

Prediksi Resiko Kematian Pada Penderita Penyakit Kardiovaskular Menggunakan Metode Ensemble Learning

Ahmadien Hafizh Yusufi¹, Adithya Kharisma², Adrian Dwi Adinata³, Daffy Fayyadhya Ramzy⁴, Mayanda Mega Santoni⁵

Informatika / Fakultas Ilmu Komputer

Universitas Pembangunan Nasional Veteran Jakarta

Jl. RS. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12460, Indonesia

ahmadienhy@upnvj.ac.id¹, adithyak@upnvj.ac.id², adrianda@upnvj.ac.id³, daffyfr@upnvj.ac.id⁴,
megasantoni@upnvj.ac.id⁵

Abstrak. Penyakit kardiovaskular, atau lebih populer di Indonesia sebagai penyakit jantung, merupakan penyakit tidak menular yang menyumbang kematian terbesar di dunia. Penyakit kardiovaskular adalah penyakit yang disebabkan gangguan fungsi jantung dan pembuluh darah. Diperlukannya estimasi prediksi resiko kardiovaskular untuk meningkatkan kesadaran publik terhadap penyakit kardiovaskular. *Machine Learning* merupakan fokus pengembangan sistem yang mampu belajar secara “mandiri” tanpa harus berulang kali diprogram manusia. *Ensemble learning* merupakan salah satu algoritma *supervised learning* yang menggabungkan satu atau lebih dari algoritma single model sehingga memiliki fleksibilitas yang tinggi. Data yang digunakan pada penelitian ini sebanyak 299 data dengan 13 atribut dimana target kolom yang dicari dalam pengklasifikasian berupa resiko kematian. Data diolah menggunakan ipynb dengan platform analisis *google collab*. Sebelum data diolah untuk membangun model klasifikasi menggunakan *logistic regression*, *k-nearest neighbours classifier*, *support vector machine*, *decision tree*, *random forest*, *naïve bayes*, dan *ensemble voting* data diproses terlebih dahulu dengan melakukan pembersihan data, eksplorasi data, normalisasi data dan penyeimbangan data. Model klasifikasi yang diperoleh dievaluasi menggunakan nilai akurasi, *precision* dan *recall*.

Kata Kunci: Penyakit Kardiovaskular, Machine Learning, Ensemble Learning

1 Pendahuluan

Penyakit kardiovaskular, atau lebih populer di Indonesia sebagai penyakit jantung, merupakan penyakit yang disebabkan gangguan fungsi jantung dan pembuluh darah, seperti: Penyakit Jantung Koroner, Penyakit Gagal Jantung, Hipertensi dan Stroke[1]. Pada tahun 2008 dilaporkan terjadi 17,3 juta kematian akibat penyakit kardiovaskular[1]. Pada 2017 dilaporkan sekitar 17.8 juta kematian disebabkan oleh penyakit kardiovaskular di seluruh dunia, angka kematian ini dilaporkan meningkat sebanyak 21.1% dibanding 2007[2]. Lebih dari 3 juta kematian tersebut terjadi sebelum usia 60 tahun dimana 4% di negara berpenghasilan tinggi dan 42% di negara berpenghasilan rendah[1]. Adapun faktor resiko penyakit kardiovaskuler terdiri dari faktor yang tidak dapat dimodifikasi dan yang dapat dimodifikasi. Faktor resiko yang tidak dapat dimodifikasi yaitu riwayat keluarga, umur, jenis kelamin dan obesitas. Sedangkan faktor resiko yang dapat dimodifikasi yaitu hipertensi, diabetes melitus, dislipidemia, kurang aktivitas fisik, diet tidak sehat dan stress[1]. Selain itu menurut *American Health Association* faktor-faktor lain seperti perokok, kurang nutrisi, kolesterol tinggi, tekanan darah tinggi dan metabolisme yang tidak normal juga dapat menyebabkan terjadinya penyakit kardiovaskular[2].

