分类号 密级

UDC

学 位 论 文

基于改进优化算法的DCNN的剩余使用寿命预测模型研究

|  |  |  |  |
| --- | --- | --- | --- |
| 作者姓名： | 张佳旭 | | |
| 作者学号： | 1770029 | | |
| 指导教师： | 宋叔尼 教授 | | |
|  | 东北大学理学院 | | |
| 申请学位级别： | 硕士 | 学科类别： | 应用统计 |
| 学科专业名称： | 应用统计 | | |
| 论文提交日期： | 2019年12月 | 论文答辩日期： | 2019年12月 |
| 学位授予日期： | 2020年 1月 | 答辩委员会主席： |  |
| 评阅人： |  | | |

东 北 大 学

2019年12月

**A Thesis in Applied Statistics**

**Research on Remaining Useful Life Prediction Model of DCNN Based on Improved Optimization Algorithm**

By Zhang Jiaxu

Supervisor: Professor Song Shuni

**Northeastern University**

**December 2019**

# 独创性声明

本人声明，所呈交的学位论文是在导师的指导下完成的. 论文中取得的研究成果除加以标注和致谢的地方外，不包含其他人已经发表或撰写过的研究成果，也不包括本人为获得其他学位而使用过的材料. 与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意.

学位论文作者签名：

日 期：

学位论文版权使用授权书

本学位论文作者和指导教师完全了解东北大学有关保留、使用学位论文的规定：即学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅. 本人同意东北大学可以将学位论文的全部或部分内容编入有关数据库进行检索、交流.

作者和导师同意网上交流的时间为作者获得学位后：

半年 □ 一年□ 一年半□ 两年□

学位论文作者签名： 导师签名：

签字日期： 签字日期：

# 摘 要

系统预测和健康管理 (PHM) 的目标包括通过监控设备状况，最大限度地提高操作的可用性，降低维护成本，提高系统可靠性和安全性. 剩余使用寿命(RUL)是PHM的重要一环，它可以根据历史轨迹数据进行估算. 本文改进了神经网络优化方法，从实际数据本身出发，在航空发动机机组预测常用的C-MAPSS数据集上进行了实验.

1. 对剩余使用寿命预测的背景、国内外研究方法、研究意义进行概述；介绍了本文所涉及到的数据预处理方法及深度学习理论.

2. 针对神经网络梯度下降后期会出现振荡导致迟迟不收敛的行为，利用牛顿法及其优化策略作为后期优化算法替代原始优化算法，使用UCI公共数据集验证L-BFG和信赖域搜索的组合能够具有更好的预测性能.

3. 基于航空发动机机组预测常用的C-MAPSS数据集，利用Python实现深度卷积神经网络，对剩余使用寿命预测进行预测，预测结果与文献中方法取得的结果预测的误差进行比较，测试集均方根误差最多相差0.1%.

4. 基于信赖域搜索的拟牛顿法的DCNN神经网络具有良好的预测能力，有效增加了神经网络的收敛性能，将其应用于剩余使用寿命预测的预测中，实验表明预测误差降低了3%.

关键词：剩余使用寿命预测；深度学习；深度卷积神经网络；优化算法

# Abstract

The goals of prognostic and health management (PHM) include monitoringing the equipment status, maximizing the availability of operation, reducing the maintenance costs, and improving system reliability and security. The remaining useful life (RUL) is an important part of PHM and can be estimated based on historical trajectory data. This article improves the optimization method of neural network, experiments are carried out on C-MAPSS datasets commonly used in aircraft engine unit prediction.

1. The background, research methods and significance of remaining useful life are summarized, and the data preprocessing method and deep learning theory involved in this paper are introduced.

2. In view of the behavior of delayed convergence due to oscillation in the later stage of neural network gradient descent, Newton method and its optimization strategy were used as the later stage optimization algorithm instead of the original optimization algorithm, and UCI common dataset is used to verify that the combination of L-BFGS and trust region search can have better prediction performance.

3. Based on the c-mapss dataset commonly used in the prediction of aircraft engine unit, the deep convolutional neural network is implemented by Python to predict the remaining useful life. The prediction results are compared with the prediction obtained by the methods in the literature, and RMSE error of the test set is 0.01% at most.

4. DCNN neural network based on Quasi-newton optimization method has good prediction ability and effectively increases the convergence performance of neural network. It is applied to the prediction of remaining useful life prediction. The experiment shows that the prediction error is reduced by 3%.

**Key words:** remaining useful life prediction; deep learning; deep convolutional neural networks; optimization algorithm

# 目 录

[独创性声明 I](#_Toc23673333)

[摘 要 II](#_Toc23673334)

[Abstract III](#_Toc23673335)

[目 录 IV](#_Toc23673336)

[第 1 章 绪 论 1](#_Toc23673337)

[1.1 研究背景 1](#_Toc23673338)

[1.2 国内外研究现状和意义 3](#_Toc23673339)

[1.3 论文主要工作 4](#_Toc23673340)

[第 2 章 预备知识 5](#_Toc23673341)

[2.1特征归一化处理 5](#_Toc23673342)

[2.2 卷积神经网络 6](#_Toc23673343)

[2.3 神经网络中的参数 8](#_Toc23673344)

[2.3.1 连接权重 8](#_Toc23673345)

[2.3.2 网络中的激活函数 8](#_Toc23673346)

[2.3.3 正则化参数 10](#_Toc23673347)

[2.3.4 损失函数和学习率 11](#_Toc23673348)

[2.3.5 神经网络中的优化方法 14](#_Toc23673349)

[第 3 章 算法改进与验证 16](#_Toc23673350)

[3.1牛顿法及其优化策略 16](#_Toc23673351)

[3.2. 基于搜索步长的信赖域算法 23](#_Toc23673352)

[3.3 公共数据集验证 25](#_Toc23673353)

[第 4 章 改进DCNN在RUL预测中的应用 28](#_Toc23673354)

[4.1 数据参数说明 28](#_Toc23673355)

[4.2 数据预处理 29](#_Toc23673356)

[4.2.1数据归一化 29](#_Toc23673357)

[4.2.2 时间窗口处理 31](#_Toc23673358)

[4.3 DCNN神经网络模型 31](#_Toc23673359)

[4.3.1模型结构 31](#_Toc23673360)

[4.3.2实验预测程序与结果分析 34](#_Toc23673361)

[4.3.3卷积层数和时间窗的影响 39](#_Toc23673362)

[第 5 章 总结与展望 42](#_Toc23673363)

[5.1 总结 42](#_Toc23673364)

[5.2 展望 43](#_Toc23673365)

[参考文献 44](#_Toc23673366)

[致 谢 48](#_Toc23673367)

1. 绪 论

## 1.1 研究背景

科学技术的发展带动了社会生活的进步，智能设备也发展迅速，这主要体现在设备的复杂性和自动化能力的增强. 虽然大大节省了人力，但是设备寿命是有限的，它们会在内外因素的综合作用下，导致设备的健康程度与性能指标呈现衰退的趋势，当这种趋势达到一定程度时，设备会发生故障，无法继续完成正常的作业，进而造成难以挽回的经济损失和资源浪费[1]. 因而实时掌握并增强运行设备的安全性和可靠性迫在眉睫. 预测与健康管理(PHM)作为保障运行设备安全性和可靠性的一项重要方法，在过去十几年间取得了丰硕的理论成果并且得到了广泛的落地应用[2]. 美国空军为了监控军用直升机装备的健康与使用监测系统(HUMS)，这作为PHM 的雏形，最早被实际应用. 20世纪70年代，航空等重要设备由于技术不完善，经常会出现故障，针对此问题PHM的理念由此提出，并且经过了各国军方的共同努力，这一思想被成功地应用于联合攻击机F-35 的故障检测中[3]. 现阶段，军事设备是否具有PHM系统和PHM系统的能力容量已经成为了美国国防部采购装备时通常会关注的重点，由此可见PHM技术在国防军事领域中处于重要地位. 在工业领域中，PHM技术不仅能够有效提高设备的可维护性、减少宕机时间，而且能够通过提高作业设备的可靠性，来提高作业效率[4]. 在学术领域中，清华大学以及国防科技大学等高校的相关科研机构大力展开了PHM 技术的理论研究，同时也结合实际的工业问题进行探讨和落地应用，不断丰富PHM 技术理论体系.

PHM技术在航空航天、制造业、汽车、重工业等许多工业领域都是非常重要的. 而传统的策略，例如故障检修和定期预防性检修越来越难以满足工业上日益注重的效率问题和可靠性需求. PHM的目标包括通过监控设备状况，提高运行设备的可用性，降低维护成本，提高系统可靠性和安全性. PHM方法主要包括两个方面：剩余使用寿命(Remaining useful life，RUL)预测和健康管理(Health management，HM). RUL通常是用来描述设备当前运行时刻与故障时刻之间的时间间隔，可定义为下式[5]：

 (1.1)

其中表示运行设备的故障时刻，表示当前运行时刻. RUL预测对设备的实时维修和换件决策具有指导性意义，所以剩余使用寿命预测因其重大意义，被称为PHM技术的核心. 实现RUL预测的主要思想在于根据设备的故障机制、历史传感器监测数据以及失效模式等有效信息，得出RUL的分布或者期望[6].

近些年由于航空航天技术得到了大力发展，大型飞机得到普遍应用，但其安全问题还有待保障. 2015年发生的空难次数为10次，这比十年前降低了近3倍，但是遇难人数却增加到了723 人，这比十年前提高了整整5倍. 统计表明，在所有的空难事故中，设备运行故障占26.7%. 航空发动机，它作为航空器的核心，其健康程度直接决定着乘客们的生命安全以及飞机航行的安全. 因此为保障安全性，必须及时对航空发动机设备进行维修和更换. 然而，考虑到民航公司的运营成本，必须适当地把握维修和更换时间，不能过早地维修或更换，这样才能最大限度地提升航空发动机的使用价值. 目前，通过对航空发动机剩余使用寿命进行精准预测来指导工作人员对设备进行检修，是保证航空安全等待解决的问题.

传统的剩余寿命预测和设备可靠性评估是建立在大量的失效样本基础上，但航空发动机因其属于成本昂贵、寿命周期长、数据不易采集的机电液混合复杂设备，使得研究人员无法获得足够数量的失效样本，因此无法通过对数据进行统计推断来预测设备的剩余寿命. 不同设备因内外部影响因素不同，其性能退化状况也相差甚大. 随着传感器技术的高速发展和在工业设备上的快速应用，使得在飞机运行过程中可以实时地记录发动机各个硬件的运行状况，这为预测发动机剩余寿命提供了有利的技术与充分的数据准备，进而通过对发动机的性能与状态数据建模，评估设备的健康状况. 论文从实际数据本身出发，在航空发动机机组预测常用的C-MAPSS数据集上进行了实验.

