

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

下载 (/download.csdn.net?ref=toolbar) GitChat (/gitbook.cn/?ref=csdn)

更多 ▾



0



weixin_3506... ▾ (/my.csdn.net?ref=toolbar)

(//write.blog.csdn.net/ueditor/activity?ref=toolbar)

source=csdnblog



MebiuW (http://blog.csdn...)

+ 关注

(http://blog.csdn.net/mebiuw)

码云

原创

粉丝

喜欢

未开通
(https://gite
utm_sourc

270

96

0

他的最新文章

更多文章 (http://blog.csdn.net/mebiuw)

Tensorflow 机器翻译NMT笔记 1 快速上手 (http://blog.csdn.net/MebiuW/article/details/77825642)

Leetcode 655. Print Binary Tree 打印二叉树 解题报告 (http://blog.csdn.net/MebiuW/article/details/77200704)

Leetcode 654. Maximum Binary Tree 最大二叉树 解题报告 (http://blog.csdn.net/MebiuW/article/details/77170643)

Tensorflow #3 使用DNN构造Iris分类器

原创

2016年11月18日 19:38:40

标签：机器学习 (http://so.csdn.net/so/search/s.do?q=机器学习&t=blog) /

神经网络 (http://so.csdn.net/so/search/s.do?q=神经网络&t=blog) /

iris (http://so.csdn.net/so/search/s.do?q=iris&t=blog) /

tensorlow (http://so.csdn.net/so/search/s.do?q=tensorlow&t=blog) /

分类 (http://so.csdn.net/so/search/s.do?q=分类&t=blog)

13053

1 前言

这篇文章主要是根据Tensorflow官方的API文档整理得到的。

本文的主要目的是学习使用Tensorflow提供的机器学习API，构建一个基于神经网络的分类器，对经典的Iris分类数据进行分类。这里的高级API主要是指代tf.contrib.learn里提供的API，

2 步骤

首先这里说一下基本步骤：

1. 从原始数据集的CSV里面读取数据，并且加载到Tensorflow当中
2. 构建一个基于神经网络的分类器
3. 使用训练数据进行模型训练



4. 使用测试数据进行模型评估
5. 使用训练好的模型对新的数据进行分类

2.1 数据加载

Iris这个数据集相信大家往上看的也多了，这里就不详细介绍了，只说明一些基本情况。
这个数据集包含150行数据，有三种不同的Iris品种。每一行数据出了品种的标注信息以外，还包含了一些参数如萼片的长度、宽度等信息。



在这里，已经提前将这些数据划分为训练集和测试集了，分别是120行和30行

A training set of 120 samples (iris_training.csv) (http://download.tensorflow.org/data/iris_training.csv)

A test set of 30 samples (iris_test.csv). (http://download.tensorflow.org/data/iris_test.csv).

准备好这些数据后，我们先完成第一部分代码，引入Tensorflow和加载数据

```
1 #coding:utf-8
2
3 from __future__ import absolute_import
4
5 from __future__ import division
6
7 from __future__ import print_function
8
```



在线课程



腾讯云容器服务架构实现介绍 ()
讲师：董晓杰



容器技术在58同城的实践
([http://edu.csdn.net/huiyi/cms/series_detail/73?](http://edu.csdn.net/huiyi/cms/series_detail/73?utm_source=blog9)
[http://edu.csdn.net/huiyi/cms/series_detail/73?](http://edu.csdn.net/huiyi/cms/series_detail/73?utm_source=blog9)
[utm_source=blog9](http://edu.csdn.net/huiyi/cms/series_detail/73?utm_source=blog9))

他的热门文章

Tensorflow #3 使用DNN构造Iris分类器 (<http://blog.csdn.net/mebiuw/article/details/53222000>)
13045

Python下的自然语言处理利器-LTP语言技术平台 pytp 学习手札 (<http://blog.csdn.net/mebiuw/article/details/52496920>)
11331

Python版的Word2Vec -- gensim 学习手札 中文词语相似性度量 V1.1 (<http://blog.csdn.net/mebiuw/article/details/52303622>)
9050

Tensorflow #2 深度学习-RNN LSTM版 MNIST手写识别Demo (<http://blog.csdn.net/mebiuw/article/details/52705731>)
8603

