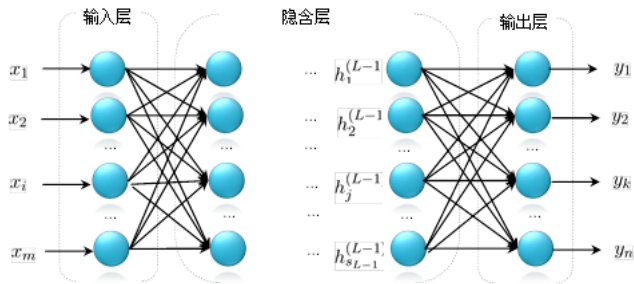


BP神经网络推导过程详解

BP算法是一种最有效的多层神经网络学习方法，其主要特点是信号前向传递，而误差后向传播，通过不断调节网络权重值，使得网络的最终输出与期望输出尽可能接近，以达到训练的目的。

一、多层神经网络结构及其描述

下图为一典型的多层神经网络。



通常一个多层神经网络由L层神经元组成，其中：第1层称为输入层，最后一层（第L层）被称为输出层，其它各层均被称为隐含层（第2层~第L-1层）。

令输入向量为：

$$\vec{x} = [x_1 \quad x_2 \quad \dots \quad x_i \quad \dots \quad x_m], i = 1, 2, \dots, m$$

输出向量为：

$$\vec{y} = [y_1 \quad y_2 \quad \dots \quad y_k \quad \dots \quad y_n], k = 1, 2, \dots, n$$

第l隐含层各神经元的输出为：

$$h^{(l)} = [h_1^{(l)} \quad h_2^{(l)} \quad \dots \quad h_j^{(l)} \quad \dots \quad h_{s_l}^{(l)}], j = 1, 2, \dots, s_l$$

其中， s_l 为第l层神经元的个数。

设 $W_{ij}^{(l)}$ 为从l-1层第j个神经元与l层第i个神经元之间的连接权重； $b_i^{(l)}$ 为第l层第i个神经元的偏置，那么：

$$h_i^{(l)} = f(net_i^{(l)})$$
$$net_i^{(l)} = \sum_{j=1}^{s_{l-1}} W_{ij}^{(l)} h_j^{(l-1)} + b_i^{(l)}$$

其中， $net_i^{(l)}$ 为l层第i个神经元的输入， $f(\cdot)$ 为神经元的激活函数。通常在多层神经网络中采用非线性激活函数，而不是用线性激活函数，因为采用基于线性激活函数的多层神经网络本质上还是多个线性函数的叠加，其结果仍然为一个线性函数。

二、激活函数

BP神经网络通常使用下面两种非线性激活函数：

$$f(x) = \frac{1}{1 + e^{-x}}$$
$$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

第一种称为sigmoid函数或者logistics函数，第二种为双曲正切函数。

Sigmoid函数的图像如下图所示，它的变化范围为(0, 1)，其导数为 $f' = f(1 - f)$ 。

<	2020	
日	一	二
1	2	3
8	9	10
15	16	17
22	23	24
29	30	31
5	6	7

导航
博客园
首页
新随笔
联系
订阅RSS
管理

统计
随笔 - 40
文章 - 0
评论 - 80
引用 - 0
公告

昵称： Alex Yu
园龄： 11年
粉丝： 296
关注： 6
+加关注

搜索

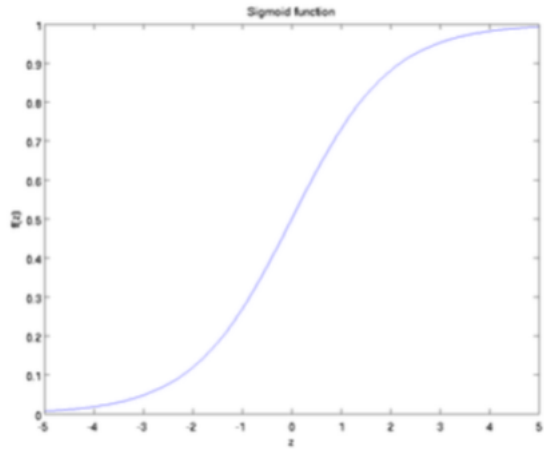
常用链接
我的随笔
我的评论
我的参与
最新评论
我的标签

最新随笔
1.萤火虫算法-pythor
2.进化策略-python
3.和声搜索算法-pyth
4.克隆选择算法-pyth
5.细菌觅食算法-pyth
6.蝙蝠算法-python
7.人工免疫算法-pyth
8.人工鱼群算法-pyth
9.人工蜂群算法-pyth
10.粒子群优化算法-p

