Tugas 3: Tugas Mandiri 3 – Regresi Linear Sederhana

**Amaya Eshia - 0110224102\***

1 Teknik Informatika, STT Terpadu Nurul Fikri, Depok

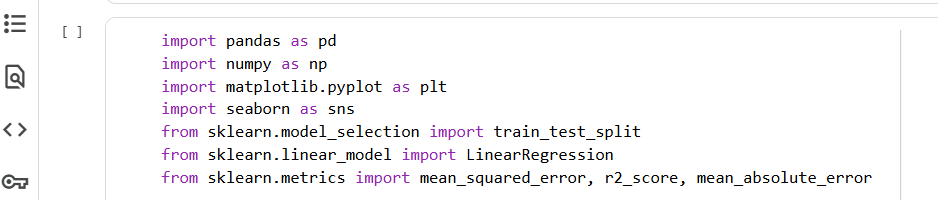
\*E-mail: [name@institution.edu](mailto:name@institution.edu) **– 0110224102@student.nurulfikri.ac.id**

**Abstract.** Praktikum ini bertujuan untuk mempelajari teknik regresi linear sederhana dalam pemodelan machine learning menggunakan dataset bike sharing (day.csv). Dataset dianalisis terlebih dahulu melalui statistik deskriptif, pemeriksaan missing values, dan analisis korelasi untuk memilih variabel independen yang paling berpengaruh terhadap variabel target 'cnt' (jumlah penyewaan sepeda). Pembagian data dilakukan dengan rasio 80:20 untuk training dan testing. Model regresi linear dibangun menggunakan library Scikit-learn dan Statsmodels untuk mendapatkan hasil yang detail. Hasil menunjukkan bahwa model memiliki R² score sekitar 0.82 pada data testing, menandakan kemampuan model dalam menjelaskan variasi data sebesar 82%. Evaluasi metrik seperti MAE, MSE, dan RMSE dilakukan, disertai visualisasi seperti scatter plot actual vs predicted, residual plot, dan distribusi residuals. Verifikasi dilakukan untuk memastikan model tidak overfitting dan residuals terdistribusi secara acak. Keterampilan ini penting dalam memprediksi variabel kontinu seperti jumlah penyewaan sepeda berdasarkan faktor cuaca dan waktu.

**Kata Kunci:** Machine Learning, Regresi Linear, Bike Sharing Dataset, Korelasi, Scikit-learn, Statsmodels, Python.

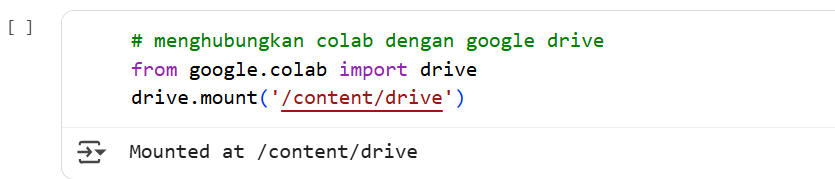
1. Penjelasan Koding Program

1.1 Import Library



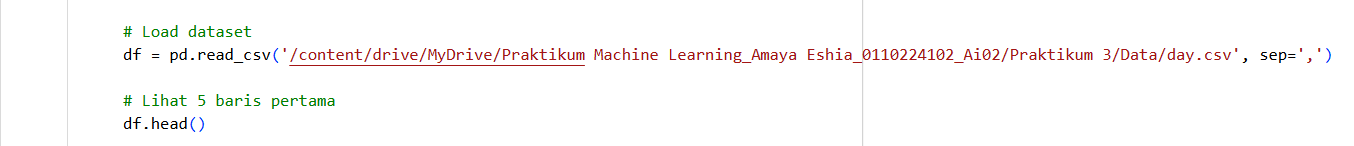
Mengimpor library Statsmodels dengan alias sm untuk analisis statistik, Pandas dengan alias pd untuk manipulasi data, Matplotlib dengan alias plt untuk visualisasi, dan NumPy dengan alias np untuk operasi matematika dan array. Selain itu, mengimpor library lain seperti Seaborn untuk visualisasi yang lebih baik, serta fungsi-fungsi dari Scikit-learn untuk pembagian data, model regresi, dan evaluasi metrik.

