计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2020,29(10):222−227 [doi: 10.15888/j.cnki.csa.007623]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

卷积优化的变分自编码聚类方法①

严晓明[1,2]

[1](福建师范大学 数学与信息学院, 福州 350117)
[2](数字福建环境监测物联网实验室, 福州 350117)
通讯作者: 严晓明, E-mail: yanxm@fjnu.edu.cn

摘　要: 传统的变分自编码器将样本展平后直接作为输入数据, 当样本为图像数据时, 采用这样的方法进行学习效果欠佳.本文提出一种卷积优化的变分自编码器, 用多个可变层数的卷积网络预处理图像数据.每个卷积网络设置了不同的参数处理输入数据, 再将不同层卷积结果拼接后, 作为变分自编码器的输入.在变分自编码模型中增加一个类别编码器, 用于计算每个样本的类别分布和原样本集中类别分布的差异, 实现聚类. 实验证明, 本文提出的卷积优化方法相较于无优化的变分自编码器在聚类准确率上得到较大提高, 生成图像的质量得到了改善, 各类别生成样本在边缘及形状等方面的多样性也都有不同程度的增加.

关键词: 卷积; 变分自编码器; 聚类; 聚类准确率

引用格式: 严晓明.卷积优化的变分自编码聚类方法.计算机系统应用,2020,29(10):222–227. http://www.c-s-a.org.cn/1003-3254/7623.html

Clustering Method Based on VAE with Convolution Optimization

YAN Xiao-Ming[1,2]

[1](College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350117, China)
[2](Digital Fujian Internet-of-Things Laboratory of Environmental Monitoring, Fujian Normal University, Fuzhou 350117, China)

Abstract: The traditional Variational AutoEncoder (VAE) takes the flattened sample as input data directly. When the sample is image data, the effect of learning by this method is weakly. In this study, VAE with the convolution optimization is proposed to preprocess image data with multiple convolution networks of variable layers. Each convolution network sets different parameters to process the input data, then splices the results of different layers as the input of VAE. Clustering is implemented through the distance between the category label distribution of original dataset and the category distribution of each sample is calculated by adding a category encoder. The experimental results show that the convolution optimization method proposed in this study improves the clustering accuracy compared with the non-optimal VAE, increases the quality of the generated image and the diversity of the generated samples in the edge and shape.

Key words: convolution; Variational AutoEncoder (VAE); clustering; clustering accuracy

变分自编码器 Variational AutoEncoder (VAE)[1] 通过编码器部分学习样本集分布的期望和方差, 提取样本的统计学特征; 通过生成器部分还原样本, 并且能生成与原样本类似的新样本. 和自编码器 AutoEncoder 对比,VAE 提取出的特征反映了原始数据集每个样本的高斯分布特性, 隐变量从单一的向量形式变成了高斯分布的期望和方差, 从不易解读的数值变成用高斯分布的形式描述.

① 收稿时间: 2020-03-03; 修改时间: 2020-03-27; 采用时间: 2020-04-14; csa 在线出版时间: 2020-09-30

在无监督学习的领域, 近年来国内外学者对变分自编码器改进和应用进行了大量研究: Fabiu 将 VAE 和 RNN 模型结合, 提出了变分循环自动编码器 VRAE, 用于提取时间序列的概率特征, 提高 RNN 的训练效率[2]; 郑欣悦等人用 VAE 提取出的特征再结合注意力机制模型, 用于小样本图像的分类, 得到了更好的准确率[3]; 曾旭禹等人用 VAE 提取数据集的分布特征, 再结合概率矩阵分解方法增加推荐系统中不同物品的评分数据, 增加了推荐精度[4]; Xie 等人提出的 Deep Embedded Clustering (DEC)[5] 通过收敛样本集基于质心的 soft assignment 和辅助目标分布的 KL 距离来实现聚类, 在 MNIST 数据集上的聚类准确率达到了 84.3%. DEC 的聚类准确率较高, 但由于用到了堆叠自编码器来进行特征表示, 缺少生成能力, 不能生成新样本.

