**深度学习作业（二）**

1. **基于梯度（Gradient-Based）的优化算法**

1、梯度下降法（GD）

最原始的梯度下降算法是指参数在沿着负梯度方向更新时，在每次迭代中使用**所有的样本**



训练期间，我们可以使用固定的学习率，而不用考虑学习率衰减的问题。它具有朝向最小值的直线轨迹，并且如果损失函数是凸的，则保证理论上收敛到全局最小值，如果损失函数不是凸的，则收敛到局部最小值。尽管程序代码可以向量化计算避免循环，但是可能仍然会很慢地遍历所有样本，特别是数据集很大的时候算法的耗时将成为严重的问题，而且学习的每一步都要遍历所有样本，这里面一些样本可能是多余的，并且对更新没有多大贡献。

2、小批量随机梯度下降法（Mini-bach SGD）

为了克服上述方法的缺点，提出了小批量梯度下降，即每次在更新参数时，不用遍历所有的样本，而只使用**一部分样本**来进行更新。 因此，每次只用小批量的b个样本进行更新学习，其主要过程如下：



比起批量梯度下降，速度更快，因为它少用了很多样本。随机选择样本有助于避免对学习没有多大贡献冗余样本或非常相似的样本的干扰，有助于改善泛化误差。

3、随机梯度下降法（SGD）

随机梯度下降（SGD）每次迭代只对**某一个样本**（xi，yi）执行参数更新，而不是遍历所有样本。因此，每次学习发生在某个样本上，其具体过程如下：



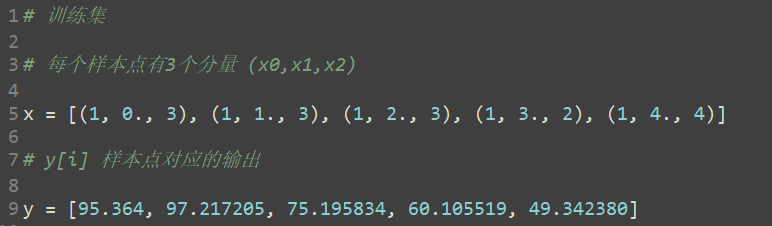
与小批量方法相比，它为学习过程增加了更多的噪声，有助于改善泛化误差。但同时也增加了运行时间。

1. **基于Python的实验分析**

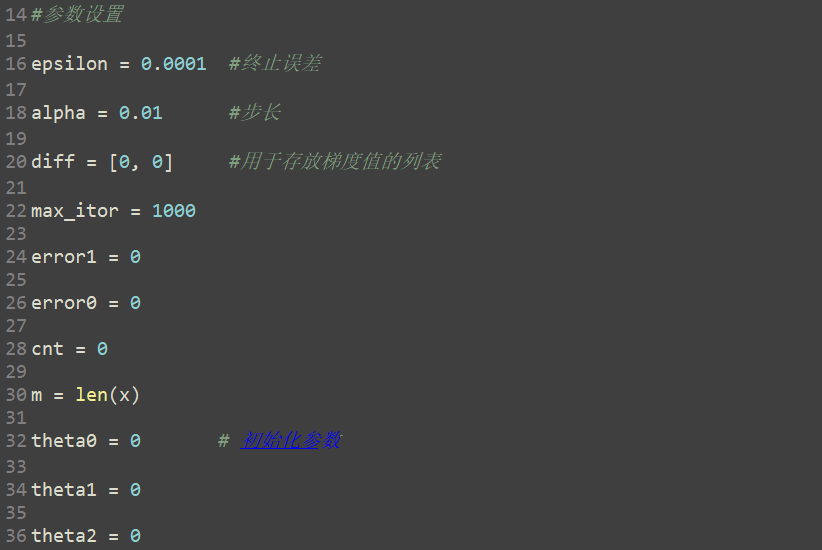
1、实验目标：利用GD解决线性回归问题

2、实验内容

1)数据准备



2）初始化算法参数与模型参数



3）利用for循环得出参数估计结果



1. **代码**

