LESSON: Histograms

FOCUS QUESTION: How can I understand and compare the distributions of two data sets?

This lesson demonstrates how to create, combine and compare histograms.

In this lesson you will:

- Calculate and display a histogram.
- Use bar charts, line graphs and stair plots to display distributions.
- Observe the characteristics of common distributions.
- Compare histograms and bar charts.

Contents

DATA FOR THIS LESSON

File	Description
	▪ The data set consists of measurements of beak sizes in mm. of one species of Darwin's ground finch (*Geospiza fortis*) taken at Daphne Island and at Santa Cruz Island in the Galápagos by Peter and Rosemary Grant. ▪ The populations of the two islands differ, although the islands are less than 10 km apart. ▪ The data was extracted from a data set distributed with the case study Natural Selection and Darwin's Finches by Martin Wikelski available on the

DaphneBeaks.txt SantaCruzBeaks.txt	web at http://wps.prenhall.com/esm_freeman_evol_3/0,8018,8412374-,00.html. • The original data is summarized in the article: "The classical case of character release: Darwin's finches (Geospiza) on Isla Daphne Major, Galápagos" by P. T. Boag and P. R. Grant that appeared in *Biological Journal of the Linnean Society* 22:243-287 (1984). See http://en.wikipedia.org/wiki/Peter_and_Rosemary_Grant for additional information on the work of Peter and Rosemary Grant.

SETUP FOR LESSON

- Create an `Histograms` directory on your `V:` drive and make it your current directory.
- Download the `DaphneBeaks.txt` and `SantaCruzBeaks.txt` to your `V:\Histograms` directory.
- Create a `HistogramsLesson.m` script file in your `Histograms` directory. Enter each of the examples in a new cell in this script.

EXAMPLE 1: Input the Daphne Island and Santa Cruz Island data (load)

Create a new cell in which you type and execute:

```
Daphne = load('DaphneBeaks.txt');
SantaCruz = load('SantaCruzBeaks.txt');
```

You should see the following 2 variables in your Workspace Browser:

- `Daphne` - a column vector with beak sizes of the Daphne Island finches
- `SantaCruz` - a column vector with the beak sizes of the Santa Cruz Island finches

EXAMPLE 2: Display a histogram of the Daphne Island beak size data

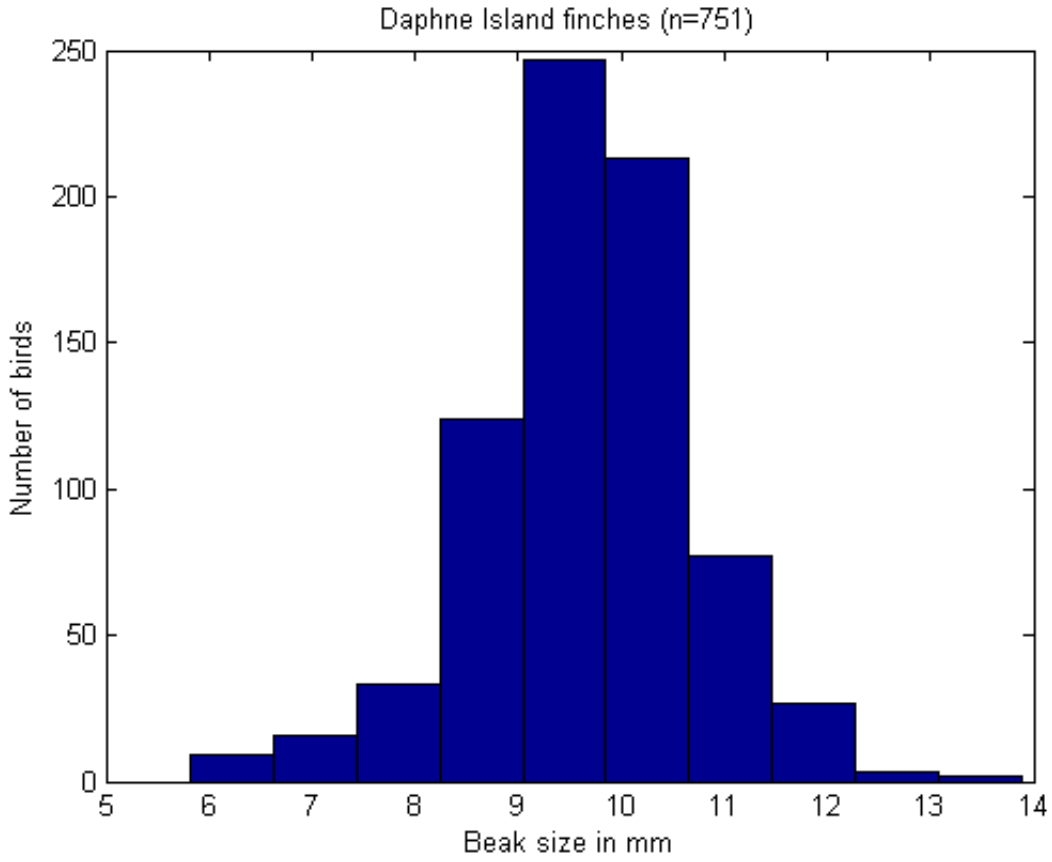
Create a new cell in which you type and execute:

```
nDaphne = length(Daphne);     % Find number of Daphne finches
titleDaphne = ['Daphne Island finches (n=' num2str(nDaphne) ')'];
figure('Name', titleDaphne); % Create a titled figure window
hist(Daphne)                 % Calculate and plot the histogram
xlabel('Beak size in mm');    % Label x-axis
ylabel('Number of birds');    % Label y-axis
title(titleDaphne);          % Use same title for plot and window
```

