BAB 2
LANDASAN TEORI

2.1 Kanker Payudara

Kanker payudara merupakan salah satu jenis kanker yang paling umum terjadi pada perempuan di 154 negara dan menjadi penyebab utama kematian akibat kanker di 103 negara [9]. Meskipun berbagai perkembangan dalam terapi terus dilakukan, kanker payudara masih menjadi salah satu penyebab utama kematian akibat kanker pada perempuan di seluruh dunia [15].

Dalam konteks tersebut, analisis kelangsungan hidup pasien menjadi penting untuk memahami perkembangan penyakit serta menentukan strategi penanganan yang tepat. Selain itu, informasi mengenai kelangsungan hidup dapat mendukung tenaga medis dalam mengambil keputusan terapi yang sesuai bagi setiap pasien [9, 16], mengevaluasi efektivitas pengobatan, mengidentifikasi faktor prognostik, serta mengembangkan strategi terapi yang lebih personal [17]. Hal ini dikarenakan setiap pasien memiliki karakteristik yang berbeda, sehingga respons terhadap pengobatan juga dapat bervariasi. Beberapa faktor diketahui memiliki pengaruh terhadap kelangsungan hidup pasien kanker payudara, di antaranya usia, stadium kanker, dan ras pasien [8].

2.2 Survival Prediction pada Kanker Payudara

Survival didefinisikan sebagai rentang waktu seorang pasien dapat bertahan setelah didiagnosis suatu penyakit [9]. Prediksi kelangsungan hidup (*survival prediction*) bertujuan untuk memperkirakan kemungkinan pasien bertahan dalam periode waktu tertentu berdasarkan data yang tersedia, sehingga dapat mendukung pengambilan keputusan klinis dan perencanaan pengobatan yang lebih tepat.

Dalam praktik klinis, prediksi kelangsungan hidup umumnya dibedakan menjadi jangka pendek dan jangka panjang [4]. Salah satu pendekatan yang sering digunakan adalah dengan menetapkan horizon waktu tertentu, seperti 3 tahun dan 5 tahun, untuk mengevaluasi kondisi pasien. Periode 5 tahun telah lama digunakan sebagai standar dalam berbagai penelitian klinis [16], sedangkan horizon waktu 3 tahun digunakan untuk memberikan gambaran kondisi pasien dalam jangka waktu yang lebih cepat.

Pendekatan yang menggunakan batas waktu tertentu dalam prediksi

kelangsungan hidup dikenal sebagai *fixed-horizon classification*. Pada pendekatan ini, model mengklasifikasikan pasien berdasarkan status kelangsungan hidupnya pada horizon waktu yang telah ditentukan, seperti 3 tahun atau 5 tahun [10].

Meningkatnya angka kejadian kanker yang tidak selalu diikuti dengan peningkatan angka kematian menunjukkan perlunya model prediksi yang lebih akurat, khususnya dalam mengidentifikasi perbedaan kelangsungan hidup pasien pada berbagai horizon waktu [6]. Oleh karena itu, pengembangan model prediksi yang mampu mengklasifikasikan pasien ke dalam kategori jangka pendek dan jangka panjang menjadi penting dalam penelitian kanker payudara.

2.3 DNA Methylation

DNA methylation merupakan salah satu bentuk modifikasi epigenetik yang terjadi melalui penambahan gugus metil pada basa sitosin di dinukleotida CpG, terutama pada wilayah yang dikenal sebagai *CpG islands* [13]. Proses ini dimediasi oleh enzim DNA methyltransferases (DNMTs) dan memiliki peran penting dalam berbagai proses biologis, seperti regulasi gen, perkembangan embrio, dan *genomic imprinting* [13].

Pada kondisi normal, DNA methylation membantu menjaga stabilitas genom dan mengatur berbagai proses biologis seluler. Namun, perubahan pola methylation yang abnormal dapat menyebabkan gangguan regulasi gen dan berkontribusi terhadap perkembangan berbagai penyakit, termasuk kanker payudara [18]. Pada kanker payudara, perubahan methylation diketahui berkaitan dengan proliferasi sel tumor, metastasis, serta mekanisme *immune escape* yang memengaruhi perkembangan penyakit [18]. DNA methylation pada platform Illumina umumnya direpresentasikan menggunakan *beta value*, yaitu rasio antara intensitas sinyal *methylated* dan total intensitas sinyal methylation [19]. Nilai beta berada pada rentang 0 hingga 1, di mana nilai mendekati 0 menunjukkan kondisi tidak termethylasi, sedangkan nilai mendekati 1 menunjukkan kondisi termethylasi penuh. Rumus *beta-value* ditunjukkan pada Persamaan 2.1 [19].

$$\beta = \frac{M}{M + U + \alpha} \tag{2.1}$$

dengan:

- M merupakan intensitas sinyal *methylated*,

- U merupakan intensitas sinyal *unmethylated*,
- α merupakan konstanta offset untuk mencegah pembagian dengan nol.

Selain berperan dalam perkembangan kanker, DNA methylation juga memiliki potensi sebagai biomarker dalam diagnosis dan prediksi kelangsungan hidup pasien. Pola methylation tertentu dapat digunakan untuk mengidentifikasi karakteristik molekuler kanker serta membantu memprediksi prognosis dan respons terapi pasien [20]. Oleh karena itu, DNA methylation menjadi salah satu jenis data biologis yang banyak dimanfaatkan dalam penelitian prediksi kelangsungan hidup pasien kanker payudara.

