Google机器学习速成课程笔记

**机器学习概念**

**机器学习简介**

**深入了解机器学习**

**降低损失**

**使用TF的基本步骤**

**泛华**

**训练集和测试集**

**验证**

**表示法**

**特征组合**

特征组合是指两个或多个特征相乘形成的合成特征，特征的组合可以提供超出这些特征单独能够提供的预测能力。

我们为什么要这么做？

线性学习器可很好地扩展到大量数据，例如vowpal-wabit、sofia0ml

如果不使用特征组合，这些模型的表现度将受到限制

使用特征组合+大量数据是学习高度复杂模型的一种有效策略

预示：神经网络可提供另一种策略

高维度线性模型可受益于使用一种基于梯度的优化方法，叫做 FTRL。该算法的优势是针对不同系数以不同方式调整学习速率，如果某些特征很少采用非零值，该算法可能比较实用（也非常适合支持 L1 正则化）。我们可以使用 [FtrlOptimizer](https://www.tensorflow.org/api_docs/python/tf/train/FtrlOptimizer" \t "_blank) 来应用 FTRL。

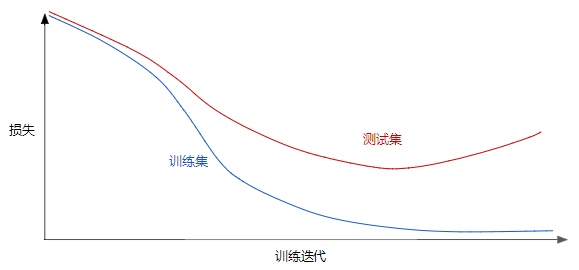
**正则化：简单性**

**逻辑回归**

**分类**

**正则化：稀疏性**

正则化是指降低模型的复杂度以减少过拟合。请查看以下**泛化曲线**，该曲线显示的是训练集和验证集相对于训练迭代次数的损失。



如上图，显示的是某个模型的训练损失逐渐减少，但验证损失最终增加。换言之，该泛化曲线显示该模型与训练集中的数据[过拟合](https://developers.google.cn/machine-learning/crash-course/generalization/peril-of-overfitting)。根据[奥卡姆剃刀定律](https://developers.google.cn/machine-learning/crash-course/generalization/peril-of-overfitting#occam)，或许我们可以通过降低复杂模型的复杂度来防止过拟合，这种原则称为**正则化**。

也就是说，并非只是以最小化损失（经验风险最小化）为目标：

minimize(Loss(Data|Model))

而是以最小化损失和复杂度为目标，这称为**结构风险最小化**：

minimize(Loss(Data|Model) + complexity(Model))

现在，我们的训练优化算法是一个由两项内容组成的函数：一个是**损失项**，用于衡量模型与数据的拟合度，另一个是**正则化项**，用于衡量模型复杂度。

机器学习速成课程重点介绍了两种衡量模型复杂度的常见方式（这两种方式有些相关）：

将模型复杂度作为模型中所有特征的权重的函数。

将模型复杂度作为具有非零权重的特征总数的函数。（[后面的一个单元](https://developers.google.cn/machine-learning/crash-course/regularization-for-sparsity/l1-regularization)介绍了这种方法。）

如果模型复杂度是权重的函数，则特征权重的绝对值越高，对模型复杂度的贡献就越大。

我们可以使用 L2 正则化公式来量化复杂度，该公式将正则化项定义为所有特征权重的平方和：

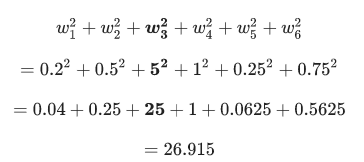
C:\Users\Administrator\AppData\Roaming\Tencent\Users\188512936\TIM\WinTemp\RichOle\L$~HYYNX}BTP(CP86A0BO1G.png