深度学习 (BOT方向) 学习笔记 (1) Sequence2Sequence 学习 (<http://blog.csdn.net/mebiuw/article/details/52222173>)



```
8
9
10
11 import tensorflow as tf
12
13 import numpy as np
14
15
16
17 # 设定数据集的位置
18
19 IRIS_TRAINING = "iris_training.csv"
20
21 IRIS_TEST = "iris_test.csv"
22
23
24
25 # 使用Tensorflow内置的方法进行数据加载
26 , target_type是最终的label的类型，这里只有012三个取值，所以用int
27 training_set = tf.contrib.learn.datasets.base.load_csv(filename=IRIS_TRAINING,
28
29                                     target_dtype=np.int)
30
31 test_set = tf.contrib.learn.datasets.base.load_csv(filename=IRIS_TEST,
32
33                                     target_dtype
34 e=np.int)
35
```

对于训练数据

和测试数据集，你都可以用.data的方式访问其特征，或者使用.target的方式访问其标签，这两部分在后面都会用到

2.2 创建一个深度神经网络分类器

tf.contrib.learn提供了很多已经实现的模型，这里被称作Estimators，通过这些已经定义好的模型，你可以快速的基于这些模型和你的数据做一些分析工作。在这里我们将会创建一个DNN分类器，这里用到了DNNClassifier。

```
1 # 每行数据4个特征，都是real-value的
2 feature_columns = [tf.contrib.layers.real_valued_column("", dimension=4)]
3
```

net/mebiuw/article/details/5283284 /)

8017

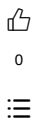
相关推荐

深度学习tensorflow教程-DNNClassifier (http://blog.csdn.net/zhangshuaizaxia/article/details/53560895)

tensorflow学习笔记十四：TF官方教程学习 tf.contrib.learn Quickstart (http://blog.csdn.net/xiaopihaierletian/article/details/62423199)

用神经网络对Iris数据集进行分类的程序--tensorflow (http://blog.csdn.net/youyuyixiu/article/details/71422130)

Python与自然语言处理（三）：Tensorflow基础学习 (http://blog.csdn.net/monkey131499/article/details/52145254)





```
4 # 构建一个DNN分类器, 3层, 其中每个隐含层的节点数量分别为10, 20, 10, 目标的分类3个, 并且指定了保存位置
5 classifier = tf.contrib.learn.DNNClassifier(feature_columns=feature_columns,
6                                           hidden_units=[10, 20, 10],
7                                           n_classes=3,
8                                           model_dir="/tmp/iris_model")
```

其中我们这里需要生命每一行（即对应的每一个样本）的特征取值，这里都使用了real-value去做
在构造DNNClassifier时候，我们构建了三层，可以直接通过hidden_units指定具体层数和每层的取值
n_classes那个地方表明了我们是3个分类问题
而在model_dir位置，是一个目录，Tensorflow可以保存一些Checkpoint，这样Tensorflow就可以随时保存训练情况了（还可以暂停后继续训练）

2.3 训练模型

这里我们直接使用fit方法将数据填入到模型中，并且进行训练

```
1 # 指定数据，以及训练的步数
2 classifier.fit(x=training_set.data,
3               y=training_set.target,
4               steps=2000)
```

2.4 模型评估

在这里我们需要通过请假期准确度的方式来衡量其性能，这里也是直接给出测试数据集，然后调用他的方法就可以了

最终我们会看到一个大约96%的结果

```
1 accuracy_score = classifier.evaluate(x=test_set.data,
2                                     y=test_set.target)["accuracy"]
3 print('Accuracy: {0:f}'.format(accuracy_score))
```

2.5 用来预测新的数据集

我们训练完成模型后，就可以使用这个模型来进行一定的数据预测了



0





```
1 # 直接创建数据来进行预测
2 new_samples = np.array(
3     [[6.4, 3.2, 4.5, 1.5], [5.8, 3.1, 5.0, 1.7]], dtype=float)
4 y = classifier.predict(new_samples)
5 print('Predictions: {}'.format(str(y)))
```

