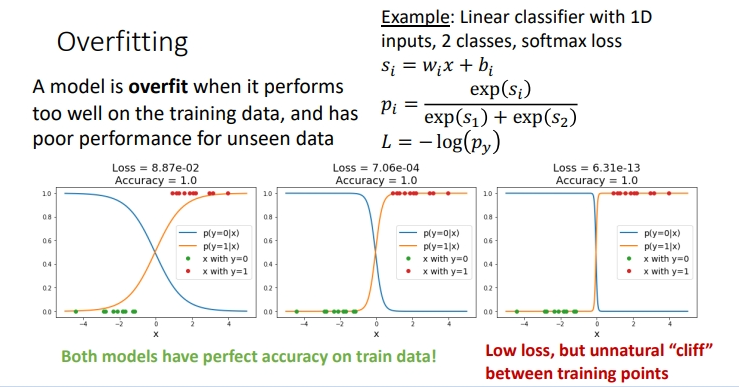
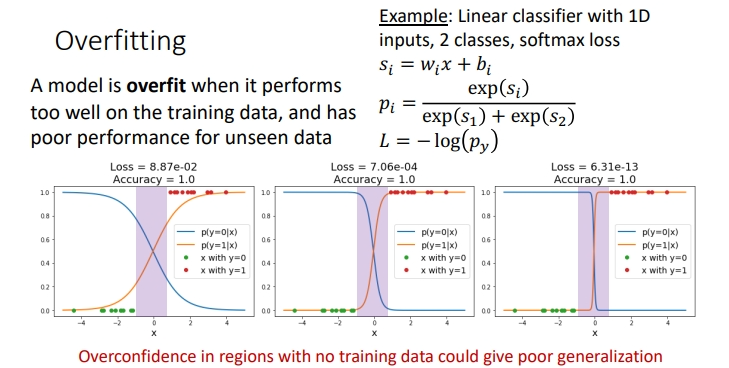
## 过拟合

当模型在训练数据上表现过于出色，而对未见数据表现不佳时，就发生了过拟合。

表现：1在对没有训练数据的区域过度自信可能会导致泛化能力差。

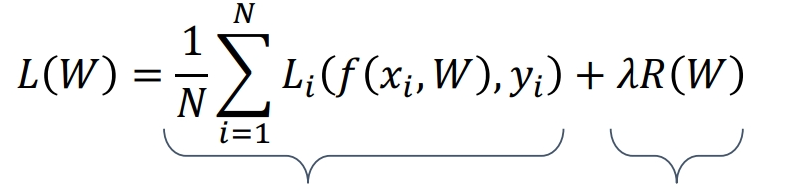
2.在展示精度图中出现损失很低，但训练点之间有不自然的“悬崖，决策边界过于尖锐。





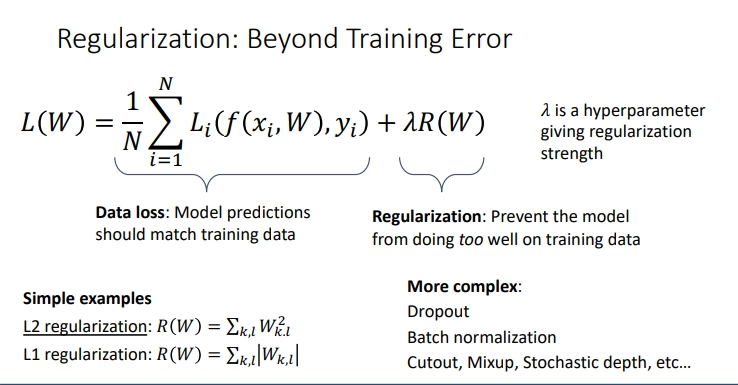
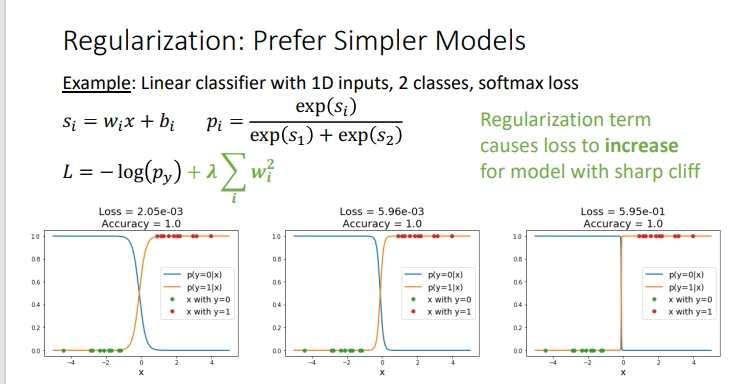
## 正则化

