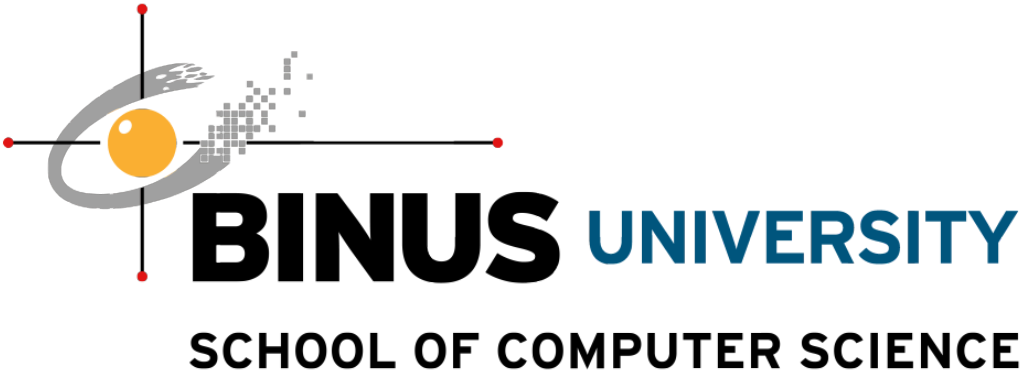
Analisis Model Prediksi Iklim dan Banjir di Daerah Jakarta Menggunakan Random Forest dan Logistic Regression



Big Data Processing - COMP6579001

Oleh:

Hendy Cahyadi Tandiono - 2702290135

Stefanus Marcellino - 2702215284

Benediktus Arthur Suparto - 27022555911

Jordi Austin Iskandar - 2702324631

Geoffrey Duncan Julianto - 2702255773

Joseph Budihartanto - 2702261712

BAB 1
LATAR BELAKANG

Jakarta merupakan salah satu kota metropolitan yang ada di Asia Tenggara. Sering kali, masalah-masalah seperti perubahan iklim dan juga banjir dapat menjadi ancaman yang serius bagi Jakarta. Kurang lebih setiap tahunnya, wilayah Jakarta mengalami banjir yang menyebabkan kerugian dari segi material dan ekonomi, terganggunya kegiatan sosial, dan seringkali mengganggu keselamatan masyarakat. Kondisi ini juga semakin diperparah dengan sistem penyerapan air yang masih kurang optimal di Jakarta.

Menurut data dari Badan Penanggulangan Bencana Daerah (BPBD) DKI Jakarta, pada awal tahun 2021, ada lebih dari 200 titik banjir di Jakarta dengan ketinggian air yang mencapai 1-2 meter. Sementara itu, menurut data BPBD pada tahun 2024, banjir di daerah Jakarta berdampak pada setidaknya 61 RT akibat hujan deras yang melanda. Hal ini juga diperparah dengan penurunan muka tanah yang menjadi penyebab banjir yang ada di Jakarta. Menurut data dari Dinas Sumber Daya Air DKI Jakarta, beberapa titik di wilayah Jakarta mengalami penurunan muka tanah yang bervariasi hingga 10 cm, dengan rata-rata 3,9 cm pada tahun 2023. Penurunan muka tanah ini sebagian besar disebabkan karena pengambilan air tanah yang berlebihan.

Di sisi lain, perubahan iklim juga memperparah kondisi di wilayah Jakarta. Perubahan iklim yang tidak teratur dapat menyebabkan cuaca yang ekstrem dan peningkatan intensitas hujan. Berdasarkan data dari BMKG, tren curah hujan yang ekstrim selalu meningkat dalam 2 tahun terakhir. Cuaca yang semakin tidak menentu ini menjadi indikator bahwa Jakarta semakin rentan terhadap bencana iklim.

Dalam menangani masalah ini, analisis terhadap perubahan iklim dan juga banjir memiliki peran yang sangat penting dalam upaya pencegahan. Data-data seperti curah hujan, kelembapan, suhu, dan kecepatan angin dapat digunakan untuk membuat sebuah model prediktif ataupun analisis yang dapat membantu pemerintah maupun masyarakat dalam mencegah dan mengambil keputusan secara cepat dan tepat.

