BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Untuk menunjang penelitian ini penulis menggunakan banyak sumber dari beberapa jurnal yang menginspirasi penulis untuk melakukan penelitian, dan tentu juga memiliki latar belakang yang sama pada penelitian ini yaitu memprediksi penyakit menggunakan metode yang ada pada machine learning.

Berdasarkan penelitian yang dilakukan dalam karya yang berjudul "Perbandingan Akurasi Algoritma Random Forest dan Algoritma Support Vector Machine pada Penyakit Diabetes", tujuan utama dari penelitian tersebut adalah mengevaluasi performa algoritma Support Vector Machine dan Random Forest dalam mengklasifikasikan Diabetes Mellitus. Penelitian ini menemukan bahwa Random Forest memberikan hasil prediksi dengan akurasi tertinggi dibandingkan Support Vector Machine. Dengan pembagian data training dan testing sebesar 75:25, algoritma Random Forest menghasilkan akurasi terbaik. Selain itu, dengan penerapan metode K-Fold Cross Validation menggunakan 10 fold, diperoleh rata-rata akurasi sebesar 80.72%. Evaluasi menggunakan confusion matrix menunjukkan nilai precision sebesar 86.92%, recall 91.86%, serta F1-score sebesar 89.31%, sementara AUC Score untuk Random Forest mencapai 85.40%[5].

Sementara itu, penelitian yang mengevaluasi perbandingan akurasi algoritma Neural Network, Naïve Bayes, dan Logistic Regression dalam mengklasifikasikan Diabetes Mellitus. Hasil penelitian ini menunjukkan bahwa algoritma Logistic Regression memiliki performa terbaik dengan akurasi 75.78% dan AUC Score sebesar 80.1%. Sebagai perbandingan, algoritma Naïve Bayes mencatatkan akurasi sebesar 74.87% dan AUC Score 79.9%, sedangkan Neural Network menunjukkan performa yang lebih rendah dengan akurasi 69.27% dan AUC Score 73.6%[6].

Pada penelitian yang dilakukan menggunakan algoritma Logistic Regression untuk memprediksi Diabetes Mellitus. Model ini dievaluasi menggunakan confusion matrix dan menghasilkan akurasi 77%, precision 75%, recall 77%, serta F1-score 76%. Penelitian ini juga mengimplementasikan hyperparameter tuning untuk meningkatkan kinerja algoritma tersebut, sehingga menghasilkan peningkatan akurasi menjadi 82%, precision 81%, recall 79%, dan F1-score 80%[7].

Terakhir, penelitian yang berjudul "Peningkatan Kinerja Akurasi Prediksi Penyakit Diabetes Mellitus Menggunakan Metode Grid Search pada Algoritma Logistic Regression" menemukan bahwa penerapan Grid Search pada Logistic Regression menghasilkan rata-rata akurasi sekitar 79% berdasarkan precision, recall, dan F1-score. Akurasi keseluruhan model mencapai 80%, dengan AUC Score sebesar 75.1%. Dari hasil pengujian data, model Logistic Regression dengan Grid Search berhasil memprediksi 15 data benar dan 3 data salah dari total 18 data yang diuji, dengan tingkat akurasi mencapai 83.33%[8].

Untuk memberikan penjelasan lebih terstruktur dan jelas terkait berbagai penelitian di atas, disusunlah tabel 2.1 sebagai rangkuman.

Tabel 2. 1 Jurnal Penelitian Terdahulu

No	Judul, Penulis, dan Tahun	Algoritma yang Digunakan	Hasil Utama Penelitian
1	"Comparison of Neural Network Algorithms, Naive Bayes and Logistic Regression to predict diabetes" (Utami, D. Y., et al., 2022)	Neural Network, Naive Bayes, Logistic Regression	Logistic Regression memiliki akurasi terbaik dibandingkan Neural Network dan Naive Bayes dalam prediksi Diabetes Mellitus.

