# TensorFlow Playground 试用报告

122106222842 王浩骅

目录

[TensorFlow Playground 试用报告 1](#_Toc131845573)

[TensorFlow Playground 简介 2](#_Toc131845574)

[数据集 DataSet 2](#_Toc131845575)

[特征 Feature 3](#_Toc131845576)

[隐藏层 Hidden Layers 3](#_Toc131845577)

[迭代 Epoch 4](#_Toc131845578)

[学习率 Learning Rate 5](#_Toc131845579)

[激活函数 Activation 6](#_Toc131845580)

[正则化 Regularization 和 正则化率 Regularization Rate 6](#_Toc131845581)

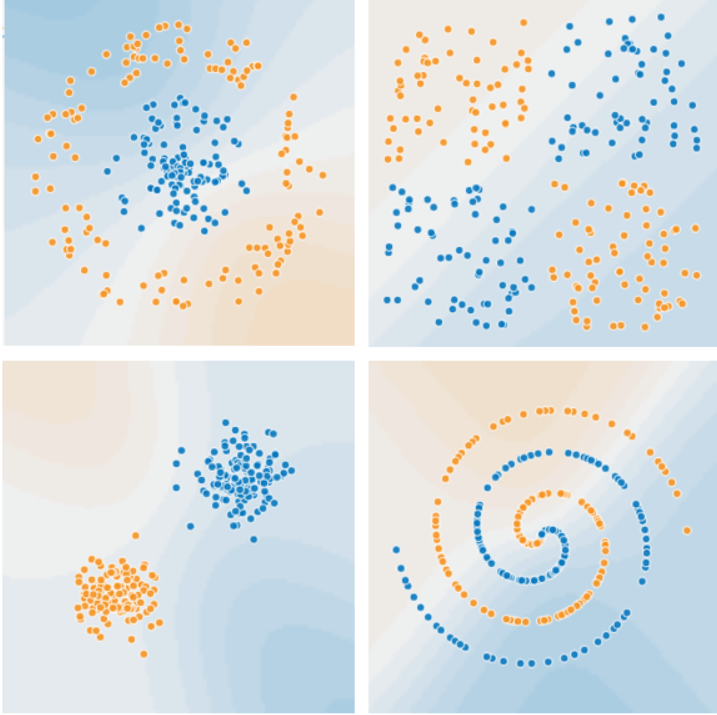
[试用展示 7](#_Toc131845582)

### TensorFlow Playground 简介

[TenforFlow Playground](https://link.juejin.cn?target=https%3A%2F%2Fgithub.com%2Ftensorflow%2Fplayground) 又名 TensorFlow 游乐场，是一个用来图形化教学的简单神经网络在线演示和实验的平台。其可将神经网络的训练过程直接可视化，从中可快速便捷的感受参数的变化对训练的影响。各部分介绍如下：

### 数据集 DataSet

在分类任务的实验数据中，提供了4种不同形态的数据，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，如下图所示：



我们的目标就是通过神经网络将这两种数据分类，可以看出螺旋形态的数据分类是难度最高的。除此之外，PlayGround还提供了非常灵活的数据配置，可以调节噪声、训练数据和测试数据的比例和Batch size的大小。

