

正则化参数求解方法研究

杨浩, 马建红

(河北工业大学 计算机科学与软件学院, 天津 300401)

摘要: 针对 BP 神经网络算法训练过程中出现的过拟合问题, 提出了利用一阶原点矩, 二阶原点矩, 方差和极大似然估计概念的推广来计算 L2 正则化中正则化参数 λ 值的方法。该方法通过对算法数据集 $[X, Y]$ 中的 X 矩阵进行运算得到的四个 λ 值, BP 神经网络算法训练时通常采用的是贝叶斯正则化方法, 贝叶斯正则化方法存在着对先验分布和数据分布依赖等问题, 而利用上述概念的推广计算的参数代入 L2 正则化的方法简便没有应用条件限制; 在 BP 神经网络手写数字识别的实验中, 将该方法与贝叶斯正则化方法应用到实验中的算法识别结果进行比较, 正确率提高了 1.14—1.50 个百分点; 因而计算得到的 λ 值应用到 L2 正则化方法与贝叶斯正则化方法相比更使得 BP 神经网络算法的泛化能力强, 证明了该算法的有效性。

关键词: BP 神经网络; 贝叶斯正则化; 矩法估计; 极大似然估计; 方差; L2 正则化

Research on Method of Regularization Parameter Solution

Yang Hao, Ma Jianhong

(Hebei University of Technology, School of Computer Science and Engineering, Tianjin 300401, China)

Abstract: Aiming at the over fitting problem of BP neural network algorithm, this method is put forward to calculate the value of the regularization parameter applied to L2 regularization by using the concept of the first order origin moment, the two order origin moment, the variance and the maximum likelihood estimation. This method based on the X matrix of data sets to compute four value. Compared with the Bayesian regularization method used to train the BP neural network, Bayesian regularization method is depend on the prior distribution and the distribution of data dependence. The calculation method in this paper is simple and has no application conditions. In BP Neural Network handwritten digit recognition experiments, this method compared with the Bayesian regularization method improve the correct rate about 1.14—1.50 percentage points. Therefore, the method in this paper makes the algorithm more efficient. This method is validity.

Keywords: BPneural network; regularization; maximum likelihood Estimation; variance; moment estimation; L2 regularization

0 引言

机器学习中常常遇到高维小样本数据, 高维小样本数据的变量空间维数很高, 而样本空间维数并不高, 因此往往会为建模带来一系列的问题, 例如, 训练样本数量的不足会导致过拟合问题, 模型的泛化能力很差, 极端情况下训练样本数小于变量数会导致建模过程需要求解病态的欠定方程组, 进而导致模型的解不唯一, 过拟合的时候, 拟合函数的系数往往非常大, 过拟合中拟合函数需要顾忌每一个点, 最终形成的拟合函数波动很大。在某些很小的区间里, 函数值的变化很剧烈。这就意味着函数在某些小区间里的导数值 (绝对值) 非常大, 由于自变量值可大可小, 所以只有系数足够大, 才能保证导数值很大, 而正则化是通过约束参数的范数使其不要太大, 所以可以在一定程度上减少过拟合情况。

正则化方法可分为迭代正则化方法, 直接正则化方法, L1 和 L2 正则化方法与贝叶斯正则化方法。迭代方法如 Landweber 迭代法^[1] 和共轭梯度法 (Conjugate Gradient, CG)^[2],

但是迭代方法中又引入了确定迭代次数等新的参数确定问题。直接正则化方法有截断奇异值 (Singular Value Decomposition, TSVD) 方法^[3] 与洁洪诺夫正则化 (Tikhonov Regularization, TR)^[4], 洁洪诺夫正则化方法是在解决不适定问题中应用广泛的正则化方法, 但由于这种方法具有饱和效应, 不能使正则解与准确解的误差估计达到阶数最优, 傅初黎^[5] 等对洁洪诺夫正则化方法进行了改进, 提出了迭代的洁洪诺夫正则化方法, 但未能使得其解决非线性算法的正则化参数问题, 并且增加了迭代次数的确定问题。迭代正则化方法和直接正则化方法主要应用于线性问题的求解, 并不能解决 BP 神经网络的非线性问题的正则化问题。

