

BAB II TINJAUAN PUSTAKA

Bab ini merupakan acuan studi pustaka yang digunakan dalam perancangan kebaruan penelitian. Dijelaskan mengenai pencapaian terdahulu dan tinjauan teoritis terkait penelitian tentang hipertensi dan machine learning. Lebih jelas dipaparkan pada sub-bab berikut.

2.1 Pencapaian Terdahulu

Penelitian ini mengacu pada pencapaian terdahulu yang telah dilakukan oleh peneliti-peneliti sebelum ini. Referensi yang didapatkan digunakan untuk acuan studi literatur dan pembandingan kebaruan pengujian algoritma. Penelitian terdahulu yang dibutuhkan dalam penelitian ini merupakan hasil penelitian tentang pengujian *machine learning* menggunakan algoritma KNN dan SVM pada kasus yang berbeda. Kasus yang diteliti beragam dengan hasil pengukuran akurasi yang juga bermacam-macam.

Pertama, referensi penelitian ini didasarkan pada jurnal oleh Rana, dkk. (2021) tentang penelitian untuk pengujian *machine learning* berupa *A Neural Network* untuk mendiagnosis kemungkinan stroke. Hal ini ditindaklanjuti dengan membuat sebuah pengujian untuk mendeteksi gejala yang dapat menimbulkan potensi stroke agar dapat segera ditangani lebih lanjut untuk mencegah kematian. Pada penelitiannya, Rana, dkk. melakukan pengambilan 5110 dataset pasien yang diambil dari Kaggle.com oleh Fedeoriano. Pengujian *machine learning A Neural Network* ini menghasilkan skor berdasarkan parameter ROC sebesar 0.84. Dengan demikian, dapat ditarik kesimpulan bahwa *machine learning* tersebut dapat diterapkan untuk mendiagnosis pasien berpotensi mengidap penyakit stroke atau tidak (Rana, et al., 2021).

Penelitian yang kedua, didapatkan referensi dari jurnal oleh Syailasya dan Kumari (2021). Pada penelitian tersebut, dilakukan pengujian untuk memprediksi potensi stroke menggunakan algoritma *machine learning* berupa *Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors, Support Vector Machine*, dan Klasifikasi *Naïve Bayes*. Pemodelan *machine learning* pada dataset

Kaggle oleh Fedesoriano tersebut didapatkan hasil ROC prediksi stroke berturut-turut sebesar 0.78, 0.66, 0.73, 0.80, 0.80, dan 0.82 (Sailasya & Kumari, 2021).

Referensi ketiga didapat dari jurnal oleh Leha, dkk. pada tahun 2019. Penelitian yang dilakukan adalah mengenai pengujian beberapa *machine learning* untuk memprediksi potensi hipertensi pulmonal (tekanan darah tinggi pada pembuluh darah di arteri paru-paru). Hasil akurasi AUC yang didapatkan pada pengujian *machine learning* tersebut cukup bervariasi. Pengujian *Support Vector Machine*, *Boosted Classification Trees*, *Lasso Penalized Logistic Regression*, *Random Forest Classification*, dan *Random Forest Regression* menghasilkan akurasi AUC berturut-turut sebesar 0.83, 0.80, 0.78, 0.85, dan 0.87 (Leha, et al., 2019).

Referensi keempat didapat dari jurnal ilmiah berdasarkan data medis untuk memprediksi penyakit hipertensi berdasarkan algoritma *machine learning* oleh Chang, dkk. tahun 2019. Disebutkan bahwa hipertensi bisa menyebabkan kematian, sehingga dilakukan pengembangan pengujian untuk memprediksi potensi penyakit tersebut agar dapat dilakukan pencegahan bagi pasien-pasien berikutnya. Dalam penelitian tersebut, dilakukan pengujian untuk algoritma *Support Vector Machine*, *Decision Tree*, *Random Forest*, dan *XGBoost*. Berdasarkan pengujian yang dilakukan, hasil akurasi algoritma tersebut diukur menggunakan parameter AUC berturut-turut adalah 75.80%, 86.03%, 88.98%, dan 94.36% (Chang, et al., 2019).

