

基于神经网络的电池 SOC 估算及优化方法

李永颖¹, 张振东¹, 朱顺良²

(1. 上海理工大学, 机械工程学院, 上海 200093;

2. 国家机动车产品质量检测与监督中心, 新能源研究所, 上海 201800)

摘要: 鉴于锂电池高度非线性和时变性使其剩余电量难以精确估算, 影响电池的管理和控制; 基于 BP 神经网络模型, 在具有随机噪声干扰下, 分析和比较不同架构的深度学习模型对电池剩余电量估算的运算时间和泛化性能, 并根据粒子群算法 (PSO)、基于 Nesterov 动量的 RMSProp 变学习率算法优化模型, 结合数学规划设计出不同深度的最优构架, 并与多种神经网络模型进行比较; 根据实验数据和模型估算结果对比表明: 此优化算法能有效减少模型的运算时间, 在双隐层最优构架下, SOC 平均估算误差在 0.1 左右。

关键词: 锂离子电池; SOC; 神经网络; 粒子群算法; RMSProp

Battery SOC Estimation and Optimization Method Based on Neural Network

Li Yongying¹, Zhang Zhendong¹, Zhu Shunliang²

(1. College of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China;

2. National Motor Vehicle Product Quality Inspection and Supervision Center, New Energy Research Institute, Shanghai 201800, China)

Abstract: In view of the high nonlinearity and time-varying of lithium batteries, it is difficult to accurately estimate the remaining power, which affects the management and control of the battery. Based on the BP neural network model, under the random noise interference, the computational time and generalization performance of the battery's remaining power estimation are analyzed and compared with the deep learning model of different architectures, and based on particle swarm optimization (PSO), Nesterov momentum-based RMSProp, the learning rate algorithm optimization model, combined with mathematical programming, designs the optimal framework at different depths and compared with a variety of neural network models. The comparison between the experimental data and the model estimation results shows that the optimization algorithm can effectively reduce the computation time of the model. Under the optimal framework of the double hidden layer, the SOC average estimation error is around 0.1.

Keywords: lithium-ion battery; SOC; neural network; particle swarm optimization; RMSProp

0 引言

电动汽车动力电池的电荷状态 (State of Charge, SOC)、健康状态 (State of Health, SOH)、峰值功率 (State-of-power, SOP)、能量状态 (State of Energy, SOE)、剩余寿命 (Residual Life, RUL) 等参数是电池管理系统 BMS (Battery Management System) 对动力电池进行实时状态分析的支撑。动力电池电化学反应过程复杂, 受到温度、湿度、老化等诸多因素的共同影响, 具有不确定性, 其数学建模涉及多学科和多领域, 一直是学术界和工业界研究的难点和重点^[1]。神经网络模型具有很好的非线性逼近能力, 在处理非线性时变问题上具有巨大优势。蔡信等^[2]利用 ADVISOR 仿真数据对 BP 神经网络进行 SOC 估计, 结果表明模型的估计值和输出值之间的误差为 4% 左

右。苏振浩等^[3]使用经过等效电路模型修正后的数据对双隐层 BP 神经网络进行训练与 SOC 预测, 证明了人工神经网络模型对 SOC 估算的有效性。刘征宇等^[4]基于量子微粒群算法 (QPSO) 的 BP 神经网络模型用于 SOC 预测, 结果表明此方法比现有的神经网络预测方法准确度更高, 且具备很好的实用性。JOHNSONVH^[5]基于 ADVISOR 平台建立一个双隐层神经网络的电池模型。用电池负载功率和 SOC 输入到模型, 输出为电流和电压。针对神经网络结构设计方法困难、模型运算时间较长、对芯片运算能力要求高等原因降低其实用性的问题。在这里提出了能有效减少模型运算时间的训练算法和基于数学规划的神经网络模型构架设计方法。为神经网络对电池状态估计的工程应用提供一个解决方案, 具有参考价值。

1 前馈神经网络模型构架设计

模型深度和宽度的选择是一个非常复杂的问题, 目前并没有理论的指导, 没有一个好的解析式来表示。隐藏层单元数目与问题的需求、输入、输出单元数都有直接的关系。一般情况下仅能靠经验来确定模型的宽度和深度。