2	"Relationship of Respondent's Characteristic with the Risk of Diabetes Mellitus and Dislipidemia" (Widyasari, N., 2022)	Analisis Statistik Deskriptif	Faktor-faktor karakteristik responden seperti usia dan gaya hidup berkontribusi signifikan terhadap risiko Diabetes Mellitus.
3	"Faktor-faktor yang mempengaruhi terjadinya diabetes mellitus" (Imelda, S. I., 2023)	Analisis Faktor	Faktor jenis kelamin, usia, dan gaya hidup sangat memengaruhi risiko Diabetes Mellitus, terutama pada pasien perempuan.
4	"Analisis Perbandingan Metode Support Vector Machine (SVM) dan Decision Tree" (Fahmi, F. A., 2022)	Support Vector Machine (SVM), Decision Tree	Decision Tree menghasilkan akurasi lebih tinggi dibandingkan SVM pada kasus klasifikasi penyakit tertentu.
5	"Percentage of older people who needed help with personal care" (Elflein, J., 2024)	Analisis Data Statistik	Melaporkan data statistik prevalensi kebutuhan perawatan pribadi bagi penderita lanjut usia dengan penyakit kronis.
6	"Klasifikasi Penyakit Diabetes Mellitus Menggunakan Algoritma Extreme Gradient Boosting (XGBoost)" (Pratama, A. R., & Sitompul, B., 2024)	XGBoost	XGBoost menunjukkan performa klasifikasi yang sangat baik dengan tingkat akurasi tinggi pada dataset pasien diabetes.

7	"Deteksi Dini Penyakit Diabetes Menggunakan Machine Learning" (Marlim, Y. N., et al., 2022)	Machine Learning (General)	Algoritma Machine Learning mampu mendeteksi Diabetes Mellitus dengan tingkat akurasi yang signifikan pada data pasien.
8	"Optimasi Klasifikasi Diabetes Mellitus Berbasis K-Nearest Neighbors Menggunakan Particle Swarm Optimization" (Ramadhan, H., & Wijaya, A., 2023)	K-Nearest Neighbors, PSO	Penerapan Particle Swarm Optimization (PSO) berhasil meningkatkan akurasi model KNN secara signifikan dalam klasifikasi diabetes.
9	"Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Diabetes" (Suryanegara, G. A. B., & Purbolaksono, M. D., 2022)	Random Forest	Normalisasi data sebelum penerapan Random Forest mampu meningkatkan akurasi prediksi pasien Diabetes Mellitus secara efektif.
10	"Komparasi Deep Learning dan Support Vector Machine untuk Deteksi Retinopati Diabetik" (Santoso, T., & Lestari, S., 2025)	Deep Learning (CNN), SVM	Metode Deep Learning (CNN) memberikan hasil akurasi dan sensitivitas yang lebih superior dibandingkan SVM konvensional.

Berbagai penelitian sebelumnya telah menunjukkan bahwa algoritma machine learning mampu memberikan performa yang baik dalam memprediksi Diabetes Mellitus. Namun demikian, hasil penelitian tersebut masih menunjukkan variasi performa karena dipengaruhi oleh karakteristik dataset, teknik prapemrosesan data, pemilihan fitur, serta metode optimasi parameter yang digunakan. Selain itu, sebagian besar penelitian hanya berfokus pada pengukuran akurasi tanpa memberikan perhatian yang memadai terhadap nilai *recall* atau sensitivitas, padahal

dalam konteks diagnosis penyakit kemampuan model mendeteksi pasien yang benar-benar menderita diabetes merupakan aspek yang jauh lebih penting dibandingkan sekadar memperoleh nilai akurasi yang tinggi [4].

Penelitian terdahulu juga umumnya menggunakan dataset dengan jumlah observasi yang relatif terbatas sehingga kemampuan generalisasi model masih perlu diuji pada dataset yang lebih besar dan lebih beragam. Di sisi lain, penelitian yang secara khusus membandingkan algoritma Random Forest dan Support Vector Machine menggunakan dataset Behavioral Risk Factor Surveillance System (BRFSS) dengan proses *undersampling*, standarisasi data, serta optimasi *hyperparameter* menggunakan *Grid Search* masih relatif terbatas. Kondisi tersebut menunjukkan adanya peluang penelitian untuk memperoleh model klasifikasi yang lebih stabil dan memiliki kemampuan deteksi yang lebih baik terhadap kasus Diabetes Mellitus [5, 6).