You should see the following 2 variables in your Workspace Browser:

- `nDaphne` - the number of Daphne Island finches in the study
- `titleDaphne` - a string to be used for a figure title.

You also should the following labeled and titled plot:



EXERCISE 1: Create a histogram for the NYC chicken pox data

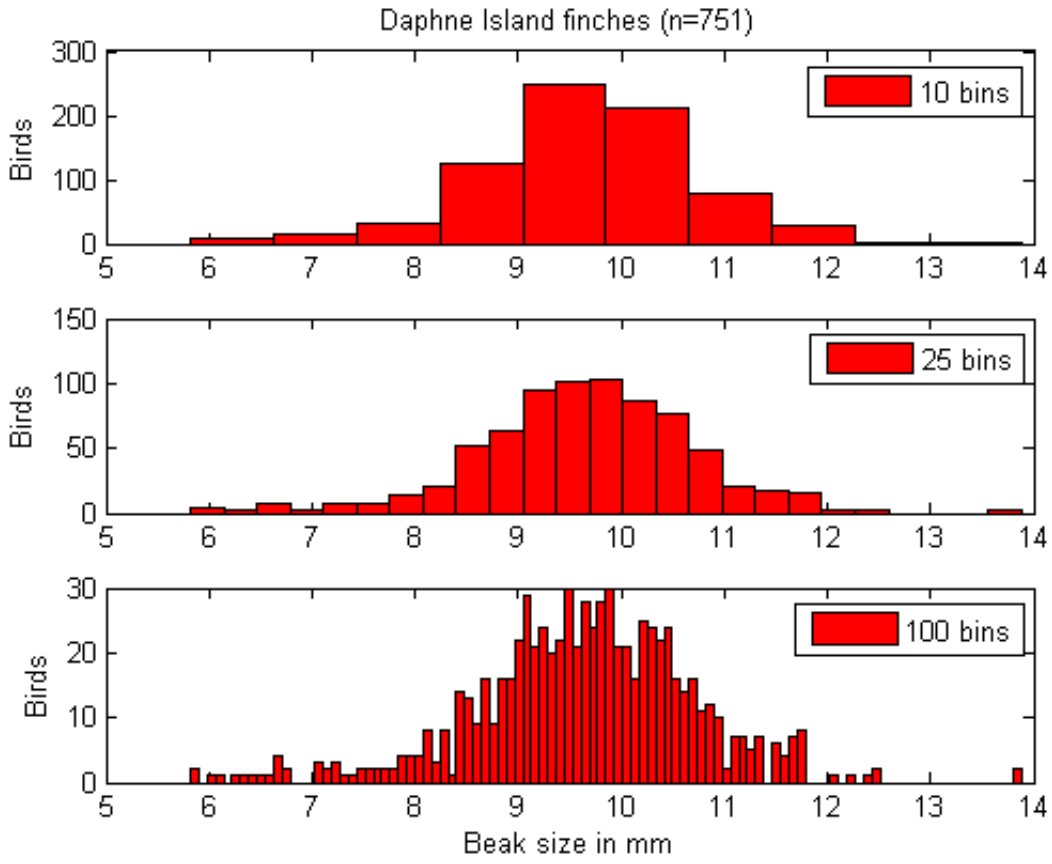
Download the NYCDiseases.mat dataset and create a new figure with a histogram showing the overall distribution of chicken pox counts. (Note: if your plot looks like a rainbow, you have done it incorrectly.) How many data points are represented by this histogram?

