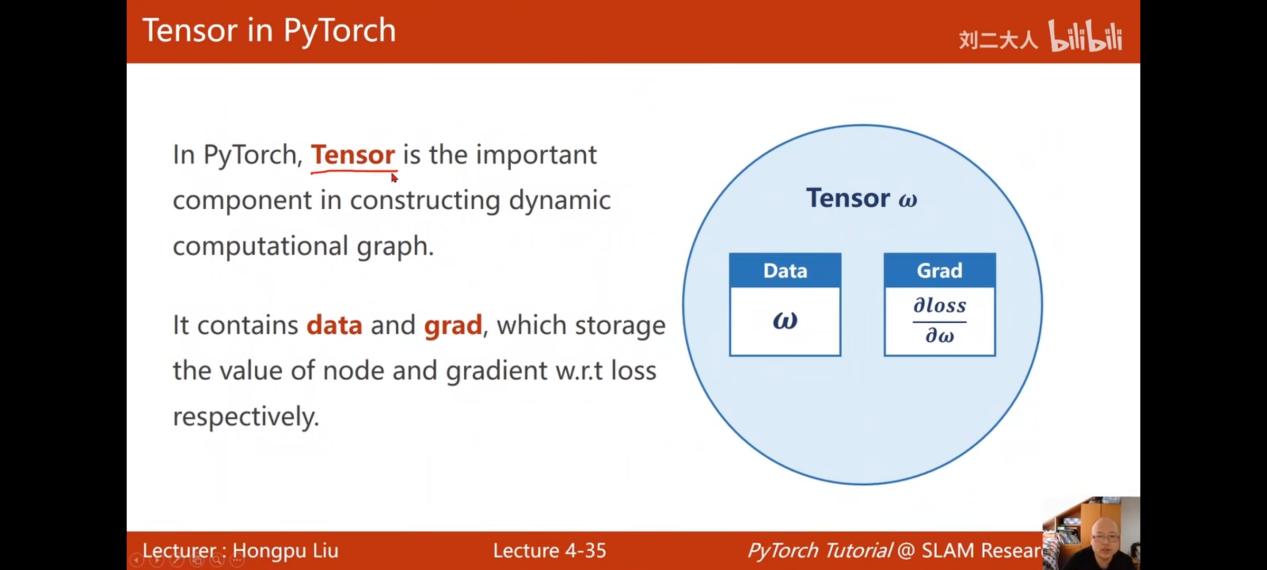
在pytorch里，Tensor是一个重要的组成部分，包括权重和梯度值(这两者也是张量)。Tensor可以是数，或者是n维矩阵。



分为两步，前向算法求出损失值，反向算法求出梯度值

portorch的用法：

Step 1：构建数据集

Step 2：设计模型

Step 3：构造损失函数和优化器

Step 4：训练周期（前馈，反馈，更新）

广播机制：将原来的矩阵扩充满足条件的矩阵

Class torch.nn.Linear(in\_feature, out\_feature,, bias = True)

这里生成了一个线性转换式：

参数说明： in\_features - 每个输入样例的尺寸

Out\_features - 每个输出样例的尺寸

注：通过以上着两个量就可以确定权重和转置矩阵的维数

Bias - 是一个bool量，确定是否需要加上转置量，默认值是Ture

MSE的计算示例





Class torch.nn.MSELoss(size\_average=True, reduce=True)

