

Prediction of Nile Tilapia (*Oreochromis niloticus*) Harvest Yield in Brackishwater Pond Aquaculture Using XGBoost

Salamet Nur Himawan ^{1*}, Wisnu Arif ^{2*}, Nur Budi Nugraha ^{3*}

* Teknik Informatika, Politeknik Negeri Indramayu

snhimawan@polindra.ac.id¹, wisnuarif071@gmail.com², nurbudinugraha@polindra.ac.id³

Article Info

Article history:

Received 2025-09-29

Revised 2025-12-19

Accepted 2025-12-23

Keyword:

Nile Tilapia,
Brackishwater Ponds,
Random Forest,
Support Vector Machine,
XGBoost.

ABSTRACT

Nile tilapia aquaculture is one of the aquaculture subsectors with significant development potential. However, the productivity of Nile tilapia cultured in brackishwater ponds is often constrained by variability in technical factors such as the number of fingerlings stocked, pond area, stocking density, land status, planting season, and feed quantity. To address these challenges, a predictive model based on machine learning was developed. Data were collected through field observations and interviews with Nile tilapia farmers in Wanantara, Sindang, Indramayu. The data were then processed using label encoding and normalization techniques. The dataset was divided into 80% for training and 20% for testing. XGBoost, Random Forest, and Support Vector Regression algorithms were trained using hyperparameter tuning and five-fold cross-validation, and evaluated using RMSE and R² metrics. The results show that XGBoost achieved the best performance (R² = 0.9798 and RMSE = 442.05 kg), followed by Random Forest (R² = 0.955 and RMSE = 679.742 kg) and SVR (R² = 0.888 and RMSE = 1065.367 kg).



This is an open access article under the [CC-BY-SA](#) license.

I. PENDAHULUAN

Ikan nila (*Oreochromis niloticus*) merupakan salah satu komoditas perikanan air tawar yang memiliki nilai ekonomis tinggi [1], dan salah satu ikan air tawar yang mengandung protein tinggi dapat menjadi sumber protein penting bagi masyarakat Indonesia [2]. Selain itu, ikan nila merupakan ikan yang sangat berpotensi dikembangkan oleh pengusaha perikanan karena mudah berkembangbiak, tumbuh cepat, serta dapat bertahan hidup pada salinitas yang tinggi [3].

Kabupaten Indramayu dikenal sebagai salah satu sentra produksi ikan nila dengan potensi budidaya yang besar karena ketersediaan lahan tambak dan tradisi masyarakat dalam perikanan. Namun demikian, produktivitas tambak ikan nila masih menghadapi berbagai tantangan, seperti fluktuasi kualitas air, kepadatan tebar yang kurang tepat, jenis pakan, serta perubahan iklim yang berdampak pada tingkat kelangsungan hidup dan hasil panen [4], [5].

Beberapa penelitian menunjukkan faktor-faktor yang mempengaruhi pertumbuhan ikan nila, diantaranya kualitas air [6], [7], padat tebar [8] dan pemilihan sistem budidaya yang sesuai, seperti sistem bioflok atau kolam beton, dapat

memengaruhi pertumbuhan ikan nila [9]. Selain itu pemanfaatan pakan yang tepat juga merupakan salah satu faktor yang mempengaruhi pertumbuhan ikan [10].

Di sisi lain, perkembangan teknologi machine learning membuka peluang baru dalam bidang perikanan. Metode ini memungkinkan analisis data historis budidaya untuk membangun model prediksi yang akurat terhadap hasil panen. Sejumlah studi menunjukkan bahwa machine learning mampu digunakan untuk memprediksi parameter budidaya, seperti Random Forest, Support Vector Regression (SVR), maupun Neural Network mampu memberikan performa baik dalam memprediksi hasil panen ikan air tawar [11], [12]. Namun, penerapannya pada skala lokal seperti budidaya ikan nila di tambak Indramayu masih terbatas.

Selain meningkatkan akurasi prediksi, penggunaan machine learning juga membantu dalam mengoptimalkan penggunaan sumber daya budidaya. Salah satu manfaat penggunaan machine learning adalah model prediksi dapat digunakan untuk menyesuaikan jadwal pemberian pakan, mengatur padat tebar, serta mengantisipasi perubahan kualitas air yang dapat menurunkan produktivitas [11]. Dengan demikian, integrasi teknologi prediktif ini tidak hanya

meningkatkan hasil panen, tetapi juga berkontribusi terhadap keberlanjutan lingkungan dan efisiensi biaya operasional [12].