Berdasarkan kondisi tersebut, penelitian ini menawarkan kebaruan melalui penerapan pipeline pemodelan yang terintegrasi, mulai dari proses *data preprocessing*, penyeimbangan kelas menggunakan metode *undersampling*, optimasi parameter menggunakan *Grid Search*, hingga evaluasi komprehensif menggunakan *Accuracy*, *Precision*, *Recall*, *F1-Score*, *Receiver Operating Characteristic* (ROC), dan *Area Under Curve* (AUC). Pendekatan tersebut diharapkan mampu menghasilkan model prediksi yang tidak hanya memiliki tingkat akurasi yang tinggi, tetapi juga sensitivitas yang baik dalam mendeteksi individu yang berisiko mengalami Diabetes Mellitus.

Berdasarkan analisis dari jurnal di tabel 2.1 membahas prediksi Diabetes Mellitus menggunakan algoritma machine learning, dapat disimpulkan bahwa Random Forest dan Support Vector Machine merupakan dua algoritma yang paling efektif dan sering digunakan dalam penelitian ini. Random Forest umumnya menunjukkan akurasi yang lebih tinggi dan hasil yang lebih konsisten dibandingkan algoritma lain seperti Naive Bayes, KNN, dan Logistic Regression, terutama pada dataset dengan kompleksitas yang lebih tinggi. Meskipun SVM juga menunjukkan performa yang baik, Random Forest lebih unggul dalam banyak kasus. Beberapa penelitian juga menunjukkan pentingnya optimasi parameter seperti grid search dan normalisasi data untuk meningkatkan akurasi model. Secara keseluruhan, tren penggunaan algoritma ensemble dan optimasi menjadi fokus utama dalam meningkatkan performa prediksi Diabetes Mellitus, dengan potensi untuk

penelitian lebih lanjut yang dapat melibatkan Deep Learning atau kombinasi algoritma untuk hasil yang lebih baik.

2.2 Teori yang berkaitan

Diabetes Mellitus merupakan sebuah penyakit yang memiliki tingkat risiko kematian yang tinggi. Penyakit ini terjadi ketika tubuh tidak mampu memproduksi insulin dalam jumlah yang cukup atau ketika insulin yang dihasilkan tidak dapat bekerja dengan baik. Secara umum, Diabetes Mellitus terjadi ketika kadar gula dalam darah melampaui batas normal yang dianjurkan[9]. Walaupun tidak dapat disembuhkan, kualitas hidup pasien tetap dapat dipertahankan secara optimal melalui pengelolaan metabolisme yang efektif. Beberapa langkah untuk menjaga keseimbangan metabolisme meliputi pemberian insulin secara berkala melalui injeksi, menerapkan pola makan sehat, melakukan olahraga secara teratur, dan menjaga kondisi tubuh secara keseluruhan. Selain itu, edukasi yang diberikan kepada pasien dan keluarganya memegang peran penting dalam mendukung pengelolaan penyakit ini[10].

Diabetes Mellitus tergolong sebagai salah satu jenis penyakit tidak menular yang penyebarannya cukup luas di berbagai belahan dunia. Penyakit ini bahkan menduduki posisi keempat sebagai salah satu penyebab kematian utama di sejumlah negara berkembang. Diabetes Mellitus memiliki sifat yang heterogen, ditandai dengan beberapa ciri utama seperti tingginya kadar gula dalam darah, gangguan pada toleransi glukosa, serta ketidakcukupan atau ketidakberfungsian insulin[11]. Penyakit ini juga termasuk gangguan metabolisme kronis yang mengganggu regulasi glukosa darah dalam tubuh, sehingga menyebabkan kadar gula darah sulit dikontrol[10].