2.4 High-Dimensional Data

Data *high-dimensional* merupakan data yang memiliki jumlah fitur sangat besar dibandingkan dengan jumlah sampel [13]. Kondisi ini umumnya direpresentasikan dengan $p \gg n$, di mana p menyatakan jumlah fitur dan n menyatakan jumlah sampel. Karakteristik tersebut umum ditemukan pada data biologis, termasuk DNA methylation, gene expression, dan data multi-omics lainnya [21].

Salah satu permasalahan utama pada data berdimensi tinggi adalah *curse of dimensionality*, yaitu kondisi ketika peningkatan jumlah dimensi menyebabkan proses pembelajaran model menjadi lebih kompleks [22]. Selain itu, data berdimensi tinggi juga memiliki berbagai tantangan lain, seperti meningkatnya risiko *overfitting*, tingginya biaya komputasi, serta adanya fitur yang bersifat *noise*, redundan, atau tidak relevan [23]. Oleh karena itu, diperlukan teknik pengurangan dimensi seperti *feature selection* untuk memilih fitur-fitur yang paling relevan sehingga dapat meningkatkan efisiensi komputasi dan performa model dalam proses prediksi [24].

2.5 Data Acquisition

Data acquisition merupakan proses memperoleh atau mengumpulkan data dari berbagai sumber untuk digunakan dalam analisis dan pengembangan model machine learning. Tahap ini menjadi fondasi penting karena kualitas serta kelengkapan data yang dikumpulkan akan memengaruhi kinerja model yang dibangun [25].

Selain itu, proses akuisisi data juga mencakup pengumpulan data dalam jumlah yang memadai dari berbagai sumber sehingga dapat membentuk kumpulan data yang representatif dan siap digunakan pada tahap pemrosesan maupun pelatihan model [26].

2.6 Exploratory Data Analysis

Exploratory Data Analysis (EDA) merupakan pendekatan untuk menganalisis dan mengeksplorasi karakteristik utama suatu dataset sebelum dilakukan pemodelan. Menurut Tukey [27], EDA bertujuan untuk memaksimalkan pemahaman terhadap data, mengidentifikasi pola yang mendasari, mendeteksi anomali, menguji asumsi, serta memperoleh wawasan awal yang dapat mendukung proses analisis lebih lanjut. Oleh karena itu, EDA menjadi tahapan penting untuk memahami distribusi data, hubungan antarvariabel, dan kualitas data sebelum dilakukan proses *preprocessing* maupun pembangunan model *machine learning*.

2.7 Data Preprocessing

Data preprocessing merupakan tahapan yang dilakukan untuk mempersiapkan data mentah agar memiliki kualitas yang lebih baik dan sesuai untuk digunakan dalam proses analisis maupun pemodelan *machine learning*. Tahap ini bertujuan untuk meningkatkan kualitas data dengan menangani nilai hilang, inkonsistensi, data duplikat, dan perbedaan skala fitur sehingga data siap digunakan dalam proses pemodelan *machine learning*.

2.7.1 Data Cleaning

Data cleaning merupakan salah satu tahap penting dalam *preprocessing* yang bertujuan untuk meningkatkan kualitas data sebelum digunakan pada proses analisis dan pemodelan *machine learning*. Tahap ini dilakukan untuk menangani berbagai permasalahan pada data, seperti *missing values*, data duplikat, data tidak konsisten, serta fitur yang tidak relevan yang dapat memengaruhi performa model [28]. Kesalahan atau ketidakkonsistenan pada data dapat menyebabkan penurunan performa model serta menghasilkan prediksi yang tidak akurat [29]. Oleh karena itu, proses pembersihan data menjadi penting baik pada tahap pelatihan maupun implementasi model machine learning.

Pada data biologis seperti DNA methylation, proses *data cleaning* diperlukan karena data umumnya memiliki jumlah fitur yang sangat besar dan berpotensi mengandung *missing values*, data tidak konsisten, maupun *noise*. Keberadaan data yang tidak lengkap dapat memengaruhi proses pembelajaran model dan menurunkan kualitas prediksi yang dihasilkan. Oleh karena itu, diperlukan proses pembersihan data untuk memastikan bahwa data yang digunakan berada dalam kondisi yang lebih konsisten dan siap digunakan pada tahap *preprocessing* selanjutnya.

2.7.2 Data Splitting

Data splitting merupakan tahap penting dalam pengembangan model *machine learning* untuk memisahkan data menjadi data pelatihan, validasi, dan pengujian. Pemisahan ini bertujuan untuk mengevaluasi kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya serta mengurangi risiko overfitting. Praktik membagi dataset menjadi *data training* dan *testing* merupakan prosedur umum dalam pengembangan model statistik dan *machine learning*, di mana *data training* digunakan untuk membangun model, sedangkan *data testing* digunakan untuk mengevaluasi performanya pada data yang tidak digunakan selama proses pelatihan [30]. Selain itu, penggunaan validation set memungkinkan proses pemilihan model dan penyesuaian hiperparameter dilakukan tanpa melibatkan data test sehingga estimasi performa akhir model tetap objektif.

