BAB II
TINJAUAN PUSTAKA

2.1 Studi Literatur

Penelitian terdahulu mengenai penerapan fairness dalam machine learning, telah memberikan banyak wawasan tentang bagaimana model prediktif dapat mengandung bias yang memengaruhi keadilan dalam pengambilan keputusan. Beberapa studi menunjukkan bagaimana model machine learning yang digunakan untuk prediksi penyakit sering kali bias terhadap kelompok tertentu, seperti berdasarkan usia, jenis kelamin, atau faktor sosial ekonomi, yang mengakibatkan ketidakadilan dalam hasil prediksi. Akan tetapi penelitian sebelumnya kebanyakan hanya sampai pada prediksi kesehatan secara umum ataupun focus utamanya pada penyakit lainnya, belum ada yang spesifik kearah topik penyakit diabetes.

Bisa dilihat pada penelitian yang dilakukan oleh Simon Caton and Christian Haas (Fairness in Machine Learning: A Survey,2024) yang memetakan metode-metode untuk meningkatkan fairness secara umum dalam machine learning, namun tidak berfokus pada penyakit diabetes secara khusus. Penelitian lain, seperti Evaluating and Mitigating Bias in Machine Learning Models for Cardiovascular Disease Prediction (2023) yang dilaksanakan oleh Fuchen Li et al. Dimana mereka mengevaluasi bias pada prediksi penyakit kardiovaskular, tetapi penyakit tersebut berbeda dengan diabetes, meskipun tetap berhubungan dengan kesehatan kronis. Studi Fair Machine Learning in Healthcare: A Survey (2024) oleh Qizhang Feng et al. Memang mengkaji fairness dalam konteks kesehatan secara lebih luas, namun belum fokus pada diabetes. Begitu juga dengan penelitian yang dilakukan oleh Pablo Mosteiro et al. Dalam penelitian yang berjudul Bias Discovery in Machine Learning Models for Mental Health (2022) yang menganalisis bias pada model kesehatan mental.

2.2 Diabetes

Diabetes adalah salah satu penyakit kronis dengan tingkat prevalensi yang terus meningkat secara global. Menurut **Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045 : Results from the International Diabetes Federation Diabetes Atlas, 9th edition** jumlah penderita diabetes diperkirakan mencapai 463 juta pada tahun 2019 dan diprediksi akan meningkat menjadi 578 juta pada tahun 2030 dan 700 juta pada 2045[2]. Lonjakan jumlah kasus ini menunjukkan bahwa diabetes bukan hanya menjadi masalah kesehatan individu, tetapi juga telah berkembang menjadi isu global yang membutuhkan perhatian serius dari berbagai pihak. Pertumbuhan angka penderita ini berdampak luas terhadap sistem kesehatan, beban ekonomi, serta kualitas hidup masyarakat di banyak negara[4]. Tren ini mencerminkan ancaman serius bagi kesehatan masyarakat global dan menekankan pentingnya upaya yang lebih intensif dalam pencegahan, pengelolaan, serta penanganan penyakit diabetes secara menyeluruh.

2.3 Pre Processing

Tahapan Pre Processing merupakan salah satu tahap yang penting dalam mempersiapkan dataset agar model machine learning dapat mengolahnya dengan optimal, hal ini berkontribusi dalam meminimalkan terjadinya kesalahan serta meningkatkan akurasi model machine learning[15], [16]. Tanpa dilakukan preprocessing, data mentah dapat menyebabkan model machine learning gagal dilatih, menghasilkan prediksi yang tidak akurat, serta memunculkan kesalahan analisis yang berdampak pada hasil. Pre processing kali ini meliputi beberapa langkah sebagai berikut :

2.3.1 Pengecekan Nilai Hilang

Langkah ini diperlukan guna memeriksa apakah terdapat data yang hilang atau kosong (missing values) dalam dataset. Data yang hilang dapat mengganggu kinerja model, sehingga biasanya diisi dengan nilai tertentu (imputasi), misalnya nilai rata-rata, median,

atau metode lain sesuai konteks data. Dalam beberapa kasus, baris atau kolom yang memiliki terlalu banyak nilai hilang mungkin akan dihapus[17], [18].

