



INNOVATIVE: Journal Of Social Science Research

Volume 4 Nomor 2 Tahun 2024 Page 3347-3357

E-ISSN 2807-4238 and P-ISSN 2807-4246

Website: <https://j-innovative.org/index.php/Innovative>

## Deteksi Awal Kelangsungan Hidup Pasien Gagal Jantung Menggunakan *Machine Learning* Metode *Random Forest*

Boy Fransiskus Sitanggang<sup>1✉</sup>, Pardomuan Sitompul<sup>2</sup>

Universitas Negeri Medan

Email: [boysitanggang416@gmail.com](mailto:boysitanggang416@gmail.com)<sup>1✉</sup>

### Abstrak

Penyakit gagal jantung menjadi penyebab kematian tertinggi di dunia dan kedua di Indonesia. Gagal jantung adalah kondisi dimana jantung tidak mampu memompa darah ke jaringan untuk memenuhi kebutuhan metabolisme tubuh. Perkembangan teknologi telah memberikan kontribusi dalam mengatasi permasalahan di berbagai bidang terutama pada bidang kesehatan, salah satunya dalam pengklasifikasian dan deteksi penyakit menggunakan *machine learning*. *Random Forest* merupakan metode pembelajaran mesin (*machine learning*) yang terkenal dan dapat digunakan untuk mengembangkan model prediksi. Penelitian ini bertujuan untuk mengetahui kinerja *Random Forest* dan hasil peningkatan kinerja menggunakan *hyperparameter tuning* dalam mendeteksi awal kelangsungan hidup pasien gagal jantung berdasarkan metrik evaluasi model. Data yang digunakan dalam penelitian yaitu data rekam medis 299 pasien yang mengalami gagal jantung diperoleh dari situs resmi UCI *Machine Learning Repository*. Berdasarkan metrik evaluasi model, kinerja *Random Forest* sebelum menggunakan *hyperparameter tuning* mendapatkan akurasi 80%, recall 57.89% dan spesifitas 90.24%. Hasil kinerja *Random Forest* menggunakan *hyperparameter tuning* dengan *Random Search* mendapatkan akurasi 83,33%, recall 57.89% dan spesifitas 95.12%. Hasil kinerja *Random Forest* menggunakan *hyperparameter tuning* dengan *Grid Search* mendapatkan akurasi 85%, recall 57.89% dan spesifitas 97.56%. Penggunaan *hyperparameter tuning* dengan *Grid Search* memiliki kinerja yang terbaik sehingga model dapat digunakan dalam mendeteksi awal status keselamatan atau kelangsungan hidup pasien gagal jantung dengan baik.

Kata Kunci: *Gagal Jantung, Machine Learning, Random Forest, Hyperparameter Tuning*

## Abstract

Heart failure is the second leading cause of death in the world and second in Indonesia. Heart failure is a condition where the heart is unable to pump blood to the tissues to fulfil the body's metabolic needs. The development of technology has contributed to overcoming problems in various fields, especially in the health sector, one of which is in the classification and detection of diseases using machine learning. Random Forest is a well-known machine learning method that can be used to develop prediction models. This study aims to determine the performance of Random Forest and the results of performance improvement using hyperparameter tuning in detecting early survival of heart failure patients based on model evaluation metrics. The data used in the study are medical records of 299 patients with heart failure obtained from the official website of the UCI Machine Learning Repository. Based on model evaluation metrics, Random Forest performance before using hyperparameter tuning gets accuracy 80%, recall 57.89% and specificity 90.24%. Random Forest performance results using hyperparameter tuning with Random Search get accuracy 83.33%, recall 57.89% and specificity 95.12%. Random Forest performance results using hyperparameter tuning with Grid Search get accuracy 85%, recall 57.89% and specificity 97.56%. The use of hyperparameter tuning with Grid Search has the best performance so that the model can be used to detect early safety status or survival of heart failure patients well.