Untuk memudahkan dalam pembacaan referensi terdahulu, Di bawah ini, disajikan hasil ringkasan beberapa pemodelan terkait topik hipertensi dari studi literasi yang didapatkan dalam bentuk Tabel 2.1 berikut ini.

Tabel 2. 1 Perbandingan Penelitian Terdahulu

Penelitian Terdahulu	Metode	Hasil Akurasi
“Stroke Prediction Using Smote-Tomek and Neural Network”, oleh Rana, 2021.	<i>A Neural Network</i>	ROC 0.84.
“Analyzing the Performance of Stroke Prediction using ML Classification Algorithms”, oleh Sailasya & Kumari, 2021.	<i>Logistic Regression</i> , <i>Decision Tree</i> , <i>Random Forest</i> , <i>K-Nearest Neighbors</i> , <i>Support Vector Machine</i> , dan <i>Klasifikasi Naïve Bayes</i> .	ROC 0.78, 0.66, 0.73, 0.80, 0.80, dan 0.82.

A Machine Learning Approach for The Prediction of Pulmonary Hypertension, oleh Leha, dkk., 2019	<i>Support Vector Machine, Boosted Classification Trees, Lasso Penalized Logistic Regression, Random Forest Classification, dan Random Forest Regression.</i>	Berturut-turut 0.83, 0.80, 0.78, 0.85, dan 0.87.
A Machine-Learning-Based Prediction Method for Hypertension Outcomes Based on Medical Data, oleh Chang dkk. 2019.	<i>Support Vector Machine, Decission Tree, Random Forest, dan XGBoost.</i>	Berturut-turut 75.80%, 86.03%, 88.98%, dan 94.36%

Dari keempat pustaka yang peneliti tinjau tersebut, jenis algoritma yang digunakan untuk memprediksi kasus yang diangkat sebagai topik pembahasan adalah KNN dan SVM. Hasil akurasi yang beragam menjadi motivasi peneliti untuk membandingkan kedua algoritma tersebut ke dalam suatu kasus untuk penelitian. Sehingga, *machine learning* dengan menerapkan algoritma KNN dan SVM dapat digunakan sebagai pengujian untuk memprediksi berbagai potensi penyakit bahkan yang tanpa gejala. Dalam penelitian ini, KNN dan SVM dapat digunakan sebagai algoritma untuk prediksi potensi penyakit hipertensi.

2.2 Tinjauan Teoritis

Aspek-aspek teori maupun aspek-aspek teknis terapan yang berkaitan erat dengan pelaksanaan penelitian dipaparkan pada sub-bab ini. Teori dibagi menjadi beberapa bagian sub-sub-bab yang diangkat dari dokumen-dokumen, jurnal, atau artikel ilmiah terkait topik yang dibahas.

2.2.1 Hipertensi

Hipertensi adalah salah satu pengaruh dari kegagalan fungsi kerja jantung ketika tekanan pada jantung bekerja lebih cepat dari tekanan normal. Hipertensi merupakan salah satu penyakit yang tidak memunculkan gejala pada penderita sehingga sering disebut "*The Silent Killer*" (Tyas & Mukhofi, 2016, p. 1). Gejala

