|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 分类号 |  | |  | | | 密 级 | |  |
| U D C |  | |  | | | 编 号 | |  |
|  | | | | | | | | |
| 武汉大学logo  **硕 士 学 位 论 文** | | | | | | | | |
|  | | | | | | | | |
|  | | | | | | | | |
|  | | 研究生姓名 | | ： |  | |  | |
| 学号 | | ： |  | |
| 指导教师姓名、职称 | | ： | 何发智 教授 | |
| 专业名称 | | ： | 计算机应用技术 | |
| 研究方向 | | ： | 计算机图形图像处理 | |
|  | |  |  | |
|  | | | | | | | | |
| 二〇一九年四月 | | | | | | | | |
|  | | | | | | | | |
| **April 2019** | | | | | | | | |

论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的研究成果。除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

学位论文作者（签名）：

年 月 日

# 摘要

光线跟

关键字：动态任务调度；光线跟踪；GPU加速；统一计算设备架构；层次包围盒

# Abstract

Ray

**Keywords:**

目 录

[摘要 I](#_Toc4612396)

[Abstract I](#_Toc4612397)

[第1章 绪论 1](#_Toc4612398)

[1.1 研究背景和意义 1](#_Toc4612399)

[1.2分类算法 1](#_Toc4612400)

[1.3 本文的研究思路与内容 1](#_Toc4612401)

[1.4 本文的组织结构 1](#_Toc4612402)

[第2章 卷积神经网络的泛化理论与泛化方法 1](#_Toc4612403)

[泛化 2](#_Toc4612404)

[2.1 Dropout 2](#_Toc4612405)

[2.1.1 dropout 2](#_Toc4612406)

[2.1.2 Fast Dropout 2](#_Toc4612407)

[2.1.3Adaptive Dropout 3](#_Toc4612408)

[2.1.4 Multinomial dropout and evolutional dropout 3](#_Toc4612409)

[2.1.5 Nested dropout 4](#_Toc4612410)

[2.1.6 max pooling dropout 4](#_Toc4612411)

[2.1.7 DropConne 4](#_Toc4612412)

[2.2dropblock算法 7](#_Toc4612413)

[2.2.1 dropblock 7](#_Toc4612414)

[2.2.2 batch normalization（batch normalization ：accelerating deep network training by reducing internal covariate，）（是否插入这篇论文的完整公式表达） 8](#_Toc4612415)

[2.2.3weight decay（deep learning） 9](#_Toc4612416)

[2.3 数据增强技术（1 improving deep learning using Generic Data Aumentation 2 The effectiveness of Data Augmentation in Image Classfication using deep learning） 11](#_Toc4612417)

[2.4调整学习率 12](#_Toc4612418)

[2.5 12](#_Toc4612419)

[2.6本章小结 12](#_Toc4612420)

[第3章 统计学习方法概括 12](#_Toc4612421)

[3.1 统计学习 12](#_Toc4612422)

[3.1.1 统计学习的特点 12](#_Toc4612423)

[3.1.2 统计学习的对象 13](#_Toc4612424)

[3.1.3 统计学习的目的 13](#_Toc4612425)

[3.1.4 统计学习的方法 13](#_Toc4612426)

[3.2 监督学习 14](#_Toc4612427)

[3.2.1基本概念 14](#_Toc4612428)

[问题的形式化 15](#_Toc4612429)

[3.3统计学习三要素 16](#_Toc4612430)

[3.3.1模型 16](#_Toc4612431)

[3.3.2策略 17](#_Toc4612432)

[3.4模型评估与模型选择 19](#_Toc4612433)

[3.4.1训练误差与测试误差 19](#_Toc4612434)

[3.4.2过拟合与模型选择 20](#_Toc4612435)

[3.5.2交叉验证 20](#_Toc4612436)

[3.6泛化能力 21](#_Toc4612437)

[3.6.1 泛化误差 21](#_Toc4612438)

[第4章 我的算法（） 22](#_Toc4612439)

[4.1 算法技术要点 22](#_Toc4612440)

[4.1.1 ResNet残差网络 22](#_Toc4612441)

[4.1.2VGGnet（very deep convolutional networks for large-scale image recognition） 24](#_Toc4612442)

[4.1.3PyTorch包 26](#_Toc4612443)

[4.1.4 MINIST数据集 27](#_Toc4612444)

[4.1.5CiFar数据集 27](#_Toc4612445)

[4.1.6dropblock算法 27](#_Toc4612446)

[4.1.7mixup算法 27](#_Toc4612447)

[4.1.8roialign 27](#_Toc4612448)

[4.2 我的算法的实现 27](#_Toc4612449)

[4.2.1 对中间层feature map裁剪roialign 27](#_Toc4612450)

[4.2.2 dropblock与roialign相结合 27](#_Toc4612451)

[4.2.3 dropblock与mixup结合 27](#_Toc4612452)

[4.2.4 27](#_Toc4612453)

[4.2.5 28](#_Toc4612454)

[4.3 本章小结 28](#_Toc4612455)

[第5章 实验与分析 28](#_Toc4612456)

[5.1 实验平台及参数 28](#_Toc4612457)

[5.2 实验结果 28](#_Toc4612458)

[5.3 实验分析 28](#_Toc4612459)

[5.4 本章小结 28](#_Toc4612460)

[第6章 总结与展望 28](#_Toc4612461)

[6.1 总结 28](#_Toc4612462)

[6.2 展望 28](#_Toc4612463)

[致谢 29](#_Toc4612464)

[附录：攻读硕士学位期间发表学术论文情况 30](#_Toc4612465)

# 第1章 绪论

## 1.1 研究背景和意义

现如今，世界上相机的数量似乎比人类的数量还要多，每天都会有海量的图片和视频产生，人类可以依靠眼睛和大脑准确的分辨图片和视频的类别，可以清晰的解读视频中人或动物的动作，那么对于计算机，是否也可以像人类一样，对图片中的物体进行分类，能够识别不同的物体，在视频中监测物体的动作呢。当然可以，人们刚开始从生物学的角度研究视觉产生的本质，hubel&Wiesel发现了视觉系统的信息处理，他们发现可是皮层是分层的；单个细胞由处理一些视觉信息中最简单的点和边出发，逐步由多细胞构成整个视觉功能的组织、系统和整个视觉信息世界；1982年David MArr（vision）一书的问世，从严谨和长远的角度给出了计算机视觉的发展方向和一些基本算法，其中不乏现在仍为人熟知的图层的概念、边缘提取、三维重建等，这便标志着计算机视觉成为了一门独立科学；计算机视觉的研究内容包括物体检测、物体识别、图像分类、物体定位、图像分割等，大体可以分为物体视觉和空间视觉，物体视觉在于对物体进行精细分类和鉴别，而空间视觉在于确定物体的位置和形状，为动作服务。

图像分类是一个模式分类问题，可以被定义为将图像划分到不同的类别之一的任务，一张图片中是否包含某种物体，对图像进行特征描述是物体分类的主要研究内容，对于单标签的图像分类问题，可以分为跨物种语义级别的图像分类、子类细粒度图像分类、实例级图像这三大类别，图像分类也是计算机视觉中的基本问题，它是计算机视觉中定位、监测和分割等得以实现的基础(Karpathy, 2016)。在图片分类领域，人工神经网络一直是研究的热门，传统的卷积神经网络使用双阶段方法来解决图像分类问题，首先使用特征描述从图像中提取手工制作的图像特征，再用这些提取的手工特征用作可训练分类器的输入，训练分类模型，这种传统方法的主要缺陷是分类任务的准确性在很大程度上取决于特征提取阶段的设计，手工构建的特征需要耗费大量的人力成本，并且不同分类任务所需要的特征又是不同的，这通常被证明是一项艰巨的任务（LeCun，Bottou，Bengio，＆Haffner，1998）。LeCun等人是先行者，他们发表论文，确立了卷积神经网络的现代结构，后来他们又对其进行了完善，他们设计的人工神经网络LeNet-5（handwritten digit recognition with back-propagation network），将反向传播应用到了神经网络上做监督学习，可以对手写数字做分类，

近年来，已经证明利用多个网络层进行非线性信息处理、特征提取和转换以及模式分析和分类的深度学习模型可以克服这些人工提取出的图像特征的种种问题。其中卷积神经网络（CNN）（LeCun，Boser，Denker，Henderson，Hubbard，＆Jackel，1989a，1989b）已成为大多数图像识别、分类和检测任务的领先架构（LeCun，Bengio，＆Hinton，2015）。卷积神经网络在计算机视觉领域的高速发展并站取领先地位，离不开种种技术的助力，其中计算机计算能力的飞速提高为卷积神经网络的发展提供了很大助力，自从199年英伟达公司推出第一款GPU之后，计算机对于图形化运算得到了很大提升，为了充分利用计算机的计算能力，英伟达公司在2006年推出了CUDA这一编程模型，CUDA是一种通用并行计算架构；只是实现了GPU加速不够的，卷积神经网络还需要更大的数据集，Everingham等人在2006至2012年间搭建了一个大型图片数据库，称为PASCAL Visual Object Challenge，李飞飞和她的团队搭建了图像数据库ImageNet，总计两万两千种类别，高达一千四百万张照片；同时还有cifar、svhn、COCO等对于计算机视觉领域同样举足轻重的数据集；最大池化的成功运用，同样加速了CNN的研究。部分研究者将目光投向了改善卷积神经网络的架构本身，在近年来有很多的网络结构被提出，在2012年的ILSVRC竞赛上，Krizhevsky等人使用DCNN在图像分类任务上斩获了第一名的成绩；在2014年，Szegedy等人提出了具有22层网络的GoogLeNet，Simonyan等人提出了具有19层深度的VGG；在2015年，何凯明团队提出的具有152层网络的MSRA获得了ILSVRC的第一名。

基于图像分类任务的卷积神经网络研究进入了高速发展阶段，越来越多的学者将目光投向了深层卷积神经网络，但是深层卷积神经网络可能过度拟合数据，过拟合问题与神经网络的泛化能力密切相关，应该如何解释深层卷积神经网络在图像分类任务上体现出的优异能力，为什么卷积神经网络具有泛化能力，应该如何衡量深层卷积神经网络的泛化能力，如何提高深层卷积神经网络的泛化能力。

## 1.2国内外相关研究工作

深度学习已经取得了显著的效果，对机器学习和人工智能也产生了深远的影响，国内外学者也将目光投向了深度学习的理论研究，对于神经网络的expressivity，认为神经网络具有university[[1]](#endnote-1)，(*Transactions on Information theory)*则论证了相比于人工提取特征而言，神经网络具有指数级别的优势；另外一系列理论研究则考虑如何优化深层假设空间以及揭示卷积神经网络的结构性质，期待发现神经网络如何能够实现非凸优化（Choromanska等，2015; Kawaguchi，2016a）的原因。 然而仅仅具有富有表现性和可训练的假设空间并不能确保模型在预测陌生数据输入的值有良好的表现，这因为可能是因为神经网络过度拟合了训练数据。关于这一现象，我们将其称为神经网络的泛化能力。

卷积神经网络的泛化能力指的是训练后的卷积神经网络对验证数据做出的反应能力，可以说，没有泛化能力的卷积神经网络没有价值，由于泛化能力的重要性，如何提高卷积神经网络的泛化能力逐渐成为国内外的研究热点，泛化误差是模型对于验证集的数据是否能做出准确预测的度量，因为机器学习算法是在有限的训练集上训练的，所以训练后的模型可能对抽样误差敏感，但是对于训练集数据的预测误差的测量可能无法提供关于验证集中新数据的预测能力的大量信息。深层卷积神经网络在学习输入与输出之间的复杂映射时有着优异的表达模型，但是有的时候由于缺乏训练数据，在训练深层卷积神经网络时学习到的样本噪声并不存在于验证数据中，这将导致得到的模型过拟合，使得模型的泛化能力变差，在深度学习中可以用泛化误差的曲线图来表示模型的泛化能力。

