

Implementasi Algoritma *Logistic Regression* pada Pembuatan Website Sederhana untuk Prediksi Penyakit Jantung

Chyntia Raras Ajeng Widiawati^{1*}, Lisa Nurazizah², Ika Romadoni Yunita³

¹Program Studi Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto

²Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto

³Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto

^{1,2,3}Jln. Let. Jend. Pol. Soemarto Purwokerto, Indonesia

E-mail: chyntiaraw@amikompurwokerto.ac.id¹, lisanr280@gmail.com², ikarom@amikompurwokerto.ac.id³

Info Naskah:

Naskah masuk: 12 Agustus 2023

Direvisi: 27 Desember 2023

Diterima: 2 Januari 2024

Abstrak

Penyakit jantung merupakan salah satu penyakit yang mematikan, pengenalan dini menjadi penting untuk mencegah angka kematian yang cukup tinggi karena penyakit ini. Terdapat berbagai cara untuk deteksi dini penyakit jantung, salah satunya adalah dengan memanfaatkan *machine learning*. Dalam penelitian ini, penulis menggunakan data sekunder yaitu data yang diambil dari website www.kaggle.com untuk proses prediksi. Jumlah data yang digunakan sebanyak 297 data dengan rincian sebanyak 160 data tidak terdeteksi penyakit jantung, dan 137 data terdeteksi penyakit jantung. Selain melakukan prediksi dari data pasien penyakit jantung dengan menggunakan algoritma *logistic regression*, penelitian ini juga melakukan implementasi dari model yang telah dibuat ke dalam website. Hasil dari implementasi algoritma *logistic regression* pada penelitian ini adalah nilai akurasi sebesar 0.9, presisi sebesar 0.92, recall sebesar 0.86, dan f1-score sebesar 0.89. Setelah dilakukan pengukuran menggunakan 4 parameter tersebut, model yang telah dibuat kemudian diimplementasikan ke dalam website sederhana dengan menggunakan metode *Rapid Application Development* (RAD).

Keywords:

heart disease;

machine learning;

logistic regression;

website.

Abstract

Heart disease is a deadly disease, early recognition is important to prevent the fairly high death rate due to this disease. There are various ways to detect heart disease early, one of which is by utilizing machine learning. In this research, the author uses secondary data, namely data taken from the website www.kaggle.com for the prediction process. The amount of data used was 297 data, with details of 160 data not detecting heart disease, and 137 data detecting heart disease. Apart from making predictions from heart disease patient data using the logistic regression algorithm, this research also implements the model that has been created on the website. The results of implementing the logistic regression algorithm in this research are an accuracy value of 0.9, precision of 0.92, recall of 0.86, and f1-score of 0.89. After measuring using these 4 parameters, the model that has been created is then implemented into a simple website using the *Rapid Application Development* (RAD) method.

*Penulis korespondensi:

Chyntia Raras Ajeng Widiawati

E-mail: chyntiaraw@amikompurwokerto.ac.id

1. Pendahuluan

Penyakit jantung merupakan salah satu penyakit yang dapat mengakibatkan kematian dengan peringkat nomor satu di dunia. Berdasarkan *Global Burden of Disease dan Institute for Health Metrics and Evaluation (IHME) 2014-2019*, penyebab kematian tertinggi yang terjadi di Indonesia adalah penyakit jantung. Berdasarkan data yang diperoleh dari Riset Kesehatan Dasar (Riskedas) pada tahun 2013 tingkat penyakit jantung yang semula berada pada kisaran 0,5% mengalami peningkatan yang signifikan pada tahun 2018 menjadi 1,5% [1]. Penyakit ini dapat dicegah dengan deteksi dini dan pengendalian risiko, deteksi dini penyakit jantung terlebih jika ada bawaan menjadi penting karena presentasi klinis dan kemunduran kondisi umum yang mendadak dapat menimbulkan risiko morbiditas dan mortalitas yang sebenarnya dapat dihindari [2].

Kurangnya pengenalan penyakit jantung dan penerapan pola hidup sehat membuat masyarakat tidak mengetahui gejala awalnya. Pendeteksian penyakit jantung selama ini dilakukan secara manual dengan pergi ke rumah sakit dan konsultasi ke dokter, lalu akan diarahkan untuk melakukan pemeriksaan laboratorium untuk kembali diperiksa dokter kembali. Tentu pemeriksaan untuk pendeteksian ini harus dilakukan secara rutin, agar tentunya penyakit jantung dapat dicegah atau ditangani sedari awal. Hal ini pastinya memerlukan biaya yang relatif besar. Untuk itu, perlu adanya sistem yang dapat mendeteksi penyakit jantung bagi masyarakat secara akurat dan lebih efisien biaya [3].

