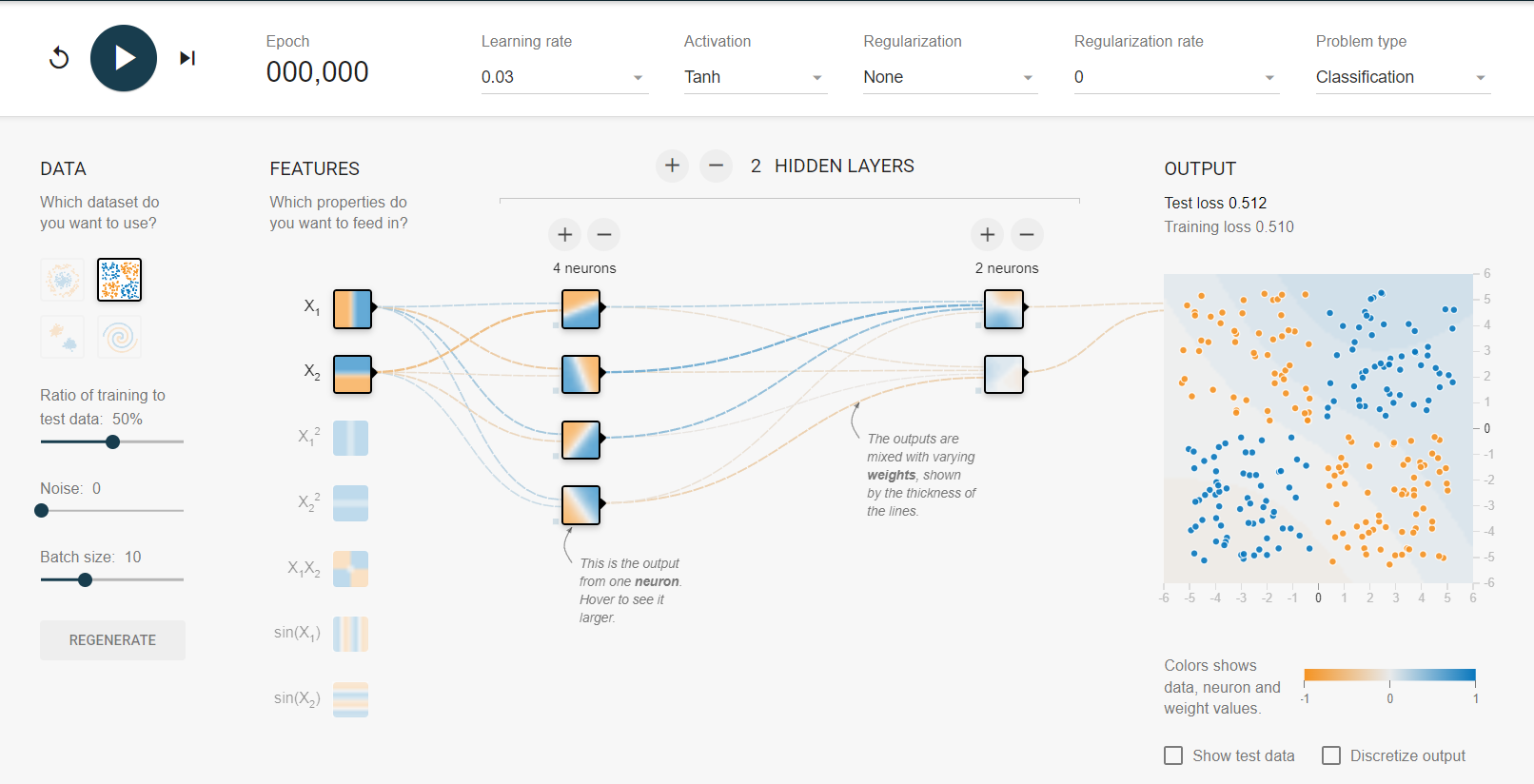
# **TensorFlow Playground试用报告**

# 概述

[TensorFlow](https://cloud.baidu.com/product/wenxinworkshop" \t "https://developer.baidu.com/article/_blank) Playground是一个简单易懂、功能强大的TensorFlow入门神器。它通过一个可视化的界面，让用户能够轻松地构建和训练[神经网络](https://cloud.baidu.com/product/wenxinworkshop" \t "https://developer.baidu.com/article/_blank)，而无需编写大量的代码。对于初学者来说，这是一个非常好的学习工具，因为它能够以直观的方式展示TensorFlow的基本概念，包括张量、操作、损失函数、优化器等。