BAB 2
METODOLOGI DAN ALUR PEKERJAAN

Penelitian ini akan menggunakan pemodelan prediktif berbasis machine learning untuk mengidentifikasi faktor-faktor yang mempengaruhi kejadian banjir di Jakarta selama tahun 2016-2020. Alur pekerjaan kami akan dipecah menjadi 5 tahap utama: (1) Data Collection, (2) Data Preparation, (3) Tipe Analisis, (4) Analisis Model, dan (5) Visualisasi Hasil.

Eksperimen dilakukan menggunakan Google Collab sebagai environment pemrograman berbasis cloud. Collab memungkinkan kolaborasi serta mendukung penggunaan hardware accelerator seperti GPU.

Dalam konteks Alur pekerjaan proyek kami ini, alur akan dimulai dari (1) Data Collection yang dilakukan dengan menggabungkan beberapa sumber data terbuka terkait banjir. Kemudian Tahap (2) Data Preparation kami mencakup pembersihan data, penanganan missing value dan outlier, serta encoding dan normalisasi data. Pada tahap (3) Tipe Analisis, pendekatan yang kami gunakan adalah analisis prediktif yang digunakan untuk mengklasifikasikan kemungkinan terjadinya banjir. Tahap (4) Analisis model dilakukan dengan menerapkan beberapa algoritma klasifikasi seperti Logistic Regression dan Random Forest, serta menggunakan teknik balancing seperti SMOTE. Terakhir, tahap (5)Visualisasi Hasil dilakukan untuk menampilkan tren kejadian banjir serta performa dari model prediktif yang dibangun.

1. Data Collection

Data yang akan digunakan dalam research kami diperoleh dari Kaggle dengan judul dataset “Climate and Flood Jakarta 2016-2020”. Dataset ini mencakup fitur-fitur meteorologi harian, seperti:

Feature	Description
Tn	min temperature (°C)
Tx	max temperature (°C)
Tavg	avg temperature (°C)
RH_avg	avg humidity(%)
RR	rainfall (mm)
ss	duration of sunshine(hour)
ff_x	max wind speed (m/s)
ddd_x	wind direction at maximum speed (°)
ff_avg	max wind speed (m/s)
ddd_car	most wind direction (°)
station_id	station id which record the data
station_name	station name which record the data
region_name	location of the station
flood	1 means True and 0 means false

2. Data Preparation

Data yang kami telah ambil akan kemudian diproses agar data tersebut siap dipakai untuk dianalisis di tahap berikutnya. Tahap ini melibatkan beberapa langkah penting:

- Loading Data: Memasukkan data ke Google Collab untuk dianalisis.

```
def load_and_examine_data(file_path):
    df = pd.read_csv(file_path)
    print("Dataset shape:", df.shape)
    print("\nFirst 5 rows:")
    print(df.head())
    print("\nData types:")
    print(df.dtypes)
    print("\nMissing values per column:")
    print(df.isnull().sum()[df.isnull().sum() > 0])
    print("\nFlood event distribution:")
    print(df['flood'].value_counts(normalize=True) * 100)
    returndf
```

- Pembersihan Data: Menghapus data yang tidak relevan dan tidak akan digunakan dalam analisis karena tidak mempengaruhi hasil, seperti station_name, station_id.

```
def remove_missing_rows(df):
    print("\nRemoving rows with missing values...")
    before = df.shape[0]
    df = df.dropna()
    after = df.shape[0]
    print(f"Removed {before - after} rows. New shape: {df.shape}")
    return df
```

- Penanganan Outlier: Menggunakan metode IQR untuk menghapus outlier dari fitur numerik.

```
def detect_and_handle_outliers(df, columns_to_check=None):
    if columns_to_check is None:
        columns_to_check = [col for col in ['Tn', 'Tx', 'Tavg', 'RH_avg', 'RR',
'ss', 'ff_x', 'ff_avg']
                            if col in df.columns and
pd.api.types.is_numeric_dtype(df[col])]
    print("\nOutlier detection and handling:")
    for col in columns_to_check:
        Q1 = df[col].quantile(0.25)
        Q3 = df[col].quantile(0.75)
        IQR = Q3 - Q1
        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR
        outliers = df[(df[col] < lower_bound) | (df[col] > upper_bound)]
        print(f"{col}: {len(outliers)} outliers capped")
        df[col] = np.where(df[col] < lower_bound, lower_bound, df[col])
        df[col] = np.where(df[col] > upper_bound, upper_bound, df[col])
    return df
```

- Penyeimbangan Data: Menggunakan SMOTE (Synthetic Minority Oversampling Technique) untuk menangani kelas pada data yang dilatih.

