Laporan Proyek Akhir

Location-Based Species Presence Prediction

Disusun Oleh:

12S22003 – Yohana Natalia Siahaan

12S22016 – Desri Stevie Natalie Dabukke

12S22017 – Lenna Febriana

INSTITUT TEKNOLOGI DEL

FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO

TAHUN AJARAN 2024/2025

BAB 1 PENDAHULUAN

Keanekaragaman hayati merupakan salah satu aspek penting dalam menjaga keseimbangan ekosistem dan keberlanjutan lingkungan hidup. Salah satu tantangan utama dalam pengelolaan keanekaragaman hayati adalah mengetahui spesies apa saja yang ada atau berpotensi ada di suatu wilayah geografis. Proses ini biasanya memerlukan survei lapangan yang intensif dan memakan banyak waktu, tenaga, serta biaya. Namun, dengan kemajuan teknologi informasi dan ketersediaan data lingkungan berskala besar, kini terbuka peluang untuk mengembangkan sistem prediksi kehadiran spesies secara otomatis.

Salah satu pendekatan yang digunakan dalam konteks ini adalah *Species Distribution Modelling (SDM)*, yaitu metode yang memodelkan hubungan antara kondisi lingkungan dengan keberadaan spesies tertentu. Pendekatan ini sangat berguna untuk menghasilkan peta distribusi spesies, mendukung konservasi, dan membantu proses identifikasi spesies. Melalui kompetisi GeoLifeCLEF 2025, tersedia data observasi kehadiran spesies serta data lingkungan seperti citra satelit, tutupan lahan, iklim, dan intensitas aktivitas manusia, yang dapat dimanfaatkan untuk membangun model prediktif berbasis lokasi.

Proyek ini dirancang untuk menjawab tantangan tersebut dengan mengembangkan model *machine learning* yang mampu memprediksi spesies tumbuhan yang mungkin hadir di suatu titik lokasi berdasarkan data GPS dan atribut lingkungan sekitarnya. Sistem ini diharapkan tidak hanya memberikan kontribusi bagi penelitian dan konservasi, tetapi juga mendorong keterlibatan masyarakat luas dalam pengamatan lingkungan melalui teknologi berbasis data.

BAB 2 BUSSINESS UNDERSTANDING

2.1 Objektif Bisnis

Proyek ini bertujuan untuk mengembangkan sistem prediksi kehadiran spesies tumbuhan pada suatu lokasi berbasis data lokasi (GPS) dan berbagai prediktor lainnya seperti citra satelit, data iklim, tutupan lahan, serta jejak aktivitas manusia. Sistem ini diharapkan dapat:

a. Membantu upaya konservasi dan manajemen keanekaragaman hayati dengan memetakan keberadaan spesies secara akurat.
b. Menyediakan alat pendukung pengambilan keputusan bagi peneliti ekologi, pemerintah, dan organisasi lingkungan.
c. Meningkatkan efisiensi alat identifikasi spesies otomatis (contoh: Pl@ntNet) dengan menyaring spesies berdasarkan lokasi.
d. Mendorong partisipasi masyarakat umum (*citizen scientists*) melalui rekomendasi spesies berbasis lokasi.
e. Mempercepat anotasi dan validasi data observasi spesies untuk memperkaya data set keanekaragaman hayati.

2.2 Tujuan Teknis

Untuk memenuhi tujuan bisnis di atas, tujuan teknis proyek ini mencakup:

1. Membangun model Species Distribution Modelling (SDM) berbasis data spasial dan lingkungan.
2. Menggunakan data observasi seperti:
    a. Koordinat GPS
    b. Citra satelit (Sentinel-2)
    c. Deret waktu satelit (Landsat ARD)
    d. Deret waktu iklim (CHELSA)
    e. Raster lingkungan (bioclimatic, tanah, dll)
3. Melatih model klasifikasi multikelas atau multilabel untuk memprediksi satu atau lebih spesies yang mungkin hadir pada titik lokasi tertentu.
4. Evaluasi performa model menggunakan metrik seperti Top-k Accuracy, Mean Average Precision (mAP), atau Recall@k

2.3 Rencana Proyek

Proyek ini dirancang untuk diselesaikan dalam jangka waktu 5 minggu, dimulai pada minggu ke-11 hingga minggu ke-15 kalender akademik kampus. Setiap minggunya difokuskan pada fase berbeda dalam siklus Data Science untuk memastikan alur kerja yang sistematis dan hasil yang optimal. Berikut adalah timeline dari pengerjaan proyek ini :

