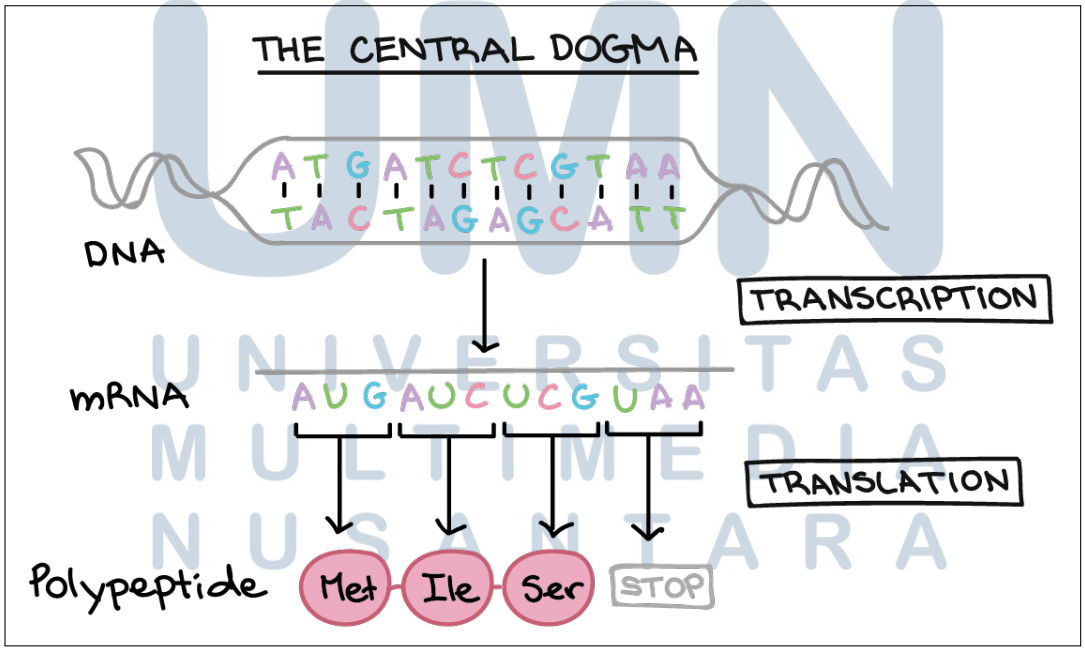
BAB 2
LANDASAN TEORI

2.1 Bioinformatika

Bioinformatika merupakan bidang interdisipliner yang menggabungkan biologi, statistika, dan ilmu komputer untuk menganalisis data biologis berskala besar. Dalam penelitian ini, bioinformatika digunakan untuk mengolah data DNA methylation sebagai dasar klasifikasi stadium kanker prostat.

2.1.1 Epigenetik dan Metilasi DNA

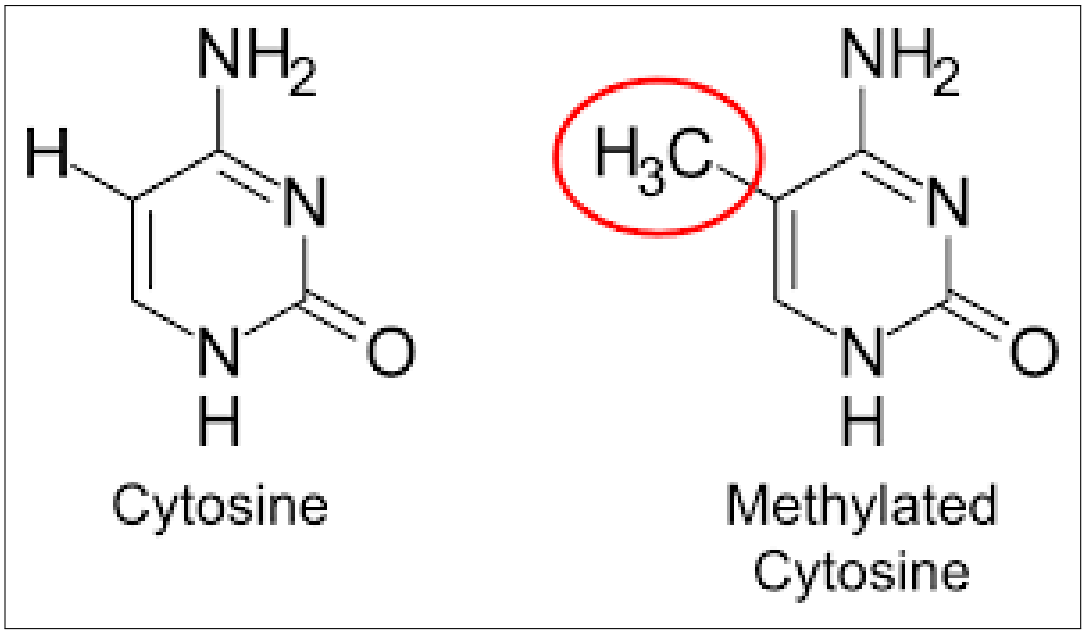
Ekspresi gen dalam sel mengikuti konsep *central dogma* biologi molekuler, yaitu aliran informasi genetik dari DNA ke RNA melalui proses transkripsi, kemudian diterjemahkan menjadi protein melalui proses translasi. Protein yang dihasilkan berperan dalam berbagai fungsi biologis sel. Regulasi terhadap proses transkripsi menjadi salah satu mekanisme penting yang menentukan apakah suatu gen akan diekspresikan atau tidak. Alur *central dogma* biologi molekuler ditunjukkan pada Gambar 2.1 [16]. Perubahan ini bersifat reversibel dan berperan penting dalam mengatur kapan dan bagaimana suatu gen diekspresikan.



