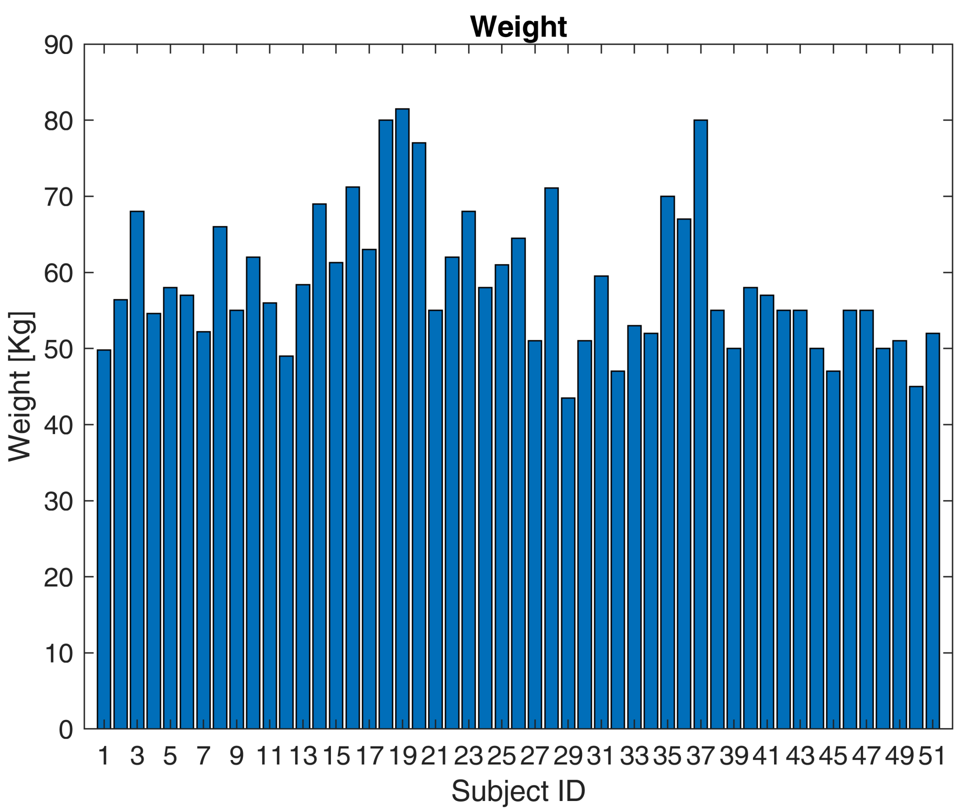
```matlab
clear all, clc,

% Load Data
load('Data.mat');

% Number of subjects
N = length(Data.Weight);
```

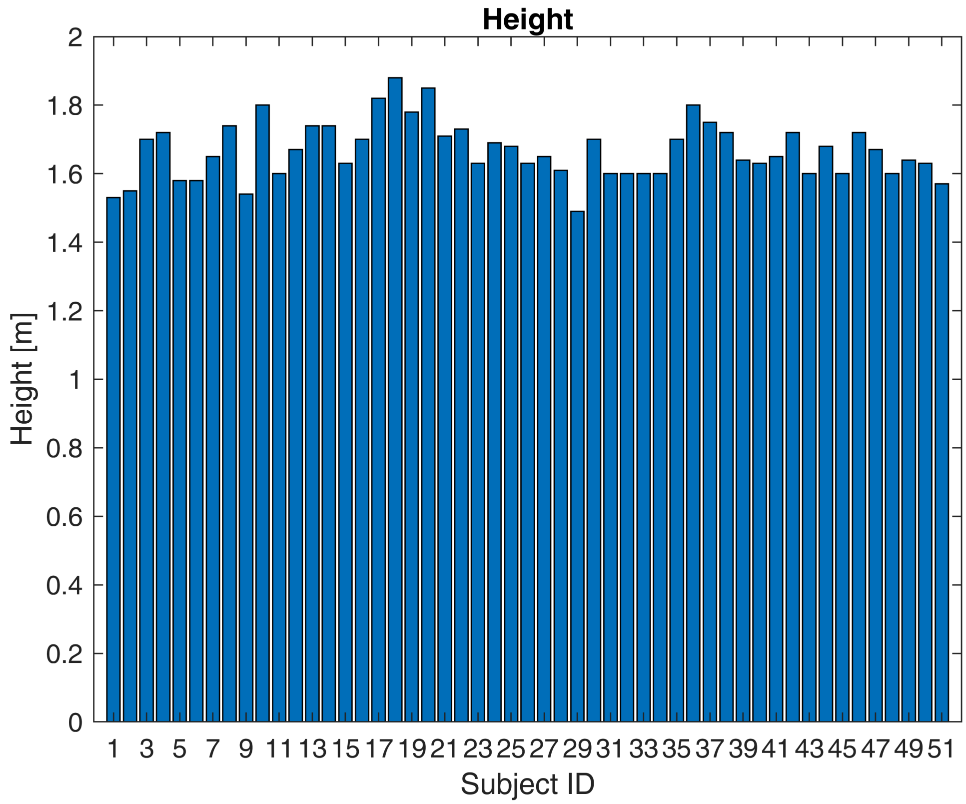
```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                   PLOTTING DATA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Plot of Weight Data
figure('NumberTitle', 'off', 'Name', 'Weight');
bar([1:N], Data.Weight);
title('Weight');
xlabel('Subject ID');
xticks([1:2:N]);
ylabel('Weight [Kg]');
```



