**泛化部分基础知识总结**

//参考《深度学习》5.2

在先前未观测到的输入上表现良好的能力被称为**泛化**

在训练集上计算的度量误差被称为**训练误差**

**泛化误差**（也被称为测试误差）被定义为新输入的误差期望。

用度量模型在训练集中分出来的测试集样本上的性能，来评**估机**器学习模型的**泛化误差**。

机器学习和优化不同的地方在于，希望泛化误差（不只是训练误差）也很低。

训练误差和测试误差之间的直接联系是，随机模型训练误差的期望和该模型测试误差的期望是一样的。

决定机器学习算法效果是否好的因素：

1. 降低训练误差。————欠拟合

2. 缩小训练误差和测试误差的差距。————过拟合

**欠拟合**是指模型不能在训练集上获得足够低的误差。

**过拟合**是指训练误差和和测试误差之间的差距太大

模型的**容量**是指其拟合各种函数的能力。

一种控制训练算法容量的方法是选择**假设空间**（hypothesis space），即学习算法可以选择为解决方案的函数集。

**奥卡姆剃刀原则**：在同样能够解释已知观测现象的假设中，我们应该挑选”最简单”的那一个。

Vapnik-Chervonenkis维度**VC维**是一种量化模型容量的方法。VC维定义为该分类器能够分类的训练样本的最大数目。

从预先知道的真实分布 p(x,y) 预测而出现的误差被称为**贝叶斯误差**

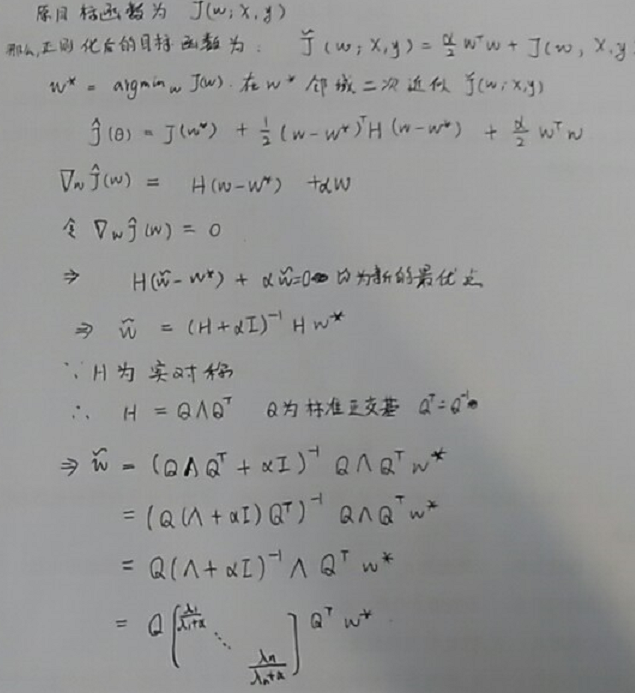
机器学习的**没有免费午餐定理**表明，在所有可能的数据生成分布上平均之后，每一个分类算法在未事先观测的点上都有相同的错误率。

**正则化**是指我们修改学习算法，使其降低泛化误差而非训练误差。

**L1L2范数区别**

//参考《深度学习》7.1

L2范数：

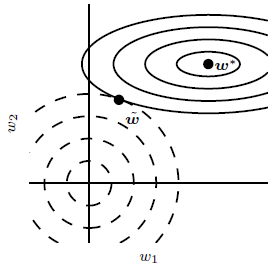


我们可以看到权重衰减的效果是沿着由H 的特征向量所定义的轴缩放 w ∗ 。具体来

说，我们会根据因子缩放与 H 第 i 个特征向量对齐的 w ∗ 的分量。

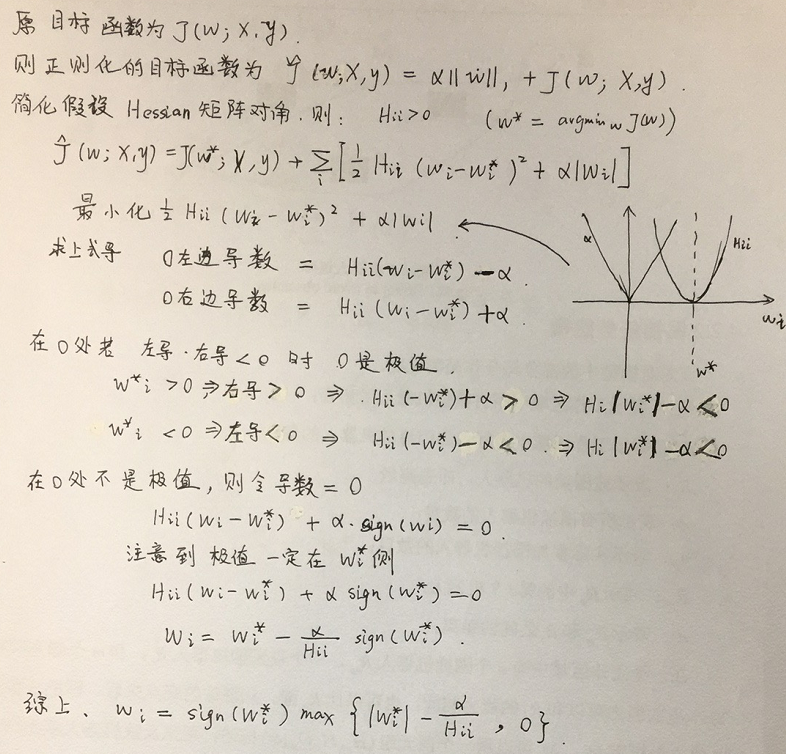
沿着 H 特征值较大的方向 (如 λ i ≫ α)正则化的影响较小。而 λ i ≪ α 的分量将