Tingginya angka kematian akibat penyakit kardiovaskular menjadi salah satu perhatian di bidang kesehatan di seluruh dunia. Dengan begitu diperlukan adanya estimasi prediksi resiko kardiovaskular untuk meningkatkan kesadaran publik terhadap penyakit kardiovaskular. Dalam praktek klinis, algoritma prediksi risiko telah digunakan paling langsung untuk mengidentifikasi individu yang berisiko tinggi untuk mengembangkan CVD (*Cardio Vascular Disease*) dalam jangka pendek untuk memilih individu-individu untuk intervensi pencegahan yang lebih intensif[3]. Data rekam medis pasien penyakit kardiovaskular dapat dimanfaatkan untuk membuat

prediksi resiko ini. Data rekam medis tersebut dapat diolah menggunakan machine learning untuk mengklasifikasikan resiko kematian terhadap pasien yang mengidap penyakit kardiovaskular.

Machine Learning atau Mesin Pembelajaran merupakan cabang dari *Artificial Intelligence* yang fokus belajar dari data (*learn from data*), yaitu fokus pengembangan sistem yang mampu belajar secara “mandiri” tanpa harus berulang kali diprogram manusia[4]. Terdapat banyak model yang digunakan dalam *Machine Learning* seperti *Logistic Regression*, *K-Nearest Neighbors*, *Support Vector Machine (SVM)*, *Decision Tree* dan *Naïve Bayes*. Dalam *Machine Learning* juga terdapat istilah *supervised learning* dan *unsupervised learning*. *Supervised Learning* mengasumsikan ketersediaan seorang guru atau supervisor yang mengklasifikasikan contoh-contoh pelatihan ke dalam kelas dan memanfaatkan informasi tentang keanggotaan kelas dari setiap contoh pelatihan. *Unsupervised Learning* mengidentifikasi pola informasi kelas secara heuristik dan pembelajaran Reinforcement belajar melalui interaksi trial and error dengan lingkungannya (*reward/penalty assignment*)[5]. Dalam pemodelan *Machine Learning* terdapat dua jenis yaitu *Single Model* dan *Ensemble Learning*. *Ensemble learning* sendiri merupakan algoritma *supervised learning* karena dapat dilatih dan digunakan untuk membuat prediksi. *Ensemble learning* menggabungkan satu atau lebih dari algoritma *single model* sehingga memiliki fleksibilitas yang tinggi karena memiliki gabungan kegunaan dari *single model* yang dipakai[6].

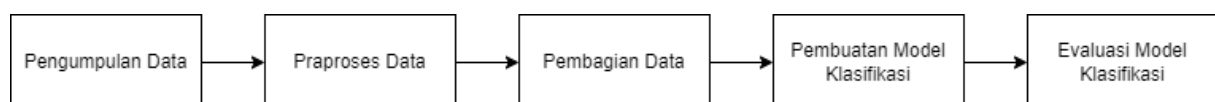
Penggunaan *Machine Learning* dalam memprediksi suatu penyakit sudah banyak dilakukan. Penelitian terkait yaitu Prediksi Hipertensi Menggunakan *Decision Tree*, *Naïve Bayes* dan *Artificial Neural Network* pada Software KNIME oleh Mayanda Mega Santoni[7]. Dengan data yang berjumlah 274 data yang digunakan, untuk *decision tree* memiliki akurasi sebesar 83%, *recall* sebesar 76%, dan *precision* sebesar 87%. Sedangkan dengan *naïve bayes* memiliki akurasi sebesar 74%, *recall* sebesar 68%, dan *precision* sebesar 87%. Lalu untuk *artificial neural network* memiliki akurasi sebesar 94%, *recall* sebesar 91%, dan *precision* sebesar 97%.

Penelitian terkait lainnya yang lebih spesifik kepada penyakit kardiovaskular yang sesuai dengan topik dari penelitian ini yaitu Prediksi Penyakit Jantung Kardiovaskular Menggunakan Model Algoritma Klasifikasi oleh Wahyu Nugraha[8]. Data yang dipakai dalam penelitian terkait ini sama dengan penelitian yang kami lakukan namun algoritma-algoritma yang dipakai berbeda. Hasil akurasi dari algoritma SVM dan XGBoost sebesar 80%, *Gradien Boosting* sebesar 78%, *Random Forest* sebesar 77%, dan Light GBM sebesar 68%.

