Error Analysis

E. F. Thacher

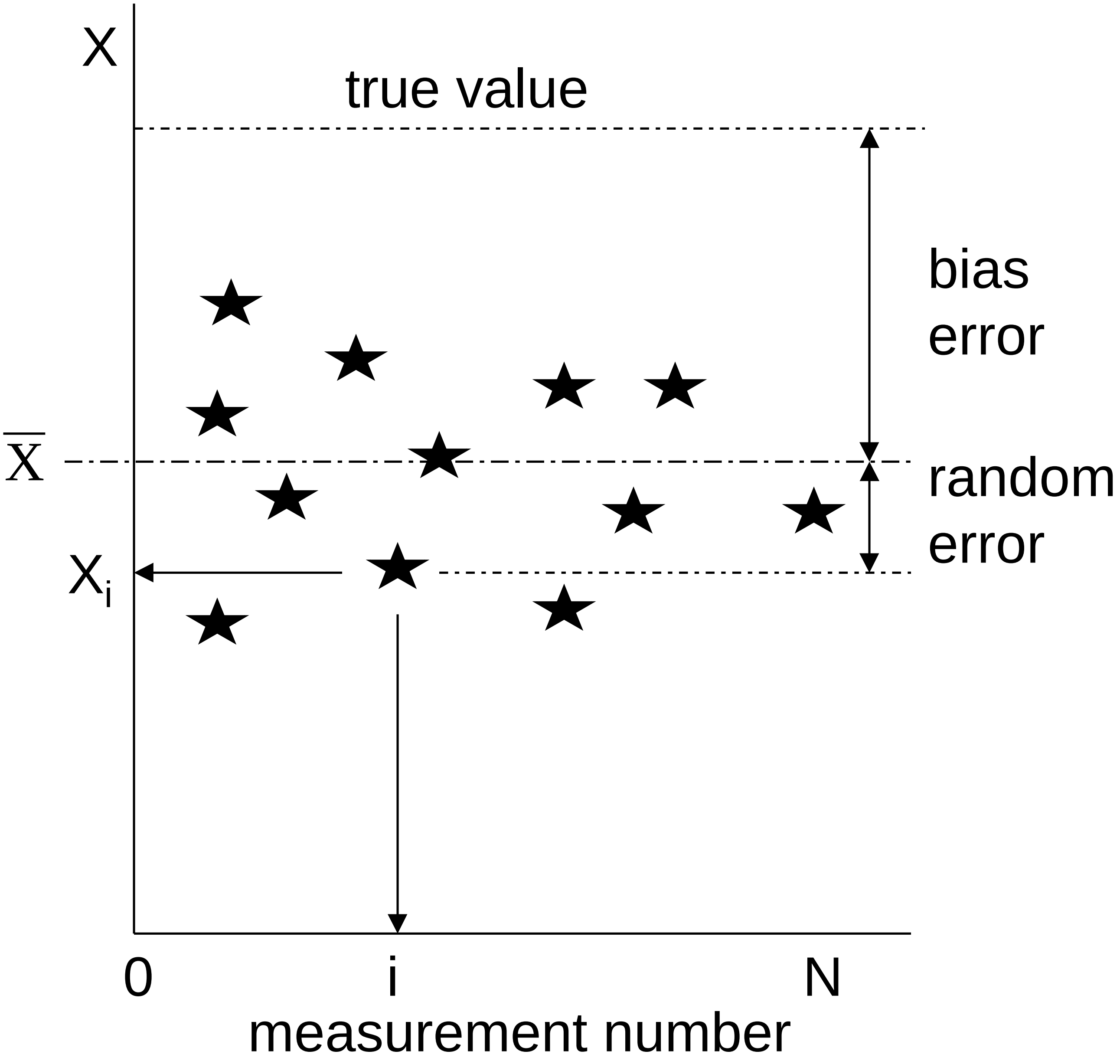
Objectives

- Learn some error analysis
- Demonstrate
  - » Overlay plots
  - » Points-only plotting
  - » Subplots
  - » Histograms
  - » Labeling