EXAMPLE 3: Use different choices of number of bins for Daphne Island histograms

Create a new cell in which you type and execute:

```
figure                % New figure window
colormap autumn       % Change figure color scheme
subplot(3, 1, 1)      % ---Top graph---
hist(Daphne, 10)      % Plot a 10-bin histogram
title(titleDaphne)    % Put title over topmost graph
legend('10 bins')
ylabel('Birds')
subplot(3, 1, 2)      % ---Middle graph---
hist(Daphne, 25)      % Plot a 25-bin histogram
legend('25 bins')
ylabel('Birds')
subplot(3, 1, 3)      % ---Bottom graph---
hist(Daphne, 100)     % Plot a 100-bin histogram
legend('100 bins')
ylabel('Birds')
xlabel('Beak size in mm') % Only label bottom x-axis
```

You should see a subplot with three axes aligned vertically:



EXERCISE 2: How many bins?
Which choice for number of bins do you think gives the best representation of the data distribution in EXAMPLE 3?

EXERCISE 3: Use square root rule to choose number of bins.
The square root rule is used by some spreadsheet programs to pick the number of bins. It simply uses the square root of the number of points in the data set. Find the number of bins suggested by this rule for the Daphne data.

EXERCISE 4: Picking number of bins.
Create a graph similar to that of EXAMPLE 3 for the chicken pox data in Exercise 1. Which of the three bin sizes (10, 25, 100) gives you a better picture of the histogram shape. How would you describe this shape?

EXAMPLE 4: Compare beak distributions of Daphne and Santa Cruz Islands

Create a new cell in which you type and execute:

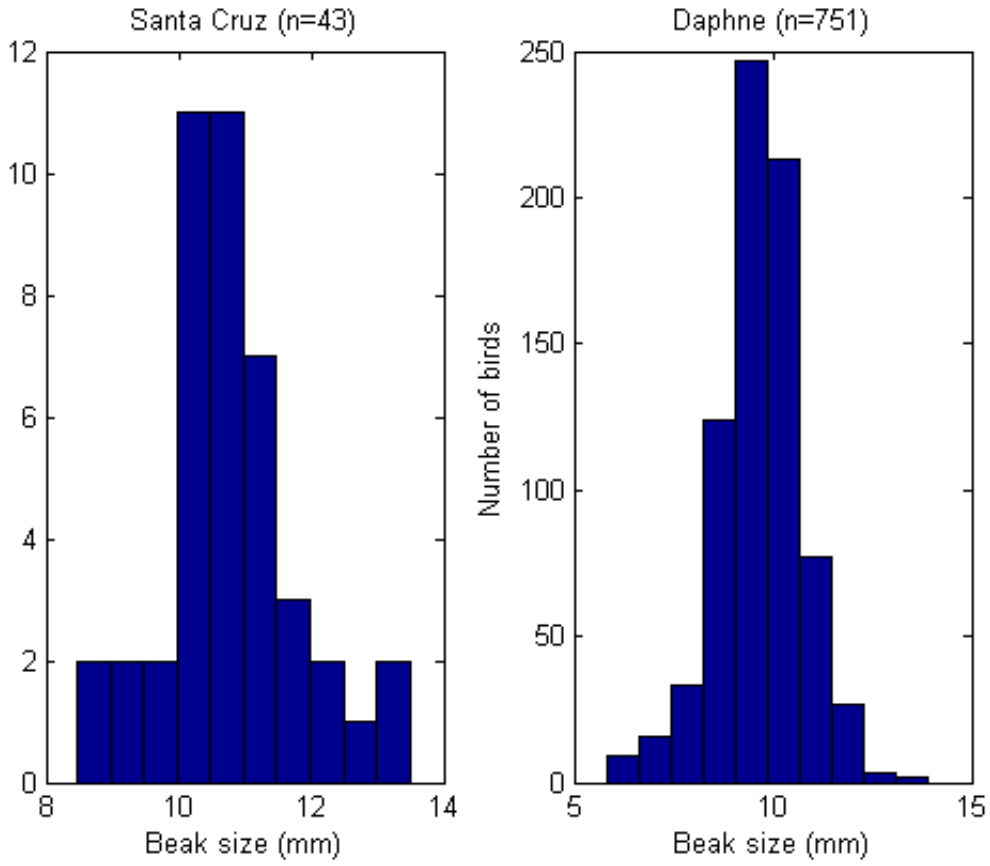
```
nSantaCruz = length(SantaCruz);     % Find number Santa Cruz Island finches
figure
subplot(1, 2, 1)
hist(SantaCruz)                     % Histogram of Santa Cruz Island finches
title(['Santa Cruz (n=' num2str(nSantaCruz) ')'])
xlabel('Beak size (mm)')
subplot(1, 2, 2)
```

```
hist(Daphne)                        % Histogram of Daphne Island finches
title(['Daphne (n=' num2str(nDaphne) ')'])
xlabel('Beak size (mm)')
ylabel('Number of birds')           % Only use one y label for both axes
```

