

Comparison of Support Vector Machine (SVM) and Random Forest (RF) Algorithm Performance with Random Undersampling Technique to Predict Gestational Diabetes Mellitus Risk

Annisa Damayanti ^{1*}, Anna Baita ^{2**}

* Informatika, Ilmu Komputer, Universitas Amikom Yogyakarta, Indonesia
annisa_damayanti@students.amikom.ac.id ¹, anna@amikom.ac.id ²

Article Info

Article history:

Received 2024-12-04

Revised 2024-12-30

Accepted 2025-01-21

Keyword:

Gestational Diabetes Mellitus,
Prediction,
Random Forest,
Random Undersampling,
Support Vector Machine.

ABSTRACT

Gestational Diabetes Mellitus (GDM) is a condition of glucose intolerance that develops during pregnancy until the birth process, which is characterized by an abnormal increase in blood sugar levels. Accurate early diagnosis is very important to provide information that can accelerate the treatment process and reduce complications in the mother and baby. One of the machine learning methods that can be used to predict GDM is the Support Vector Machine (SVM) algorithm and the Random Forest (RF) algorithm. This study aims to compare, and evaluate GDM disease prediction models using the SVM and RF algorithms by balancing the target data using the Random Undersampling Technique. The approach using the random undersampling technique managed to increase accuracy by 18% from the accuracy before using the random undersampling technique. The SVM model in this study also uses hyperparameter tuning with kernel parameters, C (cost), and gamma, while the RF model uses Scoring Metrix and four other parameters, namely N_estimators, max_depth, min_samples_split, and min_samples_leaf. The best parameter search process is carried out using GridSearchCV on both models. The results of the study showed that the SVM classification model with random undersampling technique and hyperparameter tuning with K-Fold achieved an average accuracy of 100% with precision, recall, f1-score values also reaching 100%, with the Best Parameter Kernel Linear, C value = 0.1 and gamma value = 0.001 reaching the highest accuracy of 1.0, with a ROC-AUC value of 99% indicating very good prediction performance. While the RF model showed an accuracy result of 99%, tuning was also carried out using the appropriate parameters resulting in the same accuracy of 99%, with a ROC-AUC value of 99% as well. From both models, it shows that the SVM and RF algorithms have very good prediction performance in predicting DMG, but the SVM algorithm can predict DMG better than RF because the number of prediction errors is lower.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Diabetes Melitus Gestasional adalah keadaan *intoleransi glukosa* yang berkembang semasa kehamilan hingga persalinan yang ditandai dengan peningkatan kadar *glukosa* yang tidak wajar pada tubuh dan kurangnya produksi *insulin*[1]. *Insulin* adalah hormon yang dihasilkan oleh sel β di pankreas untuk mengatur kadar glukosa, kurangnya insulin disebabkan kerusakan sel β di pancreas[2]. Berdasarkan data *organisasi Internasional Diabetes Federation* (IDF), hampir

483 juta orang dewasa pada usia 2-79 tahun di dunia hidup dengan diabetes[3]. Angka ini setara dengan sekitar 9,3% dari populasi dewasa global. Prevalensi diabetes lebih tinggi 19.9% pada kelompok usia 65-79 dan diperkirakan terus meningkat dalam beberapa dekade mendatang[4]. Menurut data dari *World Health Organization* (WHO) terdapat 3-5% ibu hamil yang mengalami DMG, dengan jumlah kasusnya mencapai sekitar 20.000 orang setiap tahunnya[5]. Ibu hamil dengan riwayat DMG memiliki risiko keguguran, bayi lahir prematur dan ibu mengalami persalinan caesar[6]. *American*

Pregnancy Association (APA) mengatakan pemeriksaan kadar gula darah saat hamil bertujuan untuk mengetahui ibu hamil mengidap DMG atau tidak[7]. Oleh karena itu, penting untuk memprediksi dan melakukan klasifikasi pada penderita DMG.

Teknologi kecerdasan buatan (*Artificial Intelligence*) menjadi salah satu teknologi yang signifikan dalam beberapa tahun terakhir dan *machine learning* telah menunjukkan potensi yang besar untuk pengklasifikasian[8]. Dalam dunia kesehatan, penerapan *machine learning* telah membuka peluang baru untuk diagnosis yang cepat dan akurat[9]. Algoritma *machine learning* yang populer dan efektif di dunia kesehatan antara lain: *Logistic Regression*[10], [11] *Random Forest*[10], [12], *K-Nearest Neighbors* (KNN)[10], [11], *Neural Network*[12], *Naïve Bayes*[11], dan *Support Vector Machine* (SVM)[12], [13], [14]. Dengan kemampuannya dalam mengklasifikasikan data kompleks, SVM telah membuka peluang baru untuk diagnosis yang lebih akurat, pengobatan yang lebih efektif, dan peningkatan kualitas hidup pasien[12], [13], [14]. Algoritma SVM merupakan algoritma klasifikasi data *linear* dan *non-linear* yang melibatkan data latih untuk memprediksi kelas berdasarkan data kompleks[14]. Dalam konteks SVM, pemilihan *hyperparameter tuning* memiliki peran penting dalam menentukan cara data ditransformasikan dalam *hyperline*. *Hyperparameter tuning* meliputi parameter *cost* (C), *gamma*, dan *kernel*[12].

Terdapat beberapa penelitian yang menerapkan *machine learning* untuk memprediksi penyakit diabetes. Penelitian Aldama dan Nasir menggunakan algoritma SVM untuk pengklasifikasian diagnosis pasien diabetes, tetapi hasil akurasi terlalu rendah (65%) untuk mendiagnosis penyakit, sehingga memiliki ruang untuk perbaikan[13]. Penelitian Ridwan dan Setiawan melakukan perbandingan dengan beberapa model *machine learning* seperti *K-Nearest Neighbors* (KNN), *AdaBoost*, *Logistic Regression* (LR), *Light Gradient Boosting* (LGB), *Random Forest* (RF), dan *Support Vector Machine* (SVM). Dataset yang digunakan adalah dataset Pima Indian dan mengalami ketidakseimbangan kelas, untuk mengatasi hal tersebut, peneliti menggunakan *Teknik Over-sampling Minoritas Sintetis* (SMOTE) untuk meningkatkan kinerja model. Dari hasil perbandingan algoritma dengan teknik SMOTE tersebut KNN menunjukkan performa terbaik[10]. Penelitian Dewi, dkk melakukan perbandingan dua algoritma *Neural Network* (ANN) dan *Support Vector Machine* (SVM), dan menggunakan parameter untuk menaikkan performa, ANN menggunakan parameter *learning rate* dan *momentum*, sedangkan SVM menggunakan fungsi *kernel Radial Basis Function* (RBF) dan parameter *gamma* dan C. Berdasarkan hasil analisis algoritma ANN menunjukkan klasifikasi yang lebih baik[12]. Dari ketiga penelitian di atas menunjukkan SVM memiliki performa yang lebih rendah dibandingkan dengan algoritma lainnya, tetapi SVM memiliki potensi untuk menunjukkan hasil akurasi yang terbaik jika dilakukan penanganan terhadap data. *Teknik oversampling* (SMOTE) atau *undersampling* untuk menangani ketidakseimbangan

kelas, pemilihan fitur yang relevan dan penggunaan *hyperparameter tuning* dapat menjadi solusi untuk meningkatkan performa SVM[15]. Penelitian Jeong, dkk menyimpulkan bahwa, pendekatan *undersampling* terbukti efektif untuk menangani kumpulan data yang tidak seimbang dalam pengklasifikasian. Secara signifikan juga meningkatkan kinerja model dalam menghasilkan akurasi klasifikasi yang tinggi[16].

