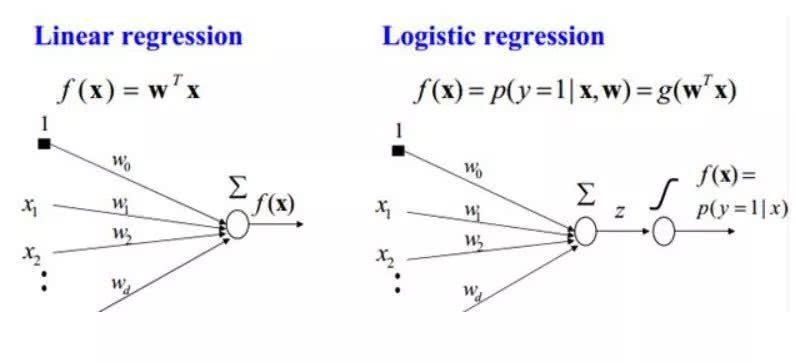
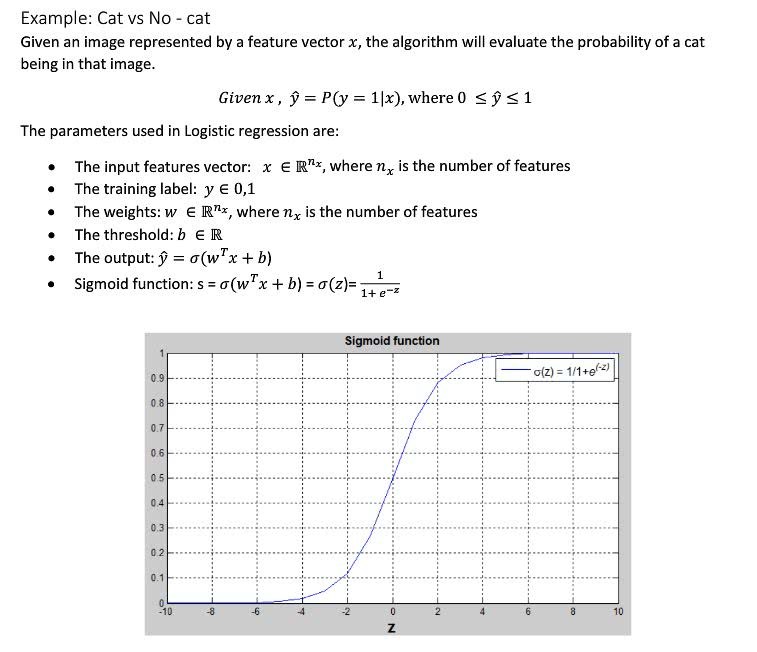
神经网络基础

**1，逻辑回归模型**

逻辑回归模型可以看成是最简单的神经网络，它只有一个神经元。逻辑回归模型是一个经典的二元分类算法，它将输入特征的线性组合通过Sigmoid函数映射到(0，1)之间作为输出值，表示预测样本为正样本的概率。

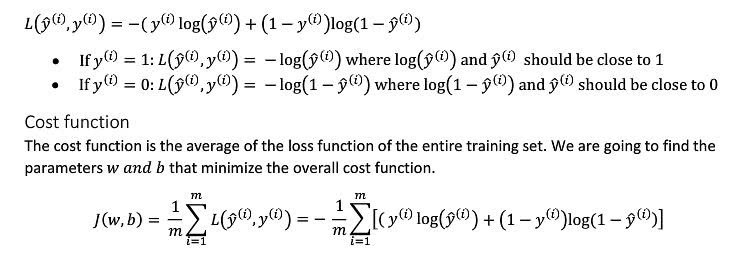




**2，交叉熵损失函数**

为了求解逻辑回归模型中的w和b，需要定义一个目标函数。机器学习中的目标函数通常由损失函数和正则化项组成。其中损失函数衡量模型对训练集样本的拟合程度，函数值越小说明模型对训练集样本偏差越小，拟合程度越好，通常通过提高模型的复杂性能够降低模型对训练集样本的偏差。正则化项描述模型的复杂程度，正则化项越小说明模型越简单。通常较简单的模型的预测结果较稳定，具有更强的抗噪能力，即较低的方差，在未知数据上的泛化能力也更好。偏差和方差均衡是机器学习模型需要解决的普遍问题。