一些经典的理论研究者将神经网络的泛化能力归因于使用低容量假设，这种分析基于统计学习理论研究（*Statistical learning theory*，*Foundations of machine learning*）；但是从与小容量相关的紧凑表达观点来看，对于为了表示某些类别的自然目标函数而言（Pascanu等人，2014;Montufar等人，2014;Livni等人，2014;Telgarsky，2016;Poggio等人，2017），与浅层假设空间相比，深层假设空间具有巨大优势。换句话说，当假设空间中隐含一些假设时（例如深目标函数近似地满足分段线性变换的组成）与不依赖于该假设的方法相比，模型可以实现非常好的泛化。但是一个最新的研究论文（Zhang et al。，2017）表明，深度假设空间具有足够可以完全记忆随机标签的能力，这种现象显然使得上述理论具有了明显的悖论，这引起了许多研究人员的积极讨论（Arpit等，2017;Krueger等，2017;Hoffer等，2017;Wu等，2017;Dziugaite和Roy，2017;Dinh等，2017）。（Zhang et al。，2017）以一个公开的问题得出结论，说明理解这些现象需要重新思考神经网络的泛化理论，而（Dinh等人2017）指出，尽管深度学习模型可以很好地泛化，尽管它们具有压倒性的能力，但关于神经网络的泛化能力仍是一个开放的研究领域。

在承认深层卷积神经网络具有强大的记忆力的前提下，应该如何定义神经网络的泛化能力，我们可以通过经验风险最小化理论、结构风险最小化理论和邻域风险最小化理论来确定神经网络的泛化边界（三种理论的引文标注）。（exploring generalization in deep learning）研究了常见的衡量神经网络泛化能力的尺度有norm-based control、sharpness、robustness等，（generalization gap and sharp minima）则提出了在训练神经网络时，过大的batch-size可能会导致神经网络陷入sharp minima的局部鞍点。

对神经网络进行模型复杂度和模型容量控制可以在实验层面有效的预防过拟合，使模型达到良好的泛化，在这一方面，正则化是必不可少的方法，在传统的神经网络研究及其相关的优化算法中，正则化指的是仅仅用于损失函数中的罚项[[[2]](#endnote-2)]，在最近的研究中，正则化指的是修改学习算法，使其降低降低泛化误差而非训练误差[[[3]](#endnote-3)]。正则化是为了旨在使模型获得更好的泛化性能，在测试集上产生更好的结果的补充技术。可以将现有的正则化技术进行分类。

**基于数据的正则化**：模型的泛化能力很大程度上取决于训练数据的数量，除了自发的搜集更多的数据，还可以对已有的数据集进行适当的处理来获得更多的训练数据。一类变换执行特征提取或者预处理，这会改变特征空间和数据分布，另一类变换是对数据集进行处理，以期望得到更大的数据集；通过平移和水平变换、改变颜色通道、对像素点执行PCA等简单的数据增强方式，可以获得更多的训练数据[[4]](#endnote-4)[[5]](#endnote-5)；可以在训练过程中向输入数据加入高斯噪声1，也可以在训练过程中向隐层层特征向量添加高斯噪声[[6]](#endnote-6)；也可以对激活函数的不确定部分添加噪声[[7]](#endnote-7)；batch normalization[[8]](#endnote-8)针对每个隐藏层输出的标准化处理；Label smoothing[[9]](#endnote-9)在输出中添加噪声；GAN[[10]](#endnote-10)可以将图片从一种一个域转换为另一个域，它将一个图像的风格（纹理、氛围、外观），将其与另一个图像的风格混合；基于数据的正则化方法基本上都是在输出层或者隐藏特征层使用随机变换，这会保留数据本身的标签值，基于经验风险理论，生成的数据和原生数据服从同一分布，但都只是接近于真实数据分布，并且有可能使得数据偏离真实数据分布，同时可能会导致数据的不真实等缺点。

**基于网络架构的正则化**：Nitish Srivastava等人在2012年提出dropout的方法，通过在训练时随机将全连接层中的部分节点传递的值置为零来提高CNN的泛化能力，加入dropout后，基于分类任务的CNN模型的泛化能力显著提升（引文），但是会使模型的训练速度变慢；受dropout的启发，谁提出了Fast Dropout，谁提出了Adaptive Dropout，谁提出了Multinomial dropout and evolutional dropout，谁提出了Nested dropout，谁提出了max pooling dropout，谁提出了dropconnect，谁提出了dropblock，谁提出了cutout，谁提出了spatialdropout，。（Understanding data augmentation for classification：when to warp）文章阐述了数据增强在input层和feature map层的使用，dropblock的作者提出了在隐藏层使用规则化dropout的思想，也是基于对feature map层的处理。

**基于误差函数的正则化**：误差函数是卷积网络输出值与真实值之间的差值，它从一定程度上反映了模型的泛化能力，也在一定程度上反映了训练数据对于真实数据的相似程度，Dice coefficiebt optimization表现出对于不均衡类的鲁棒性，在类似的损失函数不同于最小化经验误差理论指导下的损失函数，虽然很少有这种通过对误差函数采用正则化的方法，但这未尝也不是一种合适的尝试，因为在类似于多任务深度学习中，误差函数也应当做适当的变换。

**基于正则化项的正则化**：可以通过向损失函数添加正则化项的方法来达到正则化的效果，正则化项是独立于真实值的，是基于训练数据的分布与真实数据分布间存在一定偏差，为神经网络提供归纳偏差，常见的有权重衰减[[11]](#endnote-11)、frobenius范数、L1范数、与雅克比罚项[[12]](#endnote-12)类似的小方差高斯噪声注入[[13]](#endnote-13)等。

## 1.3 本文的研究思路与内容

本文是旨在对提高基于分类任务的深层卷积神经网络的泛化能力做出的相应研究，我们从dropout中得到灵感，dropout旨在通过对全连接层在训练时加入存活率来提高模型的泛化能力，但是将dropout直接运用于中间的卷积层时效果并不显著，并且dropout存在明显的缺陷，会降低神经网络的学习速率；通过对图像以及卷积神经网络本身的研究，我们发现对于图像分类任务来说，输入的图片像素点之间有很强的空间相关性，如果只是随机丢失某些节点的话，丢失节点的信息也会随着空间相关点传递到下一层，所以本文从这些角度出发，旨在通过在卷积神经网络的卷积层中加入部分处理，从而提高基于图像分类任务的卷积神经网络的泛化能力，我们尝试规则化的丢失一个节点及其邻近的节点，并且继承dropout中神经节点存活率的思想，我们也给予特征图中每个节点一定的存活率，但是我们在丢弃这个节点的同时也会丢弃它邻近的节点；可以将特征图中丢弃部分的值设为零，也可以随机填充进同样数据分布的其他图片的的特征图，这是基于邻域风险理论的思想。我们旨在经典的深层卷积神经网络上进行试验，以验证我们的泛化方法的有效性。

本文的具体工作内容如下：

（1）搜集用于图像识别的数据集。Cifar10、cifar100、SVHN、MINIST等数据集都是广泛运用于图像识别领域的数据集。

（2）实验：用深层卷积神经网络Resnet、VGG训练模型，并完成对比实验。

（3）结果分析：基于衡量模型泛化能力的各类尺度，对加入我们的方法后的试验结果进行分析，与已有的对比实验结果对比。

## 1.4 本文的组织结构

本文的正文部分分为6个章节：

第一章 绪论。介绍基于分类任务的卷积神经网络的研究背景和重要意义，以及国内外对提高神经网络泛化能力的相关研究工作。接着简单说明了本文的主要工作以及本文的组织框架。

第二章 深度学习重要理论的相关知识介绍，研究神经网络各类有效的正则化方法。

第三章 基于分类任务的卷积神经网络。研究提高卷积神经网络泛化能力相关算法， 研究神经网络的主要泛化理论。

第四章 提高基于分类任务的神经网络的泛化能力的规则化dropout方法。介绍基于分类任务卷积神经网络模型，研究了广泛用于图像分类任务的各类卷积神经网络模型，用于实验的数据集收集及数据集介绍，我们的算法的理论支持以及算法推导。

第五章 实验设计与实现。介绍模型训练、性能评价指标以及实验验证和结果分析。

第六章 总结与展望。对论文所做的工作进行总结，并对下一步的工作进行了展望。

# 第2章 卷积神经网络的泛化理论与泛化方法

## 泛化

## 2.1 Dropout

研究表明可以通过正则化来改善由此导致的模型泛化能力变差常用的正则化方法有数据增强、L1正则化、L2正则化、早停法、随机池化等方法；

本文提出的数据增强受到Dropout算法思想的启发，下面介绍基于Dropout而提出的相关研究方向。

#### 2.1.1 dropout

在Dropout中，每一层输出的每个神经节点都有概率p被保留，不被保留的节点输出被设置为零，概率为1-p，0.5是p的常用值。当Dropout应用于深度卷积神经网络的全连接层时，每层的输出，可以表示为

其中表示按元素逐乘，m为二元掩模矢量，输入矢量，W（维数d×n）为权重矩阵，然后是非线性激活函数a。在等式中，二元掩模矢量具有尺寸d，并且每个元素j独立的被伯努利（p）分布绘制，偏差包括在W中，为了简化方程将偏差固定为一。Dropout的主要优势在于其通过验证集所表现的泛化能力，主要方法是通过有效防止特征的共振适应从而很大程度上减少了过拟合，同时它还能够实现模型均化。

#### 2.1.2 Fast Dropout

尽管Dropout具有突出的优点，但在多个模型的实际采样或训练中，Dropout会使训练速度变慢。并且在使用非冗余数据的情况下，当采取不同的数据的采样方式时，有可能会导致训练效率的进一步降低。为了有效的解决上述问题，并且能同时利用Dropout训练的优势，不通过实际采样的方式，而是有效地使用了所有数据，Wang和Manning在2013年提出了Fast Dropout的思想。Fast Dropout训练是通过从高斯近似中采样或与高斯近似相结合来完成的，这种方法可以通过中心极限定理和经验证据来证明。具体而言，当Fast Dropout与常用的softmax结合时，损失可以通过下面的损失函数计算：

其中样本直接取自高斯近似输入，其中，而集合y代表所有可能的预测。Fast Dropout还可以与SVM和Maxout等类似的损损失类型相结合，通过实验证实了Fast Dropout在回归分析、文档分类方面具有显著效果，对于数据集CIFAR-10和MNIST数据的图像分类任务上表现尤为显著，与Dropout相比训练速度也有了提升。但是Fast Dropout在结合反向传播的深度学习训练中存在某些限制。但仍然需要进一步研究以确定其应用于不同DCNN架构时的益处和缺点。

#### 2.1.3Adaptive Dropout

由于在Dropout中使用固定的概率随机丢弃节点，那么在训练中，即使是能够对特征的存在与否进行准确预测的节点也将被丢弃50％（p=0.5）。为了改善这一点，Ba和Frey提出了一个名为Standout的Dropout改进，使用与深度网络共享参数的二元掩码矩阵计算每个隐藏单元的Dropout概率。节点的丢失概率是自适应的，而不是一个固定的参数，并且有别于标准的Dropout，在Standout中节点丢失概率取决于前一层的输入：

在Adaptive Dropout网络中表示从单元i到单元j的权重，值域为0到1，使用这种方法，对特征的存在进行确切预测的单元具有更高的被保留概率，使得在训练集中由于噪声而造成的训练参数有效减少，在提高模型泛化能力的同时，保留模型的学习能力不被损失。

