



Prediksi Stok Tanaman Hidroponik dengan Artificial Intelligence: Ensemble Learning dengan Optimasi Evolusioner

Putu Bagus Adidyana Anugrah Putra¹, Septian Geges², Oktaviani Enjela Putri³, I Made Bayu Artha Pratama⁴

putubagus@it.upr.ac.id¹, septian.geges@it.upr.ac.id², oktavianienjelaputri@mhs.eng.upr.ac.id³,

madebayu26@mhs.eng.upr.ac.id⁴

Universitas Palangka Raya

Teknik Informatika

Jl. Hendrik Timang, Palangka Raya

ARTICLE INFO

Article history:

Received : 29 Agustus 2024

Received in revised : 18 November 2024

Accepted : 2 Desember 2024

Available online : 12 Desember 2024

ABSTRACT

Hydroponic plant cultivation is booming, but stock and sales are hard to predict. Poor prediction can cause farmers to overstock and lose money. This study suggests a framework that uses several machine learning models, including Linear Regression (LR), Random Forest (RF), Decision Tree (DT), and Extreme Gradient Boosting. "Ensemble Learning," which combines these models, should yield more accurate and generalizable results than a single model. This framework is assessed using historical hydroponic plant sales data and related factors like price, weather, and market trends. The model's performance is measured by the difference between predictions and actual values using RMSE and MAE metrics. This framework should improve hydroponic plant stock and sales predictions. Farmers can make better production, inventory, and harvest distribution decisions. Besides reducing financial losses, this reduces food waste and improves food security.

Keywords: Ensemble Learning, Stock Prediction, Artificial Intelligence.

1. Pendahuluan

Dalam menghadapi meningkatnya populasi global dan akibatnya meningkatnya permintaan pangan, praktik pertanian berkelanjutan menjadi sangat penting [1][2]. Salah satu tantangan signifikan dalam pertanian adalah manajemen produksi tanaman yang efisien untuk meminimalkan limbah dan meningkatkan ketahanan pangan [3][4][5]. Pertanian hidroponik, merupakan model inovatif untuk menanam tanaman tanpa tanah yang menghadirkan solusi menjanjikan, karena penggunaan sumber daya yang efisien dan kondisi pertumbuhan yang terkendali [6]. Tanaman hidroponik memang memiliki banyak manfaat, namun mengelola inventaris dan memprediksi penjualannya masihlah rumit. Faktor-faktor seperti siklus pertumbuhan tanaman, permintaan pasar, dan kondisi lingkungan perlu dianalisis secara mendalam untuk mencapai hasil yang optimal [7][8]. Namun, mengoptimalkan manajemen inventaris dalam pertanian hidroponik tetap menjadi masalah penting yang secara langsung berdampak pada limbah makanan dan pemanfaatan sumber daya.

Prediksi stok dan penjualan tanaman yang akurat sangat penting untuk mengoptimalkan manajemen persediaan dalam pertanian hidroponik [9][10]. Metode tradisional sering gagal dalam memberikan yang dibutuhkan untuk pengambilan keputusan yang efektif [11][12]. Sebagai solusi untuk mengatasi kompleksitas ini, model prediktif canggih yang menggunakan machine learning mulai banyak diminati [9]. Penelitian ini memperkenalkan model *ansambel* yang menggabungkan model *regresi linier* (LR), *random forest* (RF), *decision tree* (DT), dan *extreme gradient boosting*

Received August 29, 2024; Revised November 18, 2024; Accepted December 2, 2024

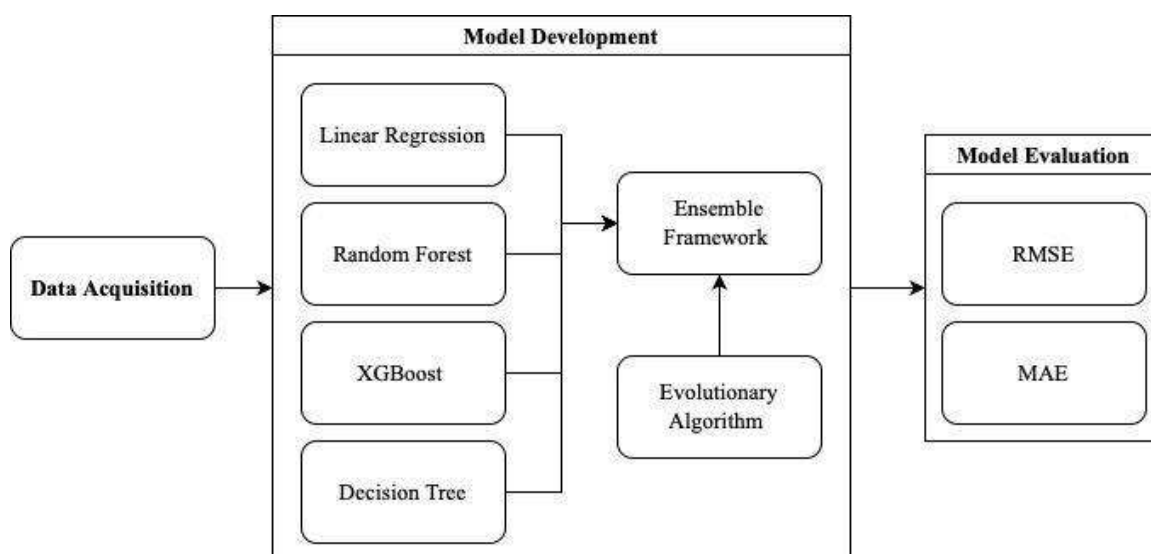
* Putu Bagus Adidyana Anugrah Putra, e-mail address: putubagus@it.upr.ac.id

