

## **Penerapan *Machine Learning* dalam Prediksi Bencana Banjir Berbasis Data Historis**

Zaima Firoos Likan (G641221002), Sulthan Farras Razin (G641221058), Muh Farid FB (G641221060), Aleeka Kiana S (G6401221089)<sup>1\*</sup>  
Kelompok: 2, Kelas Paralel: 1

### **Abstrak**

Peristiwa banjir di Indonesia tercatat menjadi bencana alam yang paling sering terjadi dengan 652 kejadian atau 37,95% dari total bencana yang terjadi di Indonesia pada tahun 2023. Dampak yang ditimbulkan oleh bencana alam ini sangat merugikan, mulai dari penyebab penghambat kegiatan perekonomian hingga menelan korban jiwa. Penelitian ini bertujuan untuk mengembangkan serta menerapkan model prediksi banjir menggunakan metode *Neural Network* (NN). Model ini diharapkan dapat menganalisis serta memprediksi potensi terjadinya banjir berdasarkan data historis. Berdasarkan prediksi model *Neural Network* menunjukkan bahwa nilai *Root Mean Squared Error* (RMSE) sebesar 0.0018 dan *Mean Absolute Error* (MAE) sebesar 0.0014 yang mengindikasikan tingkat error prediksi model sangat kecil. Selain itu, nilai *R-squared* sebesar 99,87% menunjukkan bahwa model mampu memprediksi potensi banjir berdasarkan data uji. Hal ini menunjukkan bahwa model prediksi *Neural Network* dapat memprediksi bencana banjir dengan sangat baik dalam upaya menanggulangi bencana banjir

Kata Kunci: Banjir, Data Historis, Model, *Neural Network*, Prediksi

### **Abstract**

*Flood events in Indonesia are recorded as the most frequently occurring natural disaster, with 652 incidents or 37.95% of the total disasters that occurred in Indonesia in 2023. The impact of this disaster is highly detrimental, ranging from disrupting economic activities to causing loss of life. This study aims to develop and implement a flood prediction model using the Neural Network method. The model is expected to analyze and predict the potential for flooding based on historical data. The prediction results using the Neural Network model show a Root Mean Squared Error (RMSE) of 0.0018 and a Mean Absolute Error (MAE) of 0.0014, indicating a very low prediction error. Furthermore, the R-squared value of 99.87% demonstrates that the model is capable of accurately predicting flood potential based on test data. These findings indicate that the Neural Network prediction model performs very well and can serve as an effective tool in flood disaster mitigation efforts.*

*Keywords: Flood, Historical Data, Model, Neural Network, Prediction*

---

<sup>1</sup>Program Studi Sarjana Ilmu Komputer, Sekolah Sains Data, Matematika dan Informatika (SMMI), Institut Pertanian Bogor, Bogor 16680

\*Mahasiswa Program Studi Sarjana Ilmu Komputer, SSMI IPB; Surel: [firooslikan@gmail.com](mailto:firooslikan@gmail.com), [username1@yahoo.co.id](mailto:username1@yahoo.co.id), [username1@yahoo.co.id](mailto:username1@yahoo.co.id)

## PENDAHULUAN

### Latar Belakang

Air memiliki peran penting bagi kehidupan makhluk hidup. Selain memberikan banyak manfaat, air juga dapat menjadi salah satu faktor yang memengaruhi terjadinya bencana alam, seperti banjir dan kekeringan. Sungai berfungsi sebagai tempat penampungan dan pengaliran air dari hulu ke hilir. Namun, apabila aliran air sungai datang secara berlebihan tanpa adanya daerah resapan dan penampungan yang memadai, kondisi ini dapat memicu terjadinya banjir (Sandiwarno 2024).

Bencana disebabkan oleh sekumpulan peristiwa alam yang mengancam dan mengganggu kehidupan masyarakat disebut bencana alam. Bencana yang disebabkan oleh alam diantaranya yaitu banjir. Banjir merupakan suatu keadaan Dimana suatu permukaan atau daerah tergenang oleh air dengan jumlah air yang sangat banyak. Banjir sendiri dipengaruhi oleh beberapa faktor, di antaranya kemampuan filtrasi tanah yang lemah, penumpukan sampah di sepanjang daerah aliran sungai yang menghambat aliran air, tingginya intensitas curah hujan, serta durasi hujan yang terlalu lama (Hasanah *et al.* 2021). Menurut Badan Nasional Penanggulangan Bencana (BNPB) tercatat sudah terjadi 1.718 peristiwa bencana alam yang sudah terjadi di Indonesia sejak periode 1 Januari-8 Juni 2023. Sampai saat ini banjir masih menjadi bencana alam yang paling banyak terjadi di Indonesia dengan 652 kejadian atau 37,95% dari total bencana alam nasional pada periode tersebut. Bencana ini juga cenderung meningkat setiap tahunnya (Ahdiat 2023).

Berdasarkan pentingnya peran air dalam kehidupan serta risiko yang ditimbulkan khususnya dalam bencana banjir yang frekuensinya terus meningkat setiap tahun di Indonesia, diperlukan upaya mitigasi yang lebih presisi dan berbasis data. Mengingat banjir merupakan bencana yang kompleks dan dipengaruhi oleh banyak faktor. *Neural Network* menjadi solusi potensial dalam mengidentifikasi pola dan memprediksi kejadian banjir secara lebih akurat. Oleh karena itu, diperlukan pengembangan model prediktif berbasis data historis guna mendukung upaya peringatan dini serta pengambilan keputusan dalam penanggulangan bencana banjir.

### Tujuan

Tujuan dari penyusunan tugas akhir ini adalah untuk mengembangkan dan menerapkan model prediksi banjir menggunakan metode *Neural Network* (NN). Model ini diharapkan mampu memprediksi potensi terjadinya banjir berdasarkan data historis. Selain itu, tugas akhir ini bertujuan untuk menganalisis akurasi dan performa model *Neural Network* dalam mendeteksi potensi banjir sehingga dapat digunakan sebagai sistem pendukung keputusan dalam upaya mitigasi bencana banjir di suatu wilayah.

