

**Prediksi Kasus Aktif COVID-19 di Indonesia Menggunakan Jaringan Saraf Tiruan
*Multilayer Perceptron (MLP) dengan Algoritma Backpropagation***



Kelompok 7

Aulia Haq	235150201111079
Aurelia Salsabilla Yunanto Putri	235150201111075
Rania Putri Zayyanti	235150201111073

PROGRAM STUDI TEKNIK INFORMATIKA
DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2025

BAB 1 – PENDAHULUAN

1.1 Latar Belakang

Pandemi COVID-19 membawa tantangan besar dalam memantau dinamika penyebaran kasus di Indonesia. Salah satu indikator penting adalah kasus aktif, yaitu jumlah pasien yang masih dalam proses penanganan. Nilai kasus aktif berubah setiap hari dan membentuk sebuah deret waktu (*time series*) sehingga dapat diprediksi menggunakan metode pemodelan prediktif. Pandemi *Coronavirus Disease 2019* (COVID-19) menghadirkan tantangan besar bagi sistem kesehatan global, termasuk di Indonesia. Pola penyebaran kasus yang sangat dinamis, fluktuatif, dan bersifat non-linear dari waktu ke waktu menuntut adanya mekanisme peramalan yang akurat dan responsif.

Karakteristik data epidemiologi COVID-19 sebagai data deret waktu (*time series*) yang sarat dengan pola non-linear, periodisitas mingguan, dan ketergantungan temporal yang kompleks, seringkali melampaui kemampuan model peramalan linear tradisional (seperti ARIMA). Oleh karena itu, diperlukan pendekatan yang lebih canggih yang mampu menangkap pemetaan hubungan non-linear antar periode waktu.

Dalam konteks ini, Jaringan Saraf Tiruan (JST) tipe Multilayer Perceptron (MLP) menonjol sebagai solusi yang efektif. MLP, yang secara teoritis dikenal sebagai *Universal Function Approximator*, mampu memodelkan hubungan kompleks ini, terutama ketika data deret waktu telah diubah menjadi format pembelajaran terawasi (*supervised learning*). Pelatihan MLP didukung oleh Algoritma Backpropagation yang sistematis dalam menyesuaikan bobot jaringan untuk meminimalkan error prediksi.

Model dibangun untuk memprediksi kasus aktif COVID-19 Indonesia berdasarkan data 7 hari sebelumnya. Pemilihan parameter input, yaitu penggunaan lag window tujuh hari ($L = 7$), didasarkan pada dua justifikasi utama. Pertama, $L = 7$ hari selaras dengan pola pelaporan kasus COVID-19 yang sering menunjukkan bias mingguan.¹ Kedua, periode ini konsisten dengan rata-rata waktu inkubasi atau latensi klinis virus, yang umumnya berkisar antara lima hingga tujuh hari. Dengan pendekatan ini, diharapkan diperoleh model yang stabil, konvergen, dan mampu mengikuti tren naik-turun kasus aktif.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, penelitian ini merumuskan masalah-masalah utama yang harus dipecahkan, meliputi aspek pra-pemrosesan data, implementasi algoritma, perancangan arsitektur model, dan evaluasi akurasi prediksi :

1. Bagaimana memodelkan prediksi kasus aktif COVID-19 menggunakan JST MLP?
2. Bagaimana proses pra-pemrosesan data time series untuk model backpropagation?
3. Seberapa akurat model dalam memprediksi nilai kasus aktif harian?

1.3 Tujuan Penelitian

Tujuan dari pelaksanaan Tugas Akhir ini adalah:

1. Mendesain dan menerapkan *pipeline* pra-pemrosesan data deret waktu yang robust untuk data Kasus Aktif COVID-19 agregat nasional.
2. Merancang dan mengimplementasikan arsitektur JST MLP (7-32-16-1) serta melatihnya menggunakan algoritma Backpropagation yang dikombinasikan dengan Adam Optimizer yang diimplementasikan secara mandiri (manual).
3. Mengevaluasi dan menganalisis secara kritis kinerja model dalam memprediksi Kasus Aktif harian menggunakan metrik RMSE dan MAE

1.4 Batasan Masalah

Untuk memastikan fokus dan ketercapaian tujuan penelitian, batasan-batasan berikut ditetapkan :

1. Fokus Data: Data yang digunakan dibatasi hanya pada agregat nasional Indonesia, yang didapatkan dari dataset Kaggle (COVID-19 Indonesia - Hendratno).
2. Variabel Input: Input model hanya menggunakan satu variabel tunggal (*univariate*), yaitu nilai Kasus Aktif dari tujuh hari sebelumnya ($L = 7$). Faktor eksternal seperti kebijakan pemerintah (PPKM, vaksinasi) tidak dipertimbangkan.
3. Model Prediksi: Penelitian difokuskan secara eksklusif pada model JST *Multilayer Perceptron* (MLP) dengan algoritma pelatihan Backpropagation dan optimasi Adam manual. Tidak dilakukan perbandingan dengan model machine learning atau deep learning lainnya (seperti LSTM atau GRU).

