



Implementasi Algoritma *Backpropagation* Untuk Memprediksi Jumlah Jiwa Terdampak Banjir Di DKI Jakarta

Mira Kartika¹, Kemal Ade Sekarwati²

^{1,2}Departemen Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma

^{1,2}Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

¹ mirakrtika08@gmail.com, ² ade@staff.gunadarma.ac.id

DOI:

Info Artikel

Sejarah Artikel:

Disubmit 2 Desember 2024

Direvisi 9 Desember 2024

Disetujui 17 Agustus 2024

Keywords:

Backpropagation, DKI Jakarta, Flood, Forecasting, Number of Souls Affected;

Abstrak

Banjir merupakan bencana alam yang sering terjadi di wilayah DKI Jakarta. Pemerintah DKI Jakarta perlu mengantisipasi dampak bencana banjir dengan melakukan estimasi jumlah penduduk yang terkena banjir. Jumlah penduduk yang terkena banjir yang tidak menentu setiap bulannya dapat diprediksi untuk masa yang akan datang. Ada banyak cara yang dapat dilakukan untuk memprediksi jumlah penduduk yang terkena banjir, salah satunya adalah dengan metode jaringan syaraf tiruan. Salah satu algoritma pembelajaran pada jaringan syaraf tiruan adalah algoritma *backpropagation*. Penelitian ini menerapkan metode jaringan syaraf tiruan dengan algoritma *backpropagation* untuk memprediksi jumlah penduduk yang terkena banjir di DKI Jakarta. Pada penelitian ini, proses pelatihan dilakukan sebanyak 100 kali pada masing-masing arsitektur jaringan (12-10-1, 12-12-1, 12-14-1) dengan beberapa parameter seperti epoch, konstanta momentum, dan learning rate. Hasil terbaik pada proses pelatihan dilakukan pada proses pengujian untuk menguji jaringan. Pada proses pengujian, hasil terbaik adalah arsitektur 12-10-1 dengan tingkat akurasi 98,704%. Berdasarkan hasil tersebut dapat dikatakan bahwa jaringan ini dapat melakukan prediksi dengan baik dan dapat diimplementasikan untuk peramalan jumlah jiwa yang terkena dampak banjir di DKI Jakarta.

Kata kunci: *Backpropagation, DKI Jakarta, Banjir, Peramalan, Jumlah Jiwa yang Terdampak.*

Abstract

Floods are natural disasters that often occur in the DKI Jakarta area. DKI Jakarta government needs to anticipate the impact of the flood disaster by estimating the number of people affected by the flood. The number of people affected by floods that are uncertain every month can be predicted for the future. There are many ways that can be predict the number of people affected by floods, one of them is artificial neural network method. One of learning algorithms in artificial neural networks is *backpropagation* algorithm. This research applies an artificial neural network method with *backpropagation* algorithm to predict the number of people affected by floods in DKI Jakarta. In this research, training process was carried out 100 times on each network architecture (12-10-1, 12-12-1, 12-14-1) with several parameters such as epoch, momentum constant, and learning rate. The best results in the training process are carried out to testing process to test the network. In the testing process, the best results are 12-10-1 architecture with an accuracy rate 98.704%. Based on these results, it can be said that this network can predict well and can be implemented for forecasting the number of people affected by floods in DKI Jakarta.

Keywords: *Backpropagation, DKI Jakarta, Flood, Forecasting, Number of Souls Affected.*

1. Pendahuluan

Banjir adalah suatu peristiwa akibat air yang jatuh ke tanah bertumpuk dan tidak dapat ditampung oleh tanah [6]. Peristiwa ini terjadi karena air jatuh ke dataran yang tidak memiliki daerah tangkapan atau tanah-tanah diperkotaan jenuh air. Banjir sering melanda perkotaan seperti di wilayah DKI Jakarta. Badan Nasional Penanggulangan Bencana (BNBP) menyatakan bahwa banjir menjadi bencana alam yang paling mematikan dari awal Januari 2020 sampai Agustus 2020. BNBP mencatat lebih dari 100 jiwa meninggal akibat banjir dan 17 jiwa menghilang. Banjir juga menyebabkan kelumpuhan ekonomi, sektor perumahan, dan kerusakan lingkungan. DKI Jakarta merupakan wilayah yang sering terjadi banjir. Faktor-faktor yang menyebabkan wilayah DKI Jakarta sering mengalami banjir adalah wilayah DKI Jakarta memiliki ketinggian kurang dari 10 mdpl, dinamika dan pembangunan perkotaan, demografi perkotaan, dan tata guna lahan serta alih fungsi lahan.