Selanjutnya penelitian Yolanda, dkk melakukan implementasi algoritma *Neighbor Weighted K-Nearest Neighbor* (NWKNN), NWKNN merupakan metode pengembangan dari KNN untuk mengklasifikasikan data tidak seimbang. Dataset yang digunakan memiliki karakteristik tidak seimbang sehingga algoritma NWKNN diterapkan pada dataset tersebut. Berdasarkan pengujian yang telah dilakukan menunjukkan kinerja yang baik, dan efektif dalam mengidentifikasi kasus diabetes gestasional[17]. Penelitian Delvika, dkk melakukan perbandingan klasifikasi antara *Naïve Bayes* dan KNN untuk mendiagnosa risiko diabetes pada ibu hamil. Setelah melakukan perbandingan hasil menunjukkan bahwa NBC memiliki nilai lebih tinggi dibanding KNN dibuktikan *K-Fold* K=10[11]. Penelitian Simanjuntak dan Hasibuan menerapkan metode *Principal Component Analysis* (PCA) untuk mengurangi dimensi data yang awalnya 12 variabel menjadi 9 komponen utama untuk mengidentifikasi kombinasi *linear* dan menerapkan *K-Means Clustering*. Hasil penelitian menunjukkan bahwa kombinasi metode PCA dan *K-Means Clustering* merupakan pendekatan yang efektif untuk mengidentifikasi pasien diabetes[18].

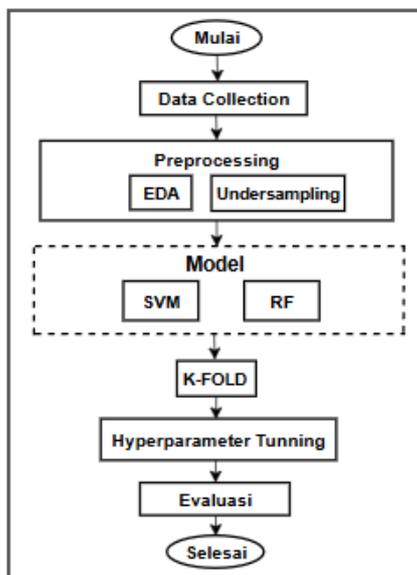
Penelitian Saputro dan Rosiyadi menguji metode *Random Over-Under Sampling* (ROUS) pada dataset yang tidak seimbang untuk mengatasi ketidakseimbangan kelas. Penelitian ini menggunakan teknik *resampling* ROUS dan *non-resampling* dengan algoritma C4.5, KNN, SVM, ANN, *Naive Bayes* dan *Random Forest*. Hasil dari metode ROUS meningkatkan kinerja semua model kecuali KNN yang menurun, sedangkan RF memberikan hasil terbaik dengan metode ROUS dalam klasifikasi pasien diabetes. Model ROUS menunjukkan kinerja yang sangat baik dan dapat digunakan dalam klasifikasi penyakit diabetes[19]. Penelitian Wafa, dkk melakukan penerapan *forward selection* dan *kernel RBF* pada algoritma SVM menghasilkan model yang lebih baik dibandingkan dengan C4.5, KNN, dan *Naïve Bayes*[20].

Klasifikasian risiko DMG menggunakan AI, khususnya algoritma SVM menunjukkan potensi yang baik untuk prediksi DMG. Terdapat beberapa penelitian menunjukkan SVM memiliki kinerja yang lebih rendah dari algoritma lain, kinerja SVM dapat ditingkatkan melalui penanganan data yang tepat, pengaturan parameter, *teknik forward selection*, *undersampling*, *k-fold*, *hyperparameter tuning* dan pemilihan *kernel* yang sesuai seperti *kernel linear*, *kernel RBF*, *kernel polynomial*, dan *kernel sigmoid*. Belum ada penelitian yang mengaplikasikan algoritma SVM dan RF, *teknik random undersampling* dan penyesuaian parameter untuk DMG. Karena itu, penulis bertujuan untuk mengimplementasikan algoritma SVM dan RF dengan teknik *random undersampling*

dan penyesuaian parameter pada *Diabetes Health Indicators Dataset*, yang memiliki distribusi data tidak seimbang yang kemudian divalidasi dengan *k-fold*. Dengan melakukan implementasi algoritma SVM dan RF dengan Teknik *random undersampling* diharapkan penelitian ini dapat memberikan hasil yang baik dalam memprediksi risiko terkena DMG secara akurat.

II. METODE

Penelitian ini dilakukan dengan menerapkan salah satu model *machine learning* dengan algoritma *Support Vector Machine (SVM)* dan *Random Forest (RF)*. Metode SVM RF digunakan untuk memprediksi risiko terkena penyakit DMG. Pembelajaran mesin terdiri dari beberapa tahapan proses penelitian yang berperan penting dalam membangun model yang efektif. Alur penelitian dapat dilihat pada Gambar 1.



Gambar 1. Alur Penelitian

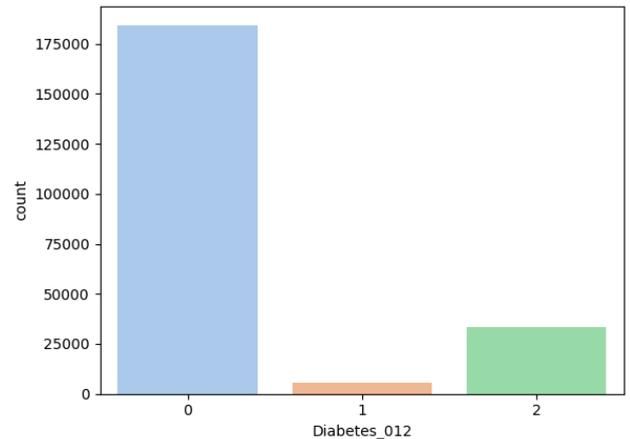
A. Data Collection

Jenis penelitian ini adalah penelitian kuantitatif dan menggunakan data numerik yang berbentuk binary untuk menguji performa SVM dalam memprediksi risiko DMG. Dataset yang digunakan bersumber dari situs Kaggle[21]. Untuk identifikasi lebih jelasnya dapat dilihat pada identifikasi data pada tabel I.

