今日统计了当前数据的正负样本个数。起因是发现蛋白质中有61条名字“重复”的蛋白质条目。

这些不是真的重复，而是最后一位字母大小写不同。而根据输出看到，这两条蛋白挨在一起，序列竟然是一样的。为啥一样的不删除？师姐又不会没有发现它们一样，事出反常必有妖：我发现标签位点不一样！！！ ！！！

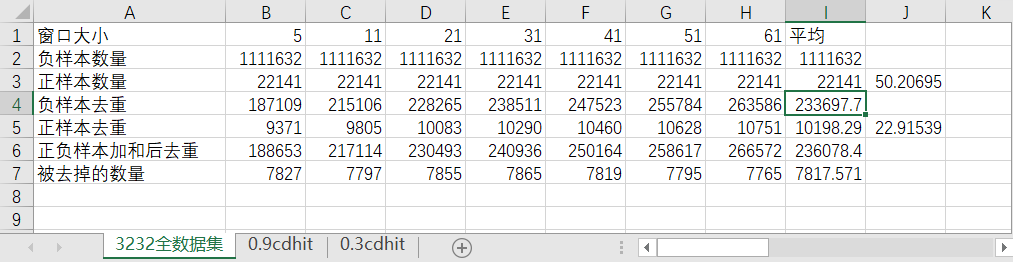
由此，我想到：有的点被标识为正样本，但是有和它同样序列的点标识为负样本，那这个情况下，机器改如何判断它是正样本还是负样本？于是我想找找这些点有多少。

数据集分别是全部，0.9去重，0.3去冗余。

窗口大小我分别截取5，11，21，31，41，51，61。

上结果

我的实验方案是分别处理得到正负样本字符窜，之后把它们放入set中，python集合这个数据类型中不能有一样的数据，起到了去重的效果。

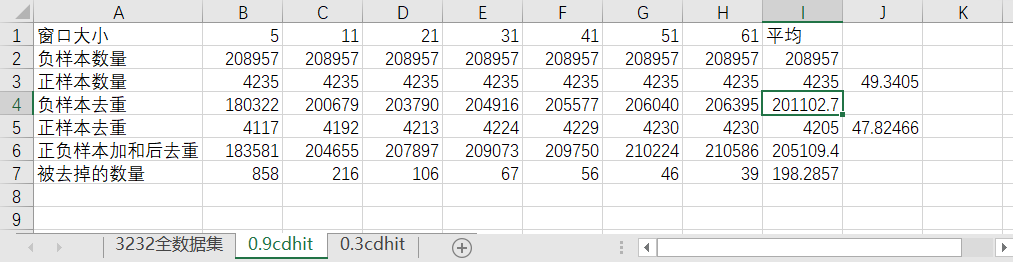


【3232】全数据集。正样本达到2万条之多！负样本则是100万。一旦经过去重，正负样本严重缩水，正样本被腰斩，只有1万条。负样本则被去掉了80%的条目。可见冗余之大。

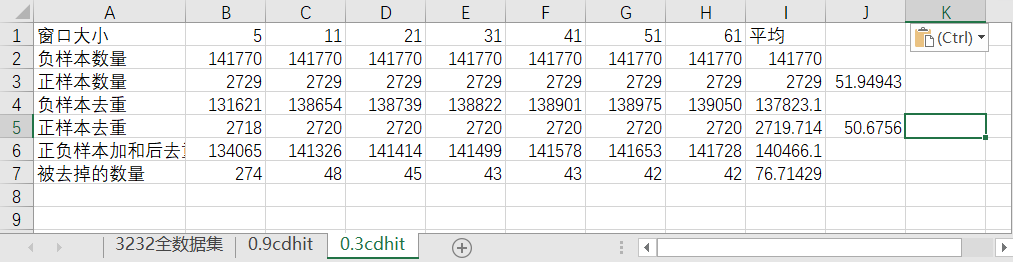
（原来训练效果好，其实都是那些冗余得到的良好效果。）

重点出现在这里：正样本+负样本，它俩之间有一样的么？一样的条目是多少呢？占比正样本多少呢?

我把正负样本做了并集，之后使用集合运算去冗余，算出差值：7817（平均）。也就是1万条数据中，76%的数据，在其他蛋白质上标识为“负样本”。。。 。。。



去重后，4.7%的数据是和负样本有交集的。



去冗余后，2.7%和负样本有交集。

（注意，我使用的是平均数算的。真实情况中我不会使用滑窗为5的去计算。。。而滑窗为5的被去掉的最多。）

其他总结：

1，正样本数量被大幅度削减。

10198——4205——2719，（100%——41%——26%）随着去冗余的进行，正样本大规模被削减。这针对我们的课题是不合理的。我们需要预测所有潜在的药物作用点。而删去的是冗余的蛋白质条目。虽然那部分蛋白质冗余，它们序列中记录的点依然是潜在的位点！！！则我们应该使用10198的正样本。

2，负样本削减数量不太大。

233697-201102-137823。（100%——86%——59%）负样本总共就那么多。去了冗余也就那么多。（其实我个人认为很多都是带有“\_\_\_\_\_”的边界数据。这样的数据真的不想要。）

3，正样本选取1万条的数据。负样本选取0.3去冗余后的数据集。这样选取的负样本冗余性更小。将会取得更好的识别效果。