Lecture 2

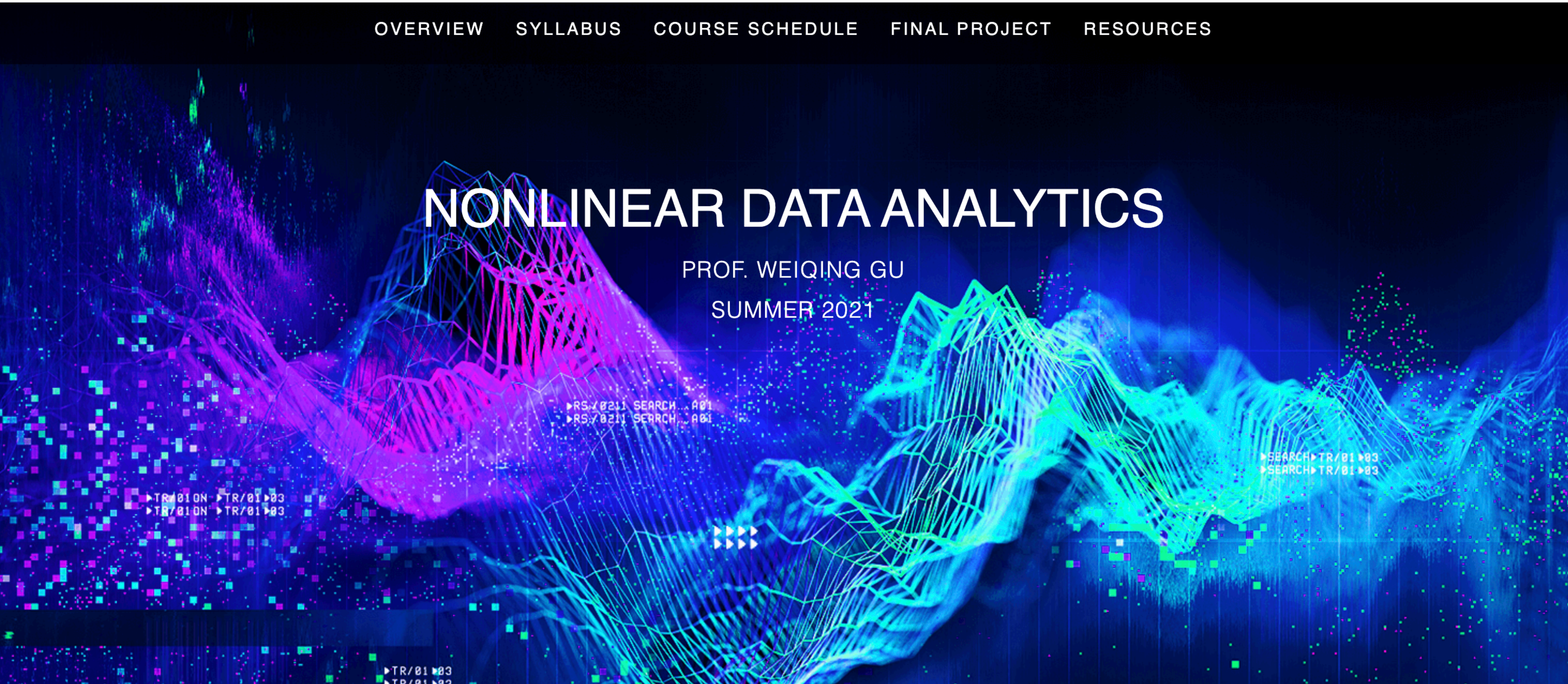
Math 178

Nonlinear Data Analytics

Prof. Weiqing Gu

Course Webpage

- https://non-linear-big-data.github.io/class-website/

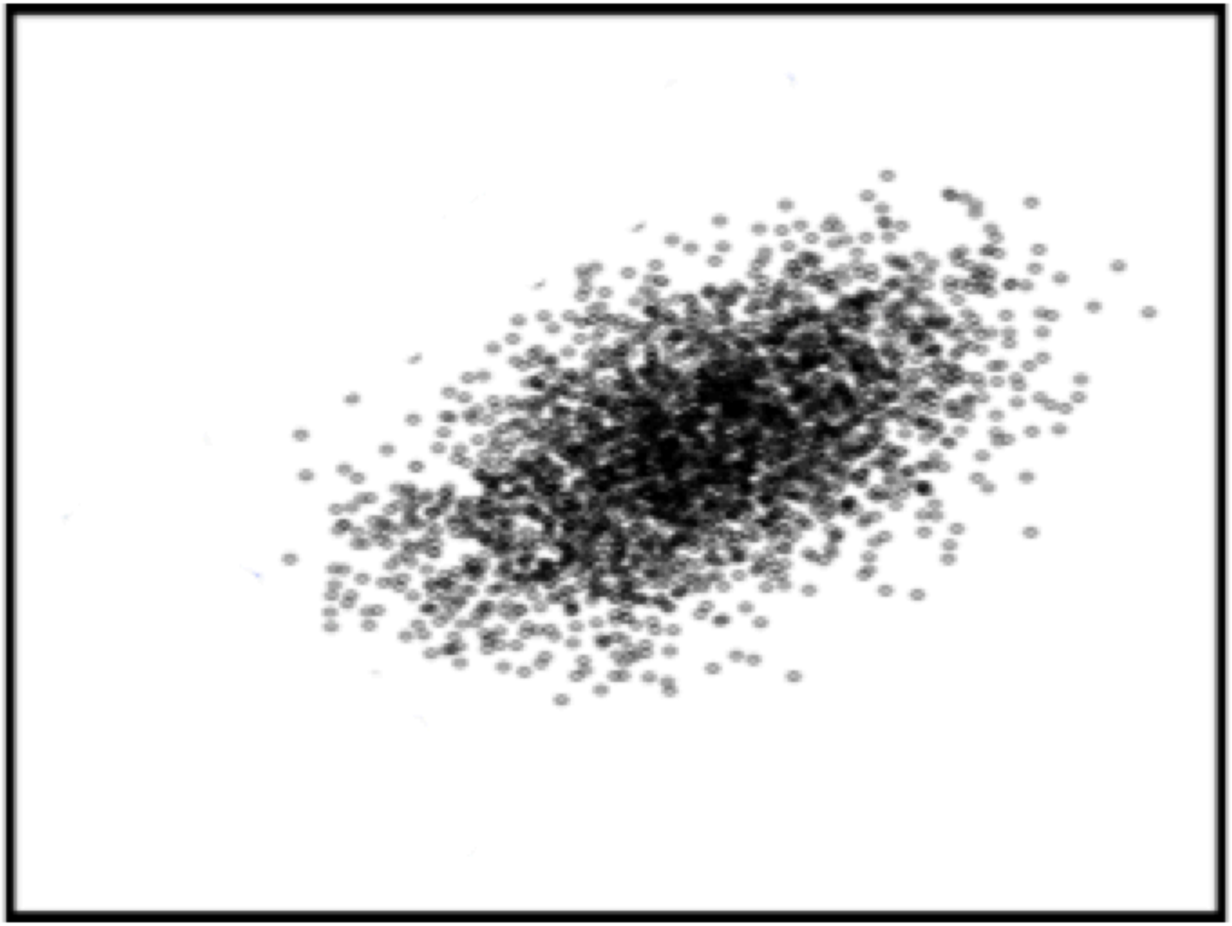


Today's topics

- ML: Spectral Analysis and its Geometry with Examples:
  - Principal Component Analysis (PCA)
  - Singular Value Decomposition (SVD)
  - Quadratic Surfaces
- Applications, especially for Data Compression.
- Curves
  - Frenet frame (*a classical example of moving frames)*
  - Frenet formula
  - Curvature and Torsion
  - Fundamental Theorem of the Local Theory of Curves