Review 1

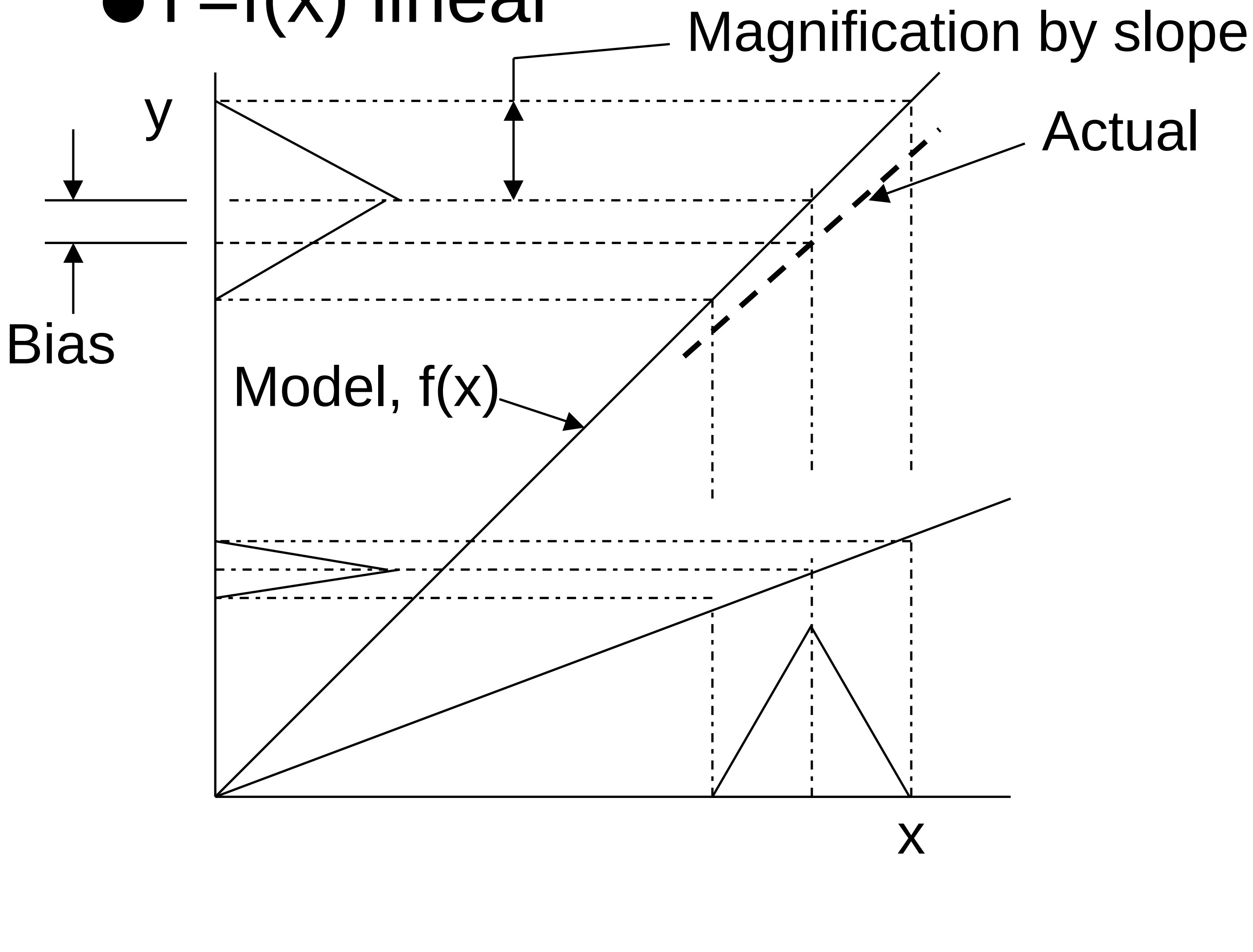
- Random error (scatter)
  - » Random fluctuations in measurement conditions
  - » Noise introduced by signal processing
- Bias error (constant offset)
  - » Poor calibration, laboratory conditions, etc.
  - » Built into model
- Both propagate through model