BAB 2 – TINJAUAN PUSTAKA

2.1 COVID-19 dan Kasus Aktif

Kasus Aktif adalah metrik yang didapatkan dari perhitungan deterministik, yaitu :

$$\text{Kasus Aktif} = \text{Total Kasus} - \text{Total Sembuh} - \text{Total Meninggal}$$

Metrik ini menunjukkan jumlah pasien yang masih menjalani perawatan atau isolasi. Dalam penelitian ini, langkah validasi awal dilakukan secara eksplisit untuk memastikan bahwa perhitungan matematis Kasus Aktif konsisten dengan data yang dilaporkan. Validasi ini memastikan bahwa variabel target bersifat *reliable* untuk peramalan.

2.2 Time Series Forecasting

Deret Waktu (*Time Series*) adalah sekumpulan data yang dicatat dalam interval waktu yang berurutan dan konsisten (misalnya, harian, mingguan, bulanan). Data Kasus Aktif COVID-19 termasuk data deret waktu karena nilai kasus pada hari tertentu sangat dipengaruhi oleh nilai kasus pada hari-hari sebelumnya. Karakteristik utama data deret waktu adalah adanya ketergantungan temporal (*temporal dependence*) dan seringkali mengandung pola seperti tren, musiman (*seasonality*), dan siklus.

Untuk menerapkan model kecerdasan buatan, data deret waktu perlu ditransformasi menjadi format yang sesuai untuk pembelajaran terarah (*Supervised Learning*). Proses ini dikenal sebagai teknik Jendela Geser (*Sliding Window*) atau Lag Window. Dalam konteks ini, nilai kasus aktif yang telah terjadi pada L hari ke belakang (*lag*) digunakan sebagai variabel input (X) untuk memprediksi nilai kasus aktif pada hari berikutnya sebagai variabel output (Y).

$$\text{Input (X)} = \{ X_{t-L}, X_{t-L+1}, \dots, X_{t-1} \}$$

$$\text{Output (Y)} = \{ Y_t \}$$

Dalam penelitian ini, nilai $L = 7$ (tujuh hari) dipilih sebagai *lag* input, sehingga model menggunakan data kasus aktif dari $t-7$ hingga $t-1$ untuk memprediksi t . Pemilihan $L = 7$ ini relevan untuk menangkap pola mingguan yang sering terjadi dalam pelaporan data kasus

2.3 Jaringan Saraf Tiruan (JST)

2.3.1 Jaringan Saraf Tiruan (JST)

Jaringan Saraf Tiruan (JST) adalah model komputasi yang terinspirasi dari struktur dan fungsi jaringan saraf biologis di otak manusia. JST terdiri dari lapisan-lapisan neuron yang saling terhubung, di mana setiap koneksi memiliki bobot (weight) yang akan disesuaikan selama proses pembelajaran. Kemampuan JST untuk memodelkan hubungan non-linear yang kompleks menjadikannya ideal untuk peramalan deret waktu.

2.3.2 Multilayer Perceptron (MLP)

MLP adalah JST *feedforward* yang terdiri dari setidaknya tiga lapisan: lapisan input, satu atau lebih lapisan tersembunyi (*hidden layer*), dan lapisan output.:

1. Lapisan Input (*Input Layer*): Menerima data masukan (dalam kasus ini, 7 hari data kasus aktif).
2. Lapisan Tersembunyi (*Hidden Layer*): Satu atau lebih lapisan yang memproses input melalui fungsi aktivasi non-linear. Lapisan ini bertanggung jawab untuk mengekstrak fitur dan pola kompleks dari data.
3. Lapisan Output (*Output Layer*): Menghasilkan hasil prediksi akhir (dalam kasus ini, 1 nilai prediksi kasus aktif untuk hari ke-8).

Arsitektur yang digunakan dalam penelitian ini mencakup dua lapisan tersembunyi dengan fungsi aktivasi non-linear.

2.3.3 Fungsi Aktivasi

Fungsi aktivasi (misalnya ReLU dan Linear) digunakan dalam JST untuk memperkenalkan non-linearity, yang sangat penting untuk memecahkan masalah non-linear seperti prediksi COVID-19 :

- ReLU (*Rectified Linear Unit*): Digunakan pada lapisan tersembunyi. Fungsi ini didefinisikan sebagai $f(x) = \max(0, x)$. Kelebihannya adalah mengatasi masalah *vanishing gradient* dan mempercepat proses pelatihan.
- Linear: Digunakan pada lapisan *output* untuk masalah regresi (peramalan nilai kontinu). Fungsi ini $f(x) = x$, memungkinkan hasil *output* berupa bilangan riil yang tidak dibatasi.