Task	Minggu				
Bussines Understanding					
Data Understanding					
Data Cleaning & Preparation					
Label Construction					
Modeling					
Evaluation & Improvement					
Final Integration & Reporting					
Presentation					

Dalam perencanaan proyek ini, algoritma Random Forest dipilih sebagai model utama untuk membangun sistem klasifikasi berdasarkan metadata pengamatan tanaman. Pemilihan model ini didasarkan pada beberapa pertimbangan, antara lain kemampuannya yang baik dalam menangani data tabular, robust terhadap outlier dan noise, serta minimnya kebutuhan preprocessing yang kompleks. Random Forest juga memiliki kelebihan dalam menangani feature non-linear dan interaksi antar fitur, yang sangat relevan dalam konteks data spasial dan temporal seperti latitude, longitude, tanggal, dan waktu pengamatan. Selain itu, model ini menyediakan interpretasi yang baik melalui *feature importance*, yang dapat membantu dalam menganalisis fitur mana yang paling berpengaruh dalam prediksi. Dengan kemudahan implementasi dan performa yang kompetitif, Random Forest menjadi pilihan yang tepat untuk digunakan dalam proyek ini, terutama untuk membangun baseline model yang dapat dikembangkan lebih lanjut di tahap evaluasi dan pengujian.

BAB 3 DATA UNDERSTANDING

3.1 Pengumpulan Data

Proses pengumpulan data merupakan langkah awal yang sangat krusial dalam memahami dan mempersiapkan data untuk analisis lebih lanjut. Dalam kompetisi GeoLifeCLEF25, data yang digunakan terdiri dari dua kategori utama, yaitu Data Observasi Spesies dan Data Lingkungan. Data tersebut berasal dari beberapa sumber yang berbeda dan memiliki format yang beragam. Berikut adalah penjelasan rinci mengenai sumber data, jenis data, dan format yang disediakan.

a. Data Observasi Spesies

Data observasi spesies merupakan data yang sangat penting dalam memahami distribusi spesies di berbagai wilayah. Data ini terbagi menjadi dua jenis utama: Presence-Absence (PA) dan Presence-Only (PO). Data Presence-Absence (PA) mencatat keberadaan atau ketiadaan spesies di lokasi tertentu berdasarkan hasil survei. Data PA ini sangat berguna untuk mengatasi masalah pengamatan yang tidak lengkap, terutama dalam menghadapi *false absence* di mana suatu spesies mungkin tidak terdeteksi di suatu tempat meskipun sebenarnya ada. Terdapat sekitar 100.000 survei untuk sekitar 10.000 spesies flora di Eropa. Setiap survei ini mencakup informasi tentang apakah spesies tersebut hadir atau tidak pada lokasi yang diawasi. Data PA disediakan dalam format CSV, yang mencakup informasi seperti surveyId, speciesId, dan location, serta metadata terkait survei dan spesies. Semua data ini dapat diakses melalui platform Kaggle dan repositori Seafile dalam folder PresenceAbsenceSurveys/.

Sementara itu, data Presence-Only (PO) hanya mencatat keberadaan spesies tanpa mencatat ketidakhadirannya, yang lebih umum ditemukan karena data PO lebih banyak dan tersebar luas. Data PO ini, meskipun sangat berguna, memiliki kelemahan karena tidak menyertakan informasi tentang ketidakhadiran spesies, sehingga dapat menimbulkan bias. Data PO dalam kompetisi ini mencakup sekitar 5 juta pengamatan yang diperoleh dari berbagai sumber, termasuk Global Biodiversity Information Facility (GBIF). Data ini berformat CSV, dengan kolom yang mencakup informasi seperti surveyId, species Id, location, dan waktu pengamatan. Sama seperti data PA, data PO dapat diakses melalui repositori Seafile, di folder PresenceOnlyOccurences/.

b. Data Lingkungan

Selain data observasi spesies, data lingkungan juga sangat penting untuk memahami faktor-faktor yang mempengaruhi distribusi spesies. Data lingkungan ini mencakup berbagai informasi geospasial, termasuk citra satelit dan variabel iklim yang mempengaruhi ekosistem. Salah satu jenis data lingkungan yang tersedia adalah Citra Satelit. Citra ini berasal dari satelit Sentinel-2 dan memberikan gambaran visual mengenai kondisi vegetasi serta lanskap di sekitar lokasi pengamatan spesies. Citra satelit ini memiliki resolusi 10 meter per piksel dan tersedia dalam format TIFF, dengan ukuran 64x64 piksel yang mewakili area 640mx640m. Setiap citra satelit dapat dihubungkan dengan ID pengamatan spesies tertentu, sehingga memungkinkan kita untuk mengaitkan citra dengan data spesies yang relevan. Data citra satelit ini dapat diakses melalui folder SatellitePatches/ di repositori Seafile.

