

CSDN新首页上线啦，邀请你来立即体验！(http://blog.csdn.net/)

立即体验

CSDN

博客 (http://blog.csdn.net/?ref=toolbar)学院 (http://edu.csdn.net/?ref=toolbar)

下载 (http://download.csdn.net/?ref=toolbar)更多 ▾

登录 (https://passport.csdn.net/account/login?ref=toolbar)注册 (http://passport.csdn.net/account/mobileregister?ref=toolbar&action=mobileRegister)

http://www.blog.csdn.net/posted/1697561.html

/activity?utm_source=csdnblog1)

实战xgboost与sklearn与pandas训练模型

原创 2017年11月09日 16:32:24 标签： xgboost (http://so.csdn.net/so/search/s.do?q=xgboost&t=blog)

64

[python] view plain copy print?

```
1. import cPickle
2. import xgboost as xgb
3. import numpy as np
4. from sklearn.model_selection import KFold, train_test_split, GridSearchCV
5. from sklearn.metrics import confusion_matrix, mean_squared_error
6. from sklearn.datasets import load_iris, load_digits, load_boston
7.
8. #用Xgboost建模，用sklearn做评估
9. #二分类问题，用混淆矩阵
10.
11. digits = load_digits()
12. y = digits['target']
13. X = digits['data']
14.
15. X.shape
16.
17. (1797, 64)
18. y.shape
19.
20. #K折的分器
21. kf = KFold(n_splits=2, shuffle=True, random_state=1234)
22. for train_index, test_index in kf.split(X):
23.     xgboost_model = xgb.XGBClassifier().fit(X[train_index], y[train_index])
24.     #预测结果
25.     pred = xgboost_model.predict(X[test_index])
26.     #标准答案
27.     ground_truth = y[test_index]
28.     print (confusion_matrix(ground_truth, pred))
29.
30. [[78 0 0 0 0 0 0 0 1 0]
31. [ 0 92 1 0 0 0 0 0 0 0]
32. [ 0 2 82 0 0 0 2 0 0 0]
33. [ 0 1 1 88 0 0 0 1 0 3]
34. [ 2 0 0 0 99 0 2 3 1 0]
35. [ 0 0 0 1 0 95 2 0 0 4]
36. [ 0 2 0 0 0 0 84 0 2 0]
37. [ 0 0 0 0 0 0 0 86 0 2]
38. [ 0 6 0 2 0 0 0 0 73 1]
39. [ 0 1 0 0 1 0 0 5 2 71]]
40. [[98 0 0 0 0 0 0 1 0 0]
41. [ 0 84 2 1 0 0 1 0 0 1]
42. [ 1 0 88 0 0 0 0 1 1 0]
43. [ 0 0 1 86 0 1 0 0 0 1]
44. [ 0 0 0 74 0 0 0 0 0 0]
45. [ 1 0 0 0 1 73 0 0 1 4]
46. [ 0 0 0 0 1 1 91 0 0 0]
47. [ 0 0 0 1 0 0 0 89 1 0]
48. [ 1 1 0 0 1 1 0 0 87 1]
49. [ 0 2 0 1 0 1 0 0 2 94]]
50.
51. #多分类
52. iris = load_iris()
53. y_iris = iris['target']
54. X_iris = iris['data']
55. kf = KFold(n_splits=2, shuffle=True, random_state=1234)
56. for train_index, test_index in kf.split(X_iris):
57.     xgboost_model = xgb.XGBClassifier().fit(X_iris[train_index], y_iris[train_index])
58.     #预测结果
59.     pred = xgboost_model.predict(X_iris[test_index])
60.     #标准答案
```

春华秋实369 (http://blog...)

+关注

(http://blog.csdn.net/qq_36330643)	码云
原创	粉丝
428	34
喜欢	未开通
9	(https://git

他的最新文章 更多文章 (http://blog.csdn.net/qq_36330643)

谈谈batchsize参数 (http://blog.csdn.net/qq_36330643/article/details/78661787)

Frequency 频率统计 (http://blog.csdn.net/qq_36330643/article/details/78646612)

NLTK完成简单的情感分析 (http://blog.csdn.net/qq_36330643/article/details/78646521)



Unable to Conn

The Proxy was unable to connect to the remote site, responding to requests. If you feel you have reached please submit a ticket via the link provided below.

