

MDAM Code Explanation

A. Modeling

1. Refer to the “**MULTIFUNCTION MODELLING**” file.
The model requires the following supporting functions: F1, CC, Dkrn, Fitting, U.
These are automatically called within the main script.
2. Line 9: Reads “**DATA.xlsx**.”
 - Row 1 – Years (1993–2022)
 - Row 2 – GHG emission data (target)
 - Other rows – Independent variables (Fossil, Renewables, Electricity Demand)
3. Line 21: Fractional order (α) is initially set as $\alpha_{\text{manual}} = 0.89$ and later optimized based on the minimum error rate achieved during fitting.
4. Lines 24–26: Main parameters are defined and optimized according to the lowest Error rate value obtained in model trials:
 $M = 7$ (polynomial degree), $l = 10$ (memory length), mm (number of variables).
5. Lines 30–36: The F1 function calculates polynomial coefficients (a_0 – a_7) for each independent variable and saves them in “**alfar.dat**.”
These coefficients are determined iteratively to minimize the fitting error.
6. Line 38: The Fitting function performs model optimization by calculating:
 v (optimal fractional order, $0 < v \leq 1$) — selected where the error rate is minimal
7. Line 41: Both v and error results are recorded in “**alfa.dat**.”
8. Lines 43–63: The Dkrn function checks the Caputo fractional kernel and updates the **DDD_all** matrix, ensuring no NaN/Inf values occur during computations.
9. Line 67: Displays the first five values of **DDD_all** for control and validation.
10. CC, Dkrn, and U serve as supporting functions for F1 and Fitting:
 - CC – computes fractional coefficients for polynomial expansion
 - Dkrn – evaluates the Caputo fractional kernel
 - U – defines the unit step function
11. Running the script prints the optimized modeling results in the console:
 - Optimal fractional order (v , at minimum error)
 - Modeled values $\hat{f}(x)$

B. Impact Factor

1. Open the “**Impact_Factor**” file to calculate the impact factor (A_{Ir}) of each variable.

2. m = number of drivers (3 in this study); $N_{out} = 1$ defines GHG as the target variable.
3. $l = 31$ sets the memory length (adjustable for sensitivity tests).
4. Matrices A and C are built from cross-products of driver and target data.
5. Coefficients are solved by $B = \text{pinv}(A) * C$.
6. The average lag weights ($\text{impact} = \text{mean}(\alpha, 2)$) provide the influence direction and strength of each factor on GHG emissions.
7. Results are printed in the console.

C. Prediction

1. Refer to the “**Prediction**” file, which estimates future GHG emissions using the optimized MDAM parameters.
2. The file reads “DATA.xlsx”, separating target (GHG) and drivers.
3. $N_{out} = 1$ selects GHG as the prediction target.
4. `predstep` defines the forecast horizon:
 - 1 → one year ahead
 - 2 → two years ahead
5. For each lag length ($l = 2-9$), the script builds A and C matrices and solves $\alpha = \text{pinv}(A)*C$.
6. Each configuration's error rate is evaluated, and the one with the minimum prediction error is selected.
7. The optimal coefficient vector (`optimalalfa`) corresponding to this minimum error is displayed as the best parameter set for forecasting.
8. Future GHG values can be calculated by multiplying `optimalalfa` with the most recent driver data.