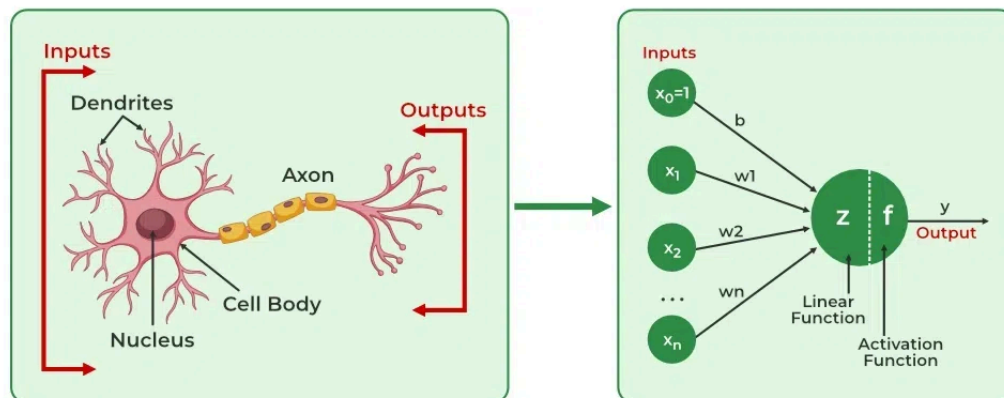
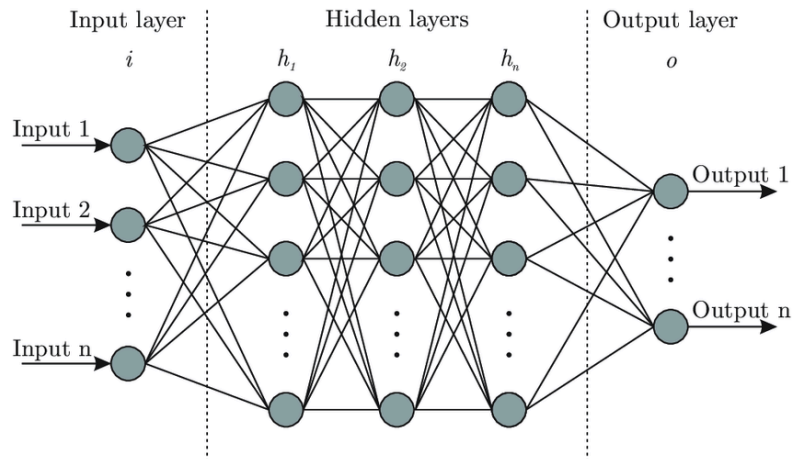


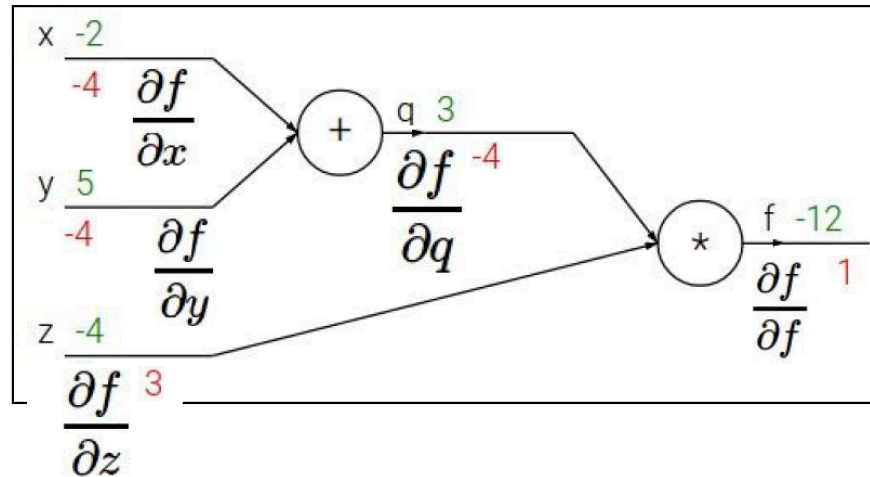
# Artificial Neural Network (ANN)