3. Analysis Type

Tipe Analisis yang akan kami gunakan adalah predictive analysis, dimana tipe analisis ini akan menggunakan supervised learning, metode supervised learning ini akan menggunakan data berlabel, dimana fitur-fitur input (seperti curah hujan, ketinggian wilayah jumlah saluran air, dan sebagainya) akan digunakan untuk memprediksi target/output (kejadian banjir atau tidak banjir).

Pendekatan ini memungkinkan untuk sistem belajar dari data bersifat historis dan mengenali pola yang berkaitan dengan potensi banjir. Kemudian hasil yang didapat dari tahap ini nantinya akan digunakan dalam tahap Analisis Model.

Terdapat dua algoritma utama yang akan kami gunakan dalam research ini. Dua algoritma tersebut adalah:

- Logistic Regression (LR): Model linear sebagai baseline yang baik untuk interpretabilitas.
- Random Forest Classifier (RFC): Model ensemble berbasis Decision Tree yang dapat menangani interaksi antar fitur secara kompleks.

4. Model Analysis

Setiap model akan dievaluasi berdasarkan beberapa kriteria berikut:

- Classification report: Precision, Recall, F1-score, dan Accuracy.
- Confusion Matrix: Untuk melihat distribusi prediksi benar dan salah.
- ROC AUC Score: Mengukur kemampuan model dalam membedakan antara dua kelas.
- ROC Curve: Visualisasi trade-off antara True Positive Rate dan False Positive Rate.

5. Visualization

Visualisasi dilakukan untuk membantu dalam tahap analisis agar mendukung dan membantu pemahaman hasil, seperti Grafik ROC Curve untuk membandingkan performa model, Flood Count by Region untuk membandingkan jumlah kejadian banjir yang terjadi di setiap daerah Jakarta, dan Monthly serta Yearly Flood Count untuk membandingkan jumlah kejadian banjir yang terjadi di tiap bulan dan tahun.

