

Algoritma Multilayer Perceptron untuk Klasifikasi Anak Putus Sekolah di Jawa Barat Tahun 2024

Aditya Haryanto Ramdhani *, Ilham Faishal Mahdy

Prodi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Islam Bandung, Indonesia.

adityaharyanto3@gmail.com, ilhamfaishalmahdy@unisba.ac.id

Abstract. The phenomenon of school dropouts among adolescents is one of the most serious challenges in the education sector in Indonesia, particularly in West Java Province, which had the highest rate of school dropouts in Indonesia in 2024. To identify children at risk of dropping out of school, this study developed a classification model using the Multilayer Perceptron (MLP) algorithm. The predictor variables used in this study include age, number of household members, gender, marital status, highest education level of the head of household, residential area, and activity status. This study employs data preprocessing, application of the MLP algorithm using the backpropagation method, and model evaluation using accuracy, precision, recall, and F1-score. The evaluation results of the multilayer perceptron model show an accuracy of 81%, precision in the school dropout class of 28%, recall in the school dropout class of 87%, and an F1-score of 42%. These values indicate that the model is capable of identifying cases of children at risk of dropping out of school, although it is not yet sufficiently accurate in determining who falls into that category. The results of this study are expected to serve as a reference for authorized institutions to undertake preventive measures against school dropout cases using an effective prediction model, as well as a reference for further research in the field of social classification using machine learning.

Keywords: *Machine Learning, Multilayer Perceptron, Dropout.*

Abstrak. Fenomena putus sekolah pada kalangan remaja adalah salah satu tantangan yang sangat serius dalam sektor pendidikan di Indonesia, khususnya di Provinsi Jawa Barat yang menduduki kasus putus sekolah tertinggi di Indonesia pada tahun 2024. Untuk mengidentifikasi anak-anak yang berisiko putus sekolah, penelitian ini mengembangkan model klasifikasi menggunakan algoritma *Multilayer Perceptron* (MLP). Variabel prediktor yang digunakan pada penelitian ini meliputi usia, jumlah anggota rumah tangga, jenis kelamin, status pernikahan, pendidikan tertinggi kepala keluarga wilayah tempat tinggal, dan status kegiatan. Penelitian ini menggunakan tahapan *data preprocessing*, penerapan algoritma MLP dengan metode *backpropagation*, serta evaluasi model menggunakan *accuracy*, *precision*, *recall*, dan *f1-score*. Hasil evaluasi pada model *multilayer perceptron* yang dibuat menunjukkan *Accuracy* sebesar 81%, *Precision* pada kelas putus sekolah sebesar 28%, *Recall* pada kelas putus sekolah sebesar 87%, dan *F1-Score* sebesar 42%. Nilai-nilai tersebut menunjukkan bahwa model mampu mengenali kasus anak yang berisiko putus sekolah, meskipun belum cukup tepat dalam menentukan siapa yang termasuk dalam kategori tersebut. Hasil penelitian ini diharapkan menjadi referensi bagi lembaga yang berwenang untuk melakukan upaya pencegahan terjadinya kasus putus sekolah dengan model prediksi yang efektif serta menjadi referensi untuk pengembangan penelitian selanjutnya dalam bidang klasifikasi sosial menggunakan *machine learning*.

Kata Kunci: *Machine Learning, Multilayer Perceptron, Putus Sekolah.*

A. Pendahuluan

Pendidikan adalah salah satu aspek yang sangat penting bagi pembangunan bangsa, yaitu sebagai karakter bangsa dan juga sebagai wadah untuk pembentukan generasi yang dibutuhkan di masa mendatang, sehingga hampir semua bangsa menempatkan pendidikan sebagai prioritas utama dalam program pembangunan nasional. Manusia dapat memperoleh pendidikan dari berbagai sarana yang berbeda, seperti keluarga, pendidikan formal (sekolah), dan pendidikan informal (Zamhari et al., 2023). Dengan alasan tersebut, idealnya setiap anak mengikuti pendidikan formal (sekolah) agar menjadi lebih matang secara kognitif, afektif, maupun motorik. Namun, pada kenyataannya tidak setiap anak mempunyai kesempatan untuk berpartisipasi dalam pendidikan di sekolah.

