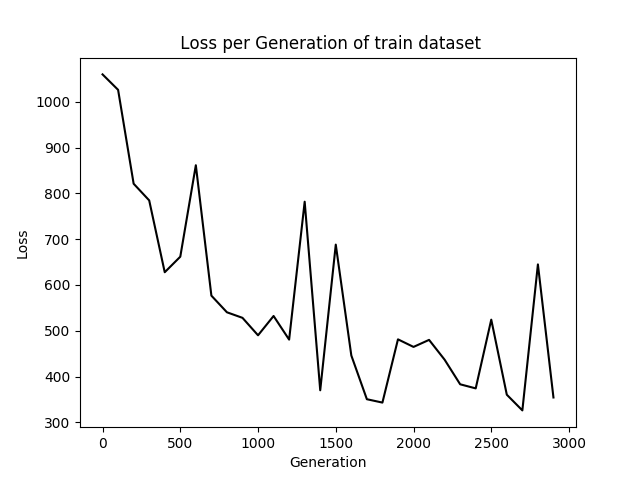
1. 模型的持久化，使用tensorflow中saver=tf.train.Saver()函数创建保存器，使用saver.save()函数可以保存图模型和变量取值，方便在断电或者内存不够时恢复模型。避免因为意外原因没有得到任何结果（任何一次训练都是宝贵的）。
2. 使用合适的评价函数来评价模型，对于一个模型来说，首先是训练数据和测试数据其次就是评价指标，如果没有明确的评价指标，就像一个无头苍蝇，不知道下一步要调节的是什么，模型在训练集和测试集上出了什么错误。

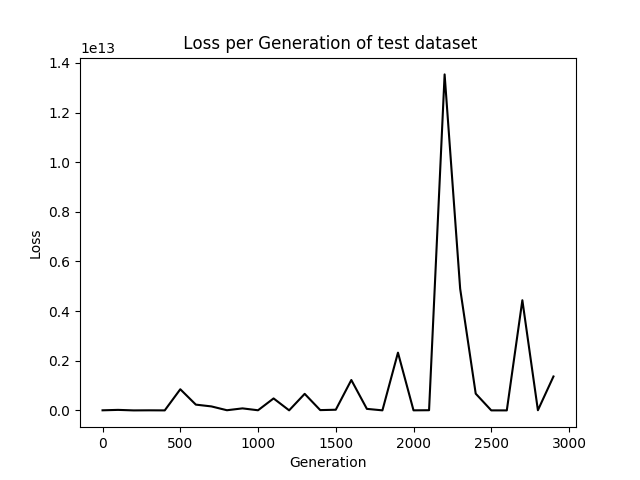
2.1回归模型可以使用：R2系数决定系数：总体平方和SStot,残差平方和SSerr R2=1-(SSerrr/SStot)表示模型对于数据的拟合度，此值越大越好，最好的情况是此值为1.

2.2 相对分析误差RPD=(stdev/remse)此值只在测试集上测定，stdev表示测定值标准差，rmse表示测试集上均方根误差。RPD值越大越好。

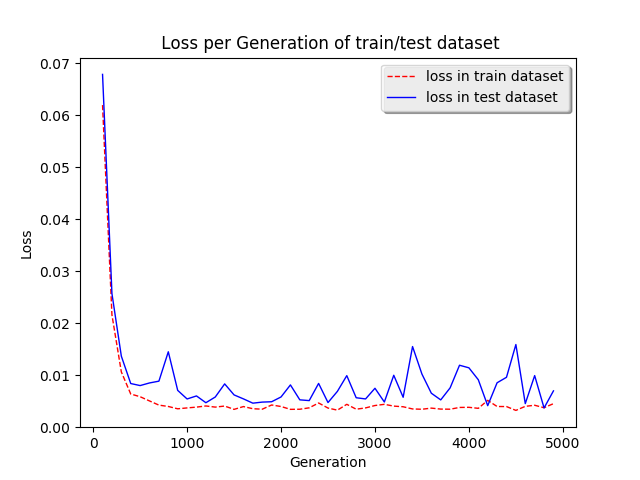
2.3对于多元素的预测问题，使用多个神经元表示输出，要在不同的元素上分别在训练集和测试集上使用评价指标（因为数据的分布和数据的范围是不同的，但是对于多任务学习其loss值是多种任务loss值的加和，我们不能知道导致loss值大的原因是来自于模型对哪一种预测元素的偏差，只有在多种任务上分别使用测试指标，才能判出模型对于不同任务的准确程度。）