TABEL I
IDENTIFIKASI DATA

Judul	Diabetes Health Indicators Dataset
Dataset	diabetes_012_health_indicators_BRFSS2021
Atribut	22
Total Data	236.378
Duplicated	12.828
Imbalance Degree	ID : 0.67 (Dataset tidak seimbang, dengan kelas minoritas jauh lebih kecil daripada kelas mayoritas)
Target	Diabetes_012 (0=tidak DMG, 1=praDMG, 2=ya DMG)

Pada tabel I dapat dilihat jumlah sampel atau ukuran datasetnya berisi 236.378, dan memiliki 22 atribut atau fitur. Dataset ini memiliki tiga kelas target dengan distribusi class pasien sehat (0), praDMG (1), dan menderita DMG (2), yang dapat dilihat pada gambar 2.



Gambar 2. Bar Chart Distribusi Class Diabetes_012

Pada gambar 2 dapat dilihat dengan jelas bahwa dataset memiliki data tidak seimbang, yang dibuktikan dengan pengecekan *imbalance degree* senilai 0.67 pada fitur Diabetes_012 dengan kelas minoritas jauh lebih kecil daripada kelas mayoritas. Distribusi label Diabetes_012 pada kelas 0 = 184.542, kelas 1 = 5.613, dan kelas 2 = 33.395 butuh penanganan lebih lanjut dengan *oversampling* atau *undersampling*. Penggunaan *imbalance degree* dilakukan karena dataset ini merupakan *multiclass* dan *binary*.

B. Preprocessing Data

Preprocessing data bertujuan untuk membersihkan dan mengubah data mentah agar sesuai untuk digunakan dalam model pembelajaran mesin. Pada tahap ini dilakukan pembersihan data dengan menghapus data *duplikat* pada dataset. Data *duplikat* ditangani untuk meningkatkan kualitas data dan meningkatkan pemrosesan data, kemudian mengatasi *missing values*, mengatasi format yang tidak sesuai seperti dataset yang tidak seimbang dan butuh penanganan dengan Teknik *Random Undersampling*.

C. Exploratory Data Analysis (EDA)

Tahap eksplorasi data melibatkan pemeriksaan jumlah data dari masing-masing kelas yang ada pada fitur Diabetes_012 yang digunakan sebagai data target, jumlah dari masing-masing kelas seimbang atau tidak. Hal ini dilakukan untuk melihat *signifikansi* dari data Diabetes_012 apakah perlu ditangani dengan teknik penyeimbangan data antar kelas pada fiturnya atau tidak. Tahap selanjutnya, *transformasi* data dilakukan untuk membantu meningkatkan performa model *machine learning* dan mempermudah *interpretasi* hasil model. Analisis fitur juga dilakukan untuk memilih fitur-fitur terbaik menggunakan RFE. *Recursive Feature Elimination (RFE)* adalah metode yang digunakan untuk meningkatkan

kualitas data dengan memilih fitur atau atribut yang paling relevan. RFE menjadi metode yang populer untuk seleksi fitur dalam berbagai aplikasi *machine learning*, seperti klasifikasi dan prediksi terutama pada algoritma SVM[22]. RFE menghasilkan 10 fitur terbaik dari 21 fitur awal, berikut ini fitur yang dihasilkan RFE dapat dilihat pada table II.

TABEL II
HASIL TRANSFORMASI DATA DENGAN RFE

No	Fitur	Type Data
1	HighBP	Int64
2	HighChol	Int64
3	CholCheck	Int64
4	Stroke	Int64
5	HeartDiseaseorAttack	Int64
6	PhysActivity	Int64
7	HvyAlcoholConsump	Int64
8	GenHlth	Int64
9	DiffWalk	Int64
10	Sex	Int64

Dengan hanya memilih 10 fitur terbaik pada dataset dengan 21 fitur, model akan menangkap interaksi yang kompleks antar variable dengan lebih baik, dan model menjadi lebih efisien karena hanya fokus pada fitur yang relevan.

D. Handling Imbalance Data

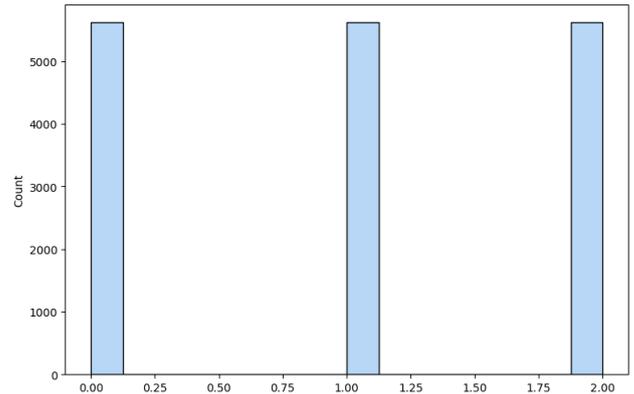
1) Teknik Random Undersampling

Metode yang digunakan untuk mengatasi masalah ketidakseimbangan kelas dalam dataset adalah *resampling*, yaitu menambahkan *record* ke kelas minoritas atau menghapus *record* dari kelas mayoritas. *Random Undersampling* digunakan untuk mengekstrak informasi yang lebih akurat dari kumpulan data yang awalnya tidak seimbang dan menghasilkan sample sintetis[15]. Hasil dari kelas data sebelum dan sesudah *Random undersampling* dapat dilihat pada tabel III.

TABEL III
TEKNIK RANDOM UNDERSAMPLING

Diabetes_012	Jumlah baris data	
	Sebelum	Sesudah
Kelas 0	33720	5619
Kelas 1	16860	5619
Kelas 2	5619	5619
Total data	223.550	16.857

Tabel III menunjukkan bahwa data sebelum melakukan *random undersampling* mengalami data tidak seimbang, maka diperlukan penanganan dan setelah melakukan penanganan menggunakan *teknik random undersampling* menghasilkan sampel sintetis baru menjadi data seimbang. Dengan 16.857 baris data dan masing-masing atribut bernilai yang sama yaitu 5619 baris data. Visualisasi dari data yang sudah seimbang dengan teknik *random undersampling* dapat dilihat pada gambar 3.



Gambar 3. Visualisasi jumlah data setelah *random undersampling*

Gambar 3 merupakan distribusi jumlah kelas setelah melakukan penanganan dengan teknik *random undersampling* menjadi data yang seimbang.

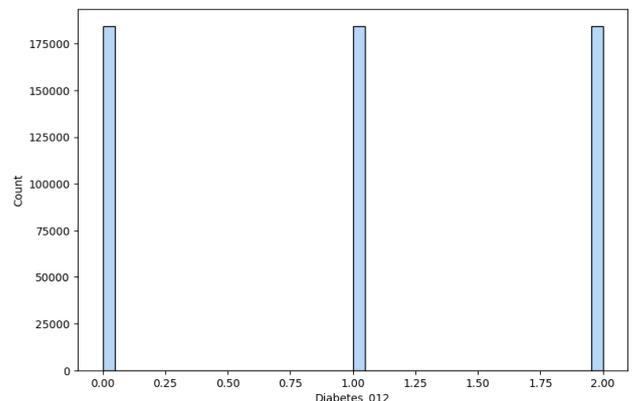
2) Teknik Oversampling (SMOTE)

Teknik SMOTE digunakan untuk mengatasi masalah *imbalance* pada dataset, SMOTE dapat meningkatkan kelas minoritas ke mayoritas dalam dataset sehingga menjadi data yang seimbang dan menghasilkan sample sintetis[23]. Hasil sebelum dan sesudah teknik SMOTE dapat dilihat tabel IV.