*Keywords: Heart Failure, Machine Learning, Random Forest, Hyperparameter Tuning*

## PENDAHULUAN

Jantung adalah salah satu organ penting yang mempunyai fungsi untuk memompa darah ke seluruh tubuh. Apabila jantung mengalami kesulitan dalam memompa darah yang mengandung oksigen sesuai dengan kebutuhan tubuh, maka dapat meningkatkan kemungkinan terjadinya suatu kondisi medis yang disebut gagal jantung. Pada umumnya, gagal jantung dapat ditandai dengan beberapa gejala seperti kesulitan bernapas saat melakukan aktivitas fisik atau beristirahat, mudah lelah, dan mudah berkeringat di area perut, pergelangan tangan atau kaki. Beberapa faktor yang biasanya menyebabkan penyakit gagal jantung adalah diabetes, tekanan darah tinggi, anemia, pola hidup yang kurang sehat dan minimnya aktifitas fisik.

Menurut WHO (2020) [1] penyakit jantung telah menjadi penyebab kematian tertinggi di dunia yang menyebabkan 16% dari total kematian di dunia. Sejak tahun 2000, menunjukkan peningkatan kematian dari 2 juta menjadi 8,9 juta kematian pada tahun 2019. Penyakit gagal jantung menjadi penyebab kematian tertinggi kedua di Indonesia setelah penyakit stroke. Berdasarkan diagnosis dokter yang dilaporkan oleh Riset Kesehatan Dasar Kementerian Kesehatan Indonesia pada tahun 2018, tercatat sekitar 1,5% atau sekitar 1.017.290 orang menderita gagal jantung di Indonesia. Penyakit gagal jantung menunjukkan peningkatan pasien dibandingkan dengan tahun 2013 yang hanya sekitar 0.5% atau sekitar

883.447 orang [2]. Hal ini yang menjadi masalah serius dikarenakan pentingnya organ vital jantung dan tingginya tingkat kematian yang diakibatkan penyakit gagal jantung. Dengan tingkat kematian yang tinggi, maka diperlukan perhatian yang lebih untuk mendeteksi awal pasien gagal jantung.

Perkembangan teknologi telah memberikan kontribusi dalam mengatasi permasalahan di berbagai bidang terutama pada bidang kesehatan, salah satunya dalam pengklasifikasian dan deteksi penyakit menggunakan metode *machine learning*. *Machine learning* atau pembelajaran mesin merupakan suatu cabang dari kecerdasan buatan yang mempelajari cara komputer belajar dari data dengan tujuan meningkatkan kecerdasannya [3]. Beberapa metode yang umum digunakan dalam *machine learning* adalah *Artificial Neural Network*, *K-Nearest Neighbors*, *Decision Tree*, *Random Forest*, *Naive Bayes*, dan *Support Vector Machine*. Dalam mendeteksi awal kelangsungan hidup pasien gagal jantung, penelitian ini akan menggunakan metode *Random Forest*. Hal ini dikarenakan metode *Random Forest* dapat memberikan tingkat akurasi yang tinggi dan dapat menangani data yang besar dan kompleks. Metode *Random Forest* adalah metode *ensemble* yang menggabungkan hasil dari beberapa pohon keputusan yang dapat meningkatkan akurasi dan mengurangi masalah *overfitting* serta dapat mengidentifikasi fitur yang paling penting untuk digunakan pada data latih [4].

Dalam membangun model *Random Forest*, tantangan yang sering muncul adalah menentukan parameter yang optimal untuk meningkatkan kinerja model. Menentukan kombinasi parameter yang tepat secara manual bisa menjadi rumit dan menghabiskan waktu. Maka, pilihan tepat yang dapat dilakukan untuk mendapatkan kombinasi parameter terbaik yaitu *hyperparameter tuning*. Dalam proses *hyperparameter tuning*, terdapat dua metode yang umum digunakan adalah *Grid Search* dan *Random Search*. Metode tersebut bertujuan untuk mencari kombinasi *hyperparameter* yang memberikan kinerja model yang optimal, tetapi keduanya mengadopsi pendekatan yang berbeda dalam mengeksplorasi ruang *hyperparameter*.