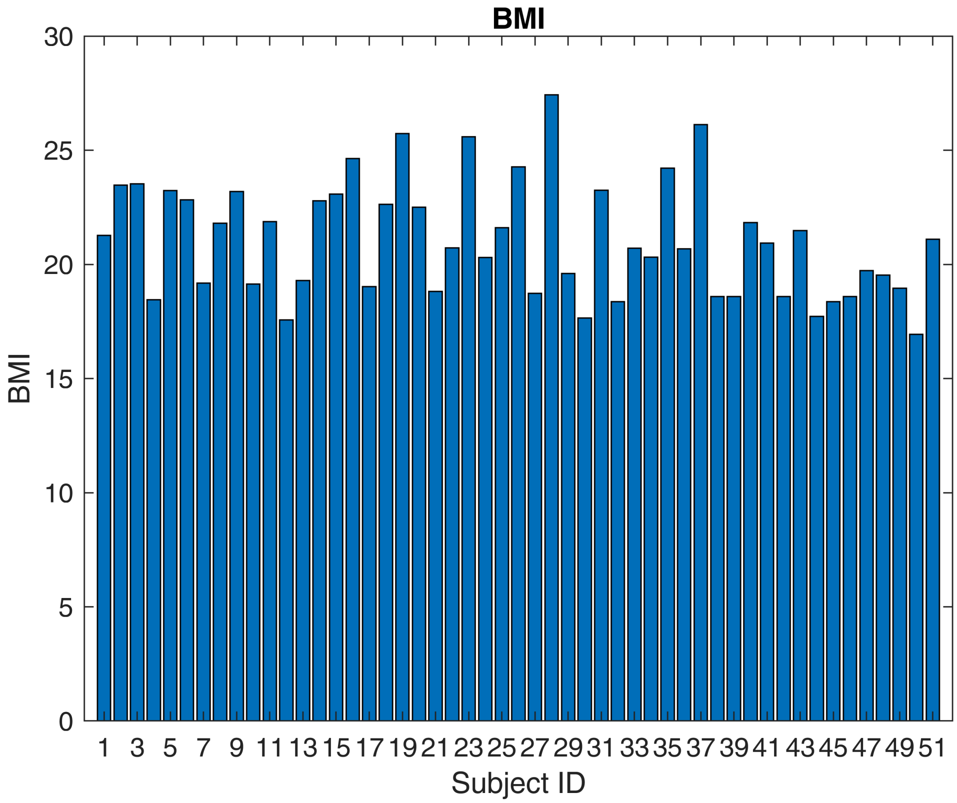
Plot of Height Data

```matlab
figure('NumberTitle', 'off', 'Name', 'Height');
bar([1:N], Data.Height);
```

```
title('Height');
xlabel('Subject ID');
xticks([1:2:N]);
ylabel('Height [m]');
```

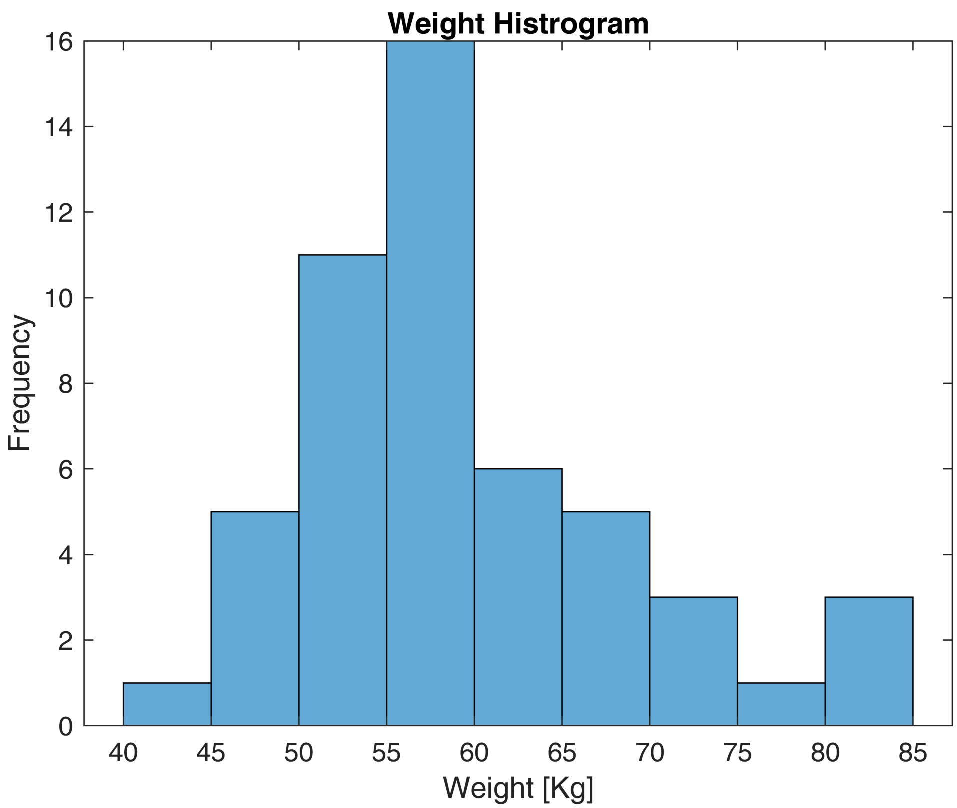



```
% Compute Body Mass Index - BMI
Data.BMI = Data.Weight./Data.Height.^2;

% Plot of BMI Data
figure('NumberTitle', 'off', 'Name', 'BMI');
bar([1:N], Data.BMI);
title('BMI');
xlabel('Subject ID');
xticks([1:2:N]);
ylabel('BMI');
```