## 1.2 国内外研究现状和意义

剩余寿命预测是通过收集设备在作业过程中产生的监测数据，预测设备未来的健康状况，即剩余使用寿命(RUL)，进而根据预测结果指导设备的管理策略. 如果能够在其性能衰退的初期，精准地对设备的剩余使用寿命进行预测，就可以有效节约成本，保障设备运行安全. 对剩余使用寿命预测的方法可以分为三种[7]：(1)根据设备的失效机理建立物理仿真模型预测设备的RUL；(2)基于设备在运行中得到的大量传感器数据，利用算法模型，结合阈值进而对RUL进行预测；(3)基于物理机制和数据建模方法相结合.

在前人的工作中，一部分学者对涡轮发动机剩余使用的研究，通过建立物理仿真模型[8]，从而对设备的剩余寿命进行预测. Marine Jouin等人[9]提出了使用粒子滤波器为剩余使用寿命预测建立了物理失效机制模型. Jaouher Ben Ali等人[10]使用Weibull分布建立剩余使用寿命模型，Weibull分布是一个典型的概率分布，它可以描述许多不同的失效类型. 然而由于运行设备的故障的复杂性和所处环境的多样性，物理模型需要大量的专业的先验知识，且泛化能力也不好，后期也需要进行维护.

一部分学者选择使用数据驱动的预测方法，数据驱动方法能够不依赖于已有的专家知识，基于历史传感器数据对退化特性进行建模，可以揭示其潜在的相关性和因果关系，并推断出相应的系统信息. 近年来提出的数据驱动算法，取得了良好的预测结果. Felix O. Heimes[11]先使用了多层感知机对RUL进行了预测，然而多层感知机并没有考虑数据的时间依赖关系，然后使用了循环神经网络(RNN)进行预测，当数据前后距离增加时，RNN变得无法连接相关信息，并且会伴随梯度消失或梯度爆炸问题的发生. 为了解决上述问题，Yu Peng等人[12]使用回声状态神经网络(ESN)框架建模，回声状态神经网络作为一种新型的递归神经网络，不会存在梯度问题，但是它的选取训练集和预测集的方式导致有20个发动机不能被预测. C. Louen等人[13]使用了支持向量机(SVM)算法和Weibull分布，先把该问题看成一个分类问题，使用SVM计算点到分离超平面的距离，假设该距离服从威布尔分布，使用加权最小二乘算法对威布尔分布的参数进行拟合，反过来计算剩余时间，由于SVM的核函数选择问题，效果并不理想. G. Sateesh Babu等人[14]则第一次使用了卷积神经网络(CNN)对RUL建模，CNN 则可以很好地识别出数据中的简单特征，然后使用这些简单特征在更高级的层中生成更复杂的特征，但是卷积核的尺寸选取不合适，使提取的局部特征不太好，导致模型拟合不充分. Pin Lim等人[15]使用K-means算法提取特征，并用一个多层神经网络作为主要的算法. 他的结果虽然比G. Sateesh Babu等人结果好，但是手工提取特征是不充分的. J.J Zhang等人[16]使用了LSTM考虑了数据的前后依赖关系，对RUL预测，J.J Wang等人[17]提出了一种基于双向长短时记忆(BiLSTM)网络的鲁棒估计数据驱动方法，充分利用了传感器数据序列的双向性，取得了更好的效果，但是他们都没有考虑到数据的局部特征. Xiang Li等人[18]则再一次把卷积神经网络(CNN)用来发动机预测RUL，他们使用conv1d提取特征，且没有使用池化操作过滤有用信息. 经过与现有方法使用均方误差(RMSE)作比较，结果表明其所提出的方法更有效.

## 1.3 本文主要工作

剩余使用寿命(RUL)可以根据历史轨迹数据进行估算，从而对工程起到指导作用. 前人工作中更多的是更换算法、调整网络的结构来对RUL进行预测，还未有通过改进优化算法对深度神经网络进行迭代计算来提高模型的准确性. 本文将基于航空发动机机组预测常用的C-MAPSS数据集，利用大数据和分析方法，建立剩余使用寿命模型，通过数据模型得到与实际剩余使用寿命更加贴切的预测值，提高预测结果的准确率. RUL的预测精度的提高，可以保证航空安全与公司效益.

1. 预备知识

## 2.1特征归一化处理

归一化处理就是把数据的取值范围约束到一定取值之间. 假设有两类数值型指标分别为身高和体重，其取值范围分别为[0.00，2.00]和[10，100]，如果对它们不做处理的话直接扔进算法中训练，一是可能会造成训练过程中模型更倾向于数值更大的特征；另外，对于一些依靠梯度下降求最优解的算法模型，如果不同维度的数据量纲差异较大，会使损失函数的下降路径变曲折，会延缓神经网络参数的更新速度，在搜索步长相同的情况下，身高维度的更新速度会大于体重维度的更新速度，身高和体重的更新速度的不同会使得在寻找最优解的过程中，需要迭代更多的次数.

为了消除数据之间的量纲差异对模型的影响，使得特征之间属于同一量纲，就需要对数据的不同维度进行归一化处理. 最经典的归一化方式是最大最小归一化. 它的主要工作方式是根据数据的每一个维度中的最大值和最小值，通过对数据进行线性变换，将原始数据约束到[0,1]范围内，当然也可以根据实际需要将数据映射到[-1,1]范围内，根据实际需要对数据进行最大最小化缩放. 其公式如下：



其中，表示原始数据，和分别表示原始数据各个维度中的最大值和最小值.

归一化处理后的特征能够使模型训练的收敛速度明显提升，同时也会提高参数在反向传播中的计算效率. 当然，对训练集合进行了归一化处理训练出来的网络模型，在对测试集进行预测和验证算法的泛化性能时也需要做出相同的操作，此处相同的操作是指按照训练样本各个维度的最大最小值来对测试数据进行最大最小化操作，这样才能保持训练集和测试集的数据分布一致.

特征归一化处理并不适用于所有算法，只有依靠梯度下降法来求解模型最优解的算法才需要对数据进行归一化处理，比如说Logistics回归和经典神经网络之类的，当然也有例外，例如PCA降维. 但像决策树这样主要学习数据的条件分布的模型，归一化处理是不适用的，因为归一化并不能改变决策树对维度划分时样本在特征上的划分标准.

## 2.2 卷积神经网络

在图像处理领域中，真实的图像是由一个个像素点组成的，每个像素点包含三个通道，它们分别代表红(R)、绿(G)、蓝(B)三个颜色. 例如一个图像的尺寸是，就是代表这个图像的是一个长为28，宽为28，通道数为1的图像，此处1代表灰色图像，如果是3则为彩色图像. 如果对图像使用常见的全连接网络结构，即上一层的神经元与下一层上的每个神经元均连接，那就意味着网络模型需要个神经元，比如hidden层采用了15个神经元，那么通过计算，模型需要计算的参数个数就达到:个，由此可见参数太多了，随便对参数进行一次反向传播计算量都是相当大的. 传统的神经网络无论是从计算效率还是从调参的角度都不建议使用在图像处理的领域.

卷积神经网络(CNN)最早由LeCun提出用于图像处理[19]，它具有空间共享权值和局部感知的两个特点. 卷积其实是图像处理中一种常用的线性滤波方法，使用卷积可以达到图像降噪、锐化等多种滤波效果.

卷积网络将多个卷积核与原始输入数据进行卷积并且生成特征，卷积层是构建卷积神经网络的核心层，它产生了网络中大部分的计算量. 所谓的卷积核，实际上就是一个权值矩阵，表示如何处理单个像素与其邻域像素之间的关系. 局部感知域、权值共享、多核卷积这三项是卷积层与全连接层最主要的不同. 局部感知域指的是，对于每一次卷机操作来说，仅仅需要考虑其像素位置附近的输入，没有必要与上一层的所有的节点相连. 权值共享是另一项可以减少模型参数个数的方式，对每一张图片进行卷积操作的时候，卷积核会逐一通过该图片的每个像素，也就是说，每个像素点的卷积核参数都相同. 每个卷积核是一个特征提取器，若只有一个卷积核的话，就只能提取一种特征，这会导致欠拟合. 所以可以使用多个卷积核，比如6个卷积核同时提取6种特征，以多核卷积进行卷积操作可以充分提取数据特征. 通过卷积层提取特征后，得到的特征图代表了比原始数据更高级的特征，然后输入到下面的池化层提取最重要的局部特征，卷积层原理图如下：

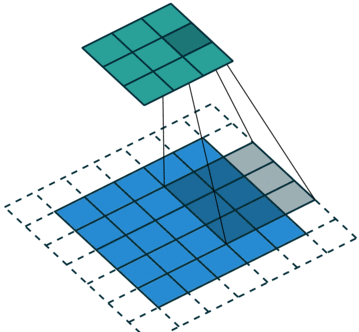


图2.1 卷积层原理图.

Fig. 2.1 The Schematic Figure of Convolutional Layer.

池化层主要位于卷积层之后， 用于降低数据的数据和减少参数的数量，降低过拟合的风险. 简而言之，如果输入是图像的话，那么池化层会起到压缩图像的作用. 池化层(采样层)主要是将图像按照设定的窗口大小划分成不同的窗口，然后对每一个窗口内的元素进行聚合. 通常使用的窗口大小为，聚合方式主要分为两种，一种是最大池化，即取窗口内的最大值. 另一种是平均池化，即取平均值. 比如窗口为的池化操作，处理完的图像长和宽都将是原始图像的一半，即输出尺寸是输入尺寸的.

## 2.3 神经网络中的参数

神经网络的参数主要包括神经元连接权重、激活函数、正则化参数、学习率和损失函数以及优化方法.

### 2.3.1 连接权重

神经网络不同网络层之间是依靠连接权重来连接的. 训练神经网络的目的就是获取一组最优的连接权重，使得神经网络拟合出的函数与待估函数足够接近.神经网络层与层之间的连接一般依靠权重连接矩阵来描述，它的实质是一个线性变换. 对于本层的每一个神经元，其输入为上一层的所有输出. 实际问题的不同，网络的连接方式也不同，有些网络上一层的输出只是连接到本层的部分单元，这种策略大大减少了参数的数量.