在这个公式中，接近于 0 的权重对模型复杂度几乎没有影响，而离群值权重则可能会产生巨大的影响。

例如，某个线性模型具有以下权重：

C:\Users\Administrator\AppData\Roaming\Tencent\Users\188512936\TIM\WinTemp\RichOle\1WRR`GP05EZXK`RKWKFW2WH.png

L2 正则化项为 26.915：



但是 w3（上述加粗内容）的平方值为 25，几乎贡献了全部的复杂度。所有 5 个其他权重的平方和对 L2 正则化项的贡献仅为 1.915。

模型开发者通过以下方式来调整正则化项的整体影响：用正则化项的值乘以名为 **lambda**（又称为**正则化率**）的标量。也就是说，模型开发者会执行以下运算：

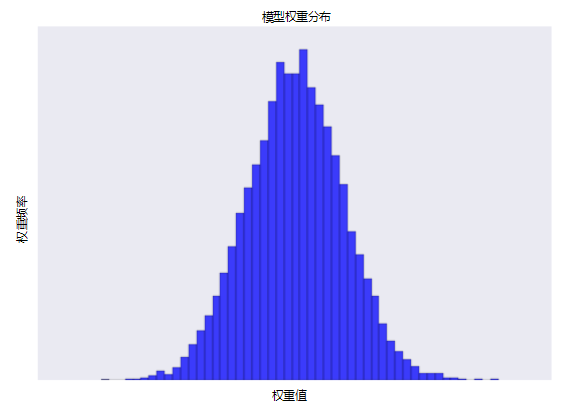
minimize(Loss(Data|Model) + λ complexity(Model))

执行 L2 正则化对模型具有以下影响：

使权重值接近于 0（但并非正好为 0）

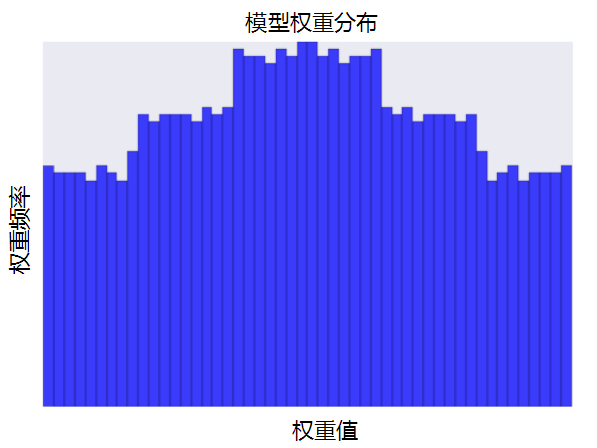
使权重的平均值接近于 0，且呈正态（钟形曲线或高斯曲线）分布。

增加 lambda 值将增强正则化效果。 例如，lambda 值较高的权重直方图可能会如图 2 所示。



**权重直方图**

降低 lambda 的值往往会得出比较平缓的直方图，如图 3 所示。



**较低的 lambda 值得出的权重直方图**

在选择 lambda 值时，目标是在简单化和训练数据拟合之间达到适当的平衡：

如果您的 lambda 值过高，则模型会非常简单，但是您将面临数据欠拟合的风险。您的模型将无法从训练数据中获得足够的信息来做出有用的预测。

如果您的 lambda 值过低，则模型会比较复杂，并且您将面临数据过拟合的风险。您的模型将因获得过多训练数据特点方面的信息而无法泛化到新数据。

*注意：将 lambda 设为 0 可彻底取消正则化。 在这种情况下，训练的唯一目的将是最小化损失，而这样做会使过拟合的风险达到最高。*

理想的 lambda 值生成的模型可以很好地泛化到以前未见过的新数据。 遗憾的是，理想的 lambda 值取决于数据，因此您需要手动或自动进行一些调整。

**神经网络简介**

**训练神经网络**

**多类别神经网络**

**嵌入**

**机器学习工程**

**生产环境机器学习系统**

**静态训练与动态训练**

**静态推理与动态推理**

**数据依赖关系**

**机器学习现实世界应用实例**

**癌症预测**

**18世纪文字**

**现实世界应用准则**

**总结**