Penelitian terdahulu mengenai kelangsungan hidup pasien penyakit gagal jantung telah dilakukan oleh Ahmad,dkk [5] yang menunjukkan bahwa dengan faktor-faktor seperti usia, disfungsi ginjal, tekanan darah tinggi, fraksi ejeksi rendah, dan anemia secara signifikan meningkatkan risiko kematian pada pasien gagal jantung. Penelitian serupa mengenai kelangsungan hidup pasien gagal jantung telah dilakukan oleh Prahasti dan Fauzi [6] yang terdapat 3 variabel yang mempengaruhi risiko kematian pada gagal jantung bahkan setelah mempertimbangkan variabel lain, yaitu status anemia, denyut jantung dan status trombositopenia.

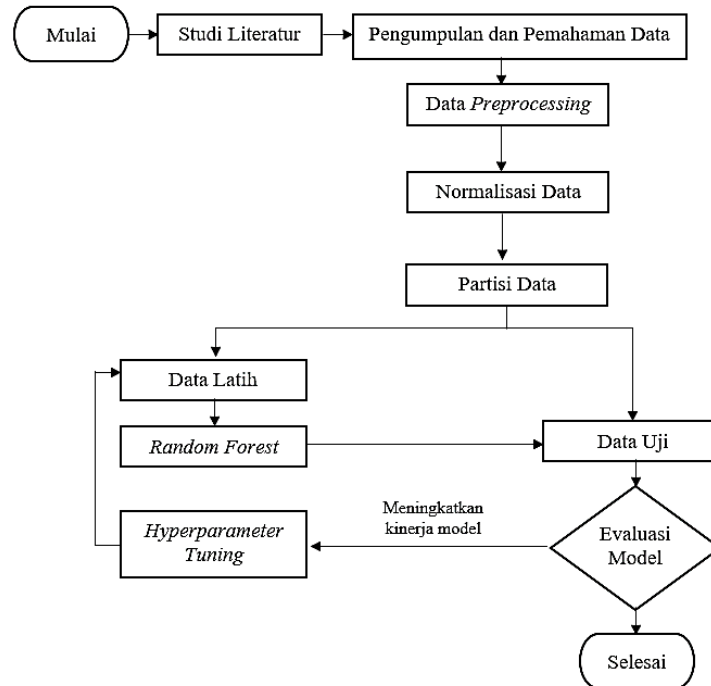
Penelitian terdahulu selanjutnya dilakukan dengan membandingkan algoritma *Random Forest* dengan algoritma *Artificial Neural Network* dalam diagnosa penyakit yang telah dilakukan oleh Eky Cahya Putra Witjaksana, dkk [7]. Hasil dari penelitian ini adalah algoritma *Random Forest* memiliki akurasi 90.62% sedangkan algoritma *Artificial Neural Network* memiliki akurasi 82.29%.

Berdasarkan uraian diatas, maka penulis melakukan penelitian dengan tujuan untuk mengetahui kinerja *Random Forest* dan hasil peningkatan kinerja menggunakan *hyperparameter tuning* dalam mendeteksi awal kelangsungan hidup pasien gagal jantung berdasarkan metrik evaluasi model.

## METODE PENELITIAN

Tempat penelitian dilaksanakan di *Digital Library* Universitas Negeri Medan dan sumber data yang digunakan dalam penelitian diperoleh dari UCI *Machine Learning Repository* tahun 2020 berjudul "*Heart Failure Clinical Records Dataset*". Data yang diperoleh merupakan data sekunder yang berupa data rekam medis 299 pasien yang mengalami gagal jantung dan dikumpulkan selama masa tindak lanjut dimana dataset memiliki 12 variabel prediktor dan 1 variabel target.

