JurnalLitbangEdusaintech (JLE)

http://journal.pwmjateng.com/index.php/jle

Algoritma *Decision Tree* Dan *Smote* Untuk Klasifikasi Serangan Jantung Miokarditis Yang *Imbalance*

Aldi Fianda Putra[*1], *Ahmad Saifuddin*[2], *Noverio Athariq Syafaz*[3], *Novanto Yudistira*[4]

Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Jalan Veteran No. 8 Malang, Jawa Timur 65145
*Correspondence: E-mail: [1]aldifp8492@student.ub.ac.id

ABSTRACTS

Heart attack or in medical terms called myocardial infarction is a serious heart problem. can be detected by using the complications suffered by patients. using Naive Bayes, Decision Tree, and Support Vector Machine as algorithms that we evaluate will detect heart attack. However, it is not immediately possible to evaluate those algorithms. Before evaluating these three algorithms, the dataset was repaired, because it contained empty data. Improvements are made by imputing data where the value is estimated based on the average of the cluster members in the same class. After that, it can be normalized using the MinMax method with aiming the feature range, especially continuous numeric data, thus the feature range is not too large. After preprocessing is committed, we can evaluate data using K-fold Cross Validation method. Yet again, an error was found, the training data that we used was not balanced. Therefore, oversampling is implemented on the data, thus the data becomes balanced. Once the dataset is balanced, we can re-evaluate and obtain a suitable algorithm to classify the data of Myocardial Infarction Complication dataset using Decision Tree algorithm with 98% accuracy, followed by the Support Vector Machine algorithm with 91% accuracy and Naïve Bayes with the lowest accuracy of 49%.

ARTICLE INFO

Article History:
Received 15 Agustus 2021
Revised 30 Des 2021
Accepted 31 Des 2021
Available online 31 Des 2021

Keyword:
algorithm,
classification,
dataset,
evaluation,
myocardial infarction

ABSTRAK

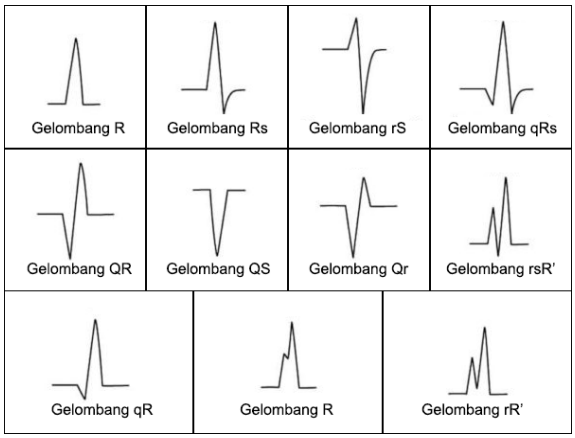
Serangan jantung atau dalam medis bernama *Myocardial Infarction* atau infark miokard adalah gangguan jantung yang sangat serius. Dalam pendeteksian ini menggunakan komplikasi-komplikasi yang diderita oleh pasien. Algoritma yang akan dievaluasi yaitu Naive Bayes, Decision Tree, dan Support Vector Machine. Namun tidak serta merta dapat dilakukan evaluasi. Sebelum mengevaluasi ketiga algoritma ini dilakukan perbaikan dataset, karena pada dataset ini sendiri terdapat data yang kosong. Perbaikan dilakukan dengan cara mengimputasikan data dimana nilai diperkirakan berdasarkan rata-rata dari anggota klaster pada kelas yang sama. Setelah melakukan imputasi data, maka dapat dilakukan normalisasi dengan metode MinMax dengan tujuan agar rentang fitur terutama data numerik kontinu tidak terlalu besar. Setelah pemrosesan data awal dilakukan maka barulah kita dapat melakukan evaluasi dengan menggunakan metode K-fold Cross Validation. Namun lagi-lagi ditemukan kesalahan yakni data latih yang digunakan ternyata tidak seimbang. Oleh sebab itu dilakukan oversampling pada data agar data menjadi seimbang. Setelah seimbang maka kita dapat melakukan evaluasi kembali dan diperolehlah algoritma yang cocok untuk mengklasifikasikan data seperti dataset Myocardial Infarction Complications adalah algoritma Decision Tree dengan akurasi 98%, diikuti algoritma Support Vector Machine dengan akurasi 91% dan Naïve Bayes dengan akurasi paling rendah yakni 49%.