(XGBoost), yang dioptimalkan melalui algoritma *evolutioner* (EA) untuk meningkatkan akurasi prediksi dalam budidaya tanaman hidroponik.

2. Metode penelitian

2.1 Alur Penelitian

Metodologi penelitian merupakan pendekatan sistematis yang digunakan untuk merancang, melaksanakan, dan menganalisis data dalam suatu penelitian. Metodologi ini menggabungkan 4 *machine learning* dan analisis data untuk mengoptimalkan manajemen stok tanaman hidroponik. Adapun tahapan dalam penelitian ini terdiri dari beberapa tahapan seperti pada gambar 2.1 di bawah ini.



Gambar 2.1. Alur Penelitian

2.2 Data Acquisition

Data Acquisition atau Pengumpulan Data adalah proses mengumpulkan data mentah dari berbagai sumber untuk kemudian dianalisis. Ini adalah langkah pertama yang sangat penting dalam setiap proyek yang berbasis data. Kualitas dan relevansi data yang dikumpulkan akan sangat mempengaruhi akurasi dan manfaat dari hasil analisis yang dihasilkan. Data Acquisition sangat penting dilakukan karena akan menjadi dasar untuk melakukan analisis. Data yang baik adalah fondasi yang kuat untuk membangun analisis yang akurat. Data yang relevan membantu dalam membuat keputusan bisnis yang lebih baik. Data juga dapat membantu mengidentifikasi tren pasar, perilaku konsumen, dan lain sebagainya. Dalam membangun model, data digunakan untuk melatih model machine learning dan membuat prediksi.

Proses pengumpulan data dilakukan dengan beberapa langkah:

- Identifikasi Sumber Data: Menentukan dari mana data akan diambil, bisa dari sumber internal (data perusahaan) atau eksternal (data publik, data pasar).
- Metode Pengumpulan: Memilih metode yang tepat, seperti survei, wawancara, pengamatan, web scraping, atau menggunakan API.
- Pengumpulan Data: Melakukan proses pengumpulan data sesuai dengan metode yang telah dipilih.
- Pembersihan Data: Memeriksa dan membersihkan data dari kesalahan, inkonsistensi, atau nilai yang hilang.

2.3 Model Development

Pengembangan model adalah proses menciptakan sebuah rumus atau persamaan matematika yang dapat memprediksi suatu kejadian di masa depan berdasarkan data yang ada. Dalam kasus penelitian tentang tanaman hidroponik, model ini akan digunakan untuk memprediksi jumlah stok tanaman yang tersisa atau jumlah penjualan yang akan terjadi. Proses pengembangan model melibatkan beberapa tahap, mulai dari pengumpulan data historis, pemilihan model yang sesuai, pelatihan model menggunakan data tersebut, hingga evaluasi kinerja model. Setelah model dianggap baik, model ini dapat digunakan untuk membuat prediksi.

Pentingnya pengembangan model terletak pada kemampuannya untuk membantu memahami hubungan antara berbagai faktor yang mempengaruhi stok tanaman hidroponik, serta memberikan prediksi yang akurat sehingga dapat digunakan untuk pengambilan keputusan yang lebih baik dalam bisnis budidaya hidroponik. Dengan model yang baik, Anda dapat mengatur produksi, distribusi, dan harga jual dengan lebih efektif.

2.4 Model Evaluation

Model dievaluasi menggunakan RMSE dan MAE untuk mengukur metrik evaluasi regresi standar. Mengambil individu dengan bobot terbaik yang ditemukan oleh EA. Dengan menggunakan bobot yang dioptimalkan tadi, akan dibuat prediksi ansambel pada tes set. Untuk menunjukkan manfaat menggabungkan prediksi, kinerja model ansambel akan dibandingkan dengan model tunggal. Untuk melakukan perbandingan, kita akan menghitung nilai RMSE dan MAE untuk masing-masing model. Kemudian, kita akan membandingkan nilai RMSE dan MAE dari model ensemble dengan nilai RMSE dan MAE dari model tunggal.

