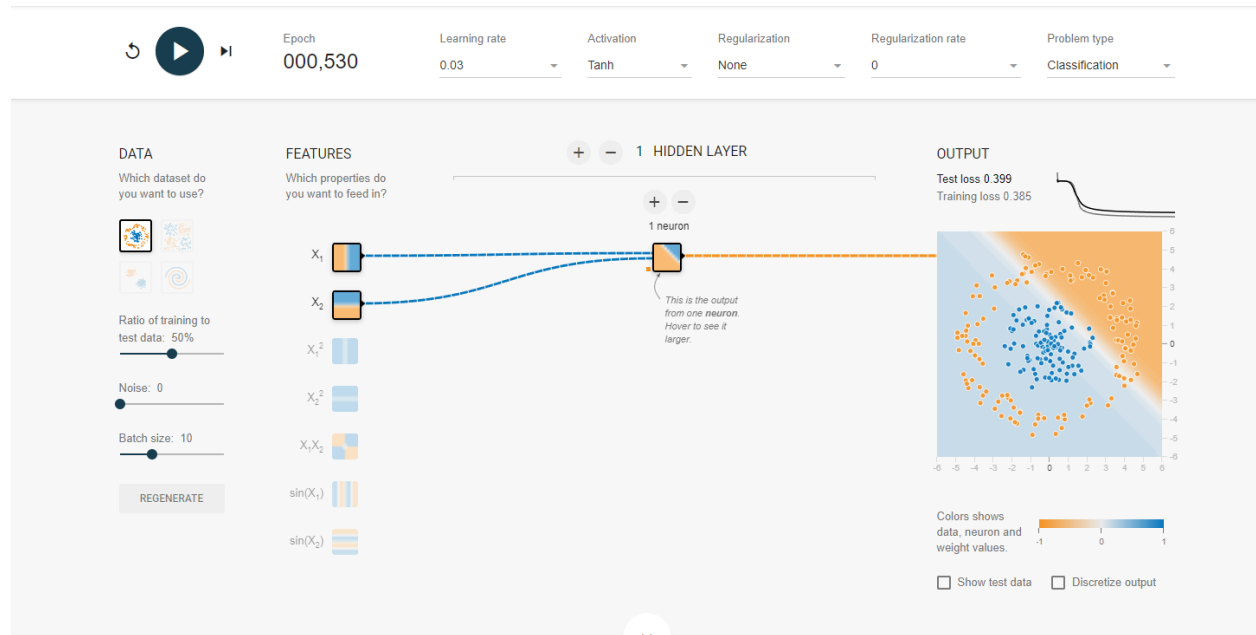


به نام خدا

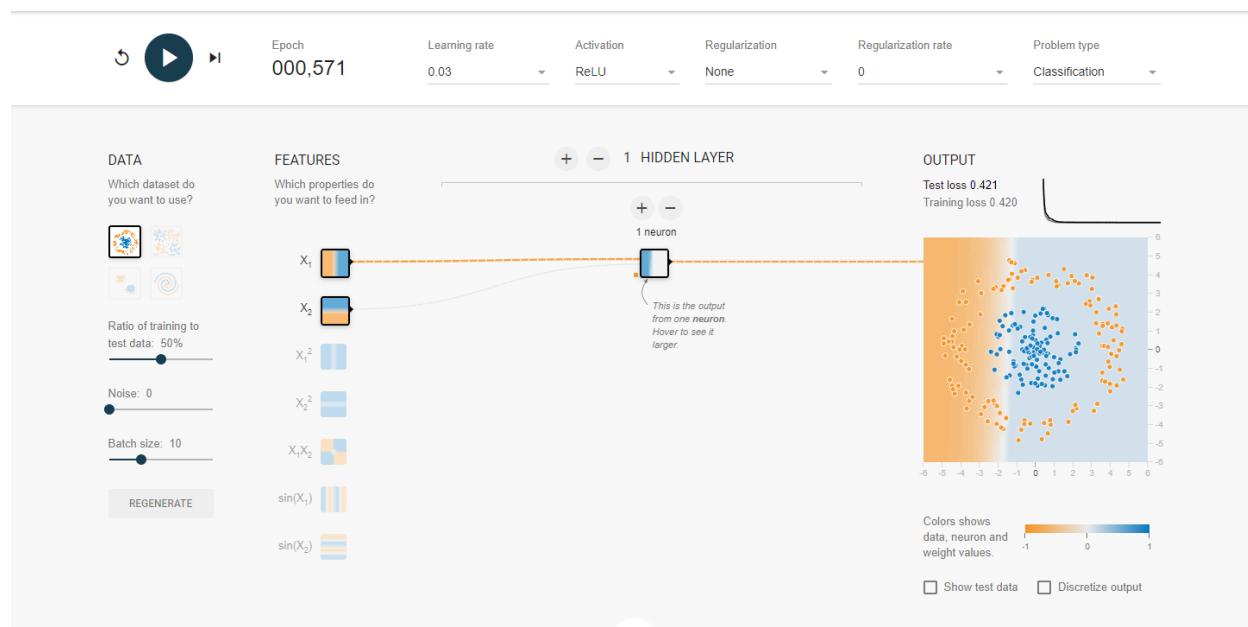
حالت ۱

<http://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=1&seed=0.19844&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



