Detection of steps in single molecule data

Tutorial: Under Construction

Tanuj Aggarwal[1], Donatello Materassi[1], Robert Davison[2],
Thomas Hays[2], Murti Salapaka[1,3]

[1]Department of Electrical and Computer Engineering,
University of Minnesota, 200 Union St. SE, Minneapolis, MN 55455
[2]Department of Department of Genetics, Cell Biology and Development,
University of Minnesota, 321 Church St. SE, Minneapolis, MN 55455
[3]To whom correspondence should be addressed. E-mail: murtis@umn.edu

1 Usage

The step fitting package is named as 'stepfit1.m' where '1' will be replaced by the version number for future versions. The command `help stepfit1.m` on MATLAB command interpreter window displays the following usage of this package.

```
USAGE: x=stepfit1(y,'parameter',value)

 x  Step Fit
 y  Data on which to fit steps
Optional Parameters:
 Fs           Sampling time.  Required if tau is being specified
 tau          Time constant of first order dynamics
 outputnoise  Standard deviation of the noise measured at output.
              Estimates noise at the output if unspecified.
              Noise estimation is not reliable under extreme
              dynamical distortion and high stepping speed
 measnoise    Standard deviation of the noise in the measurement system.
              Assumes zero measurement noise if unspecifed
 passes       Number of iterations in histogram refinement
 verbose      verbose=1:  Display progress of histogram.
              verbose=2:  Save the progress in fig files
```

`y` is a vector that stores the time data. It is assumed that the sampling is at a uniform rate. `x` is the variable in which fit will be stored after processing. The length of vector `x` is same as the data, `y`. The program can run without any specified parameters. i.e., `x=stepfit(y)` will run with default settings of the optional parameters. Any number of parameters can be specified in any order. A detailed description of each of the optional parameters is explained in Table 1.

2 Examples

Example 1

```
t=0:1000; % create time vector
steplocation=round(t(end)*rand(1,5)); % create random location for steps
x=zeros(1,length(t)); % create empty step
x(steplocation)=10; % Specify stepsize of 10 units
```

Parameter Name	Description
Fs	Sampling time in seconds. The parameter is not used if unspecified.
tau	The rise-time of first order dynamical model in seconds. For a simple spring-damper system, with stiffness k and damping constant, c, tau=c/k. The parameter is not used if unspecified.
outpunoise	This is the net standard deviation of the noise in the measured data, y. The function estimates this number if not provided by the user.
measnoise	This is the standard deviation of the noise due to instrumentation only. Default value is 0 unless specified. It is recommended to keep this zero if tau is not specified. Include all noise contributions in the outputnoise parameter instead.
passes	This is the number of iterations for which the function will refine its step-histogram and step-fit estimates. Default value is 10
verbose	This parameter takes values 0,1, or 2. Default is 0 and corresponds to minimal display of information about the progress of the function. verbose=1 will plot step fit and histogram estimates at the end of each iteration. It will also show progress of each iteration through textual status bar on the command window. verbose=2 will save the plots in files with name 'StepFitIterationProgress *.fig' with * corresponding to the iteration number.

Table 1: Function Help

```
x=cumsum(x); % Generate stepping signal
y=x+10*randn(1,length(t)); % add gaussian noise of std=10 to the stepping signal
xhat=stepfit(y); % run program without any parameters
p=plot([y;xhat;x]');
legend('Data','Estimate','True','location','best')
axis tight
set(p(1),'color',[.6 .6 .6]);
```

output:

```
Fs=1.00e+000
tau= 0.00e+000
outputnoise=9.36e+000 (Estimate of the noise)
measnoise=0.00e+000
passes= 1.00e+001
verbose=0.00e+000
starting parallel MATLAB sessions...Done
Stepping LLR=0.076
```

Plot output is shown in Figure 1.