3. Hasil dan Pembahasan

3.1. Data Acquisition

Data yang digunakan dalam penelitian ini berasal dari kegiatan pertanian hidroponik, khususnya budidaya tanaman sawi. Data dikumpulkan selama 28 minggu dan mencakup beberapa siklus panen untuk memastikan kualitas data yang baik. Tujuan utama dari penelitian ini adalah untuk memprediksi jumlah stok tanaman sawi yang tersisa. Variabel-variabel yang digunakan sebagai fitur dalam model prediksi meliputi jumlah tanaman yang ditanam, dipanen, dan total penjualan. Data telah melalui proses pembersihan dan normalisasi untuk meningkatkan kualitas dan keseragaman data.

Himpunan data yang telah disiapkan akan digunakan untuk membangun model prediksi yang dapat memperkirakan jumlah stok tanaman sawi yang tersisa. Dengan adanya model prediksi yang akurat, diharapkan dapat membantu dalam pengambilan keputusan terkait produksi, dan pengelolaan stok tanaman sawi di masa mendatang. Model prediksi yang akan dibangun akan mempertimbangkan faktor-faktor seperti jumlah tanaman yang ditanam, dipanen, serta persediaan yang ada.

Tabel 3.1 Contoh Parameter Data

Week	Number of Plants Planted	Number Harvested	Harvest Failure	Stock Data	Sales Data	Remaining Stock
1	1000	973	27	973	967	6
2	1000	968	32	974	969	5
3	1000	976	24	981	978	3
4	1000	971	29	974	974	0
5	1000	965	35	965	962	3
6	1000	979	21	982	979	3
7	1000	974	26	977	971	6
8	1000	976	24	982	979	3

9	1000	980	20	983	977	6
10	1000	973	27	979	972	7

3.2. Model Development

Tahap kedua adalah membangun kerangka kerja ansambel yang menggabungkan kemampuan prediktif model XGBoost, LR, DT dan RF yang merupakan tujuan dari fase pengembangan model.

3.2.1. Ensemble Framework Construction

Tahap pengembangan model melibatkan pembuatan kerangka kerja ensemble yang menggabungkan XGBoost, LR, DT dan RF. Proses pengembangan model meliputi langkah-langkah berikut:

Model regresi linear (LR) dilatih untuk menangkap hubungan linear antara fitur dan variabel target. Regresor random forest (RF) digunakan untuk menangani non-linearitas dan interaksi antara fitur. Model ini dikonfigurasi dengan 100 pohon untuk menyeimbangkan kinerja dan efisiensi komputasi. Regresor XGBoost digunakan untuk kinerja tinggi dan kemampuannya menangani hubungan kompleks dalam data. Model ini diatur dengan 100 putaran boosting dan dioptimalkan untuk tujuan kesalahan kuadrat.

```
# Individual Model Training Functions
def train_linear_regression(X_train, y_train):
    model = LinearRegression()
    model.fit(X_train, y_train)
    return model

def train_random_forest(X_train, y_train):
    model = RandomForestRegressor(n_estimators=100, random_state=42)
    model.fit(X_train, y_train)
    return model

def train_xgboost(X_train, y_train):
    model = XGBRegressor(objective='reg:squarederror', n_estimators=100, random_state=42)
    model.fit(X_train, y_train)
    return model

def train_gradient_boosting(X_train, y_train):
    model = GradientBoostingRegressor(learning_rate=0.1, n_estimators=100, random_state=42)
    model.fit(X_train, y_train)
    return model

def train_decision_tree(X_train, y_train):
    model = DecisionTreeRegressor(random_state=42)
    model.fit(X_train, y_train)
    return model

# Train Individual Models
model_lr = train_linear_regression(X_train, y_train)
model_rf = train_random_forest(X_train, y_train)
model_xgb = train_xgboost(X_train, y_train)
model_gbr = train_gradient_boosting(X_train, y_train)
model_dt = train_decision_tree(X_train, y_train)
```

Dataset dibagi menjadi dua bagian: 20% digunakan untuk pengujian dan 80% untuk pelatihan model. Data pelatihan digunakan untuk melatih setiap model dalam ensemble secara independen. Setiap model mempelajari pola dan hubungan dalam data melalui proses pelatihan ini. Setelah tahap pelatihan model, sebuah fungsi didefinisikan untuk membuat prediksi ensemble.

```

# Train Individual Models
model_lr = train_linear_regression(X_train, y_train)
model_rf = train_random_forest(X_train, y_train)
model_gh = train_gradient_boosting(X_train, y_train)
model_gbr = train_gradient_boosting(X_train, y_train)
model_dt = train_decision_tree(X_train, y_train)

# Ensemble Prediction Function
def ensemble_predict(X, model_lr, model_rf, model_gh, model_gbr, model_dt, weight_lr, weight_rf, weight_gh, weight_gbr, weight_dt):
    predictions_lr = model_lr.predict(X)
    predictions_rf = model_rf.predict(X)
    predictions_gh = model_gh.predict(X)
    predictions_gbr = model_gbr.predict(X)
    predictions_dt = model_dt.predict(X)
    predictions_ensemble = weight_lr * predictions_lr + weight_rf * predictions_rf + weight_gh * predictions_gh + weight_gbr * predictions_gbr + weight_dt * predictions_dt
    return predictions_ensemble