Backpropagation merupakan salah satu algoritma pembelajaran pada jaringan saraf tiruan. *Backpropagation* melakukan proses pembelajaran dengan menyesuaikan bobot-bobot yang ada di jaringan saraf tiruan dengan arah mundur dan berdasarkan nilai *error* selama proses pembelajaran [1]. MATLAB merupakan program komputer dengan lingkungan yang nyaman untuk mengerjakan banyak perhitungan dan menyediakan berbagai alat untuk mengimplementasikan berbagai metode numeris [4].

Beberapa penelitian telah membuktikan bahwa algoritma *backpropagation* berhasil melakukan peramalan. Pada tahun 2016, Triyono, Santoso, dan Pranowo melakukan penelitian untuk meramalkan harga saham (IHSG) menggunakan metode jaringan syaraf

tiruan *backpropagation* [3]. Hasil penelitian ini menunjukkan *backpropagation* berhasil melakukan peramalan dengan menghasilkan tingkat keakuratan sebesar 99,98% dan MSE sebesar 0,9915%. Pada tahun 2017, Eko Riyanto juga melakukan penelitian yang bertujuan untuk meramalkan harga saham dengan menggunakan algoritma *backpropagation* [2]. Penelitian ini berhasil dilakukan dan menunjukkan tingkat keakuratan sebesar 98%. Musli, Eka, Dewi, dan Yuhandri juga melakukan penelitian pada tahun 2018 yang berjudul Peramalan Penjualan pada Toko Retail Menggunakan Algoritma *Backpropagation Neural Network* [5]. Hasil prediksi yang didapat pada penelitian ini memiliki total kesalahan *error* sebesar 3,57%.

Berdasarkan uraian tersebut, penelitian ini bertujuan untuk membuat sebuah model prediksi jumlah jiwa yang terdampak banjir di wilayah DKI Jakarta dengan menggunakan algoritma *Backpropagation* pada metode Jaringan Saraf Tiruan (JST).

2. Metodologi

Metode penelitian yang digunakan untuk menyelesaikan masalah adalah *Knowledge Discovery in Databases* (KDD). Proses KDD dibagi dalam beberapa tahap, yaitu :

2.1. Data Selection.

Tahap *data selection* merupakan tahap pencarian data yang dibutuhkan selama penelitian. Data didapatkan melalui *website* resmi pemerintah DKI Jakarta seperti pada Gambar 3.2. Alamat website tersebut adalah <https://data.jakarta.go.id/>. Data yang diambil dimulai dari Januari 2016 sampai dengan Desember 2020. Kolom-kolom yang terpilih untuk digunakan adalah tahun, bulan, dan jiwa terdampak. Kolom tahun dan bulan digunakan untuk keterangan waktu dari setiap data, sedangkan kolom jiwa terdampak

merupakan data jumlah jiwa terdampak banjir di wilayah DKI Jakarta. Setelah

dilakukan seleksi data, maka data dapat dilihat pada Tabel 1

Tabel 1. Data Asli Jumlah Jiwa Terdampak Banjir di DKI Jakarta

Data Jiwa yang Terdampak Banjir di DKI Jakarta												
Bln/Thn	Jan	Feb	Mar	Apr	Mei	Juni	Juli	Agust	Sept	Okt	Nov	Des
2016	3853	70218	51234	27607	6271	6025	8068	31921	15331	3840	8209	n.a
2017	n.a	19325	10166	2131	3420	4547	705	n.a	n.a	n.a	6505	404
2018	0	19456	1000	988	716	1068	0	0	0	1001	7597	877
2019	453	1186	1010	8851	355	0	0	0	0	1203	0	1601
2020	83406	43464	9687	2966	3117	320	0	0	1300	6250	0	827