1.2 Menghubungkan Google Colab dengan GDrive



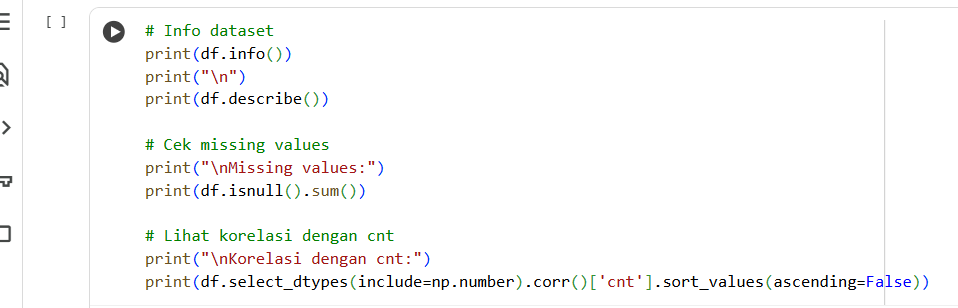
Menghubungkan Google Colab dengan Google Drive agar dapat mengakses file dataset yang tersimpan di Drive. Setelah dijalankan, akan muncul link untuk autorisasi akses.

1.3 Membaca File CSV dan Menampilkan Data Awal



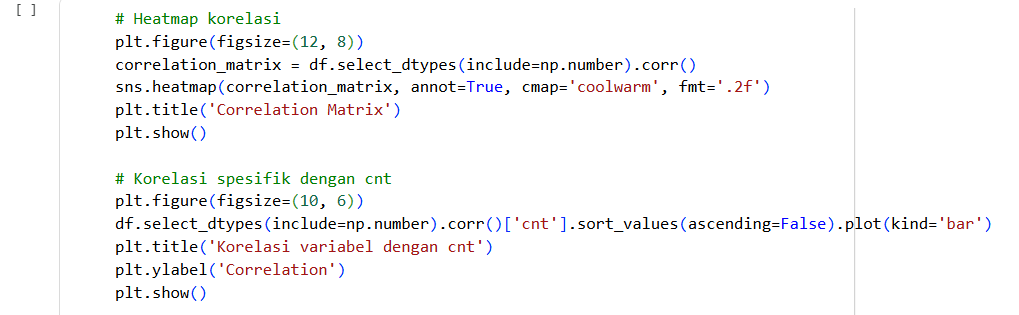
* Membaca file CSV dengan nama day.csv menggunakan fungsi pd.read\_csv() dan menyimpannya dalam DataFrame df.
* Menampilkan 5 baris pertama data menggunakan fungsi .head() untuk verifikasi awal.
* Menampilkan ringkasan statistik deskriptif menggunakan .describe() untuk memahami distribusi data..

1.4 Menganalisa Dataset di Awal



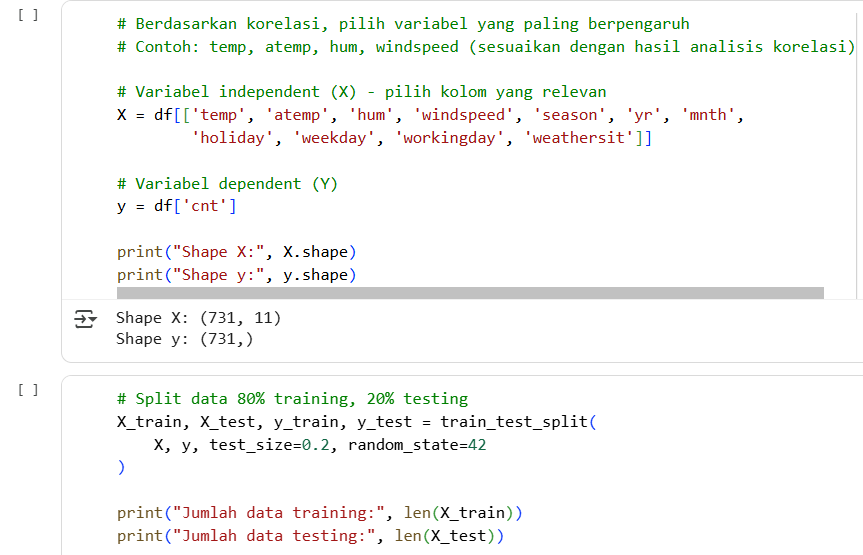
* Memeriksa informasi dataset menggunakan .info() untuk melihat tipe data dan jumlah entri.
* Memeriksa missing values menggunakan .isnull().sum() untuk memastikan tidak ada data yang hilang.
* Menghitung korelasi variabel numerik dengan target 'cnt' dan menampilkannya dalam urutan menurun