3 完整代码



0



```
1 #coding:utf-8
2 from __future__ import absolute_import
3 from __future__ import division
4 from __future__ import print_function
5
6 import tensorflow as tf
7 import numpy as np
8
9 # 设定数据集的位置
10 IRIS_TRAINING = "iris_training.csv"
11 IRIS_TEST = "iris_test.csv"
12
13 # 使用Tensorflow内置的方法进行数据加载
14 training_set = tf.contrib.learn.datasets.base.load_csv(filename=IRIS_TRAINING,
15                                                         target_dtype=np.int)
16 test_set = tf.contrib.learn.datasets.base.load_csv(filename=IRIS_TEST,
17                                                     target_dtype=np.int)
18
19 # 每行数据4个特征，都是real-value的
```

```
20 feature_columns = [tf.contrib.layers.real_valued_column("", dimension=4)]
21
22 # 构建一个DNN分类器，3层，其中每个隐含层的节点数量分别为10，20，10，目标的分类3个，并且指定了保存位置
23 classifier = tf.contrib.learn.DNNClassifier(feature_columns=feature_columns,
24                                           hidden_units=[10, 20, 10],
25                                           n_classes=3,
26                                           model_dir="/tmp/iris_model")
27
28 # 指定数据，以及训练的步数
29 classifier.fit(x=training_set.data,
30              y=training_set.target,
31              steps=2000)
32
33 # 模型评估
34 accuracy_score = classifier.evaluate(x=test_set.data,
35                                     y=test_set.target)["accuracy"]
36 print('Accuracy: {0:f}'.format(accuracy_score))
37
38 # 直接创建数据来进行预测
39 new_samples = np.array(
40     [[6.4, 3.2, 4.5, 1.5], [5.8, 3.1, 5.0, 1.7]], dtype=float)
41 y = classifier.predict(new_samples)
42 print('Predictions: {}'.format(str(y)))
43
```



0



版权声明：本文为博主原创文章，未经博主允许不得转载。



发表你的评论

(http://my.csdn.net/weixin_35068028)

相关文章推荐


深度学习tensorflow教程-DNNClassifier (<http://blog.csdn.net/zhangshuaizaxia/article/detai...>)

支持原文 <http://shuaizhang.tech/2016/12/10/Tensorflow-Tutorial-4-DNNClassifier/DNNClassifier>从前面的例子可以看出，...

zhangshuaizaxia (<http://blog.csdn.net/zhangshuaizaxia>) 2016年12月10日 20:47 4483

tensorflow学习笔记十四：TF官方教程学习 tf.contrib.learn Quickstart (<http://blog.csdn.net/youyuyixiu>...

TensorFlow高级API(tf.contrib.learn)及可视化工具TensorBoard的使用 一.TensorFlow高层次机器学习API (tf.contrib.learn)...

 xiaopihaierletian (<http://blog.csdn.net/xiaopihaierletian>) 2017年03月16日 14:55 6514



霸气！2017薪资报告：AI工程师荣登技术薪酬NO.1！


不多说！一张图带你看懂AI工程师薪酬比例！听项目经理说AI应届生开价20-50万，都没招上人来...

广告

(http://www.baidu.com/cb.php?c=lgF_pyfqHmknjnvPjn0IZ0qnfK9ujYzP1ndPWb10Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1YkPH6Yn1TvrjzrH61rHNW0AwY5HDdnHc1njmsPHn0lgF_5y9YIZ0IQzq-uzR8mLPbUB48ugfEIAqspynEmybz5LNYUNq1ULNzmvRqmhkEu1Ds0ZFb5HD0mhYqn0KsTWYs0ZNGUjYkPHTYn1mk0AqGujYknWb3rjDY0APGujYLnWm4n1c0ULI85H00TZbqnW0v0APzm1Ydrj6vns)


用神经网络对Iris数据集进行分类的程序--tensorflow (<http://blog.csdn.net/youyuyixiu>...

先确保你已经安装了tensorflow...# 引入必要的module from __future__ import absolute_import from __future__ import divi...

 youyuyixiu (<http://blog.csdn.net/youyuyixiu>) 2017年05月08日 15:58 1270


Python与自然语言处理（三）：Tensorflow基础学习 (<http://blog.csdn.net/monkey131499>...

TensorFlow实现简单神经网络，并对iris花的种类进行预测

 monkey131499 (<http://blog.csdn.net/monkey131499>) 2016年08月08日 21:34 4343

用DNN对Iris数据分类的代码--tensorflow--logging/monitoring/earlystopping/visualizing (ht...

本博客是对 用神经网络对Iris数据集进行分类的程序--tensorflow 里面的代码进行修改,使其可以记录训练日志,监控训练指标,并在TensorBoard中进行可视化.from __fut...

 youyuyixiu (<http://blog.csdn.net/youyuyixiu>) 2017年05月09日 16:23 977



程序员跨越式成长指南

完成第一次跨越，你会成为具有一技之长的开发者，月薪可能翻上几番；完成第二次跨越，你将成为拥有局部优势或行业优势的专业人士，获得个人内在价值的有效提升和外在收入的大幅跃迁.....

