

# **Examples of Applying Differential Geometry in Big Data Analytics**

Prof. Weiqing Gu

- **We first give examples of applying Differential Geometry in computer vision.**

Reference: Those examples are discussed in the book I sent to you last night.

- **Then give examples in other areas.**

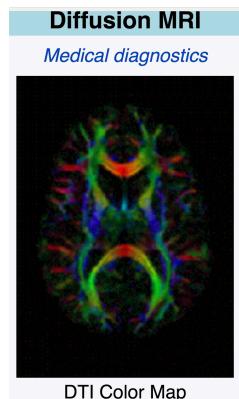
Reference: MULTISCALE REPRESENTATIONS FOR MANIFOLD-VALUED DATA by RAHMAN, Donoho etc.

1. **Spheres, Histograms, Bag of Words:** As mentioned previously, fixing the norm of the vectors restricts the relevant space of vectors to a sphere. In case the norm is selected to be one, the corresponding space is a unit sphere  $\mathbb{S}^n \subset \mathbb{R}^{n+1}$ . The unit spheres arise in situation where one is interested in direction of the vector rather than its magnitude. In statistical analysis, such considerations have led to an interesting area of study called *Directional Statistics* [44]. In computer vision, one often uses normalized histograms as representations or summaries of image or video data. Let a histogram  $h$  be a vector of frequency data points in  $n + 1$  predetermined bins. One way to normalize  $h$  is to impose the condition that  $\sum_{i=1}^{n+1} h_i = 1$ . In other words,  $h$  is a probability vector of size  $n + 1$ . If we define  $p_i = +\sqrt{h_i}$ , then we see that  $\|p\| = \sqrt{\sum_{i=1}^{n+1} p_i^2} = \sqrt{\sum_{i=1}^{n+1} h_i} = 1$ , or  $p \in \mathbb{S}^n$ . Thus, one can conveniently represent normalized histograms as elements of a unit sphere, after a square-root transformation, and utilize the geometry of this sphere to analyze histograms [59]. It has also been argued [39] that the popular bag-of-words representation, which is commonplace in vision applications (cf. [48]), is best studied in terms of the statistical manifold of the  $n$ -dimensional multinomial family. Lafferty and Lebanon [39] proposed this approach and drew from results in information geometry to derive heat kernels on this statistical family, demonstrating the benefits of a geometric approach over a traditional vector space approach.

**2. Rotation Matrices, Rigid Motions, Structure from Motion:** One of the central problems in computer vision is recognition of objects from their images or videos, invariant to their pose relative to the camera. Related problems include pose tracking, pose-invariant detection of objects, and so on. How is the pose of a 3D object, relative to a chosen coordinate system, represented in analysis? It is most naturally represented as an element of  $SO(3)$ , the set of all  $3 \times 3$  orthogonal matrices with determinant +1 [29]. This representation of pose enjoys a special structure that has the correct physical interpretations. The set  $SO(3)$  is a Lie group with the group operation given by matrix multiplication. If one applies two rotations  $O_1, O_2 \in SO(3)$  on a rigid object, in that order, then the cumulative rotation is given by  $O_1 O_2 \in SO(3)$ . Similarly, the transpose of a rotation  $O \in SO(3)$ , denoted by  $O^T$ , represents the inverse rotation and nullifies the effect of  $O$  since  $O O^T = I$ . Incidentally,  $SO(n)$  is a subset (and subgroup) of a more general transformation group  $GL(n)$ , the set of all  $n \times n$  non-singular matrices.  $GL(n)$  is useful in characterizing affine transformation in computer vision. Geometrical techniques for rotation averaging in structure from motion problems have been investigated in computer vision literature [9, 27].

**3. Covariance Tensors, Diffusion Tensor Imaging, Image-Patch Modeling:** In statistics one often represents a random quantity by a collection of its lower order moments. In particular, the use of mean and covariance matrices to characterize random variables is very common, especially under Gaussian models. A covariance matrix is a square, symmetric, and positive-definite matrix. When analyzing variables using their covariance matrices, the geometry of the space of such symmetric, positive-definite matrices (SPDMs) becomes important. This is a nonlinear space whose geometry is naturally studied as a quotient space of a Lie group,  $GL(n)$  modulo  $SO(n)$ . In other words, one forms the action of  $SO(n)$  on  $GL(n)$  and identify the orbits of  $SO(n)$  with individual SPDMs. A field of  $3 \times 3$  tensors forms an intermediate representation of data in diffusion-tensor MRI, and one needs the geometry of SPDMs for interpolation, denoising, and registrations of image data [15, 47, 51, 54]. This representation has also proved highly successful in modeling textures in patches and its application to pedestrian detection and tracking [66, 67].

MRI =magnetic resonance imaging



**4. Subspaces, Dynamical Models, Projections:** Due to severity of dimensions in vision data sets (images and videos) encountered in computer vision, the task of dimension reduction is a central theme. One idea here is to use a linear projection to a low-dimensional subspace of the original observation space. In order to make these projections optimal, for the given data and the given application, one needs to optimize a certain chosen objective function over the space of subspaces. The space of all  $d$ -dimensional orthogonal bases of an  $n$ -dimensional space ( $d \ll n$ ) is called a Stiefel manifold  $\mathcal{S}_{n,d}$ , while the space of all  $d$ -dimensional subspaces of  $\mathbb{R}^n$  is called a Grassmann manifold  $\mathcal{G}_{n,d}$ . These manifolds are naturally studied as quotient spaces of larger rotations modulo smaller rotations:  $\mathcal{S} \equiv SO(n)/SO(n-d)$  and  $\mathcal{G}_{n,d} = SO(n)/(SO(n-d) \times SO(d))$ . The use of linear projections of image data, optimally related to specific vision tasks, is an important area in itself [42, 58]. The Grassmann manifold also plays an important role in characterizing linear, time-invariant, dynamical systems where the observability matrix of the system is represented as a subspace of appropriate dimensions [64, 65]. The Grassmann manifold also arises as the representation space in face recognition when using a subspace to model a collection of faces. This representation has been used to devise very effective face recognition algorithms [24, 25].

**5. Deformations, Image Registration:** One of the most studied problems in image processing, especially medical image analysis, is the problem of registration of pixels/voxels across images. This task is performed by fixing one image  $I_1$  and deforming the other image  $I_2$  such that the corresponding pixel locations are considered matched. The deformation  $\gamma$  is a process of changing pixel locations, while keeping the pixel values fixed, in a smooth fashion so that the pixels do not cross each other. In mathematical terms, one defines deformation using diffeomorphic transformations of the set of pixel locations. Thus, the set of diffeomorphisms is synonymous with the set of deformations of images used in image registration [5, 71]. The cost function used for image registration is most commonly of the type

$$\arg \min_{\gamma \in \Gamma} \left( \|I_1 - I_2 \circ \gamma\|^2 + \lambda \mathcal{R}(\gamma) \right),$$

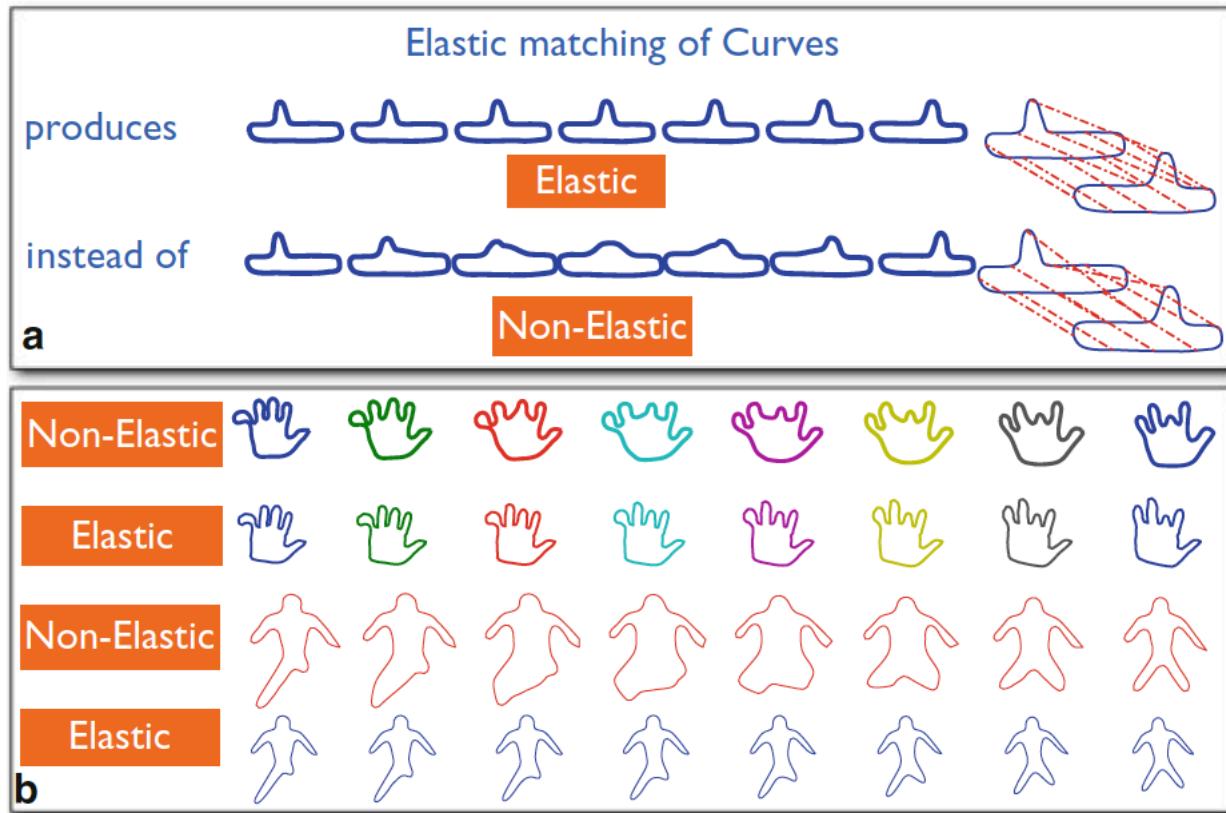
where  $\Gamma$  is the set of all deformations and  $\mathcal{R}$  is a roughness penalty on  $\gamma$ .