hipertensi tidak dapat dirasakan oleh penderita. Namun, berdasarkan literasi yang peneliti miliki, dapat disimpulkan bahwa potensi hipertensi dapat diprediksi melalui pola rata-rata level gula darah, indeks massa tubuh, dan usia. Kadar gula darah yang terkontrol dapat mencegah risiko munculnya penyakit hipertensi karena tekanan darah dapat dipertahankan dalam *range* normal (Winta, et al., 2018, p. 163). Menurut Riset Kesehatan Dasar tahun 2018, hipertensi juga dipengaruhi oleh angka IMT merujuk pada peningkatan jumlah angka obesitas dari tahun 2013 hingga 2018 yang juga meningkatkan angka penderita hipertensi (Yulnefia, 2020, p. 70). Selain itu, berdasarkan survey yang dilakukan ke 30 responden yang terdiri dari 18 orang dewasa dan 12 remaja, 15 orang dewasa menderita hipertensi dan hanya 2 remaja yang menderita hipertensi (Aristoteles, 2018, p. 14). Sehingga usia juga menjadi indikasi yang memperkuat kemungkinan penyakit hipertensi. Dengan demikian, rata-rata level gula darah, indeks massa tubuh, dan usia cukup untuk menjadi variabel independen dalam penelitian ini guna memprediksi potensi penyakit hipertensi.

2.2.2 Algoritma

Algoritma merupakan suatu alur pemikiran untuk menyelesaikan suatu masalah secara komputasi. Dalam pembuatan suatu alur algoritma, pemikiran memiliki makna yang bermacam-macam namun tetap sistematis (Azzahra, 2020). Alur pemikiran suatu algoritma dituangkan ke dalam kata-kata/kalimat (bahasa natural), gambar atau tabel tertentu, atau berupa kode tertentu. Hal ini ditujukan agar pemikiran tersebut dapat dimengerti oleh pihak pembaca algoritma seperti salah satunya komputer. Komputer bekerja secara logis dan tersistematis membaca perintah berdasarkan algoritma.

Algoritma, lebih lengkapnya, berisi suatu urutan tahapan untuk menyelesaikan suatu permasalahan secara komputasi (Pratiwi, 2017). Berdasarkan pemahaman tersebut, algoritma menjadi tinjauan teoritis peneliti untuk menyelesaikan masalah. Konteks permasalahan diambil dari variabel masukan sehingga menghasilkan keluaran yang dapat dilaksanakan oleh program komputer. Dalam hal ini metode algoritma yang peneliti gunakan untuk *machine learning* adalah *K-Nearest Neighbors* dan *Support Vector Machine*.

2.2.3 Machine Learning

Teknologi machine learning merupakan cara penyelesaian suatu masalah dengan menerapkan metode algoritma dalam perhitungan komputer. Teknologi *machine learning* digunakan untuk menggantikan atau menirukan perilaku manusia untuk menyelesaikan masalah. Sesuai dengan namanya, mesin mencoba untuk belajar menirukan bagaimana proses manusia atau makhluk cerdas belajar dan menggeneralisasi (Ahmad, 2017, p. 2). *K-Nearest Neighbors* dan *Support Vector Machine* merupakan algoritma yang diterapkan ke pemodelan *machine learning*. Peneliti menerapkan *machine learning* tersebut agar memudahkan dalam mengklasifikasikan data uji terhadap data latih pemodelan potensi penyakit hipertensi. Dalam penerapannya, *machine learning* memerlukan kumpulan data, atribut atau fitur, serta model yang merupakan hasil dari *machine learning*.

Secara garis besar, tahapan kerja pengujian machine learning adalah sebagai berikut (Zailani, et al., 2020, p. 9) :

- a. Mengumpulkan data latih dan data uji,
- b. Membuat model masukan data berdasarkan data latih,
- c. Memprediksi berdasarkan klasifikasi data uji ke hasil *machine learning* terhadap data latih,
- d. Mengevaluasi model untuk mengidentifikasi akurasi, lalu
- e. Menyempurnakan model *machine learning* agar dapat diterapkan berkelanjutan.

