

Estimasi Tinggi Muka Air Menggunakan Neural Network dengan GRG *Nonlinear* dan ADAM pada Wilayah Hilir POS Bojonegoro

The Estimation of Water Level Using Neural Network with GRG Nonlinear and ADAM in Downstream Area of Bojonegoro POST

Hanif Pamuji^{1*}, Dimara Kusuma Hakim²

^{1,2})Program Studi Teknik Informatika, Fakultas Teknik dan Sains
Universitas Muhammadiyah Purwokerto

Jl. Raya Dukuwaluh, Kembaran 53182, Indonesia

email: *¹hanifpamuji8@gmail.com, ²dimarakusumahakim@gmail.com

ABSTRAK

Wilayah hilir Pos Bojonegoro merupakan salah satu wilayah yang memerlukan estimasi tinggi muka air secara akurat untuk mencegah terjadinya banjir dan pengelolaan air secara efektif. Pemanfaatan Neural Network dalam pemodelan hidrologi sebagai solusi untuk meningkatkan akurasi estimasi tinggi muka air di wilayah hilir Pos Bojonegoro. Dengan tujuan menggunakan metode Neural Network dalam memprediksi dengan akurat tinggi muka air sungai di wilayah hilir Pos Bojonegoro. Keunggulan Neural Network terletak pada efisiensinya dalam perhitungan cepat dan kemampuannya menangani data tidak stabil yang umumnya ditemui dalam prakiraan cuaca. Hasil estimasi untuk bulan Januari hingga September 2023 menghasilkan nilai MSE 3.4424465503109847 dan MAPE 17.875026297198314%. Hasil evaluasi menunjukkan bahwa model Neural Network memiliki tingkat ketepatan yang baik dalam memprediksi tinggi muka air, ditunjukkan oleh nilai MSE yang relatif rendah. Selain itu, tingkat kesalahan yang rendah, sebagaimana tercermin dari nilai MAPE, mengindikasikan kualitas yang baik dalam peramalan tinggi muka air. Hasil penelitian ini memberikan kontribusi penting dalam pengembangan metode estimasi tinggi muka air yang akurat, memberikan landasan untuk pengambilan keputusan yang efektif dalam manajemen sumber daya air dan risiko banjir di wilayah hilir Pos Bojonegoro.

Kata Kunci: Estimasi, *Neuron Network*, Tinggi Muka Air, Pos Bojonegoro

(Dikirim: 22 Januari 2024, Direvisi: 23 Januari 2024, Diterima: 27 Januari 2024)

ABSTRACT

The downstream area of Bojonegoro Post is one of the regions requiring accurate water levels estimates to prevent floods and manage water effectively. The utilization of Neural Network in hydrological modeling serves as a solution to enhance the accuracy of water level estimation in the downstream area of Bojonegoro Post. The aim is to use the Neural Network method for accurately predicting river water levels in the downstream area of Bojonegoro Post. The advantage of Neural Network lies in its efficiency in quick calculations and its ability to handle unstable data commonly encountered in weather forecasting. The estimation results for January to September 2023 yielded an MSE value of 3.4424465503109847 and MAPE of 17.875026297198314%. Evaluation results indicate that the Neural Network model demonstrates a high level of accuracy in predicting water levels, as evidenced by the relatively low MSE value. Additionally, the low error rate, as reflected in the MAPE value, indicates a good quality in forecasting water levels. This research contributes significantly to the development of accurate water level estimation methods, providing a foundation for effective decision-making in water resource management and flood risk in the downstream area of Bojonegoro Post.

Keywords: *Estimation, Neural Network, Water Level, Bojonegoro Post*

1. Pendahuluan

Geografis dan populasi di Indonesia memiliki karakteristik alam yang berpotensi menghadapi risiko banjir. Negeri ini terdiri dari setidaknya 16,771 pulau serta dilintasi oleh 65,017 anak sungai yang tersebar di seluruh kepulauan (Prasetyo & Hayati, 2019). Informasi mengenai aliran sungai adalah data utama yang sangat signifikan bagi pihak yang mengelola sumber daya air. Untuk perencanaan bangunan pengendali banjir, data mengenai debit puncak (banjir) menjadi krusial. Di sisi lain, informasi tentang debit aliran sungai pada tingkat yang lebih rendah sangat diperlukan untuk mengatur alokasi air demi berbagai keperluan, terutama ketika menghadapi musim kemarau Panjang (Vulandari & Parwitasari, 2018). Peningkatan kejadian banjir disebabkan oleh salah satu faktor, yaitu pemanasan global. Dampak dari pemanasan global adalah perubahan iklim yang terjadi karena kenaikan suhu rata-rata global dari tahun ke tahun, yang diakibatkan oleh efek rumah kaca atau fenomena *greenhouse effect* (Roy, 2022).

Daerah Aliran Sungai (DAS) merupakan suatu kawasan yang menampung, menyimpan, dan mengalirkan air yang berasal dari curah hujan sampai akhirnya bermuara di danau atau laut (Khulyati, 2018). Tinggi muka air sungai, waduk, dan danau harus diukur secara teratur untuk mengontrol banjir, memperkirakan kapasitas penyimpanan, dan untuk tujuan manajemen air lainnya. Meski demikian, seringkali menjadi tantangan untuk melakukan pengukuran langsung tinggi permukaan air, terutama di daerah yang terpencil atau sulit dijangkau. Neural Network (NN) adalah sistem pemrosesan informasi dengan sifat-sifat yang mirip dengan jaringan saraf biologis (Rustika, 2018).

Wilayah hilir Pos Bojonegoro merupakan salah satu wilayah yang memerlukan estimasi tinggi muka air yang akurat untuk menghindari terjadinya banjir dan memastikan pengelolaan air yang efektif. Wilayah ini terletak di Kabupaten Bojonegoro, Provinsi Jawa Timur, Indonesia. Banjir Bengawan Solo terjadi pada tanggal 26 Desember 2007 sampai 7 Januari 2008 sebagai peristiwa bencana alam terbesar dalam puluhan tahun terakhir yang menimbulkan kerugian secara material maupun non material bagi kehidupan masyarakat yang terdampak pada kehidupan ekonomi, sosial, maupun lingkungan sekitar (Nawiyanto & Arianto, 2022).