Jawa Barat merupakan salah satu provinsi dengan jumlah penduduk terbesar di Indonesia. Berdasarkan data (Kemendikdasmen, 2024), angka putus sekolah di Jawa Barat adalah yang tertinggi di Indonesia, dengan jumlah kasus sebanyak 5.080. Faktor yang mempengaruhi tingginya angka putus sekolah adalah kondisi ekonomi, kondisi sosial, lingkungan keluarga, dan akses terhadap fasilitas pendidikan (Pandu et al., 2022; Purba, 2023; Sari & Karliani, 2023; Yaneri et al., 2022). Anak-anak yang putus sekolah cenderung tidak memperoleh pekerjaan yang layak, sangat berisiko mengalami kemiskinan, dan dapat terjerumus kedalam berbagai permasalahan sosial, seperti pernikahan dini dan kriminalitas. Oleh karena itu, upaya untuk memahami dan mengatasi faktor-faktor yang menyebabkan anak putus sekolah menjadi sangat penting dalam rangka meningkatkan kualitas pendidikan di Indonesia (Soetrisnaadisendjaja & Sari, 2019).

Seiring dengan perkembangan teknologi, berbagai metode berbasis kecerdasan buatan (*Artificial Intelligence*) telah digunakan dalam bidang pendidikan untuk menganalisis dan memprediksi berbagai fenomena, termasuk putus sekolah. Salah satu metode yang dapat digunakan adalah jaringan saraf tiruan (*Artificial Neural Network*) (Rumelhart et al., 1986). Pada awalnya, jaringan saraf hanya terdiri dari satu lapisan saja yang dikenal dengan istilah *perceptron* yang diperkenalkan oleh (Rosenblatt, 1958). Namun, seiring perkembangan zaman dan penyelesaian masalah yang lebih kompleks dan tidak linear, hal ini menyebabkan model *perceptron* tunggal memiliki keterbatasan, terutama dalam menyelesaikan permasalahan yang tidak dapat dipisahkan secara linear, seperti kasus XOR pada gerbang logika. Berikut adalah gambar yang menggambarkan setiap gerbang logika (Heaton, 2005).

AND



INPUT		OUTPUT
A	B	
0	0	0
0	1	0
1	0	0
1	1	1

OR



INPUT		OUTPUT
A	B	
0	0	0
0	1	1
1	0	1
1	1	1

XOR



INPUT		OUTPUT
A	B	
0	0	0
0	1	1
1	0	1
1	1	0

NAND



INPUT		OUTPUT
A	B	
0	0	1
0	1	1
1	0	1
1	1	0

NOR



INPUT		OUTPUT
A	B	
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



INPUT		OUTPUT
A	B	
0	0	1
0	1	0
1	0	0
1	1	1

NOT



INPUT	OUTPUT
A	
0	1
1	0

Gambar 1. Gerbang Logika (Mikrodata.co.id, 2020)

Pada gerbang logika OR, AND, NOR, dan NAND adalah gambaran dari klasifikasi linear, sedangkan pada gerbang logika XOR dan XNOR adalah gambaran dari klasifikasi non-linear, selanjutnya pada Gambar 1.2 menjelaskan tentang bentuk visual dari klasifikasi linear dan klasifikasi non-linear.



Gambar 2. Klasifikasi Linear dan Klasifikasi Non-Linear

Dalam klasifikasi kasus anak putus sekolah, variabel-variabel yang mempengaruhi risiko anak putus sekolah kemungkinan besar lebih kompleks sehingga pemilihan metode *Multilayer Perceptron* (MLP) dirasa lebih cocok dibandingkan *perceptron* tunggal. MLP adalah jenis saraf tiruan yang terdiri dari tiga lapisan utama, yaitu *input layer*, *hidden layer* (bisa lebih dari satu), dan *output layer*.

Penambahan *hidden layer* memungkinkan MLP untuk melakukan pemetaan non-linear yang jauh lebih kompleks dibandingkan dengan *perceptron* tunggal.

Pemilihan *Multilayer Perceptron* untuk klasifikasi anak putus sekolah merupakan pilihan yang tepat dibandingkan dengan metode ANN yang lain seperti *Convolutional Neural Network* (CNN) atau *Recurrent Neural Network* (RNN), karena CNN umumnya dirancang untuk memproses data *grid* seperti gambar dan RNN umumnya dirancang untuk memproses data sekuensial.

Penerapan algoritma *backpropagation* oleh (Rumelhart et al., 1986), menjadi pondasi penting dalam pengembangan metode MLP, karena algoritma *backpropagation* memungkinkan jaringan untuk melakukan pembelajaran melalui proses umpan balik yang disebut *backward*, dengan cara memperbaharui bobot berdasarkan *error output*. Dengan alasan tersebut, algoritma MLP menjadi salah satu metode pembelajaran mesin yang paling umum digunakan hingga saat ini (Ulfa, 2021).

