Reason for reading

最近一直在公开数据集上做了很多对比实验研究global feature的Normalization对结果的影响以及以前遇到的一些问题，和NormFace的一些想法不谋而合，吸引了我来读这一系列可以借鉴的Face相关的论文。

NormFace

这篇文章让我理解了之前做的一个实验（global feature做L2 normalization送去算Triplet和过FC算center）没有收敛的原因，一定程度上，它是后面CosFace、ArcFace和AdaCos的基础，对一些结论给出了详细的数学推导。他解决了两个非常重要的问题：（1）为什么测试要norm；（2）为什么在用softmax训练中对feature做norm网络不收敛。

- 为什么测试要norm？ 因为softmax会鼓励分类正确的feature的幅度越大越好，因此feature是成放射状分布的。即使两个feature同属一个类，他们和其他类的feature的距离是不一样的。所以norm后压缩到一个超球面上，以角度来区分。

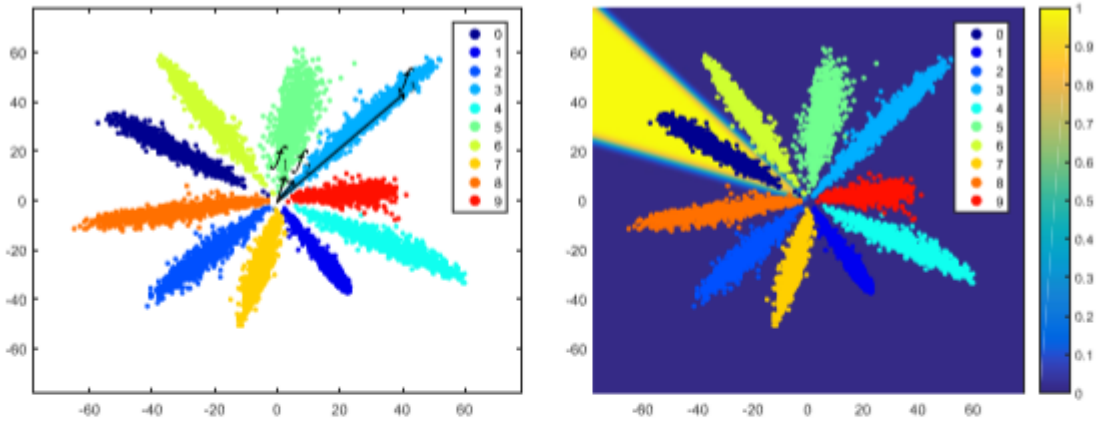

Figure 2: *Left:* The optimized 2-dimensional feature distribution using softmax loss on MNIST[14] dataset. Note that the Euclidean distance between f_1 and f_2 is much smaller than the distance between f_2 and f_3, even though f_2 and f_3 are from the same class. *Right:* The softmax probability for class 0 on the 2-dimension plane. Best viewed in color.

此外，FC去掉Bias这一点在这篇论文也有据可循，在有bias的情况下，可能有些类FC的Weights是相同的，这时候需要Bias来区别两个类，对他们做norm以后，这些类的点会分散到整个单元超求面上影响分类。