2.7.3 Cross Validation

Cross validation merupakan salah satu teknik evaluasi yang banyak digunakan dalam *machine learning* untuk mengukur kemampuan generalisasi suatu model terhadap data yang belum pernah digunakan pada proses pelatihan. Metode ini dilakukan dengan membagi dataset ke dalam beberapa subset (*fold*) dan menjalankan proses pelatihan serta evaluasi secara berulang menggunakan kombinasi data yang berbeda, sehingga setiap sampel memiliki kesempatan untuk digunakan sebagai data pelatihan maupun data validasi [31].

Salah satu metode yang paling umum digunakan adalah *k-fold cross validation*. Pada metode ini, dataset dibagi menjadi k subset dengan ukuran yang relatif sama. Pada setiap iterasi, sebanyak $k-1$ subset digunakan sebagai data pelatihan (*training set*), sedangkan satu subset sisanya digunakan sebagai data

validasi (*validation set*). Proses tersebut diulang sebanyak k kali hingga setiap subset pernah digunakan sebagai data validasi tepat satu kali. Selanjutnya, performa model diperoleh dengan menghitung rata-rata nilai metrik evaluasi dari seluruh *fold*.

Rata-rata hasil evaluasi pada *k-fold cross validation* dapat dihitung menggunakan Persamaan 2.2 [31].

$$CV = \frac{1}{k} \sum_{i=1}^{k} M_i \tag{2.2}$$

dengan:

- CV merupakan nilai rata-rata hasil *cross validation*,
- k merupakan jumlah *fold* yang digunakan,
- M_i merupakan nilai metrik evaluasi pada *fold* ke-i, seperti *Accuracy*, *Precision*, *Recall*, *F1-Score*, atau *ROC-AUC*.

2.8 Feature Selection

Feature selection merupakan proses pemilihan subset fitur yang paling relevan untuk digunakan dalam proses pemodelan [24]. Teknik ini banyak digunakan pada data berdimensi tinggi untuk mengurangi kompleksitas data serta meningkatkan performa model pembelajaran mesin. Pada data DNA methylation, jumlah fitur yang sangat besar dibandingkan dengan jumlah sampel menyebabkan proses pemodelan menjadi lebih kompleks, sehingga diperlukan proses seleksi fitur sebelum dilakukan klasifikasi.

Tujuan utama *feature selection* adalah mengurangi fitur yang tidak relevan dan redundan agar model dapat mempelajari pola data dengan lebih efektif [22]. Selain itu, proses ini dapat membantu mengurangi dimensi data, meningkatkan akurasi prediksi, menurunkan risiko *overfitting*, serta meningkatkan efisiensi komputasi. Oleh karena itu, *feature selection* menjadi salah satu tahapan penting dalam pengolahan data biologis berdimensi tinggi.

Pada penelitian ini, proses seleksi fitur dilakukan menggunakan dua pendekatan, yaitu *Differentially Methylated Positions* (DMP) dan *Mutual Information* (MI). Kedua metode tersebut digunakan untuk mengidentifikasi fitur-fitur DNA methylation yang memiliki relevansi tinggi terhadap target prediksi kelangsungan hidup pasien kanker payudara.

2.8.1 Differentially Methylated Positions (DMP)

Differentially Methylated Positions (DMP) merupakan metode yang digunakan untuk mengidentifikasi situs CpG yang memiliki perbedaan tingkat metilasi yang signifikan antara dua kelompok sampel [32]. Dalam penelitian kanker, DMP sering digunakan untuk menemukan biomarker epigenetik yang berkaitan dengan perkembangan penyakit maupun prognosis pasien [33].

Secara umum, analisis DMP dilakukan dengan membandingkan tingkat metilasi pada setiap situs CpG antara dua kelompok sampel menggunakan uji statistik. Hasil pengujian tersebut menghasilkan nilai *p-value* yang digunakan untuk menilai signifikansi perbedaan tingkat metilasi pada setiap situs CpG. Semakin kecil nilai *p-value*, semakin besar indikasi bahwa perbedaan metilasi yang diamati tidak terjadi secara kebetulan sehingga situs CpG tersebut dianggap lebih relevan untuk membedakan kedua kelompok sampel.

2.8.2 Variance Filtering

Variance filtering merupakan salah satu metode seleksi fitur sederhana yang digunakan untuk mengurangi fitur dengan variasi nilai yang sangat rendah pada dataset. Pada metode ini, fitur dengan variansi nol atau sangat rendah dihapus karena dianggap memiliki kontribusi yang kecil terhadap variasi data secara keseluruhan dan kurang memberikan informasi yang signifikan dalam proses klasifikasi [34]. Oleh karena itu, fitur-fitur tersebut dapat dihapus untuk mengurangi kompleksitas data sebelum dilakukan proses pemodelan machine learning.

Pada data biologis berdimensi tinggi seperti DNA methylation, jumlah fitur yang sangat besar dapat meningkatkan kompleksitas komputasi serta risiko *overfitting*. Proses *variance filtering* membantu mengurangi jumlah fitur awal dengan mempertahankan fitur yang memiliki variasi nilai lebih tinggi antar sampel sehingga lebih berpotensi memberikan informasi yang relevan dalam proses klasifikasi.