收稿日期: 2019-10-20; 修回日期: 2019-11-05。

作者简介: 李永颖 (1994-), 男, 上海人, 在读研究生, 主要从事 BMS 及智能控制技术方向的研究。

张振东 (1968-), 男, 教授, 博士生导师, 主要从事汽车系统仿真和电子控制技术方向的研究。

关于 BP 神经网络的隐藏层的设计有以下几种设计经验参考：

1) 1987 年 Hechtmielsen^[6] 提出隐藏层节点的数目为 $2n + 1$ ，式中 n 为输入的节点数。

2) 1988 年 Liu W^[7] 根据实验发现，在高维输入时，第一隐藏层对第二隐藏层的最佳节点比列为 3: 1。

3) 1990 年 Nelson 和 Illingsworth^[8] 建议隐藏层节点数应为 $4n$ 。

4) 隐含层的的节点数设计经验公式^[9]：

$$l = \sqrt{m+n} + d \quad (1)$$

式中， n 为输入层神经元数目； m 为输出层神经数目； d 为 1~10 之间的整数。

5) 最佳隐藏层单元数可参照以下公式^[10]：

$$l < n - 1 \quad (2)$$

$$l = \log_2 n \quad (3)$$

式中， n 为输入层单元数。

根据实验数据，采用数学规划的思想对神经网络构架进行选择。取模型训练至预测时间称为运算时间 Ct ，及每个预测点的平均绝对误差 E_{ss} 为性能指标。 E_{ss} 计算公式如式 (4)。 Ct 与 CUP 性能、模型结构及算法等有关，运算时间 Ct 和平均绝对误差 E_{ss} 作为设计模型的重要参考。

$$E_{ss} = \frac{1}{n} \sum_{i=1}^n |SOC_i^R - SOC_i^I| \quad (4)$$

式中， n 表示数据样本数， SOC_i^R 表示第 i 个 SOC 实验值， SOC_i^I 则是对应的模型预测值。

可以建立多目标线性规划模型来选择最优模型。其数学模型如下：

$$\min Ct \quad (5)$$

$$\min E_{ss} \quad (6)$$

s. t.

$$0 \leq Ct \leq \max(Ct_i)$$

$$0 \leq E_{ss} \leq \max(E_{ss}_i)$$

式中， i 是按模型隐含层数划分的模型序号 ($i=1, \dots, 5$)。

Hornik, Cybenko 等^[11-13] 表明，一个前馈神经网络如果具有线性输出层和至少一层具有任何一种“挤压”性质的激活函数的隐藏层，只要给予网络足够数量的隐藏单元，它可以以任意精度来近似任何一个有限维空间到另一个有限维空间的 Borel 函数。

在网络结构和阈值相同的情况下，BP 神经网络预测误差和均方误差、输出单元激活函数的关系如表 1^[14] 所示。隐含层和输出层激活函数的选择对 BP 神经网络预测精度有较大的影响。

在权衡收敛速度和预测误差后，用 tan sig 函数为隐藏层激活函数，purelin 函数为输出层激活函数，它们的表达式为：

$$\tan \operatorname{sig}(x) = \frac{1 - e^{-ax}}{1 + e^{-ax}} \quad (7)$$

$$\operatorname{purelin}(x) = x \quad (8)$$

表 1 不同激活函数对应预测误差

隐藏层函数	输出层函数	误差百分比	均方误差
logsig	tansig	40.63%	0.9025
logsig	purelin	0.08%	0.0001
logsig	logsig	352.65%	181.2511
tansig	tansig	31.90%	1.1733
tansig	logsig	340.90%	162.9698
tansig	purelin	1.70%	0.0107
purelin	logsig	343.36%	143.76334
purelin	tansig	120.08%	113.0281
purelin	purelin	196.49%	99.0121

tan sig 和 purelin 函数图形如图 1。tansig 函数具有“挤压性质”。

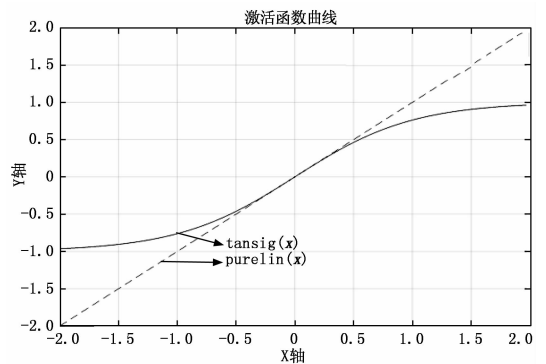


图 1 激活函数图形

tansig 函数导数为：

$$\frac{d}{dx} \tan \operatorname{sig}(x) =$$

$$\frac{a}{2} [1 + \tan \operatorname{sig}(x)] \times [1 - \tan \operatorname{sig}(x)] \quad (9)$$

