

Nama : Rizki Ramadhan

NIM : 1103213091

Machine Learning

Dataset *Dummy* :

```
X, y = make_classification(n_samples=1000, n_features=20, n_classes=2, random_state=42)
```

Setelah melakukan *Hypertuning* dengan beberapa konfigurasi :

```
hidden_layers_options = [1, 2, 3] # Number of hidden layers
hidden_neurons_options = [4, 8, 16, 32, 64] # Number of neurons in each hidden layer
activation_functions = ['linear', 'Sigmoid', 'ReLU', 'Softmax', 'Tanh'] # Activation functions
epochs_options = [25, 50, 100, 250] # Number of epochs
learning_rates = [0.1, 0.01, 0.001, 0.0001] # Learning rates
batch_sizes = [64, 128, 256, 512] # Batch sizes
```

Didapatkan beberapa hasil yang dibagi berdasarkan aktivasi fungsinya :

#### A. *Linear*

```
Best Configuration for Activation Function linear:
Hidden Layers: 3
Hidden Neurons: 8
Activation Function: linear
Epochs: 250
Learning Rate: 0.1
Batch Size: 64
Test Accuracy: 87.33%
```

Setelah melakukan pengujian, didapatkan konfigurasi terbaik untuk fungsi aktivasi *linear* dengan tiga hidden *layers* dan 8 *neuron* di setiap layer. Penggunaan tiga hidden *layers* memberikan kapasitas yang lebih besar bagi model untuk mempelajari representasi data yang lebih kompleks, meskipun jumlah *neuron* yang relatif kecil (8) di setiap layer membantu menjaga model tetap sederhana dan mencegah *overfitting*. Fungsi aktivasi *linear*, meskipun tidak ideal untuk masalah klasifikasi non-*linear*, tetap berhasil dalam konfigurasi ini untuk menghasilkan hasil yang cukup baik, mungkin karena kesederhanaannya dalam pengolahan data.

Model dilatih selama 250 *epoch* dengan *learning rate* sebesar 0.1. Penggunaan jumlah *epoch* yang cukup banyak memberikan waktu yang cukup bagi model untuk beradaptasi dengan data dan mencapai konvergensi yang lebih baik. *Learning rate* yang cukup besar memungkinkan pembaruan parameter yang cepat dalam setiap iterasi, yang dapat mempercepat proses pelatihan dan meningkatkan kinerja model secara keseluruhan. Kombinasi antara *epoch* yang banyak dan *learning rate* yang tepat memungkinkan model untuk belajar dengan lebih efektif.

Dengan *test accuracy* sebesar 87.33%, model menunjukkan performa yang sangat baik dalam klasifikasi data uji. Hasil ini menunjukkan bahwa meskipun menggunakan fungsi aktivasi *linear*,

model dapat mencapai akurasi yang sangat tinggi berkat konfigurasi *Hyperparameter* yang tepat, seperti jumlah hidden *layers*, *epoch*, dan *learning rate* yang telah dipilih secara optimal.

## B. Sigmoid

```
Best Configuration for Activation Function Sigmoid:  
Hidden Layers: 1  
Hidden Neurons: 8  
Activation Function: Sigmoid  
Epochs: 25  
Learning Rate: 0.1  
Batch Size: 256  
Test Accuracy: 87.00%
```

Setelah melakukan pengujian, didapatkan konfigurasi terbaik untuk fungsi aktivasi *Sigmoid* dengan satu hidden layer dan 8 *neuron*. Penggunaan satu hidden layer membuat model relatif sederhana, tetapi dengan 8 *neuron*, model masih bisa mempelajari pola dari data dengan baik tanpa terlalu rumit. Fungsi aktivasi *Sigmoid* bekerja dengan baik untuk klasifikasi biner karena menghasilkan output antara 0 dan 1, meskipun terkadang bisa menghadapi masalah saat digunakan pada model yang lebih dalam.

Model ini dilatih selama 25 *epoch* dengan *learning rate* sebesar 0.1. Meskipun jumlah *epoch* cukup sedikit, *learning rate* yang lebih tinggi membantu model belajar dengan lebih cepat, memperbarui parameter secara efisien. *Batch size* sebesar 256 memungkinkan model memproses data dalam jumlah besar sekaligus, yang mempercepat proses pelatihan.