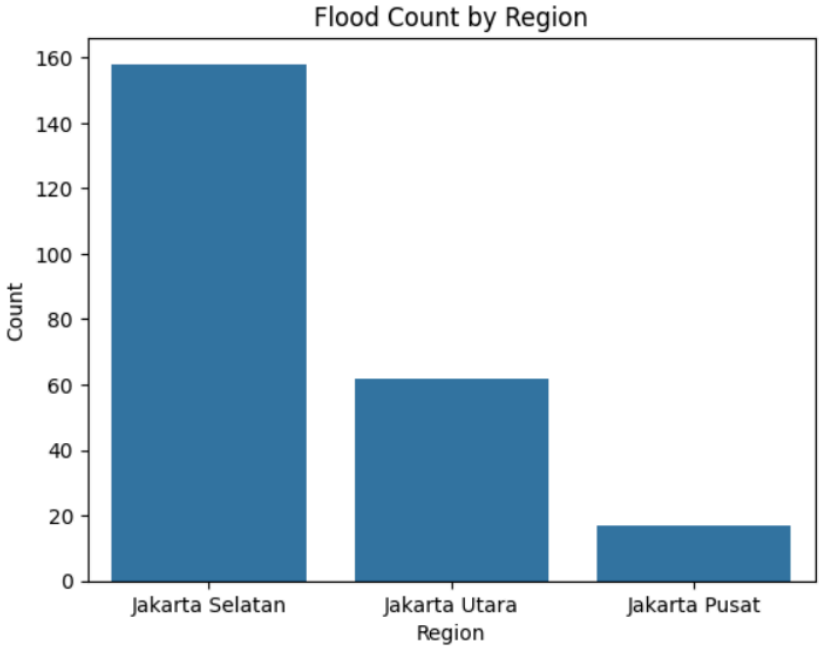
BAB 3
EVALUASI DAN DETAIL DARI ALUR PEKERJAAN

Berdasarkan hasil dari dua model yang kami gunakan, yaitu Random Forest dan Logistic Regression, setiap model memiliki kelebihan dan kekurangannya masing-masing. Dari segi akurasi, Random Forest memiliki nilai akurasi yang lebih tinggi yaitu sebesar 92%, sementara model Logistic Regression hanya menghasilkan akurasi sebesar 77%. Akurasi ini sendiri menjelaskan seberapa akurat sebuah model dapat memprediksi atau mengklasifikasikan suatu hal secara benar. Selanjutnya, Random Forest juga menghasilkan nilai precision yang lebih tinggi dalam memprediksi banjir yaitu sekitar 42%, sementara Logistic Regression hanya menghasilkan precision terhadap banjir sebesar 22%. Precision disini merupakan salah satu matrik evaluasi yang mengukur seberapa banyak dari prediksi positif yang memang dipastikan benar. Untuk nilai recall, Random Forest menghasilkan nilai sebesar 30% sementara Logistic Regression menghasilkan nilai sebesar 81%. Recall juga merupakan sebuah matriks evaluasi yang mengukur seberapa banyak kasus positif yang berhasil ditemukan oleh model. Kemudian untuk metric yang terakhir yaitu nilai ROC AUC, Random Forest menghasilkan nilai sebesar 82% sementara Logistic Regression menghasilkan nilai sebesar 87%. Untuk perbandingan secara lebih singkat dapat dilihat melalui tabel di bawah ini

Metric	Model	
	Random Forest	Logistic Regression
Accuracy	92%	77%
Precision (Banjir)	42%	22%
Recall (Banjir)	30%	81%
ROC AUC	82%	87%

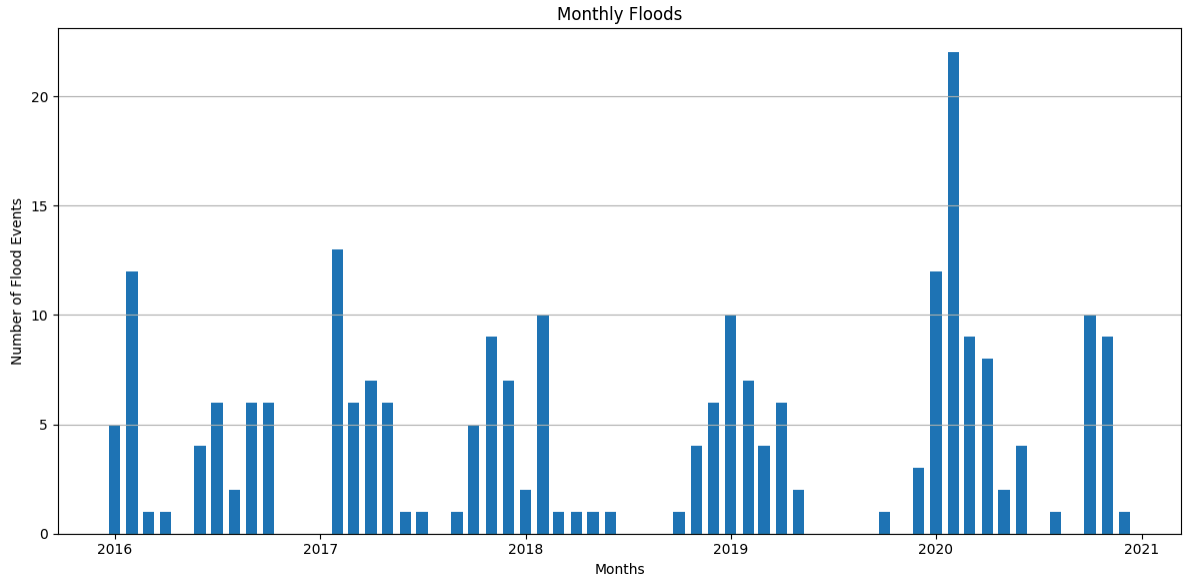
Random Forest dan Logistic Regression memiliki perbedaan dari segi cara algoritma bekerja, kompleksitasnya, dan juga kemampuan masing-masing model dalam menangkap pola pada data. Dalam segi algoritma, Random Forest menggunakan model ensemble atau gabungan dari beberapa decision tree. Cara kerja dari Random Forest ini adalah dengan membangun banyak decision tree dan mengambil mayoritas dari voting. Sementara itu, Logistic Regression menggunakan model linear dengan cara kerja mengasumsikan hubungan linear antara fitur dan probabilitas. Hal ini membuat Random Forest lebih kuat terhadap outlier dan memiliki performa yang lebih tinggi, tetapi dapat menyebabkan overfit jika tidak ditangani. Sementara itu, Logistic Regression cenderung lebih cepat dan sederhana, tetapi dapat terjadi underfit jika pola dalam data terlalu kompleks.