Gambar 2.1. Ilustrasi *central dogma* biologi molekuler.

Epigenetik merupakan cabang ilmu biologi yang mempelajari perubahan fungsi gen yang tidak disebabkan oleh perubahan urutan basa DNA, melainkan oleh modifikasi kimia yang bersifat dinamis dan dapat diwariskan. Mekanisme epigenetik utama meliputi metilasi DNA dan modifikasi histon, yang keduanya berkontribusi dalam mengontrol aktivitas gen tanpa mengubah struktur dasar genom [17].

Salah satu mekanisme epigenetik yang paling banyak dipelajari adalah metilasi DNA (*DNA methylation*). Proses ini terjadi melalui penambahan gugus metil ($-CH_3$) pada basa sitosin, khususnya pada wilayah CpG *site*, yaitu lokasi pada DNA di mana nukleotida sitosin (C) diikuti oleh guanin (G) yang dihubungkan oleh ikatan fosfodiester. Kumpulan CpG *site* yang terkonsentrasi pada suatu wilayah dikenal sebagai CpG *island*, yang umumnya terletak pada daerah promotor gen dan memiliki peran penting dalam regulasi ekspresi gen[18]. Secara fungsional, tingkat metilasi pada CpG *site* berkorelasi dengan aktivitas gen. Hipermetilasi pada daerah promotor gen cenderung menghambat proses transkripsi sehingga gen menjadi tidak aktif, sedangkan hipometilasi umumnya berkaitan dengan peningkatan ekspresi gen. Secara skematis, proses metilasi DNA pada CpG *site* ditunjukkan pada Gambar 2.2 [19]. Oleh karena itu, perubahan pola metilasi dapat memberikan dampak signifikan terhadap fungsi sel.



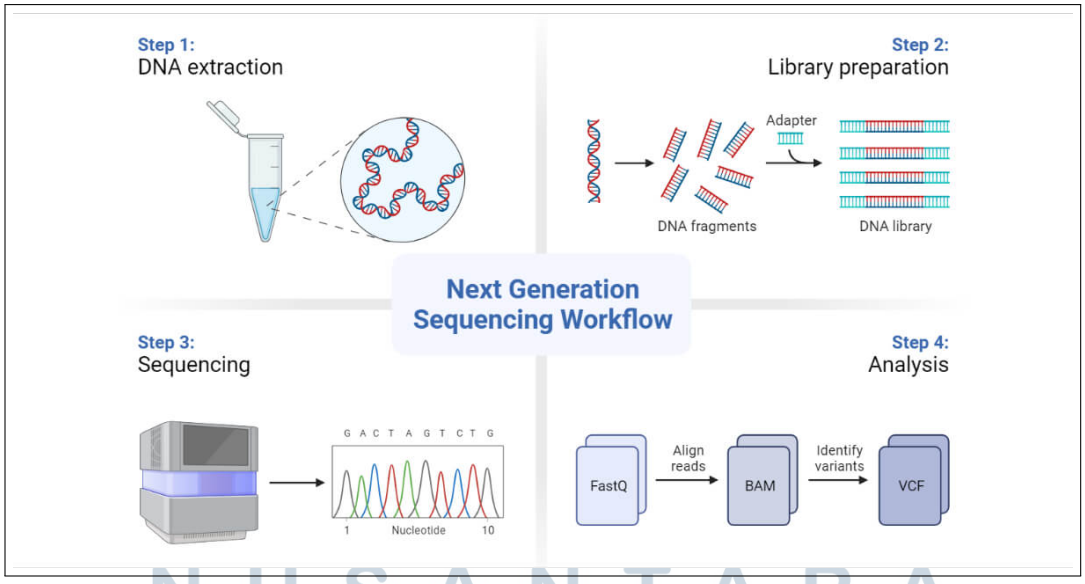
Gambar 2.2. Ilustrasi proses metilasi DNA pada CpG site.