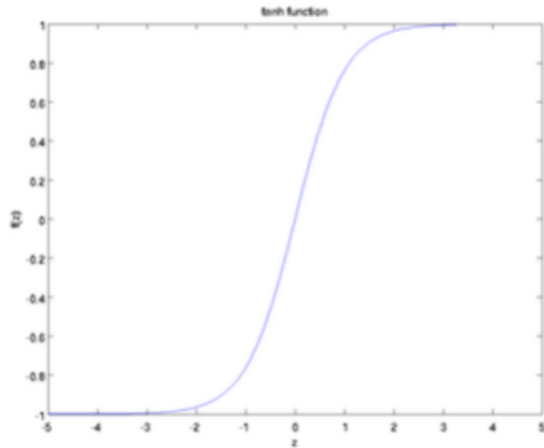
我的标签
python(23)
Eclipse(6)
RCP(5)
matlab(3)
问题求解(3)
遗传算法(3)
优化(2)
TSP(2)
插件(2)
扩展点(2)
更多

积分与排名
积分 - 71065
排名 - 9902

随笔分类 (105)
AI, Semantic Web,
Algorithm(18)
CAD
Eclipse RCP(5)
Graphics
Java(3)
LaTex(1)



双曲正切函数的图像如下图所示，它的变化范围为(-1, 1)，其导数为 $f' = 1 - f^2$ 。



三、BP算法推导过程

假定我们有 m 个训练样本 $\{(x(1), y(1)), (x(2), y(2)), \dots, (x(m), y(m))\}$ ，其中 $d(i)$ 为对应输入 $x(i)$ 的期望输出。BP算法通过最优化各层神经元的输入权值以及偏置，使得神经网络的输出尽可能地接近期望输出，以达到训练（或者学习）的目的。

采用批量更新方法，对于给定的 m 个训练样本，定义误差函数为：

$$E = \frac{1}{m} \sum_{i=1}^m E(i)$$

其中， $E(i)$ 为单个样本的训练误差：

$$E(i) = \frac{1}{2} \sum_{k=1}^n (d_k(i) - y_k(i))^2$$

因此，

$$E = \frac{1}{2m} \sum_{i=1}^m \sum_{k=1}^n (d_k(i) - y_k(i))^2$$

BP算法每一次迭代按照以下方式对权值以及偏置进行更新：

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha \frac{\partial E}{\partial W_{ij}^{(l)}}$$
$$b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial E}{\partial b_i^{(l)}}$$

其中， α 为学习速率，它的取值范围为(0, 1)。BP算法的关键在于如何求解 $W_{ij}^{(l)}$ 和 $b_i^{(l)}$ 的偏导数。

对于单个训练样本，输出层的权值偏导数计算过程：

Python(15)

产品生命周期管理(2)

数据结构与算法(26)

问题求解(18)

随笔档案 (40)

2015年10月(12)

2015年9月(10)

2015年6月(1)

2014年3月(1)

2014年1月(1)

2012年10月(2)

2012年9月(1)

2012年7月(1)

2012年6月(2)

2012年4月(1)

2011年12月(3)

2011年1月(1)

2010年3月(4)

