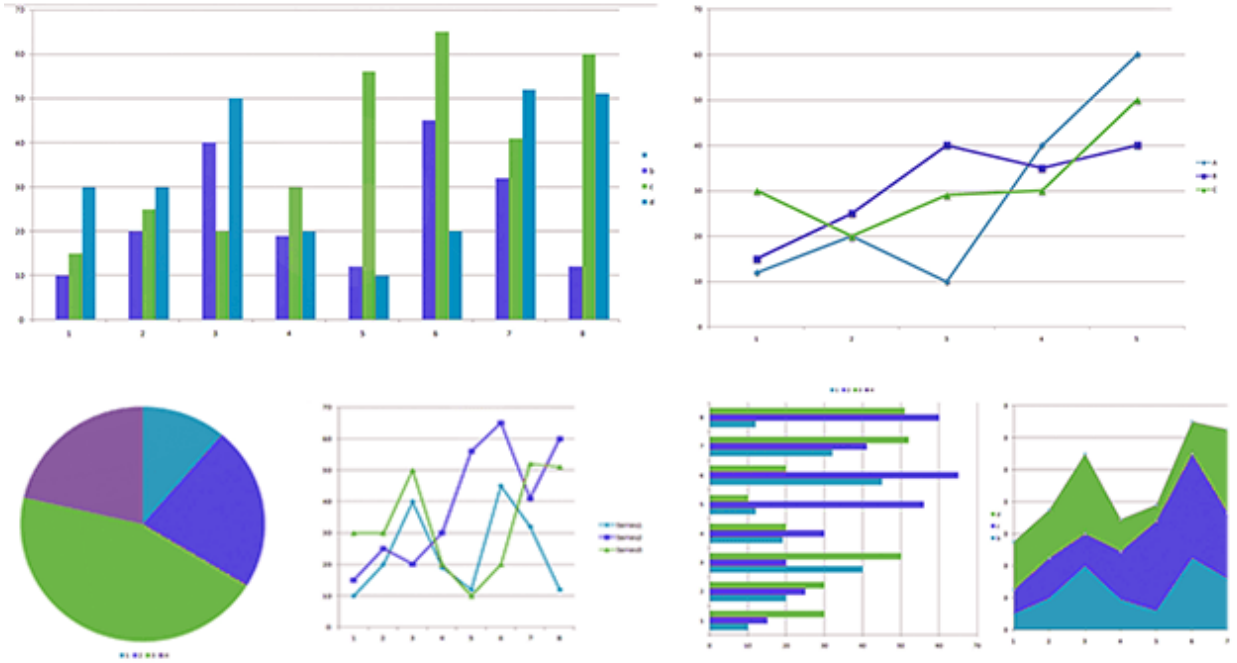
Python Programming

Dr. Cao Tien Dung
School of Engineering - Tan Tao University

Vicohub
https://vicohub.com

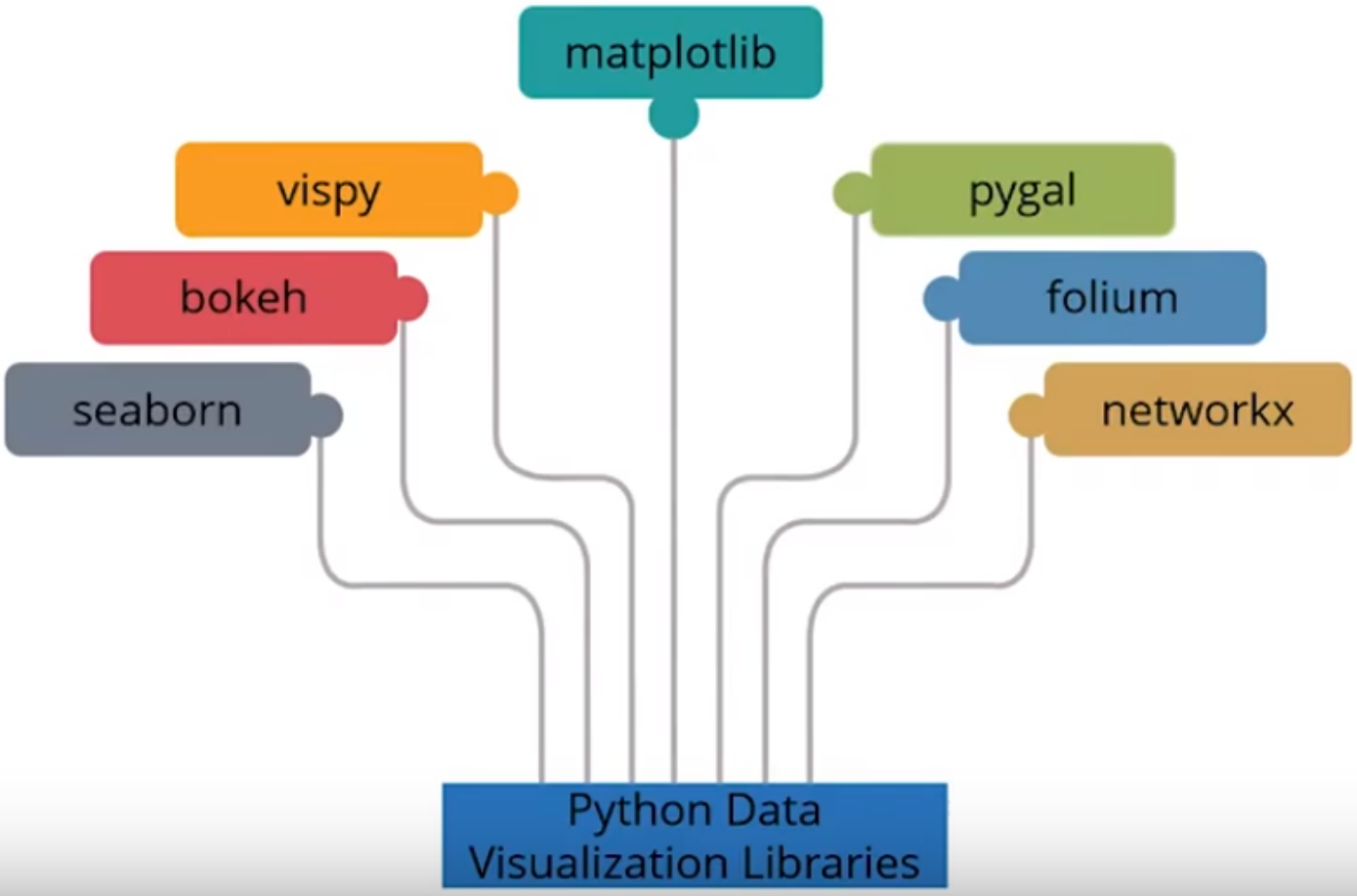
Data visualization

- The graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and **understand** trends, outliers, and patterns in data.



Data visualization on Python

Many new Python data visualization libraries are introduced recently such as:

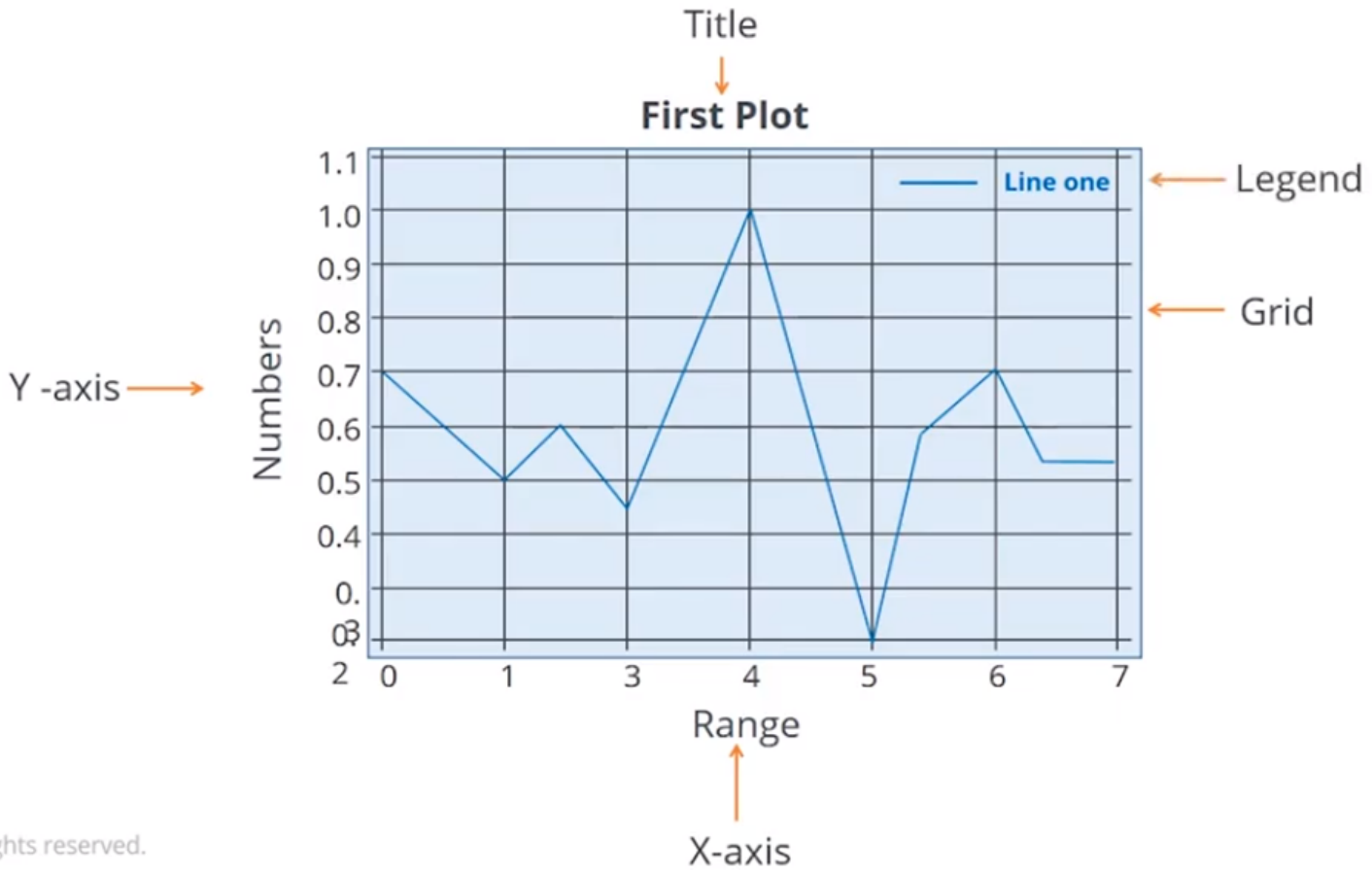


Matplotlib

- 2D plotting library.
- Can use in Python scripts (the Python and IPython shells), the Jupyter notebook, web application servers, and four graphical user interface toolkits.
- Easy to use, generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc., with just a few lines of code.
- Integrated with Pandas
- See sample: https://matplotlib.org/gallery/index.html

Understanding the plot

A plot is a graphical representation of data which shows relationship between two variables or the distribution of data.



Step to create a plot

You can create a plot using four simple steps.