Dalam konteks kanker, metilasi DNA menjadi salah satu indikator

epigenetik yang krusial. Hipermetilasi pada gen penekan tumor dapat menyebabkan inaktivasi gen tersebut, sementara hipometilasi global dapat meningkatkan instabilitas genom. Pola-pola perubahan ini menjadikan metilasi DNA sebagai biomarker yang potensial untuk deteksi, klasifikasi, dan analisis perkembangan kanker [5].

Namun demikian, pola metilasi DNA tidak bersifat universal. Variasi epigenetik dapat terjadi antar individu maupun antar kelompok populasi, yang dipengaruhi oleh faktor genetik, lingkungan, gaya hidup, serta faktor sosial. Sejumlah penelitian menunjukkan bahwa perbedaan ras atau populasi dapat menghasilkan variasi signifikan pada pola metilasi di berbagai CpG *site* [9]. Implikasi dari variasi ini sangat penting dalam pengembangan model prediksi berbasis data biologis, karena biomarker yang relevan pada satu populasi belum tentu memiliki signifikansi yang sama pada populasi lainnya. Oleh sebab itu, pemahaman terhadap variasi epigenetik menjadi aspek penting dalam analisis data dan pengembangan model *machine learning*, terutama untuk mengurangi bias serta meningkatkan kemampuan generalisasi model.

2.1.2 Next Generation Sequencing (NGS)



Gambar 2.3. Illustrasi proses umum NGS.

Teknologi NGS menghasilkan data dengan dimensi yang sangat tinggi, sering kali mencakup ratusan ribu hingga jutaan CpG *site*. Hal ini menimbulkan

tantangan dalam pengolahan data, seperti kebutuhan akan metode seleksi fitur dan model yang mampu menangani kompleksitas tinggi [6].

Dalam penelitian ini, data DNA methylation yang diperoleh dari hasil NGS digunakan sebagai fitur utama dalam model *machine learning*. Setiap CpG *site* direpresentasikan sebagai variabel numerik (*beta value*) yang mencerminkan kondisi epigenetik pasien, dan digunakan untuk mengklasifikasikan stadium kanker prostat.

2.1.3 Differentially Methylated Positions (DMP)

Differentially Methylated Positions (DMP) adalah CpG *site* individual yang menunjukkan perbedaan tingkat metilasi secara signifikan antara dua kelompok sampel, seperti pada analisis fenotipe biner. DMP berfokus pada pengujian setiap CpG *site* secara independen untuk mendeteksi perubahan metilasi spesifik pada posisi tertentu di genom [9]. Dalam praktiknya, analisis DMP dilakukan menggunakan model statistik seperti regresi logistik atau uji linier, di mana status fenotipe biner digunakan sebagai variabel dependen dan nilai metilasi (beta value) sebagai variabel prediktor. Setiap CpG kemudian diuji untuk mengetahui apakah terdapat asosiasi signifikan antara tingkat metilasi dan kondisi biologis yang diamati [11]. Dalam konteks analisis dengan fenotipe biner, DMP diidentifikasi dengan membandingkan distribusi metilasi antara dua kelompok menggunakan pendekatan statistik yang menghasilkan nilai *P-value* dan ukuran seperti koefisien regresi [10]. Untuk mengatasi masalah pengujian multipel yang timbul akibat jumlah CpG yang sangat besar, dilakukan koreksi seperti *False Discovery Rate* (FDR) untuk mengontrol tingkat kesalahan tipe I. CpG *site* yang memenuhi ambang signifikansi setelah koreksi ini kemudian diklasifikasikan sebagai DMP.