**6. Elastic Functions, Trajectories, Activities:** In experiments involving dynamic systems one studies the evolution of certain feature(s) of interest over an observation period. These variables are mathematically represented as real- or vector-valued functions over time and the area of statistics that deals with such data is called *functional data analysis*. The main challenges in this area come from two sources: (1) the infinite dimensionality of representation spaces and (2) the fact that the systems evolve under different rates in different observations. The latter source, termed the phase variability in functional data, is a nuisance variable and needs to be removed in statistical data analysis. This requires proper metrics and models to be able to formally define the concepts of amplitude and phase components in functions [34]. The metrics that allow comparisons of functions under different phase components are termed *elastic metrics* and the resulting analysis *elastic FDA*. An extension of this problem involves studying temporal evolutions of systems whose representations take values on Riemannian manifolds. The sample paths, or observations, of these systems are studied as trajectories on Riemannian manifolds [35, 62]. For instance, consider a human performing an action or an activity in front of a depth sensor where one forms a skeletal representation of the human body and studies the evolution of its shape to characterize the activity. This activity can be represented as a trajectory on an appropriate shape space of skeletons and one needs metrics/models for classifying activities using noisy depth data (cf. [3]).

**7. Shape Representations:** In the context of detection, tracking, and recognition of objects in image and video data, the shapes of silhouettes of objects play a very important role. Since these silhouettes of objects form contours and surfaces in 2D and 3D images, the area of shape analysis of curves and surfaces has become very active. Even though one starts with Euclidean representations of these objects, shape analysis quickly becomes complicated because shape is a property that is invariant to rigid motion, global scaling, and re-parameterization of the objects being studied. Thus, shape spaces are typically quotient spaces representing original objects (curves or surfaces) modulo the actions of these shape-preserving transformation groups. The shape preserving groups are rotation  $SO(n)$ , translation  $\mathbb{R}^n$ , scaling  $\mathbb{R}_+$ , and re-parameterization.

*An example of  
human silhouettes:*

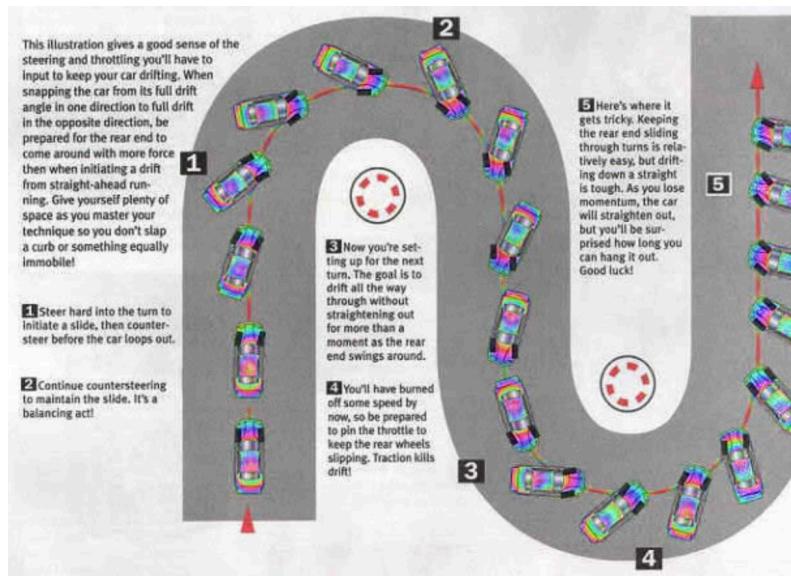




**Fig. 10.6** Examples of geodesics between shapes of curves for nonelastic ( $\gamma^*(s) = s$ ) and elastic matching. **(a)** An example of matching toy curves along with the correspondence between them. **(b)** Examples of hand and human shapes

# More examples of differential geometry in big data analytics

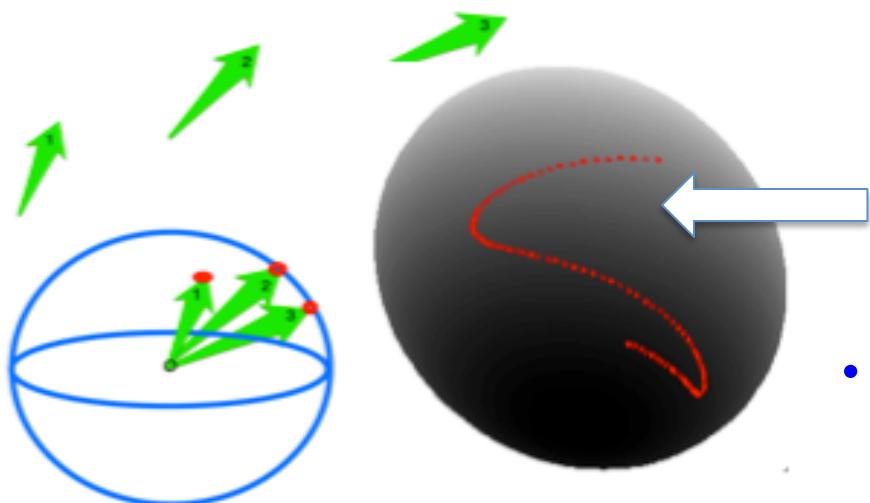
- Manifold-valued data
- Car and UAV headings
- *Headings.* Here  $p$  specifies directions in  $\mathbf{R}^2$  or  $\mathbf{R}^3$ , and so  $M$  is either the unit circle  $S^1 \subset \mathbf{R}^2$  or the unit sphere  $S^2 \subset \mathbf{R}^3$ . Such data can arise as a time series of observations of vehicle headings.



## Extract Data features by Identifying “Characteristic Submanifold of UAV Behaviors” to Use Only Most Important Data

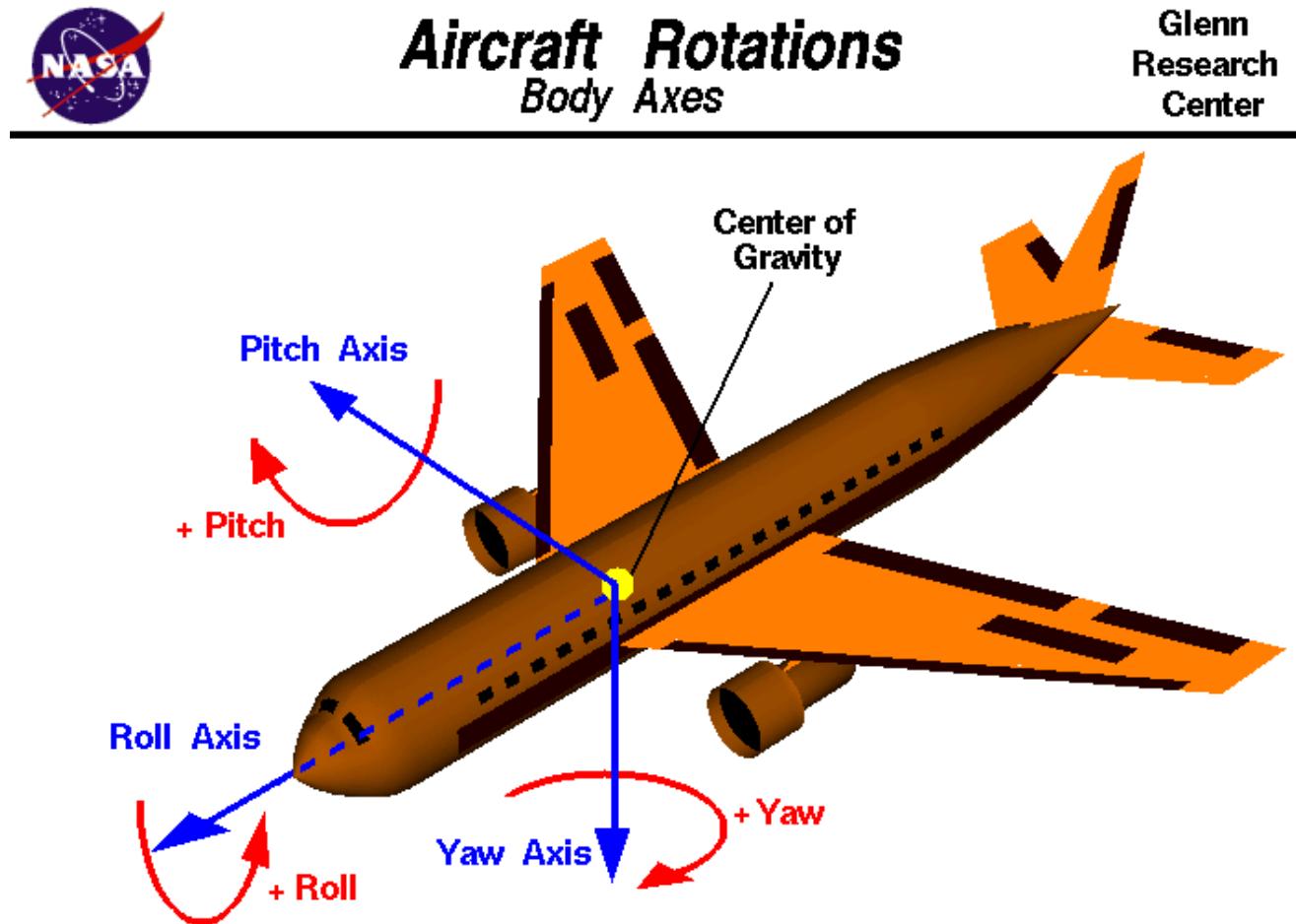


Only consider UAV heading directions here,  
but works for any other UAV characteristics

