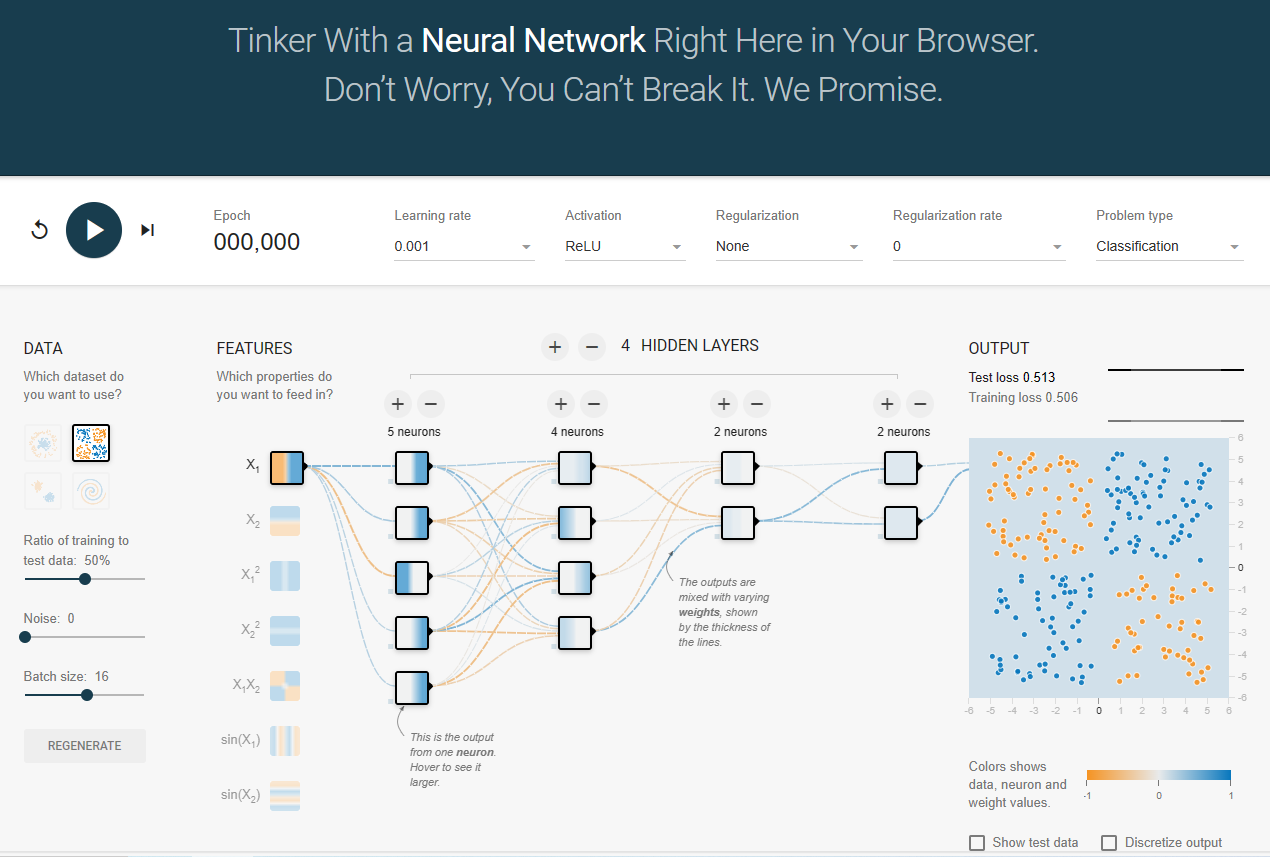
**TensorFlow-Playground 试用报告**

理解神经网络：通过调整网络结构和参数，观察训练过程和结果，用户可以更深入地理解神经网络的工作原理。

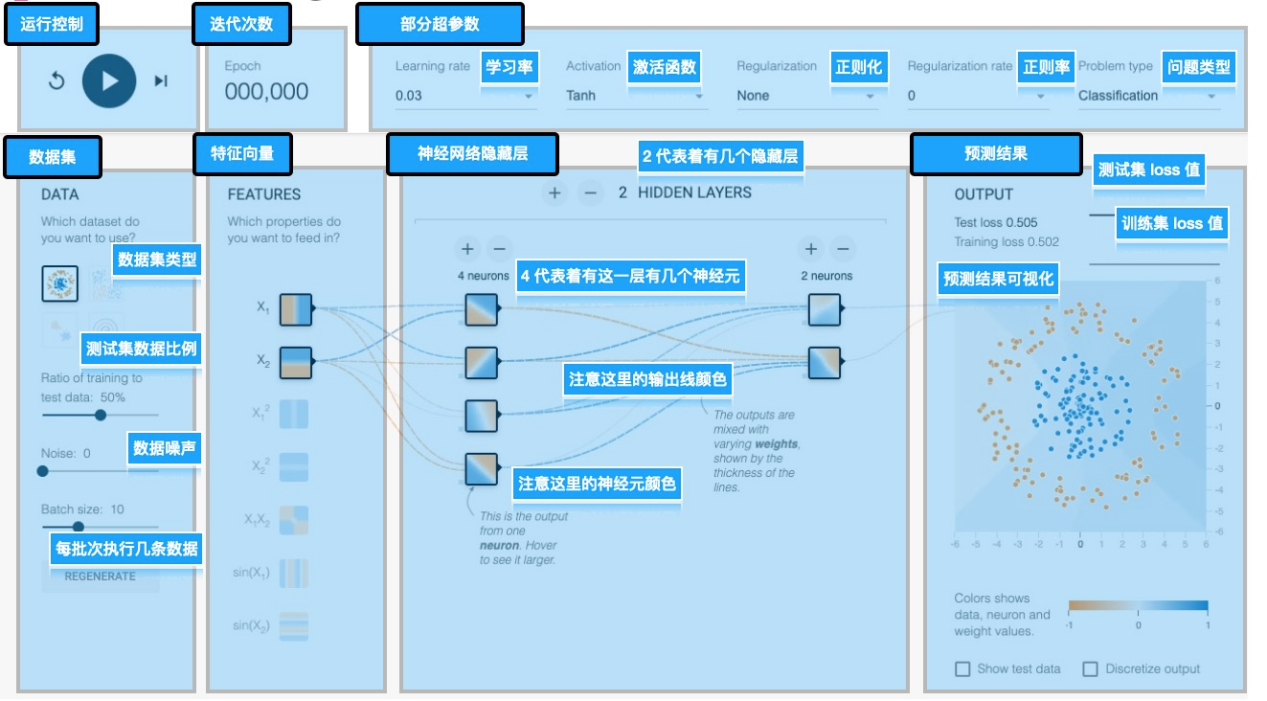
实验和探索：TensorFlow Playground提供了一个安全的环境，用户可以自由地进行实验和探索，而无需担心影响实际项目。

直观的反馈：实时的可视化反馈可以帮助用户理解模型的行为，发现问题，并调整策略。

总的来说，TensorFlow Playground是一个强大而直观的工具，对于初学者来说，它可以帮助理解神经网络的基本概念；对于有经验的用户来说，它可以作为一个实验平台，用于测试新的想法和方法。



1. 运行控制区  
   运行控制区位于页面的右上角，包含三个按钮：Play、Pause和Reset。点击Play按钮，模型开始训练，按钮会变为Pause，此时可以点击Pause按钮暂停训练。点击Reset按钮，模型的训练状态和结果将被重置，可以重新开始训练。  
   二、迭代次数展示区  