Secara biologis, DMP merepresentasikan perubahan metilasi yang spesifik dan terlokalisasi, yang dapat berperan dalam regulasi gen, terutama jika berada pada daerah fungsional seperti *promoter*, *enhancer*, atau CpG *island*. Namun, karena analisis dilakukan pada tingkat individual, DMP cenderung lebih sensitif terhadap variasi lokal dan *noise* dibandingkan DMR. Meskipun demikian, DMP sangat berguna untuk mengidentifikasi potensi kandidat biomarker spesifik yang berhubungan langsung dengan perubahan fenotipe. Dalam penelitian berbasis penyakit, termasuk kanker, identifikasi DMP antara kelompok kondisi (misalnya sampel normal dan tumor) dapat mengungkap CpG *site* yang mengalami metilasi abnormal. CpG yang signifikan ini dapat digunakan sebagai fitur dalam analisis

lanjutan, seperti klasifikasi atau pemodelan prediktif. Dalam konteks analisis fenotipe biner menggunakan *methylize*, hasil DMP memberikan daftar CpG yang paling berasosiasi dengan kondisi yang diteliti, sehingga berperan penting dalam memahami mekanisme epigenetik yang mendasari perbedaan biologis antar kelompok [20].

2.2 Machine Learning

Machine learning merupakan cabang dari kecerdasan buatan yang memungkinkan model untuk belajar dari data secara otomatis dan meningkatkan performanya seiring waktu tanpa diprogram secara eksplisit untuk setiap tugas. Secara umum, machine learning dibagi menjadi tiga model pembelajaran utama: *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. Pada *supervised learning*, model dilatih menggunakan pasangan data masukan dan label keluaran yang telah diketahui, sehingga model dapat belajar memetakan hubungan antara fitur dan kelas target. Model *supervised learning* inilah yang digunakan dalam penelitian ini, di mana data metilasi DNA beserta label stadium kanker digunakan sebagai basis pelatihan model [21].

Dalam konteks bioinformatika, machine learning memiliki peran yang semakin penting karena data omik seperti metilasi DNA memiliki dimensi yang sangat tinggi (ratusan ribu fitur per sampel) namun jumlah sampel yang tersedia relatif terbatas. Kondisi ini dikenal sebagai masalah *curse of dimensionality*, di mana peningkatan dimensi data secara drastis mempersulit proses pembelajaran model. Oleh karena itu, teknik machine learning yang dikombinasikan dengan seleksi fitur menjadi pendekatan yang tepat untuk mengekstrak pola biologis antar fitur yang bermakna dari data [22].

2.2.1 Feature Selection

Feature selection adalah proses identifikasi dan pemilihan subset fitur yang paling informatif dari keseluruhan ruang fitur untuk digunakan dalam pembangunan model prediktif. Tujuan utama dari proses ini adalah mengurangi dimensionalitas data, meningkatkan performa dan generalisasi model, mempercepat waktu komputasi, serta meningkatkan interpretabilitas model [23].

Metode seleksi fitur secara umum dikategorikan menjadi tiga pendekatan: *filter methods*, *wrapper methods*, dan *embedded methods*. *Filter methods*

mengevaluasi relevansi fitur secara independen dari model pembelajaran berdasarkan ukuran statistik seperti korelasi, chi-square, atau T test. *Wrapper methods* menggunakan model pembelajaran sebagai fungsi evaluasi untuk menilai subset fitur, sehingga mempertimbangkan interaksi antar fitur dalam konteks model tertentu. *Embedded methods* mengintegrasikan proses seleksi fitur ke dalam algoritma pelatihan model itu sendiri, seperti regularisasi L1 (LASSO). Dalam penelitian ini, digunakan kombinasi antara filter method (analisis DMP dengan uji statistik) dan wrapper method (RFE), yang memungkinkan seleksi fitur berjalan dalam dua tahap yang saling mengkomplemen untuk mendapatkan subset CpG yang paling relevan secara biologis maupun statistik.

A Recursive Feature Elimination (RFE)

Recursive Feature Elimination (RFE) adalah metode seleksi fitur berbasis *wrapper* yang secara iteratif memangkas fitur-fitur yang dianggap kurang penting berdasarkan bobot yang diberikan oleh model klasifikasi yang digunakan sebagai estimator. Proses RFE dimulai dengan melatih model pada seluruh fitur yang tersedia, kemudian menghapus fitur dengan nilai kepentingan terendah pada setiap iterasi, dan melatih ulang model hingga diperoleh jumlah fitur yang diinginkan [24].