2.4 Backpropagation

Backpropagation (*Backward Propagation of Errors*) adalah algoritma kunci yang digunakan untuk melatih JST MLP. Tujuannya adalah meminimalkan fungsi kerugian (*Loss Function*) dengan menyesuaikan bobot koneksi jaringan.

Algoritma ini bekerja dalam dua fase utama:

1. Umpan Maju (*Forward Pass*): Input data diteruskan dari lapisan input melalui lapisan tersembunyi hingga mencapai lapisan output untuk menghasilkan nilai prediksi (\hat{Y}).
2. Umpan Mundur (*Backward Pass*):
 - Perhitungan Error: Kerugian dihitung dengan membandingkan prediksi (\hat{Y}) dengan nilai aktual (Y) menggunakan fungsi kerugian *Mean Squared Error* (MSE), di mana:

$$MSE = \frac{1}{N} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2$$

- Propagasi Error: Error disebarluaskan kembali dari lapisan output ke lapisan-lapisan tersembunyi. Dengan menggunakan Aturan Rantai (*Chain Rule*) dari kalkulus, gradien (turunan parsial) dari fungsi kerugian terhadap setiap bobot dihitung. Gradien ini menunjukkan arah dan besarnya perubahan bobot yang diperlukan untuk mengurangi error.
- Pemutakhiran Bobot (*Weight Update*): Bobot dan bias diperbarui menggunakan algoritma optimasi (*Optimizer*) berdasarkan gradien yang telah dihitung.

2.5 Optimizer

Optimizer adalah algoritma yang digunakan untuk mengontrol bagaimana bobot jaringan disesuaikan berdasarkan gradien yang dihitung oleh Backpropagation.

Adam (*Adaptive Moment Estimation*) adalah algoritma optimasi yang banyak digunakan dalam deep learning dan dipilih dalam penelitian ini karena keunggulannya:

1. Kombinasi Momentum dan RMSProp: Adam menggabungkan keunggulan dari *Stochastic Gradient Descent* (SGD) dengan momentum (yang membantu mempercepat konvergensi di arah yang relevan) dan RMSProp (yang menyesuaikan learning rate untuk setiap parameter).
2. Laju Pembelajaran Adaptif: Adam secara terpisah menghitung laju pembelajaran adaptif untuk setiap bobot jaringan. Adam menggunakan estimasi momen pertama

- (rata-rata bergerak gradien) dan momen kedua (rata-rata bergerak kuadrat gradien) untuk menyesuaikan langkah pembaruan bobot (*step size*) secara dinamis.
3. Efisiensi: Adam terbukti efisien secara komputasi, mudah diimplementasikan, dan parameter konfigurasinya (*default setting*) cenderung bekerja dengan baik pada sebagian besar masalah deep learning.

BAB 3 – METODOLOGI PENELITIAN

3.1 Sumber Dataset

Dataset yang digunakan adalah COVID-19 Indonesia Time Series Dataset dari Hendratno (Kaggle). Dataset berisi data harian perkembangan COVID-19 untuk seluruh provinsi di Indonesia.

- Dataset: COVID-19 Indonesia Time Series (Hendratno).
- Sumber: Platform Kaggle.
- Fokus Data: Data agregat nasional, yaitu baris dengan Location = 'Indonesia'.

Jumlah baris total : 31822

Jumlah kolom : 38

Kolom utama yang digunakan, termasuk target prediksi, adalah:

- Date
- Total Cases
- Total Recovered
- Total Deaths
- Total Active Case

Tabel 3.1 Dataset Awal (5 baris pertama)

Date	Location	ISO Code	New Cases	New Deaths	New Recovered	New Active Cases	Total Cases	Total Deaths	Total Recovered	Total Active Cases
3/1/2020	DKI Jakarta	ID-JK	2	0	0	2	39	20	75	-56
3/2/2020	DKI Jakarta	ID-JK	2	0	0	2	41	20	75	-54
3/2/2020	Indonesia	ID-RI	2	0	0	2	2	0	0	2
3/2/2020	Riau	ID-RI	1	0	0	1	1	0	0	1
3/3/2020	DKI Jakarta	ID-JK	2	0	0	2	43	20	75	-52

3.2 Tahapan Pra-Pemrosesan Data

Tahap pra-pemrosesan data bertujuan untuk menyiapkan data deret waktu agar siap digunakan oleh model JST MLP.

1. Filter dan Pengurutan Data: Data di filter untuk lokasi 'Indonesia' dan diurutkan berdasarkan tanggal (*Date ascending*).

