



文档管理系统



哈佛大学的录取条件



访问：382573次

积分：5046

等级：**BLOG > 6**

排名：第6250名

原创：73篇

转载：270篇

xgboost原理及应用

标签：**xgboost**

2016-08-18 16:17

1228人阅读

评论(0)

收藏

分类：

Machine Learning (39)

目录(?)

[+]

1.背景

关于xgboost的原理网络上的资源很少，大多数还停留在应用层面，本文提供xgboost导读和实战地址，希望对xgboost原理进行深入理解。

2.xgboost vs gbdt

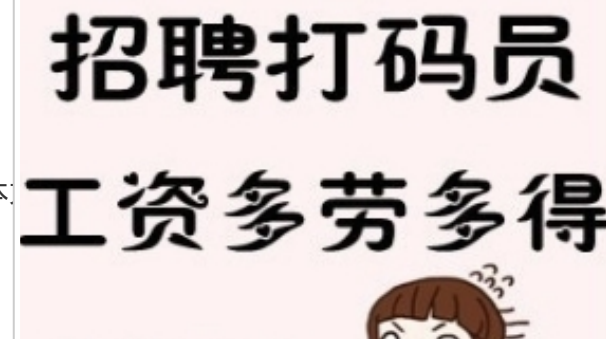
目录视图

摘要视图

RSS 订阅

图灵赠书——程序员11月书单 【思考】Python这么厉害的原因竟然是！ 感恩节赠书：《深度学习》等异
作译者评选启动！ 每周荐书：京东架构、Linux内核、Python全栈

关闭



在家兼职赚钱

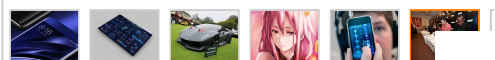




文档管理系统



哈佛大学的录取条件



Windows (4)

前端 (1)

算法 (41)

NLP (20)

推荐 (0)

Data mining (21)

Machine Learning (40)

Deep Learning (17)

软件开发设计 (1)

说到xgboost，不得不说gbdt。了解gbdt可以看我这篇文章 [地址](#)，gbdt无论在理论推导还是在应用场景实践都是相当完美的，但有一个问题：第n颗树训练时，需要用到第n-1颗树的（近似）残差。从这个角度来看，gbdt比较难以实现分布式（ps：虽然难，依然是可以的，换个角度思考就行），而xgboost从下面这个角度着手

- 目标 $Obj^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) + \text{constant}$

- 用泰勒展开来近似我们原来的目标

- 泰勒展开: $f(x + \Delta x) \simeq f(x) + f'(x)\Delta x + \frac{1}{2}f''(x)\Delta x^2$
- 定义: $g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$, $h_i = \partial_{\hat{y}^{(t-1)}}^2 l(y_i, \hat{y}^{(t-1)})$

$$Obj^{(t)} \simeq \sum_{i=1}^n \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) + \text{constant}$$