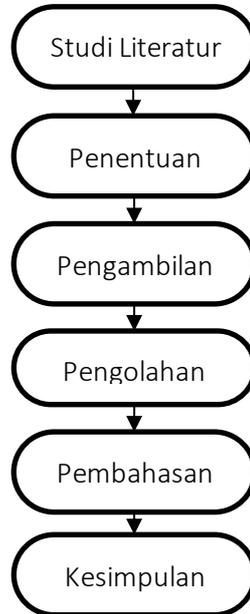
Penelitian ini bertujuan untuk menggunakan metode neural *network* dengan menggunakan optimasi GRG *Nonlinear* dan Adam yang dapat digunakan untuk mengestimasi tinggi muka air sungai di wilayah hilir Pos Bojonegoro. Neural Network dapat belajar dari data dalam jumlah besar dan melakukan generalisasi dengan baik terhadap data baru, menjadikannya berguna untuk berbagai aplikasi (Hamm, 2007). Penelitian yang dilakukan oleh (Hung, 2009) menggunakan Neural Network dengan menggunakan data dari tahun 1991 sampai 2003 yang diambil dari 51 pos hujan di Bangkok, studi ini menunjukkan bahwa Neural Network lebih efisien dalam perhitungan cepat dan mampu menangani data yang tidak stabil yang khas dalam kasus data untuk prakiraan cuaca. Metode neural *network* yang tepat dapat memprediksi tinggi muka air dengan akurasi yang tinggi, sehingga dapat membantu pihak berwenang dalam pengambilan keputusan dan tindakan pencegahan terhadap resiko air meluap atau banjir.

Penelitian ini akan dilakukan dengan menggunakan data klimatologi dan data tinggi muka air sungai yang telah dikumpulkan selama 2,9 tahun. Kemudian, data tersebut akan digunakan sebagai input untuk estimasi menggunakan metode neural *network*. Metode tersebut akan diuji dan divalidasi menggunakan data yang telah dikumpulkan. Setelah itu dilakukan evaluasi menggunakan *Mean Squared Error* dan *Mean Absolute Percentage Error* dengan menggunakan optimasi GRG *Nonlinear* dan Adam untuk memahami seberapa baik atau buruk model neural *network* dalam memprediksi atau menggeneralisasi pola pada data.

Salah satu faktor yang menyebabkan banjir di wilayah hilir Pos Bojonegoro adalah peningkatan tinggi muka air sungai. Oleh karena itu, perlu dilakukan estimasi tinggi muka air secara akurat agar dapat dilakukan pengambilan tindakan pencegahan.

2. Metode

Pada penelitian kali ini, penulis menggunakan metode kuantitatif untuk melakukan analisis data yang mencakup variabel seperti curah hujan, kelembapan, dan suhu udara. melakukan estimasi tinggi muka air dengan menggunakan metode *Neural Network*. Data yang digunakan di dalam penelitian ini diperoleh dari hidrologi.bbws-bsolo.net.



Gambar 1. Alur Penelitian

2.1. Studi Literatur

Dalam studi literatur, pada tahap ini dilakukan pencarian artikel ilmiah, buku, laporan penelitian, dan sumber informasi lainnya yang membahas berbagai metode dan teknik estimasi tinggi muka air.

2.2. Penentuan Metode

Dalam menentukan metode estimasi tinggi muka air, terdapat beberapa faktor yang perlu menjadi pertimbangan utama, ketersediaan data yang telah ada, serta tujuan dan tingkat akurasi yang diinginkan. Oleh karena itu, penelitian ini menggunakan metode *Neural Network* untuk estimasi tinggi muka air di wilayah Pos Bojonegoro.

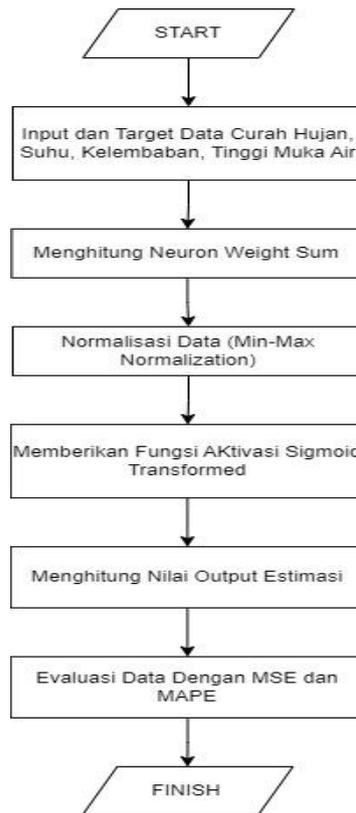
2.3. Pengambilan Data

Dalam penelitian estimasi tinggi muka air di Pos Bojonegoro, penggunaan data yang telah terkumpul sebelumnya sangat penting. Data mencakup serangkaian pengukuran curah hujan, suhu, kelembapan dan tinggi muka air yang telah dilakukan mencakup periode waktu selama 2,9 tahun.

Data penelitian ini diperoleh dari lembaga Kementerian PUPR Ditjen Sumber Daya Air BBWS Bengawan Solo pada *website* hidrologi.bbws-bsolo.net. Data ini tersedia dalam bentuk informasi tabel.

2.4. Pengolahan Data

Data yang diperoleh dari hidrologi.bbws-bsolo.net. Kemudian akan diproses menggunakan metode *Neural Network*. Penggunaan metode tersebut dalam penelitian ini yang digunakan adalah data terkait tinggi muka air, curah hujan, suhu, dan kelembapan udara selama 2 tahun untuk data *training* dan 9 bulan untuk data uji.



Gambar 2. Alur Neural Network

Dataset yang telah diperoleh akan dianalisis dengan menghitung nilai *neuron weighted sum* sebagai langkah awal dalam komputasi pada setiap *input neuron*. *Neuron weighted sum* menggabungkan *input* yang diberikan kepada *neuron* dengan bobot yang sesuai dan menambahkan bias, sebagai contoh ditunjukkan pada persamaan (1) :

$$\sum_i^n = W_i \cdot X_i + b \quad (1)$$

(Du & Swamy, 2006)

Keterangan persamaan (1) :

n adalah jumlah *input* ke *neuron* tersebut.