Kata kunci:
algoritma,
dataset,
evaluasi,
klasifikasi,
serangan jantung

1. INTRODUCTION

Serangan jantung merupakan gangguan jantung yang sangat serius. Gangguan ini terjadi ketika otot jantung tidak mendapatkan aliran darah yang baik. Kondisi inilah yang akan mengganggu fungsi jantung dalam mengalirkan aliran darah ke seluruh tubuh. Hal ini dapat berakibat fatal bagi kesehatan manusia (Fadli, R., 2021). Di Amerika Serikat sendiri penyakit ini dialami oleh 1,5 juta orang setiap tahunnya (Zafari, A.M., 2019). Berdasarkan data dari WHO pada tahun 2019 terdapat 17.9 juta kematian yang merupakan akibat dari *Cardiovascular Disease* atau CVD yang mana angka tersebut mewakili 32% kematian yang ada pada tahun tersebut, selain itu dari total kematian akibat CVD tersebut 85% merupakan korban dari ganasnya penyakit serangan jantung dan stroke ini (WHO, 2021). Dalam dunia medis nama lain dari serangan jantung adalah *myocardial infarction* atau infark miokard.

Penyakit serangan jantung ini dapat dideteksi dengan memanfaatkan alat yang bernama *Elektrokardiogram* atau dikenal dengan sebutan mesin EKG (Pane, M.D.C., 2020). Seperti yang kita ketahui denyut jantung sendiri memiliki irama, irama atau aktivitas jantung inilah yang dideteksi dengan menggunakan mesin *elektrokardiogram* ini (Kinyanjui, Y., 2018). Gelombang ini sendiri terdiri dari 3 jenis gelombang yakni sinyal Q, R dan S yang mana tidak harus semua gelombang ini ada dalam sebuah jenis gelombang. Nilai dari QRS ini tergantung dari besar dan letak gelombang tersebut, apabila gelombang negatif terletak di depan gelombang positif maka gelombang itu adalah gelombang Q, namun apabila terletak di belakang gelombang positif maka gelombang tersebut adalah gelombang S. Ukuran gelombang juga menentukan penamaan, apabila gelombangnya besar maka akan ditulis dalam huruf kapital (QRS), namun jika gelombangnya kecil maka akan ditulis dalam huruf kecil juga (qrs) (Burns, E., 2021). Terdapat 11 macam gelombang denyut jantung yang dapat diamati pada mesin EKG ini, gelombangnya seperti gambar di bawah:



Gambar 1. Gelombang EKG sinyal QRS (Larkin, J., 2021)

Dalam studi kasus ini, dilakukan sebuah evaluasi terhadap model pembelajaran mesin dengan memanfaatkan tiga buah model yakni dengan memanfaatkan metode klasifikasi *Naive Bayes*, *Decision Tree*, dan *Support Vector Machine.*

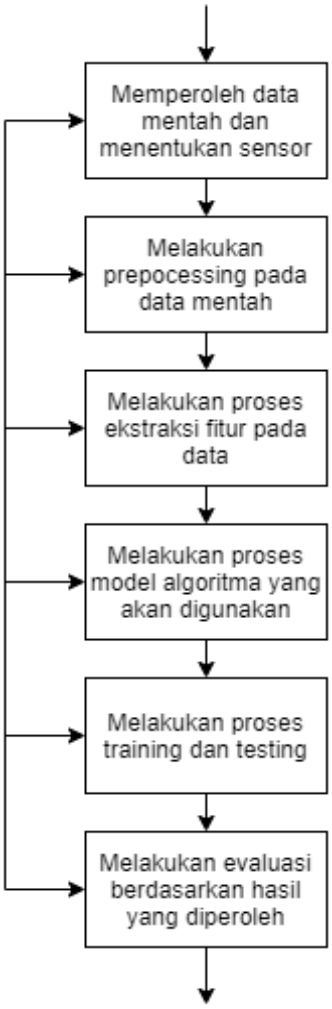
Dalam hal ini evaluasi tidak dapat dilakukan secara langsung karena masih terdapat data yang hilang sehingga perlu dilakukan perbaikan terhadap kumpulan data ini dengan menggunakan metode *Chained Imputation* dan akan dilanjutkan dengan normalisasi data menggunakan *Min-Max*. Hasil yang diperoleh dari evaluasi ini, nantinya akan diujikan akurasi kebenaran klasifikasi dengan memanfaatkan ketiga algoritma di atas agar ditemukan sebuah model algoritma yang paling cocok dan akurat dalam mengklasifikasikan kumpulan data seperti data *Myocardial Infarction Complication*.