避免过拟合是 BP 神经网络算法设计中的一个核心任务。通常采用增大数据量, 测试样本集的方法或者采用正则化的方法对算法性能进行提升。目前在 BP 神经网络算法正则化方面主要的解决办法是采用贝叶斯正则化方法 (Bayesian Regularization, BR) 和 L2 正则化方法。贝叶斯正则化由洁洪诺夫正则化方法发展而来的。在性能函数误差反馈的基础上, 加上了权值的反馈以此提高算法的泛化能力, 但贝叶斯正则化方法得到 λ 值的准确性存在依赖数据的分布和先验概率的问题^[6-11]。L2 正则化方法则需要确定它的正则化参数 λ 值, 但目前尚未有有效的方法进行计算。

因此研究一种计算 L2 正则化中正则化参数 λ 值的方法是一件很有意义的一件事。本文通过应用统计中的矩法估计, 方

收稿日期: 2017-02-24; 修回日期: 2017-03-15。

作者简介: 杨浩 (1994-), 男, 河北石家庄人, 硕士研究生, 主要从事数据挖掘方向的研究。

马建红 (1965-), 女, 河北保定人, 教授, 博士生导师, 主要从事计算机辅助创新设计过程与方法、TRIZ、软件工程及数据挖掘等方向的研究。

差和极大似然估计的概念推广的计算方法来得到正则化参数 λ 值, 并应用到 L2 正则化方法中与采用贝叶斯正则化方法和人工选取得到的正则化参数确定正则化参数进行实验对比, 人工选取正则化参数, 如从 $\lambda = 0.1$, 按照一定的步长增长如 0.1, 对比不同值得情况下, 代价函数取得较小值时且假设函数未发生过拟合时的值, 实验表明本论文中的方法得到的正则化参数值 λ 应用到 L2 正则化方法中相比于贝叶斯正则化方法更能够使得代价函数值小且假设函数不发生过拟合的现象。

1 解决原理

1.1 BP 神经网络

BP 神经网络^[12-14]是基于前馈和反向传播算法的神经网络, 包含输入层、隐含层和输出层, 隐藏层节点数越多, 隐藏层神经元数目越多, 神经网络泛化能力越差, 并且当遇到高维特征向量会给 BP 神经网络分类器带来过拟合的风险, 影响 BP 神经网络在新的数据集上的识别率, 也就是说 BP 神经网络的泛化表现不符合预期要求。

1.2 L2 正则化

向量的范数可以简单形象地理解为向量的长度, 或者向量到零点的距离, 或者相应的两个点之间的距离。向量的范数定义: 向量的范数是一个函数 $\|x\|$, 满足非负性 $\|x\| \geq 0$, 齐次性 $\|cx\| = |c| \|x\|$, 三角不等式 $\|x+y\| \leq \|x\| + \|y\|$ 。

$\|x\|_2$ 为 x 向量各个元素平方和的 1/2 次方, L2 范数又称 Euclidean 范数或者 Frobenius 范数。

由 L2 范数而来的 L2 正则化方法, L2 正则化就是在代价函数后面再加上一个正则化项:

$$C = C_0 + \frac{\lambda}{2n} \sum_w \omega^2 \quad (1)$$

C_0 代表原始的代价函数, 后面一项是 L2 正则化项表示所有参数 ω 平方的和, 除以训练集的样本大小 n 。 λ 就是正则项参数, 权衡正则项与 C_0 项的比重, 另外一个系数 1/2, 是为了后面求导的结果方便, 后面那一项求导会产生一个 2, 与 1/2 相乘刚好凑整。

