Descriptive Analytics

Lecture 8 – Monday April 10, 2017

Outline

- Reading the data
- Exploring the data
- Data Summarization

Outline

- **Reading the data**
- Exploring the data
- Data Summarization

Reading the Data

- **Oil Production per Countries**

The Excel sheet 'countries.xlsx' contains the oil production for the oil producers from 1965 to 2014 provided by BP.

Million tonnes	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
US	413.0	397.0	387.5	383.6	382.1	380.0	368.1	352.6	347.6	344.5	342.0	332.5	325.4	308.8	304.6	305.2	302.3	322.3	333.1	345.4	394.7	448.5	519.9
Canada	97.2	102.3	106.7	111.9	115.5	120.7	125.1	121.0	124.6	125.7	132.6	140.2	144.8	142.3	150.6	155.3	152.9	152.8	160.3	169.8	182.6	194.4	209.8
Mexico	153.0	153.3	154.2	150.2	162.4	169.6	173.5	165.5	170.3	175.9	177.8	188.2	190.0	186.5	182.5	172.2	156.9	146.7	145.6	144.5	143.9	141.8	137.1
Argentina	29.0	31.1	34.5	37.5	40.8	43.4	44.0	41.8	40.7	43.7	43.1	42.8	40.9	39.4	39.1	38.0	36.5	33.6	33.0	30.6	30.4	29.9	29.5
Brazil	34.2	34.8	36.2	37.6	42.4	45.5	52.6	59.5	66.9	70.0	78.5	81.3	80.7	89.1	93.8	95.2	98.9	105.8	111.4	114.1	112.1	109.8	122.1
Colombia	23.3	24.0	24.1	31.0	33.4	34.8	40.4	43.7	36.3	31.8	30.4	28.5	27.9	27.7	27.9	28.0	31.1	35.3	41.4	48.2	49.9	52.9	52.2
Ecuador	17.5	18.7	20.6	21.0	20.9	21.1	20.4	20.3	21.6	21.9	21.1	22.5	28.3	28.6	28.8	27.5	27.2	26.1	26.1	26.8	27.1	28.2	29.8
Peru	6.1	6.6	6.7	6.4	6.3	6.2	6.0	5.5	5.0	4.9	4.9	4.6	4.4	4.5	4.6	4.6	4.7	4.8	5.1	4.9	4.8	4.6	4.9
Trinidad & Tobago	6.9	6.5	6.8	6.7	6.7	6.4	6.4	6.6	6.9	6.9	8.0	8.7	8.2	9.0	9.6	8.2	8.7	7.6	7.4	6.9	6.0	5.7	5.5
Venezuela	131.6	136.1	144.5	155.3	165.2	174.4	179.6	160.9	159.8	162.9	152.8	147.5	170.1	169.7	171.0	165.5	165.6	155.7	145.7	140.5	139.3	137.9	139.5
Other S. & Cent. America	3.6	3.9	4.3	4.6	5.0	5.3	6.2	6.2	6.5	6.9	7.7	7.8	7.4	7.4	7.1	7.1	7.1	6.6	6.9	7.0	7.3	7.4	7.5
Azerbaijan	11.2	10.3	9.6	9.2	9.1	9.0	11.4	13.9	14.1	15.0	15.3	15.4	15.5	22.2	32.3	42.6	44.5	50.4	50.8	45.6	43.4	43.5	42.0
Denmark	7.7	8.2	9.0	9.1	10.2	11.2	11.6	14.6	17.7	17.0	18.1	17.9	19.1	18.5	16.8	15.2	14.0	12.9	12.2	10.9	10.0	8.7	8.1
Italy	4.5	4.6	4.9	5.2	5.5	5.9	5.6	5.0	4.6	4.1	5.5	5.6	5.5	6.1	5.8	5.9	5.2	4.6	5.1	5.3	5.4	5.6	5.8
Kazakhstan	25.8	23.0	20.3	20.6	23.0	25.8	25.9	30.1	35.3	40.1	47.3	51.5	59.5	61.5	65.0	67.1	70.7	76.5	79.5	80.0	79.2	81.8	80.8
Norway	106.9	114.1	128.6	138.4	154.7	156.2	149.6	149.7	160.7	162.5	157.9	153.9	150.3	138.7	129.0	118.6	114.8	108.7	98.8	93.8	87.3	83.2	85.6
Romania	6.8	6.9	7.0	7.0	6.9	6.8	6.6	6.4	6.3	6.2	6.1	5.9	5.7	5.4	5.0	4.7	4.7	4.5	4.3	4.2	4.0	4.1	4.0
Russian Federation	398.8	354.9	317.6	310.7	302.9	307.4	304.3	304.8	326.7	351.7	383.7	425.7	463.3	474.8	485.6	496.8	493.7	500.8	511.8	518.8	526.1	531.0	534.1
Turkmenistan	5.2	4.4	4.2	4.1	4.4	5.4	6.4	7.1	7.2	8.0	9.0	10.0	9.6	9.5	9.2	9.8	10.3	10.4	10.7	10.7	11.0	11.4	11.8
United Kingdom	94.3	100.2	126.5	129.9	129.7	127.9	132.6	137.4	126.2	116.7	115.9	106.1	95.4	84.7	76.6	76.6	71.7	68.2	63.0	52.0	44.6	40.6	39.7
Uzbekistan	3.3	4.0	5.5	7.6	7.6	7.9	8.2	8.1	7.5	7.2	7.2	7.1	6.6	5.4	5.4	4.9	4.8	4.5	3.6	3.6	3.2	3.2	3.1
Other Europe & Eurasia	31.3	29.0	29.3	27.6	26.3	25.1	24.2	22.7	22.3	22.2	23.6	24.0	23.4	22.0	21.7	21.6	20.6	19.9	19.2	19.2	19.2	19.6	19.1
Iran	175.7	184.3	185.0	185.5	186.6	187.0	190.8	178.1	191.7	189.8	177.5	198.5	208.2	206.4	209.2	210.9	214.5	205.5	208.7	208.8	177.3	165.8	169.2
Iraq	26.1	22.3	24.8	26.0	28.6	57.1	104.2	128.3	128.8	123.9	103.9	66.0	99.9	89.9	98.0	105.1	119.3	119.9	121.5	136.7	152.5	153.2	160.3
Kuwait	54.0	96.6	103.4	104.9	105.1	105.1	110.0	102.6	109.9	106.6	98.9	115.6	123.4	130.4	133.7	129.9	136.1	121.2	123.4	140.8	154.0	151.5	150.8
Oman	37.0	38.8	40.5	42.8	44.4	44.9	44.7	45.0	47.7	47.5	44.6	40.7	38.9	38.5	36.5	35.2	37.6	40.2	42.8	43.8	45.0	46.1	46.2
Qatar	23.6	21.8	21.3	21.8	27.1	33.3	33.6	34.3	40.2	40.0	37.4	43.8	50.0	52.6	56.8	57.9	65.0	62.4	71.7	78.5	83.4	84.3	83.5