这些对变分自编码器的改进和应用都直接利用隐变量进行后续的学习, 这对隐变量是否能最大程度地提取到数据集的特征就显得十分重要. 传统变分自编码器将样本数据直接作为输入, 对于非图像的样本, 这样做直截了当, 而对于图像数据来说, 样本所表达信息的结构比较复杂, 如果直接将图像样本展平后作为输入数据, VAE 中全连接结构不能完全解读图像所表达的信息, 得到的隐变量就需要更多的全连接层去学习图像样本, 在数值上也会出现一定程度的偏差. 本文提出了一种用卷积结构处理样本集, 再由变分自编码器实现聚类的方法. 由于卷积层中的卷积核对图像数据中的线条, 边缘, 形状等特征进行提取, 降低了 VAE 编码器理解图像的难度. 实验结果表明, 用 VAE 聚类时采用本文卷积优化后的图像样本, 和原始样本直接输入比较, 聚类准确率提高 20% 左右, VAE 中解码器生成的样本更加稳定, 生成样本的质量也得到较大的提升.

1 损失函数

变分自编码器的损失函数由用来还原样本的重构损失以及保持样本多样性的 KL 散度损失两部分组成. 对于原始样本集 $X=\{x^1,x^2,\cdots x^n\}$, VAE 通过最小化重构损失使得生成的样本集 $\hat{X}=\{\hat{x}^1,\hat{x}^2,\cdots,\hat{x}^n\}$ 接近原始样本集, 这和自编码器还原样本的方法是类似的. Kingma 在文献 [1] 中给出了公式:

$$\log q\left(z|x^{(i)}\right)=\log N\left(z;\mu^{(i)},\sigma^{2(i)}I\right) \tag{1}$$

式 (1) 表示编码器 $q\left(z|x^{(i)}\right)$ 要拟合成期望为 $\mu^{(i)}$, 方差为 $\sigma^{2(i)}$ 的正态分布. 随后生成器 $p(\hat{x}^{(i)}|z)$ 从这些分布中分别采样, 还原得到和样本 $x^{(i)}$ 对应的样本 $\hat{x}^{(i)}$, 再求所有样本的均方误差作为变分自编码器的重构误差 *reconstruction_loss*:

$$reconstruction\_loss=\frac{1}{n}\sum_{i=1}^{n}(x^{(i)}-\hat{x}^{(i)})^2 \tag{2}$$

当从隐层所表示的正态分布中采样时, 由于方差的存在, 使得每次采样的结果并不总是一个确定的值, 这使得 VAE 有了生成能力. 隐层特征中的方差不为 0, 那么每次从隐特征所表示的分布中采样到的结果不都是期望, 间接确保了生成器的生成能力. Kingma 在文献 [1] 中指出, 最小化 KL 散度损失 *kl_loss* 使得编码器得到的正态分布接近先验的标准正态分布, 即:

$$\begin{aligned} kl\_loss&=KL\left(N\left(\mu,\sigma^2\right)||N(0,1)\right)\\ &=\frac{1}{2}\left(-\log\sigma^2+\mu^2+\sigma^2-1\right)\end{aligned} \tag{3}$$

由于标准正态分布的方差为 1, 通过最小化编码器 $q\left(z|x^{(i)}\right)$ 和标准正态分布之间的 *kl_loss*, 等价于让编码器得到的分布接近标准正态分布, 也等价于编码器得到的分布的方差不为 0, 保证了 VAE 的生成能力.

变分自编码器实现聚类操作时在损失函数中加入类别损失 *category_loss* 并进行最小化. 首先要根据数据集的类别总数设置类别集合 $Y=\{y^1,y^2,\cdots,y^k\}$. 这里把类别值看成是分布 $p(y^i)$, 在 VAE 模型结构中, 增加一个类别编码器 $q(y^i|x^i)$, 类别编码器是一个从原始样本集的每个样本中得到该样本类别的神经网络, 把类别编码器拟合出的条件概率分布 $q(y^j|x^j)$ 和分布 $p(y^i)$ 之间求出它们的 *KL* 散度:

$$KL(q(y|x)\Big\|p(y))=\int q(y|x)\ln\frac{q(y|x)}{p(y)}dy \tag{4}$$

由蒙特卡洛模拟法, 可得:

$$\int q(y|x)\ln\frac{q(y|x)}{p(y)}dy\approx\frac{1}{k}\sum_{j=1}^{k}\ln\frac{q(y^j|x^j)}{p(y^j)} \tag{5}$$

其中, $y^j\sim q(y^j|x^j)$.