Review 2


X
true value
bias
error
$\overline{X}$
random
error
X_i
0
i
N
measurement number

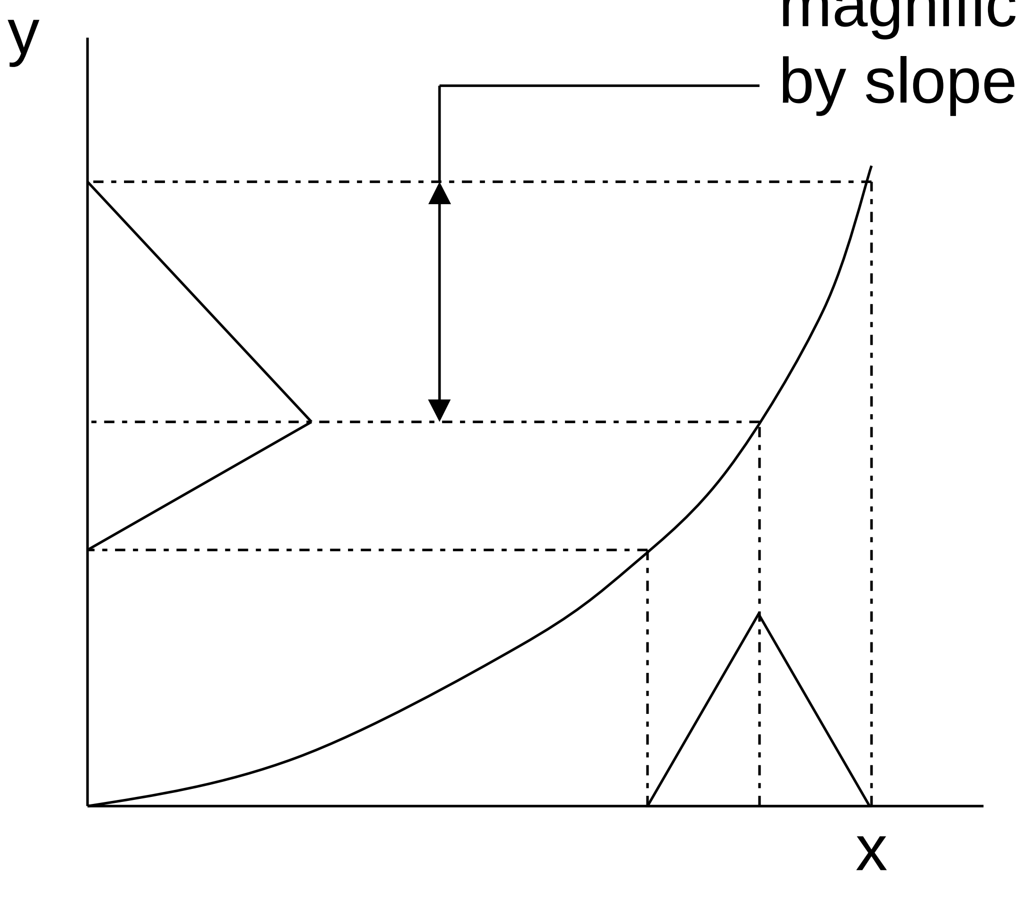
Error Propagation of Linear Model

- Y=f(x) linear



Error Propagation of Non-Linear Model

- Y=f(x) non-linear



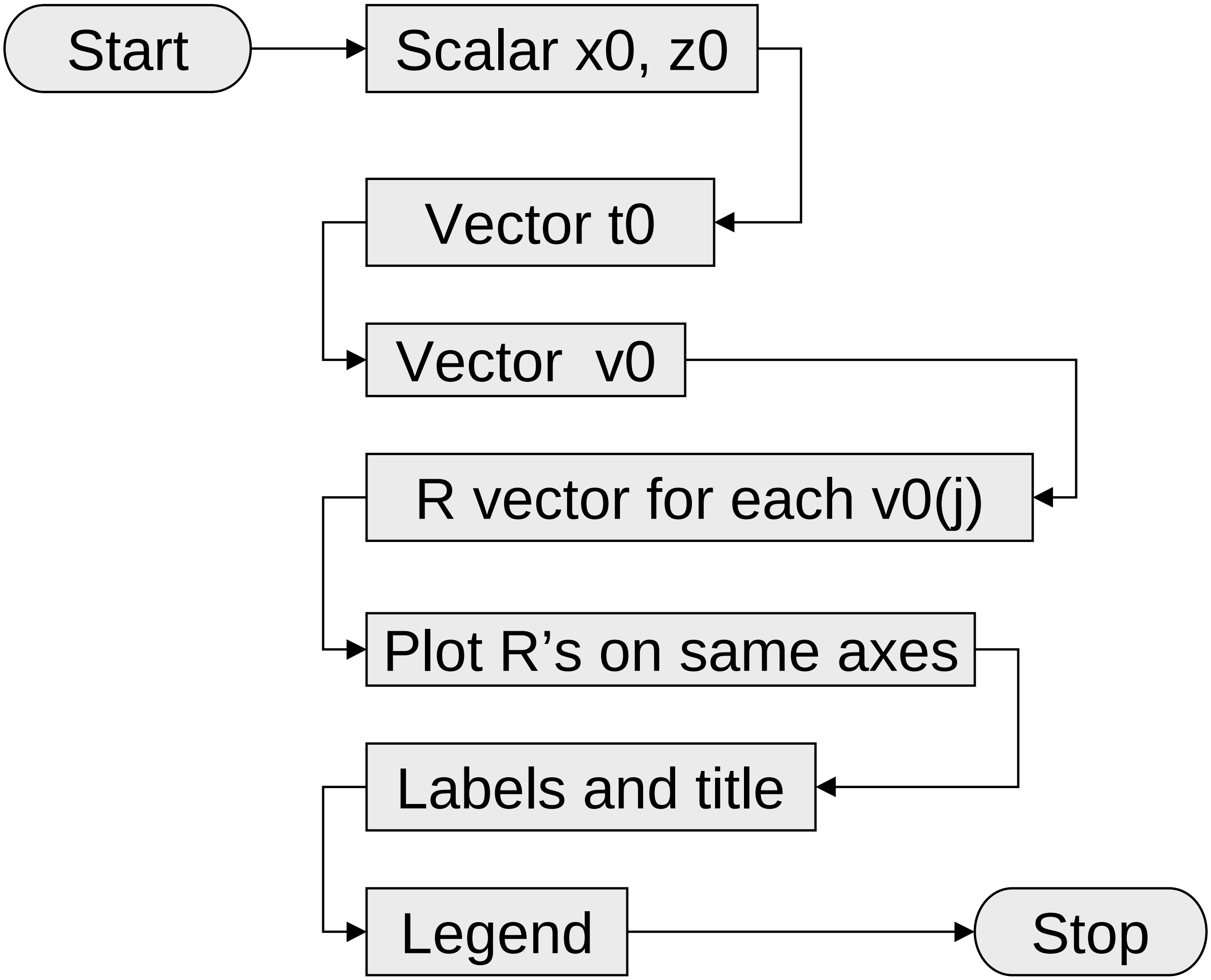
Distortion and magnification by slope

[Bias error not shown]

Program's Task

- Plot range vs. launch angle
- One plot for each of three launch velocities
- All plots on same axes

Flow Chart for OVERLAY1.M



OVERLAY1.M Statements

- X = [start:increment:stop]
- plot (x1, y1, [‘symbol1’], x2, y2, [‘symbol2’], …)
- xlabel, ylabel, title
- R = range0 (v0, t0, x0, z0)
  - » Function to calculate range using vectors v0, t0
- Rule (about to be broken): plot data with symbols, theory with lines

OVERLAY1.M DEMO

- <alt-tab> toggles between MATLAB and PP

OVERLAY2.M

- How do you connect the symbols?

```
plot (t0, R1, t0, R1, ’o’,
      t0, R2, t0, R2, ’x’,
      t0, R3, t0, R3, ’Pentagram’ )
```

You have to give the variable names twice for each overlaid plot.

