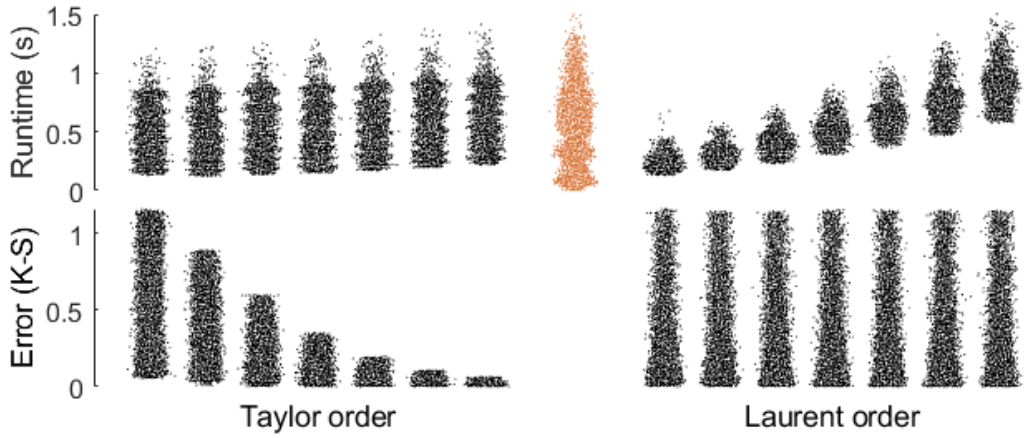
```
function benchmark_visualizer
clear;clc;close all;
D =load('data/gill_syn_data_1.mat');

analyt_col = [217 116 55]/255;

%initialize arrays to receive the data
I_ = 7; J_ = 7;
[I__,J__] = ndgrid(1:I_,1:J_,1:(50*50)); I__ = I__(:); J__ = J__(:);
[D__, T_a__, T_n__] = deal(NaN(I_, J_, 50*50));

for i = 1:I_
    for j = 1:J_
        %import the results of each scan
        F = load(sprintf('landscape/gg_200131_land_scan_1_%i_
%i.mat',i,j));

        %store them in a single array
        T_n__(i,j,:) = F.T_numint;
        T_a__(i,j,:) = F.T_analytint;
        %this selects the KS error; EMD and KL can be used instead
        D__(i,j,:) = F.div_ks;
    end
end

D__ = D__(:);
T_a__ = T_a__(:);
T_n__ = T_n__(:);


%initialize axes
figure(1);
top = 0.04;
bottom = 0.11;
left = 0.1; right = 0.04;
mid_h = 0.06; mid_v = 0.04;
width = (1-left-right-mid_h)/2;
height = (1-top-bottom-mid_v)/2;

COE_1=0.1;
COE_2 = 1-COE_1*2;

ax_1_pos = [left, bottom+height+mid_v, width, height];
ax_2_pos = [left+width+mid_h, bottom+height+mid_v, width, height];
ax_3_pos = [left, bottom, width, height];
ax_4_pos = [left+width+mid_h, bottom, width, height];
ax_5_pos = [left+width+mid_h*COE_1, bottom+height+mid_v,...
    mid_h*COE_2, height];

%%%%%%%%%%
% plot Taylor modulation runtime
axes('Position',ax_1_pos);
```

```
jit = randn(size(I__))/10;
scatter(I__+jit, T_a__, 1,'k','filled');
ylim([0, 1.5]);
ylabel('Runtime (s)');
h=get(gca);
set(gca,'xtick',[],'xcolor','none','color','none','FontSize',12)

%%%%%%%%%%
% plot Laurent modulation runtime
axes('Position',ax_2_pos);
jit = randn(size(J__))/10;
ylim([0, 1.5]);
scatter(J__+jit, T_a__, 1,'k','filled');
set(gca,'xtick',[],'ytick',
[],'ycolor','none','FontSize',12,'visible','off')

%%%%%%%%%%
% plot Taylor modulation KS error
axes('Position',ax_3_pos);
jit = randn(size(I__))/10;
scatter(I__+jit, D__, 1,'k','filled');
xlabel('Taylor order');
ylabel('Error (K-S)');

h=get(gca);
set(gca,'xtick',[],'xcolor','none','FontSize',12,'color','none')
h.XAxis.Label.Color=[0 0 0];
h.XAxis.Label.Visible='on';
h.YAxis.Label.Color=[0 0 0];
h.YAxis.Label.Visible='on';
yl=ylim;

%%%%%%%%%%
% plot Laurent modulation KS error
axes('Position',ax_4_pos);
jit = randn(size(J__))/10;
scatter(J__+jit, D__, 1,'k','filled');
xlabel('Laurent order');
h=get(gca);
set(gca,'xtick',[],'ytick',
[],'ycolor','none','FontSize',12,'visible','off')
h.XAxis.Label.Color=[0 0 0];
h.XAxis.Label.Visible='on';
ylim(yl);

%%%%%%%%%%
% plot numerical integration runtime
axes('Position',ax_5_pos);
jit = randn(size(I__))/10;
ylim([0, 1.5]);
scatter(jit, T_n__, 1,analyt_col,'filled');
h=get(gca);
set(gca,'xtick',[],'ytick',
[],'ycolor','none','FontSize',12,'visible','off')
```

```
set(gcf,'Position',[531.4000 390.6000 674.4000 264]);

return
```



Published with MATLAB® R2019b