

fmtatfun用户手册

2018 年 10 月 15 日

目录

1	简介	2
1.1	组成	2
1.2	安装	2
2	使用指南	2
2.1	作图类函数	2
2.1.1	多条曲线绘制与图像放大	2
2.1.2	图像字体大小设置	4
2.1.3	已打开figure的批量操作	4
2.1.4	对同一目录figure文件的批量操作	5
2.1.5	图像文件数据恢复	5

1 简介

fmatfun 是一个灵活易用的Matlab函数集，通过将日常使用的功能进行封装形成，涉及到图像、数据、文本、文件等，编写目的是提高使用Matlab和生成科技报告写作材料的效率。

fmatfun中的大多数功能只需要一到几行代码便可实现。fmatfun各模块根据功能组织，可重用性高，基于fmatfun的二次开发也相当便捷。

1.1 组成

目前，可使用的函数有：

1. matfig

matfig包含一系列用于作图，修改图像的函数，覆盖基本的多条曲线绘制、图像字体大小重设到图像的标记放大、批量化设置等。

2. demo

demo是fmatfun使用范例，从这里可以找到各个函数的用法示例。

1.2 安装

1. 下载 fmatfun

2. 在Matlab中运行install.m

2 使用指南

2.1 作图类函数

2.1.1 多条曲线绘制与图像放大

一般绘制多条曲线时，基本的要求是能够区分不同的曲线，这就要对曲线使用不同的线型、标记、颜色、线宽，且设置不同的图例。若曲线的点数太多，则设置标记后的密度太高，曲线的清晰度、可读性将会非常差。

使用三层曲线绘制方法可解决该问题，第一层为只有线型、颜色、线宽的曲线，第二层为稀疏的标记，线型为空，第三层为单点曲线，仅用于显示图例。plotfig为对应的函数，Code 1给出了三种不同参数设置的调用示例。

Code 1: plotfig调用代码(demo/matfig/multilines.m)

```

1
2 [fig_rndlines,~,~,plines] = plotfig(x,y,x,y); % 注意x,y必须有一
   维维度相同
3
4 % 1/15 points will be marked; 抽取1/15的点打上标记
5 [fig_rndlines,~,~,plines] = ...
   plotfig(x,y,x(1:15:end),y(:,1:15:end));
6
7 % Change the line type, marker,color, width; 更改线型、标记、颜色、
   线宽
8 [fig_rndlines,~,~,plines] = ...
   plotfig(x,y,[],[],{'-','-','-'},{'none','o','*'},{[0.0 ...
   0.447 0.741],[0.85 0.325 0.098],[0 0 0]},{1.0});

```

效果:

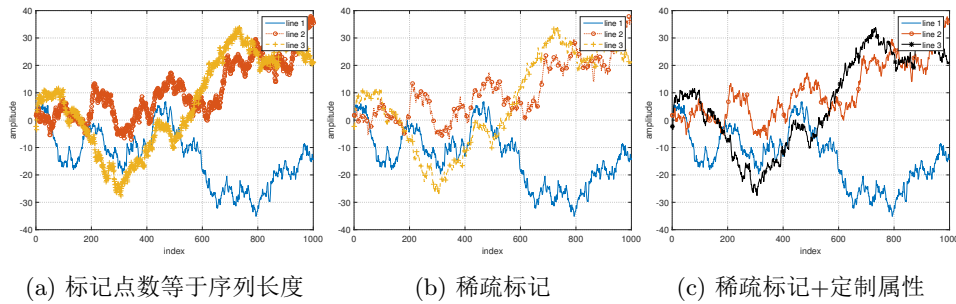


图 1: plotfig三种不同参数设置

图像放大使用copy_and_zoom_ax函数，基本思路是采用工具箱自带函数copyobj复制图像，指定图像在中的位置，设置Axes的XLim, YLim属性实现局部区域放大，最后在原图像上进行标注。

Code 2: copy_and_zoom_ax调用代码(demo/matfig/multilines.m)

```

1 % Zoom in and plot in the same figure; 放大并显示在同一幅图上
2 copy_and_zoom_ax(fig_rndlines,[],[0.2 0.6 0.35 0.30],[400 ...
   600],[200 300],'rectangle','-');
3 % Zoom in and plot in a new figure; 放大并显示在新的图上
4 fig_rndlines_new = figure;
5 copy_and_zoom_ax(fig_rndlines,fig_rndlines_new,[],[400 ...
   600],[200 300],'rectangle','-');