W_i adalah bobot yang diberikan kepada *input* ke- i .

X_i adalah *input* ke- i .

b adalah bias yang ditambahkan ke *weighted sum*.

Selanjutnya, mencari nilai minimum dan maksimum pada setiap *neuron weight sum* yang digunakan sebagai proses data mining yang bertujuan untuk membantu proses perhitungan data, sebagai contoh ditunjukkan pada persamaan (2) dan (3) :

$$\text{Data terbesar} = \max(x_1, x_2, \dots, x_n) \quad (2)$$

$$\text{Data terkecil} = \min(x_1, x_2, \dots, x_n) \quad (3)$$

Keterangan persamaan (2) dan (3) :

x adalah *input* yang akan dibandingkan.

Selanjutnya proses normalisasi data dilakukan untuk menjadikan data berada dalam skala yang sama dan dapat membantu dalam meningkatkan kinerja *neural network*, terutama jika ada perbedaan skala yang signifikan antara atribut atau fitur yang digunakan. *Min-Max* Normalisasi digunakan untuk mengubah rentang data menjadi rentang yang lebih terdefinisi dari -1 hingga 1. Rumus *Min-Max* Normalisasi ditunjukkan pada persamaan (4) :

$$x^i = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (4)$$

(Wiranto, 2023)

Keterangan persamaan (4) :

x^i adalah nilai data yang dinormalisasi.

x adalah nilai data asli.

$\min(x)$ adalah nilai minimum dari data.

$\max(x)$ adalah nilai maksimum dari data.

Selanjutnya menghitung sigmoid. Untuk masalah klasifikasi atau estimasi, kombinasi dari fungsi sigmoid memberikan hasil yang lebih baik (Sharma, 2020). Fungsi sigmoid tetap menjadi pilihan favorit terutama karena teknik komputasi ini lebih cepat. Selain itu, penggunaan teknik komputasi perkiraan fungsi telah secara drastis mengurangi kompleksitas komputasi ke tingkat akurasi yang memadai (Mulindwa & Du, 2023). Fungsi sigmoid adalah sebuah fungsi yang mengubah nilai numerik *input* menjadi rentang antara 0 hingga 1. Rumus *sigmoid transformation* ditunjukkan pada persamaan (5) :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

(Fausett, 1969)

Keterangan persamaan (5) :

$f(x)$ adalah *output* hasil transformasi sigmoid dari *input* x .

e adalah bilangan Euler, yang merupakan konstanta matematika sekitar 2.71828.

Untuk pelatihan dan akurasi yang lebih baik, lebih banyak lapisan tersembunyi harus digunakan jika akurasi hasil merupakan kriteria utama. Lapisan tersembunyi tunggal digunakan ketika waktu merupakan faktor utama (Asthana, 2017).

Olah data yang dilakukan akan ditemukan hasilnya dari setiap data yang akan digunakan *Neural Network*. *Neural Network* terdiri dari jaringan neuron buatan (*artificial neurons*) yang saling terhubung. Setiap *neuron* menerima *input*, menghitung *weighted sum* dari *input* tersebut dengan bobot terkait, dan meneruskan hasilnya melalui fungsi aktivasi. Rumus *neural network* ditunjukkan pada persamaan (6) :

$$y = \sigma(W \cdot x + b) \quad (6)$$

(Du & Swamy, 2006)

Keterangan persamaan (6) :

y adalah vektor output dari lapisan neuron tersebut.

σ adalah fungsi aktivasi yang diterapkan pada hasil *weighted sum*.

W adalah matriks bobot yang menghubungkan lapisan neuron sebelumnya dengan lapisan neuron saat ini.

x adalah vektor *input* dari lapisan neuron sebelumnya.

b adalah vektor bias yang ditambahkan ke hasil *weighted sum*.

Mean Square Error (MSE) merupakan suatu perhitungan yang digunakan dalam mengukur rata-rata dari kuadrat yang diperoleh antara selisih nilai yang diharapkan dengan nilai hasil keluaran dari prediksi. Semakin rendah nilai yang didapatkan dari perhitungan MSE, maka hasil prediksi yang diperoleh akan dianggap semakin akurat (Wiranto, 2023). Rumus MSE ditunjukkan pada persamaan (7) :

$$MSE = \sum \frac{(y_i - \hat{y}_i)^2}{n} \quad (7)$$

(Wiranto, 2023)

Keterangan persamaan (7):

n adalah ukuran data sampel

y_i adalah nilai aktual data i

\hat{y}_i adalah nilai prediksi data i

Nilai MAPE didapatkan dengan cara membagi hasil kesalahan absolut dengan nilai observasi (Habibi & Riksakomara, 2017). Semakin kecil nilai MAPE dari hasil komputasi prediksi maka semakin mendekati data aslinya. Satu model prediksi mempunyai kinerja yang sangat akurat jika nilai MAPE

dibawah 10% dan mempunyai kinerja baik jika nilai MAPE antara 10% dan 20% (Arwin Datumaya Wahyudi Sumari, 2020). Rumus MAPE ditunjukkan pada persamaan (8):

$$\frac{1}{n} \sum_{i=1}^n \left| \frac{A_i - F_i}{A_i} \right| \times 100 \quad (8)$$

(Habibi & Riksakomara, 2017)

Keterangan persamaan (8) :

n adalah ukuran data *sample*

A_i adalah nilai aktual data i

F_i adalah nilai prediksi data i

2.5. Pembahasan

Bagian pembahasan dari laporan penelitian ini akan membahas secara rinci hasil perhitungan data yang telah dilakukan menggunakan metode *Neural Network*.

2.6. Kesimpulan

Tujuan dari kesimpulan adalah untuk memberikan ringkasan yang jelas dan singkat tentang temuan penelitian mengenai estimasi tinggi muka air menggunakan metode *Neural Network*.