Steps to create plot - example

In [1]:
```
#import numpy for generating random numbers
import numpy as np
#import matplotlib library
import matplotlib.pyplot as plt
from matplotlib import style
%matplotlib inline
```

In [21]:
```
#generate random numbers (total 10)
randomNumber = np.random.rand(10)
```

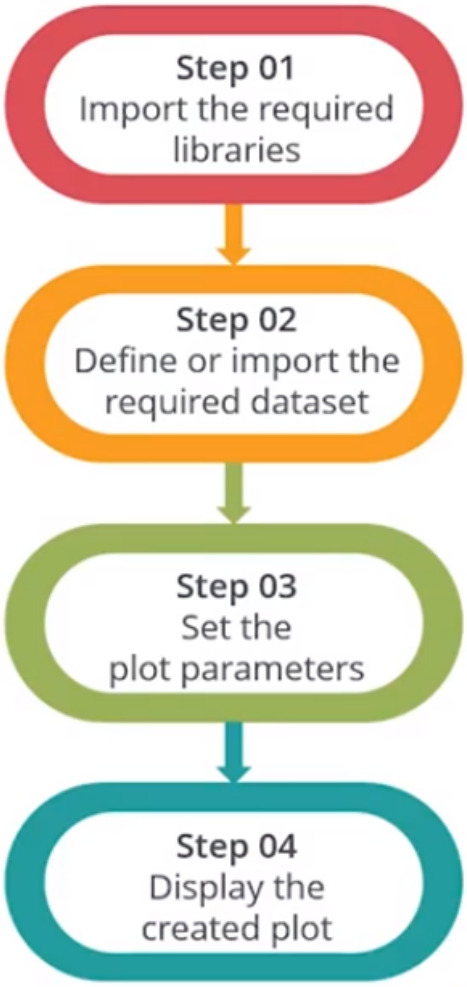
In [22]:
```
#view them
print randomNumber
```

```
[ 0.71892609  0.49065612  0.61092193  0.43397501  0.94771363  0.31505178
  0.58568599  0.6929941   0.4288734   0.43774794]
```

In [23]:
```
#select the style of the plot
style.use('ggplot')
#plot the random number
plt.plot(randomNumber,'g',label='line one',linewidth=2)
#x axis is number of random numbers (index)
plt.xlabel('Range')
#y axis is actual random number
plt.ylabel('Numbers')
#Title of the plot
plt.title('First Plot')
```

```
plt.legend()
plt.show()
```

