Input:
Timeseries (vector)
qValues (vector index i)
scales (vector index j)

Example:
[1 2 1 5 3 2 9 1]
[-3 -2 -1 0 1 2 3]
[1 2 3]

Loop through i, j until all qValues and scales are used

For each qValue(i) and scales(j) calculate matricies **Ma**(i,j), **Mf**(i,j), **Md**(i,j)
Algorithm shown separately

$muScales(j) = -\log_{10}(2^{scales(j)})$

Regress the rows of **Ma**(i,:), **Mf**(i,:), **Md**(i,:) against muScales and obtain slopes α(i), f(i), D(i) for each row

Output:
α, f
$D=\frac{d}{1-qValues}$

Example:
α = [1.49 1.45 1.35 1.14 0.86 0.65 0.56]
f =[0.66 0.77 0.91 1 0.86 0.56 0.33]
D = [1.28 1.22 1.13 1 0.86 0.74 0.67]

%for the i-th qValue and j-th scale:	Example for i=1, j=1:
window=2^scales(j); q=qValues(i);	window=2^1; q=-3;
%break the time series into windows & sum TimeseriesReshaped=reshape(Timeseries,[],window); TimeseriesSummed=sum(Timeseries);	[1 2 1 5; 3 2 9 1]' 1+2+1+5+3+2+9+1 = 24

%calculate p
ps=sum(TimeseriesReshaped);
p=ps./TimeseriesSummed;
	[1+2+1+5=9 3+2+9+1=15]
	[9/24=0.375 15/24=0.625]

Nor=sum(p.^q); mu=(p.^q)/Nor;	0.375^-3+0.625^-3 = 23.059 [0.375^-3/23.059 = 0.8224 0.625^-3/23.059 = 0.1776]
Md(i,j)=log10(Nor); **Ma**(i,j)=sum(mu.*log10(p)); **Mf**(i,j)=sum(mu.*log10(mu));	log10(23.059) = 1.3628; -0.3503 + -0.0363 = -0.3866 -0.0698 + -0.1333 = -0.2032