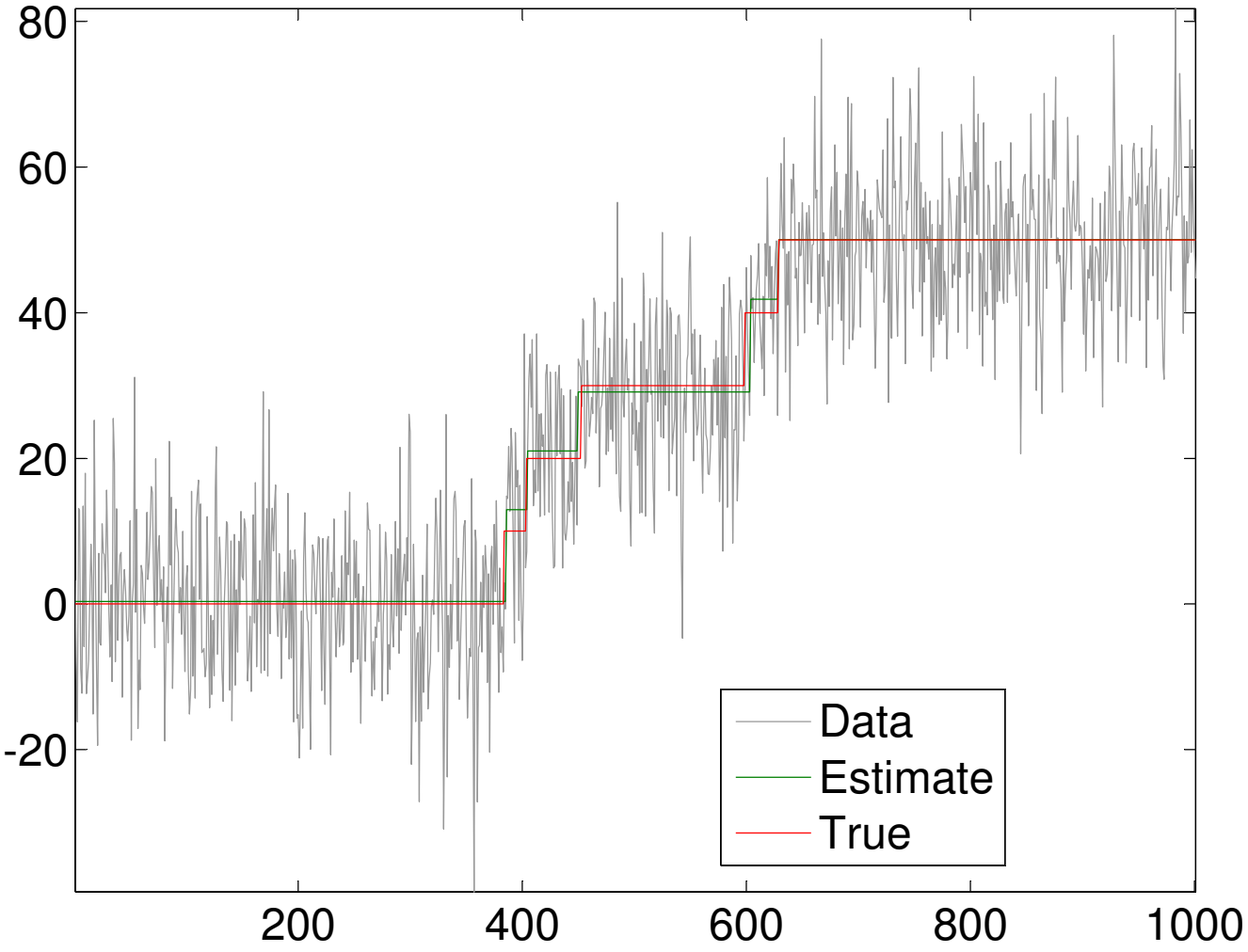

Figure 1: Example 1

The last message ‘‘starting parallel MATLAB sessions...Done’’ appears only if the parallel sessions of the matlab are not open and this is the first time the program is being executed. In subsequent runs, this message will not appear.

Example 2

Same as example 2 except that some parameters are specified. Note that the outputnoise parameter is different from its actual value. The program still works fine.

```
t=0:1000; % create time vector
steplocation=round(t(end)*rand(1,5)); % create random location for steps
x=zeros(1,length(t)); % create empty step
x(steplocation)=10; % Specify stepsize of 10 units
x=cumsum(x); % Generate stepping signal
y=x+10*randn(1,length(t)); % add gaussian noise of std=10 to the stepping signal
xhat=stepfit1(y,'outputnoise',12,'verbose',1); % run program with 2 parameters
p=plot([y;xhat;x]');
legend('Data','Estimate','True','location','best')
axis tight
set(p(1),'color',[.6 .6 .6]);
```

output: (Plot output in Figure 2, histogram updates not shown)

```
Fs=1.00e+000
tau= 0.00e+000
outputnoise=1.20e+001
measnoise=0.00e+000
passes= 1.00e+001
verbose=1.00e+000
b:  [1]
a:  [1 0]
Pass:1, Resolution=1.588, 4 steps
Pass:2, Resolution=1.588, 4 steps
Pass:3, Resolution=1.162, 4 steps
Pass:4, Resolution=0.638, 4 steps
Pass:5, Resolution=0.369, 4 steps
Pass:6, Resolution=0.236, 4 steps
Pass:7, Resolution=0.169, 4 steps
Pass:8, Resolution=0.138, 4 steps
Pass:9, Resolution=0.132, 4 steps
Pass:10, Resolution=0.130, 4 steps
Stepping LLR=0.045
```