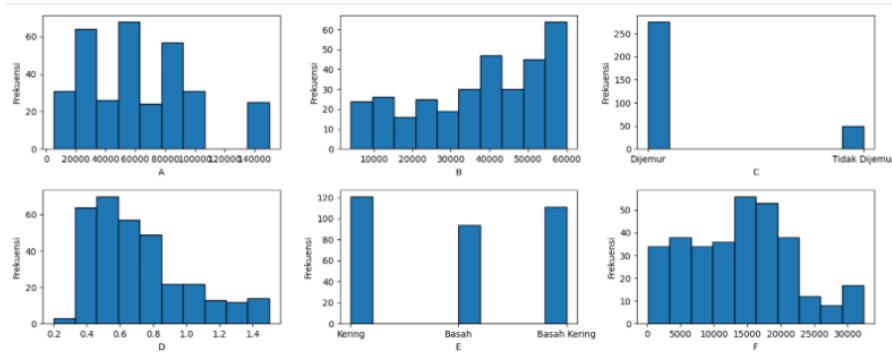
Berdasarkan hal tersebut, penelitian ini bertujuan untuk mengembangkan model prediksi hasil panen ikan nila menggunakan machine learning dengan studi kasus di Indramayu. Model ini diharapkan dapat membantu petambak dalam merencanakan budidaya, mengoptimalkan penggunaan sumber daya, serta meminimalkan risiko kegagalan produksi.

II. METODE

A. Pengumpulan Data

Proses pengumpulan data dilakukan untuk memperoleh informasi yang diperlukan dalam sistem prediksi produktivitas budidaya ikan nila. Sumber data diperoleh melalui survei langsung ke lokasi budidaya ikan nila serta wawancara dengan pemilik atau pengelola usaha budidaya. Data dikumpulkan dalam rentang waktu bulan April sampai Juni 2025 dan berlokasi di Kabupaten Indramayu.

Pengumpulan data dilakukan dengan wawancara kepada pembudidaya mengenai hal yang berpengaruh dalam hasil panen budidaya ikan nila kolam tambak. Penelitian ini menggunakan total 327 data. Data yang dikumpulkan mencakup fitur prediksi yaitu jumlah benih, luas kolam, kepadatan, status kolam, musim tebar, pakan untuk menentukan target hasil panen. Tabel 1 merupakan contoh dari data yang digunakan.



Gambar 1. Distribusi fitur

TABEL I
DATASET

Jumlah Benih (ekor)	Luas Lahan (m ²)	Status Lahan	Kepadatan (m ² /ekor)	Musim Tanam	Pakan (kg)	Hasil Panen (kg)
15000	8000	Dijemur	0.53	Kering	2400	2200
10000	8000	Dijemur	0.80	Basah	1500	1300
50000	11000	Dijemur	0.22	Basah	6000	5000
30000	11000	Dijemur	0.37	Kering	3600	3000
30000	10000	Dijemur	0.33	Basah	3900	3300
				Kering		

Dataset ini terdiri atas beberapa fitur utama yang menggambarkan proses budidaya. Fitur Jumlah Benih menunjukkan banyaknya benih yang ditebar, berkisar antara

10.000 hingga 50.000 ekor, sedangkan Luas Lahan berada dalam rentang 8.000–11.000 m² yang menentukan kapasitas tebar dan kebutuhan manajemen lahan. Status Lahan pada contoh data seluruhnya adalah “Dijemur”, menandakan bahwa lahan telah melalui proses pengeringan sebelum digunakan.

Fitur Kepadatan menggambarkan rasio antara jumlah benih dan luas lahan, dengan nilai 0,22–0,80 m²/ekor yang memengaruhi kondisi pertumbuhan ikan. Variabel Musim Tanam terdiri dari kategori Kering, Basah, dan Basah Kering, yang masing-masing dapat mempengaruhi kualitas lingkungan budidaya. Pakan, yang berkisar dari 1.500 hingga 6.000 kg, merepresentasikan jumlah nutrisi yang diberikan selama masa budidaya, sedangkan Hasil Panen, dengan nilai 1.300–5.000 kg, merupakan output akhir yang dipengaruhi oleh seluruh faktor input sebelumnya.