### Ruang Lingkup

Ruang lingkup dalam tugas akhir ini mencakup kegiatan pengumpulan, pengolahan, dan analisis data terkait faktor-faktor yang memengaruhi terjadinya banjir. Data yang digunakan diperoleh dari platform Kaggle yang berisi informasi mengenai variabel yang berkaitan dengan kejadian banjir. Teknik yang digunakan dalam penelitian ini meliputi proses pra-pemrosesan data, normalisasi, pembagian data latih dan data uji, serta penerapan metode *Neural Network* untuk melakukan prediksi potensi banjir. Selain itu, dilakukan evaluasi performa model menggunakan metrik akurasi, *R-squared* ( $R^2$ ), *Accuracy*, *Mean Squared Error* (MSE), dan *Mean Absolute Error* (MAE) untuk menilai keoptimalan model dalam memprediksi banjir.

### Manfaat

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai alat bantu dalam memprediksi potensi banjir secara lebih cepat dan akurat menggunakan model

*Neural Network*. Selain itu, model yang dihasilkan dapat digunakan sebagai referensi bagi instansi terkait dalam mengambil langkah antisipasi dan mitigasi bencana banjir, serta menjadi dasar pengembangan sistem prediksi bencana berbasis kecerdasan buatan di masa mendatang.

## TINJAUAN PUSTAKA

### a. Pengantar Bencana Banjir

Bencana banjir berdampak terhadap berbagai aktivitas masyarakat, termasuk aspek kesehatan dan sosial ekonomi, khususnya bagi penduduk yang tinggal di dataran rendah, bantaran sungai, dan kawasan perkotaan (Azizulhaq *et al.* 2021). Terjadinya banjir dipengaruhi oleh beberapa faktor, seperti kondisi tanah, aktivitas manusia, dan curah hujan. Intensitas curah hujan yang tinggi, durasi hujan yang terlalu lama, kemampuan filtrasi tanah yang lemah, serta sistem drainase yang tidak lancar menjadi faktor utama yang meningkatkan potensi terjadinya bencana banjir (Hasanah *et al.* 2021).

### b. Metode Pemodelan

#### **Support Vector Machine (SVM)**

*Support Vector Machine* (SVM) merupakan sebuah metode pengklasifikasian yang menggunakan *hyperplane* untuk memisahkan kelas-kelas data. Metode SVM memaksimalkan *margin* atau jarak antar *hyperplane* dengan titik data terdekat dari masing-masing kelas guna menghasilkan batas keputusan yang optimal dan mampu melakukan generalisasi dengan baik pada data baru. SVM digunakan dalam pengklasifikasian maupun regresi dan mampu mengatasi data nonlinear menggunakan *kernel*, yaitu memetakan data ke dimensi lebih tinggi agar dapat dipisahkan secara linear. Keunggulan SVM terdapat pada proses komputasinya yang relatif cepat dikarenakan metode ini hanya bergantung pada *support vectors* serta kemampuannya dalam menangani data dengan kompleksitas beragam (Putri 2023).

#### **Random Forest Regressor**

*Random Forest* adalah algoritma *ensemble* yang terdiri dari banyak *decision tree* yang dilatih secara independen menggunakan teknik *bagging*, yaitu teknik pelatihan sampel data secara acak secara independent dengan pengembalian untuk membentuk dataset pelatihan bagi tiap pohon yang digunakan untuk meningkatkan nilai akurasi dan stabilitas model. Prediksi akhir dihasilkan dari agregasi semua *tree* yang ada pada *decision tree*. *Random Forest* efektif dalam meningkatkan nilai akurasi dibandingkan dengan metode pada *single classifier* dan mampu menangani data dengan kompleksitas yang tinggi (Givari *et al.* 2022).

#### **LightGBM**

LightGBM merupakan sebuah *framework Gradient Boosting* yang cepat dan efisien dengan menggunakan algoritma *decision tree*. LightGBM dirancang untuk meningkatkan kecepatan pelatihan dan efisiensi memori, serta mampu dalam menangani data berskala besar dan mendukung pembelajaran paralel maupun GPU (Rizky 2022). LightGBM memiliki performa yang sangat baik dalam menangani permasalahan ketidakseimbangan data. Hal ini menunjukkan bahwa metode LightGBM memiliki spesifikasi yang lebih baik dibandingkan metode lain seperti XGBoost pada kelas minoritas.

## XGBoost

XGboost adalah metode *Gradient Boosting Decision Tree* yang dikembangkan untuk efisiensi dan performa tinggi. XGBoost menggabungkan teknik *boosting* untuk menyusun model dari beberapa pohon keputusan secara berurutan, dengan optimasi yang membuatnya 10 kali lebih cepat dibandingkan metode *boosting* lain. XGBoost sering menunjukkan performa terbaik dalam klasifikasi dan regresi, serta mampu menangani data tidak seimbang dengan baik (Givari 2022).

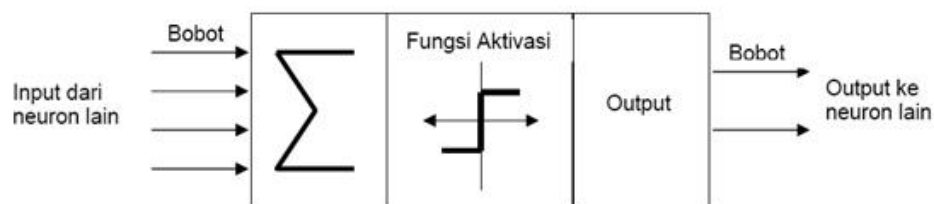
## TabNet

TabNet adalah model *deep learning* yang dirancang khusus untuk data tabular. TabNet menggunakan mekanisme *sequential attention* untuk memilih fitur penting dalam setiap langkah pengambilan keputusan, sehingga dapat mengekstraksi representasi fitur yang lebih relevan. Dalam beberapa penelitian, TabNet menunjukkan performa yang unggul dalam klasifikasi data tabular dibandingkan model neural network tradisional seperti FCNN, dengan nilai akurasi dan metrik evaluasi yang sedikit lebih baik (Ismanto *et al.* 2024).

## Neural Network (NN)

*Neural Network* (NN) merupakan hasil inspirasi dari mekanisme kerja otak manusia yang telah diteliti sejak tahun 1836. Struktur dasar dari sistem saraf terdiri atas neuron yang memiliki nukleus, dendrit, dan akson. Dendrit berfungsi menerima impuls dari neuron lain, sementara akson mengirimkan impuls ke neuron berikutnya melalui sinapsis, yaitu celah antar neuron. Setiap neuron dapat menerima sinyal dari banyak neuron lainnya, dan sinyal-sinyal ini dijumlahkan di dendrit sebelum dibandingkan dengan ambang batas untuk menentukan apakah neuron tersebut akan mengirim sinyal lanjutan (Cristina dan Kurniawan 2018).