最新评论

1. Re:萤火虫算法-py

你好，请问运行之后

NameError: name
是怎么回事呢

2. Re:人工蜂群算法-
这个是在解决什么问
能不能用在找最大值

3. Re:粒子群优化算法
问一下这段代码的目的
释看不太懂

4. Re:简单遗传算法-
@ 任新宇xrange改成

5. Re:粒子群优化算法
这里有整理好的 Pyth

6. Re:克隆选择算法-
请问计算delta的时候
self.params[2] + s
(self.params[3] - s
self.MAXGEN这句话

7. Re:蚁群算法java
法求解
楼主你好，我想请问一
点，该在那个位置判断
呢？因为通常的路径节
点和目标点的

8. Re:遗传算法Java
法求解
大佬，CX文字描述中
现了重复的基因，OX
列也有问题

9. Re:萤火虫算法-py
你好 请问有萤火虫算
最近在看这个

10. Re:差分进化算法
这里基本上涵盖了主流
性能DE算法，有详细

11. Re:粒子群优化算
objFunction 这个函

12. Re:简单遗传算法
ObjFunction.py 这
呢？？？

13. Re:简单遗传算法
for i in xrange(0, s
中，xrange报错，显
呢？

14. Re:BP神经网络并
@ 陆杰你好，如果可
推导过程的话，能麻烦

$$\begin{aligned}\frac{\partial E(i)}{\partial W_{kj}^{(L)}} &= \frac{\partial}{\partial W_{kj}^{(L)}} \left(\frac{1}{2} \sum_{k=1}^n (d_k(i) - y_k(i))^2 \right) \\ &= \frac{\partial}{\partial W_{kj}^{(L)}} \left(\frac{1}{2} (d_k(i) - y_k(i))^2 \right) \\ &= -(d_k(i) - y_k(i)) \frac{\partial y_k(i)}{\partial W_{kj}^{(L)}} \\ &= -(d_k(i) - y_k(i)) \frac{\partial y_k(i)}{\partial net_k^{(L)}} \frac{\partial net_k^{(L)}}{\partial W_{kj}^{(L)}} \\ &= -(d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} \frac{\partial net_k^{(L)}}{\partial W_{kj}^{(L)}} \\ &= -(d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} h_j^{(L-1)}\end{aligned}$$

即：

$$\frac{\partial E(i)}{\partial W_{kj}^{(L)}} = -(d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} h_j^{(L-1)}$$

同理可得，

$$\frac{\partial E(i)}{\partial b_k^{(L)}} = -(d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}}$$

令：

$$\delta_k^{(L)} = -(d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}}$$

则：

$$\begin{aligned}\frac{\partial E(i)}{\partial W_{kj}^{(L)}} &= \delta_k^{(L)} h_j^{(L)} \\ \frac{\partial E(i)}{\partial b_k^{(L)}} &= \delta_k^{(L)}\end{aligned}$$

对隐含层L-1层：

$$\begin{aligned}\frac{\partial E(i)}{\partial W_{ji}^{(L-1)}} &= \frac{\partial}{\partial W_{ji}^{(L-1)}} \left(\frac{1}{2} \sum_{k=1}^n (d_k(i) - y_k(i))^2 \right) \\ &= \frac{\partial}{\partial W_{ji}^{(L-1)}} \left(\frac{1}{2} \sum_{k=1}^n (d_k(i) - f(\sum_{j=1}^{s_{L-1}} W_{kj}^{(L)} h_j^{(L-1)} + b_k^{(L)}))^2 \right) \\ &= \frac{\partial}{\partial W_{ji}^{(L-1)}} \left(\frac{1}{2} \sum_{k=1}^n (d_k(i) - f(\sum_{j=1}^{s_{L-1}} W_{kj}^{(L)} f(\sum_{i=1}^{s_{L-2}} W_{ji}^{(L-2)} h_i^{(L-2)} + b_j^{(L-1)} + b_k^{(L)})))^2 \right) \\ &= - \sum_{k=1}^n (d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} \frac{\partial net_k^{(L)}}{\partial W_{ji}^{(L-1)}}\end{aligned}$$

因为，

$$\begin{aligned}net_k^{(L)} &= \sum_{j=1}^{s_{L-1}} W_{kj}^{(L)} h_j^{(L-1)} + b_k^{(L)} \\ &= \sum_{j=1}^{s_{L-1}} W_{kj}^{(L)} f(\sum_{i=1}^{s_{L-2}} W_{ji}^{(L-2)} h_i^{(L-2)} + b_j^{(L-1)}) + b_k^{(L)} \\ &= \sum_{j=1}^{s_{L-1}} W_{kj}^{(L)} f(net_j^{(L-1)})\end{aligned}$$

所以，

$$\begin{aligned}\frac{\partial E(i)}{\partial W_{ji}^{(L-1)}} &= \sum_{k=1}^n (d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} \frac{\partial net_k^{(L)}}{\partial W_{ji}^{(L-1)}} \\ &= \sum_{k=1}^n (d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} \frac{\partial net_k^{(L)}}{\partial f(net_j^{(L-1)})} \frac{\partial f(net_j^{(L-1)})}{\partial net_j^{(L-1)}} \frac{\partial net_j^{(L-1)}}{\partial W_{ji}^{(L-1)}} \\ &= \sum_{k=1}^n (d_k(i) - y_k(i)) f(x)' \Big|_{x=net_k^{(L)}} W_{kj}^{(L)} f(x)' \Big|_{x=net_j^{(L-1)}} h_i^{(L-2)}\end{aligned}$$