B. Data Pre-processing

1) Distribusi Fitur

Histogram dibuat untuk melihat pola distribusi data pada suatu fitur secara visual. Gambar 1 menunjukkan bahwa fitur Jumlah Benih memiliki distribusi yang cenderung merata, sedangkan fitur Luas Lahan dan Kepadatan tampak condong kiri dan kanan dengan rentang nilai yang tidak seimbang serta adanya outlier pada fitur Kepadatan. Fitur Pakan memiliki distribusi yang hampir normal.

Sementara itu, fitur kategorikal Status Lahan dan Musim tanam menunjukkan ketidakseimbangan kategori. Secara

keseluruhan, kombinasi fitur numerik yang condong dan fitur kategorikal yang tidak seimbang membuat model seperti Random Forest dan XGBoost menjadi pilihan yang sangat sesuai karena robust terhadap distribusi non-normal.

2) Pelabelan Fitur Kategorikal

Terdapat 2 fitur kategorikal yaitu Status Lahan dan Musim Tanam. Agar dapat digunakan oleh algoritma machine learning yang hanya menerima input numerik, dilakukan encoding pada nilai fitur kategorikal tersebut menjadi bilangan bulat menggunakan LabelEncoder dari pustaka scikit-learn [13]. Objek Label Encoder dibuat untuk masing-masing fitur. Berikut ini hasil pelabelan fitur kategorikal:

Label Encoding untuk fitur Status Lahan

0 = Dijemur, 1 = Tidak Dijemur

Label Encoding untuk fitur Musim Tanam

0 = Basah, 1 = Basah Kering, 2 = Kering

3) Normalisasi Data

Proses ini menggunakan StandardScaler dari library scikit-learn untuk proses normalisasi fitur [13]. StandardScaler akan menghitung rata-rata (mean) untuk setiap fitur. Kemudian menghitung varians sebagai rata-rata kuadrat dari selisih nilai terhadap mean menggunakan biase estimator yang hasilnya disimpan sebagai standar deviasi fitur. Setelah itu akan dilakukan transformasi nilai untuk setiap data dengan Persamaan 1 sebagai berikut:

$$z_i = \frac{x_i - \mu}{s} \quad (1)$$

z_i = Nilai normalisasi (z-score) dari data ke-i

s = Standar deviasi dari suatu fitur

x_i = Nilai data ke-i dari fitur tersebut

μ = Nilai rata-rata dari suatu fitur

C. Pengembangan Model Machine Learning

1) eXtreme Gradient Boosting (XGBoost)

XGBoost merupakan pengembangan dari metode *gradient boosting* yang memanfaatkan turunan pertama (gradien) dan turunan kedua (hessian) dari fungsi loss untuk mempercepat konvergensi dan meningkatkan stabilitas pembelajaran. XGBoost menambahkan regularisasi $L1$ dan $L2$ pada bobot daun pohon, sehingga mampu mengendalikan kompleksitas model dan mengurangi risiko *overfitting* [14]. Berikut algoritma XGBoost yang digunakan pada penelitian ini.

Algorithm 1. eXtreme Gradient Boosting (XGBoost)

Require: Data latih $D = \{(x_i, y_i)\}_{i=1}^N$, fungsi loss $L = (\hat{y}, y)$, jumlah iterasi boosting T , learning rate η , parameter regulasi λ, α, γ , rasio *subsample*, rasio *colsample*, parameter pohon: *max depth*, *min child weight*

Ensure: Model akhir $F(x)$

- 1: Inisialisasi model: $F_0(x) = \arg\min_c \sum_{i=1}^N L(c, y_i)$
- 2: **for** $t = 1$ **to** T **do**
- 3: **for** $i = 1$ **to** N **do**
- 4: Hitung gradien: $g_i = \frac{\partial L(F(x_i), y_i)}{\partial F(x_i)}$
- 5: Hitung hessian: $h_i = \frac{\partial^2 L(F(x_i), y_i)}{\partial F(x_i)^2}$
- 6: **end for**
- 7: Lakukan subsampling baris dan kolom
- 8: Bangun pohon regresi $f_t(x)$ dengan mencari split terbaik:
- 9: Hitung gain split: $Gain = \frac{1}{2} \left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} + \frac{(G_R + G_L)^2}{H_L + H_R + \lambda} \right) - \gamma$
- 10: Jika tidak ada split valid, bobot daun: $w^* = -\frac{G}{H + \lambda}$
- 11: Perbarui model: $F(x) \leftarrow F(x) + \eta f_t(x)$
- 12: **end for**
- 13: **return** $F(x) \leftarrow F_0(x) + \sum_{t=1}^T \eta f_t(x)$