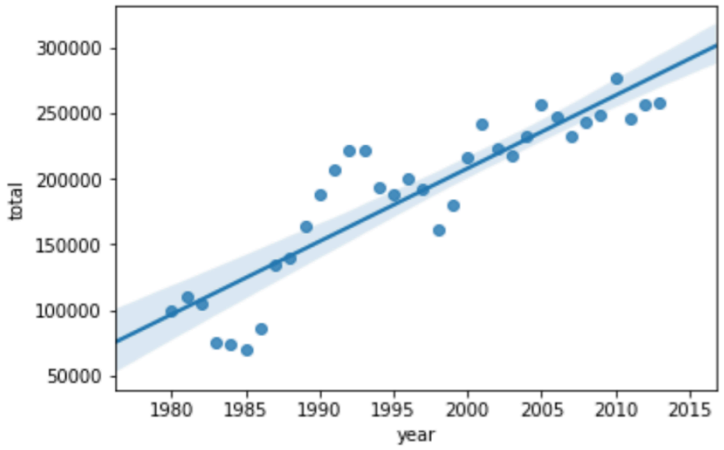
Display the created plot



Seaborn & Regression Plots

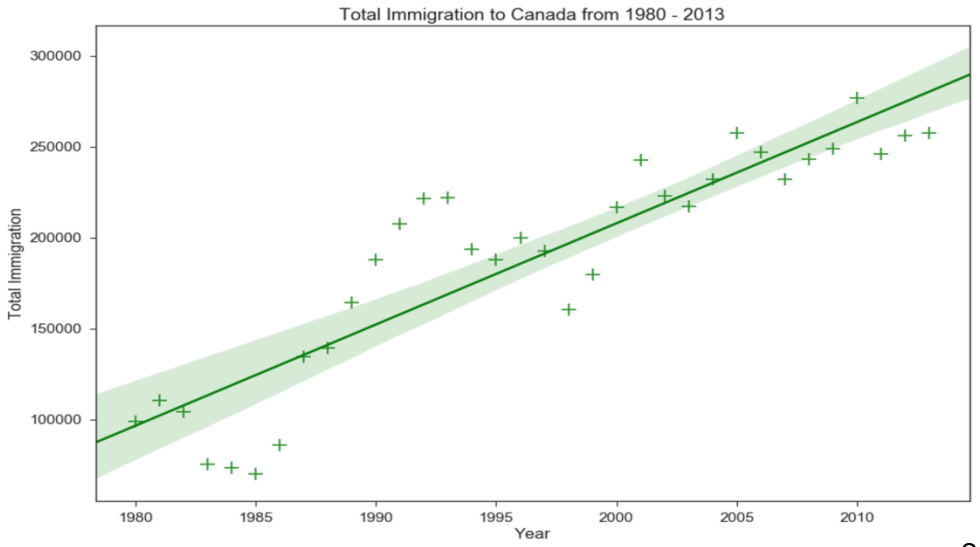
- Seaborn is another data visualization libs but it is based on Matplotlib
  - May generate codes with 5 times less than Matplotlib

```
import seaborn as sns
ax = sns.regplot(x='year', y='total', data=df_tot)
```



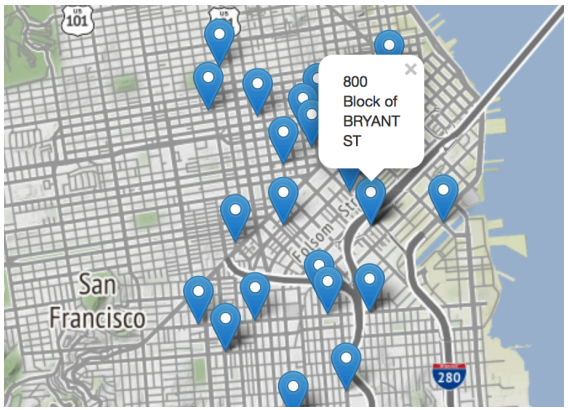

```
plt.figure(figsize=(15, 10))

sns.set(font_scale=1.5)
sns.set_style('ticks') # change background to white background

ax = sns.regplot(x='year', y='total', data=df_tot, color='green', marker='+', scatter_kws={'s': 200})
ax.set(xlabel='Year', ylabel='Total Immigration')
ax.set_title('Total Immigration to Canada from 1980 - 2013')
```

```
Text(0.5,1,'Total Immigration to Canada from 1980 - 2013')
```



Folium

- Folium is a powerful data visualization library in Python that was built primarily to help people visualize geospatial data.
  - Markers
  - Choropleth Maps




```
# create a Stamen Toner map of the world centered around Canada
world_map = folium.Map(location=[56.130, -106.35], zoom_start=4, tiles='Stamen Terrain')

# display map
world_map
```