Variansi digunakan untuk mengukur tingkat penyebaran nilai suatu fitur terhadap rata-ratanya. Pada metode *variance filtering*, fitur dengan nilai variansi yang sangat rendah akan dihapus karena dianggap kurang berkontribusi dalam proses klasifikasi. Rumus variansi ditunjukkan pada Persamaan 2.3 [34].

$$Var(X) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2 \tag{2.3}$$

dengan:

- x_i merupakan nilai data ke-i,
- μ merupakan nilai rata-rata data,
- n merupakan jumlah sampel.

Mutual Information (MI) merupakan metode seleksi fitur berbasis teori informasi yang digunakan untuk mengukur tingkat ketergantungan antara suatu fitur dengan variabel target. Semakin besar nilai MI, semakin besar informasi yang diberikan suatu fitur terhadap target prediksi. Metode ini efektif digunakan untuk mengidentifikasi fitur yang paling informatif karena mampu menangkap hubungan linear maupun nonlinier antara fitur dan target [35].

Nilai *Mutual Information* antara variabel X dan Y dapat dihitung menggunakan Persamaan 2.4 [35].

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \tag{2.4}$$

dengan:

- $p(x,y)$ merupakan probabilitas gabungan (*joint probability*) antara X dan Y,
- $p(x)$ merupakan probabilitas marginal dari X,
- $p(y)$ merupakan probabilitas marginal dari Y.

Nilai MI yang tinggi menunjukkan bahwa suatu fitur memiliki hubungan yang kuat dengan target sehingga memberikan informasi yang lebih besar dalam proses prediksi. Pada proses seleksi fitur, setiap fitur diberikan skor berdasarkan nilai *Mutual Information*-nya terhadap label target. Selanjutnya, fitur dengan skor tertinggi dipilih untuk digunakan pada tahap pelatihan model. Melalui pendekatan ini, diperoleh subset fitur DNA methylation yang lebih informatif untuk membedakan kelompok pasien dengan kelangsungan hidup jangka pendek dan jangka panjang.

2.8.3 Data Scaling

Data scaling merupakan salah satu tahapan preprocessing yang bertujuan untuk menyamakan skala antarfitur sebelum proses pelatihan model. Penerapan teknik data scaling dapat memengaruhi kinerja berbagai algoritma *machine learning*, sehingga pemilihan metode scaling perlu disesuaikan dengan karakteristik data dan algoritma yang digunakan. Berbagai penelitian juga menunjukkan bahwa penerapan data scaling mampu meningkatkan performa model pada beberapa algoritma *machine learning*, meskipun efektivitas setiap metode scaling dapat berbeda bergantung pada karakteristik data dan algoritma yang digunakan [36].

2.9 Data Balancing

Pada permasalahan klasifikasi, distribusi jumlah sampel antar kelas sering kali tidak seimbang (*imbalanced data*). Ketidakseimbangan kelas (*class imbalance*) dapat menyebabkan model cenderung mempelajari pola dari kelas mayoritas sehingga performa prediksi pada kelas minoritas menjadi kurang optimal [10]. Oleh karena itu, diperlukan teknik *data balancing* untuk menghasilkan distribusi data yang lebih seimbang dan meningkatkan kemampuan generalisasi model.

Salah satu metode yang umum digunakan untuk mengatasi ketidakseimbangan kelas adalah SMOTE (*Synthetic Minority Oversampling Technique*). SMOTE bekerja dengan menghasilkan sampel sintetis baru pada kelas minoritas berdasarkan interpolasi antara suatu sampel dengan salah satu tetangga terdekatnya (*nearest neighbors*) [37]. Proses ini bertujuan untuk meningkatkan jumlah data pada kelas minoritas tanpa melakukan duplikasi secara langsung sehingga distribusi kelas menjadi lebih seimbang.

SMOTE-Tomek merupakan metode *combined re-sampling* yang menggabungkan teknik oversampling menggunakan SMOTE dan undersampling menggunakan Tomek Links. Setelah SMOTE menghasilkan sampel sintetis pada kelas minoritas, Tomek Links digunakan untuk mengidentifikasi pasangan sampel dari kelas yang berbeda yang saling menjadi tetangga terdekat dan berada pada batas antarkelas. Sampel yang membentuk pasangan Tomek kemudian dihapus untuk mengurangi tumpang tindih antar kelas sehingga menghasilkan distribusi data yang lebih bersih dan batas keputusan yang lebih jelas [38].

Proses pembentukan sampel sintetis pada metode SMOTE dihitung menggunakan Persamaan 2.5 [37].

$$x_{new} = x_i + \lambda (x_{nn} - x_i) \tag{2.5}$$

dengan:

- x_i merupakan sampel dari kelas minoritas,
- x_{nn} merupakan salah satu tetangga terdekat dari x_i,
- λ merupakan bilangan acak dengan rentang $0 \leq \lambda \leq 1$,
- x_{new} merupakan sampel sintetis yang dihasilkan.

Melalui kombinasi antara SMOTE dan Tomek Links, metode SMOTE-Tomek tidak hanya meningkatkan jumlah sampel pada kelas minoritas, tetapi juga membersihkan data pada area batas antarkelas. Dengan demikian, metode ini diharapkan dapat membantu model machine learning mempelajari pola data secara lebih baik dan meningkatkan performa klasifikasi, khususnya pada dataset yang memiliki ketidakseimbangan kelas.