## Cara Kerja



Artificial neural network yang sederhana bekerja seperti sebuah multilayer perceptron. Setiap layer adalah sekumpulan perceptron yang fully connected. Sebuah perceptron adalah sebuah model linear dengan weights dan bias dan activation function yang diterapkan setelah kalkulasi input terhadap model linear tersebut. Fungsi aktivasi bisa berupa fungsi linear, ReLu, sigmoid. Pada tahap training, terdapat dua proses, yakni feedforward dan backpropagation. Feedforward adalah tahap memasukkan setiap input  $x_i$  pada setiap layer yang dimiliki neural network, mulai dari input layer, melalui hidden layers, lalu mencapai output layer. Proses ini adalah proses kalkulasi model linear biasa yang dilakukan pada setiap perceptron. Setelah mencapai output layer, hasil feedforward dibandingkan dengan label target sesungguhnya dengan menghitung error. Setelah error dihitung, dilakukan backpropagation, yakni mengatur ulang nilai weights dan

bias pada setiap perceptron agar neural network dapat memprediksi dengan lebih baik lagi. Perubahan weights dan bias dari setiap perceptron dihitung dengan persamaan gradien yang didapatkan menggunakan aturan rantai. Perhitungan ini seakan memberitahu setiap perceptron seberapa besar weights dan bias harus diubah dan seberapa besar kontribusi weights dan bias tersebut terhadap error yang terjadi. Proses ini dilakukan untuk sejumlah epoch hingga error menjadi minimum.