1.5 Membuat Heatmap korelasi



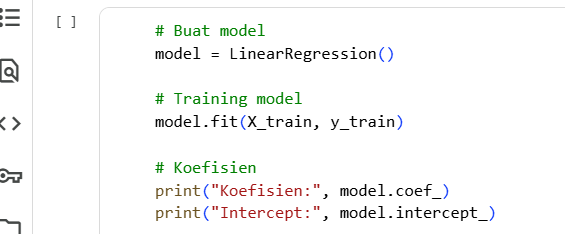
* Membuat heatmap korelasi menggunakan Seaborn untuk visualisasi matriks korelasi.
* Membuat bar plot korelasi spesifik dengan 'cnt' untuk melihat variabel yang paling berpengaruh.

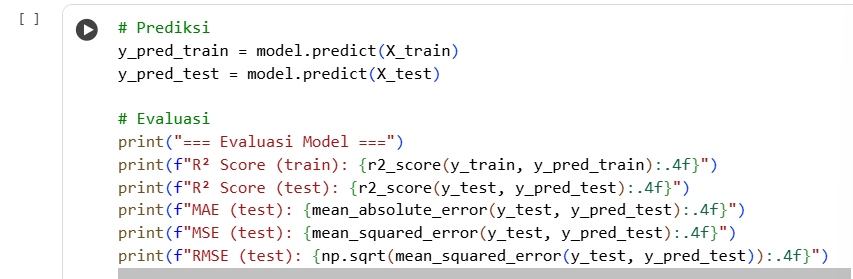
1.6 Pemilihan Variabel dan Pembagian Dataset



* Memilih variabel independen (X) berdasarkan analisis korelasi (contoh: 'temp', 'atemp', 'hum', dll.).
* Variabel dependen (y) adalah 'cnt'.
* Membagi data menjadi training (80%) dan testing (20%) menggunakan train\_test\_split dari Scikit-learn dengan random\_state=42 untuk reproduktibilitas.
* Menampilkan ukuran data training dan testing untuk verifikasi

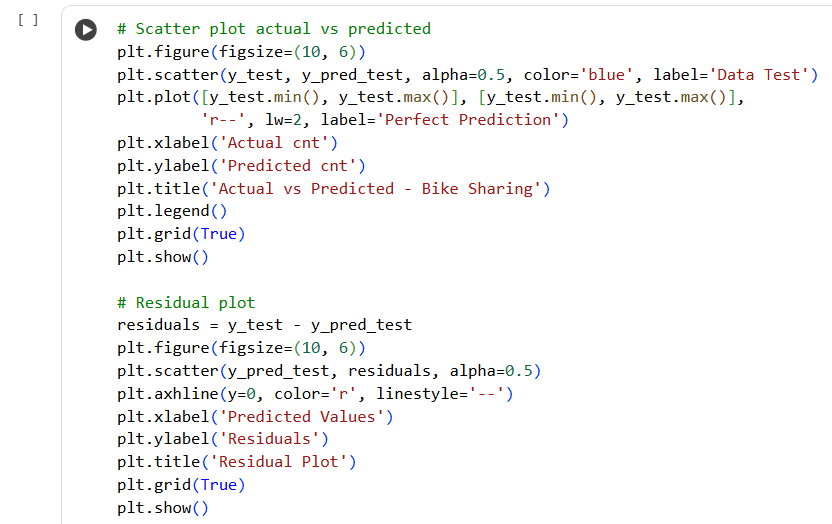
1.7 Pembuatan Model Regresi Linear Menggunakan Scikit-learn

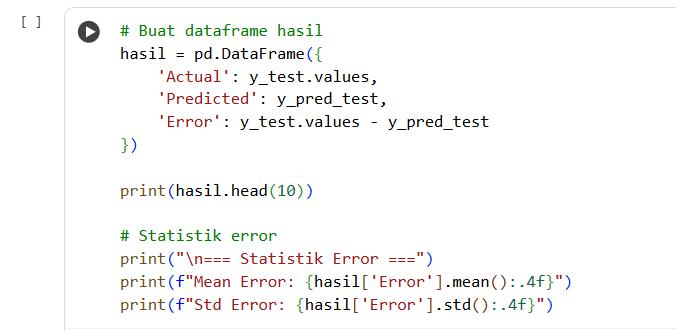




* Membuat objek model LinearRegression dari Scikit-learn.
* Melatih model menggunakan data training dengan .fit().
* Menampilkan koefisien regresi dan intercept.
* Melakukan prediksi pada data training dan testing.
* Mengevaluasi model dengan metrik R² score, MAE, MSE, dan RMSE.