Dengan kemajuan teknologi, angka kematian pasien karena penyakit jantung dapat diminimalisir. Machine learning (ML) dapat menjadi solusi untuk membantu tenaga kesehatan dalam membuat keputusan tepat terhadap hasil diagnosa pasien [4]. *Machine learning* juga merupakan teknik yang mengubah data mentah menjadi data informasional yang membantu mengambil keputusan dan prediksi [5]. Penggunaan *Machine learning* bukan berarti akan memberikan hasil yang akurat 100% dalam analisa data dan menghasilkan kesimpulan berdasarkan analisa data tersebut. Namun akurasi yang dihasilkan cukup efektif diberbagai kasus identifikasi penyakit jantung sehingga *machine learning* telah terbukti sangat membantu di bidang kesehatan [6].

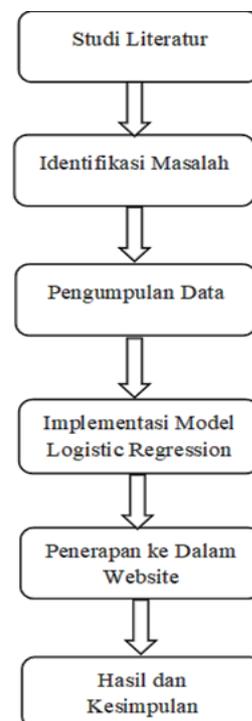
Terdapat beberapa penelitian sebelumnya terkait prediksi penyakit jantung menggunakan *machine learning*. Salah satunya oleh [4] merupakan perbandingan algoritma untuk prediksi penyakit jantung yang menggunakan dataset publik bersumber dari *UCI Machine learning*. Algoritma yang dibandingkan diantaranya *Support Vector Machine (SVM)*, *Logistic regression*, dan *Artificial Neural Network (ANN)* melalui uji komposisi pembagian data *training* dan *testing* 90:10, 80:20, 70:30, dan 60:40. Pada penelitian tersebut, dihasilkan akurasi tertinggi adalah ketika pembagian data 80:20 dengan algoritma *logistic regression*, yaitu sebesar 86%. Dan pada penelitian [7] terdapat beberapa metode yang digunakan antara lain *logistic regression* dengan akurasi 86,5%, *Random Forest classifier* dengan akurasi 80,9% *Naïve Bayes* dengan akurasi 79,8%. Berdasarkan nilai akurasi yang diperoleh dapat dilihat bahwa algoritma *Logistic regression* memiliki akurasi tertinggi.

Dari beberapa penelitian sebelumnya dapat dilihat bahwa terdapat beberapa algoritma yang memberikan performa cukup baik pada prediksi penyakit jantung. Namun, penelitian sebelumnya hanya melakukan eksperimen dan pemodelan terhadap algoritma yang digunakan dan belum diimplementasikan secara langsung pada website maupun media lainnya. Oleh karena itu, penelitian ini melakukan penerapan algoritma *logistic regression* pada prediksi penyakit jantung dengan menggunakan website sebagai aplikasi sarana. Dengan adanya aplikasi ini, nantinya masyarakat umum yang rutin melakukan *checkup* kesehatan akan dipermudah dalam melakukan deteksi dini terhadap penyakit jantung.

Penggunaan website untuk aplikasi ini dikarenakan beberapa keunggulan website yang dapat memberikan ketersediaan informasi menjadi lebih efisien dan *up to date*. Website mudah diakses oleh seluruh masyarakat di berbagai daerah dengan memanfaatkan koneksi internet. Contohnya, website digunakan pada beberapa hal seperti media pemasaran, informasi pendidikan, komunikasi dan promosi [8]. Sehingga pada penelitian ini dilakukan Implementasi Algoritma *Logistic Regression* untuk Prediksi Penyakit Jantung dalam tampilan website sederhana.