3.数据的预处理，数据的不同维度一般来说很可能来自不同的数量级，比如目标屈服强度数据范围为200-600，伸长率数据范围为20-60，而硼元素的数据范围一般在0.0005-0.0025之间，如果使用原始数据进行预测则会出现模型不拟合的后果，此时就需要上使用数据归一化处理（max-min归一化，0归一化等等）将数据各维度缩放到一个数据范围内，通常为[-1,1]和[0,1].

****

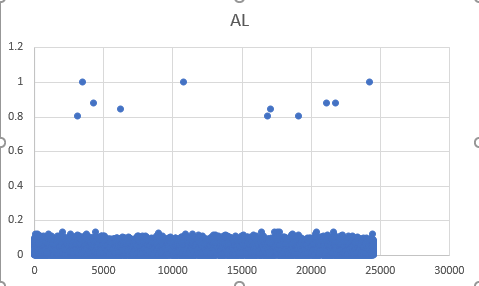
****

以上为不使用归一化的后果。（注意数量级）

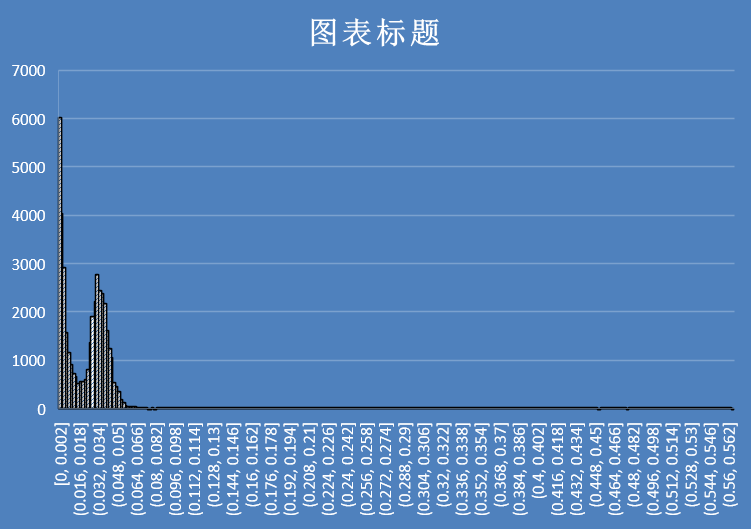


上图是使用数据归一化后的结果

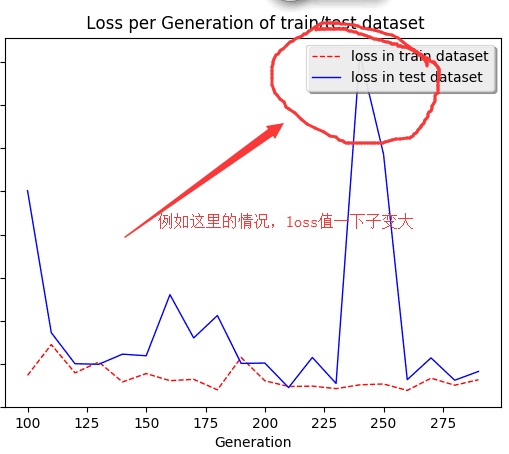
1. 数据一定要保证干净，数据在处理之前要清洗，去除错误的和整体分布明显不一致的数据。（可以使用excel查看csv文件中数据的总体分布）



在归一化的数据后我们发现AL的值一般是分布在0-0.066之间，25000个数据中出现3个值为1的值肯定是数据采集时本身有错误。



如果这种数据出现在测试集或者验证集中而训练集中没有这种数据很可能出现



Loss值突然一下爆炸的情况