1. Pendahuluan

Diabetes mellitus adalah nama umum untuk kelainan metabolisme heterogen yang ditandai dengan hiperglikemia kronis. Penyebabnya adalah gangguan sekresi insulin atau efek insulin, atau kedua-duanya [1]. Jutaan orang di seluruh dunia terkena penyakit diabetes melitus (DM), yang merupakan penyakit tidak menular kronis. Prevalensi diabetes meningkat di berbagai kelompok sosial ekonomi karena meningkatnya tingkat stres dan penurunan aktivitas fisik. Hal ini pada akhirnya mengakibatkan obesitas dan komplikasi terkait, termasuk hipertensi dan diabetes tipe II [2].

Menurut Federasi Diabetes Internasional (IDF), prevalensi diabetes di kalangan orang dewasa berusia 20 hingga 79 tahun telah meningkat secara signifikan sejak pertama kali dipublikasikan pada tahun 2000. Perkiraan jumlah kasus diabetes meningkat dari sekitar 151 juta orang (merupakan 4,6% dari populasi global). pada saat itu) menjadi 537 juta pada tahun 2021 (mewakili 10,5% populasi). Ini berarti peningkatan lebih dari tiga kali lipat selama periode tersebut [3]. Penderita diabetes berisiko mengalami berbagai komplikasi yang dapat berujung pada kecacatan atau kematian [3].

Berdasarkan temuan Riset Kesehatan Dasar (Riskesdas) tahun 2018, prevalensi diabetes di Indonesia pada individu berusia 15 tahun ke atas yang didiagnosis dokter tercatat sebesar 2%. Jika dibandingkan dengan prevalensi diabetes pada penduduk usia 15 tahun ke atas pada hasil Riskesdas tahun 2013, terjadi peningkatan sebesar 1,5%. Dari hasil pemeriksaan kadar gula darah terlihat bahwa prevalensi diabetes melitus meningkat dari 6,9% pada tahun 2013 menjadi 8,5% pada tahun 2018. Data ini menunjukkan bahwa hanya sekitar 25% penderita diabetes yang menyadari bahwa dirinya mengidap penyakit tersebut. [4]. Oleh karena itu, perlu adanya suatu algoritma yang dapat membantu memprediksi apakah seseorang menderita diabetes atau tidak. Untuk membuat prediksi yang akurat tentang kemungkinan terjadinya penyakit, diperlukan penerapan algoritma data mining. Data mining merupakan suatu disiplin ilmu yang fokus menganalisis data untuk memperoleh informasi tambahan yang lebih luas dari informasi yang tersedia saat ini melalui penggunaan kata kunci atau informasi yang ada [5].

Penelitian ini juga akan membandingkan metode data mining dalam memprediksi apakah seseorang menderita diabetes atau tidak. Metode data mining yang digunakan adalah metode klasifikasi dengan menggunakan algoritma Support Vector Machine dan Naïve Bayes. Dalam penelitian ini, kami akan menyajikan beberapa kontribusi utama. Pertama, kami akan menguji keakuratan metode Support Vector Machine dan Naive Bayes dalam mengolah dataset diabetes melitus. Kedua, kita akan membandingkan tingkat akurasinya dengan penelitian sebelumnya yang menggunakan metode Support Vector Machine dan Naive Bayes. Ketiga, kita akan mengatur parameter yang paling optimal untuk setiap metode yang digunakan, seperti parameter gamma, C, var_smoothing, Alpha. Keempat,

kita akan membandingkan nilai akurasi metode Support Vector Machine dan Naive Bayes dari percobaan yang dilakukan.

Penelitian ini terdiri dari tiga bagian utama. Bagian Bahan dan Metode akan mencakup pekerjaan terkait dan metodologi yang kami rencanakan untuk diterapkan dalam penelitian ini. Pada bagian Hasil dan Pembahasan akan diuraikan mengenai setting dan hasil percobaan yang telah kami lakukan. Terakhir, pada bagian Kesimpulan, kami akan memberikan kesimpulan dan saran untuk penelitian selanjutnya.

1. Tinjauan Pustaka

Dalam beberapa tahun terakhir, para peneliti telah mengembangkan berbagai algoritma rekomendasi berdasarkan data teks yang disediakan. Alghamdi dkk. [6] menyelidiki kinerja relatif dari berbagai metode pembelajaran mesin, mengembangkan model prediktif berbasis ensemble, dan menggunakan pendekatan Synthetic Minority Oversampling Technique (SMOTE) untuk memprediksi kejadian diabetes menggunakan catatan medis kebugaran kardiorespirasi. Penelitian yang dilakukan oleh Poonia dkk [7]. berjudul "Model Prediksi dan Klasifikasi Diagnosis Cerdas untuk Deteksi Penyakit Ginjal". Studi ini menggunakan analisis prediktif berbasis pembelajaran mesin untuk mendeteksi penyakit ginjal pada tahap awal. Penelitian ini menyediakan model prediktif berbasis fitur untuk deteksi penyakit ginjal. Berbagai algoritma pembelajaran mesin digunakan, termasuk k-nearest neighbors (KNN), artificial neural networks (ANN), support vector machine (SVM), naive Bayes (NB), dan lainnya. Penggunaan Recursive Feature Elimination (RFE) dan teknik seleksi fitur Chi-Square diperlukan untuk membangun dan menganalisis berbagai model prediktif pada dataset yang terdiri dari pasien sehat dan penyakit ginjal. Dalam penelitian lain, Zou et al. [8] yang berjudul "Memprediksi Diabetes Mellitus Menggunakan Machine Learning," menggunakan decision tree, random forest, dan neural network untuk memprediksi diabetes mellitus. Kumpulan data yang digunakan terdiri dari data pemeriksaan fisik rumah sakit di Luzhou, Tiongkok, dengan 14 atribut. Teknik validasi silang lima lipatan digunakan untuk menguji model-model tersebut. Demikian pula, Vigneswari et al. [9] menerapkan klasifikasi Machine Learning untuk memprediksi penyakit pasien dan mengevaluasi kinerja pohon keputusan dalam memprediksi Diabetes Mellitus (DM). Analisis ini didasarkan pada akurasi dan True Positive Rate (TPR). Penelitian berjudul "Memprediksi Risiko Terjadinya Diabetes Mellitus Tipe 2 pada Lansia Tiongkok Menggunakan Teknik Machine Learning" yang dikembangkan oleh Liu et al. [10] bertujuan untuk membuat model prediksi yang efektif berdasarkan machine learning (ML) untuk risiko Diabetes Mellitus Tipe 2 (T2DM) di kalangan lansia di Tiongkok. Studi ini menggunakan data pemeriksaan kesehatan dari orang dewasa yang berusia di atas 65 tahun di Wuhan, Tiongkok, dari tahun 2018 hingga 2020. Empat algoritma ML digunakan dalam penelitian ini: logistic regression (LR), decision tree (DT), random forest (RF), dan extreme gradient boosting (XGBoost). Evaluasi kinerja model-model tersebut akan dilakukan berdasarkan area di bawah kurva karakteristik operasi penerima (AUC), sensitivitas, spesifisitas, dan akurasi. Maulidah et al. [11] melakukan penelitian yang berjudul "Prediksi Diabetes Mellitus Menggunakan Mesin Pendukung Vektor dan Metode Naive Bayes." Studi ini dikembangkan dengan memproses data basis data kesehatan sekunder menggunakan Mesin Pendukung Vektor dan metode Naive Bayes untuk menentukan akurasi diagnosis diabetes. Berdasarkan beberapa studi yang dilakukan, mereka dapat berfungsi sebagai referensi