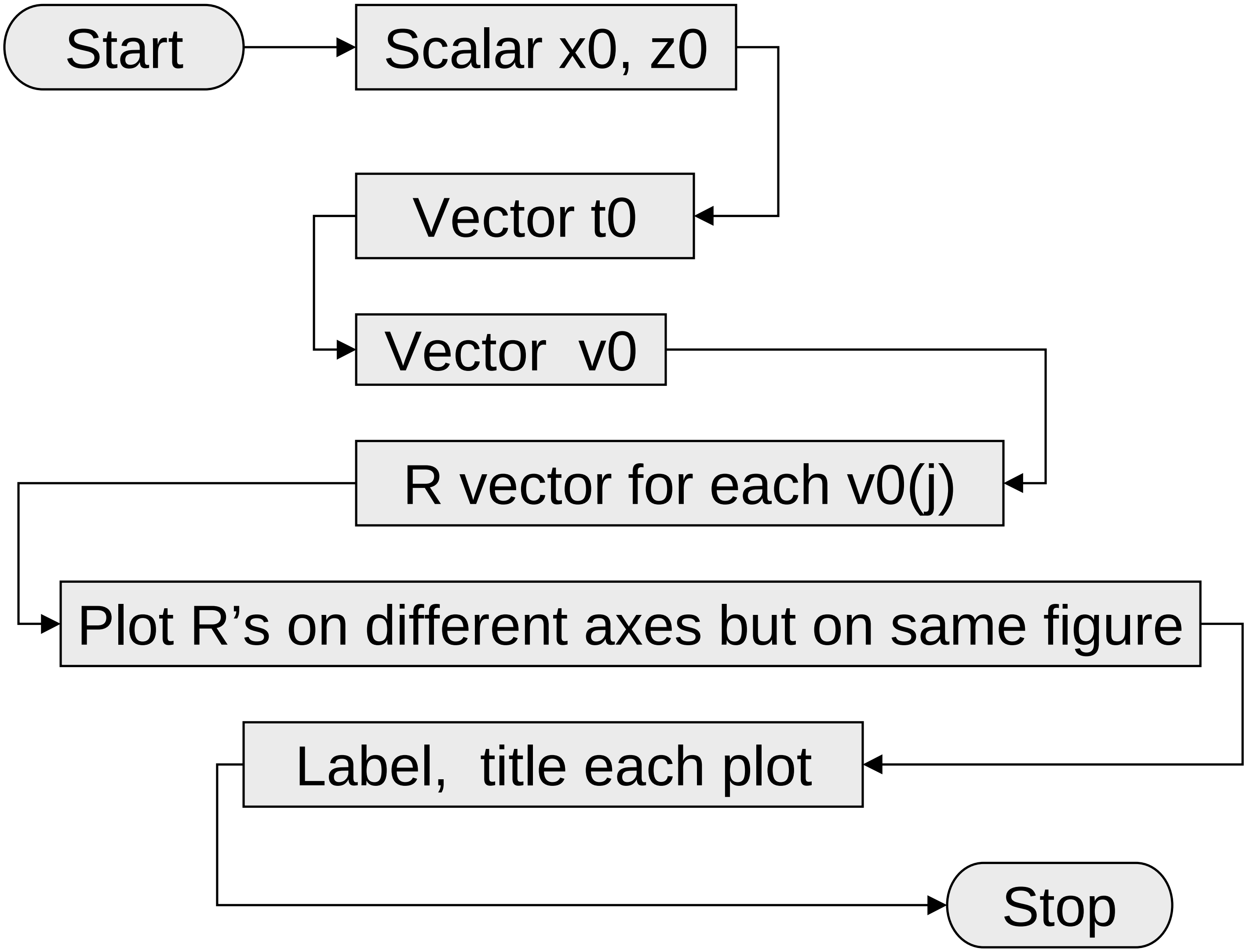
OVERLAY2.M DEMO

- <alt-tab> toggles between MATLAB and PP

Program's Task

- Plot range vs. launch angle.
- One plot for each of three different launch velocities
- Three separate plots on the same figure

Flow Chart form SUBPLT.M



SUBPLT.M
New Statements

The following statements accomplish the flow chart’s objective:

```
subplot (1, 3, 1)
plot (x1, y1, [‘symbol1’] )
xlabel ...
ylabel…
title …

subplot (1, 3, 2 )
…

subplot (1, 3, 3 )
...
```

[Note: x1, y1, symbol1, etc. are generic]