注：红色箭头指向的l即为损失函数；红色方框为正则项，包括L1、L2；红色圆圈为常数项。

利用泰勒展开三项，做一个近似，我们可以很清晰地看到，最终的目标函数只依赖于每个数据点的_____上的一阶导数和二阶导数。

3.原理

(1) 定义树的复杂度

对于f的定义做一下细化，把树拆分成结构部分q和叶子权重部分w。下映射到叶子的索引号上面去，而w给定了每个索引号对应的叶子分数是

关闭

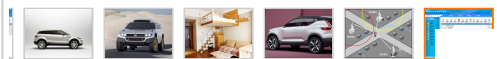
招聘打码员

工资多劳多得

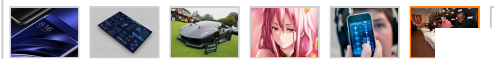
在家兼职赚钱



文档管理系统



哈佛大学的录取条件

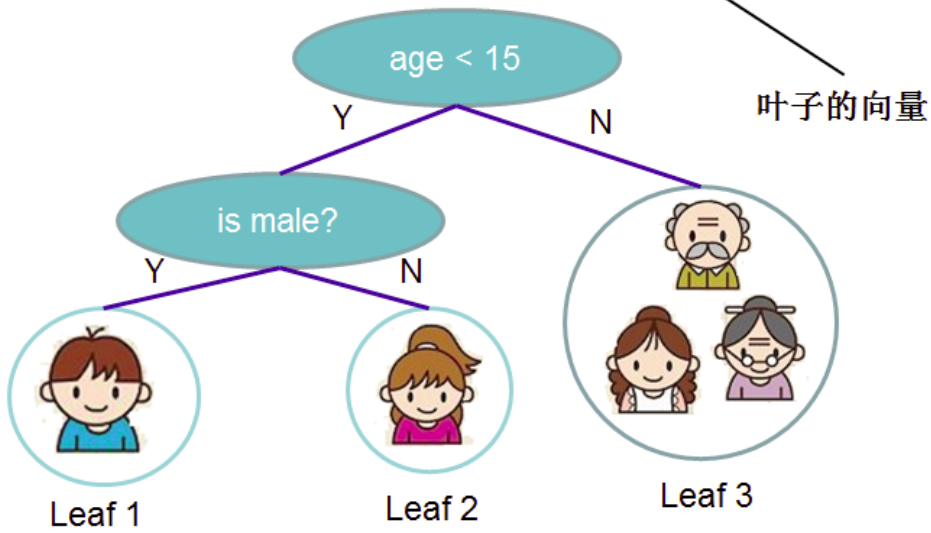


展开

阅读排行

- [mysql忽略主键冲突、避免重复..](#) (11200)
- [windows 目录表示（上级目录...](#) (8704)
- [R语言曲线拟合函数（绘图）](#) (7873)
- [时间序列完全教程（R）](#) (7828)
- [Word 表格换页自动“续表”方法](#) (7210)

$$f_t(x) = w_{q(x)}, \quad w \in \mathbf{R}^T, q: \mathbf{R}^d \rightarrow \{1, 2, \dots, T\}$$



w1=+2

w2=0.1

w3=-1

定义这个复杂度包含了一棵树里面节点的个数，以及每个树叶节点上面输出分数的L2模平方。

一种定义方式，不过这一定义方式学习出的树效果一般都比较不错。下图还给出了复杂度计算子。

关闭

招聘打码员

工资多劳多得



在家兼职赚钱

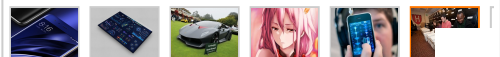




文档管理系统



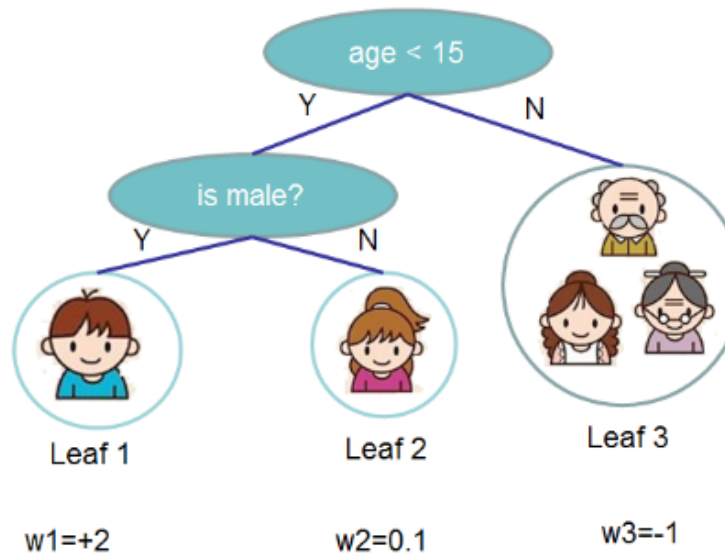
哈佛大学的录取条件



- * 【2017年11月27日】CSDN博客更新周报
- * 【CSDN】邀请您来GitChat赚钱啦！
- * 【GitChat】精选——JavaScript进阶指南
- * 改做人工智能之前，90%的人都没能给自己定位
- * TensorFlow 人脸识别网络与对抗网络搭建

$$\Omega(f_t) = \boxed{\gamma} T + \frac{1}{2} \boxed{\lambda} \sum_{j=1}^T w_j^2$$