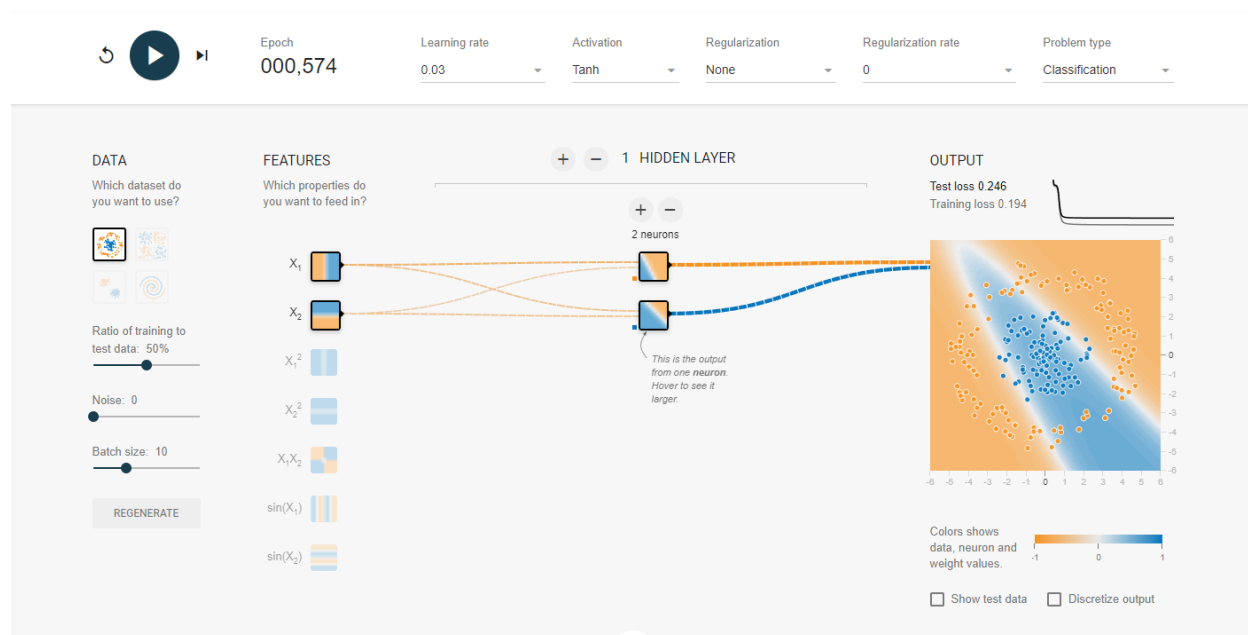
در این حالت تنها از یک perceptron استفاده شده است. چون تک perceptron قادر به حل مسائلی است که جدایی پذیر خطی باشند و این مسئله جدایی پذیر خطی نیست، انتظار داریم مقدار loss زیاد باشد. که همینطور هم می شود.