Meaning of “subplot (m,n,p)”

m = # of plot rows; n = # of plot col
p = put next plot in pth position

m = 2, n = 3

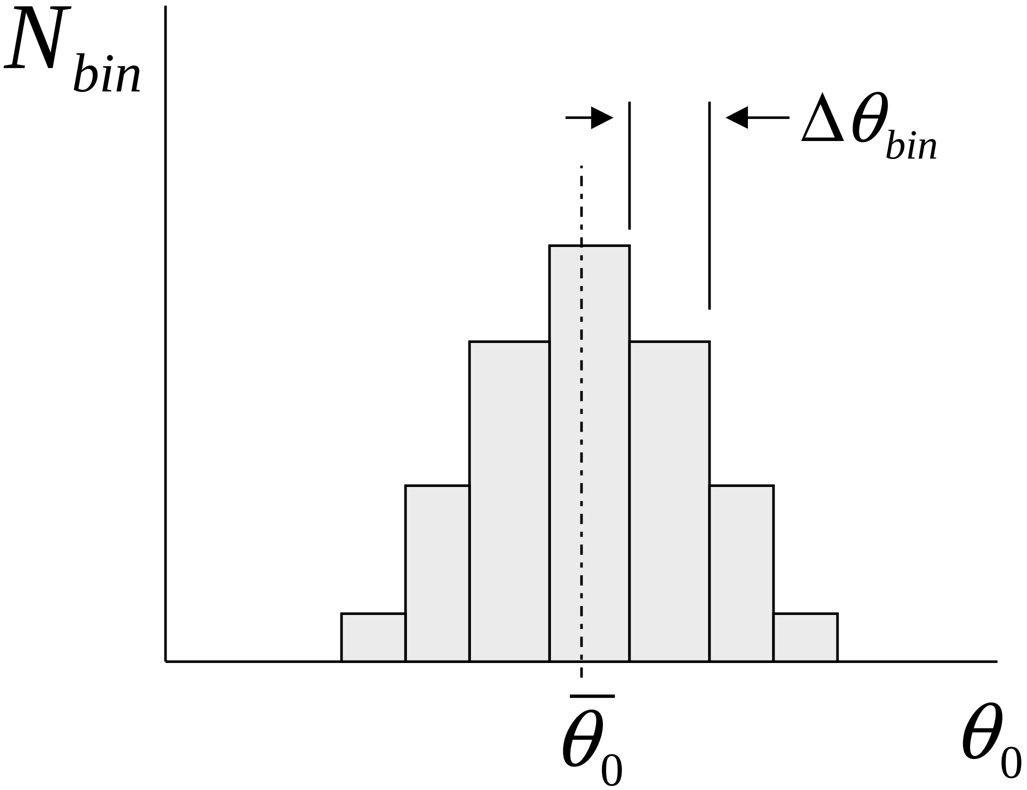
n =	1	2	3
m=1	Plot p=1	Plot p=2	Plot p= 3
m=2	Plot p=4	Plot p=5	Plot p=6

SUBPLT.M Demonstration

- <alt-tab> to toggle between MATLAB and PP.

