

Teknik Scaling Menggunakan *Robust Scaler* Untuk Mengatasi *Outlier* Data Pada Model Prediksi Serangan Jantung

Scaling Technique Using Robust Scaler to Overcome Data Outliers in the Heart Attack Prediction Model

Wahyu Nugraha¹, Raja Sabaruddin², Sri Murni³

^{1,2,3} Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika

E-mail: ¹wahyu.whn@bsi.ac.id, ²raja.rjd@bsi.ac.id, ³sri.six@bsi.ac.id

Abstrak

Serangan jantung adalah salah satu faktor utama dalam tingginya tingkat angkat penyebab kematian di seluruh dunia dan memerlukan prosedur diagnosa yang canggih sehingga dapat mengakibatkan peningkatan biaya yang signifikan. Memprediksi penyakit jantung merupakan tantangan utama dalam bidang kesehatan karena keterbatasan peralatan diagnosis penyakit ini. Prediksi penyakit jantung yang akurat sangat penting untuk mengobati pasien sebelum serangan jantung terjadi. Prediksi ini dapat dicapai dengan menggunakan model pembelajaran mesin (*machine learning*) yang optimal dengan data layanan kesehatan yang kaya (*datasets*) mengenai penyakit jantung. Namun, Permasalahan yang umumnya dihadapi oleh model prediksi penyakit jantung seperti data yang menyimpang secara ekstrim (*outliers*), data yang hilang, data yang tidak konsisten, dan data yang tercampur baik secara numerik maupun kategorikal. Data yang tidak konsisten menyebabkan kemungkinan kesalahan prediksi dan akan mempengaruhi hasil dari prediksi. Pada penelitian ini kami mencoba mengatasi masalah *outlier* pada *dataset* penyakit jantung menggunakan salah satu metode *feature scaling* yaitu *robust scaler*. Hasil Eksperimen dengan model klasifikasi algoritma *K-Nearest Neighbors* menggunakan metode *scaling robust scaler* memperoleh nilai lebih baik dibandingkan dengan tanpa *robust scaler* dengan nilai *F1 score* sebesar 0.83.

Kata kunci: Prediksi serangan jantung, *Outlier* data, *Robust scaler*, Klasifikasi.

Abstract

Heart disease is one of the main factors in the high rate of death worldwide and requires sophisticated diagnostic procedures that can result in significant increases in costs. Predicting heart disease is a major challenge in the health sector due to limited diagnostic equipment for this disease. Accurate heart disease prediction is essential to treat patients before a heart attack occurs. These predictions can be achieved using optimal machine learning models with rich healthcare data (datasets) on heart disease. However, problems generally faced by heart disease prediction models include data that has extreme deviations (outliers), missing data, inconsistent data, and data that is mixed both numerically and categorically. Inconsistent data causes the possibility of prediction errors and will affect the results of the predictions. In this research, we try to overcome the outlier problem in the heart disease dataset using one of the feature scaling methods, namely the robust scaler. Experimental results with the K-Nearest Neighbors algorithm classification model using the robust scaler scaling method obtained better scores compared to those without the robust scaler with an F1 score of 0.83.

Keywords: Heart attack prediction, Outlier data, Robust scaler, Classification

1. PENDAHULUAN

Penyakit jantung merupakan salah satu penyebab utama tingginya angka kematian di seluruh dunia dan membutuhkan proses diagnosis yang canggih dengan biaya yang cukup mahal [1]. Berdasarkan data dari organisasi kesehatan dunia atau WHO menyatakan bahwa lebih dari 18 juta kematian terjadi setiap tahun di dunia akibat penyakit jantung [2][3]. Prediksi penyakit jantung (*heart disease*) merupakan salah satu masalah kesehatan yang paling penting dan sulit untuk dilakukan karena terbatasnya peralatan untuk melakukan diagnosis [1]. Diagnosis penyakit jantung berdasarkan dari riwayat pasien dan pemeriksaan pasien seperti tekanan darah, gula darah, kolesterol dan berlanjut ke tahap pemeriksaan elektrokardiogram, tes stres olahraga, sinar-X, ekokardiogram, angiografi koroner, tes radionuklida, pemindaian MRI, dan CT scan [4]. Oleh karena itu, model prediksi penyakit jantung dengan tambahan informasi pasien dan riwayat kesehatan pada tahap awal diagnosis dapat menghemat waktu dan biaya serta bermanfaat bagi pelaku medis maupun non medis [5]. Beberapa penelitian menerapkan pendekatan *machine learning* untuk membangun sistem deteksi penyakit jantung, sehingga pengobatan yang tepat dengan didukung oleh model sistem prediksi penyakit jantung dapat mencegah meningkatnya angka kematian dan mengurangi resiko penyakit jantung [1][4].

