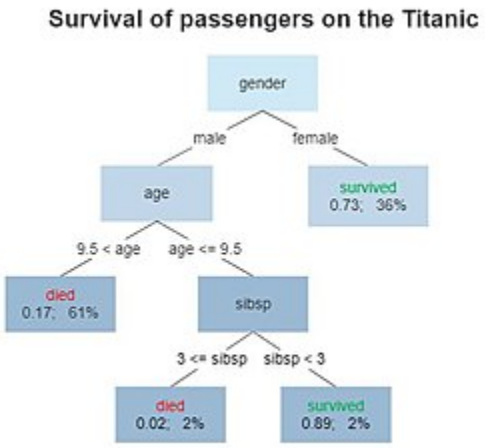
Classification and Regression Tree (CART)

Cara Kerja



Sumber: https://en.wikipedia.org/wiki/Decision_tree_learning

Pada tahap fitting, decision tree dibangun dengan pertanyaan seperti apakah nilai fitur tersebut kurang dari sebuah threshold atau tidak. Setiap node pada tree merupakan sebuah pertanyaan seperti itu. Dalam proses belajarnya, model decision tree membuat pertanyaan yang optimal dengan minimumkan impurity. Nilai impurity dapat dihitung menggunakan gini impurity atau entropy.

$$Gini = 1 - \sum_{i=1}^{n} (p_i)^2$$

Shannon's entropy equation:

$$H(X) = -\sum_{i=0}^{N-1} p_i \log_2 p_i$$

Setelah decision tree sudah dibangun, prediksi dilakukan hanya dengan men-traverse tree tersebut, menjawab pertanyaan-pertanyaan pada tiap node ketika diberikan fitur dari data yang ingin diprediksi.

Evaluasi Model

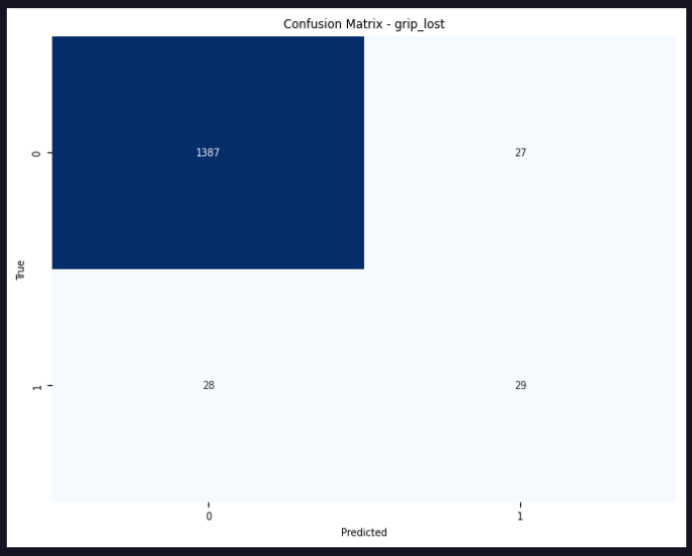
Evaluasi	Hasil
Model from Scratch	

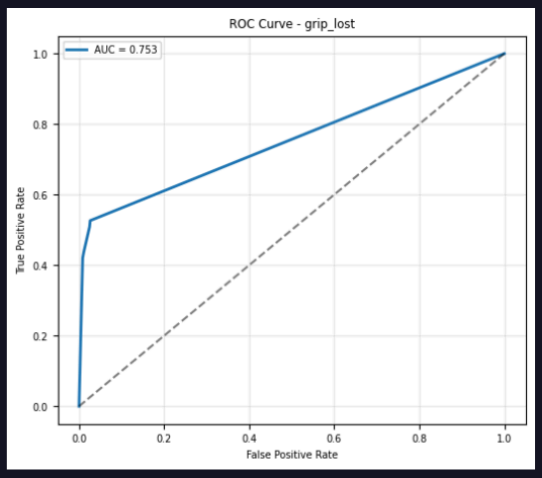
Hold-out validation untuk grip_lost

```
============================================================
Hold-Out Validation for grip_lost:
============================================================
Training time: 12.096389293670654 seconds
Prediction time: 0.004508495330810547 seconds
              precision    recall  f1-score   support

           0       0.98      0.98      0.98      1414
           1       0.52      0.51      0.51        57

    accuracy                           0.96      1471
   macro avg       0.75      0.74      0.75      1471
weighted avg       0.96      0.96      0.96      1471
```