(http://www.baidu.com/cb.php?c=lgF_pyfqHmknjzrjD0IZ0qnfK9ujYzP1f4PjnY0Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1dhnbLmWb4nADLP1nYryck0AwY5HDdnHc1njmsPHn0lgF_5y9YIZ0IQzqMpgwBUvqoQhP8QvIGIAPCmgfEmvq_lyd8Q1R4uWc4uHf3uAckPHRkPWN9PhcsmW9huWqdlAdxTv5HDknWFBmhkEusKzujYk0AFV5H00TZcqn0KdpyfqHRLPjnvnfKEpyfqHnsnj0YnsKWpyfqP1civrHnz0AqLUWYs0ZK45HcsP6KWThnqPhn3nj6)

温度当... 1... 2... 3... 4... 5... 6... 7... 8... 9... 10... 11... 12... 13... 14... 15... 16... 17... 18... 19... 20... 21... 22... 23... 24... 25... 26... 27... 28... 29... 30... 31... 32... 33... 34... 35... 36... 37... 38... 39... 40... 41... 42... 43... 44... 45... 46... 47... 48... 49... 50... 51... 52... 53... 54... 55... 56... 57... 58... 59... 60... 61... 62... 63... 64... 65... 66... 67... 68... 69... 70... 71... 72... 73... 74... 75... 76... 77... 78... 79... 80... 81... 82... 83... 84... 85... 86... 87... 88... 89... 90... 91... 92... 93... 94... 95... 96... 97... 98... 99... 100... 101... 102... 103... 104... 105... 106... 107... 108... 109... 110... 111... 112... 113... 114... 115... 116... 117... 118... 119... 120... 121... 122... 123... 124... 125... 126... 127... 128... 129... 130... 131... 132... 133... 134... 135... 136... 137... 138... 139... 140... 141... 142... 143... 144... 145... 146... 147... 148... 149... 150... 151... 152... 153... 154... 155... 156... 157... 158... 159... 160... 161... 162... 163... 164... 165... 166... 167... 168... 169... 170... 171... 172... 173... 174... 175... 176... 177... 178... 179... 180... 181... 182... 183... 184... 185... 186... 187... 188... 189... 190... 191... 192... 193... 194... 195... 196... 197... 198... 199... 200... 201... 202... 203... 204... 205... 206... 207... 208... 209... 210... 211... 212... 213... 214... 215... 216... 217... 218... 219... 220... 221... 222... 223... 224... 225... 226... 227... 228... 229... 230... 231... 232... 233... 234... 235... 236... 237... 238... 239... 240... 241... 242... 243... 244... 245... 246... 247... 248... 249... 250... 251... 252... 253... 254... 255... 256... 257... 258... 259... 260... 261... 262... 263... 264... 265... 266... 267... 268... 269... 270... 271... 272... 273... 274... 275... 276... 277... 278... 279... 280... 281... 282... 283... 284... 285... 286... 287... 288... 289... 290... 291... 292... 293... 294... 295... 296... 297... 298... 299... 300... 301... 302... 303... 304... 305... 306... 307... 308... 309... 310... 311... 312... 313... 314... 315... 316... 317... 318... 319... 320... 321... 322... 323... 324... 325... 326... 327... 328... 329... 330... 331... 332... 333... 334... 335... 336... 337... 338... 339... 340... 341... 342... 343... 344... 345... 346... 347... 348... 349... 350... 351... 352... 353... 354... 355... 356... 357... 358... 359... 360... 361... 362... 363... 364... 365... 366... 367... 368... 369... 370... 371... 372... 373... 374... 375... 376... 377... 378... 379... 380... 381... 382... 383... 384... 385... 386... 387... 388... 389... 390... 391... 392... 393... 394... 395... 396... 397... 398... 399... 400... 401... 402... 403... 404... 405... 406... 407... 408... 409... 410... 411... 412... 413... 414... 415... 416... 417... 418... 419... 420... 421... 422... 423... 424... 425... 426... 427... 428... 429... 430... 431... 432... 433... 434... 435... 436... 437... 438... 439... 440... 441... 442... 443... 444... 445... 446... 447... 448... 449... 450... 451... 452... 453... 454... 455... 456... 