untuk mengembangkan akurasi yang lebih baik dalam memprediksi diagnosis diabetes. Penelitian yang dilakukan oleh Faruque et al. [12] fokus pada "Memprediksi Diabetes Mellitus dan Menganalisis Korelasi Faktor Risiko." Studi ini bertujuan untuk menjelajahi berbagai faktor risiko seperti komplikasi ginjal, tekanan darah, gangguan pendengaran, dan komplikasi kulit yang terkait dengan penyakit ini menggunakan teknik machine learning dan pengambilan keputusan. Studi ini menggunakan empat algoritma machine learning populer, yaitu Support Vector Machine (SVM), Naive Bayes (NB), k-nearest neighbors (KNN), dan decision trees C4.5 (DT). Data yang digunakan terdiri dari data medis diagnostik dari 200 pasien diabetes di Pusat Medis Chittagong, Bangladesh, yang terdiri dari 16 atribut. Mushtaq et al. [13] melakukan penelitian tentang "Prediksi Diabetes Mellitus Berbasis Klasifikasi Voting Menggunakan Teknik Machine Learning yang Disesuaikan." Kumpulan data yang digunakan diperoleh dari repositori online. Pada tahap awal penelitian ini, logistic regression, Support Vector Machine, k-nearest neighbors, gradient boosting, Naive Bayes, dan Random Forests diterapkan untuk menilai efisiensi prediksi berdasarkan kondisi awal pasien. Selanjutnya, digunakan algoritma voting dengan tiga algoritma terbaik yang berkinerja baik.

2. Metode Penelitian

3.1. Pengumpulan data

Proses pengumpulan informasi akan menggunakan database Diabetes Mellitus yang diperoleh dari Kaggle yang berasal dari sebuah rumah sakit di Frankfurt, Jerman [14], dan juga dataset Diabetes Mellitus dari LAB01 DAT263x yang diambil dari website Kaggle [15]. Dataset tersebut masing-masing terdiri dari 2000 dan 15000 catatan, dengan berbagai variabel atau atribut prediktor medis seperti Kehamilan, Glukosa, Tekanan Darah, Ketebalan Kulit, Insulin, BMI, Fungsi Silsilah Diabetes, Usia, dan Hasil. Selanjutnya data akan diolah menggunakan tools Python.

3.2. Pemrosesan Awal Data

Pemrosesan awal memainkan peran penting dalam pembelajaran mesin karena merupakan langkah penting. Dengan kata lain, langkah ini bertujuan untuk mengubah data mentah menjadi format yang lebih mudah dipahami dan digunakan. Kumpulan data sering kali mengandung kesalahan atau ketidaksempurnaan, sehingga langkah ini dapat membantu mengatasi masalah ini dan memfasilitasi pemrosesan data [16]. Untuk menjamin keabsahan data, penelitian ini juga akan melakukan beberapa tindakan preprocessing terhadap data yang tidak relevan atau tidak digunakan.

Pemisahan Data merupakan bagian dari tahap Preprocessing. Pada langkah ini, dataset akan dibagi menjadi dua bagian: dataset pelatihan dan dataset pengujian. Kumpulan data pelatihan akan digunakan untuk

membuat model dan masih menyertakan data berlabel. Sedangkan dataset pengujian digunakan untuk memvalidasi model guna menilai keakuratan algoritma. Pada dataset pengujian, data label akan dihilangkan dan dipisahkan sebagai nilai target sebenarnya [17]. Pada penelitian ini data dibagi menjadi 80% untuk dataset pelatihan dan 20% untuk dataset pengujian.

Selanjutnya pada proses Preprocessing terdapat tahap normalisasi yang merupakan metode yang baik untuk mengurangi perbedaan data dan meningkatkan efisiensi. Dalam kasus data yang sangat besar, metode normalisasi harus memiliki aturan yang sederhana dan dapat dijalankan dengan cepat [18]. Untungnya pada penelitian ini normalisasi data sebagai salah satu langkah preprocessing tidak diperlukan karena dataset yang digunakan sudah sesuai dan tidak memerlukan penyesuaian lebih lanjut.

3.3. Metode analisis data yang diterapkan

3.3.1. Support Vector Machine

Metode SVM memiliki kemampuan yang kuat dalam membangun klasifikasi [19]. yang dapat ditemukan di [20]–[22] dan juga dianggap sebagai perpanjangan dari klasifikasi margin maksimum [23]. Support Vector Machine (SVM) adalah algoritma yang memanfaatkan contoh untuk memperoleh pengetahuan tentang cara memberi label pada objek [24]. SVM adalah metode klasifikasi diskriminatif yang sangat efektif yang secara formal ditandai dengan hyperplane yang optimal. Hyperplane yang optimal menghasilkan klasifikasi untuk contoh baru dan kumpulan data yang mendukung hyperplane tersebut disebut sebagai vektor pendukung [25]. Hyperplane disesuaikan untuk memastikan jarak maksimumnya dari titik data terdekat setiap kelas. Titik data terdekat disebut sebagai vektor pendukung. Hal ini berlaku untuk dataset pelatihan dengan Rumus (1) [19].

$$(\boldsymbol{x_1}, \boldsymbol{y_1}), \dots, (\boldsymbol{x_n}, \boldsymbol{y_n}), \boldsymbol{x_i} \in \boldsymbol{R^d}\ \boldsymbol{end}\ \boldsymbol{y_i} \in (\boldsymbol{-1}, \boldsymbol{+1}) \quad (1)$$

Dapat dilihat bahwa x_i mewakili representasi vektor fitur, sedangkan y_i sesuai dengan label kelas, yang dapat berupa positif atau negatif, dari i senyawa pelatihan. Oleh karena itu, Rumus (2) dapat digunakan untuk menentukan hyperplane optimal.