### 2.3.2 网络中的激活函数

比如最简单的网络结构逻辑回归，它在网络的输出层添加了一个Sigmoid函数，主要作用是为网络进入非线性. 这个函数运算通常称为激活函数. 神经网络中当前层的每一个节点能够将前一层的输出结果经过加权求和后，进行一个非线性的激活变化. 当然这个非线性变化是不能去掉的，因为去掉的话，那么整个网络就只剩下线性运算的一个复合计算过程，最终的效果只相当于线性模型，模型会丧失其拟合能力. 对于二分类问题可以用伯努利分布来描述，但是梯度下降算法难以推进. 另一种方案是在输出层使用Sigmoid激活函数，将取值映射为概率值，在多分类的场景下，仅在输出层使用Softmax激活函数，而对于回归问题，在输出层一般不使用激活函数，直接全连接输出即可，必要时要对结果进行线性变换. 一般来说，激活函数的选择都存在于隐藏层.

在神经网络发展的早期，Sigmoid函数因为其优良的特点成为最普遍的激活函数：



Sigmoid函数的特点是当无限小的时候，函数值无限趋近于0；无限增大时，函数值无限趋近于1，但是两端都存在数值饱和的问题，即梯度接近0. 因此，使用梯度下降法优化的时候，当非常大或非常小时，参数每次的变化量也非常小，无法有效地更新参数. 尤其在深度神经网络中，由于需要使用链式法则求导，会面临雅可比矩阵越乘越大或越乘越小的情况，伴随出现梯度消失，因此早期的神经网络层数都比较少. tanh也有相同的问题，不过它的取值范围为[-1,1]，即取值的均值为0，均值不为０就意味着自带了一个bias，在计算时是额外的负担，这会使得收敛变得更慢.

为了解决上述两个激活函数的问题，ReLu函数被提出了[20]，它的定义如下：



ReLu函数在时不可微，当时，对函数求导都为1；当 时，函数的导数为0，虽然这个函数在时不能求梯度，但是与Sigmoid函数的软饱和性相比，ReLu函数的性能更好. 一般来说，训练一个神经网络模型并不会使得损失函数值达到0点，而仅仅是显著地降低损失函数值使其收敛即可. 因此无论是从计算机的计算方式还是实际训练过程中无法到达损失函数梯度为零的点来说，损失函数取最小值时不可导都是可以接受的. 另一方面，ReLu函数在梯度反向传播的过程中，容易计算和优化. 当神经元单元处于激活状态时，导数为常数1；当神经元单元处于非激活状态时，导数为常数0 .

虽然ReLu性能比较优越，但ReLu函数无法通过梯度下降的方法更新那些使得神经元处于非激活状态的参数，即会发生“死神经元”现象，这是一个很大的不足. 在2015年，Djork等人提出了一种新型的ELU[21]激活函数，他们通过实验表明使用ELU直接替换ReLu激活函数可以提高DNN模型的准确度，该函数通过在ReLu函数的负半轴加入一个指数项，有效缓解了“死神经元”问题的发生.

### 2.3.3 正则化参数

神经网络往往会因为模型的容量过大导致过拟合，在对网络进行正则化处理时，一种比较常用的手段就是对网络参数进行范数惩罚，惩罚项附加一个惩罚系数. 参数设置的越大，则参数就越趋紧于零，这就意味着算法模型变得相对简单，越简单的模型越不容易发生过拟合. 这就意味着神经网络中很多神经单元间的连接权重接近0，这些神经元在网络中就不会起作用. 此时大的神经网络就会被稀疏成一个小的网络. 在从零逐渐增加的过程中存在一个合适的值，使得神经网络的拟合效果达到最佳.

L2正则[22]通常被称作岭回归，它是以二范数对参数进行惩罚，主要是用来控制权重的取值，该正则化形式可以使得模型参数更接近零，加入L2正则项后的损失函数为：



其中，为加入正则项后的损失函数，为未加入正则项的损失函数，矩阵是由网络参数构成，超参数决定着模型拟合能力. 若，则模型(2.4)退化成原始模型. 越大则惩罚越严重，模型拟合能力越差，过大易发生欠拟合；太小，则会发生过拟合问题.

在计算梯度时，加入了L2正则项的损失函数其实只是在原损失函数梯度的基础上再加上倍的参数本身：



加入正则化项对于参数更新的影响：在每一步对参数更新之前，会对权重参数乘以一个常数来对权重参数进行收缩. 因此L2 正则化可以起到“权重衰减”的作用. 如果我们想要对数据进行特征选择，L1正则[23]是一个很好的选择. 其表达式如下：



上式中的参数表示的意义与式(2.4)一致.

在计算梯度时，由于绝对值不可导，L1正则化后的损失函数如(2.7)所示：



其中，表示取的符号，如果，则取值为 1；如果，则取值为 -1；如果，则取值为零.

除了以上L1、L2正则这类对参数进行范数惩罚的方法之外，还有很多应用在深度神经网络中的防止过拟合的正则方法. 比如，早停和Dropout等.

当训练一个容量较大的模型时会经常发现：训练集的误差逐渐降低，但是验证集的误差会先下降后上升. 当验证误差没有进一步改善时，算法就提前终止. 这种策略被称作早停. 早停是深度学习中最常用的正则化形式，因为它简单、有效. 当训练终止时，返回的不是最新的模型参数，而是验证集误差最小的模型参数，因此需要频繁存储模型参数.

Dropout方法是在2012年由Hinton G E等人[24]提出的，方法的原理是：在网络的向前传播过程中，网络中每个隐藏层，对应的每个神经元都能按照设定丢弃率被删除，最后得到网络模型的一部分. 在反向传播过程中，仅仅针对丢弃后的网络进行权重更新. 所谓的删除，就是使该神经单元的输出都为. 如果隐神经元的权重为，则该隐神经元就不会参与到后续神经元的计算. dropout可以视作集成了非常多的神经网络的bagging集成模型，该集成模型的基分类器就是所有按照丢弃率删除隐神经元形成的子网络.

### 2.3.4 损失函数和学习率

损失函数是用来评估模型的预测值与真实值之间的差异程度. 另外损失函数也是神经网络中优化的目标函数，神经网络训练或者优化的过程就是最小化损失函数的过程，损失函数越小，说明模型的预测值就越接近真实值，模型的准确性也就越好. 常见的损失函数有交叉熵损失(Crossentropy) 、均方误差损失(MSE)等.

均方误差损失主要使用在对回归问题的研究上，它主要表示神经网络模型的预测值与真实值之间的差异. 从统计的角度看，均方误差是指对参数的估计值与真实值之差的平方求期望. 其表达式为：



MSE数值越小，表明模型的拟合效果越好，当然拟合效果好不一定代表模型好.

交叉熵损失一般用于解决分类问题，其表达的是两类数据分布的一致程度，即输入样本的预测值和真实值的差异. 其表达式为：



其中，代表真实标签，代表预测标签. 交叉熵值越小，表示模型拟合程度越高.

神经网络通常使用梯度下降算法寻找使得损失最小的最优参数，在寻优的过程中需要设置合适的学习率，它代表每次下降的步长，即每次对权重更新时，负梯度项前面的系数. 当学习率设置的很低时，寻参的过程不仅会耗费较长的时间，而且由于不能掌握具体迭代步数，会使模型更容易陷入局部最优解. 而学习率设置的很高，训练过程可能跨过局部最优解甚至不能收敛. 所以训练应当从相对较大的学习率开始，这是因为在开始时，初始的随机权重离最优值较远. 随着模型的训练，权重逼近最优解，学习率应该逐渐减小，以期找到合适的参数组合.

神经网络的参数更新是依靠梯度下降法，根据损失对中间层神经元连接权重的偏导数来更新梯度. 由于只有相邻层的梯度能够直接求导，所以隔层之间是通过链式法则的传递求得梯度.

假设每个神经元的输入用表示，输出用表示，权值用表示. 已知第层第个神经元的输出，是由该神经元的输入经过激活函数得到：



其中，是激活函数；是第层的输入；表示第层的输出. 第层的第个神经元的输入，是由上一层神经元的输出加权得到的值：



第层(输出层)中的第个神经元输出的误差，通过选用的损失函数计算：



通过梯度下降法计算迭代过程中参数的值为：



其中，为学习率. 而根据链式求导法则有：



所以，



求取上式中第层第个单元的梯度，得：



此时，需要分类讨论. 当第层为输出层，即时：



当第层不是输出层，即时：



### 2.3.5 神经网络中的优化方法

想象一个保龄球在光滑表面滚下一个平缓的斜坡：最开始会很慢，但是会迅速恢复动力，直到达到最终速度(假设有一定摩擦力和空气阻力). 这是动量优化的一个很简单的想法. 相比之下，常规梯度下降会沿着斜坡采用常规的小步前进方式，所以会花比较长的时间才能到达底部. 梯度下降并不会关心之前的梯度是多少. 如果当前梯度小，它就会走的慢. 动量优化关注以前的梯度是多少，每一个迭代会给动量m加上梯度与学习率的乘积，同时权重参数要减去这个动量，为了模拟某种摩擦机制并防止动量增长过大，该算法引入了一个新参数，其取值为[0,1]. 动量算法的参数更新过程为：



Adam[25]优化算法是随机梯度下降算法的扩展形式，它可以增快模型收敛速度，它是一种基于一阶梯度来优化损失函数的算法. Adam算法和传统的随机梯度下降不同，随机梯度下降保持恒定学习率(即)更新所有的权重. 而Adam通过计算梯度的一阶矩估计和二阶矩估计而为不同参数设置自适应学习率. Adam算法继承了梯度下降的思想，区别在于每次迭代过程的步长(学习率)都有一个确定的范围，步长不会因为极大的梯度值而导致其本身过大，但随着参数逼近模型最优解决，步长也会逐渐减小甚至使参数陷入局部次优解. Adam算法的参数更新过程为：



此外还有其他的快速优化器，例如AdaGrad算法、RMSProp算法，它们的实现都依赖于一阶偏导数，但Adam在实践中比它们的效果都要好. 在接下来的章节中，将介绍本文所主要使用的研究方法——基于改进优化算法的深度神经网络预测模型.

1. 算法改进与验证

神经网络在发展过程中，经历了3次起伏，这很重要的原因在于神经网络的优缺点在不同时代得以体现. 随着深度学习体系的逐渐完善，深度学习已经被证明在某些复杂的领域性能是强于机器学习方法的，尤其在语音识别[26]、图像处理[27]和自然语言处理[28]等方面取得了很好的效果. 在理论上讲，只包含单层隐藏层的神经网络，可以拟合任何函数，然后这在实际情况中是不常用的，往往采用含多层隐藏层的神经网络来对数据进行拟合. 并且深度神经网络能够更有效地表述实际问题，特别是针对无明确的数学表达式的问题，基于神经网络的预测模型就更凸显优越性[29]. 深度神经网络又可以极好的提取数据间的全局联系，使模型表达更完美. 深度神经网络既可以解决回归问题，也可以解决分类问题. 然而在使用一阶梯度下降算法对模型寻找最优参数的过程中，模型后期的参数收敛过程会陷入震荡，一旦达到迭代的最大步数，模型可能会取得一个次优解，这极大地影响了模型的准确性，所以论文尝试找到一种优化算法来增强模型后期的收敛性能.