## Evaluasi Model

Evaluasi	Hasil
Model from Scratch	

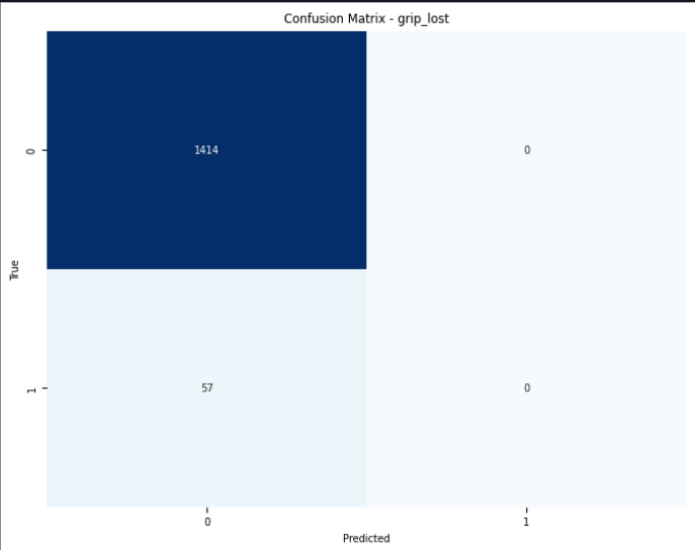
Hold-out validation untuk  
grip\_lost

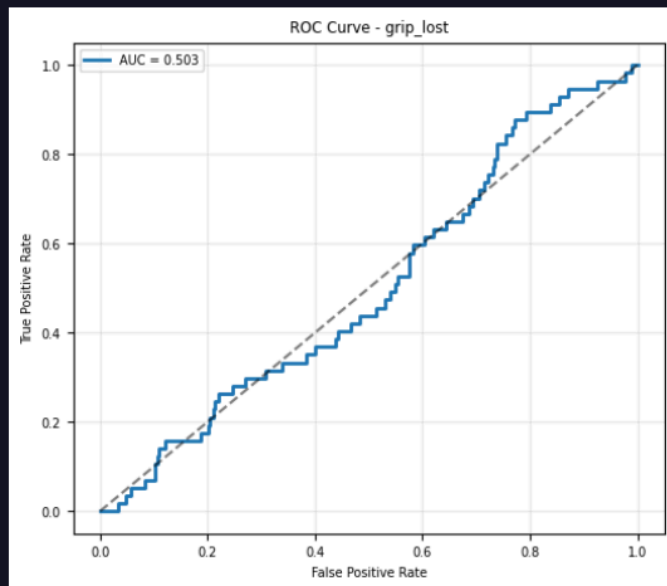
```
=====
Hold-Out Validation for grip_lost:
=====
Epoch 0, Loss: 0.24972869924854246
Epoch 100, Loss: 0.20000153146058433
Epoch 200, Loss: 0.16279544496325488
Epoch 300, Loss: 0.13526461459992875
Epoch 400, Loss: 0.11479579398799535
Epoch 500, Loss: 0.09938029620561818
Epoch 600, Loss: 0.08758321037693566
Epoch 700, Loss: 0.07840789073333763
Epoch 800, Loss: 0.07116118754556015
Epoch 900, Loss: 0.06535736148666794
Training time: 35.29235792160034 seconds
Prediction time: 0.0039975643157958984 seconds

      precision    recall  f1-score   support

     0       0.96       1.00       0.98       1414
     1       0.00       0.00       0.00         57

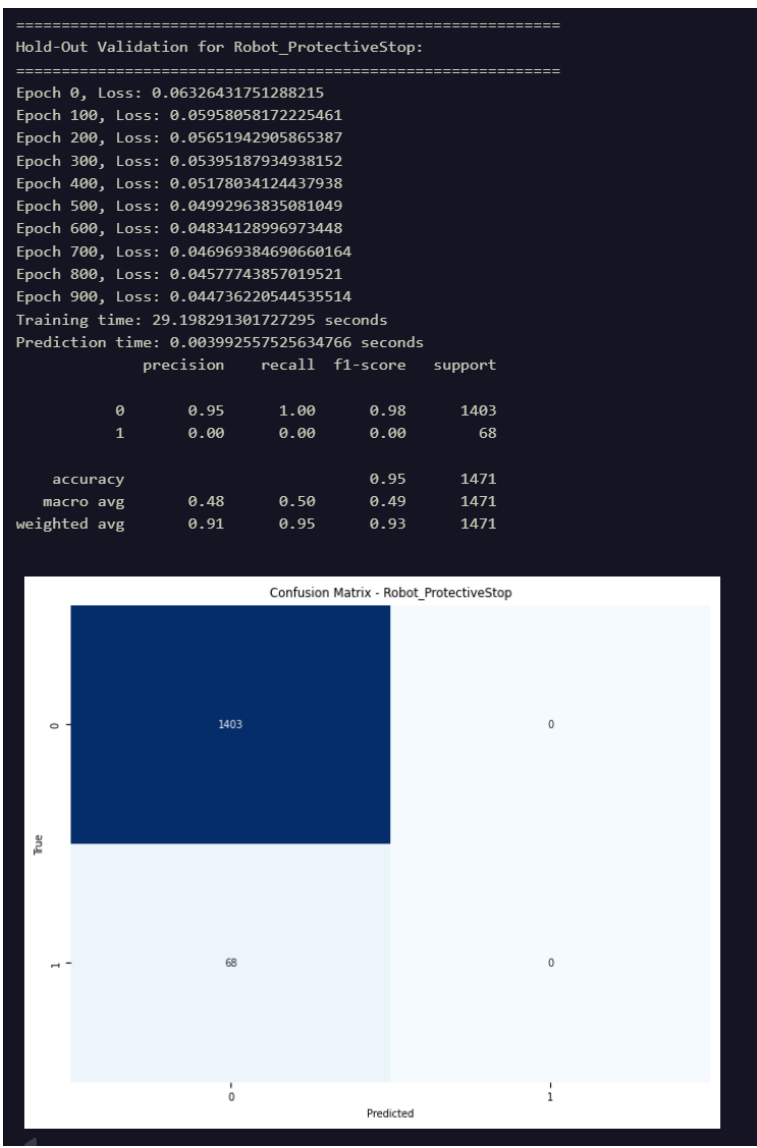
 accuracy          0.96       1471
 macro avg       0.48       0.50       0.49       1471
 weighted avg    0.92       0.96       0.94       1471
```