TABEL IV
TEKNIK SMOTE

Diabetes_012	Jumlah baris data	
	Sebelum	Sesudah
Kelas 0	184542	184542
Kelas 1	33395	184542
Kelas 2	5613	184542
Total data	223.550	553.626

Tabel IV menunjukkan bahwa data sebelum melakukan SMOTE mengalami data tidak seimbang, maka diperlukan penanganan dan setelah melakukan penanganan dengan teknik SMOTE menghasilkan sintetis baru dan menjadi data yang seimbang.



Gambar 4. Visualisasi jumlah data setelah SMOTE

Dengan 553.626 total baris data dan masing-masing atribut atau fitur memiliki nilai yang sama yaitu 184.542 baris data. Visualisasi dari data yang sudah seimbang dengan menggunakan SMOTE dapat dilihat pada gambar 4.

Setelah melakukan perbandingan antara *Random Undersampling* dan SMOTE, peneliti memilih teknik *Random Undersampling* untuk dijadikan teknik penanganan data yang tidak seimbang karena pada pemodelan *machine learning random undersampling* dapat membaca ke-tiga kelas lebih optimal dapat dilihat pada gambar 5.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	547
1	1.00	1.00	1.00	594
2	1.00	1.00	1.00	545

Gambar 5. Hasil pemodelan ML random undersampling

Dibandingkan dengan SMOTE yang tidak dapat membaca kelas minoritas yaitu kelas 1 dan 2 dapat dilihat gambar 6.

	precision	recall	f1-score	support
0	0.83	1.00	0.91	36982
1	0.00	0.00	0.00	1153
2	0.00	0.00	0.00	6575

Gambar 6. Hasil pemodelan ML SMOTE

Dengan hasil pemodelan *machine learning* tersebut peneliti memilih untuk menggunakan teknik *random undersampling* sebagai teknik *imbalance* data pada dataset yang digunakan.

E. Support Vector Machine (SVM)

Support Vector Machine adalah metode klasifikasi untuk data *linear* dan *non-linear*. SVM merupakan salah satu teknik *machine learning* yang banyak digunakan untuk klasifikasi dan regresi. SVM juga menggunakan fungsi matematika yang didefinisikan sebagai kernel yang secara *sistematis* menciptakan *support vector classifier* dalam dimensi yang lebih tinggi. Peneliti menggunakan empat jenis *kernel*, yaitu: *kernel linear* yang hanya melakukan *dot product* dari dua vektor fitur, *polynomial* mampu memisahkan data yang tidak dapat dipisahkan secara *linear*, *Radial Basic Function* (RBF) untuk mengukur kesamaan antara dua titik data dalam ruang fitur, dan *sigmoid* digunakan pada data yang memiliki struktur yang kompleks dan *non-linear*. untuk mengevaluasi kinerja model.

Rumus parameter kernel:

1) Linear Kernel

$$\kappa(x, y) = x \cdot y \quad (1)$$

Keterangan:

- x = fitur dari satu titik data
- y = fitur dari titik data lain

2) Polynomial Kernel

$$\kappa(x, y) = (\gamma x \cdot y + r)^d \quad (2)$$

Keterangan:

- γ = parameter bebas yang menskalakan produk dalam vector
- r = parameter yang memungkinkan penyesuaian, dan
- d = derajat dari *polynomial*.

3) Radial Basis Function (RBF) Kernel

$$\kappa(x, y) = \exp(-\gamma \|x - y\|^2) \quad (3)$$

Keterangan:

- \exp = *eksponensial*. Secara matematis
- $\exp(z) = e^z$, dimana e adalah *konstanta Eluler* dan
- γ = menentukan seberapa jauh pengaruh satu titik data terhadap titik lainnya.

4) Sigmoid Kernel

$$\kappa(x, y) = \tanh(\gamma x \cdot y + r) \quad (4)$$

Keterangan:

- \tanh = *hyperbolic tangent*,
- γ = parameter bebas yang menskalakan produk dalam vector, dan
- r = *konstanta*.

F. Random Forest (RF)

Random Forest adalah pembelajaran mesin berbasis *ensemble* yang digunakan untuk tugas klasifikasi dan regresi. Algoritma RF menggunakan sekumpulan predictor pohon keputusan dan termasuk dalam *supervised learning*[24]. Cara kerja dari RF ini dengan mengambil n buah data point dari suatu dataset kemudian menyatukannya dan menghasilkan keputusan yang stabil[25]. Peneliti menggunakan matrik *Scoring Metrics*, dan empat parameter, yaitu $n_estimators$, max_depth , $min_samples_plit$, dan $min_samples_leaf$. Dari parameter-parameter ini dapat memberikan control fleksibel atas kompleksitas dan performa *Random Forest*.

G. Evaluasi

Confusion matrix adalah alat yang digunakan dalam *machine learning* dan statistik untuk mengevaluasi kinerja model klasifikasi. *Confusion matrix* memberikan gambaran yang lebih detail bagaimana model klasifikasi bekerja, dibandingkan hanya menggunakan metrik *accuracy*. *Confusion matrix* berbentuk tabel yang terdiri dari empat komponen utama, dapat dilihat pada tabel V.

TABEL V
CONFUSION MATRIX

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

Nilai dari *confusion matrix* ini yang akan digunakan untuk menghitung evaluasi *performance metric* yang mengukur kinerja model yang telah dibuat. *Performance metric* yang biasa digunakan, yaitu: *accuracy*, *precision*, *recall* atau *sensitivity*, dan *f1-score*. Rumus untuk menghitung *performance metric*:

1) Accuracy

$$(accuracy) = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (5)$$

2) Precision

$$(precision) = \frac{TP}{TP + FP} \times 100\% \quad (6)$$

3) Recall

$$(recall) = \frac{TP}{TP + FN} \times 100\% \quad (7)$$

4) F1-Score

$$(f1 - score) = 2 \times \frac{recall \times precision}{recall + precision} \quad (8)$$

H. Hyperparameter Tuning

Hyperparameter tuning adalah proses mencari kombinasi *hyperparameter* terbaik yang memungkinkan model menghasilkan performa optimal. Proses ini penting karena *hyperparameter* yang dipilih dapat secara signifikan memengaruhi hasil akurasi, efisiensi dan generalisasi model. Metode *hyperparameter* yang digunakan yaitu metode *GridSearchCV*. *GridSearchCV* adalah salah satu metode *hyperparameter tuning* yang bekerja dengan membangun model untuk setiap kombinasi *hyperparameter* yang ada dalam *grid* dan mengevaluasi setiap kombinasi berdasarkan performa pada data validasi.

