

# Prediksi Data *Time-series* menggunakan Jaringan Syaraf Tiruan Algoritma *Backpropagation* Pada Kasus Prediksi Permintaan Beras

Gita Indah Marthasari<sup>1\*)</sup>, Silcillya Ayu Astiti<sup>2</sup>, Yufis Azhar<sup>3</sup>

<sup>1,2,3</sup>Jurusan Informatika, Fakultas Teknik, Universitas Muhammadiyah Malang

<sup>1,2,3</sup>Jalan Raya Tlogomas No. 246, Kota Malang, 65144, Indonesia

email: <sup>1</sup> gita@umm.ac.id, <sup>2</sup>silcillya.ayuastiti99@gmail.com, <sup>3</sup>yufis@umm.ac.id

**Abstract** – Recently, Indonesia, as a country where the majority of the population chooses rice as the primary food source, gets a decline in the rice consumption patterns, which resulted in reduced demand for rice that should have been stable. The decrease of rice purchasing power impacts several rice suppliers, commonly referred to as rice agents, to buy rice from rice production companies. Therefore, prediction of rice stock is essential to do. This paper aims to apply the backpropagation neural network method to forecast the amount of rice demand. The data used in the study is time-series data in the form of the number of requests for rice as much as 609 data from two types of rice. The modeling scenario in this study applies one to five hidden layers with a different number of hidden neurons in each experiment. The elastic net regularization method was applied after the data denormalization process to improve the quality of the resulting model. Based on the experiments, obtained the best model on architecture 7-50-200-300-250-300-1 with MSE = 0.001278, RMSE = 0.301950 in the training process and MSE results = 0.002391, RMSE = 0.204972 in the testing process.

**Abstrak** – Indonesia sebagai negara yang mayoritas penduduknya memilih beras sebagai sumber pangan utama, di tahun 2020 mengalami penurunan pola konsumsi yang mengakibatkan permintaan beras yang seharusnya stabil menjadi berkurang. Menurunnya daya beli masyarakat akan beras berdampak pada beberapa agen beras atau yang biasa disebut dengan agen beras untuk membeli beras pada perusahaan produksi beras. Oleh karena itu, prediksi stok beras penting untuk dilakukan. Tujuan penelitian ini adalah menerapkan metode jaringan syaraf tiruan backpropagation untuk melakukan peramalan jumlah permintaan beras. Data yang digunakan pada penelitian yaitu data time series berupa jumlah permintaan beras sebanyak 609 data dari dua jenis beras. Skenario pemodelan pada penelitian menerapkan 1-5 hidden layer dengan jumlah neuron hidden yang berbeda di setiap percobaannya. Untuk meningkatkan kualitas dari model yang dihasilkan, metode regularisasi elastic net yang diterapkan setelah proses denormalisasi data. Hasil yang diperoleh memperlihatkan bahwa menggunakan jaringan syaraf tiruan backpropagation menunjukkan prediksi yang baik yang pada penelitian ini hasil terbaik terdapat pada arsitektur 7-50-200-300-250-300-1 dengan hasil MSE = 0.001278, RMSE = 0,301950 di proses pelatihan and hasil MSE = 0.002391, RMSE = 0,204972 di proses pengujian.

**Kata Kunci** – Permintaan Beras, Prediksi, Jaringan Syaraf Tiruan *Backpropagation*.

\*) penulis korespondensi: Silcillya Ayu Astiti  
Email: silcillya.ayuastiti99@gmail.com

## I. PENDAHULUAN

Indonesia merupakan salah satu negara yang mengalami pandemi pada tahun 2020 akibat COVID-19 (*Coronavirus Deases 2019*). Dampak yang ditimbulkan sangat beragam, salah satunya yaitu kelumpuhan di sektor ekonomi yang membuat menurunnya konsumsi atau permintaan daya beli masyarakat [1]. Sektor ekonomi adalah salah satu sektor kehidupan yang terdapat pada manusia dengan pangan sebagai kebutuhan paling mendasar untuk bertahan hidup di tingkat kebutuhan manusia [2][3]. Menurunnya tingkat permintaan masyarakat akan kebutuhan pangan di masa pandemi COVID-19 membawa pengaruh yang besar terhadap perusahaan produksi di bidang pangan. Salah satu perusahaan yang terdampak adalah Perusahaan CV. PUSPA.

CV. PUSPA merupakan salah satu perusahaan industri penggilingan padi menjadi beras yang menghasilkan berbagai jenis beras dengan dua merek unggulan, yaitu Rajawali dan Joged. Pada awalnya perusahaan ini merupakan perusahaan kecil dengan jumlah permintaan beras per hari sebanyak 600 kg (kilogram) dan terus mengalami peningkatan permintaan pembelian dari tahun ke tahun dengan pencapaian permintaan awal terbesar di tahun 2017 sebanyak 4.576.000 ton beras. Kemudian di tahun 2018 meningkat 5.292.900 ton dan 6.622.000 ton di tahun 2019. Namun, di masa pandemi yang terjadi di tahun 2020 ternyata membuat jumlah permintaan beras per hari yang terdapat di Perusahaan CV. PUSPA tidak stabil. Tingkat konsumsi masyarakat yang rendah di masa pandemi mengakibatkan menurunnya daya beli agen beras dalam membeli beras di Perusahaan CV. PUSPA. Berdasarkan permasalahan terkait menurunnya tingkat permintaan beras dikhawatirkan jika tidak ditangani lebih mendalam akan berdampak pada produksi beras yang secara berlebihan. Maka dari itu sangat perlu dilakukan suatu tindakan pencegahan, salah satunya dengan memprediksi data dari hasil permintaan beras.