## 3.1牛顿法及其优化策略

给定一个具有参数的目标函数，我们想要找到一个使得目标函数取得最大值或最小值. 优化算法就是帮助我们找到这个的算法. 常用的SGD、Adam、RMSProp等基于梯度的优化算法都属于一阶优化算法. 梯度下降法及其变种主要是对恒定learning rate的梯度进行迭代，这几种方法都有一个共同的缺点：对learning rate的设置较为敏感，太小则训练的太慢，太大则容易使目标函数发散掉. 针对不同的参数，learning rate都是一样的. 这对于稀疏数据来说尤为不方便，因为我们更想对那些经常出现的数据采用较小的步长，而对于叫较为罕见的数据采用更大的步长. 最重要的是，使用这些方法对模型求解最后只是在一定范围内收敛，从而产生次优解.

二阶优化算法通过计算目标函数*f*对参数的二阶偏导数来最小化损失函数. 一阶优化算法虽然只需要计算一阶偏导数，计算更容易，但更容易陷入鞍点. 二阶优化算法虽然计算复杂，但是不容易陷入鞍点，且有可用的优化策略来减小计算复杂度. 二阶优化算法从最佳搜索方向和寻求最优迭代步长两方面寻找最优值， 比一阶梯度下降法只是从斜率上寻找最优值更能够求得全局最优解. 二阶优化算法眼光更加长远，所以少走弯路；而梯度下降法只考虑了局部的最优，没有全局思想. 如果更通俗地说的话，比如你想找一条最短的路径走到一个盆地的最底部，梯度下降法每次只从你当前所处位置选一个坡度最大的方向走一步，二阶优化算法在选择方向时，不仅会考虑当前坡度是否够大，还会考虑你走了一步之后，坡度是否会变得更大.

在机器学习中，牛顿法是和梯度下降法地位相当的主要优化算法，它是求解无约束最优化问题的常用方法[30]，有收敛速度快的优点.

牛顿法是迭代算法，但是每一步需要求解目标函数的海赛矩阵的逆矩阵，计算比较复杂.

牛顿法，考虑无约束最优化问题：



其中为目标函数的极小点.

假设具有二阶连续偏导数，若第*k*次迭代值为，则可将在附近进行二阶泰勒展开，展开式如下式：



这里， 是的梯度向量在点的值，是的海赛矩阵：



在点的值. 函数有极值的必要条件是在极值点处一阶导数为0，即梯度向量为0. 特别是当是正定矩阵时，函数的极值有极小值.

牛顿法利用极小点的必要条件：



每次迭代中从点开始，求目标函数的极小点，作为第*k+1*次迭代值. 具体地，假设满足：



由(3.2)有：



其中. 这样，(3.6)成为：



因此，



或者



其中，



用式(3.9)作为迭代公式的算法就是牛顿法.

只要海森矩阵保持正定，牛顿法就能够迭代使用. 但是海森矩阵的特征值并不一定全为正，即既非正定又非负定，这样就会导致损失函数在靠近鞍点附近，参数更新会主动朝着鞍点附近移动. 更严重的是海森矩阵负定，参数更新会沿着完全相反的方向移动. 解决的方法是使用正则化的海森矩阵，即在海森矩阵的对角线上增加常数，于是更新过程如下：



这个正则化策略是牛顿法的近似. 只要海森矩阵的负特征值都在零点附近，则可以起到效果. 当有很强的负曲率存在时（即存在绝对值很大的负的特征值），可能需要特别大的值. 但是如果增大到一定程度，则海森矩阵就变得由对角矩阵主导. 此时，牛顿法选择的方向就是的方向，这就和一阶梯度下降没有区别了，所以对值的选取需要慎重.

针对牛顿法计算海森矩阵的逆矩阵时的庞大的计算量，需要通过一个变换来解决该问题，所以考虑使用LU分解来求解逆矩阵以达到降低计算量的目的.

LU分解主要应用在数值分析中，用来解线性方程、求反矩阵或计算行列式. LU分解(LU Decomposition)是矩阵分解的一种. 本质上，LU分解是高斯消元的一种表达方式，它可以将一个矩阵分解为一个单位下三角矩阵和一个上三角矩阵的乘积（有时是它们和一个置换矩阵的乘积），即, 具体形式如下式：



LU 分解的意义在于，将矩阵的分解与方程的求解分离. 在不少应用场景中，当需要求解的时候，左边的矩阵很多时候是不变的，而右边的随着输入而变化. 做分解时，只会用到矩阵，所以可以预先准备好与，当有求解的需求时，直接拿来用就好了. ，首先解方程，得到，再通过，得到.

将分解为的时间复杂度是，而求解只需要.

当求解海森的逆矩阵时，可以将求解看成求解，这里的I是和海森矩阵相同维度的对角线元素都为1的对角矩阵，通过对进行分解，求得的矩阵即为.

此外，拟牛顿法也是一种优化计算效率的策略. 拟牛顿法的思路是通过差分的方法构造一个*n*阶矩阵来近似代替. 这就是拟牛顿法的基本想法.

先看牛顿法迭代中海赛矩阵满足的条件. 首先，满足以下关系，在式(3.7)中取，即得：



记，，则



或



式(3.14)或式(3.15)称为拟牛顿条件.

BFGS算法是最流行的拟牛顿算法，可以考虑用逼近海赛矩阵的逆矩阵，也可以考虑用逼近海赛矩阵，相应的拟牛顿条件是：



可以用同样的方法得到另一迭代式. 首先令



考虑使和满足：



找出合适的和，得到BFGS算法矩阵的迭代公式：



下面给出BFGS算法具体实现过程.

|  |
| --- |
| 算法3.1： BFGS算法 |
| 输入：目标函数，，精度要求；  输出：的极小点. |
| 算法： |
| 1、选定初始点，取为正定对称矩阵，置*k*=0； |
| 2、计算. 若，则停止计算得近似解；否则转3； |
| 3、由求出； |
| 4、一维搜索：求使得； |
| 5、置； |
| 6、计算，若，则停止计算，得近似解；否则，按(3.19)式算出； |
| 7、置*k*=*k*+1，转3； |

BFGS拟牛顿近似算法虽然免去了计算海森矩阵的烦恼，但是我们仍然需要保存每次迭代的和的历史值. 这依然没有减轻内存负担，要知道我们选用拟牛顿法的初衷就是减小内存占用.

L-BFGS是limited BFGS的缩写. 在BFGS算法中，需要用到一个的矩阵，当N很大时，存储这个矩阵将很耗计算资源. L-BFGS对BFGS算法进行了近似，其基本思想是：不再存储完整的矩阵，而是存储计算过程中的向量序列，需要用到矩阵时，利用向量序列的计算来代替. 而且，向量序列也不是所有的都存，而是固定存最新的m个(参数m可由用户根据自己机器的内存自行指定). 每次计算时，只利用最新的m个. 显然，这样一来，存储由原来的降到了.

接下来，讨论L-BFGS算法的具体实现过程. 出发点是式（3.19），它可以写成下式：



记，，则上式可以写成：



如果给定初始矩阵，则利用上式，可得：



由上式可见，计算时需用到，因此，若从 开始连续的存储m组的话，只能存储到，即只能依次计算 直到. 那如何计算呢，自然地，优先丢弃那些最早生成的向量. 具体来说，计算时，保存，丢掉了；计算，保存，丢掉了；….

但是舍弃掉一些向量后，就只能近似计算了，当时，按照上式，可以构造近似计算公式为：



## 3.2. 基于搜索步长的信赖域算法

如3.1节所述，无论是使用牛顿方法还是L-BFGS方法作为模型的优化方法，当计算完二阶梯度时，都要对最优迭代步长进行选取. 搜索步长的方法主要分为两种，一种是线性搜索，一种是信赖域搜索. 与线性搜索相比，信赖域方法对数值有着更好的收敛性质和数值稳定性. 本节主要说明信赖域搜索步长[31]的方法.

在点，我们想要求解下降方向，但我们不可能求解极小值问题去求解，因为这个问题与原问题复杂程度相同. 解决这个问题的可行方案是：利用 展开公式，在点的邻域中，用的一阶近似函数和二阶近似函数代替求解得到. 记这个函数为. 求这个函数的极小点，然后将其作为迭代方向，即求解



得到，其中



由在处的Taylor展开公式知与的误差为，如果处的邻域太大，就无法保证是的好的近似函数. 当然这个邻域也不能太小，因为邻域的大小决定了步长的长短，太短的步长会增加算法的迭代次数，影响算法的收敛速度. 因此，每步迭代在处选择一个合适的邻域，在这个邻域中求解问题，这就是信赖域方法的思想.

假定在第*k*步迭代已得以及信赖域的半径，则



称为信赖域的子问题，其解依旧记作. 在得到新的迭代点后，我们可以判断是否是下一步迭代的合适的信赖域半径，若不合适，可以修正得下一步的.

下面我们考虑在第*k*步迭代如何修正得到. 假定在点已求得，信赖域半径的修正应该根据处近似的好坏来决定：

从到，的实际减少量为



近似函数的的减少量为



注意到这里，两种减少量之比



的大小反映了近似的程度，当接近1时，表示近似的程度好，下一步迭代应增大；当为接近于0的正数时，表明近似的程度不好，下一步迭代应减小；当为0或者负数时，缩小信赖域的半径，重新求解问题. 下面给出信赖域方法的具体流程.

|  |
| --- |
| 算法3.2： 信赖域搜索算法 |
| 输入：  输出：若终止准则满足，则输出，迭代停止. |
| 算法： |
| 1、求解问题(3.29)得; |
| 2、由(3.30)式计算.  若>0.75，且，且；  若<0.25，则；  否则； |
| 3、若，则;否则，转2； |

信赖域搜索算法和线性搜索算法是求解非线性优化问题的两类主要的数值方法. 信赖域算法从给定的初始解出发，通过逐步迭代，不断改进，直到获得满意的近似最优解为止. 信赖域算法具有可靠性、有效性和很强的收敛性. 与线性搜索算法相比，信赖域方法是直接通过模型求解得到试探步长，而不是先确定搜索的方向，再寻找步长，所以它的效果更加鲁棒.

例如求解：



作出图像如图3.1所示. 如果直接使用线性搜索，初始点在0处无法进行下一步的搜索，因为梯度为0；而使用信赖域搜索的话，可以计算出：, 即仍可以向方向搜索.

