Long tail learning via logit adjustment（ICLR 2021）

Introduction

Related work（introduction)

这篇paper前面主要把长尾相关的方法分为三类，分别是改变模型输入(re-sampling 和 transfering 这些）、改变模型输出、改变模型中的损失函数。文中提到一般处理长尾问题时，改变输出或损失函数时都会搭配使用一下改变模型输入。这篇文章主要follow关注的是后面这两类方法（在后面会有相关公式）：

- 改变模型输出（也就是改变最后的分类器）：
  - 对训练好的模型最后分类器做权重标准化（Post-hoc weight normalisation）
- 改变损失函数：
  - 偏样本频率角度出发改变
  - 从调整margin出发改变

看了论文开头两部分，按我的理解，作者follow的这两类方法所涉及的大部分methods有两个特征：一是属于re-balancing，需要将训练样本分布（即样本频率）作为先验特征；二是都是调整 logit 的（看论文标题就显而易见了），而不是提出新的model或者数据增强那些方向的。

Problem analysis

论文介绍了文章主要用的一个 balanced error，和对应这个balanced error的最优 bayes 解。

- 引入最传统方法（softmax 交叉熵损失）:$l(y, f(x)) = log(1 + \sum_{y' \neq y} e^{f_{y'}(x) - f_y(x)})$
  - 这个交叉熵损失是基于softmax的，也就是说 $p(y|x) \propto \frac{e^{f_y(x)}}{\sum e^{f_{y'}(x)}}$，所以 $p(y|x) \propto e^{f_y(x)}$
  - 事实上，由于 $p(y|x) = \frac{p(y)}{p(x)} \times p(x|y)$，所以 $p(y|x) \propto p(y) \times p(x|y)$
    - 这里实际上$p(x)$是可以忽略的，因为每个样本是均匀采样的
  - 因此未处理过的 softmax 交叉熵损失中 $e^{f_y(x)} \propto p(y) \times p(x|y)$, 这就默认引入了$p(y)$这一不平衡的数据分布频率，这会有可能导致比如head类的评分$e^{f_y(x)}$ 对任何样本都偏高。

- 对于这样一个长尾情况，如果用一般的分类准确率就不是和合适了，因为假若把所有样本都归类为head类，说不定准确率还不低。

  面对这个问题，很自然就想到评估分类准确度时，采样一种均衡的方式：
  $BER(f) = \frac{1}{L} \sum P(y \notin argmax_{y' \in [L]} f_y^{'}(x))$
  - 这其实也默认等价于我们在训练时的loss符合： $p_{bal}(y|x) \propto \frac{1}{L} \times p(x|y)$

- L 是类别数，这样原来的不均衡的$p(y)$就被强制换成了每类相同的$\frac{1}{L}$

- 根据前面的分析，对于前面的 $BER(f)$，假设它的最优解是f^*，那么它应该满足：
  - $f^* \in argmin_f BER(f)$。
  - $argmax_{y\in[L]} f_y^*(x) = argmax_{y\in[L]} P^{bal}(y|x) = argmax_{y\in[L]} P(x|y)$
    - 这个公式她的实际含义是说，我们判别一个样本所属类别时，应该看每个样本在各自类别的样本空间下的所属概率，再放在一起比较，而不是直接把样本放在整个空间计算各个类别的所属概率

- 此外，论文标题里说的 logit 在这篇文章里其实是指：
  - $f_y(x) = w^T\phi(x)$

Related work（analysis)

基于上面的公式化语言，论文分析了一波前面提到的两大类相关工作，并指出了他们的问题，也就是本篇论文的改进方向。

- **对训练好的模型最后分类器做权重标准化（Post-hoc weight normalisation）：**
- **method**
  - 这类方法的范式就是在训练完以后的分类器中事后调整，引入原始数据的数据分布 或者 直接做权重normalization来缓解训练过程中产生的bias，公式如下：
  - $argmax_{y\in[L]} \frac{w_y^T \times \phi(x)}{v_y^T} = argmax_{y\in[L]} \frac{f(x)}{v_y^T}$
  - 这里如果 v_y 取 $P(y)$, 就表示在引入先验分布的数据类别频率信息来人为提升尾部类的权重， 如果 v_y 取 $||w_y^2||$, 就表示在做normalization 。