叶子的个数 w的L2模平方



$$\Omega = \gamma 3 + \frac{1}{2} \lambda (4 +$$

注：方框部分在最终的模型公式中控制这部分的比重

在这种新的定义下，我们可以把目标函数进行如下改写，其中I被定义为 $I_j = \{i | q(x_i) = j\}$

$$\begin{aligned} Obj^{(t)} &\simeq \sum_{i=1}^n \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] \\ &= \sum_{i=1}^n \left[g_i w_{q(x_i)} + \frac{1}{2} h_i w_{q(x_i)}^2 \right] \\ &= \sum_{j=1}^T \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i \right) w_j^2 \right] \end{aligned}$$

这一个目标包含了TT个相互独立的单变量二次函数。我们可以定义

关闭

招聘打码员

工资多劳多得

在家兼职赚钱

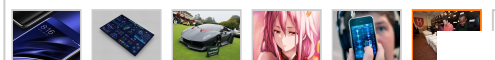




文档管理系统



哈佛大学的录取条件



吗？新手哈

用深度学习 (CNN RNN Attention) 解决大...
chvalrous : 我现在的场景是要对文章标题这种短文本进行分类,分类的类别为40+个,我使用textCNN进行试验,目...

网络爬虫工作原理分析

qq_36423458 : Python基础与爬虫技术 课程学习地址: <http://www.xuetuwuyou.com/cou...>

在windows下安装scala出现错误: 找不到或..
疯狂的赣江 : @vermouthlove:应该是空格的问题,不要安装在C:\Program Files下面,同

$$G_j = \sum_{i \in I_j} g_i \quad H_j = \sum_{i \in I_j} h_i$$

$$\begin{aligned} Obj^{(t)} &= \sum_{j=1}^T [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T \\ &= \sum_{j=1}^T [G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2] + \gamma T \end{aligned}$$

最终公式可以化简为

$$w_j^* = -\frac{G_j}{H_j + \lambda}$$






通过对 w_j 求导等于0,可以得到

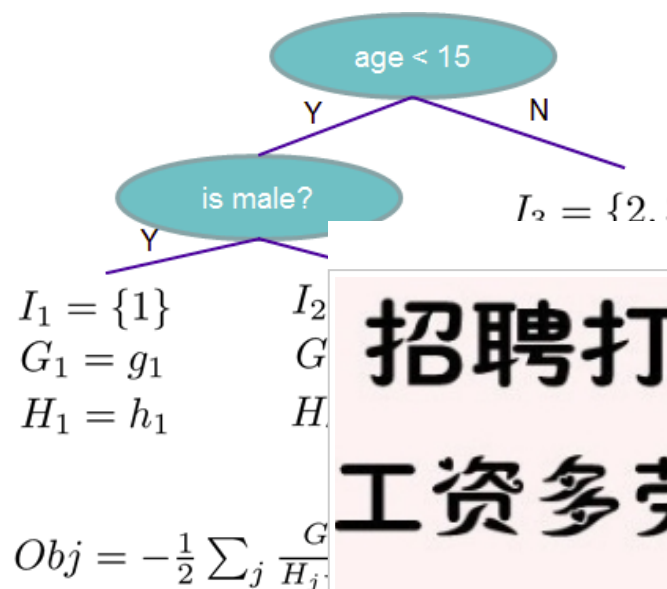
$$Obj = -\frac{1}{2} \sum_{j=1}^T \frac{G_j^2}{H_j + \lambda} + \gamma T$$

然后把 w_j 最优解代入得到:

(2) 打分函数计算示例

Obj代表了当我们指定一个树的结构的时候,我们在目标上面最多减少多少。我们可以把它叫做 (structure score)