Reading the Data

- **Importing the data**
  - ***xlsread***: EXCEL data
  - ***dlmread***: tab-delimited text (or any other form of delimited text, e.g., whitespace)
  - ***csvread:*** comma-separated numbers
  - ***textread***: any mixture of text and numbers
  - ***fopen/fread***: any formatted data by line, but need extensive user specification of format
  - ***importdata***: any formatted data as a full file (looks for the most appropriate function to use)
  - ***help <functionname>*** and ***doc <functionname>*** give instructions and examples
  - MATLAB can also be used to save data in the corresponding formats (e.g., ***dlmwrite***, ***csvwrite***, ***fopen/fwrite/fprintf***)

Reading the Data

- **Importing the data**

To import the data into Matlab, use

```
>> ds=importdata('countries.xlsx')

ds =

          data: [55x50 double]
      textdata: {55x1 cell}
    rowheaders: {55x1 cell}
```

All the numeric data is stored in a double array called data

All the char data is stored in a cell array called textdata

Row headers (country names)

Reading the Data

- **Oil Production per Countries**

```
>> ds.data
ans =
  1.0e+03 *
 Columns 1 through 13
   1.9650   1.9660   1.9670   1.9680   1.9690   1.9700   1.9710   1.9720   1.9730   1.9740   1.9750   1.9760   1.9770
   0.4277   0.4545   0.4842   0.5029   0.5114   0.5335   0.5259   0.5279   0.5147   0.4914   0.4698   0.4580   0.4628
   0.0439   0.0482   0.0527   0.0571   0.0622   0.0701   0.0752   0.0867   0.1003   0.0944   0.0816   0.0753   0.0756
   0.0181   0.0185   0.0205   0.0219   0.0230   0.0242   0.0241   0.0251   0.0259   0.0324   0.0402   0.0448   0.0544
   0.0138   0.0146   0.0160   0.0175   0.0181   0.0200   0.0216   0.0222   0.0216   0.0211   0.0203   0.0204   0.0221
   0.0050   0.0061   0.0077   0.0085   0.0093   0.0088   0.0092   0.0090   0.0091   0.0095   0.0093   0.0091   0.0087
   0.0107   0.0104   0.0101   0.0093   0.0112   0.0118   0.0117   0.0106   0.0099   0.0091   0.0085   0.0079   0.0075
   0.0004   0.0004   0.0003   0.0003   0.0002   0.0002   0.0002   0.0042   0.0112   0.0095   0.0086   0.0101   0.0098
   0.0034   0.0034   0.0038   0.0040   0.0039   0.0039   0.0033   0.0035   0.0038   0.0041   0.0038   0.0040   0.0048
   0.0067   0.0076   0.0089   0.0095   0.0078   0.0069   0.0064   0.0070   0.0082   0.0093   0.0107   0.0105   0.0113
   0.1841   0.1788   0.1879   0.1918   0.1908   0.1972   0.1899   0.1738   0.1814   0.1607   0.1271   0.1248   0.1215
   0.0022   0.0025   0.0036   0.0039   0.0039   0.0030   0.0036   0.0040   0.0040   0.0038   0.0035   0.0035   0.0030
      NaN      NaN      NaN      NaN      NaN      NaN      NaN      NaN      NaN      NaN      NaN      NaN      NaN
        0        0        0        0        0        0        0   0.0001   0.0001   0.0001   0.0001   0.0002   0.0005
   0.0023   0.0019   0.0017   0.0016   0.0016   0.0015   0.0014   0.0012   0.0011   0.0011   0.0012   0.0013   0.0012
...
```