457... 458... 459... 460... 461... 462... 463... 464... 465... 466... 467... 468... 469... 470... 471... 472... 473... 474... 475... 476... 477... 478... 479... 480... 481... 482... 483... 484... 485... 486... 487... 488... 489... 490... 491... 492... 493... 494... 495... 496... 497... 498... 499... 500... 501... 502... 503... 504... 505... 506... 507... 508... 509... 510... 511... 512... 513... 514... 515... 516... 517... 518... 519... 520... 521... 522... 523... 524... 525... 526... 527... 528... 529... 530... 531... 532... 533... 534... 535... 536... 537... 538... 539... 540... 541... 542... 543... 544... 545... 546... 547... 548... 549... 550... 551... 552... 553... 554... 555... 556... 557... 558... 559... 560... 561... 562... 563... 564... 565... 566... 567... 568... 569... 570... 571... 572... 573... 574... 575... 576... 577... 578... 579... 580... 581... 582... 583... 584... 585... 586... 587... 588... 589... 590... 591... 592... 593... 594... 595... 596... 597... 598... 599... 600... 601... 602... 603... 604... 605... 606... 607... 608... 609... 610... 611... 612... 613... 614... 615... 616... 617... 618... 619... 620... 621... 622... 623... 624... 625... 626... 627... 628... 629... 630... 631... 632... 633... 634... 635... 636... 637... 638... 639... 640... 641... 642... 643... 644... 645... 646... 647... 648... 649... 650... 651... 652... 653... 654... 655... 656... 657... 658... 659... 660... 661... 662... 663... 664... 665... 666... 667... 668... 669... 670... 671... 672... 673... 674... 675... 676... 677... 678... 679... 680... 681... 682... 683... 684... 685... 686... 687... 688... 689... 690... 691... 692... 693... 694... 695... 696... 697... 698... 699... 700... 701... 702... 703... 704... 705... 706... 707... 708... 709... 710... 711... 712... 713... 714... 715... 716... 717... 718... 719... 720... 721... 722... 723... 724... 725... 726... 727... 728... 729... 730... 731... 732... 733... 734... 735... 736... 737... 738... 739... 740... 741... 742... 743... 744... 745... 746... 747... 748... 749... 750... 751... 752... 753... 754... 755... 756... 757... 758... 759... 760... 761... 762... 763... 764... 765... 766... 767... 768... 769... 770... 771... 772... 773... 774... 775... 776... 777... 778... 779... 780... 781... 782... 783... 784... 785... 786... 787... 788... 789... 790... 791... 792... 793... 794... 795... 796... 797... 798... 799... 800... 801... 802... 803... 804... 805... 806... 807... 808... 809... 810... 811... 812... 813... 814... 815... 816... 817... 818... 819... 820... 821... 822... 823... 824... 825... 826... 827... 828... 829... 830... 831... 832... 833... 834... 835... 836... 837... 838... 839... 840... 841... 842... 843... 844... 845... 846... 847... 848... 849... 850... 851... 852... 853... 854... 855... 856... 857... 858... 859... 860... 861... 862... 863... 864... 865... 866... 867... 868... 869... 870... 871... 872... 873... 874... 875... 876... 877... 878... 879... 880... 881... 882... 883... 884... 885... 886... 887... 888... 889... 890... 891... 892... 893... 894... 895... 896... 897... 898... 899... 900... 901... 902... 903... 904... 905... 906... 907... 908... 909... 910... 911... 912... 913... 914... 915... 916... 917... 918... 919... 920... 921... 922... 923... 924... 925... 926... 927... 928... 929... 930... 931... 932... 933... 934... 935... 936... 937... 938... 939... 940... 941... 942... 943... 944... 945... 946... 947... 948... 949... 950... 