```

效果:

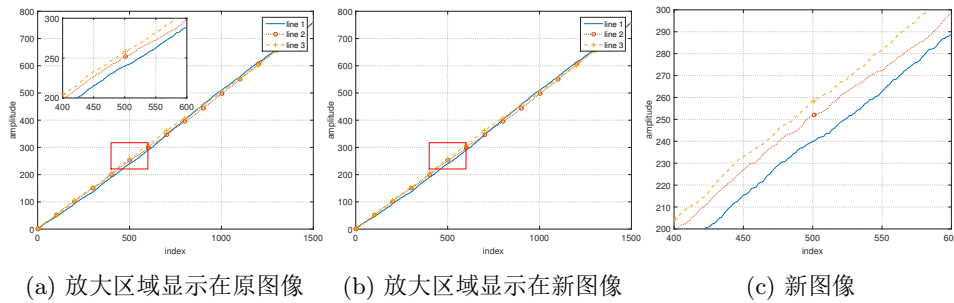


图 2: copy_and_zoom_ax

2.1.2 图像字体大小设置

`set_fig_fontsize`通过图像句柄实现对坐标轴字体、图例字体大小进行设置，只需调用一行代码`set_fig_fontsize(fig, font_size)`，其中`fig`为图像句柄，`font_size`为字体大小。无需分别单独对各个坐标轴、图例等进行字体大小设置。结合后续的图像批量处理函数可进一步提高效率。

2.1.3 已打开figure的批量操作

使用Matlab过程中有时为了简化代码编写或调试需要，经常只是简单地把结果进行绘图，等到写报告却发现图像不太美观，需要重新设置。figure的批量操作针对该场景。

Matlab的所有figure都对应一个窗口，窗口在系统中有唯一的句柄，通过`findall`函数可得到所有已打开figure的句柄，继而对它们的属性进行设置。

目前，可通过`set_openfig_fontsize`对figure字体大小进行批量设置，通过`save_openfig`对figure进行批量保存。

Code 3: 已打开图像批量操作(demo/matfig/openfigop.m)

```
1  multilines;
2  % save_openfig('fig\','fig','fig');
3  set_openfig_fontsize(14);
4  save_openfig('..\..\doc\demo\','mlines','eps');

```

2.1.4 对同一目录figure文件的批量操作

`reset_fig_font_size_batch(input_file_path,output_file_path, ... save_opt,font_size)` 同一目录figure文件进行读取，并重新设置大小，最后再输出到文件中。其中

2.1.5 图像文件数据恢复

Matlab的figure文件/句柄存储了大量信息，包括图像的外观参数及数据，这也是figure文件比其他类型文件占用空间大的原因。带来的好处之一便是我们可以从figure文件中恢复出曲线/图像对应的数据。

如果figure是对某一次实验数据或中间结果的绘制而我们又恰好想对这些数据/结果进行进一步处理，那么完全可从figure中恢复数据接着处理。适用于程序丢失或程序重新运行时间太长且未将重要数据保存成.mat文件的情况。

`fig2mat_line`、`fig2mat_image`分别实现从figure文件中提取曲线及二维图像的值。

Code 4: 图像文件数据恢复(demo/matfig/figfile_data.m)

```

1  %% fetch the values of lines from the .fig file; 从.fig图中
   提取曲线的值
2  y = cumsum(randn(1,1000));
3  fig_line = figure; plot(1:1000,y);
4  saveas(fig_line,'rline.fig','fig');
5  [line_data] = fig2mat_line('rline.fig');
6  %% fetch the values of image from the .fig file; 从 figure
   图中提取二维图像的值
7  img = randn(500,500);
8  fig_image = figure; imagesc(abs(img));
9  saveas(fig_image,'rimg.fig','fig');
10 [Xdata, Ydata, Cdata]= fig2mat_image('rimg.fig');
```

`fig2mat_line`的输出为元包数组，每一元素为结构体，对应一个Axes，示例如下所示。

```

1  >> line_data{1}
2
```

```
3      ans =  
4  
5      XData: {[1x1000 double]}  
6      YData: {[1x1000 double]}  
7      ZData: {[1x0 double]}
```