Selain itu kemampuan sebuah model dalam menangkap pola juga mempengaruhi hasil analisisnya. *Random Forest* mampu menangkap pola non-linear dan interaksi antar fitur yang membuatnya lebih fleksibel terhadap bentuk data yang kompleks. Sementara *Logistic Regression* hanya efektif jika hubungan antar variabel dan target bersifat linear yang membuatnya tidak cocok jika data kompleks atau mengandung banyak interaksi non-linear.

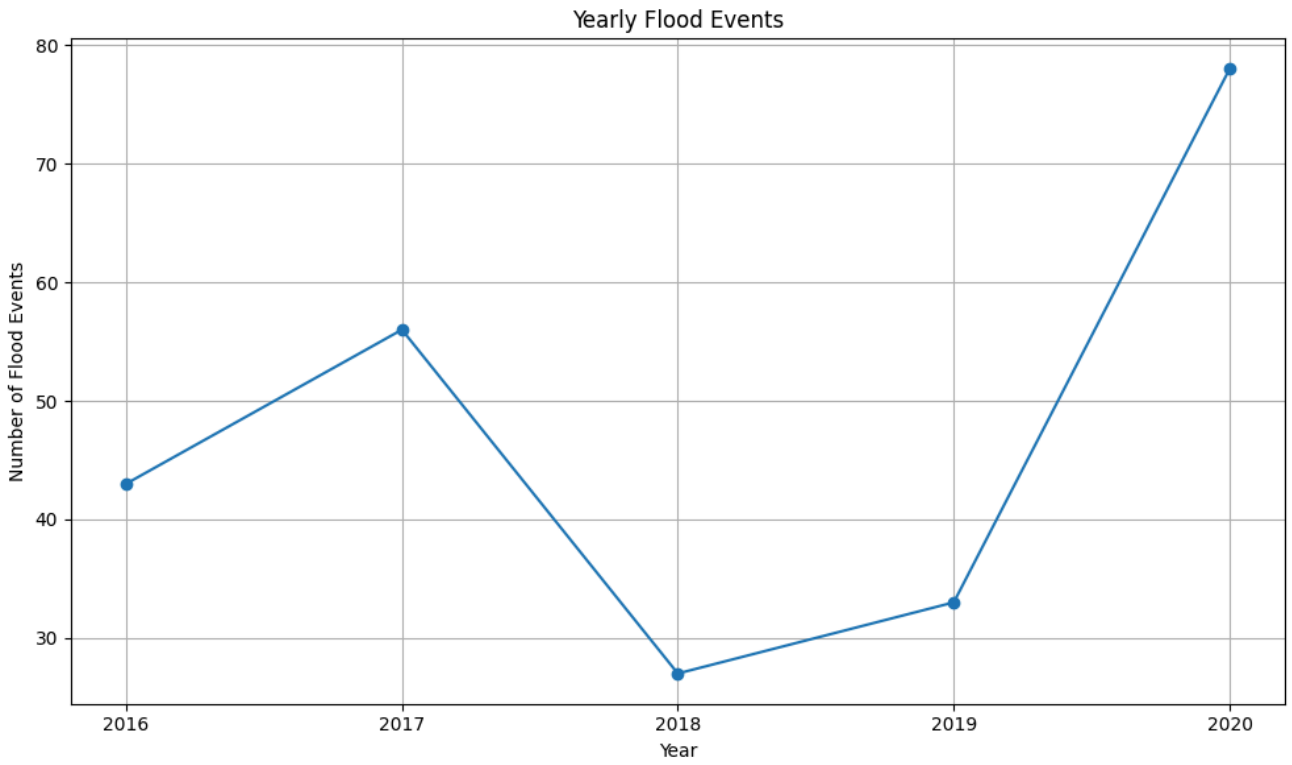


Gambar diatas menunjukan jumlah kejadian banjir yang ada di 3 wilayah di Jakarta yaitu, Jakarta Selatan, Jakarta Utara, Jakarta Pusat. Jakarta Selatan mencatat data tertinggi