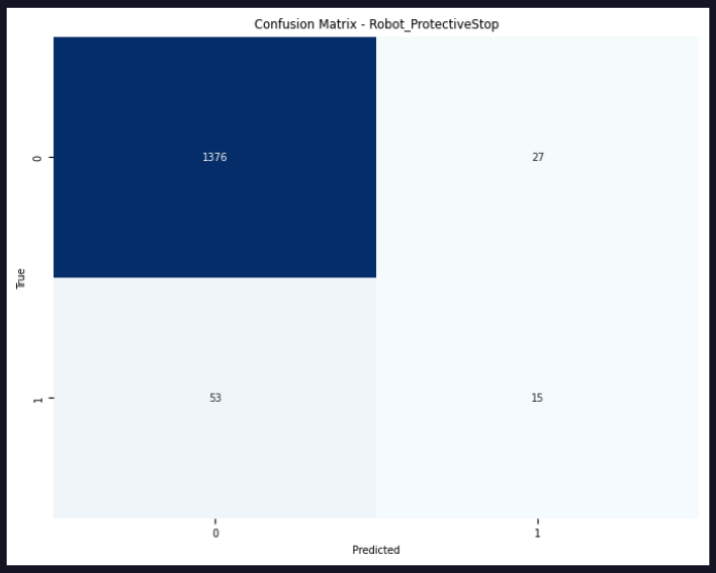

```
AUC-ROC: 0.7533
```

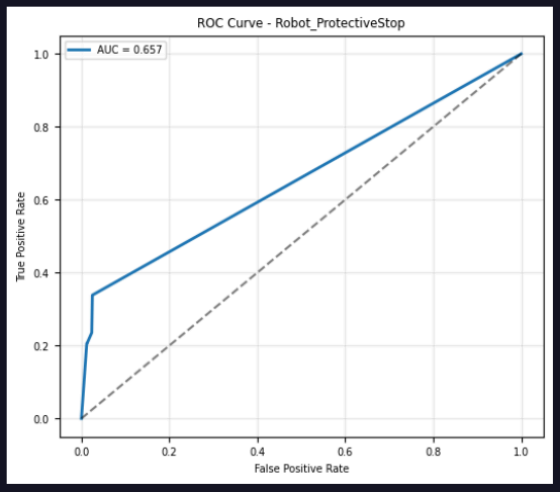
Hold out validation untuk Robot_ProtectiveStop

```
============================================================
Hold-Out Validation for Robot_ProtectiveStop:
============================================================
Training time: 12.437869548797607 seconds
Prediction time: 0.0025010108947753906 seconds
              precision    recall  f1-score   support

           0       0.96      0.98      0.97      1403
           1       0.36      0.22      0.27        68

    accuracy                           0.95      1471
   macro avg       0.66      0.60      0.62      1471
weighted avg       0.93      0.95      0.94      1471
```






```
AUC-ROC: 0.6566
```

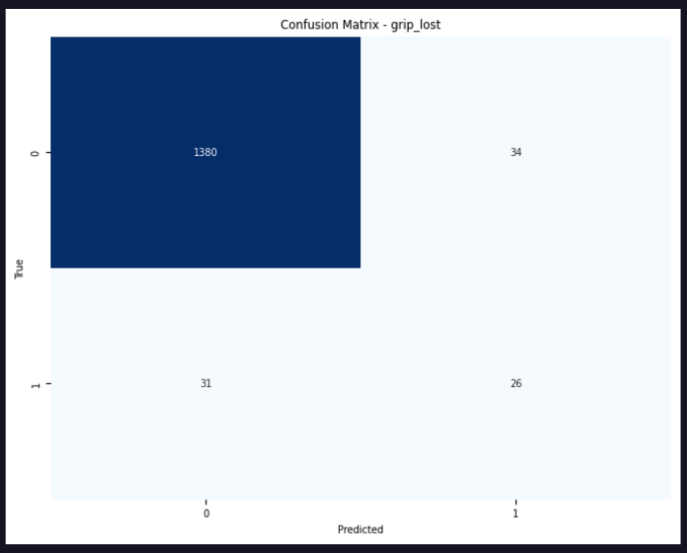
K-fold cross validation untuk grip_lost	``` == Cross Validation for grip_lost: == Model Performance (5-Fold Cross Validation): fit_time: [13.86461473 11.76693964 12.76786256 17.00507283 12.92125082] score_time: [0.01351595 0.00800014 0.01052403 0.00866127 0.01003408] test_precision: [0.61904762 0.43137255 0.4 0.50847458 0.44680851] Average test_precision: 0.48 test_recall: [0.45614035 0.39285714 0.36363636 0.71428571 0.47727273] Average test_recall: 0.48 test_f1: [0.52525253 0.41121495 0.38095238 0.59405941 0.46153846] Average test_f1: 0.47 ```
K-fold cross validation untuk Robot_ProtectiveStop	``` == Cross Validation for Robot_ProtectiveStop: == Model Performance (5-Fold Cross Validation): fit_time: [13.88592958 15.45658708 18.09196377 19.93619633 20.57934141] score_time: [0.01250935 0.01501632 0.0169735 0.01051688 0.01301575] test_precision: [0.51219512 0.52 0.51666667 0.43478261 0.575] Average test_precision: 0.51 test_recall: [0.30882353 0.49056604 0.51666667 0.42553191 0.46] Average test_recall: 0.44 test_f1: [0.3853211 0.50485437 0.51666667 0.43010753 0.51111111] Average test_f1: 0.47 ```
Model Scikit-Learn	