2. METHODS

Pada studi kasus ini, dataset dari komplikasi serangan jantung atau *Myocardial Infarction Complication* adalah dataset yang dibuat oleh beberapa ilmuwan yang mempelajari kecerdasan buatan untuk kebutuhan medis seperti serangan jantung, yang mana ilmuwan tersebut adalah S.E. Golovenkin, V.A. Shulman, D.A. Rossiev, P.A. Shesternya, S.Yu. Nikulina, Yu.V. Orlova yang mana dibantu oleh professor V.F. Voino dari Yasenetsky Krasnoyarsk State Medical University, dan pada akhir tahun 2020, dataset dikirimkan oleh E.M. Mirkes ke machine learning repository yang nantinya bisa digunakan untuk melakukan

DOI: http://dx.doi.org/10.51402/jle.v2i2.48
p- ISSN 2746-3478 e- ISSN 2746-376X

prediksi komplikasi Myocardial Infraction sehingga bisa membantu tenaga medis dalam upaya pencegahan atau penanganan yang bisa menyelamatkan pasien. Dengan demikian, dataset ini dapat diunduh pada halaman https://archive.ics.uci.edu/ml/datasets/Myocardial+infarction+complications. Dataset yang dikembangkan S.E. Golovenkin dkk ini merupakan hasil pengamatan dari total 1700 pasien dengan jumlah fitur sebanyak 124 fitur (Golovenkin, S.E., 2021). Fitur-fitur ini diperoleh dengan cara pengamatan seperti penggunaan mesin EKG sebelumnya, dengan memanfaatkan alat pengukur tekanan darah ataupun dengan menanyakan langsung pada pasien seperti penggunaan obat-obatan. Sehingga dengan adanya fitur-fitur yang menunjukkan hasil pengamatan dan juga pengonsumsian obat-obatan oleh pasien tersebut dapat dilakukan klasifikasi yang mana dalam studi kasus ini terdapat 8 klaster hasil klasifikasi yang menunjukkan gangguan jantung yang dialami oleh pasien.



Gambar 2. Alur pelatihan

Gambar 2 menunjukan alur proses dari pembelajaran mesin yang akan dilakukan untuk evaluasi algoritma pembelajaran mesin terhadap dataset *Myocardial Infarction Compilation* (MIC). Pada studi kasus ini pertama kali dilakukan pengunduhan dataset dari halaman *Machine Learning Repository,* dari penjelasan mengenai dataset MIC pada halaman *Machine Learning Repository* dijelaskan bahwa data yang multivariasi ini terdapat 1700 data dengan fitur sebanyak 124. Kemudian pada dataset ini juga terdapat *missing value* yang ditandai dengan simbol "?". Dengan demikian pada tahapan berikutnya yakni tahapan *preprocessing* atau pemrosesan data awal dilakukan imputasi dengan menggunakan metode *Chained Imputation.* Metode imputasi sendiri adalah mengisi data hilang dengan nilai yang diperkirakan cukup layak dan kemudian dianalisis dengan metode baku untuk data lengkap (Evriyanto, Y., 2004).

Namun pada studi kasus kami ini terdapat sebuah permasalahan, dimana dengan melakukan imputasi rata-rata terhadap data yang hilang, tidak semua *missing value* terisi. Seperti yang kita ketahui imputasi dengan mengubah missing value dengan rata-rata per-kolom nya memiliki rumus sebagaimana pada persamaan (1) berikut (Norazian, M.N., 2008):

$$y = (y_n + y_n + 1)/2 \quad (1)$$

Maka untuk mengatasi permasalahan ini, kami menggunakan metode *chained imputation,* dimana satu hasil imputasi yang memprediksi nilai untuk sebuah *missing value* akan menjadi patokan bagi imputasi untuk *missing value* berikutnya. Karena pada dataset MIC ini tidak hanya berisikan data numerik yang bersifat kontinu, namun mayoritas berisikan data numerik bersifat kategorikal yang berarti nilainya harus berupa bilangan bulat dan tidak bisa bilangan desimal. Maka pada metode *chained imputation* sebelumnya akan ditambahkan metode pembulatan *round* dimana akan membulatkan ke nilai yang lebih cocok.

Pada dataset sendiri apabila diperhatikan pada sebuah fitur, bisa saja fitur tersebut memiliki beberapa baris data yang tidak memiliki nilai atau *missing value*. Dengan menggunakan metode *chained imputation* ini, sebuah *missing value* akan diprediksi berdasarkan nilai-nilai yang berdekatan dengan data lain yang satu kelas dengan data tersebut. Kemudian nilai tersebut dirata-ratakan untuk diperoleh nilai yang dimungkinkan untuk fitur yang hilang pada data tersebut. Namun untuk mengantisipasi adanya data yang bersifat kategorikal yang direpresentasikan dengan angka maka untuk itu nilai hasil imputasi ini dibulatkan dengan metode *round* dimana dibulatkan ke atas jika lebih besar dari 0.5 dan dibulatkan ke bawah jika kurang dari 0.5 dengan tidak ada nilai desimal di belakang koma (pada aplikasi sendiri tetap terdapat 0 di belakang koma). Kemudian dilakukan imputasi sekali lagi namun dengan menggunakan cara yang sedikit berbeda dari sebelumnya yakni cukup mencari rata-rata dari kolom fitur yang sebelumnya terdapat data yang hilang. Dengan demikian terjadi dua kali proses imputasi pada sebuah data agar dapat mengisikan nilai yang hilang meskipun telah dilakukan normalisasi.