$$\boldsymbol{wx^T + b = 0} \quad (2)$$

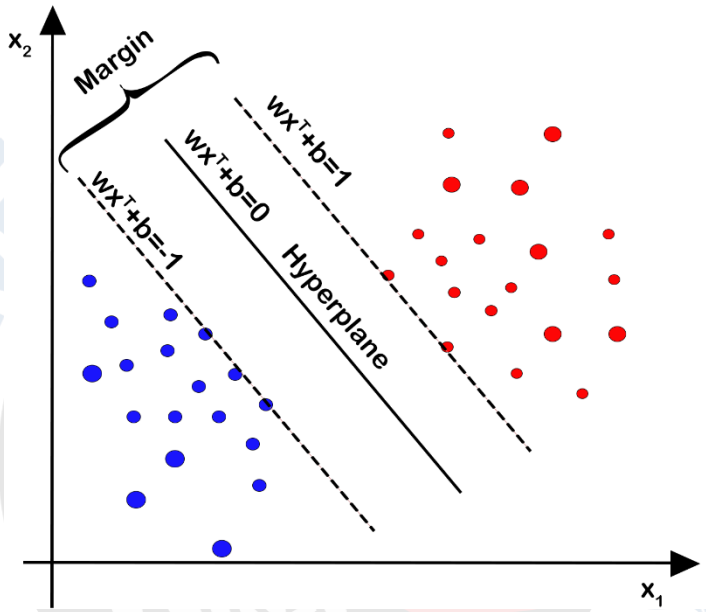
Vektor bobot direpresentasikan sebagai w, di mana x menunjukkan vektor fitur masukan. Biasanya dilambangkan dengan b. Baik vektor bobot, w, maupun bias, b, harus memenuhi pertidaksamaan berikut untuk semua elemen dalam himpunan pelatihan sesuai Rumus (3).

$$\begin{aligned} \boldsymbol{wx_i^T + b \geq +1\ if\ y_i = 1} \\ \boldsymbol{wx_i^T + b \leq -1\ if\ y_i = -1} \end{aligned} \quad (3)$$

Tujuan dari pelatihan model SVM adalah untuk menentukan nilai w dan b yang memungkinkan hyperplane memisahkan data secara efektif dengan margin maksimum, yang ditentukan oleh Rumus (4).

$$\frac{1}{\|w\|^2} \tag{4}$$

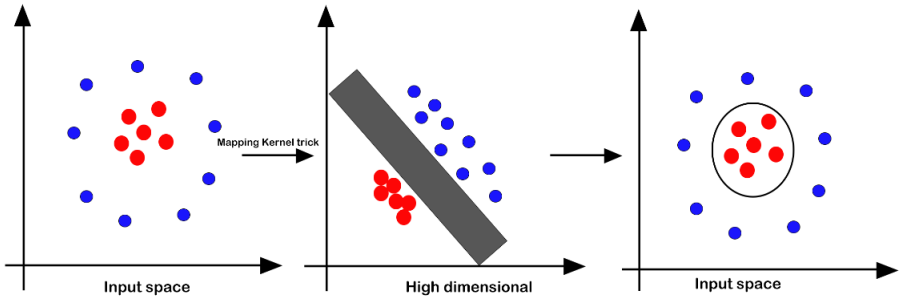
Vektor tumpuan adalah istilah yang digunakan untuk $x_i$vektor yang memenuhi persamaan $|y_i|\,(wx_i^T + b) = 1$, seperti ditunjukkan pada Gambar 1.



Gambar 1 *Model SVM linier digunakan untuk mengklasifikasikan dua kelas yaitu merah dan biru.*

Selain itu, salah satu teknik alternatif yang saat ini digunakan adalah Support Vector Machine (SVM) Nonlinear, yang menggunakan fungsi kernel untuk mengklasifikasikan sekumpulan data dengan tujuan menemukan hyperplane optimal dalam ruang fitur berdimensi tinggi seperti pada (gambar 2) [26]. Kernel merupakan suatu metode yang digunakan untuk menyelesaikan permasalahan nonlinier dengan memanfaatkan klasifikasi linier dan melibatkan transformasi data yang tidak dapat dipisahkan secara linier [25]. Dengan menggunakan $K(x_n, x_i)$, data asli mengalami transformasi menjadi ruang berdimensi lebih tinggi. Dalam proses ini, fungsi transformasi diterapkan pada perkalian titik $\phi\,(x)$, seperti yang digambarkan pada Rumus (5) [27].

$$\boldsymbol{K(x_n, x_i) = \emptyset(x_n)\emptyset(x_i)} \tag{5}$$



Gambar 2 *Fungsi kernel digunakan untuk mengubah dan memisahkan data yang tidak dapat dipisahkan oleh SVM linier.*

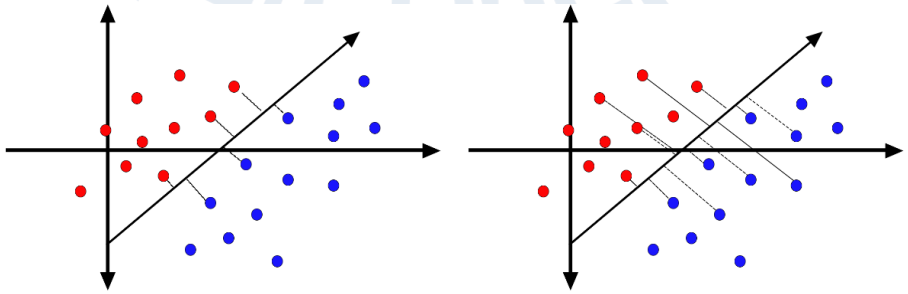
Dalam algoritma SVM, terdapat beberapa jenis fungsi kernel yang umum seperti linier, radial basis function (RBF), sigmoid, dan polinomial yang tercantum pada Tabel 1. Setiap fungsi kernel memiliki parameter tertentu yang perlu dioptimalkan untuk mencapai kinerja terbaik [27].

Tabel 1. Empat kernel umum

No	Kernel Function	Formula
1	Linear	$K(x_n, x_i) = (x_n, x_i)$
2	RBF	$K(x_n, x_i) = exp(-\gamma \|x_n - x_i\|^2 + C)$
3	Sigmoid	$K(x_n, x_i) = tanh(\gamma(x_n, x_i) + r)$
4	Polynominal	$K(x_n, x_i) = (\gamma(x_n, x_i) + r)^d$

3.3.2. Parameter regularisasi SVM

Parameter regularisasi C digunakan untuk menentukan besarnya penalti kesalahan. Hal ini berdampak pada keseimbangan antara kelancaran batas keputusan dan kemampuan untuk mengklasifikasikan data pelatihan secara akurat [28]. Jika nilai C tinggi maka data pelatihan akan diklasifikasikan secara akurat berdasarkan hyperplane; sebaliknya jika nilai C rendah maka optimasi akan mencari margin yang lebih tinggi yang memisahkan hyperplane [25].



Gambar 3 *Gamma Tinggi (kiri) dan Gamma Rendah (Kanan).*

Salah satu faktor yang mempengaruhi kinerja klasifikasi SVM adalah gamma, yang termasuk dalam subruang sampel dengan perubahan

yang kompleks [29]. Nilai gamma yang tinggi (Gambar 3 kiri) memberikan bobot lebih pada titik data yang dekat dengan batas keputusan. Sebaliknya, nilai gamma yang rendah (Gambar 3 kanan) memperhitungkan titik data yang jauh dari batas keputusan dalam penghitungan batas keputusan. [25].