```

Fungsi ini memiliki peran penting dalam menggabungkan prediksi individual dari ketiga model yang telah dilatih. Aspek kunci di sini adalah fungsi tidak hanya rata-rata prediksi. Sebaliknya, ia memasukkan bobot untuk setiap model. Bobot ini menentukan kepentingan relatif dari setiap prediksi model dalam prediksi ensemble akhir. Konsep di balik bobot adalah bahwa beberapa model mungkin lebih andal untuk titik data atau jenis masalah tertentu. Dengan menetapkan bobot yang dioptimalkan untuk model dengan kinerja lebih kuat dan menyesuaikan bobot sesuai kebutuhan, prediksi ensemble berpotensi mengungguli model tunggal mana pun dalam ensemble.

3.2.2. Evolusioner Algorithm for Weight Optimization

Untuk meningkatkan akurasi prediksi dari model ensemble, digunakan sebuah teknik optimasi yang disebut algoritma evolusioner. Tujuannya adalah menentukan bobot yang tepat untuk setiap model dalam ensemble sehingga menghasilkan prediksi terbaik.

Algoritma evolusioner bekerja dengan mencoba berbagai kombinasi bobot secara iteratif. Dalam proyek ini, digunakan library DEAP untuk implementasi algoritma evolusioner. Proses optimasi melibatkan langkah-langkah berikut:

1. **Inisialisasi Populasi:** Dimulai dengan membuat sejumlah individu (dalam hal ini, kombinasi bobot untuk ketiga model) secara acak. Populasi awal terdiri dari 100 individu.
2. **Evaluasi Individu:** Setiap individu dievaluasi berdasarkan kinerja prediksi ensemble menggunakan bobot yang dimilikinya. Kinerja diukur menggunakan kombinasi metrik RMSE dan MAE.
3. **Seleksi:** Individu-individu dengan kinerja terbaik (nilai RMSE dan MAE terendah) dipilih untuk menjadi induk dalam generasi berikutnya.
4. **Penyilangan (Crossover):** Individu-individu terpilih melakukan penyilangan untuk menciptakan individu baru dengan menggabungkan sebagian informasi dari kedua induk.
5. **Mutasi:** Perubahan acak dilakukan pada beberapa individu untuk meningkatkan keragaman dalam populasi.
6. **Iterasi:** Proses seleksi, persilangan, dan mutasi diulang selama 50 generasi.
7. **Pemilihan Solusi Terbaik:** Setelah 50 generasi, individu dengan kinerja terbaik dipilih sebagai solusi optimal. Bobot dari individu terbaik ini digunakan untuk melakukan prediksi ensemble pada data uji.

Dengan mengoptimalkan bobot menggunakan algoritma evolusioner, diharapkan dapat meningkatkan akurasi prediksi ensemble secara signifikan. Bobot optimal ini akan membantu menentukan kontribusi relatif dari setiap model dalam menghasilkan prediksi akhir yang lebih baik.

```

# Evolutionary Algorithm for Weight Optimization
creator.create("FitnessMax", base.Fitness, weights=(-1,))
creator.create("Individual", list, fitness=creator.FitnessMax)

def generate_individuals():
    return [random.uniform(0, 1) for _ in range(5)]

def evaluate(individual, x_test, y_test, model_lr, model_rf, model_gb, model_dt):
    weight_lr, weight_rf, weight_gb, weight_dt = individual
    predictions_ensemble = ensemble_predictions(x_test, model_lr, model_rf, model_gb, model_dt, weight_lr, weight_rf, weight_gb, weight_dt)
    rmse = np.sqrt(mean_squared_error(y_test, predictions_ensemble))
    mae = mean_absolute_error(y_test, predictions_ensemble)
    # Simulasi fitness antara RMSE dan MAE
    combined_metric = 0.5 * rmse + 0.5 * mae
    return combined_metric

toolbox = base.Toolbox()
toolbox.register("individual", tools.randomize, creator.Individual, lambda: generate_individuals())
toolbox.register("population", tools.randomize, list, toolbox.Individual)
toolbox.register("evaluate", evaluate, x_test=x_test, y_test=y_test, model_lr=model_lr, model_rf=model_rf, model_gb=model_gb, model_dt=model_dt)
toolbox.register("select", tools.selTournament, tournsize=1)
toolbox.register("mate", tools.cxTwoPoint)
toolbox.register("mutate", tools.mutGaussian, mu=0, sigma=0.1, indpb=0.2)
toolbox.register("map", map)

# Run the Evolutionary Algorithm
population = toolbox.population(n=100) # Population size of 100
ngen = 50 # Number of generations

algorithm.eaSimple(population, toolbox, cxpb=0.5, mutpb=0.2, ngen=ngen, stats=None, halloffame=None, verbose=False)

# Get best individual and weights
best_individual = tools.selBest(population, k=1)[0]
best_weights = best_individual

# Main ensemble prediction with optimized weights
predictions_ensemble_test = ensemble_predictions(x_test, model_lr, model_rf, model_gb, model_dt, best_weights[0], best_weights[1], best_weights[2], best_weights[3])
# Evaluate ensemble model with optimized weights
    
```