3. Hasil dan Pembahasan

Penelitian ini menggunakan data yang diperoleh dari sumber hidrologi.bbws-bsolo.net pada Pos Bojonegoro yang mencakup tinggi muka air, suhu, kelembaban. Data *training* yang digunakan dalam penelitian ini mencakup periode dari tahun 2021 sampai 2022 dengan 730 data, sementara data uji digunakan dalam penelitian 2023 selama 9 bulan dengan 273 data.

Tabel 1. Data Penelitian

Tanggal	Curah Hujan (mm)	Suhu (°C)	Kelembaban (%)	Tinggi Muka Air (mdpl)
01/01/2021	2,2	31	82	14,12
02/01/2021	13	32	82	13,80
03/01/2021	3	31	83	13,84
04/01/2021	0	32	84	13,18
.....
27/03/2023	0	37	90	6,80
28/09/2023	0	37,1	96	6,76
29/09/2023	0	37,4	95	6,76
30/09/2023	0	37,2	95	6,72

Pada Tabel 1 menjelaskan data tersebut mencakup informasi mengenai tanggal pengamatan, curah hujan, suhu udara, tingkat kelembapan, serta tinggi muka air di lokasi penelitian Pos Bojonegoro dengan total data sebanyak 1003.

A. Pengolahan Data Dengan Microsoft Excel

Pada pengolahan data menggunakan Excel hanya menggunakan data sebanyak 1003 terbagi menjadi data *training* mulai 01 Januari 2021 - 31 Desember 2022 dan data uji mulai 01 Januari 2023 - 30 September 2023. Optimasi pada Excel menggunakan GRG *Nonlinear* karena penggunaannya dalam masalah-masalah besar lebih teruji.

a) Data *Training*

Table 2. Data *Training* 2021 - 2022

Tanggal	Curah Hujan (mm)	Suhu (°C)	Kelembaban (%)	Tinggi Muka Air (mdpl)
01/01/2021	2,2	31	82	14,12
02/01/2021	13	32	82	13,80
03/01/2021	3	31	83	13,84

04/01/2021	0	32	84	13,18
.....
28/12/2022	5	30,1	95	11,96
29/12/2022	0	29,2	90	11,05
30/12/2022	9	29,5	80	11,80
31/12/2022	2,1	29,4	90	11,71

Pada Tabel 2 menjelaskan data tersebut mencakup informasi mengenai tanggal pengamatan, curah hujan, suhu udara, tingkat kelembapan, serta tinggi muka air di lokasi penelitian Pos Bojonegoro dengan total data *training* sebanyak 730.

1. *Weight* dan Bias

Table 3. *Weight* dan Bias

Neuron	w1	w2	w3	b1
1	2,02	0,00	0,00	1,00
2	3,79	0,00	0,00	1,50
3	7,39	0,00	0,00	2,00

Pada Tabel 3 menjelaskan bahwa 3 Neuron memiliki masing-masing nilai *weight* dan bias, (w1, w2, w3) adalah *weight* sesuai dengan jumlah dari variabel yaitu 3 variabel, dan b1 adalah nilai dari bias. *Weight* dan Bias ini disesuaikan selama proses pelatihan (*training*) jaringan untuk meminimalkan kesalahan (*error*) antara estimasi dan target yang sebenarnya.

2. *Neuron Weight Sum*

Table 4. *Neuron Weight Sum*

Tanggal	Neuron Weight Sum		
	1	2	3
01/01/2021	5,45	9,85	18,26
02/01/2021	27,31	50,81	98,08
03/01/2021	7,07	12,88	24,17
04/01/2021	1,00	1,50	2,00
.....
28/12/2022	11,12	20,47	38,96
29/12/2022	1,00	1,50	2,00
30/12/2022	19,21	35,64	68,52
31/12/2022	5,25	9,47	17,52

Pada Tabel 4 menjelaskan mengenai nilai neuron *weight* sum untuk tiga neuron yang berbeda, yaitu Neuron 1, Neuron 2, dan Neuron 3.

3. Max dan Min

Table 5. Max dan Min

Neuron Weight Sum	Max	Min
1	207,42	1,00
2	388,43	1,50
3	755,89	2,00

Pada Tabel 5 menjelaskan nilai maksimal dan minimal dengan mencari nilai tersebut dari 730 data pada setiap neuron weight sum.

4. *Normalized* dan *Sigmoid Transformed*

Table 6. Hasil Normalisasi Data [-1,1] dan Sigmoid [0,1]

Tanggal	Normalized			Sigmoid Transformed		
	1	2	3	1	2	3

01/01/2021	-	-	-	0,277506765	0,277506765	0,277506765
	0,956862745	0,956862745	0,956862745			
02/01/2021	-	-	-	0,32189035	0,32189035	0,32189035
	0,745098039	0,745098039	0,745098039			
03/01/2021	-	-	-	0,280662762	0,280662762	0,280662762
	0,941176471	0,941176471	0,941176471			
04/01/2021	-1	-1	-1	0,268941421	0,268941421	0,268941421
.....
28/12/2022	-	-	-	0,288647722	0,288647722	0,288647722
	0,901960784	0,901960784	0,901960784			
29/12/2022	-1	-1	-1	0,268941421	0,268941421	0,268941421
	-	-	-	0,305014978	0,305014978	0,305014978
30/12/2022	0,823529412	0,823529412	0,823529412			
	-	-	-	0,277113806	0,277113806	0,277113806
31/12/2022	0,958823529	0,958823529	0,958823529			

Pada Tabel 6 dalam normalisasi data diperlukan nilai pada setiap neuron weight sum dilakukan proses perhitungan dengan nilai max dan min yang sudah ada pada proses sebelumnya. Sigmoid transformed pada setiap neuron yang merupakan hasil perhitungan dengan nilai normalized.