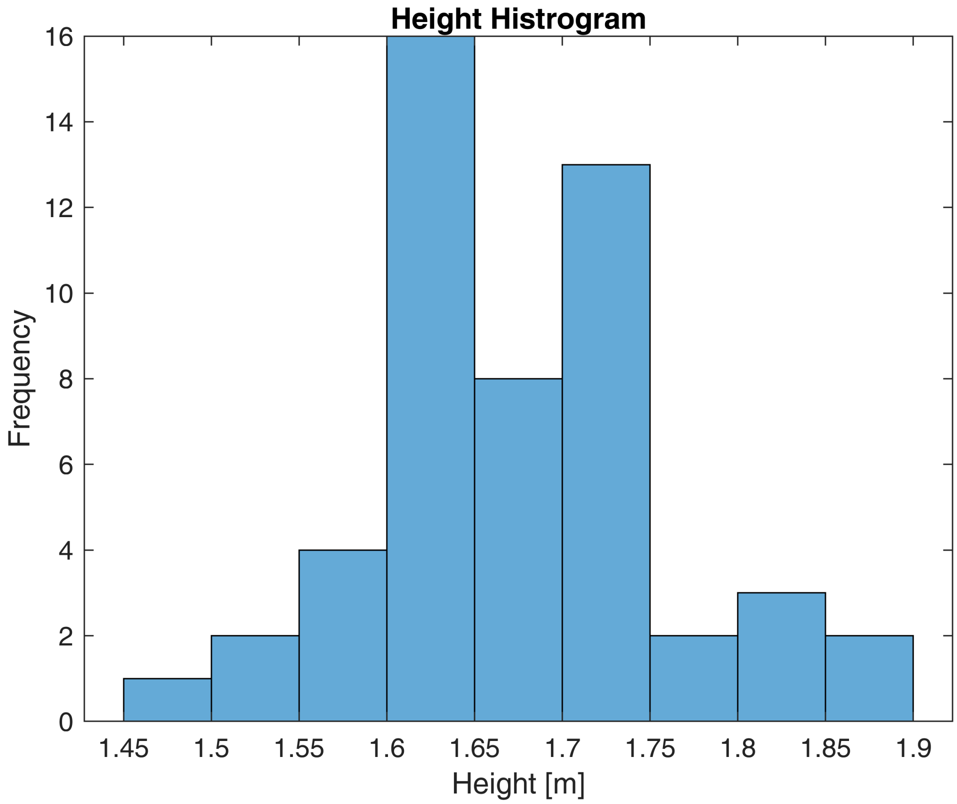


Frequency Distribution

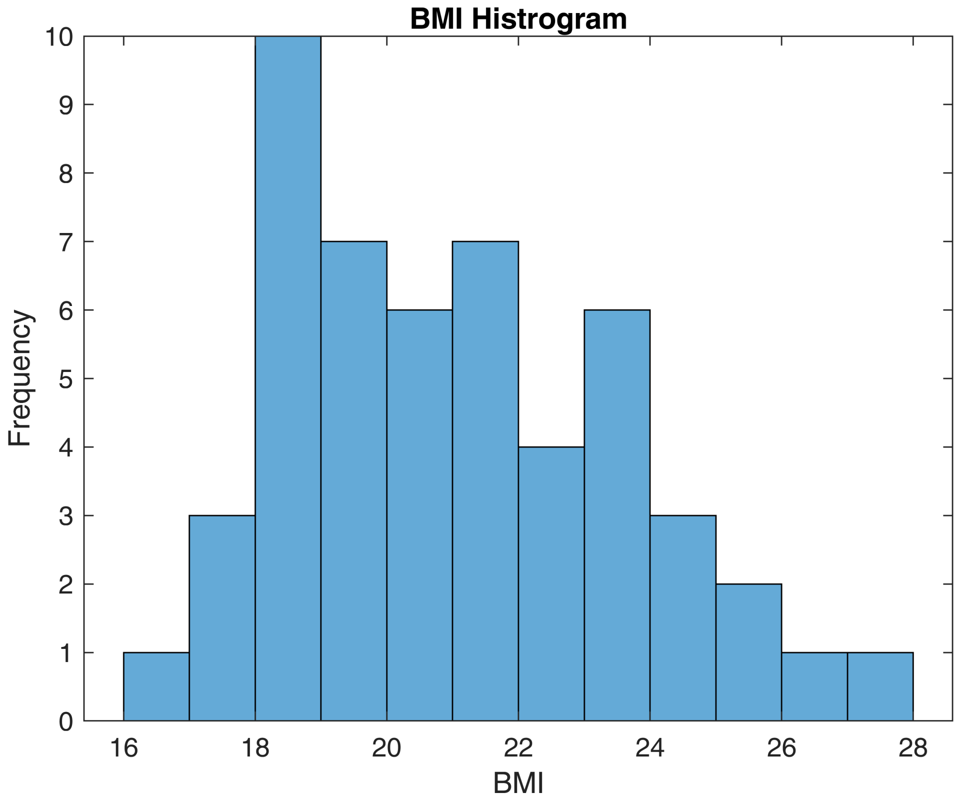
```
% Plot Weight Frequency Distribution
figure('NumberTitle', 'off', 'Name', 'Weight Histrogram');
histogram(Data.Weight,'BinEdges',40:5:85)
title('Weight Histrogram');
xlabel('Weight [Kg]');
ylabel('Frequency');
```




```
% Plot Height Frequency Distribution
figure('NumberTitle', 'off', 'Name', 'Height Histrogram');
histogram(Data.Height,'BinEdges',1.45:0.05:1.9)
title('Height Histrogram');
xlabel('Height [m]');
ylabel('Frequency');
```




```
% Plot BMI Frequency Distribution
figure('NumberTitle', 'off', 'Name', 'BMI Histrogram');
histogram(Data.BMI,'BinEdges',16:1:28)
title('BMI Histrogram');
xlabel('BMI');
ylabel('Frequency');
```




```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                         MEASURES OF CENTRAL TENDENCY
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% https://it.mathworks.com/help/stats/measures-of-central-tendency.html

% In this section we will see some statistics that can be used to
% represent the “center” of the distribution.

% The phrase “measures of central tendency”, or sometimes
% “measures of location”, % refers to the set of measures that reflect
% where on the scale the distribution is centered.

%-------------------------
% MODE
%-------------------------
% The mode can be defined simply as the most common score, that is,
% the score obtained from the largest number of subjects.
% Thus, the mode is that value of X that corresponds to the highest point
% on the distribution.

freq_Weight = histcounts(Data.Weight,40:5:85);
freq_Height = histcounts(Data.Height,1.45:0.05:1.9);
freq_BMI    = histcounts(Data.BMI,16:1:28);
Data.Weight_Mode = max(freq_Weight);
Data.Height_Mode = max(freq_Height);
```