2.2 Pre-processing

a. Pengolahan Missing Value

Data mentah yang hilang pada tabel 1 digambarkan dengan huruf “n.a”. data tersebut perlu diubah dengan rata-rata dari keseluruhan data atau biasa disebut *mean*. Rumus *mean* dapat dilihat pada persamaan (1).

$$\bar{x} = \frac{\sum fi \cdot \sum xi}{\sum fi} \quad (1)$$

Keterangan :

\bar{x} : nilai rata-rata

fi : nilai tengah kelompok data ke-i

xi : frekuensi kelompok data ke-i

Rumus *mean* pada persamaan 1 diaplikasikan pada tabel 1, maka hasil perhitungan adalah :

$$\bar{x} = \frac{3853 + 70218 + 51234 + 27607 + \dots + 827}{60}$$

$$\bar{x} = 7974,65$$

Rata-rata dari Tabel 1 adalah 7974,65. Selanjutnya, nilai tersebut dibulatkan menjadi 7975 dan dimasukkan ke dalam tabel untuk menggantikan data atau nilai yang hilang.

b. Normalisasi Data

Seluruh data dinormalisasi dengan cara diubah menjadi fungsi aktivasi. Fungsi aktivasi yang digunakan yaitu fungsi aktivasi *Sigmoid Biner*. Fungsi aktivasi *Sigmoid Biner* akan mengubah data menjadi nilai dengan interval 0,1 sampai 0,9. Interval tersebut didapat dari persamaan 2.

$$x' = \frac{0,8(x-b)}{(a-b)} + 0,1 \quad (2)$$

Keterangan :

x' = data hasil normalisasi

x = data asli / data awal

a = nilai maksimum data asli

b = nilai minimum data asli

Sebagai contoh data asli jumlah jiwa terdampak banjir pada bulan November 2018. Rincian perhitungan dari normalisasi tersebut dapat dilihat sebagai berikut.

Diketahui :

$$x = 7597$$

$$a = 83406$$

$$b = 0$$

Ditanyakan : x'

Jawab :

$$x' = \frac{0,8(x-b)}{(a-b)} + 0,1$$

$$x' = \frac{0,8(7597-0)}{(83406-0)} + 0,1$$

$$x' = 0,17287$$

Seluruh data yang sudah dinormalisasikan dapat dilihat pada Tabel 2.

Tabel 2. Data Jumlah Jiwa Terdampak Banjir Hasil Normalisasi

Bulan	Jumlah Jiwa Terdampak				
	2016	2017	2018	2019	2020
Jan	0,137	0,176	0,1	0,104	0,9
Feb	0,774	0,285	0,287	0,111	0,517
Mar	0,591	0,198	0,11	0,11	0,193
Apr	0,365	0,12	0,109	0,185	0,128
Mei	0,16	0,133	0,107	0,103	0,13
Juni	0,158	0,144	0,11	0,1	0,103
Juli	0,177	0,107	0,1	0,1	0,1

Agust	0,406	0,176	0,1	0,1	0,1
Sept	0,247	0,176	0,1	0,1	0,112
Okt	0,137	0,176	0,11	0,112	0,16
Nov	0,179	0,162	0,173	0,1	0,1
Des	0,176	0,104	0,108	0,115	0,108

c. Pembentukan Data Latih

Data yang sudah dinormalisasi digunakan sebagai data latih dan data uji. Data latih digunakan untuk proses *training*. Data latih berisi data dari bulan Januari 2016 sampai dengan Desember 2018. Data latih disajikan dalam bentuk 24 pola. Masing-masing pola memiliki 13 data. Data tersebut terdiri dari x1 sampai dengan x12 dan target.

Contoh pola ke-1 yaitu Januari 2016 sampai dengan Desember 2016 (x1 sampai x12) dan bulan ke-13 yaitu Januari 2017 sebagai data target. Hasil pembentukan data latih dapat dilihat pada tabel 3.