Penerapan *machine learning* dalam deteksi dini berbagai penyakit semakin banyak digunakan karena mampu membuat kesimpulan yang sejalan dengan informasi baru dengan mendeteksi pola-pola tersembunyi dari sebuah *dataset* [4]. *Machine learning* dapat memberikan kontribusi yang signifikan terhadap dunia medis dengan memberikan diagnosis penyakit yang akurat dan efisien [6]. Karena deteksi dini penyakit jantung dapat menurunkan angka kematian, sehingga salah satu cara efektif dalam mengidentifikasi dan memprediksi penyakit jantung adalah dengan menggunakan *algoritma machine learning* [7]. *Machine learning* dapat membuat model yang dapat digunakan untuk membuat prediksi atau keputusan berdasarkan data hasil observasi [2]. *Machine learning* mampu mengatasi berbagai kerumitan dalam proses mendiagnosis penyakit jantung dengan model prediksi menggunakan beberapa algoritma seperti *support vector machine* (SVM), *k-nearest neighbors algorithm* (k-NN), *artificial neural network* (ANN), *decision tree* (DT), *logistic regression* (LR), *AdaBoost* (AB), *Naive Bayes* (NB), *logika fuzzy* (FL) dan lainnya [8].

Tantangan yang terjadi terhadap penyakit jantung ialah kualitas *dataset* yang mengalami masalah seperti data yang menyimpang secara ekstrim (*outliers*), data yang hilang, data yang tidak konsisten, dan data yang tercampur baik secara numerik maupun kategorikal [1][2]. Data yang tidak konsisten menyebabkan kemungkinan kesalahan prediksi dan akan mempengaruhi hasil dari prediksi. Sebelum menentukan model untuk prediksi seringkali memerlukan banyak langkah pra-pemrosesan. Beberapa langkah prapemrosesan yang penting meliputi pembersihan data, pemangkasan, pemilihan fitur, dan penskalaan (*feature scaling*) [9]. *Feature scaling* penting dalam beberapa algoritma pembelajaran mesin dan teknik analisis data. Beberapa algoritma, seperti *regresi linear*, *regresi logistik*, dan algoritma berbasis jarak seperti *k-nearest neighbors* (k-NN) dan *support vector machines* (SVM), bergantung pada perhitungan jarak antar-fitur [10]. Jika fitur-fitur memiliki skala yang berbeda, fitur dengan skala yang besar akan memiliki pengaruh yang lebih besar terhadap perhitungan jarak. Hal ini dapat mengarah pada hasil yang bias atau tidak akurat.

Robust scaler merupakan salah satu metode *feature scaling* yang digunakan dalam analisis data. Metode ini bertujuan untuk mengubah skala fitur-fitur dalam dataset agar lebih tahan terhadap *outlier* atau nilai ekstrem [11]. *Outlier* dapat mempengaruhi perhitungan statistik seperti rata-rata dan simpangan baku, dan dapat memiliki dampak yang signifikan pada hasil analisis data [12]. Oleh karena itu, dalam beberapa kasus, penting untuk melakukan normalisasi atau standarisasi yang tidak terpengaruh oleh nilai *outlier*. *Robust Scaler* menggunakan pendekatan yang berbeda untuk menangani *outlier* dibandingkan dengan metode *feature scaling* lainnya seperti *min-max scaling* atau standarisasi. Metode ini menggunakan estimasi statistik yang lebih tahan terhadap *outlier*, yaitu median dan kuartil, untuk melakukan *scaling*.

Pada penelitian ini kami mencoba menganalisa dan mencoba mengatasi masalah *outlier*