#### 2.1.4 Multinomial dropout and evolutional dropout

断言标准Dropout导致次优收敛，并且对于不同神经元及其相关特征使用非均匀多项式采样概率更合乎逻辑，有学者新提出了multinomial dropout。更具体地说，为了确定最佳的Dropout概率，而不是在dropout中每个节点无相关性的和在初始化网络时固定丢失概率地的原始技术，并证明多项式采样应用于浅层学习系统，他们正式建立了多项式丢失的随机优化的风险界限。这允许他们通过最小化来自风险界限的采样依赖因子来获得依赖于分布的丢失。这种依赖于分布的丢失要求采样概率基于数据分布的二阶统计量。基于这种依赖于多项分布的丢失，他们提出了一种有效但自适应的Dropout版本，称为evolutional dropout，其目的是解决内部协变量偏移的深度学习问题。

2.1.5 Spatial dropout

在使用深度卷积神经网络训练时，有研究者发现在1×1卷积层之前应用常规的Dropout会增加训练时间但不会阻止模型过拟合。因此，他们提出了spatial dropout。具体而言，对于给定的卷积特征张量，它们仅在特征张量中执行Dropout并使用整个特征图来扩展Dropout值。特征图中执行dropout的点的相邻像素全部为零或全部保留。实验结果表明，spatial dropout非常适合于具有少量训练样本的数据集，因此spatial dropout成为减少较小数据集的过拟合问题的很好的解决方法，在仅有较小数据集的卷积神经网络中，加入spatial dropout，能有效的提高模型泛化能力。同时spatial dropout在人体姿势和关节运动估计方面的深度学习训练中表现出特别的效果，但仍需要进一步的工作以进行分类特定的任务。值得关注的是，spatial dropout应用于细粒度分类任务似乎是很有效的，因为该spatial dropout可以恢复在池化操作期间丢失的信息而不会损失通过池化实现的计算收益。

#### 2.1.5 Nested dropout

为了学习其中不同维度具有不同重要程度的数据的有序表示，使得包含在表示的每个维度中的信息根据作为维度索引的预定义的衰减函数而减小，有学者提出nested dropout。nested dropout从几何分布中随机绘制单位索引。它不像在标准dropout中独立地删除具有预定义概率的单位，而是省略了跟随丢失节点的所有单位。更具体地说，对于具有维度K的表示空间，在表示索引子集上定义的分布pB具有如下特征：如果第j个单元出现在特定掩模中，那么所有前面的单元也这样做，从而允许第j个单位依赖它们。因此，当Dropout强制在模型中的每个单独单元上进行分布时，Nested Dropout会在表示单元的嵌套子集上分配分布。

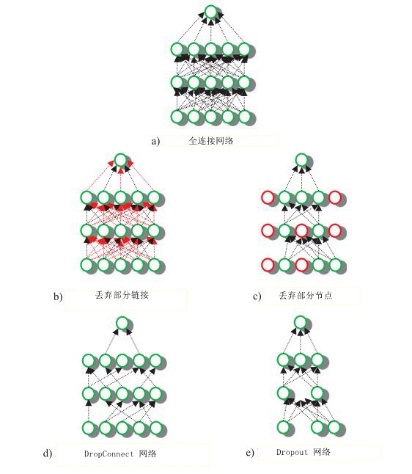
#### 2.1.6 max pooling dropout

Dropout最初设计用于深层架构层中的全连接层，很少运用到神经网络的其他层中。受此启发，相关学者的的实证研究发现，Dropout对深层卷积神经网络中最大池化层的影响相当于在训练时采用基于多项分布的随机选择激活，这在性质上与随机池化相似。因此，为了更准确地近似平均所有可能的Dropout单元，他们提出了使用概率加权池化的方法，而不是常用的最大池化。池化的参数通过的每个特征的所有激活单元来计算：

第l层的池化区域j表示为，p是丢失概率。

**总结**

一种正则化模型的合适方法是采纳平均来自几个不同网络的结果的方法，然而，对于大型深层卷积神经网络，执行这种操作所需的计算资源将是天文数字。所以导致了研究者提出类似于Dropout的正则化思想，Dropout提供了一种以有效方式粗略地合并指数个深层卷积神经网络的方法，当在卷积神经网络中运用Dropout后，在许多网络中取得了许多经验上的成功，这刺激研究人员进一步推进该技术并研究和减轻其不足之处。

为了解决应用标准Dropout后的卷积神经网络的训练不足问题，提出了一种Fast Dropout的方法，Fast Dropout方法能够在训练和验证期间为DCNN提供显著的速度增益，同时能提供更高的稳定性。然而，Fast Dropout方法的缺点是在训练过程的反向传播时的反向传递期更为复杂，而前向传播仍然是直截了当的。减轻这种情况的可能解决方案是为了简单起见，可以在训练是运用具有标准Dropout的DCNN，并且仅在推断期间使用快速丢失以获得计算速度增益；当然，进一步的工作是简化这个结合过程。Standout实施了一项技术措施，以降低出现丢失可用预测特征的单位的风险，但与fast Dropout和Spatial Dropout一样，尽管他们已经在某些网络和深度学习任务中显示了有希望的结果，但是对于其他网络结构和深度学习任务，仍然需要进一步研究，研究重点是关注它们在DCNN处理的图像分类任务中的应用。特别是，在知识转移之后使用它们进行判别性调整是一个需要进一步研究的有趣途径。具有类似于Standout的自适应特性的Evolutional Dropout可以改善DCNN的收敛特性并提高分类性能；但是，与Standout和Max pooling Dropout一样，互补概率计算增加了使用它们的训练模型的计算负担。虽然DropConnect允许在不过载的情况下训练大型模型，但它比使用Dropout或没有Dropout的模型慢很多。

Dropout和dropout的各种改进方法都有着很好的经验结果，仍但是然需要进行进一步的研究在理论上证明它们。Dropout及相关技术可以是降低利用它的模型的计算成本。真正智能深度学习模型本质上是高度适应性的，可预见的是，在未来的研究工作中将与适应性相结合，类似于Standout和Evolutional Dropout。。最重要的是研究者仍然需要在进一步的理论分析中证明dropout成功的原因。特别是，它的泛化能力尚未以可接受的数学精度证明。事实上，它的所有相关改进都将受到这种审问。提到的关于进一步分析的另一个有希望的途径是调查尖峰或随机神经元与Dropout之间的二元性和联系，因为这些可能在学习期间用于实现相同的目标这种模型规则的正则化技术。

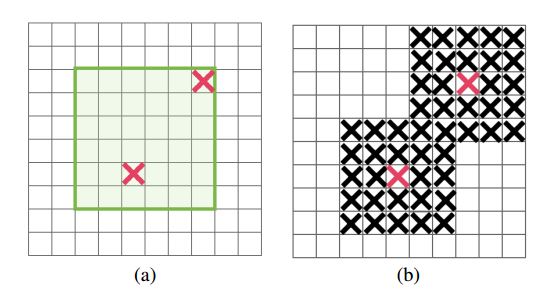
虽然输出正则化技术仍处于研究DCNN的早期阶段，但初步结果已经很令人鼓舞，并且鉴于所引入的技术通常适用于许多其他正则化方法（如Dropout），并且应当针对dropout方法进一步开展深一层的研究工作。

## 2.2dropblock算法

在具有大量参数的卷积神经网络中，常用的提高模型泛化能力的正则化方法是权重衰减和dropout，但是由于dropout主要应用于全连接层，在最近的实验中已经很少用到dropout来提高模型的泛化能力了。

虽然dropout可以随机地让部分节点值不传递到下一层，但这只在全连接层有效，在网络中间的卷积层却不太有效，在计算机视觉当中的分类任务、目标检测里，输入的图像信息是空间相关的，尽管可以随机丢失卷积层中某些特征的值，由于特征的空间相关性，可以使特征随着空间相关的特征继续传递到下一层，所以在卷积层中的dropout需要一种更为结构化的形式来提高dropout的泛化能力。

### 2.2.1 dropblock（得修改，简要介绍它）

DropBlock十分相似于Dropout方法，与Dropout随机无相关的drop节点不一样，DropBlock会丢失特征图上的一片连续区域。

Dropblok的具体算法思想可以描述为下：

1.输入为前一层的特征图，drop\_posibility,block\_size,模式

2.如果模式为FALSE，对特征图不做处理

3.如果模式为true，对特征图做以下处理

根据随机选取丢失节点组成丢失的伯努利矩阵M，扩充矩阵M中值为零的区域，以为中心点，block\_size为尺寸，将此区域值全部设为零。用输入的特征张量A与矩阵M相乘，再将A归一化。

图中（a）表示了丢失点的选取，（b）表示了丢失区域的扩充。

增加的超参数为block\_size和。block\_size表示以随机丢失的点为中心扩展的将要丢失的块的大小。控制将要被丢失的点数量，对于每一个节点它的保留概率为keep\_prob，在dropout中我们将二元掩码矩阵M进行伯努利分布采样的值为1-keep\_prob。但是在DropBlock当中，每一个将被置为零的节点都将被扩充到以此点为中心的面积为的大小，所以我们需要将值设置为合理的值，可以通过下式计算：

feat\_size表示特征图的输入尺寸，由于沿着丢失点扩充丢失的区域大小，会使得丢失区域有一部分重叠，上式只是一个估算丢失节点数的算式，所以在实际的实验中当中我们通过调整keep\_prob来实际调整的大小。

在模型训练过程中，将keep\_prob固定会使得所选择的最优模型效果较差，所以运用线性增加的方式，逐渐的将keep\_prob的值从1减小到目标值，采取这样的策略，使得模型的泛化能力获得了明显的提升。

## 2.2.2 batch normalization（batch normalization ：accelerating deep network training by reducing internal covariate，）（是否插入这篇论文的完整公式表达）

批量标准化指的是每个隐藏层输出的标准化，它作用在大小为m的小的mini-batch ，它是针对隐藏层中的输出所做的线性变换，x是一个d维向量，在将传递到下一层处理之前，对进行处理，可以用下面的公式表示：

其实质是对隐藏层中的某层输出做归一化处理，层k是对一小批N个样本的输出，每个样本具有D个特征，有N\*D的特征矩阵，取矩阵的每一列并减去平均值且除以标准差，以此来规范数据，如果在下一层的输出处重复这个操作，依次内推。

批量标准化解决了一个称为内部协变量偏移的问题，“协变量”是“输入特征”的另一个称谓，通常写作X。协变量移位意味着特征的分布在训练和测试数据的不同部分是不同的，这打破了深度学习中的IID假设（A brief primer on probability distribution），内部协变量偏移指的是在神经网络内部发生的协变量偏移，当网络学习并且更新权重的时候，网络中特定层的输出分布发生变化，这迫使较高层去适应该偏移，这将导致模型学习速度变慢。

批量标准化有助于使网络中间层之间的数据流动看起来像白化数据，不必担心输入特征的规模在层与层之间有极大的不同，因此，梯度下降可以在接近最小点时减少震荡并且收敛的更快，这意味着批量标准化有助于提高网络的学习速率，这可以让我们尽量的减少Dropout的运用，dropout会使得神经网络的学习速率变慢。同时批量标准化会为每个层添加微小噪声，这会一定程度上为神经网络带来正则化的效果， 可以减少其他正则化手段的运用。