Dari latar belakang masalah dan penelitian-penelitian terkait yang sudah dipaparkan diatas, maka penelitian ini akan menggunakan *Ensemble Learning* yang menggabungkan algoritma-algoritma *Logistic Regression*, *K-Nearest Neighbours Classifier*, *Support Vector Machine*, *Decision Tree*, *Random Forest* dan *Naïve Bayes* untuk mengklasifikasikan atau memprediksi resiko kematian pada pasien yang memiliki penyakit kardiovaskular.

2 Metode Penelitian

Metodologi dari penelitian ini dimulai dengan pengumpulan data penelitian, selanjutnya dilakukan praproses data untuk mempersiapkan data agar bisa diolah, selanjutnya dilakukan pembagian data, lalu dilanjutkan dengan pembuatan model klasifikasi, dan terakhir melakukan evaluasi model klasifikasi. Metodologi diatas dapat dilihat pada Gambar 1.



Gambar 1. Metodologi Penelitian.

2.1 Pengumpulan Data

Data penelitian diperoleh melalui website Kaggle.com yang disediakan oleh Larxel atau Andrewmvd yang merupakan *senior data scientist* dari Rumah Sakit Israelita Albert Einstein, São Paulo, State of São Paulo, Brazil. Data yang digunakan berupa dataset `heart_failure_clinical_records_dataset.csv`. Deskripsi data penelitian dapat dilihat pada Tabel 1.

Tabel 1. Deskripsi Data Penelitian

No	Nama Variabel	Deskripsi Variabel	Tipe Data
1	age	Umur	Number (Integer)
2	anaemia	Pengurangan sel darah merah atau hemoglobin	0 : Ya, 1 : Tidak (Boolean)
3	creatinine_phosphokinase	Jumlah enzim kreatin kinase dalam darah	Number (Float)
4	diabetes	Jika pasien memiliki diabetes	0 : Ya, 1 : Tidak (Boolean)
5	ejection_fraction	Persentase darah yang keluar dari jantung setiap kontraksi	Percentage (Float)
6	high_blood_pressure	Jika pasien memiliki hipertensi	0 : Ya, 1 : Tidak (Boolean)
7	platelets	Trombosit dalam darah	Kiloplatelets/mL (Float)
8	serum_creatinine	Jumlah serum kreatin dalam darah	mg/dL (Float)
9	serum_sodium	Jumlah serum sodium dalam darah	mEq/L (float)
10	sex	Jenis kelamin	0 : Pria, 1 : Wanita (Boolean)
11	smoking	Jika pasien seorang perokok	0 : Ya, 1 : Tidak (Boolean)
12	time	<i>Follow-up</i> per hari	Number (Integer)
13	DEATH_EVENT	Jika pasien meninggal saat <i>follow-up</i>	0 : Ya, 1 : Tidak (Boolean)

2.2 Praproses Data

Praproses data dalam penelitian ini dilakukan untuk memproses data mentah menjadi data yang siap digunakan. Pada tahap ini dilakukan tiga hal yaitu memuat data, membersihkan data, dan eksplorasi data. Pemuatan data dilakukan dengan mengimpor pustaka dari python lalu memuat dataset ke dalam proyek dengan fungsi `pd.read_csv()`. Setelah data dimuat selanjutnya melihat informasi atau deskripsi seputar dataset seperti melihat isi kolom dengan fungsi `df.columns`, melihat informasi kolom dengan fungsi `df.info()`, melihat deskripsi seputar dataset dengan fungsi `df.describe()`, melihat nilai *uniqu* dari setiap kolom dengan fungsi `df.nunique()`, terakhir melihat jumlah baris, kolom dan elemen dengan fungsi `df.shape[]` dan `df.size`.

Setelah pemuatan data dilakukan maka dilakukan pembersihan data. Pertama melakukan pengecekan terhadap kolom yang memiliki data kosong dengan fungsi `df.isnull().sum()`. Selanjutnya melakukan pengecekan terhadap data yang memiliki duplikasi dengan fungsi `df.duplicated().sum()`. Terakhir, untuk mempermudah penamaan kolom maka dilakukan penggantian nama pada kolom 'anaemia' menjadi 'anemia' dan 'DEATH_EVENT' menjadi 'death'.