951... 952... 953... 954... 955... 956... 957... 958... 959... 960... 961... 962... 963... 964... 965... 966... 967... 968... 969... 970... 971... 972... 973... 974... 975... 976... 977... 978... 979... 980... 981... 982... 983... 984... 985... 986... 987... 988... 989... 990... 991... 992... 993... 994... 995... 996... 997... 998... 999... 1000... 1001... 1002... 1003... 1004... 1005... 1006... 1007... 1008... 1009... 1010... 1011... 1012... 1013... 1014... 1015... 1016... 1017... 1018... 1019... 1020... 1021... 1022... 1023... 1024... 1025... 1026... 1027... 1028... 1029... 1030... 1031... 1032... 1033... 1034... 1035... 1036... 1037... 1038... 1039... 1040... 1041... 1042... 1043... 1044... 1045... 1046... 1047... 1048... 1049... 1050... 1051... 1052... 1053... 1054... 1055... 1056... 1057... 1058... 1059... 1060... 1061... 1062... 1063... 1064... 1065... 1066... 1067... 1068... 1069... 1070... 1071... 1072... 1073... 1074... 1075... 1076... 1077... 1078... 1079... 1080... 1081... 1082... 1083... 1084... 1085... 1086... 1087... 1088... 1089... 1090... 1091... 1092... 1093... 1094... 1095... 1096... 1097... 1098... 1099... 1100... 1101... 1102... 1103... 1104... 1105... 1106... 1107... 1108... 1109... 1110... 1111... 1112... 1113... 1114... 1115... 1116... 1117... 1118... 1119... 1120... 1121... 1122... 1123... 1124... 1125... 1126... 1127... 1128... 1129... 1130... 1131... 1132... 1133... 1134... 1135... 1136... 1137... 1138... 1139... 1140... 1141... 1142... 1143... 1144... 1145... 1146... 1147... 1148... 1149... 1150... 1151... 1152... 1153... 1154... 1155... 1156... 1157... 1158... 1159... 1160... 1161... 1162... 1163... 1164... 1165... 1166... 1167... 1168... 1169... 1170... 1171... 1172... 1173... 1174... 1175... 1176... 1177... 1178... 1179... 1180... 1181... 1182... 1183... 1184... 1185... 1186... 1187... 1188... 1189... 1190... 1191... 1192... 1193... 1194... 1195... 1196... 1197... 1198... 1199... 1200... 1201... 1202... 1203... 1204... 1205... 1206... 1207... 1208... 1209... 1210... 1211... 1212... 1213... 1214... 1215... 1216... 1217... 1218... 1219... 1220... 1221... 1222... 1223... 1224... 1225... 1226... 1227... 1228... 1229... 1230... 1231... 1232... 1233... 1234... 1235... 1236... 1237... 1238... 1239... 1240... 1241... 1242... 1243... 1244... 1245... 1246... 1247... 1248... 1249... 1250... 1251... 1252... 1253... 1254... 1255... 1256... 1257... 1258... 1259... 1260... 1261... 1262... 1263... 1264... 1265... 1266... 1267... 1268... 1269... 1270... 1271... 1272... 1273... 1274... 1275... 1276... 1277... 1278... 1279... 1280... 1281... 1282... 1283... 1284... 1285... 1286... 1287... 1288... 1289... 1290... 1291... 1292... 1293... 1294... 1295... 1296... 1297... 1298... 1299... 1300... 1301... 1302... 1303... 1304... 1305... 1306... 1307... 1308... 1309... 1310... 1311... 1312... 1313... 1314... 1315... 1316... 1317... 1318... 1319... 1320... 1321... 1322... 1323... 1324... 1325... 1326... 1327... 1328... 1329... 1330... 1331... 1332... 1333... 1334... 1335... 1336... 1337... 1338... 1339... 1340... 1341... 1342... 1343... 1344... 1345... 1346... 1347... 1348... 1349... 1350... 1351... 1352... 1353... 1354... 1355... 1356... 1357... 1358... 1359... 1360... 1361... 1362... 1363... 1364... 1365... 1366... 1367... 1368... 1369... 1370... 1371... 1372... 1373... 1374... 1375... 1376... 1377... 1378... 1379... 1380... 1381... 1382... 1383... 1384... 1385... 1386... 1387... 1388... 