Dengan *Test accuracy* sebesar 87.00%, model menunjukkan hasil yang sangat baik dalam mengklasifikasikan data uji. Hasil ini menunjukkan bahwa meskipun model ini sederhana, ia tetap mampu mencapai performa yang tinggi berkat kombinasi yang tepat antara jumlah *epoch*, *learning rate*, dan *batch size*.

## C. ReLU

```
Best Configuration for Activation Function ReLU:  
Hidden Layers: 3  
Hidden Neurons: 4  
Activation Function: ReLU  
Epochs: 250  
Learning Rate: 0.1  
Batch Size: 64  
Test Accuracy: 87.00%
```

Setelah melakukan pengujian, didapatkan konfigurasi terbaik untuk fungsi aktivasi *ReLU* dengan tiga hidden *layers* dan 4 *neuron* di setiap layer. Penggunaan tiga hidden *layers* memungkinkan model untuk belajar representasi data yang lebih kompleks, meskipun jumlah *neuron* yang relatif kecil (4) membantu model tetap ringan dan tidak berisiko *overfitting*.

Fungsi aktivasi *ReLU* sangat efektif dalam mempercepat konvergensi pada masalah klasifikasi, karena dapat mengatasi masalah vanishing gradient dan mempercepat pembelajaran pada layer-layer yang lebih dalam.

Model ini dilatih selama 250 *epoch* dengan *learning rate* sebesar 0.1. *Epoch* yang banyak memberikan cukup waktu bagi model untuk beradaptasi dan menemukan pola yang lebih kompleks dalam data. *Learning rate* yang lebih tinggi memungkinkan model melakukan pembaruan parameter yang cepat, mempercepat proses pelatihan dan membantu model mencapai hasil yang optimal lebih cepat. *Batch size* sebesar 64 juga cukup baik untuk menjaga keseimbangan antara efisiensi dan stabilitas dalam pelatihan.

Dengan *test accuracy* sebesar 87.00%, model menunjukkan performa yang sangat baik dalam mengklasifikasikan data uji. Hasil ini menunjukkan bahwa kombinasi antara jumlah hidden *layers*, *epoch*, *learning rate*, dan *Batch size* yang tepat dapat menghasilkan model yang efektif, bahkan dengan jumlah *neuron* yang relatif kecil di setiap layer.

#### D. Softmax

```
Best Configuration for Activation Function Softmax:  
Hidden Layers: 1  
Hidden Neurons: 4  
Activation Function: Softmax  
Epochs: 50  
Learning Rate: 0.1  
Batch Size: 128  
Test Accuracy: 87.33%
```

Setelah melakukan pengujian, didapatkan konfigurasi terbaik untuk fungsi aktivasi *Softmax* dengan satu hidden layer dan 4 *neuron*. Penggunaan satu hidden layer membatasi kompleksitas model, tetapi dengan 4 *neuron*, model tetap dapat mempelajari pola yang cukup baik tanpa menjadi terlalu rumit. Fungsi aktivasi *Softmax*, meskipun lebih umum digunakan dalam masalah klasifikasi multi-kelas, juga dapat bekerja efektif dalam klasifikasi biner, karena mampu menghasilkan probabilitas kelas yang jelas dan memudahkan pemahaman tentang keputusan model.

Model ini dilatih selama 50 *epoch* dengan *learning rate* sebesar 0.1. Dengan jumlah *epoch* yang lebih banyak, model diberi waktu yang cukup untuk beradaptasi dan mengoptimalkan parameter, menghasilkan model yang lebih akurat. *Learning rate* yang lebih tinggi memungkinkan pembaruan parameter yang lebih cepat, mempercepat konvergensi dan meningkatkan kecepatan pelatihan. *Batch size* sebesar 128 membantu menjaga keseimbangan antara kecepatan pelatihan dan stabilitas model selama proses pembelajaran.