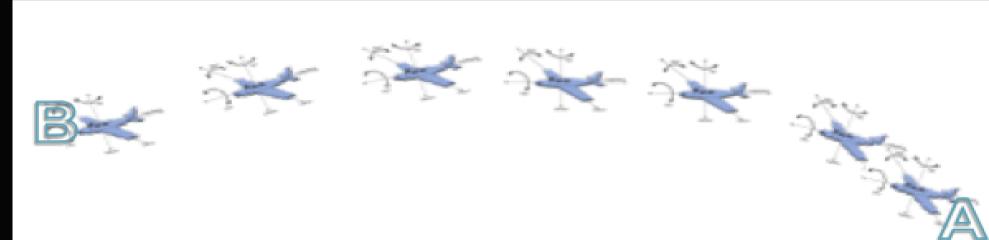
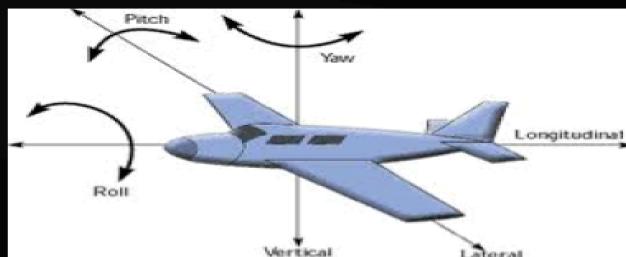


- Example: Only look at UAV “headings”
  - All possible headings for all UAVs form a sphere.
  - Define such a manifold a “Characteristic submanifold” of TM.
- 
- Just Like in Euclidean space, one extracts important data by “project them to a feature subspace”. Here we “project” to a feature submanifold.
- A projection of a behavior curve—call it a Signature Curve of UAV behaviors.*
- Key: Created a dimension-reduction technique for nonlinear data.

- *Orientations.* Here  $p$  gives ‘tripods’, i.e. orientations belonging to  $M = SO(3)$ . Such data can arise as a time series of aircraft orientations (pitch, roll, yaw).



- *Rigid Motions.* Here  $p$  specifies rigid motions in the special Euclidean group  $M = SE(3)$ . Such data can arise as a time series of placements of an object in space (position, orientation), or as a spatially-organized array giving the displacements and orientations of marker particles having undergone a deformation.



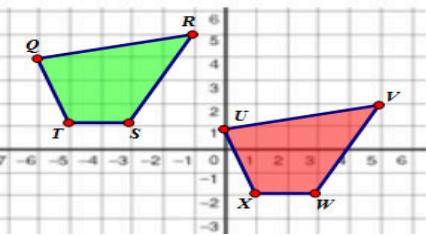
- The sensor data transformed to pitch, roll , yaw, and positions for any UAVs
- **M** = the set of data in  $SO(3) \times \mathbb{R}^3$  transformed from all possible collections of roll, pitch, yaw, and plus position data.
- **M = the set of oriented rigid motions.**
- **TM** = the tangent bundle of M which is M “plus” the velocity vectors.

**Key: TM captures dynamical behaviors of any UAVs**  
**TM is defined to be a manifold of any UAV behaviors**  
 **$\dim(TM) = 12$ .**

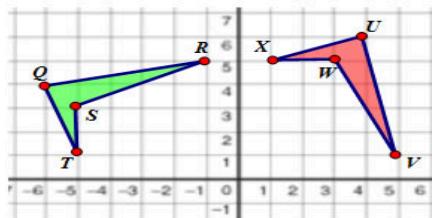
**As a UAV flying, it's behaviors trace out a curve on TM.**

# Many Kinds of Rigid Motion Applications

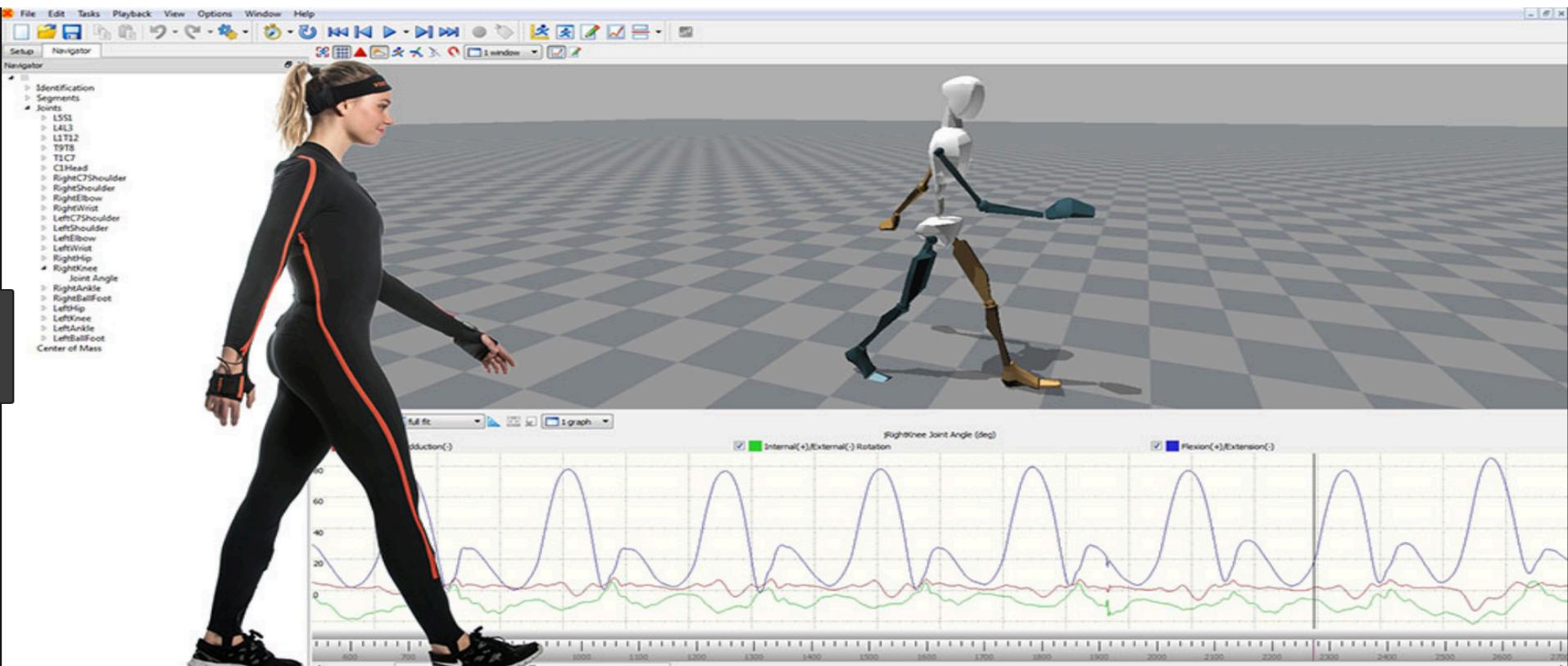
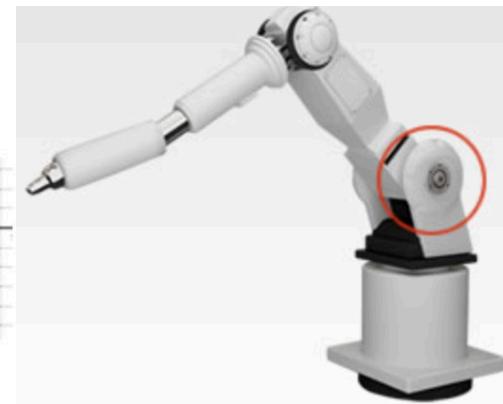
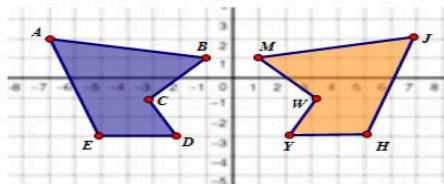
A translation  $(x, y) \rightarrow (x + 6, y - 3)$   
maps these two quadrilaterals,  
so Quad QRST  $\cong$  Quad UVWX



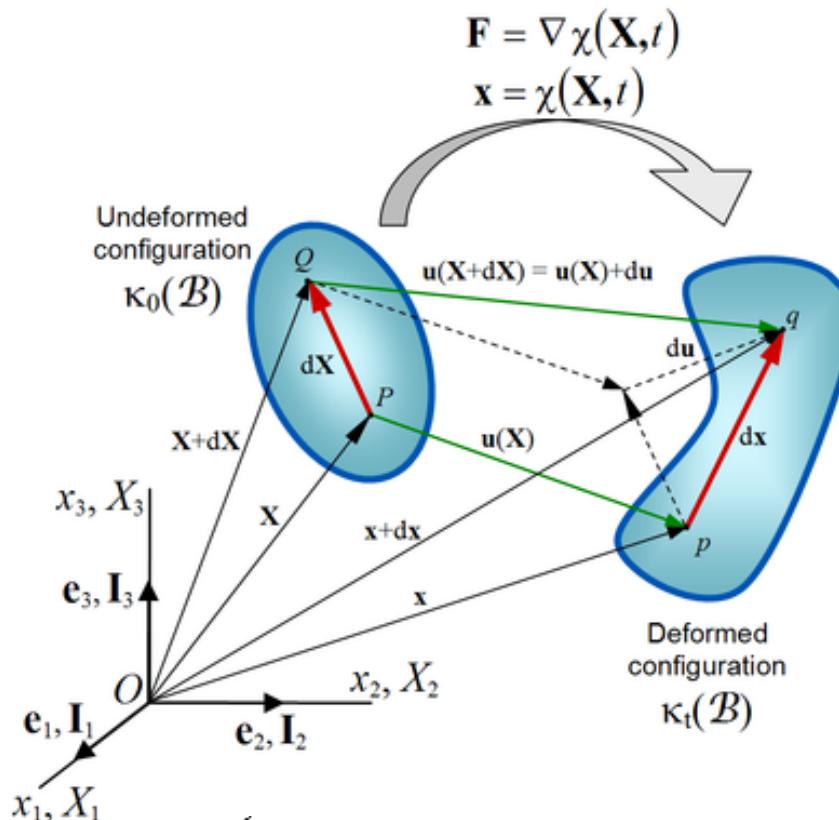
A rotation of  $270^\circ$  about the origin  
maps these two quadrilaterals,  
so Quad QRST  $\cong$  Quad UVWX