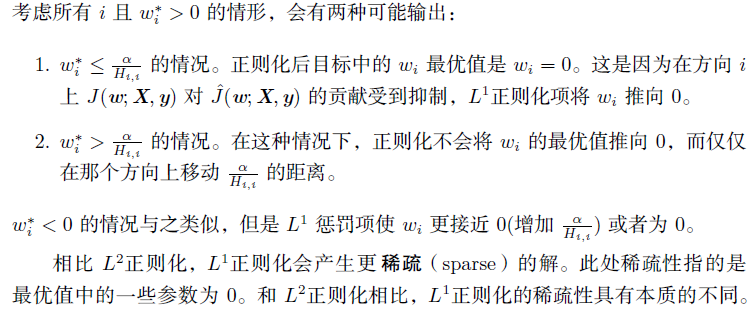
会收缩到几乎为零。



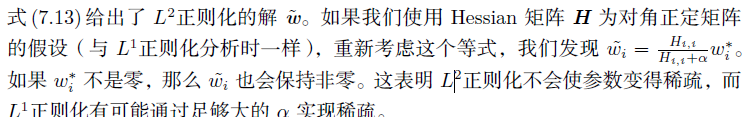
经典的用图表示这个推导，这个图中实线椭圆表示没有正则化目标的等值线。虚线圆圈表示L2 正则化项的等值线。实际上是个三维图，自己脑补，能形象的说明2范数正则化在干什么。建议给面试官画三维图。

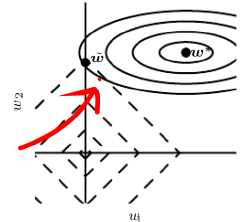
L1范数：











**提前停止**

//参考深度学习 7.8

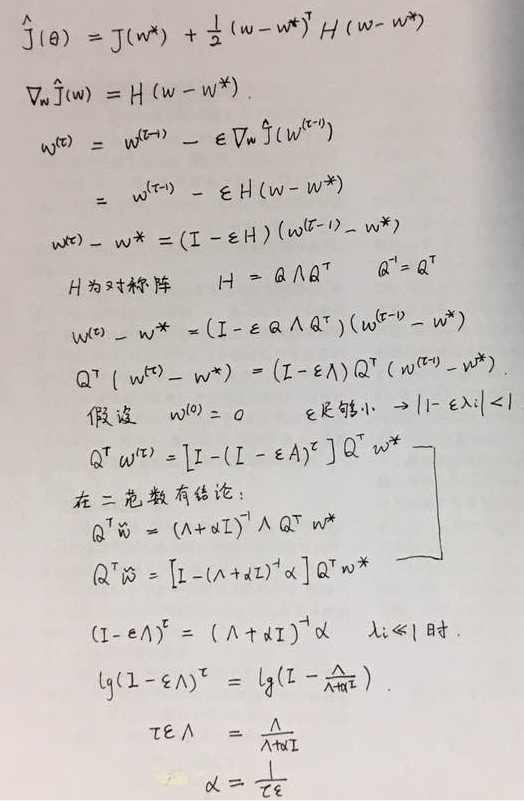
在每次验证集误差有所改善后，我们存储模型参数的副本。当验证集上的误差在事先指定的循环次数内没有进一步改善时，算法就会终止。当训练算法终止时，我们返回这些参数而不是最新的参数。

**提前终止为何具有正则化的效果**

1. 验证集误差的学习曲线是一个 U 型曲线（考虑偏差方差，泛化误差，过拟合角度）

2. 提前终止可以将优化过程的参数空间限制在初始参数值 θ 0 的小邻域内

3. 提前终止相当于 L 2 正则化，证明如下

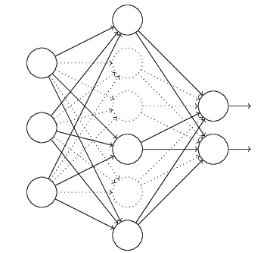


**dropout**

//参考 深度学习7.12

Dropout提供了一种廉价的Bagging集成近似,能够训练和评估指数级的神经网络。具体而言,Dropout训练的集成包括所有从基本的基础网络除去非输出单元形成子网络

超参数的采样概率为 1，隐藏层的采样概率通常为 0.5，输入的采样概率通常为 0.8。

****

**dropout为什么有效：**

1.bagging 2.参数共享 3.噪声

在Bagging的情况下，每一个模型在其相应训练集上训练到收敛。在Dropout的情况下，通常大部分模型都没有显式地被训练。而是在单个步骤中我们训练一小部分的子网络，参数共享会使得剩余的子网络也能有好的参数设定。除了这些，Dropout与Bagging算法一样。Dropout强大的大部分原因来自施加到隐藏单元的掩码噪声。权值的更新不再依赖于有固定关系的隐含节点的共同作用，阻止了某些特征仅仅在其它特定特征下才有效果的情况。迫使网络去学习更加鲁棒的特征。这可以看作是对输入内容的信息高度智能化、自适应破坏的一种形式，而不是对输入原始值的破坏。