

Università degli Studi di Pavia
Dipartimento di Ingegneria Industriale e dell'Informazione

Corso di Identificazione dei Modelli e Analisi dei Dati

Random Variables

Prof. Giuseppe De Nicolao, Federica Acerbi, Alessandro Incremona

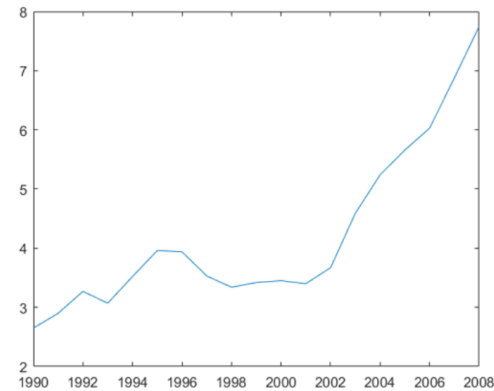
Outline

1. Data visualization basics
2. Random number generation
3. Random variables visualization
4. Outliers
5. Functions of Random Variables

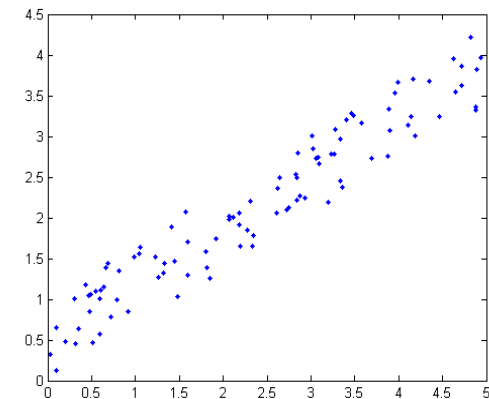
Data Visualization

- Linear Plot
- Scatter Plot
- Histogram
- Box Plot
- etc

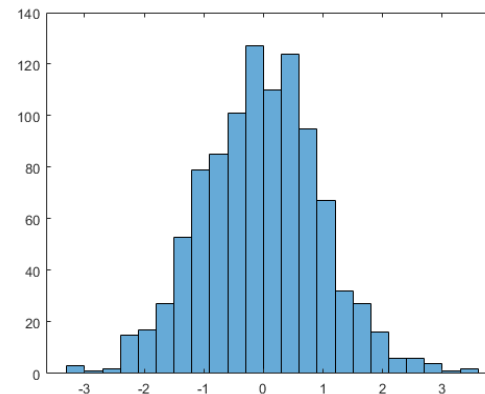
Linear Plot



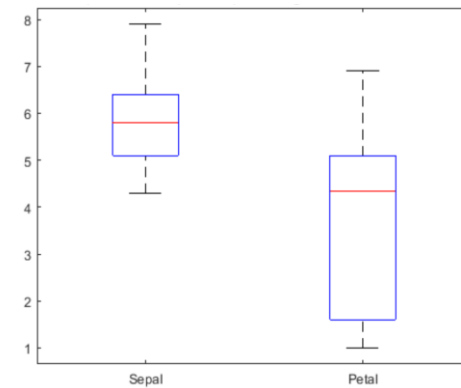
Scatter Plot



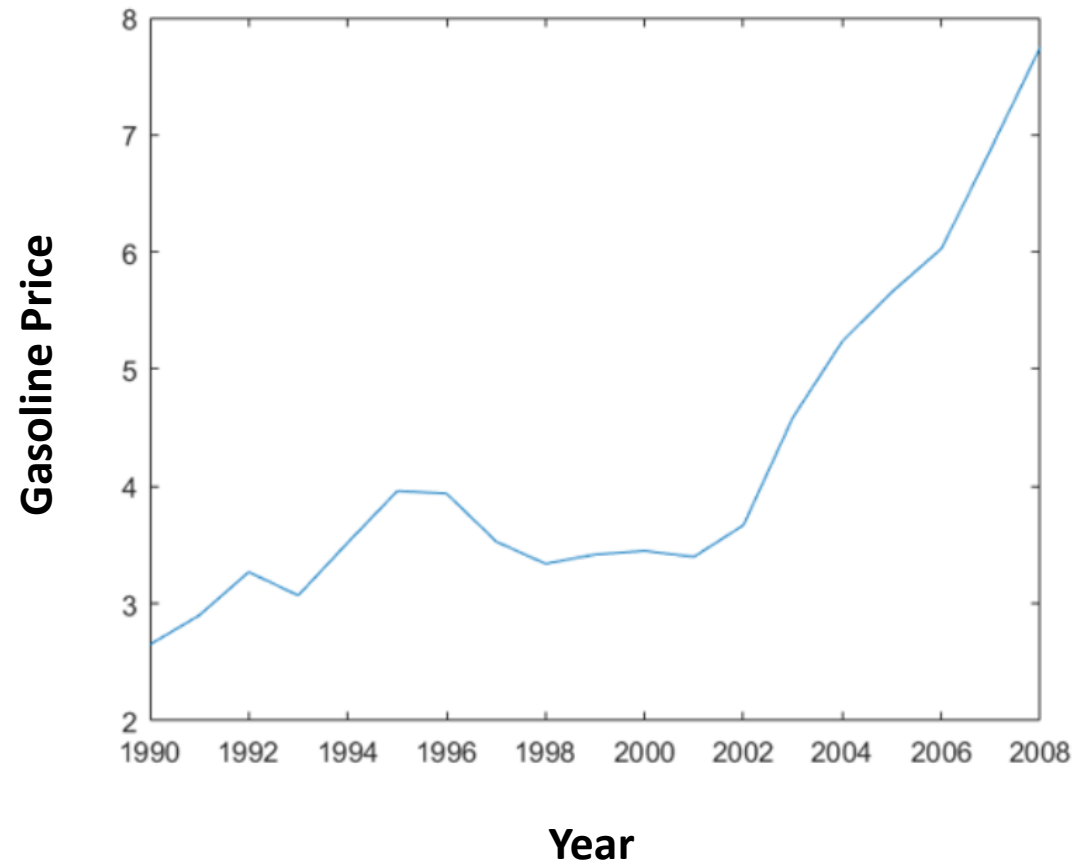
Histogram



Box Plot



Linear Plot



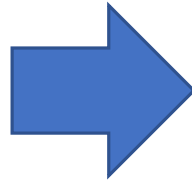
Linear Plot

Germany

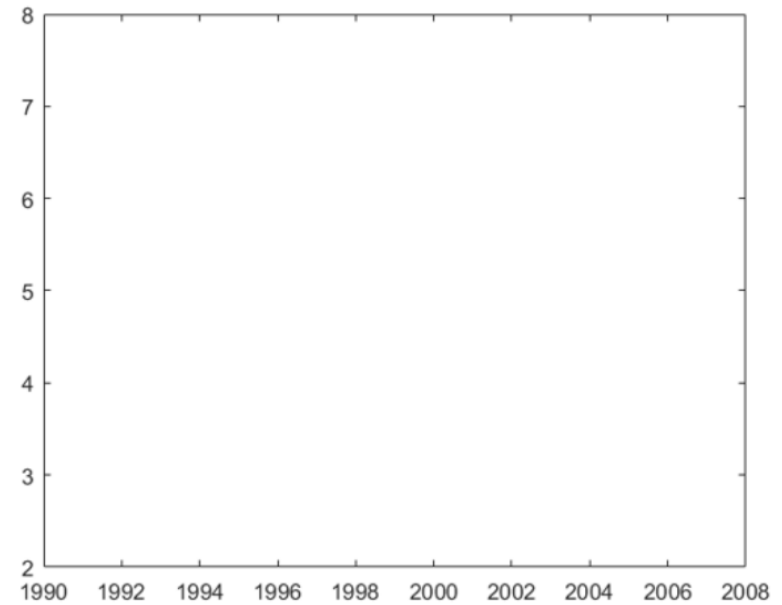
2.65
2.90
3.27
3.07
3.52
3.96
3.94
3.53
3.34
3.42
3.45
3.40
3.67
4.59
5.24
5.66
6.03
6.88
7.75

Year

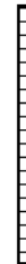
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008



Germany



Year



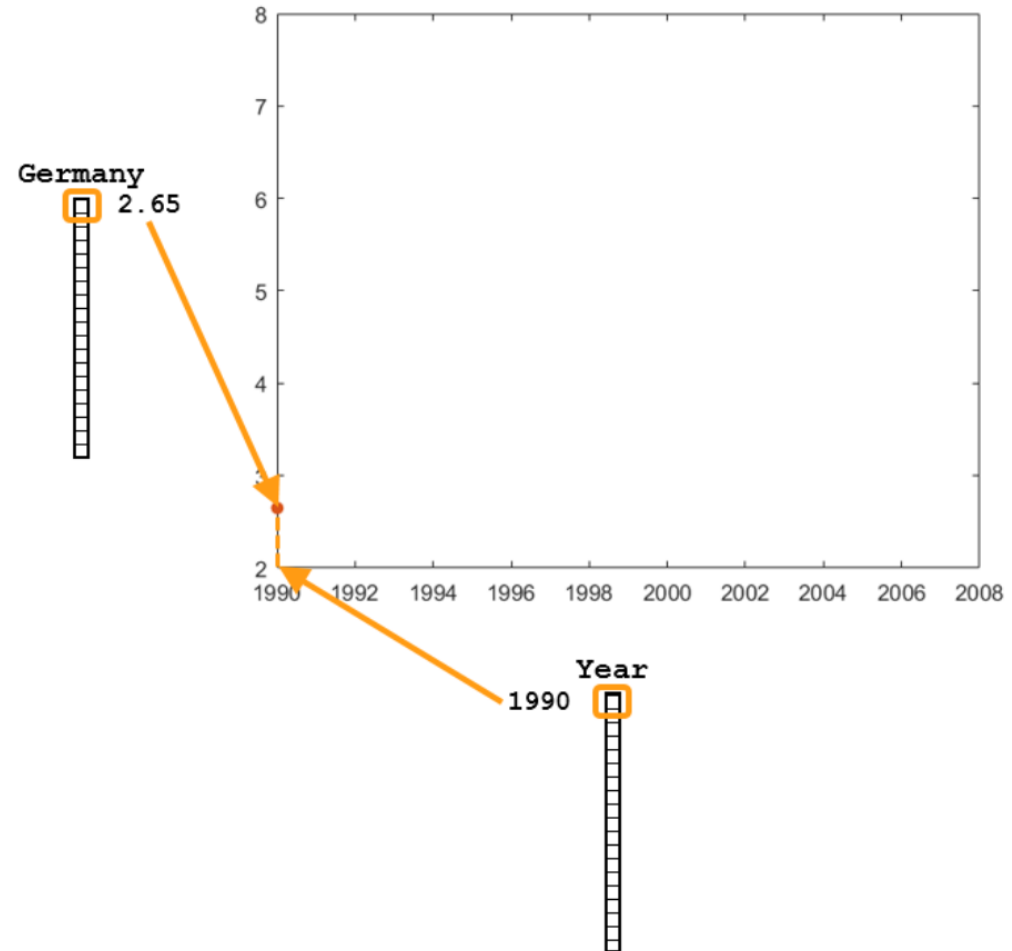
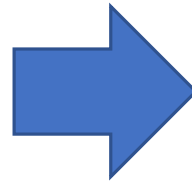
Linear Plot