Tabel 3. Data Latih

Pola	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	Target
1	0,137	0,774	0,591	0,365	0,16	0,158	0,177	0,406	0,247	0,137	0,179	0,176	0,176
2	0,774	0,591	0,365	0,16	0,158	0,177	0,406	0,247	0,137	0,179	0,176	0,176	0,285
3	0,591	0,365	0,16	0,158	0,177	0,406	0,247	0,137	0,179	0,176	0,176	0,285	0,198
4	0,365	0,16	0,158	0,177	0,406	0,247	0,137	0,179	0,176	0,176	0,285	0,198	0,12
5	0,16	0,158	0,177	0,406	0,247	0,137	0,179	0,176	0,176	0,285	0,198	0,12	0,133
6	0,158	0,177	0,406	0,247	0,137	0,179	0,176	0,176	0,285	0,198	0,12	0,133	0,144
7	0,177	0,406	0,247	0,137	0,179	0,176	0,176	0,285	0,198	0,12	0,133	0,144	0,107
8	0,406	0,247	0,137	0,179	0,176	0,176	0,285	0,198	0,12	0,133	0,144	0,107	0,176
9	0,247	0,137	0,179	0,176	0,176	0,285	0,198	0,12	0,133	0,144	0,107	0,176	0,176
10	0,137	0,179	0,176	0,176	0,285	0,198	0,12	0,133	0,144	0,107	0,176	0,176	0,176
11	0,179	0,176	0,176	0,285	0,198	0,12	0,133	0,144	0,107	0,176	0,176	0,176	0,162
12	0,176	0,176	0,285	0,198	0,12	0,133	0,144	0,107	0,176	0,176	0,176	0,162	0,104
13	0,176	0,285	0,198	0,12	0,133	0,144	0,107	0,176	0,176	0,176	0,162	0,104	0,1
14	0,285	0,198	0,12	0,133	0,144	0,107	0,176	0,176	0,176	0,162	0,104	0,1	0,287
15	0,198	0,12	0,133	0,144	0,107	0,176	0,176	0,176	0,162	0,104	0,1	0,287	0,11
16	0,12	0,133	0,144	0,107	0,176	0,176	0,176	0,162	0,104	0,1	0,287	0,11	0,109
17	0,133	0,144	0,107	0,176	0,176	0,176	0,162	0,104	0,1	0,287	0,11	0,109	0,107
18	0,144	0,107	0,176	0,176	0,176	0,162	0,104	0,1	0,287	0,11	0,109	0,107	0,11

19	0,107	0,176	0,176	0,176	0,162	0,104	0,1	0,287	0,11	0,109	0,107	0,11	0,1
20	0,176	0,176	0,176	0,162	0,104	0,1	0,287	0,11	0,109	0,107	0,11	0,1	0,1
21	0,176	0,176	0,162	0,104	0,1	0,287	0,11	0,109	0,107	0,11	0,1	0,1	0,1
22	0,176	0,162	0,104	0,1	0,287	0,11	0,109	0,107	0,11	0,1	0,1	0,1	0,11
23	0,162	0,104	0,1	0,287	0,11	0,109	0,107	0,11	0,1	0,1	0,1	0,11	0,173
24	0,104	0,1	0,287	0,11	0,109	0,107	0,11	0,1	0,1	0,1	0,11	0,173	0,108

d. Pembentukan Data Uji

Data yang sudah dinormalisasi digunakan sebagai data latih dan data uji. Data latih digunakan untuk proses *training*. Data latih berisi data dari bulan Januari 2019 sampai dengan Desember 2020. Data latih disajikan dalam bentuk 12 pola. Masing-masing pola

memiliki 13 data. Data tersebut terdiri dari x1 sampai dengan x12 dan target.

Contoh pola ke-1 yaitu Januari 2019 sampai dengan Desember 2019 (x1 sampai x12) dan bulan ke-13 yaitu Januari 2020 sebagai data target. Hasil pembentukan data latih dapat dilihat pada tabel 4.