同理，

谢。邮箱111501341

15. Re:人工蜂群算法
这个跑出来，结果没有
调整让算法收敛呢？

16. Re:差分进化算法
感觉博主写的太乱了
面

17. Re:粒子群优化算
@ 颯、阳参考Java类

18. Re:粒子群优化算
相当于C#里面的this

19. Re:粒子群优化算
self是自身啊，pytho

20. Re:人工蜂群算法
那个 fi 应该是作者打错

阅读排行榜

- 1. 简单遗传算法MAT
- 2. BP神经网络推导过
- 3. 蚁群算法java实现解(41240)
- 4. 遗传算法Java实现解(18861)
- 5. Python一些特殊用er、lambda、列表推
- 6. 简单遗传算法-pytl
- 7. 如何用TortoiseSV6)
- 8. 粒子群优化算法-py
- 9. Eclipse RCP开发3
- 10. 差分进化算法-py
- 11. Eclipse RCP 开发8)
- 12. LyX中文问题(50
- 13. Eclipse RCP开发入Console视图(5052
- 14. 人工蜂群算法-py
- 15. 萤火虫算法-pyth
- 16. N皇后问题java实
- 17. Protege 插件开发
- 18. 和声搜索算法-py
- 19. 和声搜索算法-py
- 20. 人工免疫算法-py

评论排行榜

- 1. BP神经网络推导过
- 2. 蚁群算法java实现解(11)
- 3. 差分进化算法-pytl
- 4. 粒子群优化算法-py
- 5. 简单遗传算法MAT
- 6. 简单遗传算法-pytl
- 7. 人工蜂群算法-pytl
- 8. 蝙蝠算法-python5
- 9. Protege 插件开发
- 10. 遗传算法Java实现求解(4)
- 11. 萤火虫算法-pyth
- 12. Python一些特殊用ter、lambda、列表推
- 13. 如何用TortoiseS
- 14. Eclipse RCP 开发
- 15. 细菌觅食算法-py
- 16. 克隆选择算法-py
- 17. 和声搜索算法-py
- 18. 人工免疫算法-py

推荐排行榜

- 1. 简单遗传算法MAT
- 2. 蚁群算法java实现解(9)
- 3. BP神经网络推导过
- 4. Python一些特殊用er、lambda、列表推
- 5. 萤火虫算法-pytho
- 6. 粒子群优化算法-py
- 7. 人工鱼群算法-pytl
- 8. 简单遗传算法-pytl
- 9. 差分进化算法-pytl
- 10. 细菌觅食算法-py

令：

$$\begin{aligned}\delta_j^{(L-1)} &= \sum_{k=1}^n (d_k(i) - y_k(i)) f'(x)|_{x=net_k^{(L)} W_{kj}^{(L)}} f'(x)|_{x=net_j^{(L-1)}} \\ &= \sum_{k=1}^n W_{kj}^{(L)} \delta_k^{(L)} f'(x)|_{x=net_j^{(L-1)}} \\ \frac{\partial E(i)}{\partial W_{ji}^{(L-1)}} &= \delta_j^{(L-1)} h_i^{(L-2)} \\ \frac{\partial E(i)}{\partial b_j^{(L-1)}} &= \delta_j^{(L-1)}\end{aligned}$$

由上可推，第*l*层（ $2 \leq l \leq L - 1$ ）的权值和偏置的偏导可以表示为：

$$\begin{aligned}\frac{\partial E(i)}{\partial W_{ji}^{(l)}} &= \delta_j^{(l)} h_i^{(l-1)} \\ \frac{\partial E(i)}{\partial b_j^{(l)}} &= \delta_j^{(l)}\end{aligned}$$

其中，

$$\delta_j^{(l)} = \sum_{k=1}^{s_{l+1}} W_{kj}^{(l+1)} \delta_k^{(l+1)} f'(x)|_{x=net_j^{(l)}}$$