Prediksi merupakan suatu kegiatan peramalan nilai di masa mendatang yang terdiri dari dua jenis, yaitu secara kualitatif yang digunakan saat tidak adanya data kurun waktu atau kurun waktu yang dihasilkan sedikit dan kuantitatif yang dilakukan secara sistematis, analisis runtun waktu (*time series*) [4]. Pada dasarnya metode yang digunakan dalam mengolah data prediksi sangatlah beragam, Jaringan Syaraf Tiruan (JST) salah satunya. Metode yang memiliki nama lain *Artificial Neural Network* (ANN) merupakan jaringan yang berasal dari pemrosesan informasi dan memiliki sistem kerja serta karakteristik yang hampir sama dengan jaringan syaraf biologis pada manusia yang memiliki kelebihan untuk memproses data masuk dengan meneliti dan melakukan perulangan hingga kondisi yang diinginkan tercapai. Dalam melakukan suatu peramalan atau prediksi yang akurat, jaringan syaraf tiruan memiliki teknik *backpropagation* dalam pengimplementasiannya. Teknik *backpropagation* adalah salah satu teknik yang sering digunakan pada jaringan syaraf tiruan dalam meramalkan suatu data di masa mendatang melalui data – data yang sudah tersedia sebelumnya [5].

Kelebihan yang terdapat pada jaringan syaraf tiruan algoritma *backpropagation* adalah adanya pembelajaran yang berulang yang dapat membuat sistem berjalan dengan baik dan tepat [6]. Hal tersebut terbukti dengan adanya beberapa peneliti terdahulu yang melakukan perbandingan menggunakan jaringan syaraf tiruan algoritma *backpropagation* dengan metode lain serta tingginya tingkat akurasi dan kecilnya *error* yang dihasilkan khususnya pada data *time series* menggunakan jaringan syaraf tiruan *backpropagation*.

Berdasarkan permasalahan terkait menurunnya permintaan beras yang terdapat pada Perusahaan CV. PUSPA serta kelebihan dan kemampuan yang dimiliki oleh jaringan syaraf tiruan algoritma *backpropagation* dalam melakukan sebuah prediksi terutama pada data *time series*, maka penelitian tentang “Implementasi Jaringan Syaraf Tiruan Algoritma *Backpropagation* Pada Kasus Prediksi Permintaan Beras” perlu untuk dilakukan dengan harapan dapat memprediksi permintaan beras secara akurat berdasarkan tingkat *error* yang dihasilkan dengan tujuan untuk meminimalisir kerugian yang terdapat pada perusahaan.

## II. PENELITIAN YANG TERKAIT

Terdapat beberapa penelitian sebelumnya yang menerapkan jaringan syaraf tiruan *backpropagation* sebagai metode untuk menyelesaikan permasalahan dalam memprediksi data *time series*. Pada penelitian sebelumnya, jaringan syaraf tiruan *backpropagation* diimplementasikan oleh A. Wanto dan Windarto (2017) untuk memprediksi indeks harga konsumen berdasarkan kelompok kesehatan [4]. Hasil menunjukkan bahwa melalui percobaan dengan 8 model arsitektur, metode yang digunakan berhasil mendapatkan hasil prediksi terbaik pada arsitektur 12-70-1 dengan hasil MSE 0.3659742 [4].

D. Monika., dkk (2019) menerapkan metode jaringan syaraf tiruan *backpropagation* dengan memprediksi ketersediaan cabai dengan arsitektur 4-2-1, 4-3-1, 4-4-1, dan 4-5-1 [7]. Hasil menunjukkan bahwa metode yang digunakan menghasilkan nilai *error* yang baik pada model arsitektur 4-5-1 dengan hasil MSE 0,010651 [7].

B. Bagasta., dkk (2019) yang dalam penelitiannya memprediksi besarnya jumlah nilai ekspor yang terletak di Provinsi NTB [8]. Penelitian ini dilakukan dengan menggunakan dua skenario yaitu fungsi aktivasi Sigmoid Biner pada *hidden layer* dan *output layer* dan dengan menggunakan fungsi aktivasi ReLU pada *hidden layer* dan Linear pada *output layer* dengan hasil yang menunjukkan bahwa fungsi aktivasi ReLU – Linear lebih bagus untuk digunakan pada kasus peramalan jumlah nilai ekspor di provinsi NTB dibandingkan dengan menggunakan fungsi aktivasi Sigmoid Biner dengan hasil *error* terbaik 0.0012 dan nilai MSE sebesar 0.0309 serta MAPE sebesar 53.04% pada tahap pengujian [8].

M. Rahul., dkk (2020) yang dalam penelitiannya menerapkan metode jaringan syaraf tiruan *backpropagation* memprediksi pembuatan SIM [9]. Hasil yang diperoleh menunjukkan nilai MSE terbaik terletak pada model arsitektur 12-3-1 dengan hasil MSE pelatihan 0,01791138 dan MSE pengujian 0,04068227 [9].