Histogram Review

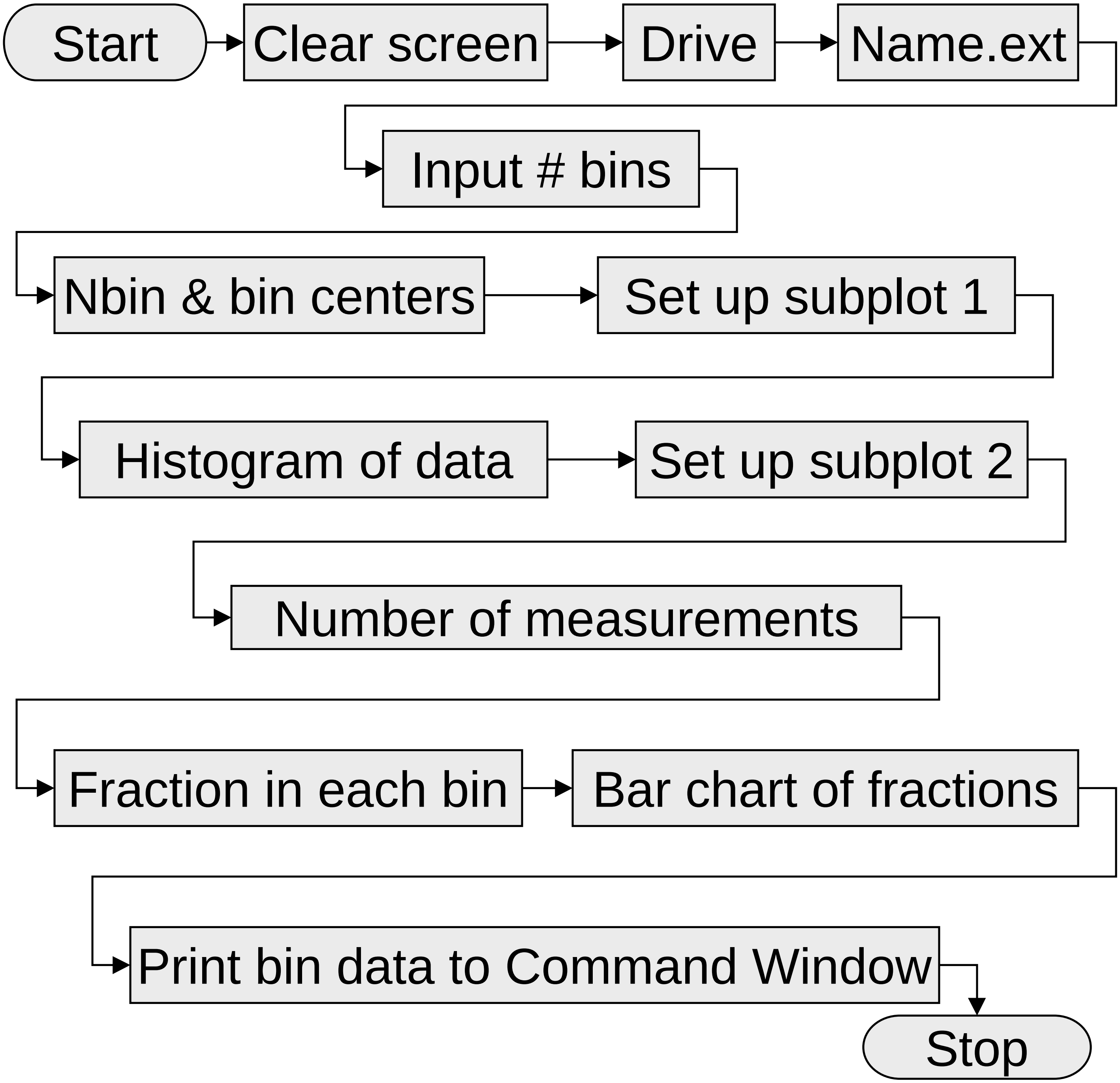
- After N measurements



Program’s Task

- Plot histogram of p_data.txt or p_data32.txt
- Plot bin chart of fraction of measurements in each bin
- Show both plots on same figure

Function HISTPLOT.M Flow Chart



HISTPLOT.M New Statements 1

Input the path and then ...

```
addpath(p);
```

Input the filename.ext, then ...

```
load(fname);
```

Parse into path, name, ext ...

```
[path,name,ext]=fileparts(fname)
```

Load the data into “data” ...

```
data=eval(name);
```

Input the number of bins & call

```
hgramf2 (bins, data);
```

HGRAMF2.M New Statements 1

In function “hgramf2.m (bins, data):”

m ← bins vector ← data

Obtaining number/bin and bin centers:

```
[n, bin_centers] = hist (vector, m);
```

Plotting the histogram:

```
hist (vector, m);
```

Finding the number of measurements and the fractiion in each bin:

```
num_meas = length (vector);
frequency = n/num_meas;
```

HGRAMF2.M New Statements 2

Printing the bin data:

```
fprintf ( ‘\n There were
          %3.0fmeasurements.\n\n',
          num_meas );

disp (' bin Center (psi)
        count frequency')
```

You have to put them in an array.

```
A = [ bin_centers; n; frequency ];
```

Blanks left for orderly appearance.

```
fprintf ('           %4.3f
              %2.0f    %6.4f\n', A)
```

Note: MATLAB takes the transpose of A when printing.

HISTPLOT.M Demonstration

- <alt-tab> to toggle between MATLAB and PP.

!!Extra Credit!!

- On a sheet of paper, describe the differences or similarities between script m-files, function m-files, and MATLAB commands such as plot