pada dataset penyakit jantung menggunakan salah satu metode *feature scaling* yaitu *robust scaler*. Penggunaan metode ini diharapkan dapat meningkatkan performa algoritma *machine learning* dan menghindari bias yang disebabkan oleh skala yang berbeda di antara fitur-fitur. Penelitian kami bertujuan untuk membuat model terbaik untuk prediksi jantung dan melakukan analisa apakah *robust scaler* mempengaruhi hasil prediksi menjadi lebih baik. Dalam penelitian ini, kami menggunakan *dataset* yang memiliki masalah *outlier* yang berasal dari kumpulan *kaggle datasets* yang dapat di unduh <https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset>. *Dataset* ini berisi rincian data pasien penyakit jantung dengan 14 fitur. Variabel target adalah variabel *biner* dengan data 0 = *less chance of heart attack* 1 = *more chance of heart attack*. Eksperimen ini menggunakan bahasa pemrograman *python language* dengan beberapa *library*. Dari model prediksi yang telah dibangun selanjutnya akan dilakukan perbandingan hasil menggunakan *Confussion Matrix*. Dari matrik konfusi akan diperoleh nilai *accuracy*, *recall*, *precision* dan *F1 Score*. Hasil Eksperimen memperlihatkan bahwa model prediksi klasifikasi algoritma *K-Nearest Neighbors* menggunakan metode *scaling robust scaler* memperlihatkan rata-rata nilai lebih baik dibandingkan dengan tanpa *robust scaler* dengan nilai *accuracy* sebesar 0.82, *precision* sebesar 0.87, *recall* sebesar 0.80 dan *F1 score* sebesar 0.83.

Penelitian yang dilakukan [13] terhadap penyakit parkinson, penelitiannya berfokus pada standarisasi data yang diberikan dan pemilihan fitur sekuensial untuk menganalisis karakteristik yang dipilih. Penelitiannya menggunakan algoritma klasifikasi *Random Forest*, *Logistic Regression*, *SVM*, *Gaussian Naive Bayes*, dan *KNN*. *Robust scaler* digunakan pada penelitian ini untuk menskala fitur dengan *outlier* yang ekstrim. Hasil penelitiannya memberikan hasil terbaik pada pengklasifikasi *AdaBoost* dengan tingkat akurasi 100%.

Penelitian yang dilakukan [4] untuk prediksi penyakit jantung menggunakan algoritma *Classification and Regression Tree (CART)*. Metode pembelajaran mesin mengekstrak aturan keputusan dalam memperjelas hubungan antara variabel masukan dan keluaran. Selain itu, temuan penelitian ini mengurutkan fitur-fitur yang mempengaruhi penyakit jantung berdasarkan kepentingannya. Saat mempertimbangkan semua parameter kinerja, akurasi prediksi sebesar 87% memvalidasi keandalan model. Di sisi lain, aturan pengambilan keputusan yang dilaporkan dalam penelitian ini dapat menyederhanakan penggunaan tujuan klinis tanpa memerlukan pengetahuan tambahan. Secara keseluruhan, algoritma yang diusulkan dapat mendukung tidak hanya profesional kesehatan tetapi juga pasien.

Penelitian yang dilakukan oleh [14] membuat model algoritma *Generalized Logistic (GL)* yang mengubah skala data secara seragam ke interval yang sesuai dengan mempelajari fungsi logistik umum untuk sesuai dengan fungsi distribusi kumulatif empiris dari data. Algoritma *GL* sederhana namun efektif; secara intrinsik tahan terhadap *outlier*, sehingga sangat cocok untuk model diagnostik/klasifikasi dalam aplikasi klinis/medis di mana jumlah sampel biasanya kecil, ia mengubah skala data secara *non-linear*, yang dapat meningkatkan akurasi potensial. eksperimen pada 16 tugas klasifikasi biner dengan jenis variabel yang berbeda dan mencakup berbagai aplikasi. Kinerja hasil dalam hal area di bawah kurva karakteristik operasi penerima (*AUROC*) dan persentase klasifikasi yang benar menunjukkan bahwa model yang dipelajari menggunakan data yang diubah skala oleh algoritma *GL* memiliki kinerja lebih baik dibandingkan dengan yang menggunakan data yang diubah skala oleh algoritma *Min-max* dan *Z-score*, yang merupakan algoritma perubahan skala data yang paling umum digunakan. Algoritma ini tahan terhadap *outlier*, sehingga tidak diperlukan langkah tambahan untuk membersihkan atau mendeteksi *outlier* dalam pra-pemrosesan data. Hasil empiris juga menunjukkan bahwa model yang dipelajari dari data yang diubah skala oleh algoritma *GL* memiliki akurasi yang lebih tinggi dibandingkan dengan algoritma perubahan skala data yang umum digunakan.