## Cutout

## Spatialdropout

## Dropconnect

## 2.2.3weight decay（deep learning）

常见的对权重系数w施加罚项的手段有被称为权重衰减的参数范数惩罚。这个正则化策略是通过向目标函数添加一个正则化项，是权重参数更加容易接近参考点。

假定模型函数的偏置参数为零，它的目标函数如下：

它的梯度可以计算出来：

使用单步梯度下降更新权重：

加入权重衰减后会使得原有的模型函数发生变化，在执行每一步的梯度更新之前，会首先收缩权重向量。当这个变化发生在整个训练过程中的时候，会导致显著减小目标函数方向上的参数保留的相对完好，而在无助于目标函数减小的方向上改变参数不会显著增加速度，这种不重要方向对应的分量会在训练过程中因正则化而衰减掉。

我们以权重系数w对目标函数作二次近似来具体讨论，假设为未采用正则化手段的目标函数在取得最小训练误差时的权重向量。近似的如下：

H是J在处取得的关于Hessian矩阵的值。在处梯度消失，且在处J取得最优值之一，可以看出H为半正定的。

当取得最小时，他的梯度为：

在上式中天健权重衰减的梯度，讨论加入正则化后的，此时的最优点梯度为，

当趋近于零时，正则化后的解将会趋近于，由于H是实对称的，可以由数学推导得知当增大时，权重衰减的效果是沿着H的特征向量所定义的轴缩放

参数范数正则化可以使得模型感知具有较高协方差的输入x，因此与输出目标的协方差较小的特征的权重将会收缩。

**正则化**

除了常见的正则化，还可以选择正则化提高模型的泛化能力，对于权重系数的正则化的定义为：

由定可知正则化为各个参数的绝对值之和。通过正则化对简单线性模型的影响，我们来分析正则化对模型的影响。施加正则化后，简单线性回归模型的目标函数变为：

对应的梯度为：

正则化与正则化的效果大不相同，与正则化相比，正则化正则化会产生更稀疏的解，稀疏性指的是最优解当中的一些参数为零，而正则化并不会使得解变得相对稀疏，

## 2.3 数据增强技术（1 improving deep learning using Generic Data Aumentation 2 The effectiveness of Data Augmentation in Image Classfication using deep learning）

在深度学习中，往往会遇到到的问题就是，数据集的容量有限，在数据缺乏的情况下，训练不良的神经网络会认为某些相同的图像是不同的类别，如：

神经网络可能会认为这三幅图是不同的图像，已经训练好的让模型泛化的更好方法就是使用更多的数据进行训练，在提供的数据集当中我们无法给到足够多的数据去训练，在这种情况下，我们只需要对现有的图片进行微小改动，有人提出了数据增强技术，

数据增强技术可以有效的提升目标是分类的深度学习分类器，常用的传统数据增强技术有：

1. 翻转：在水平或者竖直轴上翻转图像
2. 旋转：将图像旋转任意角度
3. 裁剪：随机地从给定图像裁剪一个部分并调整大小
4. 添加噪声：向给定图像添加高斯噪声
5. 颜色抖动：对图片进行随机颜色处理，调整图像亮度、饱和度、对比度等

在实际的研究中，有的站照片存在季节或者景观特征，类似于这样的自然的数据在多种条件下不能通过传统的数据增强技术生成，有的人提出了名为对抗网络（GAN）的（Unpaired Image-toImage translation using cycle-consistent Adversarial Networks）另一种先进的数据增强技术。GAN可以将图片从一种一个域转换为另一个域，它将一个图像的风格（纹理、氛围、外观），将其与另一个图像的风格混合。

数据增强技术对一个具体的分类问题来说是特别有效的方法，图像是高维的并包含各种巨大的变化因素，其中有许多可以轻易的模拟。即使分类器可以使用卷积和池化技术对部分平移保持不变，更具体地，CNN可以对平移、视点、大小和亮度等保持不变，沿训练图像每个方向平移几个像素的操作往往可以大大改善模型的泛化能力。

## 2.4早停法

## 2.5

## 2.6本章小结

# 第3章 统计学习方法概括

本章简要叙述统计学习方法的一些基本概念。首先介绍统计学习的定义、研究对象与方法；然后叙述监督学习；接着提出统计学习方法的三要素：模型、策略和算法；介绍模型选择，包括正则化、交叉验证与学习的泛化能力；介绍生成模型与判别模型；最后介绍监督学习的应用：分类问题、标注问题与回归问题。

## 3.1 统计学习

### 3.1.1 统计学习的特点

统计学习是关于计算机基于数据构建概率统计模型并运用模型对数据进行预测与分析的一门学科。统计学习也成为统计机器学习。

统计学习的主要特点是：（1）统计学习以计算机及网络为平台，是建立在计算机及网络之上的；（2）统计学习以数据为研究对象，是数据驱动的学科：（3）统计学习的目的是对数据进行预测和分析；（4）统计学习以方法为中心，统计学习方法构建模型并应用模型进行预测和分析；（5）统计学习是概率论、统计学、信息论、计算理论、最优化理论及计算机科学等多个领域的交叉学科，并且在发展中逐步形成独立的理论与方法论。

统计学习就是计算机系统通过运用数据及统计方法提高系统性能的机器学习。现在人们提及机器学习时，通常指的就是统计学习。

### 3.1.2 统计学习的对象

统计学习的对象是数据，它从数据出发，提取数据的特征，抽象出数据的模型，发现数据中的知识，又回到对数据的分析与预测中，作为统计学习的对象，数据是多样，包括存在于计算机及网络上的各种数字、文字、图像、视频、音频数据以及它们的组合。

统计学习关于数据的基本假设是同类数据具有一定的统计规律性，这是统计学习的前提。这里的同类数据是指具有共同性质的数据，例如英文文章、互联网网页、中文文章等。由于他们具有统计规律性，所以可以用概率统计方法来加以处理。比如，可以用随机变量描述数据中的特征，用概率分布描述数据的统计规律。

在统计学习过程中，以变量或变量组表示数据。数据分为连续变量和离散变量表示的类型。

### 3.1.3 统计学习的目的

统计学习用于对数据进行预测和分析，特别是对未知新数据进行预测和分析。对数据的预测可以使计算机更加的智能化，或者说使计算机的某些性能得到提高；对数据的分析可以提高人类的生活质量，促进工业现代化发展。

统计学习对数据的分析与预测是通过构建概率统计模型实现的。统计学习的目标是考虑学习什么样的模型和如何学习模型，一是模型能对数据进行准确的预测与分析，同时也要考虑尽可能的提高学习效率。

### 3.1.4 统计学习的方法

统计学习的方法是基于数据构建统计模型从而对数据进行预测与分析，统计学习由监督学习、非监督学习、半监督学习和强化学习等组成。

卷积神经网络采用的就是典型的监督学习，监督学习的定义可以归纳为：从给定的、有限的、用于学习的训练数据集合出发，假设数据是独立分布产生的；并且假设要学习的模型属于某个函数的集合，成为假设空间；应用某个评价标准，从假设空间中选取一个最优模型，使它对已知训练数据及未知测试数据在给定的评价标准下有最优的预测；最优模型的选取由算法实现。这样，统计学习方法包括模型的假设空间、模型选择的准则以及模型学习的算法，称其为统计学习的三要素，简称为模型、策略与算法。

实现统计学习方法的步骤如下：

1. 得到一个有限的训练数据集合；
2. 确定包含所有可能的模型的假设空间，及学习模型的集合
3. 确定模型选择的准则，即学习的策略；
4. 实现求解最优模型的算法，即学习的算法；
5. 通过学习方法选择最优模型；
6. 利用学习的最优模型对新数据进行预测和分析。

## 3.2 监督学习

统计学习包括监督学习、非监督学习、半监督学习和强化学习。

监督学习的任务是学习一个模型，使模型能对任何给定的输入，对其相应的输入做一个好的预测，计算机的基本操作就是对于一个给定的输入产生对应的输出，所以监督学习是极其重要的统计学习的分支。

#### 3.2.1基本概念

**1输入空间、特征空间与输出空间**

在监督学习中，将输入与输出所有可能取值的集合分别称为输入空间与输出空间。输入与输出空间可以是有限元素的集合，也可以是整个欧式空间。输入空间与输出空间可以是同一个空间，也可以是不同的空间；但通常输出空间远小于输入空间。

每个具体的输入是个实例，通常是由特征向量表示。所有特征向量存在的空间称为特征空间。特征空间内的每一维对应一个特征。有时假设输入空间与特征空间为相同的空间，对它们不予区分；有时假设输入空间与特征空间为不同的空间，将实例从输入空间映射到特征空间。模型实际上都是定义在特征空间的。

在监督学习过程中，将输入与输出看做是定义在特征空间与输出空间上的随机变量的取值。输入、输出变量用大写字母表示，习惯上输入变量写作X，输出变量写作Y。输入、输出变量的值用小写字母表示，输入变量的取值写作x，输出变量的取值写作y。变量可以是标量，也可以是矢量，输入实例x的特征向量记做：

表示x的第i个特征。表示多个输入变量的第i个，即

监督学习从训练数据集合中学习模型，对测试数据进行预测。训练数据由输入特征向量与输出对组成，训练集通常表示为

测试数据也由相应的输入与输出对组成。输入与输出对又称为样本或者样本点。

输入变量X与输出变量Y有不同的类型，可以是连续的，也可以是离散的。根据输入变量、输出变量的不同类型，对预测任务飞鱼不同的名称：输入变量与输出变量均为连续变量的预测问题成为回归问题；输出变量为有限个离散变量的预测问题成为分类问题；输入变量与输出变量均为变量序列的预测问题称为标注问题。

2联合概率分布

监督学习假设输入与输出的随机变量X和Y遵循联合概率分布。表示分布函数，或分部密度函数。联合概率分布的具体定义是未知的，训练数据与测试数据被看作是依据联合概率分布独立同分布产生的。统计学习假设数据存在一定的统计规律，X和Y具有联合概率分布的假设就是监督学习关于数据的基本假设。

**3假设空间**

监督学习的目的在于学习一个由输入到输出的映射，这一映射由模型来表示。模型属于由输入空间到输出空间的映射的集合，这个集合就是假设空间。假设空间的确定意味着学习范围的确定。

监督学习的模型可以是概率模型或非概率模型，由条件分布或决策函数表示，随具体的学习方法而定。

#### 问题的形式化

监督学习利用训练数据学习一个模型，再用模型对测试样本集进行预测，由于在模型训练的过程中需要训练数据集，而训练数据集往往是人工给出的，所以称为监督学习。监督学习分为学习和预测两个过程，由学习系统与预测系统完成，可以用下图描述

首先给定一个数据集

其中，称为样本或者样本点。是输入的观测值，也称为输入或者实例，是输出的观测值，也称为输出。

在监督学习中，假设训练数据与测试数据是依据联合概率分布独立同分布产生的。

在学习过程中，学习系统利用给定的训练数据集，通过学习（或训练）得到一个模型，表示为条件概率分布或决策函数。条件概率分布或决策函数描述输入与输出随机变量之间的映射关系。

在学习过程中，学习系统试图通过训练数据集中的样本带来的信息学习模型。具体来说，对输入，一个具体的模型可以产生一个输出，而训练数据集中对应的输出为，如果这个模型能有很好的预测能力，训练样本输出和模型输出之间的差值就应该足够小。学习系统通过不断地尝试，选取最好的模型，以便对训练数据集有着足够好的预测，同时对于未知的测试数据集也有着更好的泛化能力。