2) Random Forest (RF)

Random Forest adalah algoritma berbasis *ensemble* yang membangun sekumpulan pohon keputusan menggunakan teknik *bootstrap aggregating (bagging)* dan pemilihan subset fitur secara acak pada setiap node. Prediksi akhir diperoleh dengan merata-ratakan hasil semua pohon pada kasus regresi. Pendekatan ini membuat Random Forest tahan terhadap *overfitting* dan robust terhadap data yang mengandung noise [15]. Berikut algoritma RF yang digunakan pada penelitian ini.

Algorithm 2. Random Forest

Require: Data latih $D = \{(x_i, y_i)\}_{i=1}^N$, jumlah pohon T , jumlah fitur acak m , kedalaman maksimum *max_depth*

Ensure: Model akhir $F(x)$

- 1: **for** $t = 1$ **to** T **do**
- 2: Ambil sampel bootstrap D_t dari D
- 3: Bangun pohon regresi $h_t(x)$ dengan langkah:
 - 4: **for** setiap node dalam pohon **do**
 - 5: Pilih acak m fitur dari total fitur
 - 6: Cari split terbaik berdasarkan *MSE* minimum
 - 7: Bagi node ke kiri/kanan sesuai split
 - 8: Hentikan jika kedalaman = *max_depth* atau jumlah sampel terlalu kecil
 - 9: **end for**
 - 10: Simpan pohon $h_t(x)$
- 11: **end for**
- 12: Definisikan prediksi akhir: $F(x) = \frac{1}{T} \sum_{t=1}^T h_t(x)$
- 13: **return** $F(x)$

3) Support Vector Regression (SVR)

SVR merupakan pengembangan dari *Support Vector Machine (SVM)* yang dirancang untuk menyelesaikan masalah regresi. Algoritma ini mencari fungsi regresi yang memiliki deviasi maksimum dari data pelatihan, dengan meminimalkan kompleksitas model. SVR dapat menggunakan kernel linear maupun kernel non-linear untuk memetakan data ke ruang berdimensi lebih tinggi sehingga pola non-linear dapat dipelajari [16]. Berikut algoritma SVR yang digunakan pada penelitian ini.

Algorithm 3. SVR

Require: Data latih $D = \{(x_i, y_i)\}_{i=1}^N$, parameter $C > 0$, $\epsilon \geq 0$, pilihan kernel K : Linear $K(x_i, x_j) = x_i^T x_j$ atau *RBF*:

$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, toleransi *tol*, iterasi maksimum *max_iter*

Ensure: Koefisien dual (α, α^*) , bias b , prediktor $\hat{y}(x)$

- 1: Inisialisasi $\alpha = \alpha^* = 0$, $\alpha = 0$
- 2: **for** $t = 1$ **to** max_iter **do**
- 3: Cek kondisi KKT; jika semua pelanggaran $< tol$, **break**
- 4: Pilih pasangan (i, j) (heuristik SMO)
- 5: Hitung $\eta = K(x_i, x_i) + K(x_j, x_j) - 2K(x_i, x_j)$
- 6: **if** $\eta \leq 0$ **then**
- 7: **continue**
- 8: **end if**
- 9: Perbarui (α_i, α_j) atau (α_i^*, α_j^*) , proyeksikan ke $[0, C]$, jaga $\sum_i \alpha_i - \alpha_i^* = 0$

-
- 10: Perbarui bias b dari rumus SMO
 11: **end for**
 12: Definisikan prediktor: $\hat{y}(x) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(x_i, x) + b$
 13: **return** (α, α^*, b) dan $\hat{y}(\cdot)$
-

D. Evaluasi Model

Untuk menilai performa model dalam memprediksi hasil panen ikan nila kolam tambak, penelitian ini menggunakan dua metrik utama, yaitu Root Mean Square Error (RMSE) dan Koefisien Determinasi (R^2).

