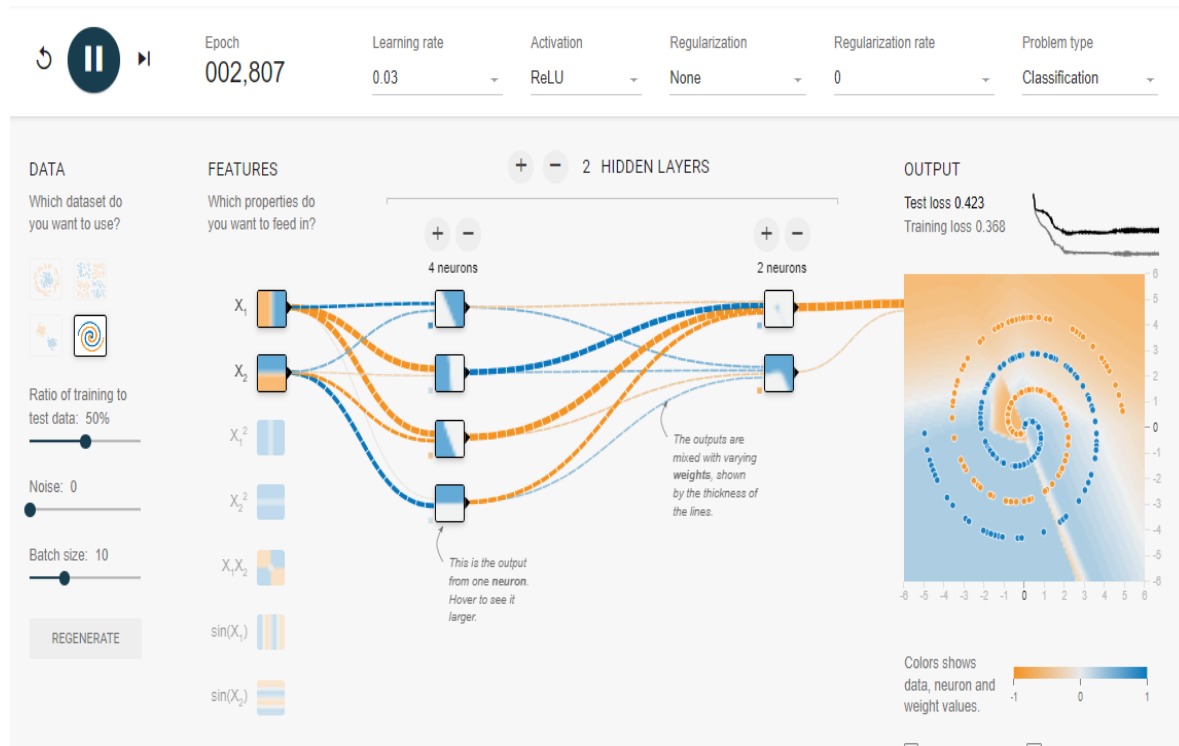


## Informe de la implementación de Tensorflow playground

### Configuración de Parámetros y Visualización de la Red Neuronal

#### Ejemplo 1: Clasificación de Datos en Espiral



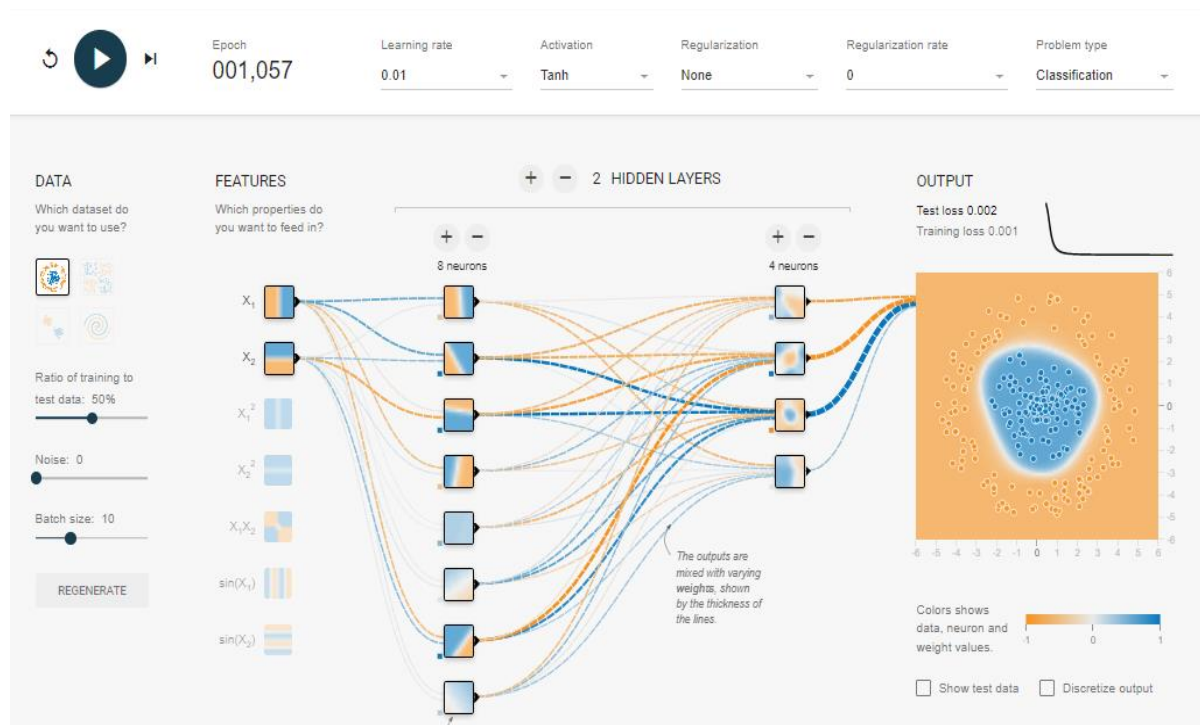
#### Conclusiones:

**Capacidad de la Red Simple:** Una red neuronal simple con una sola capa oculta puede aprender patrones no lineales complejos, como los datos en espiral, aunque puede requerir más tiempo de entrenamiento.

**Eficiencia de ReLU:** La función de activación ReLU es efectiva para entrenar redes profundas, pero en redes simples puede no siempre ser la mejor opción si los datos son extremadamente no lineales.

**Tasa de Aprendizaje:** Una tasa de aprendizaje moderada (0.03) permite a la red aprender de manera eficiente sin hacer grandes saltos que podrían hacer que el modelo no converja.

## Ejemplo 2: Clasificación de Datos en Forma de Círculos



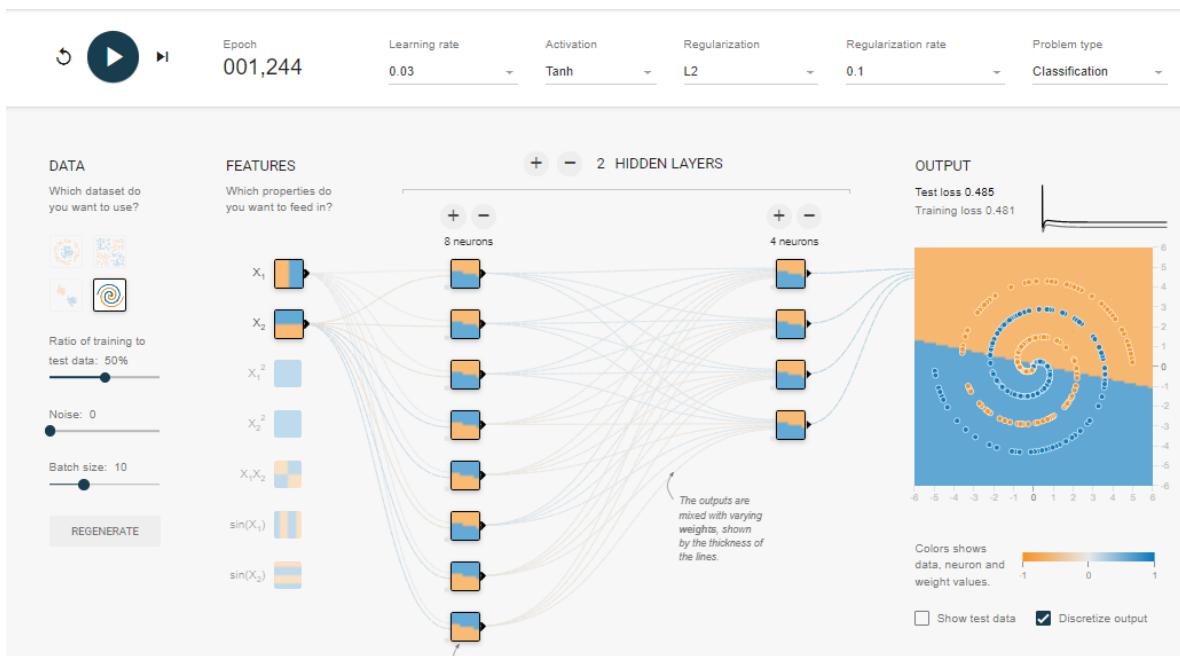
### Conclusiones:

**Profundidad y Capacidad de la Red:** Añadir más capas y neuronas permite que la red aprenda patrones más complejos y capture mejor la estructura del conjunto de datos. En este caso, dos capas ocultas con 8 y 4 neuronas respectivamente fueron suficientes para aprender los datos en forma de círculo.

**Función de Activación tanh:** La función de activación tanh, que puede modelar relaciones más complejas y no lineales que ReLU, puede ser más adecuada para ciertos tipos de datos.

**Balance de Complejidad:** Demasiadas capas o neuronas pueden resultar en un modelo que sobreajuste los datos de entrenamiento, mientras que muy pocas pueden resultar en subajuste. Encontrar el equilibrio es clave.