1389... 1390... 1391... 1392... 1393... 1394... 1395... 1396... 1397... 1398... 1399... 1400... 1401... 1402... 1403... 1404... 1405... 1406... 1407... 1408... 1409... 1410... 1411... 1412... 1413... 1414... 1415... 1416... 1417... 1418... 1419... 1420... 1421... 1422... 1423... 1424... 1425... 1426... 1427... 1428... 1429... 1430... 1431... 1432... 1433... 1434... 1435... 1436... 1437... 1438... 1439... 1440... 1441... 1442... 1443... 1444... 1445... 1446... 1447... 1448... 1449... 1450... 1451... 1452... 1453... 1454... 1455... 1456... 1457... 1458... 1459... 1460... 1461... 1462... 1463... 1464... 1465... 1466... 1467... 1468... 1469... 1470... 1471... 1472... 1473... 1474... 1475... 1476... 1477... 1478... 1479... 1480... 1481... 1482... 1483... 1484... 1485... 1486... 1487... 1488... 1489... 1490... 1491... 1492... 1493... 1494... 1495... 1496... 1497... 1498... 1499... 1500... 1501... 1502... 1503... 1504... 1505... 1506... 1507... 1508... 1509... 1510... 1511... 1512... 1513... 1514... 1515... 1516... 1517... 1518... 1519... 1520... 1521... 1522... 1523... 1524... 1525... 1526... 1527... 1528... 1529... 1530... 1531... 1532... 1533... 1534... 1535... 1536... 1537... 1538... 1539... 1540... 1541... 1542... 1543... 1544... 1545... 1546... 1547... 1548... 1549... 1550... 1551... 1552... 1553... 1554... 1555... 1556... 1557... 1558... 1559... 1560... 1561... 1562... 1563... 1564... 1565... 1566... 1567... 1568... 1569... 1570... 1571... 1572... 1573... 1574... 1575... 1576... 1577... 1578... 1579... 1580... 1581... 1582... 1583... 1584... 1585... 1586... 1587... 1588... 1589... 1590... 1591... 1592... 1593... 1594... 1595... 1596... 1597... 1598... 1599... 1600... 1601... 1602... 1603... 1604... 1605... 1606... 1607... 1608... 1609... 1610... 1611... 1612... 1613... 1614... 1615... 1616... 1617... 1618... 1619... 1620... 1621... 1622... 1623... 1624... 1625... 1626... 1627... 1628... 1629... 1630... 1631... 1632... 1633... 1634... 1635... 1636... 1637... 1638... 1639... 1640... 1641... 1642... 1643... 1644... 1645... 1646... 1647... 1648... 1649... 1650... 1651... 1652... 1653... 1654... 1655... 1656... 1657... 1658... 1659... 1660... 1661... 1662... 1663... 1664... 1665... 1666... 1667... 1668... 1669... 1670... 1671... 1672... 1673... 1674... 1675... 1676... 1677... 1678... 1679... 1680... 1681... 1682... 1683... 1684... 1685... 1686... 1687... 1688... 1689... 1690... 1691... 1692... 1693... 1694... 1695... 1696... 1697... 1698... 1699... 1700... 1701... 1702... 1703... 1704... 1705... 1706... 1707... 1708... 1709... 1710... 1711... 1712... 1713... 1714... 1715... 1716... 1717... 1718... 1719... 1720... 1721... 1722... 1723... 1724... 1725... 1726... 1727... 1728... 1729... 1730... 1731... 1732... 1733... 1734... 1735... 1736... 1737... 1738... 1739... 1740... 1741... 1742... 1743... 1744... 1745... 1746... 1747... 1748... 1749... 1750... 1751... 1752... 1753... 1754... 1755... 1756... 1757... 1758... 1759... 1760... 1761... 1762... 1763... 1764... 1765... 1766... 1767... 1768... 1769... 1770... 1771... 1772... 1773... 1774... 1775... 1776... 1777... 1778... 1779... 1780... 1781... 1782... 1783... 1784... 1785... 1786... 1787... 1788... 1789... 1790... 1791... 1792... 1793... 1794... 1795... 1796... 1797... 1798... 1799... 1800... 1801... 1802... 1803... 1804... 1805... 1806... 1807... 1808... 1809... 1810... 1811... 1812... 1813... 1814... 1815... 1816... 1817... 1818... 1819... 1820... 1821