- **problem**
  - 对于第一种引入先验分布的，后面会说它实际上不符合前面的balance最优解f^*,也就是它并不能最小化$BER(f)$;
  - 对于第二种normalization的实际上是有研究发现训练出来的分类器的权重和$P(y)$有关系，这个实验结果我自己是在decoupling的那篇文章里看到的，但是本篇论文作者说做了实验发现这个结果和optimizer有关，如果选用SGD确实权重和$P(y)$呈正相关关系，而如果用Adam却得不到类似的结论。
  - 那关于第二个normalization的问题，我又回去看了发现这个结论的decoupling这篇文章，发现他们确实用的是SGD。论文没具体展开讲为什么就是和optimizer有关，关于SGD和长尾分布，其实我在另一篇因果分析的neurlps文章里看到过一点相关分析，我结合着自己理解猜测可能是SGD里面用的一阶动量在引入整个不平衡分布时还是能最后通过一个线性的分类器权重体现出来，而Adam用的二阶动量可能带来了更多的非线性因素，这导致了最后结论没有那么直接。
  - 所以我觉得这里的问题可能就是说用ADAM时候，训练过程引入的不平衡信息不只包含在最后的线性分类层，可能在模型其它模块就明显体现了。

- **改变损失函数：**
- **method**
  - 偏样本频率角度出发改变：$l(y,f(x))=\frac{1}{P(y)}\times log(1+\sum_{y'\neq y}e^{f_{y'}(x)-f_y(x)})$
    - 这个其实就是直接引入先验分布的数据类别频率信息，调高尾部类没预估准的损失所占权重，因为他们数量少，权重大才均衡。
  - 从调整margin出发改变: $l(y,f(x))=log(1+\sum_{y'\neq y}e^{\delta_y}e^{f_{y'}(x)-f_y(x)}),\delta_y\propto\frac{1}{p^4(y)}$
    - 这个其实目的是在处理尾部类时，让那些尾部类和其它所有类别的margin都能调大一点，减少混淆概率。换句话说，就是强调在尾部类样本判别时，要加大和头部类的区分阈值，即margin变大一些。
  - 从调整margin出发改变: $l(y,f(x))=log(1+\sum_{y'\neq y}e^{\delta_y'}e^{f_{y'}(x)-f_y(x)}),\delta_y'\propto P(y^{'})$
    - 这个和上面一个改margin稍有不同，其实主要是把和 y 有关的概率换成了 $y^{'}$ ，这样一来不管是在处理头部还是尾部类，都要注意和尾部类之间的margin要大一些。我觉得可以看成是前者的升级版，这里不止是强调在尾部类样本判别上提升区分阈值，同时强调在头部类样本判别时，也要注意提升与尾部类的区分阈值。
- **problem**
  - 论文在后面有证明，上面这两种改变margin的方法其实是改变了决策边界，这两种方法和我们用来评价的错误率之间不存在fisher一致性，也就是说用这两种loss训练只能达到错误率的局部最优解；
  - 第一类balanced loss也是和我们用来评价的错误率之间不存在fisher一致性。

- **小总结**
  - 除了第一类方法 Post-hoc weight normalisation 中的对最后权重最标准化（这个前面以前说了它的问题），其它方法基本上都是引入先验分布$P(y)$的，只不过一类训练完引入，一类在训练中就引入，论文事实上改进的就是这两种引入$P(y)$的方法。

Logit adjustment

论文的核心公式如下：
$argmax_{y\in[L]}P^{bal}(y|x)=argmax_{y\in[L]}\frac{exp(f(x))}{P(y)}=argmax_{y\in[L]}f(x)-ln(P(y))$

其实但看最终结果，其实就是说，原来是在$f(x)$这步 logit 就直接拿$P(y)$处理，论文结论是认为应在最后与概率成正比的$exp(f(x))$这步处理$P(y)$。

- **Post-hoc logit adjustment**
- **method**
  - 首先是在训练完引入频率，论文得出的公式很简单直接：
    $argmax_{y\in[L]}\frac{exp(f(x))}{\pi^\tau(y)}=argmax_{y\in[L]}f(x)-\tau ln(P(y))$
    - 公式里面的$\pi(y)$ 其实就是$P(y)$ 的估计，τ 就是 temperature scaling 中常用到的一个参数，在这里呢，它表示了一种对模型输出的修正。
    - τ 如果取 1 ，就是最一般的表示 $p(y|x)\propto exp(f_y(x))$；

    - 但是实际上，输出的 logit 有可能需要进一步标定，我还没看这里参考的那篇paper，所以暂时只能猜测这里未标定说的是 $f(x)$ 输出值太大或者带有太多的bias。
    - τ 如果不取 1 ，就是表示 $p(y|x) \propto exp(\tau^{-1} f_y(x))$ ，也就是把 logit 按比例缩小了一点。