正则化的基本思想是在模型的损失函数中增加一个正则项，以此来惩罚模型参数的大小，鼓励模型学习更简单的、泛化能力更强的模式。

正则化后的损失函数为

λ是一个超参数，给出正则化的强度。

1. 一般有两种正则化方法L1正则化（Lasso正则化）：使用参数的绝对值之和作为正则项。L1正则化倾向于产生稀疏的系数，即许多系数变为零，从而简化模型的解释性。
2. L2正则化（Ridge正则化）：使用参数的平方之和作为正则项。L2正则化会使系数变小，但不会导致系数完全为零

## 优化

在任意方向上的斜率都是该方向与梯度的点积

用微积分来计算一个解析梯度

数值梯度（numerical gradient）是对真实梯度的近似计算，通常用于不可微函数。它的优点是容易编程实现，但缺点是通常是近似解，求解速度较慢。

解析梯度（analytic gradient）直接写出了闭式形式的梯度表达式，适用于可微函数。

损失函数具有较高的条件数：黑森矩阵的最大奇异值与最小奇异值之比较大

Nesterov动量和标准动量之间的区别体现在梯度的计算。

Nesterov动量中，梯度的计算在施加当前速度之后。Nesterov动量可以

理解为往标准动量方法中添加了一个校正因子。

根据每个维度的历史平方和添加梯度的元素级缩放

Adam可以被看作结合RMSProp和具有一些重要区别的动量的变种。

偏差校正：在Adam中，为了解决加权平均计算中的冷启动问题，即初始时动量项和梯度平方项的指数加权移动平均值偏向于0，所以要对动量项和梯度平方项进行了偏差修正。

改变权重使得损失函数最小的方法

1. 随机搜索
2. 沿着斜率走

两种梯度：

1. 数值梯度：近似计算，速度较慢，容易编写

2.解析梯度：精确计算，速度较快，容易出错

梯度下降方法

1. 批梯度下降法

计算整个训练集上的损失函数L对权重W的梯度，即∇L。

使用这个梯度来更新权重W，即W = W - α∇L，其中α是学习率。

重复以上步骤，直到收敛

2. 随机梯度下降法

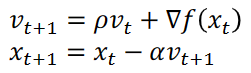
在每次迭代中，仅随机选择一个训练样本，计算该样本上的损失函数L对权重W的梯度，即∇L。

使用这个随机梯度来更新权重W，即W = W - α∇L，其中α是学习率。

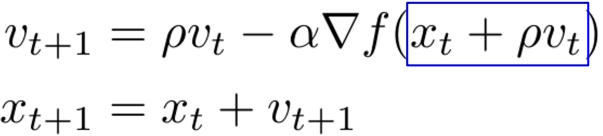
重复以上步骤，直到收敛

随机梯度下降法的问题：

1. 如果损失函数在一个方向上迅速变化，而在另一个方向上缓慢变化，那么梯度下降算法会怎么进行更新。在平坦维度上进展非常缓慢，在陡峭维度上会有抖动
2. 失函数具有较高的条件数：黑森矩阵的最大奇异值与最小奇异值之比很大。
3. 会被困在损失函数的鞍点或极小值点，此时导数为0
4. 使用小批量进行随机梯度下降可能会有噪声
5. 随机梯度与动量结合SGD



1. Nesterov Momentum算法



1. AdaGrad（一种自适应梯度下降算法）

主要思想是根据每个参数的历史累积梯度平方值来调整学习率，以解决学习率难调的问题。

特点：在“陡峭”方向上的梯度下降进展被抑制；在“平坦”方向上的梯度下降进展被加速。

1. RMSProp

引入了指数衰减机制，使得历史累积梯度平方和只与最近一段时间的梯度平方相关，而与更早的梯度平方无关。