样本号	梯度数据
1 	g_1, h_1
2 	g_2, h_2
3 	g_3, h_3
4 	g_4, h_4
5 	g_5, h_5



这个分数越小

(3) 枚举不同树结构的贪心法

关闭

招聘打码员

工资多劳多得

在家兼职赚钱







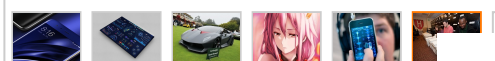




文档管理系统



哈佛大学的录取条件



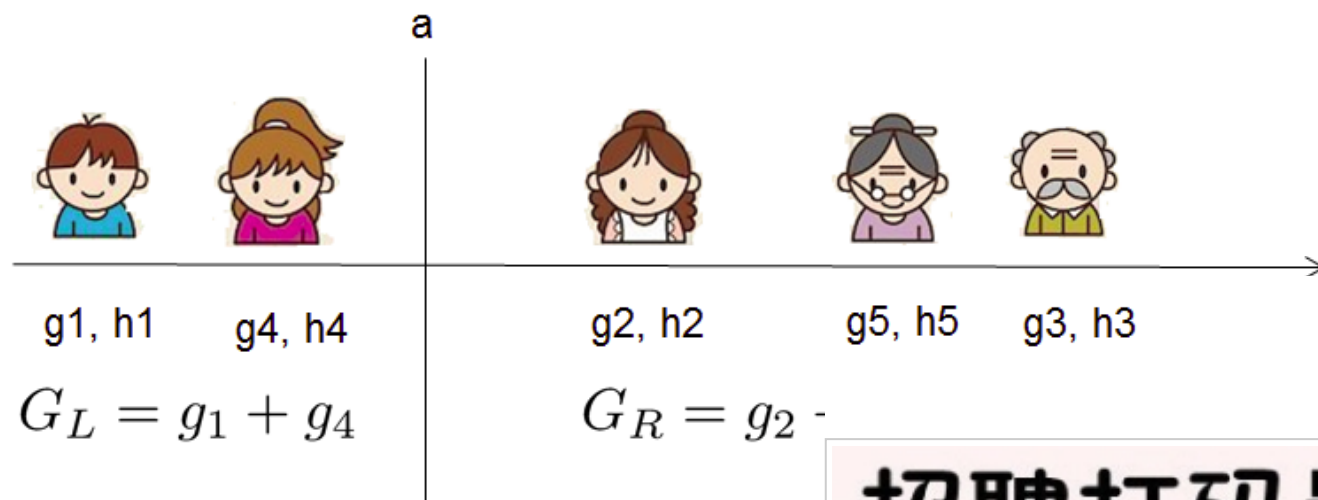
贪心法：每一次尝试去对已有的叶子加入一个分割

$$Gain = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma$$

左子树分数
右子树分数
加入新叶子节点引入的复杂度代价

不分割我们可以拿到的分数

对于每次扩展，我们还是要枚举所有可能的分割方案，如何高效地枚举所有的分割呢？我假设我们要枚举所有 $x < a$ 这样的条件，对于某个特定的分割 a 我们要计算 a 左边和右边的导数和。



关闭

我们可以发现对于所有的 a ，我们只要做一遍从左到右的扫描就可以枚举上面的公式计算每个分割方案的分数就可以了。

观察这个目标函数，大家会发现第二个值得注意的事情就是引入分割不引入新叶子的惩罚项。优化这个目标对应了树的剪枝，当引入的分割以剪掉这个分割。大家可以发现，当我们正式地推导目标的时候，像这样，而不再是一种因为heuristic而进行的操作了。

4.自定义损失函数

招聘打码员

工资多劳多得

在家兼职赚钱

在实际的业务场景下，我们往往需要自定义损失函数。这里给出一个官方的 [链接](#) [地址](#)

5.Xgboost调参

由于Xgboost的参数过多，使用GridSearch特别费时。这里可以学习下这篇文章，教你如何一步一步去调参。[地址](#)

6.python、R对于xgboost的简单使用

任务：二分类，存在样本不均衡问题（scale_pos_weight可以一定程度上解读此问题）

【Python】

```

109 def xgboost_predict():
110     import xgboost as xgb
111     #xgboost start here
112     dtest = xgb.DMatrix(test_x)
113     dval = xgb.DMatrix(val_X, label=val_y)
114     dtrain = xgb.DMatrix(X, label=y)
115     params={
116         'booster':'gbtree',
117         'objective': 'binary:logistic',
118         'early_stopping_rounds':100,
119         'scale_pos_weight': weight,
120         'eval_metric': 'auc',
121         'gamma': '0.1',
122         'max_depth':8,
123         'lambda':550,
124         'subsample':0.7,
125         'colsample_bytree':0.4,
126         'min_child_weight':3,
127         'eta': 0.02,
128         'seed':random_seed,
129         'nthread':7
130     }
131     watchlist = [(dval,'val'), (dtrain,'train')]
132     xgboost_model = xgb.train(params,dtrain,num_boost_round=1000)
133     #xgboost_model.save_model('./model/xgb.model')
134
135     #predict test set (from the best iteration)
136     xgboost_predict_y = xgboost_model.predict(dtest,ntree=1000)