You should see the following variable in your Workspace Browser:

- `nSantaCruz` - the number of Santa Cruz Island measurements

You should also see a subplot with two side-by-side axes:



EXERCISE 5: What is wrong with the display of EXAMPLE 4?.
Hint: Can you find three issues?

EXAMPLE 5: Calculate explicit histogram bin positions

Create a new cell in which you type and execute:

```
minBeak = min([min(Daphne), min(SantaCruz)]); % Smallest of the two
maxBeak = max([max(Daphne), max(SantaCruz)]); % Largest of the two
xEdges = linspace(minBeak, maxBeak, 11);  % Find evenly spaced points
xCenters = 0.5*(xEdges(2:end) + xEdges(1:end-1)); % Get bin centers
nD = hist(Daphne, xCenters);        % Daphne counts for these bins
nS = hist(SantaCruz, xCenters);     % Santa Cruz counts for these bins
```

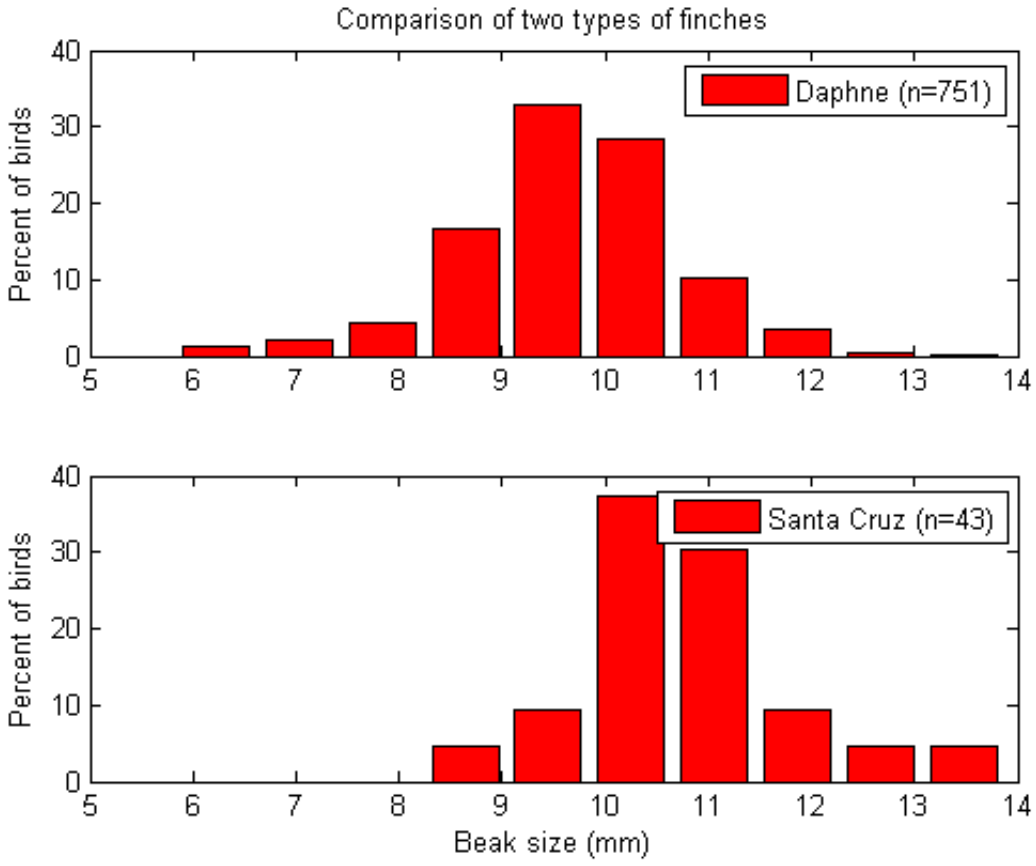
You should see the following variables in your Workspace Browser:

- `minBeak` - smallest beak size in both data sets
- `maxBeak` - largest beak size in both data sets
- `xEdges` - edges of our bins
- `xCenters` - vector with the centers needed to plot
- `nD` - vector with counts of Daphne Island beak sizes at specified bin positions
- `nS` - vector with counts of Santa Cruz Island beak sizes at specified bin positions

EXAMPLE 6: Compare percentages using scaling and explicit bin positions

Create a new cell in which you type and execute:

```
figure                          % New figure window
colormap autumn                 % Change figure color scheme
subplot(2, 1, 1)                % ---Top graph---
bar(xCenters, 100*nD/nDaphne)   % Histogram for Daphne percents
legend(['Daphne (n=' num2str(nDaphne) ')'])
title('Comparison of two types of finches')
ylabel('Percent of birds')
subplot(2, 1, 2)                % ---Bottom graph---
bar(xCenters, 100*nS/nSantaCruz) % Histogram for Santa Cruz percents
legend(['Santa Cruz (n=' num2str(nSantaCruz) ')'])
ylabel('Percent of birds')
xlabel('Beak size (mm)')        % One x-axis label for readability
```



EXERCISE 6: Modify the code of EXAMPLE 6 to show fractions.
Create a new figure in which you display fractions on the y-axis rather than percentages in each histogram.

EXAMPLE 7: Calculate and display a histogram using a bar chart, line graph and stair plot