式中， a 为常数，tansig 函数导数的这个性质将导数运算转换为乘法运算可以节省大量的计算时间，同时也提高了运算精度。

2 模型优化算法

2.1 模型训练算法

虽然随机梯度下降 (SGD) 在一般的机器学习中是应用最多的优化算法，特别是在深度学习中，但其学习过程有时会很慢。动量方法可以加速学习，特别是处理高曲率、小但一致的梯度，或者是带有噪声的梯度。随机梯度下降和动量方法的结合，有利于神经网络性能的提升。Nesterov 动量算法^[15]，是带动量的随机梯度下降算法，在凸批量梯度的情况下，Nesterov 动量将额外误差收敛率从 $O(1/k)$ (k 步后) 改进到 $O(1/k^2)$ ，有效减少了神经网络陷入局部最小值的风险。使用粒子群算法对权值和阈值优化，有效的提高了模型的泛化性能。

神经网络的学习率的取值在 $[0, 1]$ 之间，学习率越大，对阈值的修改就越大，网络学习速度越快。但是过大的学习率将使阈值学习过程产生震荡，过小的学习率使网

络收敛慢, 阈值难以稳定。变学习率在 BP 神经网络进化初期较大, 网络收敛迅速, 随着学习过程的进行, 学习率不断减小, 网络趋于稳定。变学习率计算公式为:

$$\epsilon(k) = \epsilon_{\max} - \frac{k(\epsilon_{\max} - \epsilon_{\min})}{k_{\max}} \quad (10)$$

式中, ϵ_{\max} 为最大学习率; ϵ_{\min} 为最学习率; k 为当前迭代次数; k_{\max} 为最大迭代次数。

基于上述深度神经网络模型的学习优化算法 Nesterov 动量算法和参数初始化算法 RMSProp 算法^[16], 两者相结合, 在此基础增加了变学习率算法得到模型训练算法如下所示, 式中用符号 \odot 表示 Hadamard 乘积。

基于 Nesterov 动量的 RMSProp 变学习率算法:

Resquire: 全局衰减率 ρ , 动量参数 α

Resquire: 初始参数 θ , 学习率 ϵ , 速度 v , 初始累计变量 $\gamma=0$

while 没有到达停止准则 do

从训练集中采含 m 个样本 $\{x_1, x_2, \dots, x_m\}$ 的小批量, 对应目标为 y_i 。

应用临时更新: $\tilde{\theta} \leftarrow \theta + \alpha v$

计算梯度: $g \leftarrow \frac{1}{l} \nabla_{\theta} \sum_{i=1}^l L(f(x_i; \theta), y_i)$

累计平方梯度: $\gamma \leftarrow \rho \gamma + (1 - \rho) g \odot g$

计算速度更新: $v \leftarrow \alpha v - \frac{\epsilon}{\sqrt{\gamma}} \odot g$

权值/阈值更新: $\theta \leftarrow \theta + v$

学习率更新: $\epsilon \leftarrow \epsilon_{\max} - \frac{k(\epsilon_{\max} - \epsilon_{\min})}{k_{\max}}$

end

2.2 模型参数优化算法

粒子群优化^[17] (Particle Swarm Optimization, PSO) 算法是由 Kennedy 和 Eberhart 在 1995 年在受到鸟类觅食现象启发, 提出的一种全局智能随机优化算法。PSO 算法, 每只鸟”被看作一个“粒子”。在 R 维解空间中第 i 个粒子的位置可表示为 $p_i = (p_{i1}, p_{i2}, \dots, p_{iR})$; 第 i 个粒子的速度表示为 $v_i = (v_{i1}, v_{i2}, \dots, v_{iR})$ 。第 i 个粒子在空间飞行路线中的最好点表示为 $t_i = (t_{i1}, t_{i2}, \dots, t_{iR})$; 粒子群内部所有粒子飞行过的最好的点表示为 $t_g = (t_{g1}, t_{g2}, \dots, t_{gR})$, 粒子群的每个粒子的速度和位置更新迭代公式为:

$$v_{id}^{k+1} = \omega_i v_i^k + c_1 r_1 (t_i^k - p_i^k) + c_2 r_2 (t_{gi}^k - p_i^k) \quad (11)$$

$$p_i^{k+1} = p_i^k + v_i^{k+1} \quad (12)$$

式中, ω_i 为惯性权重, r_1 和 r_2 为 0 到 1 的随机数, c_1 为局部学习因子、 c_2 为全局学习因子。

对模型参数使用粒子群算法优化, 避免了目标函数陷入局部最小值的风险, 同时也提高了模型的训练速度和泛化性能。

2.3 过拟合优化措施

过拟合 (overfitting) 是神经网络模型不可避免的, 目前有两种常用于缓解神经网络过拟合的方法, 第一种策略是“早停” (early stopping): 将数据分成训练集和测试集, 训练集用来计算梯度、更新权值和阈值, 验证集用来估计误差, 若训练集误差低但测试集误差升高, 则停止训练, 同时返回具有最小验证集误差权值和阈值。第二种策略是”

正则化” (regularization)^[10], 其基本思想是在误差目标函数中增加一个用于描述网络复杂度的部分。

3 模型优化算法数学证明

总优化算法流程如图 2 所示。

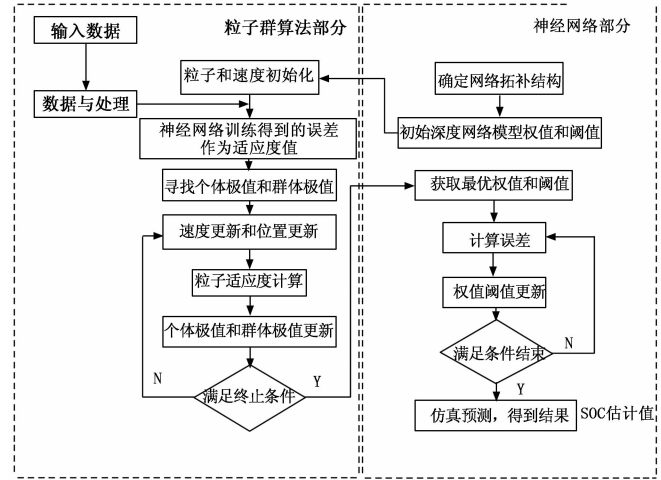


图 2 模型算法流程

给定训练集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ $x_i \in R^{s \times m}$, $y_i \in R^{n \times m}$, 即输入示例有 s 个特征属性描述, 输出 $n \times m$ 实值矩阵。基于 Nesterov 动量的 RMSProp 变学习率算法参数迭代更新公式推导如下, 符号说明如表 2。

表 2 符号说明

符号	说明
θ	粒子群算法优化的模型参数, 包含所有阈值和权值
w_{hj}	隐藏层第 h 个神经元与输出层第 j 个神经元之间的连接权值
u_j	输出层第 j 个神经元接收到的输入
ϵ	学习率取值范围(0,1)
ρ	衰减率取值范围(0,1)
α	动量系数取值范围(0,1)
γ_{hj}	隐层第 h 个神经元与输出层第 j 个神经元之间的累计变量
v_{hj}	隐层第 h 个神经元与输出层第 j 个神经元间的速度
d	输入神经元个数
l	隐藏层神经元个数
q	隐藏层层数

对训练集 (x_k, y_k) 假定网络输出为:

$$\hat{y}_j^k = f(u_j - b_j) \quad (11)$$

网络在训练集 (x_k, y_k) 的均方误差为:

$$E_k = \frac{1}{l} \sum_{j=1}^l (\hat{y}_j^k - y_j^k)^2 \quad (12)$$

参数更新估计式为:

$$\Delta \theta = \theta + v \quad (13)$$

以隐藏层到输出层的连接权值 w_{hj} 例来推导。给初始学习率 ϵ , 有:

$$\Delta w_{hj} = -\epsilon \frac{\partial E_k}{\partial w_{hj}} \quad (14)$$

临时更新:

$$\tilde{w}_{hj} = w_{hj} + \alpha v_{hj} \quad (15)$$

考虑 ω_{hj} 影响到第 j 个输出层神经元的输入值 β_j ，再影响到其输出值 \hat{y}_j^k ，然后影响到 E_k ，有：

$$\frac{\partial E_k}{\partial \omega_{hj}} = \frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial u_j} \cdot \frac{\partial u_j}{\partial \omega_{hj}} \quad (16)$$