Setelah pembersihan data dilakukan maka dilakukan eksplorasi data. Dalam tahap ini eksplorasi data menggunakan query, grouping, visualisasi sederhana, dan lainnya. Pertama memeriksa korelasi antar fitur lalu memvisualisasikannya dengan heatmap. Setelah itu melakukan visualisasi distribusi data menggunakan histogram. Setelah itu melakukan uji kemiringan dengan skewness. Setelah itu melakukan uji normalitas data, melihat nilai unik pada variable target, memeriksa distribusi target, melihat distribusi umur pasien, jenis kelamin pasien, anemia, diabetes, `high_blood_pressure`, smoking, dan time berdasarkan resiko kematian

Sebelum meneruskan ke tahap pembagian data, data perlu dinormalisasikan. Normalisasi data dilakukan untuk merubah data sedemikian rupa sehingga mean tiap variable menjadi sama dengan nol dan standar deviasi tiap variable adalah satu. Dengan begitu akan didapatkan skala yang sama untuk semua variable.

2.3 Pembagian Data

Pada tahap ini dilakukan pembagian data menjadi data latih dan data uji. Namun sebelum dilakukan hal tersebut dibutuhkan penyeimbangan kelas target dengan Teknik *Smote*. Setelah *Smote* dilakukan data telah seimbang dan siap dibagi menjadi data latih dan data uji dengan perbandingan 80% : 20%.

2.4 Pembuatan Model Klasifikasi

Pada penelitian ini terdapat enam model klasifikasi yang akan diimplementasikan. Model klasifikasi tersebut yaitu *Logistic Regression*, *K-Nearest Neighbours Classifier*, *Support Vector Machine*, *Decision Tree*, *Random Forest*, dan *Naïve Bayes*. Data latih akan digunakan untuk pembuatan model klasifikasi dari algoritma-algoritma yang digunakan.

2.4.1 *Logistic Regression*

Logistic regression merupakan jenis analisis statistic yang digunakan untuk memperkirakan sebuah kejadian berdasarkan dataset yang terdiri dari variable independen. Karena *output*-nya probabilitas, variabel dependen akan bernilai 0 dan 1 (ya dan tidak). Rumus untuk *logistic regression* dapat dilihat pada Rumus 1.

$$\ln = \left(\frac{p}{1-p} \right) = B_0 + B_1 X \quad (1)$$

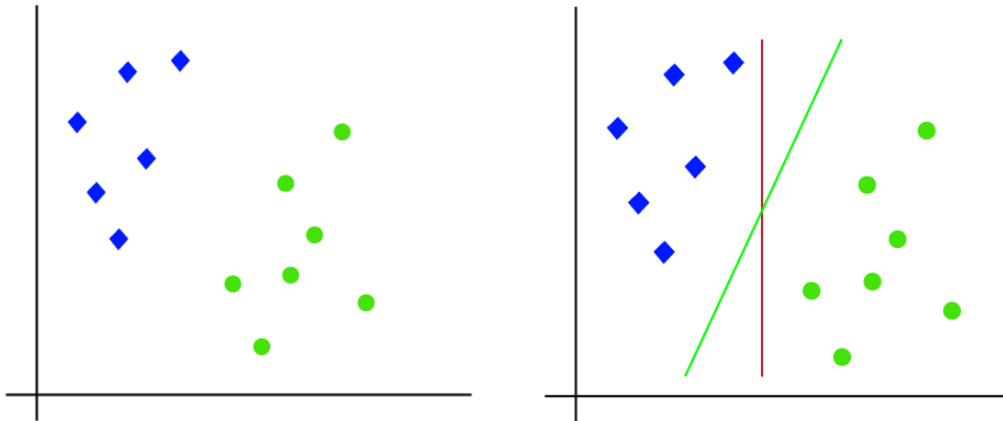
2.4.2 *K-Nearest Neighbours Classifier*

K-Nearest Neighbours atau KNN mengklasifikasikan data berdasarkan kemiripan antara data-data. Jarak antara dua titik pada algoritma KNN dapat dicari dengan metode *Euclidean Distance* dengan jumlah variabel independen 1. *Euclidean Distance* dapat dihitung dengan Rumus 2.