2.3.2 Encoding Data

Encoding merupakan salah satu proses tranformasi data menjadi format numerik agar dapat digunakan oleh model machine learning. Terdapat beberapa teknik encoding, seperti One-Hot Encoding yang mengubah setiap kategori menjadi kolom biner, dan Label Encoding yang mengubah kategori menjadi angka unik. Pemilihan metode tergantung pada tipe data kategori yang digunakan[17], [18].

2.3.3 Normalisasi Data

Normalisasi membantu menempatkan data numerik dalam skala yang seragam, umumnya antara 0 dan 1. Hal ini penting karena beberapa algoritma machine learning sensitif terhadap perbedaan skala, sehingga normalisasi membuat pelatihan model lebih efektif. Pada kali ini teknik normalisasi yang digunakan adalah standart scaler[17], [18].

2.3.4 Split data

Pembagian data atau split data adalah proses memisahkan dataset menjadi yang umumnya menjadi dua atau tiga bagian. Pada penelitian ini data dipisahkan menjadi dua yaitu data pelatihan (training set) dan data pengujian (testing set). Data pelatihan digunakan untuk melatih model, sementara data pengujian berfungsi untuk mengevaluasi kinerja model pada data yang belum pernah dilihat model sebelumnya. Proporsi yang umum digunakan adalah 70:30 untuk pelatihan dan pengujian[19].

2.4 Machine Learning

Machine learning (ML) merupakan salah satu bidang dalam kecerdasan buatan (artificial intelligence) yang menitikberatkan pada pembuatan algoritma dan model yang memungkinkan komputer belajar dari data serta membuat prediksi atau keputusan. Konsep dasar dari machine learning berakar pada ide bahwa sistem dapat dioptimalkan melalui pengalaman, mirip dengan cara manusia belajar dari pengalaman mereka[20]. Machine learning sendiri terdiri dari beberapa pendekatan, termasuk supervised learning, unsupervised learning, dan reinforcement learning[21]. Pada supervised learning, pelatihan model dilakukan dengan menggunakan data yang sudah memiliki label, sehingga dapat memprediksi output berdasarkan input yang diberikan, sedangkan unsupervised sebaliknya[22].

Pada penelitian ini, digunakan algoritma tree-based seperti Decision Tree dan Random Forest serta algoritma probabilistik biner seperti Logistic Regression dan Naive Bayes. Naive Bayes dan Logistic Regression memiliki waktu komputasi yang ringan, menjadikannya efisien untuk pengolahan dataset yang besar, sedangkan outputnya sangat berguna dalam konteks klinis untuk menilai probabilitas risiko penyakit[23], [24]. Selain itu, algoritma tree-based mampu menangkap hubungan non-linear dan interaksi antar fitur yang kompleks. Decision Tree menyajikan visualisasi aturan keputusan yang intuitif serta interpretasi yang tinggi[25]. Sedangkan Random Forest dengan pendekatan ensembelnya, membantu meningkatkan kestabilan hasil prediksi serta menurunkan kemungkinan terjadinya overfitting yang kerap dialami oleh model decision tree tunggal, sehingga sangat efektif untuk menangani data dalam jumlah besar[26]. Dukungan dari berbagai pustaka machine learning membuat keempat algoritma ini mudah diterapkan dan dioptimalkan. Kombinasi ini dipilih karena mampu memberikan keseimbangan yang baik antara akurasi, kemudahan pemahaman, dan efisiensi dalam proses komputasi.