L2 正则化避免 overfitting, 推导如下:

$$\frac{\partial C}{\partial w} = \frac{\partial C_0}{\partial w} + \frac{\lambda}{n} w \quad (2)$$

$$\frac{\partial C}{\partial b} = \frac{\partial C_0}{\partial b} \quad (3)$$

可以发现 L2 正则化项对 b 的更新没有影响, 但是对于 w 的更新有影响:

$$w = w - \eta \frac{\partial C_0}{\partial w} - \frac{\eta \lambda}{n} w \quad (4)$$

$$w = \left(1 - \frac{\eta \lambda}{n}\right) w - \eta \frac{\partial C_0}{\partial w} \quad (5)$$

在不使用 L2 正则化时, 求导结果中 w 前系数为 1, 现在 w 前面系数为 $1 - \eta \frac{\lambda}{n}$, 因为 η, λ, n 都是正的, 所以 $1 - \eta \frac{\lambda}{n}$ 小于 1, 它的效果是减小 w , 这也就是权重衰减 (weight decay) 的由来。当然考虑到后面的导数项, w 最终的值可能增大也可能减小。

L2 正则化项有让 w “变小” 的效果, 更小的权值 w , 表

示网络的复杂度更低, 对数据的拟合更好即奥卡姆剃刀法则, 而在实际应用中, 也验证了这一点, L2 正则化的效果往往好于未经正则化的效果。

1.3 BP 神经网络与 L2 正则化

在 BP 神经网络的代价函数后面添加正则化项如下:

$$J(\theta, \lambda) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{k=1}^k y_k^{(i)} \log(h_{\theta}(x^{(i)}))_k + (1 - y_k^{(i)} \log(1 - (h_{\theta}(x^{(i)}))_k)) \right] + \frac{\lambda}{2m} \sum_{l=1}^{l-1} \sum_{i=1}^{S_l} \sum_{j=1}^{S_{l+1}} (\theta^{(l)})^2 \quad (6)$$

其中: m 是训练集的数目, l 是神经网络的总层数, S_l 是第 l 层的神经元数量, λ 是正则化参数, θ 是神经网络的权重矩阵, k 是指第 k 个输出单元, $h_{\theta}(x^{(i)})$ 是 Sigmoid 函数。数据集 $[X, Y]$ 有两部分, 一部分是 X 矩阵即神经网络的输入端数据, 一部分是矩阵 y , 也就是神经网络的输出端数据。其中有两个参数是未知的, 一个是神经网络的权重矩阵 θ , 另外一个就是正则化参数 λ 。代价函数中的 θ 矩阵参数, 可以在没有 λ 的情况下进行训练, 对权重 θ 的训练完成之后, 此时的代价函数仅包含一个未知参数 λ , λ 与 θ 的求解是独立的。因而本文在算法开始训练之前计算得到 λ 值。

1.4 矩法估计, 方差, 极大似然概念推广求解 λ

矩法估计^[15], 有时候要估计的未知参数恰是总体 X 的某阶原点矩 $a_j = E(X^j), j = 1, 2, \dots$, 由于样本 (X_1, X_2, \dots, X_n) 反映了总体, 一种较为自然的方法是用样本原点矩 $A_j = \frac{1}{n} \sum_{i=1}^n X_i^j, j = 1, 2, \dots$, 来估计总体 X 的原点矩 a_j , 即对未知参数 a_j 采用估计量 $a_j = \frac{1}{n} \sum_{i=1}^n X_i^j = A_j, j = 1, 2, \dots$ 。

统计中的方差^[15], 是各个数据 (X_1, X_2, \dots, X_n) 分别与其平均数之差的平方的和的平均数。在许多实际问题中, 研究方差即偏离程度有着重要意义。