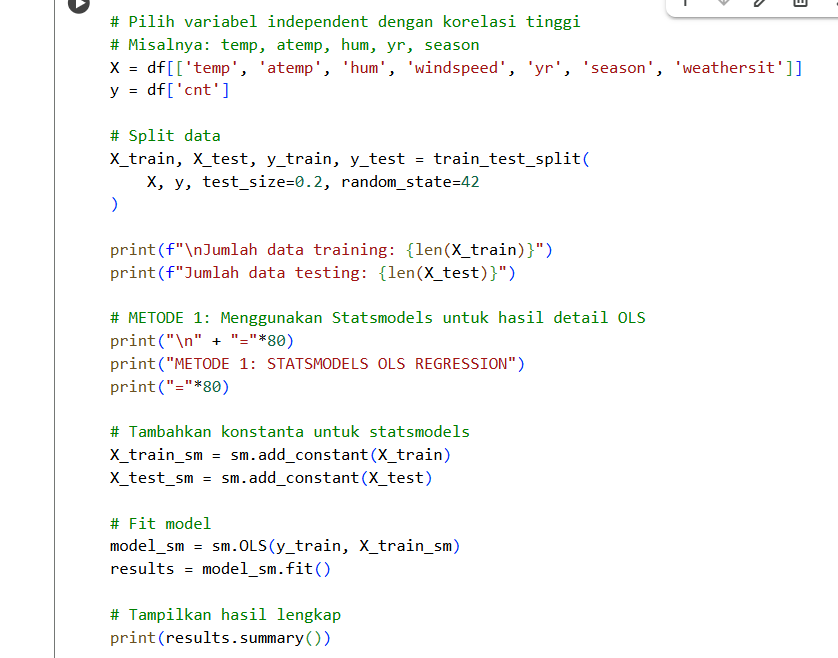
1.8 Visualisasi Hasil Prediksi (Scikit-learn)





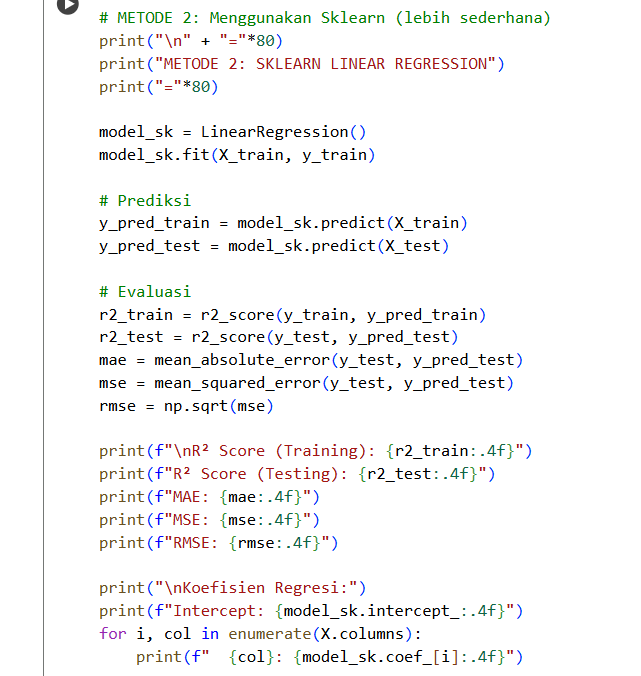
* Membuat scatter plot actual vs predicted untuk data testing.
* Membuat residual plot untuk memeriksa distribusi error.
* Membuat DataFrame hasil prediksi dengan kolom actual, predicted, dan error untuk sampel verifikasi.