Germany

2.65
2.90
3.27
3.07
3.52
3.96
3.94
3.53
3.34
3.42
3.45
3.40
3.67
4.59
5.24
5.66
6.03
6.88
7.75

Year

1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008



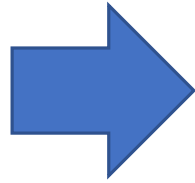
Linear Plot

Germany

2.65
2.90
3.27
3.07
3.52
3.96
3.94
3.53
3.34
3.42
3.45
3.40
3.67
4.59
5.24
5.66
6.03
6.88
7.75

Year

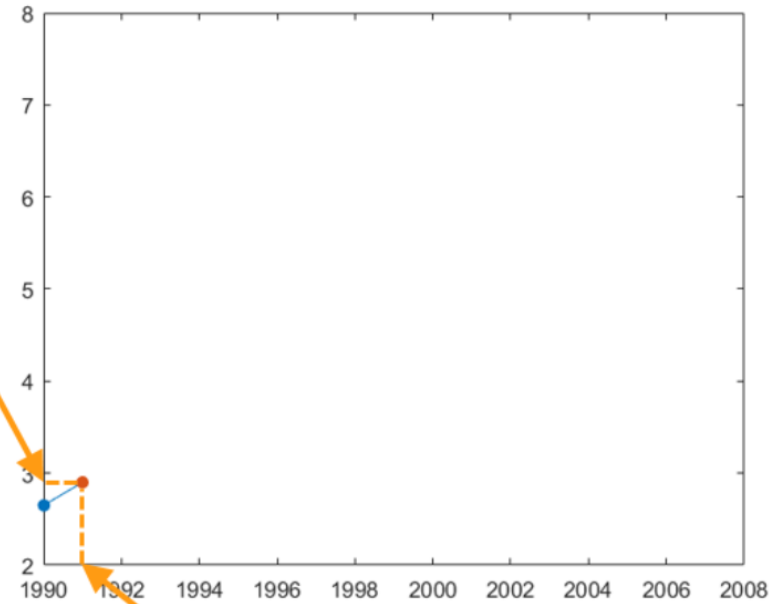
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008



Germany



2.90



Year

1991



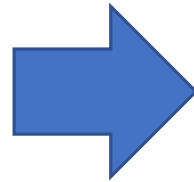
Linear Plot

Germany

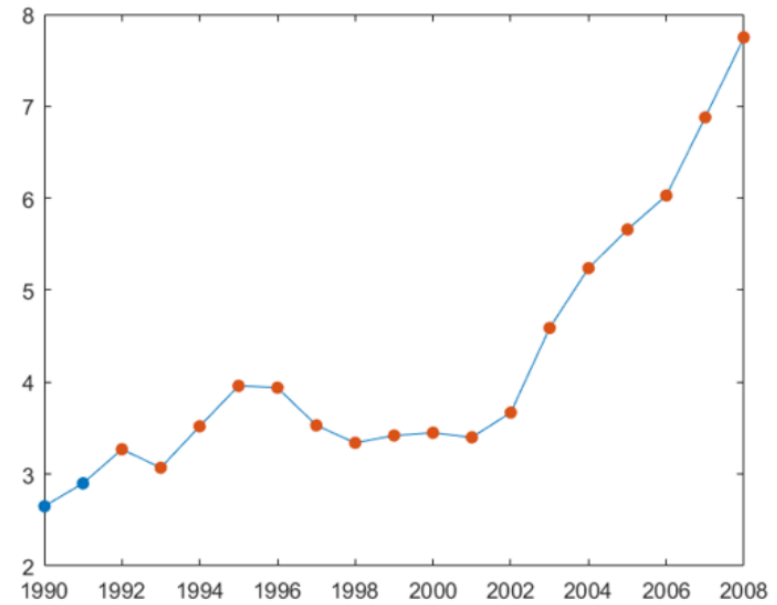
2.65
2.90
3.27
3.07
3.52
3.96
3.94
3.53
3.34
3.42
3.45
3.40
3.67
4.59
5.24
5.66
6.03
6.88
7.75