### 特征 Feature

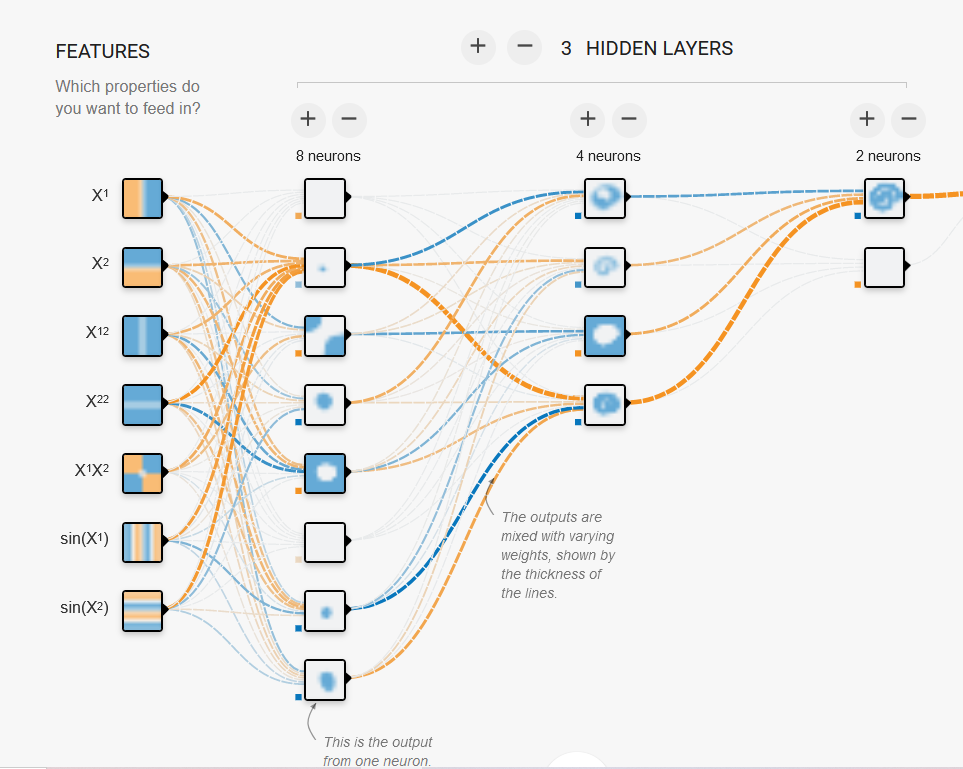
Feature一栏包含了可供选择的 7 种特征：，，，，，，。在这里，特征的小方块中同样有 黄蓝两种颜色，这个颜色是数值大小的意思，蓝色代表大，黄色代表小，基本可以认为蓝色是正数，黄色是负数。方块本身代表着整个的取值空间，可以认为中心点就是坐标原点，那么以 x1x2 特征来看，一三象限内值都是蓝色，也就是正数，二四象限内都是负数，很形象的可视化表述了我们的数据特征。

### 隐藏层 Hidden Layers

隐藏层一栏可设置多少隐含层。一般来讲，隐含层越多，衍生出的特征类型也就越丰富，对于分类的效果也会越好，但不是越多越好，层数多了训练的速度会变慢，同时收敛的效果不一定会更好，容易过拟合。我们可以通过解释 playground 里面的颜色来感知隐藏层权重的意义，他们的颜色和数据点蓝黄颜色的意义还稍稍不一样：

