机器学习中的各种范数与正则化

对于统计机器学习算法一般为缓解过拟合现象的发生需要在进行正则化操作,通过正则化以偏差的增加换取方差的减小。最常见的方法即为在损失函数中引入矩阵范数，以对模型的复杂程度做出惩罚，其目标函数一般如下式所示：

$$ min\quad \tilde J(\theta;X,y)=\underbrace{J(\theta;X,y)}*{(1)}+*underbrace{\alpha \Omega(\theta)}*{(2)}\tag{1}
$$

上式中，第一项即为经验风险，第二项即为正则化项。其中$\alpha\geq0$，为调整两者之间关系的系数。当$\alpha=0$时，则表示无正则化项，α越大则表示对应正则化惩罚越大。

L^2 范数正则化

$$ L_2:\qquad \Omega(\theta)=\frac12||w||^2_2\tag{2} $$ 这里我们假设模型的偏置参数均为0，则参数θ即为w，因此其目标函数为： $$ \tilde J(w;X,y)=J(w;X,y)+\frac{\alpha}{2}w^Tw\tag{3}
$$

对上式求其梯度有： $$ \nabla_w\tilde J(w;X,y)=\nabla_wJ(w;X,y)+\alpha w\tag{4}
$$

使用梯度下降法更新权重w ，则w将向负梯度方向收敛，如下： $$ w;\leftarrow;w-
\epsilon(\nabla_wJ(w;X,y)+\alpha w)=(1-\epsilon\alpha)w-\epsilon\nabla_wJ(w;X,y)\tag{5}
$$

从中可以看出每次权值w更新时都将乘以$(1-\epsilon\alpha)$，该常数因子小于0，即权重将会逐渐收缩，趋近于0。

进一步地，这里令$w^=*arg,min_wJ(w)$，将$J(w;X,y)$简记为$J(w)$，即$w*^$为损失函数$J(w;X,y)$取得最小训练误差的权值。并在w^*的邻域对损失函数$J(w)$做二次近似（泰勒展开）,记为 $\hat J(w)$，如下：

$$ \hat J(w)=J(w^)+*frac12(w-w*^)^TH(w-w^*)\tag{6} $$

上式中H为J在$w^$处计算的Hessian矩阵，且该矩阵为半正定矩阵。由上述知，$w*^$为损失函数的最优解，因此$\hat J(w)$的梯度将为0，即式（6）对w求偏导为0，如下所示：

$$ \nabla_w \hat J(w)=(w-w^*)H=0\tag{7} $$

记$\widetilde w$为最优权值w，将式（7）代入式（4）： $$ \nabla_w\tilde J(\tilde w)=\alpha \tilde w+
(\tilde w-w^*)H=0\tag{8}
$$

故：

$$ (H+\alpha I)\tilde w=Hw^\ *tilde w=(H+\alpha I)^{-1}Hw*^\tag{9} $$

（由于Hesian矩阵为半正定矩阵，故其为实对称阵。因此有$(\tilde w-w^)H=H(*tilde w-w*^)$）

当α趋向于0时，$\tilde w$将趋近于w。我们将实对称Hesian矩阵H分解为一个对角矩阵$\wedge$和一组特征向量的标准正交基Q，因此有$H=Q\wedge Q^T$，代入式（9）,可得

$$ \tilde w=(Q\wedge Q^T+\alpha I)^{-1}Q\wedge Q^Tw^=*[Q(\wedge +\alpha I)Q^T]^{-1}Q\wedge Q^Tw^=\ Q^{T^{-1}}(\wedge+\alpha I)^{-1}Q^{-1}Q\wedge Q^Tw^=Q(\wedge +\alpha I)^{-1}\wedge Q^Tw^\tag{10}
$$

从上式中可以看出经过正则化后，权重$\tilde w$将会沿着由H特征向量所定义的轴缩放未经标准化的损失函数最优解$w^$。*具体来说，我们会根据$\frac{\lambda_i }{\lambda_i+\alpha}$因子收缩与$H$第$i$个特征向量对齐的$w^$*的分量。如下图所示。