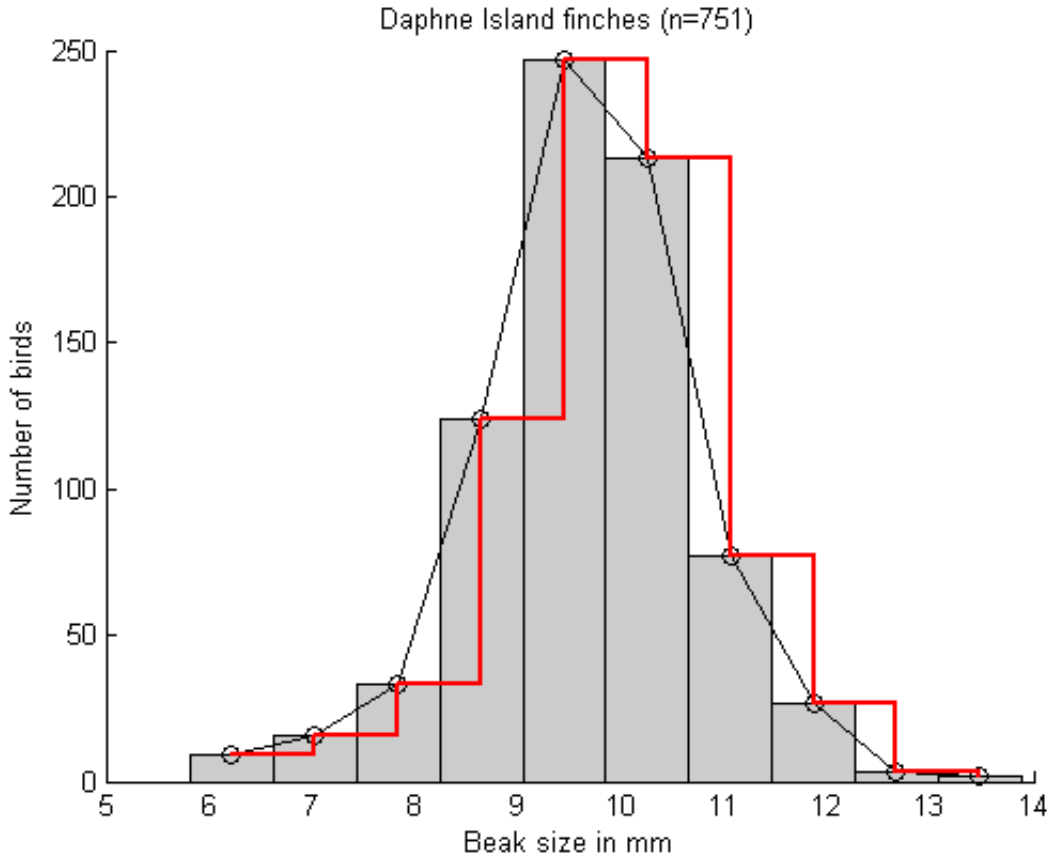
Create a new cell in which you type and execute:

```
[n, xout] = hist(Daphne);      % Calculate histogram but don't display
figure
hold on
bar(xout, n, 1.0, 'FaceColor', [0.8, 0.8, 0.8]); % Plot using a bar chart
plot(xout, n, '-ok')                          % Plot using a line graph
stairs(xout, n, 'r', 'LineWidth', 2)          % Plot using a stair plot
hold off
xlabel('Beak size in mm');
ylabel('Number of birds');
title(titleDaphne);
```

You should see the following 2 variables in your Workspace Browser:

- `n` - vector with the counts of number of beak sizes at each position of `xout`
- `xout` - vector with the positions of the centers of the bins for the histogram

You should the following labeled and titled plot:



Stair plots and line graphs are useful for overlaying histograms for comparison.

EXERCISE 7: Correctly align the stair plot of EXAMPLE 4
Notice that the stairs plot is offset by half of the bin size in EXAMPLE 7 since the first argument of stairs gives the positions that the stairs change. In contrast, `xout` from `hist` gives the centers of the bins. You can fix this by calculating half the bin size as:

```
binHalf = 0.5*(xout(2) - xout(1));
```

If you subtract `binHalf` from `xout` in the stairs plot, everything will line up correctly:

```
stairs(xout - binHalf, n, 'r', 'LineWidth', 2)'
```

Redo EXAMPLE 7, adjusting the stairs so that they correctly align.

EXAMPLE 8: Generate "random" numbers from three common probability distributions

Create a new cell in which you type and execute:

```
yNormal = random('norm', 0, 1, [1000, 1]);  % Normal with zero mean and unit sd
yUniform = random('unif', -1, 1, [1000,1]); % Uniform in the interval [-1, 1]
yExp = random('exp', 1, [1000, 1]);         % Exponential with mean 1
```

You should see the following variables in your Workspace Browser:

- `yNormal` - vector of 1000 normally distributed "random" values

- `yUniform` - vector of 1000 "random" values uniformly distributed in [-1, 1]
- `yExp` - vector of 1000 exponentially distributed "random" values, mean 1