```

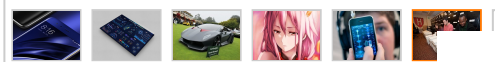
关闭



文档管理系统



哈佛大学的录取条件



招聘打码员

工资多劳多得



在家兼职赚钱

招聘打码员
工资多劳多得
赶快报名

招聘打码员
工资多劳多得
赶快报名

招聘打码员
工资多劳多得
赶快报名

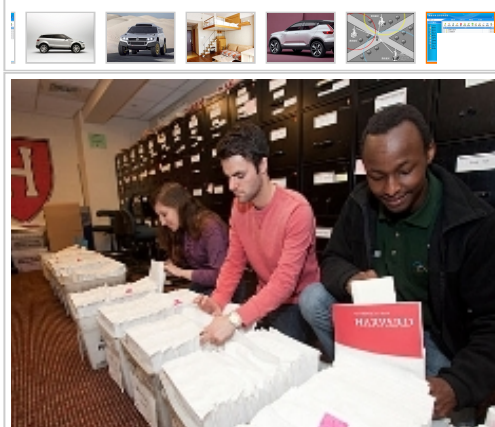
招聘打码员
工资多劳多得
赶快报名

招聘打码员
工资多劳多得
赶快报名

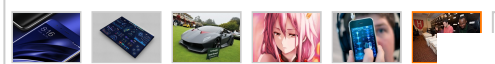
招聘打码员
工资多劳多得
赶快报名



文档管理系统



哈佛大学的录取条件



【R】

```

124 # fit xgboost model
125 dtrain=xgb.DMatrix(data=train.new[,-1],label=1-train.y$y) #train.new[,-1]表示去掉第一列，即uid列
126 dtest= xgb.DMatrix(data=test.new[,-1])
127 # dunlabeled = xgb.DMatrix(data=train_unlabeled.new[,-1])
128
129 p = nrow(train.y[train.y[,2] == 1,])
130 n = nrow(train.y[train.y[,2] == 0,])
131 weight = p/n
132
133 model_xgboost=xgb.train(
134     booster='gbtree',
135     objective='binary:logistic',
136     scale_pos_weight=weight,
137     gamma=0,
138     lambda=700,
139     subsample=0.7,
140     colsample_bytree=0.3,
141     min_child_weight=5,
142     max_depth=8,
143     eta=0.02,
144     data=dtrain,
145     nrounds=1520,
146     metrics='auc',
147     nthread=2
148 )
149
150 # predict probabilities
151 predict_xgboost=1-predict(model_xgboost,dtest)

```

7.xgboost中比较重要的参数介绍

(1) objective [default=reg:linear] 定义学习任务及相应的学习目标，

关闭

- “reg:linear” –线性回归。
- “reg:logistic” –逻辑回归。
- “binary:logistic” –二分类的逻辑回归问题，输出为概率。
- “binary:logitraw” –二分类的逻辑回归问题，输出的结果为wTx。
- “count:poisson” –计数问题的poisson回归，输出结果为poisson分布，缺省值为0.7。(used to safeguard optimization)
- “multi:softmax” –让XGBoost采用softmax目标函数处理多分类问题(输出为类数)

招聘打码员

工资多劳多得



在家兼职赚钱



招聘打码员
工资多劳多得
赶快报名





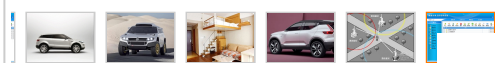




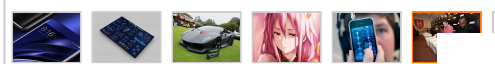




文档管理系统



哈佛大学的录取条件



- “multi:softprob” –和softmax一样，但是输出的是 $ndata * nclass$ 的向量，可以将该向量reshape成 $ndata$ 行 $nclass$ 列的矩阵。没行数据表示样本所属于每个类别的概率。
- “rank:pairwise” –set XGBoost to do ranking task by minimizing the pairwise loss