2.2.4 Data Pre-processing

Kumpulan data latih yang didapat tidak selalu ideal dan merupakan data mentah. Sehingga, menggunakan data *pre-processing* dapat memperbaiki format data dan meminimalisasi *noise* pada data latih (Gunawan, 2016). Pada penelitian ini, Data *Pre-Processing* yang diterapkan berupa normalisasi data dan seleksi fitur. Dilakukan normalisasi data menggunakan *Min-Max Scaler*. Proses *Min-Max Scaler* pada umumnya bekerja dengan merubah rentang nilai dalam data menjadi antara 0 sampai 1. Selain itu, dilakukan juga seleksi fitur dengan mengeliminasi variabel yang tidak diperlukan dalam pengujian. Pada penelitian ini, peneliti hanya mengambil variabel berupa angka rata-rata glukosa, indeks massa tubuh, dan usia

dari kumpulan set data yang telah dikumpulkan. Diperlukan rumus untuk melakukan *pre-processing* data. Persamaan dalam normalisasi data menggunakan *Min-Max Scaler* adalah sebagai berikut :

$$Newdata = \frac{(x - min)}{max - min}$$

Keterangan :

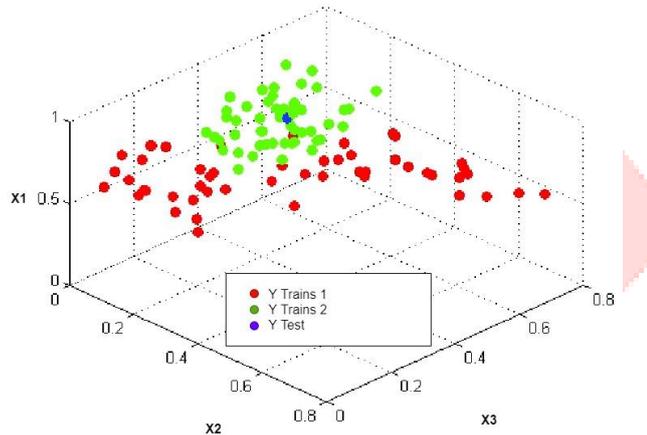
Newdata = Data hasil normalisasi

x = Nilai data

min = Nilai minimum data dalam satu kolom

max = Nilai maksimum data dalam satu kolom

2.2.5 *K-Nearest Neighbors* (KNN)



Gambar 2. 1 Ilustrasi KNN 3 dimensi (Garcia, et al., 2019)

Algoritma *K-Nearest Neighbors* (KNN) merupakan algoritma yang umum digunakan untuk mengklasifikasikan objek ke dalam data berdasarkan jarak terdekat objek ke data latih (Rivki & Bachtiar, 2017, p. 33). KNN sering kali digunakan dalam penerapan *machine learning* untuk pemecahan masalah klasifikasi atau prediksi. KNN termasuk dalam *instance-based learning*. KNN bekerja dengan mencari kelompok *k* objek di dalam data latih yang terdekat atau memiliki kemiripan dengan objek baru atau data latih. Pada penelitian ini, sistem klasifikasi algoritma KNN diperlukan untuk mengklasifikasikan pasien baru dengan pasien lama untuk memprediksikan apakah pasien baru tersebut menderita penyakit hipertensi.

Algoritma *K-Nearest Neighbors* mengklasifikasi kumpulan data berdasarkan data yang sudah terklasifikasi. KNN bekerja berdasarkan jarak terpendek dari sample uji ke sample latih hingga dapat ditentukan KNN. Setelah data terkumpul, KNN mengambil mayoritas data untuk dijadikan prediksi sample uji. Jauh atau dekat ketetanggaan dihitung berdasarkan jarak *euclidean*. Tahapan menghitung algoritma KNN adalah sebagai berikut :

- a. Menentukan parameter k ,
- b. Menghitung jarak Euclidean antara data latih dan data uji menggunakan rumus berikut :