Principal Component Analysis (PCA)

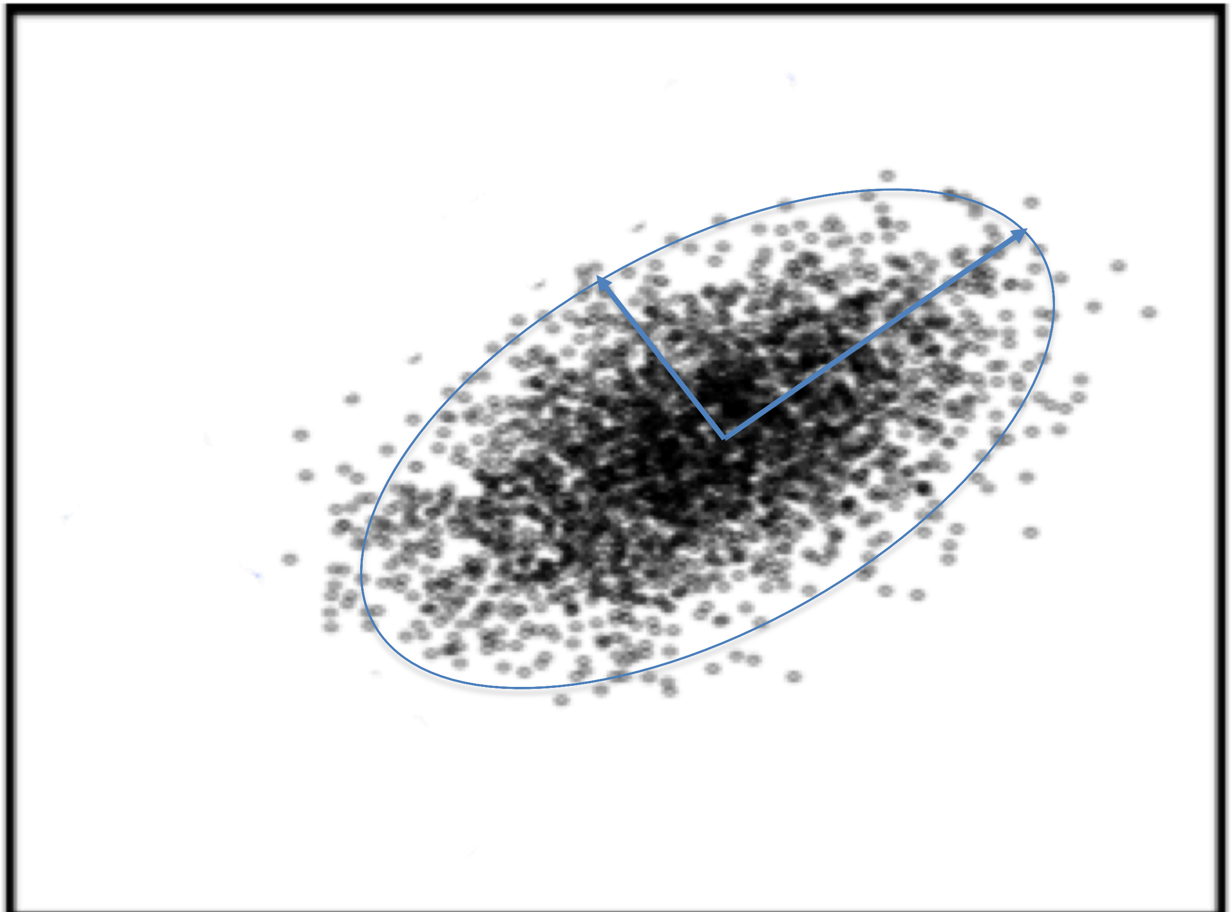
An intuitive example



What kind of geometric curve will include almost all the data above?