URL: http://pos.baidu.com/s?hei=250&wid=300&di=u%2Fblog.csdn.net%2Fqq_36330643%2Farticle%2Fdi

博主专栏

Tensorflow学习

TensorFlow 3D

Tensorflow学习

Tensorflow学习 (http://blog.csdn.net/column/details/16952.html)

(http://blog.csdn.net/column/details/16952.html)

6879

在线课程

腾讯云容器服务架构

腾讯云容器服务架构

腾讯云容器服务架构

腾讯云容器服务架构 (http://edu.csdn.net/course/series_detail/73?utm_source=blog9)

0

腾讯云容器服务架构

腾讯云容器服务架构

腾讯云容器服务架构

腾讯云容器服务架构 (http://edu.csdn.net/course/series_detail/73?utm_source=blog9)

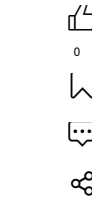
0

他的热门文章

scala通过mkString方法把一个集合转化为一个字符串 (http://blog.csdn.net/qq_36330643/article/details/76489573)

Spark SQL来读取现有Hive中的数据 (http://blog.csdn.net/qq_36330643/article/details/772722390)

Failed to install the TensorFlow installation (http://blog.csdn.net/qq_36330643/article/details/772722390)



```
61. ground_truth = y_iris[test_index]
62. print (confusion_matrix(ground_truth, pred))
63.
64. #回归问题
65. boston = load_boston()
66. y_boston = boston['target']
67. X_boston = boston['data']
68. kf = KFold(n_splits=2, shuffle=True, random_state=1234)
69. for train_index, test_index in kf.split(X_boston):
70.     xgboost_model = xgb.XGBRegressor().fit(X_boston[train_index], y_boston[train_index])
71.     #预测结果
72.     pred = xgboost_model.predict(X_boston[test_index])
73.     #标准答案
74.     ground_truth = y_boston[test_index]
75.     print (mean_squared_error(ground_truth, pred))
76.
77. 优化超参数(参数选择)
78.
79. boston = load_boston()
80. y_boston = boston['target']
81. X_boston = boston['data']
82. xgb_model = xgb.XGBRegressor()
83. #参数字典
84. param_dict = {'max_depth':[2,4,6], 'n_estimators':[50, 100, 200]}
85.
86. rgs = GridSearchCV(xgb_model, param_dict)
87.
88. rgs.fit(X_boston, y_boston)
89.
90. print (rgs.best_score_)
91.
92. print (rgs.best_params_)
```

ElasticSearch大批量数据入库 (http://blog.csdn.net/qq_36330643/article/details/71750886)
1301

Error (http://blog.csdn.net/qq_36330643/article/details/71750886)
further parser pars
1108

版权声明： 作者原创，转载请附上文章链接。



相关文章推荐

利用sklearn训练LDA主题模型及调参详解 (<http://blog.csdn.net/TiffanyRabbit/article/details/71750886>)

sklearn不仅提供了机器学习基本的预处理、特征提取选择、分类聚类模型接口，还提供了很多常用语言模型的接口，LDA主题模型就是其中之一。本文除了介绍LDA模型的基本参数、调用训练以外，还将提供两种...

TiffanyRabbit (<http://blog.csdn.net/TiffanyRabbit>) 2017年07月31日 15:50 1162

xgboost参数说明，模型训练，模型预测java接口相关说明 (<http://blog.csdn.net/luoyexuge>)

xgboost参数说明在网上找了很多关于xgboost的文章，基本上90%都是以python在说明的，java的很少，xgboost参数说明<http://blog.csdn.net/zc02051>...

luoyexuge (<http://blog.csdn.net/luoyexuge>) 2017年05月07日 14:02 1162



不止20K，Python薪酬又飙升了？？

2017年 Python薪酬曝光啦！看完后薪资报告后，同事说了一句：人生苦短，不学Python算白活...

..


