增量训练全连接神经网络模型

运行环境：python3 Tensorflow

依赖： numpy

压缩包密码：e752s38

# 准备自己的训练数据集

一个回归或者分类器，可以抽象为N输入、K输出的黑箱。需要给定一个训练集文件，文本格式。

每一行是一个样本，含有N+K个浮点元素，逗号分割，回车换行。前N个元素为特征值，后K个为预期的结果值。越多越好。

训练时，网络不关心这些值的意义，只尽可能的调整参数，使得输出值与期望值之间的差异最小。这个文件存为“train\_input.txt”，与程序位于同一文件夹。

# 指定网络规模

新建或者修改配置文件“train\_task.cfg”。与程序位于同一文件夹。

#1.网络描述

[network]

#1.1输入节点（特征向量元素）数

input\_nodes=9

#1.2隐层规模,逗号分隔字符串

hidden\_layer\_size="17,23,23,16"

#1.3输出节点（判决向量元素）数

output\_nodes=3

#1.4正则化参数，防止权值发散

lamda=0.000001

#2.性能参数

[performance]

#2.1.单个文件读取次数

file\_deal\_times = 2

#2.1.trunk是每次用于训练的最大样本数

trunk=2048

#2.2.iterate\_times是每次训练迭代的次数

iterate\_times=256

#2.3.train\_step 是每轮训练样本集窗口向下移动的行数

train\_step = 1024

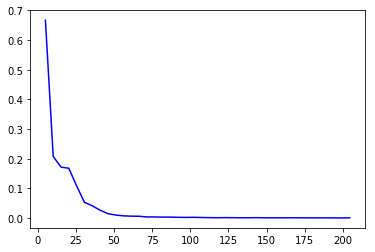
这个文件【network】块应该在一次应用中保持不变。

# 训练、测试与应用

## Spyder IDE模式

### 首次训练

运行gn\_first\_training.py，会产生模型文件夹saved\_model。同时，交互环境会输出学习曲线（代价函数曲线）



### 增量训练

尽可能找到更多样本，并更新训练数据集“train\_input.txt”，与程序位于同一文件夹。

运行gn\_next\_training.py，会继续上次的进展，训练模型文件夹saved\_model

### 使用测试集合(可选)

在收集更多样本的同时，可以匀出来少许样本，存入测试数据集合“test\_set.txt”，与程序位于同一文件夹。

运行gn\_test\_model.py，会输出测试集合的代价函数返回值（越小越好）。

### 应用训练结果

把实际的数据（N个特征）存成任务文件task.txt，每一行是一个样本，含有N个浮点元素，逗号分割，回车换行。这个文件存到程序同一文件夹。

运行gn\_run\_model.py，会产生结果集合result.txt.前N个为原始输入，后K个为计算输出。

## 命令行模式

### 首次训练

运行cmd\_first\_training.py

(C:\Anaconda) C:\Anaconda\projects\general\_nn>**python cmd\_first\_trainning.py -t 任务配置.cfg -i 训练样本.txt -m 模型文件夹**

### 增量训练

运行cmd\_next\_training.py，会继续上次的进展

(C:\Anaconda) C:\Anaconda\projects\general\_nn>**python cmd\_next\_trainning.py -t 任务配置.cfg -i 训练样本.txt -m 模型文件夹**

### 使用测试集合(可选)

运行cmd\_test\_model.py，会输出测试集合的代价函数返回值（越小越好）。

(C:\Anaconda) C:\Anaconda\projects\general\_nn>**python cmd\_test\_model.py -t 任务配置.cfg -i 测试样本.txt -m 模型文件夹**

### 应用训练结果

把实际的数据（N个特征）存成任务文件，每一行是一个样本，含有N个浮点元素，逗号分割，回车换行。这个文件存到程序同一文件夹。

运行cmd\_run\_model.py，会产生结果集合result.txt.前N个为原始输入，后K个为计算输出。

(C:\Anaconda) C:\Anaconda\projects\general\_nn>**python cmd\_run\_model.py -t 任务配置.cfg -i 特征集合.txt -m 模型文件夹 -o 输出文件.txt**