Year

1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008



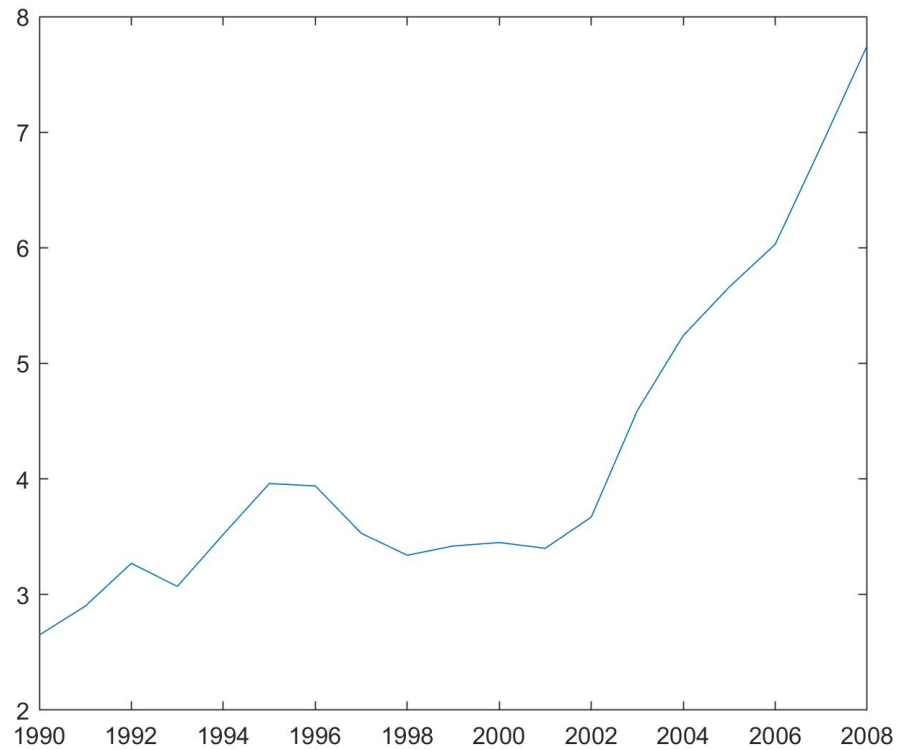
Germany



Year

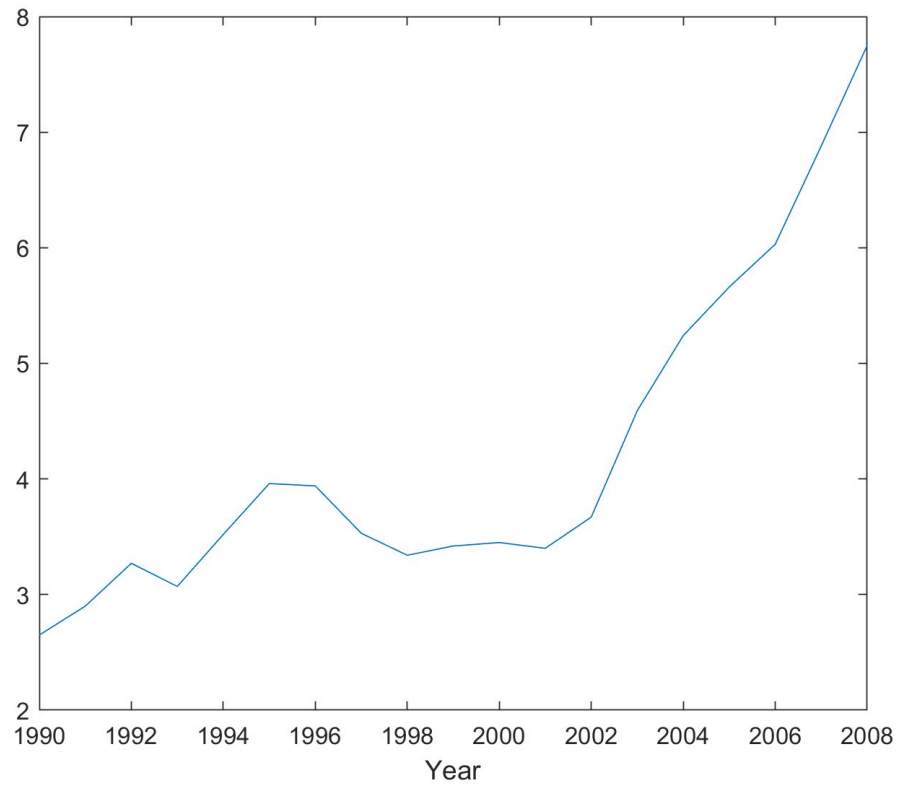


Linear Plot



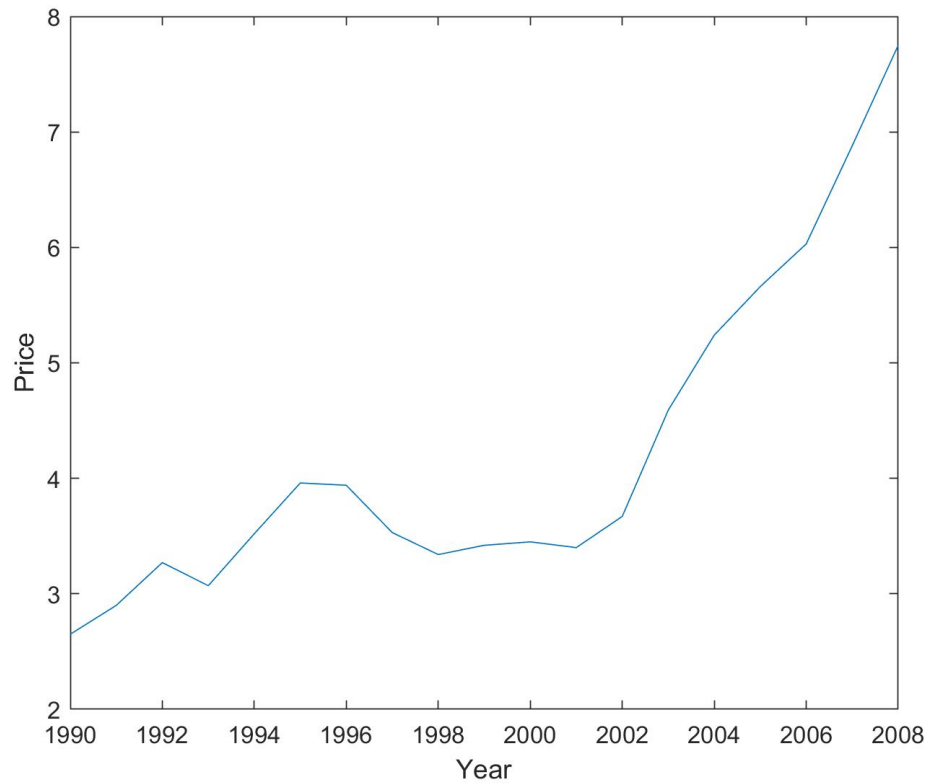
```
>> plot(Year,Germany)
```

Linear Plot



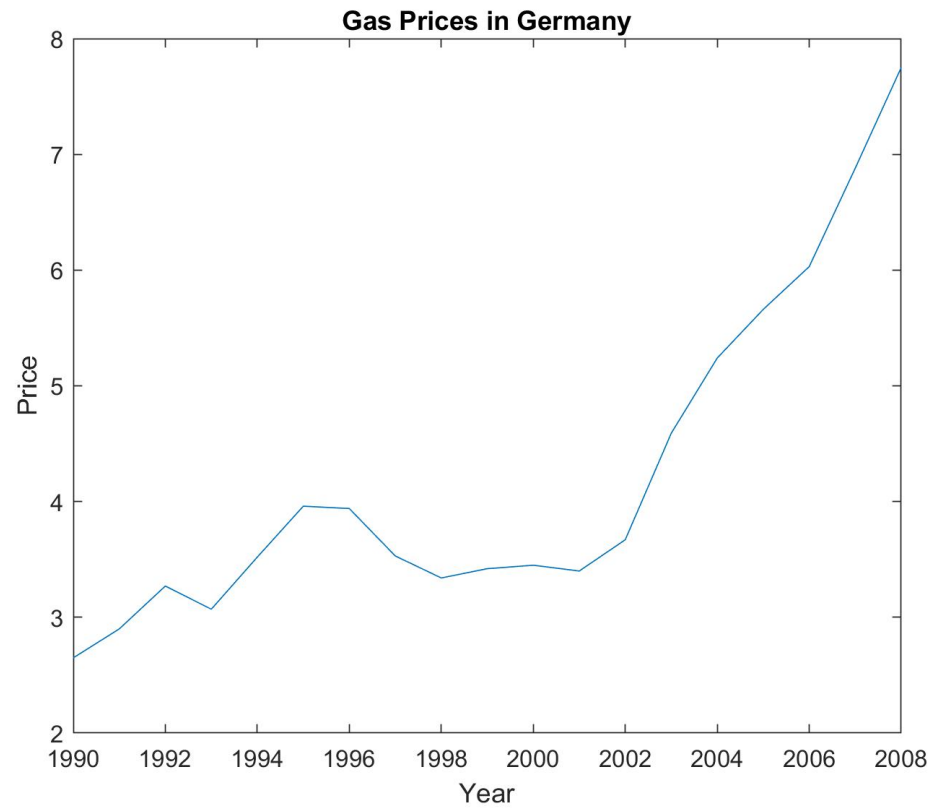
```
>> plot(Year,Germany)  
>> xlabel('Year')
```

Linear Plot



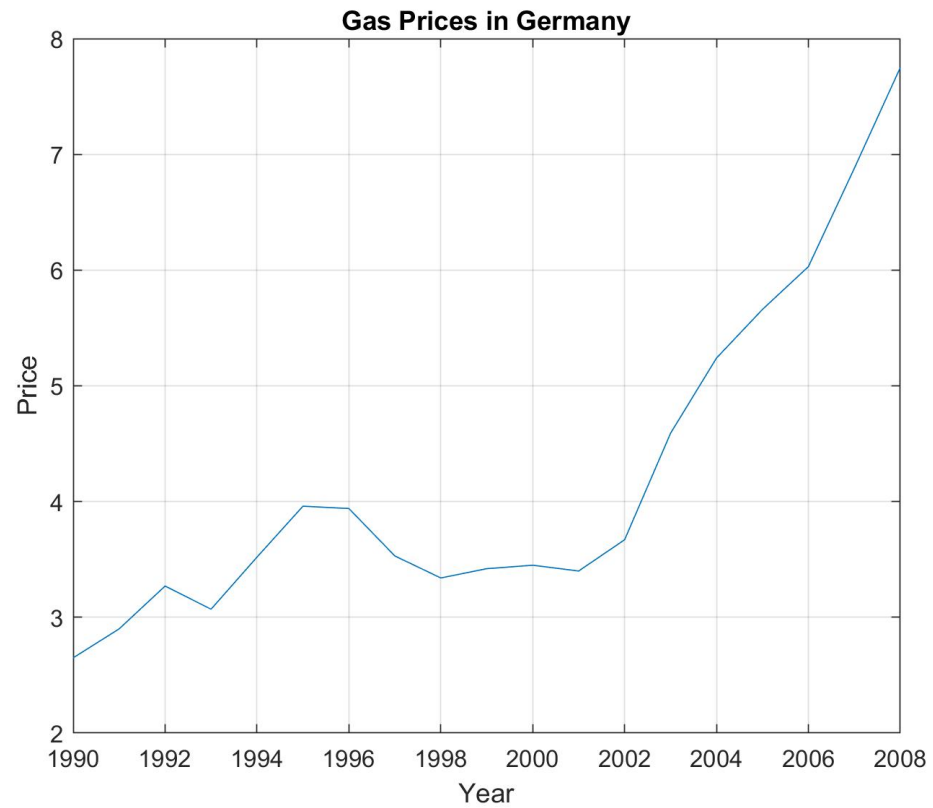
```
>> plot(Year,Germany)  
>> xlabel('Year')  
>> ylabel('Price')
```

Linear Plot



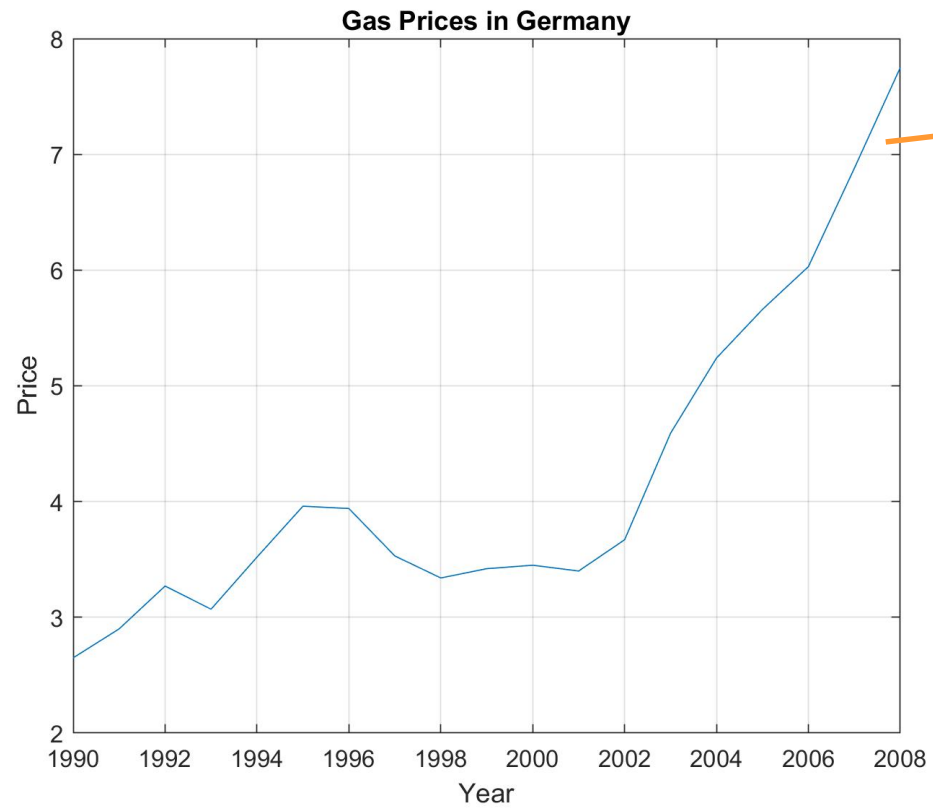
```
>> plot(Year,Germany)
>> xlabel('Year')
>> ylabel('Price')
>> title('Gas Prices in Germany')
```