1) Root Mean Square Error (RMSE)

RMSE digunakan untuk mengukur besarnya deviasi antara nilai prediksi model dengan nilai aktual pada data uji. Metrik ini sangat relevan karena menyajikan kesalahan dalam satuan yang sama dengan target (kg hasil panen). Nilai RMSE yang lebih kecil menunjukkan bahwa prediksi model lebih mendekati nilai sebenarnya. Dengan kata lain, semakin rendah RMSE, semakin tinggi ketepatan prediksi model. RMSE dapat dihitung menggunakan Persamaan 2.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2)$$

- y_i : nilai aktual ke- i
 \hat{y}_i : nilai prediksi ke- i
 n : jumlah data

2) Koefisien Determinasi (R^2)

R^2 digunakan untuk mengukur proporsi variasi data target yang dapat dijelaskan oleh model. Nilai R^2 berkisar antara 0 hingga 1, di mana nilai mendekati 1 menunjukkan bahwa model mampu menjelaskan sebagian besar variasi data. Sebaliknya, nilai R^2 yang rendah menunjukkan bahwa model tidak mampu menangkap pola hubungan antara fitur dan hasil panen dengan baik. R^2 dihitung menggunakan Persamaan 3.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (3)$$

- y_i : nilai aktual ke- i
 \hat{y}_i : nilai prediksi ke- i
 \bar{y} : nilai rata-rata dari data aktual
 n : jumlah data

III. HASIL DAN PEMBAHASAN

Dataset yang telah diproses melalui tahapan preprocessing data terlebih dahulu dipersiapkan dengan memisahkan fitur target dari fitur prediksi. Pada proses ini, fitur target berupa Hasil Panen, sementara fitur lainnya digunakan sebagai fitur input. Selanjutnya, dataset dibagi menjadi dua bagian, dengan proporsi 80 % sebagai data latih (*training*) dan 20 % sebagai data uji (*testing*).

A. Hyperparameter Tuning

Dalam pengembangan model machine learning, pemilihan hyperparameter yang tepat merupakan salah satu faktor penting untuk mendapatkan performa prediksi yang optimal. Hyperparameter merupakan parameter yang tidak dipelajari langsung oleh model dari data, melainkan harus ditentukan sebelum proses pelatihan. Pada penelitian ini, proses hyperparameter tuning dilakukan menggunakan metode grid search dan cross-validation.

Grid Search digunakan untuk menguji berbagai kombinasi nilai hyperparameter secara sistematis dalam ruang pencarian yang telah ditentukan. Cross-validation diterapkan untuk mengurangi risiko overfitting dan memastikan model dapat melakukan generalisasi dengan baik. Teknik k-fold cross-validation $k = 5$ digunakan, sehingga dataset dibagi menjadi 5 bagian, dan setiap bagian bergantian digunakan sebagai data uji.

Melalui kombinasi kedua metode tersebut, diperoleh konfigurasi hyperparameter terbaik untuk masing-masing algoritma. Hasil tuning kemudian dibandingkan menggunakan metrik evaluasi RMSE dan R^2 . Model dengan kombinasi hyperparameter terbaik inilah yang digunakan untuk memprediksi hasil panen ikan nila di Indramayu. Berikut Tabel II, Tabel III dan Tabel IV merupakan hyperparameter dengan ruang pencarian dan nilai yang dipilih.

TABEL II
HYPERPARAMETER TUNING XGBOOST

Hyperparameter	Ruang Pencarian	Nilai
<code>n_estimator</code>	100, 300, 500	300
<code>learning_rate</code>	0,01; 0,05; 0,1	0,05
<code>max_depth</code>	3, 5, 7	3
<code>subsample</code>	0,7; 0,9; 1,0	0,7
<code>colsample_bytree</code>	0,7; 0,9; 1,0	0,9

TABEL III
HYPERPARAMETER TUNING RANDOM FOREST

Hyperparameter	Ruang Pencarian	Nilai
<code>n_estimators</code>	200,300,500,800	500
<code>max_depth</code>	5, 10, 15	15
<code>min_samples_split</code>	2, 5, 10	2
<code>min_samples_leaf</code>	1, 2, 4, 6	1
<code>max_features</code>	sqrt; log2; 0,7; 0,9	sqrt
<code>bootstrap</code>	True	True

TABEL IV
HYPERPARAMETER TUNING SVR

Hyperparameter	Ruang Pencarian	Nilai
C	0,1; 1; 10; 100	10
epsilon	0,01; 0,1; 0,5	0,1
kernel	rbf, linear	rbf
gamma	scale, auto	scale

B. Evaluasi Model

Setelah dilakukan proses hyperparameter tuning, setiap algoritma diuji menggunakan data uji untuk menilai performa prediksi. Evaluasi dilakukan dengan dua metrik utama, yaitu RMSE dan R². Tabel V merupakan hasil evaluasi pada setiap model.