(http://www.baidu.com/cb.php?c=lgF_pyfqHmknjnPjc0lZ0qnfK9ujYzP1f4PjDs0Aw-5Hc3rHnYnHb0TAq15HfLPWRznb0T1dhP1KWj6YPy7bnW0srHb40AwY5HDdnHcsn1DdPHn0lgF_5y9YIZ0IQzq-uZR8mLPbUB48ugfEIAqspynETZ-YpAq8nWqdlAdxTvqdThP-5yF_UvTk0KzujY4rHb0mhYqn0KsTWYs0ZNGujYkPHTYn1mk0AqGujYknWb3rjDY0APGujYLnWm4n1c0ULI85H00TZbqnW0v0APzm1YvPjbkP0)

sklearn-pandas (http://download.csdn.net/detail/panda_aj/9494377)

http://download.csdn.net/detail/panda_aj/9494377 2016年04月18日 11:34 84KB [下载 \(1\)](#)

mxnet实战笔记(1) - 使用自己的图片数据训练CNN模型 (http://blog.csdn.net/storMouse/...

将自己的图片数据集导入mxnet训练一个cnn模型。

 storMouse (http://blog.csdn.net/storMouse) 2017年02月21日 01:22 00816

实战深度学习（一）——利用digits训练road模型 (http://blog.csdn.net/klqulei123/article...

参考：<http://www.cnblogs.com/denny402/p/5136155.html> <http://www.cnblogs.com/denny402/p/5136262.html> 第一步： ...

 klqulei123 (http://blog.csdn.net/klqulei123) 2016年10月10日 20:55 00872



一学就会的 **WordPress** 实战课

学习完本课程可以掌握基本的 **WordPress** 的开发能力，后续可以根据需要开发适合自己的主题、插件，打造最个性的 **WordPress** 站点。

(http://www.baidu.com/cb.php?c=lgF_pyfqhHmknjfvP1m0lZ0qnfK9ujYzP1f4Pjnz0Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1YYnvNWPj9bP1m4mhfkH6Y0AwY5HDdnHcsn1DdPHm0lgF_5y9YlZ0lQzqMpgwBUvqoQhP8QviGIAPCmgfEmvq_lyd8Q1N9nHmvnj7hnHPWnjFhPAD1Py5HDknWw9mhhEusKzujY4rHb0mhYqn0KsTWYs0ZNGujYkPHTYn1mk0AqGujYkn10snjf10APGujYLnWm4n1c0ULI85H00TZbqnW0v0APzm1Y1nHmkP6)


【神经网络与深度学习】深度学习实战——caffe windows 下训练自己的网络模型 (http...

1、相关准备 1.1 手写数字数据集这篇博客上有.jpg格式的图片下载，附带标签信息，有需要的自行下载，博客附带百度云盘下载地址(手写数字.jpg 格式)：<http://blog.csdn.net/LG1259156776>

 LG1259156776 (http://blog.csdn.net/LG1259156776) 2016年09月15日 10:00 001875


深度学习 Caffe实战笔记（21）Windows平台 Faster-RCNN 训练好的模型测试数据 (http...

前一篇博客介绍了如何利用Faster-RCNN训练自己的数据集，训练好会得到一个模型，这篇博客介绍如何利用训练好的模型进行测试数据。1、训练好的模型存放位置训练好的模型存放在faster_rcnn...

 gybheroin (http://blog.csdn.net/gybheroin) 2017年05月21日 19:32 001126


深度学习实战——caffe windows 下训练自己的网络模型 (http://blog.csdn.net/Eddy_zhe...

1、相关准备 1.1 手写数字数据集这篇博客上有.jpg格式的图片下载，附带标签信息，有需要的自行下载，博客附带百度云盘下载地址(手写数字.jpg 格式)：<http://blog.csdn.net/ed...>

 Eddy_zheng (http://blog.csdn.net/Eddy_zheng) 2016年03月19日 16:27 005085


Caffe实战：LeNet模型训练测试MNIST数据集 (http://blog.csdn.net/u010834458/article/d...