2. Metode

Metode yang dilakukan pada penelitian dipaparkan pada Gambar 1. Alur penelitian yang dipaparkan pada Gambar 1 memiliki penjelasan untuk setiap tahapan sebagai berikut:



Gambar 1. Alur Penelitian

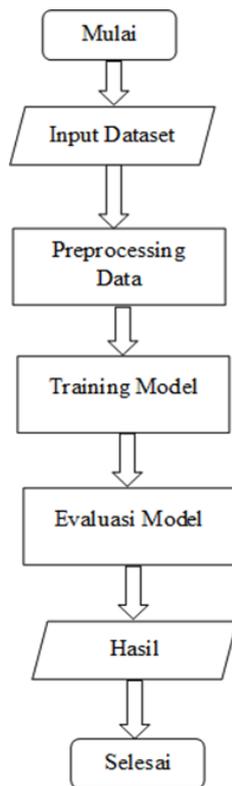
2.1 Pengumpulan Data

Data sekunder yang digunakan pada penelitian ini diperoleh dari Kaggle yang dapat diakses pada link berikut ini <https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci>.

2.2 Implementasi Model Logistic Regression

Tahap selanjutnya yaitu implementasi model dari algoritma *logistic regression*. Model *logistic regression* merupakan salah satu model yang digunakan dalam *machine learning* untuk mengetahui pengaruh variabel independen (X) terhadap variabel dependen (Y) dengan variabel dependen merupakan nilai biner 1 menyatakan variabel tersebut memiliki kriteria yang sesuai, dan nilai 0 adalah menyatakan variabel tersebut tidak memiliki kriteria yang sesuai [6].

Pada tahap implementasi Model *Logistic Regression* terdapat beberapa proses didalamnya seperti yang ditunjukkan pada Gambar 2.



Gambar 2. Implementasi Logistic Regression

Berikut penjelasan proses yang dilakukan pada implementasi model pada Gambar 2.

- Input Dataset:** Pada tahap ini dilakukan input dataset yang akan digunakan, yang sebelumnya telah didownload pada link <https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci> yaitu dataset penyakit jantung yang memiliki 13 kolom features dan 1 kolom label.
- Pre-processing data:** Pada tahap ini dilakukan dua tahap, diantaranya yaitu cek data duplikat (data ganda) dan cek *missing value* (data kosong). Selanjutnya akan dilakukan seleksi antara *features* dan label pada dataset yang telah dimasukkan. Berikutnya adalah pemisahan data *training* dan data *testing*, dalam hal ini penulis akan membagi dataset dengan perbandingan 80:20.
- Training Model:** Dalam penelitian ini, dilakukan teknik *random sampling* untuk membagi data *training* dan *testing* menggunakan *library* dari *scikit-learn*.

Kemudian dataset yang sudah dibagi menjadi data pelatihan (*training*) dan data percobaan (*testing*) diterapkan ke model.

- Evaluasi Model:** data *training* yang sudah dilakukan *training*, selanjutnya adalah evaluasi model untuk menghitung akurasi algoritma. Dalam evaluasi, data *testing* yang akan dihitung adalah akurasi, presisi, *recall*, dan *f1 score* nya.

Akurasi merupakan parameter pengukuran yang digunakan untuk melihat performa model dalam melakukan klasifikasi kelas yang benar [9]. Presisi merupakan parameter pengukuran yang digunakan untuk mengetahui perbandingan dari prediksi benar positif dibagi dengan keseluruhan hasil yang diprediksi positif [10]. *Recall* adalah parameter pengukuran yang digunakan untuk melihat rasio perhitungan dari jumlah data prediksi benar positif terhadap hasil benar positif keseluruhan [11]. *F1-Score* adalah parameter pengukuran untuk mengetahui rata-rata perbandingan dari nilai presisi dengan nilai *recall* yang telah diperoleh [12]. Akurasi dapat dihitung menggunakan persamaan (1), presisi dapat dihitung menggunakan persamaan (2), *recall* dapat dihitung menggunakan persamaan (3), dan *F1 score* dapat dihitung menggunakan persamaan (4).

$$\text{Akurasi} = \frac{TP + TN}{FP + FN + TP + TN} \quad (1)$$

$$\text{Presisi} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{F1 - Score} = 2 \times \frac{\text{Recall} \times \text{Presisi}}{\text{Recall} + \text{Presisi}} \quad (4)$$

Keterangan:

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

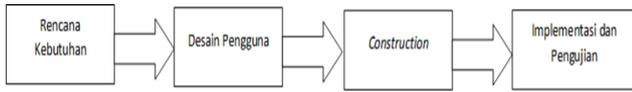
Dari persamaan (1), (2), dan (3) Nilai-nilai TP, TN, FP, dan FN tersebut akan didapat pada saat setelah dilakukan proses data *training* dan *testing*. Sedangkan pada persamaan (4) maka bergantung pada persamaan (2) dan (3).

