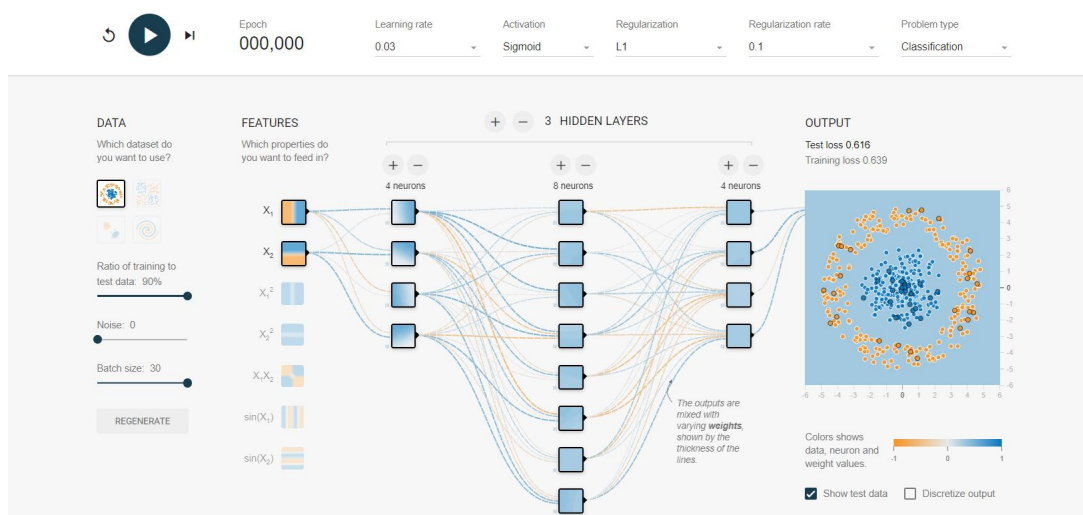
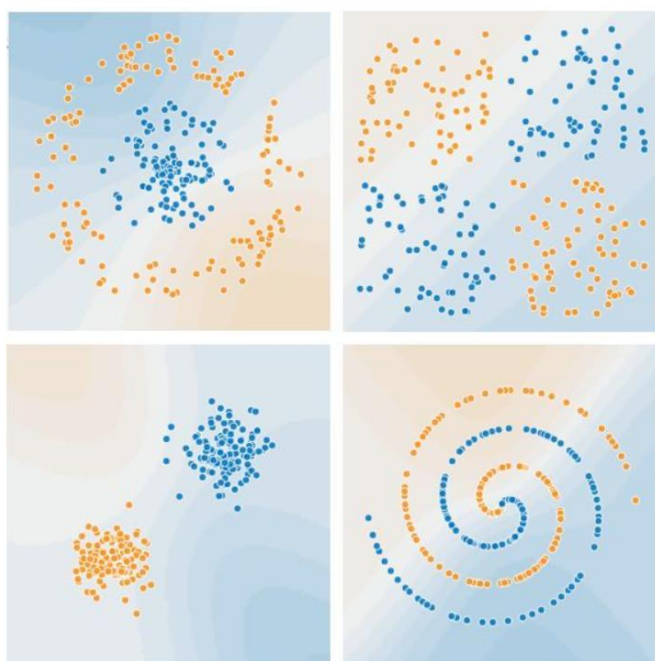


Tensorflow Playground 试用报告

进入网站后的界面是这样的。



我们可以通过这个网站去模拟一个 MLP 实现分类或者回归任务的过程。网站中的输入数据共有四种，分别是圆形、异或、高斯和螺旋。平面内的数据分为蓝色和黄色两类，如下所示。



我们的目标就是使用一个 MLP 去对这四种数据实现分类或者回归。输入的特征一共有 7 类， X_1 、 X_2 、 X_1X_1 、 X_2X_2 、 X_1X_2 、 $\sin(X_1)$ 、 $\sin(X_2)$ 。

通过 Playground, 对于数据, 我们能够调节训练集和测试集的比例, batchsize 大小以及噪声大小。对于训练方面, 我们能够调节学习率, 激活函数, 正则化方式及权重。对于模型, 我们能调节隐藏层的数量 (上限为 6 个) 和每层的神经元个数 (上限为 8 个)。模型最后的效果可以通过 test_loss 体现。