2.4.1 Decision Tree

Decision Trees (DT) adalah model yang berbentuk pohon, di mana keputusan diambil berdasarkan serangkaian pertanyaan mengenai fitur input. Algoritma ini bersifat non-parametrik, artinya tidak membuat asumsi tentang distribusi data, sehingga dapat digunakan pada data yang tidak terdistribusi normal. Selain itu, decision tree memiliki representasi grafis yang intuitif, memudahkan pemahaman dan interpretasi hasil[27].

Decision Trees (DT) memiliki kelebihan utama dalam kemudahan pemahaman dan interpretasi, karena modelnya yang berbentuk pohon membuat keputusan dapat dilacak dengan jelas. Selain itu, decision tree dapat menangani data numerik dan kategorikal tanpa memerlukan transformasi, serta berfungsi baik dengan dataset yang besar. Namun, kelemahan utama dari algoritma ini adalah kecenderungannya untuk overfit pada data pelatihan, terutama ketika pohon terlalu dalam dan kompleks, yang dapat mengurangi kemampuan generalisasi pada data baru[28].

2.4.2 Random Forest

Random Forest (RF) adalah algoritma ensemble yang bekerja dengan membentuk sejumlah pohon keputusan secara paralel, lalu mengombinasikan output dari masing-masing pohon untuk meningkatkan ketepatan prediksi. Dengan pendekatan ini, random forest dapat mengurangi varians serta menurunkan kemungkinan terjadinya overfitting yang kerap dialami oleh model decision tree tunggal. Algoritma ini juga dikenal karena kemampuannya dalam menangani data yang hilang dengan baik[29].

Random Forest (RF), di sisi lain, mengatasi masalah overfitting dengan membangun ensemble dari beberapa decision trees, sehingga meningkatkan akurasi dan stabilitas model. Kelebihan lain dari random forest adalah kemampuannya untuk menangani data

yang hilang dan mengukur pentingnya fitur. Namun, kelemahannya adalah model yang lebih kompleks dan kurang interpretatif dibandingkan dengan decision tree tunggal, serta membutuhkan lebih banyak waktu untuk pelatihan dan prediksi[30].

2.4.3 Logistic Regresion

Logistic Regression (LR) adalah model linier yang digunakan untuk memprediksi probabilitas kejadian dua kelas (biner). Meskipun dinamakan regresi, algoritma ini merupakan metode klasifikasi yang menggunakan fungsi sigmoid untuk mengubah output menjadi nilai probabilitas antara 0 dan 1 [31].

Logistic Regression (LR) memiliki kelebihan dalam kesederhanaan dan kecepatan pelatihan, serta interpretasi hasil yang mudah, menjadikannya pilihan yang baik untuk masalah klasifikasi biner dengan data yang terdistribusi normal. Namun, kelemahan dari algoritma ini adalah keterbatasan dalam menangkap hubungan non-linear antara fitur dan target, sehingga kurang efektif pada dataset yang kompleks dan non-linear tanpa transformasi atau teknik lain yang mendukung[32].

2.4.4 Naive Bayes

Naïve Bayes merupakan algoritma klasifikasi yang didasarkan pada pendekatan probabilistik dengan menerapkan Teorema Bayes, di mana diasumsikan bahwa setiap fitur bersifat independen. Algoritma ini menentukan probabilitas masing-masing kelas berdasarkan fitur yang ada, lalu memilih kelas dengan probabilitas tertinggi sebagai hasil prediksi[33]. Karena asumsinya yang "naïf" atau sederhana yaitu setiap fitur dianggap tidak saling bergantung, algoritma ini sangat efisien dalam menghitung probabilitas meskipun fitur-fitur dalam data sebenarnya bisa jadi saling berhubungan.

Kelebihan utama Naïve Bayes adalah cepat, efisien, dan bekerja dengan baik meskipun data memiliki jumlah fitur yang besar, algoritma ini juga tetap memberikan performa yang baik meskipun jumlah data pelatihan terbatas[24]. Meski demikian, kelemahan utama dari algoritma ini terletak pada asumsi bahwa fitur-fitur bersifat independen, yang dalam praktiknya sering kali tidak sesuai dengan kondisi data di dunia nyata, yang dapat menyebabkan penurunan akurasi. Selain itu, jika suatu kategori tidak muncul dalam data pelatihan (zero probability problem), maka model bisa gagal memberikan prediksi yang benar tanpa teknik smoothing seperti Laplace Smoothing.