I. K-Fold Cross Validation

K-Fold Cross Validation adalah teknik yang digunakan untuk mengevaluasi kinerja suatu model *machine learning* dan merupakan teknik yang sangat penting dalam membangun model SVM yang baik. Teknik ini membagi dataset menjadi beberapa *fold* (lipatan) yang sama besar. Setiap *fold* secara bergantian akan digunakan sebagai data *testing* dan sisanya dijadikan data *training*, proses ini diulang

sebanyak jumlah *fold* yang telah ditentukan $k=10$. Jika kinerja model konsisten setiap *fold*, maka model dapat dianggap generalisasi, stabil dan memiliki kinerja yang optimal.

III. HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk membandingkan, dan mengembangkan model prediksi Diabetes Melitus Gestasional (DMG) yang akurat dengan menerapkan algoritma *machine learning*, yaitu *Support Vector Machine* (SVM) dan *Random Forest* (RF). Model ini dikembangkan dengan menggabungkan teknik *random undersampling* untuk mengatasi ketidakseimbangan kelas data, *hyperparameter tuning* untuk mengoptimalkan kinerja model dan validasi *K-Fold* untuk mengevaluasi performa model secara konsisten. Penggunaan *random undersampling* pada penelitian ini diharapkan dapat mengatasi masalah data yang tidak seimbang dan meningkatkan keakuratan model dalam memprediksi atau mengidentifikasi kasus DMG.

A. Support Vector Machine (SVM)

1) Teknik Random Undersampling & K-Fold CV

Penerapan teknik *random undersampling* dan *K-Fold Cross Validation* menunjukkan perbaikan dalam keseimbangan kelas mayoritas dan minoritas, ditunjukkan dengan hasil tabel VI.

TABEL VI
TEKNIK RANDOM UNDERSAMPLING SVM

Metrik	20% Testing		10 Fold	
	Sebelum undersampling	Setelah undersampling	Sebelum undersampling	Setelah undersampling
Accuracy	0.82	1.00	0.83	1.00
Precision	0.83	1.00	0.83	1.00
Recall	1.00	1.00	1.00	1.00
F1-Score	0.91	1.00	0.90	1.00

Tabel VI menunjukkan bahwa setelah menerapkan teknik *undersampling*, model menjadi sangat akurat dengan semua metrik mencapai 100%, dan distribusi kelas terdeteksi atau terbaca. Kemudian menerapkan metode *K-Fold CV* dengan *fold* yang digunakan $K=10$ menunjukkan performa yang baik, dan mengindikasikan bahwa model SVM memiliki performa yang stabil dengan generalisasi yang lebih kuat dibanding hanya melakukan pengujian *random undersampling* 20% testing.

2) Hyperparameter Tuning

Hyperparameter tuning pada algoritma SVM dilakukan menggunakan *GridSearchCV* untuk menemukan kombinasi optimal *parameter* SVM, seperti *kernel*, *C* (*Cost*) dan *gamma*. Kombinasi optimal yang akan dihasilkan *kernel linear* dengan nilai *C* dan *gamma* yang telah ditentukan dan akan memberikan performa terbaik, dan best parameter dapat dilihat pada tabel VIII.

TABEL VII
NILAI & PARAMETER PADA HYPERPARAMETER TUNING ALGORITMA SVM

Parameter	Value	Best param	Accuracy
Kernel	Linear	Linear	1.0
C (Cost)	0.1; 1.0; 10.0; 100.0	0.1	
Gamma	0.001; 0.01; 0.1; 1	0.001	

Hasil *tuning parameter* pada tabel VII menunjukkan parameter terbaik dengan akurasi 1.0 pada data pengujian yaitu *kernel linear*. Kernel terbaik yang dihasilkan oleh *GridSearchCV* yaitu *kernel Linear*, dengan nilai C = 0.1 dan nilai gamma = 0.001. Hasil *tuning best parameter* menunjukkan bahwa *kernel liner* mampu mencapai akurasi tertinggi sebesar 1.0. Pemilihan kernel linear selain dari akurasi tertinggi juga tidak teridentifikasi overfitting. Oleh karena itu, *kernel linear* menjadi pilihan terbaik karena tidak memberikan akurasi yang efisien, dan generalisasi.

Tabel VII juga membuktikan bahwa model dapat lebih efektif dalam mengklasifikasikan kasus DMG. Setelah melakukan *hyperparameter tuning* pada algoritma SVM, hasilnya menunjukkan tidak ada peningkatan atau penurunan yang signifikan dalam akurasi prediksi DMG. *Hyperparameter tuning* yang diharapkan dapat menstabilkan performa model ternyata tidak memberikan dampak signifikan pada akurasi karena sebelum tuning model yang dihasilkan sudah sangat akurat, hal ini menunjukkan bahwa kombinasi *undersampling*, *K-Fold*, SVM, dan RF sudah mencapai batas optimal, dan *tuning* lebih lanjut tidak memberikan dampak yang berarti.

Evaluasi metrik kinerja untuk *training data* dan *validation data* dilakukan untuk pengecekan *overfitting* dan *underfitting*. Metrik yang digunakan untuk evaluasi yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-Score* untuk performa model. Model dapat dinyatakan *overfitting* jika selisih performa antar data training dan testing signifikan atau akurasi pada data train sangat tinggi sedangkan pada data testing lebih rendah, sedangkan *underfitting* jika performa antar data train dan testing sama-sama rendah. Pada algoritma SVM menunjukkan hasil performa yang baik dan seimbang pada data train dan testing, dapat dilihat pada gambar 7.

```

Training Metrics:
Accuracy: 1.00
Precision: 1.00
Recall: 1.00
F1-Score: 1.00

Validation Metrics:
Accuracy: 1.00
Precision: 1.00
Recall: 1.00
F1-Score: 1.00

Model memiliki performa yang baik dan seimbang pada data pelatihan dan pengujian.

```

Gambar 7. Pengecekan overfitting & underfitting SVM

B. Random Forest (RF)

1. Teknik Random Undersampling & K-Fold CV

Metode *Random Undersampling* dan *K-Fold Cross Validation* juga diterapkan pada algoritma RF tetapi diuji dengan metrik yang berbeda, dengan penanganan yang sama yaitu 20% pengujian testing yang dapat dilihat pada tabel VIII.