Untuk menjaga agar kadar glukosa darah tetap mendekati batas normal, pasien perlu melakukan manajemen diri secara aktif. Manajemen ini mencakup pemantauan kadar glukosa darah secara rutin dan pengambilan tindakan yang tepat, seperti menyesuaikan pola makan serta penggunaan obat-obatan, termasuk insulin. Ketidaknormalan pada kadar glukosa darah, yang dikenal sebagai anomali glukosa

darah, dapat dijelaskan melalui pemahaman lebih lanjut mengenai penyebabnya. Anomali ini dapat berupa variasi penyebab normal yang diketahui secara jelas atau variasi penyebab khusus yang belum teridentifikasi sepenuhnya pada pasien[12].

2.2.1 Diabetes Mellitus Tipe 1

Diabetes Mellitus tipe 1, yang juga dikenal sebagai insulin-dependent diabetes, merupakan kondisi autoimun yang umumnya menyerang individu berusia muda, terutama di bawah 20 tahun. Penyakit ini sering disebut sebagai juvenile-onset diabetes. Pada tipe ini, sel pankreas yang berfungsi memproduksi insulin dirusak oleh sistem kekebalan tubuh. Untuk mengelola kondisi ini, penderita memerlukan suntikan insulin secara teratur, pemantauan kadar gula darah, dan pola makan yang sesuai[11].

2.2.2 Diabetes Mellitus Tipe 2

Diabetes Mellitus tipe 2 menyumbang sekitar 90% dari keseluruhan kasus diabetes dan sering disebut sebagai adult-onset diabetes atau non-insulin-dependent diabetes[13]. Pada tipe ini, tubuh menjadi resisten terhadap insulin, yang mengakibatkan peningkatan kebutuhan insulin. Namun, pankreas tidak mampu memproduksi insulin dalam jumlah yang cukup untuk memenuhi kebutuhan tersebut. Pencegahan Diabetes Mellitus tipe 2 dapat dilakukan dengan pola makan sehat, olahraga rutin, dan pemantauan kadar glukosa darah secara berkala. Pada umumnya, penderita tipe ini memiliki kadar gula darah yang tinggi, meskipun tidak setinggi pada penderita Diabetes Mellitus tipe 1[12].

2.2.3 Diabetes Mellitus Gestasional (DMG

Diabetes Mellitus Gestasional adalah kondisi yang terjadi pada wanita hamil, ditandai dengan peningkatan kadar gula darah akibat penurunan sensitivitas terhadap insulin. Diagnosis dilakukan melalui pengukuran kadar gula darah dengan metode enzimatik menggunakan sampel darah

plasma vena. Diabetes Mellitus Gestasional dapat didiagnosis jika glukosa puasa melebihi 126 mg/dL, glukosa plasma mencapai lebih dari 200 mg/dL dua jam setelah Tes Toleransi Glukosa Oral (TTGO), atau glukosa sewaktu lebih dari 200 mg/dL. Kondisi ini sering muncul pada trimester kedua kehamilan, sekitar usia kehamilan enam bulan. Sebagian besar wanita pulih setelah melahirkan, tetapi hampir separuh kasus berisiko kambuh di masa mendatang[14].

2.3 Framework/Algoritma yang digunakan

Pemilihan algoritma merupakan salah satu tahap penting dalam penelitian machine learning karena setiap algoritma memiliki kelebihan dan keterbatasan. Pada penelitian ini dipilih Random Forest dan Support Vector Machine karena keduanya merupakan algoritma klasifikasi yang banyak digunakan pada penelitian bidang kesehatan dan memiliki kemampuan yang baik dalam menangani data berdimensi tinggi maupun hubungan antarvariabel yang kompleks. Selain itu, kedua algoritma memiliki pendekatan yang berbeda sehingga hasil perbandingan dapat memberikan gambaran mengenai algoritma yang paling sesuai untuk prediksi Diabetes Mellitus.