Kemudian terdapat penelitian lainnya yang memecahkan permasalahan dengan melakukan perbandingan metode antara metode jaringan syaraf tiruan *backpropagation* dengan metode lainnya, yang membuktikan bahwa jaringan syaraf tiruan *backpropagation* menghasilkan *output* yang

lebih unggul. Hal tersebut dibuktikan pada penelitian sebelumnya yang dilakukan oleh Masruroh (2020) dalam memprediksi nilai ujian nasional (UN) siswa SMP pada metode *regresi linier* dan *neural network backpropagation* menggunakan *Software R* [10]. Hasil menunjukkan tingkat akurasi dan *error* terbaik terdapat pada jaringan syaraf tiruan algoritma *backpropagation* dengan hasil RMSE 7,28 dan MAPE 0,55% [10].

## III. METODE PENELITIAN

### A. Dataset

Data yang akan digunakan pada penelitian merupakan data jumlah permintaan agen beras yang didapat dari perusahaan beras CV. PUSPA selama 20 bulan, yakni dari bulan Januari 2019 hingga Agustus 2020 per harinya dengan jumlah data penelitian sebanyak 609 data di setiap mereknya. Data permintaan beras yang terdiri dari dua jenis merek beras, yaitu beras Rajawali dan Jaged dibagi dengan perbandingan 80:20, yaitu 80% pada data *training* dan 20% pada data *testing*. Adapun sampel data jumlah permintaan beras dapat dilihat pada Tabel I dan Tabel II.

TABEL I  
SAMPEL DATA PERMINTAAN BERAS RAJAWALI

No	Waktu	Permintaan
1	01/01/2019	20025
2	02/01/2019	20100
3	03/01/2019	20100
.....	.....	.....
609	31/08/2020	13225

TABEL II  
SAMPEL DATA PERMINTAAN BERAS JOGED

No	Waktu	Permintaan
1	01/12/2019	16000
2	02/12/2019	16000
3	03/12/2019	16100
.....	.....	.....
609	31/08/2020	10225

### B. Pra Proses Data

Tahapan pra proses pada data dilakukan dengan cara normalisasi dataset menggunakan normalisasi *min-max* dengan mengubah dataset ke dalam rentang 0 – 1. Pengubahan nilai ke dalam rentang 0 – 1 pada normalisasi *min-max* memiliki tujuan agar nilai yang terdapat pada setiap data memiliki proporsi yang sama dalam pemrosesannya. Normalisasi *min-max* tertuang melalui persamaan berikut.

$$x_i = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

Keterangan :

- $x_i$  = Hasil normalisasi
- $x$  = Data yang akan dinormalisasi
- $x_{min}$  = Nilai minimum
- $x_{max}$  = Nilai maksimum

C. Implementasi Metode Jaringan Syaraf Tiruan Backpropagation

Terdapat dua tahapan pada jaringan syaraf tiruan *backpropagation*, yang meliputi tahapan pelatihan dan pengujian. Pelatihan pada jaringan syaraf tiruan *backpropagation* bertujuan untuk mengidentifikasi kerangka data berdasarkan inputan parameter yang terdapat pada jaringan syaraf tiruan, seperti neuron *input*, neuron *hidden*, neuron *output*, laju pembelajaran, jumlah *epoch*, dan toleransi error. Tahapan pelatihan *backpropagation* terdiri dari 3 fase, yaitu fase maju, fase mundur, dan fase memodifikasikan bobot untuk menurunkan tingkat kesalahan [11]. Adapun langkah – langkah tahapan pelatihan algoritma *backpropagation* antara lain [11] :

1. Fase Propagasi Maju

Langkah 0: Inisialisasi bobot – bobot dengan bilangan kecil

Langkah 1: Lakukan langkah 2 – 9 jika kondisi penghentian belum terpenuhi

Langkah 2: Lakukan langkah 3 – 8 pada setiap rangka pelatihan

Langkah 3: Tiap unit *input* menerima sinyal daan kemudian dilakukan penerusan ke unit *hidden*.

Langkah 4: Menghitung nilai neuron pada unit *z* (unit  $z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ij}$ ) (2)

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}} \quad (3)$$

Langkah 5: Menghitung nilai neuron pada unit *y* (*output layer*)  $y_k (k = 1, 2, \dots, m)$

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj} \quad (4)$$

$$z_k = f(z_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}} \quad (5)$$

2. Fase Propagasi Mundur

Langkah 6: Menghitung kesalahan *error* lapisan *output*  $\delta_k$  berdasarkan nilai *output*  $y_k (k = 1, 2, \dots, m)$

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \quad (6)$$

Kemudian menghitung besarnya koreksi bobot  $W_{kj}$

$$\Delta W_{kj} = \alpha \delta_k z_j, k = 1, 2, \dots, m; j = 0, 1, \dots, p \quad (7)$$

Langkah 7: Menghitung kesalahan *error* lapisan *hidden*  $z_j$  berdasarkan nilai *hidden layer*  $z_j (j = 1, 2, \dots, p)$

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj} \quad (8)$$

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j) \quad (9)$$

Kemudian menghitung besarnya koreksi bobot  $v_{ji}$

$$\Delta v_{ij} = \alpha \delta_j x_i, j = 1, 2, \dots, p; i = 0, 1, \dots, n \quad (10)$$

3. Fase Perubahan Bobot

Langkah 8: Menghitung semua perubahan bobot

Perubahan bobot yang menuju unit *output* :

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj} \quad (11)$$

Perubahan bobot yang menuju unit *hidden* :

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (12)$$

Langkah 9: Melakukan pengujian apakah kondisi sudah terpenuhi atau belum.