- **comparison**

    - 和其它方法对比起来呢，这个方法看似没什么改动，τ 等于1的情形是长尾问题中最最基本的用法，不等于 1 的呢别的论文中实际上也已经提出过了。那作者把除以 $P(y)$ 这一步从原来的 logit $f(x)$ 那一步放到了后面 $exp(f(x))$ 这个小的改动到底有什么用处呢，作者在这里给出了证明。
    - 首先，两种方法带来的结果确实是不一样的：$\frac{w_1^T\Phi(x)}{\pi_1} < \frac{w_2^T\Phi(x)}{\pi_2} < \cdots < \frac{w_L^T\Phi(x)}{\pi_L}$ 和 $e^{\frac{w_1^T\Phi(x)}{\pi_1}} < e^{\frac{w_2^T\Phi(x)}{\pi_2}} < \cdots < e^{\frac{w_L^T\Phi(x)}{\pi_L}}$ 之间并不存在等价关系
    - 这就意味着，原来直接拿 logit 除$\pi(y)$的方法的处理，并不是完全朝着消除bias的方向进行的，举个例子，如果开始一个tail类样本的分数$f(x)$是 -0.1 而 一个head 类分数$f(x)$是0.1 ，那按第一种方法处理过后，还是一正一负，head类样本分数不管怎么样都是比tail类样本分数高的。
    - 也即，这与我们最小化的 balanced error $BER(f)$ 并不完全一致；而论文是在最后概率那步除的 $\pi(y)$ ，是符合前面f^*的要求的，与最小化的$BER(f)$ 显然一致。

- **Logit adjusted loss**

- **method**

    - 对于改变 loss ，因为这里是针对多分类任务的，所以 loss 是 softmax交叉熵损失函数。
    - 前面提到 $p_{bal}(y|x) \propto \frac{1}{L} \times p(x|y)$， 而直接拿长尾分布数据训练出来的是 $p(y|x) \propto p(y) \times p(x|y)$ ，所以 $p_{bal}(y|x) \propto p(y|x)/p(y)$ 。
    - 按我的理解，对于我们的这样一个长尾分布数据集来做训练，最后最小化的loss肯定也是要符合这个分布的，但是这样的话这个loss就和我们所期望的 $argmin_f BFR(f)$ 不一致了。而我们的最终目的是让训练出来的$e^{f_y(x)}$ 是一个 balance 的评分，而不是为了最小化那个不均衡的 loss，所以要让机器知道我们需要的是什么，公式如下：
    $l(y, f(x)) = -log\frac{exp(f(x)+\tau log(\pi(y)))}{\sum_{y' \neq y} exp(f_{y'}(x)+\tau log(\pi(y)))} = log(1 + \sum_{y' \neq y}(\frac{\pi_{y'}}{\pi_y})^\tau e^{f_{y'}(x)-f_y(x)})$
        - 这里实际上和原来相比就在最后的概率处，添加了一个偏移项，用这个loss训练既能让机器知道我们要的，符合$argmin_f BFR(f)$ 要求的平衡训练出的概率 $e^{f_y(x)}$ ； 同时 $exp(f(x) + \tau log(\pi(y)))$ 这个概率又符合我们的数据集情况。
        - 事实上，这和我们上面的后验修改本质时一样的，上面最后得到 $P^{bal} = argmax_{y \in [L]} f(x) - \tau ln(P(y))$, 这里我们拿$f(x) = P^{bal} + \tau ln(P(y))$作为概率评分。
        - 论文指出，这里实际上是在理论上对凸函数可行，也就是说如果函数是凸函数，那么最后加上这个bias $\tau ln(P(y))$ 是能够得到 $argmin_f BFR(f)$的，但是如果是非凸函数，就不一定了，可能会收敛到局部最优。这里我理解是，我们的神经网络往往就是非凸的，打个比方 $relu(-x)$ 这个就是非凸的，也就是说我们用上面这个公式还是会存在一些问题。
    - 为了分析方便，论文定义了最基本的loss公式：
    $l(y, f(x)) = \alpha log(1 + \sum_{y' \neq y} e^{\Delta_{yy'}} e^{f_{y'}(x)-f_y(x)})$
        - $\Delta_{yy'}$ 可以看成是类别之间的gap；
        - 这个公式如果 $alpha$ 取1，$\Delta_{yy'}$取$ln(\frac{\pi_{y'}}{\pi_y})$，就对应了论文提出来的上面 τ 取1的公式。