在TensorFlow Playground中，用户可以通过拖拽不同的模块来构建神经网络。这些模块包括输入层、隐藏层和输出层等，用户可以根据自己的需求自由组合这些模块。同时，用户还可以通过调整模块的参数来优化神经网络的性能。这种可视化的构建方式使得神经网络的构建过程变得非常简单明了，用户可以轻松地理解神经网络的结构和工作原理。  
 除了神经网络的构建，TensorFlow Playground还提供了数据集选择和训练设置等功能。用户可以选择不同的数据集进行训练，例如MNIST手写数字数据集、CIFAR-10图像数据集等。同时，用户还可以通过调整训练设置来控制训练过程，例如学习率、训练轮数等。这些功能使得用户可以更加灵活地控制训练过程，从而更好地理解神经网络的训练原理。

# 功能介绍



（1）控制运行

从左到右三个功能分别是：（a）重启；（b）运行；（c）一次运行一个周期

（2）运行周期数

用于查看训练的周期数

（3）参数调整区域

名称 功能说明

Learning rate 学习率（是一个超参数，在梯度下降算法中会用到；学习率是人为根据实际情况来设定）。

Activation 激活函数（默认为非线性函数Tanh；如果对于线性分类问题，这里可以不使用激活函数）。

Regularization 正则化（正则化是利用范数解决过拟合的问题）。 Problem type 问题类型（在这里我们要解决的是一个二分类问题，简单解释一下分类问题是指，给定一个新的模式，根据训练集推断它所对应的类别（如：+1，-1），是一种定性输出，也叫离散变量预测；回归问题是指，给定一个新的模式，根据训练集推断它所对应的输出值（实数）是多少，是一种定量输出，也叫连续变量预测；在这里我们属于分类问题。）。

（4）数据区域

DATA 数据集类型（这里提供了四种数据集，我们默认选中第一种；被选中的数据也会显示在最右侧的OUTPUT中；在这个数据中，我们可以看到二维平面上有蓝色和黄色的小点；每一个小点代表一个样例例子；点的颜色代表样例的标签；因为只有两种颜色，所以这里是一个二分类问题；在这里我们以判断某工厂零件是否合格为例子来说明，那么黄色就代表不合格零件，蓝色就代表合格零件）。

Ratio of training to test 数据用于测试的比例（直接对进度条进行操作即可调整）。 Noise对数据中引入噪声。

Batch size 调整batch size的大小。

（5）网络结构调整区域

FEATURES 特征向量（为了将一个实际问题对应到空间中的点，我们需要提取特征。在这里我们可以用零件的长度和质量来大致描述；所以这里x1就代表零件长度，x2代表零件质量;特征向量是神经网络的输入）。

HIDDEN LAYERS 隐藏层（在输入和输出之间的神经网络称为隐藏层；一般神经网络的隐藏层越多这个神经网络越深；这里我们默认有一个隐藏层，这个隐藏层上有4个节点）。