Secara formal, diberikan himpunan fitur awal $F = \{f_1, f_2, \ldots, f_n\}$, RFE pada setiap iterasi t melakukan langkah-langkah berikut:

1. Melatih model klasifikasi M pada fitur saat ini F_t
2. Menghitung skor kepentingan w_i untuk setiap fitur $f_i \in F_t$ berdasarkan bobot atau koefisien model
3. Menghapus fitur dengan skor kepentingan terendah: $F_{t+1} = F_t \setminus \{\arg\min_i w_i\}$
4. Mengulangi langkah di atas hingga $|F_t|$ mencapai jumlah fitur target

Dalam penelitian ini, Support Vector Classifier (SVC) dengan kernel linear digunakan sebagai estimator pada RFE. Pemilihan SVC didasarkan pada kemampuannya menghasilkan koefisien fitur yang terstruktur melalui margin maksimum, yang memberikan dasar yang kuat untuk menilai kontribusi relatif setiap fitur CpG terhadap klasifikasi stadium.

2.2.2 Multilayer Perceptron (MLP)

Multilayer Perceptron (MLP) adalah salah satu jenis arsitektur jaringan saraf tiruan, yang terdiri dari setidaknya tiga lapisan: lapisan input, satu atau lebih *hidden layer*, dan lapisan output. MLP termasuk dalam kategori *feedforward neural network*, di mana sinyal mengalir secara satu arah dari lapisan input menuju lapisan output tanpa adanya siklus atau rekursi. Setiap neuron pada suatu *layer* terhubung dengan semua neuron pada *layer* berikutnya melalui bobot yang dapat dipelajari, sehingga MLP juga disebut sebagai *fully connected network* [25].

Secara matematis, output dari satu *hidden layer* dapat dituliskan sebagai:

$$\mathbf{h} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{2.1}$$

di mana $\mathbf{x}$ adalah vektor input, $\mathbf{W}$ adalah matriks bobot, $\mathbf{b}$ adalah vektor bias, dan σ adalah fungsi aktivasi. Proses pembelajaran pada MLP dilakukan melalui algoritma *backpropagation* yang dikombinasikan dengan metode optimasi berbasis gradien. Algoritma ini menghitung gradien dari fungsi kerugian terhadap setiap bobot dalam jaringan menggunakan aturan rantai (*chain rule*), kemudian memperbarui bobot secara bertahap untuk meminimalkan kesalahan prediksi.

Menurut *universal approximation theorem*, sebuah MLP dengan satu *hidden layer* yang memiliki jumlah neuron yang cukup dapat mengaproksimasi fungsi kontinu apapun dengan presisi yang tidak menentu. Meskipun demikian, jaringan dengan dua *hidden layer* seringkali lebih efisien secara komputasi karena mampu merepresentasikan fungsi kompleks dengan jumlah parameter yang lebih sedikit dibandingkan jaringan satu *layer* yang sangat lebar [26]. Dalam penelitian ini, MLP dengan satu hingga dua *hidden layer* digunakan untuk menghindari resiko *overfitting* pada dataset biomedis berdimensi tinggi namun berukuran sampel terbatas.

A Fungsi Aktivasi

Fungsi aktivasi merupakan komponen dalam MLP yang memperkenalkan non-linearitas ke dalam jaringan. Tanpa fungsi aktivasi, lapisan-lapisan linear hanya akan ekuivalen dengan satu transformasi linear tunggal, sehingga jaringan tidak akan mampu memodelkan hubungan atau pola yang kompleks dalam data.

ReLU (Rectified Linear Unit) didefinisikan sebagai:

$$\text{ReLU}(x) = \max(0, x) \tag{2.2}$$

ReLU banyak digunakan pada *hidden layer* karena beberapa keunggulannya. Pertama, gradiennya bersifat konstan (bernilai 1) untuk input positif, sehingga secara signifikan mengurangi masalah *vanishing gradient* yang umum terjadi pada fungsi sigmoid atau tanh dalam jaringan yang dalam. Kedua, karena ReLU menghasilkan output nol untuk semua input negatif, ia mendorong *sparsity* dalam aktivasi jaringan, yang dapat meningkatkan efisiensi representasi. Ketiga, operasinya yang sederhana membuat komputasi menjadi lebih efisien.

Sigmoid didefinisikan sebagai:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.3}$$

Fungsi ini mengkompres nilai input ke rentang $(0, 1)$, yang menjadikannya interpretasi alami sebagai probabilitas untuk klasifikasi biner. Pada lapisan output jaringan dengan tugas klasifikasi biner, output sigmoid dapat diinterpretasikan langsung sebagai probabilitas bahwa suatu sampel termasuk ke dalam kelas positif. Ambang batas 0.5 umumnya digunakan untuk menentukan kelas prediksi akhir.

B Optimizer

Optimizer menentukan bagaimana bobot jaringan diperbarui selama proses pelatihan berdasarkan gradien yang dihitung melalui *backpropagation*. Adam (*Adaptive Moment Estimation*) adalah algoritma optimasi berbasis gradien stokastik yang dikembangkan oleh Kingma dan Ba dan telah menjadi salah satu optimizer paling populer dalam deep learning [27].