5. Layer 1 Output dan Error

Table 7. Hasil Layer 1 *Output* dan *Error*

Tanggal	Sigmoid Transformed			Output1	Error1
	1	2	3		
01/01/2021	0,277506765	0,277506765	0,277506765	9,45	21,82
02/01/2021	0,32189035	0,32189035	0,32189035	9,80	16,00
03/01/2021	0,280662762	0,280662762	0,280662762	9,48	19,07
04/01/2021	0,268941421	0,268941421	0,268941421	9,38	14,40
.....
28/12/2022	0,288647722	0,288647722	0,288647722	9,54	5,87
29/12/2022	0,268941421	0,268941421	0,268941421	9,38	2,77
30/12/2022	0,305014978	0,305014978	0,305014978	9,67	4,53
31/12/2022	0,277113806	0,277113806	0,277113806	9,45	5,12
Error Sum					2977,72
Wo1	1,12	2,56	4,18		
Bo1	7,27	1,50	2,00		

Pada Tabel 7 menjelaskan proses perhitungan untuk *output* yaitu menggunakan *sigmoid transformed* dengan nilai W_{o1} dan B_{o1} sebagai *Weight* dan *Bias* untuk lapisan *output*. Untuk memperoleh hasil error diperlukan perhitungan selisih antara hasil *output* dengan nilai tinggi muka air (dikuadratkan). Hasil error sum adalah 2977,72.

6. 2nd Layer *Weight* dan *Bias*

Table 8. 2nd Layer *Weight* dan *Bias*

Neuron	w21	w22	w23	b21
1	1,10	1,10	1,10	0,00
2	1,49	1,49	1,49	0,00
3	1,34	1,34	1,34	0,00

Pada Tabel 8 menjelaskan bahwa 3 Neuron memiliki masing-masing nilai *weight* dan *bias*, (w_{21} , w_{22} , w_{23}) adalah *weight* pada layer 2 sesuai dengan jumlah dari variabel yaitu 3 variabel, dan b_{21} adalah nilai dari *bias* pada layer 2. *Weight* dan *Bias* ini disesuaikan selama proses pelatihan (*training*) jaringan untuk meminimalkan kesalahan (*error*) antara estimasi dan target yang sebenarnya.

7. 2nd Hidden Layer Aggregation

Dengan menggunakan dua *hidden layers*, model memiliki lebih banyak kapasitas untuk mempelajari representasi yang lebih abstrak dan hierarkis dari data, yang dapat membantu meningkatkan pemahaman dan analisis data.

Table 9. 2nd Hidden Layer Aggregation

Tanggal	2 nd Hidden Layer Aggregation		
	1	2	3
01/01/2021	0,92	1,24	1,11
02/01/2021	1,06	1,44	1,29
03/01/2021	0,93	1,25	1,13
04/01/2021	0,89	1,20	1,08
.....
28/12/2022	0,95	1,29	1,16
29/12/2022	0,89	1,20	1,08
30/12/2022	1,01	1,36	1,22
31/12/2022	0,91	1,24	1,11

Pada Tabel 9 menjelaskan hasil dari 2nd hidden layer aggregation yang diperoleh dari proses perhitungan menggunakan sigmoid transformed pada Tabel 6 dengan menggunakan rumus seperti menghitung neuron weight sum. Pada hasil 2nd hidden layer aggregation yang dapat dilihat pada Tabel 9 diberikan fungsi aktivasi sigmoid sehingga didapatkan hasil seperti pada tabel diatas.

8. 2nd Layer Output dan Error

Table 10. 2nd Layer Output dan Error

Tanggal	2 nd Hidden Layer Transformed			Output2	Error2
	1	2	3		
01/01/2021	0,714185242	0,775393056	0,752677478	9,49	21,50
02/01/2021	0,743123816	0,808018026	0,784306992	9,84	15,71
03/01/2021	0,716306467	0,777837592	0,755026117	9,51	18,76
04/01/2021	0,708380607	0,768662691	0,746227474	9,41	14,18
.....
28/12/2022	0,721630662	0,783937316	0,760900805	9,58	5,69
29/12/2022	0,708380607	0,768662691	0,746227474	9,41	2,68
30/12/2022	0,732349956	0,796058155	0,772637711	9,71	4,36
31/12/2022	0,713920459	0,775087348	0,752383988	9,48	4,97
Error Sum					2957,76
Wo21	1,89	3,71	5,54		
Bo21	1,09	1,50	2,00		

Pada Tabel 10 menjelaskan proses perhitungan untuk output yaitu menggunakan 2nd hidden layer transformed yang merupakan hasil perhitungan pada Tabel 10 dengan nilai Wo1 dan Bo1 sebagai Weight dan Bias untuk lapisan output. Untuk memperoleh hasil error diperlukan perhitungan selisih antara hasil output dengan nilai tinggi muka air (dikuadratkan). Hasil error sum adalah 2957,76.

9. MAPE (Mean Absolute Percentage Error)

Table 11. Hasil Mean Absolute Percentage Error

Tanggal	Output2	ABS Error2 (%)	MAPE (%)
01/01/2021	21,50	32,83368196	18,16
02/01/2021	15,71	28,72206625	
03/01/2021	18,76	31,28972925	
04/01/2021	14,18	28,56946966	

.....
28/12/2022	5,69	19,94319499	
29/12/2022	2,68	14,80390004	
30/12/2022	4,36	17,70519859	
31/12/2022	4,97	19,03170198	

Pada Tabel 11 dapat dijelaskan bahwa hasil *mean absolute percentage error* yaitu sebesar 18,16%. Didapatkan dari hasil perhitungan menggunakan nilai rata-rata *absolute error* pada 730 data.

b) Data Uji

Data yang digunakan untuk pengujian pada perhitungan manual menggunakan *excel* terdiri dari 31 sampel data dari 1 Januari hingga 31 Januari 2023 yang diperoleh dari hidrologi.bbws-bsolo.net pada Pos Bojonegoro. Data yang digunakan dalam pengujian mencakup tinggi muka air, curah hujan, suhu dan kelembapan. Data uji dapat dilihat pada Tabel 12 dibawah ini :

Table 12. Data Uji Januari 2023

Tanggal	Curah Hujan	Suhu	Kelembaban	Tinggi Muka Air
	(mm)	(°C)	(%)	(mdpl)
01/01/2023	0	31	91	11,58
02/01/2023	0	32,1	95	9,80
03/01/2023	3	32,1	91	9,94
04/01/2023	0	32,3	95	10,20
.....
28/01/2023	2,2	31,2	96	12,03
29/01/2023	39	32	95	12,18
30/01/2023	0	33	91	11,30
31/01/2023	21,4	32	95	10,53

Pada Tabel 12 menjelaskan data uji yang mencakup Curah Hujan dengan satuan (milimeter), Suhu dengan satuan (celcius), Kelembaban dengan satuan (persen) dan Tinggi Muka Air dengan satuan (mdpl/meter diatas permukaan laut).