dengan jumlah kejadian banjir sebanyak 160. Kemudian, data ini disusul dengan Jakarta Utara yang mencatat kejadian banjir sebanyak 60 kejadian. Sementara itu, Jakarta Pusat memiliki jumlah kejadian banjir yang paling sedikit, hanya sekitar 20. Perbedaan yang cukup signifikan ini menunjukan bahwa Jakarta Selatan merupakan wilayah yang paling rentang banjir diantara ketiga wilayah lainnya.



Gambar diatas menunjukkan jumlah kejadian banjir bulanan dari tahun 2016 hingga 2020 di Jakarta. Terlihat bahwa puncak kejadian banjir terjadi pada awal tahun 2020 dengan jumlah lebih dari 20 kejadian dalam satu bulan, yang menandakan adanya peristiwa banjir besar pada periode tersebut. Secara umum, terdapat peningkatan dan penurunan kejadian banjir yang cukup signifikan setiap tahunnya. Kejadian banjir cenderung meningkat di awal setiap tahun dan menurun menjelang pertengahan hingga akhir tahun. Hal tersebut juga disebabkan oleh musim hujan di Indonesia yang biasanya terjadi pada awal tahun.



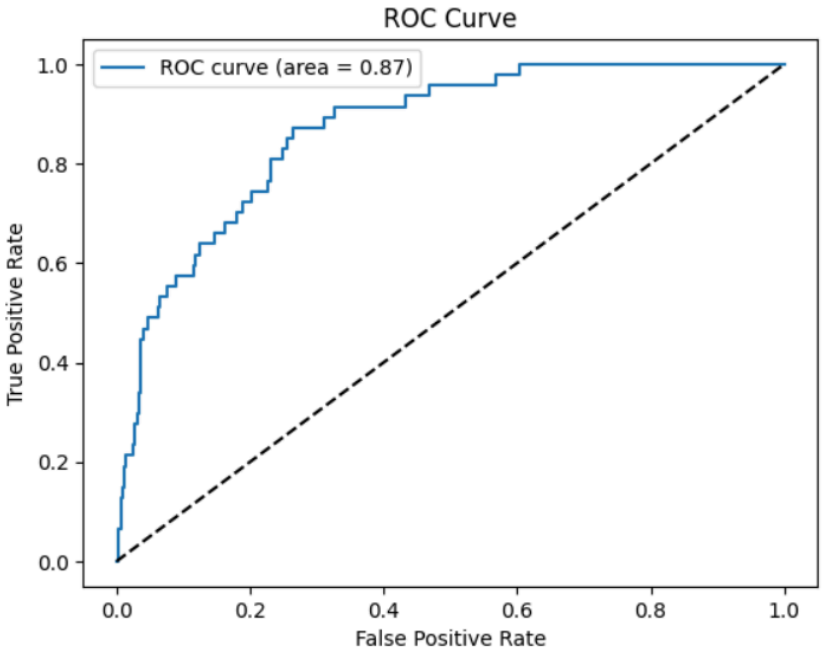
Gambar diatas menunjukkan jumlah kejadian banjir tahunan dari tahun 2016 hingga 2020 di Jakarta. Data ini memperlihatkan tren banjir yang meningkat dalam lima tahun terakhir, dengan peningkatan yang signifikan pada tahun 2020. Pada tahun 2016, tercatat sekitar 43 kejadian banjir. Jumlah ini meningkat pada tahun 2017 menjadi sekitar 56 kejadian. Namun, pada tahun 2018, terjadi penurunan yang cukup besar dengan hanya terjadinya sekitar 27 kejadian banjir. Tahun berikutnya, 2019, terdapat kenaikan jumlah kejadian banjir menjadi sekitar 33 kejadian. Puncaknya terjadi pada tahun 2020, di mana jumlah kejadian banjir mengalami kenaikan signifikan sehingga mencapai sekitar 78 kejadian. Hal ini menunjukkan bahwa tahun tersebut merupakan periode dengan intensitas banjir tertinggi selama lima tahun terakhir dari 2016.