极大似然估计^[15], 设总体 X 服从某类离散型分布, 它的概率密度函数为 $P(x; \theta), (X_1, X_2, \dots, X_n)$ 是取自总体的一个样本, 在一次实验中获得样本值 (X_1, X_2, \dots, X_n) 的概率为:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P\{X_i = x_i\} = \prod_{i=1}^n P(x_i; q) \quad (7)$$

当 θ 的值在参数空间 Θ 内变动时, $\prod_{i=1}^n P\{X_i = x_i\}$ 一般随之取不同的值, 把 $\prod_{i=1}^n P\{X_i = x_i\}$ 视为 θ 的函数, 记为

$L(\theta; x_1, x_2, \dots, x_n)$ 或简记为 $L(\theta)$, 适当的选取 θ , 使得 $L(\theta)$ 的值达到最大, 把所选取的 θ 值作为未知参数 θ 的估计值, 这种方法获得参数估计的方法就是极大似然估计法。

设 (X_1, X_2, \dots, X_n) 是取自总体 $X \sim b(m, p)$ 的样本, 其中 $0 < p < 1$ 未知, 计算 p 的极大似然估计。

总体 X 的概率函数为:

$$P(x; p) = \binom{m}{x_i} p^x (1-p)^{m-x}, x = 0, 1, 2, \dots, m \quad (8)$$

因此似然函数为:

$$L(p) = \prod_{i=1}^n P(x_i; p) = \prod_{i=1}^n \binom{m}{x_i} p^{x_i} (1-p)^{m-x_i} =$$

$$\prod_{i=1}^n \binom{m}{x_i} p^{\sum_{i=1}^n x_i} (1-p)^{m-\sum_{i=1}^n x_i}, x_i = 0, 1, 2, \dots, m$$

$$\ln L(p) = \sum_{i=1}^n \ln \binom{m}{x_i} + \left(\sum_{i=1}^n x_i \right) \ln p + (mm - \sum_{i=1}^n x_i) (\ln(1-p)) \tag{9}$$

$$\frac{d \ln L(p)}{dp} = \frac{\sum_{i=1}^n x_i}{p} - \frac{mm - \sum_{i=1}^n x_i}{1-p} = 0 \tag{10}$$

解得 p 的估计值:

$$p = \frac{\frac{1}{n} \sum_{i=1}^n x_i}{m} = \frac{\bar{X}}{m} \tag{11}$$

以上是矩法估计、方差和极大似然估计的概念介绍，下面是对矩法估计、方差和极大似然概念的推广过程。

矩法估计、极大似然估计和方差在向量形式数据上的应用举例。假设判断一个灯泡质量的优劣，当灯泡寿命在 4 500 小时以上时，我们判断灯泡质量是优，灯泡寿命在 4 500 小时以下，判断是劣，总共测试 $n=5\ 000$ 个灯泡，如第一个灯泡寿命 $x_1=4\ 600$ ，第二个灯泡寿命 $x_2=1\ 800$ 等，根据每个灯泡的寿命 x_i 得到每个灯泡对应的评价， x_1 为优， x_2 为劣等，因而得到 5 000 个灯泡寿命数据及优劣评价，同时得到的是一维的 1 行 5 000 列的灯泡寿命数据的向量 $(x_1 x_2 \dots x_n)$ ， $n=5\ 000$ 。此时灯泡寿命数据组成的向量用于矩法估计时，计算一阶原点矩公式如下：

$$\lambda = \frac{1}{n} \sum_{i=1}^n x_i \tag{12}$$

计算二阶原点矩公式如下：

$$\lambda = \frac{1}{n} \sum_{i=1}^n x_i^2 \tag{13}$$

在计算极大似然估计时，用这个向量数据计算灯泡寿命的平均值除以灯泡个数得到极大似然估计值，计算公式如下：

$$\lambda = \frac{\frac{1}{n} \sum_{i=1}^n x_i}{n} = \frac{\bar{X}}{n} \tag{14}$$