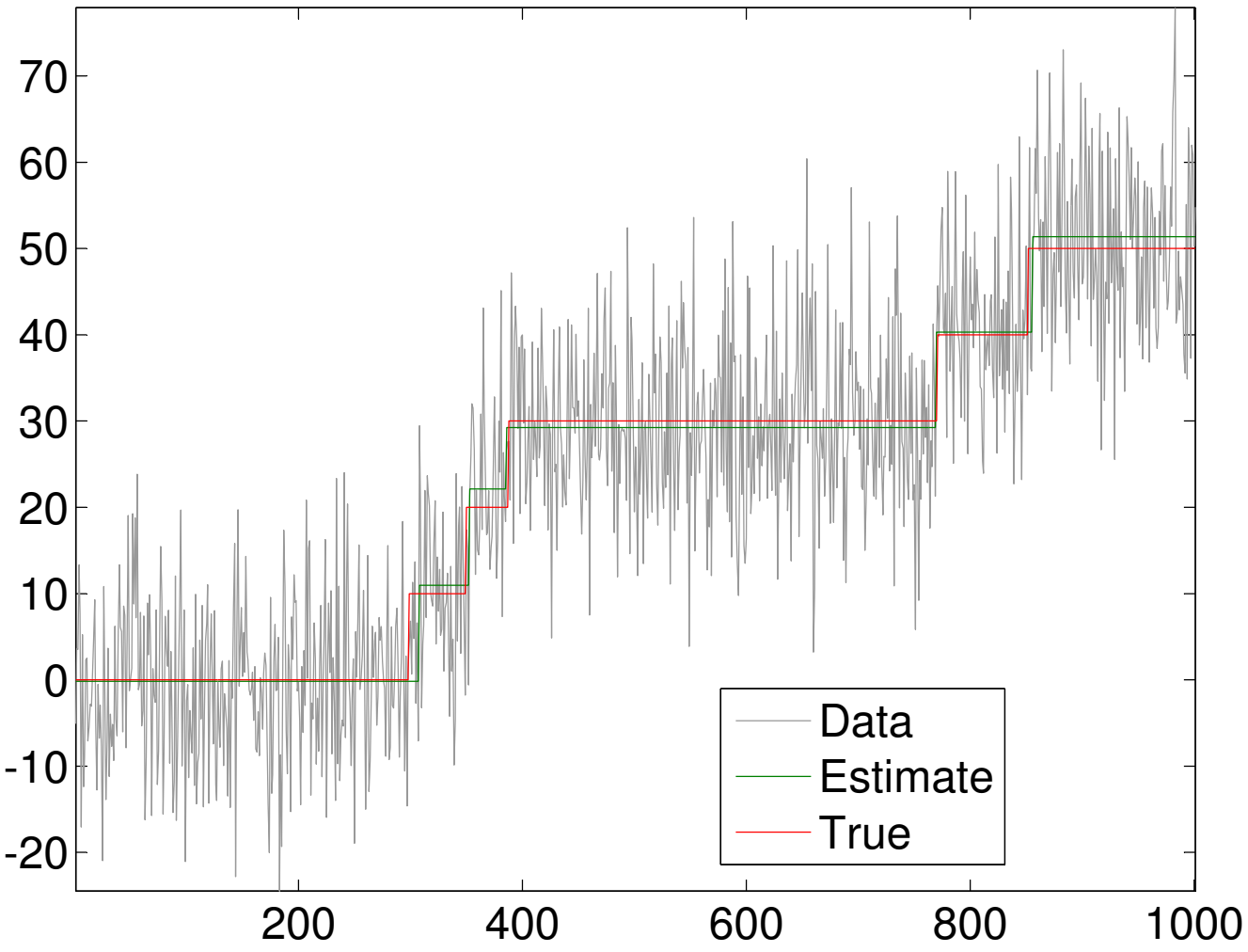

Figure 2: Example 2