2.3.1 Random Forest

Random Forest adalah algoritma pembelajaran mesin yang termasuk dalam metode ensemble, yang menggabungkan beberapa pohon keputusan (decision trees) untuk meningkatkan akurasi prediksi. Algoritma ini pertama kali diperkenalkan oleh Leo Breiman pada tahun 2001. Random Forest bekerja dengan cara membuat banyak Random Forest secara independen menggunakan subset data yang diambil secara acak (dengan pengembalian) dari dataset asli. Pada setiap Random Forest, hanya subset fitur yang dipilih secara acak yang digunakan untuk menentukan split terbaik. Hasil akhir diperoleh dengan menggabungkan prediksi dari semua Random Forest, baik melalui rata-rata (regresi) atau voting mayoritas (klasifikasi) [15].

Random Forest merupakan salah satu algoritma *ensemble learning* yang bekerja dengan menggabungkan banyak *decision tree* untuk menghasilkan prediksi yang lebih stabil dibandingkan penggunaan satu pohon keputusan saja. Pendekatan ini memanfaatkan teknik *bootstrap aggregating* (bagging), yaitu membangun setiap pohon menggunakan sampel data yang berbeda serta memilih sebagian fitur secara acak pada setiap proses pemisahan (*split*). Strategi tersebut mampu mengurangi varians model dan meningkatkan kemampuan generalisasi terhadap data baru, sehingga risiko *overfitting* menjadi lebih kecil dibandingkan algoritma *Decision Tree* tunggal [6, 13].

Dalam bidang kesehatan, Random Forest banyak digunakan karena mampu menangani data dengan jumlah variabel yang cukup banyak, memiliki hubungan yang kompleks, serta mengakomodasi interaksi nonlinier antarvariabel klinis. Selain itu, algoritma ini relatif tahan terhadap *noise*, *outlier*, maupun data yang memiliki korelasi antarfitur. Keunggulan tersebut menjadikan Random Forest sesuai untuk memprediksi Diabetes Mellitus yang dipengaruhi oleh berbagai faktor risiko seperti usia, indeks massa tubuh (*Body Mass Index*), kadar glukosa, hipertensi, aktivitas fisik, dan riwayat penyakit penyerta [10].

Penelitian-penelitian terdahulu juga menunjukkan bahwa Random Forest secara konsisten memberikan performa klasifikasi yang kompetitif pada berbagai dataset diabetes. Namun demikian, kinerja algoritma ini tetap dipengaruhi oleh kualitas data, keseimbangan kelas, serta pengaturan *hyperparameter*. Oleh karena itu, penelitian ini menerapkan proses *data preprocessing*, penyeimbangan kelas menggunakan *undersampling*, dan optimasi parameter melalui *Grid Search* agar model yang dihasilkan memiliki kemampuan prediksi yang optimal.

Proses Kerja:

1. **Bootstrap Sampling**

   Data pelatihan diambil secara acak dengan pengembalian untuk membuat beberapa subset data (bootstrap).

2. **Pembentukan Random Forest**

   Pada setiap Random Forest, pembentukan keputusan dilakukan dengan memilih atribut secara acak dan membagi node berdasarkan atribut yang memberikan *split* terbaik (biasanya diukur dengan *Gini Impurity* atau *Entropy*).

3. **Voting atau Averaging**

   - Untuk masalah klasifikasi: hasil akhir ditentukan oleh suara mayoritas (*majority voting*) dari semua pohon.
   - Untuk masalah regresi: hasil akhir adalah rata-rata dari prediksi semua pohon.

Rumus-Rumus:

- **Gini Impurity**

  Digunakan untuk mengukur kemurnian node:

  $$G = 1 - i = 1\sum C \ pi2$$

  Dimana:

  G = Gini Impurity

  C = Jumlah kelas

  Pi = Proporsi kelas i dalam node

- **Entropy (Informasi Gain)**

  Untuk mengukur ketidakteraturan:

  $$H = - i = 1\sum C \ pi \ log2 \ (pi \ )$$

  Dimana:

  H = Entropi

  C = Jumlah kelas

  Pi = Proporsi kelas i dalam node

Berdasarkan karakteristik tersebut, Random Forest dipilih sebagai salah satu algoritma yang dibandingkan dalam penelitian ini karena mampu menghasilkan model yang stabil pada data kesehatan dengan jumlah observasi yang besar. Kemampuan algoritma dalam mengurangi *overfitting*, menangani hubungan nonlinier, serta menyediakan informasi mengenai pentingnya setiap fitur (*feature importance*) diharapkan dapat meningkatkan kualitas prediksi Diabetes Mellitus pada dataset BRFSS.