Pada penelitian ini, kami berupaya menganalisis dan mengatasi permasalahan *outlier* pada dataset penyakit jantung dengan menggunakan metode *feature scaling*, yakni *robust scaler*. Penggunaan metode ini diharapkan dapat meningkatkan kinerja algoritma *machine learning* dan mengurangi bias yang timbul akibat perbedaan skala antar fitur. Tujuan penelitian kami adalah menciptakan model prediksi jantung terbaik dan mengevaluasi apakah penggunaan *robust scaler*

berdampak positif pada hasil prediksi.

2. METODE PENELITIAN

2.1 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini adalah data yang terkait penyakit jantung dan juga terkait dengan masalah *outlier* sehingga sesuai dengan tujuan dari penelitian ini. *Dataset* diperoleh dari *kaggle dataset repository* dengan format ekstensi CSV (*a comma separated values*) yang diunduh dari <https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset>. *Dataset* terdiri dari 14 variabel yang berisi rician variabel-variabel *heart attack*. Variabel target adalah variabel biner dengan data 0 = *less chance of heart attack* 1 = *more chance of heart attack*. Variabel prediktor terdiri dari *age, sex, cp, trtbps, chol, fbs, restecg, thalachh, exng, oldpeak, slp, caa* dan *thall*. Tabel 1 merupakan *overview* data primer *heart attack* yang akan digunakan pada penelitian ini.

Tabel 1 Sampel *dataset heart attack analysis prediction*

<i>age</i>	<i>sex</i>	<i>cp</i>	<i>trtbps</i>	<i>chol</i>	<i>fbs</i>	<i>restecg</i>	<i>thalachh</i>	<i>exng</i>	<i>oldpeak</i>	<i>slp</i>	<i>caa</i>	<i>thall</i>
63	1	3	145	233	1	0	150	0	2.3	0	0	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2
41	0	1	130	204	0	0	172	0	1.4	2	0	2
56	1	1	120	236	0	1	178	0	0.8	2	0	2
57	0	0	120	354	0	1	163	1	0.6	2	0	2

2.2 Outlier Data

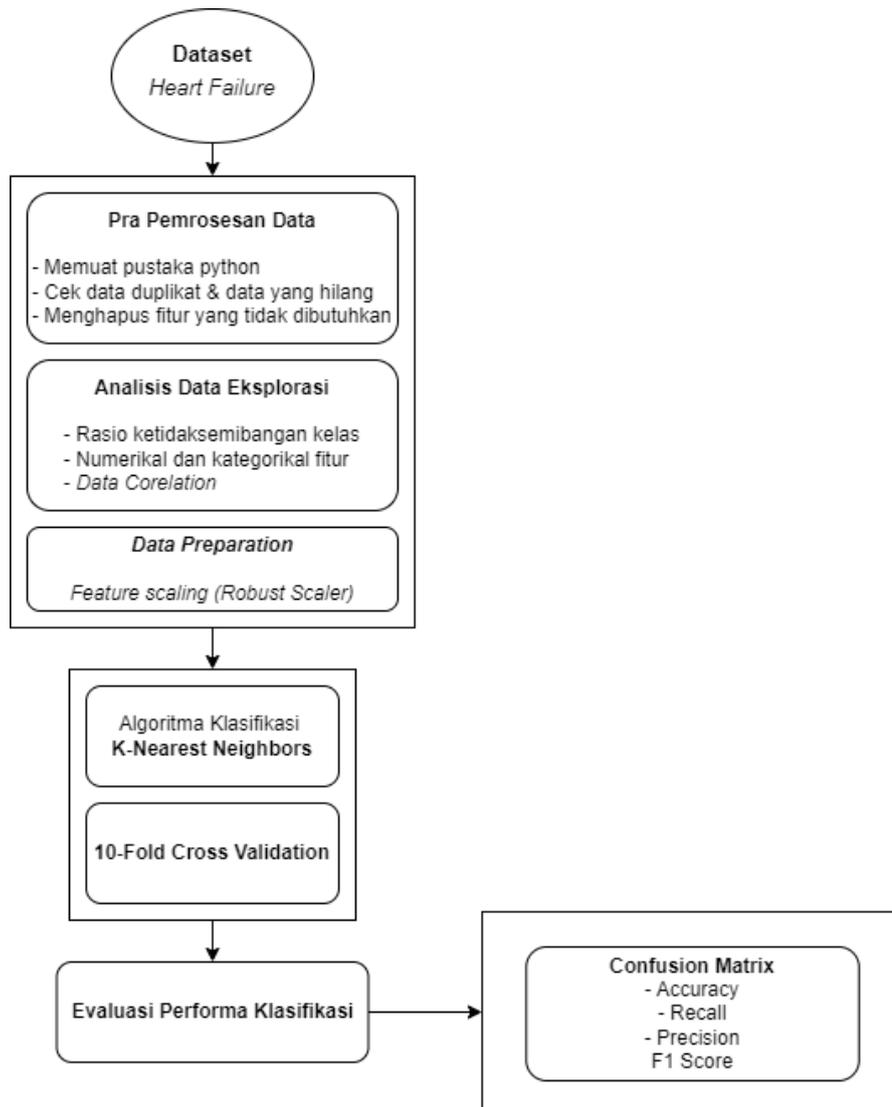
Outlier dalam konteks *data mining* merujuk pada observasi atau titik data yang secara signifikan berbeda dari mayoritas data dalam sebuah *dataset*. *Outlier* sering kali merupakan data yang langka atau tidak biasa, dan mereka bisa menjadi penting karena mereka dapat mengindikasikan kesalahan dalam pengumpulan data, atau mereka bisa menjadi informasi berharga yang mengungkapkan pola atau anomali yang tidak terduga dalam data [15]. Salah satu teknik untuk mengatasi *outlier* pada data adalah menggunakan *robust scaler*. Metode *Robust Scaler* adalah salah satu teknik *pre-processing* dalam analisis data yang digunakan untuk menormalkan data dengan mengurangi pengaruh *outlier*. Rumus dari metode *Robust Scaler* dapat dijelaskan pada persamaan di bawah ini.