最后可以得出类别损失函数为:

$$category\_loss=\frac{1}{k}\left(\sum_{j=1}^{k}\ln(q(y^j|x^j))-\sum_{j=1}^{k}\ln(p(y^j))\right) \tag{6}$$

在式 (6) 中, $p(y^j)$是原始样本集中类别的先验分布, 由于类别总数是一个确定值, 可以将其看成是均匀分布, 则式 (6) 的后半部分的值为一个常数. 最后的类别损失函数可以简化成:

$$category\_loss = \frac{1}{k}\sum_{j=1}^{k}\ln(q(y^j|x^j)) \tag{7}$$

2 卷积优化的变分自编码器

样本集为图像时, 传统的 VAE 将样本展平成一维数组, 作为输入数据. 展平操作后, 样本中邻近像素点点之间的顺序被重新排列, 导致这些点之间原本存在的信息被打乱, 对后续的学习不利. 本文在 VAE 的编码器前加入多个卷积网络, 对图像样本先进行卷积操作, 将卷积的结果作为编码器的输入, 同时在解码器后加入相应的反卷积操作, 还原成图像.

在图像处理领域, 卷积操作应用广泛, 在很多深度学习模型[6–8] 中用到了卷积操作, 通过不同的卷积核对图像中的边缘, 形状等信息进行处理, 取得了不错的效果. 在传统的 VAE 前加入多个卷积网络, 用不同的卷积核对图像中的特征进行预处理后, 再作为 VAE 的输入, 与采用直接将图像作为输入数据比较, 卷积后的图像数据中包含了更多隐藏在图像中的不易识别到的信息, 有利于编码器对图像分布特征的提取. 本文卷积优化的 VAE 聚类模型如图 1 所示.

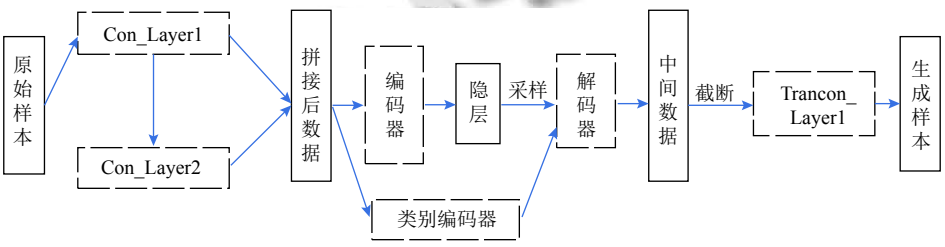

图 1 带卷积优化的 VAE 聚类模型

图 1 中实线矩形表示数据, 矩形的高度表示了数据维度的大小, 虚线矩形为网络模型. Con_Layer1 是一个卷积层, 包括一个卷积层, 一个激活层和一个池化层, 输入数据为原始的样本; 卷积层 Con_Layer2 的输入数据为 Con_Layer1 的输出结果, 和 Con_Layer1 不同的是, 这一层的卷积核的大小和数量发生了改变. 最后将两个卷积层的结果进行拼接, 作为 VAE 编码器的输入数据, 用不同的卷积核来分析样本图像, 能最大限度地将图像数据中的特征加以描述, 卷积核参数共享也能大大降低卷积网络需要优化的参数数量. 当数据集中样本图像长宽较大或通道数较多时, 还可以把这两个卷积层变成多层的卷积网络, 卷积核参数, 激活函数和池化参数也可以适当调整.

图 1 中的中间数据表示解码器得到的数据, 其维度和拼接后数据相同, 按 Con_Layer1 卷积网络所得到的维度截取后, 作为与 Con_Layer1 对应的反卷积层 TranCon_Layer1 的输入数据, 还原生成样本.