   迭代次数展示区位于运行控制区下方，显示了当前的迭代次数。例如，如果显示为"Epoch 1000"，表示模型已经进行了1000次迭代。  
   三、超参数选择区  
   超参数选择区位于页面的右侧，包含多个选项，如学习率、激活函数、正则化方法和比率等。例如，你可以选择学习率为0.03，激活函数为ReLU，正则化方法为L2，正则化比率为0.001。这些设置将影响模型的训练过程和结果。  
   四、数据集调整区  
   数据集调整区位于页面的左上角，你可以在这里选择不同的预设数据集，如圆形、异或、高斯等。例如，如果你选择了圆形数据集，那么模型的任务就是要学习如何区分位于圆形区域内外的数据点。  
   五、特征向量选择区  
   特征向量选择区位于数据集调整区下方，你可以在这里选择输入特征。例如，如果你选择了X1和X2^2，那么模型将使用这两个特征来进行训练。  
   六、神经网络区域  
   神经网络区域位于页面的中间，你可以在这里添加隐藏层和神经元。例如，你可以添加两个隐藏层，每个隐藏层包含三个神经元。  
   七、预测结果区  
   预测结果区位于页面的左侧，显示了模型的预测结果。在训练过程中，你可以看到决策边界的变化，以及每个数据点的预测类别。