$$X_{scaled} = \frac{X - Q1(X)}{Q3(X) - Q1(X)} \quad (1)$$

X_{scaled} adalah nilai yang telah dinormalisasi. X adalah nilai asli pada *dataset*. $Q1(X)$ adalah kuartil pertama dari *dataset* X . $Q3(X)$ adalah kuartil ketiga dari *dataset* X . Persamaan di atas mengurangi setiap nilai pada *dataset* dengan kuartil pertama, kemudian dibagi dengan selisih antara kuartil ketiga dan kuartil pertama. Hal ini dilakukan agar nilai-nilai yang memiliki jarak yang jauh dari kuartil pertama dan kuartil ketiga (*outlier*) tidak memiliki pengaruh yang besar terhadap proses normalisasi.

2.3 Pemodelan

Pada penelitian ini optimasi model *machine learning* menggunakan *feature scaling* yaitu *robust scaler*. Gambar 1 Menunjukkan tahapan eksperimen yang dilakukan yaitu dimulai dari import *dataset*, data *preprocessing*, *Exploratory Data Analysis* (EDA), Data preparation, uji coba menggunakan algoritma klasifikasi K-NN dan langkah akhir dilakukan evaluasi performa model menggunakan *confusion matrix*.



Gambar 1 Model eksperimen *heart attack analysis prediction*

Langkah awal eksperimen yaitu melakukan pemilihan *dataset* yang tepat sesuai tujuan penelitian. *Dataset* penyakit jantung dipilih karena memiliki kasus *outlier* dan ini sesuai dengan kriteria tujuan penelitian ini. Selanjutnya pada tahap preprocessing dilakukan dengan memuat pustaka *python*, mengecek duplikasi data dan menghilangkan fitur yang dianggap tidak dibutuhkan untuk membangun model prediksi. Langkah selanjutnya setelah data siap dilakukan tahapan analisis eksplorasi data yang bertujuan untuk melihat tingkat korelasi hubungan antar variabel, ketidakseimbangan kelas, melihat variabel yang numerikan maupun kategorikal yang bertujuan untuk proses pembuatan model.

Proses membangun model prediksi data dibagi menjadi 70% data *training* dan 30% sebagai data *testing*. Data *training* harus lebih besar dari data *testing* agar pembelajaran mesin menjadi lebih baik. Untuk rasio data *training* dan *testing* dapat mempengaruhi kinerja dari model prediksi [16]. Eksperimen ini akan menggunakan *ratio* 70/30 untuk *training* dan *testing* karena secara umum sebagian besar peneliti menggunakan rasio ini. Proses membangun model algoritma K-NN akan diuji menggunakan *dataset* yang telah melalui proses scaling dan akan dibandingkan dengan *dataset* tanpa proses *scaling*. Tujuannya agar dapat diperoleh perbedaan dan mengetahui

apakah proses *scaling* menggunakan *robust scaler* mampu meningkatkan kinerja dari model prediksi. Hasil pengujian menggunakan metrik evaluasi *confusion matrix* dimana matrik ini sangat umum digunakan pada klasifikasi binari [17].