Dengan *test accuracy* sebesar 87.33%, model menunjukkan performa yang sangat baik dalam mengklasifikasikan data uji. Hasil ini menunjukkan bahwa meskipun menggunakan fungsi aktivasi

*Softmax*, model dapat mencapai akurasi yang sangat tinggi berkat kombinasi yang tepat antara jumlah hidden *layers*, *epoch*, *learning rate*, dan *batch size*.

## E. Tanh

```
Best Configuration for Activation Function Tanh:  
Hidden Layers: 1  
Hidden Neurons: 32  
Activation Function: Tanh  
Epochs: 25  
Learning Rate: 0.01  
Batch Size: 512  
Test Accuracy: 87.33%
```

Setelah melakukan pengujian, didapatkan konfigurasi terbaik untuk fungsi aktivasi *Tanh* dengan satu hidden layer dan 32 *neuron*. Penggunaan satu hidden layer memungkinkan model tetap sederhana, namun dengan jumlah *neuron* yang lebih banyak (32), model mampu mempelajari pola data yang lebih kompleks tanpa menjadi terlalu rumit. Fungsi aktivasi *Tanh* efektif dalam menangani masalah non-linear karena dapat menghasilkan output dalam rentang yang lebih lebar (-1 hingga 1), memberikan model fleksibilitas dalam menangkap variasi dalam data.

Model ini dilatih selama 25 *epoch* dengan *learning rate* sebesar 0.01. Jumlah *epoch* yang lebih sedikit menunjukkan bahwa model mampu mencapai konvergensi yang cukup cepat, sementara *learning rate* yang lebih kecil memberikan pembaruan parameter yang lebih halus dan mengurangi risiko *overshooting*. *Batch size* sebesar 512 memungkinkan model memproses sejumlah besar data dalam satu iterasi, mempercepat pelatihan dan membantu model mencapai hasil yang lebih stabil.

Dengan *test accuracy* sebesar 87.33%, model menunjukkan performa yang sangat baik dalam mengklasifikasikan data uji. Hasil ini menunjukkan bahwa meskipun menggunakan fungsi aktivasi Tanh, model dapat mencapai akurasi yang sangat tinggi berkat konfigurasi *Hyperparameter* yang tepat, termasuk jumlah hidden layers, epoch, learning rate, dan Batch size yang optimal.

### Best Hyperparameter

```
Best Hyperparameter Configuration (Overall):  
Hidden Layers: 1  
Hidden Neurons: 4  
Activation Function: Softmax  
Epochs: 50  
Learning Rate: 0.1  
Batch Size: 128  
Test Accuracy: 87.33%
```

Setelah melakukan Test dan Train didapatkan hasil konfigurasi terbaik dengan konfigurasi *Hidden Layers: 1 Hidden Neurons: 4, Activation Function: Softmax, Epochs: 50, Learning rate: 0.1, Batch size: 128*, dengan Test Accuracy: 87.33%. Dalam konfigurasi *Hyperparameter* terbaik ini pada dataset dummy, model menggunakan satu hidden layer dengan 4 *neuron* dan fungsi aktivasi *Softmax*. Penggunaan satu hidden layer membatasi kemampuan model dalam menangkap

pola yang lebih kompleks, tetapi dengan jumlah *neuron* yang relatif kecil (4), model tetap dapat menjaga keseimbangan antara kompleksitas dan generalisasi. Fungsi aktivasi *Softmax*, yang lebih sering digunakan dalam masalah klasifikasi multi-kelas, tetap bekerja dengan baik dalam klasifikasi biner meskipun biasanya *ReLU* lebih disarankan untuk tugas tersebut, karena dapat mempercepat konvergensi dan mencegah masalah vanishing gradient.

Model ini dilatih selama 50 *epoch* dengan *learning rate* sebesar 0.1. *Epoch* yang cukup banyak menunjukkan bahwa model memiliki waktu yang cukup untuk beradaptasi dengan data dan mencapai konvergensi yang lebih baik. *Learning rate* yang lebih besar seperti 0.1 memungkinkan model untuk memperbarui parameter lebih cepat, yang meningkatkan kecepatan pelatihan. Namun, pada beberapa kasus, *learning rate* yang terlalu besar bisa menyebabkan pembaruan parameter yang terlalu besar, tetapi pada konfigurasi ini model masih mampu mengoptimalkan hasil dengan baik.