EXAMPLE 9: Display the histograms of the generated distributions

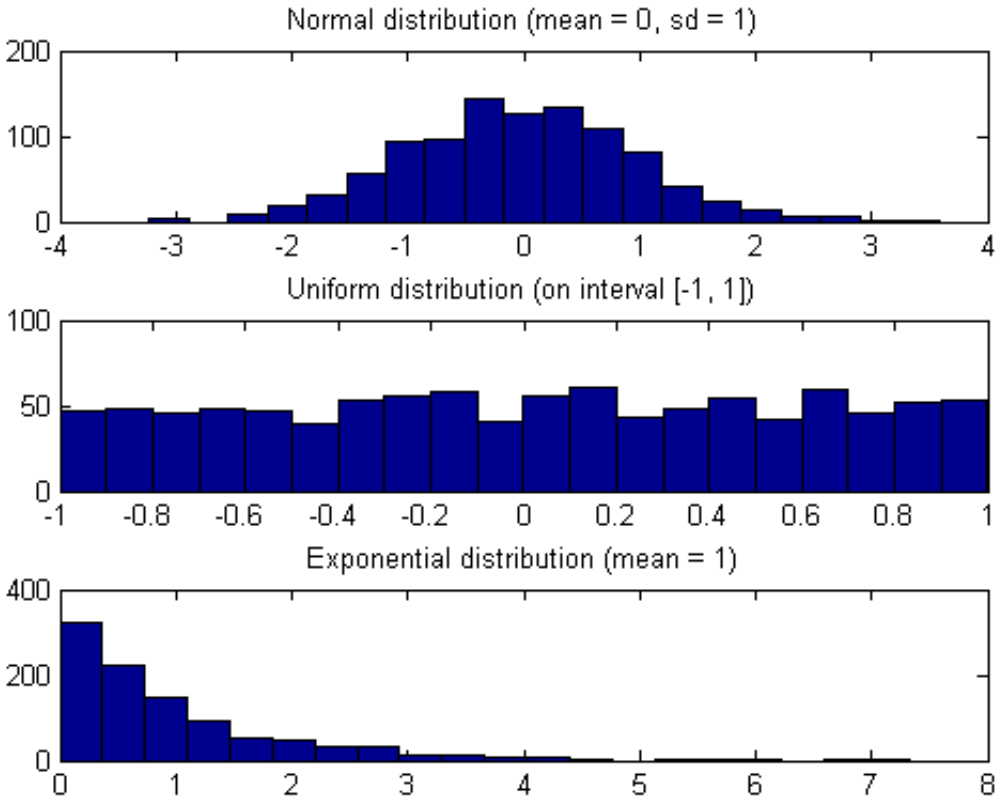
Create a new cell in which you type and execute:

```
figure
subplot(3,1,1)
hist(yNormal, 20)
title('Normal distribution (mean = 0, sd = 1)')

subplot(3,1,2)
hist(yUniform, 20)
title('Uniform distribution (on interval [-1, 1])')

subplot(3,1,3)
hist(yExp, 20)
title('Exponential distribution (mean = 1)')
```

You should see a subplot three axes vertically aligned. Note however that the scales are not aligned so you can only compare shape:



EXERCISE 8: Create a sample of 1000 values from a normal distribution.
Create a variable `yNormal1` that holds a vector of 1000 values drawn from the normal distribution with mean 1 and

standard deviation 1.

EXERCISE 9: Compare the histograms of two normal distributions.
Display the histograms of the normal distribution of EXAMPLE 8 and the normal distribution of Exercise 8 on the same graph (using line plots). Use the square root rule to choose the number of bins.

SUMMARY OF SYNTAX

MATLAB syntax	Description
`hist(x)`	creates a histogram plot of the values in the vector `x`.
`[n, xout] = hist(x)`	calculates the histogram of the vector `x`, but does not plot anything. The `n` variable contains the counts and the `xout` variable contains bin center locations.
`random(distName, parameters, [n, m])`	generates an n x m array of numbers randomly selected from the specified probability distribution. The `parameters` item represents the values of the parameters needed to define the particular probability distribution. For example, a normal distribution is specified by its mean and standard deviation. On the other hand, the exponential distribution is specified only by its mean. The uniform distribtion is specified by its two end points (i.e., values are evenly distributed between the two end point values).
`stairs(Y)`	plots stair-step graphs of the columns of the array `Y` against the positive integers.
`stairs(X, Y)`	plots stair-step graphs of the columns of the array `Y` against the columns of the array `X`.

This lesson was written by Kay A. Robbins of the University of Texas at San Antonio and last modified on March 23, 2015. Please contact kay.robbins@utsa.edu with comments or suggestions. *The image, Medium Ground Finch Geospiza fortis, Santa Cruz, Galapago taken by Mark Putney. The original source is http://www.flickr.com/photos/putneymark/13516124843/in/set-72157601810082531/. The image is available under common license at http://commons.wikimedia.org/wiki/File:Geospiza_fortis.jpg.*

Published with MATLAB® 8.3