2.5 Dalex

Penelitian ini menggunakan library yang mana merupakan sebuah library untuk membantu peneliti dan praktisi dalam menganalisis model machine learning dengan fokus pada keadilan. DALEX, singkatan dari *Model Agnostic Language for Exploration and Explanation* adalah sebuah kerangka kerja yang dirancang untuk membantu analisis dan interpretasi model machine learning tanpa bergantung pada jenis atau struktur model tertentu. Salah satu tahap utama dalam penggunaan library Dalex adalah membangun explainer, yaitu objek yang merepresentasikan model beserta data latihannya, sehingga memungkinkan analisis lebih lanjut terkait perilaku atau keadilan model[34].

Dalam konteks deteksi bias, DALEX menerapkan prinsip aturan empat perlima, sebuah standar umum yang digunakan untuk mengevaluasi tingkat diskriminasi dalam model[35]. Aturan ini diterapkan dengan nilai ambang batas ε yang secara default ditetapkan sebesar 0,8, meskipun pengguna dapat menyesuaikannya sesuai dengan kebutuhan analisis. Rasio skor metrik yang mendekati angka 1 mengindikasikan bahwa model memiliki tingkat keadilan yang tinggi terhadap kelompok-kelompok yang dibandingkan. Untuk mempermudah interpretasi, skor dari masing-masing metrik bias dihitung menggunakan rumus berikut :

$$\forall_{i\in\{a,b,\ldots,z\}^{\varepsilon}} < \frac{metric_i}{metric_{previledged}} < \frac{1}{\varepsilon} \quad (1)$$

Melalui analisis yang dilakukan dengan Dalex, pengguna dapat mengevaluasi berbagai metrik keadilan, seperti disparitas dalam prediksi antara kelompok yang berbeda. Selain itu, Dalex juga memungkinkan analisis sensitivitas dan interpretabilitas model, sehingga pengguna dapat memahami faktor-faktor mana yang paling memengaruhi keputusan model. Dengan pendekatan ini, Dalex membantu untuk lebih memahami masalah keadilan dalam machine learning dan memberikan solusi yang lebih adil dan inklusif dalam pengembangan model prediktif.

2.6 Machine Learning Fairness

Machine Learning Fairness mengacu pada usaha untuk menjamin bahwa model machine learning menghasilkan keputusan yang tidak hanya tepat, tetapi juga mempertimbangkan aspek keadilan[36]. Dalam hal ini, keadilan mengacu pada prinsip bahwa model tidak mendiskriminasi kelompok tertentu, baik secara langsung maupun tidak langsung, berdasarkan atribut sensitif seperti jenis kelamin, ras, umur, atau faktor genetik lainnya[37]. Ketidakadilan dalam model machine learning dapat menghasilkan keputusan yang merugikan kelompok tertentu dan memperkuat ketidaksetaraan yang ada dalam masyarakat[38].

Dalam penelitian fairness membutuhkan atribut yang dilindungi atau biasa di sebut protected attribute, yang diperoleh dari kombinasi beberapa atribut yang berpotensi menimbulkan sebuah ketidakadilan. Penelitian menggunakan kombinasi dari atribut usia dan jenis kelamin, dikarenakan kedua atribut tersebut berpotensi menjadi sumber bias dalam prediksi model. Protected attribute adalah variabel yang merepresentasikan karakteristik sensitif individu yang harus dijaga agar tidak menjadi dasar diskriminasi dalam pengambilan keputusan model. Dalam konteks fairness, model harus diupayakan untuk memperlakukan semua kelompok pada protected attributes tersebut secara adil.