1. *Neuron Weight Sum* dan Normalisasi

Dalam mencari hasil neuron *weight sum* dan *normalized*, dilakukan proses seperti pada data *training*. Karena penelitian ini menggunakan Microsoft Excel, jadi hanya memakai *auto fill* sebanyak 31 data sesuai banyaknya hari pada bulan Januari 2023. Untuk hasilnya dapat dilihat pada Tabel 13 dibawah ini :

Table 13. Neuron Weight Sum dan Normalisasi

Tanggal	Neuron Weight Sum			Normalisasi		
	1	2	3	1	2	3
01/01/2023	1,00	1,50	2,00	-1	-1	-1
02/01/2023	1,00	1,50	2,00	-1	-1	-1
03/01/2023	7,07	12,88	24,17	-0,941176471	-0,941176471	-0,941176471
04/01/2023	1,00	1,50	2,00	-1	-1	-1
.....
28/01/2023	5,45	9,85	18,26	-0,956862745	-0,956862745	-0,956862745
29/01/2023	79,92	149,44	290,25	-0,235294118	-0,235294118	-0,235294118
30/01/2023	1,00	1,50	2,00	-1	-1	-1
31/01/2023	44,31	82,68	160,17	-0,580392157	-0,580392157	-0,580392157

Pada Tabel 13 menjelaskan hasil dari neuron *weight* sum dan normalisasi yang didapatkan dari perhitungan *input* 3 variabel. Pada normalisasi yang digunakan yaitu pada rentang (-1,1).

2. Sigmoid Transformed, Output1, Error1, MAPE, MSE

Table 14. Hasil *Sigmoid Transformed*, Output1, Error1, MAPE, MSE

Tanggal	Sigmoid Transformed			Output1	Error1	MAPE	MSE
	1	2	3				
01/01/2023	0,268941	0,268941	0,268941	9,38	4,83	9,84	1,36
02/01/2023	0,268941	0,268941	0,268941	9,38	0,18		
03/01/2023	0,280663	0,280663	0,280663	9,48	0,21		
04/01/2023	0,268941	0,268941	0,268941	9,38	0,67		
...
28/01/2023	0,277507	0,277507	0,277507	9,45	6,65		
29/01/2023	0,441446	0,441446	0,441446	10,74	2,07		
30/01/2023	0,268941	0,268941	0,268941	9,38	3,67		
31/01/2023	0,358842	0,358842	0,358842	10,09	0,19		

Pada Tabel 14 menjelaskan hasil dari *sigmoid transformed* yang menggunakan 3 neuron, hasil dari *output* untuk layer 1, dan hasil *error* untuk layer 1. Hasil MAPE 9,84% dan MSE 1,36. Karena pada training model error pada hidden layer 2 lebih kecil dari hidden layer 1, maka proses pada data uji cukup pada hidden layer 2.

B. Pengolahan Data Dengan *Python*

Pada pengolahan data dengan *Python* menggunakan data *training* mulai 01 Januari 2021-31 Desember 2022 dan data uji mulai 01 Januari 2023-30 September 2023. Optimasi pada *Python* menggunakan Optimasi Adam karena penggunaannya dalam masalah-masalah besar lebih baik dan menghasilkan hasil yang lebih akurat dan efisien dalam waktu yang lebih singkat.

```
import pandas as pd
import numpy as np
import tensorflow as tf
from datetime import date, timedelta
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt

# Membaca data pelatihan dari Excel
data_training = pd.read_excel('Data 21-22.xlsx')

# Menyiapkan data pelatihan
X_train = data_training[['Curah Hujan', 'Suhu', 'Kelembaban']].values
y_train = data_training['Tinggi Muka Air'].values

# Normalisasi data menggunakan Min-Max Scaling
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()
X_train_scaled = scaler_X.fit_transform(X_train)
y_train_scaled = scaler_y.fit_transform(y_train.reshape(-1, 1))

# Membangun model neural network dengan 3 neuron dan aktivasi sigmoid
model = tf.keras.Sequential([
    tf.keras.layers.InputLayer(input_shape=(3,)), # Input layer
    tf.keras.layers.Dense(3, activation='sigmoid'), # Hidden layer
    tf.keras.layers.Dense(1, activation='sigmoid') # Output layer
])

model.compile(optimizer='adam', loss='mean_squared_error')

# Melatih model dengan data pelatihan
model.fit(X_train_scaled, y_train_scaled, epochs=100, verbose=0)

# Membaca data sebenarnya untuk 9 bulan di 2023 dari Excel
data_real_2023 = pd.read_excel('Data 23.xlsx')

# Menyiapkan data sebenarnya
X_real_2023 = data_real_2023[['Curah Hujan', 'Suhu', 'Kelembaban']].values
y_real_2023 = data_real_2023['Tinggi Muka Air'].values
```