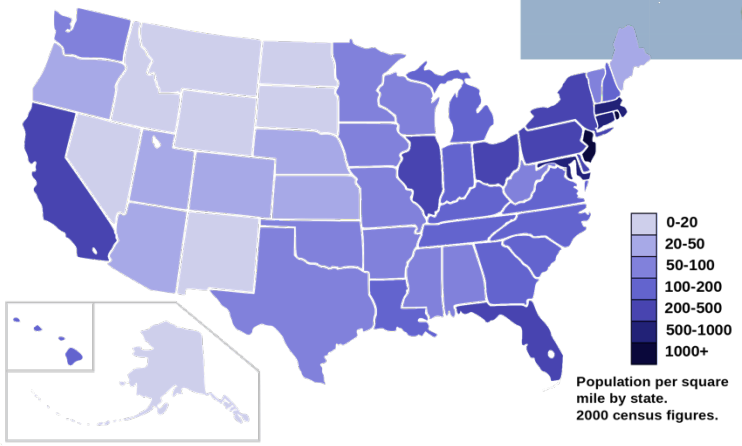
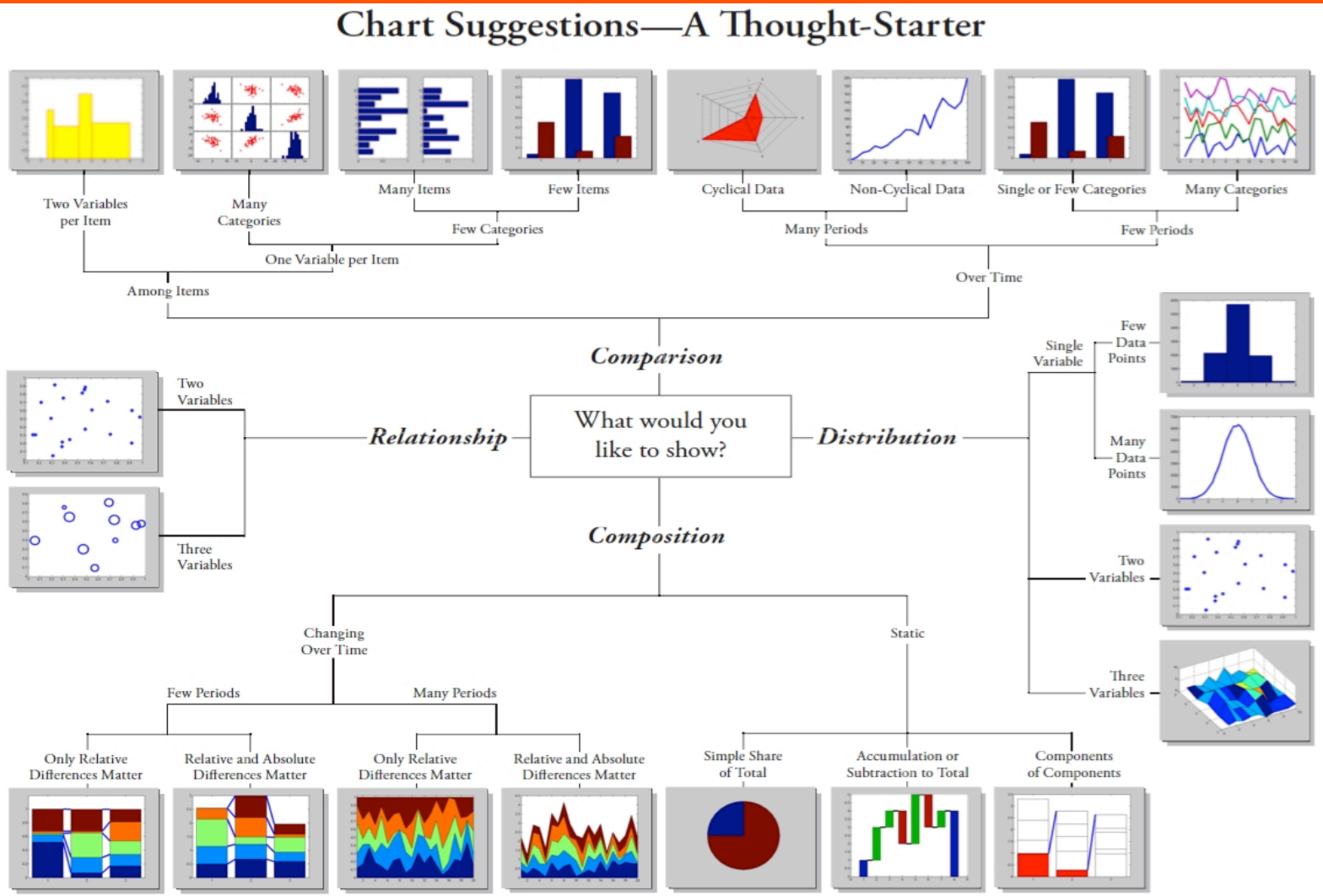


Chart selection



Modified with permision -Doug Hull
blogs.mathworks.com/videos
hull@mathworks.com 2009

Let practice

- See example file