<http://playground.tensorflow.org/#activation=relu&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=1&seed=0.19844&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



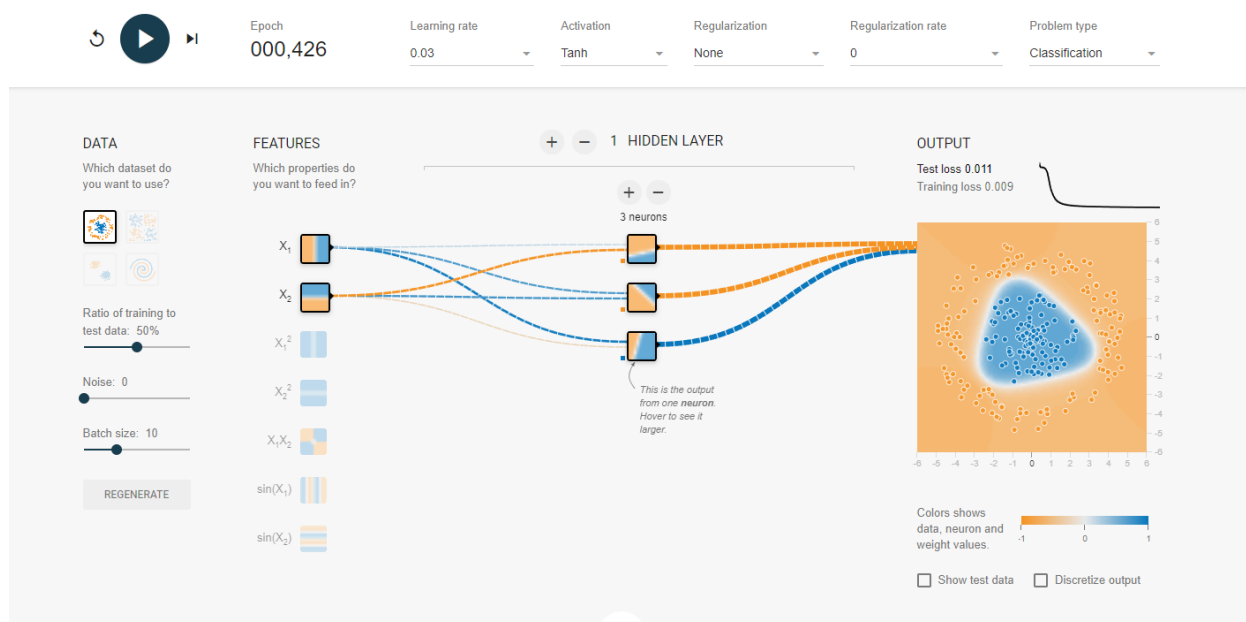
این حالت نیز مشابه حالت قبلی است و تنها activation function این تک perceptron را تغییر دادم (کار بیهوده ای بود).

<http://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=2&seed=0.19844&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



در این حالت از یک لایه و دو نرون استفاده کردم. انتظار داشتم که با این ۲ نرون بتوان ۲ تا خط رسم کرد و feature space را با این دو خط جدا کرد.