Gambar 3. Program *Python*

```

# Normalisasi data sebenarnya menggunakan Min-Max Scaling
X_real_2023_scaled = scaler_X.transform(X_real_2023)
y_real_2023_scaled = scaler_y.transform(y_real_2023.reshape(-1, 1))

# Melakukan estimasi pada data sebenarnya
estimasi_scaled = model.predict(X_real_2023_scaled)
estimasi = scaler_y.inverse_transform(estimasi_scaled)

# Menghitung Mean Squared Error (MSE)
mse = mean_squared_error(y_real_2023, estimasi)
print("Mean Squared Error (MSE) pada data sebenarnya 2023:", mse)

# Menghitung Mean Absolute Percentage Error (MAPE)
def mape(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

mape_score = mape(y_real_2023, estimasi)
print("Mean Absolute Percentage Error (MAPE) pada data sebenarnya 2023:", mape_score)

# Mencetak hasil estimasi dan data sebenarnya beserta informasi curah hujan, suhu, dan kelembapan
tanggal_awal = date(2023, 1, 1) # Tanggal awal (1 Januari 2023)
print("Curah Hujan Suhu Kelembapan Data Aktual Data Estimasi")
for i, (pred, actual, rain, temp, humidity) in enumerate(zip(estimasi, y_real_2023, X_real_2023[:, 0], X_real_2023[:, 1], X_real_2023[:, 2])):
    tgl = tanggal_awal + timedelta(days=i)
    print(f"{rain:.2f}mm {temp:.2f}°C {humidity:.2f}% {actual:.2f} mdpl {pred[0]:.2f} mdpl")

# Plot hasil estimasi dan data sebenarnya
plt.figure(figsize=(12, 6))
plt.plot(y_real_2023, label='Data Sebenarnya')
plt.plot(estimasi, label='Estimasi')
plt.legend()
plt.title('Estimasi Tinggi Muka Air vs. Data Sebenarnya')
plt.xlabel('Hari')
plt.ylabel('Tinggi Muka Air')
plt.show()

```

Gambar 4. Lanjutan Program Python

Pada Gambar 3 dan 4 melakukan estimasi selama 9 bulan pada tahun 2023 menggunakan program *Python*, nantinya menampilkan data aktual dan nilai estimasi, menampilkan nilai MAPE dan MSE, serta menampilkan grafik.

```

9/9 [=====] - 0s 6ms/step
Mean Squared Error (MSE) pada data sebenarnya 2023: 3.4424465503109847
Mean Absolute Percentage Error (MAPE) pada data sebenarnya 2023: 17.875026297198314

```

Gambar 5. Hasil Mean Squared Error dan Mean Absolute Percentage Error

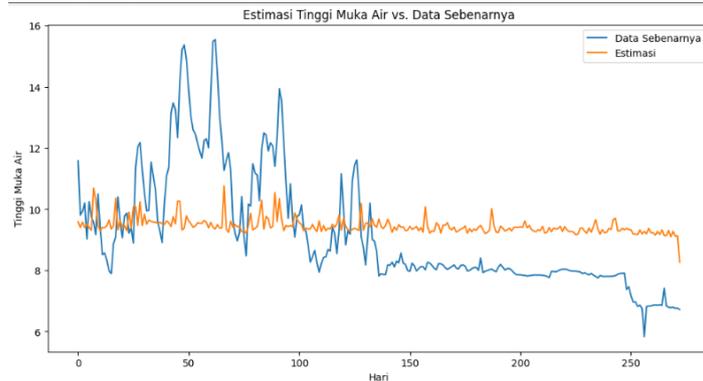
Pada Gambar 5 nilai MSE adalah 3.4424465503109847, yang menunjukkan bahwa estimasi model memiliki kesalahan rata-rata sebesar 3.4424465503109847. Semakin mendekati nol, semakin baik. MAPE mengukur tingkat kesalahan estimasi dalam persentase dari data sebenarnya. Nilai MAPE adalah 17.87502697198314%, yang berarti estimasi model memiliki kesalahan rata-rata sebesar 17.87502697198314% dari data sebenarnya. Semakin mendekati nol, semakin baik.

Table 15. Hasil Estimasi dan Data Aktual Bulan Januari-September 2023

Tanggal	Data Aktual	Data Estimasi
2023-01-01	9.59 mdpl	11.58 mdpl
2023-01-02	9.47 mdpl	9.80 mdpl
2023-01-03	9.58 mdpl	9.94 mdpl
2023-01-04	9.46 mdpl	10.20 mdpl
2023-01-05	9.53 mdpl	9.02 mdpl
-----	-----	-----
2023-09-26	9.15 mdpl	6.78 mdpl
2023-09-27	9.26 mdpl	6.80 mdpl
2023-09-28	9.18 mdpl	6.76 mdpl
2023-09-29	9.18 mdpl	6.76 mdpl
2023-09-30	7.61 mdpl	6.72 mdpl

Pada Tabel 15 Kolom pertama adalah tanggal yang dimulai dari 1 Januari 2023 hingga 30 September 2023. Kolom kedua adalah hasil estimasi tinggi muka air selama 9 bulan pada 2023. Nilai ini diberikan dalam satuan "mdpl" (meter di atas permukaan laut). Kolom ketiga adalah data tinggi muka air

sebenarnya atau data aktual. Nilai ini juga disajikan dalam satuan "mdpl". Data aktual ini merupakan data yang sebenarnya terjadi pada setiap hari selama 9 bulan pada 2023.



Gambar 6. Grafik Estimasi dan Data Sebenarnya

Pada Gambar 6 menampilkan grafik estimasi tinggi muka air dan data sebenarnya dari 1 Januari sampai 30 September 2023. Dengan data sebenarnya ditunjukkan dengan grafik berwarna biru, dan nilai estimasi ditunjukkan dengan grafik berwarna oranye.

4. Kesimpulan

Pada penelitian ini, metode neural *network* dengan menggunakan optimasi GRG *Nonlinear* dan Adam yang telah digunakan mampu memberikan estimasi tinggi muka air pada wilayah hilir Pos Bojonegoro selama 9 bulan yaitu dari 01 Januari sampai 30 September 2023. Estimasi ini didasarkan pada data pelatihan sebelumnya yang mencakup curah hujan, suhu, dan kelembapan.

Pada perhitungan manual menggunakan Excel dengan optimasi GRG *Nonlinear* untuk estimasi di bulan Januari 2023, jika hasil *error* lebih kecil pada model dua *hidden layer* dibandingkan satu *hidden layer*, ini menunjukkan bahwa dua *hidden layer* sudah cukup baik. Metode *neural network* yang digunakan memberikan hasil *Mean Squared Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE) terhadap data sebenarnya. Dari hasil evaluasi, ditemukan bahwa nilai MSE 1,29 dan MAPE 9,65%. Sedangkan perhitungan menggunakan *python* dengan optimasi Adam untuk estimasi bulan Januari-September 2023 memberikan hasil *Mean Squared Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE) terhadap data sebenarnya. Dari hasil evaluasi ditemukan bahwa nilai MSE 3.4424465503109847 dan MAPE 17.875026297198314%. Nilai MSE yang relatif rendah menunjukkan bahwa model memiliki tingkat ketepatan yang baik dalam memprediksi tinggi muka air, sementara nilai MAPE yang baik menunjukkan tingkat kesalahan rendah.