3.3.3. Naïve Bayes

Naive Bayes Classifier merupakan salah satu teknik dalam text mining yang berguna untuk menangani permasalahan dalam opinion mining dengan mengkategorikan ke dalam dua kategori yaitu opini positif dan negatif. Oleh karena itu, Naive Bayes Classifier efektif sebagai metode untuk mengklasifikasikan teks [30].

Naive Bayes Classifier merupakan pendekatan yang memanfaatkan teorema Bayes, yaitu metode klasifikasi sederhana berbasis probabilitas yang mengasumsikan bahwa setiap atribut dalam data bersifat independen [31]. Dalam mengklasifikasikan data menggunakan naif bayes, diwakili oleh sekumpulan atribut "x1, x2,…, xn ". Rumus (6) dapat digunakan untuk menyatakan model probabilitas setiap kelas K.

$$\boldsymbol{P(y_k|x_1, x_2, \dots, x_n)} \tag{6}$$

Selanjutnya n mewakili jumlah atribut, sedangkan k mewakili jumlah kelas yang ada dalam kumpulan y data kelas. Dalam perspektif probabilitas, klasifikasi digambarkan sebagai aturan Bayes yang dapat dituliskan menurut Rumus (7):

$$\boldsymbol{P(y_k|x_i) = \frac{P(y_k).P(x_i|y_k)}{P(x_i)}} \tag{7}$$

Dalam rumus berikut, $P(y_k|x_i)$ Menunjukkan kemungkinan suatu peristiwa y_k terjadi dengan adanya peristiwa x_i, $P(x_i|y_k)$ menyatakan kemungkinan x_n terjadinya suatu peristiwa ketika y_k terjadi, $P(y_k)$ menyatakan kemungkinan suatu peristiwa y_k, dan $P(x_i)$ menyatakan kemungkinan suatu peristiwa x_i. Untuk mencari nilai probabilitas tertinggi setiap kelas yang dapat dipilih menjadi kelas optimal, dapat digunakan rumus (8):

$$\mathbf{arg\,max}_{\mathbf{y_k \in y}} = \frac{\mathbf{P(y_k).P(x_i|y_k)}}{\mathbf{P(x_i)}} \tag{8}$$

Rumus (9) diturunkan dengan mengasumsikan nilai konstan untuk $P(x_i)$ di semua kelas.

$$\mathbf{arg\,max}_{\mathbf{y_k \in y}} = \mathbf{P(y_k).P(x_i|y_k)} \tag{9}$$

3.4. Evaluasi Model

Ada banyak metrik untuk mengevaluasi pemrosesan teks dan sistem pengambilan informasi. Kinerja sistem yang mengklasifikasikan dokumen ke dalam kategori dapat diukur menggunakan berbagai ukuran, seperti presisi, perolehan kembali, dan rata-rata makro [32]. Definisi presisi dan recall dapat dilihat pada Tabel 2.

Tabel 2. Pengertian FN, FP, TN, dan TP

	Negatif (N)	Positif (P)
Salah (P)	FN hasil prediksi: N hasil sebenarnya: P	FP hasil prediksi: P hasil sebenarnya: N
Benar (T)	TN hasil prediksi: N hasil sebenarnya: N	dll hasil prediksi: P hasil sebenarnya: P

TP adalah jumlah dokumen relevan yang diklasifikasikan oleh manusia dan pengklasifikasi, FN adalah jumlah dokumen yang dianggap relevan oleh manusia tetapi tidak diklasifikasikan relevan oleh pengklasifikasi, FP adalah jumlah dokumen yang dianggap tidak relevan oleh manusia tetapi diklasifikasikan relevan oleh pengklasifikasi , dan TN adalah jumlah dokumen yang dianggap tidak relevan oleh manusia dan pengklasifikasi [32].

Presisi berkaitan dengan proporsi prediksi positif yang benar dibandingkan dengan keseluruhan prediksi positif [33]. Presisi ditentukan dengan menggunakan rumus yang disediakan (10) untuk menghitung nilainya:

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \tag{10}$$

Penarikan kembali, yang juga disebut sebagai tingkat atau sensitivitas positif sebenarnya, mengacu pada proporsi prediksi yang benar (positif) dari total sampel positif aktual. Rumus (11) dapat digunakan untuk menghitung recall.

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \tag{11}$$

Hasilnya, skor F1, yang diperoleh dari rata-rata harmonik presisi dan perolehan, dapat dihitung menggunakan rumus khusus (12).

$$\boldsymbol{f1\ score} = \frac{\boldsymbol{2 \times precision * recall}}{\boldsymbol{precision + recall}} = \frac{\boldsymbol{2 \times TP}}{\boldsymbol{2 \times TP + FP + FN}} \tag{12}$$

3. Analisis hasil

Pada tahap ini akan dijelaskan hasil percobaan yang telah dilakukan. Penelitian ini menguji dan mengevaluasi kinerja suatu algoritma untuk

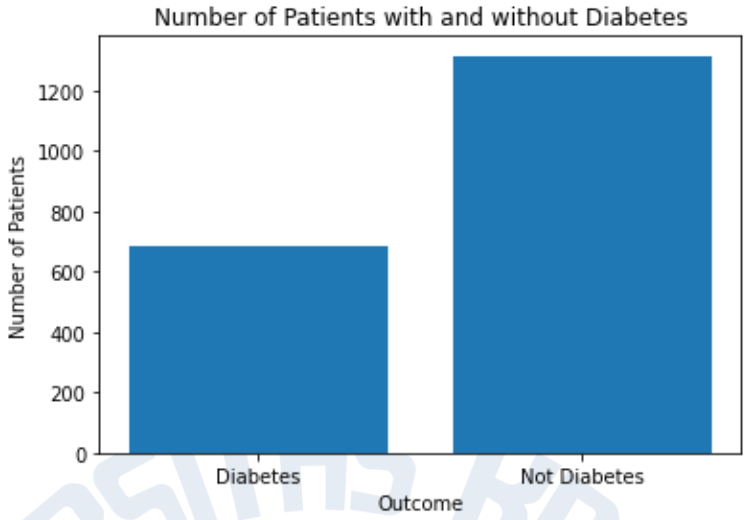
menentukan pasien diabetes melitus menggunakan dataset dari sebuah rumah sakit di Frankfurt, Jerman. Dataset yang digunakan berisi 2000 titik data, dengan berbagai variabel atau atribut prediktor medis seperti Kehamilan, Glukosa, Tekanan Darah, Ketebalan Kulit, Insulin, BMI, Fungsi Silsilah Diabetes, Usia, dan Hasil. Deskripsi dataset yang digunakan dalam percobaan ini dapat dilihat pada Tabel 3. Selanjutnya Tabel 4 dan Gambar 5 memberikan gambaran Dataset 1 yang dijadikan contoh.