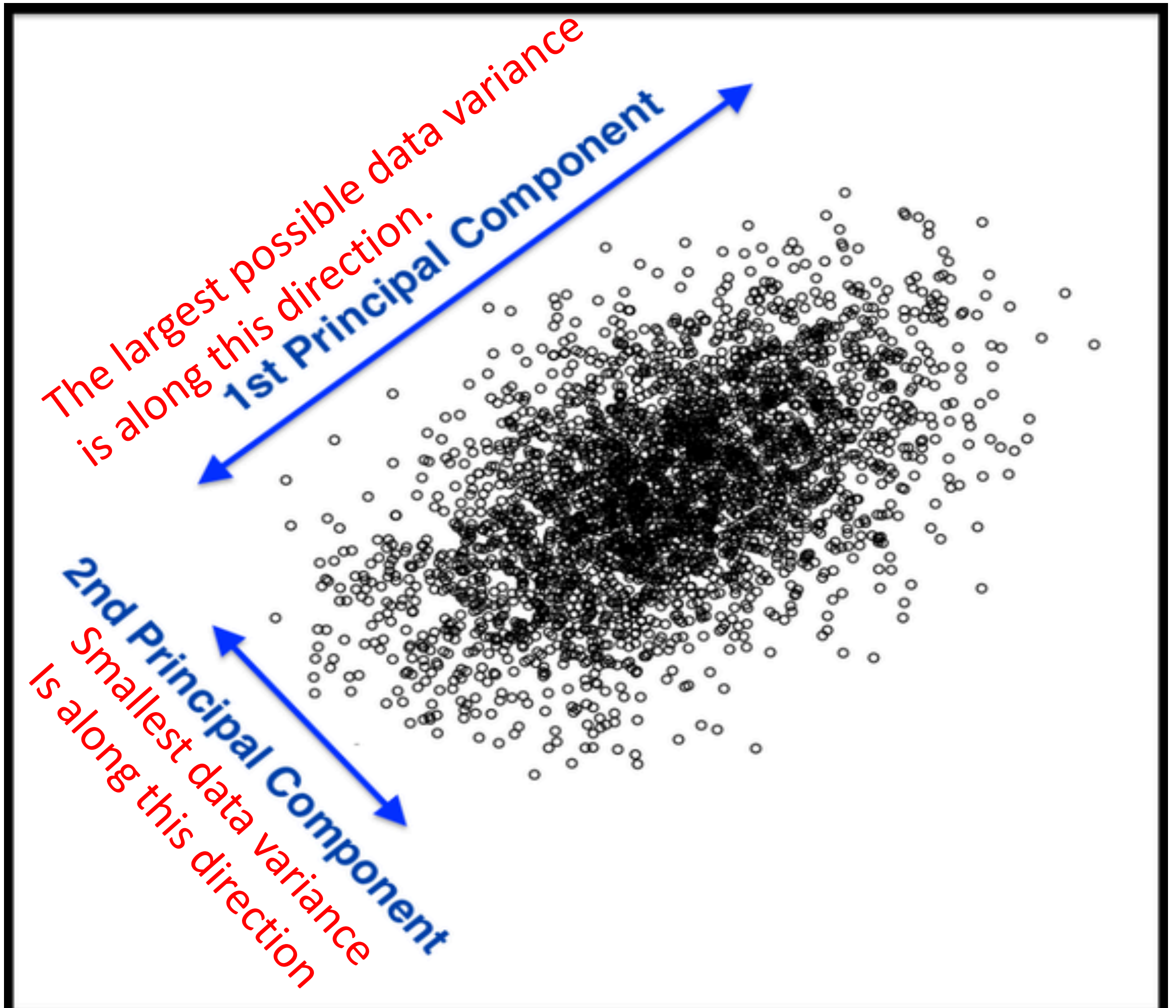
Principal Component Analysis (PCA)