Reading the Data

- **Oil Production per Countries**

```
>> ds.textdata
ans =
    'Million tonnes'
    'US'
    'Canada'
    'Mexico'
    'Argentina'
    'Brazil'
    'Colombia'
    'Ecuador'
    'Peru'
    'Trinidad & Tobago'
    'Venezuela'
    'Other S. & Cent. America'
    'Azerbaijan'
    'Denmark'
    'Italy'
    'Kazakhstan‘
...
```

```
>> ds.rowheaders
ans =
    'Million tonnes'
    'US'
    'Canada'
    'Mexico'
    'Argentina'
    'Brazil'
    'Colombia'
    'Ecuador'
    'Peru'
    'Trinidad & Tobago'
    'Venezuela'
    'Other S. & Cent. America'
    'Azerbaijan'
    'Denmark'
    'Italy'
    'Kazakhstan‘
...
```

Reading the Data

- **Avoid multiple import of the same data file**

```
% read the data from the Excel sheet

% make sure that the data is not loaded

if ~exist('ds')

 ds=importdata('countries.xlsx');

end
```

Outline

- Reading the data
- **Exploring the data**
- Data Summarization

Exploring the data

- **Plot the data**

```
% close all the figures

close all;



% read the data from the Excel sheet

% make sure that the data is not loaded

if ~exist('ds')

 ds=importdata('countries.xlsx');

end
```

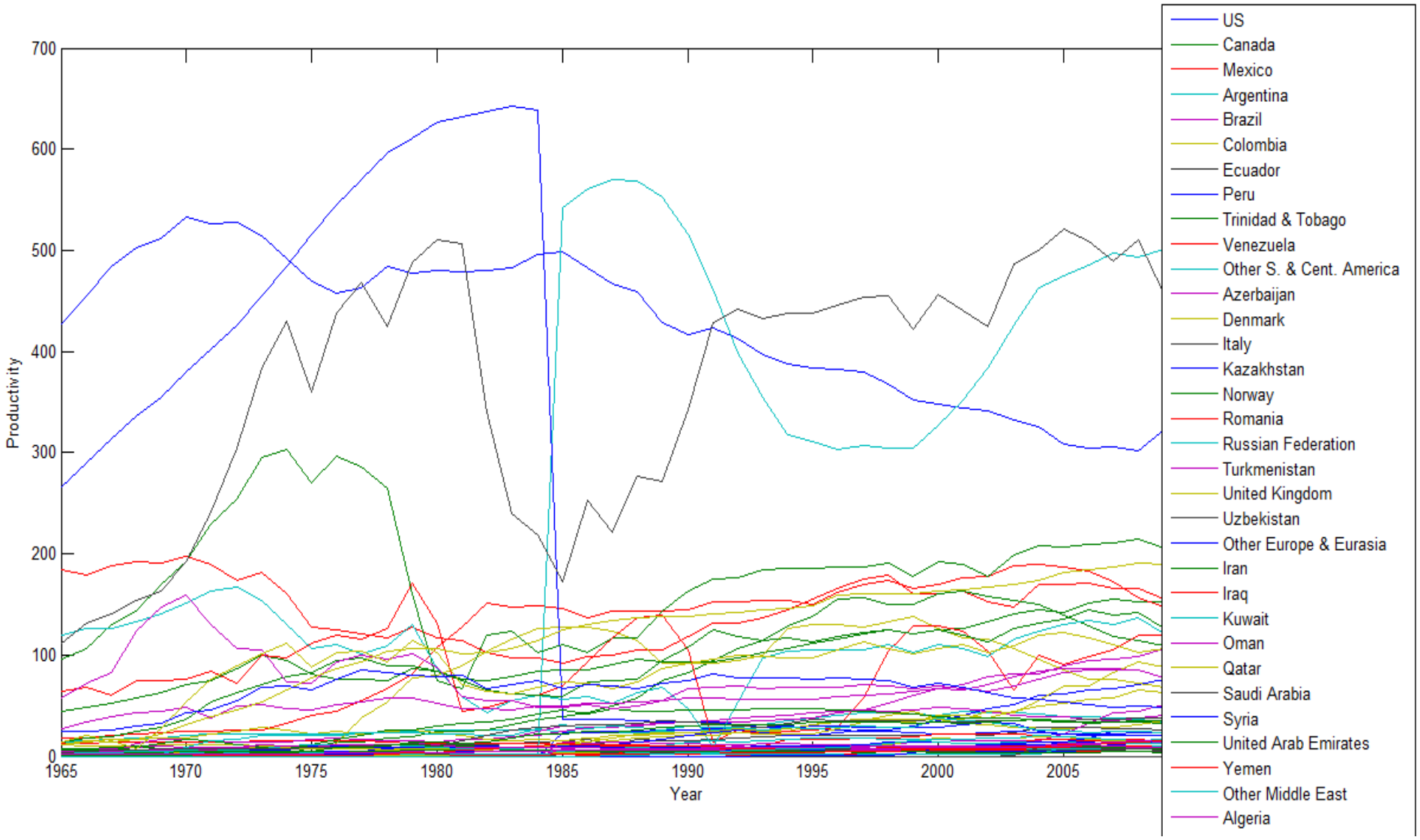
Exploring the data

- **Plot the data**

```
% Plot the data of all countries

plot(ds.data(1,:),ds.data(2:length(ds.data),:));

% Add labels, titles and legend

xlabel('Year');

ylabel('Productivity');

set(gcf,'color','w');

legend(ds.textdata(2:length(ds.textdata)));
```