Diagram alir dari prosedur penelitian yang akan dilakukan:



Gambar 1. Diagram Alir Prosedur Penelitian

Gambar 1 menjelaskan bahwa tahapan pertama peneliti mencari, memahami, dan mengumpulkan informasi dari penelitian serupa untuk mendapatkan teori pendukung.

Pengumpulan dan pemahaman data menggunakan dataset dari *UCI Machine Learning* dan selanjutnya melakukan *data preprocessing* seperti pembersihan dan normalisasi data. Setelah itu dataset akan dipartisi menjadi dua bagian yaitu data latih dan data uji, kemudian diproses menggunakan metode *Random Forest*. Tahapan berikutnya adalah peningkatan kinerja menggunakan *hyperparameter tuning* dengan *Random Search* dan *Grid Search*. Proses kinerja model dievaluasi menggunakan *confusion matrix* untuk mengetahui nilai akurasi, recall, dan spesitifitas. Proses selesai saat hasil kinerja dengan metode *Random Forest* dengan penggunaan *hyperparameter tuning* mendapatkan performa yang optimal dalam mendeteksi awal kelangsungan hidup pasien gagal jantung.

## HASIL DAN PEMBAHASAN

Kumpulan data yang digunakan dalam penelitian diperoleh dari *UCI Machine Learning Repository* tahun 2020 berjudul "*Heart Failure Clinical Records Dataset*". Data yang diperoleh merupakan data sekunder yang berupa data rekam medis 299 pasien yang mengalami gagal jantung dan dikumpulkan selama masa tindak lanjut dimana dataset memiliki 12 variabel prediktor dan 1 variabel target. Adapun variabel penelitian yang digunakan dideskripsikan pada Tabel 1 berikut.

Tabel 1. Variabel Penelitian

No	Variabel	Deskripsi	Tipe Data	Contoh Data
1	<i>Age</i>	Umur dari pasien	<i>Integer</i>	40 (tahun)
2	<i>Anaemia</i>	Penurunan sel darah merah atau hemoglobin	<i>Boolean</i>	1 = Ya 0 = Tidak
3	<i>Creatinine Phosphokinase (CPK)</i>	Tingkat enzim CPK dalam darah	<i>Integer</i>	582 (mcg/L)
4	<i>Diabetes</i>	Pasien menderita diabetes atau tidak	<i>Boolean</i>	1 = Ya 0 = Tidak
5	<i>Ejection Fraction</i>	Persentase darah yang meninggalkan jantung pada setiap kontraksi	<i>Integer</i>	10 (%)
6	<i>High Blood Pressure</i>	Pasien menderita hipertensi atau tidak	<i>Boolean</i>	1 = Ya 0 = Tidak
7	<i>Platelets</i>	Jumlah trombosit dalam tubuh	<i>Float</i>	263358.03 (kiloplatelets/mL)
8	<i>Serum Creatinine</i>	Jumlah kreatin serum yang terdapat pada darah	<i>Float</i>	1.9 (mg/dL)

9	<i>Serum Sodium</i>	Jumlah natrium serum yang terdapat pada darah	<i>Integer</i>	120 (mEq/L)
10	<i>Sex</i>	Jenis kelamin pasien	<i>Boolean</i>	1 = Pria 0 = Wanita
11	<i>Smoking</i>	Pasien merokok atau tidak	<i>Boolean</i>	1 = Ya 0 = Tidak
12	<i>Time</i>	Waktu tindak lanjut	<i>Integer</i>	5 (hari)
13	<i>Death Event</i>	Pasien telah meninggal atau tidak selama waktu tindak lanjut	<i>Boolean</i>	1 = Tidak Selamat 0 = Selamat