TABEL V
HASIL EVALUASI MODEL

Model Machine learning	RMSE (kg)		R-squared	
	Mean	Standar Deviasi	Mean	Standar Deviasi
XGBoost	442,049	47,960	0,979	0,006
Random Forest	679,742	64,520	0,955	0,007
SVR	1065,725	87,673	0,888	0,031

Hasil evaluasi menunjukkan bahwa XGBoost memiliki performa terbaik di antara model yang diuji. Model ini mampu menghasilkan RMSE sebesar 422,049 kg dan R² sebesar 0,979. Nilai R² yang mendekati 1 menunjukkan bahwa model dapat menjelaskan sebagian besar variasi hasil panen ikan nila berdasarkan fitur input.

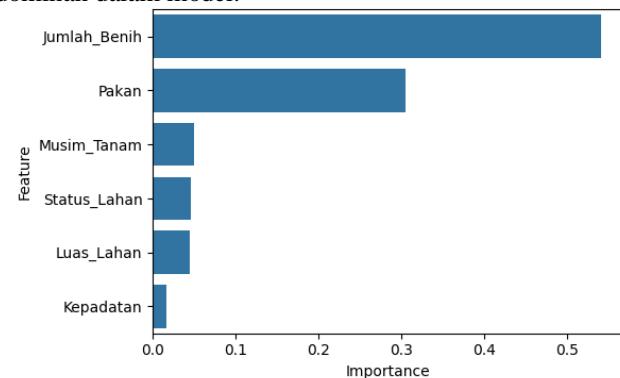
Random Forest (RF) menempati posisi kedua dengan performa yang cukup kompetitif. RF menghasilkan RMSE sebesar 679,742 kg dan R² sebesar 0,955. Sementara itu, Support Vector Regression (SVR) menghasilkan RMSE sebesar 1065,725 kg dan R² sebesar 0,888. Walaupun nilai ini masih menunjukkan kemampuan prediksi yang baik, performa SVR cenderung lebih rendah dibandingkan dua model lainnya.

Secara keseluruhan, hasil evaluasi membuktikan bahwa XGBoost adalah model paling optimal untuk prediksi hasil panen ikan nila di Indramayu. Perbedaan nilai error antar model tidak terlalu jauh, namun konsistensi XGBoost dalam memberikan hasil dengan error rendah dan nilai R² tinggi menjadikannya pilihan utama.

C. Interpretasi Fitur

Hasil analisis feature importance dari model dapat dilihat pada Gambar 2, menunjukkan bahwa fitur paling berpengaruh terhadap hasil panen ikan nila adalah jumlah benih dan jumlah pakan. Jumlah benih menentukan potensi biomassa awal yang tersedia di tambak, sedangkan jumlah pakan merupakan faktor utama yang memengaruhi pertumbuhan ikan hingga mencapai ukuran panen. Kedua fitur ini berhubungan

langsung dengan jumlah panen, sehingga kontribusinya dominan dalam model.



Gambar 2. Feature importance

Selain itu, fitur musim tanam juga tercatat berpengaruh, meskipun kontribusinya lebih kecil dibandingkan benih dan pakan. Hal ini wajar karena kondisi musim, seperti suhu rata-rata dan curah hujan, dapat memengaruhi kualitas air tambak serta ketersediaan pakan alami. Faktor status lahan berhubungan dengan kesuburan tanah dan kondisi dasar tambak yang berdampak pada kualitas media budidaya.

Fitur lain seperti luas lahan dan kepadatan juga memberikan pengaruh signifikan. Luas lahan menentukan kapasitas produksi total, sedangkan kepadatan tebar memengaruhi tingkat kompetisi antar ikan dalam hal pakan dan oksigen. Penebaran dengan kepadatan terlalu tinggi berisiko meningkatkan stres ikan dan menurunkan survival rate, sementara kepadatan yang optimal akan meningkatkan efisiensi pakan serta produktivitas lahan.