Selain citra satelit, terdapat juga data Time Series Satelit, yang berisi informasi mengenai perubahan vegetasi dan kondisi lingkungan sepanjang waktu. Time series ini mencakup data dari enam band satelit (R, G, B, NIR, SWIR1, SWIR2) yang merepresentasikan perubahan musiman dari musim dingin 1999 hingga musim gugur 2020. Data ini disediakan dalam dua format, yaitu CSV dan tensor 3D, yang mencakup nilai rata-rata setiap band satelit untuk setiap musim. Data ini sangat berguna untuk menganalisis perubahan jangka panjang di lingkungan tempat spesies diamati. Semua data time series satelit ini dapat ditemukan di folder SatelliteTimeSeries/ di Seafile.

Selanjutnya, data Raster Lingkungan mencakup informasi tentang iklim, tanah, penutup lahan, dan jejak manusia. Data raster ini sangat penting untuk memahami faktor-faktor yang mempengaruhi distribusi spesies di wilayah tertentu. Misalnya, data tentang bioclimatic rasters dapat menunjukkan suhu dan kelembapan, sedangkan soil rasters memberikan informasi tentang jenis tanah di area pengamatan. Data ini juga tersedia dalam format GeoTIFF dan CSV, yang masing-masing menyimpan informasi terkait dengan lokasi pengamatan spesies. Semua data raster lingkungan ini dapat ditemukan dalam folder EnvironmentalRasters/ di repositori Seafile, dengan subfolder yang terorganisir berdasarkan jenis data, seperti Climate, SoilGrids, dan LandCover.

Berikut tampilan code untuk meload data, dan kemudian head() untuk menampilkan 5 data teratas.

```
In [1]: # Import dan load dataset
        import pandas as pd

        # Load data
        path = r'C:\Users\ASUS\Downloads\GLC25_P0_metadata_train.csv'
        df = pd.read_csv(path)

        df.head()
```

Out[1]:

	publisher	year	month	day	lat	lon	geoUncertaintyInM	taxonRank	date	dayOfYear	speciesId	surveyId
0	Pl@ntNet	2019	5.0	5.0	43.74605	1.573057	6.0	SPECIES	2019-05-05	125	3383.0	1
1	Pl@ntNet	2021	3.0	17.0	42.12559	0.314948	5.0	SPECIES	2021-03-17	76	1152.0	2
2	Pl@ntNet	2021	6.0	5.0	48.29520	-0.934518	24.9	SPECIES	2021-06-05	156	6772.0	3
3	iNaturalist.org	2021	6.0	9.0	53.63367	-2.644535	8.0	SPECIES	2021-06-09	160	3318.0	4
4	iNaturalist.org	2021	4.0	1.0	49.79471	7.925086	15.0	SPECIES	2021-04-01	91	3374.0	5

3.2 Penelaahan Data

Dataset ini merupakan kumpulan data observasi spesies flora Eropa yang bersumber dari berbagai platform seperti GBIF, iNaturalist, dan Pl@ntNet. Data ini terdiri dari dua jenis utama, yaitu Presence-Only (PO) dan Presence-Absence (PA). PO merupakan data observasi yang hanya mencatat kehadiran spesies tanpa informasi ketidakhadiran, dengan jumlah sekitar lima juta observasi. Sementara itu, PA mencakup sekitar seratus ribu survei dengan pencatatan yang lebih sistematis karena mencakup kehadiran dan ketidakhadiran spesies. Kehadiran PA data sangat penting karena membantu mengatasi kelemahan dari PO data, terutama dalam hal kemungkinan kesalahan asumsi tentang tidak adanya spesies.