Dalam konteks kecerdasan buatan, *Neural Network* (NN) memiliki hubungan antar simpul dalam jaringan yang memiliki bobot dan bias yang disesuaikan selama pelatihan agar kesalahan antara output yang dihasilkan dan output yang diharapkan seminimal mungkin. Proses ini menyerupai pencarian kondisi energi terendah untuk mengenali pola dari data yang diberikan. Sebuah model *Neural Network* umumnya terdiri dari lapisan input, beberapa *hidden layer*, dan satu lapisan output (Halim dan Wibisono 2000).



Gambar 1 Struktur *Neural Network*

Proses kerja *Neural Network* diawali dengan memasukkan data input ke neuron, disertai dengan nilai bobot masing-masing. Nilai-nilai ini kemudian dijumlahkan menggunakan fungsi penjumlahan yang dilambangkan dengan simbol sigma ( $\Sigma$ ). Hasil dari penjumlahan selanjutnya diproses oleh fungsi aktivasi dengan membandingkan nilai tersebut dengan ambang batas tertentu. Bila hasil penjumlahan berada di bawah ambang, maka neuron akan diaktifkan, namun jika melampaui ambang, aktivasi neuron akan dihentikan. Setelah neuron aktif, output kemudian diteruskan ke neuron lain melalui bobot, dan proses ini berulang untuk setiap input berikutnya.

### c. Studi Terkait

Penelitian terkait prediksi risiko banjir telah banyak dilakukan sebelumnya. Salah satunya dilakukan oleh peneliti dalam Septiadi *et al* (2024) yang menggunakan metode Analytic Hierarchy Process (AHP) dan Sistem Informasi Geografis (SIG) untuk memetakan daerah rawan banjir. Hasilnya menunjukkan tingkat akurasi sebesar 85% berdasarkan validasi lapangan. Sementara itu, penelitian oleh Wibowo dan Abadi (2022) memanfaatkan model Fuzzy Logic untuk memprediksi tingkat bahaya banjir berdasarkan parameter seperti curah hujan, elevasi tanah, dan penggunaan lahan. Model ini berhasil memperoleh nilai evaluasi Mean Absolute Percentage Error (MAPE) sebesar 8,5%, yang menunjukkan tingkat prediksi yang cukup baik.

Selain itu, penelitian oleh Adnan *et al* (2023), algoritma Random Forest (RF), K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP), serta kombinasi Genetic Algorithm–Radial Basis Function–Support Vector Regression (GA-RBF-SVR) digunakan untuk memprediksi banjir di wilayah pantai barat daya Bangladesh. Meskipun model-model tersebut mencapai akurasi prediksi yang tinggi, terdapat perbedaan spasial dalam hasil prediksi, dengan nilai koefisien korelasi piksel antar model berkisar antara 0,62 hingga 0,91. Model yang telah dioptimasi menunjukkan akurasi prediksi yang lebih baik dan meningkatkan kesesuaian spasial dengan mengurangi jumlah kesalahan klasifikasi.

Penelitian lain dilakukan oleh Fatonah (2021) memfokuskan deteksi bencana banjir berdasarkan data kejadian banjir pada bulan-bulan sebelumnya, sehingga pihak terkait dan masyarakat di sekitar wilayah terdampak dapat mengetahui besarnya dampak bencana pada tahun ini dan tahun-tahun mendatang. Metode yang digunakan dalam penelitian tersebut adalah algoritma naïve bayes yang diimplementasikan menggunakan perangkat lunak RapidMiner. Hasilnya menunjukkan tingkat akurasi sebesar 76,73% dalam mendeteksi bencana banjir berdasarkan data kejadian banjir beberapa tahun sebelumnya sebagai dasar mitigasi bencana banjir di masa mendatang.

### d. Penelitian dengan Data yang Sama

Salah satu penelitian relevan dilakukan oleh pengguna vandelta1 di platform Kaggle melalui proyek “Deteksi Banjir RF & ANN”. Studi ini menggunakan dataset historis yang sama dan membangun model *Artificial Neural Network* (ANN) dengan dua hidden layer berukuran 64 dan 32 neuron. Data diproses menggunakan *StandardScaler*, dan model dilatih selama 100 epoch dengan batch size 32 menggunakan optimizer Adam dan *loss function* MSE. Hasil evaluasi menunjukkan nilai *Mean Squared Error* (MSE) sebesar  $1.32 \times 10^{-7}$  dan *R-squared* ( $R^2$ ) sebesar 0.999946, yang mengindikasikan tingkat akurasi prediksi yang sangat tinggi. Penelitian ini membuktikan bahwa ANN berbasis data historis efektif dalam memodelkan risiko banjir dan dapat menjadi alat pendukung dalam mitigasi bencana.

## METODE

### Data

Tugas akhir ini menggunakan Flood Prediction Dataset, yang tersedia secara publik dan dapat diakses melalui situs internet Kaggle. Skor penggunaan data bernilai sebesar 9.41, yang menunjukkan kualitas dan kemudahan penggunaan dataset dalam kegiatan analisis data. Selain itu, dataset ini dilisensikan di bawah CC BY 4.0 sehingga dapat digunakan secara bebas dengan menyebutkan sumbernya.

Dataset terdiri dari 50.000 baris dan 21 fitur numerik, seluruh fitur memiliki tipe data int64, tanpa ada nilai yang hilang, dan tanpa ada variabel kategorik. Fitur-fitur tersebut

mencakup indikator lingkungan dan sosio-ekonomi yang diketahui berkontribusi terhadap banjir. Tabel 1 menunjukkan variabel input yang digunakan dalam tahap pemodelan.