Tabel 3. Deskripsi Kumpulan Data

TIDAK	Nama	Himpunan data	Jumlah data	Fitur
1	Kumpulan data 1	Dataset Diabetes Mellitus dari sebuah rumah sakit di Frankfurt, Jerman [14].	2000	8
2	Kumpulan data 2	Kumpulan data Diabetes Melitus dari LAB01 DAT263x [15].	15000	8

Tabel 4. Dataset diabetes melitus dari sebuah rumah sakit di Frankfurt, Jerman (Dataset 1).

Kehamilan	Glukosa	Tekanan darah	Ketebalan Kulit	Insulin	BMI	Fungsi Silsilah Diabetes	Usia	Hasil
2	138	62	35	0	33.6	0,127	47	2
0	84	82	31	125	38.2	0,233	23	0
0	145	0	0	0	44.2	0,630	31	0
0	135	68	42	250	42.3	0,365	24	0



Gambar 4 *Jumlah titik data kasus diabetes dan non-diabetes (Dataset 1).*

Pada tahap awal percobaan, dilakukan penelitian untuk menguji keakuratan metode seperti Support Vector Machine dan Naive Bayes dalam mengolah dataset diabetes melitus yang diperoleh dari sebuah rumah sakit di Frankfurt, Jerman. Hasil penelitian pada Tabel 5 menunjukkan hasil pengujian akurasi metode Support Vector Machine. Eksperimen dilakukan dengan menggunakan beberapa jenis kernel pada Support Vector Machine, seperti kernel linier, kernel polinomial (poli), dan kernel Radial Basis Function (rbf). Dari hasil eksperimen yang diperoleh dari dataset pengujian, akurasi tertinggi sebesar 83,50% dicapai ketika menggunakan kernel Radial Basis Function (RBF).

Tabel 5. Hasil eksperimen metode Support Vector Machine dengan Dataset 1.

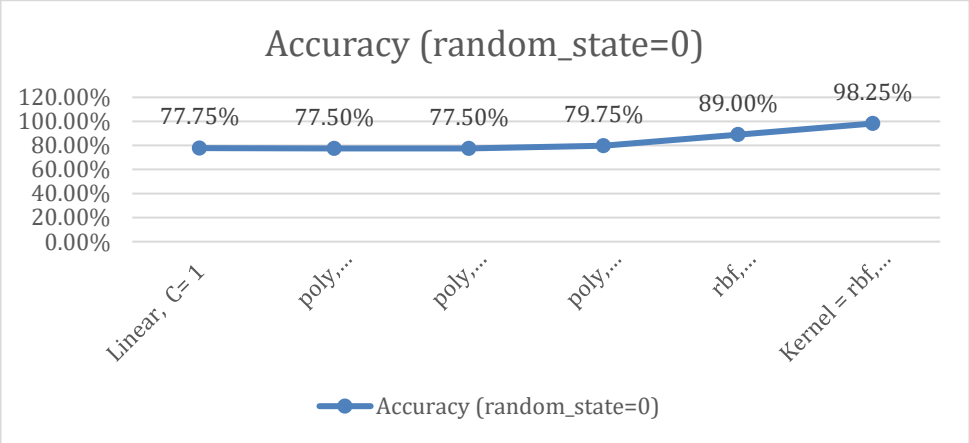
Mendukung Mesin Vektor	Akurasi (keadaan_acak =0)
Kernel = Linier	77,75%
Kernel = poli	77,00%
Kernel = rbf	78,50%
Kernel = rbf, gamma=0,001	83,50%

Tabel 6. Hasil eksperimen metode Naive Bayes dengan Dataset 1.

Bayes yang naif	Akurasi (keadaan_acak =0)
naif_bayes = GaussianNB ()	75,75%
naif_bayes = GaussianNB (var_smoothing =0,001)	77,00%
naif_bayes = BernoulliNB ()	66,25%
naif_bayes = MultinomialNB ()	60,25%

Sedangkan hasil uji akurasi metode Naive Bayes dengan Dataset 1 yang disajikan pada Tabel 6 juga dilakukan dengan beberapa variasi, seperti Naive Bayes Gaussian, Naive Bayes Multinomial, dan Naive Bayes Bernoulli. Dari hasil percobaan diperoleh akurasi tertinggi sebesar 77,00%

diperoleh ketika menggunakan variasi Naive Bayes Gaussian. Untuk mencapai tingkat akurasi yang optimal, diperlukan penyesuaian parameter yang paling optimal pada setiap metode. Untuk Support Vector Machine parameter yang harus disesuaikan adalah C dan gamma untuk kernel Radial Basis Function (RBF), sedangkan untuk metode Naive Bayes parameter yang harus disesuaikan adalah var_smoothing untuk variasi Naive Bayes Gaussian dan parameter alpha untuk Naive Bayes Multinomial.

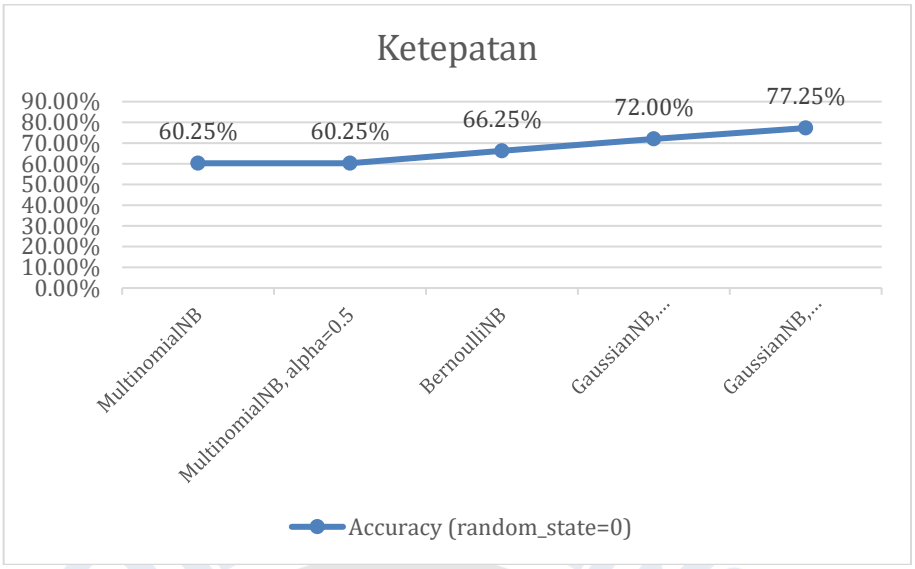


Gambar 5. *Hasil percobaan menggunakan metode Support Vector Machine dengan beberapa kernel.*

Pada Gambar 5 dilakukan percobaan menggunakan metode Support Vector Machine dengan parameter regularisasi C dan parameter yang mempengaruhi contoh pelatihan seperti derajat dan gamma. Awalnya percobaan dilakukan menggunakan kernel linier dengan C = 1 yang menghasilkan akurasi sebesar 77,75%. Namun setelah mencoba menggunakan kernel polinomial dengan nilai derajat yang berbeda, ditemukan bahwa derajat=3 dan C = 10 menghasilkan akurasi tertinggi yaitu 79,75%. Selanjutnya percobaan dilakukan dengan menggunakan kernel Radial Basis Function (rbf) dan berbagai parameter gamma. Hasilnya, diperoleh gamma=0,01 dan C=10 menghasilkan akurasi tertinggi sebesar 98,25%. Tabel 7 menunjukkan bahwa metode Support Vector Machine dengan kernel rbf dan gamma=0.01 menghasilkan akurasi tertinggi.