MNIST转成lmdb格式首先下载minist数据文件。此处我放的路径为E:\installpacket\编程软件\caffe源码\caffe-master\exampl...

 u010834458 (http://blog.csdn.net/u010834458) 2017年04月05日 23:30 00100


数据挖掘（pandas&xgboost） (http://blog.csdn.net/a_victory/article/details/72842051)

类别特征处理 1、利用pd.get_dummies方法将类别特征进行编码。使用get_dummies进行one-hot编码（查看时间字段的类型，如果不是datetime类型需要to_datetime转化...

 a_victory (http://blog.csdn.net/a_victory) 2017年06月02日 10:32 00245


利用python-pandas和sklearn进行天池移动推荐离线赛的全过程 (http://blog.csdn.net/L...

python, pandas, sklearn

 LY_ysys629 (http://blog.csdn.net/LY_ysys629) 2017年06月30日 22:23 001461


Python利用pandas/sklearn处理类别型特征；手动分段与离散化处理连续型特征 (http://...

类别型变量特征：独热向量编码/One-Hot-Encoding (Dummy variables) 颜色：红、黄、紫[1,0,0] [0,1,0] [0,0,1] LR=...

 cymy001 (http://blog.csdn.net/cymy001) 2017年11月19日 19:12 0099

数据预处理（1）——数据清洗 使用python（sklearn, pandas, numpy）实现 (http://...


数据预处理的主要任务有：一、数据预处理 1.数据清洗 2.数据集成 3.数据转换 4.数据归约 1.数据清洗现实世界的数据一般是不完整的、有噪声的和不一致的。数据清理例程试图填充缺失的值，光...

 daxiaofan (http://blog.csdn.net/daxiaofan) 2016年11月06日 18:55 003956



关于在Ubuntu下安装配置numpy,scipy,matplotlib,pandas 以及sklearn (http://blog.csdn.net/qq_36330643/article/details/78...)

关于在Ubuntu下安装配置numpy,scipy,matplotlib,pandas 以及sklearn最近在用Python学习机器学习，正好需要配置相关组件。也在网上查了一些，汇总了一下。顺便分...

 Yakumoyukarilan (http://blog.csdn.net/Yakumoyukarilan) 2016年05月07日 22:02 10377

数据预处理（2）——数据变换与数据离散化 使用python（sklearn，pandas，numpy...

数据预处理的主要任务有：一、数据预处理 1.数据清洗 2.数据集成 3.数据转换 4.数据归约 3.数据变换数据变换是指将数据转换或统一成适合于挖掘的形式。（1）数据泛化：...

 daxiaofan (http://blog.csdn.net/daxiaofan) 2016年11月09日 17:37 4200

数据预处理（3）——数据归约 使用python（sklearn，pandas，numpy）实现 (http://...)


数据预处理的主要任务有：一、数据预处理 1.数据清洗 2.数据集成 3.数据转换 4.数据归约 4.数据归约（1）数据立方体聚集：聚集操作作用于数据立方体结构中的数据。...


 daxiaofan (http://blog.csdn.net/daxiaofan) 2016年11月09日 17:46 1255

python pandas 基本使用，sklearn的10种分类器实践 (http://blog.csdn.net/ass7798/arti...)

由于最近代码任务不多，就想把pandas的日常使用和sklearn的建模使用更加规范和熟练，因此就以泰坦尼克的相关数据集为测试数据进行分析。选择了 glm tree svm ...

 ass7798 (http://blog.csdn.net/ass7798) 2017年11月09日 15:35 43

 pythont xgboost 速度快效果好的boosting模型 文档 (http://download.csdn.net/detail/sw...)

 2017年05月09日 20:13 810KB 下载 (1)

 Python实现xgboost模型的Demo (http://download.csdn.net/detail/sw...)

 2017年06月02日 09:47 3KB 下载 (1)

xgboost算法原理与实战 (http://blog.csdn.net/JasonZhangOO/article/details/73061060)

xgboost算法原理与实战之前一直有听说GBM，GBDT（Gradient Boost Decision Tree）渐进梯度决策树 GBRT（Gradient Boost RegressionTr...

 JasonZhangOO (http://blog.csdn.net/JasonZhangOO) 2017年06月11日 19:12 1174