## 3.3统计学习三要素

#### 3.3.1模型

统计学习首要考虑的问题是学习怎么样的模型。在监督学习过程中，模型就是所要学习的条件概率分布或决策函数。模型的假设空间包含所有可能的条件概率分布或决策函数。

假设空间用表示。假设空间可以定义为决策函数的集合

其中，X和Y是定义在输入空间和输出空间上的变量。这时候通常是由一个参数向量决定的函数族：

参数向量取值于n维欧式空间，称为参数空间。

假设空间也可以定义为条件概率的结合

其中，参数X和Y是定义在输入空间和输出空间上的随机变量。这是一个参数向量决定的条件概率分布族

参数向量取值于n维欧式空间，称为参数空间。

#### 3.3.2策略

有了模型的假设空间，统计学习接着需要考虑的是按照怎么样的准则学习或如何去选择最优的模型。统计学习的目标在于从假设空间中选取最优模型。

首先引入损失函数与风险函数的概念。损失函数度量模型一次预测的好坏，风险函数度量平均意义下模型预测的好坏。

**1损失函数和风险函数**

监督学习问题在假设空间中选取模型作为决策函数，对于给定的输入X，由给出相应的输出Y，这个输出的预测值与真实值Y可能一直也可能不一致，用一个损失函数或者代价函数来度量预测错误的程度。损失函数是与Y的非负实值函数，记做。

统计学习中常用的损失函数有以下几种：

（1）0-1损失函数

（2）平方损失函数

（3）绝对损失函数

（4）绝对损失函数或对数似然损失函数

损失函数值越小，模型就越好。由于模型的输入、输出是随机变量，遵循联合分布，所以损失函数的期望是

这是模型上模型关于联合分布的平均意义下的损失，成为风险函数或期望损失。在分类任务中，y的值域为，是阶跃函数；但是在回归任务中，y是实际的数字，是。（Vicinal Risk Minimization）

学习的目标就是选择期望风险最小的模型。由于联合分布是未知的，不能直接运算。实际上，如果知道联合分布，可以从联合分布直接求出条件概率分布，也就不需要学习了。正是因为不知道联合概率分布，所以才需要进行统计学习。这样一来，既要根据期望风险最小学习模型要用到联合分布，另一方面联合分布又是未知的，所以监督学习是一个病态问题。

给定一个训练数据集

模型关于训练集的平均损失称为经验风险或经验损失，记做：

期望风险是模型关于联合分布的期望损失，经验风险是是模型关于训练集样本的平均损失。根据大数定律，当样本趋于无穷时，经验风险是趋近于期望风险的。所以我们可以用经验风险来预估期望风险。但是往往由于训练样本的有限数目，由经验风险预估出的期望风险并不十分理想，需要对计算出的经验风险进行一定程度上的修正，这关系到了监督学习中的经验风险最小化和结构风险最小化这两个基本策略。

**2经验风险最小化和结构风险最小化**

在假设空间中、损失函数和训练数据集一定的情况下，经验风险可由公式确定。经验风险最小化（empirical risk minimization ERM）的策略认为，经验风险最小化的训练模型是最优的模型。所以将求解最优模型问题转化为了求解经验风险最小化问题：

当样本数量足够充足时，通过采纳经验风险最小化的学习策略可以是学习效果得到好的保证，所以经验风险最小化策略在实际实验中被广泛运用。经验风险最小化策略的典型例子之一就是极大似然估计，当模型是条件概率分布，损失函数是对数损失函数，极大似然估计就是经验风险最小化的实际体现。

令人懊恼的是，当没有充足的训练样本时，也就是训练集数量很小时，经验风险最小化学习策略不能使得模型表现出良好效果，将会使得训练出的模型产生“过拟合”现象。

结构风险最小化（structural risk minimization，SRM）是为了防止模型过拟合而提出的学习策略。结构风险最小化也就是我们常说的正则化。结构风险是在经验风险之上附加一系列参数，表示模型复杂度的正则化项或者罚项。在确定的假设空间、训练集数据、损失函数下，结构风险被定义为：

表示的是模型的复杂度，是定义在假设空间上的泛函数，与模型的复杂度成正相关。复杂度表示对复杂模型的惩罚，所以又称为罚项。是复杂度的权重系数，结构风险最小化策略需要经验风险与模型复杂度都取值为最小值。经验表明，采用结构风险最小化的学习策略可以使得模型对于未知的数据具有很好的预测能力，也就是模型有着很好的泛化能力。

结构化风险最小化的典型示例就是常见的最大后验概率估计。结构风险最小化学习策略认为结构风险最小的模型就是所得到的最优模型，所以将求解最优模型问题转化为了求解结构风险最小化问题：

**Vicinal risk minimization（引文）**

经验风险最小化（ERM）相当于通过位于每个样本上的概率密度函数形成的经验分布来最小化损失函数的期望：

但如果概率密度函数和目标函数都是平滑的，那么经验风险最小化就可能不是对期望风险最小化的最佳逼近，假设未知概率密度函数任意点的一个邻域内是平滑的，使得风险最小化的函数也是平滑的，且在任意点的领域内对称，可以对所有输入向量构造其邻域函数，然后再用这些邻域函数构造领域风险函数，通过将概率密度函数替换为点的邻域分布来改进对联合分布，的估计：

我们可以将邻域风险函数定义为：

邻域风险最小化认为邻域风险最小的模型就是最优的模型，用邻域最小风险来估计期望风险，构建VRM可以使用任何联合分布的近似，而不是限制我们必须逐点预估。

**3.3.**3**算法**

算法是指学习模型的具体方法

## 3.4模型评估与模型选择

#### 3.4.1训练误差与测试误差

统计学习的目的是对于训练后的模型能够对于未知的数据也能够具有很好的预测能力，也就是好的泛化能力。采用不同的学习方法会训练出不同的模型，当损失函数给定时，基于损失函数的模型的训练误差与模型的测试误差成为了评判学习方法优劣的标准。

假设通过学习达成的模型可以表示为，模型关于训练数据集的平均损失即为训练误差：

模型关于测试数据集的平均损失即为测试误差：

为训练集，为测试集。

测试误差反应了学习方法对于未知的数据所能达到什么程度预测的能力，在统计学习中极为重要。在统计学习中，又将学习方法对于未知数据的预测能力成为泛化能力。

#### 3.4.2过拟合与模型选择

当假设空间中每个模型的复杂度不相同时，我们就要衡量如何去选择最优模型，我们所选择的模型在理论上与实际上的理想模型应当十分相近，那么所选择的模型与理想模型的参数个数和参数向量应当近乎相同，

但是如果选择的学习方法一味地追求对于训练数据的预测能力，那么训练出的模型就会具有更高的复杂度，甚至可能会高于理想模型，在统计学习中，将这种现象称为过拟合现象，过拟合是指学习时所选择的的模型包含的参数过多，以至于模型虽然能够对于训练集可以做出很好的预测，但是对于未知的数据进行预测时，往往表现出很差的预测能力的现象。

3.5正则化与交叉验证

3.5.1正则化

选择模型的典型方法就是正则化，正则化是结构风险最小策略的实现，其基本思想是在经验风险最小的基础上再加上一个正则化项，正则化项一般是关于模型复杂度的单调递增函数，与模型复杂度正相关。

正则化项有着不同的形式，常见的有正则化、正则化。正则化符合奥克姆剃刀原理。奥卡姆剃刀原理的定义为：在所有可能的模型中，能够很好地预测已知数据并且十分简单的模型才是最好的模型。从贝叶斯估计的角度来看，正则化项对应于模型的先验概率来说。可以假设复杂的模型有较小的先验概率，简单的模型有较大的先验概率。

#### 3.5.2交叉验证

另外的选择模型的典型方法就是交叉验证。

如果给定的样本数据充足，进行模型选择的简单方法是随机地将数据切分成三部分，分别为训练集、验证集、测试集。训练集用来训练模型，验证集用来选择模型，二测试集用于对所应用学习方法的评估。在训练后具有不同复杂度的模型里，从中选出对验证集有最小预测误差的模型。当有足够充足数据量的验证集时，用它来进行模型选择往往很有效。

但是，在许多实际的实验中，实验的数据往往是不充足的。为了选择更好的模型，可以采取交叉验证的模型选择方法。交叉验证的基本思想是重复的使用数据，把给定的数据集进行切分，将数据分为训练集和验证集，反复的应用训练集和验证集将模型进行训练、测试、选择。

1简单交叉验证

简单交叉验证方法的定义是：首先随机地将已给的数据集分为两部分，一部分数据作为训练集，另一部分数据作为测试集。然后用训练集反复进行统计学习训练，得到在各种不同条件下的具有不同参数的模型；继而用验证集数据评价各个模型的测试误差，选出测试误差最小的模型。

2 S折交叉验证

在众多的交叉验证方法中，应用的最多的模型选择方法是S折交叉验证，具体定义是：首先随机地将给定的数据集切分为S个互不相交的大小相同的子集；然后利用S-1个子集的数据训练模型，用剩下的一个子集的数据测试模型；将这一过程中的S种选择方法进行重复，最后选出S次中具有最小测试误差的模型。

3 留一交叉验证

S折交叉验证的特殊情况是S=N，称为留一交叉验证，往往在缺乏数据的情况下使用，N为给定数据的容量。

## 3.6泛化能力

#### 3.6.1 泛化误差

学习方法的泛化能力是指由该学习方法学习到的模型对未知数据的预测能力，是学习方法最重要的本质属性，

研究表明在统计学习中，f（x）基于一些输入数据x预测输出值y的函数。预期的错误，期望误差的表达式为

V代表损失函数，是关于x和y的联合概率分布，在不清楚的情况下，可能用替代，在给定的n个数据中，我们可以通过结算经验误差

泛化误差为期望误差和经验误差间的差值，表达式为：

如果泛化性能很好，泛化误差接近于0。

因为无法计算未知的概率分布，那么也就无法计算泛化误差。相反，统计学习理论中许多问题的目的是约束或在概率上表征的泛化误差：

表征的泛化误差的概率为，泛化误差界限为（称为学习率，一般依赖于和n）。

泛化误差与过拟合问题密切相关。

3.6.2 泛化误差上界

学习方法的泛化能力分析往往是通过研究泛化误差的概率上界进行的，简称为泛化误差上界。通过比较两种学习方法的泛化误差上界来比较学习方法的优劣。泛化误差上界通常具有以下性质：泛化误差上界与样本容量成负相关关系，当样本容量增加时，泛化误差上界趋近于0；泛化误差上界也是假设空间容量的函数，与假设空间容量成正相关关系，假设空间容量越大，模型越为复杂，泛化误差上界就越大。

3.7分类问题

分类是监督学习的核心问题。在监督学习中，当输出变量Y取有限个离散值时，预测问题便成为了一个分类问题。对于分类问题，输入变量可以是离散的，也可以是连续的，监督学习从数据中学习一个分类模型或分类决策函数，称为分类器。分类器对新的输入进行输出的预测，这称之为分类。可能的输出称为类。分类的类别为多个时，称为多分类问题。

分类问题主要分为学习和分类这样两个过程。在学习过程中，根据已知的训练数据利用有效的方法学习一个分类器；在分类过程中，利用学习完成的分类器对新的输入实例进行分类。分类问题可以用图表示

评价分类器性能的指标为分类准确率 ，其定义为：对于给定的测试数据集，分类器正确分类的样本数与总样本数之比。

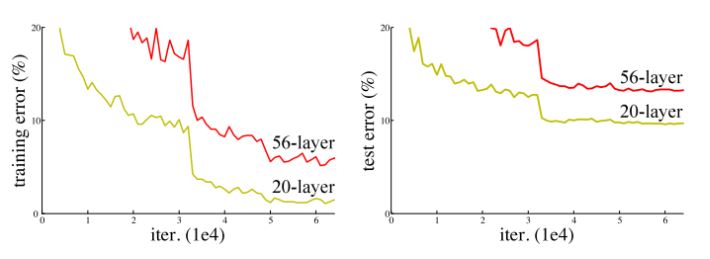
# 第4章 我的算法（）

## 4.1 算法技术要点

### 4.1.1 ResNet残差网络

何凯明团队的Residual Network在2015年的ImageNet上获得了极大地成功，在COCO的目标检测和语义分割获得了第一名的成绩，在ImageNet的分类任务、目标检测和目标定位获得了第一名的成绩。同时也在团队论文Deep Residual for Image Recognition也获得了CVPR2016的最佳论文。