Pemrosesan data awal tidak selesai pada tahap ini, masih terdapat metode yang dilakukan untuk pemrosesan awal yakni dengan melakukan normalisasi. Metode normalisasi yang digunakan adalah normalisasi *Min-Max* dimana rentang nilai pada sebuah kolom fitur hanya berjarak 0 sampai 1, sehingga normalisasi ini cukup dilakukan pada data kontinu. Untuk melakukan normalisasi kita dapat menggunakan persamaan seperti pada persamaan (2) berikut:

$$X_{baru} = (X_{lama} - X_{min}) / (X_{max} - X_{min}) \quad (2)$$

Setelah dilakukan pemrosesan awal dan ekstraksi fitur maka selanjutnya adalah menentukan model algoritma yang akan digunakan. Tentunya pada tahap ini algoritma yang akan dievaluasi adalah algoritma *Naive Bayes, Support Vector Machine* dan *Decision Tree.* Evaluasi ini dilakukan dengan menggunakan metode *K-Fold Cross Validation.* Sebelum masuk pada metode evaluasinya, data dibagi menjadi dua kelompok yakni data uji dan data latih dengan perbandingan 20:80. Kemudian dengan menggunakan metode *K-Fold Cross Validation* ini data latih akan dibagi sebanyak k, pada studi kasus ini nilai k yang dimasukan adalah nilai *default* yakni 10. Dengan demikian data latih akan dibagi menjadi 10 bagian yang mana 10 bagian ini berarti terdapat 1 bagian yang digunakan sebagai percobaan dan 9 bagian lagi sebagai data latih.

Setelah menentukan metode pelatihannya, maka masuk pada tahap berikutnya yakni tahap pelatihan itu sendiri atau *training.* Metode *training* ini dilakukan dengan cara menentukan model algoritma yang akan dievaluasi, semisal algoritma SVM maka kita dapat menggunakan model *SVC(gamma="auto")* yang terdapat pada *library* SKlearn. Untuk nilai *gamma* yang "*auto*" sendiri menunjukan bahwa SVM yang digunakan adalah SVM linier. Selanjutnya dengan menggunakan fungsi *make classification* yang dimiliki *library* akan menghasilkan nilai *precision, recall* dan *f1-score* dari sebuah evaluasi algoritma. Nilai *precision, recall* dan *f1-score* ini menggunakan nilai TP, TN, FN dan FP dari suatu kelas. Namun tabel *confusion matrix* tidak dibuat secara eksplisit pada fungsi *make classification* tersebut. Sehingga kita dapat menambahkan tabel *confusion matrix* untuk melihat nilai dari *true positive*(TP), *true*

DOI: http://dx.doi.org/10.51402/jle.v2i2.48
p- ISSN 2746-3478 e- ISSN 2746-376X

negative(TN)*, false positive* (FP) dan *false negative*(FN) dari evaluasi sebuah algoritma. Studi kasus ini dilakukan dengan menggunakan layanan Google Collaboratory sehingga membutuhkan koneksi internet untuk melakukan penelitian. Tentunya pada Google Collab ini menggunakan bahasa pemrograman Python, lalu kami juga menggunakan *library* Numpy, Pandas, SKLearn, Matplotlib, Collections, dan IMBLearn.

1. EKSPERIMEN

Dari metode penelitian yang telah dipilih sebelumnya untuk mengklasifikasikan penyakit jantung yang dialami pasien dengan memanfaatkan pembelajaran mesin ini menjadi 8 kluster yakni:

1) 0 berarti kondisi sehat.
2) 1 berarti mengidap syok kardiogenik.
3) 2 berarti mengidap edema paru.
4) 3 berarti mengidap myocardial rupture.
5) 4 berarti mengidap gagal jantung kongestif.
6) 5 berarti mengidap thromboembolism.
7) 6 berarti mengidap asystole.
8) 7 berarti mengidap ventrikular fibrilasi.

Eksperimen selanjutnya dilakukan dengan melakukan imputasi data (menggunakan metode chained imputation sebelumnya), normalisasi Min-Max, dan memilih model evaluasi K-Fold Cross Validation untuk mengevaluasi algoritma Naive Bayes, Support Vector Machine dan Decision Tree terhadap dataset MIC. Maka diperoleh hasil prediksi (perlu diketahui hasil prediksi bersifat random yang selalu berbeda setiap kali menjalankan kode program) dari eksperimen pertama yang mana hasilnya sebagai tabel berikut:

Tabel 1. Prediksi dengan algoritma Naïve Bayes

NO	Precision	Recall	F1-score	Support
0.0	0.99	0.50	0.67	296
1.0	0.44	0.21	0.29	19
2.0	0.05	0.67	0.10	3
3.0	1.00	1.00	1.00	8
4.0	0.00	0.00	0.00	7
5.0	0.02	0.33	0.03	3
6.0	0.00	0.00	0.00	2
7.0	0.00	0.00	0.00	2
accuracy			0.48	340
Macro avg	0.31	0.34	0.26	340
W. avg	0.91	0.48	0.62	340

Tabel 2. Prediksi dengan algoritma Decision Tree

NO	Precision	Recall	F1-score	Support
0.0	1.00	1.00	1.00	296
1.0	0.86	1.00	0.93	19
2.0	1.00	0.33	0.50	3
3.0	1.00	1.00	1.00	8
4.0	1.00	1.00	1.00	7
5.0	0.67	0.67	0.67	3
6.0	1.00	1.00	1.00	2
7.0	0.50	0.50	0.50	2

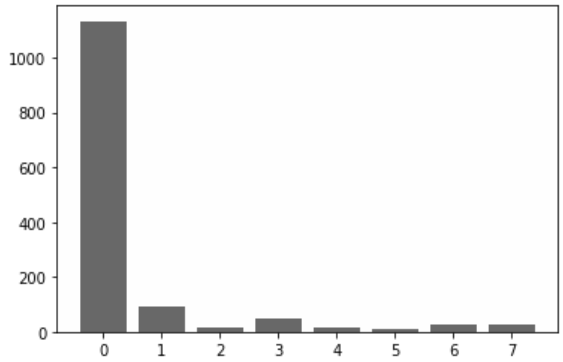
accuracy			0.99	340
Macro avg	0.88	0.81	0.82	340
W. avg	0.99	0.99	0.98	340

Tabel 3. Prediksi dengan algoritma SVM (Support Vector Machine)

NO	Precision	Recall	F1-score	Support
0.0	0.97	1.00	0.98	296
1.0	0.52	0.74	0.61	19
2.0	0.00	0.00	0.00	3
3.0	0.50	0.38	0.43	8
4.0	1.00	0.14	0.25	7
5.0	0.00	0.00	0.00	3
6.0	0.00	0.00	0.00	2
7.0	1.00	0.50	0.67	2
accuracy			0.92	340
Macro avg	0.50	0.34	0.37	340
W. avg	0.91	0.92	0.91	340

Setelah kami melakukan eksperimen atau percobaan pertama dengan menggunakan 3 algoritma yang berbeda untuk prediksi MIC, dimana pada ketiga hasil tersebut, untuk akurasi dari algoritma Naive Bayes, Decision Tree, dan SVM berturut - turut adalah 48% , 99%, dan 92%. Sekilas kita akan langsung mengetahui bahwa akurasi dari algoritma Decision Tree yang paling baik.

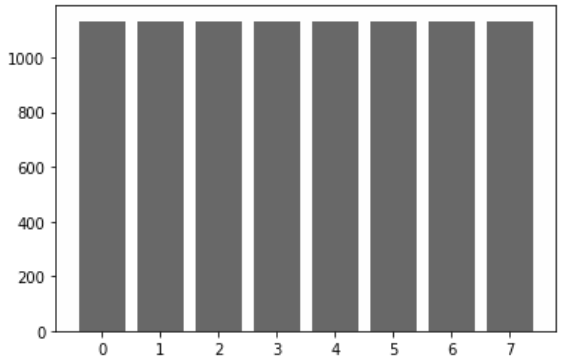
Diketahui bahwa meskipun preprocessing data telah dilakukan untuk meminimalisir kesalahan, namun masih terdapat suatu kekurangan yaitu perbedaan jumlah signifikan pada penderita yang sehat dengan jumlah yang jauh lebih besar dibandingkan dengan total jumlah penderita yang berjumlah lebih sedikit pada data latih. Perbedaan perbandingan jumlah pasien yang pada data latih ini bisa menyebabkan miskalkulasi atau bisa disebut juga dengan imbalanced dataset, Imbalanced dataset sendiri adalah masalah umum dalam klasifikasi pembelajaran mesin di mana terdapat rasio yang tidak proporsional di setiap kelas (Rumagit, R.Y., 2019). Sehingga akan lebih baik jika melakukan penyeimbangan pada jumlah data yang akan dilatih. Tampilan jumlah data pada setiap klaster data latih sebelum dilakukan balancing adalah sebagai berikut:



Gambar 3. Tampilan imbalanced data pada data latih

Pada tampilan di atas, kluster 0 (Sehat) memiliki data sebanyak 1133 data yang mana sangat jauh berbeda dari kluster lainnya. Teknik yang bisa dilakukan dalam penyeimbangan ini adalah resample, yaitu oversampling / undersampling pada data latih, jadi data uji tidak perlu dilakukan oversampling, begitu juga dengan dataset secara keseluruhan atau variabel x dan y. Pada studi kasus kami ini, kami menggunakan teknik oversampling untuk menyeimbangkan data latih. Oversampling sendiri adalah mengambil kelas minoritas sedemikian rupa sehingga proporsinya dalam sampel lebih besar dibandingkan proporsi asalnya (Sartono, B., 2018). Dengan kata lain semua klaster yang memiliki anggota lebih kecil daripada klaster dengan anggota terbesar akan dipaksakan memiliki anggota sama banyak. Dengan demikian, semua data pada klaster-klaster akan mengikuti banyaknya data pada klaster dengan anggota terbanyak, yakni semua kluster memiliki anggota sebanyak 1133 yang dimana mengikuti jumlah anggota klaster 0 (Sehat) seperti gambar berikut:

DOI: http://dx.doi.org/10.51402/jle.v2i2.48
p- ISSN 2746-3478 e- ISSN 2746-376X



Gambar 4. Tampilan data latih yang sudah *balanced*

Pada bahasa pemrograman Python sendiri, untuk melakukan oversampling kita dapat memanfaatkan algoritma SMOTE atau Synthetic Minority Oversampling Technique. Metode SMOTE ini diperoleh dari library "imblearn" dan ditampilkan dengan menggunakan histogram seperti gambar di atas dengan memanfaatkan library "matplotlib". Dengan menggunakan library imblearn sendiri oversampling dapat mudah dilakukan, yakni cukup dengan membuat sebuah variabel bernama terserah (pada eksperimen kami menggunakan nama oversampling agar dapat mudah diingat) dan kemudian meng-import fungsi SMOTE pada variabel tersebut. Variabel oversampling sebelumnya dikombinasikan dengan fungsi resampling yang ditujukan pada data latih, yakni variabel X_train dan Y_train. Dengan demikian, proses oversampling pun dilakukan dan akan diperoleh anggota pada setiap klaster sama (Brownlee, J., 2019).

Dengan data latih yang sudah di-oversampling tersebut, maka akan terjadi perubahan pada hasil evaluasi algoritma klasifikasi. Perubahan yang terjadi tidak begitu signifikan karena jumlah data uji yang digunakan tetap sama yakni 340 data, dimana 340 data uji ini merupakan 20% total dari semua data. Pada data uji ini 340 data tersebut juga merupakan komposisi dari 20% data dari setiap klaster. Dengan sudah dilakukannya oversampling, maka satu klaster seperti misalnya klaster 7 dimana memiliki anggota paling sedikit yakni 2, akan dilatih dengan data latih yang setiap klaster pada data latih memiliki anggota sebanyak 1133. Seperti tampilan tabel hasil evaluasi sebelumnya, dengan menggunakan fungsi make classification maka akan diperoleh nilai recall, precision, f1-measure, accuracy, macro avg dan weighted avg. Nilai tersebut menggunakan nilai dari TP, TN, FP, FN yang tidak dituliskan secara eksplisit pada fungsi tersebut. Untuk mendapatkan nilai dari recall, precision dan f1-measure sendiri diperoleh dari persamaan berikut. Dimana persamaan 3 adalah recall, persamaan 4 adalah precision dan persamaan 5 adalah f1-measure.

$$Recall = TP / TP + FN \quad (3)$$

$$Precision = TP / TP + FP \quad (4)$$

$$F1 - Measure = 2 * (R * P) / (R + P) \quad (5)$$

Recall atau sensitivity merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Precision merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. F1 - Measure atau F1 - score merupakan perbandingan rata-rata presisi(P) dan recall(R) yang dibobotkan (Arthana, R., 2019).

3. RESULTS AND DISCUSSION

Kami telah melakukan percobaan di atas untuk mendapatkan hasil evaluasi algoritma yaitu *Naive Bayes, Decision Trees,* dan *SVM.* Terdapat 8 klasifikasi/klaster pada dataset MIC ini yang direpresentasikan 0 sampai 7. Model ini dievaluasi dengan *k-fold cross validation* sebanyak k nya adalah 10 kali. Data uji dan data latih telah dibagi menjadi 20:80. Setelah melakukan eksperimen sebelumnya, yang mana masih terdapat kesalahan dimana persebaran data yang tidak seimbang pada setiap klaster. Maka setelah diperbaiki dengan menggunakan fungsi SMOTE dan melakukan eksperimen ulang untuk memperoleh hasil evaluasi dengan menggunakan data yang seimbang atau *balanced,* maka diperoleh hasil *precision, recall,* dan *f1-score* seperti tabel berikut.