Berdasarkan hasil penelaahan terhadap struktur dan isi dataset, kami memutuskan untuk menggunakan data Presence-Only (PO) dalam proses analisis lebih lanjut. Keputusan ini didasarkan pada kenyataan bahwa data PO mencakup informasi yang serupa dengan data Presence-Absence (PA), baik dari segi atribut seperti waktu, lokasi, identitas spesies, hingga keterkaitan dengan data lingkungan melalui surveyId. Bahkan, data PO memiliki cakupan yang jauh lebih luas, baik dari jumlah observasi (sekitar lima juta entri) maupun dari keragaman wilayah dan spesies yang tercatat. Dengan kelengkapan dan cakupan yang lebih besar ini, data PO memberikan gambaran yang lebih komprehensif mengenai distribusi spesies di Eropa. Meskipun data PO dikumpulkan secara oportunistik dan memiliki potensi bias, dengan pendekatan pemodelan yang tepat, kelemahan tersebut masih dapat diatasi. Oleh karena itu, penggunaan data PO dipandang sebagai pilihan yang lebih efisien dan informatif dalam konteks studi distribusi spesies ini.

Berikut adalah gambar potongan code yang dimana mendeskripsikan Presence-Only.

```
In [7]: df.info()
        df.describe()
        df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 5010338 entries, 0 to 5079796
Data columns (total 12 columns):
 #   Column             Dtype
---  ------             -----
 0   publisher          object
 1   year               int64
 2   month              float64
 3   day                float64
 4   lat                float64
 5   lon                float64
 6   geoUncertaintyInM  float64
 7   taxonRank          object
 8   date               datetime64[ns]
 9   dayOfYear          int64
 10  speciesId          float64
 11  surveyId           int64
dtypes: datetime64[ns](1), float64(6), int64(3), object(2)
memory usage: 496.9+ MB
```

Out[7]:

	publisher	year	month	day	lat	lon	geoUncertaintyInM	taxonRank	date	dayOfYear	speciesId	surveyId
0	PI@ntNet	2019	5.0	5.0	43.74605	1.573057	6.0	SPECIES	2019-05-05	125	3383.0	1
1	PI@ntNet	2021	3.0	17.0	42.12559	0.314948	5.0	SPECIES	2021-03-17	76	1152.0	2
2	PI@ntNet	2021	6.0	5.0	48.29520	-0.934518	24.9	SPECIES	2021-06-05	156	6772.0	3
3	iNaturalist.org	2021	6.0	9.0	53.63367	-2.644535	8.0	SPECIES	2021-06-09	160	3318.0	4
4	iNaturalist.org	2021	4.0	1.0	49.79471	7.925086	15.0	SPECIES	2021-04-01	91	3374.0	5

Berdasarkan hasil eksplorasi awal menggunakan df.info(), dataset ini memuat sebanyak 5.012.338 entri dengan 12 kolom. Kolom-kolom tersebut meliputi informasi penting seperti publisher (sumber data), year, month, dan day yang menunjukkan waktu pengamatan, serta lat dan lon yang menunjukkan lokasi geografis pengamatan. Selain itu, terdapat juga kolom geoUncertaintyInM yang mengindikasikan tingkat ketidakpastian posisi lokasi dalam satuan meter. Informasi taksonomi spesies dicantumkan dalam kolom taxonRank dan speciesId, sementara date dan dayOfYear memberikan informasi waktu dalam format datetime dan ordinal hari dalam tahun. Kolom surveyId berfungsi sebagai identitas unik dari setiap observasi, yang juga digunakan untuk menghubungkan data ini dengan data lingkungan seperti citra satelit dan raster.

Dataset ini memiliki karakteristik geospasial dan temporal yang kuat. Dengan kombinasi lokasi (lat, lon) dan waktu (year, date, dayOfYear), pengguna dapat melakukan analisis tren musim, perubahan distribusi spesies, maupun integrasi dengan data lingkungan. Namun, dari tipe data yang digunakan, seperti float64 pada kolom day, month, geoUncertaintyInM, dan speciesId, terdapat potensi keberadaan nilai kosong (missing values) yang perlu ditangani pada tahap pembersihan data. Selain itu, data PO yang dikumpulkan secara oportunistik tanpa protokol pengamatan yang baku dapat mengandung bias pengambilan sampel. Hal ini menuntut perhatian ekstra dalam membangun model, agar tidak terjebak pada kesimpulan yang keliru akibat *false absence* atau *sampling bias*.

