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

Corso di Identificazione dei Modelli e Analisi dei Dati

Multivariate Random Variables

Prof. Giuseppe De Nicolao, Federica Acerbi, Alessandro Incremona

Multivariate normal distributions

- **probability density function (pdf)**

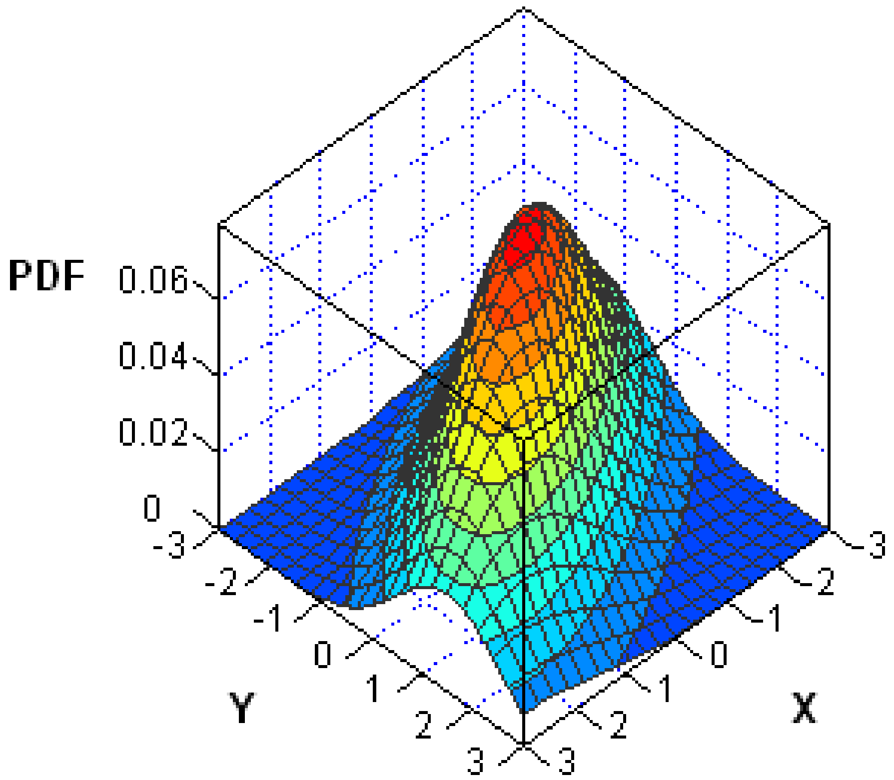
```
>> y = mvnpdf(X,MU,SIGMA)
```

- **cumulative distribution function (cdf)**

```
>> c = mvncdf(X,MU,SIGMA)
```

- **random numbers generation**

```
>> r = mvnrnd(MU,SIGMA,cases)
```



Exercise 1 (part 1)

1. Generate 10000 pairs of values (X,Y) from a bivariate normal distribution $\mathcal{N}(\mu, \Sigma)$,

   with $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} 1 & 0.6 \\ 0.6 & 2 \end{bmatrix}$

Exercise 1 (part 1)

1. Generate 10000 pairs of values (X,Y) from a bivariate normal distribution $\mathcal{N}(\mu, \Sigma)$,

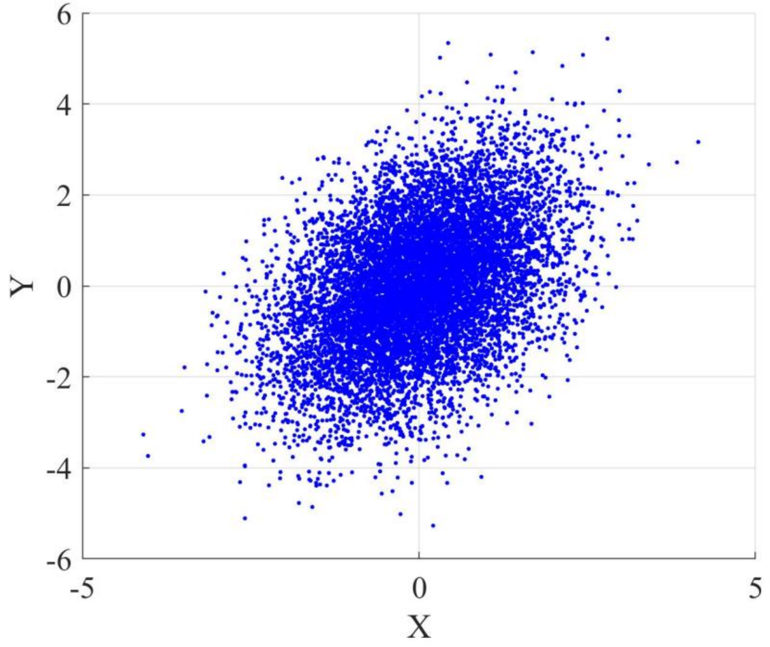
with $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} 1 & 0.6 \\ 0.6 & 2 \end{bmatrix}$