2.10 Machine Learning

Machine learning merupakan bagian dari kecerdasan buatan (*artificial intelligence*) yang digunakan untuk menganalisis data dan mempelajari pola untuk menghasilkan prediksi [13]. Metode ini banyak digunakan dalam bidang kesehatan dan bioinformatika karena mampu menangani data biologis yang kompleks dan berdimensi tinggi.

Perkembangan metode machine learning memungkinkan integrasi berbagai jenis data biologis dalam suatu model prediksi sehingga dapat meningkatkan performa dan akurasi model [1]. Integrasi berbagai jenis data tersebut dapat membantu meningkatkan performa dan akurasi model dalam proses prediksi.

2.10.1 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan algoritma pembelajaran mesin terawasi (*supervised machine learning*) yang digunakan untuk menyelesaikan permasalahan klasifikasi maupun regresi. Pada permasalahan klasifikasi, SVM bekerja dengan membangun sebuah *hyperplane* optimal yang mampu memisahkan

data dari kelas yang berbeda dengan margin maksimum. Margin merupakan jarak antara *hyperplane* dengan data terdekat dari masing-masing kelas yang disebut sebagai *support vector*. Semakin besar margin yang diperoleh, semakin baik kemampuan model dalam melakukan generalisasi terhadap data baru. Selain itu, SVM dikenal memiliki performa yang baik pada data berdimensi tinggi dan tetap efektif ketika jumlah fitur jauh lebih besar dibandingkan jumlah sampel, sehingga banyak diterapkan pada berbagai bidang, khususnya kesehatan, untuk diagnosis penyakit, prediksi prognosis, dan klasifikasi data biomedis [39].

Fungsi keputusan pada SVM linear dinyatakan menggunakan Persamaan 2.6 [39].

$$f(x) = w^T x + b \tag{2.6}$$

dengan:

- w merupakan vektor bobot (*weight vector*),
- x merupakan vektor fitur masukan,
- b merupakan nilai bias.

Pada praktiknya, sebagian besar data tidak dapat dipisahkan secara sempurna oleh sebuah *hyperplane*. Oleh karena itu, SVM menggunakan pendekatan *soft-margin* yang memperbolehkan sejumlah data berada di dalam margin atau mengalami kesalahan klasifikasi. Tingkat toleransi terhadap kesalahan tersebut dikendalikan oleh parameter regularisasi C. Nilai C yang besar menghasilkan penalti yang lebih tinggi terhadap kesalahan klasifikasi sehingga model cenderung membentuk batas keputusan yang lebih ketat, sedangkan nilai C yang kecil menghasilkan margin yang lebih lebar dengan toleransi kesalahan yang lebih besar [40].

Pada banyak kasus, hubungan antar data bersifat nonlinier sehingga tidak dapat dipisahkan menggunakan *hyperplane* linear. Untuk mengatasi kondisi tersebut, SVM memanfaatkan konsep *kernel trick*, yaitu memetakan data ke ruang fitur berdimensi lebih tinggi tanpa harus menghitung transformasi tersebut secara eksplisit. Pendekatan ini memungkinkan data yang semula tidak dapat dipisahkan secara linear menjadi lebih mudah dipisahkan oleh sebuah *hyperplane*. Beberapa fungsi kernel yang umum digunakan pada SVM meliputi *linear*, *polynomial*,

sigmoid, dan *Radial Basis Function* (RBF). Di antara berbagai jenis kernel tersebut, kernel RBF merupakan salah satu yang paling banyak digunakan karena mampu memodelkan hubungan nonlinier yang kompleks serta memberikan performa yang baik pada data berdimensi tinggi, termasuk data medis dan genomik [39].

Pada penelitian ini digunakan kernel *Radial Basis Function* (RBF). Fungsi kernel RBF dihitung menggunakan Persamaan 2.7 [39].

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right) \tag{2.7}$$

dengan:

- $K(x_i, x_j)$ merupakan nilai kemiripan antara dua data,
- γ merupakan parameter yang mengatur jangkauan pengaruh setiap *support vector*,
- $\|x_i - x_j\|^2$ merupakan kuadrat jarak Euclidean antara dua data.

Parameter γ mengendalikan bentuk batas keputusan yang dihasilkan oleh kernel RBF. Nilai γ yang besar menyebabkan pengaruh setiap *support vector* menjadi lebih lokal sehingga menghasilkan batas keputusan yang lebih kompleks, sedangkan nilai γ yang kecil menghasilkan batas keputusan yang lebih halus. Oleh karena itu, kinerja SVM dengan kernel RBF dipengaruhi oleh kombinasi parameter C dan γ, sehingga kedua parameter tersebut umumnya ditentukan melalui proses optimasi hiperparameter sebelum model digunakan untuk melakukan klasifikasi [39].

2.10.2 Extreme Gradient Boosting (XGBoost)

Extreme Gradient Boosting (XGBoost) merupakan algoritma *ensemble learning* berbasis *gradient boosting* yang dikembangkan untuk meningkatkan kecepatan komputasi, akurasi prediksi, dan kemampuan generalisasi model. XGBoost membangun model secara bertahap dengan menambahkan pohon keputusan baru untuk memperbaiki kesalahan prediksi yang dihasilkan oleh model sebelumnya [41].