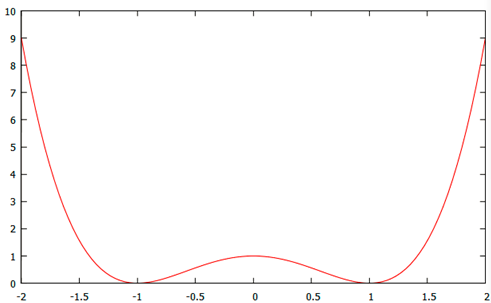


图3.1 函数图像

Fig. 3.1 The Figure of Function

## 3.3 公共数据集验证

本文主要是考虑深度神经网络模型后期数值收敛性问题对结果的影响，这个过程与前期的参数迭代过程无关，所以这里先使用Adam算法配合深度神经网络模型对数据进行大部分的预训练，将数据损失逼近到一个局部凸的区域，再使用通过Adam算法得到的参数对一个新的模型进行参数初始化，最后讨论使用不同的优化算法组合对深度神经网络模型的损失函数进行数值收敛的表现. 根据所得到的损失结果，从时间效率和准确率两个方面找到快速并且有效的优化方法，以解决提出的问题，不同的优化算法组合如下表.

表3.1 不同优化算法组合

Table3.1 Different Combinations of Optimization Algorithms

|  |
| --- |
| 优化方法 |
| Adam |
| 牛顿+LU分解+信赖域搜索 |
| 正则化牛顿+LU分解+信赖域搜素 |
| L-BFGS+信赖域搜索 |

UCI数据库是加州大学欧文分校提出的用于机器学习的数据库，这个数据库目前共有335个公共数据集，数据的数目还在不断增加，UCI数据集是一个常用的标准测试数据集.

为验证在上一节中提出的改进优化算法的深度神经网络的性能，作者选择了UCI数据中关于住宅电器能耗预测的数据进行验证. 该数据集由43824个数据样本组成，数据集为每10分钟采样，持续了约4.5个月. Luis M. Candanedo和Véronique Feldheim等人在2017年使用该数据，建立DNN模型对住宅电器能耗数据集进行预测. 最终在RMSE衡量指标下，训练集和测试集的损失值分别达到17.56和66.65.

本文中采用深度神经网络，使用3.1节中提出的三个优化算法作为优化器来迭代模型时，其中信赖域算法和拟牛顿法的组合效果最佳，其RMSE指标在训练集和预测集分别达到15.20和50.89，低于Luis M. Candanedo等人的结果，迭代过程的损失函数变化曲线如图3.2所示.

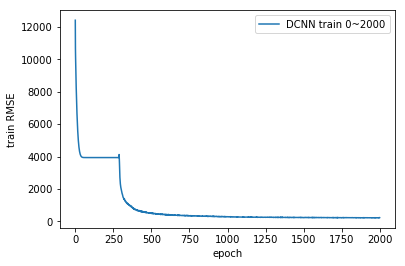


图3.2 损失函数变化曲线

Fig. 3.2 The loss function variation curve

通过公共数据集得出的结果，拟牛顿法和信赖域算法的组合作为优化算法，拟牛顿法可以对模型参数进行更准确的迭代，信赖域算法又可以帮助拟牛顿法得到更精确的步长，能够在一定程度上提高模型性能，所以提出的方法对于模型的准确性的提高是有帮助的.

在下一章中，本文将回归到研究的实际问题中，将第四章中提到的牛顿法、正则化牛顿和拟牛顿法的优化算法应用到深度卷积神经网络，对航空涡轮发动机剩余使用寿命进行预测，在实际问题背景下，检验模型的效果，以达到本课题的研究目的.

1. 改进DCNN在RUL预测中的应用

回归至本文的研究背景，航空涡轮发动机剩余使用寿命，第三章中提出的基于信赖域搜索的牛顿法、正则化牛顿以及拟牛顿法的优化方法将应用于对剩余使用寿命的预测中. 由于这三种方法都是二阶优化方法，因此在本章中将要探索他们实际应用的准确性以及时间效率.

改进优化算法的深度卷积神经网络结构，采用第三章中提到的基于信赖域搜索步长的牛顿法、正则化牛顿以及L-BFGS算法对模型进行后期收敛，论文还会探索卷积层数和时间窗口对模型性能的影响，其他超参数均不作改变. 在改进的模型上面使用航空涡轮发动机数据集进行验证.

## 4.1 数据参数说明

本文中，所使用的数据是NASA 涡轮发动机退化问题数据集. 这个流行的数据集包含四个子数据集，它们都是由基于模型的仿真程序生成的仿真数据，即由NASA开发的商用模块化航空推进系统仿真(C-MAPSS). C-MAPSS数据由21个传感器[32]采集的多变量时间数据组成. 每个子数据集包含一个训练集和一个测试集. 论文主要使用其中的第四个数据集来进行试验. 训练数据集包括在不同运行条件和故障模式下收集的多个航空发动机从运行到故障的传感器记录. 每个发动机单元启动时的初始磨损程度不同，制造过程的变化也不同，这是未知的，但这个状态可以认为是健康的. 随着时间的推移，发动机部件开始退化，直到出现系统故障为止，即最后一个数据样本对应于发动机部件被认为不健康的时间周期. 另一方面，测试数据集中的传感器记录在系统发生故障之前的某个时刻终止，该任务的目标是估计测试数据集中每个引擎的剩余使用寿命. 为了验证，还提供了测试引擎单元的实际RUL值.

表4.1 *C-MAPSS*数据集的信息

Table4.1 The information of *C-MAPSS* dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据 | FOOD1 | FOOD2 | FOOD3 | FOOD4 |
| 训练用发动机 | 100 | 260 | 100 | 249 |
| 测试用发动机 | 100 | 259 | 100 | 248 |
| 训练样本 | 17731 | 48819 | 20820 | 57522 |
| 测试样本 | 100 | 259 | 100 | 248 |
| 故障模式 | 1 | 1 | 2 | 2 |

在本研究中， C-MAPSS数据集的信息如表4.1所示，其中FOOD4训练集总共近6万条数据，测试集总共近248条数据. 在训练过程中，将所有可用的发动机测量数据点作为训练样本，并将每个数据点与其作为目标的RUL标签相关联. 采用分段线性退化模型对每个训练样本进行RUL标记. 在测试过程中，通常使用每个发动机单元最后记录周期对应的一个数据点作为测试样本. 数据集中还提供了测试样本的实际RUL.

## 4.2 数据预处理

### 4.2.1数据归一化

C-MAPSS数据集中的多变量时间数据包含来自21个传感器的发动机单元测量值.

首先，一些传感器的测量值在发动机的使用过程中有恒定的输出，它们并不能提供对发动机剩余使用寿命进行估计的有价值的信息. 因此，通过对数据的各个维度做出的数据分布图，如图4.1可以观察到，在21个传感器中，有14个传感器的测量值可以作为原始输入，其指标对应的列分别为2，3，4，6，7，8，9，11，12，13，14，15，17，20和21，由于方差越小的指标对结果的预测能力越差，所以删除了其他分布比较单一的指标.

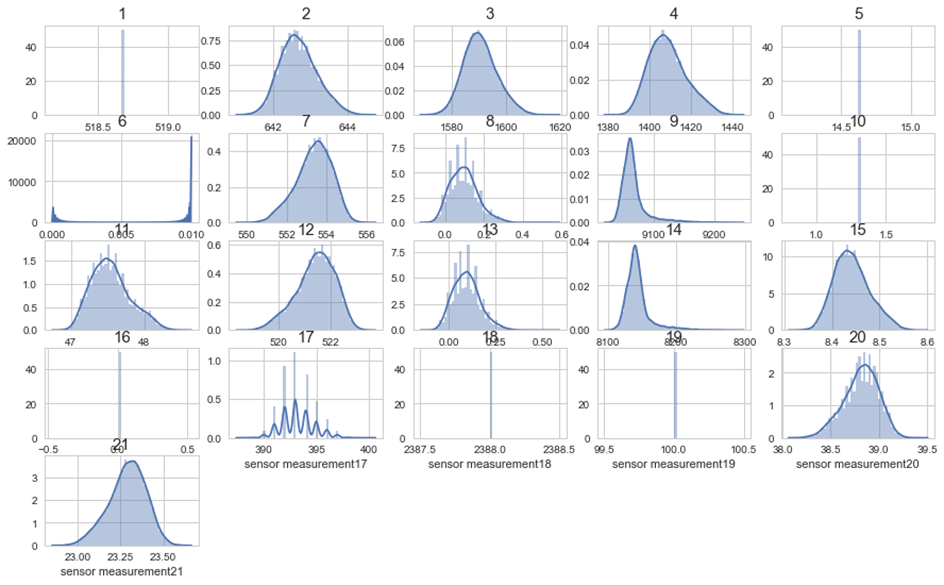


图4.1 RUL数据分布图.

Fig. 4.1 The Distribution Figure of RUL Data.

然后，对于C-MAPSS数据集，采用最小最大归一化方法，将每个传感器采集到的测量数据归一化到[- 1,1]范围内，



其中表示原始数据的第i个数据点的第j个传感器的数据，是的归一化值. 和分别表示原始测量数据中第j个传感器的最大值和最小值.

最后，一般来说，发动机机组在刚开始使用的时候工作是正常的，随着使用的时间增加，才开始出现线性退化，所以可以认为在设备初始阶段有一个常数RUL标签. 根据文献的最新研究，是一个常数RUL值[33]，它可以作为前期数据点的目标标签. 更值得注意的，的大小对模型的预测性能有明显的影响，所以在对剩余使用寿命label进行处理的时候，可按照下式处理.



这里RUL表示真实值，表示规定的常数值.

4.2.2 时间窗口处理

在基于多变量时间序列的RUL估计等问题中，与单步采样的多变量数据点相比，通常可以从时序数据中获得更多的信息. 对时间序列进行细致划分具有较大的潜力，可以提高模型的容量，获得更好的预测性能. 本文利用多变量时间序列信息，采用时间窗口对数据进行处理. 设为时间窗口的大小. 在每个时间步长中，收集时间窗内所有过去的传感器数据，形成一个高维特征向量，作为网络的输入. 时间窗口的大小和卷积层数对网络性能的影响将在4.3.3节中讨论.