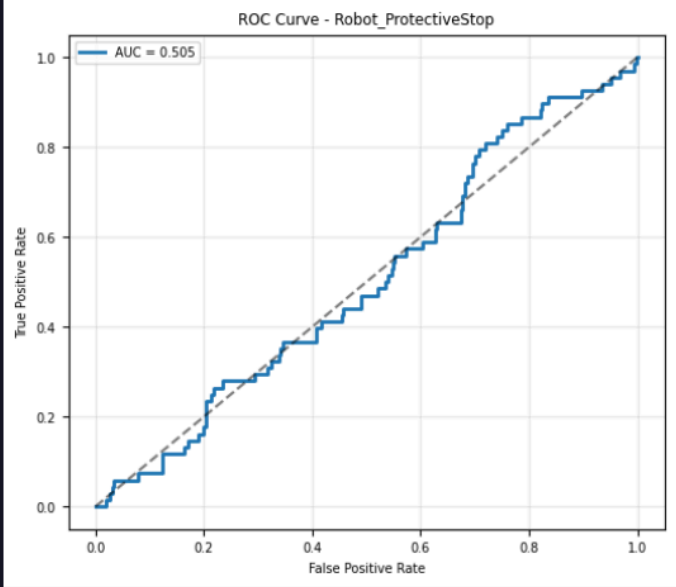




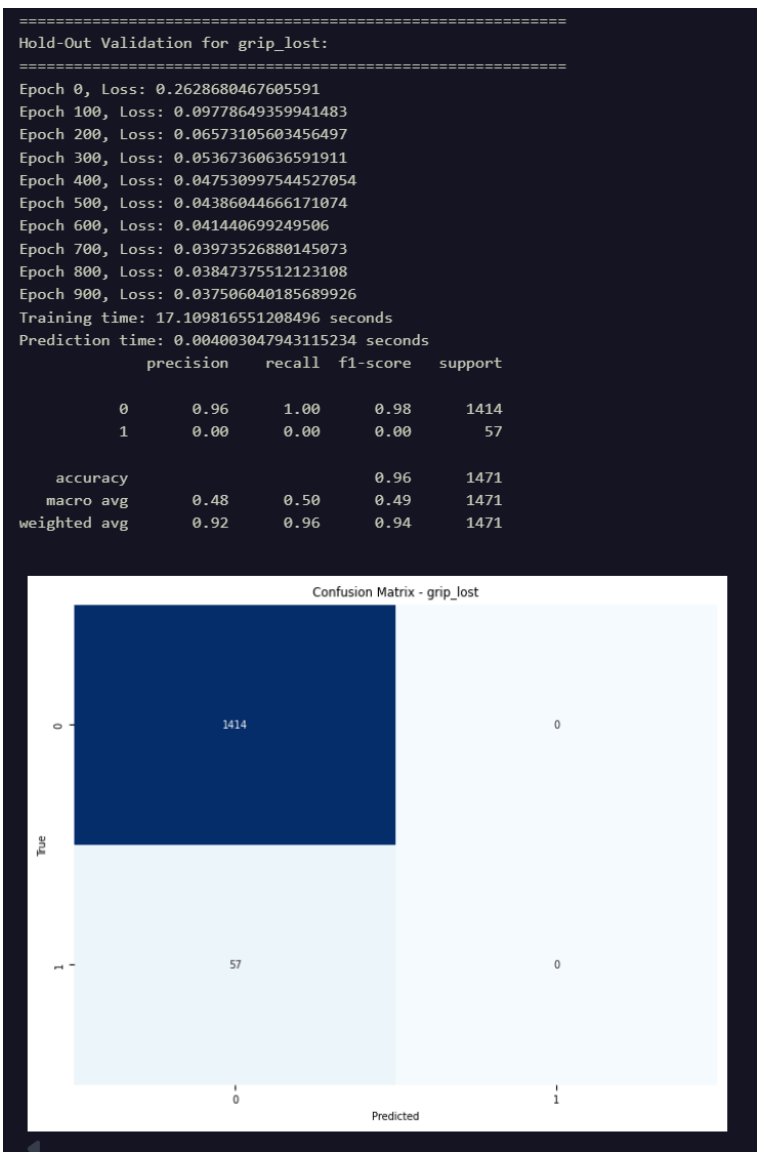
AUC-ROC: 0.5030

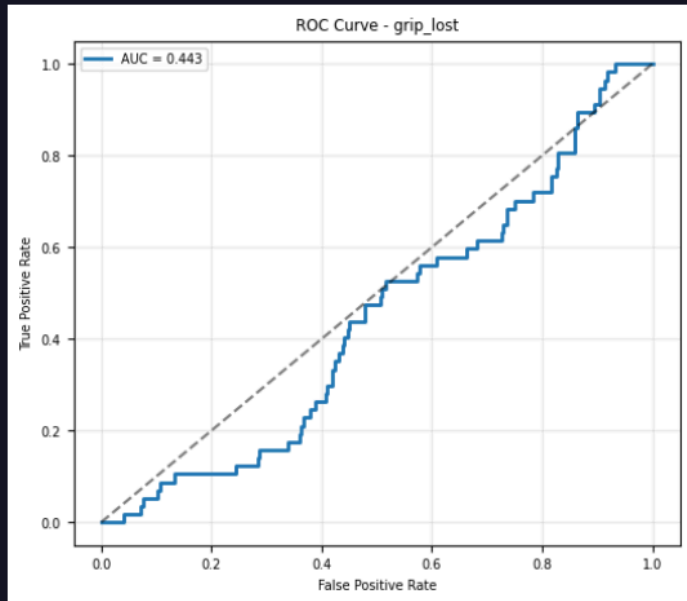
Hold out validation untuk  
Robot\_ProtectiveStop



	<div><p>AUC-ROC: 0.5045</p></div>