Pengujian dilakukan ketika tahapan pelatihan sudah terselesaikan. Pengujian pada jaringan syaraf tiruan *backpropagation* hanya dapat dilakukan pada fase pertama, yaitu fase propagasi maju yang dimana seluruh bobot *input* diambil berdasarkan inputan parameter pada pelatihan data [12].

D. Denormalisasi Dataset

Hasil dari proses prediksi yang sudah dilakukan pada tahapan pengujian masih menghasilkan data dengan range 0 – 1. Untuk mengetahui banyaknya jumlah permintaan beras yang dihasilkan oleh prediksi jaringan syaraf tiruan *backpropagation*, maka perlu dilakukan sebuah denormalisasi data yang ditujukan untuk merubah data dari hasil prediksi ke dalam data asli. Persamaan denormalisasi dapat dilihat pada persamaan berikut :

$$x_i = y(x_{max} - x_{min}) + x_{min} \quad (13)$$

Keterangan :

$x_i$  = Hasil denormalisasi

$y$  = Hasil normalisasi

$x_{min}$  = Nilai minimum

$x_{max}$  = Nilai maksimum

E. Regularisasi Elastic Net

Regularisasi merupakan fungsi yang digunakan untuk meningkatkan kinerja pada *artificial neural network* agar tidak terjadi *overfitting*. Pada dasarnya regularisasi bekerja dengan menambah suatu pinalti pada *loss function* ketika terjadi peningkatan kompleksitas data [13]. Pada penelitian ini, teknik regulasi yang digunakan adalah teknik regulasi *elastic-net*. *Elastic-net* adalah metode yang menggabungkan dua penalti dari LASSO (*L1 Regularization*) dan Ridge (*L2 Regularization*) yang mampu mengatasi permasalahan tingginya korelasi [13]. Kombinasi yang terdapat pada *elastic net* memiliki peran dan fungsinya masing – masing, yaitu L1 yang berperan sebagai seleksi variabel dan L2 sebagai solusi untuk menstabilkan [14]. Adapun bentuk gabungan L1 dan L2 penalti pada *elastic net* dapat dilihat pada persamaan berikut.

$$p(\beta) = \lambda \left[ \alpha \sum_{i=1}^n \beta_i^2 + (1 - \alpha) \sum_{i=1}^n |\beta_i| \right] = \lambda [\alpha \|\beta\|_2^2 + (1 - \alpha) \|\beta\|_1], \alpha \in [0,1] \quad (14)$$

F. Performa Hasil

Setelah melakukan pelatihan dan pengujian data, tahapan selanjutnya yaitu mengukur tingkat kinerja yang dihasilkan dengan menghitung rata – rata kesalahan atau *error* pada setiap *epoch*. RMSE (*Root Mean Squared Error*) dan MSE (*Mean Squared Error*) digunakan pada penelitian ini dalam menghitung tingkat *error* yang tertuang pada persamaan berikut.

$$MSE = \frac{1}{n^{polin}} \sum_k^{n^{polin}} (t_k - y_k)^2 \quad (19)$$

$$RMSE = \sqrt{\frac{\sum |t_k - y_k|^2}{n}} \quad (15)$$

Keterangan :

$t_k$  = Data sebenarnya

$y_k$  = Data hasil prediksi

$n$  = Jumlah data

G. Analisis Hasil

Kegiatan analisis hasil dilakukan pada hasil *error* yang didapatkan pada masing – masing proses pelatihan dan pengujian. Hasil tersebut kemudian di analisis dengan melihat

beberapa parameter yang telah diujikan dalam memprediksi data permintaan beras menggunakan metode jaringan syaraf tiruan *backpropagation*.

IV. HASIL DAN PEMBAHASAN

Pada penelitian ini, metode jaringan syaraf tiruan *backpropagation* diimplementasikan melalui skenario pemodelan yang sudah ditentukan sebelumnya, yaitu pengujian arsitektur jaringan yang terdiri dari 1 - 5 *hidden layer* serta pengujian *learning rate* dengan *output* atau keluaran yang dihasilkan berupa plot *loss*, mse, rmse, *loss*, disetiap proses pelatihan dan pengujian serta hasil prediksi berupa nilai data serta divisualisasikan melalui grafik