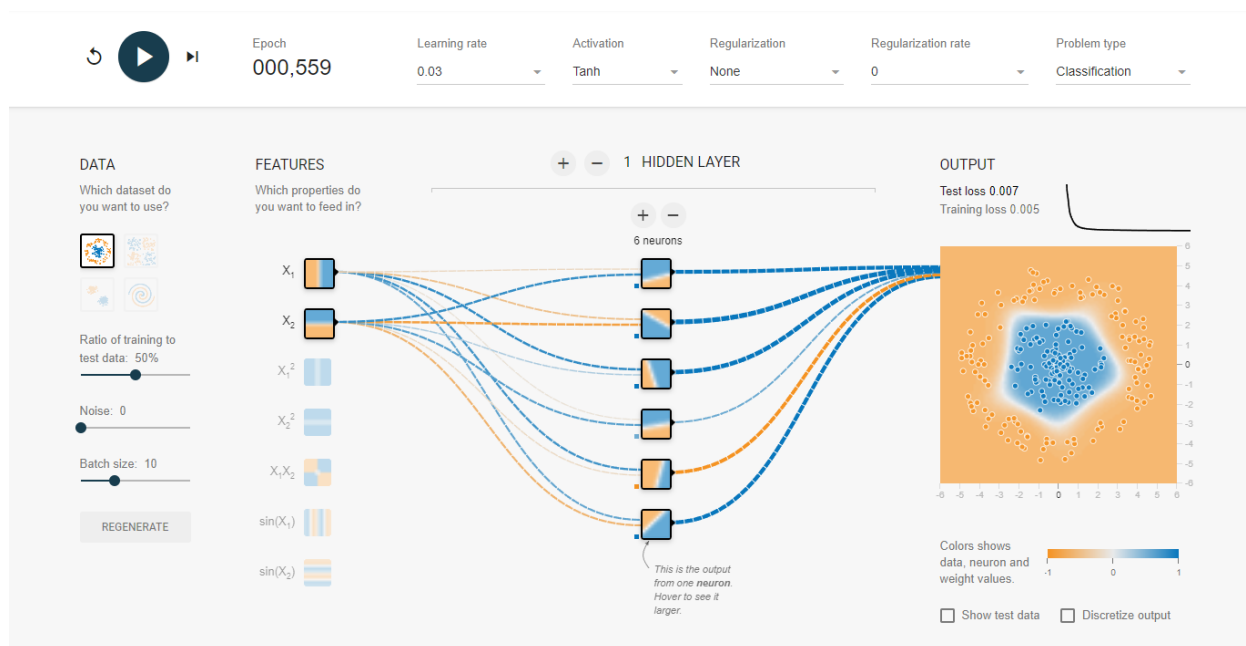
<http://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=3&seed=0.19844&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



تعداد نرون های این تک لایه را به ۳ تا رساندم. همانطور که در تصویر قابل مشاهده است، این feature space توسط ۳ عمل منطقی (and/or) به دو بخش تقسیم شده است.

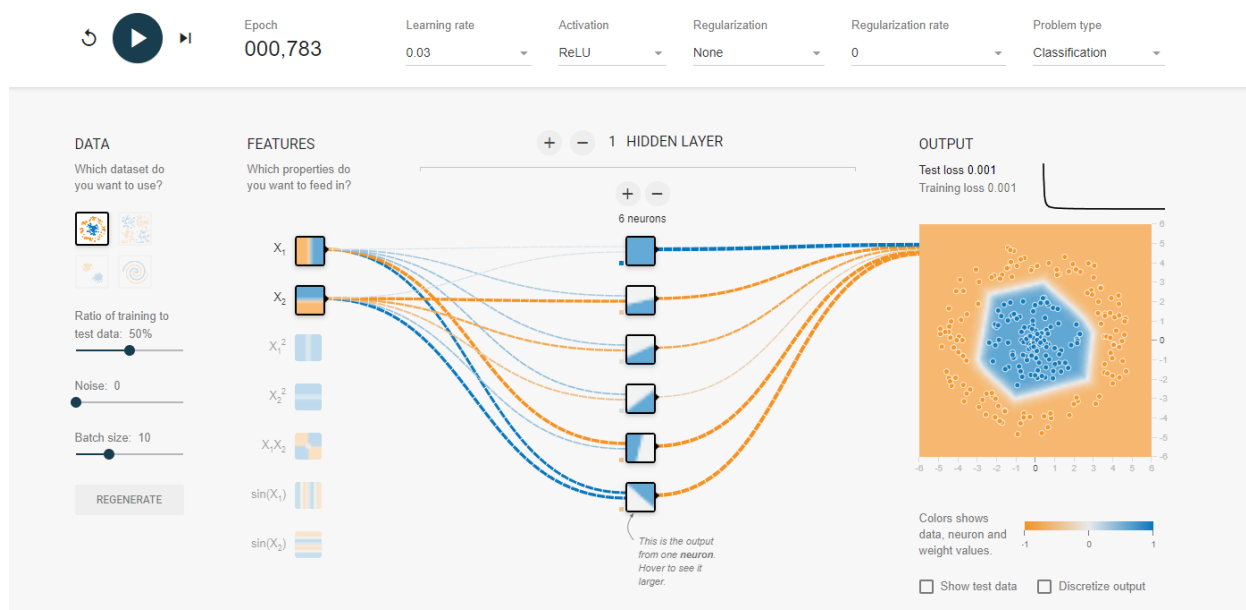
جداسازی ای که هر نرون انجام داده (خطی که رسم کرده) بر روی آن نرون قابل مشاهده است.