Solution:

```
>> MeanVec = [0 0];
>> CovMatrix = [1 0.6; 0.6 2];

>> Samples = mvnrnd(MeanVec, CovMatrix, 10000);

>> figure(1)
>> scatter(Samples(:,1),Samples(:,2),'.b')
```



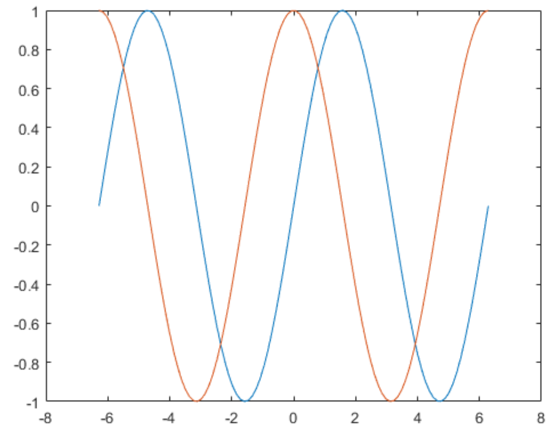
What is the difference between `scatter` and `plot` commands in Matlab?

`plot`

- has a concept of the order of the points. It is usually used to make line plots (**Fig. 1**)

- gives the possibility to draw the points, without connecting line, specifying only the marker (e.g. 'o','+','*',etc.)

Fig. 1

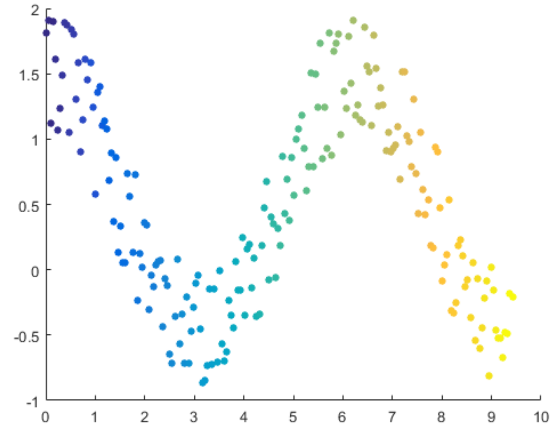


`scatter`

- creates a scatter plot with circles at the locations specified by the vectors x and y (**Fig. 2**). This type of graph is also known as a bubble plot.