Linear Plot



```
>> plot(Year,Germany)
>> xlabel('Year')
>> ylabel('Price')
>> title('Gas Prices in Germany')
>> grid on
```

Linear Plot



line



Color

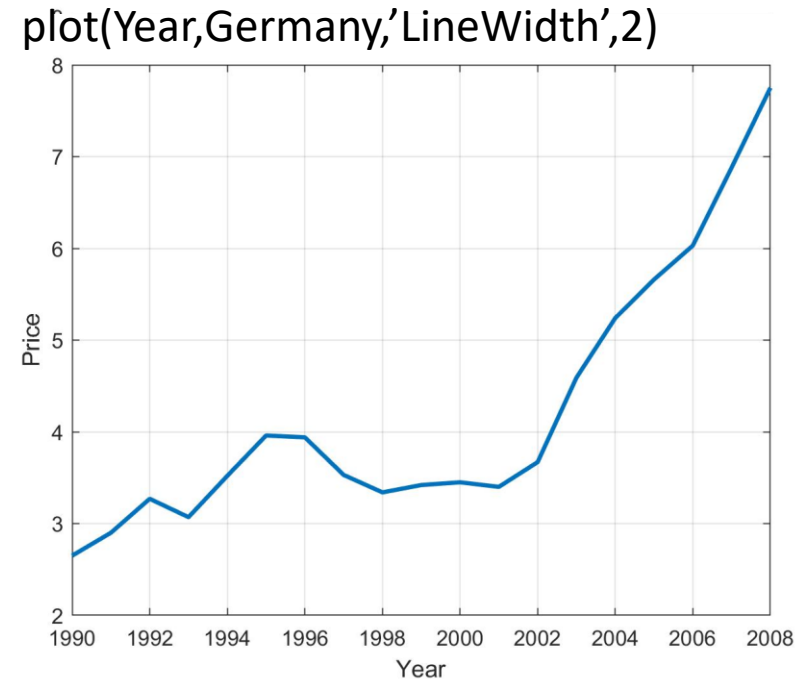
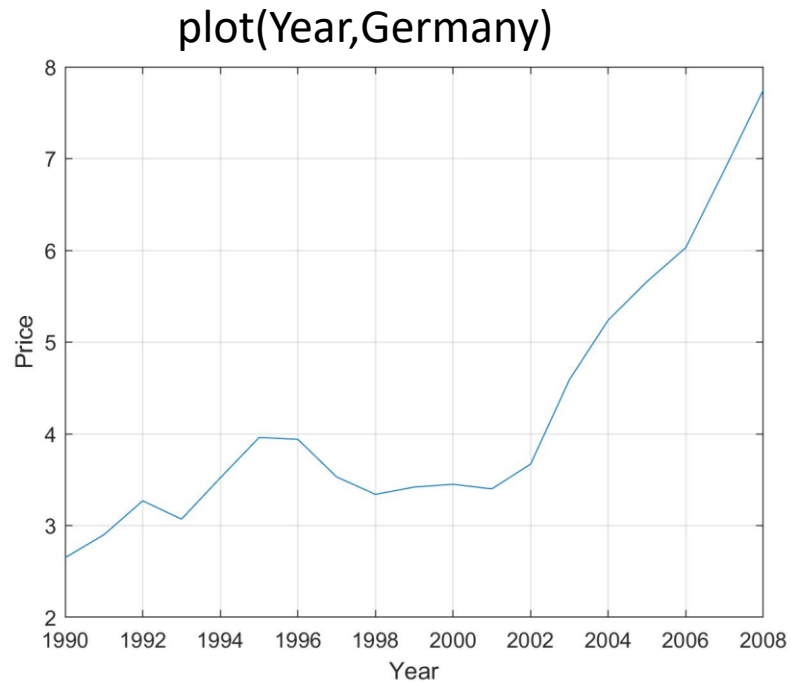
Marker

LineWidth

Linear Plot

NAME,VALUE plot customization:

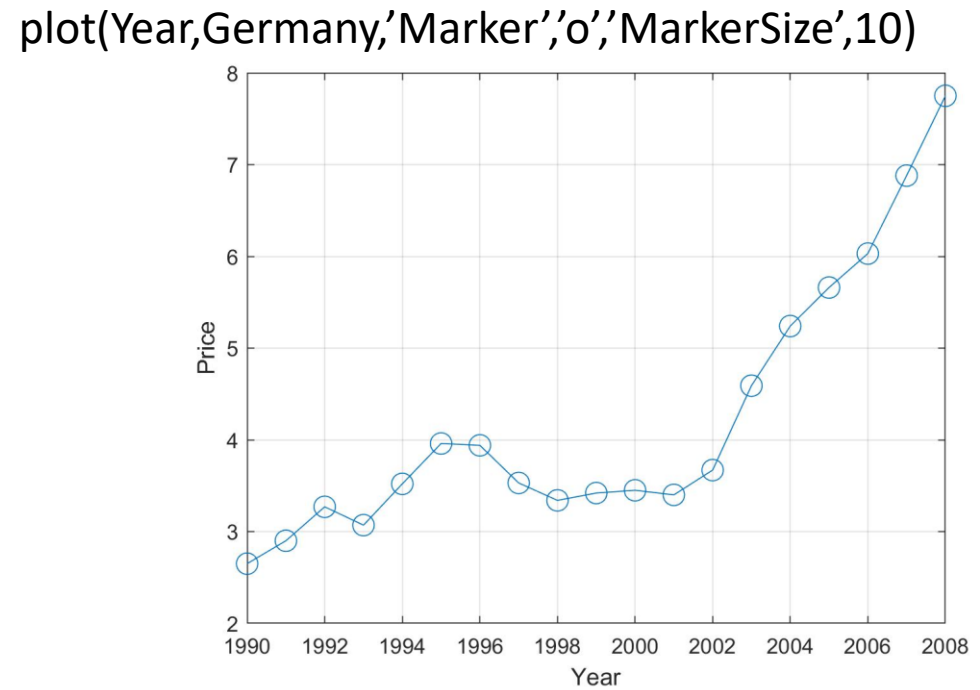
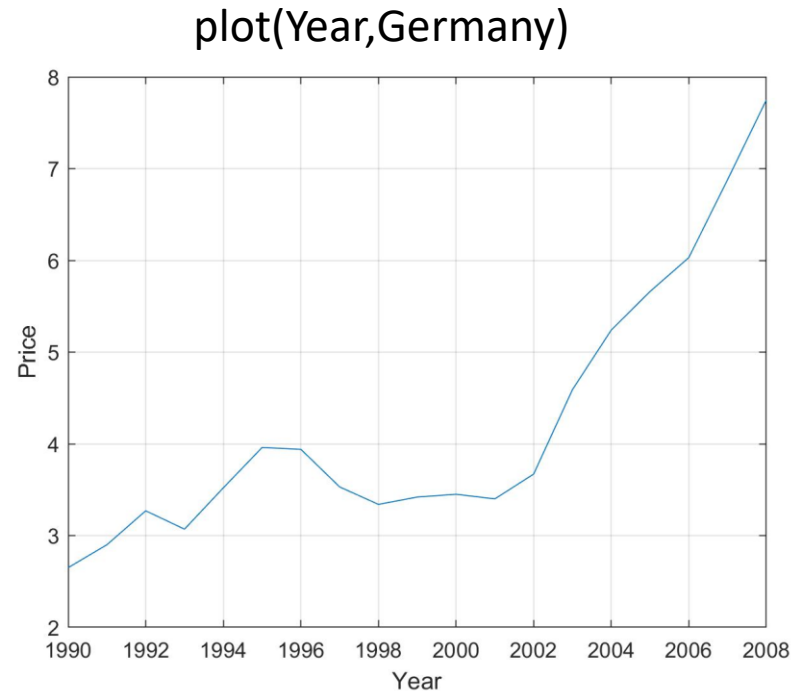
- 'LineWidth' – Line Width (0.5 default)



Linear Plot

NAME,VALUE plot customization:

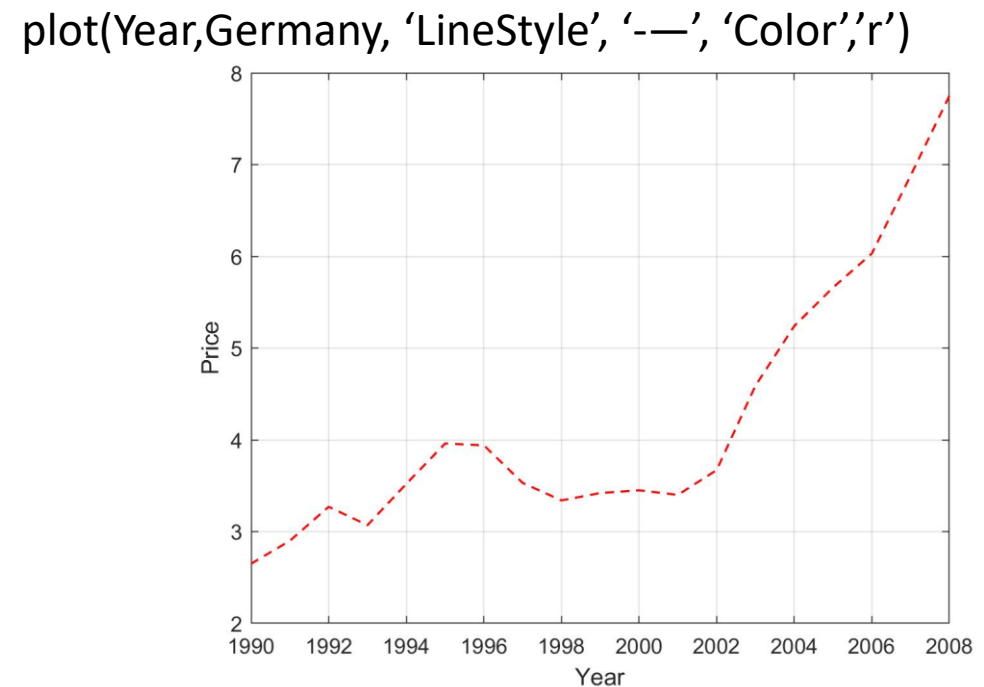
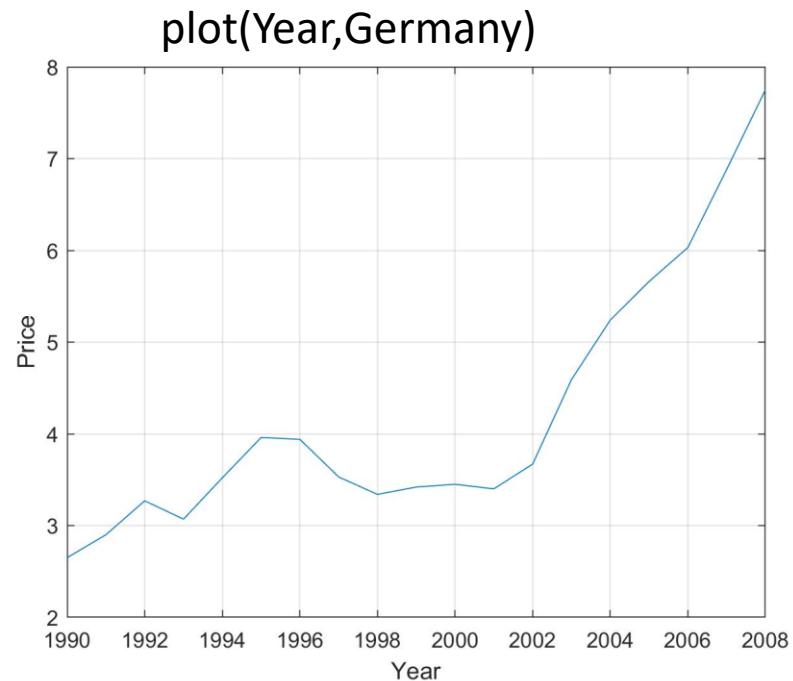
- 'Marker' – Marker Symbol
- 'MarkerSize' - Marker Size (6 default)



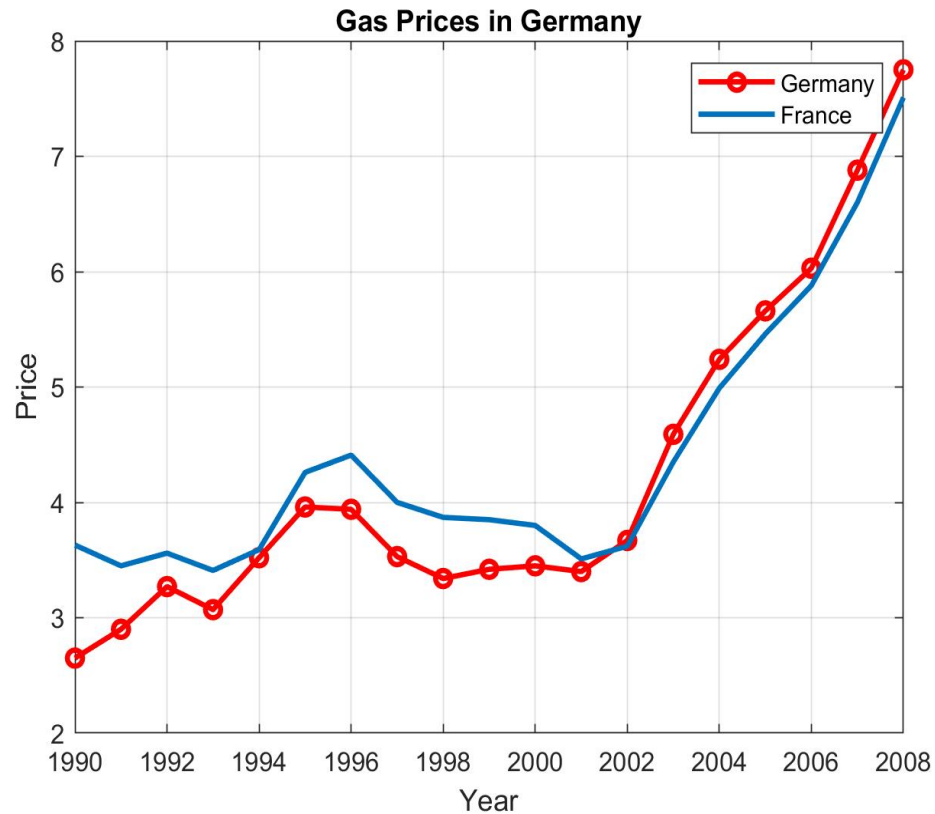
Linear Plot

NAME,VALUE plot customization:

- **'Color'** – Line Color ([0 0.4470 0.7410] default)
- **'LineStyle'** - Line Style ('-' default)



Linear Plot



```
>> plot(Year,Germany,'r','LineWidth',2,'Marker','o')
```

```
>> hold on
```

```
>> plot(Year,France)
```

```
>> xlabel('Year')
```

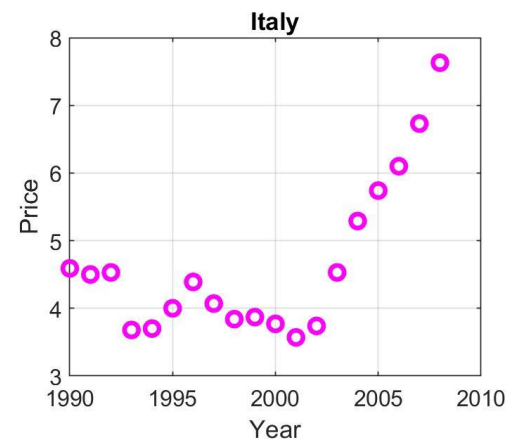
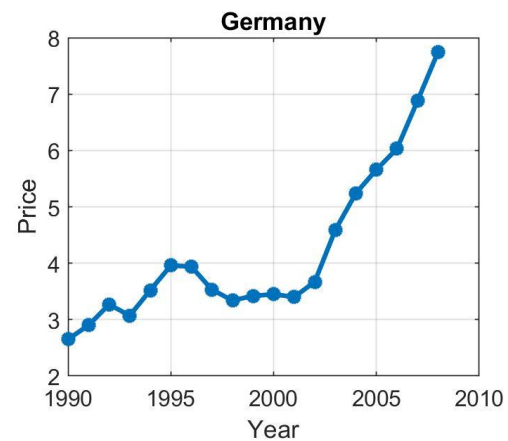
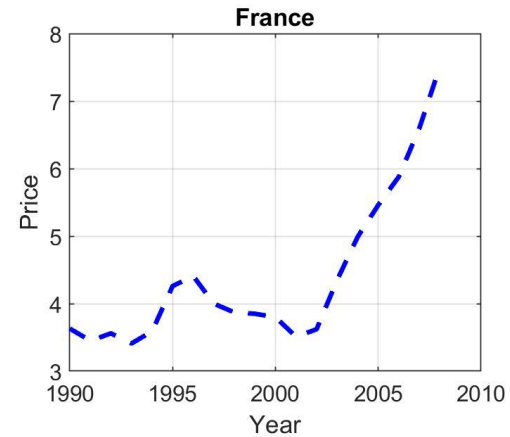
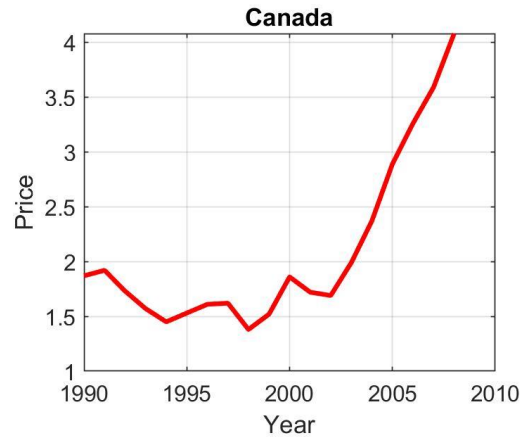
```
>> ylabel('Price')
```

```
>> title('Gas Prices in Germany')
```

```
>> grid on
```

```
>> legend('Germany','France')
```

Linear Plot



```
>> figure(1)
```

```
>> subplot(2,2,1)
```

```
>> plot(Year,Canada,'r','LineWidth',2)
```

```
>> subplot(2,2,2)
```

```
>> plot(Year,France,'--b','LineWidth',2)
```

```
>> subplot(2,2,3)
```

```
>> plot(Year,Germany,'-*','LineWidth',2)
```

```
>> subplot(2,2,4)
```

```
>> plot(Year,Italy,'om','LineWidth',2)
```

Exercises

1. Import the file 'gasprices.csv' in the table dat
2. Plot the data of the gas price in Italy
3. Compare the gas price in Italy and USA plotting them on the same figure
4. Modify the plot adding: axis labels, title, legend.