3.3 Validasi Data

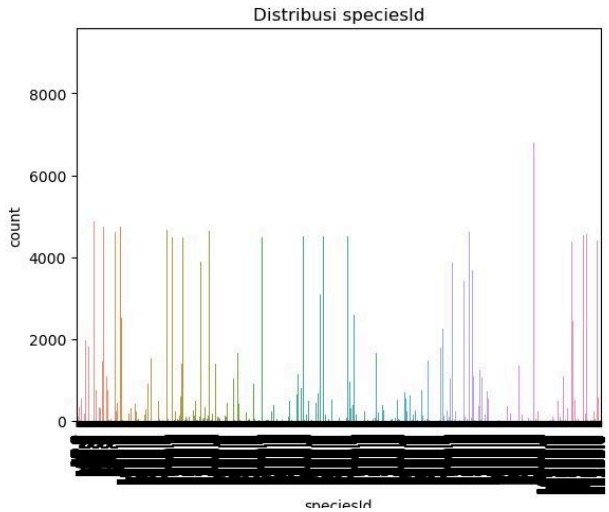
3.3.1 Distribusi dan Struktur Data

Pada tahap awal *data understanding*, dilakukan eksplorasi terhadap distribusi data untuk memahami karakteristik dasar dari dataset. Visualisasi distribusi speciesId menunjukkan bahwa data bersifat tidak seimbang (imbalanced), di mana terdapat beberapa spesies dengan jumlah observasi yang sangat tinggi, sementara sebagian besar spesies lainnya hanya memiliki sedikit kemunculan. Hal ini menjadi indikasi awal bahwa permasalahan *class imbalance* perlu diperhatikan dalam proses modeling.

Distribusi spasial juga dianalisis melalui visualisasi boxplot nilai latitude terhadap speciesId. Hasilnya menunjukkan bahwa masing-masing spesies tersebar dalam rentang lintang yang bervariasi, dengan keberadaan sejumlah outlier. Namun secara keseluruhan, nilai lat dan lon menunjukkan sebaran yang wajar sehingga dapat digunakan sebagai fitur spasial dalam proses prediksi.

```
In [8]: import seaborn as sns
        import matplotlib.pyplot as plt

        sns.countplot(data=df, x='speciesId')
        plt.xticks(rotation=90)
        plt.title('Distribusi speciesId')
        plt.show()
```



3.2.2 Analisis Korelasi Fitur Numerik

Analisis korelasi dilakukan terhadap seluruh fitur numerik dalam dataset untuk mengevaluasi adanya hubungan linier antar fitur. Hasilnya menunjukkan beberapa temuan penting, antara lain:

1. Korelasi Tinggi:
    a. month dan dayOfYear memiliki korelasi sangat tinggi (0.99), yang wajar karena dayOfYear merupakan turunan langsung dari month dan day.
    b. lat dan lon memiliki korelasi sedang (0.36), menunjukkan bahwa ada hubungan geografis antar lokasi.
2. Korelasi Rendah/Mendekati Nol:
   speciesId, surveyId, dan geoUncertaintyInM memiliki korelasi sangat rendah terhadap hampir semua fitur lain, menunjukkan bahwa fitur-fitur ini bersifat independen atau tidak memiliki hubungan linier yang kuat.
3. Korelasi Negatif :
   year memiliki korelasi negatif dengan lat (-0.26) dan lon (-0.07), bisa jadi karena adanya pergeseran lokasi pengamatan dari waktu ke waktu.

Berdasarkan analisis korelasi antar atribut numerik, beberapa fitur menunjukkan hubungan yang cukup kuat satu sama lain, sementara sebagian besar lainnya bersifat independen. Fitur month dan dayOfYear memiliki korelasi sangat tinggi (nilai korelasi 0.99), yang menunjukkan bahwa keduanya menyampaikan informasi waktu yang hampir identik. Oleh karena itu, dalam proses pemodelan hanya salah satu dari keduanya yang akan dipilih untuk menghindari multikolinearitas.

Sementara itu, fitur spasial seperti lat dan lon memiliki korelasi rendah terhadap seluruh fitur lainnya, termasuk target (speciesId). Hal ini mengindikasikan bahwa informasi lokasi bersifat independen dan penting untuk dipertahankan dalam model. Fitur geoUncertaintyInM, yang merepresentasikan tingkat ketidakpastian geografis, juga menunjukkan korelasi sangat rendah terhadap fitur lainnya, namun tetap relevan sebagai indikator kualitas data lokasi.