Tabel 1 Penjelasan variabel input prediksi banjir

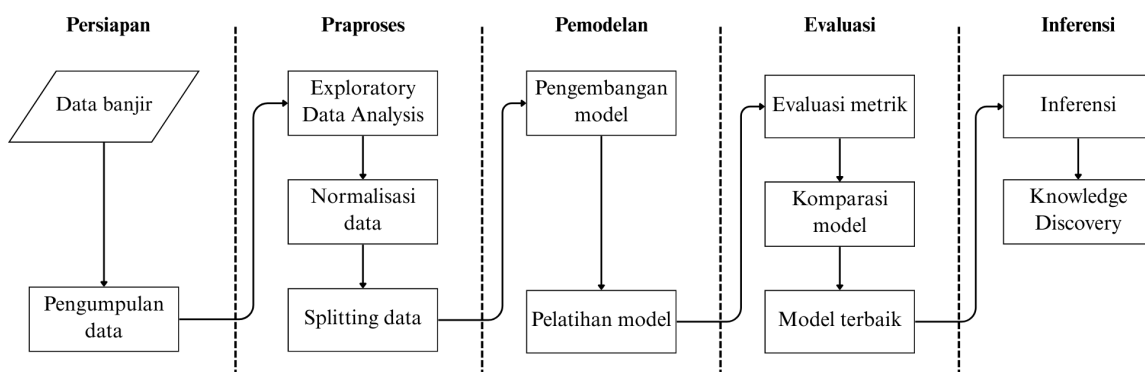
<b>Variabel</b>	<b>Keterangan</b>
MonsoonIntensity	Menggambarkan intensitas curah hujan saat musim hujan
TopographyDrainage	Kapasitas drainase berdasarkan topografi wilayah
RiverManagement	Kualitas dan efektivitas pengelolaan sungai
Deforestation	Tingkat deforestasi yang memengaruhi daya serap tanah dan limpasan air
Urbanization	Tingkat urbanisasi yang menciptakan permukaan kedap air
ClimateChange	Dampak perubahan iklim terhadap pola curah hujan ekstrem
DamsQuality	Kondisi fisik dan pemeliharaan bendungan
Siltation	Tingkat pendangkalan sungai dan waduk akibat sedimentasi
Agricultural Practices	Praktik pertanian yang mempengaruhi lingkungan, seperti penggundulan hutan dan irigasi
Encroachments	Tingkat pembangunan di wilayah resapan dan daerah aliran sungai
IneffectiveDisasterPreparedness	Kesiapsiagaan bencana yang tidak memadai, termasuk sistem peringatan dini
DrainageSystems	Efektivitas sistem drainase buatan dalam mengalirkan air hujan
CoastalVulnerability	Kerentanan wilayah pesisir terhadap banjir rob dan kenaikan permukaan laut
Landslides	Potensi tanah longsor akibat curah hujan tinggi di wilayah lereng
Watersheds	Jumlah dan karakteristik daerah aliran sungai yang mempengaruhi distribusi air
DeterioratingInfrastructure	Infrastruktur yang memburuk seperti gorong-gorong dan saluran air
PopulationScore	Tingkat kepadatan penduduk yang meningkatkan risiko kerugian akibat banjir
WetlandLoss	Kehilangan lahan basah yang berfungsi sebagai penyangga alami air berlebih
InadequatePlanning	Perencanaan tata ruang yang tidak memperhitungkan risiko

banjir

PoliticalFactors	Faktor kebijakan seperti kurangnya investasi atau komitmen politik dalam manajemen banjir
FloodProbability	Variabel target yang merepresentasikan skor probabilitas terjadinya banjir di suatu wilayah berdasarkan pengaruh dari seluruh fitur lainnya

### Tahapan Kegiatan

Pengerjaan tugas akhir melibatkan beberapa tahapan seperti pada Gambar 1. Tahapan pertama adalah tahapan pengumpulan data dari situs web Kaggle. Setelah itu, data akan masuk ke tahap praproses yang dilakukan untuk memahami karakteristik data dan mempersiapkannya untuk proses pemodelan. Tahap praproses pertama yang dilakukan adalah *Exploratory Data Analysis* (EDA), yakni proses eksplorasi data yang dilakukan untuk memahami distribusi fitur, hubungan antar variabel, serta mendeteksi outlier. Pada tahap ini pun dilakukan eksplorasi statistika deskriptif dari data melalui nilai mean, median, dan eksplorasi untuk memahami keperluan imputasi atau transformasi data. Setelahnya, dilakukan normalisasi data untuk menyamakan nilai untuk selanjutnya dibagi menjadi data latih dan data uji untuk validasi kinerja model.



Gambar 1 Diagram alir pengerjaan tugas akhir

Selanjutnya, pengerjaan masuk ke dalam tahap pemodelan, yakni pembangunan dan pelatihan model prediksi. Pada tugas akhir ini, digunakan beberapa algoritma regresi, yakni *Support Vector Machine* (SVM), *Random Forest Regressor*, *XGBoost*, *LightGBM*, *Neural Network*, dan *TabNET*. Penggunaan beberapa algoritma ini bertujuan untuk menjadi komparasi model terbaik nantinya. Setelah model dilatih, dilakukan evaluasi terhadap performa masing-masing model dengan metrik evaluasi *R-squared* ( $R^2$ ), *Accuracy*, *Mean Squared Error* (MSE), dan *Mean Absolute Error* (MAE). Evaluasi dilakukan secara objektif berdasarkan hasil prediksi data latih dan data uji. Hasil evaluasi dibentuk dalam komparasi model untuk memilih model terbaik yang digunakan. Tahap akhir dari proses adalah melakukan inferensi menggunakan data baru dan melakukan *knowledge discovery*.