$$dis(x_1, x_2) = \sqrt{\sum_{i=0}^n (x_{1i} - x_{2i})^2} \quad (2)$$

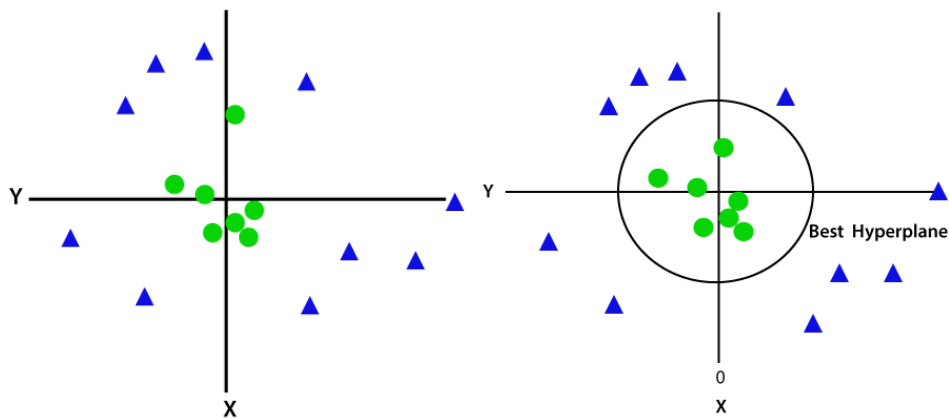
2.4.3 *Support Vector Machine*

Support Vector Machine adalah metode supervised learning yang digunakan untuk klasifikasi dan regresi. Secara matematis, konsep SVM lebih matang dibandingkan teknik klasifikasi lainnya. SVM digunakan untuk mencari hyperplane terbaik dengan memaksimalkan jarak antara kelas-kelas. SVM memiliki 2 jenis, yaitu Linear dan Non-Linear. SVM Linear digunakan untuk data yang dapat dipisahkan secara linear. Jika sebuah dataset dapat diklasifikasikan menjadi 2 kelas dengan 1 garis lurus, SVM Linear dapat digunakan seperti pada Gambar 2.



Gambar 2. SVM Linear

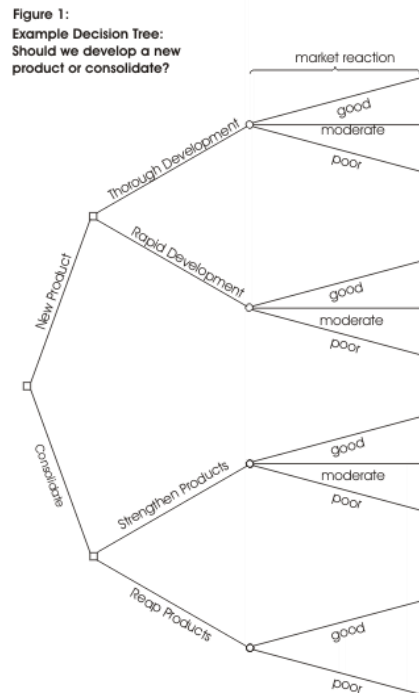
SVM Non-Linear digunakan untuk dataset yang tidak dapat diklasifikasikan dengan garis lurus seperti pada Gambar 3.



Gambar 3. SVM Non-linear

2.4.4 Decision Tree

Decision tree adalah sebuah diagram untuk memvisualisasikan proses pemilihan keputusan dengan menggambarkan berbagai pilihan dan dampak yang mungkin terjadi dari pilihan-pilihan tersebut. Diagram ini dapat membantu melihat dan mempertimbangkan sebanyak mungkin tindakan yang dapat diambil. Secara umum, decision tree mulai dengan 1 node yang akan bercabang ke pilihan-pilihan yang ada. Lalu, cabang-cabang tersebut akan memiliki cabangnya sendiri. Gambar 4 merupakan ilustrasi dari sebuah decision tree.



Gambar 4. Ilustrasi Decision Tree

2.4.5 Random Forest

Random forest adalah algoritma klasifikasi dataset dalam jumlah besar. Algoritma ini membangun beberapa decision tree dan menggabungkannya untuk mendapatkan prediksi yang lebih akurat. Kumpulan tree ini biasanya dilatih dengan metode bagging, yaitu kombinasi model pembelajaran untuk meningkatkan hasil keseluruhan.