A rotation over the y axis  
maps these two pentagons,  
so ABCDE  $\cong$  JMWHY

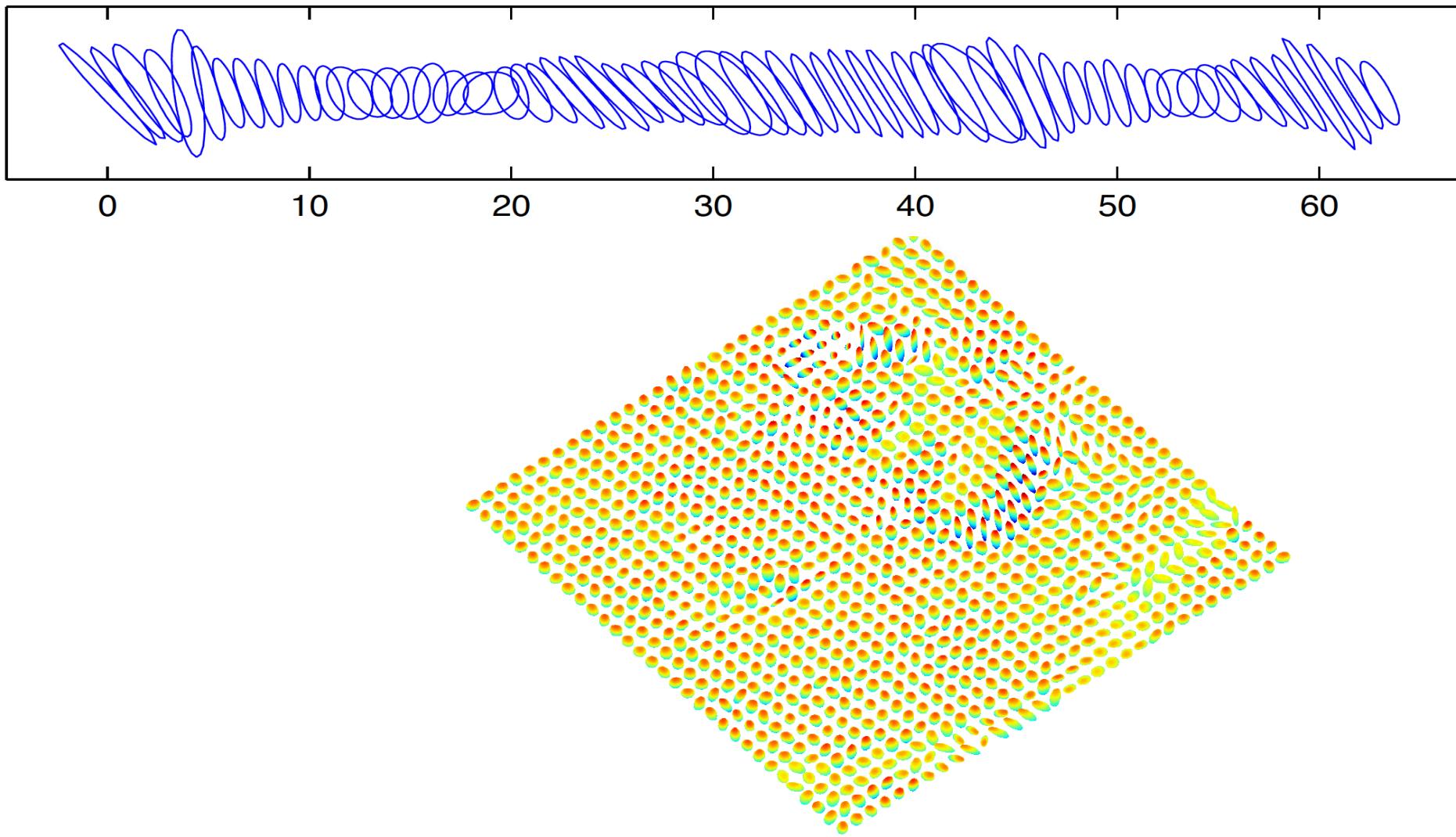


- *Deformation Tensors.* Here  $p$  is a symmetric positive definite matrix in  $M = SPD(n)$ . Spatially-organized data of this kind can arise from measurements of strain/stress and deformation in materials science and earth science. Arrays of this kind also arise in cosmological measurements of gravitational lensing.



$$E_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \frac{\partial u_k}{\partial x_j} \frac{\partial u_k}{\partial x_i} \right)$$

**6.4. Exchange rate data.** We now consider a dataset  $p(t)$  of  $2 \times 2$  symmetric nonnegative-definite matrices. The matrices are covariances between exchange rates for the US Dollar vs. Euro and the US Dollar vs. British Pound, within a 10 day sliding window. Figure 6.6 shows the time series in which the symmetric matrices are depicted as ellipses. The Frobenius norms of the wavelet coefficients are depicted in



- *Distance Matrices.* Here each  $p$  is an  $n$  by  $n$  matrix giving the pairwise distances between all pairs in a cloud of  $n$  points. Time series of this kind can arise as representing the state of a swarm of maneuvering vehicles, each of which can sense its distance to all other members of the swarm.

## Multidimensional scaling

- What does the MDS algorithm do?

CITIES	ATLA	CHIC	DENV	HOUS	L.A.	MIAMI	N.Y.	S.F.	SEAT	WASH D.C.
ATLANTA		587	1212	701	1936	604	748	2139	2182	543
CHICAGO	587		920	940	1745	1188	713	1858	1737	597
DENVER	1212	920		879	831	1726	1631	949	1021	1494
HOUSTON	701	940	879		1374	968	1420	1645	1891	1220
LOS ANGELES	1936	1745	831	1374		2339	2451	347	959	2300
MIAMI	604	1188	1726	968	2339		1092	2594	2734	923
NEW YORK	748	713	1631	1420	2451	1092		2571	2408	205
SANFRANCISCO	2139	1858	949	1645	347	2594	2571		678	2442
SEATTLE	2182	1737	1021	1891	959	2734	2408	678		
WASHINGTON D.C.	543	597	1494	1220	2300	923	205	2442	232	

(B) AIRLINE DISTANCES BETWEEN TEN U.S. CITIES

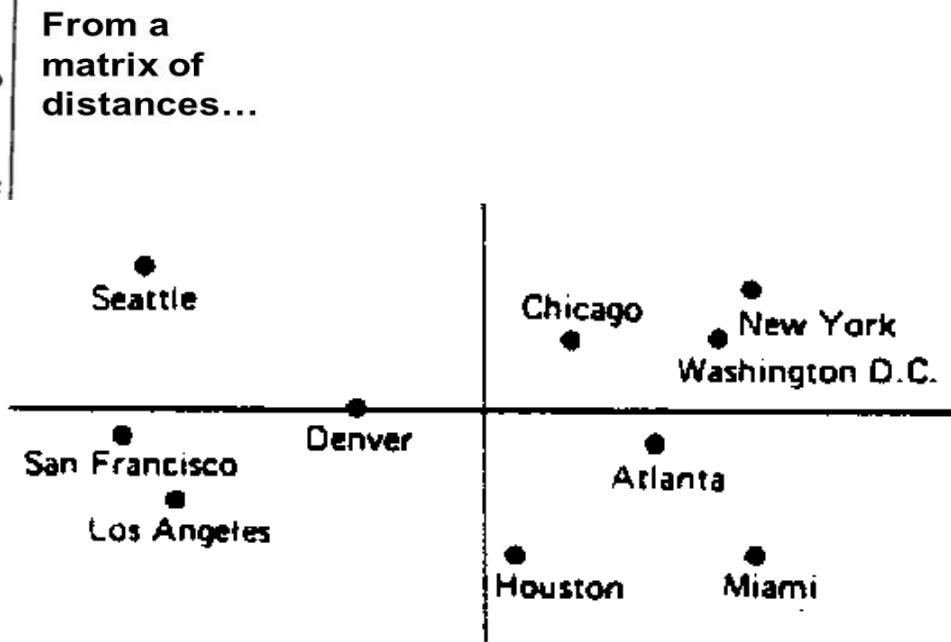
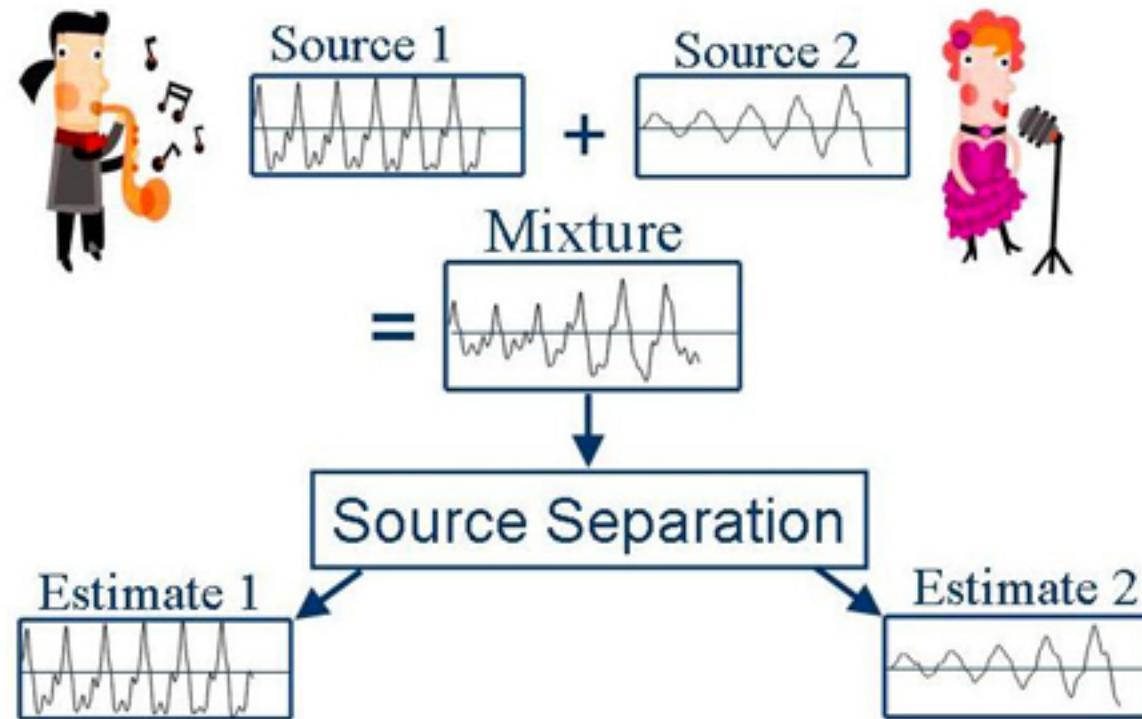