(2) 'eval_metric' The choices are listed below，评估指标:

- “rmse”: root mean square error
- “logloss”: negative log-likelihood
- “error”: Binary classification error rate. It is calculated as $\#(\text{wrong cases})/\#(\text{all cases})$. For the predictions, the evaluation will regard the instances with prediction value larger than 0.5 as positive instances, and negative instances.
- “merror”: Multiclass classification error rate. It is calculated as $\#(\text{wrong cases})/\#(\text{all cases})$.
- “mlogloss”: Multiclass logloss
- “auc”: Area under the curve for ranking evaluation.
- “ndcg”: Normalized Discounted Cumulative Gain
- “map”: Mean average precision
- “ndcg@n”, “map@n”: n can be assigned as an integer to cut off the top positions in the lists for evaluation.
- “ndcg-“, “map-“, “ndcg@n-“, “map@n-”: In XGBoost, NDCG and MAP will evaluate the score of a any positive samples as 1. By adding “-” in the evaluation metric XC consistent under some conditions.

(3) lambda [default=0] L2 正则的惩罚系数

(4) alpha [default=0] L1 正则的惩罚系数

(5) lambda_bias 在偏置上的L2正则。缺省值为0（在L1上没有偏置项）

(6) eta [default=0.3]

为了防止过拟合，更新过程中用到的收缩步长。在每次提升计算之后

关闭

招聘打码员

工资多劳多得



在家兼职赚钱



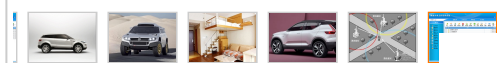




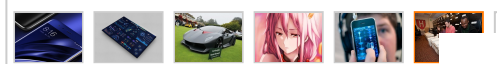




文档管理系统



哈佛大学的录取条件



缩减特征的权重使提升计算过程更加保守。缺省值为0.3

取值范围为： $[0,1]$

(7) `max_depth [default=6]` 数的最大深度。缺省值为6，取值范围为： $[1,\infty]$

(8) `min_child_weight [default=1]`

孩子节点中最小的样本权重和。如果一个叶子节点的样本权重和小于`min_child_weight`则拆分过程结束。在现行回归模型中，这个参数是指建立每个模型所需要的最小样本数。该成熟越大算法越conservative

取值范围为： $[0,\infty]$

更多关于Xgboost学习地址

(1) <https://github.com/dmlc/xgboost>

本文转载自: <http://blog.csdn.net/a819825294/article/details/51206410>

顶

0

踩

0

关闭

- 上一篇 xgboost: 速度快效果好的boosting模型
- 下一篇 IntelliJ IDEA 快捷键和设置

相关文章推荐

- 王小草【机器学习】笔记--提升之XGBoost工具的应..
- xgboost入门

招聘打码员

工资多劳多得



在家兼职赚钱

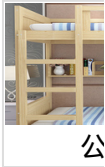
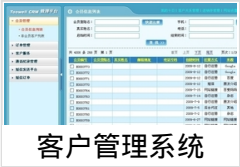




- 腾讯云容器服务架构实现介绍--董晓杰
- Python可以这样学（第三季：多线程与多进程编程）
- 使用vs2013 debug xgboost C++码源
- Linux下安装xgBoost
- 容器技术在58同城的实践--姚远
- 华为工程师，带你实战C++
- xgboost C++ window编译问题解决与安装
- 机器学习笔记（七）Boost算法（GDBT,AdaBoost,...
- Tensorflow项目实战-文本分类
- XGBoost解决多分类问题
- xgboost的使用简析
- win10 下xgboost的安装----终极版
- MySQL深入浅出
- XGBoost：参数解释

普林斯顿大學	5	5
劍橋大學	6	6
麻省理工學院	7	3
倫敦帝國學院	8	9
芝加哥大學	9	>10

世界排名大学



查看评论

暂无评论

发表评论

用户名： weixin_35068028

评论内容：



关闭

招聘打码员

工资多劳多得

在家兼职赚钱



提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 | 江苏乐知网络技术有限公司

SDN.NET, All Rights Reserved



文档管理系统



哈佛大学的录取条件



关闭

招聘打码员
工资多劳多得



在家兼职赚钱