根据 u_j 的定义，显然有：

$$\frac{\partial u_j}{\partial \omega_{hj}} = b_h \quad (17)$$

式中， $u_j = \sum_{h=1}^q \omega_{hj} b_h$ 。

输出层使用 purelin 函数，由式 (11) 和 (12) 有：

$$g_j = -\frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial u_j} = -(\hat{y}_j^k - y_j^k) \text{purelin}'(u_j - b_j) = (\hat{y}_j^k - y_j^k) \quad (13)$$

累计梯度：

$$r_{hj} = \rho r_{hj} + (1 - \rho) g_j \cdot g_j \quad (14)$$

速度更新：

$$\bar{v}_{hj} = av_{hj} - \frac{\epsilon}{\sqrt{\gamma_{hj}}} \cdot g_j \quad (15)$$

得连接权值 ω_{hj} 更新公式：

$$\bar{\omega}_{hj} = \omega_{hj} + av_{hj} - \frac{\epsilon}{\sqrt{\gamma_{hj}}} \cdot g_j \quad (16)$$

类似可以得到其阈值的更新公式：

$$\bar{b}_h = b_h + av_{hj} - \frac{\epsilon}{\sqrt{\gamma_{hj}}} \cdot g_j \quad (17)$$

4 前馈神经网络 SOC 估计结果与分析

模型的输入为负载功率 P 、电流 I 、温度 T 及随机干扰信号 D ，输出为 SOC 预测值。训练数据采用温度为 20℃ 下，纯电动车整车在 CYC_NEDC（新欧洲行驶循环）工况下的 8 次循环中电池包的电流、负载功率、温度的 0.1 s 采样值，使用经预处理的 15 381 个高质量数据样本对模型进行训练和预测。整车参数如表 3。8 次新欧洲行驶工况曲线如图 3，其对应的电池组参数变化曲线如图 4，可看出 SOC 变化是高度非线性，并且受到老化、温度等多因素影响。使用如图 5 所示的干扰信号模拟随机因素对 SOC 估计的干扰。

表 3 整车技术参数

项目	参数
总长×总宽×总高(mm)	2500×1480×1480
迎风面积/m ²	2
整车整备质量/kg	897
轴距/cm	260
轮胎滚动半径/cm	28.2
三元锂电池容量/A·h	60
蓄电池组单元个数/个	25
电动机最大效率/%	0.92
电动机最大功率/kW	75

运算时间在 Intel (R) Core (TM) i7-4510U CUP@2GHz 处理器，Matlab 平台上对对电池包实验数据进行训练和预测所计算得到，因为不同的芯片运算性能不同，SOC 预测时间也不同，但是模型的优化效果具有相同的趋势，这里对模型运算时间进行定性分析。