Fitur year dan day_of_week juga menunjukkan korelasi yang rendah terhadap atribut lainnya, namun dapat memberikan kontribusi penting dalam membedakan pola observasi berdasarkan waktu dan musim. Secara keseluruhan, fitur-fitur yang dipilih (lat, lon, year, month, dayOfYear, geoUncertaintyInM, dan day_of_week) mencakup dimensi spasial, temporal, dan kualitas observasi, dan dipertimbangkan sebagai fitur yang relevan untuk pemodelan prediksi kehadiran spesies tanaman.

```
In [9]: # Contoh korelasi numerik
        df.corr(numeric_only=True)
```

Out[9]:

	year	month	day	lat	lon	geoUncertaintyInM	dayOfYear	speciesId	surveyId
year	1.000000	-0.071752	-0.007250	-0.261237	-0.073941	0.041526	-0.072042	-0.010201	0.001433
month	-0.071752	1.000000	-0.101343	0.122559	0.057670	-0.072411	0.990006	0.005292	0.000856
day	-0.007250	-0.101343	1.000000	-0.002875	-0.005559	-0.006707	0.039658	0.000469	0.001252
lat	-0.261237	0.122559	-0.002875	1.000000	0.358924	-0.081073	0.122031	-0.009588	-0.000460
lon	-0.073941	0.057670	-0.005559	0.358924	1.000000	-0.020131	0.056881	-0.002063	0.000543
geoUncertaintyInM	0.041526	-0.072411	-0.006707	-0.081073	-0.020131	1.000000	-0.073480	-0.009255	-0.001885
dayOfYear	-0.072042	0.990006	0.039658	0.122031	0.056881	-0.073480	1.000000	0.005471	0.001037
speciesId	-0.010201	0.005292	0.000469	-0.009588	-0.002063	-0.009255	0.005471	1.000000	0.001027
surveyId	0.001433	0.000856	0.001252	-0.000460	0.000543	-0.001885	0.001037	0.001027	1.000000

Dalam bentuk HeatMap :

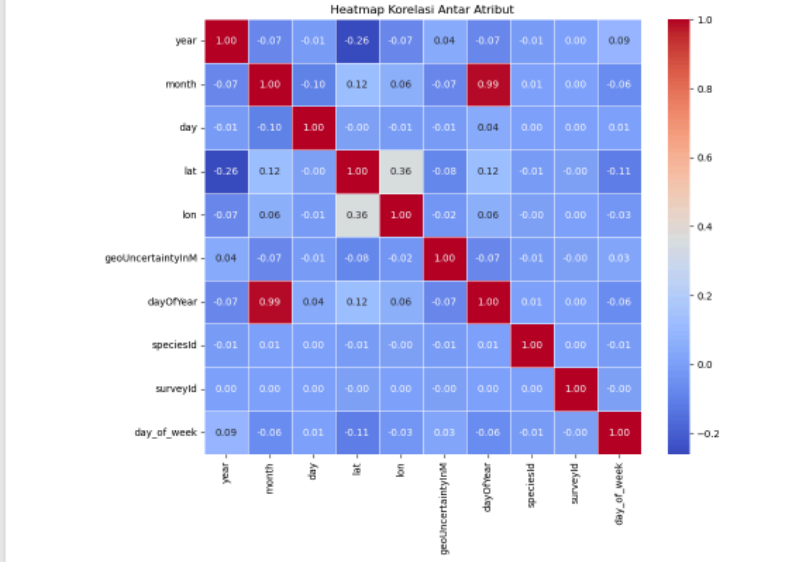
```
In [12]: import matplotlib.pyplot as plt
         import seaborn as sns

         # Hitung korelasi antar fitur numerik
         correlation_matrix = df.corr(numeric_only=True)

         # Atur ukuran plot
         plt.figure(figsize=(10, 8))

         # Buat heatmap dengan anotasi nilai korelasi
         sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=0.5)

         # Judul plot
         plt.title("Heatmap Korelasi Antar Atribut")
         plt.show()
```