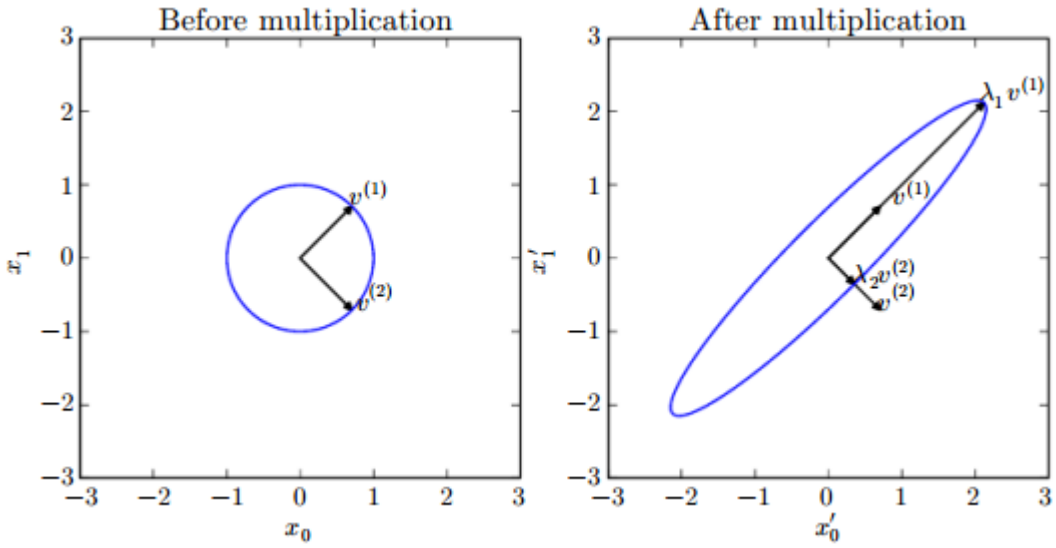

图1. 特征向量作用效果图

上图为特征向量的作用效果图，这里矩阵有两个标准正交特征向量，对应的特征值分别为$v^{(1)}$和$v^{(2)}$。其中左图为所有单位向量$\mu\in\Bbb R^2$集合构成的单位圆。右图为特征值的拉伸结果。

由于特征向量的缩放因子为$\frac{\lambda_i }{\lambda_i+\alpha}$，故沿着特征值较大的方向（如$\lambda_i >>\alpha$）正则化影响较小。而$\lambda_i <<\alpha$将会收缩至几乎为零。因此$L^2$范数将使模型的参数趋近于0附近。

通过L^2正则化，在显著减小目标函数方向上的参数会被相对完整的保留，而对于无助于目标函数减小的方向（对应Hessian矩阵较小的特征值），其参数的改变将不会显著参加梯度，因此其在训练过程中会因正则化项而衰减至0。

此外，在线性回归的平方误差损失函数中引入二范数，即在原来逆矩阵的基础上加入对角阵，使得矩阵求逆可行，同时缓解过拟合的问题。而由于加入的对角矩阵，其就像一条“山岭”一样，因此，L^2在统计学中也被称为岭回归或Tikhonov正则。

线性回归目标函数一般为：

$$ L(w)=(Xw-y)^T(Xw-y) \tag{a} $$ $$ \nabla_wL(w)=X^T(Xw-y)=0\tag{b}
$$

固有：

$$ X^TXw=X^Ty \tag{c} $$

即：

$$ w=(X^TX)^{-1}X^Ty \tag{d} $$

当引入L^2正则项后，目标函数变为：

$$ L(w)=(Xw-y)^T(Xw-y)+\frac12\alpha w^Tw\tag{e} $$

则： $$ \nabla_wL(w)=X^T(Xw-y)+\alpha w^T=0\tag{f} $$ $$ w=(X^TX+\alpha I)^{-1}X^Ty \tag{g}
$$