Salah satu keunggulan utama XGBoost adalah penggunaan regularisasi yang membantu mengurangi risiko *overfitting* serta meningkatkan performa model pada berbagai permasalahan klasifikasi dan prediksi [41].

Fungsi objektif pada XGBoost terdiri atas fungsi kerugian (*loss function*) dan komponen regularisasi yang dihitung menggunakan Persamaan 2.8 [42].

$$Obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{2.8}$$

dengan:

- $l(y_i, \hat{y}_i)$ merupakan fungsi kerugian (*loss function*),
- $\Omega(f_k)$ merupakan fungsi regularisasi,
- K merupakan jumlah pohon keputusan.

Fungsi regularisasi pada XGBoost dinyatakan menggunakan Persamaan 2.9 [42].

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{2.9}$$

dengan:

- γ merupakan parameter yang mengontrol kompleksitas model,
- T merupakan jumlah daun (*leaf*) pada pohon keputusan,
- λ merupakan parameter regularisasi terhadap bobot daun,
- w_j merupakan bobot pada daun ke-j.

2.10.3 Random Forest

Random Forest merupakan metode *ensemble learning* yang terdiri atas sekumpulan pohon keputusan (*decision tree*) yang dibangun secara independen. Setiap pohon dilatih menggunakan sampel data yang dipilih secara acak melalui teknik *bootstrap sampling*, sedangkan pemilihan fitur pada setiap node dilakukan secara acak untuk meningkatkan keragaman model [43].

Metode ini menggabungkan hasil prediksi dari banyak pohon keputusan untuk menghasilkan model yang lebih stabil dan memiliki kemampuan generalisasi yang lebih baik dibandingkan satu pohon keputusan tunggal. Selain itu, Random

Forest mampu mengurangi risiko *overfitting* serta meningkatkan akurasi klasifikasi pada berbagai permasalahan prediksi [43].

Pada proses klasifikasi, prediksi akhir diperoleh melalui mekanisme *majority voting*, yaitu kelas yang paling banyak diprediksi oleh seluruh pohon keputusan akan dipilih sebagai hasil klasifikasi akhir. Persamaan *majority voting* ditunjukkan pada Persamaan 2.10 [43].

$$\hat{y} = \text{mode}\left(h_1(x), h_2(x), \ldots, h_B(x)\right) \tag{2.10}$$

dengan:

- $h_b(x)$ merupakan prediksi yang dihasilkan oleh pohon keputusan ke-b,
- B merupakan jumlah pohon keputusan dalam *Random Forest*,
- $\hat{y}$ merupakan hasil klasifikasi akhir.

2.10.4 Multilayer Perceptron (MLP)

Multilayer Perceptron (MLP) merupakan salah satu arsitektur *Artificial Neural Network* (ANN) bertipe *feed-forward* yang banyak digunakan dalam berbagai tugas klasifikasi dan prediksi. MLP memiliki kemampuan yang baik dalam memodelkan hubungan nonlinier antar variabel sehingga dapat menghasilkan performa prediksi yang tinggi pada berbagai jenis data [44]. Selain itu, MLP juga telah banyak diterapkan pada bidang kesehatan untuk menyelesaikan permasalahan klasifikasi maupun prediksi berbasis data medis [45].

Secara umum, arsitektur MLP terdiri atas tiga komponen utama, yaitu *input layer*, satu atau lebih *hidden layer*, dan *output layer*. Setiap neuron pada suatu lapisan terhubung dengan neuron pada lapisan berikutnya melalui bobot (*weights*) yang akan diperbarui selama proses pelatihan menggunakan algoritma *backpropagation*. Informasi diproses secara berurutan dari lapisan masukan menuju lapisan keluaran sehingga model mampu mempelajari pola yang kompleks dari data.

Proses komputasi pada sebuah neuron di dalam *Multilayer Perceptron* (MLP) dihitung menggunakan Persamaan 2.11 [46].

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{2.11}$$

dengan:

- x_i merupakan fitur masukan ke-i,
- w_i merupakan bobot yang terkait dengan fitur ke-i,
- b merupakan nilai bias,
- $f(\cdot)$ merupakan fungsi aktivasi, dan
- y merupakan keluaran neuron.

Melalui kombinasi beberapa lapisan tersembunyi dan fungsi aktivasi nonlinier, MLP mampu mempelajari representasi data yang kompleks sehingga banyak digunakan dalam berbagai aplikasi machine learning, termasuk prediksi dan klasifikasi pada data berdimensi tinggi.

2.10.5 Threshold Tuning

Threshold tuning merupakan proses penentuan nilai ambang (*decision threshold*) yang digunakan untuk mengonversi probabilitas keluaran model menjadi label kelas. Pada klasifikasi biner, nilai ambang bawaan umumnya ditetapkan sebesar 0,5, namun nilai tersebut belum tentu menghasilkan performa terbaik, terutama pada dataset yang memiliki distribusi kelas tidak seimbang. Oleh karena itu, penyesuaian nilai *threshold* dapat dilakukan untuk memperoleh keseimbangan yang lebih baik antara *precision* dan *recall*, sehingga meningkatkan kualitas hasil klasifikasi. Penelitian mengenai klasifikasi berbasis *threshold* juga menunjukkan bahwa proses *threshold tuning* setelah pelatihan model dapat digunakan untuk meningkatkan performa prediksi tanpa mengubah model yang telah dilatih [47].