TABEL VIII
TEKNIK RANDOM UNDERSAMPLING & K-FOLD CV RF

Metrik	20% Testing		10 Fold	
	Sebelum undersampling	Setelah undersampling	Sebelum undersampling	Setelah undersampling
Accuracy	0.82	0.99	0.83	0.99
Precision	0.77	1.00	0.78	0.99
Recall	0.82	1.00	0.83	0.99
F1-Score	0.78	1.00	0.78	0.99

Pada tabel VIII menunjukkan bahwa penggunaan *random undersampling* membuat distribusi data lebih seimbang, sehingga model dapat memprediksi dengan akurasi yang lebih tinggi seperti pada tabel VIII diatas. Pada RF diuji dengan *K-Fold CV* menggunakan metrik *Scoring Metrics*. *Scoring Metrics* memberikan panduan kuantitatif tentang seberapa baik model dapat memprediksi data berdasarkan hasil penelitian dan validasi. Meskipun *Scoring Metrics* bukan bagian khusus dari algoritma RF, *Scoring Metrics* merupakan alat evaluasi yang digunakan untuk menilai kinerja model, termasuk RF. Penerapan *K-Fold CV* juga memastikan stabilitas dan konsistensi dari performa model. Nilai *fold* yang digunakan K=10, karena memberikan estimasi kinerja yang lebih stabil dan akurat. Hasil menggunakan *K-Fold CV* K=10 menunjukkan bahwa model memiliki kinerja yang baik dan konsisten metrik evaluasi. Nilai metrik yang tinggi dan stabil menegaskan bahwa model tidak hanya akurat tetapi juga generalisasi dengan baik terhadap data baru.

2. Hyperparameter Tuning

Hyperparameter tuning juga dilakukan pada algoritma RF untuk mendapatkan penanganan yang sama agar dapat dibandingkan hasil akhirnya. *GridSearchCV* digunakan untuk menemukan kombinasi parameter RF yang optimal, parameternya menghasilkan nilai akurasi yang baik dan dapat dilihat pada tabel IX.

TABEL IX
HYPERPARAMETER TUNING ALGORITMA RF

Parameter	Value	Accuracy
N_estimators	10, 20, 30	0.99
Max_depth	10, 20, 30	
Min_samples_split	2, 5, 10, 15, 20	
Min_samples_leaf	1, 2, 4, 6, 8	

Pada tabel IX menunjukkan bahwa kombinasi parameter terbaik untuk model RF menghasilkan akurasi 99% yang sangat tinggi. Ini menunjukkan bahwa model bekerja dengan

baik pada data latih sekaligus memiliki kemampuan generalisasi yang baik terhadap data uji. Dengan kata lain, parameter ini telah mengoptimalkan keseimbangan antara bias dan *variance*, sehingga menghasilkan kinerja yang sangat baik.

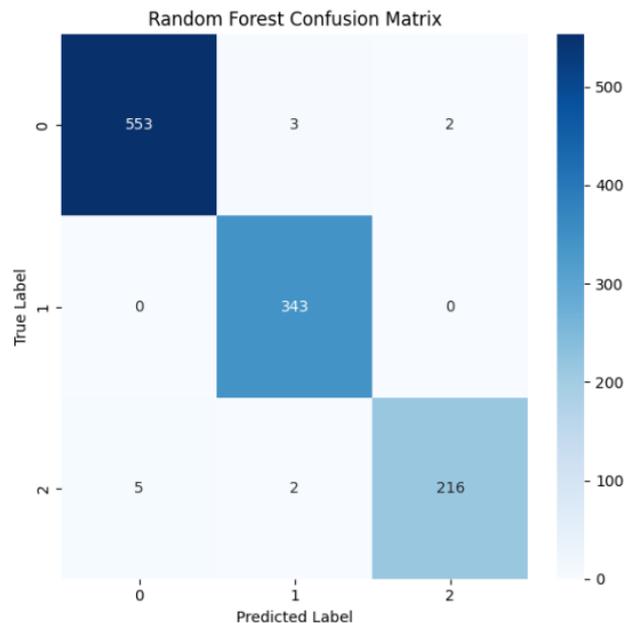
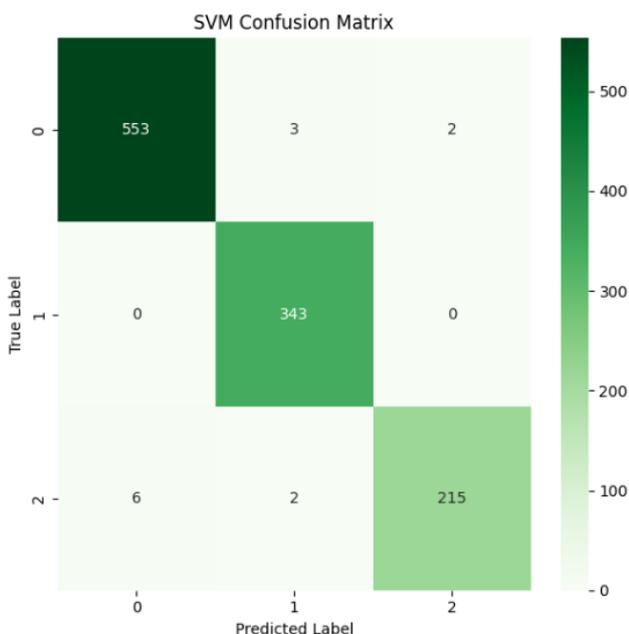
Metode evaluasi performa model berdasarkan data training dan testing dilakukan untuk pengecekan overfitting dan underfitting. Pada algoritma RF menunjukkan hasil overfitting dengan akurasi training data sebesar 1.00 dan akurasi testing data 0.49 dapat dilihat pada gambar 8.

Accuracy on Training Data: 1.00
 Accuracy on Testing Data: 0.52
 Model menunjukkan overfitting.

Gambar 8. Pengecekan overfitting & underfitting RF

C. Confusion Matrix SVM & RF

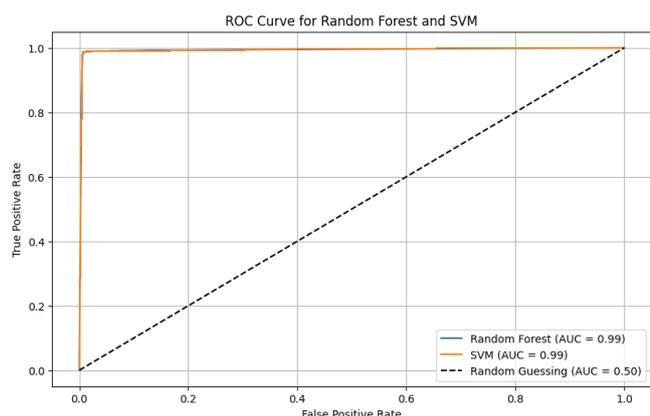
Berikut ini ditampilkan visualisasi *confusion matrix* dari algoritma SVM dan RF. *Confusion matrix* adalah representasi visual yang menunjukkan performa model klasifikasi dengan membandingkan prediksi yang dibuat model terhadap nilai target actual, dan dapat dilihat pada gambar 8.