Confusion matrix berperan penting untuk mengetahui seberapa baik model yang telah dibangun. Dari tabel konfusi akan diperoleh *True Positif (TP)*: Prediksi *class* positif yang diprediksi benar atau *real transaction*. *False Negatif (FN)*: *Prediksi class* negatif, tetapi nilai sebenarnya positif atau prediksi salah (*type II error*). *False Positif (FP)*: *Prediksi class* positif, tetapi nilai sebenarnya negatif atau prediksi salah (*type I error*). *True Negatif (TN)*: *Prediksi class* negatif yang diprediksi benar atau nilai sebenarnya negatif. Hasil pengujian akan diukur menggunakan *confusion matrix* atau matrik konfusi yang dapat dilihat pada Tabel 2 untuk mendapatkan nilai akurasi, *precision*, *recall* dan *F1 Score* yang dinyatakan dalam pernyataan 2, 3, 4, dan 5. *Confusion matrix* memberikan penilaian kinerja model klasifikasi berdasarkan jumlah objek yang diprediksi dengan benar dan salah.

Tabel 2 Model *Confusion Matrix*

Classification	Predicted Class		
	Class = YES	Class = NO	
Observed Class	Class = YES	A (true positive-TP)	B (false negative-FN)
	Class = NO	C (false positive-FP)	D (true negative-TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1\ Score = 2 * \frac{Recall * Precision}{Recall + Precision} \quad (5)$$

2.4 Validation and Evaluation Model

Model validasi untuk pembelajaran dan pengujian menggunakan *10-fold cross-validation* melibatkan pembagian data latih menjadi 10 *subset* yang sama ukurannya. Proses pembelajaran dilakukan sebanyak 10 kali, di mana setiap kali pembelajaran dilakukan pada 9 *subset* dan pengujian dilakukan pada *subset* yang tersisa. Dari Tabel 3, terlihat bahwa dalam setiap iterasi, satu bagian dipilih sebagai data pengujian, sementara sisa data sebanyak 9 bagian digunakan sebagai data pembelajaran. Setelah itu, dihitung nilai rata-rata dan nilai penyimpangan (*deviation value*) dari 10 kali jumlah pengujian yang berbeda [18]. Penelitian ini menggunakan *k-fold cross validation* karena metode ini telah menjadi metode validasi standar dan *state-of-the-art* [19].

Tabel 3 *Stratified 10 Fold Cross Validation*

n-validation	Dataset's Partition									
1	■									
2		■								
3			■							
4				■						
5					■					
6						■				
7							■			
8								■		
9									■	
10										■

3. HASIL DAN PEMBAHASAN

Eksperimen untuk mencapai hasil dari penelitian ini terdiri dari beberapa tahap eksperimen diantaranya *data preprocessing*, *exploratory data analysis* (EDA) dan model *fitting* (pengujian model yang menggunakan *robust scaler* dan tanpa *robust scaler*) dan evaluasi hasil untuk menentukan hasil terbaik dari semua pengujian model. Eksperimen dimulai dari *import* beberapa *library* yang dibutuhkan dalam prapemrosesan data. Eksperimen ini menggunakan bahasa *python* versi 3 dengan *tool google colab* dimana *library* yang digunakan diantaranya *numpy*, *pandas*, *matplotlib*, *seaborn*, *plotly* dan lainnya. Selanjutnya dengan *library pandas dataset* dengan format CSV dimuat. Selanjutnya dilakukan observasi data untuk menemukan duplikasi data serta mengecek apakah ada data yang hilang atau *missing value*.

Pengujian model klasifikasi pada eksperimen ini menggunakan algoritma klasifikasi K-Nn dengan komposisi 70% data *training* dan 30% sebagai data *testing*. Pengujian model klasifikasi K-Nn pada *dataset* dilakukan apa adanya tanpa proses *scaling*. Kemudian akan dibandingkan dengan model yang menggunakan teknik *scaling robust scaler*. Dari seluruh model yang dibangun akan dicari model terbaik berdasarkan hasil eksekusi. Penentuan model terbaik dari dua pengujian maka pengukuran terhadap model algoritma klasifikasi menggunakan metrik pengukuran *confusion matrix*. Dari hasil pengujian yang telah dilakukan diperoleh nilai *confusion matrix* dari masing-masing algoritma klasifikasi yang dapat dilihat pada tabel 4 dan tabel 5. Nilai yang diperoleh dari matriks konfusi adalah akurasi, *recall*, *precision* dan *F1 score*.

