**随机梯度下降SGD**

1. SGD主要用在大规模稀疏数据问题上，经常用在文本分类和自然语言处理，因为这些数据是稀疏的。可以轻松解决超过的训练样本及的特征。
2. 可以使用sklearn.linear\_model中的SGDClassifier直接来使用SGD方法。通过penalty参数，设置对应的惩罚项：

l2正则化，它对于最后的特征权重的影响是，尽量打散权重到每个特征维度上，不让权重集中在某些维度上，出现权重特别高的特征。

而l1正则化，它对于最后的特征权重的影响是，让特征获得的权重稀疏化，也就是对结果影响不那么大的特征，干脆就拿不着权重。

penalty=“l1”: 对coef的L2范数罚项  
penalty=“l2”: 对coef的L1范数罚项  
penalty=“elasticnet”:

L2和L1的convex组合; (1 - l1\_ratio) \* L2 + l1\_ratio \* L1



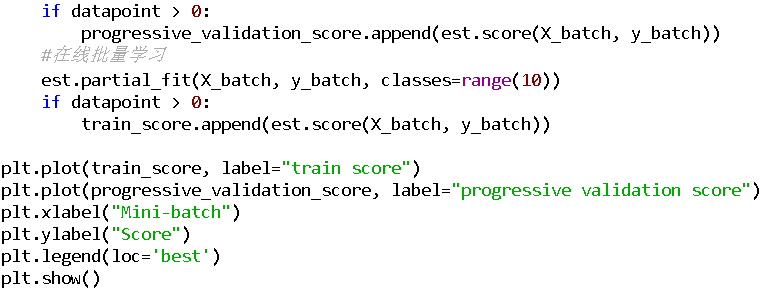


1. sklearn中提供了很多增量学习算法。虽然不是所有的算法都可以增量学习，但是学习器提供了partial\_fit的函数的都可以进行增量学习。事实上，使用小batch的数据中进行增量学习（有时候也称为online learning）是这种学习方式的核心，因为它能让任何一段时间内内存中只有少量的数据。

其中对于分类问题，在第一次调用partial\_fit时需要通过classes参数指定分类的类别。

另外有一点需要考虑，所有的学习器在学习过程中不会对每个样例赋予同样的权重。对于感知机，它对于bad样本会敏感，即使学习器已经学习了很多样本了，而对于SGD和PassiveAggressive，对于这种情况会更鲁棒一点，后者在学习的时候，后来学习样本的权重会随着学习器学习率的下降而降低。







从这个图上的得分，我们可以看出在50个mini-batch迭代之后，数据上的得分就已经变化不大了。但是好像得分都不太高，所以我们猜测一下，这个时候我们的数据，处于欠拟合状态。我们刚才在小样本集合上提到了，如果欠拟合，我们可以使用更复杂的模型，比如把核函数设置为非线性的，但遗憾的是像rbf核函数是没有办法和SGDClassifier兼容的。因此我们只能想别的办法了，比如这里，我们可以把SGDClassifier整个替换掉了，用多层感知神经网来完成这个任务，我们之所以会想到多层感知神经网，是因为它也是一个用随机梯度下降训练的算法，同时也是一个非线性的模型。当然根据机器学习算法使用图谱，也可以使用核估计(kernel-approximation)来完成这个事情。