根据通用的逼近理论，单层前馈神经网络在获得足够的容量之后可以有效的实现任何功能，但是这种情况下网络容易变得过于庞大，而且模型容易过度拟合训练数据，为了提高模型的泛化能力，极需要提出更深层的深度学习网络。

自从AlexNet提出以来，最先进的CNN网络结构越来越深入。虽然AlexNet只有5个卷积层，但VGG（Very Deep Convolutional networks for Large-scale image recognition）和Google（Going deeper with convolutions）分别有19层和22层。但是如果只是简单的堆叠网络层数是行不通的，这会导致梯度消失问题，在梯度反向传播的过程中，重复的乘法会导致梯度会变得非常小，单纯的叠加卷积神经网络的深度反而会导致模型的性能变差或者迅速饱和。

在ResNet之前，也有人提出其他的在DCNN中可以解决梯度消失问题的方法，但都不太奏效。

**ResNet的实现**

ResNet的核心是引入一个跳过一个或多个层的“恒等映射”，如下图所示：



有两种恒等映射：

（1）当输入和输出具有相同尺寸时，可以直接使用快捷身份映射而不做任何处理。

（2）当输入与输出尺寸不同时，仍然执行快捷身份映射，用增加的维度参数填充x中与之相比没有的参数，用下面的方式匹配恒等映射x的维度参数。

ResNet表明通过加入identity（本层不做任何处理）层，堆叠网络会降低神经网络的性能，并且与浅层神经网络相比，不会降低分类任务的准确率，与让堆叠层直接与底层映射相适应相比，残差网络可以使堆叠层更容易地和残差映射相适应。

**常见的resnet网络结构**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| layer name | output size | 18-layer | 34-layer | 50-layer | 101-layer | 152-layer |
| conv1 | 112×112 | 7×7,64，stride 2 | | | | |
| conv2\_x | 56×56 | 3×3 max pool，stride 2 | | | | |
|  |  |  |  |  |
| conv3\_x | 28×28 |  |  |  |  |  |
| conv4\_x | 14×14 |  |  |  |  |  |
| conv5\_x | 7×7 |  |  |  |  |  |
|  | 1×1 | Average pool ，1000-d fc，softmax | | | | |
| 参数 |  |  |  |  |  |  |

每个ResNet块都是两层深度或者三层深度。可以用Pytorch实现ResNet的块结构，用Bottleneck类实现3层块，用Basicblock实现2层块。

### 4.1.2VGGnet（very deep convolutional networks for large-scale image recognition）

在2012年的ILSVRC竞赛上，Karen Simonvan等人提出的VGGnet获得了定位和分类的竞赛中分别获得了第一名和第二名的成绩，同时也在其他数据集上也取得了很好的泛化性能。

卷积神经网络在计算机视觉领域表现出了优异的性能，浅层卷积神经网络容易出现过拟合的问题，越来越多的学者将研究转向了如何扩充卷积神经网络的深度，VGG通过使用小的卷积核，添加更多的卷积层来稳定地增加网络的深度。3个的小卷积核所获得的感受野和1个的卷积核所获得的感受野是相同的，而且参数的数量并不会显著增加。（有篇博客 找找引文）

在训练期间，VGG的输入是固定大小为的RGB图像，对其做的预处理是从每个像素点中减去训练集上计算得RGB均值，所使用的卷积核大部分为和，的卷积滤波器可以看做是对输入变量作的线性变换，卷积步长为1个像素；为了保持卷积层的输入在的卷积之后还有足够的空间分辨率，在的卷积层中padding为1个像素，池化层选择的是最大池化。最大池化的卷积核大小为。

在测试阶段将网络中原本的全连接层依次变为1个conv、2个conv，也就是三个卷积层，改变之后，相当于整个网络没有全连接层，网络中间的feature map不会固定，所以在测试阶段网络可以处理任意大小的输入。

具体的网络结构如下图：

VGG的配置如下表：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| vggnet | | | | | |
| A | A-LRN | B | C | D | E |
| 11 layers | 11 layers | 13 layers | 16 layers | 16 layers | 19 layers |
| 输入 （的RGB图像） | | | | | |
| Conv3-64 | Conv3-64  LRN | Conv3-64  Conv3-64 | Conv3-64  Conv3-64 | Conv3-64  Conv3-64 | Conv3-64  Conv3-64 |
| maxpool | | | | | |
| Conv3-128 | Conv3-128 | Conv3-128  Conv3-128 | Conv3-128  Conv3-128 | Conv3-128  Conv3-128 | Conv3-128  Conv3-128 |
| maxpool | | | | | |
| Conv3-256  Conv3-256 | Conv3-256  Conv3-256 | Conv3-256  Conv3-256 | Conv3-256  Conv3-256  Conv1-256 | Conv3-256  Conv3-256  Conv3-256 | Conv3-256  Conv3-256  Conv3-256  Conv3-256 |
| Maxpool | | | | | |
| Conv3-512  Conv3-512 | Conv3-512  Conv3-512 | Conv3-512  Conv3-512 | Conv3-512  Conv3-512  Conv1-512 | Conv3-512  Conv3-512  Conv3-512 | Conv3-512  Conv3-512  Conv3-512  Conv3-512 |
| Maxpool | | | | | |
| Conv3-512  Conv3-512 | Conv3-512  Conv3-512 | Conv3-512  Conv3-512 | Conv3-512  Conv3-512  Conv1-512 | Conv3-512  Conv3-512  Conv3-512 | Conv3-512  Conv3-512  Conv3-512  Conv3-512 |
| Maxpool | | | | | |
| FC-4096 | | | | | |
| FC-4096 | | | | | |
| FC-1000 | | | | | |
| Soft-max | | | | | |

BCDE这后四个网络均使用预训练的模型A的某些层来做参数初始化。

### 4.1.3PyTorch包

**Tensor**

张量是矢量和矩阵的推广，很容易理解为多维数组（deep learning），标量可以看做是零阶张量或零秩张量，矢量是一维或者一阶张量，矩阵是二维或二维张量

可视化张量矢量图

从本质上说张量是一个数据容器，它主要包含数字，可以将张量理解为一堆数字，但是往往容易将张量和多维数组混淆，张量和多维数组是不同类型，张量is a function，矩阵是张量在坐标系中的合适的数据结构，张量在数学上的定义为复合线性函数，一个复合线性函数包含多种矢量变量，所以在使用时，张量是需要定义的函数或者容器。当输入数据时产生计算，张量产生，常见的张量可能是矩阵或者多维数据。

张量的表示与矩阵的表示相似，大写字母表示张量，带有下标的小写字母表示张量内的标量值。原来使用标量、矩阵、矢量的计算都可以用张量来替换。

**Pytorch**

pytorch是基于python的深度学习框架，是facebook在2017年推出的开源深度学习框架，作为NumPy的替代品，Pytorch可以使用GPU的强大功能，使用基于张量的深度学习运算，具有很大的灵活性和计算速度。

pytorch具有很多独特功能：

（1）Pytorch采用张量存储，张量的原始数据并不是立即保存在张量结构中，而是保存在我们称之为“存储”的地方，它是张量结构的一部分，Pytorch可以做到在numpy和Pytorch张量间的自由转换，Pytorch给出了两个函数（from\_numpy()和numpy()），同时也提供各种张量程序，可以做到加速各种科学计算，例如……….（去torch官网看）

（2）Pytorch用一种独特的方法构建神经网络：using and replaying a tape recorder

（3）大多数的深度学习框架都是静态的世界观，人们必须建立一个神经网络并一次又一次地重复使用相同的结构，当希望改变网络结构时必须从头开始。

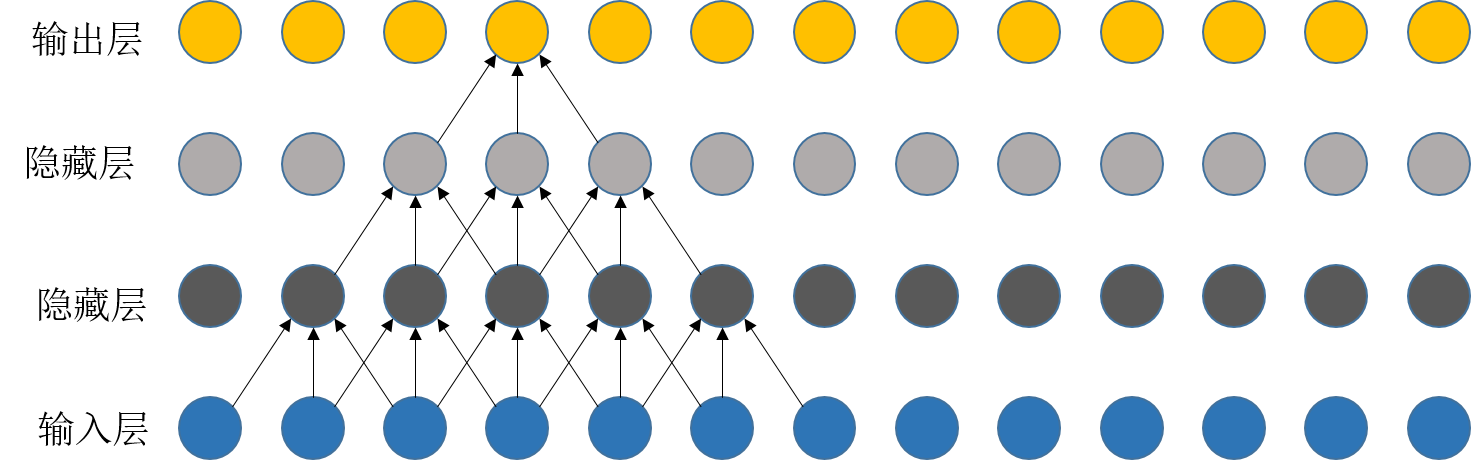
（4）Pytorch使用一种称为自动微分的反向模式的技术，它允许以零滞后或零开销的任意改变神经网络的方式，但这这并不是Pytorch独有的技术。

（5）Pytorch具有最小的框架开销，它集成了英特尔MKL和NVIDIA等加速库，最大限度的提高运算速度，核心是CPU和GPU的张量计算，他的后端是编写为带有C99 API的独立库。

（6）Pytorch与其他深度学习框架相比，它对内存的使用效率很高，Pytorch的创建者为GPU编写了自定义内存分配器，可以确保深度学习模型具有最大的内存效率，这使得研究者可以训练比以前更深的深度学习模型。

### 4.1.4 特征图感受野

**感受野与有效感受野**

**（Receptive fields of single neurones in the cat's striate cortex.）**提出了猫纹状皮层单个神经元的感受野，这是感受野这一概念首次被提出来，感受野（RF）被定义为卷积神经网络特征所能看到输入图像的区域，换句话说特征输出受感受野区域内的像素点的影响。如下图所示：