3.2.2. Model Performance

Untuk membangun titik perbandingan dan memahami kekuatan masing-masing model, terlebih dahulu dilakukan evaluasi kinerja model dasar secara individu, yaitu Regresi Linear (LR), Random Forest (RF), Decision Tree (DT), Gradient Boosting (GB) dan XGBoost pada data uji.

```

Ensemble Model Results (Evolutionary Algorithm):
- Weights: (Linear Regression, Random Forest, XGBoost, Gradient Boosting, Decision Tree): 0.62, -0.04, 0.06, 0.49, -0.03
RMSE:
- Linear Regression: 1.78
- Random Forest: 2.89
- XGBoost: 4.27
- Gradient Boosting: 1.84
- Decision Tree: 2.33
- Ensemble: 1.04
MAE:
- Linear Regression: 1.04
- Random Forest: 1.49
- XGBoost: 1.19
- Gradient Boosting: 0.94
- Decision Tree: 2.33
- Ensemble: 0.93
    
```

Tabel 3.2. Hasil Model Evolution Algorithm

Model	RMSE	MAE
Linear Regression	1.78	1.04
Random Forest	2.89	1.49
XGBoost	4.27	1.19
Gradient Boosting	1.84	0.94
Decision Tree	2.33	1
Ensemble	1.04	0.93

Tabel diatas menyajikan perbandingan kinerja beberapa model machine learning yang telah dilatih pada suatu dataset. Kinerja model-model ini dinilai berdasarkan dua metrik evaluasi yang umum digunakan dalam regresi, yaitu:

- a. RMSE (Root Mean Squared Error): Mengukur rata-rata perbedaan kuadrat antara nilai prediksi model dengan nilai aktual. Semakin kecil nilai RMSE, semakin baik model dalam memprediksi nilai.

- b. MAE (Mean Absolute Error): Mengukur rata-rata selisih absolut antara nilai prediksi model dengan nilai aktual. Sama seperti RMSE, nilai MAE yang lebih kecil menunjukkan kinerja model yang lebih baik.

3.2.3. Model Evaluation

Model dievaluasi menggunakan RMSE dan MAE untuk mengukur metrik evaluasi regresi standar. Mengambil individu dengan bobot terbaik yang ditemukan oleh EA. Dengan menggunakan bobot yang dioptimalkan tadi, akan dibuat prediksi ansambel pada tes set. Untuk menunjukkan manfaat menggabungkan prediksi, kinerja model ansambel akan dibandingkan dengan model tunggal.

Setelah proses optimasi bobot menggunakan algoritma evolusioner selesai, kita akan memiliki sejumlah individu (kombinasi bobot) yang telah dievaluasi. Untuk memilih individu terbaik, kita menggunakan metrik evaluasi regresi yang umum, yaitu Root Mean Squared Error (RMSE) dan Mean Absolute Error (MAE). Kedua metrik ini mengukur rata-rata kesalahan antara nilai prediksi dan nilai aktual. Semakin kecil nilai RMSE dan MAE, semakin baik kinerja model.

Individu dengan kombinasi bobot yang menghasilkan nilai RMSE dan MAE terendah dianggap sebagai individu terbaik. Kombinasi bobot inilah yang akan digunakan untuk membuat prediksi ensemble pada data uji. Dengan kata lain, kita akan memberikan bobot yang telah dioptimalkan ini kepada masing-masing model (regresi linear, random forest, decision tree dan XGBoost) untuk menghasilkan prediksi akhir.