Penelitian terdahulu adalah kajian literatur seberapa baik performa dari *Multilayer Perceptron* (MLP). Dari kajian literatur tersebut didapat 30 artikel penelitian yang mengangkat masalah prediksi dan klasifikasi menggunakan algoritma *Multilayer Perceptron*, dan diperoleh hasil rata-rata performa adalah 91,98%. Hal tersebut menunjukkan bahwa algoritma MLP cukup baik untuk digunakan dalam penelitian yang berhubungan dengan masalah prediksi maupun klasifikasi (Pardede & Herawan Hayadi, 2022; Rindri & Fitriyani, 2023).

B. Metode

Pada penelitian ini, data yang digunakan merupakan data sekunder hasil Survei Sosial Ekonomi Nasional (SUSENAS) Provinsi Jawa Barat Bulan Maret Tahun 2024, dengan 17.729 observasi dengan variabel atau karakteristik yang mempengaruhinya. Penelitian ini menggunakan 1 (satu) variabel target dan 7 (tujuh) variabel prediktor.

Tabel 1. Data Penelitian

No	Usia	Jumlah ART	Jenis Kelamin	Status Pernikahan	Pendidikan Tertinggi KRT	Wilayah	Kegiatan	Status Sekolah
1	15	3	Laki-Laki	Tidak Menikah	Dasar	Pedesaan	Tidak Bekerja	Sekolah
2	13	4	Laki-Laki	Tidak Menikah	Menengah	Pedesaan	Tidak Bekerja	Sekolah
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
17728	16	3	Laki-Laki	Tidak Menikah	Dasar	Perkotaan	Tidak Bekerja	Putus Sekolah
17729	9	3	Perempuan	Tidak Menikah	Dasar	Perkotaan	Tidak Bekerja	Sekolah

Sebelum melakukan setiap tahapan pada algoritma MLP, data akan dinormalisasi terlebih dahulu. Namun, variabel yang bersifat biner tidak perlu diubah menggunakan persamaan 2.1.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad \dots (2.1)$$

Tahapan *Feedforward*

Tahap *Feedforward* dimulai dengan proses penerimaan sinyal oleh setiap *input layer* dan meneruskannya ke *hidden layer*. Kemudian, lakukan penjumlahan pada bobot dari sinyal *input* (z_{inj}) yang dinyatakan dengan rumus sebagai berikut (Haykin, 1999; Rana et al., 2018).

$$z_{net} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad \dots (2.2)$$

Selanjutnya, penelitian ini menggunakan fungsi aktivasi *sigmoid* yang berfungsi untuk menghitung sinyal *output* yang dikirim oleh *hidden layer* (z_j) menggunakan persamaan (2.3).

$$z_j = \frac{1}{1 + e^{-z_{net}}} \quad \dots (2.3)$$

Setelah melalui perhitungan dengan fungsi aktivasi *sigmoid* pada *neuron* di *hidden layer*, lalu akan dilakukan perhitungan seluruh bobot sinyal pada jaringan *output layer* (y_{net}) dengan menggunakan persamaan (2.4).

$$y_{net} = w_{0k} + \sum_{k=1}^n z_j w_{jk} \quad \dots (2.4)$$

Langkah terakhir untuk tahap *feedforward* adalah menghitung nilai *output* dari *neuron* di *output layer* setelah melalui aktivasi *sigmoid* (y_k), dengan menggunakan persamaan (2.5).

$$y_k = \frac{1}{1 + e^{-y_{net}}} \quad \dots (2.5)$$

Dikarenakan pada penelitian ini memiliki kemungkinan ketidakseimbangan, maka digunakan teknik pembobotan kelas dengan persamaan (2.6) (Hairani et al., 2020).

$$w_c = \frac{N}{n_c \cdot C} \quad \dots (2.6)$$

Tahapan *Backward*

Tahap *backward* dimulai dengan menentukan nilai *error* (δ_k) atau kesalahan di *output layer* dengan persamaan (2.7), yang dilanjutkan dengan melakukan perhitungan perubahan bobot (Δw_{jk}) dari *neuron* di *hidden layer* ke *neuron* di *output layer* menggunakan persamaan (2.8).

$$\delta_k = w_c(t_k - y_k)y_k(1 - y_k) \quad \dots (2.7)$$

$$\Delta w_{jk} = \alpha \delta_k z_j \quad \dots (2.8)$$

Selanjutnya dilakukan hal yang sama pada *hidden layer* yakni menghitung *error* (δ_j) dengan persamaan (2.9). Lalu, diikuti dengan menghitung perubahan bobot (Δv_{ij}) dengan persamaan (2.10).