Figure 1 CMDS of flying mileages between 10 American cities.

- *Projections, Subspaces.* Here  $p$  is a projector with  $k$ -dimensional range, or what is the same thing, a  $k$ -subspace of  $\mathbf{R}^n$ . Such values belong to the Grassmann manifold  $G(k, n)$ . Time series of this kind can arise in array signal processing, where the subspace is associated with the signal-generating sources.



**How to use Grassmannian techniques? Open problem!**  
**If you can do it, a Ph.D. thesis!**

# Back up slides

# Covariance, and Covariance Matrix

- The **covariance** between two rv's X and Y measures the degree to which X and Y are (linearly) related; defined as

$$\text{cov}[X, Y] \triangleq \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$$

Exercise

$$= \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$$

If  $\mathbf{x}$  is a d-dimensional random vector, its **covariance matrix** is defined to be the following symmetric, positive definite matrix:

↑  
Often denoted by  $\Sigma$

$$\text{cov}[\mathbf{x}] \triangleq \mathbb{E}[(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{x} - \mathbb{E}[\mathbf{x}])^T]$$
$$= \begin{pmatrix} \text{var}[X_1] & \text{cov}[X_1, X_2] & \cdots & \text{cov}[X_1, X_d] \\ \text{cov}[X_2, X_1] & \text{var}[X_2] & \cdots & \text{cov}[X_2, X_d] \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}[X_d, X_1] & \text{cov}[X_d, X_2] & \cdots & \text{var}[X_d] \end{pmatrix}$$

# correlation coefficient & correlation matrix

- The (Pearson) **correlation coefficient** between two rvs  $X$  and  $Y$  is defined as

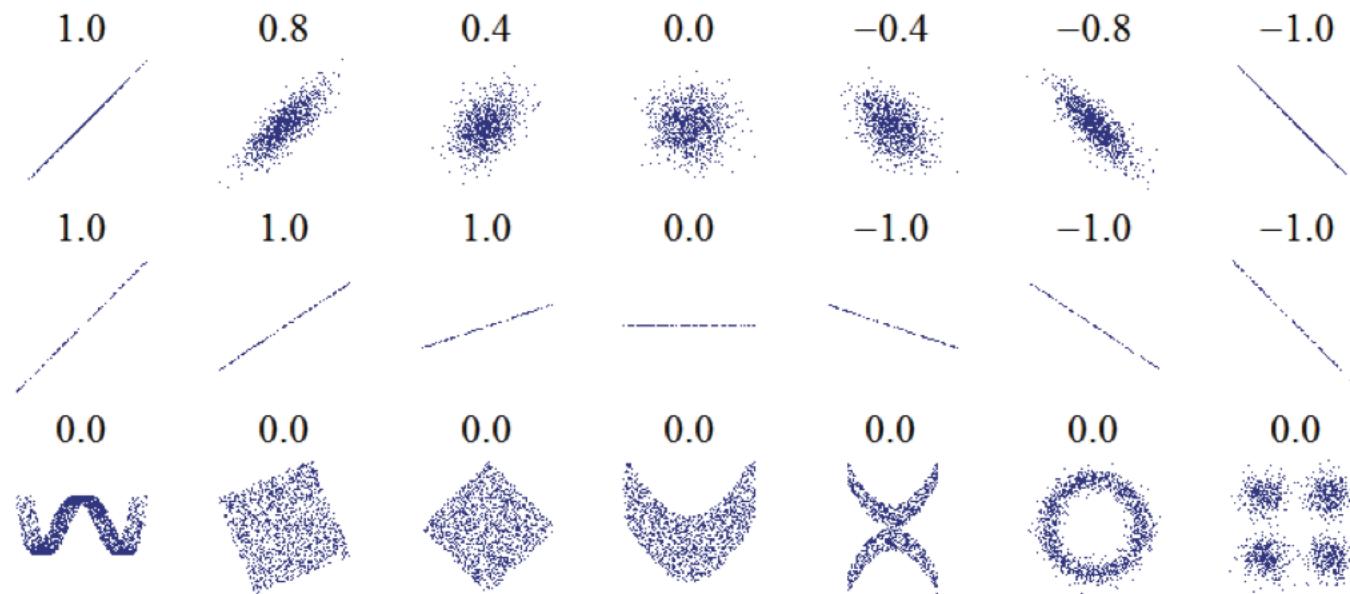
$$\text{corr} [X, Y] \triangleq \frac{\text{cov} [X, Y]}{\sqrt{\text{var} [X] \text{var} [Y]}}$$

- If  $X$  and  $Y$  are indep., then  $\text{cov} [X, Y] = 0$ ; say  $X$  and  $Y$  are uncorrelated.
- A **correlation matrix** of a random vector has the form:

$$\mathbf{R} = \begin{pmatrix} \text{corr} [X_1, X_1] & \text{corr} [X_1, X_2] & \cdots & \text{corr} [X_1, X_d] \\ \vdots & \vdots & \ddots & \vdots \\ \text{corr} [X_d, X_1] & \text{corr} [X_d, X_2] & \cdots & \text{corr} [X_d, X_d] \end{pmatrix}$$

Exercise: show that  $-1 \leq \text{corr} [X, Y] \leq 1$  and  
Show that  $\text{corr}[X,Y] = 1$  iff  $Y = aX + b$  for some parameters  $a$  and  $b$ .

# Example of Correlation Coefficients



**Figure 2.12** Several sets of  $(x, y)$  points, with the correlation coefficient of  $x$  and  $y$  for each set. Note that the correlation reflects the noisiness and direction of a linear relationship (top row), but not the slope of that relationship (middle), nor many aspects of nonlinear relationships (bottom). N.B.: the figure in the center has a slope of 0 but in that case the correlation coefficient is undefined because the variance of  $Y$  is zero. Source: [http://en.wikipedia.org/wiki/File:Correlation\\_examples.png](http://en.wikipedia.org/wiki/File:Correlation_examples.png)

# The multivariate Gaussian (distribution) or multivariate normal (MVN)

(The most widely used joint probability density function for continuous variables)

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \triangleq \frac{1}{(2\pi)^{D/2} |\boldsymbol{\Sigma}|^{1/2}} \exp \left[ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$

determinant

where  $\boldsymbol{\mu} = \mathbb{E}[\mathbf{x}] \in \mathbb{R}^D$  and  $\boldsymbol{\Sigma} = \text{cov}[\mathbf{x}]$