A. Perancangan Model Pengujian

Perancangan model arsitektur dimulai dari insialisasi parameter yang terdiri dari 1, 2, 3, 4, dan 5 arsitektur *hidden layer*, neuron *input* sebanyak 7 neuron yang dihitung berdasarkan 7 hari ke belakang, neuron *output* sebanyak 1 neuron yang merupakan data yang akan diprediksi, serta terdapat neuron *hidden*, *learning rate* ( $\alpha$ ), ReLu sebagai fungsi aktivasi, *epoch*, dan *loss threshold* yang merupakan penentuan *loss minimum* yang diambil. *Learning rate* sebesar 0.001 dan *epoch* sebesar 2000 digunakan ketika proses pelatihan dan pengujian dalam melakukan pengujian arsitektur jaringan di 1 - 5 *hidden layer*. Sedangkan *learning rate* 0.0001, 0.00001, 0.00003, 0.00005 dan *epoch* sebesar 5000 digunakan ketika proses pelatihan dan pengujian dalam melakukan pengujian *learning rate*. Tujuan dari adanya perancangan model pengujian yang berbeda adalah untuk mendapatkan nilai *error* terbaik pada hasil akhir yang sangat dipengaruhi oleh rancangan pemodelan yang dibuat, baik dari segi laju pembelajaran, fungsi aktivasi, jumlah *hidden layer* dan banyaknya *neron* pada *hidden layer*, serta jumlah *epoch* yang digunakan melalui insialisasi parameter pada Tabel III.

TABEL III  
INSIALISASI PARAMETER

Parameter	Spesifikasi
Arsitektur	1, 2, 3, 4, dan 5 <i>hidden layer</i>
Neuron <i>input</i>	7
Neuron <i>hidden</i>	3, 5, 10, 25, 50, 75, 100, 150, 200, 250, 300, 500, 750, 1000
Neuron <i>output</i>	1
<i>Learning rate</i> ( $\alpha$ )	0.001, 0.0001, 0.00001, 0.00003, dan 0.00005
<i>Epoch</i>	2000 dan 5000
<i>Loss threshold</i>	0.0005
Fungsi Aktivasi	ReLu

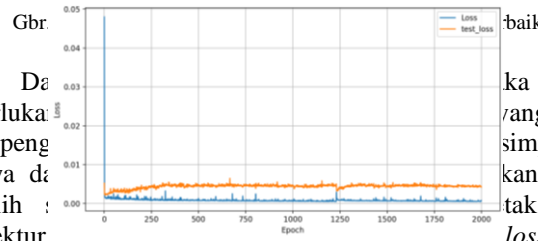
B. Hasil Pengujian Arsitektur Jaringan

Hasil dari pengujian arsitektur jaringan pada proses pelatihan dan pengujian setiap *hidden layer* dilakukan pemilihan arsitektur terbaik pada *hidden layer* dengan mengevaluasi hasil *loss* pelatihan di setiap *hidden layer* yang dilanjutkan dengan evaluasi *loss* pengujian dari *loss* pelatihan yang lolos uji kurang dari *loss* minimum yang ditentukan, yaitu kurang dari 0.0005. Keluaran yang dihasilkan berupa MSE, RMSE, dan *loss* pada proses pelatihan dan pengujian yang mana hasil menunjukkan bahwa dari percobaan 1-5

*hidden layer* yang menghasilkan *loss* kurang dari 0.0005 adalah *hidden layer* 3, 4, dan 5. Hasil dari pengujian arsitektur jaringan untuk menentukan arsitektur terbaik dapat dilihat pada Gambar 1.

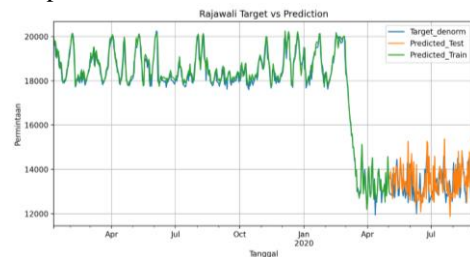
Pelatihan		Pengujian		Loss Pelatihan	Loss Pengujian	Arsitektur Jaringan
MSE	RMSE	MSE	RMSE			
0,000358	0,306269	0,004226	0,211625	0,000348	0,004226	7-50-200-300-250-300-1
0,000409	0,304769	0,004250	0,208841	0,000441	0,004250	7-50-200-300-250-1
0,000512	0,303325	0,00462	0,207597	0,000476	0,00462	7-50-200-300-1

Gbr. 1 menunjukkan grafik *loss* dan *test\_loss* yang menunjukkan bahwa *loss* pelatihan dan *test\_loss* sudah mencapai nilai yang diinginkan, yaitu kurang dari 0.0005. Hal ini menunjukkan bahwa arsitektur yang digunakan sudah optimal. Dari grafik tersebut dapat disimpulkan bahwa arsitektur terbaik adalah yang memiliki *loss* terendah, yaitu arsitektur 7-50-200-300-250-1.

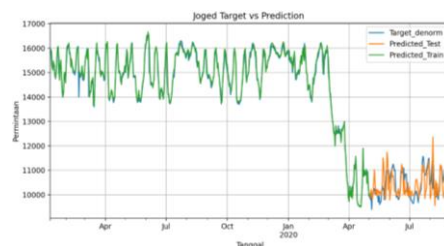


Gbr. 2 Plot Hasil Loss Arsitektur Terbaik

Pada Gambar 2 terlihat bahwa garis biru yang mempresentasikan *loss* pengujian dan garis orange sebagai *loss* pelatihan belum menghasilkan performa yang baik. Dari hasil *loss* pelatihan dan pengujian kemudian didapatkan hasil prediksi di setiap jenis merek beras yang divisualisasikan berupa grafik pada Gambar 3 dan Gambar 4.