D. Implikasi

Implikasi dari hasil ini adalah bahwa petambak perlu memperhatikan keseimbangan antara jumlah benih, jumlah pakan, kepadatan, dan kapasitas lahan dalam merencanakan budidaya. Selain itu, faktor musiman dan status lahan tidak boleh diabaikan, karena dapat memengaruhi keberhasilan panen secara tidak langsung. Model prediksi yang dikembangkan dapat membantu petambak menyesuaikan strategi budidaya berdasarkan kondisi musim dan karakteristik lahan, sehingga hasil panen dapat lebih konsisten dan risiko kerugian dapat diminimalkan.

Dari sisi praktis, model ini dapat diimplementasikan sebagai sistem pendukung keputusan (decision support system) bagi petambak di Indramayu. Dengan memasukkan data kondisi budidaya yang sedang berlangsung, petambak dapat memperoleh estimasi hasil panen sebelum waktu panen sebenarnya.

Hasil analisis feature importance menunjukkan bahwa variabel Jumlah Benih dan Pakan memiliki pengaruh paling dominan terhadap hasil panen. Secara praktis, hal ini mengindikasikan bahwa kebijakan pengelolaan tambak perlu menitikberatkan pada penentuan padat tebar yang proporsional serta pengaturan pemberian pakan yang efisien. Model menunjukkan bahwa peningkatan jumlah benih tidak

selalu berbanding lurus dengan hasil panen apabila tidak diimbangi dengan kapasitas lahan dan ketersediaan pakan.

Di sisi lain, meskipun variabel seperti Musim Tanam, Status Lahan, dan Kepadatan menunjukkan pengaruh relatif kecil, informasi ini tetap penting untuk merumuskan kebijakan pendukung. Seperti, rendahnya kontribusi musim tanam mengindikasikan bahwa produktivitas dapat tetap stabil sepanjang musim, selama manajemen pakan dan penebaran dilakukan dengan benar. Hal ini dapat menjadi dasar bagi pemilik atau pengelola budidaya terkait untuk menyusun kalender tanam yang lebih fleksibel serta memberikan panduan teknis mengenai pengeringan lahan sebelum penebaran benih.

Secara keseluruhan, temuan ini menunjukkan bahwa integrasi model prediktif dapat memperkuat pengambilan keputusan berbasis data, baik dalam optimasi praktik budidaya di tingkat pembudidaya maupun dalam perumusan kebijakan peningkatan produktivitas sektor perikanan. Namun, terdapat keterbatasan yang perlu diperhatikan. Dataset yang digunakan dalam penelitian ini relatif terbatas dan hanya mencakup beberapa tambak di Indramayu. Variasi faktor lingkungan di lokasi lain bisa berbeda sehingga akurasi model mungkin berkurang jika diterapkan secara luas tanpa penyesuaian. Untuk itu, penelitian selanjutnya perlu memperluas cakupan data, menambahkan fitur baru seperti curah hujan atau salinitas, serta melakukan uji coba implementasi lapangan dengan partisipasi langsung petambak.

Dengan demikian, hasil analisis dan implikasi ini menegaskan bahwa model prediksi berbasis machine learning tidak hanya berfungsi sebagai alat estimasi, tetapi juga sebagai pedoman praktis dalam manajemen budidaya, dengan potensi besar untuk meningkatkan produktivitas dan keberlanjutan usaha perikanan di Indramayu.

IV. KESIMPULAN

Prediksi hasil panen ikan nila (*Oreochromis niloticus*) sangat penting untuk mendukung keberlanjutan budidaya di Indramayu. Dengan adanya sistem prediksi, petambak dapat memperkirakan produktivitas lebih awal, sehingga dapat memaksimalkan hasil panenan dan mengurangi risiko kerugian.

Dalam penelitian ini, digunakan sejumlah fitur penting yang memengaruhi produktivitas tambak, antara lain jumlah benih, luas lahan, kepadatan ikan, status lahan, musim tanam dan jumlah pakan. Fitur-fitur tersebut dipilih berdasarkan wawancara dengan petambak dan diambil berdasarkan fitur yang mudah didapat oleh petambak.

Hasil implementasi machine learning menunjukkan bahwa model prediksi mampu memberikan performa yang baik. Di antara model yang diuji, Xgboost memberikan hasil yang paling unggul dengan R Score dan RMSE sebesar 0,97 dan 442,05 kg.

UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Kementerian Pendidikan Tinggi, Sains dan Teknologi dalam skema Penelitian Dosen Pemula.