Data yang telah terkumpul kemudian dilakukan tahap preprocessing untuk mengubah data tersebut menjadi data yang lebih normal dan tidak memiliki penyimpangan yang besar, dalam artian semua data akan memiliki skala dan rentang nilai yang sama. *Data preprocessing* pada penelitian menggunakan MinMaxScaler dapat dilihat pada Gambar 2.

```
#Data setelah normalisasi dengan MinMaxScaler
from sklearn.preprocessing import MinMaxScaler

col = data.drop('DEATH_EVENT',axis=1).select_dtypes(exclude='uint8').columns

#standarisasi
MinMax = MinMaxScaler()
data.loc[:, col] = MinMax.fit_transform(data.loc[:, col])
data.head() #cek 5 baris data teratas setelah normalisasi
```

age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking	time
0.636364	0.0	0.071319	0.0	0.090909	1.0	0.290823	0.157303	0.485714	1.0	0.0	0.000000
0.272727	0.0	1.000000	0.0	0.363636	0.0	0.288833	0.067416	0.657143	1.0	0.0	0.007117
0.454545	0.0	0.015693	0.0	0.090909	0.0	0.165960	0.089888	0.457143	1.0	1.0	0.010676
0.181818	1.0	0.011227	0.0	0.090909	0.0	0.224148	0.157303	0.685714	1.0	0.0	0.010676
0.454545	1.0	0.017479	1.0	0.090909	0.0	0.365984	0.247191	0.085714	0.0	0.0	0.014235

Gambar 2. Hasil *data preprocessing*

Pada Gambar 2. data telah dilakukan normalisasi menggunakan MinMaxScaler yang mengubah skala data dari rentang asli, sehingga seluruh nilai berada dalam rentang antara 0 dan 1.

Data dibagi menjadi dua bagian, yaitu data latih dengan proporsi 80% dan data uji dengan proporsi 20%. 239 data latih akan digunakan untuk pemodelan dan 60 data uji akan digunakan untuk menghitung pengukuran akurasi klasifikasi. Pembagian data dilakukan secara acak dengan stratifikasi pada variabel terikat (Y). Pembagian dataset dilakukan menggunakan *software Python* dengan *syntax* sebagai berikut:

```
#SPLIT/PARTISI DATA TRAIN DAN DATA TEST
x = data.iloc[:, :-1]
y = data.iloc[:, -1].values
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 42, stratify=y)
```

Data latih yang telah dipartisi akan digunakan untuk mengenali pola menggunakan metode *Random Forest*, kemudian akan dilakukan pengujian pada data uji. Hasil evaluasi kinerja *Random Forest* menggunakan *confusion matrix* dapat dilihat pada Tabel 2.

Tabel 2. *Confusion matrix* sebelum *Hyperparameter Tuning*

<i>Classification</i>		<i>Predict Class</i>	
		<i>Negatif</i>	<i>Positif</i>
<i>Actual Class</i>	<i>Negatif</i>	37	4
	<i>Positif</i>	8	11

Ukuran-ukuran ketepatan klasifikasi dapat dihitung dengan perhitungan sebagai berikut:

$$TP = 11$$

$$TN = 37$$

$$FP = 4$$

$$FN = 8$$

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FN+FP} \times 100\% = \frac{11+37}{11+37+8+4} \times 100\% = 80\%$$

$$\text{Recall} = \frac{TP}{TP+FN} \times 100\% = \frac{11}{11+8} \times 100\% = 57.89\%$$

$$\text{Spesitifitas} = \frac{TN}{TN+FP} \times 100\% = \frac{37}{37+4} \times 100\% = 90.24\%$$

Hasil kinerja *Random Forest* nilai akurasi 80% yang menunjukkan bahwa terdapat 80% pasien dapat memprediksi data secara benar positif (meninggal) dan negatif (selamat). Nilai recall 57.89% yang menunjukkan bahwa terdapat 57.89% pasien dengan kelas asli positif (meninggal) yang diprediksi secara benar untuk didiagnosis positif (meninggal). Nilai spesitifitas 90.24% menunjukkan bahwa terdapat 90.24% pasien dengan kelas asli negatif (selamat) yang diprediksi secara benar untuk didiagnosis negatif (selamat).