图 1 中类别编码器是一个多层神经网络, 最后一层用 Softmax 多分类器求得每个样本的类别, 类别总数为原始样本集中的样本总类别数. 求得的类别根据公式 7 计算类别损失, 加入到损失函数中, 参与总损失的梯度优化, 不参与隐层的采样.

本文提出的卷积优化的变分自编码器聚类算法步骤如算法 1.

算法 1. 卷积优化的变分自编码器聚类算法

1) 计算数据集的多卷积层拼接数据.
2) 构造全连接网络, 根据公式 7 求得样本的类别损失*category_loss*.
3) 构造两个全连接网络, 拟合样本$x^{(i)}$所属高斯分布的均值$\mu^{(i)}$和方差$\log\sigma^{2(i)}$.
4) 根据式 (3) 求*kl_loss*.
5) 从 3) 得到的高斯分布中采样, 构造全连接网络根据式 (2) 计算*reconstruction_loss*.
6) 令总损失*loss*为 2),4),5) 步中 3 个损失之和, 应用梯度下降最小化*loss*.
7) 返回 2), 直到达到指定的迭代次数.
8) 通过反卷积操作得到指定的生成样本, 并计算聚类准确率.

3 实验与结果分析

本文选取手写数字数据集 MNIST[9] 和服饰图像数据集 Fashion_MNIST[10] 展开实验. MNIST 数据集包含 10 个类别的手写数字图像; Fashion_MNIST 数据集中样本总类别数也为 10, 包含了外套, 包, 短靴, 牛仔裤

等不同服饰, 与 MNIST 数据集中的手写数字比较, Fashion_MNIST 数据集中的服饰有着更复杂的描述对象, 也包含了更多形状, 边缘等信息. 两个数据集的图像的大小和样本数都相同, 分别为 28×28 和 70 000 个, 适合作为验证本文算法使用. 实验环境的计算机配置为:Intel i7 CPU, 8 GB 内存, Windows 10 操作系统, 语言环境为 Python.

本文实验中, 将隐层的期望和方差的维数设为可调的参数s, 这两个向量的维度相同, 测试期望和方差的维数大小对聚类正确率以及生成样本质量的影响. 将 Con_Layer1 和 Con_Layer2 都设置为可调卷积层数的卷积网络, 层数由参数n指定, 卷积网络 Con_Layer1 中所有卷积层的其他参数均相同: 卷积核 16 个, 大小为 3×3, 步长为 1, 采用零填充的方法, 激活函数为 ReLU, 采用 2×2 的最大池化; 卷积网络 Con_Layer2 的卷积核调整为 32 个, 大小为 5×5, 其它的参数都和卷积网络 Con_Layer1 相同. 出于代码实现上的考虑, 通过对层数n值的改变, 能方便地实现卷积网络中卷积层数的变化而其它的参数不作修改, 这样的实现方法在 VGG16 模型中取得了不错的表现. 由于公式 3 在计算kl_loss时包含有方差的对数形式, 实验中隐层拟合的正态分布的方差更换为方差的对数形式, 由于对数函数值存在负数, 编码器的全连接网络没有加上激活函数, 层数设为 1. 本实验中为了验证编码器全连接层神经元总数对实验效果的影响, 将其设为可调的参数m. 两个数据集中的 70 000 个样本, 训练集均设为 60 000 个样本, 其余的样本作为测试集. 实验中迭代次数均设为 50. 实验结果如表 1 所示.

表 1 实验结果

数据集	隐层维数s	卷积层数n	神经元数m	总损失值		聚类正确率(%)	
				训练集	测试集	训练集	测试集
MNIST	2	2	128	41.23	41.13	66.40	66.39
	5	2	128	33.15	33.42	87.61	88.32
	10	2	128	32.04	31.85	85.65	86.54
	20	1	512	37.71	37.4	84.63	85.06
	50	1	512	47.20	46.59	93.94	94.0
	60	1	512	50.13	50.16	94.22	94.42
	100	1	512	66.12	65.36	83.66	83.92
Fashion_MNIST	10	4	512	154.20	160.18	67.45	66.51
	20	6	512	141.74	146.70	64.83	64.01
	30	4	512	143.77	146.93	67.62	67.6
	40	5	512	148.73	150.72	66.19	65.67
	60	5	512	158.39	162.43	68.03	68.0
	70	5	512	161.70	163.41	64.54	64.8
	100	5	512	178.10	180.82	66.08	65.67

