保证对比分类结果的有效性, 采用与文献[4]一致的分类性能衡量指标(CP), 即星系的分类正确率, 其定义如下:

$$\mathrm{CP}(v) = 100 \times \frac{N_{\mathrm{gal-gal}}(v)\delta v}{N^{\mathrm{tot}}_{\mathrm{galaxy}}(v)\delta v}\ , \tag{17}$$

其中, $N_{\mathrm{gal-gal}}(v)\delta v$代表星等值在$(v - \frac{\delta v}{2}, v + \frac{\delta v}{2})$区间内的数据样本被正确分类为星系的数量, $N^{\mathrm{tot}}_{\mathrm{galaxy}}(v)\delta v$代表星等值在$(v - \frac{\delta v}{2}, v + \frac{\delta v}{2})$区间内数据样本中星系的总数量. 数据集合是按照modelMag (模型星等)值的大小来划分区间的. 其中亮星等区间(14–19)、暗星等区间(19–21)和最暗星等区间(20.5–21)分别代表各个modelMag对应星等值大小的数据集合.

表 2 SDSS-DR7星系分类正确率
Table 2 The accuracy of SDSS-DR7 galaxy classification

Method \ Set	CP(14–19)/%	CP(19–21)/%	CP(20.5–21)/%
XGBoost	99.87	95.72	79.48
GBDT	99.84	95.74	77.64
Adaboost	99.89	95.80	77.56
RF	99.88	95.44	75.71
SDAE	99.87	95.70	73.08
DBN	99.60	96.01	74.45
FT	99.64	84.98	74.04

从通过仿真实验得出的表2中可以看出, XGBoost的星系分类准确率要优于FT. 尤其是在暗星等区间, XGBoost相比FT提高了近10%的准确率. 而与其他较为先进的DBN[12]、SDAE、RF、Adaboost、GBDT相比, 在modelMag值为20.5–21的最暗星等区间, 也提高了2%–5%的星系分类准确率, 由此可见, XGBoost算法模型具有更强的泛化能力, 在恒星星系分类问题上的表现优于其他算法. 另外, 本文利用modelMag属性值为14–19的约88万条数据, 来测试XGBoost、GBDT和Adaboost在亮星等数据集上训练模型时的效率. 使用亮源是因为在暗星等或者最暗星等数据量小的数据集上, 对比结果差异不明显. 结果如表3所示.

之所以没有测试其他模型的训练时间, 是因为其他模型的星系分类准确率要远低于以上3个模型. 实验结果表明, 在数据集不变的情况下, XGBoost在训练模型上所消耗的时间要远远低于GBDT和Adaboost. 相对于GBDT, XGBoost使用了2阶信息, 可以更快地在训练集上收敛. 因此, XGBoost不仅在准确率上优于其他模型, 而且在效率上也远高于GBDT和Adaboost.

表 3 模型训练时间
Table 3 The time of the model training

Method \ SetTime	CP(14–19) /h
XGBoost	1.44
GBDT	15.58
Adaboost	28.05

5 总结与展望

本文通过使用SDSS-DR7测光数据集, 并且采用十折交叉验证的方法, 研究了基于XGBoost算法的恒星/星系的分类问题. 最后通过使用经验值调参、栅格搜索等常用方法对模型不断调优, 基于星系分类准确率的评价指标, 与FT、Adaboost、RF、GBDT、SDAE、DBN等模型进行对比. 实验结果表明, 调优后的XGBoost算法模型在恒星/星系数据集上的分类效果要远好于其他模型. 同时, 在训练模型时, XGBoost要比GBDT和Adaboost更加高效. 因此, 无论是准确性还是高效性, XGBoost模型无疑都具有更加明显的优势. 虽然, 在恒星/星系暗源的准确性还有待进一步提高, 但是, 我相信随着XGBoost算法在天文学数据挖掘方面的研究逐步深入, 天文学相关领域将会快速发展.

参 考 文 献

[1] 张彦霞, 赵永恒. 科研信息化技术与应用, 2011, 2: 13
[2] Messier C. Connoissance des Temps for 1784, 1781: 227
[3] 严太生, 张彦霞, 赵永恒, 等. 中国科学G辑, 2009, 39: 1794
[4] Vasconcellos E C, De Carvalho R R, Gal R R, et al. AJ, 2010, 141: 189
[5] Sevilla-Noarbe I, Etayo-Sotos P. A&C, 2015, 11: 64
[6] Kim E J, Brunner R J. MNRAS, 2017, 464: 4463
[7] 李俊峰, 汪月乐, 胡升, 等. 光谱学与光谱分析, 2016, 36: 3261
[8] 刘蓉, 乔学军, 张健楠, 等. 光谱学与光谱分析, 2017, 37: 1553
[9] Xan M A, Ben H, David B. MNRAS, 2018, 481: 4194
[10] Jerome H. Friedman. The Annals of Statistics, 2001, 29: 1189
[11] Chen T, Guestrin C. ACM SIGKDD and International Conference on Knowledge Discovery and Data Mining, 2016: 785
[12] Hinton G E, Osindero S, Yee-Whye T. Neural Computation, 2006, 18: 1527

Research on Star/Galaxy Classification Based on XGBoost Algorithm

LI Chao[1,2] ZHANG Wen-hui[3,4] LIN Ji-ming[1]

(1 College of Information and Communication Engineering, Guilin University of Electronic Technology, Guilin 541004)
(2 Key Laboratory of Cognitive Radio and Information Processing, the Ministry of Education, Guilin University of Electronic Technology, Guilin 541004)
(3 Guangxi Cooperative Innovation Center of Cloud Computing and Big Data, Guilin University of Electronic Technology, Guilin 541004)
(4 Guangxi Colleges and Universities Key Laboratory of Cloud Computing and Complex Systems, Guilin University of Electronic Technology, Guilin 541004)

Abstract Machine learning, especially the life algorithm, has achieved great success in many areas today. The lifting algorithm has a strong ability to adapt to various scenarios with high accuracy, and has played a great role in many fields. But in astronomy, the application of lifting algorithms is rare. In response to the low classification accuracy of dark source sets in star/galaxy in the Sloan Digital Sky Survey (SDSS), a new research result in machine learning, eXtreme Gradient Boosting (XGBoost), was introduced. The complete photometric data set is obtained from the SDSS-DR7, and divided into a bright source set and a dark source set according to the magnitude. Firstly, the ten-fold cross-validation method is used for the bright source set and the dark source set respectively, and the XGBoost algorithm is used to establish the star/galaxy classification model. Then, the grid search and other methods are used to tune the XGBoost parameters. Finally, based on galaxies' classification accuracy and other indicators, the classification results are analyzed, comparing with the models of function tree (FT), Adaptive boosting (Adaboost), Random Forest (RF), Gradient Boosting Decision Tree (GBDT), Stacked Denoising AutoEncoders (SDAE), and Deep Belief Nets (DBN). The experimental results show that, the XGBoost improves the classification accuracy of galaxies in dark source classification by nearly 10% compared to the function tree algorithm, and improves the classification accuracy of galaxies in the darkest magnitude of dark source set by nearly 5% compared to the function tree algorithm. Compared with other traditional machine learning algorithms and deep neural networks, the XGBoost also has different degrees of improvement.

Key words stars: fundamental parameters, galaxies: fundamental parameters, techniques: photometric, methods: data analysis