此即为岭回归。

上式中第（1）项X^TX即为线性回归标准最小二乘项，第（2）项αI即为正则化项，为一对角阵。

另外从另一个角度理解，当X^TX非正定时（某些参数线性相关），其无法求逆。此时解决的方法一般包括两种：

（a）通过PCA构建线性无关组，进行降维处理，删除无关特征，求逆；

（b）二范数正则化，即通过增加取值较小的对角阵使得矩阵求逆可行。由于PCA方法处理后其剔除了一些贡献程度较小的特征，而二范数只是将部分无关特征权值缩放置0附近，因此二范数也被称为Soft-PCA。 L^1 范数正则化

L^1 范数形式如下：

$$ \Omega(\theta)=||w||_1=\sum_i|w_i|\tag{11} $$

如上式所示，L^1范数为各参数的绝对值之和。（L^1范数求导、优化困难，因此较L^2范数相比使用较少）对于，L^1范数其目标函数如下所示：

$$ \tilde J(w;X,y)=J(w;X,y)+\alpha ||w||_1\tag{12}
$$

其对应的梯度如下：

$$ \nabla_w\tilde J(fw;X.y)=\nabla_wJ(w;X,y)+\alpha sign(w)\tag{13}
$$

上式中，$sign(w)$为符号函数，其取值结果只与个权值w的正负号有关。

同理，这里令$w^*=arg,min_wJ(w)$，我们可以将 L^1正则化目标函数的二次近似解分解为关于各参数求和的形式：

$$ \hat J(w)=J(w^)+\sum_i[\frac12H_{i,i}(w_i-w_i^)^2+\alpha |w_i|]\tag{14}
$$

对每一维w_i求梯度，以最小化式（14）。由于W^*为$J(w)$的最优解，因此有$\nabla_wJ(w)=0$，故：

$$ \nabla_{w_i} \hat J(w)=[H_{i,i}(w_i-w_i^)+\alpha sign(w_i^)]=0\tag{15}
$$

即：

$$ w_i=w_i^-\frac{\alpha}{H_{i,i}}sign(w_i^) $$

对w_i^*的正负号分类讨论，则上式将等价于：

$$ w_i=sign(w_i^)max\lbrace|w_i^|-\frac{\alpha}{H_{i,i}},0\rbrace\tag{16}
$$

当$w_i>0$时，会有两种结果：

a. $w_i^*\leq\frac{\alpha}{H_{i,i}}$。此时式（16）的最优值为$w_i=0$。

b. $w_i^ >\frac{\alpha}{H_{i,i}}$。此时则有$w_i=w_i^-\frac{\alpha}{H_{i,i}}$，即$w_i$在此方向上向0收缩$\frac{\alpha}{H_{i,i}}$个距离。

同理，当$w_i<0$时，w_i也将等于0或向0收缩。

与L^2范数相比，L^1范数正则化使得部分参数为0。因此，L^1范数会产生更稀疏的解，该性质已被大量应用于特征的选择机制。著名的LASSO回归即将L^1范数引入至线性模型中，并使用最小二乘代价函数。通过L^1正则化使得部分权值为零，而忽略相应的特征。如图2所示。