* 每个神经元只有蓝白两色，蓝色部分是此神经元的敏感区域。
* 每个神经元输出的线有蓝黄两色，蓝色是正反馈，黄色是负反馈。越粗则说明下游节点眼里这个特征越重要。

隐藏层中神经元和权重可视化结果如下图所示：

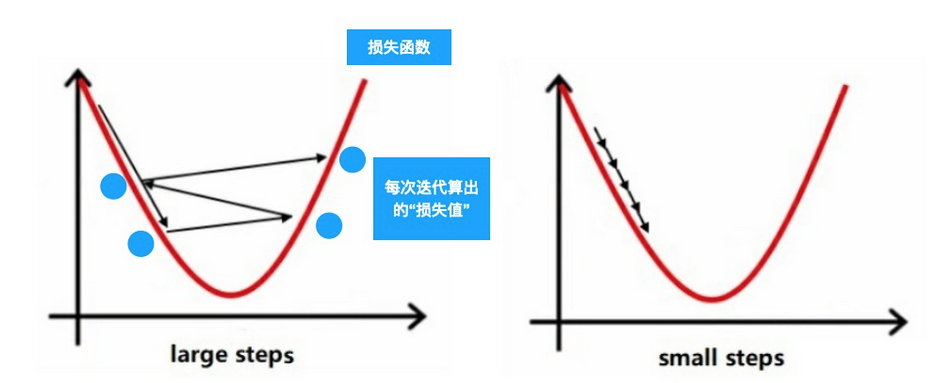


### 迭代 Epoch

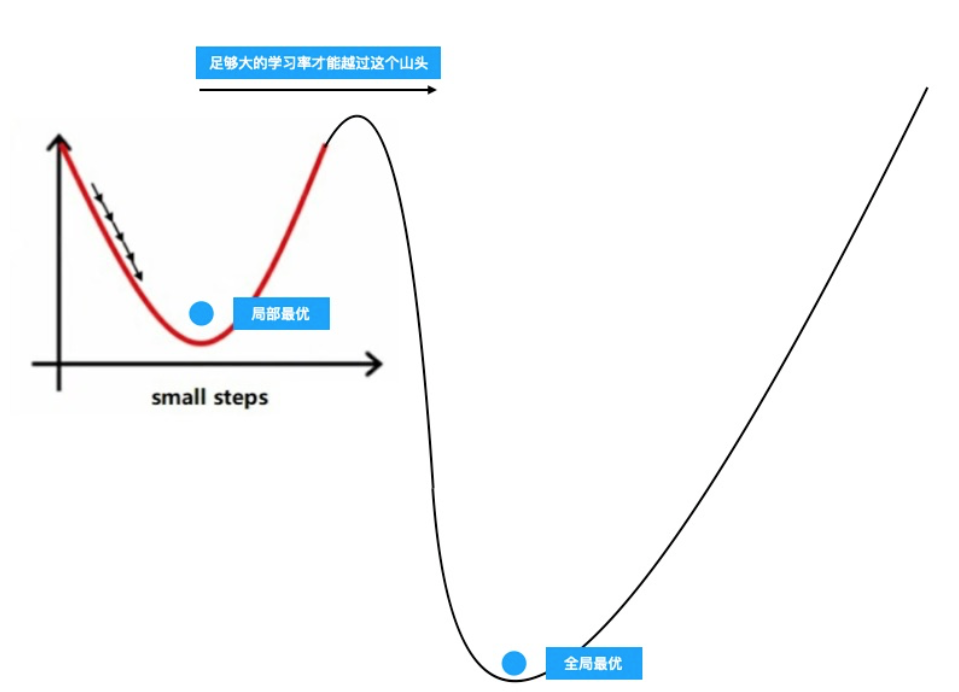
迭代理念是机器学习不同于传统统计学分析的最大的点之一。全量的数据在网络里跑完一次后计算出当前的损失函数大小，然后往下个方向继续行进。这个过程就是一个迭代（Epoch），我们在 Tensorflow Playground 里面运行算法时，随着 Epoch 的增加，Output 里 loss（损失函数值）也在随之变化，直到 N个迭代后到达损失函数最优点。这时候我们可以说，已经找到了当前数据样本的最佳拟合函数，我们的模型已经训练完成。

### 学习率 Learning Rate

学习率就是每个迭代完之后的行进步长，步长过大则有可能越过了最优解，反复横跳不能得到最优效果，步长过小则不仅收敛效果慢。如下图所示：



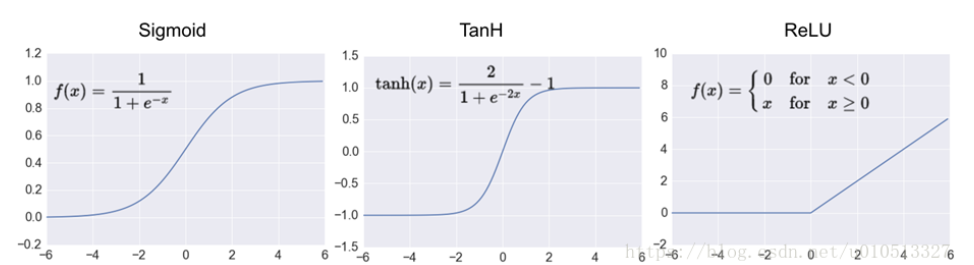
步长过小还有可能如下图一样陷入局部最优，这时候加大学习率才能越过山头找到真正的最优解，如下图所示：



总之，选用合适的学习率很重要，不同的模型也有不同的合适学习率可选择。甚至你可以用一些动态学习率的方法来在运算的过程中动态调整学习率，loss 下降明显时加大学习率，loss 趋缓时减小学习率。