直接通过点击各个图标即可选择Features的类型，对于隐藏层的操作，可以直接选择加减号即可获得想要的隐藏层层数以及每层的神经元个数。

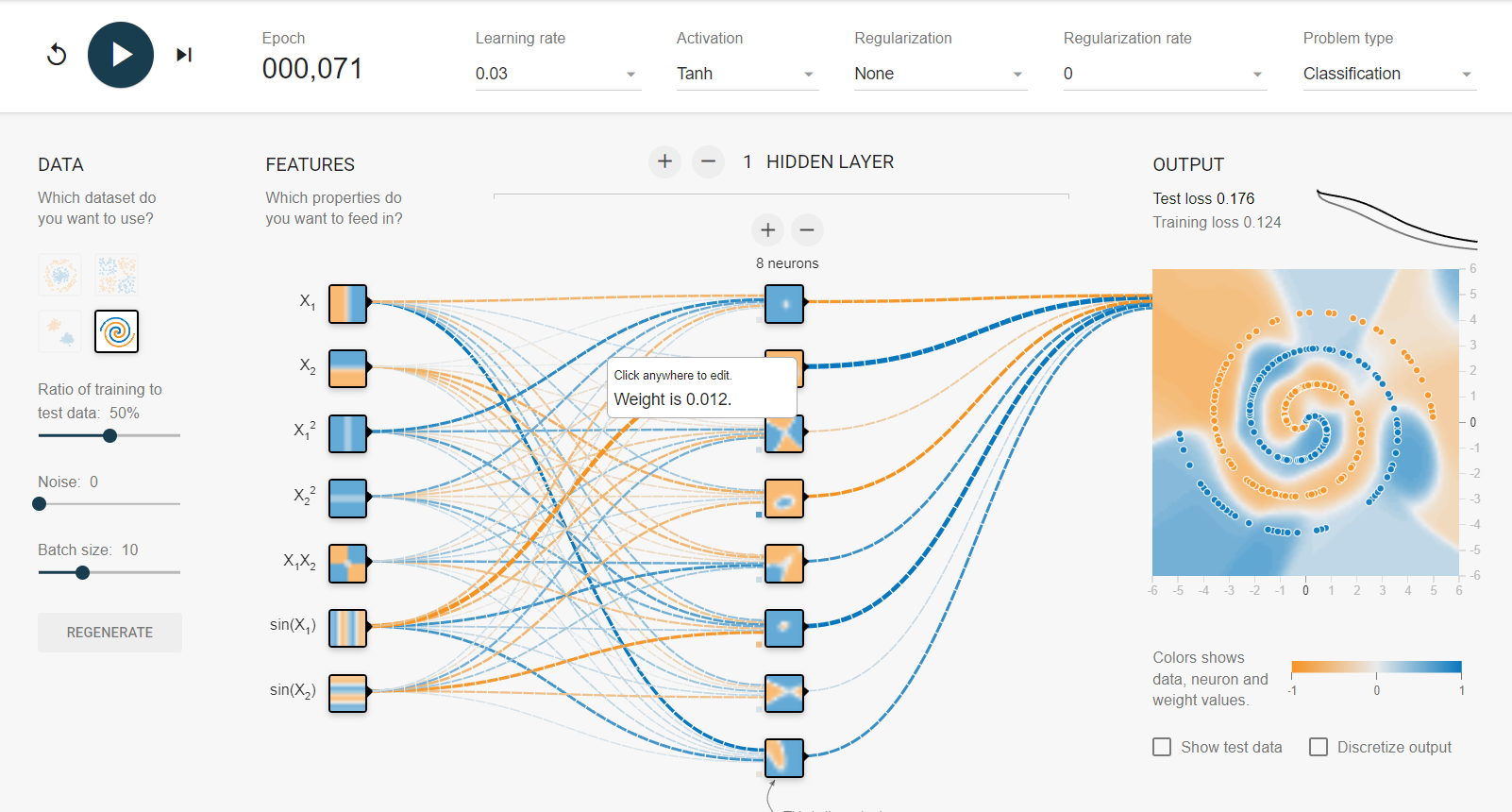
（6）输出结果区域

设置完上面的参数，点击运行即可观测到输出结果的变化。