只需要y,yhead

size\_average -是否除以N

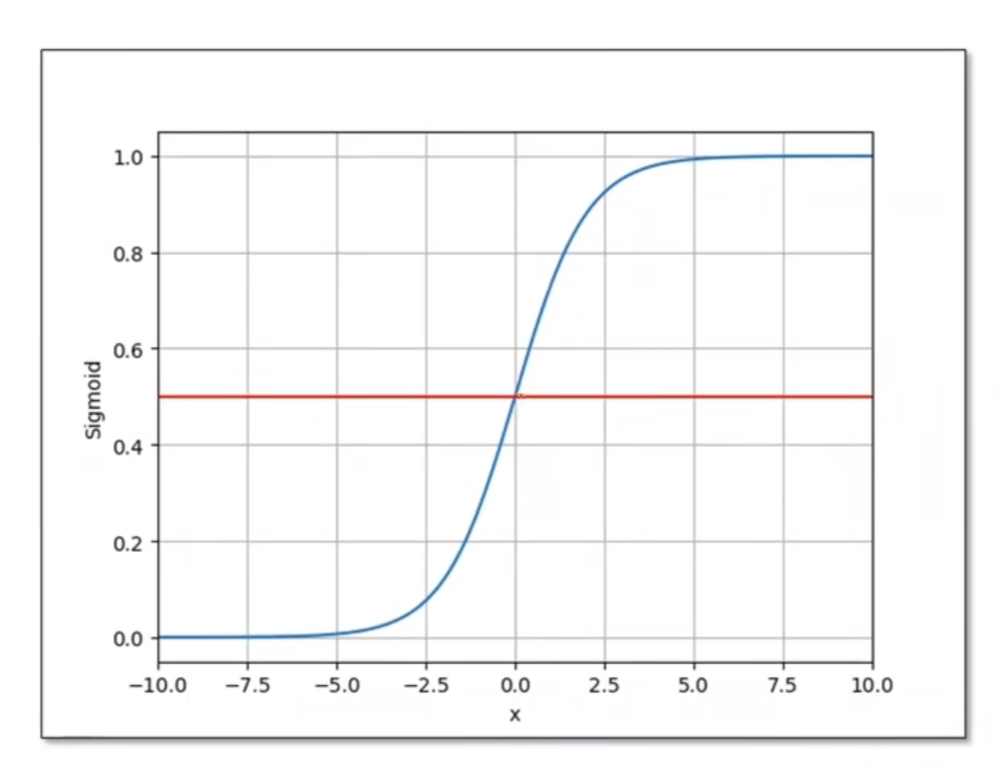
Class torch.optim.SGD(params,lr(学习率))

训练次数不能过多，不然会造成训练集的损失比较小，但是测试集的损失比较大

Logistical regression

数学公式：





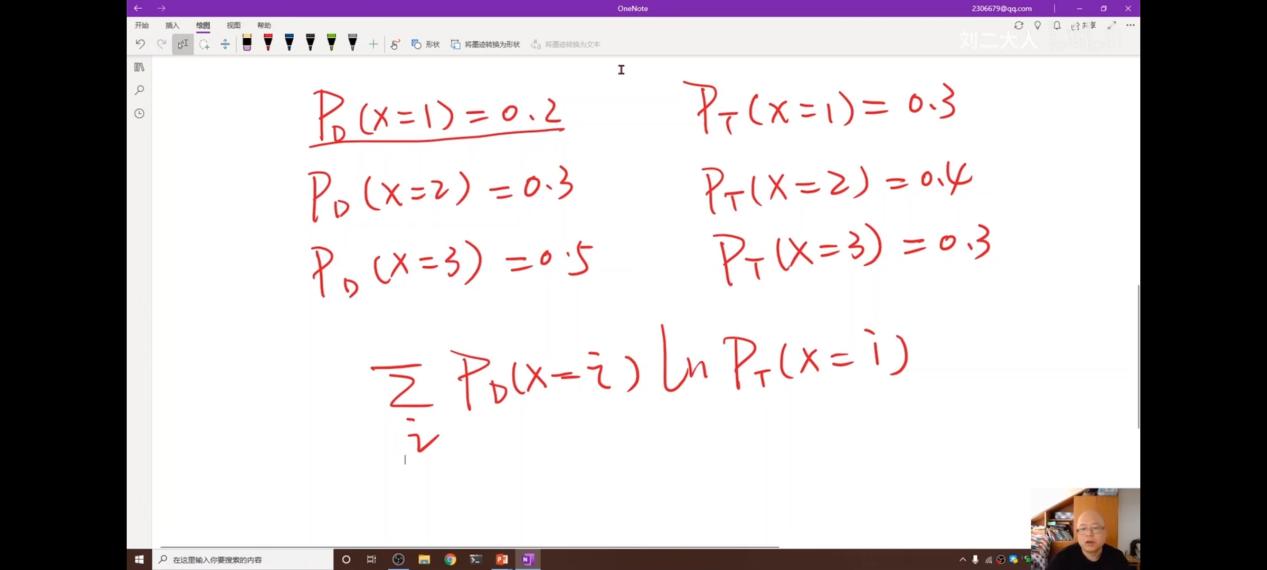
函数图像如上图所示

饱和函数到达极限值附近时，导数趋于0

Sigmoid 函数：1.函数值有极限 2.单调增函数 3.饱和函数

Logistic函数时sigmoid函数里面比较出名的，所以有的教材直接把它称为sigmoid函数

交叉熵：



表示两个分布的差异

二分类的损失函数（BCE Loss)：