Tabel 4. Naive Bayes Resample

NO	Precision	Recall	F1-score	Support
0.0	0.99	0.49	0.66	296
1.0	0.50	0.37	0.42	19

2.0	0.04	0.67	0.07	3
3.0	1.00	1.00	1.00	8
4.0	0.04	0.14	0.06	7
5.0	0.03	0.33	0.05	3
6.0	0.00	0.00	0.00	2
7.0	0.00	0.00	0.00	2
accuracy			0.49	340
Macro avg	0.32	0.38	0.28	340
W. avg	0.91	0.49	0.62	340

6.0	0.50	0.50	0.50	2
7.0	0.33	0.50	0.40	2
accuracy			0.98	340
Macro avg	0.67	0.69	0.68	340
W. avg	0.97	0.98	0.97	340

Tabel 5. Decision Tree Resample

NO	Precision	Recall	F1-score	Support
0.0	1.00	1.00	1.00	296
1.0	0.86	1.00	0.93	19
2.0	0.00	0.00	0.00	3
3.0	1.00	1.00	1.00	8
4.0	1.00	0.86	0.92	7
5.0	0.67	0.67	0.67	3

Tabel 6. SVM(Support Vector Machine) Resample

NO	Precision	Recall	F1-score	Support
0.0	0.99	1.00	0.99	296
1.0	0.62	0.42	0.50	19
2.0	0.00	0.00	0.00	3
3.0	0.43	0.38	0.40	8
4.0	0.00	0.00	0.00	7
5.0	0.00	0.00	0.00	3

DOI: http://dx.doi.org/10.51402/jle.v2i2.48
p- ISSN 2746-3478 e- ISSN 2746-376X

6.0	0.00	0.00	0.00	2
7.0	0.17	0.50	0.25	2
accuracy			0.90	340
Macro avg	0.28	0.29	0.27	340
W. avg	0.91	0.90	0.91	340

Dari hasil evaluasi kembali di atas yaitu tabel 4-6, apabila diperhatikan terdapat perbedaan yang signifikan dari hasil-hasil yang ada pada seperti pada tabel 1 sampai 3. Dalam kasus ini kami mengambil pada hasil prediksi dengan akurasi terbaik yakni hasil *decision tree*. Pada tabel 2 sebelumnya tidak terdapat nilai P, R maupun F1 yang bernilai 0. Karena pada tabel sebelum dilakukan *oversampling,* semua klaster memiliki nilai yang berbeda-beda, sehingga kemungkinan untuk diklasifikasikan sebagai "Sehat" yang memiliki jumlah anggota yang tinggi daripada yang lainnya sehingga kemungkinan besar akan terjadi miskalkulasi, namun masih terdapat harapan bahwa hasil klasifikasi memperoleh nilai 2. Sebagai sampel berikut tampilan tabel *confusion matrix* dari klaster 2 yakni pasien pengidap edema paru yang diperoleh dengan menjalankan fungsi *multilabel confusion matrix*.

Tabel 7. Confusion matrix klister 2 pada tabel 2 (belum oversampling)

	True	**False**
True	1	2
False	0	337

Tabel 8. Confusion matrix klaster 2 pada tabel 5 (sudah oversampling)

	True	**False**
True	0	3
False	1	336

Pada tabel 5 sendiri diperoleh nilai *precision, recall,* dan *f1-score* bernilai 0 karena sejak awal nilai dari *true positive* klaster 2 tersebut sudah 0 sehingga akan menghasilkan 0 apapun operasinya. Pada klaster 2 pada tabel 2 nilai *true positive* adalah 1, *true negative* bernilai 337, *false positive* bernilai 0, dan *false negative* bernilai 2. Maka dengan menggunakan persamaan 3, 4 dan 5 akan diperoleh nilai berikut:

$$Recall = 1/(1 + 2) = 1/3 = 0.33$$

$$Precision = 1/(0 + 1) = 1/1 = 1$$

$$F1 - Measure = 2(0.33 * 1)/(0.33 + 1 = 0.66/1.33 =$$

Dengan melakukan oversampling, pada klaster 2 setelah melakukan *oversampling* atau pada tabel 5 memang tidak diperoleh ketiga nilai tersebut, pada klaster 2 tabel 5 sendiri data berjumlah 3 diujikan dengan 9064 data latih yang menyebabkan kemungkinan data diklasifikasikan dengan benar itu sangat kecil dan bisa dikatakan tidak mungkin diklasifikasikan dengan benar. Berbeda dari klaster 2 tabel 2 dimana belum dilakukan *oversampling,* 3 data diujikan dengan 1360 data yang mana peluang untuk diklasifikasikan dengan benar itu masih ada. Dengan demikian, hasil R, P dan F1 yang diperoleh pada prediksi bergantung kepada banyaknya data uji, mengingat *oversampling* hanya dilakukan pada data latih namun tidak pada data uji. Maka tentunya data uji yang kecil akan memiliki kemungkinan besar untuk diklasifikasikan dengan tidak benar. Namun dengan adanya *oversampling* ini juga akan menyebabkan prediksi dengan benar dimana setiap data akan diujikan dengan data latih yang memiliki jumlah anggota klaster yang seimbang.