Adam mengombinasikan keunggulan dua metode optimasi sebelumnya: *momentum* yang menyimpan rata-rata bergerak dari gradien masa lalu, dan *RMSProp* yang menyimpan rata-rata bergerak dari kuadrat gradien. Secara

matematis, pembaruan parameter pada Adam dinyatakan sebagai:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{2.4}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{2.5}$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{2.6}$$

$$\theta_t = \theta_{t-1} - \frac{\alpha}{\sqrt{\hat{v}_t} + \varepsilon} \hat{m}_t \tag{2.7}$$

di mana g_t adalah gradien pada iterasi ke-t, β_1 dan β_2 adalah koefisien peluruhan (umumnya 0.9 dan 0.999), α adalah *learning rate*, dan ε adalah konstanta kecil untuk stabilitas numerik. Koreksi bias $\hat{m}_t$ dan $\hat{v}_t$ memastikan estimasi momen tidak bias pada iterasi awal pelatihan.

C Loss Function

Fungsi kerugian (*loss function*) mengukur seberapa besar perbedaan antara prediksi model dan nilai label yang sesungguhnya, dan merupakan objektif yang diminimalkan selama proses pelatihan. Untuk tugas klasifikasi biner seperti dalam penelitian ini (stadium awal vs. stadium lanjut), binary cross-entropy digunakan sebagai fungsi kerugian, yang didefinisikan sebagai:

$$\mathscr{L} = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1-y_i) \log(1-\hat{y}_i)] \tag{2.8}$$

di mana N adalah jumlah sampel, $y_i \in \{0,1\}$ adalah label aktual, dan $\hat{y}_i \in (0,1)$ adalah probabilitas prediksi model untuk sampel ke-i. Formulasi ini berasal dari prinsip *maximum likelihood estimation* (MLE) pada distribusi Bernoulli, di mana meminimalkan binary cross-entropy ekuivalen dengan memaksimalkan log-likelihood dari data pelatihan. Ketika prediksi sangat menyimpang dari label sebenarnya, nilai kerugian meningkat secara logaritmik, sehingga memberikan sinyal pelatihan yang kuat untuk memperbaiki prediksi yang salah jauh [28].

D Early Stopping

Early stopping adalah teknik regularisasi yang mencegah *overfitting* dengan menghentikan proses pelatihan secara otomatis sebelum model mencapai jumlah

epoch maksimum yang telah ditentukan. *Overfitting* terjadi ketika model terlalu menyesuaikan diri dengan pola noise dalam data pelatihan sehingga kehilangan kemampuan generalisasinya terhadap data baru yang belum pernah dilihat sebelumnya [27].

Selama pelatihan, performa model dievaluasi secara periodik pada data validasi yang tidak digunakan dalam proses pembaruan bobot. Jika nilai metrik validasi (umumnya *validation loss*) tidak menunjukkan perbaikan selama sejumlah epoch tertentu yang disebut *patience*, pelatihan dihentikan. Kondisi ini mengindikasikan bahwa model telah mencapai titik generalisasi optimalnya, dan pelatihan lebih lanjut hanya akan memperburuk performa pada data yang belum dilihat.

2.3 Evaluasi Model

Evaluasi model dilakukan untuk mengukur kinerja model dalam melakukan klasifikasi. Beberapa metrik yang umum digunakan adalah sebagai berikut:

2.3.1 Cross-Validation

(CV) Cross-validation (CV) adalah teknik evaluasi model yang bertujuan untuk mendapatkan estimasi performa model yang lebih andal dan tidak bias dibandingkan dengan pembagian data pelatihan-pengujian tunggal. Pada metode *k-fold cross-validation*, dataset dibagi menjadi k lipatan (*fold*) berukuran sama. Pada setiap iterasi, satu lipatan digunakan sebagai data validasi sementara $k-1$ lipatan sisanya digunakan untuk pelatihan. Proses ini diulang sebanyak k kali sehingga setiap lipatan mendapat giliran menjadi data validasi tepat satu kali. Estimasi performa akhir diperoleh dengan merata-ratakan hasil dari semua k iterasi tersebut [29].

2.3.2 Accuracy

Accuracy adalah proporsi prediksi yang benar dari keseluruhan data yang dievaluasi. Secara matematis, accuracy dihitung sebagai:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.9}$$

di mana TP (*true positive*) adalah jumlah sampel positif yang diprediksi benar, TN (*true negative*) adalah jumlah sampel negatif yang diprediksi benar, FP (*false positive*) adalah sampel negatif yang salah diprediksi sebagai positif, dan FN (*false negative*) adalah sampel positif yang salah diprediksi sebagai negatif [29].