深度学习(deep learning)之一步一步实现编写与深度神经网络 (DNN) (<http://blog.csdn.net/tand...>)

本文将以matlab为工具介绍下如何实现深度神经网络，以方便初学者，神经元激活函数为 ReLU。 ReLU函数的数学公式很简单 $\text{ReLU}(x)=\max(x,0)$ ，但其对DNN的贡献是巨大的。 若DN...

 tandky (<http://blog.csdn.net/tandyky>) 2015年03月10日 23:19  18593

深度学习之 (DNN) 深度神经网络 (<http://blog.csdn.net/lihaitao000/article/details/51308009>)

(DNN) 深度神经网络 简介 DNN是指深度神经网络。与RNN循环神经网络、CNN卷积神经网络的区别就是DNN特指全连接的神经元结构，并不包含卷积单元或是时间上的关联。 神经网络简史...

 lihaitao000 (<http://blog.csdn.net/lihaitao000>) 2016年05月03日 21:42  13301





Delphi7高级应用开发随书源码 (<http://download.csdn.net/detail/chenxh/3>)

(<http://download.csdn.net/detail/chenxh/3>) 2003年04月30日 00:00 676KB [下载](#)

Tensorflow:softmax处理Iris鸢尾花分类 (<http://blog.csdn.net/jmh1996/article/details/77799...>)

经典的分类模型，鸢尾花的分类。数据集样例：length sepal width petal length petal width class 5.1 3.5 1.4...

 jmh1996 (<http://blog.csdn.net/jmh1996>) 2017年09月02日 11:13  230



Delphi7高级应用开发随书源码 (<http://download.csdn.net/detail/chenxh/3>)

(<http://download.csdn.net/detail/chenxh/3>) 2003年04月30日 00:00 676KB [下载](#)



TensorFlow-4: tf.contrib.learn 快速入门 (<http://blog.csdn.net/aliceyangxi1987/article/deta...>)

学习资料： https://www.tensorflow.org/get_started/tflearn

 aliceyangxi1987 (<http://blog.csdn.net/aliceyangxi1987>) 2017年04月26日 10:39  7650



05：Tensorflow高级API的进阶--利用tf.contrib.learn建立输入函数 (http://blog.csdn.net/sinat_33761963)

标签（空格分隔）： 王小草Tensorflow笔记笔记整理者：王小草 笔记整理时间：2017年2月27日 笔记对应的官方文档：http://www.tensorflow.org/get_star...

 sinat_33761963 (http://blog.csdn.net/sinat_33761963) 2017年03月16日 18:44  3528



TensorFlow 使用之 Logistics Regression (<http://blog.csdn.net/willduan1/article/details/53...>)

这篇文章主要讲使用 TensorFlow 进行 Logistics Regression 训练。 本文要解决的问题是使用逻辑回归解决一个二分类问题，即给定了人口普查的数据如人的年龄，性别，教育程度和职...

 willduan1 (<http://blog.csdn.net/willduan1>) 2016年11月09日 21:00  12704


tensorflow之tf.contrib.learn Quickstart (<http://blog.csdn.net/CZ626626/article/details/701...>)

一、载入必要的模块from __future__ import absolute_import from __future__ import division from __future__ impo...

 CZ626626 (<http://blog.csdn.net/CZ626626>) 2017年04月16日 22:32  490



TensorFlow学习笔记[1]: TensorFlow安装及在Jupyter notebook中用Pythond代码并生成散...

TensorFlow学习笔记[1]: TensorFlow安装及在Jupyter notebook中用Pythond代码并生成散列点图 作者：雨水/家辉，日期：2016-10-15，CSDN博客：ht...

 gobitan (<http://blog.csdn.net/gobitan>) 2016年10月15日 23:13  6922



caffe 跑mnist 数据集 (<http://blog.csdn.net/lijianqing/article/details/53321551>)

1.下载数据集：可以通过脚本下载也可以直接到官网下载了放到data/mnist下，然后解压 2.设置用cpu训练 lenet_solver.prototxt: # solver mode: CP...

 lijianqing (<http://blog.csdn.net/lijianqing>) 2016年11月24日 16:15  3164

Tensorflow学习笔记--使用迁移学习做自己的图像分类器（Inception v3）(<http://blog.csdn.n...>)

本文主要使用inception v3的模型，再后面接一个softmax，做一个分类器。具体代码都是参照tf github。 整体步骤： 步骤一：数据准备，准备自己要分类的图片训练样本。 ...

 l18930738887 (<http://blog.csdn.net/l18930738887>) 2017年05月30日 22:50  4230



距离平均最近邻分类器 iris (http://download.csdn.net/detail/lantian_66/213...)

(http://download.csdn.net/detail/lantian_66/213...) 2010年03月16日 17:14 2KB [下载](#)




K-最近邻分类器 iris数据 (http://download.csdn.net/detail/lantian_66/2133...)

(http://download.csdn.net/detail/mebiuw/53222000) 2010年03月16日 17:13 2KB 下载

数据挖掘-K-近邻分类器-Iris数据集分析-根据花萼长宽分类-以散点图显示(一) (http://blog.csdn.net/zhangyingchengqi)

%load python_8_1.py 此命令以加载本脚本到到ipython中 # coding: utf-8 from sklearn import datasets iris=...

 zhangyingchengqi (http://blog.csdn.net/zhangyingchengqi) 2017年01月29日 22:29 247



0