2.4.6 Naïve Bayes

Naïve bayes adalah teknik supervised klasifikasi objek dengan menetapkan label kelas kepada instansi menggunakan probabilitas bersyarat. Supervised merujuk pada klasifikasi data training yang sudah dilabel dengan kelas. Untuk perhitungan *naïve bayes* dapat dilakukan dengan Rumus 3.

$$P(A|B) = P(B|A)P(A)P(B) \quad (3)$$

Dengan:

$P(A|B)$: Probabilitas A terjadi dengan bukti bahwa B telah terjadi (superior)

$P(B|A)$: Probabilitas B terjadi dengan bukti bahwa A telah terjadi

$P(A)$: Peluang terjadinya A

$P(B)$: Peluang terjadinya B

2.5 Evaluasi Model Klasifikasi

Setelah model klasifikasi terbentuk maka akan dilakukan evaluasi berdasarkan data uji yang telah disediakan. Pengujian evaluasi dari setiap algoritma akan menggunakan *confusion matrix*. *Confusion matrix* merupakan pengukuram performa unntuk masalah klasifikasi *machine learning* dimana keluaran dapat berupa dua kelas atau

lebih. *Confusion matrix* adalah tabel dengan 4 kombinasi berbeda dari nilai prediksi dan nilai aktual. Tabel 2 merupakan *confusion matrix*.

Tabel 2. *Confusion matrix*

		Prediksi	
		Positif	Negatif
Aktual	Positif	True Positive (TP)	False Negative (FN)
	Negatif	False Positive (FP)	True Negative (TN)

True Positive (TP) : Prediksi positif dan itu benar.

False Positive (FP) : Prediksi negatif dan itu benar.

True Negative (TN) : Prediksi positif dan itu salah.

False Negative (FN) : Prediksi negatif dan itu salah.

Nilai evaluasi yang akan digunakan berupa nilai akurasi, *precision*, dan *recall*. Nilai akurasi adalah nilai yang menunjukkan perbandingan jumlah data yang teridentifikasi benar dengan seluruh jumlah data yang diujikan. Evaluasi berdasarkan pada banyaknya data uji yang diprediksi secara benar dan tidak benar oleh model. Nilai akurasi dapat dihitung dengan Rumus 4.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} * 100\% \quad (4)$$

Nilai *precision* merupakan perbandingan prediksi benar dibandingkan dengan keseluruhan data yang diprediksi. Nilai *precision* dapat dihitung dengan Rumus 5.

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

Nilai *recall* merupakan perbandingan prediksi benar dibandingkan dengan keseluruhan data yang aktual. Nilai *recall* dapat dihitung dengan Rumus 6.

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

3 Hasil Dan Pembahasan

Pengolahan data pada penelitian ini menggunakan ipynb dengan platform Google Collab untuk menganalisis data. Data penelitian yaitu `heart_failure_clinical_records_dataset.csv` yang dimuat ke dalam proyek memiliki data awal dengan jumlah baris 299 dan jumlah kolom 13. Data awal ini akan dipraproses terlebih dahulu. Pertama dilakukan pembersihan data dan dihasilkan bahwa jumlah *missing value* atau data kosong yaitu nol dan jumlah data yang berduplikat yaitu nol. Lalu dilakukan perubahan nama kolom untuk mempermudah pemrosesan yaitu kolom 'anaemia' menjadi 'anemia' dan kolom 'DEATH_EVENT' menjadi 'death'.

Selanjutnya dilakukan normalisasi data dan penyeimbangan target kolom. Normalisasi data dilakukan dengan fungsi `StandardScaler()` dimana data akan disamakan rentangnya untuk semua variable dari 0 – 1. Setelah normalisasi dilakukan maka kelas target data diseimbangkan dengan *Smote* yang dapat dilihat pada Tabel 3.

Tabel 3. Perbandingan Target Kelas Menggunakan *Smote*

Smote	0	1
Sebelum Smote	203	96
Sesudah Smote	203	203

Dapat dilihat pada tabel diatas bahwa kolom target sebelum dilakukan *Smote* belum seimbang dengan resiko kematian (0) berjumlah 203 dan tidak beresiko kematian (1) berjumlah 96 sehingga data pada kolom target harus diseimbangkan. Setelah dilakukan *Smote*, data resiko kematian (0) dan tidak beresiko kematian (1) menjadi seimbang dengan jumlah 203. Selanjutnya dilakukan pembagian data menjadi data latih dan data uji dengan rasio 80% : 20%.