在深度为四层的卷积神经网络中，每一层卷积层卷积核的大小为3，步长为1，那么输出层一个节点对应输出层的感受野为。

**计算方式**

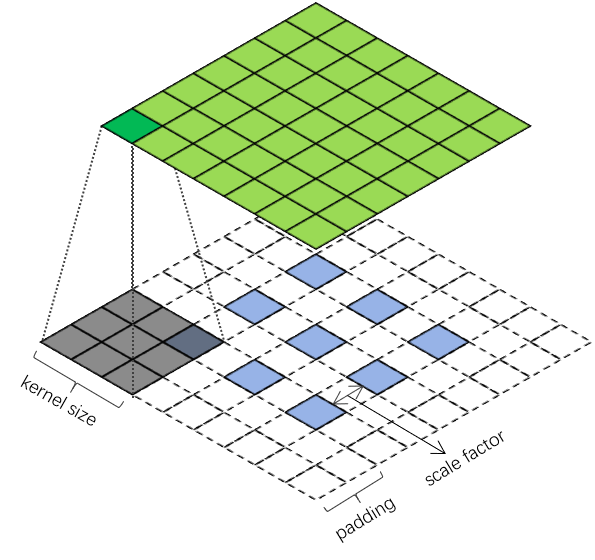
其中*RF*表示特征感受野大小，*l*表示所属层数，

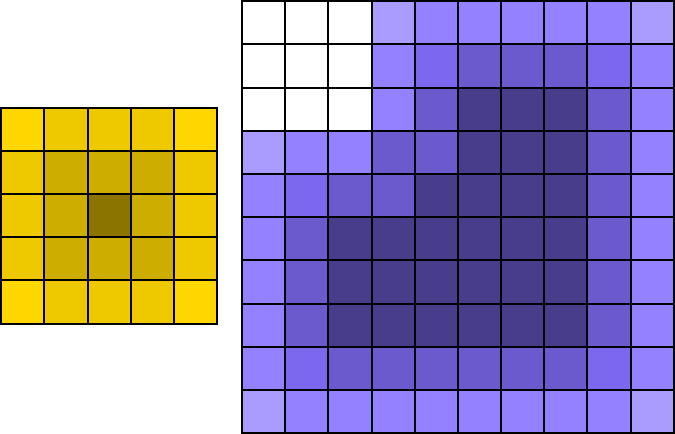
可以计算出从上到下感受野依次为3、5、7。可以看出来，神经网络中某一层特征图上某个位置的特征向量，是由前一层某些固定区域的输入计算出来的，这个区域便称为这个位置的感受野，任意两层之间都有这种对应关系，但在神经网络中，我们一班将特征图中某一位置对应输入图像的区域称为感受野。

在分类任务中，感受野越大越好，最后卷积层的感受野大于输入图像会比较好，网络深度越深感受野越大；

**有效感受野**

在（visilzing and understanding CN）中，介绍了神经网络的可视化细节，通过逆卷积将神经网络每一层所学习到的具体特征做了总结。

（S**tructured receptive fields in cnns**.）提出了深层卷积神经网络当中的感受野结构，在完全连接的网络中，每个单元的值取决于网络的整个输入，而卷积网络中的一个单元仅取决于输入的一个区域。输入中的该区域是该单元的接收场。感受野的概念对于理解和评价深度CNN的工作非常重要。由于在单元的接收场之外的输入图像中的任何地方都不影响该单元的值，因此有必要仔细地控制感受野，以确保其覆盖整个相关图像区域。对输入图像中的每个像素进行预测时，每个输出像素都必须有一个大的感受野，这样在进行预测时就不会遗漏重要的信息。

《Understanding the Effective Receptive Field in Deep Convolutional Neural Networks 》，深层卷积神经网络中感受野的这种结构化表示是符合高斯分布的，这种假设是在没有正当理由的情况下使用的。在我们下图的示例中，第一层卷积的卷积核为，从内往外，每个输入值的使用次数依次减少，很明显越靠近感受野中心的值被使用次数越多，越靠近边缘的值使用次数越少，输入是刚好符合高斯分布的特殊情况，输入时所有值的使用次数都是一，大于输入时大部分位于中心区域的值的使用次数都是9，边缘衰减到1，每层卷积都会有这种规律，经过多层卷积的堆叠，总体上感受野就会呈现高斯分布。发现并非感受野中的所有像素对输出单元的响应都有同样的贡献。直观地说，很容易看出，位于接收场中心的像素对输出的影响要大得多。在前向传播中，中心像素可以通过许多不同的路径将信息传播到输出，而在接收场的外部区域中的像素具有非常少的路径来传播其影响。在反向传播中，来自输出单元的梯度在所有路径上传播，因此来自该输出的梯度的中心像素具有大得多的幅度。该文章发展的有效感受野理论也与一些经验观察有很好的相关性。一个这样的经验观察是，目前常用的随机初始化导致一些深层CNN从一个小的有效感受野开始，然后在训练期间增长。这可能表示初始化偏差很大。

当卷积核权重均匀分布和随机分布时，有效感受野服从（近似服从）Gaussian分布；激活函数ReLU的加入也会使得有效感受野服从（近似服从）Gaussian分布。dropout的加入不会改变有效感受野的Gaussian分布；下采样和dilation 会增加有效感受野的大小，下采样效果更明显。使用skip-connections会减小有效感受野的大小。

随机丢失深层卷积神经网络中的节点，实际上也就是损失一部分图像的细节，因为图像中的像素点具有空间相关性，并且由于共享卷积核的参数，隐藏层中每个节点都有与其相关的感受野。

### 4.1.5空间相关性

图像具有空间连续性（the structed image），在人类视觉系统中仅将投射到特定序列上的局部变化置为有效信息，那么卷积神经网络其实有很大一部分属性是不必要学习的。（visualizing and understanding CN）及图证实了在CNN中通过卷积学习到的特征是局部相关的。

### 4.1.7mixup算法

Mixup数据增强

3.3.1 大型深度神经网络的缺陷

深度神经网络的优势在于全面的提取特征。进年来深度神经网络在强大越发明显。但也伴随着一些不足之处，如记忆性和对于对抗性数据的敏感性，这就导致了大的泛化误差。

只要深度神经网络的参数个数超过训练点的个数就可以获得好的效果，深度神经网络就具有了足够的容量记忆整个数据集。[thinking generliztion]使用随机标签替换真实标签训练，发现用于训练的几个标准网络也能很好的拟合跟换了标签的数据，而且仅在数次迭代之后便开始拟合。

深度神经网络对于输入数据在隐藏空间中的限制很小，在训练时会认为训练数据的分布与测试数据的分布是相同的，也就是两种在隐藏空间中都属于相同的区域。[manifold mixup] 只要测试数据与训练数据服从同一分布，深度神经网络测试时表现出好的泛化性。反之，出现对抗性样本时，容易给出错误的结果，且置信度很高，因为两种数据在隐藏空间中属于不同的局部区域。

3.3.2 ERM准则的替代方案

在SRM准则中正则化是有效的排除不必要的方法，但在深度学习中却起了不同的作用。显式正则化可以改善泛化性，但既不必要也不能有效控制泛化误差。没有正则化也能的很好泛化。[thinking generliztion]对显式和隐式正则化器的观察结果一致表明正则化器在适当调整后可以帮助改善泛化性能。然而，正规化器不太可能是泛化的根本原因，因为在所有正规化器移除后，网络继续表现良好。

ERM准则无法提供对于对抗性测试样本的泛化能力。即便是测试数据的分布与训练数据的分布只有轻微的不同之处。ERM准则也使得神经网络即使是在有正则化的情况下记忆训练数据，只要其参数数量超个训练数据数量，就具有了记忆所有训练数据的能力。

VRM准则可以是一种替代方案。VRM准则用函数将点替换为领域点，使用改进的密度估计进行训练。这就给原本在隐藏空间中位于两块不同局部区域的数据建立起了桥梁，将它们转换到同一区域。而且，训练的成功不仅取决于对于的近似，也可以由空间，即网络的参数数量决定。这种双重保证可以改善在实际训练中，输入腐蚀标签的或者是对抗性的数据所导致的泛化误差，提升网络的泛化性。

3.3.3 使用Mixup技术

VRM要求用一个优秀合成的分布来近似实际的分布。这时候就需要用到人类的知识来利用现有的训练数据合成出新的领域数据，以扩大对于训练分布的支持，通常被称为数据增强。一般的数据增强包括对图像进行翻转、缩放、旋转等。数据增强可以改善深度神经网络的泛化性，但是这些操作都默认增强的数据和原数据服从同一分布，在隐藏空间中属于相同的局部空间。

一般的数据增强操作不会使两个不同类间数据建立联系。为了更好的泛化性，势必要对隐藏空间中不同分布的数据进行约束，使得大多数数据最后都服从同一个分布，并且都是随机数据。[mixup]基于这些考虑来构造领域样本：

(9)

(10)

其中(是随机从训练集中选出的两个数据的特征向量，是由分布决定的一个混合系数。

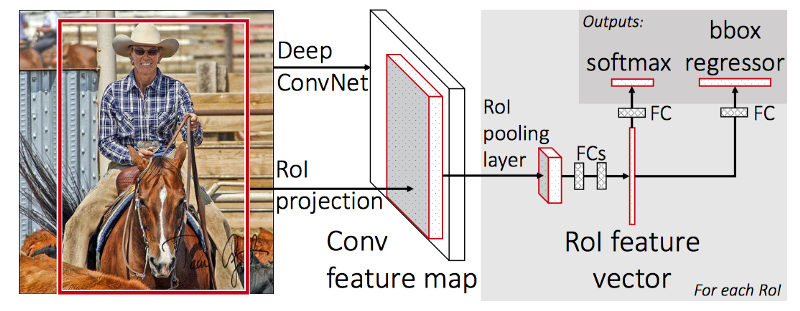
这种线性插值的方法可以将两个分属不同类的数据联系起来，这样在预测训练样本之外的数据时的会减少不良震荡，取得更好的泛化性。生成系数以及最后做插值引入的额外计算负担很小。线性插值也符合奥卡姆剃刀定律：“如无必要，勿增实体”的要求。

在深度神经网络中使用Mixup非常简单，只需将某一层的输出插值之后作为下一层的输入即可。

### 4.1.8roialign

（Mask R-CNN）（Fast R-CNN）

在Fast R-CNN中提出了ROI-pooling的概念

图

基于目标检测的边界框与特征图中的ROI（region of interest）框相互对应，Roi指的是一张图片中认为存在检测目标的区域，一般Roi的输入特征图大小不确定，输出的特征图的大小固定。

对Roi的最大池化操作的卷积核的大小A=多层卷积后的特征图的大小除以得到的特征图的大小，A存在不为整数的情况。常用的处理方法有以下两种

ROI pooling图

采用最近邻插值法：从左至右对多层卷积后的特征图进行最大池化操作，丢弃特征图靠右的值。

得到的ROI框可能是浮点数，进行取整会损失一定的精度，在ROI池化过程中也可能存在浮点数取整的情况，又会损失一定的精度，两次取整后会使得候选框的位置出现偏差，**会损失空间对称性**，最终会导致模型的错误率增加。

**ROI Align**

针对ROI池化的缺点，研究者提出了ROI Align这一改进的方法，ROI Align的思路为：取消量化操作，使用双线性内插的方法获得坐标点为浮点数的像素点上的图像数值，从而将整个特征聚集过程转化为一个连续的操作。