$$\delta_j = z_j(1 - z_j) \sum_{k=1}^n \delta_k w_{jk} \quad \dots (2.9)$$

$$\Delta v_{ij} = \alpha \delta_j x_i \quad \dots (2.10)$$

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad \dots (2.11)$$

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad \dots (2.12)$$

Cross Validation

Cross Validation adalah metode validasi untuk menemukan model terbaik. Nilai k yang terlalu besar dapat menghasilkan variansi menjadi besar sehingga dapat memicu terjadinya *overfitting*, sedangkan nilai k yang terlalu kecil dapat memicu terjadinya bias (Widyaningsih et al., 2021). Untuk penelitian ini, Nilai k yang digunakan adalah 5, artinya *data splitting* memiliki rasio 80:20.

C. Hasil Penelitian dan Pembahasan

Tahap awal sebelum masuk ke algoritma MLP dimulai dari *encoding* data yang bertipe kategorik. *Data Encoding* adalah proses mengubah data kategorik menjadi data numerik dan sangat diperlukan dalam klasifikasi menggunakan algoritma MLP.

Tabel 2. Hasil *Encoding*

Usia	Jumlah ART	Jenis Kelamin	Status Pernikahan	Pendidikan Tertinggi KRT	Wilayah	Status Bersekolah	Kegiatan
15	3	0	1	0	0	1	1
13	4	0	1	1	0	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
16	3	0	1	0	1	0	1
9	3	1	1	0	1	1	1

Setelah data sudah menjadi numerik, maka bisa dilanjutkan ke tahap selanjutnya yaitu normalisasi data (*data encoding* biner tidak dinormalisasi).

$$x_{norm} = \frac{15 - 7}{18 - 7} = 0,7273$$

Tabel 3. Data Hasil Normalisasi

Usia	Jumlah ART	Jenis Kelamin	Status Pernikahan	Pendidikan Tertinggi KRT	Wilayah	Kegiatan
0,7273	0,1667	0	1	0	0	1
0,5455	0,25	0	1	0,5	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
0,1818	0,25	0	1	0,5	1	1
0,8182	0,1667	0	1	0	1	1

Tahap *Feedforward*

Selanjutnya diambil contoh data baris pertama yang telah dinormalisasi. Berdasarkan jumlah variabel *X* yang berjumlah 7 (tujuh), dapat diketahui bahwa jumlah *neuron* pada *input layer* adalah 7 (tujuh). Untuk data pada Tabel 5. penulis akan menginisiasi bobot secara acak dari -1 sampai 1 dan dihasilkan angka acak. Pada ilustrasi ini penulis menggunakan 1 *hidden layer* 7 *neuron* di *hidden layer*.

Tabel 4. Hasil Pengacakan Bobot *Input-Hidden*

		Variabel						
		X_1	X_2	X_3	X_4	X_5	X_6	X_7
<i>Neuron</i>	1	0,3652	0,4373	-0,4454	0,1384	-0,9444	0,4641	-0,8549
	2	-0,7432	0,5942	0,8977	0,1616	-0,5226	-0,4018	-0,9710
	3	0,9045	0,8082	-0,2227	-0,4205	-0,9821	-0,3442	-0,5843
	4	0,5842	0,2030	-0,8355	0,9762	0,2773	0,8466	-0,8985
	5	-0,8767	0,0958	-0,4983	0,5030	-0,9029	-0,6006	0,7914
	6	-0,1017	-0,2460	0,9406	-0,0491	-0,0685	-0,3386	0,5873
	7	0,4287	0,6680	0,7823	-0,2287	0,9403	0,5765	-0,7222

Tabel 5. Hasil Pengacakan Bias pada *Neuron* di *Hidden Layer*

Bias pada <i>Neuron</i> di <i>Hidden Layer</i>						
1	2	3	4	5	6	7
0,6137	-0,1009	-0,0560	-0,8771	0,1392	-0,8515	-0,9270

Pada Tabel 6. menunjukkan nilai inisiasi awal untuk bias yang ada pada setiap *neuron* di *hidden layer*. Selanjutnya akan dihitung nilai pada setiap *neuron* di *hidden layer* sebelum aktivasi *sigmoid* (z_{net}) dan akan di ilustrasikan untuk nilai *neuron* pertama di *hidden layer*.

$$z_{net} = 0,6137 + ((0,7273 \times 0,3652) + \dots + (1 \times (-0,8549))) = 0,2357$$

Tabel 6. Nilai *Neuron* di *Hidden Layer* Sebelum Aktivasi *Sigmoid*

Nilai pada Setiap <i>Neuron</i> di <i>Hidden Layer</i> (z_{net})						
1	2	3	4	5	6	7
0,2357	-1,3518	-0,2684	-0,3407	0,8120	-0,4283	-1,4548