表 1 中编码器全连接层中神经元总数m和卷积网络中卷积层的层数n仅列出最好情况下的取值. 从表 1 中可以看出, 对于 MNIST 数据集, 随着隐层变量维数的增加, 达到最佳效果的卷积网络中的卷积层数也随之变少, 这是由于随着正态分布维数的增加, 编码器能更好地拟合样本的分布特征, 此时卷积网络可以用较少的层数对图像进行卷积操作; 与此同时, 由于拼接后的输入数据将原始样本中的信息充分地展现, 编码器中的全连接层要用更多的神经元个数进行拟合, 最后得到的聚类准确率也会上升. 当隐层变量维数为 50, 卷积网络中卷积层的层数为 1, 编码器神经元个数在 512 时, VAE 的聚类正确率就达到峰值 94% 左右, 较文献 [6] 中 DEC 算法的 84.3% 的聚类正确率有较大的提升; 隐层维数大于 60 后, 则出现了过拟合, 聚类正确率开始下降.

传统的变分自编码器对 MNIST 数据集的聚类实验中, 在隐层维数为 10, 编码器神经元总数为 100 时就达到了聚类准确率为 75% 的峰值. 对比这个结果, 本文提出的经过卷积优化的变分自编码器方法用多个卷积网络对图像样本的卷积操作后再进行拼接, 能在最大程度上将图像中的边缘及形状等信息通过不同卷积核进行提取, 增大了输入数据的维度, 比原始样本展平的方式更适合于自编码器的学习, 聚类准确率得到了较大程度的提高, 效果明显. 同时随着拼接后输入数据维度的增大, 式 (2) 计算所有样本各个维度值的均方误差之和也增大, 在表 1 中损失函数的结果值随之

增加.

Fashion_MNIST 和 MNIST 的图像有着同样的长宽值, 但是图像中的服饰比手写数字的面积大, 即样本中的非零元的个数多于 MNIST, 导致了当实验中设置了相同参数的情况下, 其总损失值更大, 这也是该数据集的聚类准确率小于 MNIST 数据集的主要原因. 卷积网络对该数据集中图像样本的处理需要更多的卷积层数, 在实验中, 隐层变量维度为 60, 两个卷积网络的卷积层数为 5, 编码器的神经元总数为 512 时, 获得了最好的聚类正确率 68%, 随后也出现了过拟合的情况. 传统的变分自编码器对 Fashion_MNIST 数据集的聚类实验中, 在隐层维数为 25, 编码器神经元总数为 200 时就达到了聚类准确率为 55% 的峰值, 在服饰数据集上, 本文方法也得到了更好的聚类准确率.

对比两个数据集的实验, 取得最好结果时 VAE 隐层维度都在 50 至 60 之间, 从表 1 中也可以看到对 Fashion_MNIST 数据集达到最佳效果的隐层维度比 MNIST 数据集多了 10 维. 多元高斯分布能描述更复杂样本分布, 但是隐层维度不能通过无限增大的方式来取得更好的分布结果, 这是由于高斯分布的维度每增加一维, 分布的高度就为原分布高度的$(\sqrt{2\pi})^{-1}$, 当隐层维度增大到一定数值时, 高斯分布的效果和均匀分布接近, 已经失去了隐层变量的意义. 也就是说当隐层维度越来越大时, 隐层所表示的分布的高度更趋于 0, 由于方差越来越小, VAE 失去了生成能力.

不同卷积核的两个卷积网络对输入数据的处理不仅使得聚类准确率得到明显的提升, 同时也提高了变分自编码器生成样本的多样性. 图 2 是 Fashion_MNIST 数据集在隐层维数为 60, 卷积网络的层数为 5 时, 本文方法与传统的变分自编码器对于包 (Bag) 这个类别在样本多样性上的对比, 图 3 和图 4 是相同参数下的短袖 (T-shirt) 长靴 (Ankle boot) 的对比, 这 3 个图中左边均为传统的变分编码器的结果, 右边为本文改进 VAE 在指定参数下的结果.