- **allows more customization of individual data markers**

Fig. 2



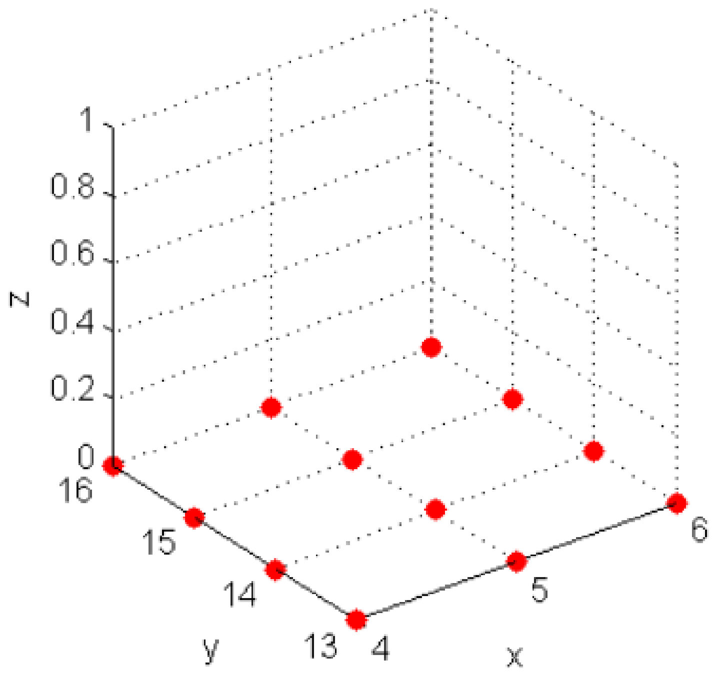
3-D Surface Plots

```
>> x = 4:6
```

$$\begin{bmatrix} 4 & 5 & 6 \end{bmatrix}$$

```
>> y = 13:16
```

$$\begin{bmatrix} 13 & 14 & 15 & 16 \end{bmatrix}$$

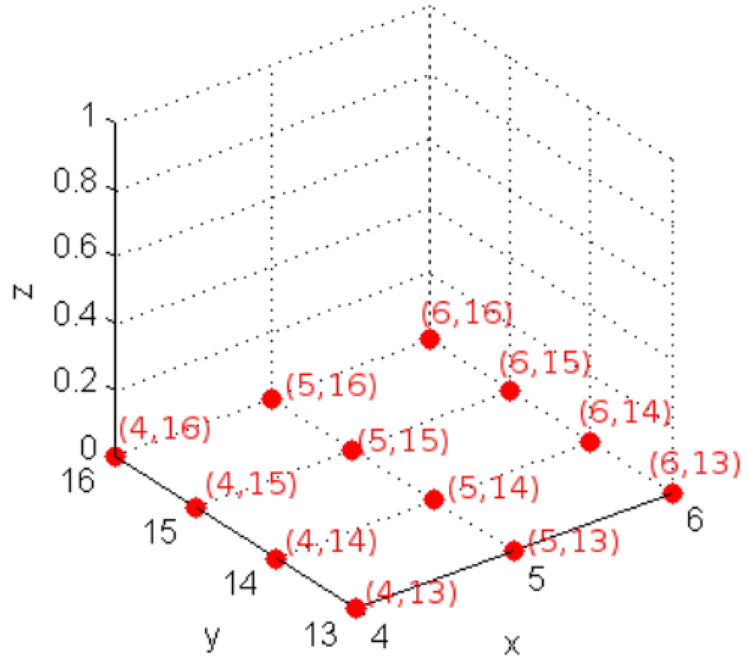


3-D Surface Plots

```
>> [X,Y] = meshgrid(x,y)
```

$$
X = \begin{bmatrix} 4 & 5 & 6 \\ 4 & 5 & 6 \\ 4 & 5 & 6 \\ 4 & 5 & 6 \end{bmatrix}
\qquad
Y = \begin{bmatrix} 13 & 13 & 13 \\ 14 & 14 & 14 \\ 15 & 15 & 15 \\ 16 & 16 & 16 \end{bmatrix}
$$

Notice that each x-value is paired with each y-value at some location within the arrays.