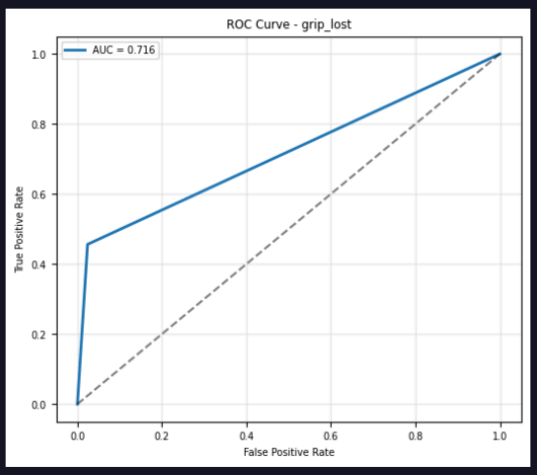
Hold-out validation untuk grip_lost

```
============================================================
Hold-Out Validation for grip_lost:
============================================================
Training time: 0.13266730308532715 seconds
Prediction time: 0.0012407302856445312 seconds
              precision    recall  f1-score   support

           0       0.98      0.98      0.98      1414
           1       0.43      0.46      0.44        57

    accuracy                           0.96      1471
   macro avg       0.71      0.72      0.71      1471
weighted avg       0.96      0.96      0.96      1471
```





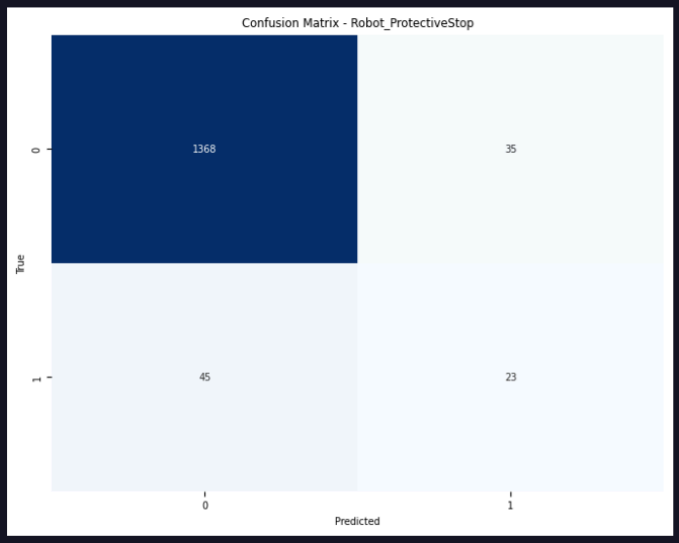

```
AUC-ROC: 0.7160
```

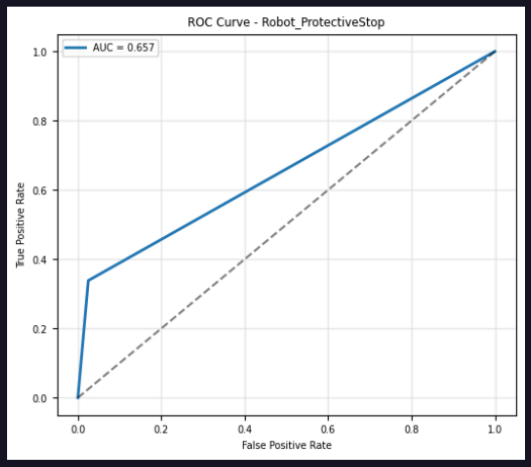
Hold out validation untuk Robot_ProtectiveStop

```
------------------------------------------------------------
Hold-Out Validation for Robot_ProtectiveStop:
============================================================
Training time: 0.12331700325012207 seconds
Prediction time: 0.0009982585906982422 seconds
              precision    recall  f1-score   support

           0       0.97      0.98      0.97      1403
           1       0.40      0.34      0.37        68

    accuracy                           0.95      1471
   macro avg       0.68      0.66      0.67      1471
weighted avg       0.94      0.95      0.94      1471
```






```
AUC-ROC: 0.6566
```

K-fold cross validation untuk grip_lost	``` == Cross Validation for grip_lost: == Model Performance (5-Fold Cross Validation): fit_time: [0.13102388 0.13313317 0.12301588 0.20087028 0.23422837] score_time: [0.00754452 0.00300312 0.00500321 0.01555777 0.00800586] test_precision: [0.44067797 0.47826087 0.32692308 0.52631579 0.46666667] Average test_precision: 0.45 test_recall: [0.45614035 0.39285714 0.38636364 0.71428571 0.47727273] Average test_recall: 0.49 test_f1: [0.44827586 0.43137255 0.35416667 0.60606061 0.47191011] Average test_f1: 0.46 ```
K-fold cross validation untuk Robot_ProtectiveStop	``` == Cross Validation for Robot_ProtectiveStop: == Model Performance (5-Fold Cross Validation): fit_time: [0.18908477 0.24933958 0.23894501 0.32544327 0.38431787] score_time: [0.00954556 0.01104641 0.02404904 0.01104879 0.04819703] test_precision: [0.42372881 0.51851852 0.58333333 0.38095238 0.525] Average test_precision: 0.49 test_recall: [0.36764706 0.52830189 0.58333333 0.34042553 0.42] Average test_recall: 0.45 test_f1: [0.39370079 0.52336449 0.58333333 0.35955056 0.46666667] Average test_f1: 0.47 ```

Improvement