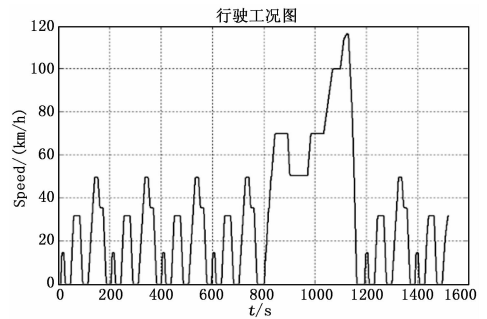


图 3 8 次 CYC_NEDC 循环工况曲线图

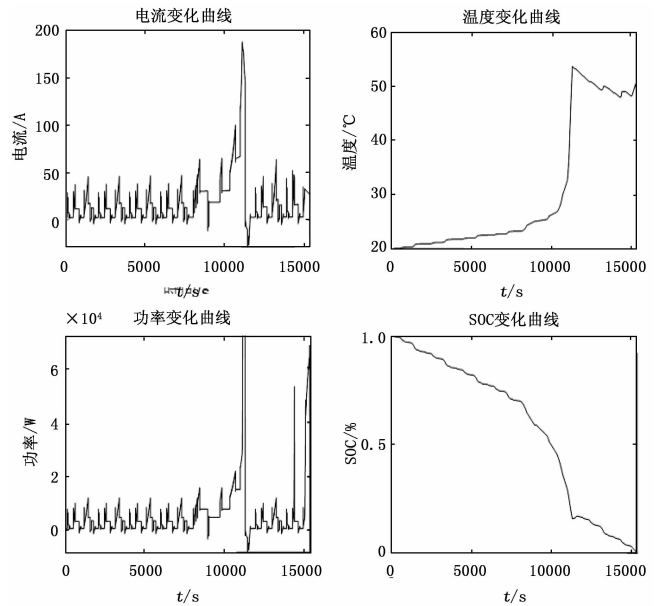


图 4 电池组参数变化曲线

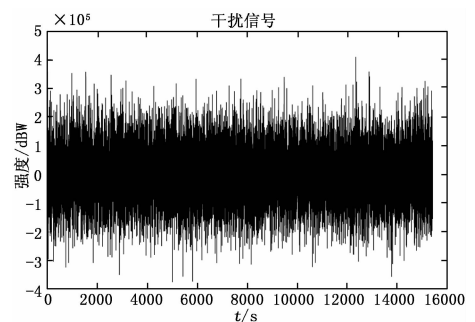


图 5 随机干扰信号

参考神经网络模型设计经验并根据实验数据，利用多目标数学规划选择出不同深度的神经网络最优构架。由于神经网络训练和预测存在不稳定性，即每次训练和预测的结果存在细微差别，故每个构架进行多次训练和预测，对运算时间和平均绝对误差取平均值。采用传统 Levenberg_Marquardt 的 BP 训练算法作对比，其优化效果如表 4 所示，经过粒子群算法和基于 Nesterov 动量的 RMSProp 变学习率训练算法优化后，在保证 SOC 平均绝对误差满足工程要求的前提下，有效减少了运算时间，特别是单元数多的构架，

这证明这套模型优化方法能快速找到最优的模型参数, 减少迭代次数, 从而减少运算时间, 并且设计出的最优构架结构简单、SOC 预测精度高, 具有一定的参考价值。

表 4 不同模型架构的优化效果

模型深度	模型构架	运算时间/s	平均绝对误差
单层	[4 27 1]	优化前:5.2339	0.0060
		优化后:0.9111	0.0546
双层	[4 2 3 1]	优化前:0.8625	0.0159
		优化后:0.4607	0.1290
3 层	[4 7 1 7 1]	优化前:1.9076	0.0120
		优化后:0.5589	0.1842
4 层	[4 1 5 1 5 1]	优化前:1.7188	0.0342
		优化后:0.4918	0.1863
5 层	[4 2 3 2 3 3 1]	优化前:2.4063	0.0621
		优化后:0.4746	0.2676

由不同构架的神经网络对电池 SOC 估计的结果分析, 同一深度前馈神经网络 SOC 预测精度随着单元数的增多而减少, 但是运算时间也增加。预测精度随着网络深度的增加而减少, 运算时间增加。因此, 在相同输入输出数据集条件下, 神经网络运算时间与模型训练算法、模型构架和芯片性能有关。

根据行业标准, 考虑到 SOC 估算时间及其平均绝对误差, 还有模型结构复杂度和可靠性。选择优化后的双隐层, 构架为 [4 2 3 1] 的神经网络模型作为锂电池 SOC 的估算模型, 运算时间为 0.4607 秒, SOC 预测平均绝对误差为 0.1290, 拟合曲线如图 6 所示, R 值在每次运算时都能达到 0.9, 接近 1, 可见该模型稳定性高, 预测精度可靠。并且迭代 77 次左右时, 残差已经收敛, 达到期望误差。如果搭载神经网络专用芯片或性能更高的芯片, 运算时间会进一步减少, 实时预测成为可能。