Epoch ---> 训练次数。

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，这里咱不讨论）。

Regularization rate ---> 正则率（这里不使用正则化所以无需考虑）。

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。）。

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）。

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。在这里我们可以用图片的颜色和纹理来大致描述；所以这里x1就代表图片颜色，x2代表图片纹理;特征向量是神经网络的输入）。

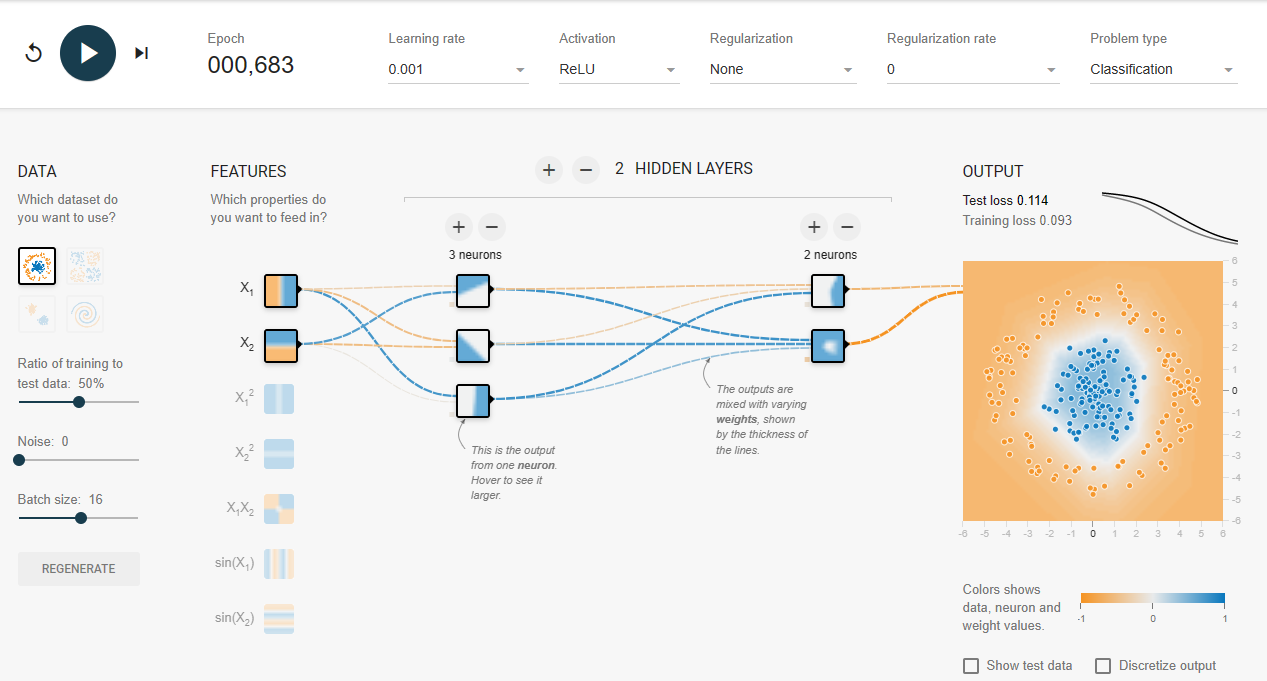
HIDDEN LAYERS ---> 隐藏层（在输入和输出之间的神经网络称为隐藏层；一般神经网络的隐藏层越多这个神经网络越深；这里我们默认有一个隐藏层，这个隐藏层上有4个节点）。