Gbr. 3 Grafik Hasil Prediksi Beras Rajawali Pada Arsitektur Terbaik



Gbr. 4 Grafik Hasil Prediksi Beras Joged Pada Arsitektur Terbaik

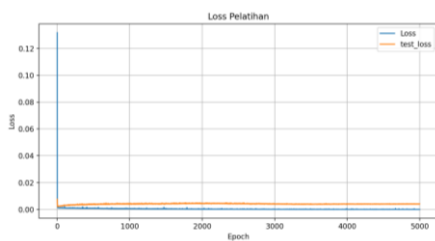
C. Hasil Pengujian Learning Rate

Pada tahapan ini dilakukan pengujian pada *learning rate* dengan tujuan untuk mendapatkan parameter terbaik dari model arsitektur terbaik yang sudah terpilih sebelumnya, yaitu arsitektur 7-50-200-300-250-300-1. *Learning rate* yang diujikan pada tahapan ini adalah 0.0001, 0.00001, 0.00003, 0.00005 dengan pengambilan *loss* minimum yaitu 0.0005 dan jumlah *epoch* sebesar 5000 *epoch*. Hasil dari pengujian *learning rate* dapat dilihat pada Gambar 5.

Pelatihan		Pengujian		Loss Pelatihan	Loss Pengujian	Learning Rate
MSE	RMSE	MSE	RMSE			
0,000033	0,305775	0,004090	0,210433	0,000028	0,004090	0,0001
0,000183	0,305705	0,005257	0,210387	0,000186	0,005257	0,00005
0,000310	0,305518	0,006107	0,209480	0,000306	0,006107	0,00003
0,000606	0,304482	0,005135	0,211450	0,000618	0,005135	0,00001

Gbr. 5 Hasil Pengujian *Learning Rate*

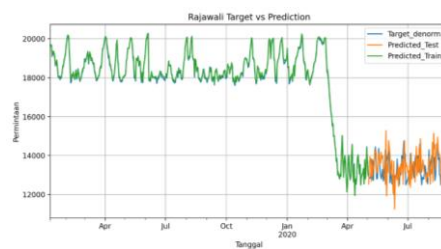
Berdasarkan hasil yang didapatkan dalam melakukan pengujian *learning rate* di Gambar 5, hasil terbaik terletak pada *learning rate* 0.0001 yang memiliki hasil *loss test* terkecil dari ketiga *loss train* yang lolos 0.0005. Hasil dari penentuan *learning rate* terbaik akan digunakan sebagai parameter terbaru pada model arsitektur terbaik dalam melakukan proses *regularization*. Hasil kinerja dari proses pelatihan dan pengujian berdasarkan *loss* yang didapatkan dan maksimum *epoch* yang diujikan memperlihatkan perbedaan hasil dari *loss* yang sangat signifikan dan membuat proses prediksi yang dilakukan tidak maksimal. Kondisi pada model ini disebut dengan *overfitting*, sehingga perlu penanganan khusus agar model menjadi lebih baik, salah satunya dengan melakukan teknik *regularization*. Adapun plot hasil *loss* yang terdapat pada proses pelatihan dan pengujian yang dihasilkan oleh arsitektur 7-50-200-300-250-300-1 dengan *learning rate* terbaru, yaitu 0.0001 dapat dilihat pada Gambar 6.



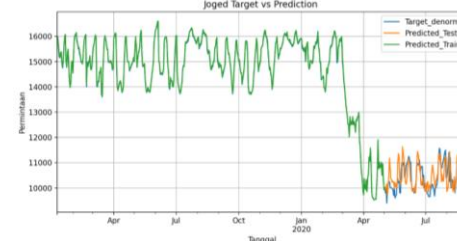
Gbr. 6 Plot Hasil Loss Learning Rate Terbaik

Gambar 6 menampilkan hasil kinerja dari proses pelatihan dan pengujian berdasarkan *loss* yang didapatkan dan maksimum *epoch* yang diujikan. Terlihat bahwa setelah dilakukan pengujian pada *learning rate*, model yang dihasilkan masih terlihat sama, yaitu adanya *loss* pelatihan yang lebih kecil daripada *loss* pengujian dengan perbedaan yang sangat signifikan dan membuat proses prediksi yang dilakukan tidak maksimal. Kondisi pada model ini disebut dengan *overfitting*, sehingga perlu penanganan khusus agar model menjadi lebih baik, salah satunya dengan melakukan teknik *regularization*. Berdasarkan hasil *loss* pelatihan dan pengujian kemudian didapatkan hasil prediksi di setiap jenis

merek beras yang terlihat pada Gambar 7 dan 8 sebagai hasil visualisasi berupa grafik dari hasil prediksi pada pengujian *learning rate* terbaik