Setelah mendapatkan prediksi ensemble menggunakan bobot terbaik, langkah selanjutnya adalah membandingkan kinerja model ensemble dengan kinerja masing-masing model tunggal (regresi linear, random forest, decision tree dan XGBoost). Perbandingan ini dilakukan untuk menunjukkan keunggulan model ensemble dalam hal akurasi prediksi.

```
import numpy as np
from sklearn.metrics import mean_squared_error, mean_absolute_error

# Assuming you have defined model_lr, model_rf, model_xgb, model_gbr, model_dt, and predictions_ensemble_test

# Calculate RMSE and MAE for each model
rmse_lr = mean_squared_error(y_test, model_lr.predict(X_test))
rmse_rf = mean_squared_error(y_test, model_rf.predict(X_test))
rmse_xgb = mean_squared_error(y_test, model_xgb.predict(X_test))
rmse_gbr = mean_squared_error(y_test, model_gbr.predict(X_test))
rmse_dt = mean_squared_error(y_test, model_dt.predict(X_test))
rmse_ensemble = np.sqrt(mean_squared_error(y_test, predictions_ensemble_test))

mae_lr = mean_absolute_error(y_test, model_lr.predict(X_test))
mae_rf = mean_absolute_error(y_test, model_rf.predict(X_test))
mae_xgb = mean_absolute_error(y_test, model_xgb.predict(X_test))
mae_gbr = mean_absolute_error(y_test, model_gbr.predict(X_test))
mae_dt = mean_absolute_error(y_test, model_dt.predict(X_test))
mae_ensemble = mean_absolute_error(y_test, predictions_ensemble_test)

# Print results
print("RMSE:")
print(f" - Linear Regression: {rmse_lr:.2f}")
print(f" - Random Forest: {rmse_rf:.2f}")
print(f" - XGBoost: {rmse_xgb:.2f}")
print(f" - Gradient Boosting: {rmse_gbr:.2f}")
print(f" - Decision Tree: {rmse_dt:.2f}")
print(f" - Ensemble: {rmse_ensemble:.2f}")

print("\nMAE:")
print(f" - Linear Regression: {mae_lr:.2f}")
print(f" - Random Forest: {mae_rf:.2f}")
print(f" - XGBoost: {mae_xgb:.2f}")
print(f" - Gradient Boosting: {mae_gbr:.2f}")
print(f" - Decision Tree: {mae_dt:.2f}")
print(f" - Ensemble: {mae_ensemble:.2f}")
```

Untuk melakukan perbandingan, digunakan nilai RMSE dan MAE untuk masing-masing model. Kemudian, nilai RMSE dan MAE dari model ensemble dibandingkan dengan nilai RMSE dan MAE

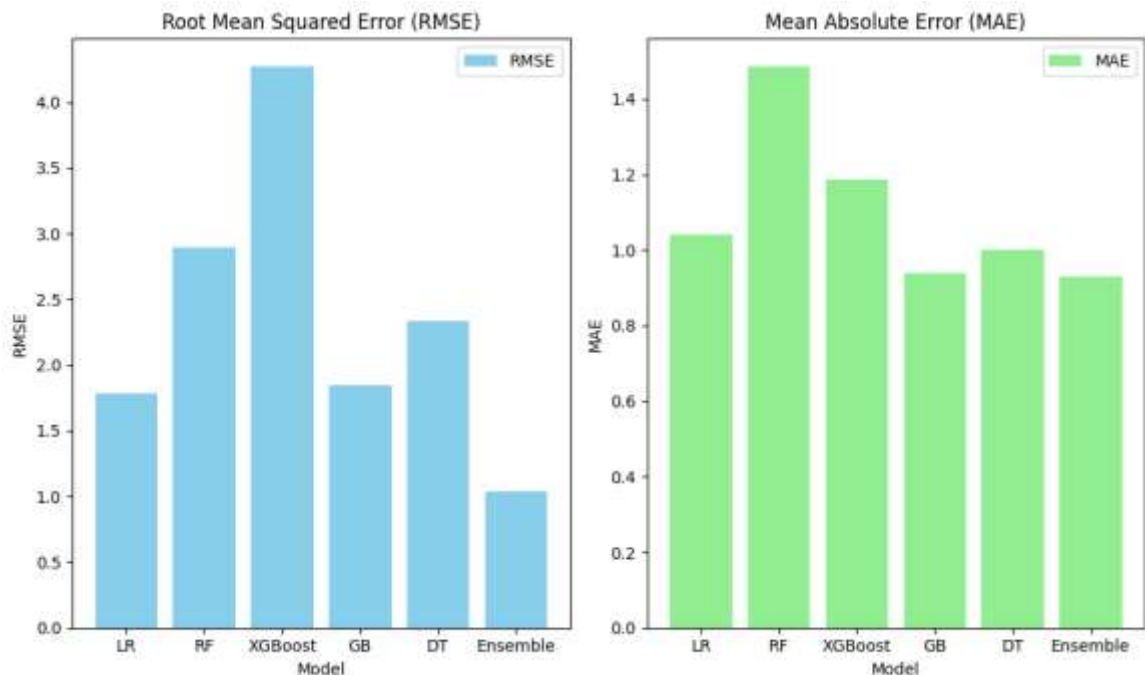
dari model tunggal. Jika nilai RMSE dan MAE dari model ensemble lebih rendah daripada nilai RMSE dan MAE dari model tunggal, maka dapat disimpulkan bahwa model ensemble memiliki kinerja yang lebih baik.

```

RMSE:
- Linear Regression: 1.78
- Random Forest: 2.89
- XGBoost: 4.27
- Gradient Boosting: 1.84
- Decision Tree: 2.33
- Ensemble: 1.04

MAE:
- Linear Regression: 1.04
- Random Forest: 1.49
- XGBoost: 1.19
- Gradient Boosting: 0.94
- Decision Tree: 1.00
- Ensemble: 0.93

```



Berdasarkan grafik diatas, dapat diamati bahwa:

- Model Ensemble: Merupakan model dengan kinerja terbaik berdasarkan kedua metrik evaluasi. Ini menunjukkan bahwa menggabungkan beberapa model (ensemble) dapat menghasilkan prediksi yang lebih akurat dibandingkan dengan menggunakan satu model tunggal.
- Gradient Boosting: Menyusul di posisi kedua dengan nilai RMSE dan MAE yang cukup baik. Model ini juga merupakan salah satu algoritma boosting yang populer dan sering digunakan dalam kompetisi machine learning.