2.3.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. Algoritma ini bekerja dengan menemukan hyperplane optimal yang memisahkan data ke dalam kelas-kelas dengan margin terbesar. Jika data tidak dapat dipisahkan secara linier, SVM menggunakan fungsi kernel untuk memetakan data ke ruang dimensi yang lebih tinggi agar memungkinkan pemisahan linier. Formulasi SVM didasarkan pada masalah optimasi kuadratik yang memaksimalkan margin sambil meminimalkan kesalahan klasifikasi[16].

Support Vector Machine (SVM) merupakan algoritma klasifikasi berbasis teori *Statistical Learning* yang dikembangkan untuk memperoleh batas pemisah (*hyperplane*) dengan margin maksimum antar kelas. Prinsip dasar tersebut memungkinkan SVM menghasilkan kemampuan generalisasi yang baik ketika dihadapkan pada data baru (*unseen data*). Berbeda dengan algoritma klasifikasi konvensional yang hanya berupaya meminimalkan kesalahan pelatihan, SVM berusaha memperoleh keseimbangan antara kompleksitas model dan kemampuan prediksi sehingga lebih tahan terhadap *overfitting* [7, 13]

Dalam penelitian di bidang kesehatan, SVM banyak digunakan karena mampu mengidentifikasi hubungan yang kompleks antarvariabel klinis, terutama

ketika hubungan tersebut tidak bersifat linier. Faktor-faktor risiko Diabetes Mellitus seperti usia, kadar glukosa darah, hipertensi, indeks massa tubuh (BMI), aktivitas fisik, serta riwayat penyakit memiliki pola interaksi yang saling memengaruhi sehingga sulit dipisahkan menggunakan batas keputusan linier. Oleh karena itu, SVM menjadi salah satu algoritma yang banyak diterapkan dalam penelitian prediksi penyakit kronis karena memiliki kemampuan klasifikasi yang tinggi pada data medis multidimensi [14, 15]

Alasan Matematis dan Medis Penggunaan Kernel RBF pada SVM

Dalam arsitektur *Support Vector Machine* (SVM), penentuan fungsi *kernel* memegang peran yang sangat krusial terhadap keberhasilan model, terutama ketika dihadapkan pada dataset berdimensi tinggi. Pada penelitian ini, implementasi model SVM secara spesifik dioptimasi menggunakan fungsi *kernel Radial Basis Function* (RBF), dan bukan menggunakan *kernel linear* biasa. Keputusan ini didasari oleh dua argumentasi utama, yakni secara komputasi matematis dan secara representasi kondisi medis.

Secara medis, faktor-faktor indikator kesehatan pada manusia—seperti hubungan antara tingkat Indeks Massa Tubuh (BMI), fluktuasi tekanan darah, rutinitas aktivitas fisik, dan pertambahan usia kronologis—memiliki interaksi yang sangat kompleks dan saling tumpang tindih (*overlapping*). Risiko seseorang terkena diabetes tidak bergerak dalam pola garis lurus yang sederhana (linear). Oleh karena itu, penggunaan *kernel linear* yang memaksakan pembentukan batas keputusan (*decision boundary*) berupa garis atau bidang datar tunggal, dinilai terlalu kaku dan tidak relevan untuk merepresentasikan kompleksitas diagnosis klinis, sehingga sering kali menyebabkan model mengalami *underfitting*.

Secara matematis, *kernel RBF* mengatasi keterbatasan tersebut dengan memproyeksikan vektor data input pasien secara implisit ke dalam ruang fitur berdimensi jauh lebih tinggi (bahkan tak terhingga). Proses transformasi ini memungkinkan algoritma SVM untuk melengkungkan batas *hyperplane* secara

dinamis guna memisahkan kelompok data pasien sehat dan pasien diabetes yang secara kasat mata sulit dipisahkan pada ruang dimensi aslinya. Penggunaan *kernel RBF* ini menjadi langkah validasi yang wajib diaplikasikan agar komparasi performa antara algoritma tunggal SVM melawan algoritma *ensemble* seperti *Random Forest* menjadi setara dan adil secara akademis.