Setelah menggunakan metode *Random Forest* selanjutnya akan dilakukan penyetulan hyperparameter (*hyperparameter tuning*) dengan pencarian acak (*random search*) untuk mengatasi *overfitting* dan mengoptimalkan kinerja *Random Forest*. Hyperparameter yang akan disetel (dituning) menggunakan pencarian acak (*random search*) sebagai berikut:

```

from sklearn.model_selection import RandomizedSearchCV
parameter = {
    'n_estimators': range(1,1000),
    'criterion': ['gini', 'entropy'],
    'max_features': ['auto', 'sqrt', 'log2'],
    'max_depth': range(1,100),
    'min_samples_leaf': range(0,10),
    'min_samples_split': range(0,10),
}
rs_cv = RandomizedSearchCV(estimator=RandomForestClassifier(), param_distributions=parameter,
                           cv=cv, n_jobs = -1, scoring='accuracy')

```

Parameter terbaik yang diperoleh dengan pencarian acak (*random search*) yaitu  $n\_estimator = 233$ ,  $min\_samples\_split = 2$ ,  $min\_samples\_leaf = 5$ ,  $max\_features = auto$ ,  $max\_depth = 37$ ,  $criterion = entropy$ . Hasil evaluasi kinerja *Random Forest* setelah dilakukan penyetelan hyperparameter (*hyperparameter tuning*) dengan pencarian acak (*random search*) menggunakan *confusin matrix* dapat dilihat pada Tabel 3.

Tabel 3. *Confusion matrix* dengan *Random Search*

<i>Classification</i>		<i>Predict Class</i>	
		<i>Negatif</i>	<i>Positif</i>
<i>Actual Class</i>	<i>Negatif</i>	39	2
	<i>Positif</i>	8	11

Ukuran-ukuran ketepatan klasifikasi dapat dihitung dengan perhitungan sebagai berikut:

$$TP = 11$$

$$TN = 39$$

$$FP = 2$$

$$FN = 8$$

$$Akurasi = \frac{TP+TN}{TP+TN+FN+FP} \times 100\% = \frac{11+39}{11+39+8+2} \times 100\% = 83.33\%$$

$$Recall = \frac{TP}{TP+FN} \times 100\% = \frac{11}{11+8} \times 100\% = 57.89\%$$

$$Spesitifitas = \frac{TN}{TN+FP} \times 100\% = \frac{39}{39+2} \times 100\% = 95.12\%$$

Hasil kinerja *Random Forest* setelah dilakukan penyetelan hyperparameter dengan *random search* didapatkan nilai akurasi 83.33% yang menunjukkan bahwa terdapat 83.33% pasien dapat memprediksi data secara benar positif (meninggal) dan negatif (selamat). Nilai recall 57.89% yang menunjukkan bahwa terdapat 57.89% pasien dengan kelas asli positif (meninggal) yang diprediksi secara benar untuk didiagnosis positif (meninggal). Nilai spesitifitas 95.12% menunjukkan bahwa terdapat 95.12% pasien dengan kelas asli negatif (selamat) yang diprediksi secara benar untuk didiagnosis negatif (selamat).

Selanjutnya akan dilakukan penyetelan hyperparameter (*hyperparameter tuning*) dengan pencarian grid (*grid search*). Hyperparameter yang akan disetel (dituning) menggunakan pencarian grid (*grid search*) sebagai berikut:

```

from sklearn.model_selection import GridSearchCV

parameter = {
    'n_estimators':[1,50,5],
    'criterion': ['gini', 'entropy'],
    'max_features':['sqrt','log2'],
    'max_depth': [10,50,5],
    'min_samples_leaf': range(1,8),
    'min_samples_split': range(1,8),
}

gs_cv = GridSearchCV(estimator=RandomForestClassifier(), param_grid=parameter, cv=cv, n_jobs = -1, scoring='accuracy')