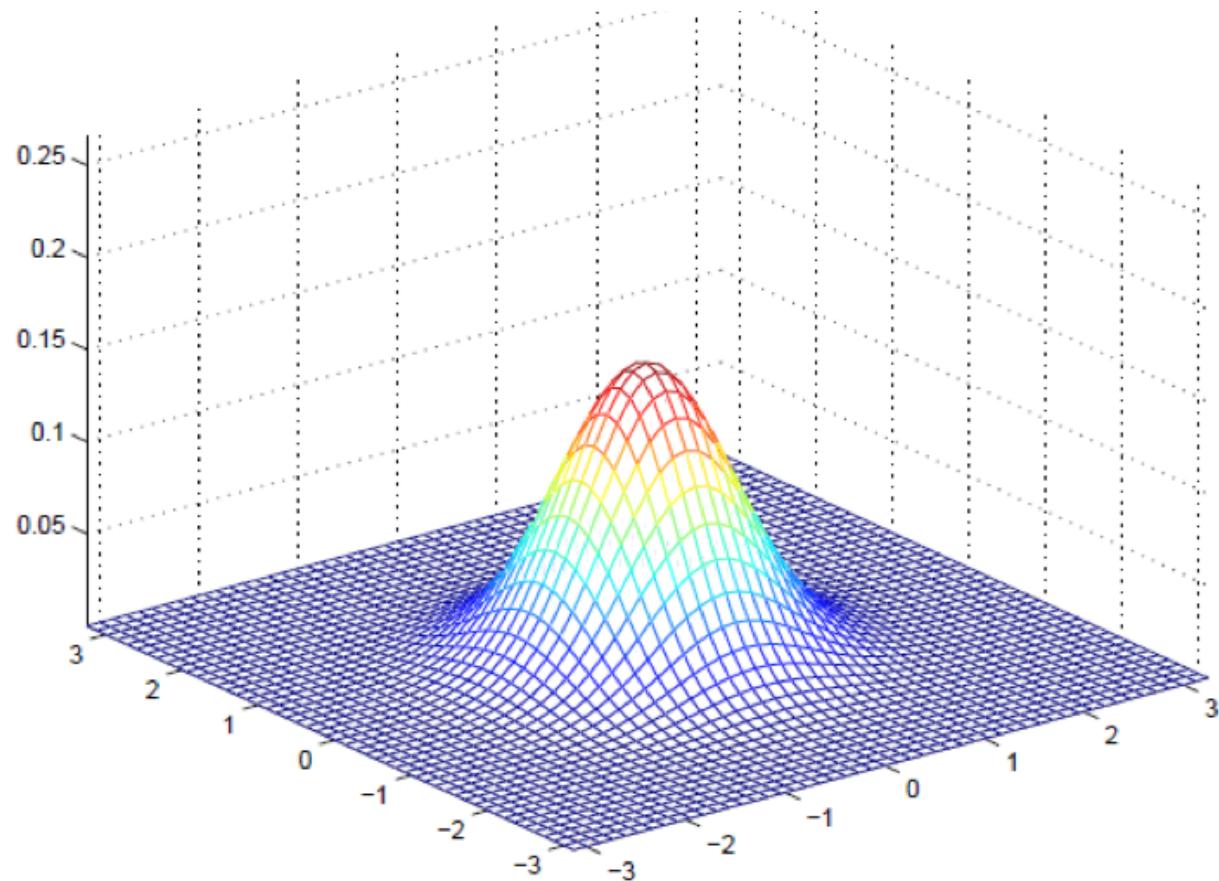
Note: the precision matrix or concentration matrix is just

the inverse covariance matrix,  $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$

A spherical or isotropic covariance  $\boldsymbol{\Sigma} = \sigma^2 \mathbf{I}_D$ ,  
has one free parameter.