Dengan *test accuracy* sebesar 87.33%, model menunjukkan performa yang sangat baik, mencapai lebih dari 87% akurasi dalam klasifikasi data uji. Hasil ini menunjukkan bahwa konfigurasi *Hyperparameter* yang dipilih dapat menghasilkan model yang cukup efektif dalam memahami dan memprediksi kelas-kelas yang ada dalam dataset dummy ini.

#### Worst Hyperparameter

```
Worst Hyperparameter Configuration:  
Hidden Layers: 1  
Hidden Neurons: 64  
Activation Function: linear  
Epochs: 25  
Learning Rate: 0.0001  
Batch Size: 512  
Test Accuracy: 28.33%
```

Setelah melakukan pengujian, didapatkan konfigurasi yang kurang optimal dengan satu hidden layer dan 64 *neuron*, serta fungsi aktivasi *linear*. Penggunaan satu hidden layer dengan jumlah *neuron* yang cukup besar (64) dapat membuat model terlalu sederhana untuk menangkap pola-pola kompleks dalam data, terutama karena fungsi aktivasi *linear* tidak dapat menangani masalah non-*linear* dengan baik. Fungsi aktivasi *linear* menghasilkan gradien yang konstan, yang membatasi kemampuan model untuk belajar dari data dengan cara yang lebih dalam dan efektif.

Model ini dilatih selama 25 *epoch* dengan *learning rate* yang sangat kecil, yaitu 0.0001. *Learning rate* yang terlalu kecil menyebabkan pembaruan parameter yang sangat lambat, sehingga model kesulitan untuk melakukan pembelajaran yang efektif dalam waktu yang terbatas. Meskipun *Batch size* yang besar (512) biasanya mempercepat pelatihan, kombinasi dengan *learning rate*

yang rendah justru menghambat konvergensi model, membuatnya tidak mampu mencapai performa yang optimal.

Dengan *test accuracy* hanya sebesar 28.33%, model menunjukkan hasil yang sangat buruk dalam klasifikasi data uji. Akurasi yang rendah ini mengindikasikan bahwa model gagal dalam mempelajari pola yang relevan dari data, yang kemungkinan disebabkan oleh pemilihan fungsi aktivasi yang kurang tepat dan *learning rate* yang terlalu kecil, yang menghambat pembaruan parameter yang efektif.

## KESIMPULAN

Berdasarkan pengujian berbagai konfigurasi *Hyperparameter* dengan berbagai fungsi aktivasi, dapat disimpulkan bahwa model dapat mencapai akurasi yang sangat baik (sekitar 87%) dengan berbagai kombinasi, meskipun menggunakan fungsi aktivasi yang berbeda-beda. Konfigurasi dengan satu hidden layer dan jumlah *neuron* yang lebih sedikit, seperti pada fungsi aktivasi *Sigmoid* dan *Softmax*, menunjukkan bahwa model dapat memberikan hasil yang baik meskipun relatif sederhana. Sementara itu, penggunaan tiga hidden *layers* dengan jumlah *neuron* yang lebih kecil, seperti pada fungsi aktivasi *ReLU*, juga berhasil mencapai performa yang optimal.

Pemilihan *epoch*, *learning rate*, dan *Batch size* yang tepat memainkan peran besar dalam menentukan keberhasilan model. Penggunaan *epoch* yang lebih banyak (seperti 250) memberikan model lebih banyak kesempatan untuk belajar, sementara *learning rate* yang lebih besar (0.1) memungkinkan pembaruan parameter yang lebih cepat dan lebih efisien. *Batch size* yang lebih besar (512) membantu dalam meningkatkan efisiensi pelatihan tanpa mengorbankan stabilitas model.

Secara keseluruhan, meskipun ada variasi dalam jumlah hidden *layers* dan jenis fungsi aktivasi yang digunakan, model dengan konfigurasi *Hyperparameter* yang tepat mampu mencapai akurasi yang sangat baik di atas 87%. Ini menunjukkan bahwa pemilihan kombinasi *Hyperparameter* yang tepat sangat penting dalam meningkatkan performa model dalam masalah klasifikasi.