Tabel 4. Data Uji

Pola	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	Target
1	0,104	0,111	0,11	0,185	0,103	0,1	0,1	0,1	0,1	0,112	0,1	0,115	0,9
2	0,111	0,11	0,185	0,103	0,1	0,1	0,1	0,1	0,112	0,1	0,115	0,9	0,517
3	0,11	0,185	0,103	0,1	0,1	0,1	0,1	0,112	0,1	0,115	0,9	0,517	0,193
4	0,185	0,103	0,1	0,1	0,1	0,1	0,112	0,1	0,115	0,9	0,517	0,193	0,128
5	0,103	0,1	0,1	0,1	0,1	0,112	0,1	0,115	0,9	0,517	0,193	0,128	0,13
6	0,1	0,1	0,1	0,1	0,112	0,1	0,115	0,9	0,517	0,193	0,128	0,13	0,103
7	0,1	0,1	0,1	0,112	0,1	0,115	0,9	0,517	0,193	0,128	0,13	0,103	0,1
8	0,1	0,1	0,112	0,1	0,115	0,9	0,517	0,193	0,128	0,13	0,103	0,1	0,1
9	0,1	0,112	0,1	0,115	0,9	0,517	0,193	0,128	0,13	0,103	0,1	0,1	0,112
10	0,112	0,1	0,115	0,9	0,517	0,193	0,128	0,13	0,103	0,1	0,1	0,112	0,16
11	0,1	0,115	0,9	0,517	0,193	0,128	0,13	0,103	0,1	0,1	0,112	0,16	0,1
12	0,115	0,9	0,517	0,193	0,128	0,13	0,103	0,1	0,1	0,112	0,16	0,1	0,108

2.3 Transformation

Tahap *transformation* adalah tahapan mengubah langkah-langkah dalam mengolah data menjadi bentuk *script* dalam aplikasi MATLAB. Langkah-langkah *transformation* adalah :

1. Membuat *folder* bernama “SKRIPSI”. Folder ini berisi *file* “Data Banjir.xls”.
2. Membuka *folder* SKRIPSI pada aplikasi MATLAB.

3. Membuat *Script* pada aplikasi MATLAB. *Script* pertama yang dibuat adalah *script training*. *Script* ini berisi kode-kode untuk tahapan *training* pada *data mining*. *Script* kedua yang perlu dibuat adalah *script testing*.

2.4 Data Mining

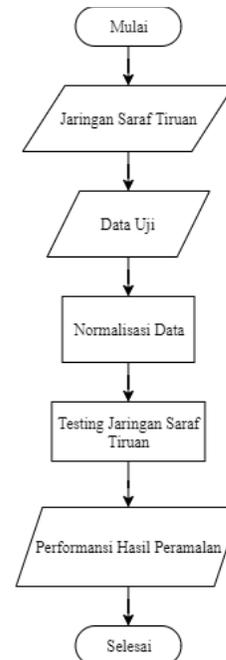
Data mining terdiri dari 2 tahapan yaitu tahapan *training* dan *testing*. Tahap *data mining* ini mencari pola atau informasi dalam

data yang telah dipilih. Pencarian pola dibantu menggunakan aplikasi MATLAB. Pencarian pola dilakukan dengan dengan cara mengolah data masukan dan nilai target. Penentuan pola dilakukan dengan cara menentukan jumlah *neuron*, *epoch*, *goal*, *show*, *momentum constant* (mc), dan *learning rate* (lr). Tahap ini dinamakan proses *training*. Alur dari proses *training* dapat dilihat pada Gambar 1.



Gambar 1 *Flowchart* Proses *Training*

Setelah pola didapat, langkah selanjutnya adalah menguji kembali pola tersebut dengan data yang lain. Tahap ini dinamakan proses *testing*. Alur dari proses *testing* dapat dilihat pada Gambar 2.



Gambar 2 *Flowchart* Proses *Testing*

Setelah didapat akurasi dari pola tersebut, maka data ditransformasikan balik. Hasil peramalan yang telah dilakukan masih dalam bentuk *Sigmoid Biner*. Hasil tersebut ditransformasikan ke bentuk semula menggunakan rumus fungsi aktivasi.