- c. Linear Regression: Meskipun sederhana, model linear regression masih memberikan hasil yang cukup kompetitif, terutama dilihat dari nilai MAE.
- d. Random Forest, XGBoost, dan Decision Tree: Model-model ini memiliki kinerja yang kurang baik dibandingkan dengan model ensemble dan gradient boosting. Nilai RMSE dan MAE yang lebih tinggi menunjukkan bahwa model-model ini cenderung membuat prediksi yang kurang akurat.

4. Conclusion

Hasil penelitian ini menunjukkan keefektifan kerangka kerja ensemble untuk mendapatkan prediksi stok sisa tanaman hidroponik yang akurat dan dapat diinterpretasikan. Pendekatan ini tidak hanya memprediksi hasil panen dengan baik, tetapi juga memberikan wawasan tentang proses pengambilan keputusan, yang sangat penting bagi para pemangku kepentingan dalam pertanian hidroponik.

Penelitian ini juga menekankan bagaimana strategi ini dapat digunakan untuk meningkatkan pengelolaan sumber daya dalam sistem hidroponik dan mendorong praktik yang berkelanjutan. Penggunaan sumber daya yang lebih baik dan pengaruh yang lebih kecil terhadap lingkungan dapat dihasilkan dari keputusan yang diambil oleh para pemangku kepentingan melalui penggunaan algoritma yang kompleks dan alat interpretasi.

Meskipun menjanjikan, penelitian ini fokus pada kerangka kerja ensemble dan teknik optimasi tertentu. Penelitian selanjutnya dapat mengeksplorasi metode alternatif untuk meningkatkan kinerja prediksi. Selain itu, integrasi data sensor *realtime* berpotensi meningkatkan kemampuan beradaptasi dan akurasi model dalam prediksi.

References

- [1] Y. Liu, X. Pan, and J. Li, "Current Agricultural Practices Threaten Future Global Food Production," *J Agric Environ Ethics*, vol. 28, no. 2, pp. 203–216, Apr. 2015, doi: 10.1007/s10806-014-9527-6.
- [2] A. Wezel and E. Silva, "Agroecological Practices for Sustainable Agriculture Agroecology and Agroecological Cropping Practices," in *Agroecological Practices For Sustainable Agriculture: Principles, Applications, And Making The Transition*, no. 2, World Scientific Publishing Co., 2017, pp. 19–51. doi: 10.1142/9781786343062_0002.
- [3] T. Friedrich and A. Kassam, "Food security as a function of Sustainable Intensification of Crop Production," *AIMS Agriculture and Food*, vol. 1, no. 2, pp. 227–238, 2016, doi: 10.3934/agrfood.2016.2.227.
- [4] A. Rehman and M. Farooq, "Challenges, constraints, and opportunities in sustainable agriculture and environment," in *Sustainable Agriculture and the Environment*, Elsevier, 2023, pp. 487–501. doi: 10.1016/B978-0-323-90500-8.00012-9.
- [5] N. Arakkal Thaiparambil and V. Radhakrishnan, "Challenges in achieving an economically sustainable aquaponic system: a review," *Aquaculture International*, vol. 30, no. 6. Springer Science and Business Media Deutschland GmbH, pp. 3035–3066, Dec. 01, 2022. doi: 10.1007/s10499-022-00946-z.
- [6] S. R. Sathyanarayana, W. V. Gangadhar, M. G. Badrinath, R. M. Ravindra, and A. U. Shriramrao, "Hydroponics: An Intensified Agriculture Practice to Improve Food Production," *Reviews in Agricultural Science*, vol. 10. Gifu University - United Graduate School of Agricultural Science, pp. 101–114, 2022. doi: 10.7831/ras.10.0_101.
- [7] T. Alkadri, D. Rukmana, and N. Hamid, "Hydroponic vegetable agribusiness business development strategy (Case Study in CV. Akar Hydroponics Moncongloe Subdistrict, Maros District)," in *IOP Conference Series: Earth and Environmental Science*, Institute of Physics, 2023. doi: 10.1088/1755-1315/1230/1/012050.