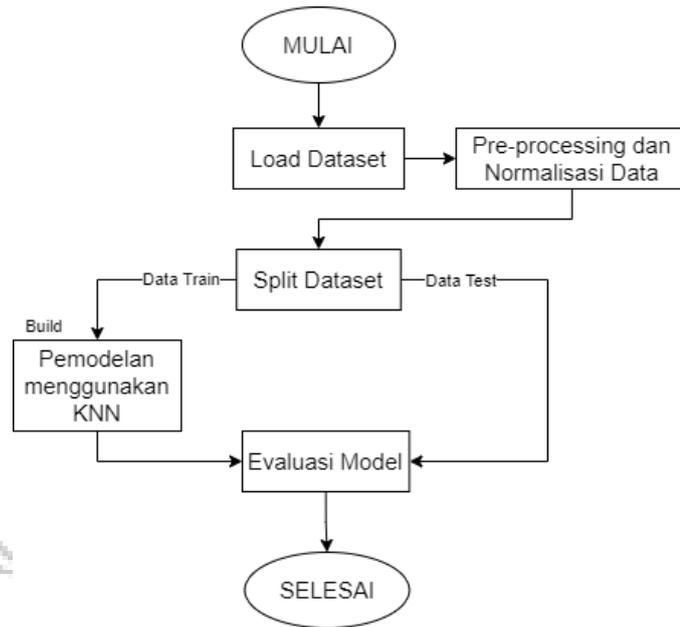
$$euc = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

Keterangan :

p_i = data latih
 q_i = data uji
 i = variable data
 n = dimensi data

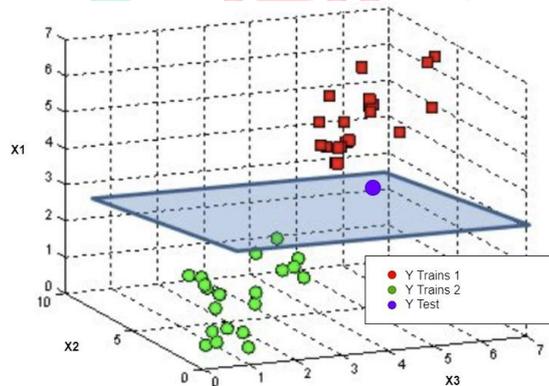
- c. Mengurutkan jarak yang terbentuk,
- d. Menentukan jarak terdekat dari k ,
- e. Mencocokkan kelas yang saling sesuai,
- f. Mencari jumlah kelas dari tetangga yang terdekat lalu menetapkan sebagai kelas data yang akan dievaluasi.

Dalam penelitian ini, KNN digunakan dalam pemodelan data dengan alur proses pengujian yang tergambar pada diagram alir Gambar 2.2 berikut ini.



Gambar 2. 2 Diagram Alir Pemodelan Data Algoritma *K-Nearest Neighbors*

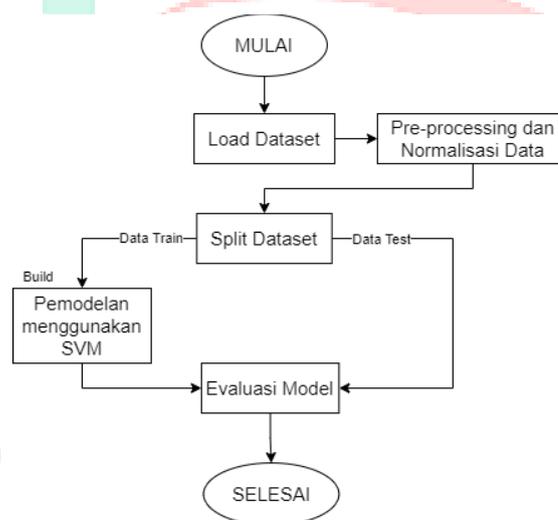
2.2.6 Support Vector Machine (SVM)



Gambar 2. 3 Ilustrasi SVM 3 dimensi (Schaeffer & Maddix, 2020)

SVM merupakan sistem *machine learning* yang menerapkan metode yaitu melatih data statistik dengan algoritma pembelajaran dan menggunakan ruang hipotetis berupa fungsi linier dalam sebuah ruang fitur dimensi tinggi (*feature space*) (Jumeilah, 2017, p. 21). Secara mendasar, SVM mengklasifikasikan data linier dan non-linier. Di sini dimaksudkan, SVM menggunakan pemetaan non-linier untuk mengubah data latih awal ke dimensi yang lebih tinggi. Pemahaman sederhana konsep SVM digambarkan sebagai usaha dalam menemukan garis *hyperplane* terbaik untuk memisahkan dua buah kelas data latih (Kasim &