### 激活函数 Activation

激活函数是神经网络独有的概念，可以这么说，没有激活函数，那么神经网络是画不出那一条细致的分类线的，再多的隐藏层和神经元都只能复合出线性运算，只能画直线拟合线性函数。激活函数在 Playground 里面供我们选择的有 Relu、Tanh、Sigmoid 等，如下图所示：



### 正则化 Regularization 和 正则化率 Regularization Rate

正则化是解决过拟合的手段之一，使用正则化可以降低模型的复杂度，增强模型的泛化能力。正则化常用的几种方法有 L1 和 L2。

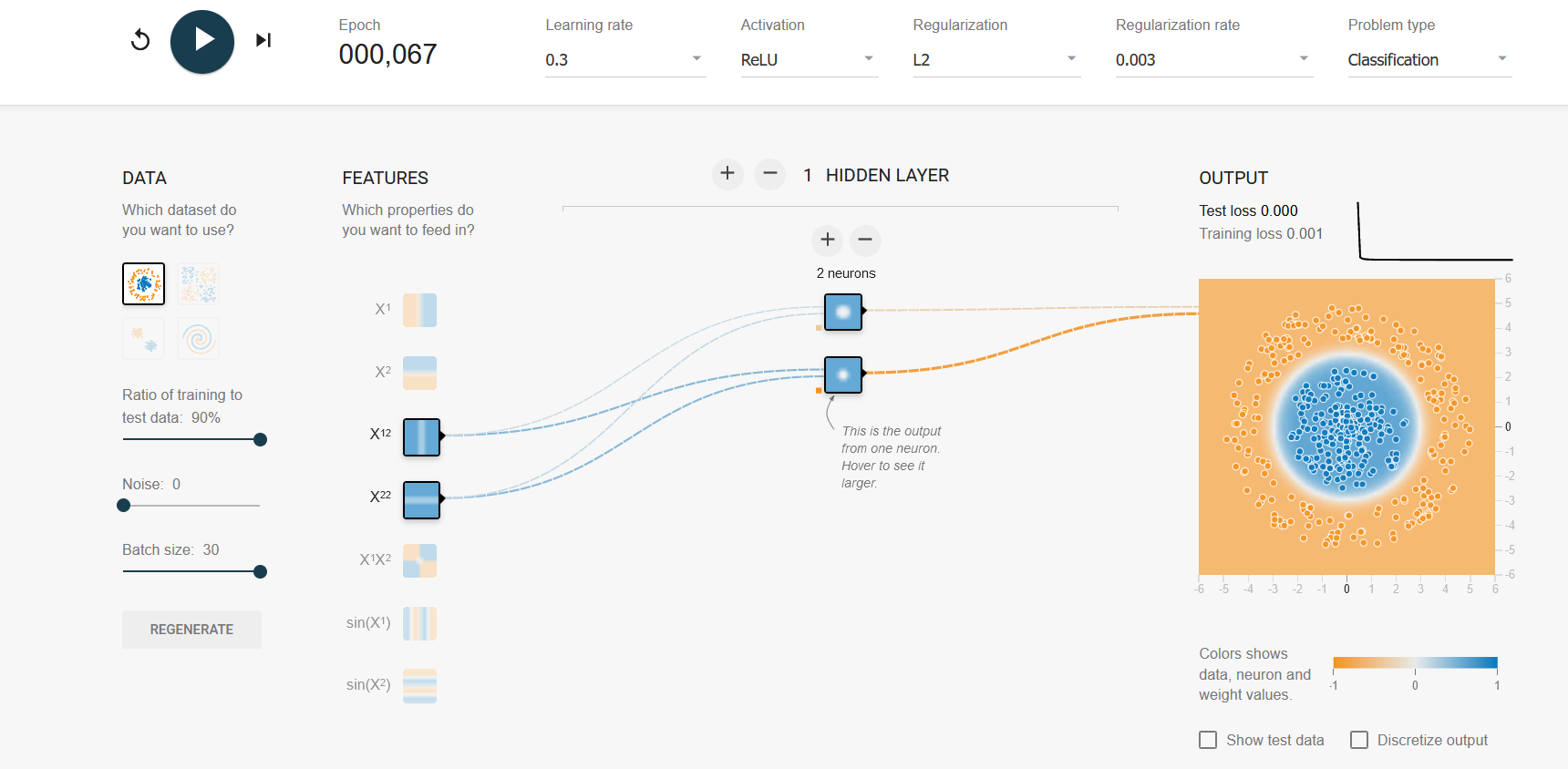
* L1 范数：权值向量 w 中各个元素的绝对值之和。
* L2 范数：权值向量 w 中各个元素的平方和求平方根。