Roi Align图

具体计算图

当A的大小不为整数时，假定采样点为4，将每个区域平分为四份，每一份取中心点位置，中心点位置的像素，采用双线性插值法进行计算。

**双线性插值**

在数学上，双线性插值是有两个变量的插值函数的线性插值扩展，其核心思想是在两个方向分别进行一次线性插值。如图

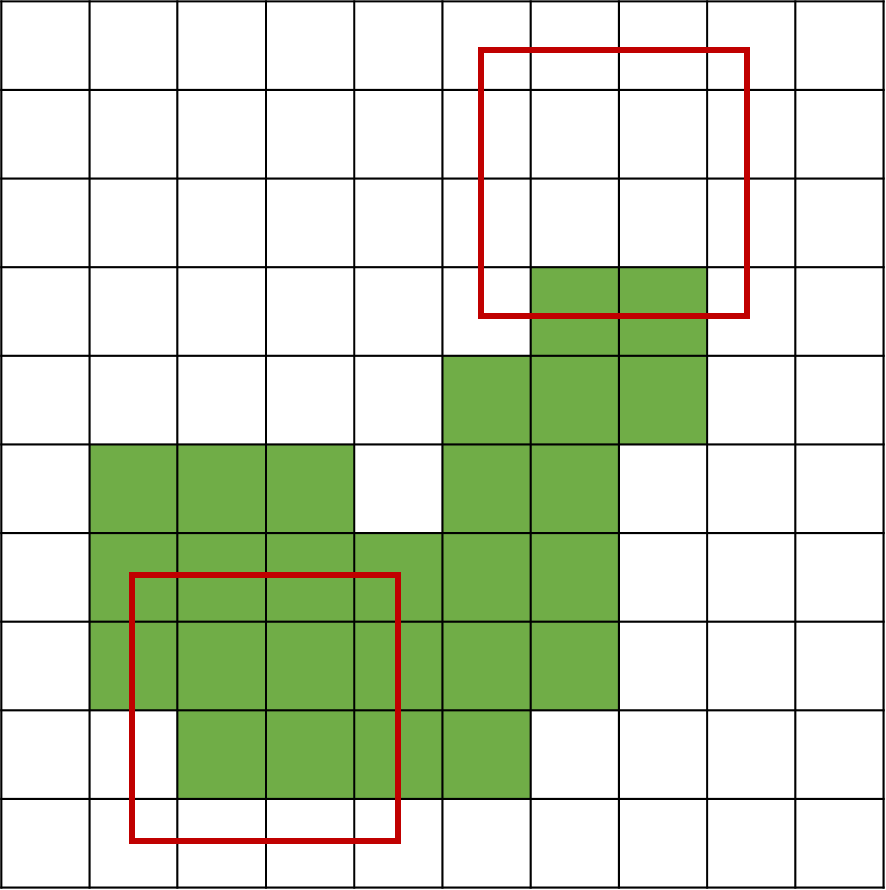
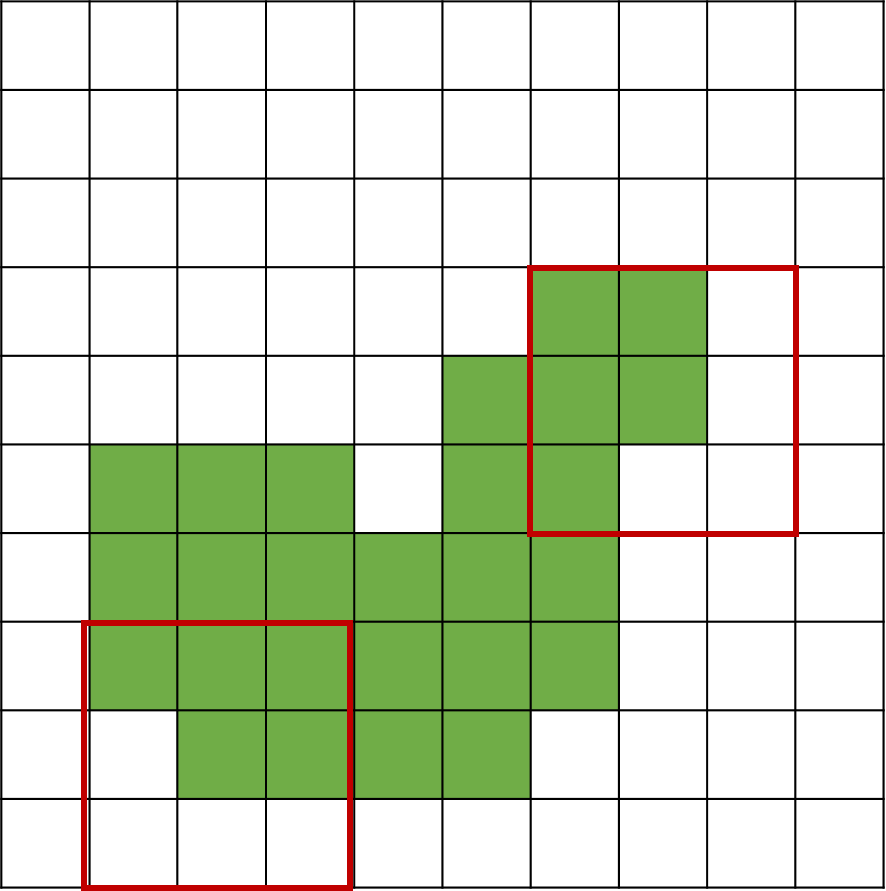


P点的像素值由它附近四个采样点经过双线性插值得到。

## 4.2 我的算法的实现

### 4.2.1 dropblock与roialign相结合层

DropblockAlign与dropblock相似，dropblock随机置零的特征图中的块为完整的框，但是随机选取的丢失中心点可能不位于一张特征图的整节点上，当采样点位于两个具有具体值的特征点之间时，原来的操作方法不再具有效果，采样点以及以采样点为中心的丢失框的示意图如下：



吹一下非规则化取drop，

**随机选取丢失框**

算法1：dropAlign

1：输入：一层卷积的输出激活单元A，特征图尺寸wh，初始参数init\_param，align\_sche，block\_size，mode

2：if（align\_sche==true）

3： 计算init\_parm参数

4：end if

5：if（mode == test）

6： return A

7：end if

8：根据init\_param计算丢失框个数

9：依据丢失框大小block\_size随机选取丢失框

10：运用align计算丢失框内伯克利矩阵M

11：应用丢失框：

12：resize A

13：

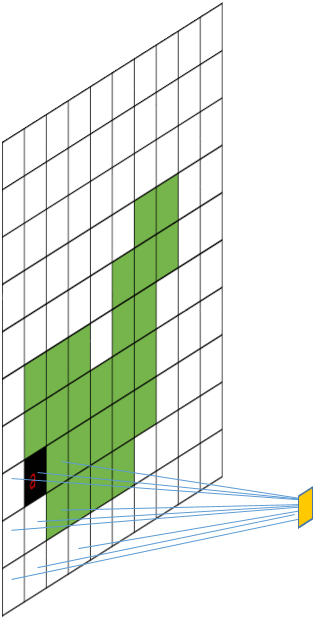
与dropout相似，在使用测试数据集测试模型期间，不运用dropAlign；仅在训练模型期间，使用dropAlign。这样在整个假设空间内，存在指数大小的模型平均预测，运用dropAlign的训练模型并不会看到连续的特征映射，但是在测试时模型可以看到完整的特征映射（特征映射的文章：SARDNET：A self-organizing feature map for sequences）。

**设置block\_size值**

在我们的算法里，block\_size是丢失框大小的参数，在整个训练过程中，固定block\_size参数大小。

**设置init\_param值**

根据输入特征图的大小，调整丢失框的数量，受到ScheduledDropPath(Learning transferable architectures for scalable image recognition)启发，如果将丢失框数量刚开始设置为固定值，会导致模型在训练初期无法学习到足够的图像特征，从而导致模型的学习速度变慢，所以我们采用线性增加init\_param参数的方式，在模型在学习尽量靠近真实的联合分布时，为了防止模型过度拟合，提高模型的泛化能力，将init\_param参数调大。

**Dropout的空间相关性**

如上图所示，卷积核的大小为3，如果运用dropout在卷积神经网络的隐藏层中，点a被丢失，它的激活单元设置为值零，但是同一卷积核感受野内的其余八个点并未丢失，那么在传播过程中，a点的值可以通过与它空间相关的8个点传递到下一层，在反向传播过程中，点a的梯度也可以通过其余八个点；这样看来，单纯的dropout在卷积神经网络的隐藏层中效果不佳，所以我们通过一种结构化的dropout，将选中的节点及其周围的一些节点都丢失掉。在spatialdropout（efficient object localization using convolutional networks）中，有一维图像是在卷积过程中线性相关的说明。

### 4.2.2 dropblock与mixup结合

我们收到mixup的启发，基于邻域风险最小化（VRM）原理，可以不将丢失框的值不插值为丢失框周围的值，也可以基于mixup的想法，将训练样本的邻域数据插值到丢失框内，混合后的数据仍是训练数据分布的i.i.d抽样。

**运用混合的算法**

算法2：局部替换

1：输入：一层卷积的输出*W*，*mode*，*block\_size*

2：if（*mode* == test） then

3： return *W*

4：end if

5：Randomly select a position *p* in *W*

6：at position *p*，creat a block *B* with the center bing *p*，the width,height being *block\_size* and set all values of B to be one and others to be zero。

7：the remanent is mask *M*.On the contary，set values in *B* to be zero，and others to be one。

8：shuffle *W*：

9：apply the mask：

算法3：感受野

1：输入：Mask *M*，kernel size，padding size ，stride ，layer（*W*）

2：set deconvolution：set all value of kernel to be 1，deconv=convolu-tion（kernel\_size=，padding=，stride=1）

3：dilate mask：*M*=Upsample（scale\_factor=，mode=‘nearest’）（*M*）

4：计算感受野：*M*=deconv（*M*）

5：将*M*的值设置在0到1之间

6：裁剪：if Size（*M*）> *W* then *M*=Crop（*M*）

## 4.2.3网络架构

在深层卷积神经网络（DCNN）中运用我的算法如图

### 4.2.4

### 4.2.5

## 4.3 本章小结

# 第5章 实验与分析

## 5.1 实验平台及参数

## 5.2 实验结果

## 5.3 实验分析

## 5.4 本章小结

# 第6章 总结与展望

## 6.1 总结

## 6.2 展望

致谢

.

Reference

1. Leshno M, Lin V Y, Pinkus A, et al. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function[J]. Neural networks, 1993, 6(6): 861-867. [↑](#endnote-ref-1)
2. [] Bishop C M. Neural networks for pattern recognition[M]. Oxford university press, 1995. [↑](#endnote-ref-2)
3. [] Goodfellow I, Bengio Y, Courville A. Deep learning[M]. MIT press, 2016. [↑](#endnote-ref-3)
4. Ciresan D C, Meier U, Masci J, et al. Flexible, high performance convolutional neural networks for image classification[C]//Twenty-Second International Joint Conference on Artificial Intelligence. 2011. [↑](#endnote-ref-4)
5. Cireşan D, Meier U, Schmidhuber J. Multi-column deep neural networks for image classification[J]. arXiv preprint arXiv:1202.2745, 2012. [↑](#endnote-ref-5)
6. DeVries T, Taylor G W. Dataset augmentation in feature space[J]. arXiv preprint arXiv:1702.05538, 2017. [↑](#endnote-ref-6)
7. Gulcehre C, Moczulski M, Denil M, et al. Noisy activation functions[C]//International conference on machine learning. 2016: 3059-3068. [↑](#endnote-ref-7)
8. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. arXiv preprint arXiv:1502.03167, 2015. [↑](#endnote-ref-8)
9. Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the inception architecture for computer vision[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2818-2826. [↑](#endnote-ref-9)
10. Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2223-2232. [↑](#endnote-ref-10)
11. Lang K J, Hinton G E. Dimensionality reduction and prior knowledge in e-set recognition[C]//Advances in neural information processing systems. 1990: 178-185. [↑](#endnote-ref-11)
12. Rifai S, Vincent P, Muller X, et al. Contractive auto-encoders: Explicit invariance during feature extraction[C]//Proceedings of the 28th International Conference on International Conference on Machine Learning. Omnipress, 2011: 833-840. [↑](#endnote-ref-12)
13. An G. The effects of adding noise during backpropagation training on a generalization performance[J]. Neural computation, 1996, 8(3): 643-674. [↑](#endnote-ref-13)