Hasil evaluasi model klasifikasi dari keenam algoritma yang dipakai yaitu *Logistic Regression*, *K-Nearest Neighbour*, *Support Vector Machine*, *Decision Tree*, *Random Forest*, *Naïve Bayes*, dan *Ensemble Learning* dapat dilihat pada Tabel 4.

Tabel 4. Hasil Evaluasi Model Klasifikasi

Algoritma	Akurasi (%)	Precision (%)	Recall (%)
Logistic Regression	81	82	87
K-Nearest Neighbour	84	82	91
Support Vector Machine	86	84	93
Decision Tree	82	88	80
Random Forest	91	88	97
Naïve Bayes	80	84	80
Ensemble Voting	89	86	95

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *logistic regression* didapat bahwa akurasi sebesar 81%. Kelas resiko kematian benar berjumlah 26 data dan kelas tidak mati benar berjumlah 41 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *logistic regression* sebesar 87% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *logistic regression* sebesar 82%.

Tabel 5. *Confusion matrix Logistic Regression*

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	26	9
	Resiko Tidak Mati	6	41

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *k-nearest neighbour* didapat bahwa akurasi sebesar 84%. Kelas resiko kematian benar berjumlah 26 data dan kelas tidak mati benar berjumlah 43 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *k-nearest neighbour* sebesar 82% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *k-nearest neighbour* sebesar 91%.

Tabel 6. *Confusion matrix K-Nearest Neighbour*

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	26	9
	Resiko Tidak Mati	4	43

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *support vector machine* didapat bahwa akurasi sebesar 86%. Kelas resiko kematian benar berjumlah 27 data dan kelas tidak mati benar berjumlah 44 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *support vector machine* sebesar 84% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *support vector machine* sebesar 93%.

Tabel 7. Confusion matrix Support Vector Machine

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	26	9
	Resiko Tidak Mati	4	43

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *decision tree* didapat bahwa akurasi sebesar 82%. Kelas resiko kematian benar berjumlah 30 data dan kelas tidak mati benar berjumlah 38 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *decision tree* sebesar 88% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *decision tree* sebesar 90%.

Tabel 8. Confusion matrix Decision Tree

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	30	5
	Resiko Tidak Mati	9	38

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *random forest* didapat bahwa akurasi sebesar 91%. Kelas resiko kematian benar berjumlah 29 data dan kelas tidak mati benar berjumlah 46 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *random forest* sebesar 88% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *random forest* sebesar 97%.

Tabel 9. Confusion matrix Random Forest

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	29	6
	Resiko Tidak Mati	1	46

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *naïve bayes* didapat bahwa akurasi sebesar 80%. Kelas resiko kematian benar berjumlah 28 data dan kelas tidak mati benar berjumlah 38 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *naïve bayes* sebesar 84% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *naïve bayes* sebesar 80%.

Tabel 10. Confusion matrix Naïve Bayes

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	29	6
	Resiko Tidak Mati	1	46

Dari *confusion matrix* yang dihasilkan melalui tahapan evaluasi dengan algoritma *ensemble learning* didapat bahwa akurasi sebesar 89%. Kelas resiko kematian benar berjumlah 28 data dan kelas tidak mati benar berjumlah 45 data dengan total keseluruhan data uji berjumlah 82. Nilai recall pada algoritma *ensemble learning* sebesar 86% dari total keseluruhan data yang benar-benar resiko mati. Nilai precision pada algoritma *ensemble learning* sebesar 95%.