1.9 Pembuatan Model Regresi Linear Menggunakan Statsmodels



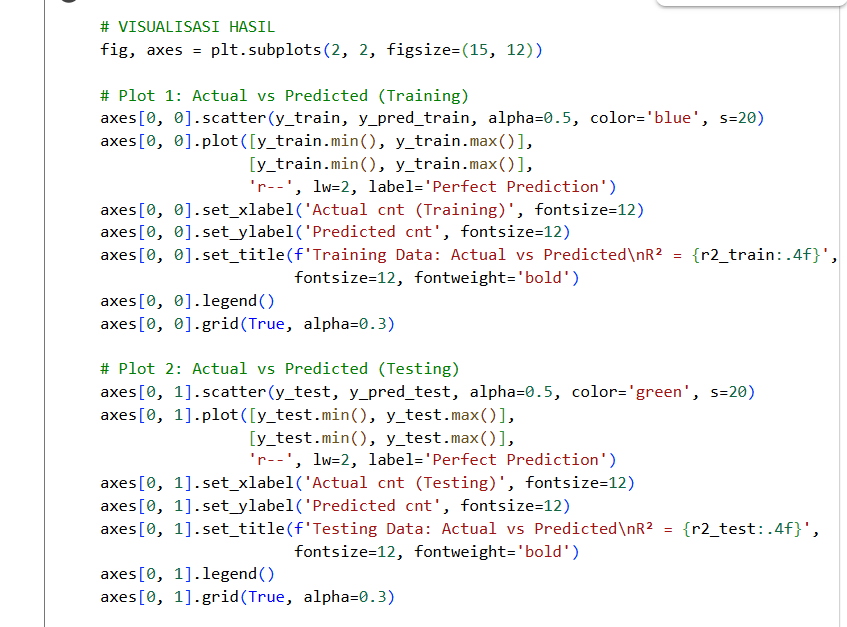
* Memilih variabel independen yang lebih spesifik berdasarkan korelasi.
* Menambahkan konstanta untuk model OLS.
* Melatih model menggunakan sm.OLS dan .fit().
* Menampilkan summary hasil OLS yang detail, termasuk p-value dan statistik lainnya.

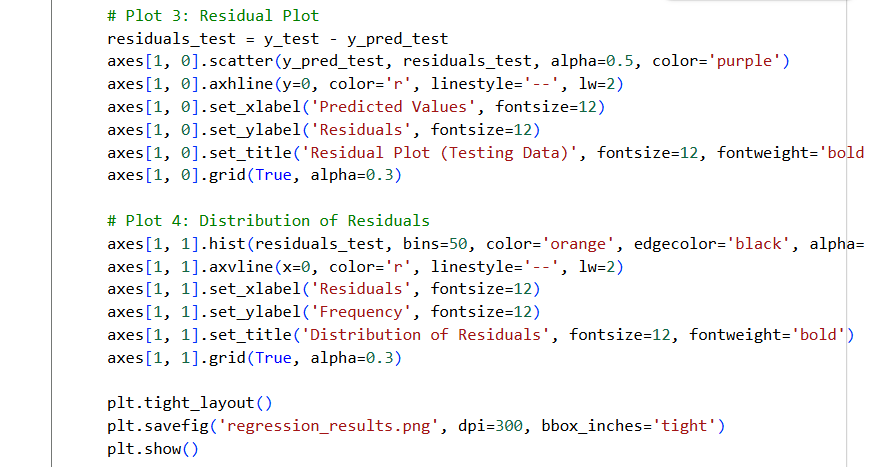
1.10 Pembuatan Model Regresi Linear Menggunakan Scikit-Learn (Versi Kedua) dan Evaluasi

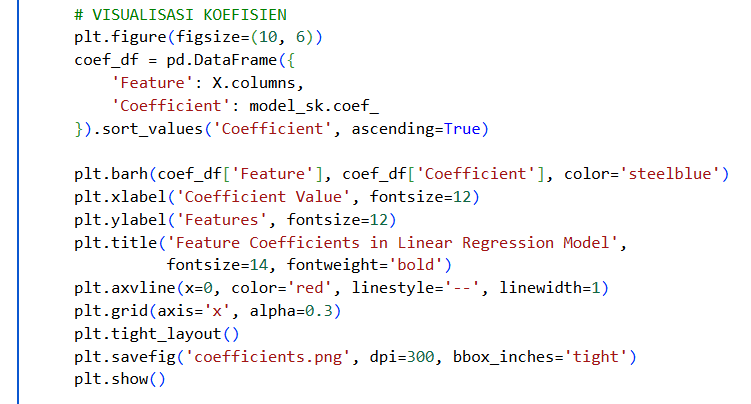


* Membuat model LinearRegression kedua dengan variabel yang disesuaikan.
* Melatih dan memprediksi.
* Mengevaluasi dengan R², MAE, MSE, RMSE.
* Menampilkan koefisien regresi untuk interpretasi pengaruh variabel.