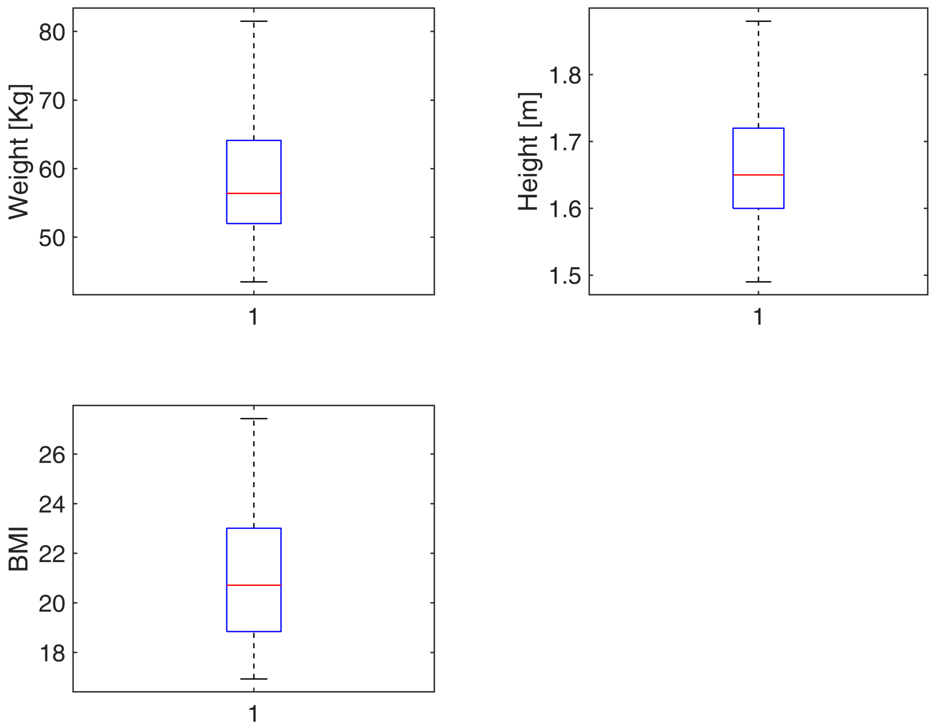
```
Data.BMI_Mode    = max(freq_BMI);

%------------------------
% MEDIAN
%------------------------
% The median is the score that corresponds to the point at or below which
% 50% of the scores fall when the data are arranged in numerical order.
% By this definition, the median is also called the 50th percentile.

Data.Weight_Median = median(Data.Weight);
Data.Height_Median = median(Data.Height);
Data.BMI_Median    = median(Data.BMI);

%------------------------
% MEAN
%------------------------
Data.Weight_Mean = mean(Data.Weight);
Data.Height_Mean = mean(Data.Height);
Data.BMI_Mean    = mean(Data.BMI);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                   MEASURES OF VARIABILITY
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% https://it.mathworks.com/help/stats/measures-of-dispersion.html

% In this section we will see some statistics that can be used to
% represent the dispersion of the distribution.

%------------------------
% RANGE
%------------------------
% The range is a measure of distance, namely the distance from the lowest
% to the highest score.
Data.Weight_Range = range(Data.Weight);
Data.Height_Range = range(Data.Height);
Data.BMI_Range    = range(Data.BMI);

%------------------------
% STANDARD DEVIATION
%------------------------
Data.Weight_STD = std(Data.Weight);
Data.Height_STD = std(Data.Height);
Data.BMI_STD    = std(Data.BMI);

%------------------------
% VARIANCE
%------------------------
Data.Weight_VAR = var(Data.Weight);
Data.Height_VAR = var(Data.Height);
Data.BMI_VAR    = var(Data.BMI);
```

```
Data
```

```
Data = struct with fields:
          Weight: [49.8000 56.4000 68 54.6000 58 57 52.2000 66 55 62 56 49 58.4000 69 61.3000 71.2000 63
          Height: [1×51 double]
          Gender: [0 1 1 0 1 1 0 1 0 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0 0 0 1 1 1 0 0 1 1 0 0
             BMI: [1×51 double]
     Weight_Mode: 16
     Height_Mode: 16
        BMI_Mode: 10
   Weight_Median: 56.4000
   Height_Median: 1.6500
      BMI_Median: 20.7157
     Weight_Mean: 58.7059
     Height_Mean: 1.6675
        BMI_Mean: 21.0687
    Weight_Range: 38
    Height_Range: 0.3900
       BMI_Range: 10.4925
      Weight_STD: 9.2255
      Height_STD: 0.0835
         BMI_STD: 2.5286
      Weight_VAR: 85.1098
      Height_VAR: 0.0070
         BMI_VAR: 6.3936
```

```
%-----------------------------------------------------------------------
% BOXPLOTS: Graphical Representations of Dispersions and Extreme Scores
%-----------------------------------------------------------------------

% A boxplot is a standardized way of displaying the distribution of data
% based on a five number summary:
% 1)“minimum”
% 2) first quartile (Q1)
% 3) median (Q2)
% 4) third quartile (Q3)
% 5) “maximum”

% The point that cuts off the lowest 25% of the distribution is called the
% first quartile and is usually denoted as Q1. Similarly, the point that
% cuts off the upper 25% of the distribution is called the third quartile
% and is denoted Q3. The median is the second quartile, Q2.

figure,
subplot(2,2,1)
boxplot([Data.Weight ]); ylabel('Weight [Kg]');
subplot(2,2,2)
boxplot([Data.Height ]); ylabel('Height [m]');
subplot(2,2,3)
boxplot([Data.BMI ]); ylabel('BMI');
```