## 4.3 DCNN神经网络模型

### 4.3.1模型结构

深度卷积神经网络能够通过多种非线性变换和近似的复杂非线性函数自适应地从原始输入信号中获取表征信息. 一般来说，该深度卷积学习模型由两个子结构组成，第一部分是多个卷积层的叠加，它主要用来提取更加细腻的局部特征，恰当的卷积层和卷积核的选取可以大大提高模型的性能；第二部分是全连接层，通过对卷积层提取的特征进行展开，并连接到合适的神经元个数的全连阶层上，来预测目标值，预测又分为类别预测和回归预测，这主要和全连接层后的功能函数有关系，类别预测在全连接层后会加上一个softmax分类函数，回归预测直接加上一个神经元为1的输出层即可.

首先，输入数据样本采用二维格式，便于卷积运算的应用. 输入的维数为 ，其中为时间序列维数，为所选特征个数. 原始特征通常是通过多个传感器测量得到的.

然后，在网络中叠加4个卷积层进行特征提取. 这4层的结构所使用的个通道数量相同，滤波器大小为. 采用零填充操作，以保持特征图维度不变. 到目前为止，得到的输出是与原始输入样本相同，尺寸为的个特征图. 我们使用另一个带有1个通道的卷积层来将之前的特征图合并为一个唯一的特征图，它的卷积核尺寸为. 这样，就得到了每个原始特征的高级表示. 对于航空发动机数据，首先按照4.2节中的数据预处理方法，对给定的训练集和测试集进行特征归一化处理，接着进行时间窗口处理，具体窗宽在后文讨论.

最后，对二维特征图进行平面化处理，并与全连通层进行连接. 注意，在最后一个feature map上使用了dropout技术，以减轻过度拟合. 最后，在网络的末端附加了一个输出神经元对样本进行预测. 图4.2给出了该模型的结构框图.

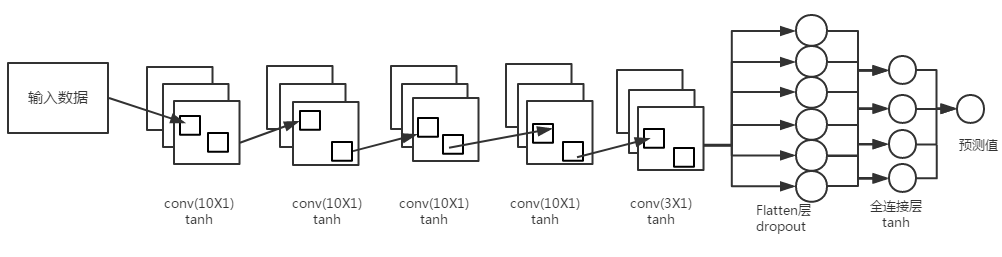


图4.2 剩余使用寿命预测模型结构.

Fig. 4.2 Architecture of Prediction Model for RUL estimation.

所有层都使用*tanh*作为激活函数，权重初始化使用Xavier初始化. 为了进一步提高预测性能，采用反向传播算法对模型参数进行微调，使训练误差达到最小. 模型前期采用Adam迭代器对网络进行优化.

需要指出的是，使用二维卷积神经网络进行特征提取时，卷积运算实际上是在一维中进行的，即每个特征的时间序列维数，可以称它为1d卷积核，其提取特征表示原理图如图4.3所示. 首先使用多层叠加卷积层分别学习每个原始特征的高级表示，然后再使用全连接层学习所有的特征表示进行最终预测. 与现有的深度CNN预测方法相比，该方法从一开始就尝试学习不同特征的空间关系，然后从学习到的多层抽象表示中提取信息，更适合从不同的传感器测量中提取特征.

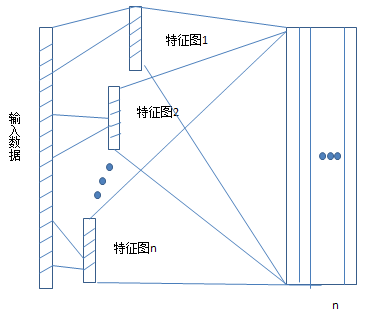


图4.3 1D CNN操作

Fig. 4.3 1D CNN operation.

在本文中，输入的数据是2D格式的，一个维度是特征的数量，另一个维度是每个特征时间窗口的长度，数据具体的细节如上述所示. 然而，考虑到所收集到的机械特征来自不同的传感器，数据样本中可能相邻特征之间的关系并不显著. 因此，当输入和对应的特征图有2个维度时，网络中的卷积滤波器实际上是一维的.

在实际应用中，卷积核的尺寸和特征图的数量两个参数的确定取决于具体任务. 根据对文献30的理解，较大的卷积核尺寸和特征图数目通常会得到较高的预测精度. 然而，计算负担也变得更重. 在实际的案例研究中必须做出权衡. 因此,关于故障诊断研究中, 首选卷积核的尺寸为10，特征图的数量也为10.

通常情况下，卷积层生成的feature map会连接到池化层. 一方面，池化操作能够提取每个feature map中最重要的信息；另一方面，该操作可以显著降低特征维数，即模型参数的个数，从而降低计算量；最后，池化操作也能起到一定防止过拟合的作用. 因此，池化操作非常适合高维数据的问题，如图像处理. 虽然这种操作可以提高计算效率，但在一定程度上过滤了明显有用的信息. 因此，尽管卷积神经网络中普遍使用池化操作，但在原始特征维数相对较低的预测问题中，不建议使用池化操作.

### 4.3.2实验预测程序与结果分析

首先，对C-MAPSS数据集进行预处理，选取14个原始传感器测量值，将相应数据归一化到[-1,1]范围内. 将用来训练和测试的数据集处理成每个样本包含窗宽的时间序列信息. 需要注意的是，标准化数据处理成2D格式后，直接作为模型输入，并不需要人工制作的数据特征，如偏度、峰度等. 因此，该方法不需要预先具备预测和特征处理方面的专业知识. 然后，根据具体的信号处理问题和数据集信息，开始构建用于RUL估计的深度卷积神经网络(DCNN)，确定其网络结构包括隐含层数、卷积滤波器数和长度等. DCNN将归一化的训练数据作为输入，将训练样本的标记RUL值作为网络的目标输出. 反向传播用于学习网络权值的更新. Adam优化算法用于小批量梯度的更新. 对于每个训练周期，将样本运行多个批次，每个批次包含128个样本，并将样本放入模型训练. 根据每个小批量处理的平均损失函数，对网络参数，即各层的权值进行优化. 需要注意的是，批量大小的选择会影响网络的训练性能. 在实验的基础上，确定了128个样本的批次大小，并将其应用于本文的所有案例研究中. 此外，模型还采用了不同的学习速度，在开始的前225个阶段，学习速率为0.001，用于快速优化，保留迭代好的参数. 然后最后25个阶段则使用第三章提到的优化方法进行稳定收敛. 最后，将测试数据样本输入训练后的网络进行RUL估计，得到预测精度.

经过训练后的DCNN神经网络，在使用测试集数据检验网络的预测性能时，本文所采用的衡量指标是网络的训练集与测试集的损失函数值(均方根误差).

RMSE的公式如下:



其中表示模型对样本的预测值，表示真实值，m表示batch的样本量.

RMSE主要是针对回归问题的评价标准，它是观测值与真值偏差的平方和与观测次数 m 比值的平方根，是用来衡量观测值同真值之间的偏差.

深度卷积神经网络的默认参数如表4.2所示.

表4.2该方法的默认参数和实验设置

Table4.2 Default Parameters of Proposed Method and Experimental Setting.

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 数值 | 参数 | 数值 |
|  | 10 | 数据的窗宽 | 15 |
|  | 10 | 全连接层神经元 | 100 |
|  | 14 | 丢弃率 | 0.5 |
|  | 125 | 卷积层数量 | 5 |
| Batch size | 128 | Epoch数量 | 250 |

图4.4给出了测试发动机单元数据使用DCNN模型预测的RUL结果图. 为了更好地观察和分析模型结果，测试发动机单元按从小到大的剩余使用寿命数值进行排序. 可以看出，用该方法预测出的RUL值与实际RUL值基本接近. 尤其在RUL值比较小的区域，模型的预测精度往往较高. 这是因为当涡轮发动机部件接近故障时，故障特征会得到明显增强，这样更利于深度卷积神经来提取局部特征，以便更好地预测结果.

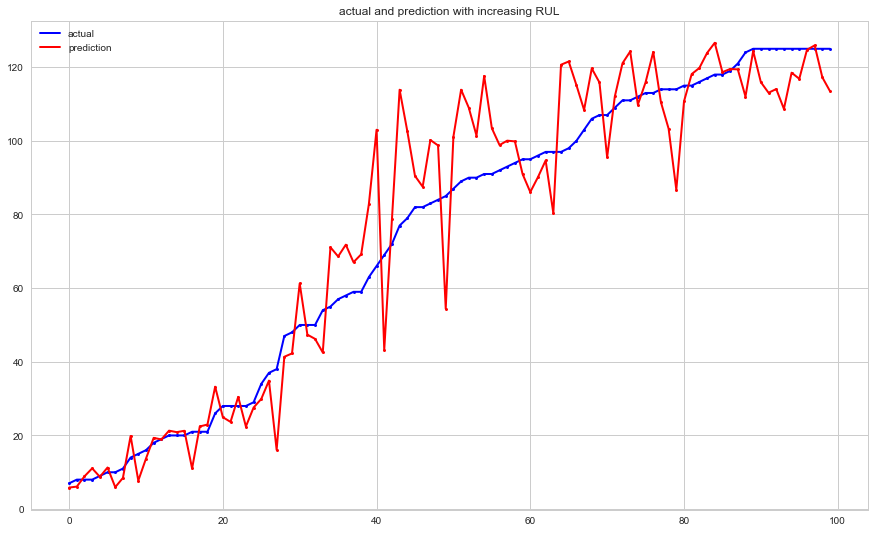


图4.4 RUL预测结果.

Fig. 4.4 The Predict Result of RUL.

具体地说，尽管预测值与真实的RUL值之间存在一些明显的误差，但预测精度很高，特别是当发动机部件接近故障时，对于我们来说 这是有工业价值的，因为在发动机生命周期的后期，健康管理是非常关键的. 对发动机后期状态进行良好的评估，可以提高运行的可靠性和安全性，降低维护成本，提高整个系统的性能.

本文使用的C-MAPSS数据集在预测研究中非常受欢迎，近年来已报道了许多最新的结果. 表4.3总结了C-MAPSS四个子数据集的最新研究成果. 可以看出，许多基于神经网络的预测方法，包括LSTM、CNN等，都显示出了各自的优点. 此外，由于子数据集难度的增加，FD002和FD004的RMSE相对较高. 一方面，主要因为运行状况和故障模式的增多使预测问题更加复杂. 另一方面，随着发动机部件数量的增加，过拟合有很大的机会发生. 这也是本文选择第四个子数据集的原因，因为它不容易预测和优化.