### Ejemplo 3: Clasificación con Regularización



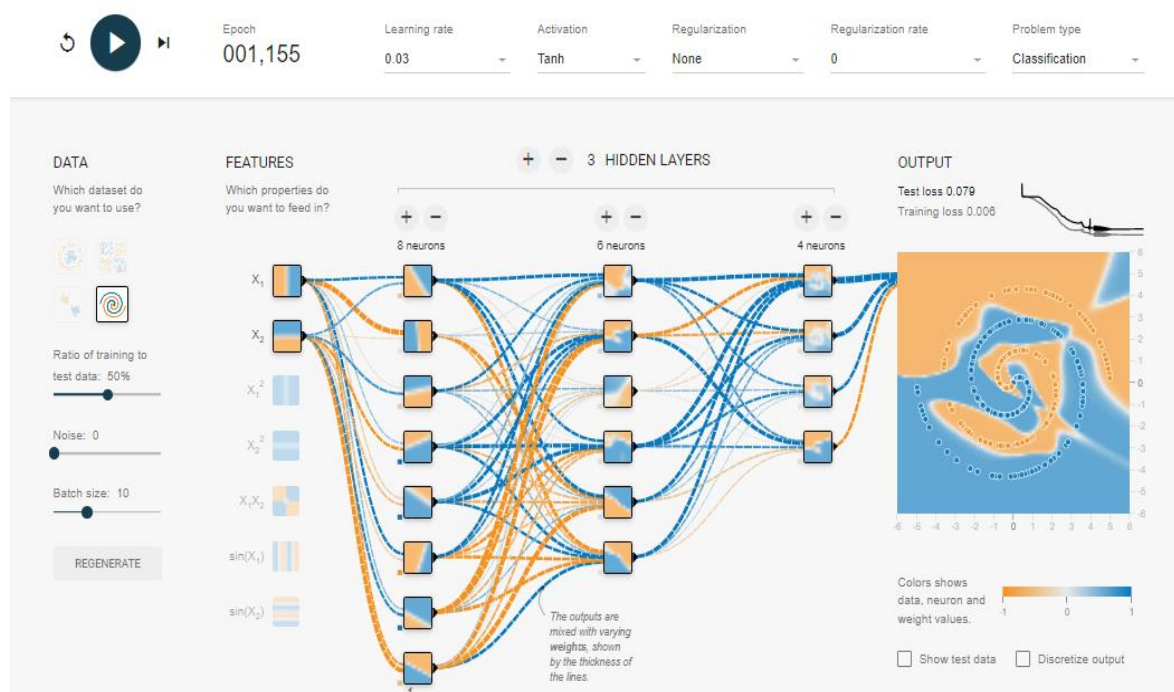
#### Conclusiones:

Efecto de la Regularización L2: La regularización L2 ayuda a reducir el sobreajuste penalizando los pesos grandes, lo que hace que el modelo sea más simple y generalice mejor a datos no vistos.

Generalización: Aplicar una regularización adecuada puede mejorar significativamente la capacidad del modelo para generalizar a nuevos datos, aunque puede reducir un poco la precisión en los datos de entrenamiento.

Balance de Regularización: Demasiada regularización puede hacer que el modelo no aprenda lo suficiente (subajuste), mientras que muy poca puede hacer que el modelo aprenda demasiado los datos de entrenamiento (sobreajuste).

## Ejemplo 4: Exploración de Sobreajuste



### Conclusiones:

**Sobreajuste en Redes Complejas:** Una red con muchas capas y neuronas puede sobre ajustarse a los datos de entrenamiento, especialmente si no se utiliza ninguna técnica de regularización. Esto se manifiesta en una alta precisión en el conjunto de entrenamiento, pero un rendimiento pobre en datos no vistos.

**Visualización de Superficie de Decisión:** Al observar la superficie de decisión del modelo, se puede ver que el modelo sobre ajustado puede capturar incluso el ruido en los datos de entrenamiento, resultando en fronteras de decisión muy complejas.

**Necesidad de Validación:** Es crucial validar el modelo con un conjunto de datos separado para evaluar su rendimiento en datos no vistos y ajustar la complejidad de la red y la regularización en consecuencia.

### Conclusión General

Experimentar con TensorFlow Playground puede ayudar a entender cómo diferentes configuraciones de redes neuronales y ajustes de hiperparámetros afectan el rendimiento del modelo

**Equilibrio entre Complejidad y Generalización:** Encontrar el balance adecuado entre la complejidad del modelo y su capacidad para generalizar a nuevos datos es crucial. Esto implica ajustar la cantidad de capas y neuronas, así como aplicar regularización cuando sea necesario.

**Importancia de la Regularización:** La regularización es una herramienta poderosa para evitar el sobreajuste y mejorar la generalización del modelo.

**Función de Activación y Tasa de Aprendizaje:** La elección de la función de activación y la tasa de aprendizaje puede tener un impacto significativo en el rendimiento del modelo y su capacidad para aprender de los datos.

**Validación y Evaluación Continua:** Evaluar continuamente el modelo con datos de validación es esencial para asegurarse de que no está sobreajustando y para realizar los ajustes necesarios en la configuración del modelo.

Estos experimentos en TensorFlow Playground te brindan una comprensión práctica e intuitiva de cómo funcionan las redes neuronales y cómo afectan los diferentes parámetros y configuraciones al rendimiento del modelo.