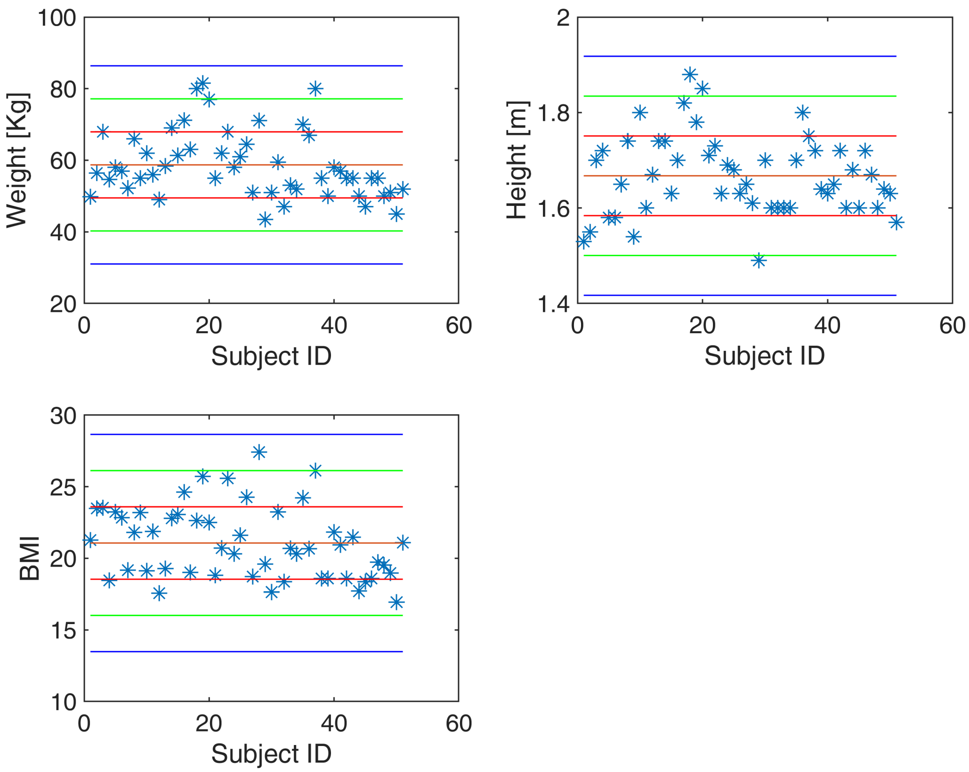

```
% Definition of "Outlier" according to the boxplot method (Tukey's fences)

% The difference between the first and third quartiles (Q3 - Q1) is the
% interquartile range.

% One could define an outlier to be any observation outside the range:
% [ Q1-k(Q3-Q1) , Q3+k(Q3-Q1) ], where k is positive (usually k=1.5).
```

```
% Es a = [ 1:14 ] a = [-100 2:13: 50]
```