# Tensorboard 可视化与参数导出

## 使用Tensorboard查看训练摘要

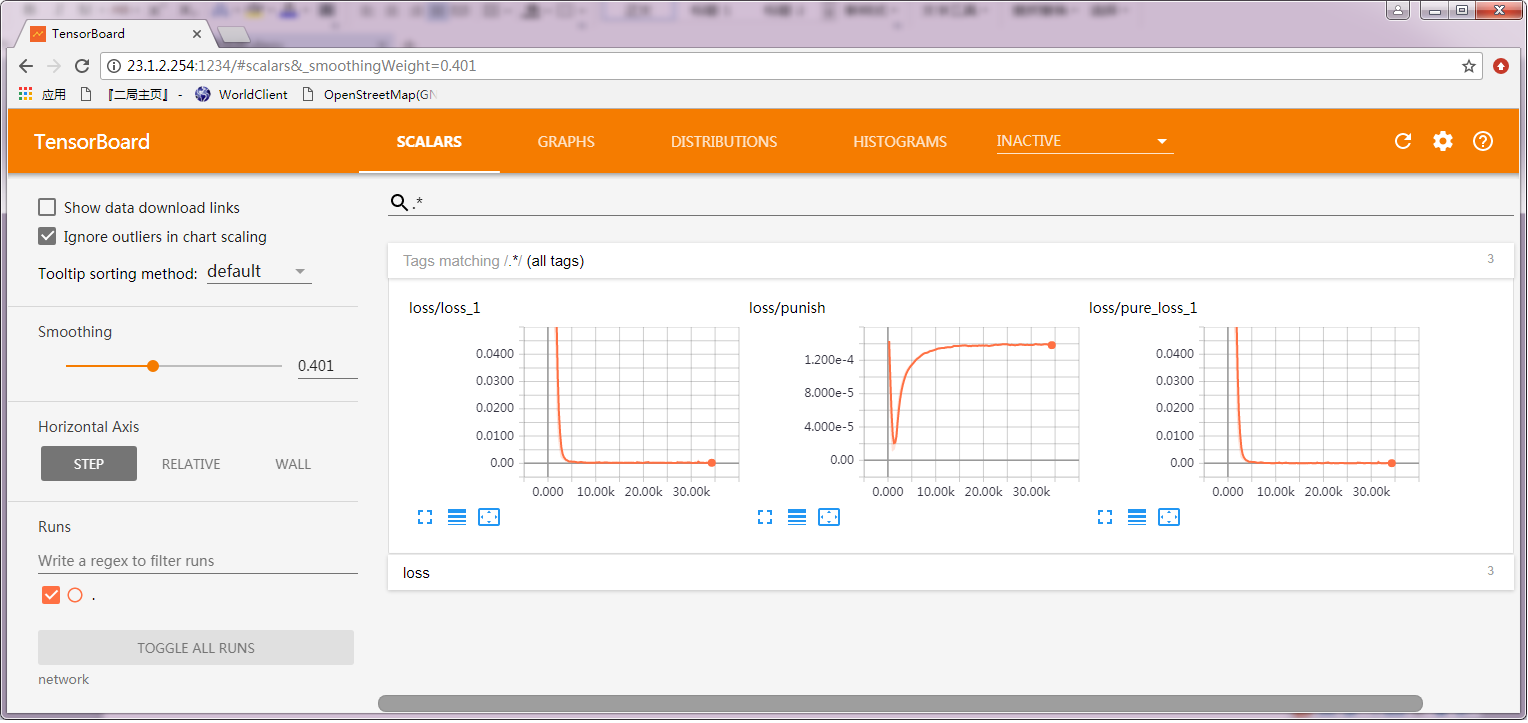
首次训练模型前，请清空可视化摘要文件夹。默认的摘要文件夹位于 network 下。

一旦训练完毕，network中会有可视化信息。

(C:\Anaconda) C:\Anaconda\projects\general\_nn>**tensorboard --port 1234 --logdir n**

**etwork**

用Chrome浏览器（新版，IE不行）打开地址:



## 获得网络的取值

运行

import gnn\_methods as gm

trainning\_task\_file = 'train\_task.cfg'

model\_path = './saved\_model/'

dp\_all = gm.dump\_all(trainning\_task\_file,model\_path)

所有网络取值都会存入dp\_all

# 范例数据

example\_gen\_trdata.py 产生一个二维平面上3区域分类的数据，用于演示。