Exploring the data

- **Plot the data**



Exploring the data

- **Plot the trend of the record of a specific country**

```
%Find a specific country

a=strfind(ds.textdata,'Egypt');



% Find the row number of the selected country

ind1=find(~cellfun(@isempty,a));
```

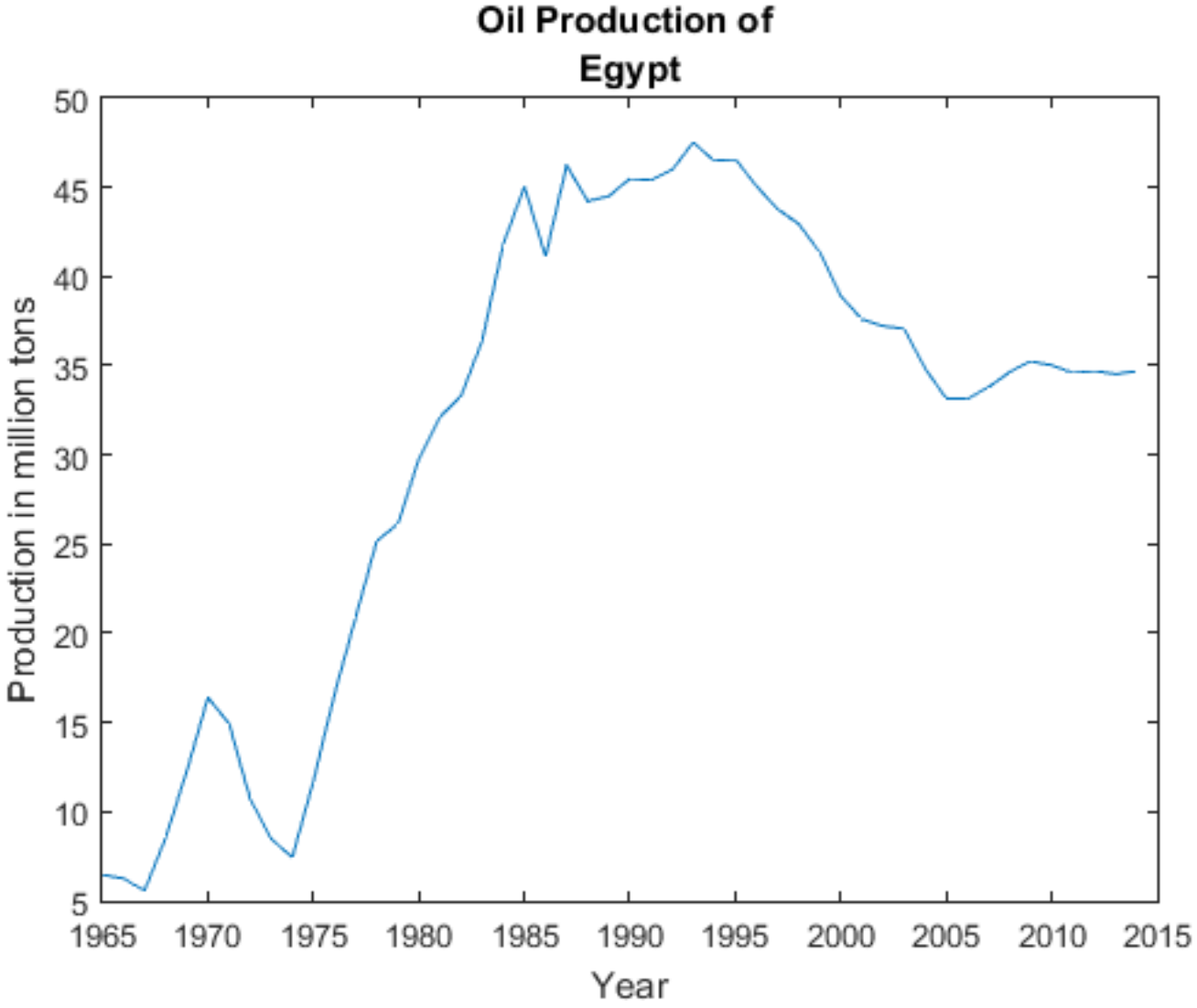
Exploring the data

- **Plot the trend of the record of a specific country**

```
% Plot the production of the selected county
against the years
figure;
plot(ds.data(1,:),ds.data(ind1,:),'--');
set(gcf,'color','w');
% Add labels and titles
xlabel('Year');
ylabel('Production in million tons');
title(['Oil Production of ', ds.textdata(ind1)]);
```