- **comparison**
  - 前面提到的三种loss（一个是balanced loss，另两个是调整margin的loss）都是上面基本loss公式的特例。
    - $alpha$ 取 $\frac{1}{\pi_y}$，$\Delta_{yy'}$取0， 就是balanced loss；
    - $alpha$ 取1，$\Delta_{yy'}$取$\pi_y{}^{-1/4}$ 或者$F(\pi_y^{'})$，其中$F(\cdot)$是一个递增函数，就对应了另外两个调整margin的loss。
  - 一方面，上面三个原来的loss引入样本频率是要么使用正例频率π_y，要么引入负例频率$\pi_y^{'}$，却没有同时引入这两者。
  - 另一方面，作者提出了一个定理（证明在论文中），首先，论文提出来的 $alpha$ 取1，$\Delta_{yy'}$取$ln(\frac{\pi_{y'}}{\pi_y})$的公式肯定是符合和 $argmin_f BFR(f)$ 存在fisher一致性；但是其它三个loss根据证明就不符合了。

Discussion

- 首先，结合论文中提到的两种method是不可行的，改loss以后本来就是认为得到的是我们需要的了，再给它加一个bias就不对了。
- 对于上面提出的关于loss符合fisher一致性的定理，可以衍生出很多其它思路的loss。
- 除了上面的定理，作者还用了一个二分类问题的例子说明了weight normalisation 和loss modification不一定会收敛到 $BER(f)$ 的最小解。

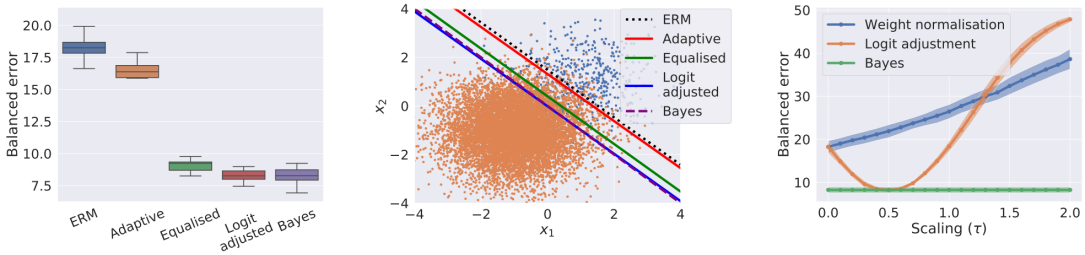


Figure 2: Results on synthetic binary classification problem. Our logit adjusted loss tracks the Bayes-optimal solution and separator (left & middle panel). Post-hoc logit adjustment matches the Bayes performance with suitable scaling (right panel); however, *any* weight normalisation fails.