Setelah filter (5 baris contoh)

Jumlah baris (Indonesia) : 929

Rentang tanggal : 2020-03-02 sampai 2022-09-16

Tabel 3.2 Hasil Filter dan Pengurutan Data

Date	Location	Total Cases	Total Recovered	Total Deaths	Total Active Cases
2020-03-02	Indonesia	2	0	0	2
2020-03-03	Indonesia	2	0	0	2
2020-03-04	Indonesia	2	0	0	2
2020-03-05	Indonesia	2	0	0	2
2020-03-06	Indonesia	4	0	0	4

2. Validasi dan Penentuan Deret Waktu Utama: Kolom Active Cases ditentukan sebagai deret waktu utama yang akan diprediksi.

Tabel 3.3 Validasi Deret Waktu Active Cases

Date	Total Cases	Total Recovered	Total Deaths	Total Active Cases	Active_calc
2020-03-02	2	0	0	2	2
2020-03-03	2	0	0	2	2
2020-03-04	2	0	0	2	2
2020-03-05	2	0	0	2	2
2020-03-06	4	0	0	4	4

3. Penanganan Missing Value: Nilai yang hilang (NaN) ditangani menggunakan kombinasi metode interpolasi(method="linear"), diikuti dengan ffill (forward fill), dan bfill (backward fill) untuk mengisi celah yang mungkin tersisa.
4. Normalisasi Skala [0, 1]: Data dinormalisasi ke rentang [0,1] menggunakan MinMaxScaler 8 untuk menstabilkan proses training.
 - Nilai minimum Kasus Aktif: 29.
 - Nilai maksimum Kasus Aktif: 8534510.

5. Data deret waktu diubah menjadi format *supervised learning* menggunakan lag 7 hari. Nilai Kasus Aktif dari $t-7$ sampai $t-1$ dijadikan input untuk memprediksi nilai hari ke- t . Bentuk umumnya adalah:

$$X_t = [a_{t-7}, \dots, a_{t-1}], \quad y_t = a_t$$

Dengan demikian, setiap sampel input terdiri dari 7 nilai Kasus Aktif pada hari-hari sebelumnya, sedangkan targetnya adalah nilai Kasus Aktif pada hari berikutnya. Setelah proses ini, diperoleh 922 sampel supervised dengan dimensi sebagai berikut:

- a. Bentuk X : (922, 7)
- b. Bentuk y : (922,)

Berikut adalah ilustrasi satu sampel input (X) dan target (y) yang digunakan oleh model:

Tabel 3.4. Contoh Sampel Data Supervised Setelah Pembentukan Lag 7 Hari

Lag	Tanggal	Active (Asli)	Active (Scaled)
Hari ke-1 (t-7)	2020-03-02	2	0.000000
Hari ke-2 (t-6)	2020-03-03	2	0.000000
Hari ke-3 (t-5)	2020-03-04	2	0.000000
Hari ke-4 (t-4)	2020-03-05	2	0.000000
Hari ke-5 (t-3)	2020-03-06	4	0.000003
Hari ke-6 (t-2)	2020-03-07	4	0.000003
Hari ke-7 (t-1)	2020-03-08	6	0.000007
Target (t)	2020-03-09	19	0.000029

Transformasi ini menghasilkan dataset yang siap digunakan untuk proses pelatihan, validasi, dan pengujian model JST MLP pada tahap selanjutnya.

3.3 Pembagian Data

Pada penelitian ini, pembagian data dilakukan secara berurutan berdasarkan waktu (chronological split) karena dataset yang digunakan merupakan deret waktu (*time series*). Metode ini penting untuk memastikan bahwa model hanya belajar dari data

“masa lalu” dan dievaluasi menggunakan data pada periode waktu setelahnya, sehingga tidak terjadi kebocoran informasi (*data leakage*).

Jumlah total sampel supervised yang diperoleh setelah proses sliding window adalah 922 sampel. Dataset tersebut selanjutnya dibagi menjadi tiga subset dengan komposisi sebagai berikut:

- a. Training set – 70% (645 sampel)

Digunakan untuk proses pelatihan model dan penyesuaian bobot menggunakan algoritma Backpropagation.

Rentang tanggal: 9 Maret 2020 – 13 Desember 2021

- b. Validation set – 15% (138 sampel)

Berfungsi untuk memantau performa model selama training serta membantu proses *hyperparameter tuning* dan *early stopping*.

Rentang tanggal: 14 Desember 2021 – 30 April 2022

- c. Testing set – 15% (139 sampel)

Digunakan sebagai evaluasi akhir untuk mengukur kemampuan generalisasi model terhadap data baru. Subset ini tidak terlibat sama sekali dalam proses pelatihan.