2.5 Interpretation

Tahap ini menjelaskan hasil peramalan. Hasil peramalan ditampilkan berupa grafik. Grafik tersebut dapat memudahkan pemerintah DKI Jakarta dalam melihat hasil peramalan. Selain grafik, tahap ini juga akan menghitung *error* dari data hasil peramalan dengan target. Nilai *error* akan ditampilkan dalam bentuk persentase.

3. Hasil dan Analisa

3.1. Training

Proses pelatihan data dilakukan sebanyak 100 kali pada tiga macam arsitektur jaringan. Arsitektur jaringan yang digunakan yaitu 12-10-1 (12 *input*, 10 *hidden layer*, dan 1 *output*), 12-12-1 (12 *input*, 12 *hidden layer*, dan 1 *output*), dan 12-14-1 (12 *input*, 14

hidden layer, dan 1 *output*). Tujuan proses *training* dilakukan secara berulang-ulang adalah untuk mendapatkan performa terbaik dari jaringan saraf tiruan *backpropagation* dengan nilai *Mean Square Error* (MSE) terkecil. Setiap percobaan menggunakan nilai parameter yang berbeda-beda. Parameter tersebut adalah parameter *epoch*, *momentum constant* (*mc*), dan *learning rate* (*lr*).

Proses *training* menghasilkan 70 data yang memiliki nilai MSE lebih kecil dari nilai *goal* yaitu 0,001. Data tersebut dipilih kembali sehingga menghasilkan 6 data dengan nilai MSE terkecil. Keenam data tersebut dapat dilihat pada Tabel 5.

Tabel 5. Hasil *Training* Jaringan Saraf Tiruan Peramalan Jumlah Jiwa Terdampak Banjir di Wilayah DKI Jakarta dengan Nilai MSE Lebih Kecil dari *Goal*

Arsitektur	Epoch	Mc	Lr	Goal	Mse	Regression	Time
12-10-1	3100	0.8	1.6	0.001	0,00099744	0,80938	0
	9800	0.95	2.4	0.001	0,0009999	0,80225	10
12-12-1	9900	0.95	2.5	0.001	0,00099991	0,80304	0
	10000	0.95	2.5	0.001	0,00099662	0,80447	0
12-14-1	4800	0.95	2.4	0.001	0,0009999	0,80467	2
	6400	0.6	0.7	0.001	0,00099999	0,80232	6

3.2. Testing

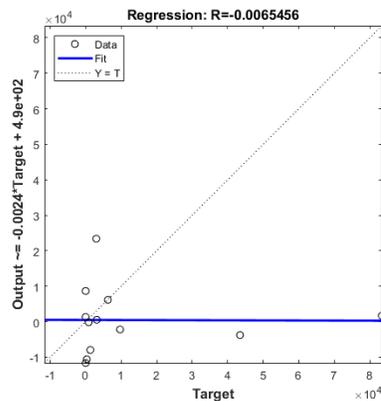
Proses *training* telah menghasilkan arsitektur jaringan yang memiliki nilai MSE lebih kecil dari *goal* yaitu 0,001. Jaringan tersebut disimpan dan digunakan kembali pada proses *testing*. Proses ini bertujuan

untuk mengetahui tingkat keakuratan nilai peramalan dengan nilai aktual pada arsitektur jaringan yang telah dilatih. Keenam data pada Tabel 5 diuji kembali dan menghasilkan performansi seperti Tabel 6.

Tabel 6. Hasil *Testing* dari Keenam Arsitektur Jaringan

Arsitektur	Epoch	Mc	Lr	Goal	MSE	Regression
12-10-1	3100	0.8	1.6	0.001	0,075822	-0,0065456
	9800	0.95	2.4	0.001	0,067757	0,14798
12-12-1	9900	0.95	2.5	0.001	0,076353	-0,040319
	10000	0.95	2.5	0.001	0,069719	0,19338
12-14-1	4800	0.95	2.4	0.001	0,06427	0,2207
	6400	0.6	0.7	0.001	0,074528	0,23477

Hasil terbaik dari proses *testing* terdapat pada arsitektur jaringan 12-10-1 dengan nilai parameter *epoch* 3100, *mc* 0,8, dan *lr* 1,6. Model jaringan ini menghasilkan nilai MSE sebesar 0,075822 dan nilai regresi sebesar -0,0065456. Grafik regresi dapat dilihat pada Gambar 3 dan grafik perbandingan keluaran JST dengan target pada Gambar 4.