- [8] K. J. Walters, B. K. Behe, C. J. Currey, and R. G. Lopez, "Historical, current, and future perspectives for controlled environment hydroponic food crop production in the United States," *HortScience*, vol. 55, no. 6, pp. 758–767, Jun. 2020, doi: 10.21273/HORTSCI14901-20.
- [9] T. van Klompenburg, A. Kassahun, and C. Catal, "Crop yield prediction using machine learning: A systematic literature review," *Computers and Electronics in Agriculture*, vol. 177. Elsevier B.V., Oct. 01, 2020. doi: 10.1016/j.compag.2020.105709.
- [10] M. Javaid, A. Haleem, I. H. Khan, and R. Suman, "Understanding the potential applications of Artificial Intelligence in Agriculture Sector," *Advanced Agrochem*, vol. 2, no. 1, pp. 15–30, Mar. 2023, doi: 10.1016/j.aac.2022.10.001.
- [11] S. Carpitella, M. Inuiguchi, V. Kratochvíl, and M. Pištěk, "Multi-criteria decision analysis without consistency in pairwise comparisons," *Comput Ind Eng*, vol. 168, Jun. 2022, doi: 10.1016/j.cie.2022.108089.
- [12] S. Barat, T. Clark, B. Barn, and V. Kulkarni, "A Simulation Based Aid for Complex Dynamic Decision Making," in *Proceedings of the Doctoral Consortium at the 9th IFIP WG 8.1 Working Conference on The Practice of Enterprise Modeling (PoEM-DC 2016) co-located with the 9th IFIP WG 8.1 Working Conference on The Practice of Enterprise Modeling (PoEM 2016)*, Soderstrom E. and Sandkuhl K., Eds., CEUR-WS, 2016, pp. 22–31. Accessed: May 23, 2024. [Online]. Available: <https://ceur-ws.org/Vol-1765/paper-05.pdf>
- [13] N. K. Choudhary, S. Sree Laya Chukkapalli, S. Mittal, M. Gupta, M. Abdelsalam, and A. Joshi, "YieldPredict: A Crop Yield Prediction Framework for Smart Farms," in *Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 2340–2349. doi: 10.1109/BigData50022.2020.9377832.
- [14] E. Sabeeh and M. Z. Al-Taie, "Enhancing Agricultural Decision-Making through Data Analysis: Predicting Crop Health Outcomes," in *BIO Web of Conferences*, EDP Sciences, Apr. 2024. doi: 10.1051/bioconf/20249700013.
- [15] V. , Mane, A. Gajbhiye, D. C. Amisha, and K. Gaikwad, "Krishi Mitra - Intelligent Crop And Fertilizer Recommender," in *2023 6th International Conference on Advances in Science and Technology (ICAST)*, IEEE, 2023. doi: 10.1007/978-981-19-2840-6_24.
- [16] J. Rajput et al., "Assessment of data intelligence algorithms in modeling daily reference evapotranspiration under input data limitation scenarios in semi-arid climatic condition," *Water Science and Technology*, vol. 87, no. 10, pp. 2504–2528, May 2023, doi: 10.2166/wst.2023.137.
- [17] M. Shahhosseini, G. Hu, I. Huber, and S. V. Archontoulis, "Coupling machine learning and crop modeling improves crop yield prediction in the US Corn Belt," *Sci Rep*, vol. 11, no. 1, Dec. 2021, doi: 10.1038/s41598-020-80820-1.
- [18] G. Pillonetto, T. Chen, A. Chiuso, G. De Nicolao, and L. Ljung, "Regularization of Linear Regression Models," in *Communications and Control Engineering*, Springer Science and Business Media Deutschland GmbH, 2022, pp. 33–93. doi: 10.1007/978-3-030-95860-2_3.
- [19] T. Zhu, "Analysis on the applicability of the random forest," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Aug. 2020. doi: 10.1088/1742-6596/1607/1/012123.
- [20] P. Nawrocki and M. Smendowski, "Long-Term Prediction of Cloud Resource Usage in High-Performance Computing," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2023, pp. 532–546. doi: 10.1007/978-3-031-36021-3_53.
- [21] A. Alsubayhin, M. S. Ramzan, and B. Alzahrani, "Crime Prediction Model using Three Classification Techniques: Random Forest, Logistic Regression, and LightGBM," *IJACSA International Journal of Advanced Computer Science and Applications*, vol. 15, no. 1, 2024, doi: 10.14569/IJACSA.2024.0150123.

- [22] D. A. Gzar, A. M. Mahmood, and M. K. Abbas, "A Comparative Study of Regression Machine Learning Algorithms: Tradeoff Between Accuracy and Computational Complexity," *Mathematical Modelling of Engineering Problems*, vol. 9, no. 5, pp. 1217–1224, Oct. 2022, doi: 10.18280/mmep.090508.
- [23] S. Ahmed, S. Madanian, F. Mirza, and S. Zain, "Prediction of Natural Gas Consumption in Bahçeşehir Using Machine Learning Models Machine Learning Models," in *ACIS 2020 Proceedings*, 2020. Accessed: May 23, 2024. [Online]. Available: <https://aisel.aisnet.org/acis2020/12/>
- [24] I. Szekely and M. H. Jijakli, "Bioponics as a Promising Approach to Sustainable Agriculture: A Review of the Main Methods for Producing Organic Nutrient Solution for Hydroponics," *Water (Switzerland)*, vol. 14, no. 23. MDPI, Dec. 01, 2022. doi: 10.3390/w14233975.