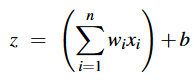
**第五章读书报告**

**09118223 吴亦珂**

**一、Classification: the sigmoid**

1. weight: ，代表输入特征的重要性。

bias term: b，在权重化的输入后面，加上一个偏差项。



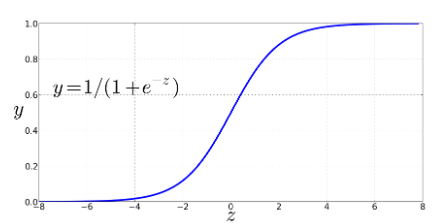
如果写成内积的形式：



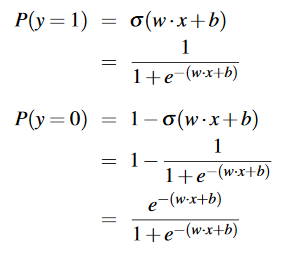
由于我们输出的是一个概率，必须在0到1之间，但是上式的输出实在负无穷到正无穷之间。为了生成这个概率，我们使用了sigmoid function：



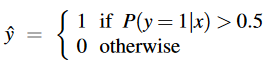
做出函数图像：



最终得出分类函数：



对于一个二分类问题，可以得出决策边界：



1. 选择分类器

Naive Bayes: 进行了条件独立假设，所以在京都方面不及logistic regression，但是在小数据上效果更好，而且更容易实现，训练更快。

Logistics regression：更鲁棒，更准确，在大型文档和数据下效果更好。

**二、Learning in Logistic Regression**

逻辑回归的损失函数采用交叉熵损失函数，优化方法一般使用梯度下降。

**三、The cross-entropy loss function**

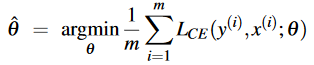
我们需要损失函数描述预测值与真实值之间的差距：



因此，可以采用交叉熵损失函数：



在训练的过程中，我们要做的就是极小化上面的损失函数：

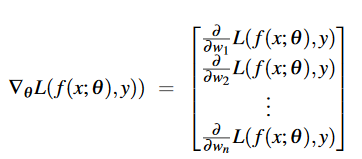


**四、Gradient Descent**

1. 由于逻辑回归中要极小化的函数是一个凸函数，因此只有极小值，而梯度下降可以找到这个极小值点。梯度下降计算出当前点的损失函数上的梯度，然后沿着梯度相反的方向移动。移动的距离取决于学习率，学习率越大，移动的距离越远。

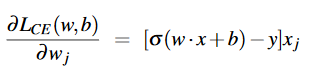


在逻辑回归当中，w可能是一个高维向量，因此在每一个维度上都要进行改变：



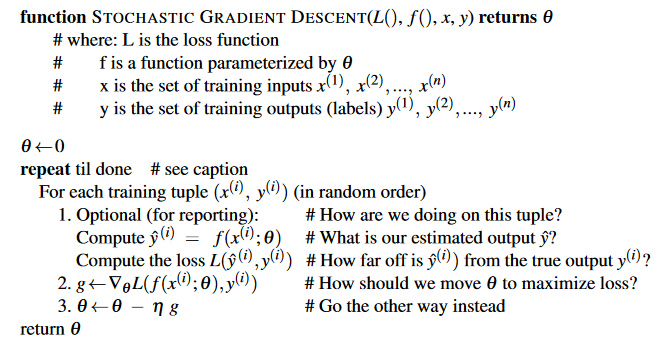


在逻辑回归当中，我们使用了交叉熵损失函数，对其使用梯度下降，则损失函数的倒数为：



可以看成是预测值与实际值的差值再乘上输入值作为权重。

2.随机梯度下降具体算法

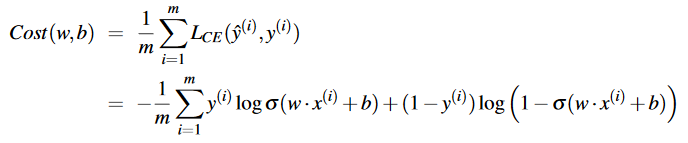


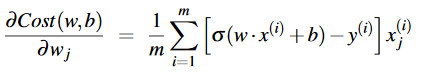
1. 梯度下降有三种：

Stochastic gradient decent：每次选择一个样本进行迭代，但这样可能造成较大的随机性。

Batch gradient decent：每次将所有训练数据输入进行一轮迭代，但这样计算量过大。

Mini-batch gradient decent：每次选择训练数据中的一部分进行一次迭代，相当于两者的这种算法：

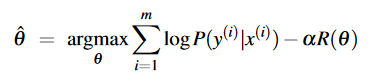




**五、Regularization**

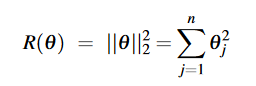
1. 过拟合：模型在训练数据上表现得非常好，但是在测试集上效果并不理想。这并不是一个好的模型，因为我们希望模型的泛化能力较好。

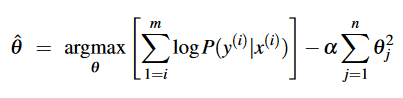
2. 正则项：为了避免模型过拟合，可以在模型后面加上正则项：



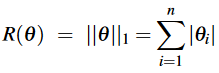
正则项有两种：

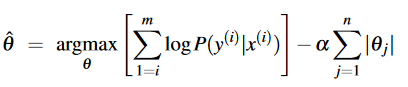
L2 regularization：也叫ridge regularization





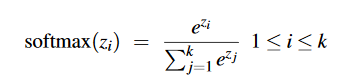
L1 regularization：也叫Lasso regularization



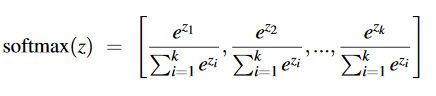


**六、Multinomial logistic regression**

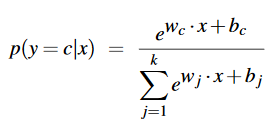
1. 如果需要进行分类的不止两类，可以使用multinomial logistics regression，使用softmax function计算p(y=c|x)：



Softmax将输出的值映射成概率分布，每一个值都在0到1之间，求和为1。

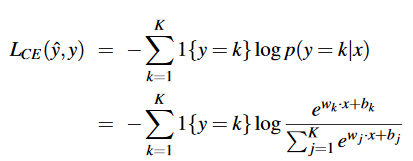


因此最终的函数为：



1. Learning in Multinomial Logistic Regression

Loss function：



Gradient：