Experiment

- 评价指标和之前论文有所不同，用的是balanced error 即 $BER(f)$。

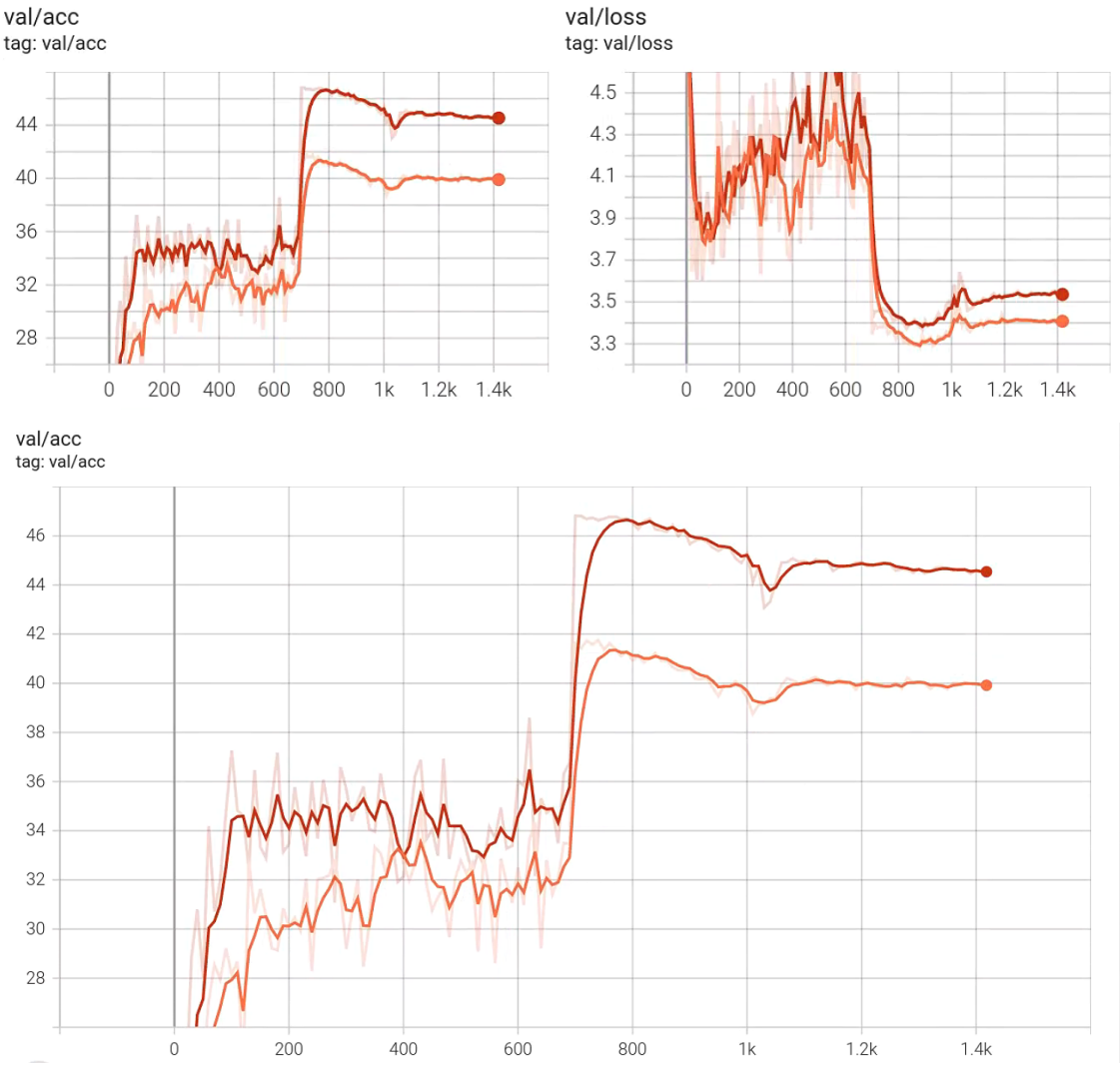
```
        results = {}
        avg_acc = 0
        for i in range(num_class):
            acc = 100.0 * n_class_correct[i] / n_class_samples[i]
            avg_acc += acc
            results["class/" + classes[i]] = acc
        results["AA"] = avg_acc / num_class
        return results
```

- 实验结果如下：

Method	CIFAR-10-LT	CIFAR-100-LT	ImageNet-LT	iNaturalist
ERM	27.16	61.64	53.11	38.66
Weight normalisation ($\tau = 1$) [Kang et al., 2020]	24.02	58.89	52.00	48.05
Weight normalisation ($\tau = \tau^*$) [Kang et al., 2020]	21.50	58.76	49.37	34.10*
Adaptive [Cao et al., 2019]	26.65†	60.40†	52.15	35.42†
Equalised [Tan et al., 2020]	26.02	57.26	54.02	38.37
Logit adjustment post-hoc ($\tau = 1$)	22.60	58.24	49.66	33.98
Logit adjustment post-hoc ($\tau = \tau^*$)	19.08	57.90	49.56	33.80
Logit adjustment loss ($\tau = 1$)	22.33	56.11	48.89	33.64

Table 3: Test set balanced error (averaged over 5 trials) on real-world datasets. We use a ResNet-32 for the CIFAR datasets, and ResNet-50 for the ImageNet and iNaturalist datasets. Here, †, * are numbers for "LDAM + SGD" from Cao et al. [2019, Table 2, 3] and "τ-normalised" from Kang et al. [2020, Table 3, 7]. Here, $\tau = \tau^*$ refers to using the best possible value of tuning parameter τ. See Figure 3 for plots as a function of τ, and the "Discussion" subsection for further extensions.

- 我自己也去复现了一下代码，结果对的上（浅橙色的是Logit adjustment loss, 深橙色的是baseline）：



Conclusion

这篇文章还是有很多可取之处的，尽管最后结果看起来没有特别高，但是论文只是提供了一个思路，事实上顺着思路能衍生出很多其它的loss和使用方法，并且如果改一下backbone和一些超参，或者是加一些trick（看了代码，论文只是简单用了随机翻转和随机裁剪这两个简单的数据增强）效果应该还会变得更好。论文最关键的是那个定理，借着论文的那个定理，可以说把很多解决长尾问题的loss给统一了一下，并且提供了实际的理论证明。看了代码部署起来也十分方便，改loss就是在原来的softmax交叉熵那里改一下，post-hoc修改就是在最后分类时候处理一下。