Meskipun mudah dipahami, accuracy dapat memberikan gambaran yang menyesatkan apabila distraibusi kelas dalam dataset tidak seimbang. Sebagai contoh, jika 95% sampel adalah kelas negatif, model yang selalu memprediksi negatif akan memperoleh accuracy 95% meskipun tidak mampu mengenali kelas positif sama sekali. Oleh karena itu, accuracy sebaiknya digunakan bersama metrik lain seperti precision, recall, dan F1-score untuk mendapatkan gambaran performa model yang lebih lengkap.

2.3.3 Precision

Precision mengukur seberapa tepat model dalam memberikan prediksi positif, yaitu dari seluruh sampel yang diprediksi positif oleh model, berapa banyak yang benar-benar positif. Precision dihitung sebagai:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{2.10}$$

Precision yang rendah berarti model banyak menghasilkan *false positive*, yaitu model terlalu sering menandai sampel negatif sebagai positif. Dalam konteks klasifikasi stadium kanker, precision yang rendah berarti banyak pasien stadium awal yang keliru diklasifikasikan sebagai stadium lanjut, yang dapat berujung pada pemberian terapi yang tidak tepat [29].

2.3.4 Recall

Recall mengukur kemampuan model dalam menemukan seluruh sampel yang benar-benar positif dari keseluruhan data positif yang ada. Recall dihitung sebagai:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{2.11}$$

Recall yang rendah berarti model banyak melewatkan sampel positif yang seharusnya terdeteksi (*false negative*). Dalam aplikasi medis, recall umumnya diprioritaskan karena konsekuensi dari tidak terdeteksinya kasus positif misalnya, kanker stadium lanjut yang tidak terdiagnosis jauh lebih serius dibandingkan

dengan *false positive*. Precision dan recall memiliki hubungan yang saling bertolak belakang misalnya; meningkatkan satu metrik cenderung menurunkan metrik yang lain, sehingga keseimbangan antara keduanya perlu dipertimbangkan sesuai konteks penggunaan model [29].

2.3.5 F1-Score

F1-score adalah rata-rata harmonik dari precision dan recall, yang memberikan satu nilai ringkasan yang mempertimbangkan keseimbangan antara kedua metrik tersebut. F1-score dihitung sebagai:

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.12}$$

Rata-rata harmonik dipilih karena lebih sensitif terhadap nilai yang rendah dibandingkan rata-rata aritmatika biasa. Artinya, F1-score yang tinggi hanya dapat dicapai jika baik precision maupun recall sama-sama tinggi; jika salah satu bernilai rendah, F1-score akan ikut turun secara signifikan. Nilai F1-score berkisar antara 0 hingga 1, di mana nilai 1 menunjukkan performa sempurna. Metrik ini sangat berguna pada dataset yang tidak seimbang di mana accuracy saja tidak cukup untuk menggambarkan kualitas model secara keseluruhan [29].

2.3.6 Area Under the Curve (AUC)

AUC (*Area Under the ROC Curve*) adalah metrik yang mengukur kemampuan model dalam membedakan antara kelas positif dan negatif di seluruh kemungkinan nilai ambang batas klasifikasi. AUC diperoleh dengan menghitung luas area di bawah kurva ROC (*Receiver Operating Characteristic*), yaitu kurva yang menggambarkan hubungan antara *True Positive Rate* (recall) pada sumbu-y dan *False Positive Rate* pada sumbu-x:

$$\text{FPR} = \frac{FP}{FP + TN} \tag{2.13}$$

Setiap titik pada kurva ROC merepresentasikan pasangan TPR dan FPR pada satu nilai ambang batas tertentu. Model yang ideal akan menghasilkan kurva yang mendekati sudut kiri atas grafik, yang bersesuaian dengan nilai AUC mendekati 1. Sebaliknya, model yang hanya menebak secara acak akan

menghasilkan kurva diagonal dengan AUC sebesar 0.5. Keunggulan AUC dibandingkan accuracy adalah kemampuannya mengevaluasi performa model secara menyeluruh tanpa bergantung pada satu nilai ambang batas tertentu, sehingga memberikan gambaran yang lebih robust mengenai kemampuan diskriminasi model, terutama pada dataset yang tidak seimbang [29].