正则化率就是正则化的程度，提高正则化率可以减少过拟合，但是也要注意不要过高导致欠拟合。

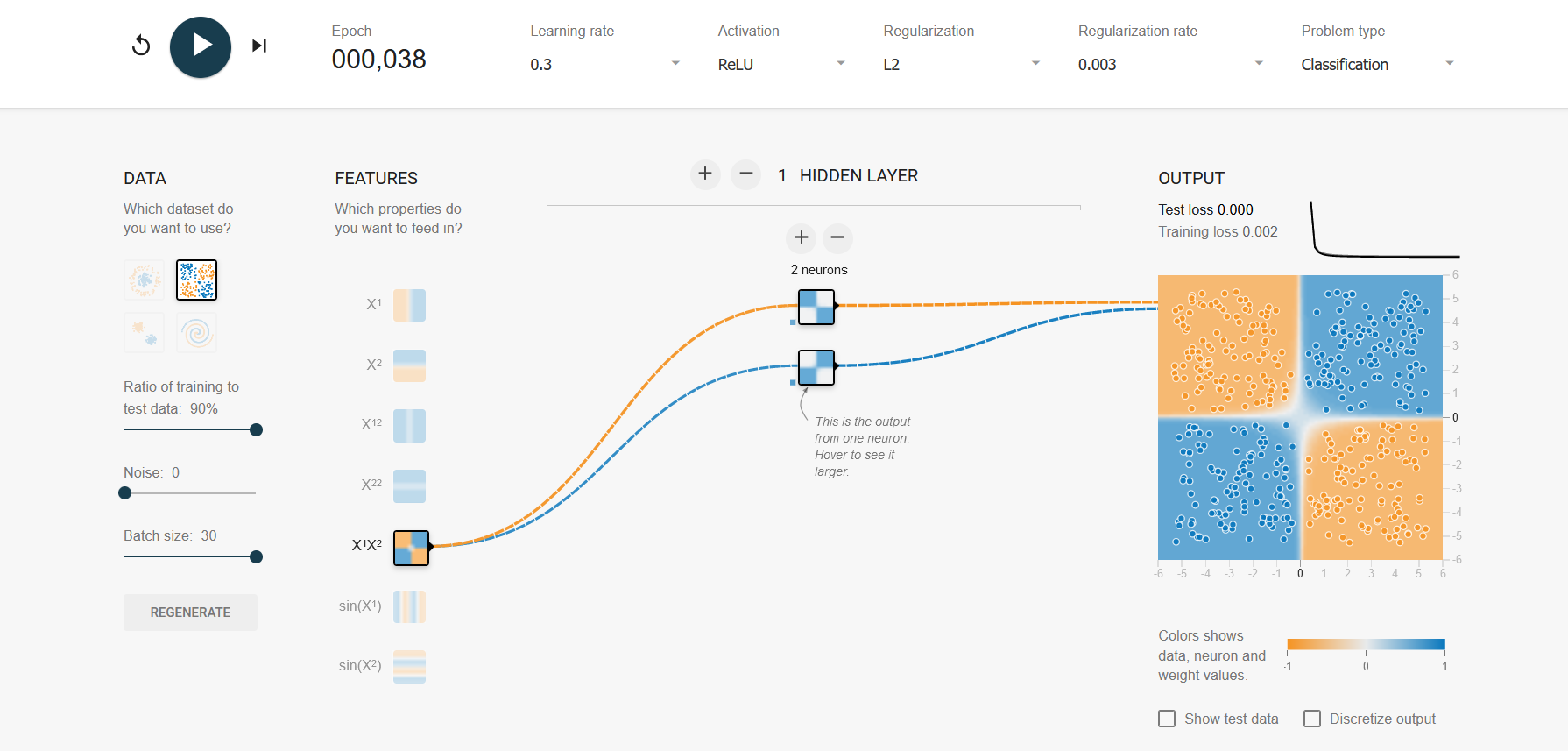