Sударsono, 2019, p. 570). *Hyperplane* pemisah terbaik dari kedua kumpulan data latih ditemukan dengan mengukur margin *hyperplane* dan menemukan titik maksimal. Margin itu sendiri merupakan jarak antara *hyperplane* dengan data terdekat ke setiap kelas. Jarak terdekat tersebut disebut *support vector*. Pada dasarnya, prinsip kerja SVM hanya untuk menangani klasifikasi dua kelas, namun telah dikembangkan sehingga dapat digunakan untuk klasifikasi lebih dari dua kelas dengan adanya *pattern recognition*. Pada pengujian ini, digunakan dua jenis kernel SVM untuk membandingkan pola kernel mana yang terbaik untuk menghasilkan akurasi hasil prediksi pemodelan antara lain *Radial Basis Function* (RBF) dan *Linear*. RBF ini digunakan untuk mengolah data apabila data tidak terpisah secara linier. Sebaliknya, *Linear* digunakan Ketika data yang akan diolah sudah terpisah secara linier (Ningrum, 2018, p. 19). Dalam penelitian ini, SVM digunakan dalam pemodelan data dengan alur proses pengujian yang tergambar pada diagram alir Gambar 2.4 berikut ini.



Gambar 2. 4 Diagram Alir Algoritma *Support Vector Machine*

2.2.7 Evaluasi Model

Dalam melakukan prediksi data menggunakan machine learning, dibutuhkan parameter evaluasi untuk memeriksa validasi hasil pemodelan. RMSE atau *Root Mean Squared Error* merupakan suatu parameter penilaian kuadrat untuk mengukur rata-rata kesalahan atau error yang dihasilkan. RMSE berupa akar kuadrat dari rata-rata perbedaan kuadrat antara prediksi dan observasi aktual

(Sutoyo & Almaarif, 2020, p. 98). Semakin kecil hasil RMSE, maka semakin kecil tingkat error yang dihasilkan. Rumus RMSE dituliskan pada persamaan berikut :

$$RMSE = \sqrt{\frac{1}{n} \left(\sum_{j=1}^n (y_j - \hat{y}_j)^2 \right)}$$

Keterangan :

y = data uji yang sebenarnya

\hat{y} = data hasil prediksi

n = jumlah data uji

Selain mengetahui error yang dihasilkan dari hasil pengujian menggunakan RMSE, dibutuhkan juga parameter akurasi menggunakan parameter *Accuracy Score* dengan rumus sebagai berikut :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Keterangan :

Accuracy = Akurasi hasil pengujian

TP = Data positif yang diprediksi benar

TN = Data negative yang diprediksi benar

FP = Data positif yang diprediksi salah

FN = Data negatif yang diprediksi salah

Parameter ini digunakan untuk mengukur akurasi hasil prediksi dibandingkan dengan data yang sebenarnya.

2.2.8 Bahasa Pemrograman Python

Bahasa pemrograman Python merupakan bahasa pemrograman yang umum digunakan. Bahasa pemrograman ini cukup populer di kalangan programmer. Python diciptakan oleh Guido Van Rossum pada tahun 1991. Python merupakan bahasa pemrograman yang berorientasi dinamis. Dari kemampuan dan kapabilitasnya, Python diklaim sebagai bahasa pemrograman yang sangat jelas dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif (Baharuddin, et al., 2019, p. 271). Bahasa pemrograman Python populer digunakan

dalam pengujian pemodelan menggunakan *machine learning*. Dalam penelitian ini, pengujian dilakukan dengan menerapkan bahasa Python untuk memprediksi potensi penyakit hipertensi dengan menerapkan *machine learning* dengan algoritma KNN dan SVM.