Exploring the data

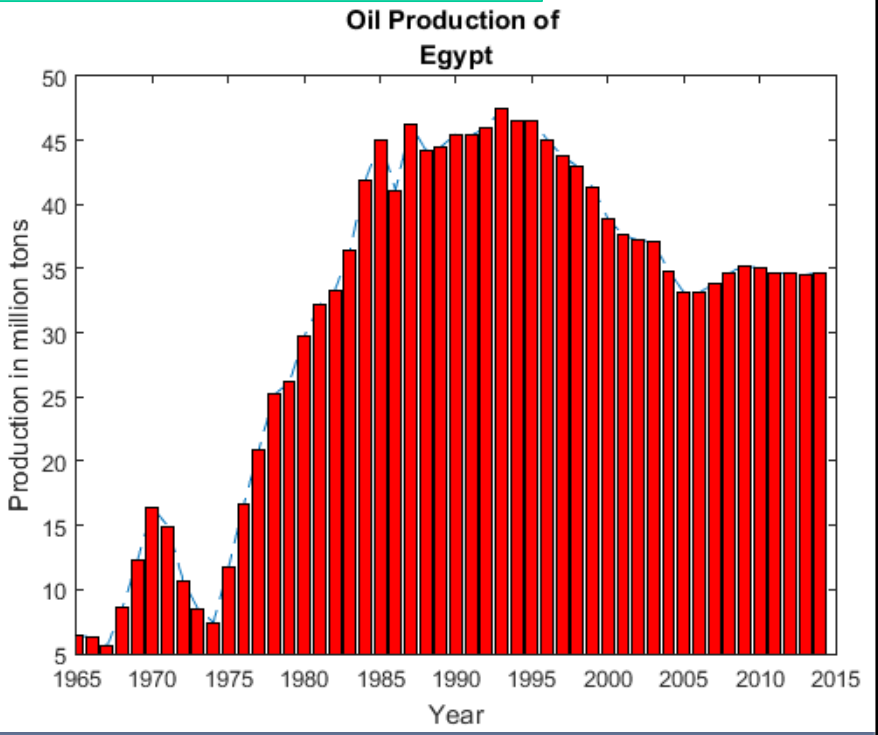
- **Plot the trend of the record of a specific country**



Exploring the data

- **Plot the record and its trend of a specific country**

```
hold on;

bar(ds.data(1,:),ds.data(ind1,:),'r');

axis([1965,2015,5,50]);
```



Exploring the data

- **Plot two countries**

```
%Find the first country

a=strfind(ds.textdata,'Egypt');


% Find the row number of the selected country

ind1=find(~cellfun(@isempty,a));
```

```
%Find the second country

b=strfind(ds.textdata,'India');

% Find the row number of the selected country

ind2=find(~cellfun(@isempty,b));
```

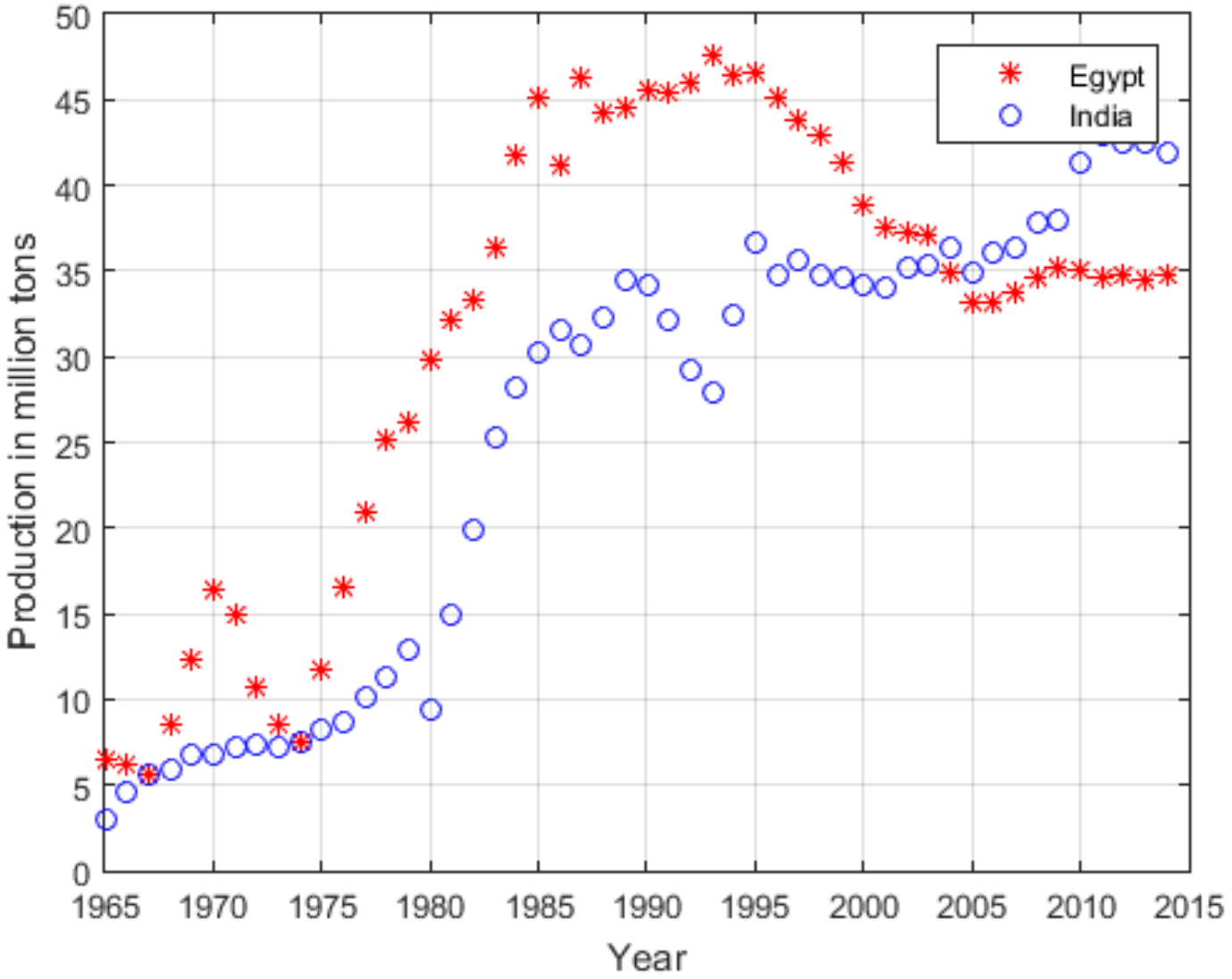
Exploring the data

- **Plot two countries**

```
figure;
% using vector notation
plot(ds.data(1,:),ds.data(ind1,:),'r*'); % Egypt
plot
hold on;
plot(ds.data(1,:),ds.data(ind2,:),'bo'); % India
plot
set(gcf,'color','w');
xlabel('Year');
ylabel('Production in million tons');
legend('Egypt','India');
grid on;
```