Selain itu, untuk keperluan analisis fairness, diperlukan juga privileged attribute, yaitu kelompok dalam protected attributes yang secara historis atau statistik memiliki keuntungan atau akses lebih besar terhadap hasil positif. Privileged atribut pada sebuah penelitian fairness umumnya di pilih secara acak atau pun bisa di coba satu persatu guna analisis lebih lanjut, akan tetapi pada penelitian ini previledge ditentukan dengan memilih kelompok dengan jumlah terbanyak.

2.7 Evaluasi

tahap evaluasi sangat penting untuk memastikan bahwa model yang dibangun tidak hanya akurat, tetapi juga adil dan tidak memihak. Evaluasi ini juga membantu dalam memahami dampak model pada berbagai kelompok, memastikan bahwa prediksi yang dihasilkan bersifat inklusif, dan mendukung keputusan yang lebih adil dalam penerapan model di dunia nyata.

2.7.1 Evaluasi Model

Penelitian ini menggunakan classification report dengan metrik accuracy, precission, recall, f1-score dan AUC guna mengevaluasi model. Classification report memberikan gambaran komprehensif tentang ketepatan dan kekomprehensifan model dalam mengklasifikasikan data.

- *Accuracy*

  Accuracy mengukur proporsi jumlah prediksi yang benar (baik positif maupun negatif) terhadap keseluruhan data yang diuji. Ini adalah metrik yang paling umum digunakan, karena mudah dipahami dan memberikan gambaran umum kinerja model. Berikut rumus untuk menghitung accuracy :

$$accuracy = \frac{TP + TN}{(TP + TN + FP + FN)} \quad (2)$$

- *Precission*

  Precision mengukur proporsi dari prediksi positif yang benar-benar merupakan kelas positif. Ini menunjukkan keandalan model dalam memberikan label positif. Berikut rumus untuk menghitung precission :

$$precision = \frac{TP}{(TP + FP)} \quad (3)$$

- Recall

  Recall mengukur seberapa baik model mendeteksi seluruh kasus positif yang benar-benar ada dalam data. Berikut rumus untuk menghitung recall :

$$recall = \frac{TP}{(TP + FN)} \quad (4)$$

- *F1-Score*

  F1-Score merupakan rata-rata harmonik dari precision dan recall, yang berguna ketika diperlukan keseimbangan antara keduanya, terutama saat data tidak seimbang. Berikut rumus untuk menghitung F1-Score :

$$F1\ score\ = 2 \times \frac{precision \times recall}{(precision\ +\ recall)} \quad (5)$$

- *AUC*

  AUC menunjukkan kemampuan model untuk membedakan antara kelas positif dan negatif berdasarkan nilai probabilitas yang dihasilkan model. Berikut rumus untuk menghitung AUC :

$$AUC = \int_0^1 TPR(FPR)\, d(FPR) \quad (6)$$

2.7.2 Deteksi Bias

Penelitian ini menggunakan metrik fairness seperti True Positive Rate (TPR), Accuracy, Positive Predictive Value (PPV), False Positive Rate (FPR) , dan Statistical Parity (STP) guna mengevaluasi aspek kinerja dan keadilan model. Metrik-metrik ini memberikan gambaran menyeluruh tentang kemampuan model dalam mendeteksi kasus positif secara akurat serta memeriksa potensi bias antar kelompok.

- TPR

  TPR mengukur seberapa sering model berhasil mengenali kasus positif secara benar dalam suatu kelompok. Ketidakseimbangan TPR antar kelompok menunjukkan bahwa model lebih baik dalam mendeteksi kasus positif pada satu kelompok dibanding yang lain, yang bisa menyebabkan ketidakadilan. Berikut rumus untuk menghitung TPR :

$$TPR_G = \frac{TP_G}{TP_G + FN_G} \tag{7}$$

  TPR Ratio adalah membandingkan TPR grup lain terhadap grup referensi. Berikut rumus untuk menghitung TPR Ratio:

$$TPR_{Ratio} = \frac{TPR_{G(Protected)}}{TPR_{G(Previledge)}} \tag{8}$$

  Nilai rasio TPR tersebut kemudian dianalisis untuk mengevaluasi tingkat fairness antar kelompok. Interpretasi dilakukan dengan membandingkan nilai rasio terhadap rentang nilai tertentu yang umum digunakan dalam literatur fairness machine learning. Rentang ini digunakan untuk mengidentifikasi apakah terdapat ketimpangan perlakuan terhadap kelompok protected dibandingkan kelompok privilege.