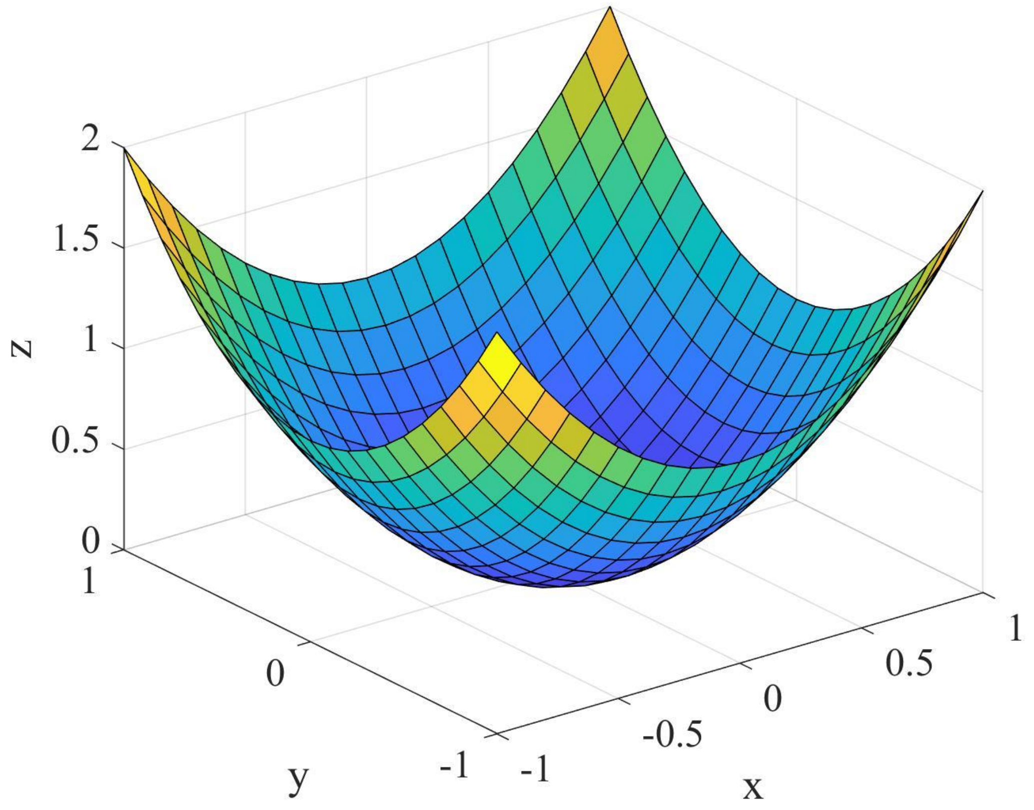
3-D Surface Plots

```
>> x = -1:0.1:1;
>> y = -1:0.1:1;

>> [X,Y] = meshgrid(x,y);

>> Z = X.^2 + Y.^2;
>> surf(X,Y,Z)

>> xlabel('x')
>> ylabel('y')
>> zlabel('z')
```