### Lingkungan Pengembangan

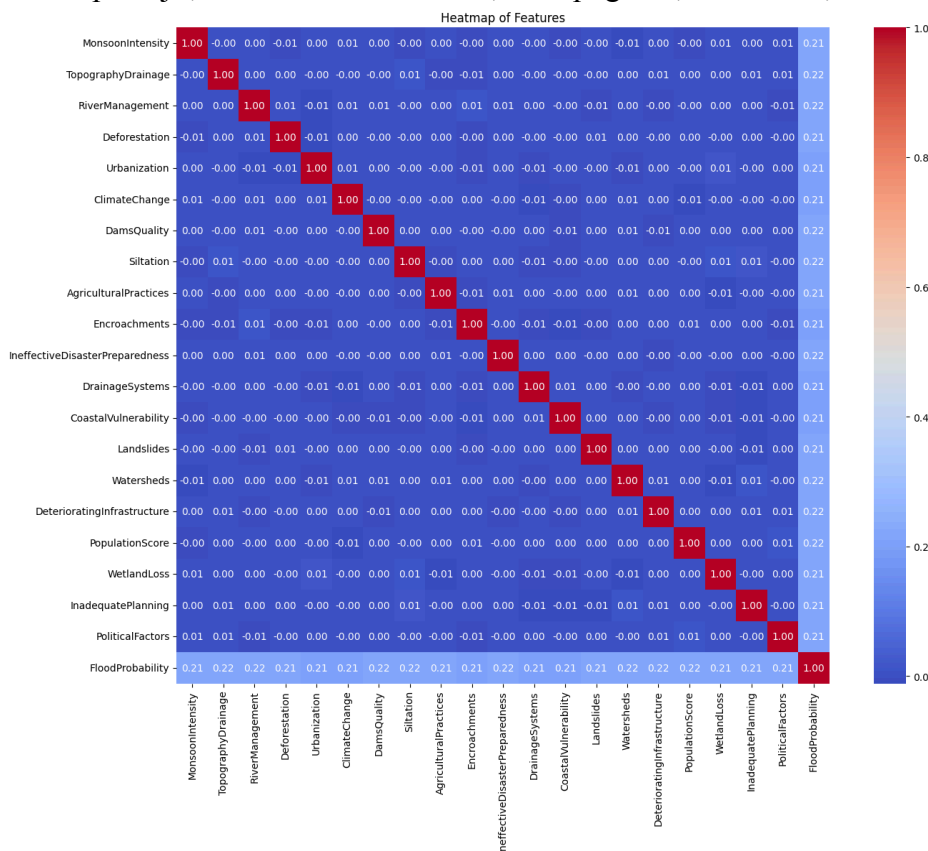
Pengerjaan tugas akhir dilakukan menggunakan *code editor* berbasis web, Google Colaboratory untuk mengolah, menganalisis, dan memvisualisasikan data. Sementara bahasa pemrograman yang dipakai dalam proses pengembangan adalah bahasa Python.

## HASIL DAN PEMBAHASAN

Pada tahap awal, dilakukan eksplorasi data secara deskriptif menggunakan beberapa fungsi yang disediakan oleh library *pandas*. Melalui proses ini, ditemukan bahwa dataset yang digunakan memiliki jumlah 50.000 baris data dan 21 kolom numerik, termasuk variabel *FloodProbability*. Setelahnya, dilakukan analisis untuk melihat parameter deskriptif dari data. Hampir seluruh fitur memiliki distribusi seimbang dan nilai median yang normal, namun ada beberapa fitur dengan variasi besar sehingga mengindikasikan potensi fitur untuk memiliki nilai pencilan. Pemeriksaan data selanjutnya menunjukkan bahwa tidak terdapat *missing values*, sehingga tidak diperlukan proses imputasi. Dataset juga tidak memiliki data duplikat, sehingga proses menjadi lebih sederhana. Terakhir, karena seluruh data sudah memiliki nilai numerik *int64*, data tidak perlu lagi dikonversi.

### Exploratory Data Analysis

Tahap selanjutnya dilakukan eksplorasi data secara visual menggunakan teknik *Exploratory Data Analysis* (EDA). Pengecekan korelasi dilakukan menggunakan diagram *heatmap* dengan metode *spearman*. Hasil dari pengecekan korelasi dapat dilihat pada Gambar 2. Korelasi setiap fitur dengan target *FloodProbability* memiliki nilai yang cukup rendah namun memiliki nilai yang kurang lebih seimbang, dengan rentang 0.21 hingga 0.22. Melihat nilai tersebut, ada dua kondisi yang dilakukan untuk dalam pemilihan fitur untuk pemodelan. Kondisi pertama adalah penggunaan seluruh fitur untuk memastikan semua informasi tertangkap oleh model. Sementara kondisi kedua adalah penggunaan 10 fitur dengan korelasi tertinggi menurut tinjauan pustaka. Pemilihan 10 fitur didasari oleh penelitian Hapsari pada 2016 yang menjelaskan berbagai faktor yang berkontribusi terhadap banjir, mulai dari faktor alam, antropogenik, tata kelola, dan faktor eksternal.



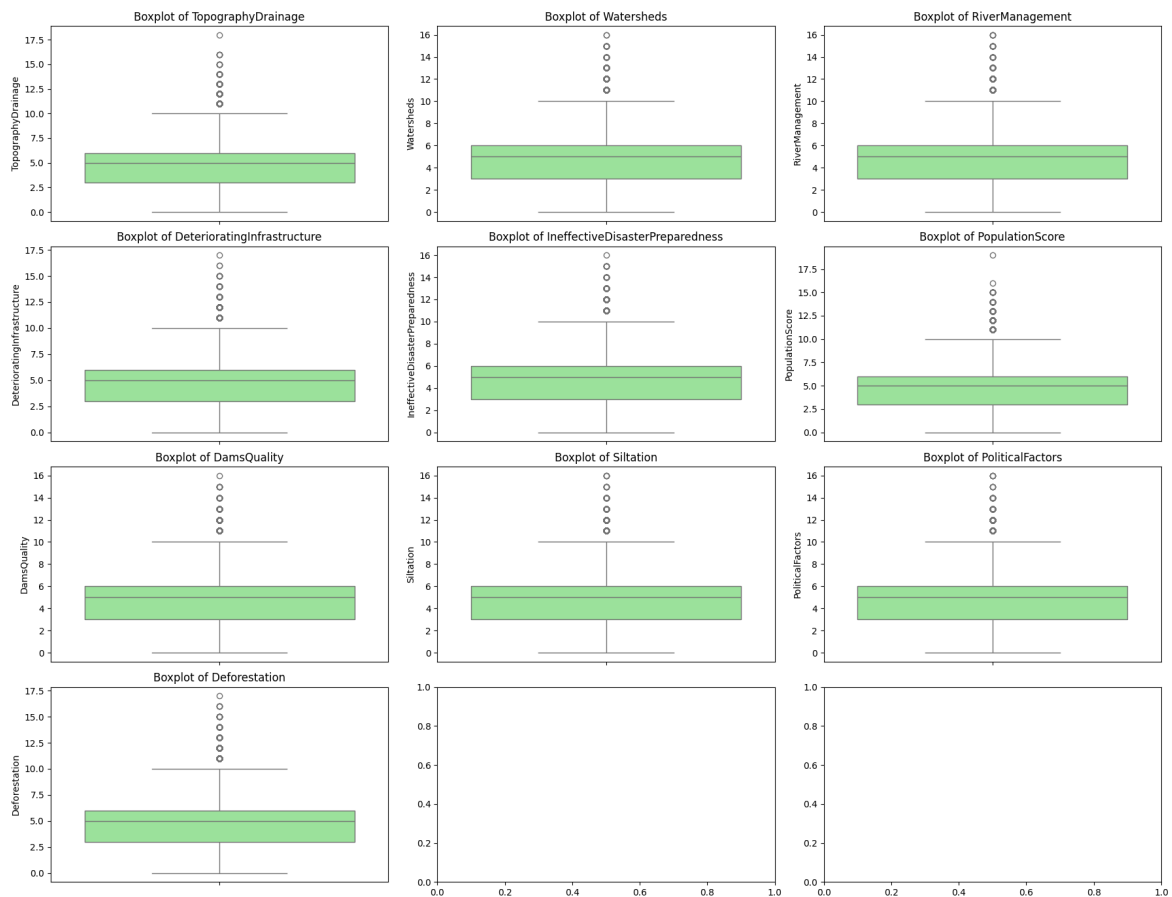
Gambar 2. Diagram *heatmap* korelasi fitur-fitur dalam dataset