2.3.7 Explainable Artificial Intelligence (XAI)

Explainable Artificial Intelligence (XAI) adalah bidang yang berfokus pada pengembangan metode dan teknik untuk membuat keputusan model machine learning dapat dipahami dan diinterpretasikan oleh manusia. Kebutuhan akan XAI muncul karena model seperti MLP pada dasarnya bekerja sebagai *black box*: meskipun menghasilkan prediksi yang akurat, proses internal yang menghubungkan fitur input dengan output tidak dapat diamati atau dijelaskan secara langsung. Dalam konteks aplikasi biomedis, kemampuan untuk menjelaskan mengapa suatu prediksi dihasilkan sama pentingnya dengan akurasi prediksi itu sendiri, karena hasil model perlu dapat diverifikasi oleh dokter atau peneliti sebelum digunakan dalam pengambilan keputusan klinis.

Metode XAI secara umum dibagi berdasarkan cakupan penjelasannya: metode *global* menjelaskan perilaku model secara keseluruhan (fitur mana yang paling berpengaruh secara rata-rata di seluruh data), sedangkan metode *local* menjelaskan alasan di balik satu prediksi spesifik untuk satu sampel tertentu. Dalam penelitian ini, digunakan dua metode XAI yang saling melengkapi, yaitu SHAP untuk interpretasi global maupun lokal, dan LIME untuk validasi lokal.

A SHAP (Shapley Additive Explanations)

SHAP adalah metode interpretabilitas yang berakar dari teori permainan kooperatif, khususnya konsep nilai Shapley yang pertama kali diperkenalkan oleh Lloyd Shapley. Dalam konteks machine learning, setiap fitur diperlakukan sebagai ”pemain” dalam permainan kooperatif, dan nilai SHAP mengukur kontribusi rata-rata setiap fitur terhadap prediksi model di semua kemungkinan kombinasi fitur lainnya [30].

Nilai SHAP untuk fitur i pada sampel tertentu dihitung sebagai:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [f(S \cup \{i\}) - f(S)] \tag{2.14}$$

di mana F adalah himpunan semua fitur, S adalah subset fitur yang tidak menyertakan fitur i, dan $f(S)$ adalah prediksi model menggunakan hanya fitur dalam subset S. Nilai SHAP yang positif menunjukkan bahwa fitur tersebut mendorong prediksi ke arah kelas positif, sedangkan nilai negatif menunjukkan kontribusi ke arah kelas negatif.

Keunggulan SHAP terletak pada sifat-sifat matematisnya yang terjamin, yaitu efisiensi: total nilai SHAP semua fitur sama dengan selisih prediksi dan nilai dasar, simetri: fitur dengan kontribusi sama mendapat nilai yang sama, dan konsistensi: jika kontribusi suatu fitur meningkat dalam model baru, nilai SHAP-nya tidak akan menurun.

B LIME (Local Interpretable Model-Agnostic Explanations)

LIME adalah metode interpretabilitas lokal yang menjelaskan prediksi suatu model kompleks untuk satu sampel tertentu dengan membangun model yang lebih sederhana dan mudah diinterpretasikan di sekitar sampel tersebut. Metode ini bersifat *model-agnostic*, artinya dapat diterapkan pada model apa pun tanpa perlu mengetahui struktur internalnya [30].

Proses LIME dimulai dengan menghasilkan sejumlah sampel baru di sekitar sampel yang ingin dijelaskan melalui penyimpangan fitur-fiturnya. Setiap sampel yang dihasilkan kemudian diprediksi menggunakan model asli, dan bobotnya ditentukan berdasarkan kedekatan dengan sampel asli menggunakan fungsi kernel:

$$\pi_x(z) = \exp\left(-\frac{D(x,z)^2}{\sigma^2}\right) \tag{2.15}$$

di mana $D(x,z)$ adalah jarak antara sampel asli x dengan sampel perturbasi z, dan σ adalah parameter lebar kernel. Selanjutnya, model sederhana (umumnya regresi linear) dilatih pada sampel-sampel perturbasi tersebut dengan mempertimbangkan bobot kedekatan, sehingga menghasilkan koefisien yang merepresentasikan kontribusi setiap fitur terhadap prediksi di sekitar sampel asli.

Berbeda dengan SHAP yang memberikan penjelasan yang konsisten secara global, LIME berfokus pada fidelitas lokal, yaitu penjelasan yang akurat hanya di sekitar satu titik data tertentu. Dalam penelitian ini, LIME digunakan untuk memvalidasi konsistensi fitur-fitur penting yang diidentifikasi oleh SHAP pada level prediksi individual, sehingga interpretasi model dapat dikonfirmasi dari dua perspektif yang berbeda.