图 2 中传统的 VAE 生成样本中的包基本维持了四边形的线条, 在包的大小, 四边形的两侧稍有变化; 而本文改进的 VAE 方法中的包变化样式更丰富并且不改变包类别的特征, 对包的提手部分的变化情况也多于传统的 VAE. 图 3 中右侧样本在短袖袖口处的变化多于左侧, 并且在短袖下摆的宽度以及整件短袖大小的变化优于传统 VAE 的生成样本. 从图 4 中可以看出, 传统 VAE 在长靴样本的多样性上体现在长靴的鞋跟的长短, 鞋帮的粗细和鞋面的弧度上, 这几个特征在本文 VAE 方法所生成的样本中更加明显, 并且右侧样本在保持长靴类别的前提下, 在鞋子的形状和边缘上有更多的变化. 对服饰数据集的实验中, 其他类别的样本也同样体现了本文方法在生成样本多样性上的提升.

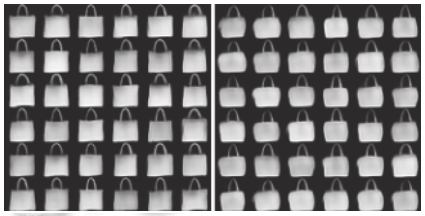

图 2 传统 VAE 与改进 VAE 在包类别上多样性对比

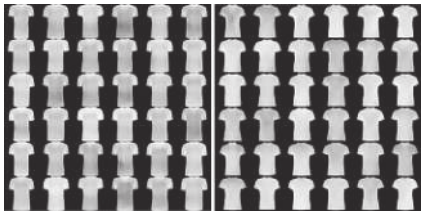

图 3 传统 VAE 与改进 VAE 在短袖类别上多样性对比

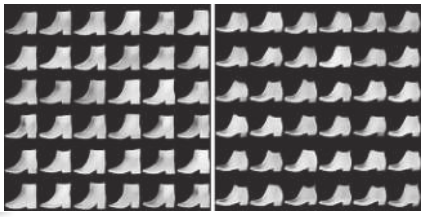

图 4 传统 VAE 与改进 VAE 在长靴类别上多样性对比

4 结语

本文提出了用多个卷积网络优化变分自编码器实现聚类的方法, 通过隐层变量的维数和卷积网络层数的调整, 在对 MNIST 和 Fashion_MNIST 数据集的实验结果表明, 该方法与传统变分自编码比较, 聚类准确率得到了明显提高, 增加了变分自编码器生成样本的多样性, 生成的图像质量更好.

参考文献

1 Kingma DP, Welling Max. Auto-encoding variational bayes. https://arxiv.org/pdf/1312.6114.pdf, 2019.

2 Fabius O, van Amersfoort JR. Variational recurrent auto-encoders. arXiv: 1412.6581, 2014.
3 郑欣悦, 黄永辉. 基于 VAE 和注意力机制的小样本图像分类方法. 计算机应用与软件, 2019, 36(10): 168–174. [doi: 10.3969/j.issn.1000-386x.2019.10.030]
4 曾旭禹, 杨燕, 王淑营, 等. 一种基于深度学习的混合推荐算法. 计算机科学, 2019, 46(1): 126–130. [doi: 10.11896/j.issn.1002-137X.2019.01.019]
5 Xie JY, Girshick R, Farhadi A. Unsupervised deep embedding for clustering analysis. Proceedings of the 33rd International Conference on Machine Learning. New York, NY, USA. 2016. 478–487.
6 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. https://arxiv.org/abs/1409.1556, 2019.
7 Szegedy C, Liu W, Jia YQ, *et al*. Going deeper with convolutions. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition(CVPR). Boston, MA, USA. 2015. 1–9.
8 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA. 2016. 770–778.
9 LeCun Y, Cortes C, Burges CJC. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/. 2019.
10 Xiao H, Rasul K, Vollgraf R. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. arXiv: 1708.07747, 2017.