Outliers According to the 3-sigma rule

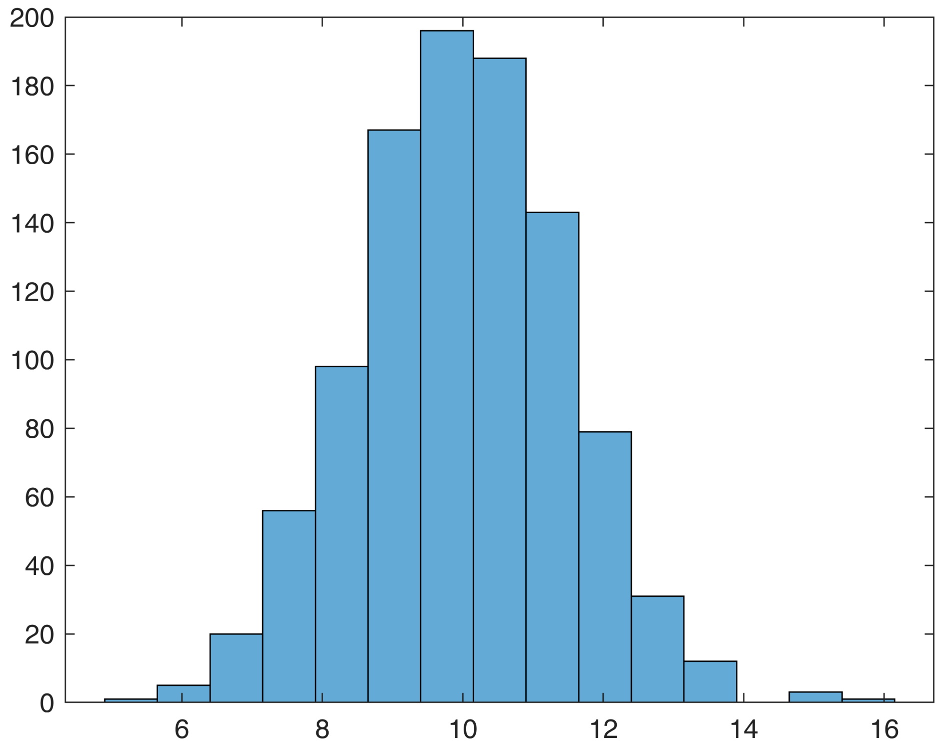
```
figure;
subplot(2,2,1)
plot(Data.Weight,'*'),
hold on
plot([1:N],Data.Weight_Mean*ones(1,N));
plot([1:N],(Data.Weight_Mean-Data.Weight_STD)*ones(1,N),'r');
plot([1:N],(Data.Weight_Mean+Data.Weight_STD)*ones(1,N),'r');
plot([1:N],(Data.Weight_Mean-2*Data.Weight_STD)*ones(1,N),'g');
plot([1:N],(Data.Weight_Mean+2*Data.Weight_STD)*ones(1,N),'g');
plot([1:N],(Data.Weight_Mean-3*Data.Weight_STD)*ones(1,N),'b');
plot([1:N],(Data.Weight_Mean+3*Data.Weight_STD)*ones(1,N),'b');
```

```
xlabel('Subject ID');  ylabel('Weight [Kg]');
hold off,

subplot(2,2,2)
plot(Data.Height,'*'),
hold on
plot([1:N],Data.Height_Mean*ones(1,N));
plot([1:N],(Data.Height_Mean-Data.Height_STD)*ones(1,N),'r');
plot([1:N],(Data.Height_Mean+Data.Height_STD)*ones(1,N),'r');
plot([1:N],(Data.Height_Mean-2*Data.Height_STD)*ones(1,N),'g');
plot([1:N],(Data.Height_Mean+2*Data.Height_STD)*ones(1,N),'g');
plot([1:N],(Data.Height_Mean-3*Data.Height_STD)*ones(1,N),'b');
plot([1:N],(Data.Height_Mean+3*Data.Height_STD)*ones(1,N),'b');
xlabel('Subject ID');  ylabel('Height [m]');
hold off,

subplot(2,2,3)
plot(Data.BMI,'*'),
hold on
plot([1:N],Data.BMI_Mean*ones(1,N));
plot([1:N],(Data.BMI_Mean-Data.BMI_STD)*ones(1,N),'r');
plot([1:N],(Data.BMI_Mean+Data.BMI_STD)*ones(1,N),'r');
plot([1:N],(Data.BMI_Mean-2*Data.BMI_STD)*ones(1,N),'g');
plot([1:N],(Data.BMI_Mean+2*Data.BMI_STD)*ones(1,N),'g');
plot([1:N],(Data.BMI_Mean-3*Data.BMI_STD)*ones(1,N),'b');
plot([1:N],(Data.BMI_Mean+3*Data.BMI_STD)*ones(1,N),'b');
xlabel('Subject ID');  ylabel('BMI');
hold off
```



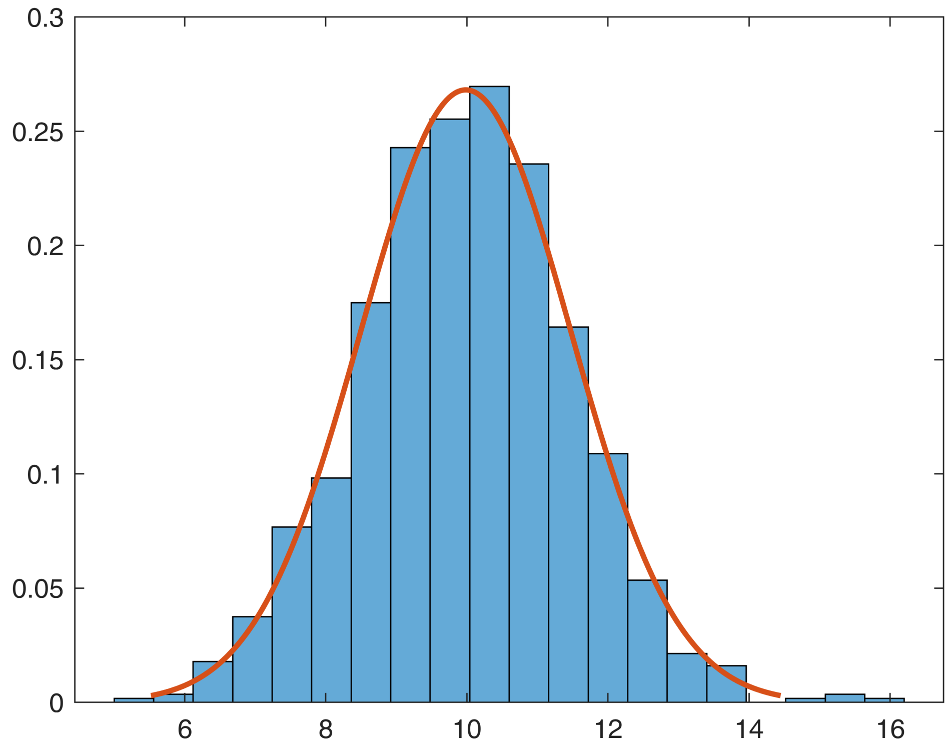
NORMAL DISTRIBUTION

```
% Generation of n random observations drawn from a Normal Distribution.

mu = 10;
sigma = 1.5;
n = 1000;
observations = normrnd(mu,sigma,1,n);
figure;
histogram(observations,'NumBins',15);
```