Selanjutnya akan dilakukan fungsi aktivasi *sigmoid* dan akan di ilustrasikan untuk nilai *neuron* pertama setelah melakukan fungsi aktivasi *sigmoid*.

$$z_j = \frac{1}{1+e^{-0,2357}} = 0,5586$$

Tabel 7. Nilai *Neuron* di *Hidden Layer* Setelah Aktivasi *Sigmoid*

Nilai pada Setiap <i>Neuron</i> di <i>Hidden Layer</i> Setelah Fungsi Aktivasi <i>Sigmoid</i> (z_j)						
1	2	3	4	5	6	7
0,5586	0,2056	0,4333	0,4156	0,6925	0,3945	0,1893

Tabel 8. Hasil Pengacakan Bobot *Hidden-Output*

Bobot pada Setiap <i>Neuron</i> di <i>Hidden Layer</i> ke <i>Output Layer</i>						
1	2	3	4	5	6	7
0,7741	0,4544	-0,0216	0,5545	-0,2412	-0,9233	0,2597

Pada Tabel 4.9 menunjukkan hasil dari inisiasi pengacakan untuk bobot dari *neuron* di *hidden layer* ke *neuron* di *output layer*. Selanjutnya melakukan inisiasi pengacakan untuk bias yang ada pada *neuron* di *output layer* dan nilai bias pada *output layer* adalah -0,5424, dan akan dihitung nilai pada setiap *neuron* di *output layer* sebelum aktivasi *sigmoid* (y_{net}).

$$y_{net} = -0,5424 + ((0,5586 \times 0,7741) + \dots + (0,1893 \times 0,2597)) = -0,2776$$

Perhitungan (y_{net}) menunjukkan nilai pada *neuron* di *output layer* sebelum dilakukan aktivasi *sigmoid* sebesar -0,2776. Selanjutnya nilai (y_{net}) akan dilakukan fungsi aktivasi *sigmoid*.

$$y_k = \frac{1}{1+e^{-0,2776}} = 0,431$$

Nilai diatas adalah hasil perhitungan dari *neuron* pada *output layer* setelah aktivasi *sigmoid*. Untuk nilai $\leq 0,7$ maka akan dikatakan bahwa prediksinya adalah “0” (Putus Sekolah) dan untuk nilai $> 0,7$ maka akan dikatakan bahwa prediksinya adalah “1” (Bersekolah).

Tahap *Backward*

Sebelum benar-benar masuk ke tahap *backward* maka akan dilakukan terlebih dahulu tahap pembobotan kelas. Tanpa pembobotan kelas, model cenderung bias terhadap kelas mayoritas (Bersekolah) yang berpotensi menurunkan sensitivitas terhadap kelas minoritas (Putus Sekolah), berikut adalah perhitungan bobot setiap kelasnya.

$$w_0 = \frac{17729}{1405 \times 2} = 6,304 ; w_1 = \frac{17729}{16324 \times 2} = 0,5431$$

Hasil Perhitungan menunjukkan bahwa bobot untuk kelas minoritas (Putus Sekolah) adalah sebesar 6,304, sedangkan bobot untuk kelas mayoritas (Sekolah) adalah sebesar 0,5431. Hal tersebut

mendeskripsikan bahwa kesalahan prediksi pada kelas “Putus Sekolah” akan diperhitungkan 11 kali lebih besar dalam fungsi *loss*, sehingga model akan lebih sensitif terhadap prediksi siswa yang berisiko putus sekolah. Selanjutnya kita masuk ke tahap *backward*. Langkah awal pada tahap *backward* dimulai dengan menentukan nilai *error* pada *neuron* yang ada di *output layer*. Nilai *error* pada *output layer* dapat dilihat pada perhitungan berikut.

$$\delta_k = 0,5431 \times (1 - 0,431) \times 431 \times (1 - 0,431) = 0,0758$$

Hasil diatas memberikan contoh kesalahan klasifikasi untuk kelas mayoritas (Bersekolah) dimana seharusnya klasifikasinya menghasilkan nilai “1” pada nilai aktualnya, tetapi pada hasil prediksi menghasilkan *output* mendekati nilai “0”. Selanjutnya akan dilakukan perhitungan perubahan bobot yang menghubungkan *neuron* pertama di *hidden layer* dan *neuron* di *output layer*.

$$\Delta w_{jk} = 0,01 \times 0,0758 \times 0,5586 = 0,0004$$

Tabel 9. Perubahan Bobot *Hidden-Output*

Perubahan Bobot Neuron <i>Hidden-Output</i>						
1	2	3	4	5	6	7
0,0004	0,0002	0,0003	0,0003	0,0005	0,0003	0,0001