四、BP算法过程描述

采用批量更新方法对神经网络的权值和偏置进行更新：

- 对所有的层 $2 \leq l \leq L$ ，设 $\Delta W^{(l)} = 0, \Delta b^{(l)} = 0$ ，这里 $\Delta W^{(l)}$ 和 $\Delta b^{(l)}$ 分别为全零矩阵和全零向量；
- For $i = 1:m$ ，
 - 使用反向传播算法，计算各层神经元权值和偏置的梯度矩阵 $\nabla W^{(l)}(i)$ 和向量和 $\nabla b^{(l)}(i)$ ；
 - 计算 $\Delta W^{(l)} = \nabla W^{(l)}(i)$ ；
 - 计算 $\Delta b^{(l)} = \nabla b^{(l)}(i)$ 。
- 更新权值和偏置：
 - 计算 $W^{(l)} = W^{(l)} + \frac{1}{m} \Delta W^{(l)}$ ；
 - 计算 $b^{(l)} = b^{(l)} + \frac{1}{m} \Delta b^{(l)}$ 。

作者：Alex Yu
出处：<http://www.cnblogs.com/biaoyu/>
本文版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章页面明显位置给出原文连接，否则保留追究法律责任的权利。

支持 飘过

分类: 数据结构与算法

好文要顶 关注我 收藏该文



Alex Yu
关注 - 6
粉丝 - 296

+加关注

« 上一篇： 如何用TortoiseSVN管理本地文档
» 下一篇： 快速排序算法-python实现

posted on 2015-06-20 23:48 Alex Yu 阅读(83737) 评论(14) 编辑 收藏

评论

#1楼 2015-11-06 15:52 夜泊枫桥

博主，你好。在输出层的权值偏导计算过程的最后， $E(i)$ 对 W_{kj} = $\Delta T_k \cdot H_j$ ，这边的 H_j 是第L层的，而上面的推导过程中却是第L-1层的，是不是存在误笔呢？

支持(3) 反对(0)

#2楼 2016-03-09 21:47 深入理解C++和OpenCV

博主你好，非常感谢你的博文。xue习了！！！！

#3楼 2016-08-24 13:16 月江西

楼主 您好。非常感谢您的分享。
我跟着你的思路一步步推下来，有几个疑问。
1.在第三部分，“所以”下面对L-1层权值求偏导时，是不是遗漏了负号？
2.第四部分，第2步 更新delt 欧米伽时，还要加上 原来的 delt 欧米伽

支持(4) 反对(0)

#4楼 2016-08-25 10:00 陆杰

@ 月江西
这里的推导大致框架没问题，个别地方的符号确实不对。我自己整理了一份推导，如果你需要的话，可以发给你。

支持(1) 反对(0)

#5楼 2016-08-25 10:08 陆杰

@ 月江西
还有第4部分的step3，应该是减去梯度，而不是加。

支持(0) 反对(0)

#6楼 2016-08-25 12:23 月江西

@ 陆杰
是的，我注意到了。公式里遗漏的“负号”与step3的“减”去梯度的两个错误一起看，结果反而是正确的...

支持(0) 反对(0)

#7楼 2016-08-25 14:41 陆杰

@ 月江西
你QQ是多少？我有几个疑惑，我们讨论下。

支持(0) 反对(0)

#8楼 2016-10-30 16:51 突然想起她的名字

想问一下E(i)的那个为何是d_k(i)-y_k(i)? 针对单个来说，不应该是 d(i)-y(i)么？自己没有相通，想问一下，谢谢。

支持(0) 反对(0)

#9楼 2016-11-10 16:43 进击的小学生

@ 突然想起她的名字
一个样本有n个输出

支持(0) 反对(0)

#10楼 2016-11-28 15:17 qige12

@ 突然想起她的名字
因为输出Y是个向量，里面有N个分量

支持(0) 反对(0)

#11楼 2016-11-28 16:45 突然想起她的名字

@ qige12
恩恩，已经理解了。thanks

支持(0) 反对(0)

#12楼 2016-11-28 16:46 突然想起她的名字

@ 进击的小学生已经理解了

支持(0) 反对(0)

#13楼 2018-01-11 16:39 哼哼1987

博主真厉害，推导的这么清楚，看西瓜书没看懂，还是你的思路清晰

支持(0) 反对(0)

#14楼 2019-06-28 10:42 zhou~

@ 陆杰
你好，如果可以的话，如果您还有这份推导过程的话，能麻烦您发给我一份吗？谢谢。邮箱1115013416@qq.com

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)， [访问](#) 网站首页。

- 【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库
- 【活动】腾讯云服务器推出云产品采购季 1核2G首年仅需99元
- 【推荐】96秒100亿！哪些“黑科技”支撑全球最大流量洪峰？
- 【推荐】精品问答：前端开发必懂之 HTML 技术五十问

Powered by:

博客园

Copyright © 2020 Alex Yu

Powered by .NET Core on Kubernetes