通过计算灯泡的平均寿命，算出每个灯泡与灯泡平均寿命值差值的平方，从而得到一组新的 1 行 5 000 列的向量 $(x_1 x_2 \dots x_n)$ ，新的向量中的数据累加之后除以总数 n 就会得到方差，计算公式如下：

$$\lambda = \frac{(x_i - \frac{1}{n} \sum_{i=1}^n x_i)^2}{n} \tag{15}$$

目前，矩法估计、极大似然估计和方差都是对向量形式的数据的计算，样本数据 $X_i, 1 < i < n$ ，组成的数据集 $(X_1 X_2 \dots X_n)$ ，对于 BP 神经网络的输入端数据来说，输入端 X 是一个矩阵，本文将向量中每个数据 X_i 向量化，即把向量数据 $(X_1 X_2 \dots X_n)$ 推广到矩阵模型，此时 $(X_1 X_2 \dots X_n)$ 代表的就是一个矩阵，以此来完成将矩法估计，方差，极大似然估计概念的推广来计算正则化参数 λ 值。公式 (12) ~ (15) 的推广计算方法对应式 (16) ~ (19)。一阶原点矩概念推广，对算法数据集 $[X, Y]$ 的 X_{mm} (m, n 代表矩阵的维数) 矩阵每个 x_{ij} (i

代表行向量， j 代表列向量) 数值全部累加并且除以它的个数和 $m * n$ 。时间复杂度为 $O(m * n)$ ，即时间复杂度与 X 矩阵的维数有关，计算公式如下：

$$\lambda = \frac{1}{m * n} \sum_{i=1}^n \sum_{j=1}^m x_{ij} \tag{16}$$

二阶原点矩概念的推广，对算法数据集 $[X, Y]$ 的 X_{mm} 矩阵每个 x_{ij} 数值的平方数值全部累加并且除以它的个数和 $m * n$ 。时间复杂度为 $O(m * n)$ ，计算公式如下：

$$\lambda = \frac{1}{m * n} \sum_{i=1}^n \sum_{j=1}^m x_{ij}^2 \tag{17}$$

方差概念的推广，对算法数据集 $[X, Y]$ 的 X_{mm} 矩阵每个 x_{ij} 减去 x_{ij} 数值累加并除以矩阵数值个数 $m * n$ 的平均值之后的平方进行累加，除以维数 $m * n$ 。时间复杂度为 $O(m * n)$ ，计算公式如下：

$$\lambda = \frac{1}{m * n} \sum_{i=1}^n \sum_{j=1}^m (x_{ij} - \frac{1}{m * n} \sum_{i=1}^n \sum_{j=1}^m x_{ij})^2 \tag{18}$$

极大似然估计概念的推广，对算法数据集 $[X, Y]$ 的 X_{mm} 矩阵的 x_{ij} 数值累加，并且除以维数 $m * n$ 。时间复杂度为 $O(m * n)$ ，计算公式如下：

$$\lambda = \frac{\sum_{i=1}^m \sum_{j=1}^n x_{ij}}{m * n} \tag{19}$$

例如在一个 BP 神经网络算法数据集 $[X, Y]$ 中， X 是 $5\ 000 * 400$ 的矩阵， Y 是 $5\ 000 * 1$ 的列向量，则上述式子 (16) (17) (18) (19) 中 $n=5\ 000, m=400$ 。

2 分析与讨论

2.1 数据采集与处理

本文 BP 神经网络手写数字识别实验研究所使用的数据集来自在线课程 coursera 中吴恩达的机器学习课程的课后算法编程中提供的数据。

数据集中有 5 000 个训练实例，每个训练实例是 20 像素 * 20 像素的灰度图像的数字。通过 20 * 20 的像素网格“打开”一个 400 维的向量。这 5 000 个训练的例子中的每一个数字成为一个行向量组成的 $5\ 000 * 400$ 数据矩阵 X 。