2.11 Model Evaluation

2.11.1 Confusion Matrix

Confusion matrix merupakan metode evaluasi yang digunakan untuk menganalisis kinerja model klasifikasi dengan membandingkan hasil prediksi terhadap label sebenarnya. Konsep ini pertama kali diperkenalkan oleh Karl Pearson pada tahun 1904 sebagai *contingency table*, kemudian dikenal sebagai *classification matrix*, dan akhirnya lebih populer dengan istilah *confusion matrix*

[48]. Matriks ini berbentuk persegi berukuran $N \times N$, dengan N menyatakan jumlah kelas pada masalah klasifikasi. Setiap baris menunjukkan jumlah sampel yang diprediksi pada suatu kelas, sedangkan setiap kolom menunjukkan jumlah sampel pada kelas sebenarnya. Dengan demikian, *confusion matrix* memberikan informasi mengenai prediksi yang benar maupun kesalahan klasifikasi yang dilakukan oleh model, baik pada permasalahan klasifikasi biner maupun multikelas. Selain itu, matriks ini menjadi dasar dalam perhitungan berbagai metrik evaluasi, seperti *Accuracy*, *Precision*, *Recall*, dan *F1-Score*, sehingga dapat digunakan untuk mengevaluasi serta meningkatkan performa model pada metode pembelajaran terawasi (*supervised learning*).

Tabel 2.1. Confusion Matrix untuk Klasifikasi Biner

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

2.11.2 Accuracy

Accuracy digunakan untuk mengukur proporsi jumlah prediksi yang benar yang dihasilkan oleh model machine learning terhadap seluruh data yang diuji. Nilai accuracy dihitung berdasarkan jumlah *true positive* (TP) dan *true negative* (TN) dibandingkan dengan total seluruh prediksi [49]. Rumus 2.12 menunjukkan cara perhitungan *Accuracy* [49].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{2.12}$$

2.11.3 Precision

Precision digunakan untuk mengukur tingkat ketepatan model dalam memprediksi kelas positif. Metrik ini menunjukkan perbandingan antara jumlah data yang diprediksi positif secara benar dengan seluruh data yang diprediksi positif oleh model [49]. Rumus 2.13 menunjukkan cara perhitungan *Precision* [49].

$$Precision = \frac{TP}{TP + FP} \times 100 \tag{2.13}$$

2.11.4 Recall atau Sensitivity

Recall atau *Sensitivity* digunakan untuk mengukur kemampuan model dalam mengidentifikasi seluruh data positif yang sebenarnya. Nilai *recall* menunjukkan proporsi data positif yang berhasil diklasifikasikan dengan benar oleh model [49]. Rumus 2.14 menunjukkan cara perhitungan *Recall* [49].

$$Recall = \frac{TP}{TP + FN} \tag{2.14}$$

2.11.5 F1-Score

F1-Score merupakan metrik evaluasi yang diperoleh dari kombinasi antara precision dan recall. Metrik ini digunakan untuk memberikan keseimbangan antara kemampuan model dalam melakukan prediksi positif secara tepat dan kemampuan mendeteksi seluruh data positif [49]. Rumus 2.15 menunjukkan cara perhitungan *F1-Score* [49].

$$F1\text{-}Score = \frac{2TP}{2TP + FP + FN} \tag{2.15}$$

2.11.6 Macro F1-Score

Macro F1-Score merupakan metrik evaluasi yang diperoleh dengan menghitung rata-rata nilai *F1-Score* dari setiap kelas. Berbeda dengan *F1-Score* yang hanya mengevaluasi satu kelas tertentu, *Macro F1-Score* memberikan bobot yang sama pada seluruh kelas tanpa mempertimbangkan distribusi jumlah sampel. Oleh karena itu, metrik ini sesuai digunakan pada permasalahan klasifikasi dengan distribusi kelas yang tidak seimbang karena mampu memberikan evaluasi yang lebih adil terhadap performa model pada setiap kelas. Pada kasus klasifikasi biner, nilai *Macro F1-Score* dihitung sebagai rata-rata *F1-Score* dari kedua kelas. Rumus 2.16 menunjukkan cara perhitungan *Macro F1-Score*.

$$\text{Macro F1} = \frac{F1_0 + F1_1}{2} \tag{2.16}$$

2.11.7 ROC-AUC

Receiver Operating Characteristic (ROC) digunakan untuk menggambarkan hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) pada berbagai nilai threshold. Luas area di bawah kurva ROC atau *Area Under Curve* (AUC) digunakan untuk mengukur kemampuan model dalam membedakan kelas positif dan negatif. Semakin tinggi nilai AUC, maka semakin baik performa model dalam melakukan klasifikasi [49]. Rumus 2.17 menunjukkan cara perhitungan *AUC* [49].

$$AUC = \int_{x=0}^{1} \frac{TP}{TP+FN} \left(\frac{FP}{FP+TN} \right)^{-1} (x)\, dx \tag{2.17}$$

2.12 Model Interpretation

SHapley Additive exPlanations (SHAP) merupakan salah satu metode *Explainable Artificial Intelligence* (XAI) yang digunakan untuk menginterpretasikan hasil prediksi model *machine learning*. SHAP didasarkan pada konsep nilai Shapley dari teori permainan (*game theory*), di mana kontribusi setiap fitur terhadap keluaran model dihitung secara adil dengan mempertimbangkan seluruh kemungkinan kombinasi fitur. Metode ini mampu memberikan interpretasi baik pada tingkat lokal untuk menjelaskan prediksi suatu sampel maupun pada tingkat global untuk mengevaluasi pengaruh setiap fitur terhadap keseluruhan model [50].