```

Parameter terbaik yang diperoleh dengan pencarian grid (*grid search*) yaitu  $n\_estimator = 50$ ,  $min\_samples\_split = 7$ ,  $min\_samples\_leaf = 7$ ,  $max\_features = \log_2$ ,  $max\_depth = 10$ ,  $criterion = entropy$ . Hasil evaluasi kinerja *Random Forest* setelah dilakukan penyetelan hyperparameter (*hyperparameter tuning*) dengan pencarian grid (*grid search*) menggunakan *confusin matrix* dapat dilihat pada Tabel 4.

Tabel 4. *Confusion matrix* dengan *Grid Search*

<i>Classification</i>		<i>Predict Class</i>	
		<i>Negatif</i>	<i>Positif</i>
<i>Actual Class</i>	<i>Negatif</i>	40	1
	<i>Positif</i>	8	11

Ukuran-ukuran ketepatan klasifikasi dapat dihitung dengan perhitungan sebagai berikut:

$$TP = 11$$

$$TN = 40$$

$$FP = 1$$

$$FN = 8$$

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FN+FP} \times 100\% = \frac{11+40}{11+40+8+1} \times 100\% = 85\%$$

$$\text{Recall} = \frac{TP}{TP+FN} \times 100\% = \frac{11}{11+8} \times 100\% = 57.89\%$$

$$\text{Spesitifitas} = \frac{TN}{TN+FP} \times 100\% = \frac{40}{40+1} \times 100\% = 97.56\%$$

Hasil kinerja *Random Forest* setelah dilakukan penyetelan hyperparameter dengan *random search* didapatkan nilai akurasi 85% yang menunjukkan bahwa terdapat 85% pasien dapat memprediksi data secara benar positif (meninggal) dan negatif (selamat). Nilai recall 57.89% yang menunjukkan bahwa terdapat 57.89% pasien dengan kelas asli positif (meninggal) yang diprediksi secara benar untuk didiagnosis positif (meninggal). Nilai

spesifitas 97.56% menunjukkan bahwa terdapat 97.56% pasien dengan kelas asli negatif (selamat) yang diprediksi secara benar untuk didiagnosis negatif (selamat).

Perbandingan kinerja *Random Forest* dalam mendeteksi awal kelangsungan hidup pasien gagal jantung sebelum dan sesudah menggunakan *hyperparameter tuning* berdasarkan metrik evaluasi model dapat dilihat pada Tabel 5 berikut.

Tabel 5. Perbandingan Kinerja Metode *Random Forest*

Metode	Akurasi	Recall	Spesifitas
<i>Random Forest</i>	80 %	57.89%	90.24%
<i>Random Forest</i> Dengan <i>Random Search</i>	83.33 %	57.89%	95.12%
<i>Random Forest</i> Dengan <i>Grid Search</i>	85 %	57.89%	97.56%

Berdasarkan Tabel 5 dapat dilihat bahwa terdapat perbedaan antara setiap nilai sebelum dan sesudah *hyperparameter tuning* dengan *Random Search* dan *Grid Search*. Hasil kinerja *Random Forest* setelah menggunakan *hyperparameter tuning* dengan *Random Search* maupun *Grid Search* berhasil mengoptimalkan kinerja, menghindari *overfitting*, meningkatkan efisiensi pelatihan. Pada penelitian ini penggunaan *hyperparameter tuning* dengan *Grid Search* memiliki kinerja yang terbaik sehingga model dapat digunakan dalam mendeteksi awal status keselamatan atau kelangsungan hidup pasien gagal jantung dengan baik.