<http://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=6&seed=0.19844&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



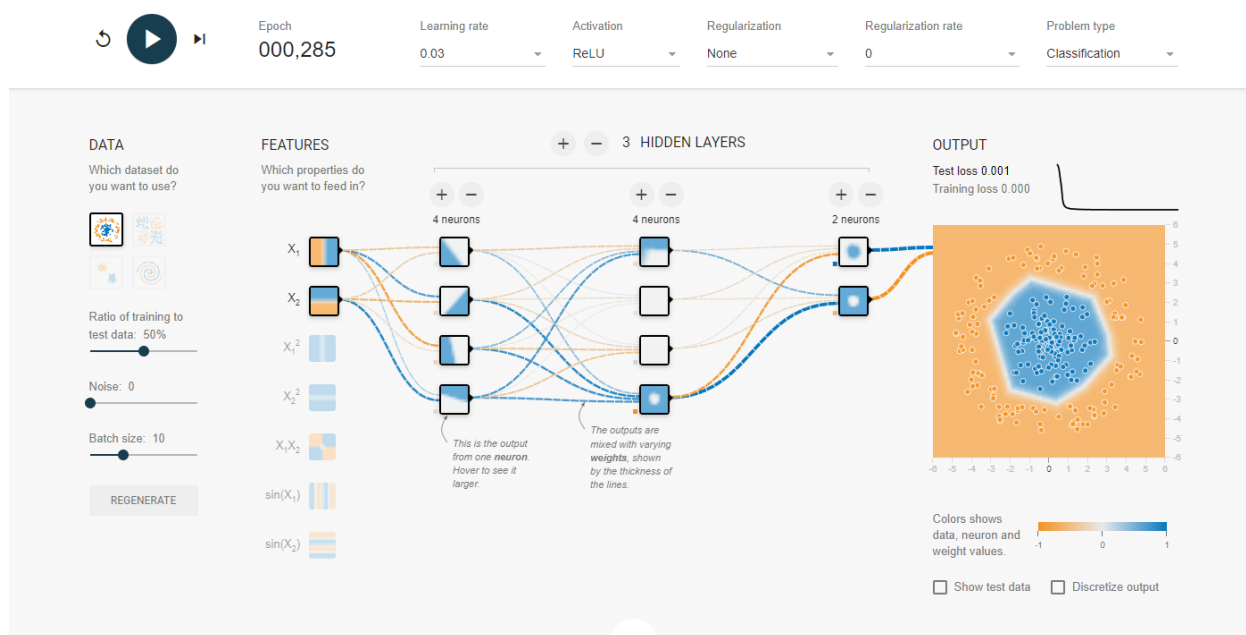
تعداد نرون های این تک لایه را به ۶ تا افزایش دادم. همینطور که در شکل قابل مشاهده است، feature space توسط یک ۵ ضلعی به دو بخش تقسیم شده است. خطوطی که در نرون رسم کرده بر روی تصویر آن قابل مشاهده است.