2.3 Penerapan ke Website

Hasil prediksi yang sudah dilakukan di aplikasi *google colab*, selanjutnya diolah dan diterapkan dalam website sederhana menggunakan *framework flask* yang merupakan *framework* yang disediakan oleh *python*. Pada tahap ini, penulis menggunakan metode *Rapid Application Development (RAD)*. Metode ini merupakan salah satu model dari *System Development Life Cycle (SDLC)*. RAD merupakan metode yang menekankan pada siklus pengembangan yang sangat singkat dan cepat, serta proses pengembangan perangkat lunak secara *linear sequential* [13].

Alasan menggunakan metode RAD ini adalah memiliki beberapa kelebihan, diantaranya memerlukan waktu yang

singkat dalam pengerjaan, menghasilkan sistem yang memenuhi kebutuhan langsung, serta waktu pengerjaan yang fleksibel [14].



Gambar 3. Metode Perancangan RAD

Berikut penjelasan proses yang dilakukan pada implementasi model pada Gambar 3.

- Rencana Kebutuhan:** Pada tahap ini, dilakukan analisis kebutuhan baik fungsional maupun non fungsional dalam pembuatan website, karena tahap ini adalah awal keberhasilan pembuatan sistem yang tepat sasaran untuk kebutuhan.
- Desain Pengguna:** Tahap ini membuat rancangan yang sesuai dengan kebutuhan, diharapkan dapat mengatasi masalah yang terjadi. Desain sistem yang dibuat menggunakan *Unified Modeling Language (UML)*.
- Construction:** Tahap ini adalah tahap memulai pembuatan website dengan menyusun suatu kode program atau *coding*, menerapkan desain yang telah dibuat ke dalam aplikasi yang telah direncanakan sebelumnya.
- Implementasi dan Pengujian:** Pada tahap ini dilakukan implementasi terhadap semua tahapan sebelumnya, yaitu rencana kebutuhan, desain, dan konstruksi. Kemudian pengujian dilakukan pada keseluruhan sistem yang dibangun, semua komponen harus diuji dengan *Black Box Testing* agar mengurangi risiko cacat sistem.

3. Hasil dan Pembahasan

3.1 Pengumpulan Data

Pada penelitian ini, penulis mengambil data sekunder dari *website* penyedia data *kaggle* dengan judul *Heart Disease Cleveland UCI* [15]. Data tersebut terdiri dari 297 baris dan 13 *features*/atribut seperti yang ditunjukkan pada Tabel 1.

Tabel 1. Atribut Dataset

No	Atribut	Deskripsi	Keterangan
1	Age	Usia dalam tahun	Numerik
2	Sex	Jenis Kelamin Pasien	0=Perempuan, 1= Laki-laki
3	Cp	Jenis Nyeri Dada	1= Angina Khas, 2= angina atipikal, 3= Nyeri non-angina, 4= Asintomatik
4	Trestbps	Tekanan Darah saat istirahat	Numerik dengan satuan mm/Hg
5	Chol	Kolestrol Serum	Numerik dengan satuan mg/dl
6	Fbs	Gula Darah saat istirahat kurang/lebih dari 120mg/dl	1= Benar, 0= salah
7	Restecg	Elektrokardiografi saat istirahat	0= normal, 1= kelainan, 2= mungkin atau pasti

No	Atribut	Deskripsi	Keterangan
8	Thalach	Detak jantung maksimum	hipertrofi ventrikel kiri Numerik
9	Exang	Ukuran apakah nyeri dada saat berolahraga	1= ya, 0= tidak
10	Oldpeak	Depresi ST yang diakibatkan latihan relative saat istirahat	Numerik
11	Slope	Kemiringan dari puncak ST setelah berolahraga	0= condong ke atas, 1= datar, 2= sedikit landai ke bawah
12	Ca	Banyaknya nadi utama	0-3 berdasarkan warna oleh flouroscopy
13	Thal	Kondisi jantung pasien	0= normal, 1= cacat tetap, 2= cacat sementara