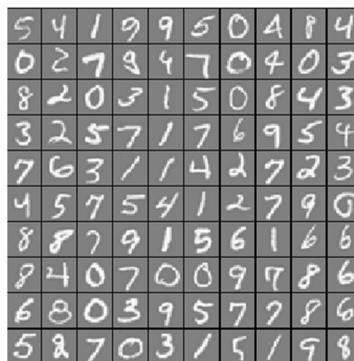


图 1 100 个手写数字训练图像

2.2 神经网络结构

Octave 设计实现一个包含两个隐藏层的 BP 神经网络，网络参数选择如表 1 所示。

表 1 实验中的神经网络结构

序号	参数	值
1	隐藏层神经元	25
2	隐藏层激活函数	Sigmoid(S型函数)
3	输出层激活函数	Sigmoid(S型函数)
4	训练次数	1000

2.3 实验结果分析

本文实验中, BP 神经网络输入端是 $5\,000 \times 400$ 的矩阵, λ 是从 -1 开始以步长 0.1 增长至 2 。其中当 $\lambda = 0$ 时, 代价函数较小为 $0.287\,63$, 实验中测试集的手写数字识别率达到了 96.5% , 泛化表现不错。

通过对数据集 $5\,000 \times 400$ 矩阵按照上述原理中的计算方法进行四个数值的运算, 利用推广的一阶原点矩概念计算的结果是 $0.131\,34$, 代价函数值是 $0.300\,26$, 利用推广的二阶原点矩概念计算结果是 $0.104\,31$, 代价函数是 $0.297\,66$, 利用推广的方差概念得到的 λ 值是 $0.087\,055$, 代价函数值是 $0.296\,00$, 利用推广的极大似然估计概念得到的值是 $0.131\,34$, 代价函数是 $0.300\,26$ 。将一阶原点矩 $0.131\,34$ 作为 λ 的值, 经过训练之后, 在实验中的测试集上手写数字识别率达到了 96.18% , 将二阶原点矩 $0.104\,01$ 作为 λ 的值, 经过训练之后在测试集上识别率达到了 96.42% , 将方差 $0.087\,055$ 作为 λ 的值, 经过训练之后, 在测试集上识别率达到了 96.06% , 将极大似然估计得到的 $0.087\,055$ 作为 λ 的值, 经过训练之后, 在测试集上识别率达到了 96.18% 。由数据集 $[\mathbf{X}, \mathbf{Y}]$ 中的 \mathbf{X} 进行矩法估计中一阶矩估计、二阶矩估计、方差及极大似然估计概念推广的公式简单计算就可以得到 λ 值, 并且算法有着不错的泛化表现。

为进一步验证所提方法的有效性, 和贝叶斯正则化进行了比较。由贝叶斯正则化方法计算得到正则化参数值 $0.999\,9$ 作为 λ 的值, 代价函数值是 $0.383\,76$, 经过训练之后, 算法识别率达到了 94.92% , 低于由矩法估计, 方差和极大似然估计概念推广得到的 λ 值训练之后手写数字识别率的最低值 96.06% , 即下表 2 所示。

表 2 试验中对比数据

方法	λ	代价函数值	识别率/%
一阶原点矩概念推广	$0.131\,34$	$0.300\,26$	96.18
二阶原点矩概念推广	$0.104\,31$	$0.297\,66$	96.42
方差概念推广	$0.087\,055$	0.296	96.06
极大似然概念推广	$0.131\,34$	$0.300\,26$	96.18
人工取值	0	$0.287\,63$	96.50
贝叶斯正则化方法	$0.999\,9$	$0.383\,76$	94.92

实验证明, 本文对一阶原点矩、二阶原点矩、方差及极大似然估计的概念推广计算得到的 λ 值, 并用这 4 个值作为 λ 取值计算得到的代价函数值与人工多次选取找到的最优 λ 值得到的代价函数值极为接近, 泛化表现也很接近。本论文的方法计算得到的 λ 应用到 L2 正则化方法进行算法的训练得到的手写数字识别率高于采用贝叶斯正则化方法的算法识别率。