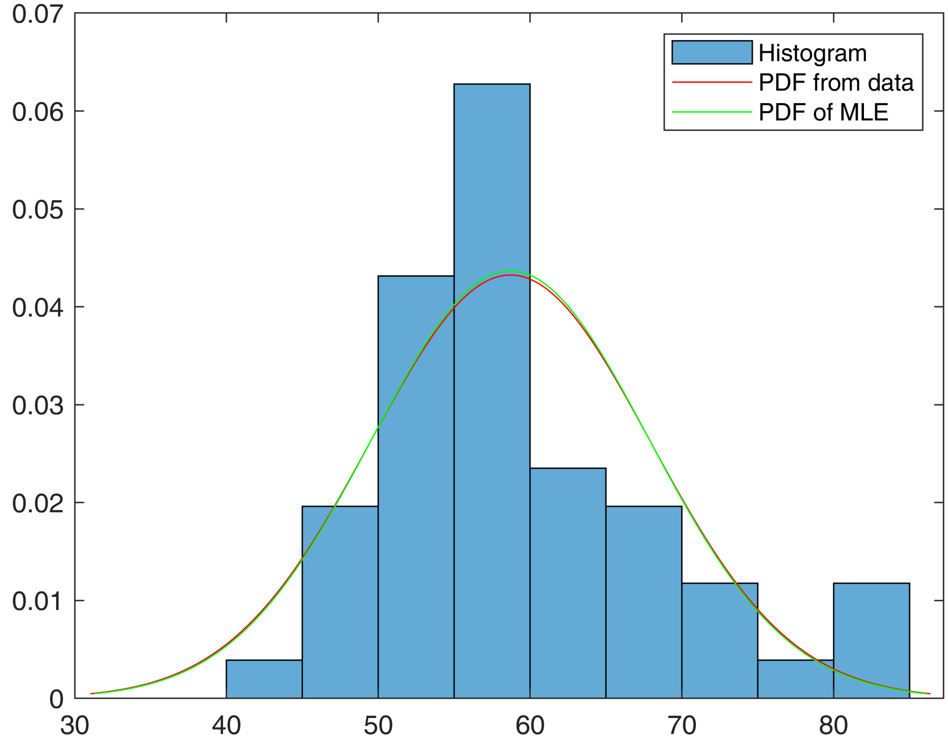
Maximum likelihood estimation - MLE

```
% MLE is a method of estimating the parameters of a statistical model,
% given observations.

p = mle(observations);
mean_est = p(1);
std_est  = p(2);

% Comparison between observation distribution and the
% Normal Probability Density Function featuring by the estimated parameters.

x = [mean_est-3*std_est:0.001:mean_est+3*std_est];
y = normpdf(x,mean_est,std_est);

figure;
histogram(observations,'NumBins',20,'Normalization','pdf');
hold on
plot(x,y, 'linewidth', 2)
```



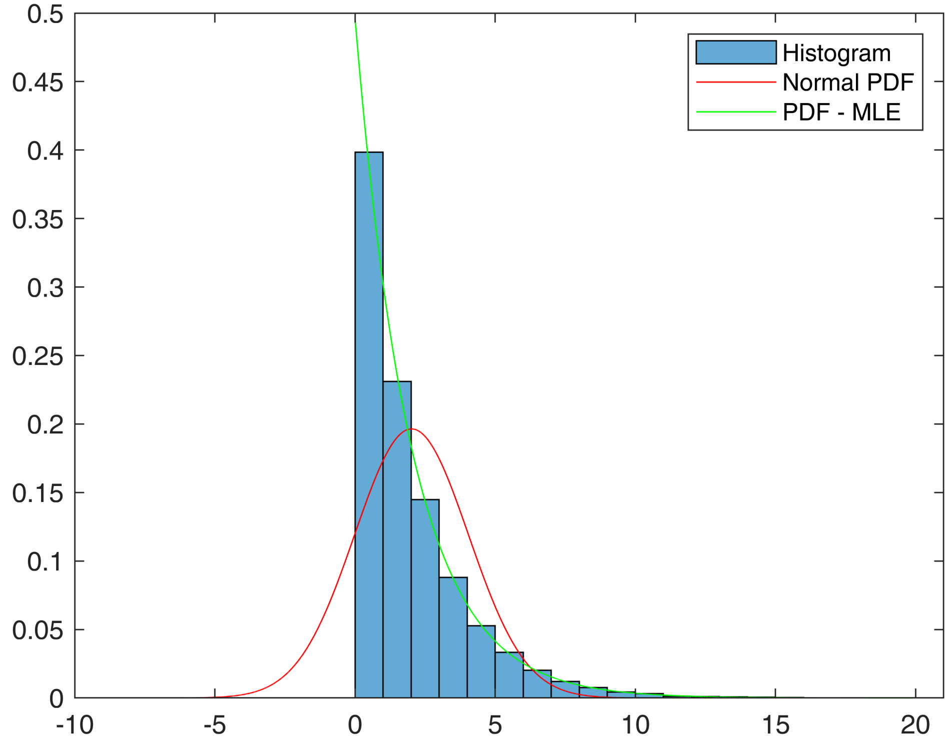
MLE on Anthropometric Data

```
figure,
histogram(Data.Weight,'BinEdges',40:5:85,'Normalization','pdf')
hold on,

x = [Data.Weight_Mean-3*Data.Weight_STD:0.001:Data.Weight_Mean+3*Data.Weight_STD];
y = normpdf(x,Data.Weight_Mean,Data.Weight_STD);
plot(x,y, 'r');

p = mle(Data.Weight);
mean_est = p(1);
std_est  = p(2);
x = [mean_est-3*std_est:0.001:mean_est+3*std_est];
y = normpdf(x,mean_est,std_est);
plot(x,y, 'g');

legend('Histogram','PDF from data', 'PDF of MLE');
```