分别使用同样的数据集, 对径向基函数神经网络 (RBF)、广义回归神经网络 (GRNN) 和 Elman 神经网络进行训练和预测, 结果如图 7 所示。由图可以看出由于 BP 神经网络结构简单, 结合上述优化算法, 使其在动力电池状态估计上比其他常见的神经网络模型具有更少的运算时间和较好的预测精度。证明经过优化后的双隐层 BP 神经网络具有更好的工程实用性。

5 结论

由整车实验数据, 参考 BP 神经网络设计经验, 研究更深的神经网络模型构架设计, 对不同构架的神经网络进行了性能分析。针对这类模型 SOC 估计运算时间长、构架设计困难、对芯片性能要求高、无法实时预测等困难, 使用能有效减少模型运算时间的粒子群优化算法和基于 Nesterov 动量的 RMSProp 变学习率训练算法, 对模型进行了优化。SOC 估计误差范围的条件下, 运用数学规划方法选择一个平均绝对误差小且训练时间短、稳定性高并具有实用性的双隐层神经网络模型用于 SOC 估计。由于神经网络训练样本数没有理论指导, 套模型优化方法与模型设计方法

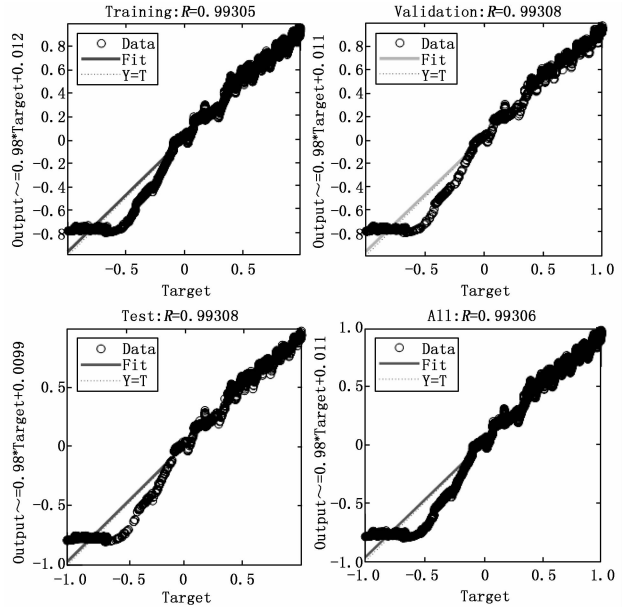


图 6 双隐层最优构架回归曲线图

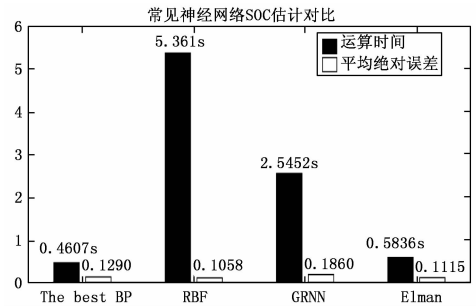


图 7 常见神经网络模型 SOC 估计对比图

没有在训练样本数上进行研究, 但为神经网络模型在电池其他状态估算上提供有价值的参考。随着神经网络专用芯片不断投入市场, 神经网络模型电池状态估计有较大的应用前景。

参考文献:

- [1] 胡晓松, 唐小林. 电动车辆锂离子动力电池建模方法综述 [J]. 机械工程学报, 2017, 53 (16): 20-31.
- [2] 蔡 信, 李 波, 汪宏华, 等. 基于神经网络模型的动力电池 SOC 估计研究 [J]. 机电工程, 2015, 32 (1): 128-132.
- [3] 苏振浩, 李晓杰, 秦 晋, 等. 基于 BP 人工神经网络的动力电池 SOC 估算方法 [J]. 储能科学与技术, 2019, 8 (5): 868-873.
- [4] 刘征宇, 杨俊斌, 张 庆, 等. 基于 QPSO-BP 神经网络的锂电池 SOC 预测 [J]. 电子测量与仪器学报, 2013, 27 (3): 224-228.
- [5] Johnson V H. Battery performance models in ADVISOR [J]. Journal of Power Sources, 2002, 110 (2).
- [6] Hechtmielsen R. Nearest matched filter classification of spatiotemporal patterns [J]. Applied Optics, 1987, 26 (10): 1892-1899.

(下转第 194 页)