3 结论

由一阶矩估计, 二阶矩估计, 方差和极大似然估计概念的推广公式对数据集 $[\mathbf{X}, \mathbf{Y}]$ 中的 \mathbf{X} 矩阵计算得到的 λ 值, 与人工多次选择正则化参数相比较效率更高, 并且在算法训练之前就可以计算出正则化参数, 很好地解决了算法有可能会出现的过拟合问题, 减少了因为选取正则化参数而导致多次计算代价函数等导致的多余的运算量。

对比在 BP 神经网络算法上应用广泛的贝叶斯正则化方法, 本文提出的正则化参数 λ 计算方法简单, 没有条件假设限制, 适用 BP 神经网络, 并且实验证明本文中的方法比贝叶斯正则化算法得到的 λ 值更能有效减少过拟合, 提高算法的泛化表现。

进一步的研究应该探讨本文方法应用于深度学习网络、线性回归及 svm 等算法中的可行性。

参考文献:

- [1] 谢正超, 王 飞, 严建华, 等. 炉膛三维温度场重建中 Tikhonov 正则化和截断奇异值分解算法比较 [J]. 物理学报, 2015, 64 (24): 17-24.
- [2] 崔 岩, 王彦飞. 基于初至波走时层析成像的 Tikhonov 正则化与梯度优化算法 [J]. 地球物理学报, 2015, 58 (4): 1367-1377.
- [3] 王 旭, 王静文, 王柯元. 阈值 Landweber 在 MIT 图像重建中的应用 [J]. 东北大学学报 (自然科学版), 2016, 37 (4): 476-480.
- [4] 雷 旒, 刘 峰, 曾喆昭. 基于共轭梯度法的 FIR 数字滤波器优化设计 [J]. 计算机仿真, 2014, 31 (12): 179-182.
- [5] 傅初黎, 李洪芳, 熊向团. 不适定问题的迭代 Tikhonov 正则化方法 [J]. 计算数学, 2006.
- [6] 张刚刚. 基于 Tikhonov 正则化的模糊系统辨识 [D]. 辽宁科技大学, 2014.
- [7] 高鹏毅. BP 神经网络分类器优化技术研究 [D]. 武汉: 华中科技大学, 2012.
- [8] 刘子翔. 基于 GA 和 LM 算法优化的 BP 神经网络在城市空气质量预测中的应用研究 [D]. 济南: 山东大学, 2015.
- [9] 赵 攀, 陈 昱, 汪一聪. 基于图像参数的 BP 网络的岩石颗粒体积估算 [J]. 计算机测量与控制, 2009, 17 (3): 571-575.
- [10] 郑德祥, 赖晓燕, 廖晓丽. 基于贝叶斯正则化 BP 神经网络的森林资源资产批量评估研究 [J]. 福建林学院学报, 2013, 33 (2): 132-136.
- [11] 刘 源, 庞宝君. 基于贝叶斯正则化 BP 神经网络的铝平板超高速撞击损伤模式识别 [J]. 振动与冲击, 2016, 35 (12): 22-27.
- [12] 宋 雷, 黄 腾, 方 剑, 等. 基于贝叶斯正则化 BP 神经网络的 GPS 高程转换 [J]. 西南交通大学学报, 2008, 43 (6): 724-728.
- [13] Zhang S H, Ou J P. BP-PSO-based intelligent case retrieval method for high-rise structural form selection [J]. Science China Technological Sciences, 2013, 56 (4): 940-944.
- [14] Rakhshkhorshid M, Teimouri S S A. Bayesian regularization neural networks for prediction of Austenite formation Temperature (Ac1 and Ac3) [J]. ScienceDirect, 2014, 21 (2): 246-251.
- [15] 刘达民, 程 岩. 应用统计 [M]. 北京: 化学工业出版社, 2005.