Tabel 11. Confusion matrix Ensemble Learning

		Prediksi	
		Resiko Kematian	Resiko Tidak Mati
Aktual	Resiko Mati	28	7
	Resiko Tidak Mati	2	45

4 Kesimpulan Dan Saran

4.1 Kesimpulan

Dari hasil dan pembahasan yang telah dilakukan dapat disimpulkan bahwa algoritma yang memiliki akurasi paling tinggi yaitu *Random Forest* dengan 91%, *Ensemble voting* dengan 89%, *Support Vector Machine* dengan 86% dengan *K-Nearest Neighbours Classifier* dengan 84%, *Decision Tree* dengan 82%, *Logistic Regression* dengan 81%, terakhir *Naïve Bayes* dengan 80%. Selanjutnya algoritma yang memiliki *precision* paling tinggi yaitu *Random forest* dengan 88%, *Decision Tree* dengan 88%, *Ensemble voting* dengan 86%, *Naïve Bayes* dengan 84%, *Support Vector Machine* dengan 84%, *Logistic Regression* dengan 82%, *K-Nearest Neighbours Classifier* dengan 82%. Selanjutnya algoritma yang memiliki *recall* paling tinggi yaitu *Random forest* dengan 97%, *Ensemble Voting* dengan 95%, *Support Vector Machine* dengan 93%, *K-Nearest Neighbours Classifier* dengan 91%, *Logistic Regression* dengan 87%, *Decision Tree* dengan 80%, terakhir *Naïve Bayes* dengan 80%.

Meskipun *Ensemble Voting* merupakan algoritma yang menggabungkan *logistic regression*, *support vector machine*, *k-nearest neighbours classifier*, *decision tree*, *random forest*, dan *naïve bayes* namun nilai akurasi, *precision* dan *recall*-nya masih kalah dibanding algoritma *Random Forest* sehingga dapat disimpulkan algoritma *Random forest* masih lebih baik dibanding *ensemble voting* yang dilakukann dalam penelitian ini.

4.2 Saran

Pada penelitian ini memiliki dataset yang sedikit yaitu sekitar 299 baris, sebaiknya pada penelitian selanjutnya diharapkan untuk menambahkan data yang banyak sehingga mesin dapat belajar dan membuat keputusan atau prediksi yang lebih akurat ketika digunakan dalam kasus nyata.

Referensi

- [1] K. K. RI and Pusat Data an Informasi Kementrian Kesehatan RI, "Situasi Kesehatan Jantung," *Kementrian Kesehatan*, pp. 1–8, 2014, Accessed: Aug. 07, 2022. [Online]. Available: <https://pusdatin.kemkes.go.id/download.php?file=download/pusdatin/infodatin/infodatin-jantung.pdf>
- [2] American Heart Association, "Heart disease and stroke statistics—2020 update a report from the American Heart Association," *AHA SStatistical Update*, vol. 141, no. 9, pp. E139–E596, Aug. 2020, doi: 10.1161/CIR.0000000000000757.
- [3] M. S. Donald M. Lloyd-Jones, "Cardiovascular risk prediction: Basic concepts, current status, and future directions," *CIRCULATION AHA*, vol. 121, no. 15, pp. 1768–1777, Apr. 2010, doi: 10.1161/CIRCULATIONAHA.109.849166.
- [4] I. Cholissodin, Sutrisno, A. A. Soebroto, and Y. I. Febiola, "AI, Machine Learning & Deep Learning (Teori & Implementasi)," Malang, Jul. 2019. [Online]. Available: <https://www.researchgate.net/publication/348003841>

- [5] R. Sathya and A. Abraham, "Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification," *International Journal of Advanced Research in Artificial Intelligence*, vol. 2, no. 2, 2013, doi: 10.14569/ijarai.2013.020206.
- [6] X. Ying, "Ensemble Learning," Georgia, May 2014. [Online]. Available: <https://www.researchgate.net/publication/262369664>
- [7] M. Mega Santoni, N. Chamidah, and N. Matondang, "Prediction of Hypertension using Decision Tree, Naïve Bayes and Artificial Neural Networks in KNIME Analytics Platform," *Techno.COM*, vol. 19, no. 4, pp. 353–363, Nov. 2020, Accessed: Aug. 08, 2022. [Online]. Available: <http://publikasi.dinus.ac.id/index.php/technoc/article/view/3872/2118>
- [8] W. Nugraha, "PREDIKSI PENYAKIT JANTUNG CARDIOVASCULAR MENGGUNAKAN MODEL ALGORITMA KLASIFIKASI," 2021. Accessed: Aug. 08, 2022. [Online]. Available: <https://repository.bsi.ac.id/index.php/unduh/item/344564/01-Nugraha---Prediksi-Penyakit-Jantung-Cardiovascular-Menggunakan-Model-Algoritma-Klasifikasi.pdf>