Tabel 7. Hasil laporan klasifikasi Support Vector Machine dengan C=10 dan gamma=0.01.

Barang	presisi	mengingat	skor f1	mendukung
0	0,98	1,00	0,99	272
1	0,99	0,95	0,97	128
Ketepatan			0,98	400
rata-rata makro	0,99	0,97	0,98	400
rata-rata tertimbang	0,98	0,98	0,98	400



Gambar 6 *Hasil percobaan metode Naive Bayes dengan Dataset 1 dan beberapa variasi.*

Pada Gambar 6 dilakukan percobaan dengan menggunakan berbagai parameter pada metode klasifikasi Naive Bayes. Hasil percobaan menggunakan Multinomial Naive Bayes dengan berbagai nilai parameter alpha menunjukkan akurasi sebesar 60,25%. Selanjutnya dilakukan percobaan dengan variasi Bernoulli Naive Bayes yang menghasilkan akurasi sebesar 66,25%. Kemudian dilakukan percobaan menggunakan Gaussian Naive Bayes dengan beberapa nilai parameter var_smoothing dan diperoleh akurasi tertinggi sebesar 77.25%. Tabel 8 menunjukkan bahwa metode klasifikasi Naive Bayes dengan variasi Gaussian dan nilai parameter var_smoothing menghasilkan akurasi tertinggi sebesar 77,25%.

Tabel 8. Hasil laporan klasifikasi naif_bayes dengan GaussianNB (var_smoothing =0.01).

Barang	presisi	mengingat	skor f1	mendukung
0	0,79	0,91	0,84	272
1	0,71	0,48	0,58	128
Ketepatan			0,77	400
rata-rata makro	0,75	0,70	0,71	400
rata-rata tertimbang	0,76	0,77	0,76	400

Oleh karena itu, kesimpulan yang diambil adalah penggunaan metode Support Vector Machine dengan tingkat akurasi sebesar 98,25% lebih unggul dibandingkan penerapan metode Naive Bayes. Hal ini menjadikan Support Vector Machine dengan kernel Radial Basis Function (rbf) sebagai metode terbaik dalam penelitian ini. Selanjutnya penelitian ini akan membandingkan tingkat akurasi dengan penelitian sebelumnya yang menggunakan metode Support Vector Machine dan Naive Bayes.

Tabel 9. Perbandingan Hasil Eksperimen dengan penelitian sebelumnya.

Algoritma	Himpunan data	Hasil Akurasi
Mendukung mesin Vektor[11]	Kumpulan data 1	78,04%
naif_bayes [11]	Kumpulan data 1	76,98%
Metode Usulan		
Mendukung Mesin Vektor dengan C=10 dan gamma=0,01.	Kumpulan data 1	98,25%
naif_bayes dengan GaussianNB (var_smoothing =0,01).	Kumpulan data 1	77,25%

Penelitian sebelumnya yang dilakukan oleh Maulidah dkk. dapat dilihat pada Tabel (9). Dengan menggunakan metode Support Vector Machine (SVM) dan Naive Bayes, mereka mencapai akurasi tertinggi sebesar 78,04% menggunakan Support Vector Machine [11]. Dalam penelitian ini, kami menggunakan Support Vector Machine dengan C=10 dan gamma=0.01 dan Naive Bayes dengan GaussianNB (var_smoothing =0.01). Namun, metode yang kami usulkan berhasil mencapai akurasi tertinggi sebesar 98,25% menggunakan SVM. Keberhasilan ini dicapai melalui penelitian kami yang berfokus pada optimasi parameter, yang bertujuan untuk mencapai tingkat akurasi yang lebih tinggi. Pemilihan parameter yang cermat menjadikan penelitian ini lebih optimal dan akurat dibandingkan penelitian sebelumnya. Langkah selanjutnya adalah melakukan percobaan pada beberapa dataset yang berbeda. Eksperimen ini dimaksudkan untuk memastikan bahwa model yang dikembangkan menggunakan Support Vector Machine dengan Radial Basis Function (rbf) dapat mencapai tingkat akurasi yang tinggi meskipun diuji pada kumpulan data yang berbeda.

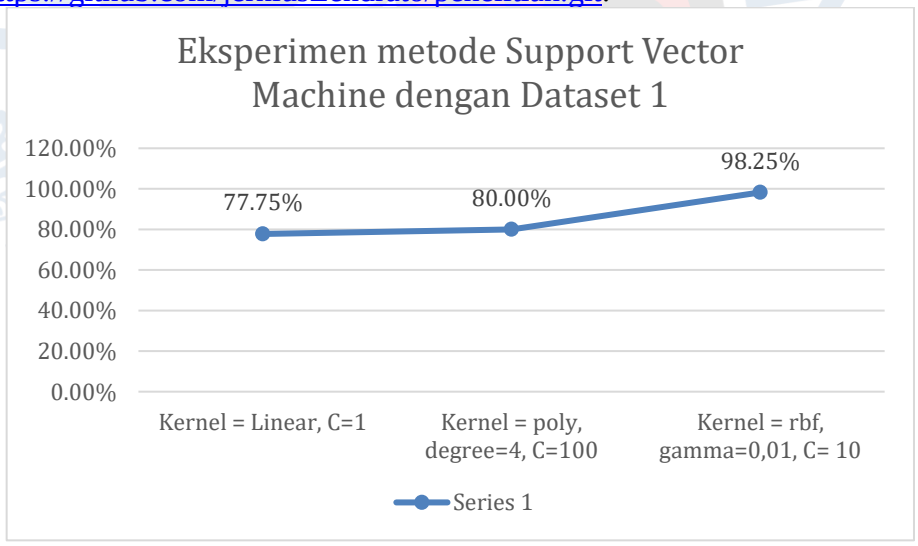
Tabel 10. Hasil percobaan metode Support Vector Machine dengan Dataset 2.