Gambar 3 Grafik Regresi *Testing* Pertama



Gambar 4 Grafik Perbandingan Keluaran JST dengan Target *Testing* Pertama

Hasil persentase *error* dari peramalan yang telah dilakukan dapat dilihat pada Tabel 7. Nilai *error* terkecil berada pada Pola ke-10 yaitu 1,296%. Hal ini menunjukkan bahwa hasil peramalan memiliki keakuratan sebesar 98,704%.

Tabel 7. Hasil Peramalan Jumlah Jiwa Terdampak Banjir

Pola	Hasil Peramalan	Target	Target-Peramalan	Persentase <i>Error</i> (Target-Peramalan)/Target * 100
1	1693	83406	81713	97,970%
2	-3761	43464	47225	108,653%
3	-2184	9687	11871	122,546%
4	23390	2966	20424	688,604%
5	542,4	3117	2574,6	82,599%
6	-10600	320	10920	3412,500%
11	1349	0	1349	-
12	-146,4	827	973,4	117,703%

4. Kesimpulan

Model prediksi jumlah jiwa yang terdampak banjir di wilayah DKI Jakarta dengan menggunakan algoritma *Backpropagation* telah berhasil dibuat. Model memiliki arsitektur jaringan 12-10-1 (12 *input*, 10 *hidden layer*, dan 1 *output*) dengan parameter *epoch* sebesar 3100, *mc* sebesar 0,8, dan *lr* sebesar 1,6. Model ini menghasilkan nilai MSE sebesar 0,075822. Nilai MSE yang

dihasilkan lebih kecil dibandingkan arsitektur jaringan lainnya. Model menghasilkan nilai regresi sebesar -0,0065456 yang mendekati angka 1. Hal ini membuktikan bahwa model tersebut berhasil memprediksi data target dan model ini memiliki tingkat keakuratan sebesar 98,704%.

5. Daftar Pustaka

- [1] Agus Perdana Windarto, D. N., Anjar Wanto, Frinto Tambunan, M. S. H., Muhammad Noor Hasan Siregar, M. R. L., & Solikhun, Yusra Fadhillah, D. N. (2019). *Jaringan Saraf Tiruan: Algoritma Prediksi dan Implementasi*. *Journal of Chemical Information and Modeling* (Vol. 53).
- [2] Riyanto, E. (2017). Peramalan Harga Saham Menggunakan Jaringan Syaraf Tiruan Secara Supervised Learning Dengan Algoritma Backpropagation. *Jurnal Informatika Upgris*, 3(2), 137–142. <https://doi.org/10.26877/jiu.v3i2.1899>
- [3] Triyono, A., Santoso, A. J., & Pranowo. (2016). Penerapan Metode Jaringan Syaraf Tiruan Backpropagation Untuk Meramalkan Harga Saham (IHSG). *Jurnal Sistem Dan Informatika*, 11, 165–172. Retrieved from <https://media.neliti.com/media/publications/129675-ID-penerapan-metode-jaringan-syaraf-tiruan.pdf>
- [4] Vivian S. dan Rismon H.S. (2020). *Pemrograman Matlab untuk Teknik Sistem Kontrol dan Sistem Komunikasi*. Balige Publishing, Sumatera Utara.
- [5] Yanto, M., Mandala, E. P. W., Putri, D. E., & Yuhandri, Y. (2018). Peramalan Penjualan Pada Toko Retail Menggunakan Algoritma Backpropagation Neural Network. *Jurnal Media Informatika Budidarma*, 2(3), 110–117. <https://doi.org/10.30865/mib.v2i3.811>
- [6] Yohana, C., Griandini, D., & Muzambeq, S. (2017). Penerapan Pembuatan Teknik Lubang Biopori Resapan Sebagai Upaya Pengendalian Banjir. *Jurnal Pemberdayaan Masyarakat Madani (JPMM)*, 1(2), 296–308. <https://doi.org/10.21009/jpmm.001.2.10>