Gbr. 7 Grafik Hasil Prediksi Beras Rajawali Pada *Learning Rate* Terbaik



Gbr. 8 Grafik Hasil Prediksi Beras Joged Pada *Learning Rate* Terbaik

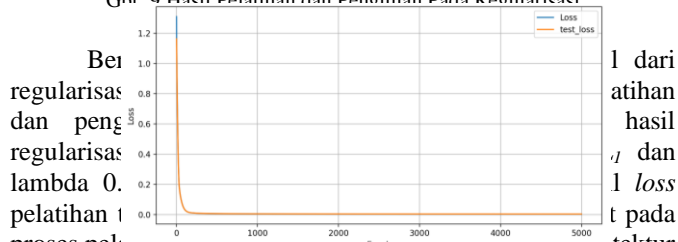
#### D. Regularisasi

Regularisasi diterapkan pada penelitian ini melihat hasil dari model arsitektur yang terbentuk mengalami kondisi *overfitting* sehingga perlu penanganan khusus agar hasil *loss* pelatihan dan pengujian yang didapatkan dalam kondisi baik sehingga hasil prediksi yang dihasilkan juga akurat. Regularisasi yang digunakan merupakan regularisasi *elastic net* dengan menggabungkan Regulasi  $L_1$  dan Regulasi  $L_2$ . Arsitektur parameter regularisasi ( $\lambda$ ) yang digunakan dalam melakukan pengujian regularisasi, yaitu 0.0001-0.0001 dan 0.0001-0.00001 dengan hasil yang terlihat pada Gambar 9.

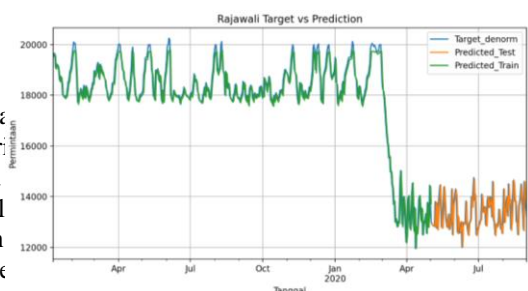
Pelatihan		Pengujian		Loss Pelatihan	Loss Pengujian	Regularisasi
MSE	RMSE	MSE	RMSE			
0,001278	0,301950	0,002391	0,204972	0,002604	0,003704	( $0,0001$ , $0,00001$ )
0,001373	0,302146	0,002401	0,205158	0,002705	0,003818	( $0,0001$ , $0,0001$ )

Gbr. 9 Hasil Pelatihan dan Pengujian Pada Regularisasi

Berregularisasi dan pengujian regularisasi  $\lambda$  0.0001 dan 0.00001 pada proses pelatihan dan pengujian pada arsitektur 7-50-200-300-250-300-1 dengan *learning rate* terbaru, yaitu 0.0001 dan sudah melalui tahap regularisasi menggunakan hasil regularisasi terbaik dapat dilihat pada Gambar 10.

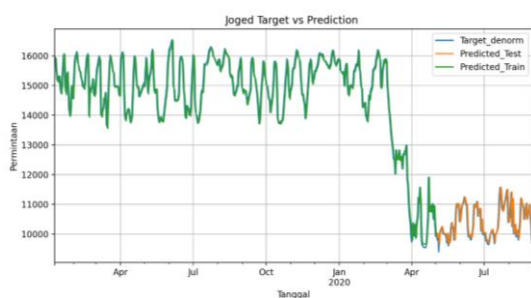


pelatih: regularisasi dengan visualisasi



loss roses atau tidak data nerek hasil prediksi setelah regularisasi.

Gbr. 11 Grafik Hasil Prediksi Beras Rajawali Setelah Regularisasi



Gbr. 12 Grafik Hasil Prediksi Beras Joged Setelah Regularisasi

E. Analisis Hasil

Berdasarkan hasil pengujian arsitektur, learning rate, hingga tahap regulasi dalam memprediksi permintaan beras menggunakan metode jaringan syaraf tiruan backpropagation terdapat beberapa hasil yang diperoleh, antara lain:

1. Pada tahap pengujian arsitektur jaringan yang menerapkan 1-5 hidden layer dengan tujuan untuk mendapatkan arsitektur terbaik, didapatkan hasil pemilihan arsitektur terbaik di 5 hidden layer dengan arsitektur 7-50-200-300-250-300-1 dengan MSE = 0.000358, RMSE = 0.306269 di proses data pelatihan dan hasil MSE = 0.004226, RMSE = 0.211625 di proses data pengujian.
2. Pengujian learning rate yang dilakukan dengan menerapkan learning rate sebesar 0.0001, 0.00001, 0.00003, 0.00005 dilakukan untuk mendapatkan parameter terbaik pada perolehan arsitektur terbaik dengan 0.0001 sebagai learning rate terbaik pada tahapan pengujian. Hasil yang diperoleh pada tahapan pengujian learning rate terbaik, yaitu MSE = 0.000033, RMSE = 0.305775 di proses data pelatihan dan hasil MSE = 0.004090, RMSE = 0,210433 di proses data pengujian.

3. Hasil loss yang terdapat pada tahapan pelatihan dan pengujian implementasi metode jaringan syaraf tiruan backpropagation memiliki perbedaan yang sangat signifikan atau dikatakan overfitting. Sehingga diperlukan teknik regularization. Hasil akhir yang diperoleh pada tahapan regularization menghasilkan loss yang baik dalam melakukan prediksi data yang sebenarnya atau data asli dengan perolehan MSE = 0.001278, RMSE = 0.301950 di proses data pelatihan and hasil MSE = 0.002391, RMSE = 0.204972 di proses data pengujian.