Random number generation

- When Matlab generates random numbers, they're not truly random; they are based on a pseudo-random number generating algorithm.
- The pseudo-number generating algorithm is based on a starting value, called “seed”.
- If we want to repeat an experiment with random numbers and you want to obtain the same random outcome, you need to “control” the seed using the command `rng`.

Example:

```
rng(2) %fix the seed, for repeatability
y = rand([10, 1]) %create an array 10x1 of pseudorandom values drawn
                from the standard uniform distribution on the open interval(0,1)
```

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'PiecewiseLinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'PiecewiseLinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'PiecewiseLinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'PiecewiseLinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'PiecewiseLinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'Piecewiselinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

Distribution Name	Description
'Beta'	Beta distribution
'Binomial'	Binomial distribution
'BirnbaumSaunders'	Birnbaum-Saunders distribution
'Burr'	Burr distribution
'Exponential'	Exponential distribution
'ExtremeValue'	Extreme Value distribution
'Gamma'	Gamma distribution
'GeneralizedExtremeValue'	Generalized Extreme Value distribution
'GeneralizedPareto'	Generalized Pareto distribution
'HalfNormal'	Half-normal distribution
'InverseGaussian'	Inverse Gaussian distribution
'Logistic'	Logistic distribution
'Loglogistic'	Loglogistic distribution
'Lognormal'	Lognormal distribution
'Multinomial'	Multinomial distribution
'Nakagami'	Nakagami distribution
'NegativeBinomial'	Negative Binomial distribution
'Normal'	Normal distribution
'Piecewiselinear'	Piecewise Linear distribution
'Poisson'	Poisson distribution
'Rayleigh'	Rayleigh distribution
'Rician'	Rician distribution
'Stable'	Stable distribution
'tLocationScale'	t Location-Scale distribution
'Triangular'	Triangular distribution
'Uniform'	Uniform distribution
'Weibull'	Weibull distribution

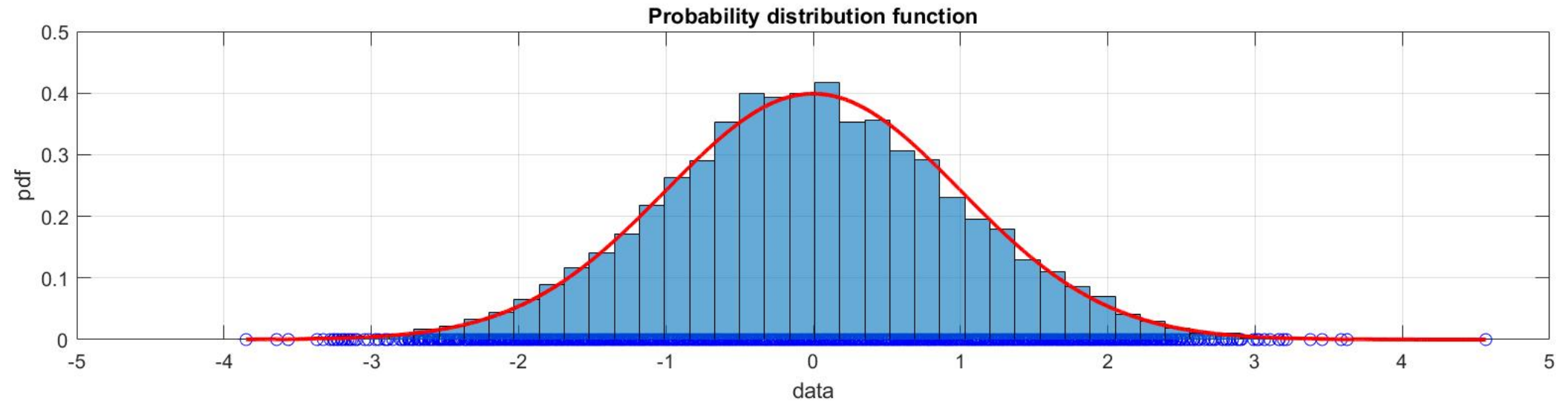
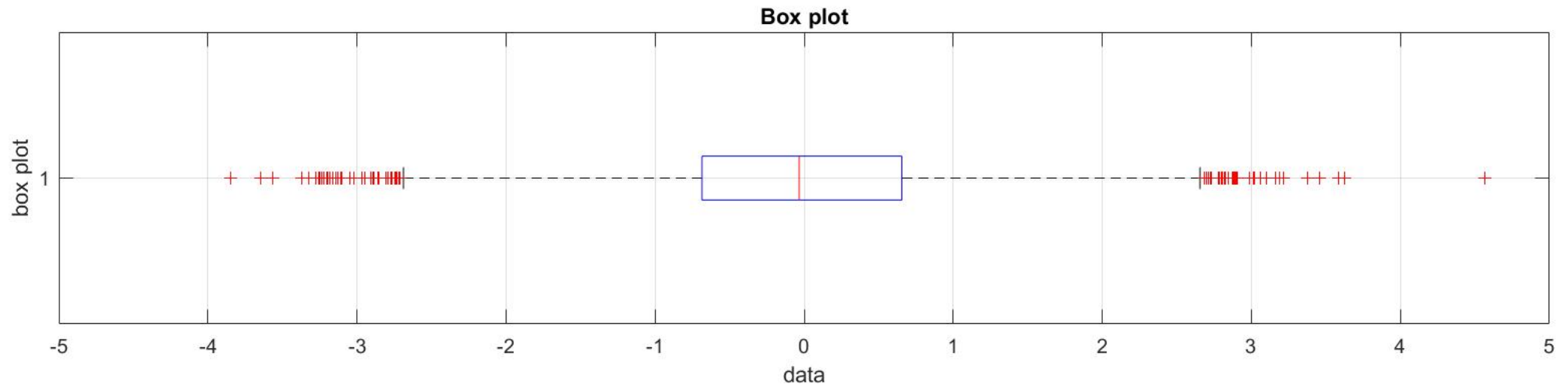
Random number generation

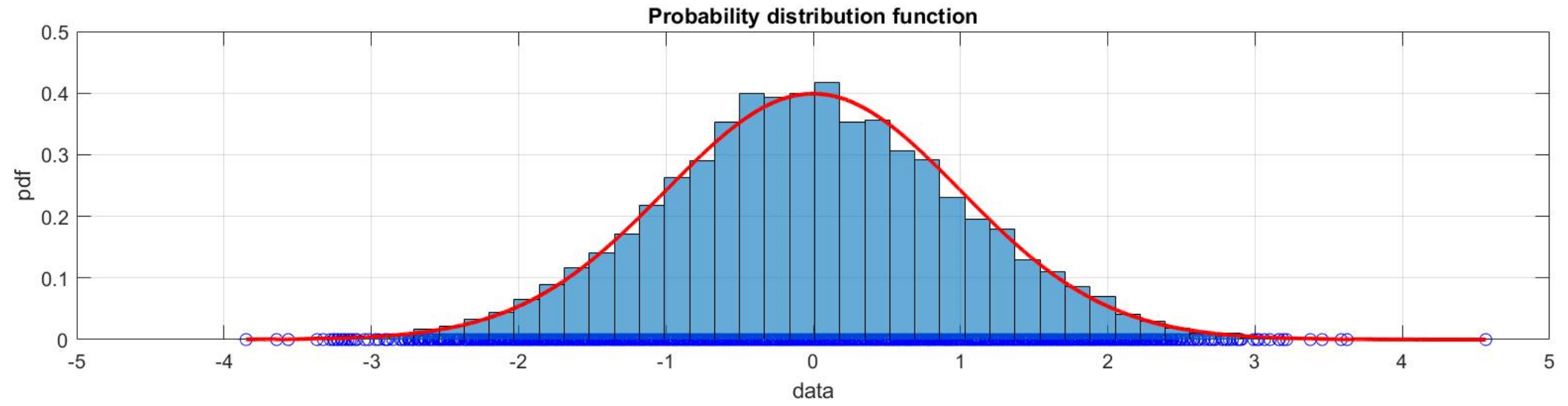
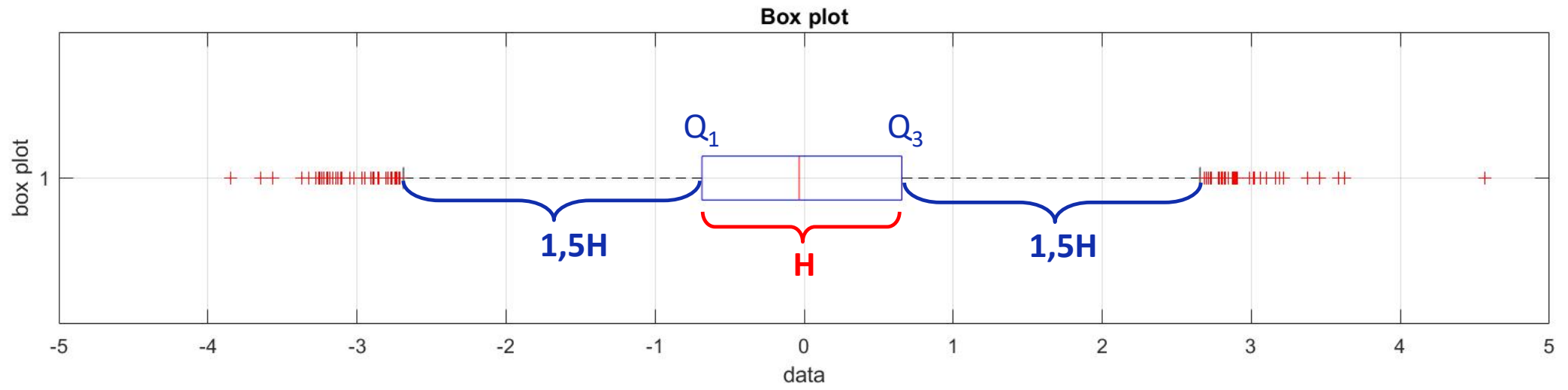
Here are some commands for handling random variables ([see documentation](#) for the syntax):