Model klasifikasi K-NN tanpa melalui proses *scaling* pada Tabel 4 menunjukkan hasil dari tabel konfusi yang kurang baik dengan nilai *accuracy score*: 0.61, *precision score*: 0.71, *recall score*: 0.67, dan *F1 score*: 0.68.

Tabel 4 *Confusion matrix* tanpa proses *scaling*

	0	80	58
<i>True label</i>	1	47	118
		0	1
		<i>Predicted Label</i>	

Model klasifikasi K-NN dengan proses *scaling* dengan *robust scaler* pada Tabel 5 menunjukkan hasil dari tabel konfusi yang lebih baik atau meningkat dibandingkan model tanpa proses *scaling* dengan nilai *accuracy score*: 0.82, *precision score*: 0.87, *recall score*: 0.80, dan *F1 score*: 0.83.

Tabel 5 *Confusion matrix* dengan *robust scaler*

	0	98	30
<i>True label</i>	1	18	124
		0	1
		<i>Predicted Label</i>	

Berdasarkan tabel 4 dan 5 menggambarkan bahwa dari model yang telah diujikan, model algoritma dengan *robust scaler* memperoleh nilai rata-rata tertinggi berdasarkan *confusion matrix*. Jika dilihat secara rata-rata maka penerapan *robust scaler* cukup memberikan dampak yang signifikan terhadap performa model dengan data yang *outlier*. Peningkatan tersebut terlihat dari nilai *F1 score* yang meningkat cukup signifikan dari 0.68 menjadi 0.83. Untuk hasil keseluruhan dapat dilihat pada tabel 6 di bawah ini.

Tabel 6 Perbandingan Hasil Berdasarkan *Confusion Matrix*

Metode Scaling	Accuracy	Precision	Recall	F1 Score
<i>Original</i>	0.61	0.71	0.67	0.68
<i>Robust Scaler</i>	0.82	0.87	0.80	0.83

4. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang dilakukan menggunakan *dataset Kaggle dataset repository (heart failure prediction)* dapat disimpulkan bahwa model klasifikasi menggunakan K-NN dengan *robust scaler* rata-rata memperoleh nilai tertinggi baik menggunakan pengukuran *accuracy*, *recall*, *precision* dan *F1 score*.

Secara umum, algoritma K-NN menggunakan teknik *scaling robust scaler* bekerja dengan baik bahkan pada data yang mengalami *outlier*. Model dibangun dari algoritma tersebut memberikan hasil yang baik pada data yang *outlier* dengan memberikan mekanisme untuk lebih fokus mengatasi masalah tersebut dan menyesuaikan algoritma pelatihan. Penelitian ini dapat disimpulkan bahwa dengan mengubah data yang mengalami *outlier* kami mengamati perubahan pada nilai *recall*, akurasi, *precision* dan *f1 score* meningkat cukup signifikan.

Peneliti menyarankan untuk lebih menekankan pentingnya untuk mendapatkan pengetahuan yang lebih mendalam mengenai masalah yang dihadapi saat proses pelatihan. Pada kasus ini outlier pada data menjadi salah satu masalah yang sering dihadapi pada model prediksi penyakit jantung. Namun, solusi yang diberikan sebaiknya berdasarkan permintaan pemangku kepentingan sehingga peneliti mampu membuat perubahan sesuai kebutuhan.

Meskipun metode *Robust Scaler* lebih tahan terhadap *outlier* dibandingkan dengan metode *Standard Scaler (z-score scaling)*, metode ini masih dapat dipengaruhi oleh *outlier* yang sangat ekstrem. *Outlier* yang sangat ekstrem masih bisa mempengaruhi perhitungan median dan kuartil, sehingga dapat mempengaruhi hasil *scaling*. Walaupun memiliki kelemahan, metode *Robust Scaler* tetap berguna dalam banyak situasi, terutama ketika data mengandung *outlier* yang perlu ditangani dengan hati-hati. Keputusan untuk menggunakan metode ini atau metode *preprocessing* lainnya harus didasarkan pada karakteristik data dan tujuan analisis atau model yang akan dibangun.