Rentang tanggal: 1 Mei 2022 – 16 September 2022

Pembagian ini menghasilkan alur data yang terstruktur dari fase awal pandemi hingga periode terbaru dalam dataset, sehingga model dapat mempelajari pola kasus aktif secara bertahap. Dengan struktur pembagian yang terurut seperti ini, evaluasi model menjadi lebih akurat dan mencerminkan kemampuan prediksi pada situasi nyata.

3.4 Arsitektur Model JST

Model yang diusulkan adalah *Multilayer Perceptron* (MLP) dengan arsitektur *feedforward* yang terdiri dari dua lapisan tersembunyi.

Rancangan arsitektur model:

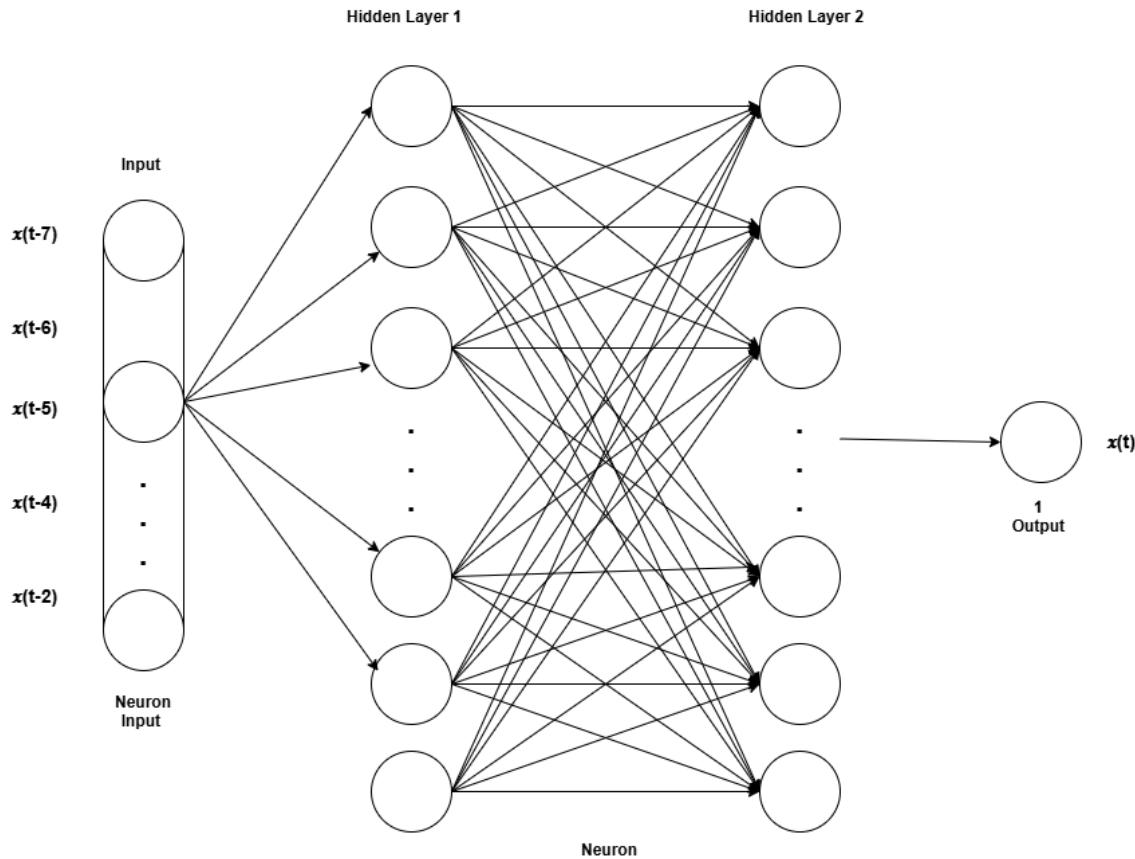
Tabel 3.5. Arsitektur JST (MLP) untuk Prediksi Kasus Aktif COVID-19

Lapisan	Jumlah Neuron	Fungsi Aktivasi	Keterangan
Input Layer	7 neuron	–	Menerima input dari 7 hari sebelumnya (<i>lag = 7</i>)
Hidden Layer 1	32 neuron	ReLU	Menangkap pola non-linear dan mempercepat konvergensi
Hidden Layer 2	16 neuron	ReLU	Menambah kedalaman jaringan untuk mempelajari hubungan kompleks
Output Layer	1 neuron	Linear	Menghasilkan prediksi nilai kasus aktif hari berikutnya

Algoritma Pembelajaran:

- a. Forward Pass: Data input diteruskan ke seluruh lapisan hingga menghasilkan prediksi \hat{Y}
- b. Perhitungan Error: Error dihitung menggunakan *Mean Squared Error* (MSE).
- c. Backward Pass: Error disebarluaskan mundur untuk memperbarui bobot dan bias menggunakan metode gradient descent.
- d. Iterasi: Proses diulang hingga 200 epoch dengan EarlyStopping untuk mencegah *overfitting*.
- e. Model akan dievaluasi menggunakan metrik MAE, RMSE, dan MAPE untuk mengukur akurasi prediksi terhadap data aktual.