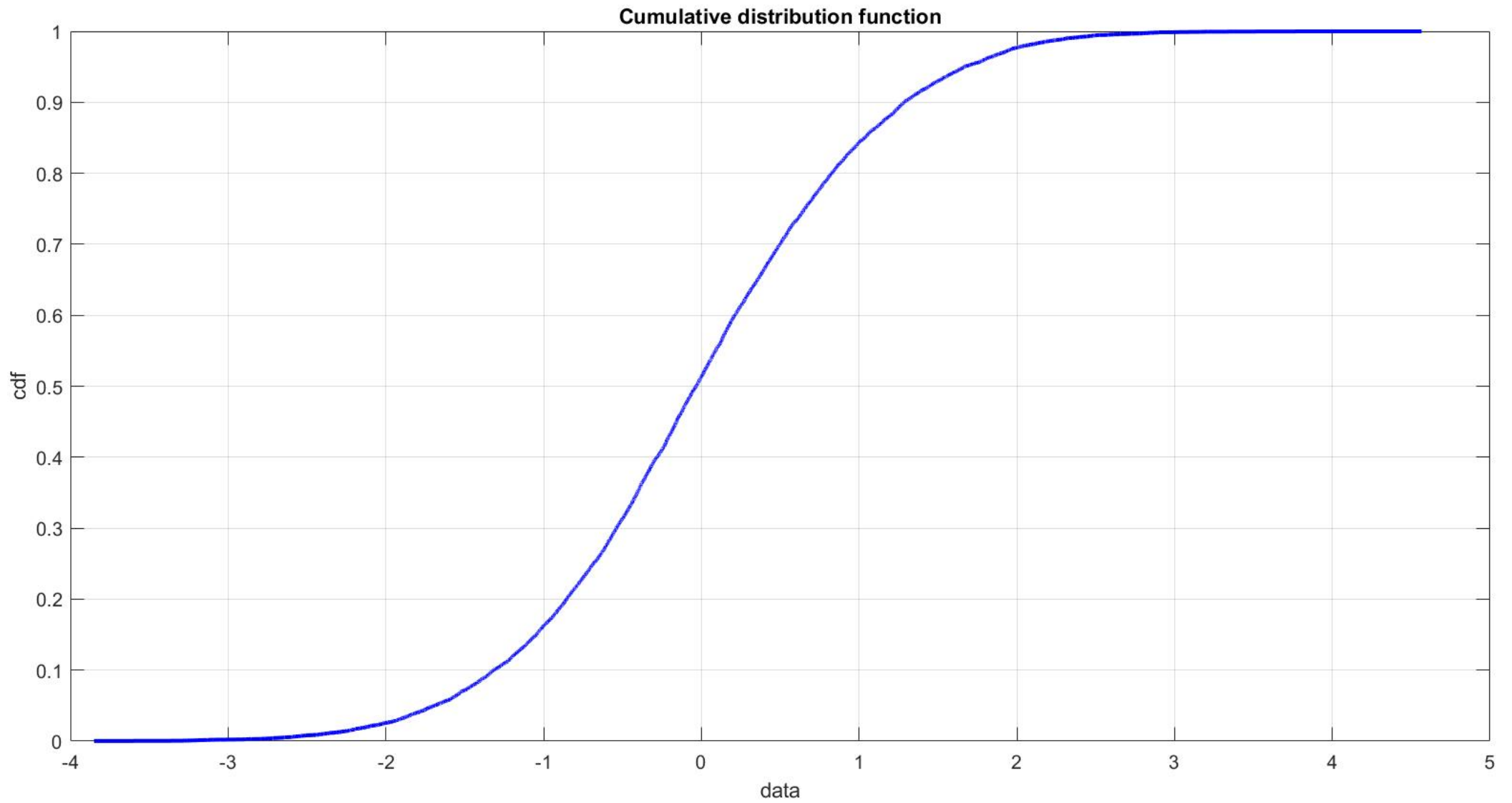
- `makedist`
- `histogram`
- `random`
- `boxplot`
- `ecdf`
- `pdf`
- `randi`
- `randn`
- `rand`

Example

1. Create a script named 'esempio_random_variables.m'.
2. Simulate 10000 random values from a standar Normal distribution.
3. Create a figure with two subplots (vertically aligned).
4. Plot a histogram on the second subplot, to check how the data are distributed.
5. Overlap to the histogram the theoretical pdf of the data.
6. Draw on the first subplot a box plot of the data, horizontally placed.
7. Resize the x-axis limits of the two subplots (using the same limits for both).
8. Draw on another figure, the empirical cumulative distribution function of the data.







Exercise 2

1. Create a script named 'esercizio_random_variables.m'.
2. Simulate 10000 random values from a Exponential distribution with mean 1.
3. Create a figure with two subplots (vertically aligned).
4. Plot a histogram on the second subplot, to check how the data are distributed.
5. Overlap to the histogram the theoretical pdf of the data.
6. Draw on the first subplot a box plot of the data, horizontally placed.
7. Resize the x-axis limits of the two subplots (using the same limits for both).
8. Draw on another figure, the empirical cumulative distribution function of the data

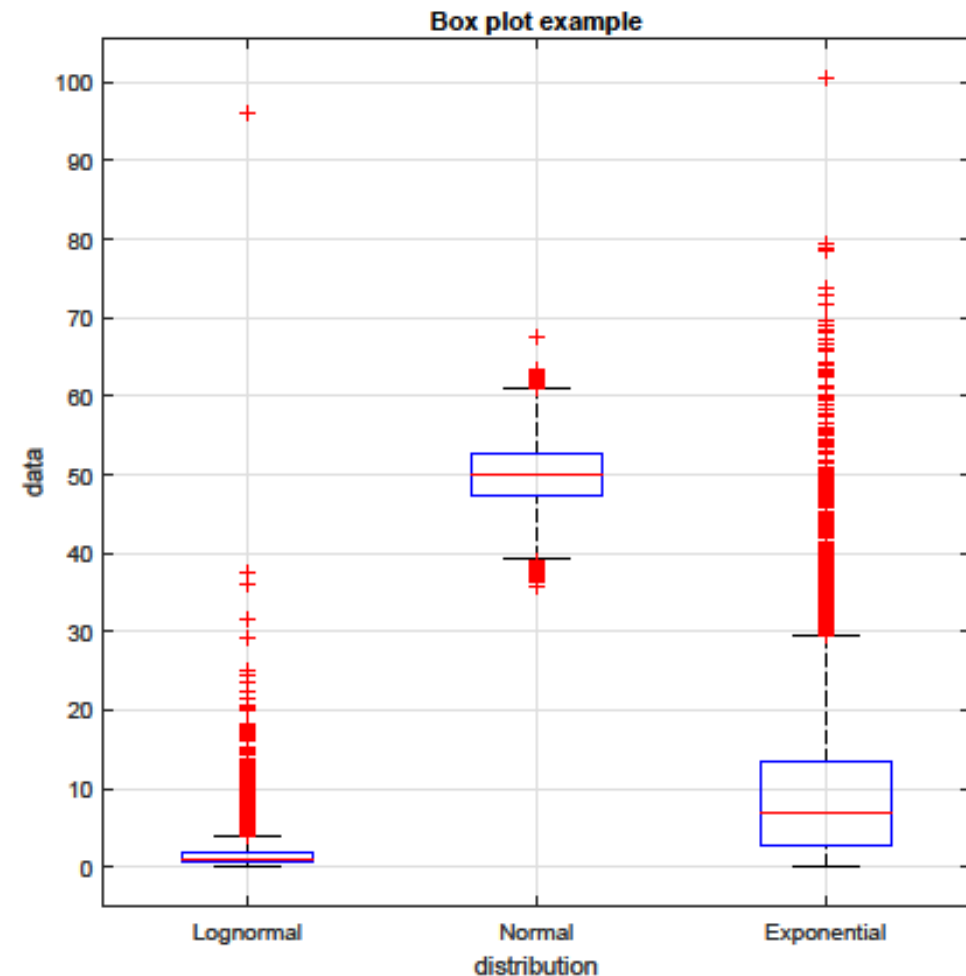
Box plot

```
rng(2) %for repeatability
pd1 = makedist('LogNormal', 0, 1);
pd2 = makedist('Normal', 3, 4);
pd3 = makedist('Exponential', 4);
```

```
Y1 = random(pd1, [10000, 1]);
Y2 = random(pd2, [10000, 1]) ;
Y3 = random(pd3, [10000, 1]) ;
```

```
Y = [Y1 Y2 Y3] ;
```

```
boxplot(Y)
```