DAFTAR PUSTAKA

- Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning. *Karya Ilmiah Mahasiswa Bertauhid (KARIMAH TAUHID)*, 2(1), 1–6.
- Aprianto, R., Ayu, P., & Puspitasari, D. (2020). Prediksi Curah Hujan Bulanan Tahun 2020 Kabupaten Sumbawa Menggunakan Artificial Neural Network (ANN) Back Propagation. *Prosiding Seminar Nasional IPPeMas 2020*, 1(1), 622–628.
- Arwin Datumaya Wahyudi Sumari, Muhammad Bisri Musthafa, Ngatmari, & Dimas Rossiawan Hendra Putra. (2020). Comparative Performance of Prediction Methods for Digital Wallet Transactions in the Pandemic Period. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(4), 642–647.
- Du, K. L., & Swamy, M. N. S. (2006). Neural networks in a softcomputing framework. In *Neural Networks in a Softcomputing Framework*.
- Habibi, M. Y., & Riksakomara, E. (2017). Peramalan Harga Garam Konsumsi Menggunakan Artificial Neural Network Feedforward-Backpropagation (Studi Kasus : PT. Garam Mas, Rembang, Jawa

- Tengah). *Jurnal Teknik ITS*, 6(2).
- Hamm, L., Brorsen, W., & Hagan, M. (2007). Comparison of Stochastic Global Optimization Methods to Estimate Neural Network Weights. *Neural Processing Letters*, 26, 145–158.
- Hartini, E. (2017). *Hidrologi dan Hidrolika Terapan* (1st ed., Vol. 1). Fakultas Kesehatan Program Studi Kesehatan Lingkungan Universitas Dian Nuswantoro.
- Hukubun, A. J. E. (2022). Neural Network. *Jurusan Pendidikan Teknologi Informasi Dan Komunikasi Universitas Negeri Manado*, 1(1), 1–6.
- Hung, N. Q., Babel, M. S., Weesakul, S., & Tripathi, N. K. (2009). An artificial neural network model for rainfall forecasting in Bangkok, Thailand. *Hydrology and Earth System Sciences*, 13(8), 1413–1425.
- Jayadianti, H., Cahyadi, T. A., Amri, N. A., & Pitayandanu, M. F. (2020). Metode Komparasi Artificial Neural Network Pada Prediksi Curah Hujan - Literature Review. *Jurnal Tekno Insentif*, 14(2), 48–53.
- Kalantar, B., Ueda, N., Saeidi, V., Janizadeh, S., Shabani, F., Ahmadi, K., & Shabani, F. (2021). Deep neural network utilizing remote sensing datasets for flood hazard susceptibility mapping in Brisbane, Australia. *Remote Sensing*, 13(13).
- Lakitan, B. (2002). *Dasar-Dasar Klimatologi*.
- Lee, J., Kim, C. G., Lee, J. E., Kim, N. W., & Kim, H. (2018). Application of artificial neural networks to rainfall forecasting in the Geum River Basin, Korea. *Water (Switzerland)*, 10(10).
- Mangunwiyoto, W., & Harjono. (2004). *Pokok-Pokok Fisika SMP: Untuk Kelas IX Widagdo Mangunwiyoto* (1st ed.). Erlangga.
- Nawiyanto, & Arianto, D. E. (2022). Banjir Bengawan Solo dan Pengaruhnya Bagi Kehidupan Masyarakat di Kabupaten Bojonegoro Tahun 2007 – 2008. *Jurnal Historia*, 4(2), 180–194.
- Prasetyo, D. N. H., & Hayati, R. (2019). Peningkatan Pengetahuan Kesiapsiagaan Banjir Pengurus Karang Taruna dengan Metode Diskusi Berbantuan Media Audio Visual di Kelurahan Sawah Besar Tahun 2018. *Edu Geography*, 7(3), 222–231.
- Roy. (2022). Apa Itu Perubahan Iklim, Penyebab, Dampak & Cara Mengatasinya.
- Rustika, I., Margana, D. B., & Putro, T. Y. (2018). Sistem Pengukuran dan Pemantauan Ketinggian dan Debit Air Berbasis Mikrokontroler untuk Mendeteksi Potensi Banjir. *Industrial Research Workshop and National Seminar*, 1(1), 57–64.
- Ruswanti, D. (2020). Pengukuran Performa Support Vector Machine Dan Neural Network Dalam Meramalkan Tingkat Curah Hujan. *Gaung Informatika*, 13(1), 66–75.
- Salsabila, A., & Nugraheni, I. L. (2020). Pengantar Hidrologi. *Pengantar Hidrologi*, 134.
- Sharma, S., Sharma, S., & Athaiya, A. (2020). Activation Functions in Neural Networks. *International Journal of Engineering Applied Sciences and Technology*, 04(12), 310–316.
- Sit, M., & Demir, I. (2019). *Decentralized Flood Forecasting Using Deep Neural Networks*.
- Suhartanto, E., Cahya, E. N., & Iul Maknun. (2019). Analisa Limpasan Berdasarkan Curah Hujan Menggunakan Model Artificial Neural Network (ANN) Di Sub DAS Brantas Hulu. *Jurnal Teknik Pengairan Universitas Brawijaya*, 10(2), 134–144.
- Wang, H., Czerminski, R., & Jamieson, A. C. (2021). Neural Networks and Deep Learning. *The Machine Age of Customer Insight*, 91–101.
- Wiranto, A. R., Setiawan, E., Nuryaman, A., & Usman, M. (2023). Implementasi Metode Backpropagation Neural Network Dalam Meramalkan Tingkat Inflasi Di Indonesia. *MATHunesa: Jurnal Ilmiah Matematika*, 11(1), 8–16.