Gambar berikut menunjukkan arsitektur jaringan yang digunakan:



Gambar 3.1. Arsitektur Jaringan Saraf Tiruan (MLP) untuk Prediksi Kasus Aktif COVID-19

Arsitektur JST terdiri dari 7 neuron input (lag 7 hari), dua hidden layer dengan masing-masing 32 dan 16 neuron (aktivasi ReLU), serta 1 neuron output linear untuk memprediksi kasus aktif harian.

3.5 Proses Training Backpropagation

Proses pelatihan menggunakan implementasi algoritma Backpropagation secara manual, dengan fungsi kerugian *Mean Squared Error* (MSE), dan pembaruan bobot menggunakan Adam Optimizer.

3.5.1 Rumus Umpam Maju (Forward Pass)

Setiap neuron menghitung net input dan nilai aktivasi. Untuk lapisan tersembunyi, l , net input ($z^{[l]}$) dan nilai aktivasi ($a^{[l]}$) dihitung sebagai:

$$z^{[l]} = a^{[l-1]} W^{[l]} + B^{[l]}$$

$$a^{[l]} = \text{ReLU}(z^{[l]})$$

Untuk lapisan output (regresi), digunakan aktivasi Linear ($y_{\text{pred}} = z^{[L]}$)

3.5.2 Fungsi Kerugian (Loss Function)

Kesalahan antara prediksi (y_{pred}) dan nilai aktual (y) diukur menggunakan Mean Squared Error (MSE)21:

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

3.5.3 Rumus Umpam Mundur (Backward Pass)

Error disebarluhan mundur untuk menghitung gradien kerugian terhadap bobot (W) dan bias (B) pada setiap lapisan.

Output Layer Error ($\delta^{[L]}$) (menggunakan turunan MSE):

$$\delta^{[L]} = a^{[L]} - y \text{ (Asumsi Linear Output)}$$

Hidden Layer Error ($\delta^{[l]}$):

$$\delta^{[l]} = (\delta^{[l+1]} W^{[l+1]}) \odot g'(z^{[l]})$$

Di mana $g'(z^{[l]})$ adalah turunan dari fungsi aktivasi ReLU (1 jika $z > 0$, dan 0 jika $z \leq 0$)
Rumus MSE

3.5.4 Pembaruan Bobot dengan Adam Optimizer (Manual)

Bobot dan bias diperbarui menggunakan Adam. Adam menggunakan rata-rata bergerak terbobot dari gradien (m , momen pertama) dan kuadrat gradien (v , momen kedua) untuk perhitungan learning rate adaptif.

- Pembaruan Momen dan Koreksi Bias:

$$\hat{m} = \frac{\beta_1 m + (1-\beta_1) g}{1-\beta_1^t}; \hat{v} = \frac{\beta_2 v + (1-\beta_2) g^2}{1-\beta_2^t}$$

Di mana g adalah gradien (grad_W atau grad_B).

- Pembaruan Bobot Akhir:

$$W \leftarrow W - \text{lr} \cdot \frac{\widehat{m}}{\sqrt{\widehat{v}} + \epsilon}$$

Hyperparameter yang digunakan adalah: lr=0.001\$, $\beta_1=0.9$, $\beta_2=0.999$, dan $\epsilon = 1e-8$.

BAB 4 – HASIL & PEMBAHASAN

Bab ini menyajikan hasil penelitian berupa proses pra-pemrosesan data, hasil pembentukan data supervised, proses pelatihan jaringan saraf tiruan (JST) Multilayer Perceptron (MLP) menggunakan algoritma Backpropagation dengan optimizer Adam manual, serta evaluasi performa prediksi terhadap data aktual. Pembahasan dilakukan secara mendalam dengan menghubungkan hasil keluaran program, grafik, serta analisis pola prediksi.

4.1 Hasil Pra-Pemrosesan Data

- Data berhasil disiapkan dengan langkah-langkah:
- Tidak ada missing value
- Deret waktu konsisten
- Normalisasi berjalan baik
- Data supervised (lag=7) berhasil dibentuk

4.2 Hasil Training (Backpropagation + Adam)

Pelatihan dilakukan menggunakan:

- Arsitektur JST: $(7 \rightarrow 32 \rightarrow 16 \rightarrow 1)$
- Aktivasi hidden: ReLU
- Aktivasi output: Linear
- Optimizer: Adam (manual)
- Epoch: 2000
- Learning rate: 0.001
- a. Perkembangan MSE Selama Training

Tabel berikut menunjukkan penurunan MSE pada beberapa epoch:

Tabel 4.1 Perkembangan MSE Selama Training

Epoch	MSE (Mean Squared Error)
50	0.017174
100	0.000887
150	0.000747
200	0.000621

Epoch	MSE (Mean Squared Error)
250	0.000458
300	0.000311
350	0.000198
400	0.000124
450	0.000086
500	0.000070
550	0.000064
600	0.000063
650	0.000061
700	0.000058
750	0.000057
800	0.000056
850	0.000055
900	0.000054
950	0.000053
1000	0.000052
1050	0.000051
1100	0.000050
1150	0.000049
1200	0.000047
1250	0.000046
1300	0.000045

Epoch	MSE (Mean Squared Error)
1350	0.000044
1400	0.000043
1450	0.000042
1500	0.000041
1550	0.000040
1600	0.000039
1650	0.000038
1700	0.000037
1750	0.000036
1800	0.000035
1850	0.000034
1900	0.000033
1950	0.000033
2000	0.000032

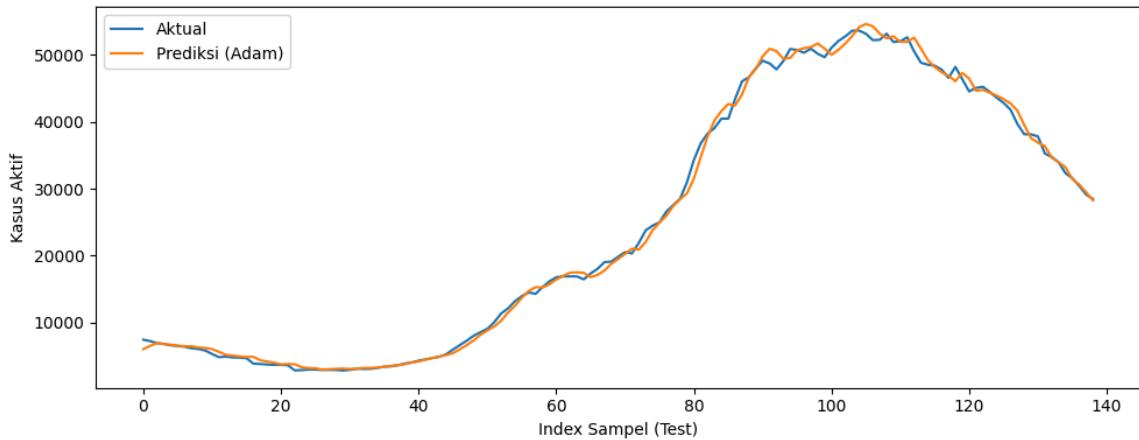
Interpretasi:

- 1) Penurunan MSE signifikan dari epoch 0 hingga ~300, menunjukkan fase belajar cepat.
- 2) Setelah itu, penurunan MSE melambat namun tetap konsisten.
- 3) Nilai MSE mendekati nol menunjukkan akurasi sangat tinggi pada skala normalisasi.
- 4) Tidak ada tanda *overfitting* karena MSE terus menurun tanpa fluktuasi ekstrem.
- 5) Adam membantu menjaga stabilitas update bobot sehingga konvergensi berlangsung halus.

4.3 Hasil Prediksi vs Data Aktual

Untuk mengevaluasi kemampuan model, dilakukan prediksi pada data testing dan hasilnya dibandingkan dengan nilai aktual.

4.3.1 Grafik Perbandingan Prediksi dan Aktual



Gambar 4.1 Grafik Perbandingan Prediksi vs Aktual

Grafik menampilkan dua garis:

- Garis biru : nilai Aktual kasus aktif
- Garis oranye : nilai Prediksi (Adam)

Kedua garis tampak sangat berdekatan dan mengikuti pola yang sama, terutama pada fase kenaikan dan puncak kasus aktif. Pada beberapa titik di bagian akhir terlihat deviasi kecil, tetapi pola besar tetap selaras.

4.3.2 Tabel 5 Sampel Pertama (Data Test)

Berikut contoh 5 sampel pertama pada data uji (dalam skala jumlah kasus):

Tabel 4.2 Perbandingan: Aktual vs Prediksi (5 Sampel Pertama)

Idx	Aktual	Prediksi	Selisih
1	7474.0	6450.8	-1023.2
2	7257.0	6840.4	-416.6
3	6951.0	6933.1	-17.9

Idx	Aktual	Prediksi	Selisih
4	6845.0	6730.8	-114.2
5	6631.0	6654.2	23.2

Selisih berada pada kisaran ratusan kasus dan sejalan dengan nilai MAE yang akan dibahas pada Subbab 4.4.