Berdasarkan karakteristik masing-masing algoritma, Random Forest dan Support Vector Machine dipilih sebagai objek komparasi dalam penelitian ini. Random Forest mewakili pendekatan ensemble berbasis pohon keputusan, sedangkan Support Vector Machine mewakili pendekatan berbasis optimasi hyperplane. Perbandingan kedua algoritma diharapkan dapat memberikan informasi mengenai model yang paling efektif dalam mendeteksi Diabetes Mellitus pada dataset BRFSS.

Proses Kerja:

1. **Fungsi Kernel**

SVM memetakan data ke dimensi yang lebih tinggi menggunakan fungsi kernel untuk membuat data dapat dipisahkan secara linier. Jenis kernel yang umum digunakan:

- Linear
- Polynomial
- Radial Basis Function (RBF)

2. **Penentuan Hyperplane**

Hyperplane adalah garis atau bidang di ruang fitur yang memisahkan dua kelas. Hyperplane terbaik adalah yang memiliki margin maksimum antara titik data dari kelas yang berbeda.

3. **Optimisasi Margin**

SVM memaksimalkan margin, yaitu jarak terdekat antara *hyperplane* dan titik data mana pun dari setiap kelas [17].

Rumus-Rumus:

- **Persamaan Hyperplane**:

$$w \cdot x + b = 0$$

Di mana:
w = Vektor bobot
x = Vektor fitur
b = Bias

- **Fungsi Optimasi** (untuk margin maksimum):

$$min21 \quad \| w \| 2s.t.yi \ (w \cdot xi \quad + b) \geq 1$$

Di mana:
||w|| = Norm dari vektor w
yi = Label kelas untuk titik data i
xi = Vektor fitur untuk titik data i

- **Fungsi Kernel (RBF)**:

$$K(x,x\prime) = exp(-\gamma \ \| x - x\prime \ \| 2)$$

Di mana:
K(x,x′) = Fungsi kernel RBF antara dua titik x dan x′
γ = Parameter kernel

Berdasarkan karakteristik tersebut, penelitian ini menggunakan kernel Radial Basis Function (RBF) sebagai fungsi kernel pada algoritma Support Vector Machine. Pemilihan kernel ini didasarkan pada kemampuan RBF dalam membentuk batas keputusan yang adaptif terhadap distribusi data kesehatan yang kompleks. Selain itu, penggunaan kernel RBF juga bertujuan agar proses komparasi dengan algoritma Random Forest dilakukan pada konfigurasi model yang telah banyak direkomendasikan dalam penelitian-penelitian terdahulu mengenai prediksi Diabetes Mellitus.

2.4 Tools/software yang digunakan

2.4.1 Python

Python adalah bahasa pemrograman tingkat tinggi yang populer dalam komunitas pembelajaran mesin dan data analitik. Dengan sintaks yang sederhana dan pustaka yang kaya, Python menjadi pilihan utama untuk pengembangan model pembelajaran mesin, termasuk implementasi algoritma Random Forest dan Support Vector Machine [18].

Python dipilih sebagai bahasa pemrograman utama dalam penelitian ini karena memiliki ekosistem pustaka (*library*) yang lengkap untuk pengolahan data, pengembangan model *machine learning*, visualisasi data, serta evaluasi performa model. Selain bersifat *open source*, Python juga mendukung proses analisis yang bersifat *reproducible*, sehingga setiap tahapan penelitian dapat didokumentasikan dan dijalankan kembali dengan hasil yang konsisten. Kemudahan integrasi antar pustaka menjadikan Python sebagai salah satu bahasa pemrograman yang paling banyak digunakan dalam penelitian kecerdasan buatan dan analisis data kesehatan [32, 33].