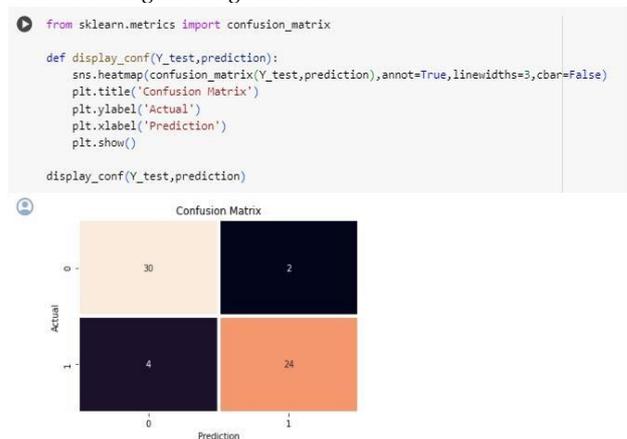
Data yang digunakan pada penelitian ini memiliki rincian sebanyak 160 data tidak menderita penyakit jantung dan sebanyak 137 menderita penyakit jantung. Pada data ini memiliki atribut sebanyak 13 dan memiliki satu kolom target sebagai kelas yang terdapat dua kelas.

3.2 Implementasi Model Logistic Regression

Tahap selanjutnya yaitu penjabaran pada implementasi dataset yang telah diperoleh ke model *logistic regression* yang terdiri dari beberapa sub bab berikut ini.

- Input Dataset:** Berikut ini adalah sampel dataset yang telah didownload ditunjukkan pada Tabel 2.
- Pre-processing data:** Di dalam *preprocessing* data dilakukan pengecekan *missing value* dan data duplikat yang mana hasilnya adalah tidak terdapat *missing value* dan data duplikat. Sehingga seluruh data digunakan pada penelitian ini.
- Training Model:** Pada tahap ini data dilatih menggunakan model algoritma *logistic regression*. Proses *training* data dilakukan pada 237 data yang merupakan 80% dari data *training*.

Pada gambar 4 adalah *confusion matrix* hasil rincian evaluasi model oleh *logistic regression*.



Gambar 4. Hasil Confusion Matrix dari Logistic Regression

Tabel 2. Sampel dataset

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
69	1	0	160	234	1	2	131	0	0.1	1	1	0	0
69	0	0	140	239	0	0	151	0	1.8	0	2	0	0
66	0	0	150	226	0	0	114	0	2.6	2	0	0	0
65	1	0	138	282	1	2	174	0	1.4	1	1	0	1
64	1	0	110	211	0	2	144	1	1.8	1	0	0	0
..
..
40	1	3	152	223	0	0	181	0	0	0	0	2	1
39	1	3	118	219	0	0	140	0	1.2	1	0	2	1
35	1	3	120	198	0	0	130	1	1.6	1	0	2	1
35	0	3	138	183	0	0	182	0	1.4	0	0	0	0
35	1	3	126	282	0	2	156	1	0	0	0	2	1

Berdasarkan Gambar 3 diperoleh nilai TP sebanyak 24, TN sebanyak 30, FP sebanyak 2 dan FN sebanyak 4. Dari data data diatas selanjutnya dihitung akurasi, presisi, *recall*, dan *F1-score* untuk model *logistic regression* dengan hasil sebagai berikut:

$$\text{Akurasi} = \frac{24 + 30}{2 + 4 + 24 + 30} = 0.9$$

$$\text{Presisi} = \frac{24}{24 + 2} = 0.9230$$

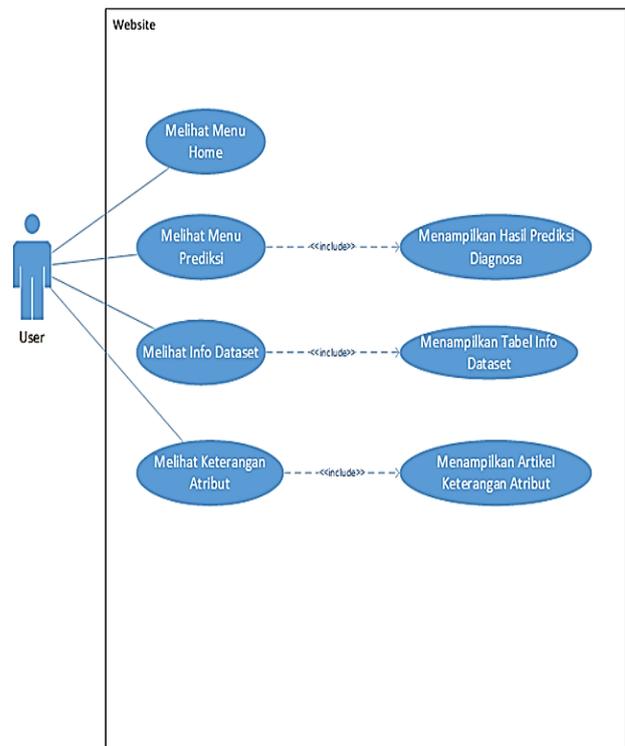
$$\text{Recall} = \frac{24}{24 + 4} = 0.8571$$

$$\text{F1-Score} = 2 \times \frac{0.92 \times 0.86}{0.92 + 0.86} = 0.8889$$

Dari persamaan perhitungan di atas pada persamaan (5) diperoleh nilai akurasi sebesar 0.9 yang artinya kemampuan model yang dibangun untuk mengenali kelas positif dan negatif sudah baik yaitu dengan tingkat akurasi 90%. Pada persamaan (6), (7), dan (8) menghitung *target* kelas 1 (kelas positif jantung) dengan nilai presisi sebesar 0.9230 yang artinya rasio prediksi benar terkena penyakit jantung dibandingkan dengan keseluruhan hasil yang diprediksi sebagai pasien penyakit jantung sebesar 92,30%. Kemudian *recall* sebesar 0.8571 yang artinya rasio prediksi benar terkena penyakit jantung dibandingkan dengan keseluruhan data yang benar merupakan pasien penyakit jantung sebesar 85,71%. Sedangkan *F1-score* sebesar 0.8889 yang merupakan perbandingan antara nilai presisi dengan recall.

3.3 Penerapan ke Website

Hasil dari model yang sudah dilatih dan dievaluasi, kemudian diterapkan ke dalam sistem informasi sederhana berbasis *website*. Berikut merupakan *use case* dari aplikasi.



Gambar 5. Use Case Diagram Aplikasi

Dari *use case* diagram di atas, dapat dilihat bahwa terdapat menu *home* yang merupakan menu awal ketika membuka *website*. Pengguna dapat mengakses menu lainnya dari menu *home*. Lalu terdapat menu prediksi untuk mendapatkan hasil diagnosis prediksi. *Use case* dimulai dari pengguna mengisi 13 atribut yang berada pada menu prediksi, kemudian aplikasi akan menampilkan hasil diagnosis sesuai dengan atribut yang sudah diisi. Selanjutnya ada menu informasi detail dataset berisi informasi terkait atribut pada dataset yang ada di menu dataset dalam bentuk tabel. Menu selanjutnya adalah menu keterangan atribut yang berisi informasi istilah bagi pengguna yang belum

pernah *checkup* kesehatan jantung berisi informasi mengenai kolom atribut dengan bentuk artikel.

Dari model perancangan tersebut menghasilkan tampilan *website* sederhana dengan menu prediksi seperti yang ditunjukkan pada Gambar 6.

Gambar 6. Menu Prediksi

4. Kesimpulan

Penelitian ini menghasilkan beberapa kesimpulan sebagai berikut: hasil performa algoritma *logistic regression* memiliki perhitungan keakuratan model algoritma dalam klasifikasi yang benar sebesar 0.9. Nilai presisi atau rasio perbandingan antara yang benar terdeteksi penyakit jantung dibagi dengan keseluruhan yang diprediksi terdeteksi penyakit jantung sebesar 0.92. Nilai *recall* atau rasio perbandingan antara benar terdeteksi penyakit jantung dengan keseluruhan data yang benar terdeteksi penyakit jantung sebesar 0.86. Nilai *f1-score* atau hasil perbandingan rata-rata presisi dan *recall* yang dibobotkan sebesar 0.89.