Gambar 8. Confusion Matrix Algoritma SVM & RF

Pada gambar 8 menunjukkan *confusion matrix* dari kedua model yaitu SVM dan RF. SVM dan RF memiliki performa yang hampir identic dengan tingkat kesalahan yang rendah. Kedua model menunjukkan akurasi yang sangat baik, terutama dalam mengklasifikasikan kelas 0 (lebih dari 550 *instance* benar), kelas 1 (343 *instance* benar), dan kelas 2 (216 *instance* benar). Kesalahan tertinggi terjadi pada kelas 2, di mana beberapa *instance* (6 *instance*) salah diprediksi sebagai kelas 0 untuk kedua model. Hal ini menunjukkan bahwa model mengalami sedikit kesulitan dalam membedakan antara kelas 2 dan kelas 0. Berdasarkan *confusion matrix* ini, SVM lebih baik dalam memprediksi, karena jumlah kesalahan prediksinya lebih rendah dibandingkan *random forest*.

Model akhir yang telah dilatih dengan *teknik undersampling* dan *hyperparameter tuning* serta divalidasi dengan *K-Fold* mencapai akurasi rata-rata sebesar 100%, dengan nilai *precision*, *recall*, *f1-score*nya juga mencapai 100%. Nilai akurasi yang mencapai 100% pada prediksi risiko DMG sangat mungkin terjadi karena proses EDA yang menyeluruh dan penanganan data yang tepat. Dataset yang digunakan penulis merupakan karakteristik dataset binary yang terdiri dari dua kelas yaitu positif dan negative atau (0 dan 1). Hasil akurasi, *precision*, *recall*, *f1-score* yang mencapai 100% pada model ini bukan semata-mata disebabkan oleh kebetulan, melainkan karena proses EDA yang komprehensif dan penanganan karakteristik data binary yang tepat dan cermat. Dengan demikian dapat disimpulkan hasil penelitian ini menunjukkan bahwa algoritma SVM dengan *teknik undersampling* dan *hyperparameter tuning* serta divalidasi dengan *K-Fold* memberikan kinerja yang sangat baik untuk memprediksi risiko terkena DMG yang diperkuat dengan nilai ROC-AUC mencapai 0.99 dapat dilihat pada gambar 9.



Gambar 9. Curva ROC-AUC Algoritma SVM & RF

Pada gambar 9 menunjukkan grafik dari kedua algoritma SVM dan RF yang memiliki kurva yang identik. Kurva ROC ini membandingkan kinerja dari dua model klasifikasi SVM dan RF, kurva ini menunjukkan kedua model memiliki performa prediksi yang sangat baik. AUC kurva untuk kedua model SVM dan RF adalah 0.99, yang menunjukkan bahwa model sangat baik dalam memisahkan kelas positif dan negatif dan baik dalam memprediksi data dengan *threshold* yang berbeda. Pengecekan ROC-AUC ini menggunakan pendekatan One-vs-Rest (OvR) atau biasa dikenal One-vs-All untuk klasifikasi biner dalam multiclass classification.

- Nilai AUC mendekati 1 berarti performa klasifikasi memiliki kinerja yang sangat baik dalam membedakan kelas positif dan negatif.
- Kedua model memiliki *True Positive Rate* (TPR) yang mendekati 1.0 di hampir semua *threshold*, yang berarti model mampu mengidentifikasi hampir semua sampel positif dengan benar
- Kedua model memiliki *False Positive Rate* (FPR) yang mendekati 0.0 di sebagian besar *threshold*, yang berarti model jarang salah memprediksi kelas *negative* sebagai *positive*.
- Titik kurva yang dekat dengan sudut kiri atas menunjukkan keseimbangan optimal antara TPR dan FPR.
- Garis diagonal putus-putus dengan $AUC = 0.50$ mempresentasikan performa tebakan acak (random guessing).

IV. KESIMPULAN

Penelitian ini berhasil membandingkan model dan mengembangkan model prediksi DMG menggunakan algoritma SVM dan RF yang diperkuat dengan *teknik undersampling*, dan *K-Fold Cross Validation*. Pada algoritma SVM dilakukan penerapan *K-Fold CV* dengan $K=10$ memastikan performa model yang stabil, ditunjukkan dengan hasil evaluasi pada berbagai kernel SVM yang memiliki kinerja yang konsisten di setiap foldnya. Selain itu, *tuning parameter* dengan *GridSearchCV* berhasil

mengidentifikasi *kernel Linear*, $C = 0.1$, dan $\gamma = 0.001$ mencapai akurasi tertinggi 1.0, dengan nilai ROC-AUC 99% menunjukkan performa prediksi yang sangat baik, tetapi penerapan *hyperparameter tuning* ini tidak meningkatkan atau menurunkan akurasi dan tidak memberikan dampak yang *signifikan* terhadap model. Sedangkan pada model RF juga dilakukan penanganan yang sama, tetapi metrik yang berbeda yaitu metrik *scoring matrix* dan menghasilkan nilai akurasi 0.99 juga dilakukan *tuning* menggunakan parameter yang sesuai dan menghasilkan akurasi yang sama yaitu 0.99, dengan nilai ROC-AUC 99% menunjukkan performa prediksi yang sangat baik. Hal ini menunjukkan bahwa penerapan *hyperparameter tuning* tidak meningkatkan atau menurunkan akurasi dan tidak memberikan dampak yang *signifikan* terhadap kedua model karena sebelum melakukan *hyperparameter tuning* model yang dihasilkan sudah sangat akurat. Dari kedua model menunjukkan bahwa algoritma SVM dan RF memiliki performa prediksi yang sangat baik dalam memprediksi DMG. Tetapi pada algoritma SVM dapat memprediksi DMG lebih baik karena jumlah kesalahan prediksinya lebih rendah dan tidak teridentifikasi *overfitting*, sedangkan algoritma RF teridentifikasi *overfitting*.

Model akhir yang dihasilkan algoritma SVM mencapai akurasi rata-rata 100% dengan nilai *precision*, *recall*, *f1-score*nya juga mencapai 100%, menandakan keseimbangan tinggi dalam memprediksi DMG. Hasil ini menunjukkan bahwa pendekatan yang diterapkan tidak hanya relevan dalam meningkatkan akurasi prediksi DMG tetapi juga stabil pada dataset yang besar dan tidak seimbang, sehingga berpotensi menjadi metode yang efektif dalam mendiagnosis DMG. Saran untuk pengguna yaitu, model ini dapat digunakan untuk memprediksi penyakit DMG secara dini, tetapi untuk lebih jelas pengguna diharapkan untuk melakukan pemeriksaan lebih lanjut di rumah sakit.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Jullien Nazreen sebagai pemilik dari dataset Diabetes Health Indicators Dataset, dengan judul dataset “diabetes_012_health_indicators_BRFSS2021”, yang telah menyediakan dataset yang sangat membantu dalam penelitian ini.