An intuitive example: Geometry

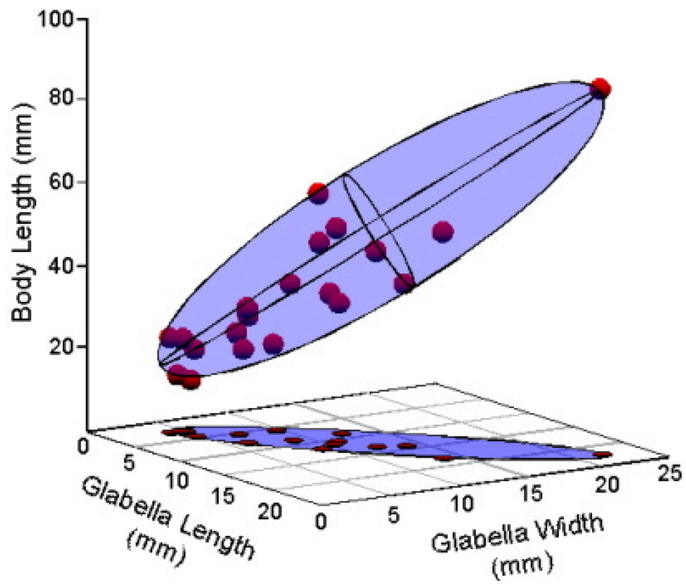


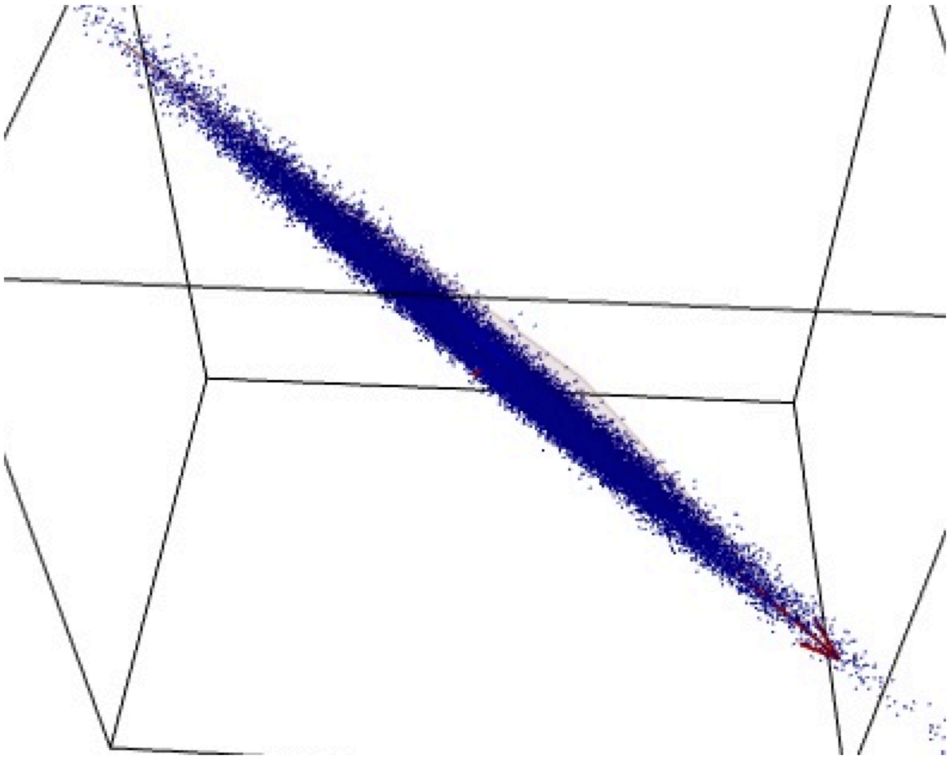
What kind of geometric curve will include almost all the data above?