Perhitungan selanjutnya adalah menghitung *error* di setiap *neuron* di *hidden layer*, dan akan diilustrasikan perhitungannya untuk *neuron* pertama di *hidden layer*.

$$\delta_j = 0,5586 \times (1 - 0,5586) \times 0,0758 \times 0,7741 = 0,0144$$

Tabel 10. *Error* pada Setiap Neuron di *Hidden Layer*

<i>Error</i> pada Neuron di <i>Hidden Layer</i>						
1	2	3	4	5	6	7
0,0144	0,0056	-0,0004	0,0102	-0,0039	-0,0167	0,003

Selanjutnya akan dilakukan ilustrasi perubahan bobot pada koneksi *neuron* pertama di *input layer* dan *neuron* pertama di *hidden layer*.

$$\Delta v_{ij} = 0,01 \times 0,0144 \times 0,7273 = 0,0001$$

Tabel 11. Perubahan Bobot *Input-Hidden*

Perubahan Bobot Neuron <i>Input-Hidden</i>						
1	2	3	4	5	6	7
0,000105193	0,0000241067	0	0,00014464	0	0	0,00014464
0,0000409029	0,00000937359	0	0,0000562415	0	0	0,0000562415
⋮	⋮	⋮	⋮	⋮	⋮	⋮
-0,00012155	-0,0000278559	0	-0,00016714	0	0	-0,00016714
0,0000219604	0,00000503259	0	0,0000301955	0	0	0,0000301955

Pembaruan Bobot

Selanjutnya penulis akan menampilkan hasil dari perubahan bobot yang sebelumnya sudah dihitung, dan akan diilustrasikan pada *neuron* pertama di *input layer* ke *neuron* pertama di *hidden layer*.

Tabel 12. Nilai Neuron di *Hidden Layer* Baru Sebelum Fungsi Aktivasi Sigmoid

Nilai pada Setiap Neuron di <i>Hidden Layer</i> (z_{net})						
1	2	3	4	5	6	7
0,2360	-1,3517	-0,2684	-0,3404	0,8119	-0,4287	-1,4547

Tabel 13. Nilai *Neuron* di *Hidden Layer* Baru Setelah Fungsi Aktivasi *Sigmoid*

Nilai pada Setiap <i>Neuron</i> di <i>Hidden Layer</i> Setelah Fungsi Aktivasi <i>Sigmoid</i> (z_j)						
1	2	3	4	5	6	7
0,5587	0,2056	0,4333	0,4157	0,6925	0,3944	0,1893

Bias pada *neuron* di *output layer* juga tidak berubah, sehingga bisa dilanjutkan untuk menghitung nilai *neuron* di *output layer* sebelum fungsi aktivasi *sigmoid*, dan dilanjutkan dengan melakukan fungsi aktivasi *sigmoid* pada *neuron* di *output layer*.

$$y_{net} = -0,5424 + ((0,5586 \times 0,7745) + \dots + (0,1893 \times 0,26)) = -0,2764$$

$$y_k = \frac{1}{1 + e^{-0,2364}} = 0,4313$$

Pada perhitungan setelah pembaruan bobot, dapat diambil kesimpulan bahwa model memperkirakan kemungkinan siswa untuk tetap bersekolah adalah sebesar 0,4313 dengan *threshold* adalah sebesar 0,7 maka untuk data pada baris pertama dikatakan bahwa anak tersebut berisiko putus sekolah ($\hat{y} = 0$). Selanjutnya penulis akan menampilkan perbandingan antara data target aktual dan data prediksi model pada *data testing* yang berjumlah 3.545 data. Untuk melihat performa model terhadap data yang belum dilihat sebelumnya, maka akan dilakukan *Loss* (kesalahan) pada *data testing* dengan percobaan 4 sampai 14 *neuron* di *hidden layer*, dan erdasarkan hasil pelatihan sebanyak 50 *epoch* dan dengan menggunakan metode *early stopping* untuk menghindari *overfitting*, diperoleh nilai *loss* terkecil pada data *testing* adalah 0,2728 dengan jumlah *neuron* di *hidden layer* sebanyak 7.