4. CONCLUSION

Dari eksperimen yang dilakukan, diperoleh algoritma *Decision Tree* memiliki akurasi yang lebih bagus dibandingkan algoritma *Naive Bayes* dan *Support Vector Machine* atau SVM, baik sebelum dilakukan ataupun setelah dilakukan *oversampling*. Hal ini dibuktikan dengan akurasi yang konsisten di atas 95%. Sehingga dapat disimpulkan, untuk sebuah dataset yang terdiri atas campuran data kategorik dan numerik akan lebih baik diklasifikasikan dengan menggunakan algoritma *Decision Tree,* dengan

Naive Bayes sendiri hasil yang diperoleh sangat tidak memuaskan dimana akurasi kebenaran hanya sekitar kurang dari 50%. Untuk SVM sendiri lumayan baik namun tidak memperoleh akurasi sebaik algoritma *Decision Tree* yang telah dijabarkan sebelumnya.

5. REFERENCES

GOLOVENKIN, S.E., GORBAN, A.N., MIRKES, E.M., 2020. Myocardial infarction complications Data Set. [online] Available at: <https://archive.ics.uci.edu/ml/datasets/Myocardial+infarction+complications> [Accessed 6 May 2021].

DARAEI, A., HAMIDI, H., 2017. An Efficient Predictive Model for Myocardial Infarction Using Cost-sensitive J48 Model. [online] Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5442282/> [Accessed 7 May 2021].

KINYANJUI, Y., 2018. LEARNING THE EKG. [online] Available at: <https://slideplayer.com/slide/12255891/> [Accessed 7 May 2021].

LARKIN, J., 2021. QRS Interval. [online] Available at: <https://litfl.com/qrs-interval-ecg-library/> [Accessed 7 May 2021].

ZAFARI, A.M., 2019. Myocardial Infarction. [online] Available at: <https://emedicine.medscape.com/article/155919-overview> [Accessed 7 May 2021].

GOLOVENKIN, S.E., GORBAN, A., MIRKES, E., SHULMAN, V.A., ROSSIEV, D.A., SHESTERNYA, P.A., NIKULINA, S.YU., ORLOVA, YU.V., DORRER, M.G., 2020. Myocardial infarction complications Database. [online] Available at: <https://leicester.figshare.com/articles/dataset/Myocardial_infarction_complications_Database/12045261/3> [Accessed 7 May 2021].

BURNS, E., BUTTNER, R., 2021. LEFT BUNDLE BRANCH BLOCK(LBBB). [online] Available at:<https://litfl.com/left-bundle-branch-block-lbbb-ecg-library/> [Accessed 7 May 2021].

WORLD HEALTH ORGANIZATION. (2021). Cardiovascular diseases (CVDs), World Health Organization (WHO).

EVRIYANTO, Y., 2004. Thesis projects: PERBANDINGAN METODE IMPUTASI UNTUK MENGESTIMASI DATA HILANG PADA DATA KESEHATAN IBU DAN ANAK DI JAWA TIMUR. Surabaya: Perpustakaan Universitas Airlangga.

RUMAGIT, R.Y., 2019. Imbalanced Dataset. [online] Available at: <https://socs.binus.ac.id/2019/12/26/imbalanced-dataset/> [Accessed 7 May 2021].

SARTONO, B., 2018. Oversampling dan Undersampling. [online] Available at: <http://bagusco.staff.ipb.ac.id/2018/01/08/kelas-tidak-seimbang-part2/> [Accessed 7 May 2021].

ARTHANA, R., 2019. Mengenal Accuracy, Precision, Recall dan Specificity serta yang diprioritaskan dalam Machine Learning. [online] Available at: <https://rey1024.medium.com/mengenal-accuracy-precission-recall-dan-specificity-serta-yang-diprioritaskan-b79ff4d77de8> [Accessed 7 May 2021].

DOI: http://dx.doi.org/10.51402/jle.v2i2.48
p- ISSN 2746-3478 e- ISSN 2746-376X

NORAZIAN, M.N., SHUKRI, Y.A., AZAM, R.N., AL BAKRI,A.M.M., 2008. Estimation of missing values in air pollution data using single imputation techniques. [e-journal] 34(2). p.343. Available at: <http://103.86.130.60/handle/123456789/6612> [Accessed 7 May 2021].

BROWNLEE, J., 2019. SMOTE for Imbalanced Classification with Python. [online] Available at: <https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/> [Accessed 7 May 2021].

FADLI, R., 2021. Serangan Jantung. [online] Available at: < https://www.halodoc.com/kesehatan/serangan-jantung> [Accessed 7 May 2021].

PANE, M.D.C., 2020. Syok Kardiogenik. [online] Available at: < https://www.alodokter.com/syok-kardiogenik> [Accessed 7 May 2021].