表4.3 C-MAPSS数据集上对DCNN与最新相关论文的性能进行比较.

Table4.3 Performance Comparisons of DCNN and the Latest Related Papers on the C-MAPSS Dataset.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | FD001 | FD002 | FD003 | FD004 |  |
| 支持向量机分类 | 29.82 | *N/A* | *N/A* | *N/A* | 没有使用 |
| 随机森林 | 17.91 | 29.59 | 20.27 | 31.12 | 没有使用 |
| 梯度提升 | 15.67 | 29.09 | 16.84 | 29.01 | 没有使用 |
| DCNN的第一次尝试 | 18.45 | 30.29 | 19.82 | 29.16 | 没有提供 |
| 基于时间窗口的神经网络 | 15.16 | *N/A* | *N/A* | *N/A* | 没有提供 |
| 多目标深度信念网络集成 | 15.04 | 25.05 | 12.51 | 28.66 | 没有使用 |
| LSTM | 12.81 | *N/A* | *N/A* | *N/A* | 125 |
| DCNN | 12.61 | 22.36 | 12.64 | 23.31 | 125 |
| DCNN | 13.32 | 24.86 | 14.02 | 29.44 | 没有使用 |

与现有的学习方法相比，该方法取得了良好的学习效果. 值得注意的是，在本文中，我们人为地提前设置了健康状态的RUL阈值，这对实验性能有明显的影响. 表4.3还提供了没有阈值的预测结果. 尽管无阈值的RUL估计的RMSE比有阈值的有所增加，但该方法的结果仍具有一定的竞争力. 前期的常数值在相应的研究中得到了广泛的应用. 然而，关于阈值的详细信息在有限的论文中进行了讨论. 虽然不同的研究采用了不同的，但是表4.3中给出的结果仍然能够提供该方法的一般比较.

此外，虽然在时间序列处理的许多任务中，通常首选LSTM算法，但和相同深度的网络比其计算量相对较高. DCNN能够实现与LSTM相同水平的预测结果，结构简单，计算量小. 因此，该方法具有良好的预测前景.

第三章中提出的基于信赖域搜索的牛顿法、正则化牛顿以及拟牛顿法的优化算法结合深度卷积神经网络，被应用到对剩余使用寿命的预测中，它们的结果损失和Adam算法的结果损失如下图：

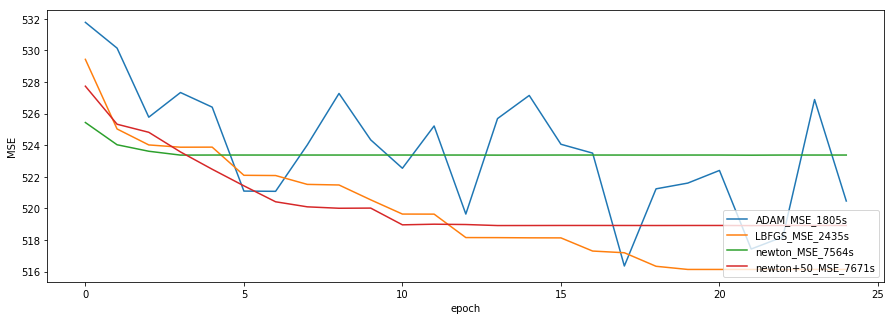


图4.5 不同优化算法的损失变化图

Fig. 4.5 Loss Variation Figure of Different Optimization Algorithms

从图4.5可以看出使用Adam优化算法的损失下降路径是抖动的，这主要与它一阶导性质和恒定学习率有关，其它三个基于信赖域搜索的二阶优化方法都是递减的曲线. 可以看出使用牛顿方法迭代四次损失曲线就趋于平稳，这和它的病态黑塞矩阵是密不可分的，使模型的迭代主动跳入了鞍点，它不能使模型得到一个最优解.

为了解决这个问题，尝试对牛顿法增加正则化项，因值不宜太大，所以其范围取值为[0，100]，以10为一个跨度对模型进行拟合，通过大量实验，最终选定50作为最佳值. 正则化牛顿法的损失比牛顿法可迭代步数更多，下降的更深，虽然由于加了正则化系数，前者开始的损失下降没有后者快，但可以看出随着步数的增加，正则化牛顿迭代到十步最终平稳，且RMSE小于Adam优化算法和牛顿法，可见它能起到一定作用.

L-BFGS算法与Adam优化算法相比，损失下降更平稳，与另外两种二阶优化算法相比，因为是想要找到一个矩阵去近似黑塞矩阵，所以开始没有另外两个的损失下降的快，但是由于L-BFGS更加稳定的性质，它迭代到18轮才开始收敛，且准确率是高于另外三个优化方法的. 再来看时间效率，这四个优化方法，全部迭代25轮，牛顿法是比较耗时的，约花费2个小时，虽然L-BFGS算法没有Adam使用的时间短，但是从表4.4可以看到，达到收敛步数时L-BFGS算法消耗时间是的低于Adam算法的，所以无论从准确率还是时间效率来看，L-BFGS算法都是可取的.

实验证明其能够提高网络的预测能力，使用基于信赖域搜索的L-BFGS优化算法可以使训练集的损失函数值较低，而在测试集对剩余使用寿命的预测中，同样也达到更低的误差值. 实验结果如图4.5所示，数据指标对比结果如表 4.4所示.

表4.4 不同优化算法的深度卷积网络结果对比表

Table4.4 Comparison of results of deep convolutional networks.

|  |  |  |  |
| --- | --- | --- | --- |
| 优化算法 | 训练集 | 测试集 | 收敛时间 |
| Adam | 22.81376475 | 23.43219381 | 1805s |
| 牛顿 | 22.87749986 | 23.53654075 | 1224s |
| 牛顿（正则化50） | 22.78393691 | 23.40385801 | 3061s |
| L-BFGS | 22.71859879 | 23.32444837 | 1753s |

DCNN模型开始通过Adam算法进行梯度下降，更新网络中的参数，当损失下降到一定程度，保存当前网络中的参数，使用保存的参数初始化DCNN神经网络的权重，再使用L-BFGS优化算法对神经网络的损失进行下降. 模型想找到一个合适的学习率会比较麻烦. 如果你把它设置的太高，训练可能会产生分歧. 如果设置的太低，训练可能需要很长的时间才能收敛到最优解. 如果你开始设置得相对高一些，可能开始运行的很快，但是最后一直在最优解附近摇摆不动. 如果计算预算有限，可能必须在训练正确收敛之前停下来，从而生成次优解. 而在其摇摆的时候使用基于信赖域搜索的L-BFGS优化算法，能帮助损失尽快达到最优解点，从而提高模型的预测能力.

再来观察基于L-BFGS优化算法对模型的泛化能力，训练集和测试集的误差损失如图4.6所示. 从图中可以看出测试集的损失伴随着训练集的损失下降而下降，新的网络在测试集和训练集的损失函数值均非常低，二者相差不大. 可以看出模型并没有出现过拟合的现象，且泛化能力较好.

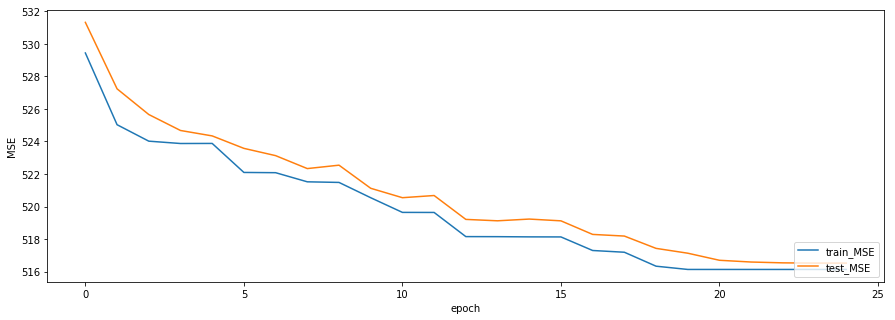


图4.6 不同优化算法的损失变化图

Fig. 4.6 Loss Variation Figure of Different Optimization Algorithms

对比表4.3中的数据可以看出，改进的DCNN模型预测出的数据集的剩余使用寿命的相对误差更低. 综上，实验证明基于L-BFGS优化算法的深度卷积神经网络的确能够达到预期的算法效果，网络训练误差与测试误差，均得到了一定程度的降低.

### 4.3.3卷积层数和时间窗的影响

在本节中，我们研究了该方法中卷积层数以及时间窗口的大小对模型的影响，并以数据集FD004对测试集进行了评价.

图4.7显示了卷积层数对网络预测性能的影响，每个层数对应的RMSE都减了一个基数，这样可以观察到明显差异. 从图中可以看出，一般情况下，隐藏层的卷积层数越多，RMSE值越低. 这表明深层架构比浅层架构能够捕获更多有用的信息.

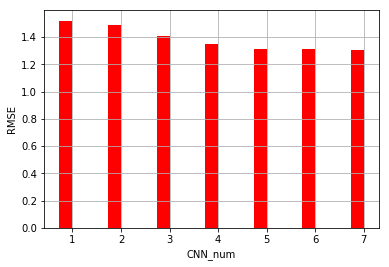


图4.7 不同卷积层数对预测结果的影响

Fig. 4.7 The influence of different convolution layers on the prediction results

另一方面，虽然更深层次的结构可以获得更高的预测精度，但训练过程的计算时间几乎随隐含层数线性增加. 通过观察可以注意到，5个卷积层的网络在计算量适中的情况下性能良好，当卷积层数超过5时虽然模型的效果有提升但是提升空间并不大，所以可以将5作为深度卷积网络结构的默认隐含层数，然后固定卷积层数探究时间窗口对模型的影响.

该方法的另一个重要参数是样本制备的时间窗口大小. 我们在图4.8中给出了时间窗口大小对网络性能的影响，对结果进行可视化时同样也减去了一个基数. 需要注意的是，在测试集中，测试发动机单元记录的数据周期长度不同，其中最短的只有20个周期. 为了对时间窗进行更全面的分析，在相应的情况下，删除了记录周期短于的测试发动机单元.

从图中可以清楚地看出，时间窗口越大，RUL估计越好. 更大的时间窗口可以覆盖更多的原始信息，这是进一步提取特征的基础. 当时间窗口从10增加到15时，RUL估计的损失误差有明显的降低. 当时间窗口超过15时，预测性能没有显著的进一步改善. 与卷积层的数量的影响相似，训练过程的计算量随着时间窗口大小的增大而增大，当时当时间窗口过大时预测性能反而降低，这应该与数据本身的周期性或者时间窗口太长带来的不稳定性有关.