Tabel 14. Data Aktual vs Data Prediksi

X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y	$\hat{Y}$
0.72727	0.16667	0	1	0	0	1	1	1
0.45455	0.41667	1	1	0	0	1	1	0
0.45455	0.25	1	1	0	0	1	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0	0.41666667	1	1	2	1	1	1	0
0.45	0.33333333	0	1	0	1	1	1	1
0.09	0.33333333	1	1	0	1	1	1	1

Tabel 16. menjelaskan tentang perbandingan data aktual vs data prediksi. Namun terdapat perbedaan antara perhitungan data yang diilustrasikan dengan perhitungan yang dilakukan oleh bantuan *software* dengan bahasa *python*. Hal tersebut dikarenakan pada *software* dengan bahasa *python* selama proses pelatihan masih ada beberapa perbedaan yang hanya bisa dilakukan di *software*, dan penulis melakukan perhitungan manual hanya bertujuan untuk bisa mengilustrasikan kepada pembaca perhitungan manual secara garis besar.

Evaluasi Model

Selanjutnya akan dilakukan evaluasi model, tujuannya untuk mengetahui seberapa akurat model belajar. Tabel 17. adalah evaluasi model dengan 7 *neuron* di *hidden layer*.

Tabel 15. Evaluasi Model

<i>Neuron</i> di <i>Hidden Layer</i>	Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Accuracy</i>
7	0 (Putus Sekolah)	0,28	0,87	0,42	0,81
	1 (Sekolah)	0,99	0,81	0,89	

Pada hasil evaluasi model menunjukkan bahwa nilai *precision* pada kelas minoritas adalah sebesar 28%. Pada nilai *recall*, model menunjukkan nilai yang baik untuk klasifikasi kelas mayoritas maupun kelas minoritas yaitu sebesar 87%, hal ini disebabkan karena penggunaan pembobotan kelas.

Strategi ini lebih menguntungkan karena kesalahan mendeteksi siswa berisiko putus sekolah lebih dapat diterima dibandingkan gagal mendeteksi siswa yang benar-benar putus sekolah.

D. Kesimpulan

Berdasarkan analisis yang dilakukan menunjukkan hasil evaluasi pada model *multilayer perceptron* yang dibuat menunjukkan *Accuracy* sebesar 81%, *Precision* pada kelas putus sekolah sebesar 28%, *Recall* pada kelas putus sekolah sebesar 87%, dan *F1-Score* sebesar 42%. Nilai-nilai tersebut menunjukkan bahwa model mampu mengenali kasus anak yang berisiko putus sekolah, meskipun belum cukup tepat dalam menentukan siapa yang termasuk dalam kategori tersebut. Secara keseluruhan, model *multilayer perceptron* terbukti efektif digunakan untuk klasifikasi sosial seperti mendeteksi dini anak berisiko putus sekolah dan dapat digunakan sebagai alat bantu dalam mendukung perumusan kebijakan pendidikan yang lebih baik di masa yang akan datang.

Berdasarkan hasil penelitian yang telah dilakukan terdapat beberapa saran yang dapat diberikan. Pertama, pengembangan model klasifikasi di masa mendatang dapat mempertimbangkan variabel-variabel yang lebih kontekstual terhadap kasus anak putus sekolah yang diharapkan dapat meningkatkan ketepatan klasifikasi dalam memprediksi kasus anak putus sekolah. Kedua, meskipun pembobotan kelas terbukti efektif dalam meningkatkan *recall*, perlu diperhatikan juga antara sensitivitas model dan ketepatan klasifikasi, karena nilai *precision* yang rendah dapat banyak menyebabkan peringatan palsu. Ketiga, penggunaan algoritma lain yang mungkin digunakan untuk mengklasifikasi seperti *Support Vector Machine* (SVM) atau *Random Forest*. Keempat, secara praktis dapat digunakan secara langsung dalam perumusan kebijakan pendidikan di lapangan. Terakhir, penelitian selanjutnya dapat menerapkan algoritma *multilayer perceptron* pada data dari provinsi dan tahun yang berbeda untuk menyesuaikan dengan dinamika sosial ekonomi yang terus berkembang.