Confusion Matrix (Actual/Predicted)	Not Flood (0)	Flood (1)
Not Flood (0)	447 (True Negative)	134 (False Positive)
Flood (1)	9 (False Negatives)	38 (True Positive)

Tabel di atas menunjukan confusion matrix dari model Logistic Regression yang digunakan untuk memprediksi kejadian banjir. Dari total data yang ada, model berhasil untuk memprediksi 447 kasus “tidak banjir” dengan benar (true negative), namun salah memprediksi 134 kasus “tidak banjir” sebagai “banjir” (false positive). Untuk kasus sebenarnya “banjir”, model berhasil memprediksi dengan benar sebanyak 38 kasus (true positive), dan hanya salah dalam 9 kasus (false positive). Dengan hasil ini, model mampu mengenali sebagian besar kejadian banjir dengan cukup baik, meskipun terdapat cukup banyak kesalahan prediksi positif palsu yang menunjukan bahwa model cenderung sedikit over sensitive.

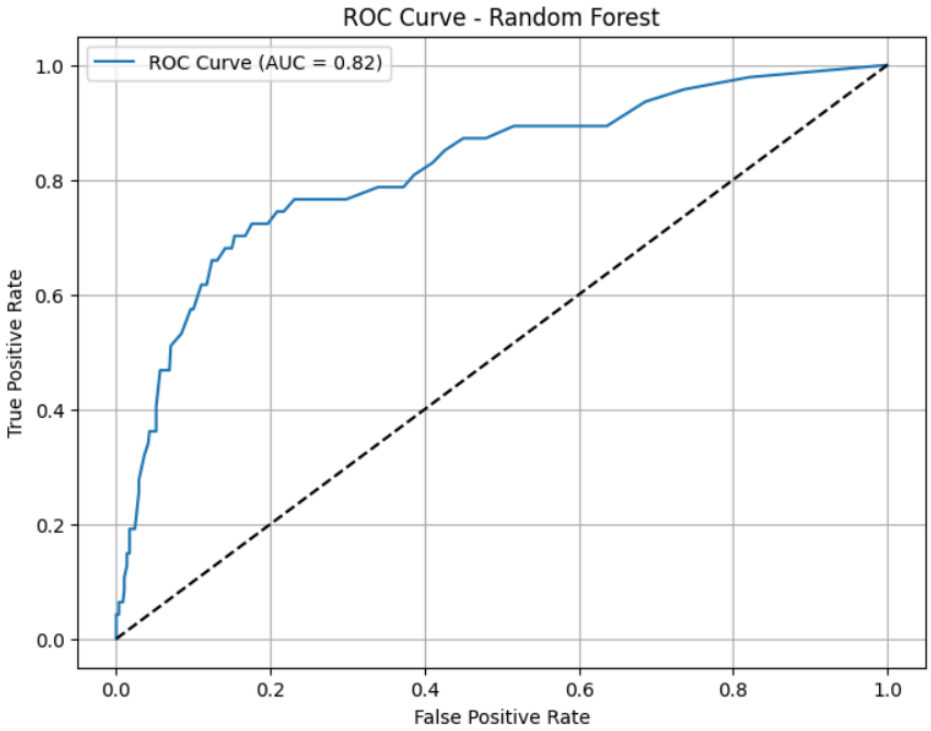
Confusion Matrix (Actual/Predicted)	Not Flood (0)	Flood (1)
Not Flood (0)	561 (True Negative)	19 (False Positive)
Flood (1)	33 (False Negatives)	14 (True Positive)

Tabel di atas menunjukan *confusion matrix* dari model *Random Forest* yang digunakan untuk memprediksi kejadian banjir. Dari total data yang ada, model berhasil untuk memprediksi 561 kasus “tidak banjir” dengan benar (*true negative*), namun salah memprediksi 19 kasus “tidak banjir” sebagai “banjir” (*false positive*). Untuk kasus sebenarnya “banjir”, model berhasil memprediksi dengan benar sebanyak 14 kasus (*true positive*), dan hanya salah dalam 33 kasus (*false positive*). Dengan hasil ini, model mampu mengenali sebagian besar kejadian tidak banjir dengan sangat baik, tetapi masih kesulitan dalam mendeteksi kasus banjir secara benar. Hal ini ditandai dengan jumlah *false negative* yang cukup tinggi.