Berdasarkan tinjauan pustaka melalui jurnal, penulis memilih 10 fitur yang merepresentasikan setiap faktor dengan baik sekaligus memperhatikan nilai dari fitur tersebut pada diagram *heatmap*. Sepuluh fitur yang dipilih adalah MoonsoonIntensity, RiverManagement, Urbanization, ClimateChange, Siltation, Enroachments, IneffectiveDisasterPreparedness, DrainageSystems, CoastalVulnerability, dan InadequatePlanning.

Tahap selanjutnya adalah pemeriksaan pencilan menggunakan diagram botplox. Gambar 3 menunjukkan diagram botplox untuk mendeteksi pencilan dari setiap fitur yang ada pada data. Melalui diagram, dapat dilihat bahwa distribusi median dari setiap distribusi memiliki nilai yang kurang lebih sama, yakni pada nilai 5. Selain itu, setiap fitur memiliki pencilan positif dengan nilai tinggi. Penulis memutuskan untuk mempertahankan nilai pencilan dengan mempertimbangkan data yang kemungkinan besar mencerminkan fenomena data sehingga data menjadi lebih realistis.

Boxplot for Outlier Analysis



Gambar 3. Diagram botplox seluruh fitur pada dataset

Setelah dilakukan *selected feature* berdasarkan tinjauan pustaka, langkah selanjutnya adalah melakukan proses normalisasi terhadap fitur-fitur yang terpilih guna menyamakan skala data agar setiap variabel memiliki kontribusi yang seimbang dalam proses pemodelan. Setelah normalisasi, data kemudian dibagi (*splitting*) menjadi data latih (*training set*) dan data uji (*testing set*) dengan proporsi data latih 80% dan data uji sebesar 20% sehingga model dapat dilatih dan diuji kinerjanya secara optimal menggunakan fitur-fitur yang telah dipilih tersebut. Dokumen kode dapat dilihat pada Tabel 1.

Tabel 1. Dokumen kode pemilihan fitur, normalisasi, dan *splitting* data

```

# used features
'''
selected_features = [
    'MonsoonIntensity',
    'RiverManagement',
    'Urbanization',
    'ClimateChange',
    'Siltation',
    'Encroachments',
    'IneffectiveDisasterPreparedness',
    'DrainageSystems',
    'CoastalVulnerability',
    'InadequatePlanning'
]
'''

# targets
X = df.drop(labels=['FloodProbability'], axis=1)
# X = df[selected_features]
y = df['FloodProbability'].values.reshape(-1, 1)

# normalizing
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# splitting
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y,
test_size=0.2, random_state=42)

```

## Pemodelan

Pada tahap pemodelan, dilakukan proses prediksi menggunakan beberapa metode regresi, yaitu *Support Vector Machine*, *Random Forest Regressor*, *LightGBM*, *XGBoost*, dan *Neural Network*. Setiap model akan dilatih menggunakan data latih yang telah disiapkan sebelumnya, kemudian dievaluasi kinerjanya menggunakan data uji berdasarkan sejumlah metrik evaluasi, seperti *R-squared* ( $R^2$ ), *Accuracy*, *Mean Squared Error* (MSE), dan *Mean Absolute Error* (MAE). Hasil dari evaluasi ini akan dibandingkan, dan model dengan performa evaluasi terbaik akan dipilih sebagai model akhir yang digunakan dalam penelitian ini. Adapun, perbandingan evaluasi model pada Tabel 2.

Tabel 2. Perbandingan evaluasi setiap model dengan data uji

Evaluasi Model	SVM	<i>Random Forest Regressor</i>	LightGBM	XGBoost	<i>Neural Network</i>	TabNet
RMSE	0.0264	0.2576	0.0013	0.0013	0.0023	0.0088
MAE	0.0201	0.0203	0.0105	0.0108	0.0018	0.0065
<i>R-squared</i>	0.7195	0.7334	0.9229	0.9252	0.9987	0.9690

Berdasarkan hasil perbandingan performa model yang ditampilkan pada Tabel 3. diketahui bahwa metode *Neural Network* menghasilkan nilai evaluasi terbaik dibandingkan metode lainnya. Metode ini menunjukkan nilai RMSE sebesar 0.0023, MAE sebesar 0.0018, dan *R-squared* sebesar 0.9987 yang menandakan tingkat akurasi prediksi yang sangat tinggi dan kesalahan prediksi yang paling rendah. Sementara itu, algoritma TabNet juga menunjukkan performa yang cukup baik dengan nilai RMSE sebesar 0.0088, MAE sebesar 0.0065, dan *R-squared* sebesar 0.9690. Meskipun demikian, performa TabNet masih berada di *bawah Neural Network* dari ketiga matrik evaluasi yang digunakan. Oleh karena itu, diputuskan untuk menggunakan model *Neural Network* sebagai model akhir dalam memprediksi variabel target karena memberikan hasil evaluasi yang paling optimal di antara semua algoritma yang dibandingkan. Setelah dilakukan komparasi performa antar model, diperoleh bahwa algoritma *Neural Network* menunjukkan hasil evaluasi paling optimal. Adapun proses yang dilakukan untuk mendapatkan performa optimal di antaranya:

### Inisiasi dan *Training Model*

Tabel 3. Dokumen kode inisiasi model dan *training model*

```
#Inisiasi model
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout,
BatchNormalization, Input
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import EarlyStopping

model = Sequential([
    Input(shape=(X_train.shape[1],)),
    Dense(128, activation='relu'),
    BatchNormalization(),
    Dropout(0.3),
    Dense(64, activation='relu'),
    Dense(32, activation='relu'),
    Dense(1) # regresi
])

model.compile(optimizer=Adam(learning_rate=0.0005), loss='mse',
```

```
metrics=['mae'])

early_stop = EarlyStopping(monitor='val_loss', patience=10,
restore_best_weights=True)

history = model.fit(
    X_train, y_train,
    epochs=100, batch_size=32,
    validation_data=(X_test, y_test),
    callbacks=[early_stop],
    verbose=1
)
```

Pada Tabel 3. dilakukan proses inisiasi model dengan membangun arsitektur menggunakan *Sequential model* dari Keras. Model terdiri atas beberapa lapisan, dimulai dengan *input layer* yang menyesuaikan jumlah fitur dari data latih. Selanjutnya, ditambahkan beberapa *hidden layer*, masing-masing menggunakan fungsi aktivasi ReLU, dengan susunan neuron sebanyak 128, 64, dan 32 unit.