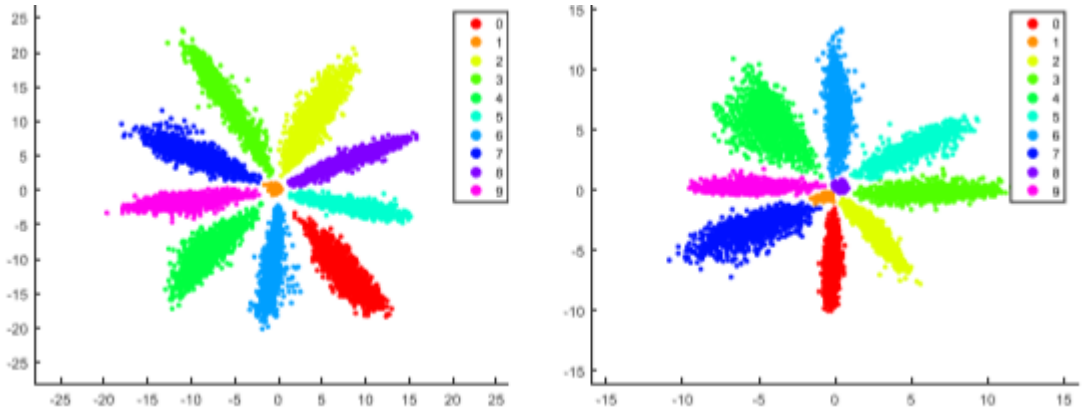

Figure 3: Two selected scatter diagrams when bias term is added after inner-product operation. Please note that there are one or two clusters that are located near the zero point. If we normalize the features of the center clusters, they would spread everywhere on the unit circle, which would cause misclassification. Best viewed in color.

- 为什么在softmax训练时直接做norm会不收敛？ 因为norm后，学好的权重和feature之间的cosine similarity的取值在[-1,1]之间，这样过了softmax后的y=gt的概率将会极其小，而训练希望y=gt的概率接近于1，显然这是达不到的。且此时对y=gt的loss很大，对y≠gt的loss很小。 此外，论文给出了详细的理论推导证明softmax的loss存在下界，下界□和数据集类别数以及feature和weights的norm有关。下界如下：

$$\log\left(1+(n-1)e^{-\frac{n}{n-1}\ell^2}\right)$$

- 如果norm太小，下界很大，即loss很难收敛，因此希望将feature norm到一个更大一点的值，使下界减小，让网络容易收敛。（这也是后面一些softmax改进loss中的scale参数的由来。）

CosFace

CosFace总结起来主要有两个贡献：（1）将feature和FC的weights都norm，然后提出Normalized version of Softmax Loss（NSL）和Large margin cosine loss（LMCL）。其中feature和weights的norm以及NSL被成功运用在SphereReID中。

- NSL

$$L_{ns}=\frac{1}{N}\sum_i-\log\frac{e^{s\cos(\theta_{y_i,i})}}{\sum_j e^{s\cos(\theta_{j,i})}}.$$

- LMCL

$$L_{lmc}=\frac{1}{N}\sum_i-\log\frac{e^{s(\cos(\theta_{y_i,i})-m)}}{e^{s(\cos(\theta_{y_i,i})-m)}+\sum_{j\neq y_i}e^{s\cos(\theta_{j,i})}}, \tag{4}$$

subject to

$$\begin{aligned} W&=\frac{W^*}{\|W^*\|},\\ x&=\frac{x^*}{\|x^*\|},\\ cos(\theta_j,i)&=W_j{}^T x_i, \end{aligned} \tag{5}$$

（2）分析了scale参数和margin参数的取值范围。

- scale取值范围

$$s\geq\frac{C-1}{C}\log\frac{(C-1)P_W}{1-P_W}.$$

- margin取值范围

$$\begin{aligned} &0\leq m\leq 1-\cos\frac{2\pi}{C}, \quad (K=2)\\ &0\leq m\leq\frac{C}{C-1}, \quad (C\leq K+1)\\ &0\leq m\ll\frac{C}{C-1}, \quad (C>K+1) \end{aligned}$$

- feature和weights的norm使得feature和weights的内积只和夹角的cosine值相关；
- LMCL使得分界面存在margin，优化问题更加hard，使得feature更容易从角度上区分开，不像NSL，在分界面处的feature不好区分。

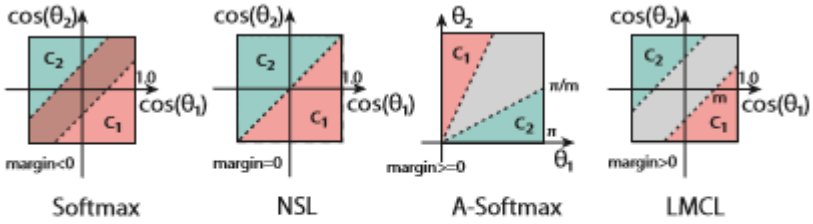

Figure 2. The comparison of decision margins for different loss functions the binary-classes scenarios. Dashed line represents decision boundary, and gray areas are decision margins.