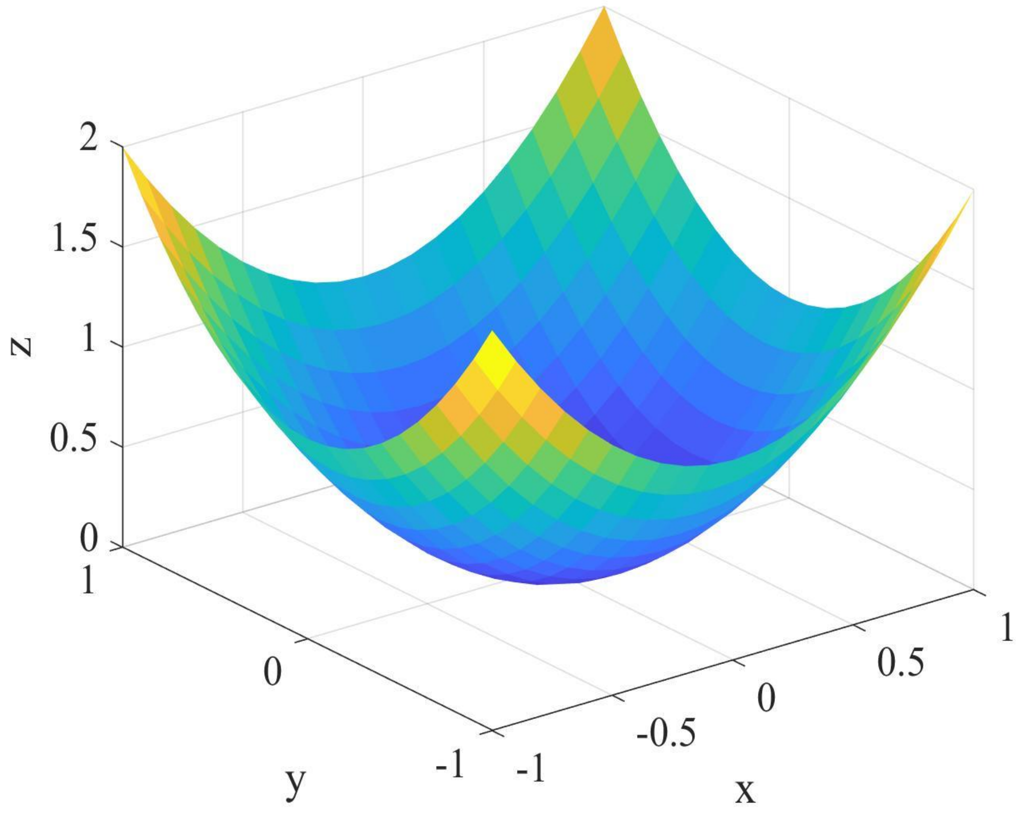


Surface properties

```
>> s = surf(X,Y,Z)
```

Turn off the display of the edges

```
>> s.EdgeColor = 'none';
```



Surface properties

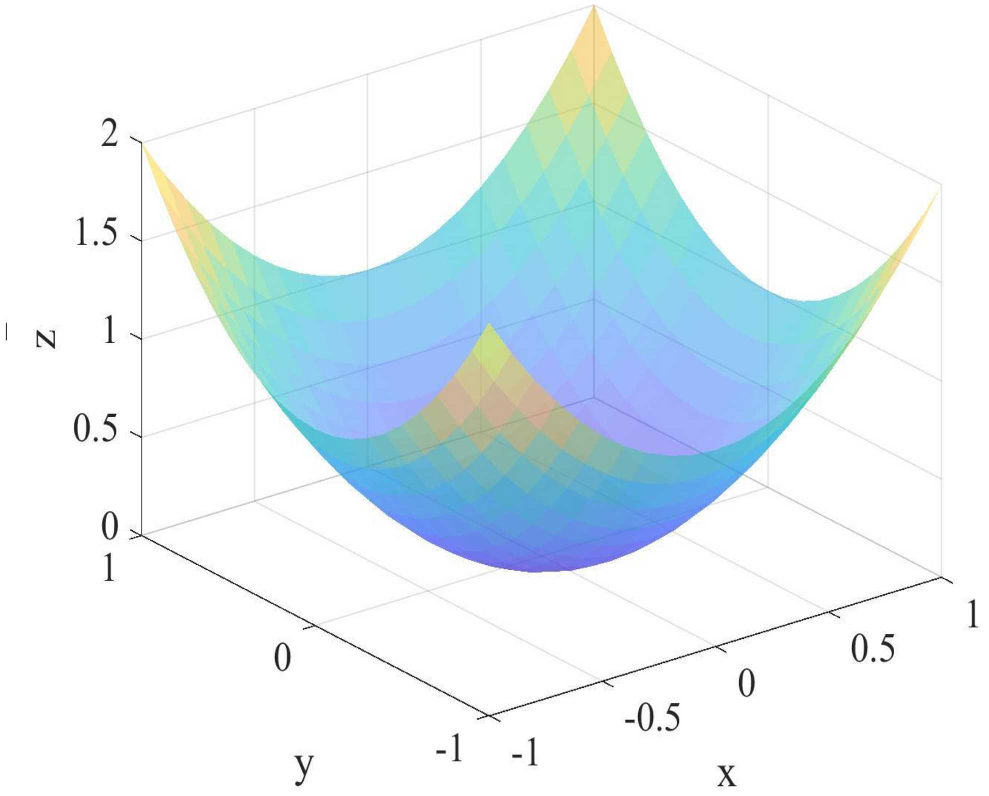
```
>> s = surf(X,Y,Z)
```

Turn off the display of the edges

```
>> s.EdgeColor = 'none';
```

Create a semitransparent surface

```
>> s.FaceAlpha = 0.5
```



Surface properties

```
>> s = surf(X,Y,Z)
```

Turn off the display of the edges

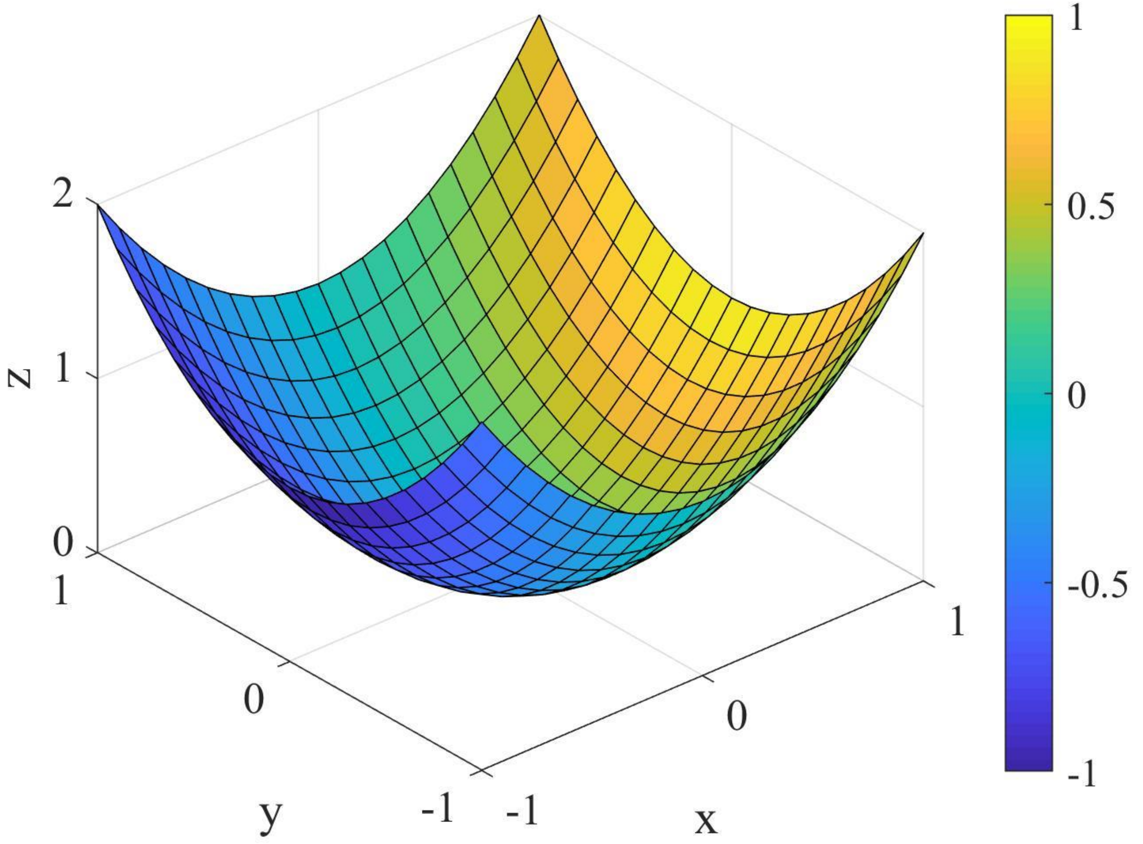
```
>> s.EdgeColor = 'none';
```

Create a semitransparent surface

```
>> s.FaceAlpha = 0.5
```

Customize the colours of the surface

```
>> C = X.*cos(Y);
>> s = surf(X,Y,Z,C)
>> colorbar
```



Exercise 2

1. Plot the function

$$f(x,y) = x\sin(xy)$$

   in the region $0 \leq x \leq 5, \pi \leq y \leq 2\pi$ using the commands `meshgrid` and `surf`

2. Set the surface transparency as 0.6 and delete the edges

3. Rename the figure axis as: `‘X’,’Y’,’f(X,Y)’`.

Exercise 1 (part 2)

2. Approximate the marginal distributions of X and Y using the histograms and draw over them the theoretical marginal distributions

3. Draw the theoretical probability density function (pdf) of the bivariate normal distribution, from which the data points X and Y was sampled, using the command **`mvnpdf`**

4. Draw the theoretical cumulative distribution function (cdf) of the bivariate normal distribution using the command **`mvncdf`**

5. Approximate the function drawn in the point 3) using the functions **`histogram2 and ksdensity`**