<http://playground.tensorflow.org/#activation=relu&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=6&seed=0.19844&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



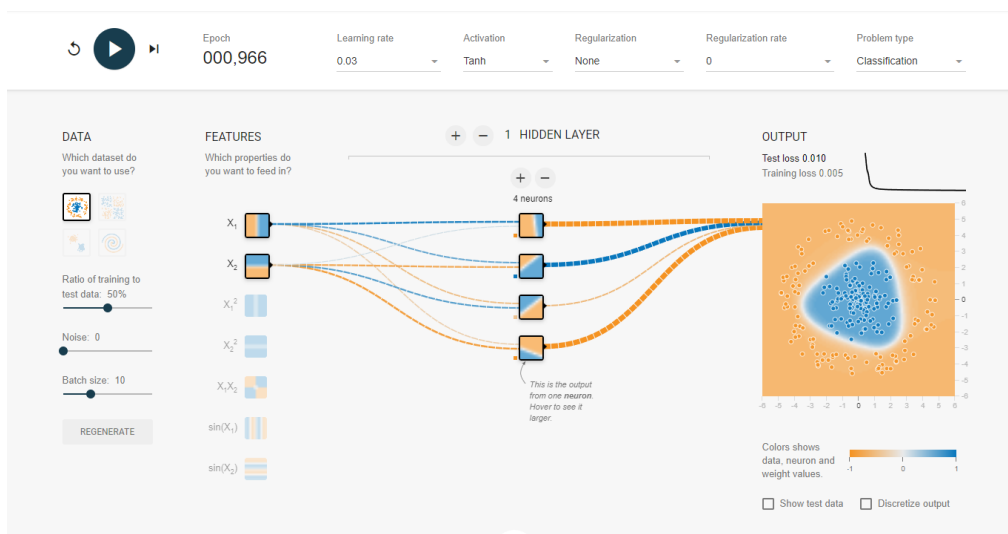
نسبت به حالت قبلی activation function را به Relu تغییر دادم. شکلی که فضا را به دو قسمت تقسیم می کرد از ۵ ضلعی به ۶ ضلعی تغییر کرد. (گویا با استفاده از Relu شکل زاویه دار تر است)

<https://playground.tensorflow.org/#activation=relu&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=4,4,2&seed=0.35533&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>