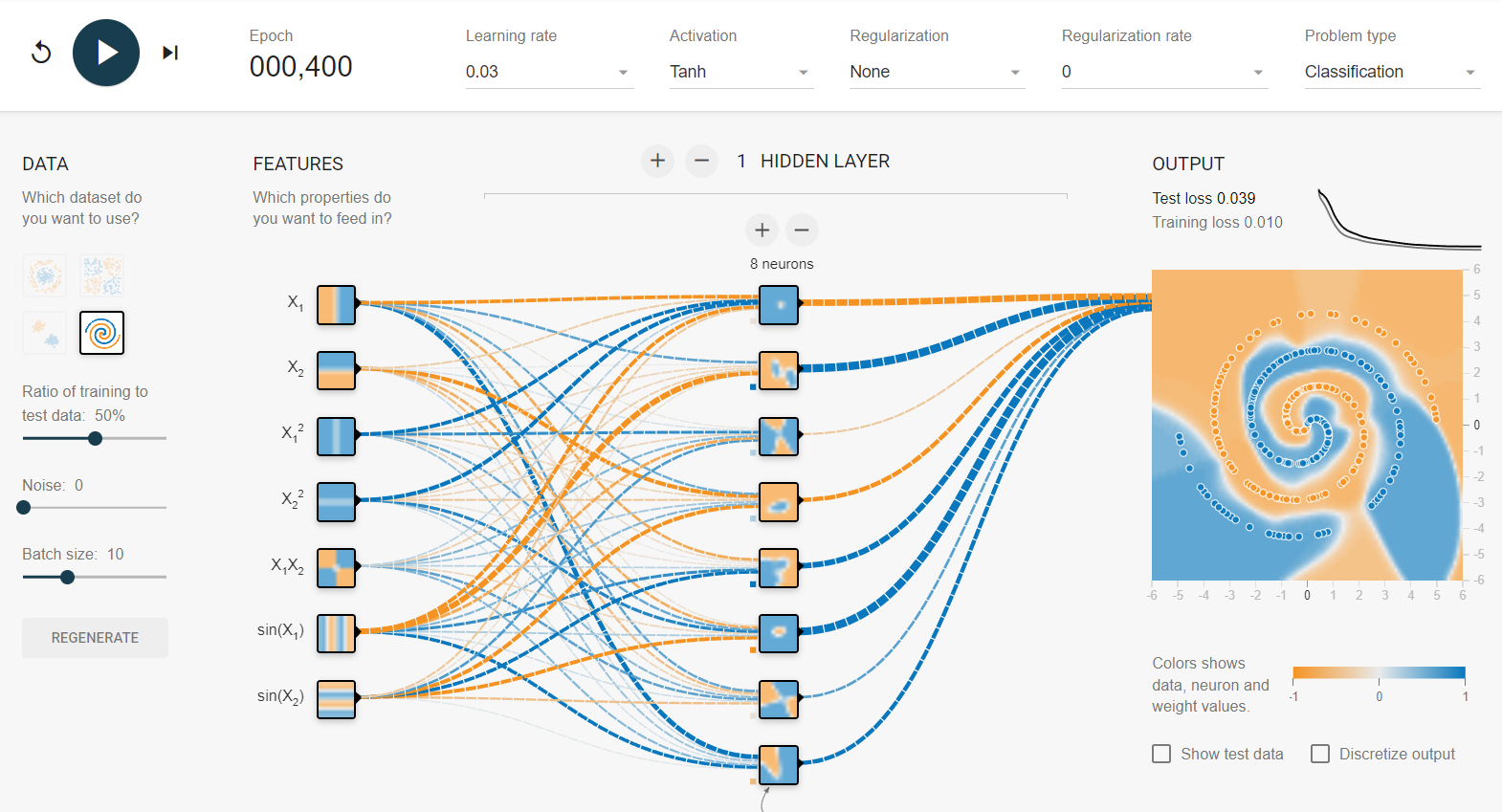
如果你选择的是分类问题，即可看到明显的边界变化以及loss在不断减小的情况，点击show test data可以显示未参与训练的test数据集的情况，点击Discretize output可以看到离散化后的结果。