## SIMPULAN

Berdasarkan hasil dan pembahasan pada bab sebelumnya, diperoleh kesimpulan sebagai berikut:

1. Kinerja *Random Forest* mendapatkan akurasi 80%, recall 57.89% dan spesifitas 90.24%. Berdasarkan metrik evaluasi tersebut, dapat disimpulkan bahwa model memiliki nilai yang cukup baik dalam mendeteksi awal kelangsungan hidup pasien gagal jantung.
2. Hasil peningkatan kinerja *Random Forest* menggunakan *hyperparameter tuning* dalam mendeteksi awal kelangsungan hidup pasien gagal jantung berdasarkan metrik evaluasi model adalah sebagai berikut:
  - a. Hasil kinerja *Random Forest* menggunakan *hyperparameter tuning* dengan *Random Search* mendapatkan akurasi 83,33%, recall 57.89% dan spesifitas 95.12%. Parameter terbaik yang diperoleh dengan pencarian acak (*random search*) yaitu  $n\_estimator = 233$ ,  $min\_samples\_split = 2$ ,  $min\_samples\_leaf = 5$ ,  $max\_features = auto$ ,  $max\_depth = 37$ ,  $criterion = entropy$ .

- b. Hasil kinerja *Random Forest* menggunakan *hyperparameter tuning* dengan *Grid Search* mendapatkan akurasi 85%, recall 57.89% dan spesifitas 97.56%. Parameter terbaik yang diperoleh dengan pencarian grid (*grid search*) yaitu  $n\_estimator = 50$ ,  $min\_samples\_split = 7$ ,  $min\_samples\_leaf = 7$ ,  $max\_features = \log_2$ ,  $max\_depth = 10$ ,  $criterion = entropy$ .

Hasil kinerja *Random Forest* setelah menggunakan *hyperparameter tuning* dengan *Random Search* maupun *Grid Search* berhasil meningkatkan, mengoptimalkan kinerja dan menghindari *overfitting*. Pada penelitian ini penggunaan *hyperparameter tuning* dengan *Grid Search* memiliki kinerja yang terbaik sehingga model dapat digunakan dalam mendeteksi awal status keselamatan atau kelangsungan hidup pasien gagal jantung dengan baik.

#### DAFTAR PUSTAKA

- WHO, "The top 10 causes of death." [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death>
- Rokom, "Penyakit Jantung Penyebab Kematian Terbanyak ke-2 di Indonesia." [Online]. Available: <https://sehatnegeriku.kemkes.go.id/baca/rilis-media/20220929/0541166/penyakit-jantung-penyebab-utama-kematian-kemenkes-perkuat-layanan-primer/>
- T. Wahyono, *Fundamental of Python for Machine Learning (Dasar-dasar Pemrograman Python untuk Machine Learning dan Kecerdasan Buatan)*. Yogyakarta: Gava Media, 2018.
- S. Polamuri, "How the Random Forest Algorithm Works In Machine Learning." [Online]. Available: <https://dataaspirant.com/random-forest-algorithm-machine-learning/>
- T. Ahmad, A. Munir, S. H. Bhatti, M. Aftab, and M. A. Raza, "Survival analysis of heart failure patients: A case study," *PLoS One*, vol. 12, no. 7, pp. 1–8, 2017, doi: 10.1371/journal.pone.0181001.
- S. D. Prahasti and L. Fauzi, "Risiko Kematian Pasien Gagal Jantung Kongestif (GJK): Studi Kohort Retrospektif Berbasis Rumah Sakit," *Indones. J. Public Heal. Nutr.*, vol. 1, no. 3, pp. 388–395, 2021, [Online]. Available: <https://journal.unnes.ac.id/sju/index.php/IJPHN/article/view/48101/20503>
- E. C. P. Witjaksana, R. R. Saedudin, and V. P. Widartha, "Perbandingan Akurasi Algoritma Random Forest dan Algoritma Artificial Neural Network untuk Klasifikasi Penyakit Diabetes," *e-Proceeding Eng.*, vol. 8, no. 5, pp. 9765–9772, 2021.