Mendukung Mesin Vektor	**Akurasi (keadaan_acak =0)**
Kernel = Linier, C=10	79,40%
Kernel = poli, derajat=4, C=100	82,97%
Kernel = rbf , gamma=0,001, C=1	**85,40%**

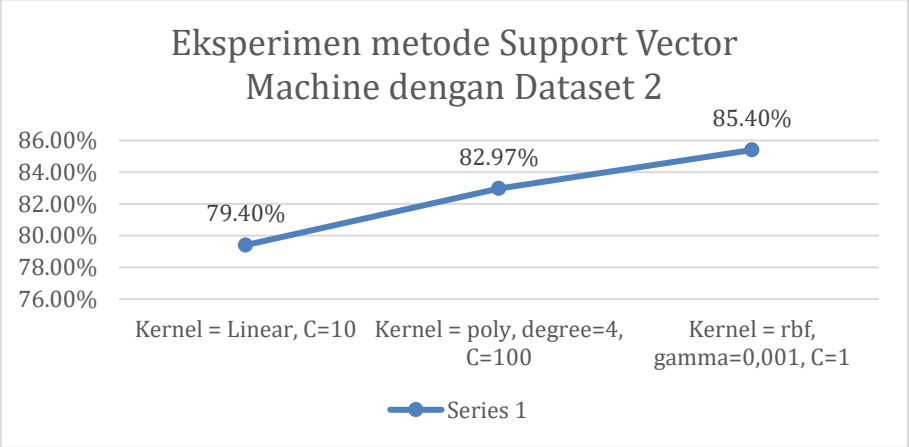
Tabel 11. Hasil percobaan metode Naive Bayes dengan Dataset 2.

Bayes yang naif	**Akurasi (keadaan_acak = 0)**
naif_bayes = MultinomialNB ()	62,30%
naif_bayes = BernoulliNB ()	66,73%
naif_bayes = GaussianNB (var_smoothing =0,00001)	**79,53%**

Hasil percobaan menggunakan metode Support Vector Machine pada dataset Diabetes Mellitus dari LAB01 DAT263x dicatat pada Tabel 10, sedangkan hasil menggunakan metode Naive Bayes dicatat pada Tabel 11. Dari hasil yang diperoleh dari dataset pengujian dapat dapat disimpulkan bahwa metode Support Vector Machine dengan kernel Radial Basis Function (RBF) masih yang terbaik dibandingkan metode lainnya. Metode ini mampu memberikan akurasi yang lebih tinggi dibandingkan dengan metode lainnya yaitu sebesar 85,40%. Dalam penelitian ini, nilai akurasi yang lebih tinggi umumnya menunjukkan kinerja yang lebih baik, sedangkan nilai akurasi yang lebih rendah menunjukkan kinerja yang kurang memuaskan. Jika ditemukan persentase akurasi tertinggi dari beberapa model eksperimen, maka eksperimen tersebut merupakan pengklasifikasi terbaik [34]. Temuan penelitian menunjukkan adanya tren peningkatan nilai akurasi pada setiap percobaan yang dilakukan untuk seluruh dataset. Berdasarkan hasil evaluasi yang disajikan pada Gambar 7, dapat disimpulkan bahwa model usulan menggunakan Support Vector Machine dengan kernel Radial Basis Function (rbf) memiliki kinerja yang lebih baik dibandingkan metode lain pada setiap dataset. Metode Support Vector Machine dengan kernel Radial Basis Function (rbf) dapat menghasilkan akurasi terbaik dalam klasifikasi yang dilakukan. Selanjutnya, kami mengunggah eksperimen kami di tautan GitHub https://github.com/jerniusZendrato/penelitian.git.



(A)



(B)

Gambar 7 *Hasil percobaan metode Support Vector Machine dengan kernel pada (a) Dataset 1, dan (b) Dataset 2.*

4. Kesimpulan

Penelitian ini bertujuan untuk meningkatkan tingkat akurasi penelitian sebelumnya dengan mengeksplorasi berbagai metode dan parameter. Pada percobaan yang dilakukan, ditemukan bahwa metode Support Vector Machine (SVM) memiliki kinerja yang lebih baik dibandingkan dengan Naive Bayes. Untuk lebih meningkatkan akurasi kedua metode, percobaan selanjutnya dilakukan dengan memvariasikan parameter seperti parameter regularisasi C, parameter pelatihan seperti derajat dan gamma pada SVM, dan berbagai parameter pada metode klasifikasi Naive Bayes seperti alpha dan var_smoothing. Dari hasil percobaan pada Gambar 5, 6, dan 7 ditemukan bahwa metode SVM dengan kernel Radial Basis Function (rbf) dapat mencapai akurasi yang lebih tinggi. Dibandingkan fungsi kernel lainnya, penggunaan SVM dengan kernel rbf mampu meningkatkan akurasi dari 77.75% menjadi 98.25% dan mengungguli akurasi tertinggi pada metode Naive Bayes yaitu 77.25%. Hal ini menunjukkan bahwa penyesuaian metode yang digunakan sangat penting untuk mengoptimalkan akurasi yang dihasilkan. Hasil percobaan juga membuktikan kelayakan dan keakuratan algoritma yang diusulkan. Dengan akurasi tinggi yang dicapai oleh SVM dengan kernel rbf, terdapat potensi signifikan untuk penerapannya di masa depan dalam sistem pengenalan pola medis. Hal ini dapat dimanfaatkan untuk mendukung dokter dalam mendiagnosis penyakit, tidak terbatas pada diabetes tetapi juga penyakit seperti kanker dan penyakit jantung, dengan lebih tepat dan cepat, sehingga meningkatkan perawatan pasien. Di masa depan, perlu mempertimbangkan untuk membandingkan metode yang berbeda dan memperbarui kumpulan data untuk meningkatkan kualitas hasil penelitian. Selain itu, kami akan mengimplementasikan SHAP untuk mendeskripsikan fitur penting.

DAFTAR PUSTAKA

[1] A. Petersmann *et al.*, "Definition, Classification and Diagnosis of Diabetes Mellitus," *Experimental and Clinical Endocrinology and Diabetes*, vol. 127, 2019, doi: 10.1055/a-1018-9078.

[2] J. E. John and N. A. John, "Imminent risk of covid-19 in diabetes mellitus and undiagnosed diabetes mellitus patients," *Pan African Medical Journal*, vol. 36, 2020, doi: 10.11604/pamj.2020.36.158.24011.

[3] I. D. Federation, "IDF Diabetes Atlas Tenth edition 2021," *International Diabetes Federation*, 2021.

[4] KemenkesRI, "infodatin Pusat Data Informasi kementrian kesehatan 2020 Diabetes Melitus.," *kementrian kesehatan RI*, vol. 15, no. 2, 2020.

[5] A. K. Tiwari, G. Ramakrishna, L. K. Sharma, and S. K. Kashyap, "Academic performance prediction algorithm based on fuzzy data mining," *IAES International Journal of Artificial Intelligence*, vol. 8, no. 1, 2019, doi: 10.11591/ijai.v8.i1.pp26-32.

[6] M. Alghamdi, M. Al-Mallah, S. Keteyian, C. Brawner, J. Ehrman, and S. Sakr, "Predicting diabetes mellitus using SMOTE and ensemble machine learning approach: The Henry Ford ExercIse Testing (FIT) project," *PLoS One*, vol. 12, no. 7, 2017, doi: 10.1371/journal.pone.0179805.