借助于 Playground, 我们可以进行以下实验。

首先是 L1 和 L2 正则化的区别, 在保持其他设置相同的情况下, 我们将正则化项分别设置为 L1 和 L2 的结果如下所示。

使用 L1 正则化时收敛 Loss 为 0.002, 第 1 个隐藏层的 1 个神经元跟输入层地 2 个神经元之间的权重如下所示。



使用 L2 正则化时收敛 Loss 为 0.001, 第 1 个隐藏层的 1 个神经元跟输入层地 2 个神经元之间的权重如下所示。



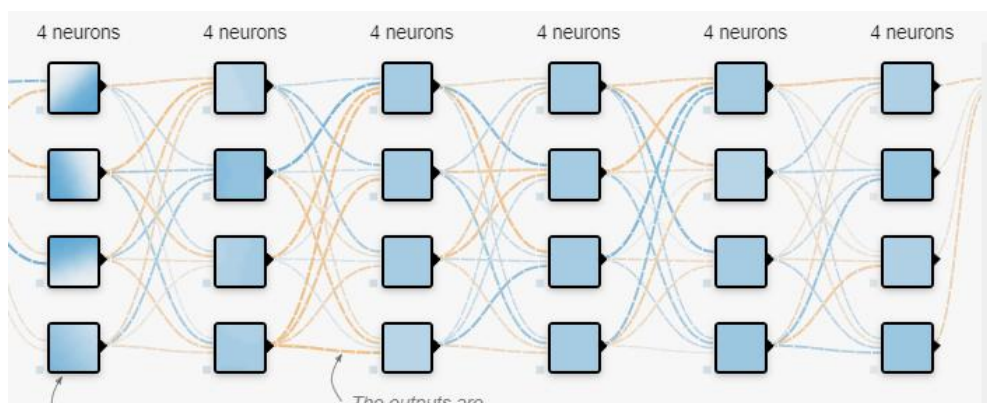
对比结果可以发现, 相较于 L1 正则化, L2 正则化得到的权重的值更加接近, 这说明 L1 正则化更偏向于将某些权重置于 0, 而 L2 正则化则更偏向于把所有的权重同等变小。

然后我们可以对比下 Sigmoid 激活函数和 ReLU 激活函数收敛速度。

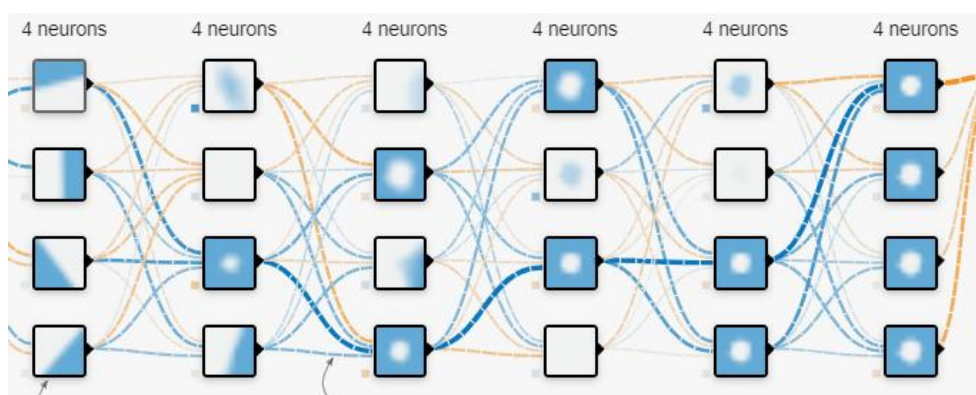
Sigmoid 激活函数的收敛次数为 2000, 收敛 Loss 为 0.005, ReLU 激活函数的收敛次数为 600, 收敛的 Loss 为 0.001。这说明 ReLU 比 Sigmoid 函数的收敛速度更快。

最后, 我们可以研究下在较深模型中, Sigmoid 和 ReLU 的选取问题。

使用 Sigmoid 函数的收敛 Loss 为 0.5, 模型权重图像如下所示。可以看到模型的权重都比较小, 而且模型的学习效果很差, 这可能是因为 Sigmoid 函数在深度模型中存在梯度消失的问题。



ReLU 函数的收敛 Loss=0.000，模型权重如下所示，可以清楚地看到，ReLU 激活函数能够很好地解决 Sigmoid 出现的梯度消失问题。



通过 Tensorflow Playground，我们可以清楚地可视化模型的训练过程，在实验结果不好时，也可以通过查看中间连接的权重值来发现问题的所在。当然它也存在一定的缺点，比如只能可视化 MLP 模型，提供的输入数据过于基础等等。但是总体来说，它能够提高研究人员对于模型的认识，是一个很好的学习工具。