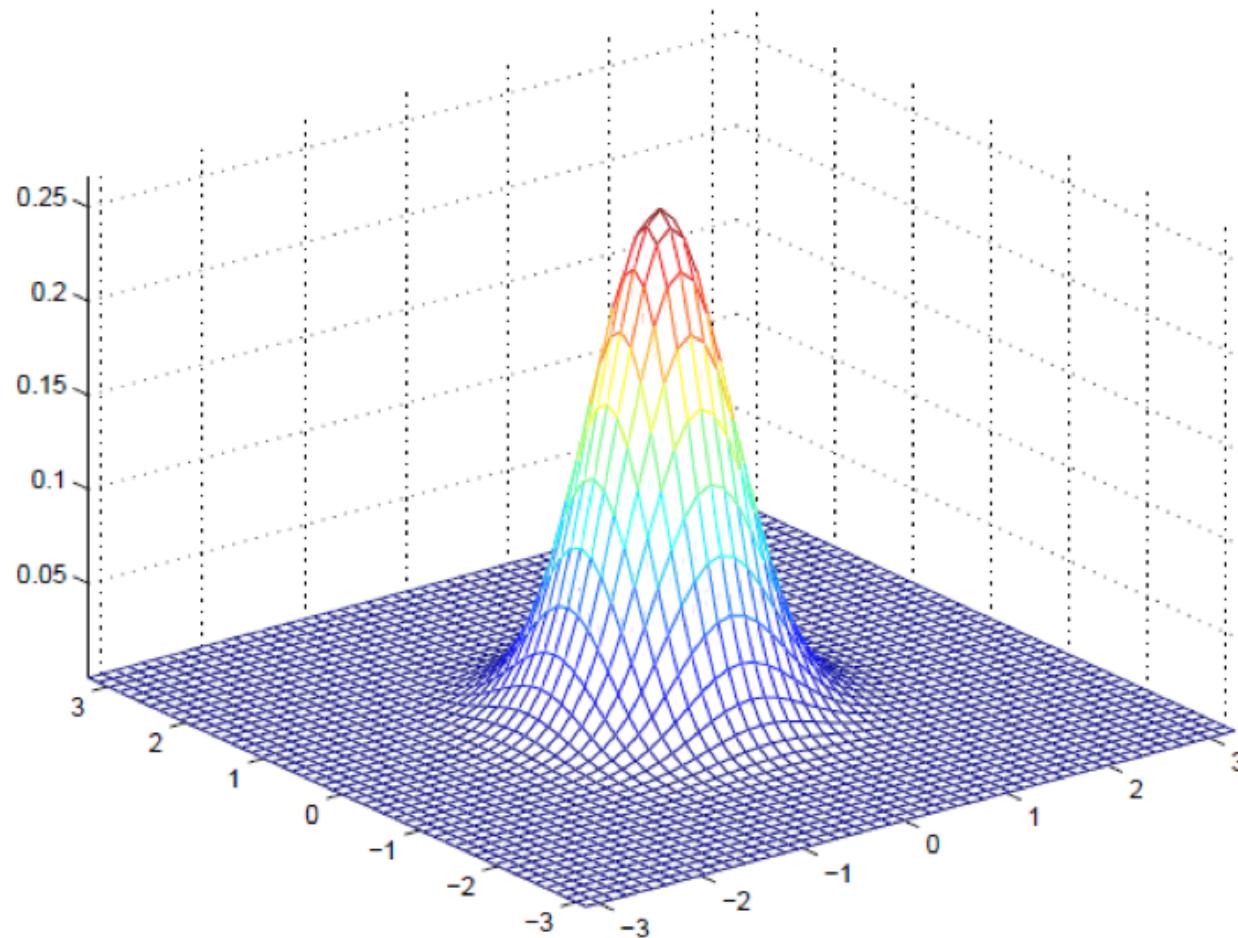
$$\mu = [0; 0]$$

$$\Sigma = [I \ 0; 0 \ I]$$



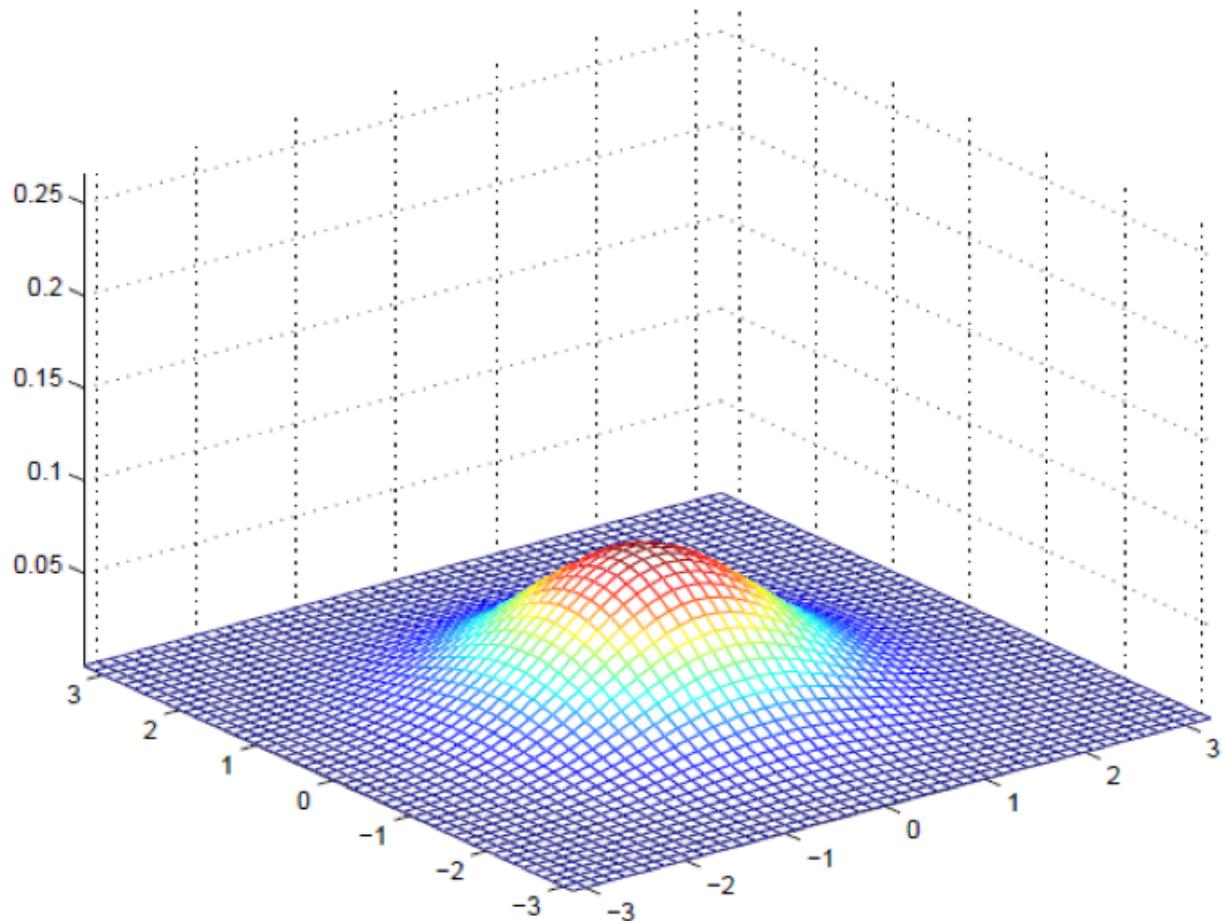
$$\mu = [0; 0]$$

$$\Sigma = [.6 \ 0 ; 0 .6]$$



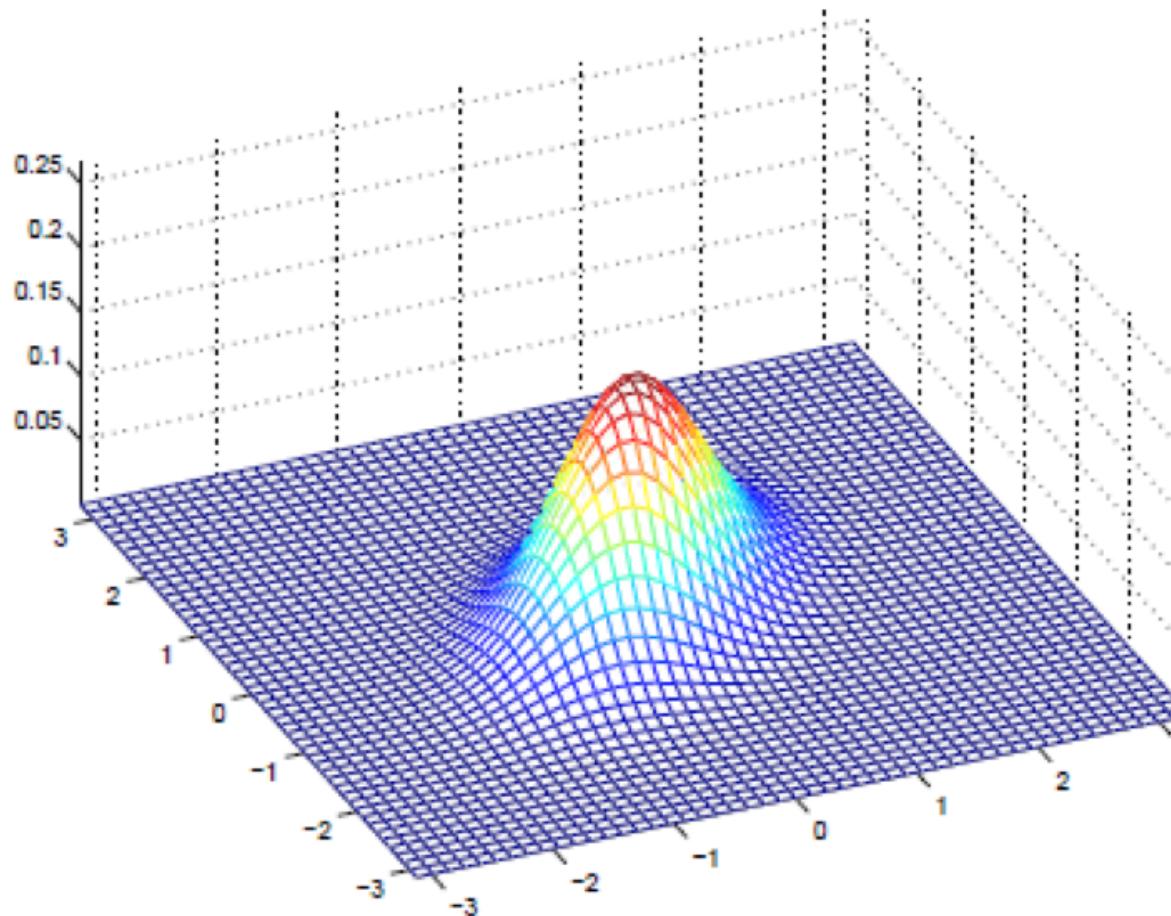
$$\mu = [0; 0]$$

$$\Sigma = [2 \ 0 ; 0 \ 2]$$



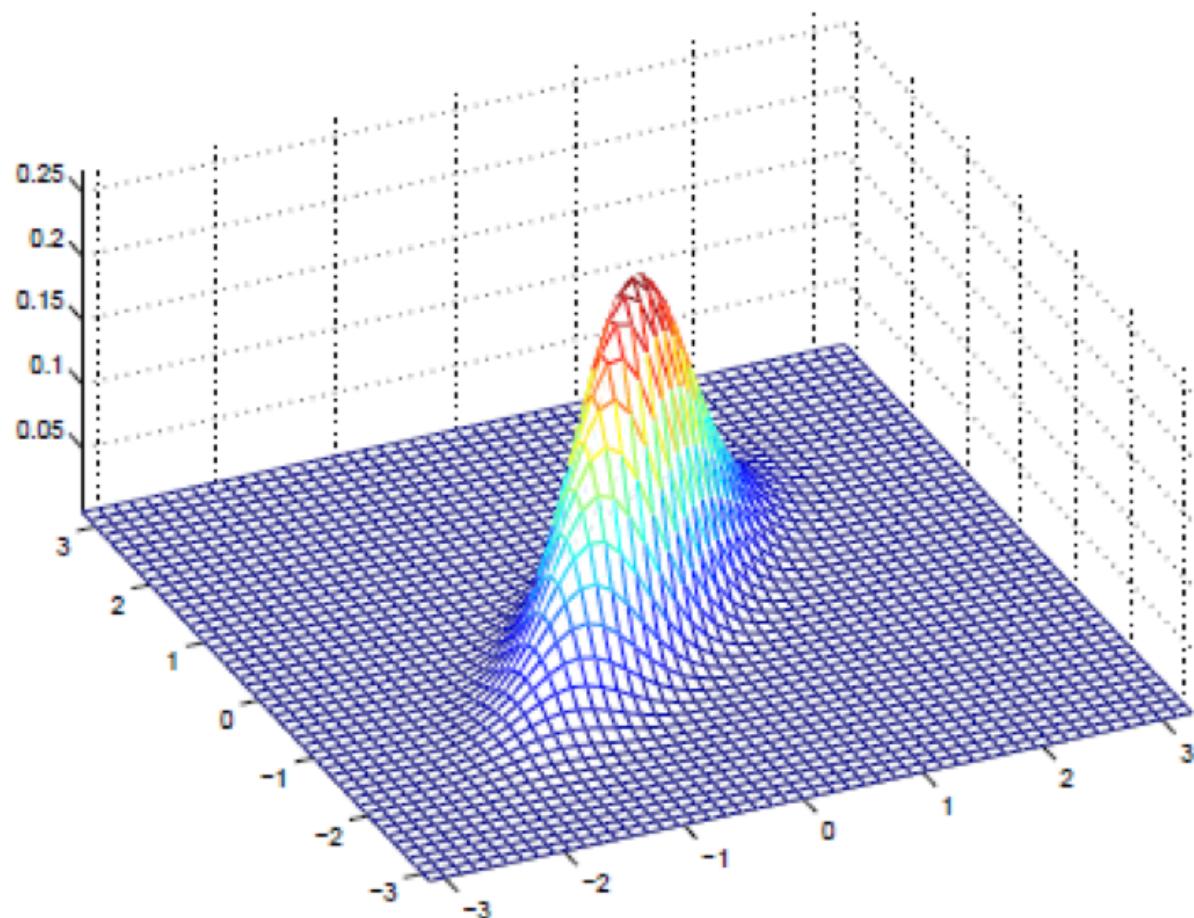
$$\mu = [0; 0]$$

$$\Sigma = [1 \ 0.5; 0.5 \ 1]$$



$$\mu = [0; 0]$$

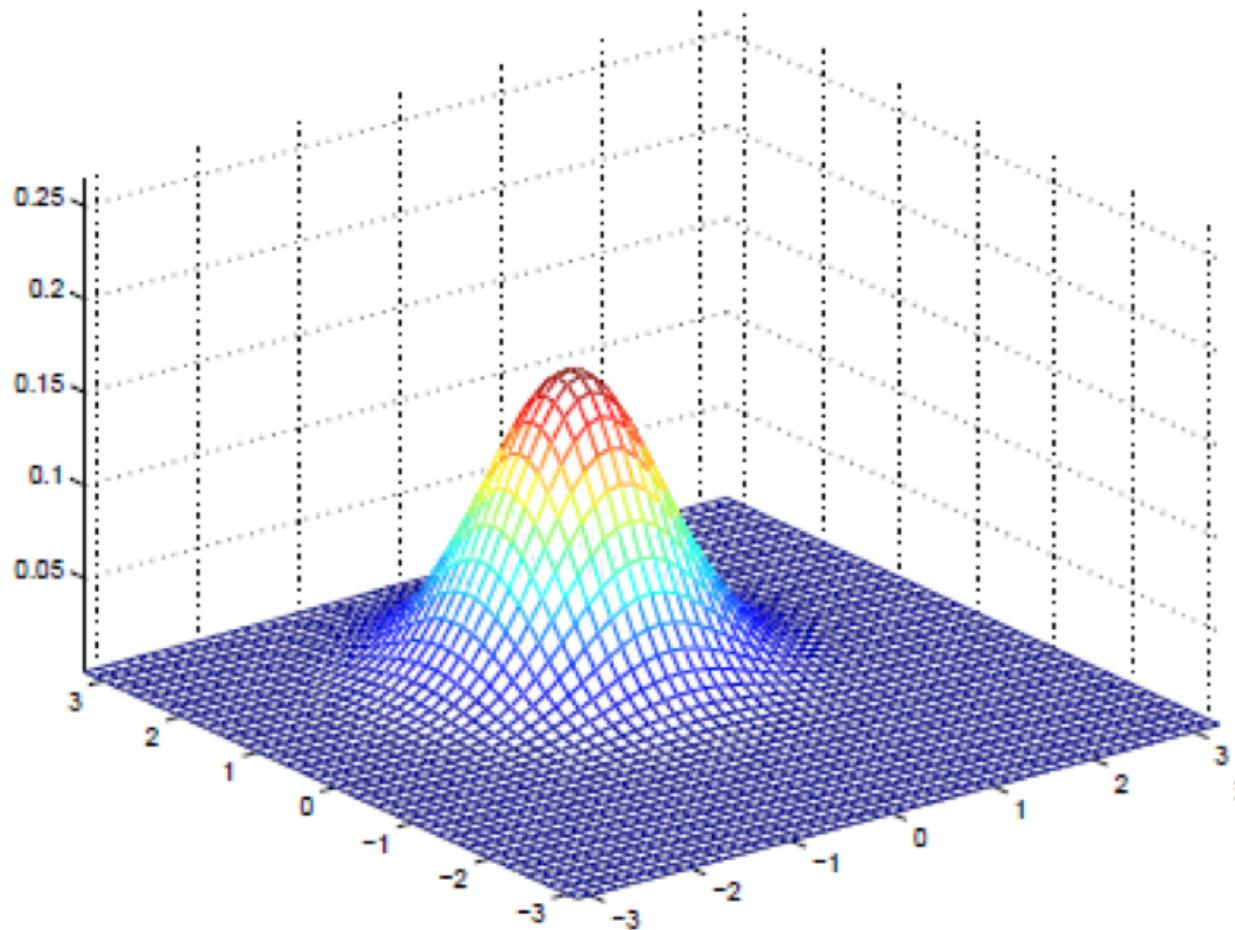
$$\Sigma = [1 \ 0.8; 0.8 \ 1]$$



