

```

% This code will calculate the fractal dimension of an image from some
% given data.
%
% Using ImageJ and the nested square method of calculating fractal
% dimension, the log of the number of pixels that represent the
% aerosol and
% the log of the increasing box lengths has been exported to
% a text file.
% They are arrays of the same size that have been combined into one
% array.
% This code will graph that data (box length is the x-axis and pixel
% count
% is the y-axis) and calculate the slope of the linear regression of
% that
% data which is the fractal dimension.
%
% Input: None
%
% Output: None
%
% Created on July 12th, 2020 CE by Timothy Honablew

%% Importing the Data and Separating It

data = importdata('imageData.txt');
pixelCount = zeros(length(data)/2, 1);
boxLength = zeros(length(data)/2, 1);

for ii = 1:length(data)/2
    pixelCount(ii) = data(ii);
    boxLength(ii) = data(ii + length(data)/2);
end

%% Calculating the Slope of the Linear Regression

% From the curve fitting toolbox (comes with MATLAB but must be
% manually
% installed).

Df = fit(boxLength, pixelCount, 'poly1', 'Exclude', boxLength > 6.5);

% , 'Exclude', boxLength > 6

disp(Df);

%% Graphing the Data and the Linear Regression

hFig = figure();
set(hFig, 'units', 'pixels', 'Position', [10 10 500 255]);

```

```

hold on;
box on;
grid on;

dataPlot = plot(boxLength, pixelCount, 'o', 'Color', 'r');
linearRegressionPlot = plot(boxLength, Df(boxLength), 'Color', 'k');

title('Nested Square Fractal Dimension Calculation');
xlabel('Log of Box Length');
ylabel('Log of Pixel Count');
xlim([min(boxLength) - 1 max(boxLength) + 1]);
ylim([min(pixelCount) - 1 max(pixelCount) + 2]);

leg = legend([dataPlot, linearRegressionPlot],...
    {'Data', 'Linear Regression'},...
    'interpreter', 'latex', 'location', 'northwest');

set(leg, 'FontSize', 12.5);

save2pdf('NestedSquare.pdf', hFig, 600);

```