1.11 Visualisasi Hasil Lengkap

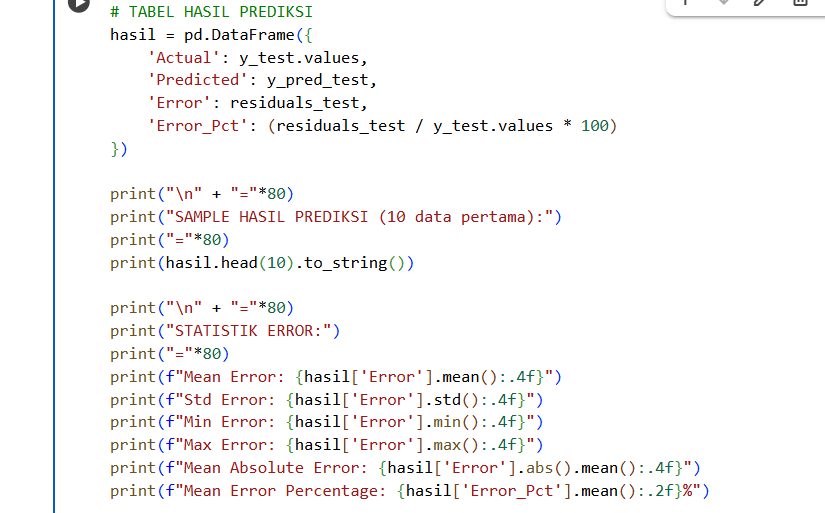


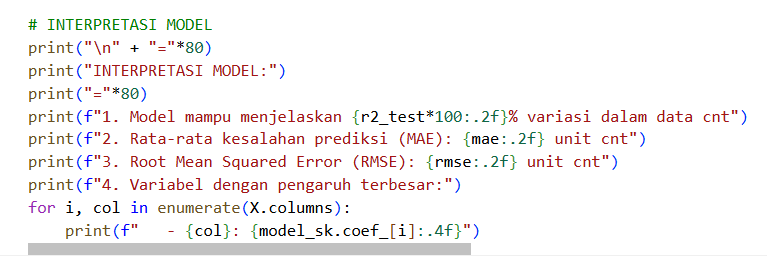




* Membuat subplot untuk actual vs predicted (training dan testing), residual plot, dan distribusi residuals.
* Menyimpan gambar sebagai 'regression\_results.png'.
* Membuat bar plot koefisien fitur untuk melihat pengaruh variabel.
* Menyimpan gambar sebagai 'coefficients.png'.

1.12 Menampilkan Hasil Prediksi dan Interpretasi Model





* Membuat DataFrame hasil prediksi dengan tambahan error percentage.
* Menampilkan sampel 10 data pertama.
* Menampilkan statistik error seperti mean, std, min, max, dan MAE.
* Memberikan interpretasi model berdasarkan metrik evaluasi dan koefisien.

Referensi:

Munir, S., Seminar, K. B., Sudradjat, Sukoco, H., & Buono, A. (2022). The Use of Random Forest Regression for Estimating Leaf Nitrogen Content of Oil Palm Based on Sentinel 1-A Imagery. *Information*, *14*(1), 10. https://doi.org/10.3390/info14010010

Seminar, K. B., Imantho, H., Sudradjat, Yahya, S., Munir, S., Kaliana, I., Mei Haryadi, F., Noor Baroroh, A., Supriyanto, Handoyo, G. C., Kurnia Wijayanto, A., Ijang Wahyudin, C., Liyantono, Budiman, R., Bakir Pasaman, A., Rusiawan, D., & Sulastri. (2024). PreciPalm: An Intelligent System for Calculating Macronutrient Status and Fertilizer Recommendations for Oil Palm on Mineral Soils Based on a Precision Agriculture Approach. *Scientific World Journal*, *2024*(1). https://doi.org/10.1155/2024/1788726