### 试用展示

各类型数据集调参后分类结果展示如下各图所示。

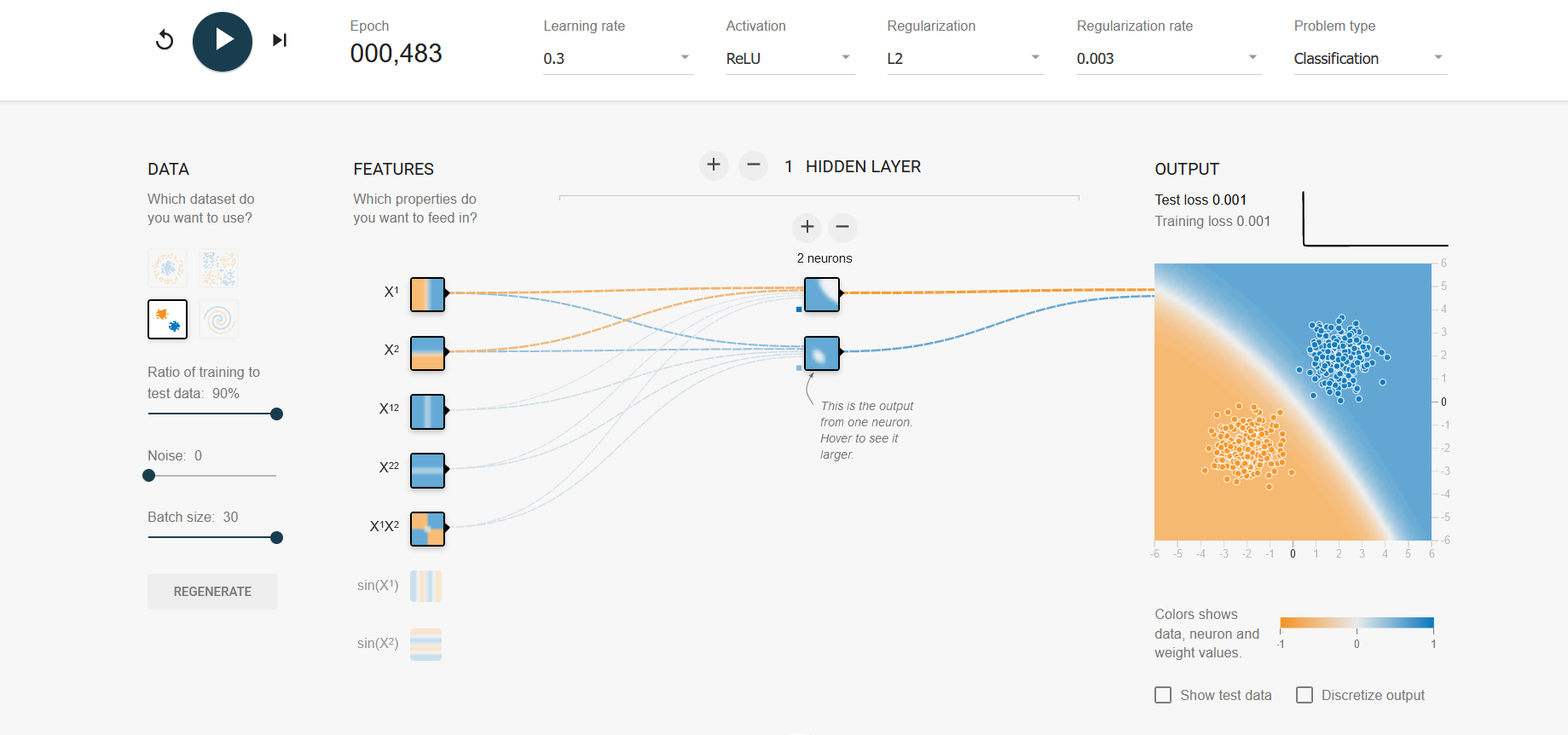
圆形：



异或：



高斯：



螺旋型：