Outliers

>> doc `isoutlier`

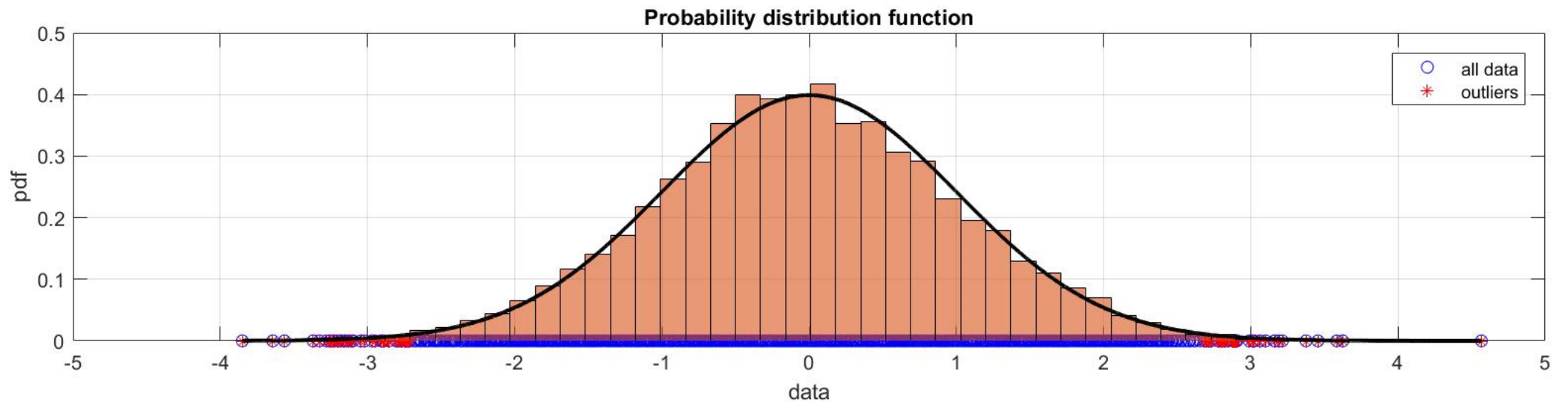
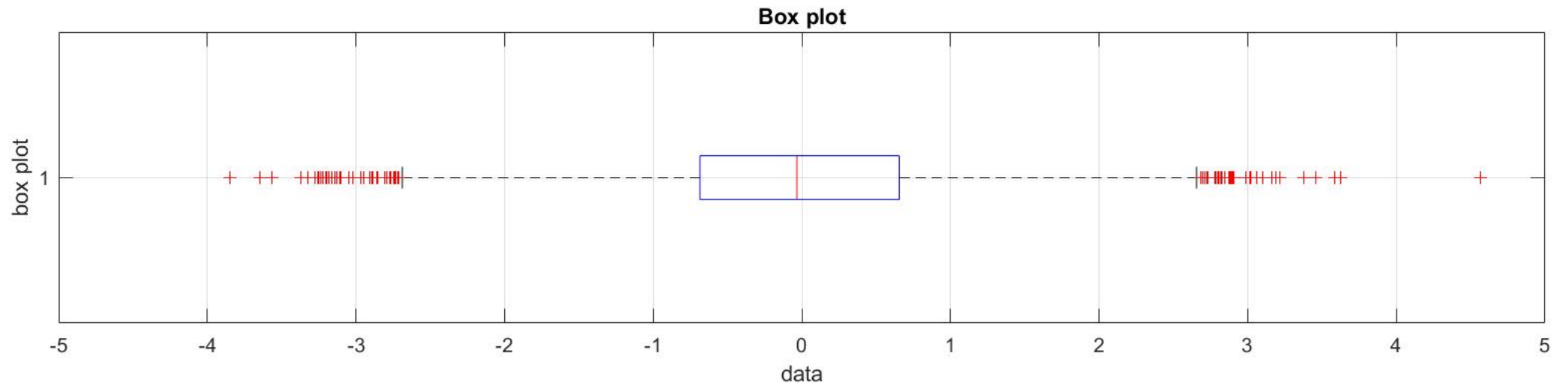
isoutlier

Find outliers in data

Syntax

```
TF = isoutlier(A)  
TF = isoutlier(A,method)
```

“`isoutlier(A)`” returns a logical array whose elements are **true** when an outlier is detected in the corresponding element of “A”.



Exercise 3

1. Open the script 'esercizio_random_variables.m'.
2. Detect the outliers using the `isoutlier` command.
3. On the second plot, overlap the outliers data (use a different color in order to distinguish them).

Functions of random variables

Example:

$$Y = 2X + 10 \quad \text{with} \quad X \sim \mathcal{N}(0, 1)$$

- The value of $f(X)$ depends on the value of X (random variable)
- ... and therefore on the result of an experiment!

In this case, Y is called **stochastic function**:

- It is a **random variable**
- It can be evaluated **running an experiment**
- It has its own **probability distribution**

Exercise 4

Example:

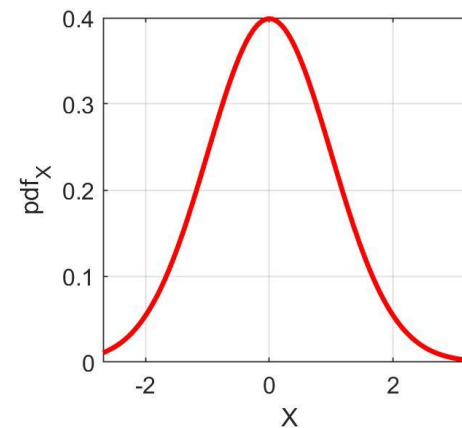
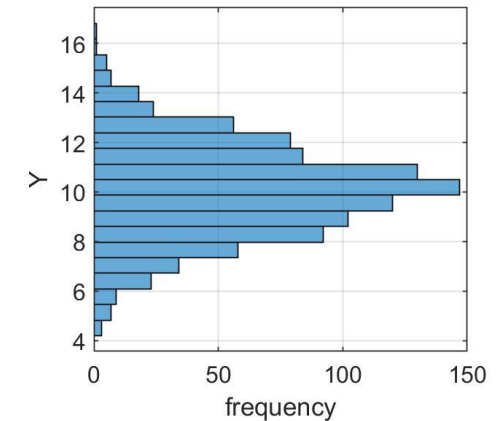
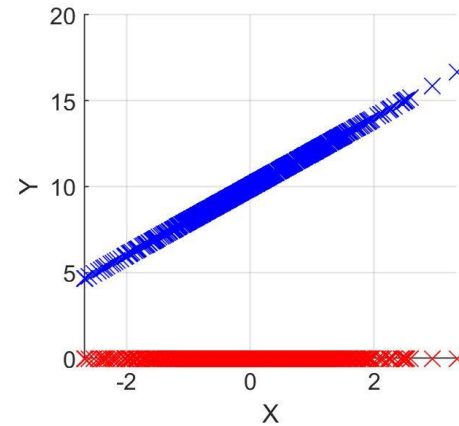
$$Y = 2X + 10 \quad \text{with} \quad X \sim \mathcal{N}(0, 1)$$

1. Generate X as a [1000x1] vector of random numbers sampled from a **normal** distribution with **mean = 0** and **variance = 1**
2. Draw the theoretical probability distribution of X
3. Compute $Y = f(X)$ and approximate its probability distribution using the **histogram** function

Exercise 4

```
pd = makedist('Normal', 0, 1);  
x = random(pd, 1000, 1);  
y = 2*x + 10;
```

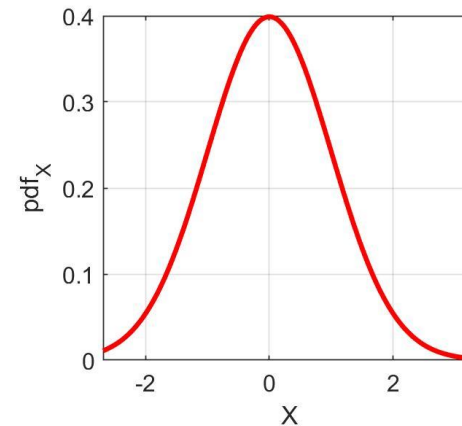
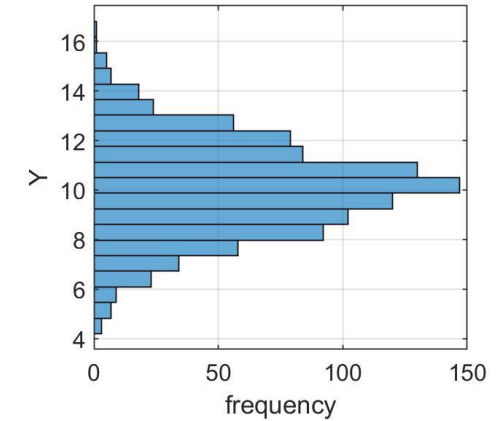
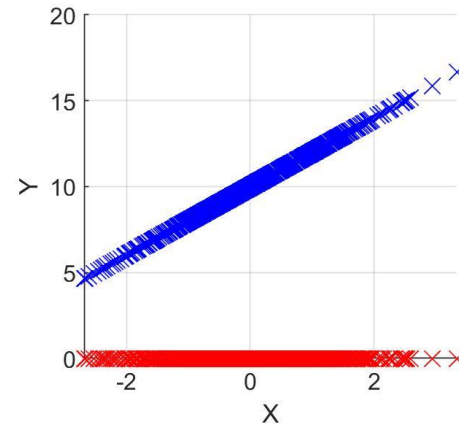
```
subplot(2,2,1)  
scatter(x, zeros(length(x), 1), 'x', 'r')  
grid on  
hold on  
scatter(x, y, 'x', 'b')  
grid on  
xlabel('X')  
ylabel('Y')  
xlim([min(x),max(x)])
```



Exercise 4

```
subplot(2,2,3)
xx = linspace (min(x),max(x),length(x));
plot(xx,pdf(pd,xx),'r','LineWidth',2);
grid on
xlabel('X')
ylabel('pdf_{x}')
xlim([min(x),max(x)])
```

```
subplot(2,2,2)
histogram(y,20)
grid on
xlabel('X')
ylabel('Y')
xlim([0,max(y)])
set(gca , 'view', [90 90])
```



Exercise 5

1. Replicate the exercise n° 4 with
 - X : a vector (dimension [1000x1]) of random numbers sampled from a **lognormal** distribution with mean = 0 and standard deviation = 1
 - $Y = 2 \log(X) + 10$

Reference Documentation:

- <https://it.mathworks.com/>
- <http://sisdin.unipv.it/labsisdin/teaching/courses/imadlt/esercitazioni>