Degree of Data Variation in different directions



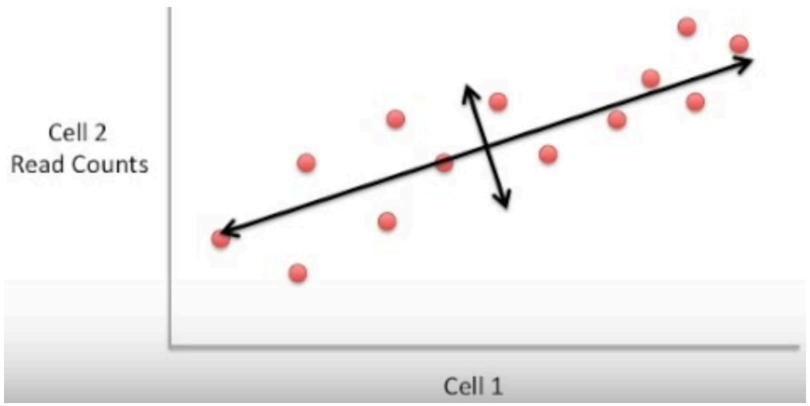
Similarly in Higher Dimensional

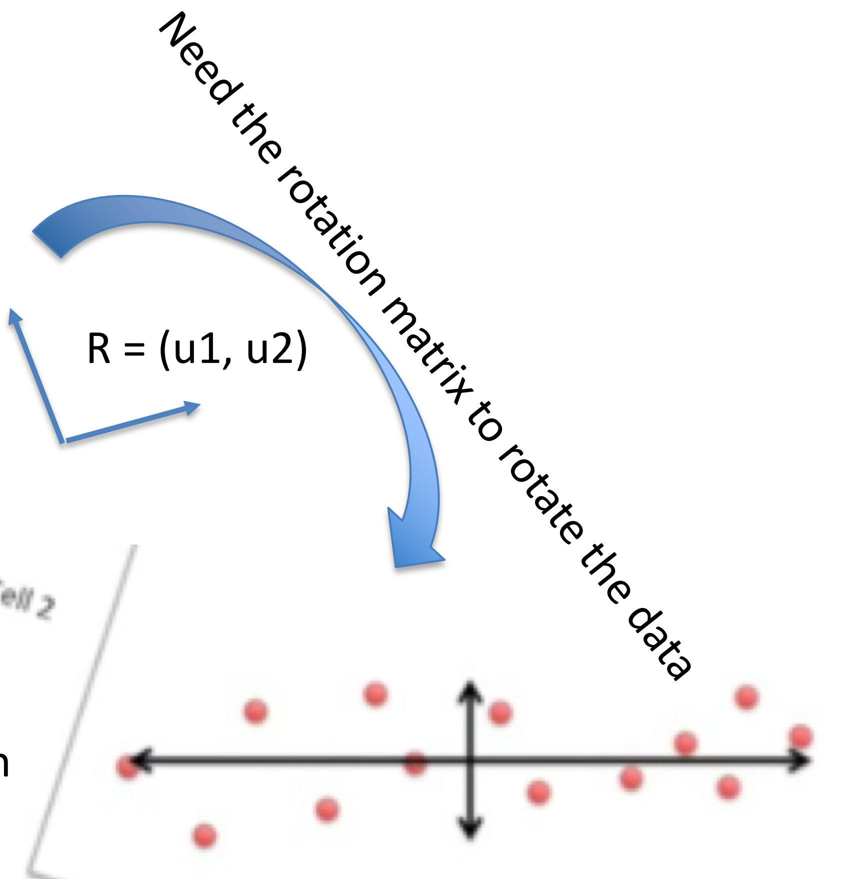




All the data can be projected to 1-dimensional line!