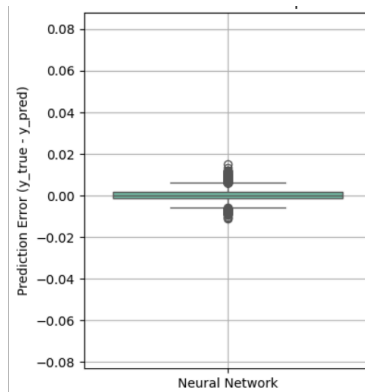
Untuk meningkatkan stabilitas *training* dan mencegah *overfitting*, digunakan Batch Normalization dan Dropout sebesar 0.3 setelah layer pertama. Proses kompilasi model dilakukan dengan menggunakan *optimizer* Adam dengan *learning rate* 0.0005, fungsi loss *mean squared error* (MSE), dan metrik evaluasi *mean absolute error* (MAE). *Training model* dilakukan selama maksimal 100 epoch dengan *batch size* 32, serta diterapkan *Early Stopping* dengan pemantauan pada nilai *validation loss*, dimana training akan dihentikan jika tidak terjadi perbaikan dalam 10 epoch berturut-turut, sekaligus mengembalikan bobot terbaik dari proses training.

### Evaluasi model

```
Neural Network Model:
MAE : 0.0019
RMSE : 0.0025
R2 : 0.9975
-----
```

Gambar 4. *Output* evaluasi model

Tahap selanjutnya adalah melakukan evaluasi terhadap performa model *Neural Network* menggunakan data uji. Prediksi dilakukan terhadap data uji yang telah dipisahkan sebelumnya, kemudian hasil prediksi dibandingkan dengan nilai aktual. Evaluasi model menggunakan tiga metrik utama, yaitu *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dan *R-squared* ( $R^2$ ). Berdasarkan hasil evaluasi pada Gambar 3, diperoleh nilai MAE sebesar 0.0019, RMSE sebesar 0.0025, dan *R-squared* sebesar 0.9975. Nilai *R-squared* yang sangat mendekati angka 1 menunjukkan bahwa model *Neural Network* mampu menjelaskan variasi data target dengan sangat baik, sementara nilai MAE dan RMSE yang rendah menandakan tingkat kesalahan prediksi model yang minimal.

Gambar 5. Boxplot *error distribution*

Pada Gambar 4. merupakan boxplot dari distribusi *error* hasil prediksi model. Terlihat bahwa nilai *error* mayoritas terkonsentrasi di sekitar nol dengan rentang interkuartil yang sangat sempit, menandakan bahwa sebagian besar prediksi memiliki selisih kecil terhadap nilai aktual. Walaupun terdapat beberapa outlier di area atas dan bawah, sebarannya masih dalam batas wajar untuk model regresi. Selanjutnya, untuk lebih meminimalkan error dan mengurangi jumlah outlier, proses *hyperparameter tuning* tambahan akan dilakukan agar performa model dapat dioptimalkan lebih lanjut.

### ***Hyperparameter Tuning***

```
Best Hyperparameters:
Units_1      : 128
Dropout      : 0.1
Units_2      : 32
Units_3      : 32
Learning Rate: 0.001
```

Gambar 6. *Best hyperparameter*

Pada tahap ini dilakukan *hyperparameter tuning* menggunakan Keras Tuner dengan metode *Random Search* untuk mencari konfigurasi terbaik model Neural Network. Setelah model terbaik dilatih ulang, diperoleh performa evaluasi dengan RMSE sebesar 0.0018 dan MAE sebesar 0.0014 yang menunjukkan peningkatan akurasi dibandingkan sebelum *tuning*. Pada tahap selanjutnya, dilakukan pengambilan hasil *hyperparameter* terbaik dari proses tuning yang telah dijalankan sebelumnya. *Hyperparameter* yang diperoleh meliputi jumlah *neuron* pada setiap *hidden layer*, nilai *dropout*, dan *learning rate*. Berdasarkan hasil *tuning*, konfigurasi terbaik terdiri dari 128 unit pada *layer* pertama, 32 unit pada *layer* kedua dan ketiga, *dropout* sebesar 0.1, serta *learning rate* sebesar 0.001.

### ***Evaluation Model Tuning***

Pada tahap ini dilakukan evaluasi kembali terhadap performa model *neural network* setelah melalui proses *hyperparameter tuning*. Hasil evaluasi menunjukkan bahwa nilai *Root Mean Squared Error* (RMSE) sebesar 0.0018 dan *Mean Absolute Error* (MAE) sebesar 0.0014, yang mengindikasikan tingkat error prediksi model sangat kecil. Selain itu, nilai  $R^2$  sebesar 0.9987 menunjukkan bahwa model mampu menjelaskan sekitar 99.87% variasi data target yang berarti model memiliki performa prediksi yang sangat baik. Dengan hasil ini, dapat disimpulkan bahwa tuning berhasil meningkatkan kualitas model dibandingkan sebelum tuning dilakukan.

### Perbandingan dengan Penelitian Sebelumnya

Salah satu studi yang relevan adalah proyek berjudul “Deteksi Banjir RF & ANN” yang dipublikasikan oleh pengguna vandelta1 di platform Kaggle. Penelitian tersebut menggunakan dataset yang sama dan menerapkan metode *Artificial Neural Network* (ANN) dengan dua hidden layer berukuran 64 dan 32 neuron. Proyek ini menghasilkan nilai *R-squared* ( $R^2$ ) sebesar 0.999946 dan *Mean Squared Error* (MSE) sebesar  $1.32e-07$ , menunjukkan tingkat akurasi prediksi yang sangat tinggi.

