

**Nama : Muhamad Rizq Rihaz**

**NIM : 1103210192**

## **1. Hidden Layers**

### **Deskripsi:**

- Hidden layers adalah lapisan tersembunyi dalam jaringan saraf yang bertugas memproses pola data.

### **Temuan:**

- **1 Hidden Layer:**
  - Memberikan akurasi tinggi pada dataset yang sederhana, tetapi kurang mampu menangkap pola kompleks.
  - Berguna untuk tugas dengan kompleksitas rendah.
  - Akurasi optimal tercapai dalam kombinasi parameter tertentu (**ReLU, 50 Epochs, LR=0.01**).
- **2 Hidden Layers:**
  - Menawarkan stabilitas yang lebih baik dibandingkan 1 hidden layer.
  - Memberikan akurasi tinggi di hampir semua kombinasi parameter.
  - Cocok untuk dataset dengan pola yang lebih kompleks tanpa terlalu membebani sumber daya komputasi.
- **3 Hidden Layers:**
  - Memberikan akurasi maksimal pada dataset kompleks, tetapi memerlukan tuning parameter yang lebih presisi.
  - Membutuhkan lebih banyak sumber daya dan waktu pelatihan.
  - Tidak selalu memberikan peningkatan signifikan dibandingkan 2 hidden layers, kecuali pada konfigurasi optimal.

### **Kesimpulan:**

- **2 Hidden Layers** adalah kompromi terbaik antara performa, stabilitas, dan efisiensi.
- Gunakan **3 Hidden Layers** untuk data yang lebih besar atau lebih kompleks.

## 2. Activation Function

### Deskripsi:

- Fungsi aktivasi bertugas memperkenalkan non-linearitas ke jaringan saraf, memungkinkan model belajar pola yang lebih kompleks.

### Temuan:

- **ReLU:**
  - Fungsi aktivasi terbaik secara keseluruhan, memberikan akurasi tertinggi dan stabil di berbagai kombinasi parameter.
  - Tidak memiliki masalah saturasi gradien seperti Sigmoid atau Tanh.
  - Cepat dan efisien karena hanya mempertahankan nilai positif.
- **Tanh:**
  - Alternatif yang baik untuk data terpusat karena rentang outputnya antara -1 hingga 1.
  - Kadang mencapai akurasi tinggi, tetapi kurang stabil dibandingkan ReLU.
- **Sigmoid:**
  - Sering memberikan akurasi lebih rendah karena masalah saturasi gradien, terutama pada nilai yang sangat kecil atau besar.
  - Tidak direkomendasikan untuk hidden layers, tetapi bisa digunakan di output layer untuk tugas klasifikasi biner.

### Kesimpulan:

- Gunakan **ReLU** untuk hidden layers karena memberikan performa optimal secara konsisten.
- **Tanh** dapat digunakan untuk dataset yang sangat terpusat, tetapi tidak seefisien ReLU.
- Hindari **Sigmoid** untuk hidden layers.

### 3. Epochs

#### Deskripsi:

- Epoch adalah jumlah iterasi penuh melalui dataset selama pelatihan.

#### Temuan:

- **10 Epochs:**
  - Tidak cukup untuk melatih model secara optimal, menghasilkan akurasi lebih rendah pada sebagian besar kombinasi.
  - Hanya cocok untuk tugas sederhana atau jika waktu pelatihan sangat terbatas.
- **50 Epochs:**
  - Memberikan akurasi maksimal dan memastikan model terlatih dengan baik.
  - Sebagian besar kombinasi mencapai akurasi tinggi atau sempurna dengan jumlah ini.

#### Kesimpulan:

- Gunakan **50 Epochs** untuk pelatihan optimal.
- Tambahkan lebih banyak epoch jika dataset lebih besar atau lebih kompleks, tetapi **monitor overfitting**.

### 4. Learning Rate

#### Deskripsi:

- Learning rate mengontrol seberapa besar perubahan bobot model selama pembaruan.

#### Temuan:

- **0.01:**
  - Memberikan hasil terbaik di hampir semua kombinasi parameter.
  - Seimbang antara kecepatan konvergensi dan stabilitas pelatihan.
- **0.001:**
  - Stabil, tetapi lebih lambat untuk mencapai konvergensi.

- Berguna untuk dataset yang kompleks atau jika model sensitif terhadap pembaruan besar.
- **Learning Rate besar (10, 1):**
  - Menyebabkan **divergensi** dan performa buruk karena langkah pembaruan terlalu agresif.
- **Learning Rate kecil (0.0001):**
  - Sangat lambat untuk konvergen, sering memberikan akurasi lebih rendah karena model tidak terlatih cukup baik.

#### Kesimpulan:

- Gunakan **0.01** untuk pelatihan yang cepat dan stabil.
- Gunakan **0.001** jika dataset lebih besar atau lebih kompleks untuk mencegah pembaruan yang tidak stabil.

## 5. Batch Size

#### Deskripsi:

- Batch size menentukan jumlah sampel yang diproses dalam satu iterasi pelatihan.

#### Temuan:

- Batch size **32 dan 128** memberikan hasil konsisten baik di hampir semua kombinasi parameter.
- Batch size kecil (16) memungkinkan pembaruan bobot lebih sering, tetapi memerlukan lebih banyak iterasi.
- Batch size besar (512) memanfaatkan GPU lebih efisien, tetapi memerlukan memori lebih besar.

#### Kesimpulan:

- Gunakan **32 atau 128** untuk pelatihan yang efisien dan hasil yang konsisten.
- Pilih batch size sesuai dengan kapasitas GPU atau CPU yang tersedia.

## Kesimpulan Akhir

Berdasarkan hasil eksperimen, berikut adalah kombinasi parameter terbaik untuk dataset ini:

- **Hidden Layers:** 2.
- **Activation Function:** ReLU.
- **Epochs:** 50.
- **Learning Rate:** 0.01.
- **Batch Size:** 128.

Kombinasi ini memberikan akurasi tertinggi dengan efisiensi yang baik dan dapat diadaptasi untuk dataset lain dengan penyesuaian kecil.