Exploring the data

- **Plot two countries**



Exploring the data

- **Find the giant producers**

```
% exclude the first row as it indicates the
year
prod=ds.data(2:length(ds.data),:);


% To get the row numbers where the maximum
data values occur in each data column,
% specify a second output parameter indx to
return the row index
[mx,inds] = max(prod);
```

Exploring the data

- **Find the giant producers**

```
% sort unique the found indices of the giant
producers
giants=unique(inds);


% Display the list of giant producers
disp('Giant producers are: ');
ds.textdata(giants+1) % 1 is added for the
first row that represents the units
```

Exploring the data

- **Find the giant producers**

```
>> Example_1
Giant producers are:


ans =

    'US'
    'Russian Federation'
    'Other Europe & Eurasia'
    'Saudi Arabia'
```

Outline

- Reading the data
- Exploring the data
- **Data Summarization**

Data Summarization

- **Descriptive Statistics**

Function	Description
max	Maximum value
mean	Average or mean value
median	Median value
min	Smallest value
mode	Most frequent value
std	Standard deviation
var	Variance, which measures the spread or dispersion of the values

Data Summarization

```
% Find the maximum value in each column
mx = max(prod);


% To find the maximum value in the entire count
matrix
mxall=max(prod(:));

% Tell the user
fprintf(' The maximum productivity in the
entire years and countries (Million tonnes):
%f\n', mxall);
```

Data Summarization

```
% Find the minimum value in each column
mn = min(prod);
% To find the minimum value in the entire count
matrix
mnall=min(prod(:));
% Tell the user
fprintf(' The minimum productivity in the
entire countries (Million tonnes): %f\n',
mnall);
% To find the minimum value in the entire count
matrix
minall=min(prod(:));
```

Data Summarization

```
% Calculate the mean of each column
mu = mean(prod);
muall = mean(prod(:));
% Tell the user
fprintf(' The mean productivity)is: %f\n',
muall);
% Calculate the standard deviation of each column
sigma = std(prod);
sigmaall = std(prod(:));
% Tell the user
fprintf(' The standard deviation is: %f\n',
sigmaall);
```