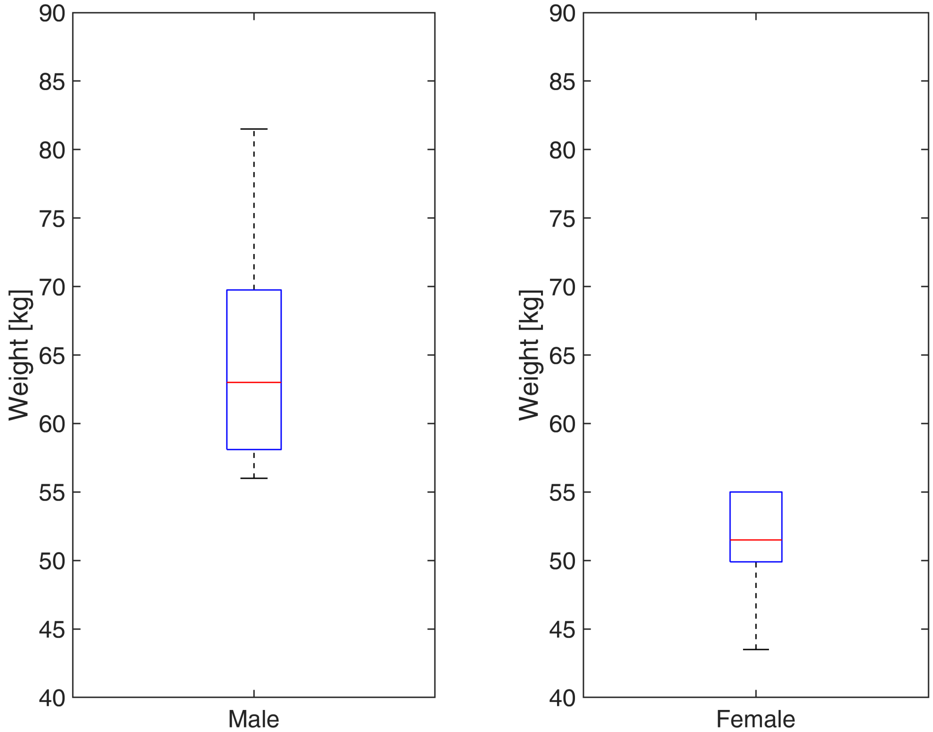
Example with samples drawn from an Exponential Distribution

```
lambda = 2;
n = 10000;
observations = exprnd(lambda,1,n);
figure;
histogram(observations,'NumBins',20,'Normalization','pdf');
hold on,

mean_obs = mean(observations);
std_obs  = std(observations);
x = [mean_obs-4*std_obs:0.001:mean_obs+4*std_obs];
y = normpdf(x,mean_obs,std_obs);
plot(x,y, 'r');

p = mle(observations,'dist', 'exp');
x = [min(observations):0.001:max(observations)];
y = exppdf(x,p);
plot(x,y, 'g');
legend('Histogram','Normal PDF ', 'PDF - MLE');
```



SUBGROUPS: Gender

```
% Data.Gender
%
% 1 -> Male
% 0 -> Female

% Plot Weight Data

figure,
subplot(1,2,1)
boxplot(Data.Weight( find(Data.Gender == 1)) ,'labels','Male');
ylim([40 90]); ylabel('Weight [kg]');
subplot(1,2,2)
boxplot(Data.Weight( find(Data.Gender == 0)) ,'labels','Female');
ylim([40 90]); ylabel('Weight [kg]');
```



Expectation maximization algorithm for learning a multi-dimensional Gaussian mixture

```
% In statistics, an expectation-maximization (EM) algorithm is an iterative method
% to find maximum likelihood in statistical models

n = 1000;

mu1 = 0;
mu2 = 8;
std1 = 2;
std2 = 2;

D1 = normrnd(mu1,std1,n,1);
D2 = normrnd(mu2,std2,n/2,1);

X = [D1;D2];

gmfit = fitgmdist(X,2);

mu = gmfit.mu
```

```
mu = 2×1
    7.8548
   -0.0675
```

```
std = sqrt(gmfit.Sigma)
```

```
std =
std(:,:,1) =

    1.9022


std(:,:,2) =

    1.9907
```

```
w = gmfit.ComponentProportion
```

```
w = 1×2
    0.3334    0.6666
```

```
figure;

x = min(X):0.001:max(X);
histogram(X,'NumBins',30,'Normalization','pdf');
hold on;
g1 = normpdf(x,mu(1),std(1));
g2 = normpdf(x,mu(2),std(2));

plot(x,w(1)*g1,'LineWidth',3),
plot(x,w(2)*g2,'LineWidth',3);
```