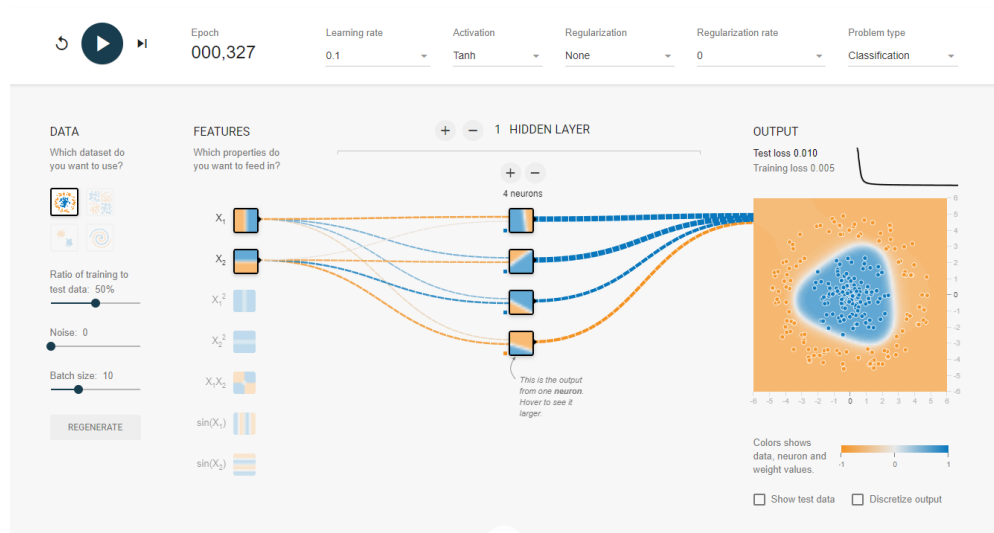


با توجه به ساده بودن جدا سازی داده ها، با اضافه کردن hidden layer های بیشتر، اتفاق خاصی در نحوه جداسازی نیافتاد (ولی با اینکار performance مدل کاهش می یابد چون پارامتر های بیشتری برای آموزش وجود دارد).

<https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=4&seed=0.35533&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>

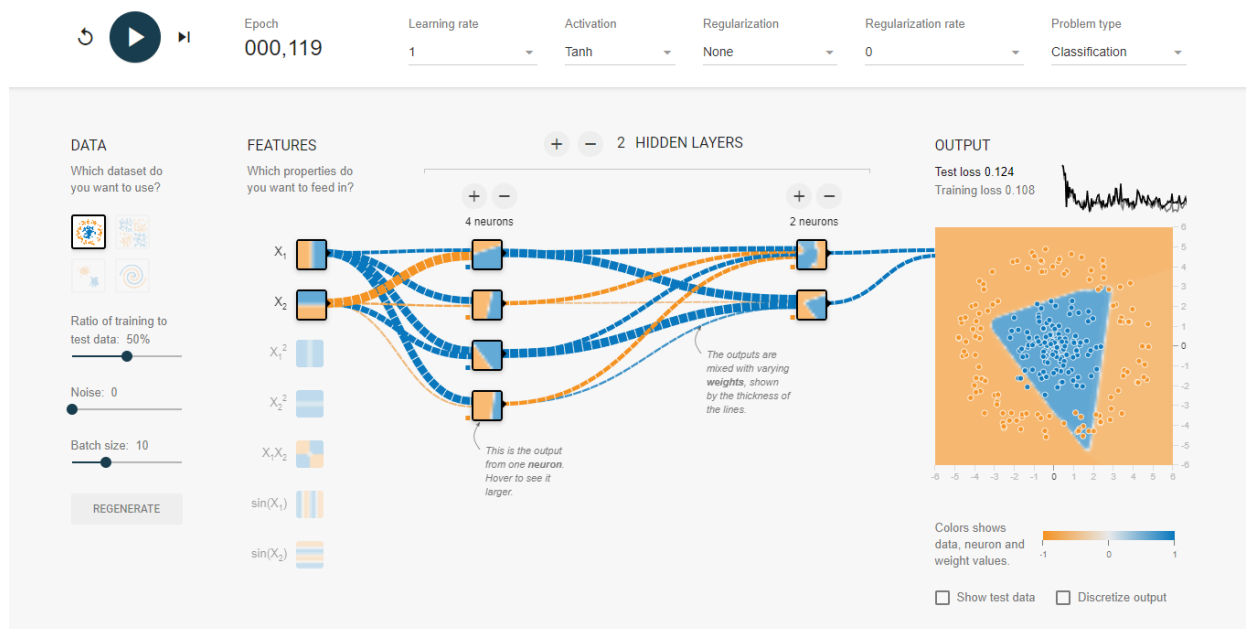


<https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.1®ularizationRate=0&noise=0&networkShape=4&seed=0.35533&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



با افزایش learning rate وزن ها سریعتر به سمت وزن های بهینه همگرا شدند. و loss سریعتر کاهش یافت.

<https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=1®ularizationRate=0&noise=0&networkShape=4,2&seed=0.35533&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>



با افزایش بیش از حد learning rate مدل دچار ناپایداری شد.