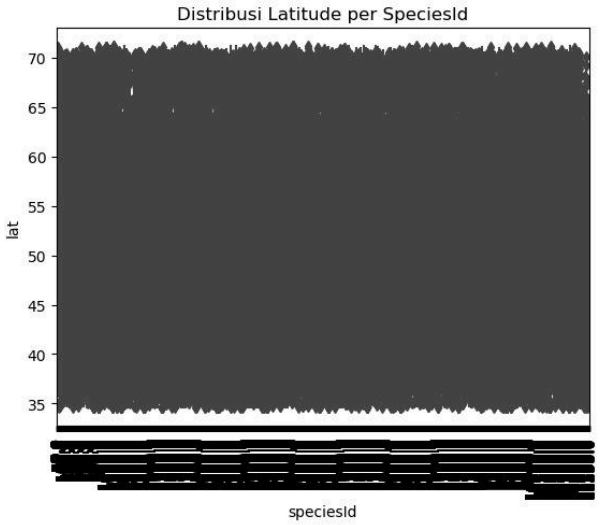
3.2.3 Pemilihan surveyId sebagai Arah Penelitian

Berdasarkan hasil eksplorasi dan validasi data, diputuskan bahwa speciesId akan dijadikan sebagai target utama dalam penelitian ini. Artinya, model yang akan dibangun diarahkan untuk memprediksi kehadiran suatu spesies berdasarkan informasi spasial (koordinat GPS), temporal (waktu observasi), serta faktor lingkungan lain yang tersedia.Pemilihan speciesId sebagai target juga didukung oleh fokus utama kompetisi GeoLife di Kaggle, yaitu memprediksi kehadiran spesies tanaman berdasarkan lokasi dan prediktor lingkungan. Dengan pendekatan ini, diharapkan model dapat mengidentifikasi pola keterkaitan antara spesies tertentu dan kondisi geografis maupun ekologis di sekitarnya.

In [10]:

```
# Visualisasi Distribusi
import seaborn as sns
import matplotlib.pyplot as plt

sns.boxplot(x='speciesId', y='lat', data=df)
plt.title('Distribusi Latitude per SpeciesId')
plt.xticks(rotation=90)
plt.show()
```



BAB 4 DATA PREPARATION

4.1 Memilah Data

Pada tahap ini, kita memisahkan data berdasarkan atribut yang relevan untuk analisis dan pelatihan model. Hal ini bertujuan untuk memastikan bahwa hanya data yang dibutuhkan untuk prediksi yang digunakan dalam pembuatan model.

Hal-hal yang perlu dilakukan adalah :

1. Memilih kolom-kolom yang relevan untuk prediksi, yaitu atribut-atribut yang memiliki hubungan signifikan dengan speciesId.
2. Mengabaikan kolom yang tidak relevan atau tidak diperlukan dalam analisis, seperti surveyId yang tidak mempengaruhi hasil prediksi spesies.

Berikut adalah code yang digunakan untuk memilah data, yaitu menggunakan fitur-fitur yang relevan digunakan untuk membangun model.

```
In [16]: # Daftar fitur relevan berdasarkan analisis sebelumnya
         fitur_relevan = ['lat', 'lon', 'year', 'month', 'dayOfYear', 'geoUncertaintyInM', 'day_of_week']

         # Memilih hanya kolom-kolom yang relevan
         X = df[fitur_relevan]
         y = df['speciesId']  # Label untuk prediksi
```

4.2 Membersihkan Data

Setelah data berhasil dipilah, tahap selanjutnya adalah pembersihan data untuk memastikan kualitas dan konsistensi data yang digunakan. Data yang diperoleh dari *Presence-Only* cenderung mengandung nilai yang hilang (*missing values*), duplikasi, atau kesalahan entri. Oleh karena itu, dilakukan proses seperti menghapus baris duplikat, mengisi atau menghapus nilai yang hilang, dan memastikan semua data memiliki format yang sesuai. Pembersihan data ini penting untuk menghindari bias atau kesalahan dalam analisis selanjutnya.