K-fold cross validation untuk grip_lost	
K-fold cross validation untuk Robot_ProtectiveStop	
Model Pytorch	

Hold-out validation untuk  
grip\_lost





AUC-ROC: 0.4426



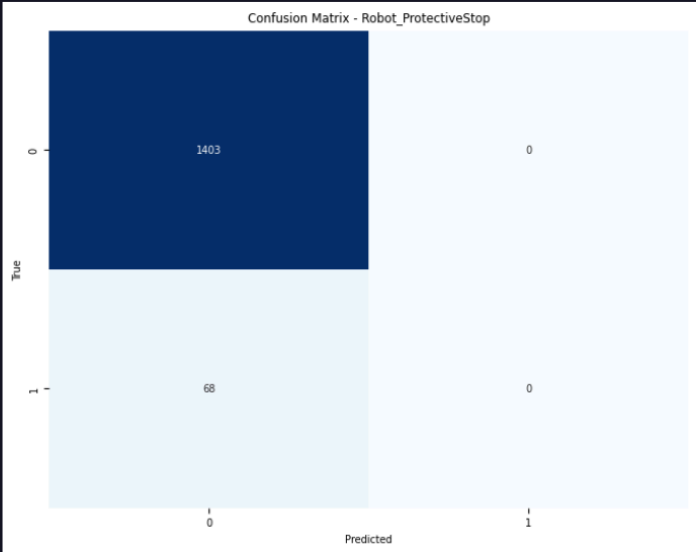
Hold out validation untuk  
Robot\_ProtectiveStop