Meskipun arsitektur jaringan dalam penelitian ini lebih kompleks, yaitu menggunakan tiga hidden layer (128, 64, dan 32 neuron) serta ditambah dengan *Dropout* dan *Batch Normalization*, nilai evaluasi model yang diperoleh sedikit lebih rendah ( $R^2 = 0.9987$ ). Hal ini mengindikasikan bahwa jumlah hidden layer yang lebih banyak tidak selalu menghasilkan model yang lebih akurat. Sebaliknya, efisiensi model juga dipengaruhi oleh kombinasi arsitektur, skema pelatihan, serta kualitas data. Oleh karena itu, dalam merancang model ANN, penting untuk mempertimbangkan keseimbangan antara kompleksitas arsitektur dan performa aktual, bukan semata-mata meningkatkan kedalaman jaringan.

### KESIMPULAN

Topografi wilayah yang cukup rendah atau berada pada dataran yang rendah serta pengelolaan sistem drainase yang buruk menjadi faktor signifikan yang mempengaruhi terjadinya bencana banjir. Berdasarkan penelitian yang dilakukan, Metode *Neural Network* berhasil digunakan sebagai alat prediksi bencana banjir menggunakan data historis sebagai acuan. Hasil dari penerapan model ini mampu memberikan prediksi banjir dengan tingkat akurasi 99,87% yang menunjukkan bahwa model mampu memprediksi potensi bencana banjir dengan akurat. Metode *Neural Network* yang digunakan menunjukkan performa yang sangat baik dengan nilai RMSE sebesar 0,0018 dan MAE sebesar 0,0014. Hal ini sudah sesuai dengan tujuan awal penelitian yaitu membangun model prediksi bencana banjir menggunakan data historis. Penelitian ini menunjukkan bahwa penerapan metode *Neural Network* memiliki potensi dalam mendukung upaya mitigasi bencana berbasis data. Model ini memiliki potensi untuk diterapkan secara luas pada wilayah yang rentan terkena banjir sebagai alat bantu dalam pengambilan keputusan. Penelitian selanjutnya disarankan untuk mengintegrasikan data spasial dan data *real-time* guna meningkatkan kualitas prediksi. Selain itu, diperlukan upaya peningkatan kelengkapan dan kualitas data historis agar akurasi model dapat terus terjaga secara optimal.

## DAFTAR PUSTAKA

- Adnan, M. S. G., et al. (2023). A novel framework for addressing uncertainties in machine learning-based geospatial approaches for flood prediction. *Journal of Environmental Management*, 326, 116813. doi:10.1016/j.jenvman.2022.116813
- Amaliyah Wibowo, N., & Maman Abadi, A. (2022). Analisis tingkat kerawanan bencana alam banjir di Kabupaten Purbalingga dengan fuzzy logic.
- Azizulhaq, M. A., Suhendi, A., & Setianingsih, C. (2021). Dashboard sistem peringatan dini prediksi banjir menggunakan metode radial basis function berbasis web. *e-Proceeding of Engineering*, 8(1), 334.
- Cristina, C., & Kurniawan, A. (2018). Sejarah, Penerapan, dan Analisis Resiko dari Neural Network: Sebuah Tinjauan Pustaka. *Jurnal Informatika: Jurnal Pengembangan IT*, 3(2), 259-270. doi:10.30591/jpit.v3i2.890.
- Farisi, I., Shadiq, J., Priyadi, W., Maulana, D., Acep, A., & Gusril, S. F. (2024). Penerapan model Recurrent Neural Network (RNN) untuk prediksi curah hujan berbasis data historis. *Information System for Educators and Professionals: Journal of Information System*, 9(2), 217–226.
- Givari, M. R., Sulaeman, M. R., & Umaidah, Y. (2022). Perbandingan Algoritma SVM, Random Forest Dan XGBoost Untuk Penentuan Persetujuan Pengajuan Kredit. *Nuansa Informatika*, 16(1), 141-149. doi: [10.25134/nuansa.v16i1.5406](https://doi.org/10.25134/nuansa.v16i1.5406).
- Hasanah, M. A., Soim, S., & Handayani, A. S. (2021). Implementasi CRISP-DM model menggunakan metode decision tree dengan algoritma CART untuk prediksi curah hujan berpotensi banjir. *Journal of Applied Informatics and Computing*, 5(2). doi:10.30871/jaic.v5i2.3200.
- Halim, S., & Wibisono, A. M. (2000). Penerapan jaringan saraf tiruan untuk peramalan. *Jurnal Teknik Industri*.
- Ismanto, E., Fadlil, A., & Yudhana, A. (2024). Analisis Perbandingan Model Fully Connected Neural Networks (FCNN) dan TabNet Untuk Klasifikasi Perawatan Pasien Pada Data Tabular. *Jurnal CoSciTech (Computer Science and Information Technology)*, 5(3), 526-532.
- Putri, S. S. M., Arhami, M., & Hendrawaty, H. (2023). Penerapan Metode SVM pada Klasifikasi Kualitas Air. *Journal of Artificial Intelligence and Software Engineering*, 3(2), 93-101.
- Rizky, P. S., Hirzi, R. H., & Hidayaturohman, U. (2022). Perbandingan Metode LightGBM dan XGBoost dalam Menangani Data dengan Kelas Tidak Seimbang. *J Statistika: Jurnal Ilmiah Teori dan Aplikasi Statistika*, 15(2), 228-236. doi: [10.36456/jstat.vol15.no2.a5548](https://doi.org/10.36456/jstat.vol15.no2.a5548).
- Salsabila, R. A. (n.d.). Perbandingan Fuzzy Inference System metode Mamdani dan Tsukamoto untuk memprediksi banjir di Provinsi Sumatera Selatan.
- Sandiwarno, S. (2024). Penerapan machine learning untuk prediksi bencana banjir. *Jurnal Sistem Informasi Bisnis*, 14(1), 62–76.
- Septiadi, M. R., Nugroho, B. I., Santoso, N. A., & Gunawan, G. (2024). Penerapan metode Simple Moving Average dan Analytic Hierarchy Process untuk prediksi tingkat kerawanan banjir rob di Brebes. *Jurnal Minfo Polgan*, 13(1), 278–285. doi:10.33395/jmp.v13i1.13618.