DAFTAR PUSTAKA

- [1] F. K. Adli, “Diabetes Melitus Gestasional : Diagnosis dan Faktor Risiko,” *Jurnal Medika Hutama*, vol. 03, no. 01, pp. 1545–1551, 2021.
- [2] D. Hardianto, “Telaah Komprehensif Diabetes Melitus: Klasifikasi, Gejala, Diagnosis, Pencegahan, Dan Pengobatan,” *Jurnal Bioteknologi & Biosains Indonesia (JBBi)*, vol. 7, no. 2, pp. 304–317, 2021, doi: 10.29122/jbbi.v7i2.4209.
- [3] I. D. Federation, “International Diabetes Federation,” *International Diabetes Federation*, 2024.
- [4] H. S. W. Hovi, A. Id Hadiana, and F. Rakhmat Umbara, “Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM),” *Informatics and Digital Expert (INDEX)*, vol. 4, no. 1, pp. 40–45, 2022, doi: 10.36423/index.v4i1.895.
- [5] W. H. Organization, “<https://iris.who.int/bitstream/handle/10665/376869/97892400947>”

- [03-eng.pdf](#),” *World Health Organization*, no. Diabetes, p. 68, 2024.
- [6] Rianti Nurpalah, Meti Kusmiati, Meri Meri, Hendro Kasmento, and Dina Ferdiani, “Deteksi Dini Diabetes Melitus Gestasional (Dmg) Melalui Pemeriksaan Glukosa Darah Sebagai Upaya Pencegahan Komplikasi Pada Ibu Hamil,” *J-ABDI: Jurnal Pengabdian kepada Masyarakat*, vol. 2, no. 9, pp. 6425–6432, 2023, doi: 10.53625/jabdi.v2i9.4880.
- [7] American Pregnancy Association, “[https://americanpregnancy.org/American Pregnancy Association](https://americanpregnancy.org/American-Pregnancy-Association), 2024.
- [8] E. N. Simanjuntak, D. Irmayani, and F. A. Nasution, “Tinjauan Penerapan Kecerdasan Buatan Dalam Keamanan Jaringan Tantangan Dan Prospek Masa Depan,” *Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI)*, vol. 7, no. 2, pp. 370–375, 2024.
- [9] R. G. Wardhana, G. Wang, and F. Sibuea, “Penerapan Machine Learning Dalam Prediksi Tingkat Kasus Penyakit Di Indonesia,” *Journal of Information System Management (JOISM)*, vol. 5, no. 1, pp. 40–45, 2023, doi: 10.24076/joism.2023v5i1.1136.
- [10] A. M. Ridwan and G. D. Setyawan, “Perbandingan Berbagai Model Machine Learning Untuk Mendeteksi Diabetes,” *Teknokom*, vol. 6, no. 2, pp. 127–132, 2023, doi: 10.31943/teknokom.v6i2.152.
- [11] P. D. Rinanda, B. Delvika, S. Nurhidayarnis, N. Abror, and A. Hidayat, “Perbandingan Klasifikasi Antara Naive Bayes dan K-Nearest Neighbor Terhadap Resiko Diabetes pada Ibu Hamil,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 2, no. 2, pp. 68–75, 2022, doi: 10.57152/malcom.v2i2.432.
- [12] D. Diana Dewi, N. Qisthi, S. S. S. Lestari, and Z. H. S. Putri, “Perbandingan Metode Neural Network Dan Support Vector Machine Dalam Klasifikasi Diagnosa Penyakit Diabetes,” *Cerdika: Jurnal Ilmiah Indonesia*, vol. 3, no. 09, pp. 828–839, 2023, doi: 10.59141/cerdika.v3i09.662.
- [13] C. Aldama and M. Nasir, “Klasifikasi Penyakit Diabetes Menggunakan Metode Support Vector Machine Pada Rumah Sakit Umum Prabumulih,” *Jurnal Ilmiah Betrik*, vol. 14, no. 02, pp. 376–383, 2023, [Online]. Available: <https://ejournal.pppmitpa.or.id/index.php/betrik/article/view/117>
- [14] G. Sanhaji, A. Febrianti, and F. Teknik, “Aplikasi DIATECT Untuk Prediksi Penyakit Diabetes Menggunakan SVM Berbasis Web,” *Jurnal TEKNO KOMPAK*, vol. 18, no. 1, pp. 150–163, 2024.
- [15] R. Mohammed, J. Rawashdeh, and M. Abdullah, “Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results,” *2020 11th International Conference on Information and Communication Systems, ICICS 2020*, pp. 243–248, 2020, doi: 10.1109/ICICS49469.2020.239556.
- [16] D. H. Jeong, S. E. Kim, W. H. Choi, and S. H. Ahn, “A Comparative Study on the Influence of Undersampling and Oversampling Techniques for the Classification of Physical Activities Using an Imbalanced Accelerometer Dataset,” *Healthcare (Switzerland)*, vol. 10, no. 7, 2022, doi: 10.3390/healthcare10071255.
- [17] V. Yolanda, I. Cholissodin, and P. P. Adikara, “Klasifikasi Diagnosis Penyakit Diabetes Gestasional pada Ibu Hamil menggunakan Algoritme Neighbor Weighted K-Nearest Neighbor (NWKNN),” *jurnal pengembangan teknologi informasi dan Ilmu Komputer*, vol. 5, no. 4, pp. 1310–1321, 2021.
- [18] A. Simanjuntak and M. S. Hasibuan, “Application of PCA and K-Means Clustering Methods to Identify Diabetes Mellitus Patient Groups Based on Risk Factors,” *Prisma Sains : Jurnal Pengkajian Ilmu dan Pembelajaran Matematika dan IPA IKIP Mataram*, vol. 11, no. 4, p. 1002, 2023, doi: 10.33394/j-ps.v11i4.9263.
- [19] E. Saputro and D. Rosiyadi, “Penerapan Metode Random Over-Sampling Pada Algoritma Klasifikasi Penentuan Penyakit Diabetes,” *Bianglala Informatika*, vol. 10, no. 1, pp. 42–47, 2022, doi: 10.31294/bi.v10i1.11739.
- [20] H. S. W. Hovi, A. Id Hadiana, and F. Rakhmat Umbara, “Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM),” *Informatics and Digital Expert (INDEX)*, vol. 4, no. 1, pp. 40–45, 2022, doi: 10.36423/index.v4i1.895.
- [21] J. Nazreen, “Diabetes Health Indicators Dataset,” Kaggle. [Online]. Available: <https://www.kaggle.com/datasets/julnazz/diabetes-health-indicators-dataset>
- [22] A. M. Priyatno and T. Widiyaningtyas, “a Systematic Literature Review: Recursive Feature Elimination Algorithms,” *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, vol. 9, no. 2, pp. 196–207, 2024, doi: 10.33480/jitk.v9i2.5015.
- [23] X. W. Liang, A. P. Jiang, T. Li, Y. Y. Xue, and G. T. Wang, “LR-SMOTE — An improved unbalanced data set oversampling based on K-means and SVM,” *Knowl Based Syst*, vol. 196, 2020, doi: 10.1016/j.knosys.2020.105845.
- [24] C. C. Olisah, L. Smith, and M. Smith, “Computer Methods and Programs in Biomedicine Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective,” *Comput. Methods Programs Biomed.*, vol. 220, p. 106773, 2022, doi: 10.1016/j.cmpb.2022.106773.
- [25] B. R. Prasetyo *et al.*, “MODEL DIABETES,” *JITET J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, 2024.