Data Summarization

The maximum productivity in the entire years and countries

(Million tonnes): 642.283000

The minimum productivity in the entire countries (Million

tonnes): 0.000000

The mean productivity)is: 58.773398

The standard deviation is: 104.133135

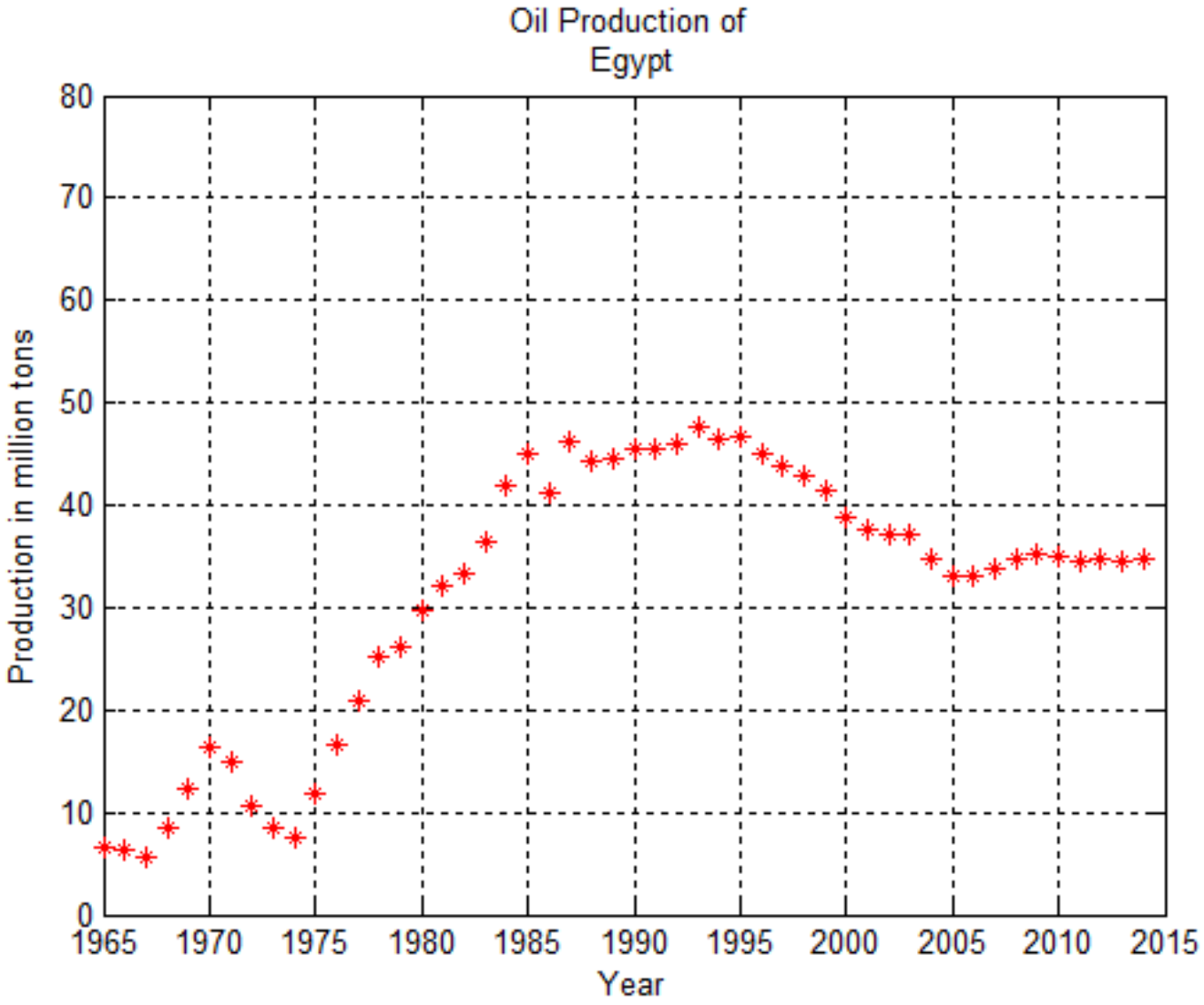
Data Summarization

- **Plot specific record and the overall mean**

```
%Find a specific country
a=strfind(ds.textdata,'Egypt');

% Find the row number of the selected country
ind1=find(~cellfun(@isempty,a));

% Plot the production of the selected county
aganist the years
figure;
plot(ds.data(1,:),ds.data(ind1,:),'r*');
set(gcf,'color','w');
```

Data Summarization

- **Plot specific record and the overall mean**



Data Summarization

- **Plot specific record and the overall mean**

```
% plot the mean
hold on;
% Create a matrix of mean values by
% replicating the mu vector for n rows
MeanMat = repmat(muall,1,length(ds.data(1,:)));
plot(ds.data(1,:),MeanMat,'--');
```