# 工具演示

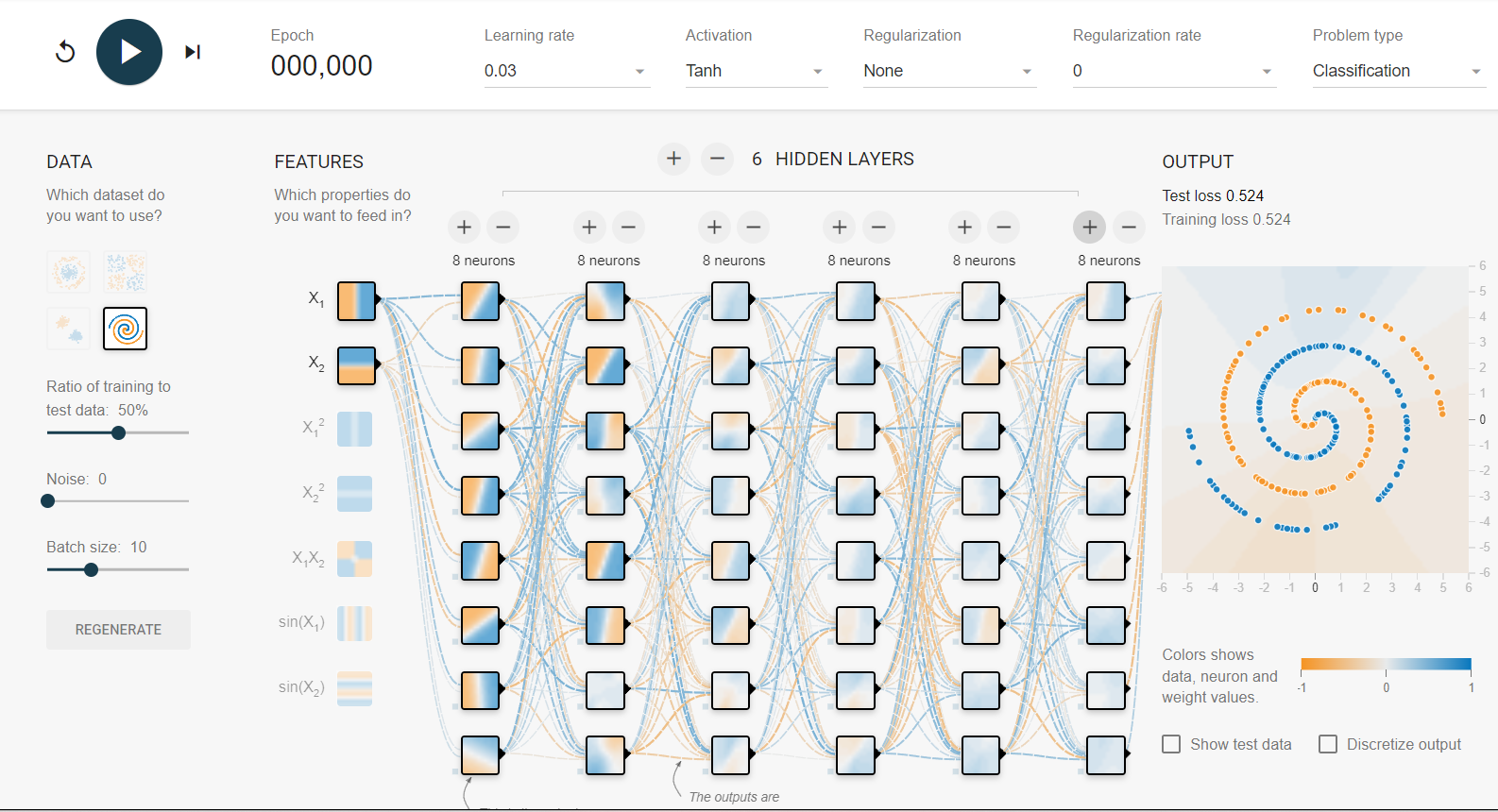
首先进行实验传统的方法。在这里，我们将所有能够想到的7个特征都输入系统，并选择只有1层的神经网络：