Beberapa pustaka Python yang sering digunakan dalam penelitian ini meliputi:

1. **Scikit-learn**
   Scikit-learn adalah pustaka pembelajaran mesin yang dirancang untuk kemudahan penggunaan dan efisiensi. Pustaka ini menyediakan implementasi dari algoritma Random Forest dan Support Vector Machine, dilengkapi dengan fungsi evaluasi model seperti *confusion matrix*, ROC curve, dan penghitungan akurasi [19].
   - **Fitur utama**:
     - Dukungan untuk regresi, klasifikasi, dan klasterisasi.

- Parameter tuning dengan GridSearchCV untuk mencari parameter terbaik.

Dalam penelitian ini, Scikit-learn digunakan sebagai pustaka utama untuk membangun model Random Forest dan Support Vector Machine. Selain implementasi algoritma klasifikasi, Scikit-learn juga dimanfaatkan untuk melakukan pembagian data (*train-test split*), standarisasi data menggunakan *StandardScaler*, optimasi *hyperparameter* melalui *GridSearchCV*, serta evaluasi model menggunakan *confusion matrix*, *classification report*, kurva ROC, dan nilai AUC. Dengan demikian, seluruh proses pemodelan dilakukan dalam satu lingkungan kerja yang terintegrasi sehingga memudahkan reproduksibilitas penelitian.

2. **Pandas**

   Pandas digunakan untuk manipulasi dan analisis data. Pustaka ini mempermudah proses membaca, membersihkan, dan menyiapkan dataset untuk pelatihan dan pengujian model [19].

   - **Fitur utama**:
     - Struktur data seperti DataFrame untuk pengolahan data tabular.
     - Operasi statistik dasar dan eksplorasi data.

   Pada tahap *Data Understanding* dan *Data Preparation*, Pandas digunakan untuk membaca dataset, mengidentifikasi struktur data, mendeteksi *missing values*, melakukan seleksi variabel penelitian, mengubah tipe data, serta melakukan manipulasi data sebelum proses pemodelan. Kemampuan Pandas dalam mengelola data tabular secara efisien sangat membantu proses eksplorasi data dan pembersihan data yang merupakan bagian penting dalam metodologi CRISP-DM [19].

3. **Matplotlib dan Seaborn**

   Matplotlib dan Seaborn digunakan untuk visualisasi data. Alat ini

membantu dalam memahami pola data awal, memvisualisasikan hasil prediksi, dan membuat grafik evaluasi model [20].

- **Fitur utama**:
  - Pembuatan grafik seperti histogram, scatter plot, dan heatmap.
  - Visualisasi yang disesuaikan dengan parameter model.

Selain digunakan untuk eksplorasi data, Matplotlib dan Seaborn juga dimanfaatkan untuk menyajikan hasil analisis dalam bentuk visualisasi yang informatif. Grafik distribusi kelas, *heatmap confusion matrix*, kurva ROC, serta visualisasi penting lainnya digunakan untuk mempermudah interpretasi hasil penelitian. Penyajian visual tersebut membantu menjelaskan performa masing-masing algoritma sehingga hasil penelitian lebih mudah dipahami [20].

2.4.2 Jupyter Notebook

Jupyter Notebook adalah aplikasi berbasis web yang digunakan untuk menulis dan mengeksekusi kode Python [19]. Dengan fitur seperti penyisipan teks deskriptif dan visualisasi langsung, Jupyter mendukung pengembangan model pembelajaran mesin yang iteratif dan mudah diakses.

- **Kelebihan**:
  - Mendukung dokumentasi inline dengan Markdown.
  - Interaktif dan mendukung eksperimen secara langsung.

Seluruh tahapan analisis dalam penelitian ini dikembangkan menggunakan Jupyter Notebook karena mendukung integrasi antara kode program, dokumentasi, visualisasi, dan hasil keluaran (*output*) dalam satu dokumen interaktif. Penggunaan Jupyter Notebook juga mempermudah proses validasi penelitian karena setiap langkah analisis dapat ditampilkan secara berurutan mulai dari pembacaan data, proses *preprocessing*, pelatihan model, hingga evaluasi hasil klasifikasi.