DAFTAR PUSTAKA

- [1] M. M. Ahsan, M. A. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, "Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance," *Technologies (Basel)*, vol. 9, no. 3, pp. 5–9, 2021, doi: 10.3390/technologies9030052.
- [2] F. Ali *et al.*, "A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion," *Information Fusion*, vol. 63, pp. 208–222, 2020, doi: 10.1016/j.inffus.2020.06.008.
- [3] J. S. Soni, U. Ansari, D. Sharma, and S. Soni, "Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction," *International Journal of Computer Applications (0975-8887)*, vol. 17, no. 8, pp. 43–48, 2011, doi: 10.4337/9781848442986.00014.
- [4] M. Ozcan and S. Peker, "A classification and regression tree algorithm for heart disease modeling and prediction," *Healthcare Analytics*, vol. 3, no. December 2022, p. 100130, 2023, doi: 10.1016/j.health.2022.100130.
- [5] W. P. Lord and D. C. Wiggins, *Medical decision support systems*, G. Spekowi. Springer: Berlin/Heidelberg, 2006.
- [6] R. Williams, T. Shongwe, A. N. Hasan, and V. Rameshar, "Heart Disease Prediction using Machine Learning Techniques," *2021 International Conference on Data Analytics for Business and Industry, ICDABI 2021*, no. October, pp. 118–123, 2021, doi: 10.1109/ICDABI53623.2021.9655783.
- [7] P. Ghosh *et al.*, "Efficient prediction of cardiovascular disease using machine learning algorithms with relief and lasso feature selection techniques," *IEEE Access*, vol. 9, pp. 19304–19326, 2021, doi: 10.1109/ACCESS.2021.3053759.
- [8] A. U. Haq, J. P. Li, M. H. Memon, S. Nazir, R. Sun, and I. Garcíá-Magarinõ, "A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms," *Mobile Information Systems*, vol. 2018, 2018, doi: 10.1155/2018/3860146.

- [9] M. S. Maulana, R. Sabarudin, and W. Nugraha, "Prediksi Ketepatan Kelulusan Mahasiswa Diploma dengan Komparasi Algoritma Klasifikasi," *JUSTIN*, vol. 07, no. 03, pp. 202–206, 2019.
- [10] H. Qian, Q. Wen, L. Sun, J. Gu, Q. Niu, and Z. Tang, "RobustScaler: QoS-Aware Autoscaling for Complex Workloads," Apr. 2022, [Online]. Available: <http://arxiv.org/abs/2204.07197>
- [11] K. V. A. Reddy, S. R. Ambati, Y. S. Rithik Reddy, and A. N. Reddy, "AdaBoost for Parkinson's Disease Detection using Robust Scaler and SFS from Acoustic Features," in *2021 Smart Technologies, Communication and Robotics (STCR)*, IEEE, Oct. 2021, pp. 1–6. doi: 10.1109/STCR51658.2021.9588906.
- [12] S. Yulianto and K. H. Hidayatullah, "Analisis kluster untuk pengelompokan kabupaten/kota di provinsi Jawa Tengah berdasarkan indikator kesejahteraan rakyat," *J. Statistika. Univ. Muhammadiyah Semarang*, vol. 2, no. 1, pp. 56–63, May 2014.
- [13] K. V. A. Reddy, S. R. Ambati, Y. S. Rithik Reddy, and A. N. Reddy, "AdaBoost for Parkinson's Disease Detection using Robust Scaler and SFS from Acoustic Features," in *2021 Smart Technologies, Communication and Robotics (STCR)*, 2021, pp. 1–6. doi: 10.1109/STCR51658.2021.9588906.
- [14] X. H. Cao, I. Stojkovic, and Z. Obradovic, "A robust data scaling algorithm to improve classification accuracies in biomedical data," *BMC Bioinformatics*, vol. 17, no. 1, Sep. 2016, doi: 10.1186/s12859-016-1236-x.
- [15] S. Agarwal, "Data Mining: Data Mining Concepts and Techniques," *2013 International Conference on Machine Intelligence and Research Advancement*, pp. 203-207, 2013, doi: 10.1109/ICMIRA.2013.45.
- [16] Q. H. Nguyen *et al.*, "Influence of data splitting on performance of machine learning models in prediction of shear strength of soil," *Math Probl Eng*, vol. 2021, 2021, doi: 10.1155/2021/4832864.
- [17] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognit*, vol. 91, pp. 216–231, 2019, doi: 10.1016/j.patcog.2019.02.023.
- [18] R. S. Wahono, N. S. Herman, and S. Ahmad, "A Comparison Framework of Classification Models for Software Defect Prediction," vol. 20, no. 10, pp. 1945–1950, 2014, doi: 10.1166/asl.2014.5640.
- [19] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining Third Edition*. Elsevier Inc, 2011.