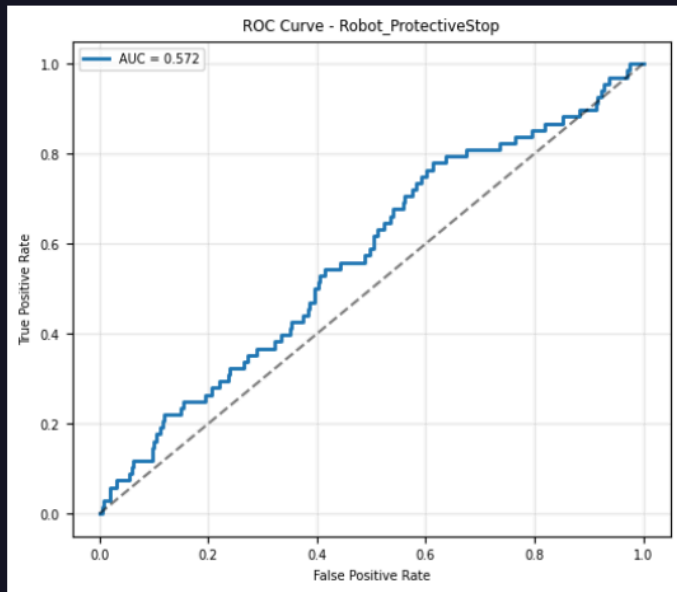
```
=====
Hold-Out Validation for Robot_ProtectiveStop:
=====
Epoch 0, Loss: 0.03924586623907089
Epoch 100, Loss: 0.03864779323339462
Epoch 200, Loss: 0.03815681114792824
Epoch 300, Loss: 0.037746988236904144
Epoch 400, Loss: 0.037400055676698685
Epoch 500, Loss: 0.03710273653268814
Epoch 600, Loss: 0.036845166236162186
Epoch 700, Loss: 0.03661998361349106
Epoch 800, Loss: 0.03642142936587334
Epoch 900, Loss: 0.03624502941966057
Training time: 9.56024432182312 seconds
Prediction time: 0.0029959678649902344 seconds

      precision    recall  f1-score   support

     0         0.95        1.00        0.98        1403
     1         0.00        0.00        0.00         68

 accuracy          0.95        1471
 macro avg         0.48        0.50        0.49        1471
 weighted avg         0.91        0.95        0.93        1471
```





AUC-ROC: 0.5721

K-fold cross validation untuk  
grip\_lost

```
Epoch 600, Loss: 0.034756239503622055
Epoch 700, Loss: 0.03467518091201782
Epoch 800, Loss: 0.034600965678691864
Epoch 900, Loss: 0.034532759338617325
Model Performance (5-Fold Cross Validation):

fit_time: [ 6.81856465  5.4051137  7.48348403  9.55359244 10.47390199]

score_time: [0.0115335  0.0110774  0.01746035 0.01991463 0.01759744]

test_precision: [0. 0. 0. 0. 0.]
Average test_precision: 0.00

test_recall: [0. 0. 0. 0. 0.]
Average test_recall: 0.00

test_f1: [0. 0. 0. 0. 0.]
Average test_f1: 0.00
```

K-fold cross validation untuk  
Robot\_ProtectiveStop

```
Epoch 600, Loss: 0.034756239503622055
Model Performance (5-Fold Cross Validation):

fit_time: [ 8.20627379  8.20563722  9.85033894 12.19046259  8.8550961 ]

score_time: [0.01305652 0.01970458 0.01995707 0.01968241 0.01253366]

test_precision: [0. 0. 0. 0. 0.]
Average test_precision: 0.00

test_recall: [0. 0. 0. 0. 0.]
Average test_recall: 0.00

test_f1: [0. 0. 0. 0. 0.]
Average test_f1: 0.00
```

# Improvement