# Now let's visualize as $\mu$ changes

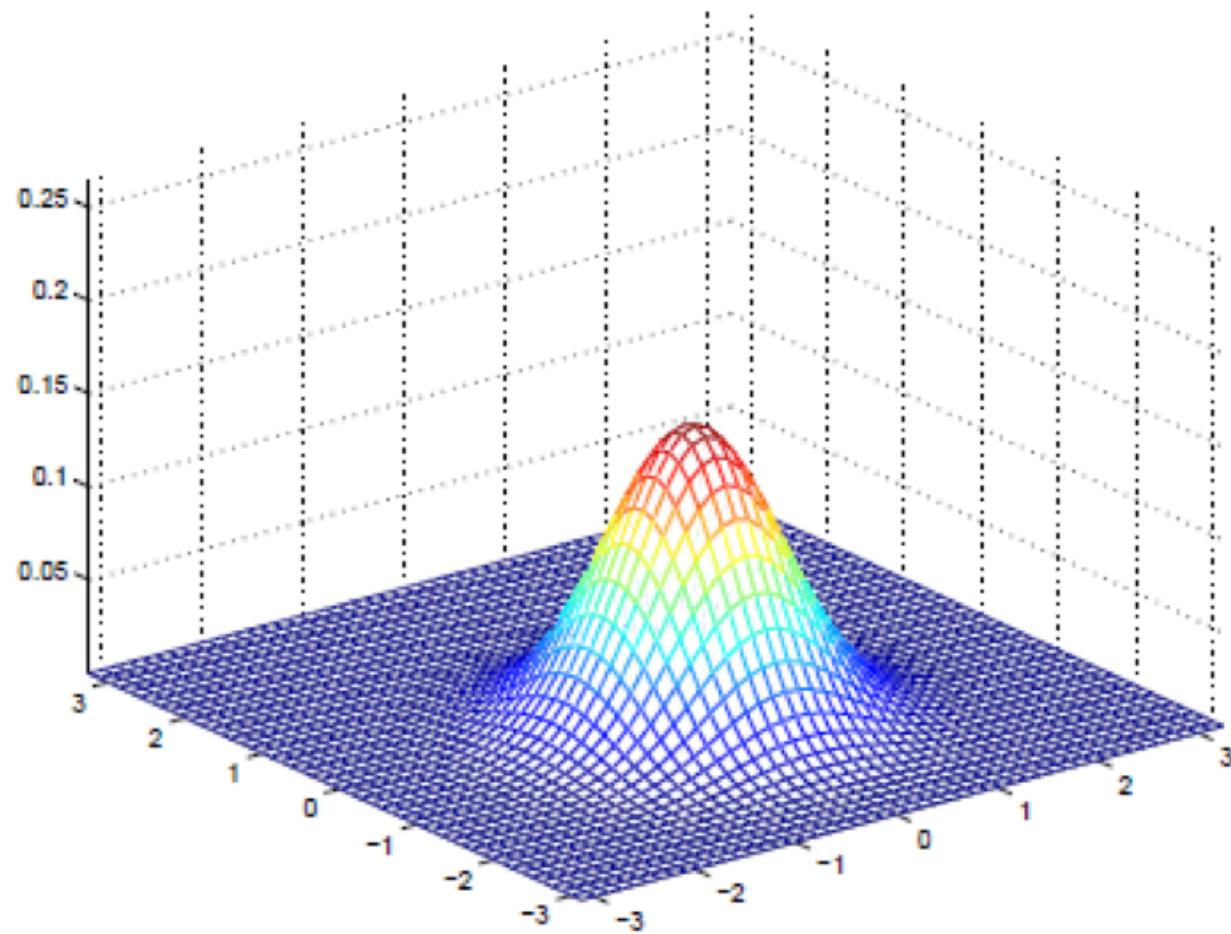
$$\mu = [1; 0]$$

$$\Sigma = [1 \ 0; 0 \ 1]$$



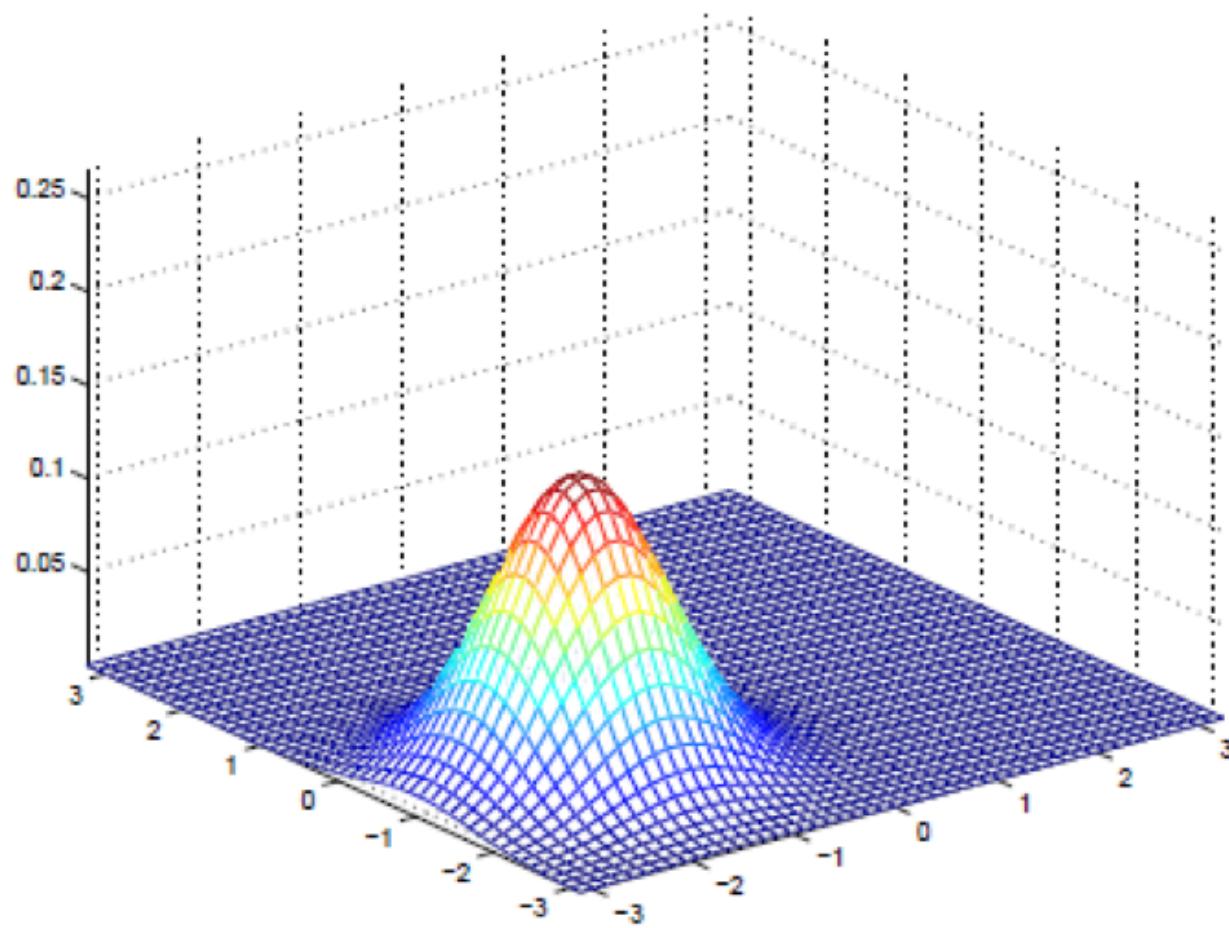
$$\mu = [-.5; 0]$$

$$\Sigma = [1 \ 0; 0 \ 1]$$



$$\mu = [-1; -1.5]$$

$$\Sigma = [1 \ 0; 0 \ 1]$$



# Level sets visualization

$$\mu = [0; 0]$$

$$\Sigma = [1 \ 0; 0 \ 1]$$

$$\mu = [0; 0]$$

$$\Sigma = [1 \ 0.5; 0.5 \ 1]$$

$$\mu = [0; 0]$$

$$\Sigma = [1 \ 0.8; 0.8 \ 1]$$