Sebagai pertimbangan pada pengembangan penelitian selanjutnya, terdapat beberapa saran yang bisa dilakukan, yaitu bekerja sama dengan rumah sakit untuk mendapatkan data asli pasien penyakit jantung sehingga hasil yang diperoleh lebih valid dan sebagai pengujian aplikasi. Mengembangkan aplikasi dengan berbasis *mobile* yang fleksibel digunakan dalam sehari-hari

Ucapan Terimakasih

Terima kasih kepada Universitas Amikom Purwokerto yang telah memberikan dukungan terhadap terlaksananya penelitian ini.

Daftar Pustaka

- [1] A. Purnama, "Edukasi dapat meningkatkan kualitas hidup pasien yang terdiagnosa penyakit jantung koroner," *J. Kesehat. Indones.*, vol. X, no. 2, pp. 66–71, 2020.
- [2] E. M. Marwali, Y. Purnama, and P. S. Roebiono, "Modalitas Deteksi Dini Penyakit Jantung Bawaan di Pelayanan Kesehatan Primer," *J. Indones. Med. Assoc.*, vol. 71, no. 2, pp. 100–109, 2021, doi: 10.47830/jinma-vol.71.2-2021-241.
- [3] A. B. Wibisono and A. Fahrurrozi, "Perbandingan Algoritma Klasifikasi Dalam Pengklasifikasian Data Penyakit Jantung Koroner," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 3, pp. 161–170, 2019, doi: 10.35760/tr.2019.v24i3.2393.
- [4] F. Handayani, "Komparasi Support Vector Machine, Logistic Regression Dan Artificial Neural Network Dalam Prediksi Penyakit Jantung," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 3, p. 329, 2021, doi: 10.26418/jp.v7i3.48053.
- [5] Balakrishnan, M., Arockia Christopher, A. B., Ramprakash, P., & Logeswari, A. (2021). Prediction of Cardiovascular Disease using Machine Learning. *Journal of Physics: Conference Series*, 1767(1). <https://doi.org/10.1088/1742-6596/1767/1/012013>
- [6] J. J. Pangaribuan, H. Tanjaya, and Kenichi, "Mendeteksi Penyakit Jantung Menggunakan Mechine Learning Dengan Algoritma Logistic Regression," *Mach. Learn.*, vol. 45, no. 13, pp. 40–48, 2021.
- [7] M. Balakrishnan, A. B. Arockia Christopher, P. Ramprakash, and A. Logeswari, "Prediction of Cardiovascular Disease using Machine Learning," *J. Phys. Conf. Ser.*, vol. 1767, no. 1, 2021, doi: 10.1088/1742-6596/1767/1/012013.
- [8] P. S. Hasugian, "Perancangan Website Sebagai Media Promosi Dan Informasi," *J. Inform. Pelita Nusant.*, vol. 3, no. 1, pp. 82–86, 2018.
- [9] Mohan, S., Thirumalai, C., & Srivastava, G. (2019). Effective heart disease prediction using hybrid machine learning techniques. *IEEE Access*, 7, 81542–81554. <https://doi.org/10.1109/ACCESS.2019.2923707>
- [10] Derisma, D. (2020). Perbandingan Kinerja Algoritma untuk Prediksi Penyakit Jantung dengan Teknik Data Mining. *Journal of Applied Informatics and Computing*, 4(1), 84–88. <https://doi.org/10.30871/jaic.v4i1.2152>
- [11] Terrada, O., Cherradi, B., Raihani, A., & Bouattane, O. (2019). Classification and Prediction of atherosclerosis diseases using machine learning algorithms. 2019 International Conference on Optimization and Applications, ICOA 2019, April, 1–5. <https://doi.org/10.1109/ICOA.2019.8727688>
- [12] Santoso, B. (2019). An Analysis of Spam Email Detection Performance Assessment Using Machine Learning. *Jurnal Online Informatika*, 4(1), 53. <https://doi.org/10.15575/join.v4i1.298>
- [13] N. Hidayat and K. Hati, "Penerapan Metode Rapid Application Development (RAD) dalam Rancang Bangun Sistem Informasi Rapor Online (SIRALINE)," *J. Sist. Inf.*, vol. 10, no. 1, pp. 8–17, 2021, doi: 10.51998/jsi.v10i1.352.
- [14] T. Pricillia and Zulfachmi, "Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, RAD)," *J. Bangkit Indones.*, vol. 10, no. 1, pp. 6–12, 2021, doi: 10.52771/bangkitindonesia.v10i1.153.
- [15] Cherngs, "Heart Disease Cleveland UCI," *Kaggle*, 2019. <https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci>.