OUTPUT ---> 输出（显示区分平面以及训练数据）。

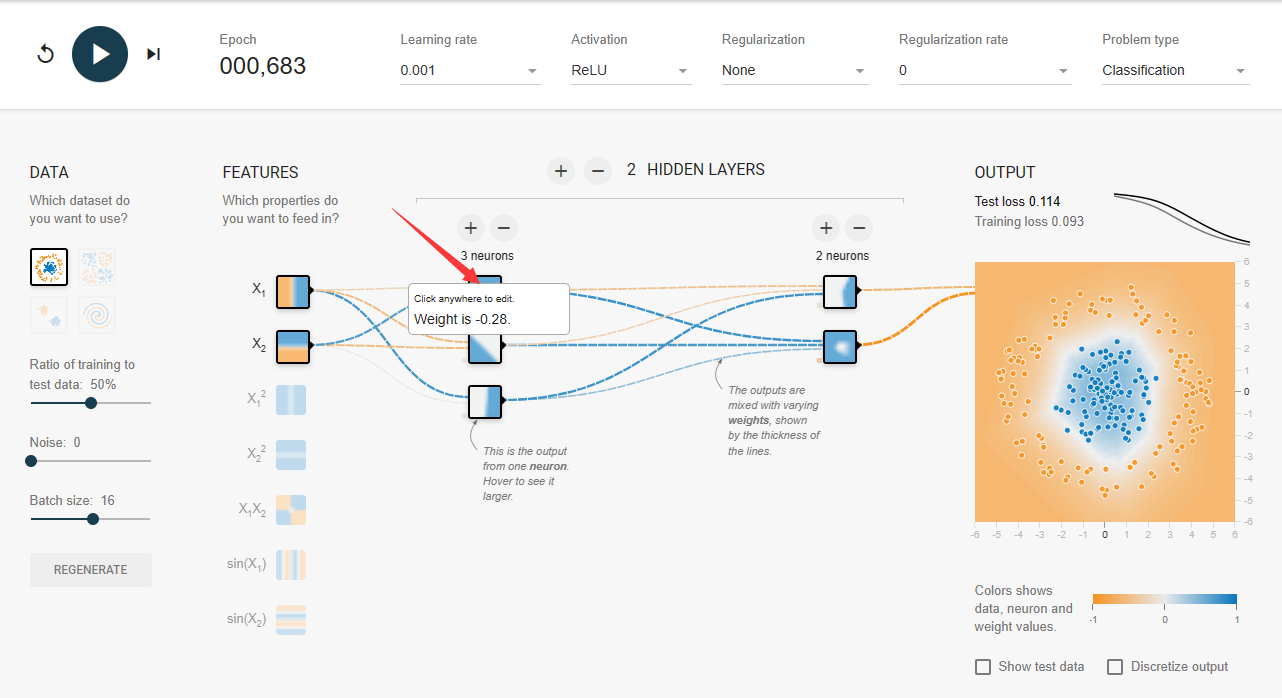
**神经网络的训练过程**

图上每一个小格子代表神经网络中的一个神经元，每一条线代表神经元之间的连接。每一个节点和边都被涂上了或深或浅的颜色。但边的颜色和格子中的颜色含义有略微区别。每一条边代表了神经网络的一个权重（可以理解为一个参数），它可以理解为任意实数。神经网络就是利用反向传播算法对权重进行合理修改从而解决分类或者回归问题。边的颜色表现了权重的取值，当边颜色越深这个参数绝对值越大；当边越接近白色，这个参数绝对值趋于0。