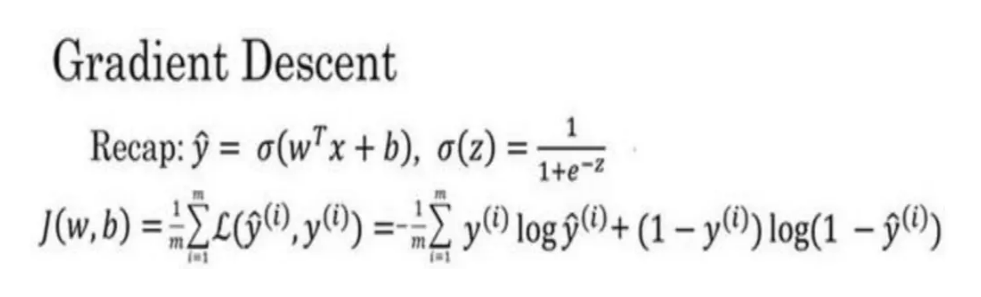
线性回归模型采用的损失函数为平方损失函数，而逻辑回归采用的损失函数是交叉熵损失函数。交叉熵损失函数的形式可以用极大似然估计算法推导得到。对于多样本的情况，逻辑回归的交叉熵损失函数是关于w和b参数的凸函数，即至多仅有一个极值点，这一优良性质使得它能够很好地被用于w和b的学习。



**3，梯度下降法**

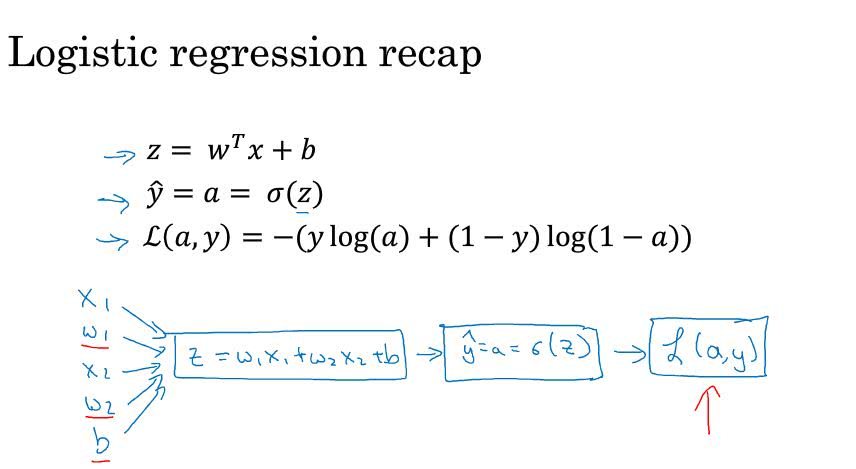
梯度下降法是求解函数最小值的简单的数值迭代算法。梯度下降算法的基本原理是函数的输入参数沿着负梯度发生微小改变，函数值将会下降。梯度下降算法只适用于求解凸函数的最小值，在目标函数有多个极小值时，即非凸函数时，使用梯度下降算法可能会落入局部极小值点。

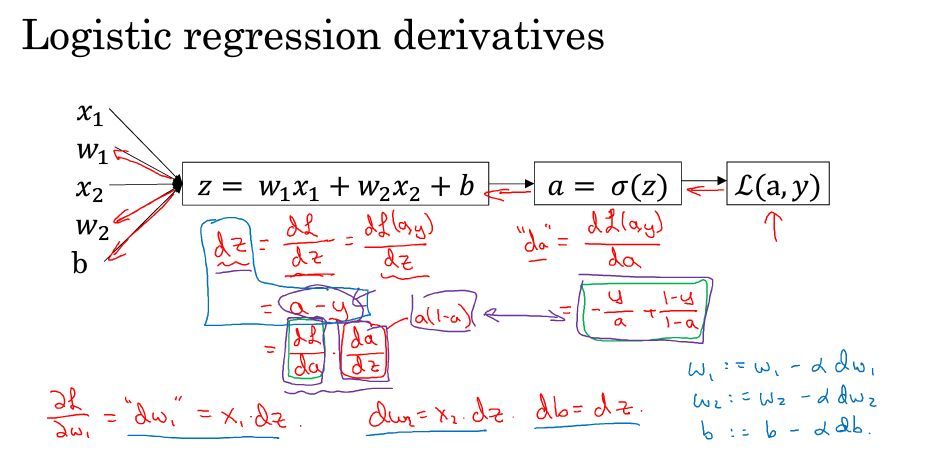
如下图所示：



**4，计算图**

对于存在复杂嵌套关系的函数，可以用计算图的可视化方法简化函数值和对各个参数偏导数的计算过程。计算图可以直观呈现函数中各个变量之间的依赖关系，在计算图的帮助下，应用导数计算的链式法则将非常清晰明了。





**5，向量化**

对于不存在先后依赖关系的相似计算任务，即可以并行的计算任务，许多时候可以用向量化的方式优化计算过程。向量化计算方式和循环计算方式相比，计算效率可以成百上千倍地提高。

在逻辑回归模型或者神经网络中，需要对多个样本进行训练。可以用矩阵形式表达计算公式，将不同的样本放在矩阵的不同列，将同一层不同节点放在矩阵的不同行。使用python中的numpy数组可以计算矩阵形式的公式，并实现高效的向量化计算。使用numpy数组进行矩阵计算时，应当使用2维的ndarray来表示矩阵，若使用0维的ndarray，容易出现各种维度错误。