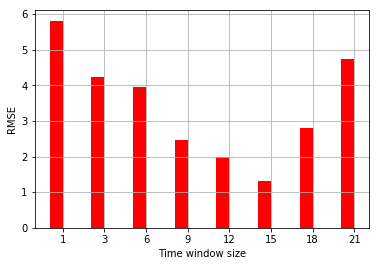


图4.8 不同时间窗口数对预测结果的影响

Fig. 4.8 The influence of different time window on the prediction results

此外，如前所述，当大于20时，所给出的结果不能代表整个测试数据集. 但是，一般显示模式的时间窗口效果可以得到体现. 基于实验和数据集的信息，将= 15作为子数据集FD004的默认设置.

1. 总结与展望

## 5.1 总结

剩余使用寿命是航空涡轮发动机运行过程中重要监控指标，它对于及时更换维修发动机保护生命安全起到了决定性作用. 对于剩余使用寿命的预测，前人工作中或是通过建立设备失效机理的物理模型，从而对设备的剩余寿命进行预测；或是基于数据驱动的方法，基于设备在运行过程中得到的性能监测数据，利用不同的机器学习模型，结合失效阈值，进而对剩余寿命进行预测. 本文使用了航空涡轮发动机数据，在深度学习的背景下，建立深度神经网络模型，对剩余使用寿命进行预测. 由于对神经网络的损失函数进行梯度下降更新参数往往是使用的一阶导数信息，且在模型收敛的后期会导致数值局部收敛. 而牛顿法及其改进算法更符合损失下降的路径，它比一阶梯度下降法只是从斜率上寻找最优值更能够求得全局最优解，同时信赖域算法在进行步长搜索时，与线性搜索相比信赖域方法有着更好的收敛性质和数值稳定性. 因此，本文使用信赖域搜索步长的牛顿法及其改进优化算法对深度卷积网络迭代参数，降低网络预测的相对误差 .

本文的主要工作首先是复现参考论文中的网络结构，按照论文中的参数训练DCNN神经网络预测剩余使用寿命，同时实验效果表明预测结果与文中相差不多，并且预测效果优于以往的方法，和最近的结果相比能够将测试集的预测值和真实值之间的误差至少降低3.14531%. 其次，为了提高模型容量，使用不同的时间窗口和卷积层数对数据进行实验，确定了合适的窗宽和卷积层数. 最后，为防止模型最后一直在最优解附近摇摆不动，从而生成次优解的问题. 本文使用基于信赖域搜索步长的牛顿法、正则化牛顿以及L-BFGS优化算法对模型进行收敛计算，主要步骤为，DCNN模型开始通过Adam算法进行梯度下降，更新网络中的参数，当损失下降到一定程度，保存当前网络中的参数，使用保存的参数初始化DCNN神经网络的权重，再使用二阶优化算法对神经网络的损失进行下降. 基于L-BFGS优化算法的DCNN模型通过UCI数据库中的公共数据集验证，证明该方法对预测准确性最有帮助后，再使用基于信赖域搜索步长的牛顿法、正则化牛顿以及L-BFGS优化算法通过航空涡轮发动机数据进行实验，实验结果表明无论是从准确性还是时间效率方面，L-BFGS算法取得了不错的效果，其对剩余使用寿命的预测性能得到了进一步的提高，误差至少降低了3%.

## 5.2 展望

基于本文的数据集，基于L-BFGS优化算法的DCNN模型的预测性能比未改进前效果更好，且优于传统的基于经验公式的计算方式，但仍然存在不足等待改进.

1.算法的实证研究只在UCI数据集与C-MAPSS数据集上进行了实验，实验数据有限，后续工作可以继续搜集相关数据，持续地在该航空发动机剩余使用寿命背景下进行更深度的研究，以判断算法的稳定性.

2.本文在对剩余使用寿命进行预测过程中，数据由卷积神经网络进行建模的，只考虑了空间上的关系和部分时序关系，考虑为模型结构加入LSTM网络结构提取时序关系.

# 参考文献

1. MOGHADDASS R, ZUO M J. An integrated framework for online diagnostic and prognostic health monitoring using a multistate deterioration process[J]. Reliability Engineering and System Safety, 2014, 124: 92-104.
2. PECHT M , JAAI R. A prognostics and health management roadmap for information and electronics rich systems[J]. Microelectronics Reliability, 2010, 50(3):317-323.
3. SMITH G, SCHROEDER J B, NAVARRO S, et al. Development of a prognostics and health management capability for the Joint Strike Fighter[C]. 1997 IEEE Autotestcon Proceedings AUTOTESTCON, 2002 :676-682.
4. YE Z S, XIE M. Stochastic modelling and analysis of degradation for highly reliable products[J]. Applied Stochastic Models in Business and Industry, 2015, 31(1):16-32.
5. ZIO E, MAIO F D. A data-driven fuzzy approach for predicting the remaining useful life in dynamic failure scenarios of a nuclear system[J]. Reliability Engineering & System Safety, 2013, 95(1):49-57.
6. GEBRAEEL N, PAN J. Prognostic degradation models for computing and updating residual life distributions in a time-varying environment[J]. IEEE Transactions on Reliability, 2008, 57(4):539-550.
7. Pecht M, Gu J. Physics of failure based prognostics for electronic products[J]. Trans Inst Meas Control, 2009, 31(3–4):309–322.
8. Qian Y, Yan R, Gao RX. A multi-time scale approach to remaining useful life prediction in rolling bearing[J]. Mech Syst Signal Process 2017, 83:549–567.
9. Jouin M, Gouriveau R, Hissel D, et al. Particle filter-based prognostics: Review, discussion and perspectives[J]. Mechanical Systems and Signal Processing, 2015, 72-73:2-31.
10. Ben Ali J, Chebel-Morello B, Saidi L, et al. Accurate bearing remaining useful life prediction based on Weibull distribution and artificial neural network[J]. Mechanical Systems and Signal Processing, 2015, 56-57:150-172.
11. Heimes F O. Recurrent neural networks for remaining useful life estimation[C]. Prognostics and Health Management (PHM), 2008 IEEE Conference on. IEEE, 2008.
12. Peng Y, Wang H, Wang J, et al. A modified echo state network based remaining useful life estimation approach[C]. Prognostics and Health Management (PHM), 2012 IEEE Conference on. IEEE, 2012.
13. Louen C, Ding S X, Kandler C. A new framework for remaining useful life estimation using Support Vector Machine classifier[C]. Control and Fault-Tolerant Systems (SysTol), 2013 Conference on. IEEE, 2013.
14. Babu GS, Zhao P, Li X L. Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life[C]. Database systems for advanced applications: 21st international conference, 2016.
15. Lim P, Goh C K, Tan K C. A time window neural network based framework for Remaining Useful Life estimation[C]. International Joint Conference on Neural Networks. IEEE, 2016.
16. Zhang J, Wang P, Yang R, et al. Long short-term memory for machine remaining life prediction[J]. Journal of Manufacturing Systems, 2018, 48:78-86.
17. Wang J, Wen G, Yang S, et al. Remaining Useful Life Estimation in Prognostics Using Deep Bidirectional LSTM Neural Network[C]. 2018 Prognostics and System Health Management Conference, 2018.
18. Li X, Ding Q, Sun J. Remaining useful life estimation in prognostics using deep convolution neural networks [J]. Reliability Engineering & System Safety, 2018, 172: 7-11.
19. 杨云, 杜飞. 深度学习实战[M]. 北京: 清华大学出版社, 2018, 72-113.
20. Hinton G E. Rectified Linear Units Improve Restricted Boltzmann Machines Vinod Nair[C]. International Conference on International Conference on Machine Learning. Omnipress, 2010.
21. Djork-Arné Clevert, Thomas Unterthiner, Sepp Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)[J]. arXiv preprint arXiv: 1511.07289, 2015.
22. Zou H, Hastie T. Regularization and variable selection via the elastic net[J]. Journal of the Royal Statistical Society, 2005, 67(5): 768-768.
23. Tibshirani R. Regression shrinkage and selection via the lasso: A retrospective[J]. Journal of the Royal Statistical Society, 1996, 58(1): 267-288.
24. Srivastava N, Hinton G E, Krizhevsky A, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting[J]. Journal of Machine Learning Research, 2014, 15(1): 1929-1958.
25. Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv: 1412. 6980, 2014.
26. Maas A L, Qi P, Xie Z, et al. Building DNN acoustic models for large vocabulary speech recognition[J]. Computer Speech & Language, 2017, 41: 195-213.
27. Krizhevsky A, Sutskever I, Hinton G E. ImageNet Classification with Deep Convolutional Neural Networks[C]. International Conference on Neural Information Processing Systems. Curran Associates Inc. 2012.
28. Morchid M. Parsimonious Memory Unit for Recurrent Neural Networks with Application to Natural Language Processing[J]. Neurocomputing, 2018, 314: 48-64.
29. 王宜怀, 王林. 基于人工神经网络的非线性回归[J]. 计算机工程与应用, 2004, 40(12): 79-82.
30. 李航. 统计学习方法[M]. 北京: 清华大学出版社, 2012, 219-224.
31. 高立. 数值最优化方法[M]. 北京: 北京大学出版社, 2014, 18-35.
32. Saxena A, Goebel K, Simon D and Eklund N. Damage propagation modeling for aircraft engine run-to-failure simulation[C]. 2008 International Conference on Prognostics and Health Management. CO, 2008.
33. Zhao Z, Bin L, Wang X, Lu W. Remaining useful life prediction of aircraft engine based on degradation pattern learning[J]. Reliability Engineering & System Safety, 2017, 164:74–83.

# 致 谢

感谢我的导师宋叔尼教授，本篇论文的完成离不开宋老师的悉心指导与亲切关怀. 宋老师十分关心论文的进展，多次询问，并在我遇到困惑时，给予精心点拨，为我指点迷津. 他对工作一丝不苟、严谨细致的态度是我努力学习的榜样. 在此谨向宋老师致以诚挚的感谢与敬意.

同时，我也要向研究生求学过程中对我传道、授业、解惑的所有老师致以真诚的感谢，感谢老师们的辛勤付出，是老师们的谆谆教导使我完善人格，丰富内涵，能够以自信、自立、自强的态度走向未来的道路. 学海无涯，学生依旧会带着老师们的教导与嘱托继续前行.

我还要感谢的是给予我帮助与陪伴的学长学姐、同学室友，感谢你们在我需要帮助时伸出援手；感谢你们在每段时光的陪伴，不离不弃；感谢你们给予我的包容与厚爱，鼓励与劝诫.

最后，谢谢家人二十多年以来对我的养育之恩，能够让我在一个和谐幸福的环境中迎接生命中的每时每刻！