[7] R. C. Poonia *et al.*, "Intelligent Diagnostic Prediction and Classification Models for Detection of Kidney Disease," *Healthcare (Switzerland)*, vol. 10, no. 2, 2022, doi: 10.3390/healthcare10020371.

[8] Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, and H. Tang, "Predicting Diabetes Mellitus With Machine Learning Techniques," *Front Genet*, vol. 9, 2018, doi: 10.3389/fgene.2018.00515.

[9] D. Vigneswari, N. K. Kumar, V. Ganesh Raj, A. Gugan, and S. R. Vikash, "Machine Learning Tree Classifiers in Predicting Diabetes Mellitus," in *2019 5th International Conference on Advanced Computing and Communication Systems, ICACCS 2019*, 2019. doi: 10.1109/ICACCS.2019.8728388.

[10] Q. Liu *et al.*, "Predicting the Risk of Incident Type 2 Diabetes Mellitus in Chinese Elderly Using Machine Learning Techniques," *J Pers Med*, vol. 12, no. 6, 2022, doi: 10.3390/jpm12060905.

[11] N. Maulidah, R. Supriyadi, D. Y. Utami, F. N. Hasan, A. Fauzi, and A. Christian, "Prediksi Penyakit Diabetes Melitus Menggunakan Metode Support Vector Machine dan Naive Bayes," *Indonesian Journal on Software Engineering (IJSE)*, vol. 7, no. 1, 2021, doi: 10.31294/ijse.v7i1.10279.

[12] M. F. Faruque, Asaduzzaman, S. M. M. Hossain, M. H. Furhad, and I. H. Sarker, "Predicting diabetes mellitus and analysing risk-factors correlation," *EAI Endorsed Trans Pervasive Health Technol*, vol. 5, no. 20, 2020, doi: 10.4108/eai.13-7-2018.164173.

[13] Z. Mushtaq, M. F. Ramzan, S. Ali, S. Baseer, A. Samad, and M. Husnain, "Voting Classification-Based Diabetes Mellitus Prediction Using

Hypertuned Machine-Learning Techniques," *Mobile Information Systems*, vol. 2022, 2022, doi: 10.1155/2022/6521532.
[14] "diabetes | Kaggle." https://www.kaggle.com/datasets/johndasilva/diabetes (accessed Apr. 20, 2023).
[15] "Diabetes from DAT263x Lab01 | Kaggle." https://www.kaggle.com/datasets/fmendes/diabetes-from-dat263x-lab01 (accessed Apr. 20, 2023).
[16] R. Ghorbani and R. Ghousi, "Comparing Different Resampling Methods in Predicting Students' Performance Using Machine Learning Techniques," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2986809.
[17] D. A. Anggoro and W. Supriyanti, "Improving accuracy Bb applying Z-score normalization in linear regression and polynomial regression model for real estate data," *International Journal of Emerging Trends in Engineering Research*, vol. 7, no. 11, 2019, doi: 10.30534/ijeter/2019/247112019.
[18] W. Li and Z. Liu, "A method of SVM with normalization in intrusion detection," in *Procedia Environmental Sciences*, 2011. doi: 10.1016/j.proenv.2011.12.040.
[19] S. Huang, C. A. I. Nianguang, P. Penzuti Pacheco, S. Narandes, Y. Wang, and X. U. Wayne, "Applications of support vector machine (SVM) learning in cancer genomics," *Cancer Genomics and Proteomics*, vol. 15, no. 1. 2018. doi: 10.21873/cgp.20063.
[20] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. 2000. doi: 10.1017/cbo9780511801389.
[21] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond Adaptive computation and machine learning*. 2001.
[22] C. González, J. Mira-McWilliams, and I. Juárez, "Important variable assessment and electricity price forecasting based on regression tree models: Classification and regression trees, Bagging and Random Forests," *IET Generation, Transmission and Distribution*, vol. 9, no. 11, 2015, doi: 10.1049/iet-gtd.2014.0655.
[23] P. Golpour *et al.*, "Comparison of support vector machine, naïve bayes and logistic regression for assessing the necessity for coronary angiography," *Int J Environ Res Public Health*, vol. 17, no. 18, 2020, doi: 10.3390/ijerph17186449.
[24] W. S. Noble, "What is a support vector machine?," *Nature Biotechnology*, vol. 24, no. 12. 2006. doi: 10.1038/nbt1206-1565.
[25] G. Battineni, N. Chintalapudi, and F. Amenta, "Machine learning in medicine: Performance calculation of dementia prediction by support vector machines (SVM)," *Inform Med Unlocked*, vol. 16, 2019, doi: 10.1016/j.imu.2019.100200.
[26] H. Cheng, P. N. Tan, and R. Jin, "Efficient algorithm for localized support vector machine," *IEEE Trans Knowl Data Eng*, vol. 22, no. 4, 2010, doi: 10.1109/TKDE.2009.116.

[27] M. A. Nanda, K. B. Seminar, D. Nandika, and A. Maddu, “A comparison study of kernel functions in the support vector machine and its application for termite detection,” *Information (Switzerland)*, vol. 9, no. 1, 2018, doi: 10.3390/info9010005.
[28] M. Kamble, P. Shrivastava, and M. Jain, “Digitized spiral drawing classification for Parkinson’s disease diagnosis,” *Measurement: Sensors*, vol. 16, 2021, doi: 10.1016/j.measen.2021.100047.
[29] Y. Wu and Y. Lu, “An intelligent machine vision system for detecting surface defects on packing boxes based on support vector machine,” *Measurement and Control (United Kingdom)*, vol. 52, no. 7–8, 2019, doi: 10.1177/0020294019858175.
[30] P. O. A. Sunarya, R. Refianti, A. B. Mutiara, and W. Octaviani, “Comparison of accuracy between convolutional neural networks and Naïve Bayes Classifiers in sentiment analysis on Twitter,” *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, 2019, doi: 10.14569/ijacsa.2019.0100511.
[31] R. Malani, A. B. W. Putra, and M. Rifani, “Implementation of the naive bayes classifier method for potential network port selection,” *International Journal of Computer Network and Information Security*, vol. 12, no. 2, 2020, doi: 10.5815/ijcnis.2020.02.04.
[32] N. Rezaeian and G. Novikova, “Persian text classification using naive bayes algorithms and support vector machine algorithm,” *Indonesian Journal of Electrical Engineering and Informatics*, vol. 8, no. 1, 2020, doi: 10.11591/ijeei.v8i1.1696.
[33] J. Guo, B. Wan, H. Wu, Z. Zhao, and W. Huang, “A Virtual Reality and Online Learning Immersion Experience Evaluation Model Based on SVM and Wearable Recordings,” *Electronics (Switzerland)*, vol. 11, no. 9, 2022, doi: 10.3390/electronics11091429.
[34] R. C. Chen, C. Dewi, S. W. Huang, and R. E. Caraka, “Selecting critical features for data classification based on machine learning methods,” *J Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00327-4.