ArcFace

这篇论文像对简单一点，即把CosFace的cosin margin改成了angular margin，loss如下：

$$L_3 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\cos(\theta_{y_i}+m))}}{e^{s(\cos(\theta_{y_i}+m))}+\sum_{j=1,j\neq y_i}^{n}e^{s\cos\theta_j}}.$$

并和SphereFace和CosFace的loss做了对比，比较了margin上的差异，并直观上给出了分界面图：

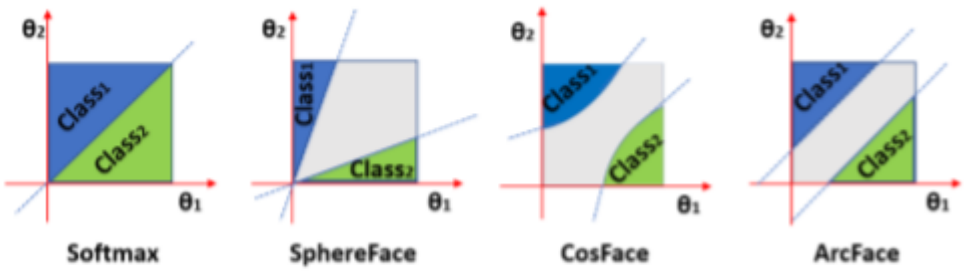

Figure 5. Decision margins of different loss functions under binary classification case. The dashed line represents the decision boundary, and the grey areas are the decision margins.

AdaCos

AdaCos有点像是对近期各种Face中的Loss的一种总结，我个人觉得论文的闪光点是它的第三部分，即对scale参数和margin参数的实验探究，和之前论文中对scale、margin相关的理论推导相辅相成。这一部分通过详细证明了一下两点：

- scale和margin太大太小都不好。scale太小，限制了y=gt的概率大小，一方面使得分类准确度不高，另一方面，即使y=gt样本被正确分类，网络对它依然还有loss；scale太大，即使feature和weight之间的夹角接近90度还有很大的概率被认为正确分类，也即网络没办法针对错分样本进行优化。margin太小，分界面分离不够，分界面周围的样本容易错分；margin太大，优化问题太hard，是的正确分类样本的概率不高。
- scale和margin的大小对结果的影响方向是一直的，即大scale和大margin对结果的影响是一致的，但是大scale和小margin对结果的往往都是有害的。且scale影响y=gt样本的概率上限，margin影响相位。其实，可以至设置一个超参数如scale，另外一个通过特定方法推算得到。

接下来，论文对scale参数进行在线的学习调整，而不是像NormFace、CosFace、ArcFace那样是提前设定且训练过程中是一直不变的。论文通过详细的证明给出了scale的在线调整方法如下：

$$\tilde{s}_d^{(t)} = \begin{cases} \sqrt{2} \cdot \log(C-1) & t = 0, \\ \dfrac{\log B_{\text{avg}}^{(t)}}{\cos\left(\min(\frac{\pi}{4}, \theta_{\text{med}}^{(t)})\right)} & t \geq 1, \end{cases}$$

Thoughts

这几篇论文都是对人脸过程中normalization和Loss的优化工作，人脸和ReID有很多相似之处。实践也证明，人脸的normalization方法和NSL损失函数被成功用到SphereReID中，而NormFace提出的直接norm网络不收敛问题我自己也亲身经历过，Bag of tricks的开源版本就是训练过程中不norm，而测试过程中做了norm，也和NormFace中提出的问题相一致。所以说，人脸的方法可以借鉴，Normalization和相关Softmax Loss的改进也可深入学习。