Daftar Pustaka

- Damayanti CR, M., & Yanti, T. S. (2022). Regresi Poisson Invers Gaussian (PIG) untuk Pemodelan Jumlah Kasus Pneumonia pada Balita di Provinsi Jawa Tengah Tahun 2019. *Jurnal Riset Statistika*, 1(2), 143–151. <https://doi.org/10.29313/jrs.v1i2.523>
- Diyandini, S. M., & Kurniati, E. (2024). Analisis Maksimalisasi Utilitas Konsumen Melalui Grafik dan Persamaan Rumus. *DataMath: Journal of Statistics and Mathematics*, 2(1), 1–10.
- Hairani, Saputro, K. E., & Fadli, S. (2020). K-means-SMOTE for handling class imbalance in the classification of diabetes with C4.5, SVM, and naive Bayes. *Jurnal Teknologi Dan Sistem Komputer*, 8(2), 89–93. <https://doi.org/10.14710/jtsiskom.8.2.2020.89-93>
- Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation (2nd ed.)* (2nd ed.). Pearson Education.
- Heaton, Jeff. (2005). *Introduction to neural networks with Java*. Heaton Research.
- Kemendikdasmen. (2024, August 22). *Jumlah Siswa Putus Sekolah Menurut Tingkat Tiap Provinsi*. <https://data.dikdasmen.go.id/dataset/p/peserta-didik/jumlah-siswa-putus-sekolah-menurut-tingkat-tiap-provinsi-indonesia-sd-2023>
- Luhung Mustika Budiharti, & Sunendiari, S. (2021). Pemodelan dan Pemetaan Jumlah Penderita Kusta di Jawa Barat dengan Regresi Binomial Negatif dan Flexibly Shaped Spatial Scan Statistic. *Jurnal Riset Statistika*, 1(2), 99–106. <https://doi.org/10.29313/jrs.v1i2.409>
- Mikrodata.co.id. (2020, December 15). *Fungsi Gerbang Logika dalam Rangkaian Elektronika*. <https://www.mikrodata.co.id/fungsi-gerbang-logika-dalam-rangkaian-elektronika/>

- Pandu, K. T., Aminuyati, A., & Atmaja, T. S. (2022). Analisis Faktor Penyebab Anak Putus Sekolah Jenjang Sekolah Menengah Pertama di Desa Maju Karya Kecamatan Parindu Kabupaten Sanggau. *Jurnal Pendidikan Sosiologi Dan Humaniora*, 13(2), 543. <https://doi.org/10.26418/j-psh.v13i2.57605>
- Pardede, D., & Herawan Hayadi, B. (2022). Kajian Literatur Multilayer Perceptron: Seberapa Baik Performa Algoritma Ini. *Journal of ICT Application and System*, 1(1), 23–34.
- Purba, M. (2023). *Faktor Internal dan Eksternal yang Memengaruhi Status Putus Sekolah di Provinsi DKI Jakarta Tahun 2021 (Internal and External Factors Affecting Drop Out Status in DKI Jakarta Province in 2021)*.
- Rana, A., Singh Rawat, A., Bijalwan, A., & Bahuguna, H. (2018). Application of Multi Layer (Perceptron) Artificial Neural Network in the Diagnosis System: A Systematic Review. *2018 International Conference on Research in Intelligent and Computing in Engineering (RICE)*, 1–6. <https://doi.org/10.1109/RICE.2018.8509069>
- Rindri, Y. A., & Fitriyani, A. (2023). Analisis Perbandingan Kinerja Algoritma Multilayer Perceptron dan K-Nearest Neighbor pada Klasifikasi Tipe Migrain. *Jurnal Teknologi Dan Informasi*, 13(1), 44–55. <https://doi.org/10.34010/jati.v13i1.9111>
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519>
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536. <https://doi.org/10.1038/323533a0>
- Sari, L., & Karliani, E. (2023). Pernikahan Dini dan Implikasinya Pada Angka Putus Sekolah di Desa Tumbang Habaon Kecamatan Tewah Kabupaten Gunung Mas. *Jurnal Pendidikan Ilmu Pengetahuan Sosial (JPIPS) Desember*, 15, 428–433. <http://e-journal.upr.ac.id/index.php/JP-IPS>
- Soetrisnaadisendjaja, D., & Sari, N. (2019). *Fenomena Anak Putus Sekolah di Kawasan Industri Kota Cilegon*.
- Ulfa, M. (2021). *Penerapan Jaringan Syaraf Tiruan Prediksi Kebutuhan Alat Lampu Penerangan Jalan Umum (LPJU) dengan Metode Backpropagation*. 14(1).
- Widyaningsih, Y., Arum, G. P., & Prawira, K. (2021). Aplikasi K-Fold Cross Validation Dalam Penentuan Model Regresi Binomial Negatif Terbaik. *BAREKENG: Jurnal Ilmu Matematika Dan Terapan*, 15(2), 315–322. <https://doi.org/10.30598/barekengvol15iss2pp315-322>
- Yaneri, A., Bandung, P., Vonika, N., & Suviani, V. (2022). *Analisis Penyebab Anak Putus Sekolah Bagi Keluarga Miskin*. 4(1). <https://monitor.co.id/2019/01/08/ferry->
- Zamhari, Noviani, D., & Zainuddin. (2023). *Perkembangan Pendidikan di Indonesia*. 1(5), 1–10.