Berikut adalah code yang digunakan untuk proses *data cleaning*:

In [7]:
```
## Membersihkan Data
# Cek missing values
print("Missing values sebelum dibersihkan:")
print(df.isnull().sum())

# Drop baris yang mengandung missing values
df = df.dropna()

# Drop duplikat
df = df.drop_duplicates()

# Validasi latitude dan longitude
df = df[(df['lat'].between(-90, 90)) & (df['lon'].between(-180, 180))]

# Konversi kolom 'date' jika ada
if 'date' in df.columns:
    df['date'] = pd.to_datetime(df['date'], errors='coerce')
    df = df.dropna(subset=['date'])  # Hapus tanggal tidak valid
```

```
Missing values sebelum dibersihkan:
publisher              0
year                   0
month                484
day                 1819
lat                    0
lon                    0
geoUncertaintyInM      0
taxonRank              0
date                   0
dayOfYear              0
speciesId              0
surveyId               0
dtype: int64
```

Berdasarkan output code di atas, dapat dilihat bahwa sebagian besar kolom memiliki nilai lengkap, kecuali untuk kolom month dan day, yang memiliki jumlah nilai hilang yang signifikan. Hal ini menunjukkan bahwa ada masalah dengan pengisian data pada bagian bulan dan hari, yang mungkin perlu ditangani lebih lanjut, misalnya dengan mengisi nilai yang hilang atau memutuskan apakah entri yang hilang perlu dihapus. Setalah dilakukan data cleaning, maka Missing Values akan menjadi nol untuk setiap fitur.

In [15]:
```
print("Missing values setelah dibersihkan:")
print(df.isnull().sum())
```

```
Missing values setelah dibersihkan:
publisher            0
year                 0
month                0
day                  0
lat                  0
lon                  0
geoUncertaintyInM    0
taxonRank            0
date                 0
dayOfYear            0
speciesId            0
surveyId             0
day_of_week          0
location             0
dtype: int64
```

4.3 Mengkonstruksi Data

Tahap konstruksi data bertujuan untuk membuat fitur tambahan atau mengubah format data agar lebih mudah digunakan oleh model. Fitur baru ini dapat membantu model memahami pola dalam data dengan lebih baik.

Langkah-langkah yang dilakukan dalam tahap ini adalah:

1. Menambahkan fitur baru berdasarkan kolom tanggal, seperti hari dalam seminggu (day_of_week).
2. Jika tidak ada kolom dayOfYear, kita buat kolom tersebut berdasarkan tanggal.

3. Membuat fitur lokasi yang menggabungkan lat dan lon dalam format string untuk menunjukkan lokasi geografis.

```
In [8]: ## Mengkonstruksi Data
        # Tambahkan fitur hari dalam seminggu jika ada kolom date
        if 'date' in df.columns:
            df['day_of_week'] = df['date'].dt.dayofweek

        # Cek jika fitur dayOfYear tidak tersedia, maka buat
        if 'dayOfYear' not in df.columns:
            df['dayOfYear'] = pd.to_datetime(df[['year', 'month', 'day']], errors='coerce').dt.dayofyear

        # Buat fitur baru (contoh: kombinasi lokasi sebagai string)
        df['location'] = df['lat'].round(2).astype(str) + "_" + df['lon'].round(2).astype(str)
```

4.4 Menentukan Label Data

Berikut potongan code untuk label yang akan diprediksi:

```
In [9]: # Label yang akan diprediksi adalah speciesId
        y = df['speciesId']  # Target label
```

Kode df['speciesId'] digunakan untuk mengambil kolom speciesId dari DataFrame df. Kolom ini berisi label atau target yang ingin diprediksi dalam model machine learning. Tujuan dari kode ini, yaitu menentukan data target yang menjadi acuan dalam proses supervised learning. Label ini nantinya akan digunakan saat melatih model agar dapat mengenali dan memprediksi kategori yang sesuai.

4.5 Mengintegrasikan Data

Setelah memisahkan fitur dan label, kita mengintegrasikan data dengan memilih kolom-kolom yang relevan dan memastikan bahwa data siap digunakan dalam model.

Langkah-langkah yang dilakukan untuk mengintegrasi data adalah:

1. Integrasi data dengan memilih fitur yang relevan (yaitu: lat, lon, year, month, geoUncertaintyInM, day_of_week, dll.).
2. Menghapus kolom yang tidak diperlukan seperti speciesId dan surveyId untuk memastikan data hanya berisi fitur yang relevan untuk prediksi.

Berikut adalah code yang digunakan untuk mengintegrasi data.

```
In [13]: # Daftar fitur relevan berdasarkan heatmap
         fitur_relevan = ['lat', 'lon', 'year', 'month', 'dayOfYear', 'geoUncertaintyInM', 'day_of_week']

         # Pilih hanya kolom-kolom yang relevan
         X = df[fitur_relevan]
         y = df['speciesId']
```