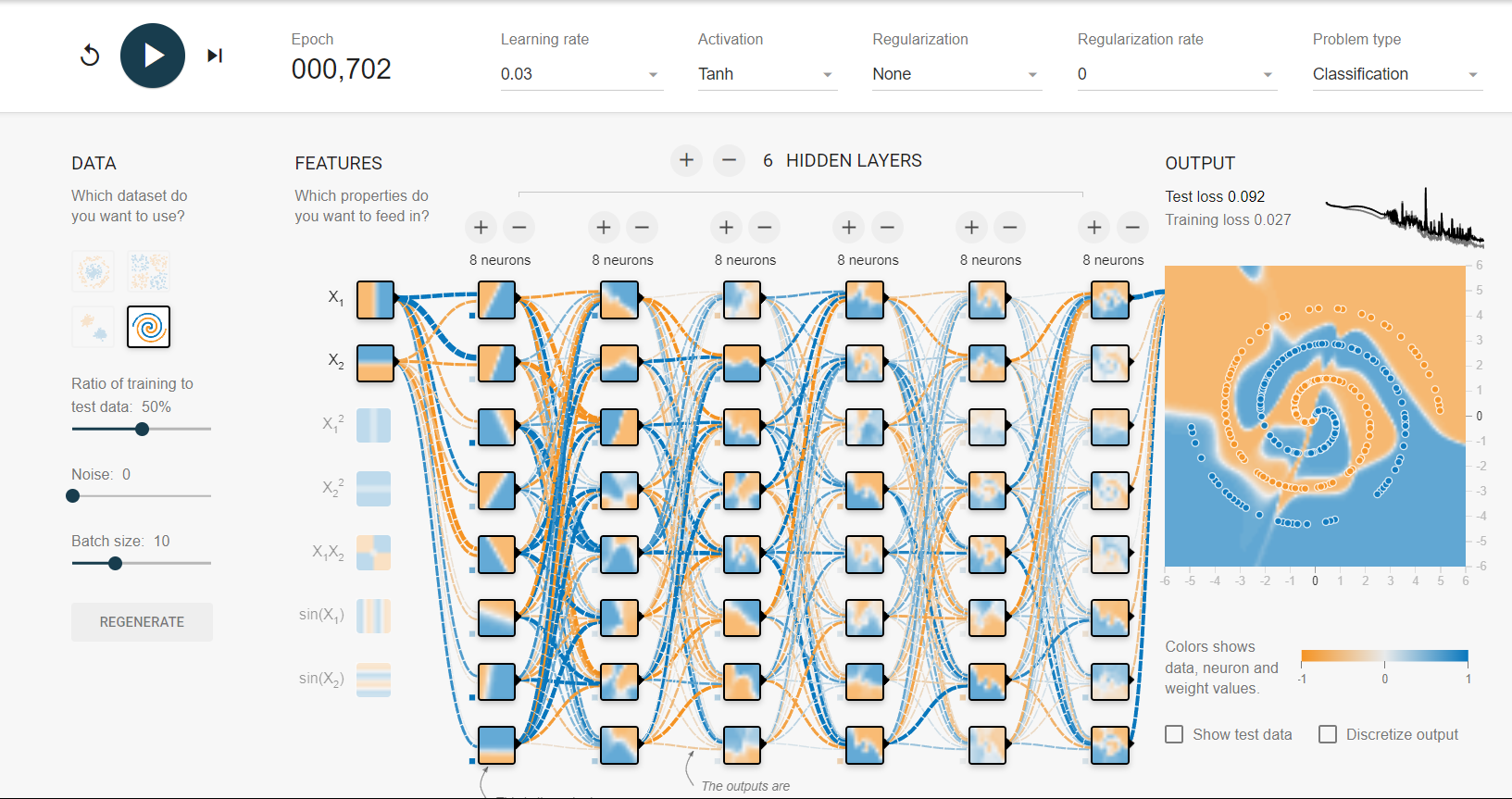
最后的结果是这样的，可以看出单层神经系统几乎完美地分离出了橙色点和蓝色点：



第二次实验中，只输入x1, x2，而选择1个6层的，每层有8个神经元的神经网络：



最后的结果发现，通过增加神经元和神经网络的层数，即使没有输入许多特征，我们也能够成功地分类：



# 结论

通过这次试验，我了解了TensorFlow Playground在对加深对神经网络的理解方面有很大优点，可以自己自由搭配各种参数，看看怎么才能最少次数迭代出收敛的模型。最重要还是可视化啊，很容易理解。而当我们有了神经网络，我们的系统自己就能学习到哪些特征是有效的、哪些是无效的，这就大大提高了我们解决复杂机器学习问题的能力。