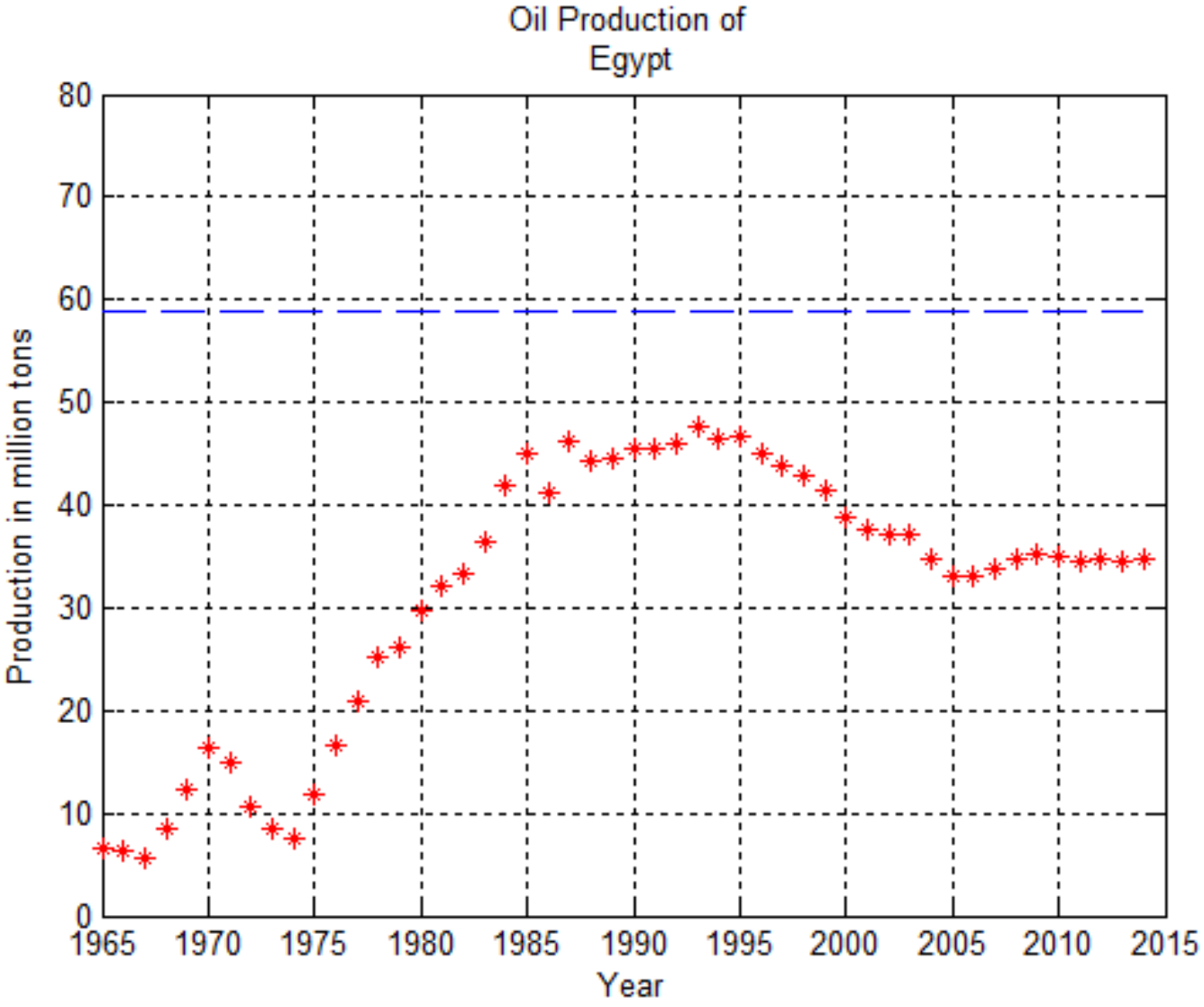
Data Summarization

- **Plot specific record and the overall mean**

```
% Add labels and titles
xlabel('Year');
ylabel('Production in million tons');
title(['Oil Production of ',
ds.textdata(ind1)]);
% Manipulating the axis
axis([1965, 2015, 0, 80]);
grid on;
```

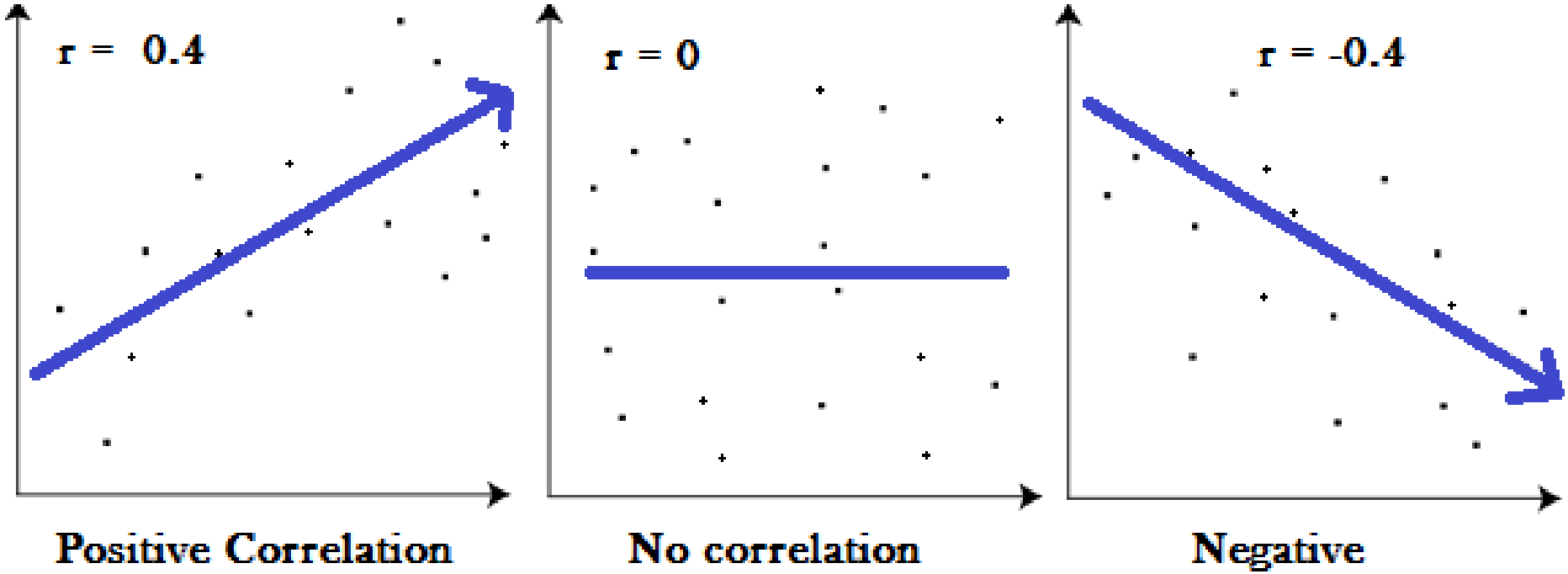
Data Summarization

- **Plot specific record and the overall mean**



Data Summarization

- Calculate the correlation-coefficient



Data Summarization

- Calculate the correlation-coefficient

Correlation coefficients r_k are given by

$$r_k = \frac{\sum_{t=1}^{N}(x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^{N}(x_t - \bar{x})^2}$$

where x_t is a data value at time step t, k is the lag, and the overall mean is given by

$$\bar{x} = \sum_{t=1}^{N} \frac{x_t}{N}$$

Data Summarization

- Calculate the correlation-coefficient

```
%Find the record of Egypt
a=strfind(ds.textdata,'Egypt');
ind1=find(~cellfun(@isempty,a));

%Find the record of India
b=strfind(ds.textdata,'India');
ind2=find(~cellfun(@isempty,b));

%Find the record of India
c=strfind(ds.textdata,'US');
ind3=find(~cellfun(@isempty,c));

prod_Egypt=ds.data(ind1,:);
prod_India=ds.data(ind2,:);
prod_US=ds.data(ind3,:);

% Calculate the correlation-coefficient
coef_EG_IN=corrcoef(prod_Egypt,prod_India);
coef_EG_US=corrcoef(prod_Egypt,prod_US);

imagesc(coef_EG_IN);
colormap(winter); % other color maps: summer, autumn, spring, copper, hsv, gray, etc.

figure;
imagesc(coef_EG_US);
colormap(gray);
```