V. KESIMPULAN

Didapatkan kesimpulan dari hasil dari penelitian dalam memprediksi permintaan beras menggunakan jaringan syaraf tiruan algoritma backpropagation, yaitu:

1. Terdapat berbagai tahapan dalam mempersiapkan data sebelum dilakukan proses mengimplementasikan metode jaringan syaraf tiruan backpropagation, baik itu dari proses pengumpulan data, pembentukan dataset, proses normalisasi menggunakan min-max, dan melakukan pembagian dataset menggunakan ratio 80:20 yang mana 80% data pelatihan dan 20% data pengujian.
2. Berdasarkan proses implementasi metode yang dilakukan dengan menerapkan arsitektur 1-5 hidden layer, fungsi aktivasi ReLu, epoch 2000, serta learning rate sebesar 0.001, arsitektur terbaik terletak pada arsitektur 7-50-200-300-250-300-1 dengan hasil model yang bersifat overfitting sehingga membuat hasil prediksi tidak maksimal. Untuk mengatasi kasus tersebut dilakukan regularisasi menggunakan teknik regularisasi elastic net dengan learning rate 0.0001 dan epoch 5000. Hasil setelah diregularisasi menunjukkan tidak overfitting dengan percobaan regulasi terbaik terdapat pada percobaan lambda 0.0001 L1 dan lambda 0.00001 L2.

DAFTAR PUSTAKA

- [1] A. K. Pakpahan, "Covid-19 Dan Implikasi Bagi Usaha Mikro, Kecil, Dan Menengah," J. Ilm. Hub. Int., vol. 0, no. 0, pp. 59-64, 2020, doi: 10.26593/jihi.v0i0.3870.59-64.
- [2] D. A. D. Nasution, E. Erlina, and I. Muda, "Dampak Pandemi COVID-19 terhadap Perekonomian Indonesia," J. Benefita, vol. 5, no. 2, p. 212, 2020, doi: 10.22216/jbe.v5i2.5313.
- [3] M. Hanif, M. Abdurrohman, and A. G. Putrada, "Rice consumption prediction using linear regression method for smart rice box system," J. Teknol. dan Sist. Komput., vol. 8, no. 4, pp. 284-288, 2020, doi: 10.14710/jtsiskom.2020.13353.
- [4] A. Wanto and A. P. Windarto, "Analisis Prediksi Indeks Harga Konsumen Berdasarkan Kelompok Kesehatan Dengan Menggunakan Metode Backpropagation," J. Penelit. Tek. Inform. Sink., vol. 2, no. 2, pp. 37-43, 2017.
- [5] Y. Andriani, H. Silitonga, and A. Wanto, "Analisis Jaringan Syaraf Tiruan untuk prediksi volume ekspor dan impor migas di Indonesia," Regist. J. Ilm. Teknol. Sist. Inf., vol. 4, no. 1, p. 30, 2018, doi: 10.26594/register.v4i1.1157.

- [6] A. Sudarsono, "Jaringan Syaraf Tiruan Untuk Memprediksi Laju Pertumbuhan Penduduk Menggunakan Metode," *Media Infotama*, vol. 12, no. 1, pp. 61–69, 2016.
- [7] D. Monika, A. Ahmad, S. Wardani, and Solikhun, "Model Jaringan Syaraf Tiruan Dalam Memprediksi Ketersediaan Cabai Berdasarkan Provinsi," *Teknika*, vol. 8, no. 1, pp. 17–24, 2019, doi: 10.34148/teknika.v8i1.140.
- [8] B. Bagasta *et al.*, "JUMLAH NILAI EKSPOR DI PROVINSI NTB ( Implementation of Artificial Neural Network Algorithm Backpropagation Method to Predict Amount of Export Value in the NTB Province )," vol. 1, no. 2, 2019.
- [9] M. Rahul, I. Gunawan, F. Anggraini, S. Sumarno, and I. O. Kirana, "Analisa JST Untuk Memprediksi Pembuatan SIM Menggunakan Metode Algoritma Backpropagation," *J. Media Inform. Budidarma*, vol. 4, no. 1, p. 124, 2020, doi: 10.30865/mib.v4i1.1742.
- [10] M. Masruroh, "Perbandingan Metode Regresi Linear Dan Neural Network Backpropagation Dalam Prediksi Nilai Ujian Nasional Siswa Smp Menggunakan Software R," *Joutica*, vol. 5, no. 1, p. 331, 2020, doi: 10.30736/jti.v5i1.347.
- [11] Y. D. Lestari, "Jaringan syaraf tiruan untuk prediksi penjualan jamur menggunakan algoritma backpropagation," *J. ISD*, vol. 2, no. 1, pp. 40–45, 2017.
- [12] D. E. Rufiyanti, "Implementasi Jaringan Syaraf Tiruan Backpropagation Dengan Input Model Arima Untuk Peramalan Harga Saham," pp. 1–124, 2015.
- [13] P. Churn, D. A. N. Strategi, and R. Pada, "Prediksi churn dan strategi retensi pada kasus perusahaan telekomunikasi," 2021.
- [14] R. P. Masini, M. C. Medeiros, and E. F. Mendes, "Machine learning advances for time series forecasting," *arXiv*, pp. 1–44, 2020.