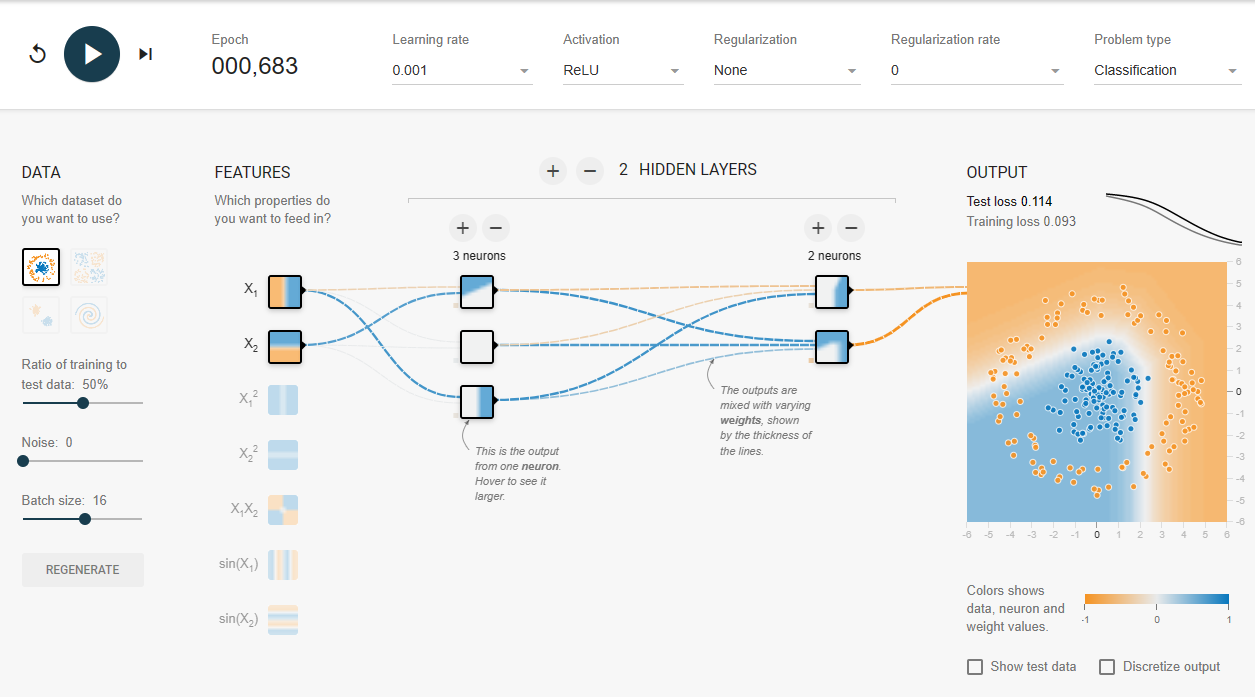
每一个节点上的颜色代表了这个节点的区分平面。在这里稍微解释一下区分平面的概念。如果我们把首先我们把这个平面当成一个笛卡尔坐标系，这个平面上的每一个点就代表了（x1,x2）的一种取值。在图中可以很明显的看到，白色已经很明显的将这两类区分出来。而这个时候的损失函数已经很小了，说明他的训练效果是非常不错的。



在这幅图中，我们的x1和x2是输入层，输入层的输出值就是x1，x2本身，当x1，x2的点在白色光圈内，就说明预测的结果是蓝色，当在白色光圈外，就说明预测结果是橘色。这里的隐藏层只有一个，这个隐藏层上有三个节点，我们把第一个隐藏层节点称为h1，第二个称为h2，以此类推。可以看到每一层都赋有不同的权重，不同的权重赋值也会导致结果的不同，在进行训练的时候需要格外的关注这个结果。



修改权重之后就会发现训练结果发生了巨大的改变：



综上所述，我们可以把神经网络解决一个分类问题大致可以分为：

1.提取特征向量作为输入，比如说本例子中零件的长度和质量。

2.定义神经网络结构。包括隐藏层数，激活函数等等。

3.通过训练利用反响传播算法不断优化权重的值，使之达到最合理水平。

4.使用训练好的神经网络来预测未知数据，这里训练好的网络就是指权重达到最优的情况。

TensorFlow Playground是一个基于Web的交互式神经网络可视化工具，旨在帮助用户理解神经网络的工作原理和调参过程。它提供了一个直观的界面，让用户可以通过实验来探索神经网络的不同配置和超参数对模型的影响。

在TensorFlow Playground中，用户可以选择不同的数据集，包括线性可分数据集和非线性可分数据集，以便观察神经网络在不同情况下的表现。用户可以调整神经网络的结构，包括层数、每层神经元的数量，还可以选择不同的激活函数和优化器。此外，用户还可以调整学习率、批量大小等训练超参数。

在实验过程中，TensorFlow Playground会实时显示神经网络的训练过程，包括损失函数的变化、训练样本和决策边界的可视化。这使用户能够直观地了解神经网络是如何学习和适应数据的，从而更深入地理解神经网络的工作原理。

总的来说，TensorFlow Playground是一个强大的教育工具，为用户提供了一个直观、交互式的学习平台，帮助我们更好地理解神经网络的工作原理和调参过程。