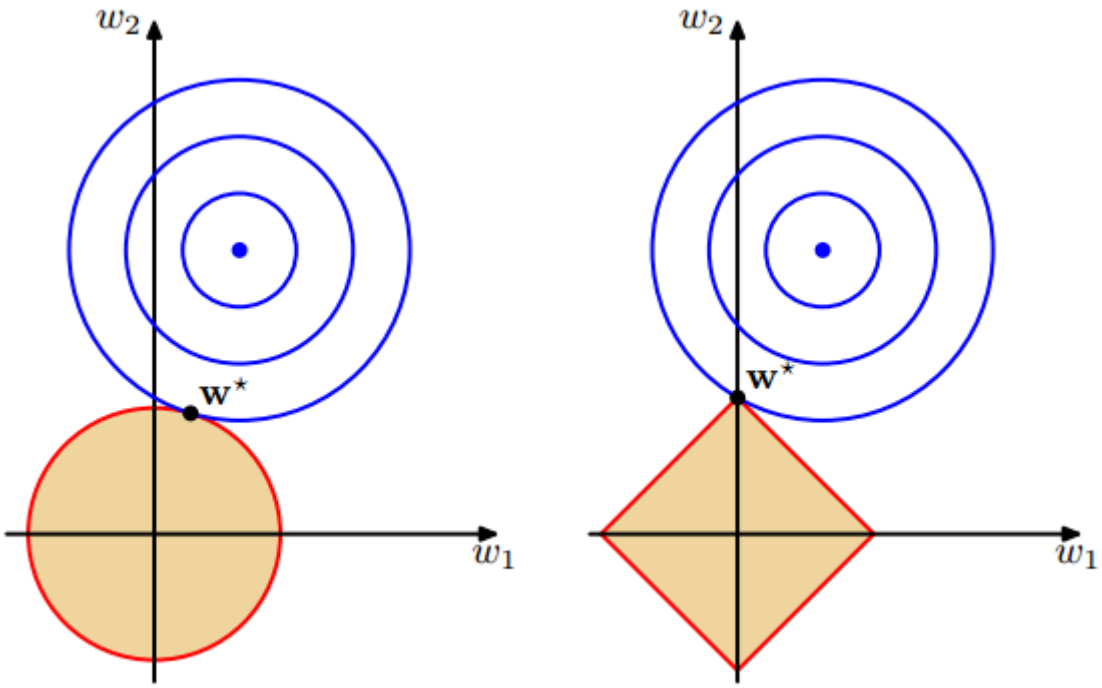

图2. 一范数与二范数示意图

上图中，蓝色的圆圈表示原问题可能的解范围，橘色的表示正则项可能的解范围。而整个目标函数（原问题（损失函数）+正则项）有解当且仅当两个解范围相切。从上图可以很容易地看出，由于L^2范数解范围是圆，所以相切的点有很大可能不在坐标轴上，而由于L^1范数是菱形，其相切的点更可能在坐标轴上。因此其只有一个坐标分量不为零，其它坐标分量为零，即L^1的解是稀疏的。

L^2范数正则化，模型权重服从高斯分布，L^1范数正则化，模型参数服从各向同性的拉普拉斯分布。即L^1正则化项$\alpha \Omega(w)=\alpha \sum_i|w_i|$与通过MAP贝叶斯推断最大化对数先验项等价。此外，相较于L^2而言，采用L^1模型的鲁棒性较差（每次至零的权值均不相同）。

$$ log(p(w))=\sum_ilogLaplace(w_i;0,\frac1{\alpha})=\sum_ilog(\frac{1}{\frac{2}{\alpha}}(\frac{-|w_i|}{\frac{1}{\alpha}}))=\sum_ilog(\frac{\alpha}{2}\cdot(-\alpha|w_i|))\=\sum_ilog(\frac{\alpha}{2})+\sum_ilog(-\alpha|w_i|)=-\alpha||w||_1+nlog\alpha-nlog2\tag{17}
$$

(拉普拉斯分布：$Laplace(x;\mu,\gamma)=\frac{1}{2\gamma}exp(\frac{-|x-\mu|}{\gamma})$)

由于目标函数是关于w的最大化进行学习，因此可以忽略$nlog\alpha-nlog2$。

L^0范数

L^0范数如下所示：

$$ ||w||_0=#(i);with;w_i\neq 0\tag{18} $$

L^0范数即为模型参数中不为0的数目。在实际问题中，模型一般较为复杂，参数数目较多，因此求解L^0范数为NP难问题，故一般不使用。在过去的前几年做压缩感知即稀疏表达时一般会使用0范数，但由于其优化困难因此会用1范替代。

Frobenius范数

Frobenius范数如下所示：

$$ ||w||F=(tr(w^tw))^{\frac12}=(\sum_i^m\sum_j^n|w{i,j}|^2)^{\frac12}\tag{19} $$

从上式可以明显看出，矩阵的Frobenius范数就是将矩阵张成向量后的L^2范数。（在此就不做推导了）

各类范数一般作为各类机器学习模型权值的约束项（惩罚项）出现在目标函数中，训练过程以使得结构风险最小化，防止过拟合的发生。