Tabel 2.1 Interpretasi Nilai Fairness Berdasarkan TPR Ratio menyajikan kondisi fairness berdasarkan nilai TPR Ratio yang dihasilkan.

Tabel 2.1 Interpretasi Nilai Fairness Berdasarkan TPR Ratio

Rentang Rasio	Kondisi	Interpretasi
< 0,8	Bias Negatif	Kelompok kurang terdeteksi sebagai positif meskipun seharusnya (underdiagnosed)
0,8 – 1,25	Fair	Kemampuan model mendeteksi kasus positif relatif seimbang
> 1,25	Bias Positif	Kelompok terlalu sering terdeteksi positif (overdiagnosed)

- ACC

ACC menunjukkan proporsi prediksi yang benar dari total kasus dalam kelompok. Perbedaan akurasi antar kelompok dapat menunjukkan bahwa model lebih efektif di satu kelompok, sehingga menimbulkan potensi bias dalam performa keseluruhan. Berikut rumus untuk menghitung ACC :

$$ACC_G = \frac{TP_G + TN_G}{TP_G + TN_G + FP_G + FN_G} \quad (9)$$

ACC Ratio adalah membandingkan ACC grup lain terhadap grup referensi. Berikut rumus untuk menghitung ACC Ratio:

$$ACC_{Ratio} = \frac{ACC_{G(Protected)}}{ACC_{G(Previledge)}} \quad (10)$$

Nilai rasio ACC digunakan untuk menilai apakah akurasi prediksi model bersifat adil antar kelompok. Analisis dilakukan dengan membandingkan akurasi kelompok protected terhadap

kelompok privilege. Rentang nilai rasio yang digunakan merepresentasikan tingkat kesetaraan performa model pada setiap kelompok. Tabel 2.2 Interpretasi Nilai Fairness Berdasarkan ACC Ratio menggambarkan kondisi fairness berdasarkan nilai ACC Ratio yang diperoleh.

Tabel 2.2 Interpretasi Nilai Fairness Berdasarkan ACC Ratio

Rentang Rasio	Kondisi	Interpretasi
< 0,8	Bias Negatif	Akurasi jauh lebih rendah dibanding kelompok lain (berisiko salah prediksi)
0,8 – 1,25	Fair	Akurasi prediksi seimbang antar kelompok
> 1,25	Bias Positif	Kelompok memiliki keakuratan berlebih (overfitting terhadap kelompok ini)

- PPV

PPV menunjukkan seberapa akurat prediksi positif model pada masing-masing kelompok. Jika PPV tidak seimbang, maka satu kelompok lebih sering menerima prediksi positif yang salah, yang bisa berdampak negatif dalam pengambilan keputusan. Berikut rumus untuk menghitung PPV :

$$PPV_G = \frac{TP_G}{TP_G + FP_G} \tag{11}$$

PPV Ratio adalah membandingkan PPV grup lain terhadap grup referensi. Berikut rumus untuk menghitung PPV Ratio:

$$PPV_{Ratio} = \frac{PPV_{G(Protected)}}{PPV_{G(Previledge)}} \tag{12}$$

Rasio PPV digunakan untuk mengukur kesetaraan dalam kualitas prediksi positif yang diberikan model antar kelompok. Nilai ini mencerminkan apakah model memberikan prediksi

positif yang sama akuratnya pada kelompok protected dan privilege. Tabel 2.3 Interpretasi Nilai Fairness Berdasarkan PPV Ratio menjelaskan interpretasi nilai rasio PPV berdasarkan batasan rentang tertentu yang umum digunakan dalam evaluasi fairness.