Gambar di atas menampilkan kurva ROC (*Receiver Operating Characteristic*) dari model *Logistic Regression*, dengan area di bawah kurva (AUC) sebesar 0.87 atau 87%. Kurva ROC ini menunjukan hubungan antara *True Positive Rate* (sensitivitas) dan *False Positive Rate* pada berbagai batas keputusan atau threshold. AUC yang memiliki nilai 0.87 atau 87% menjelaskan bahwa model memiliki kemampuan klasifikasi yang cukup baik dalam membedakan antara banjir dan tidak banjir. Semakin mendekati 1.0, semakin baik performa dari model yang digunakan. Posisi kurva yang jauh di atas garis diagonal menandakan bahwa model *Logistic Regression* memberikan prediksi yang cukup baik.

\



Gambar di atas menampilkan kurva ROC (*Receiver Operating Characteristic*) dari model *Random Forest*, dengan area di bawah kurva (AUC) sebesar 0.82 atau 82%. Nilai AUC sebesar 82% ini menandakan bahwa model *Random Forest* memiliki kemampuan yang cukup baik, meskipun sedikit lebih rendah jika dibandingkan dengan *Logistic Regression* yang memiliki nilai AUC 87%. Kurva yang terletak diatas garis diagonal menunjukan bahwa model ini masih cukup baik dalam mengklasifikasikan data.

BAB 4
Kesimpulan

Jakarta semakin rentan terhadap banjir akibat perubahan iklim, penurunan muka tanah, dan sistem drainase yang kurang optimal, sehingga diperlukan analisis prediktif berbasis data iklim untuk mitigasi dan pengambilan keputusan yang efektif. Dalam penelitian ini, model yang digunakan adalah algoritma *Logistic Regression* dan *Random Forest* dengan platform Google Collab untuk memprediksi bencana banjir di daerah Jakarta menggunakan *dataset* yang relevan. Berdasarkan hasil penelitian, masing-masing algoritma memiliki kelebihan dan kekurangan masing-masing. Logistic Regression memiliki nilai ROC AUC lebih besar di 87% dan nilai Recall di 81%, namun banyak False Positives akibat precision rendah di 22%. Random Forest memiliki nilai hasil akurasi keseluruhan yang lebih besar di 92% dan precision di 42%, namun algoritma tersebut mendapatkan nilai recall yang jauh lebih rendah di 30%. Maka, dapat disimpulkan bahwa kita dapat menggunakan Logistic Regression jika prioritas utamanya adalah untuk mendeteksi banjir karena nilai recall yang lebih tinggi, meskipun itu berarti lebih banyak false positives yang terjadi. Jika kita lebih memprioritaskan precision yang lebih tinggi, lebih sedikit false positives, serta dapat mentoleransi hilangnya beberapa flood events, maka kita dapat menggunakan algoritma Random Forest.

REFERENSI

- Richard, C. (2022). *Climate and Flood Jakarta 2016-2020* [Data set]. Kaggle. https://www.kaggle.com/datasets/christopherrichardc/climate-and-flood-jakarta
- CNN Indonesia. (2024, 28 November). 61 RT di Jakarta Terendam Banjir, Tertinggi 2,6 Meter. Diakses dari: https://www.cnnindonesia.com/nasional/20241128102945-20-1171557/61-rt-di-jakarta-terendam-banjir-tertinggi-26-meter
- Dinas Sumber Daya Air DKI Jakarta. (2023). Penurunan Muka Tanah di Jakarta. Diakses dari: https://dsda.jakarta.go.id/detail-artikel/19
- Warta Ekonomi. (2024). Jakarta Berpotensi Alami Penurunan Tanah 3 Meter, Kementerian ESDM Stop Izin Baru Sedot Air Tanah. Diakses dari: https://wartaekonomi.co.id/read554431/jakarta-berpotensi-alami-penurunan-tanah-3-meter-kementerian-esdm-stop-izin-baru-sedot-air-tanah
- BMKG. (2024). Analisis Laju Perubahan Curah Hujan Tahunan. Diakses dari: https://www.bmkg.go.id/iklim/analisis-laju-perubahan-curah-hujan

LINK COLAB

https://colab.research.google.com/drive/1emiI0TVi8-WiQFhDDUmgmnHicIS1WEg4?usp=sharing