4.3.3 Analisis Pola

Secara kualitatif dapat disimpulkan:

- a. Model mampu mengikuti pola tren naik-turun kasus aktif dengan baik.
- b. Pada fase puncak, prediksi hampir menumpuk dengan data aktual sehingga puncak dan penurunannya tertangkap cukup tepat.
- c. Deviasi yang muncul di beberapa titik kemungkinan terkait fluktuasi ekstrem pada data aktual yang sulit dipelajari hanya dari satu variabel (kasus aktif) dan lag 7 hari.

4.4 Evaluasi Model

Evaluasi kuantitatif menggunakan dua metrik yang dimana perhitungan dilakukan setelah mengembalikan skala ke nilai asli. Hasilnya:

- 1) Root Mean Squared Error (RMSE)

$$RMSE = 867.444$$

- 2) Mean Absolute Error (MAE)

$$MAE = 600.613$$

4.4.1 Interpretasi Hasil

- a. Prediksi mengikuti tren

Grafik menunjukkan bahwa model berhasil mengikuti bentuk kurva kasus aktif di Indonesia, terutama pada fase kenaikan dan penurunan setelah puncak.

- b. Besaran error

Nilai MAE sekitar 600 kasus dan RMSE sekitar 867 kasus. Jika dibandingkan dengan skala data yang dapat mencapai lebih dari 500.000 kasus

aktif pada puncaknya, kesalahan ini berada di bawah 0,2% dari nilai puncak, sehingga dapat dikategorikan cukup kecil dan masih dapat diterima.

c. Penyebab deviasi

Deviasi yang muncul pada beberapa titik dapat disebabkan oleh:

- fluktuasi data aktual yang sangat tajam,
- penggunaan satu variabel saja (kasus aktif) tanpa memasukkan faktor lain,
- keterbatasan arsitektur MLP sederhana yang digunakan.

d. Stabilitas model

Penurunan MSE yang konsisten selama training dan nilai error yang moderat di data test menunjukkan bahwa model tidak mengalami *overfitting* yang parah dan cukup stabil.

BAB 5 – KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian model Multilayer Perceptron (MLP) dengan algoritma Backpropagation, diperoleh beberapa kesimpulan sebagai berikut:

- a. Arsitektur MLP Backpropagation dengan lag 7 hari mampu mempelajari pola historis kasus aktif COVID-19 dan mengikuti tren pergerakan datanya dengan baik. Penggunaan 7 hari sebagai input historis cukup efektif untuk menangkap pola mingguan pada data.
- b. Optimizer Adam yang diimplementasikan secara manual memberikan proses pelatihan yang stabil dan konvergen. Hal ini terlihat dari nilai MSE yang menurun signifikan dari 0,017 menjadi sekitar 0,000032 selama 2000 epoch training.
- c. Prediksi model terhadap data uji menunjukkan kecocokan pola yang tinggi. Garis prediksi pada grafik hampir menumpuk dengan garis aktual pada sebagian besar periode uji, meskipun terdapat beberapa deviasi kecil di titik-titik tertentu.
- d. Nilai error RMSE dan MAE berada pada tingkat yang dapat diterima. RMSE sebesar 867,444 dan MAE sebesar 600,613 masih relatif kecil dibandingkan skala kasus aktif harian yang dapat mencapai ratusan ribu kasus.
- e. Secara keseluruhan, model MLP Backpropagation dengan optimizer Adam dapat digunakan sebagai model prediksi sederhana untuk kasus aktif COVID-19 di Indonesia, terutama untuk mempelajari dan memperkirakan tren jangka pendek.

Secara keseluruhan, model MLP Backpropagation dengan optimizer Adam layak digunakan sebagai model prediksi sederhana untuk kasus aktif COVID-19, khususnya dalam mempelajari tren jangka pendek.

5.2 Saran

Untuk pengembangan lebih lanjut, beberapa saran yang dapat dipertimbangkan adalah:

- a. Menambah fitur (multivariate)
Masukkan variabel lain seperti new cases, jumlah tes harian, indikator mobilitas, atau data kebijakan (misalnya PPKM, vaksinasi) untuk memperkaya informasi yang dipelajari model.
- b. Mencoba arsitektur time series lain
Mengembangkan model yang lebih khusus untuk deret waktu seperti LSTM atau GRU (di luar batasan tugas ini), sehingga dapat dibandingkan dengan MLP.

c. Eksperimen hyperparameter

Melakukan pengujian terhadap variasi ukuran lag, jumlah neuron, jumlah hidden layer, serta nilai learning rate untuk mencari konfigurasi yang lebih optimal.

d. Prapemrosesan tambahan

Menggunakan teknik smoothing atau moving average untuk mengurangi noise harian yang ekstrem sebelum masuk ke model.

LAMPIRAN

Kode Program :  [MLP_dengan_Backpropagation_Adam.ipynb](#)

Link dataset: <https://www.kaggle.com/datasets/hendratno/covid19-indonesia>