Tabel 2.3 Interpretasi Nilai Fairness Berdasarkan PPV Ratio

Rentang Rasio	Kondisi	Interpretasi
< 0,8	Bias Negatif	Prediksi positif kelompok sering keliru (false positive tinggi)
0,8 – 1,25	Fair	Tingkat kebenaran prediksi positif seimbang
> 1,25	Bias Positif	Prediksi positif kelompok sangat tepat (kemungkinan model lebih percaya terhadap kelompok ini)

- FPR

FPR mengukur seberapa sering model salah memberikan prediksi positif pada kasus yang sebenarnya negatif. Ketimpangan FPR antar kelompok dapat menyebabkan satu kelompok lebih sering dirugikan karena kesalahan sistem. Berikut rumus untuk menghitung FPR :

$$FPR_G = \frac{FP_G}{FP_G + TN_G} \tag{13}$$

FPR Ratio adalah membandingkan FPR grup lain terhadap grup referensi. Berikut rumus untuk menghitung FPR Ratio:

$$FPR_{Ratio} = \frac{FPR_{G(Protected)}}{FPR_{G(Previledge)}} \tag{14}$$

Rasio FPR digunakan untuk mengevaluasi sejauh mana model memberikan kesalahan prediksi positif (false positive) secara merata pada tiap kelompok. Ketimpangan nilai FPR dapat

menunjukkan bahwa suatu kelompok lebih sering dirugikan akibat prediksi yang salah. Interpretasi fairness dari nilai rasio FPR disajikan dalam Tabel 2.4 Interpretasi Nilai Fairness Berdasarkan FPR Ratio dengan mengacu pada rentang nilai yang telah ditetapkan.

Tabel 2.4 Interpretasi Nilai Fairness Berdasarkan FPR Ratio

Rentang Rasio	Kondisi	Interpretasi
< 0,8	Bias Positif	Kelompok lebih jarang dirugikan oleh false positive
0,8 – 1,25	Fair	Tingkat kesalahan false positive relatif seimbang
> 1,25	Bias Negatif	Kelompok lebih sering salah diprediksi positif (false alarm)

- STP

  STP mengukur seberapa besar peluang setiap kelompok mendapatkan prediksi positif, tanpa memperhatikan kebenaran label. Jika peluang ini tidak seimbang, maka akses atau hasil yang diberikan model cenderung tidak adil. Berikut rumus untuk menghitung STP :

$$STP_G = P(\hat{Y} = 1 \mid A = a) \tag{15}$$

STP Ratio adalah membandingkan STP grup lain terhadap grup referensi. Berikut rumus untuk menghitung STP Ratio:

$$STP_{Ratio} = \frac{STP_{G(Protected)}}{STP_{G(Previledge)}} \tag{16}$$

Rasio STP dimanfaatkan untuk menilai kesetaraan distribusi prediksi positif antar kelompok tanpa memperhatikan kebenaran label sebenarnya. Nilai ini mencerminkan apakah setiap kelompok mendapatkan peluang yang seimbang dalam

menerima keputusan positif dari model. Untuk menilai fairness, nilai rasio STP dibandingkan terhadap rentang nilai tertentu sebagaimana ditampilkan pada Tabel 2.5 Interpretasi Nilai Fairness Berdasarkan STP Ratio.

Tabel 2.5 Interpretasi Nilai Fairness Berdasarkan STP Ratio

Rentang Rasio	**Kondisi**	**Interpretasi**
< 0,8	Bias Negatif	Kelompok kurang mendapatkan prediksi positif (diskriminasi peluang)
0,8 – 1,25	Fair	Proporsi prediksi positif seimbang antar kelompok
> 1,25	Bias Positif	Kelompok terlalu sering mendapat hasil positif, bisa berlebihan atau tidak proporsional