DAFTAR PUSTAKA

- [1] K. Raje, "Tilapia Market Report 2025 (Global Edition)," Jul. 2025.
- [2] S. E. Matondang, "Perbandingan Kadar Protein Ikan Air Tawar Dan Ikan Air Laut," *LAVOISIER: Chemistry Education Journal*, vol. 1, no. 1, pp. 9–16, Jul. 2022, doi: 10.24952/lavoisier.v1i1.5723.
- [3] R. Aziz and E. Barades, "Adaptation Of Tilapia Juvenile (*Oreochromis niloticus*) On Different Salinity Increases," *Jurnal Perikanan Unram*, vol. 11, no. 2, pp. 251–258, Nov. 2021, doi: 10.29303/jp.v11i2.262.
- [4] K. Haga Mendrofa and E. Krisdila Zebua, "Analisis Faktor-Faktor yang Mempengaruhi Produktivitas Budidaya Ikan Nila di Indonesia : Studi Literatur," *Ilmu Kedokteran Hewan*, vol. 3, no. 1, 2025, doi: 10.62951/zoologi.v3i1.104.
- [5] R. R. Lamangkaraka, Mulis, Y. Koniyo, and M. Alvionita, "Analisis Kualitas Air Pada Sistem Budidaya Ikan Nila (*Oreocromis nilotius*) di Balai Benih Ikan Andalas, Kota Gorontalo," *Jurnal Ilmiah Perikanan dan Kelautan*, vol. 11, no. 2, 2024.
- [6] D. Azhari and A. M. Tomasoa, "Kajian Kualitas Air Dan Pertumbuhan Ikan Nila (*Oreochromis niloticus*) Yang Dibudidayakan Dengan Sistem Akuaponik," *Jurnal Akuatika Indonesia*, vol. 3, no. 2, p. 84, Sep. 2018.
- [7] R. Akhmad Akbar Trinanda Putra, Emma Yuliani, and Sri Wahyuni, "Pengaruh Kualitas Air Untuk Pertumbuhan Budidaya Ikan Nila (*Oreochromis Niloticus*) Di Kecamatan Glenmore Kabupaten Banyuwangi," *Jurnal Teknologi dan Rekayasa Sumber Daya Air*, vol. 5, no. 1, pp. 498–507, Jan. 2025, doi: 10.21776/ub.jtresda.2025.005.01.047.
- [8] M. Arzad and A. Fahrizal, "Pengaruh Padat Tebar Terhadap Pertumbuhan Ikan Nila (*Oreochromis niloticus*) Dalam Sistem Akuaponik," 2019.
- [9] A. Kurniaji, Y. Yunarty, A. Anton, Z. Usman, E. Wahid, and K. Rama, "Pertumbuhan dan konsumsi pakan ikan nila (*Oreochromis niloticus*) yang dipelihara dengan sistem bioflok," *Sains Akuakultur Tropis*, vol. 5, no. 2, pp. 197–203, Oct. 2021, doi: 10.14710/sat.v5i2.11824.
- [10] N. F. Bulontio, S. R. Kalaka, and S. Nursinar, "Pemberian Pakan yang Berbeda Terhadap Pertumbuhan Benih Ikan Nila (*Oreochromis niloticus*)," *Jurnal Ilmiah Perikanan dan Kelautan*, vol. 12, no. 4, Dec. 2024.
- [11] I. Fadillah, T. S. Ramadhani, and Z. A. Tifazani, "Pendugaan Suhu Dan Ph Budidaya Ikan Air Tawar Menggunakan Support Vector Regression (SVR)," vol. 11, no. 2, 2023.
- [12] C. M. Suprapto, W. S. J. Saputra, and F. P. Aditiawan, "Prediksi Hasil Panen Budidaya Ikan Lele Dari Mitra Panen Menggunakan Algoritma Support Vector Regression," *Jurnal Komputer dan Informatika*, vol. 12, no. 2, pp. 158–165, Oct. 2024, doi: 10.35508/jicon.v12i2.13187.
- [13] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. null, pp. 2825–2830, Nov. 2011.
- [14] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [15] L. Breiman, "Random Forests," *Mach Learn*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.
- [16] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Stat Comput*, vol. 14, no. 3, pp. 199–222, Aug. 2004, doi: 10.1023/B:STCO.0000035301.49549.88.