Secara matematis, nilai SHAP untuk suatu fitur diperoleh dari rata-rata kontribusi marginal fitur tersebut terhadap keluaran model ketika ditambahkan ke dalam seluruh kemungkinan subset fitur. Persamaan nilai SHAP dinyatakan sebagai berikut.

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F|-|S|-1)!}{|F|!} [f(S \cup \{i\}) - f(S)] \tag{2.18}$$

dengan:

- ϕ_i merupakan nilai SHAP untuk fitur ke-i,

- F merupakan himpunan seluruh fitur pada model,
- S merupakan subset fitur yang tidak mengandung fitur ke-i,
- $|S|$ merupakan jumlah elemen pada subset S,
- $|F|$ merupakan jumlah total fitur,
- $f(S)$ merupakan keluaran (*output*) model ketika hanya fitur-fitur dalam subset S yang digunakan, dan
- $f(S \cup \{i\})$ merupakan keluaran model setelah fitur ke-i ditambahkan ke dalam subset S.

Nilai SHAP yang positif menunjukkan bahwa suatu fitur memberikan kontribusi yang mendorong keluaran model ke arah prediksi yang lebih tinggi terhadap kelas target, sedangkan nilai SHAP yang negatif menunjukkan bahwa fitur tersebut menurunkan keluaran model terhadap kelas target. Dengan demikian, SHAP dapat digunakan untuk mengidentifikasi fitur-fitur yang paling berpengaruh terhadap keputusan model serta meningkatkan transparansi dan interpretabilitas model, terutama pada aplikasi di bidang kesehatan dan bioinformatika [50].

2.13 Penelitian Terdahulu

Berbagai penelitian telah dilakukan dalam prediksi kelangsungan hidup pasien kanker payudara dengan memanfaatkan metode *machine learning* dan data biologis, termasuk DNA methylation. Penelitian-penelitian tersebut menggunakan berbagai pendekatan, horizon waktu, serta metode klasifikasi yang berbeda. Ringkasan penelitian terdahulu yang relevan disajikan pada Tabel 2.2.

Tabel 2.2. Ringkasan penelitian terdahulu terkait prediksi kelangsungan hidup pasien kanker payudara

Penelitian	**Pasien**	**Fitur**	**Metode dan Hasil**
CAGCL: Predicting Short- and Long-Term Breast Cancer Survival With Cross-Modal Attention and Graph Contrastive Learning [4]	1035	PCA (500 fitur multi-omics dan 800 fitur WSI)	CAGCL untuk prediksi *short-term* dan *long-term survival* berdasarkan batas 5 tahun (Akurasi 93.2%, AUC 0.948).
Deep Learning-Based Feature-Level Integration of Multi-Omics Data for Breast Cancer Survival Analysis [7]	$\sim$1000	DNA methylation, miRNA, gene expression, dan CNV	ConcatAE dan CrossAE untuk prediksi *survival* berbasis multi-omics (C-index terbaik 0.641 $\pm$ 0.031 menggunakan ConcatAE).
eBreCaP: Extreme Learning-Based Model for Breast Cancer Survival Prediction [51]	578	Gene expression, CNA, DNA methylation, protein expression, dan citra patologis	Ensemble *Extreme Learning Machine* (ELM) dengan *gradient boosting* untuk prediksi *survival* kanker payudara (Akurasi 85%).
Predicting Breast Cancer Mortality Using SEER Data [10]	4005	31 variabel klinis dan demografis	*Fixed-horizon classification* pada horizon 3 dan 5 tahun menggunakan *Logistic Regression* dan *Neural Network* (ROC-AUC 0.78 pada 3 tahun dan 0.75 pada 5 tahun).

Lanjut ke halaman berikutnya

Tabel 2.2 – lanjutan dari halaman sebelumnya

Penelitian	**Pasien**	**Fitur**	**Metode dan Hasil**
Novel Models Based on Machine Learning to Predict the Prognosis of Metaplastic Breast Cancer [11]	3008	Faktor prognostik klinis dari dataset SEER	Model CatBoost untuk prediksi *survival* pada horizon 1, 3, dan 5 tahun (AUC 0.833, 0.806, dan 0.810).
DNA Methylation-Based Prognostic Subgroups and Prognosis Prediction Model for Breast Cancer Patients [52]	1232	21.122 situs CpG promoter DNA methylation	*Consensus Clustering* dan Cox Regression berbasis DNA methylation untuk prediksi *survival* kanker payudara (AUC *training* 0.757 dan AUC *testing* 0.601).
Integrative Survival Analysis of Breast Cancer with Gene Expression and DNA Methylation Data [53]	485	20.533 fitur mRNA dan 20.106 fitur DNA methylation	*Adaptive Multi-task Learning* berbasis BiLSTM dan *Ordinal Cox Model* untuk prediksi *survival* kanker payudara (C-index 0.722).