#### 正则化在机器学习中的应用

**1. 引言**

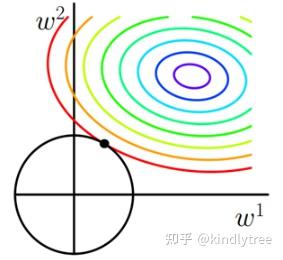
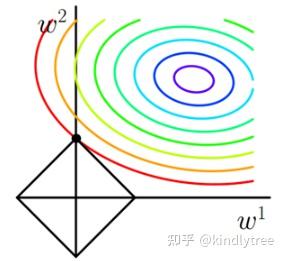
正则化是机器学习中的关键技术，旨在防止过拟合并提高模型的泛化能力。它通过向模型的损失函数中引入惩罚项，控制模型的复杂度，以适应更广泛的数据。

1. **参数正则化**

常用的正则化项是权重衰减项（weight decay）深度学习中的参数包括每个神经元中的权重系数和偏置。由于每个权重会指定两个变三量之间相互作用的方式，因而拟合权重所需要的数据量要比拟合偏置多得多。相比之下，每个偏置只控制一个变量，即使不对它做正则化也不会产生太大方差，正则化的方式不对反而还会增加算法的偏差。这是正则化的对象只包括权重而不包括偏置的原因。

在权重衰减中，正则化项是以范数的形式表示的，常用的范数包括L2范数和L1范数。回忆一下当这两种范数作为正则化项被应用在线性回归中时，分别对应的是[岭回归](https://www.zhihu.com/search?q=%E5%B2%AD%E5%9B%9E%E5%BD%92&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":919700199}" \t "_blank)和 LASSO回归。L2范数作为正则化项时，其作用是使权重系数更加接近原点。引入权重衰减后，在每一步的梯度更新之前，权重向量都会被收缩。整体来看，这使得在显著减小目标函数方向上的权重保留完好，无益于目标函数减小的方向所对应的分量会因正则化而被逐渐地衰减掉。从泛化误差的角度来说，[L2范数](https://www.zhihu.com/search?q=L2%E8%8C%83%E6%95%B0&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":919700199}" \t "_blank)能够感知具有较高方差的输入，与这些输入特征相关的权重则被收缩。相比之下，L1范数和L2范数有本质上的区别。L1正则化得到的是稀疏的解，它将一部分较小的权重直接砍掉。这样做可以从可用的特征子集中选择出有意义的特征，从而简化学习问题。

以二维空间进行几何说明：L1范数和L2范数的等值线（或等高线）分别为棱形（以原点为坐标的水平正方形旋转45度）和以圆点为中心的圆。损失函数或误差表面（error surface）的等高线在满足约束条件的区域下的最优值。



从图中可以看出，凸函数在满足约束L1-norm的条件下的最优值有更多的可能在4个顶点上（凸函数中等值线值低的曲线上的点首先与L1-norm的棱形上的顶点相交，等值线增大会和棱形的四条边中的某些边相交，也有一定的可能性是等值线值低的曲线与四条边界的某一条边相切，但概率要小），在其他边界的可能性相对较低，而在满足L2-norm的条件下的最优解在圆形边界上的每一个点的可能性相同（函数的梯度和圆的[切线](https://www.zhihu.com/search?q=%E5%88%87%E7%BA%BF&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":2849557497}" \t "_blank)的梯度在交点处平行，和[拉格朗日乘数法](https://www.zhihu.com/search?q=%E6%8B%89%E6%A0%BC%E6%9C%97%E6%97%A5%E4%B9%98%E6%95%B0%E6%B3%95&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":2849557497}" \t "_blank)的机理一样）。

**3. 广义正则化方法**

3.1 数据增强

训练模型的质量在很大程度上取决于训练数据。除了选择噪声较小的训练数据，还可以通过正则化来提升训练数据的质量。正则化处理数据的一个目的是执行预处理和特征提取，从而将[特征空间](https://www.zhihu.com/search?q=%E7%89%B9%E5%BE%81%E7%A9%BA%E9%97%B4&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":919700199}" \t "_blank)或数据分布修改为其他形式；另一个目的是通过生成新样本来创建具有更大容量，甚至是无限容量的增强数据集。这两个目的之间相互独立，因而可以结合起来使用。

对训练数据正则化的做法是在训练数据集上施加变换，从而产生新的训练数据集。变换的形式是以满足某种概率分布的随机变量为自变量的函数，最简单的实例就是向数据中添加随机的高斯噪声。由于提升机器学习模型泛化性能最直接的办法就是使用更多的数据进行训练，因而使用随机参数的变换可以用于生成“假”数据，这种方法被称为数据集增强（data augmentation）。

数据集增强的对象通常只包括输入层和隐藏层，而不包括输出层。这样做的效果是将训练数据集映射到一个新的概率分布上，并用这个新分布来计算经验风险函数。变换中参数的随机性使我们可以在不同的参数下生成多组新数据，从而通过数据量的增加来减小期望风险和经验风险之间的差别。

3.2 早停（Early Stopping）

早停是一种简单且有效的正则化技术，通过在训练过程中监测验证集误差，及时停止训练以防止模型过度拟合。

3.3 模型集成（Dropout）

Dropout是一种集成方法，通过结合多个模型来降低[泛化误差](https://www.zhihu.com/search?q=%E6%B3%9B%E5%8C%96%E8%AF%AF%E5%B7%AE&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":919700199}" \t "_blank)。之所以说 Dropout属于基于训练数据的正则化，是因为它可以构造不同的数据集来训练不同的模型，每个数据集通过对原始数据集进行“有放回采样”得到。

Dropout的关键想法是在训练期间从神经网络中随机丢弃神经元及其连接，得到简化的网络。而在测试时，一个简单的小权重网络就可以逼近所有这些简化网络预测的平均效果。其优点在于计算简单方便，同时还具有对不同的模型和训练过程的普适性。但 Dropout对训练集容量的要求很高，少量训练样本并不能发挥其优势。

从输入到输出的映射必须具有某些特质才能很好地适应数据，而对输入-输出映射进行假设的方法正对应着网络结构的选择，这催生了基于网络架构的正则化方法。对映射的假设既可以关注深度网络中不同层次的具体操作，也可以关注层与层之间的连接方式。基于网络架构的正则化通常会简化关于映射的假设，再让网络架构逐步逼近简化后的映射。这限制了模型的搜索空间，为找到更好的解提供了可能性。

**4. 正则化的优势**

* 提高泛化能力：正则化有助于模型在未见过的数据上表现良好。
* 控制过拟合：防止模型在训练集上过度拟合，增加对新数据的适应能力。
* 特征选择：L1正则化可以帮助确定影响最大的特征，提高模型的可解释性。

**5. 结论**

正则化技术在机器学习中发挥着重要作用，有效控制过拟合并提高模型的泛化能力。除了常见的L1和L2正则化外，广义正则化方法如数据增强也为模型训练提供了新的途径。理解和应用这些正则化技术是提升模型性能和稳定性的关键。