Need to Identify the Rotation Matrix

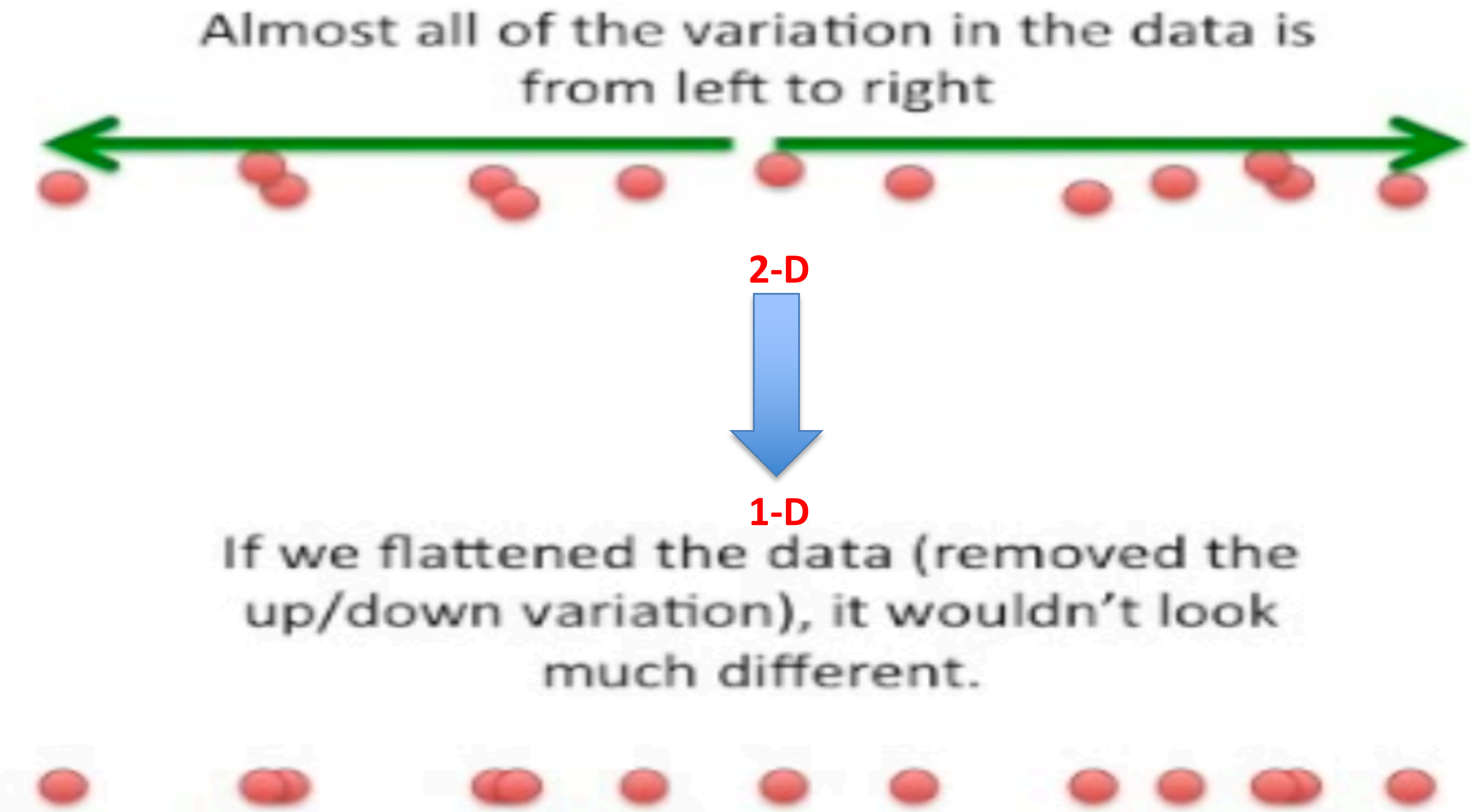




Key: How to find this orthogonal transformation
i.e. the orthogonal matrix?

Use orthogonal eigenvectors of the covariance matrix of the data as columns of this rotation matrix!

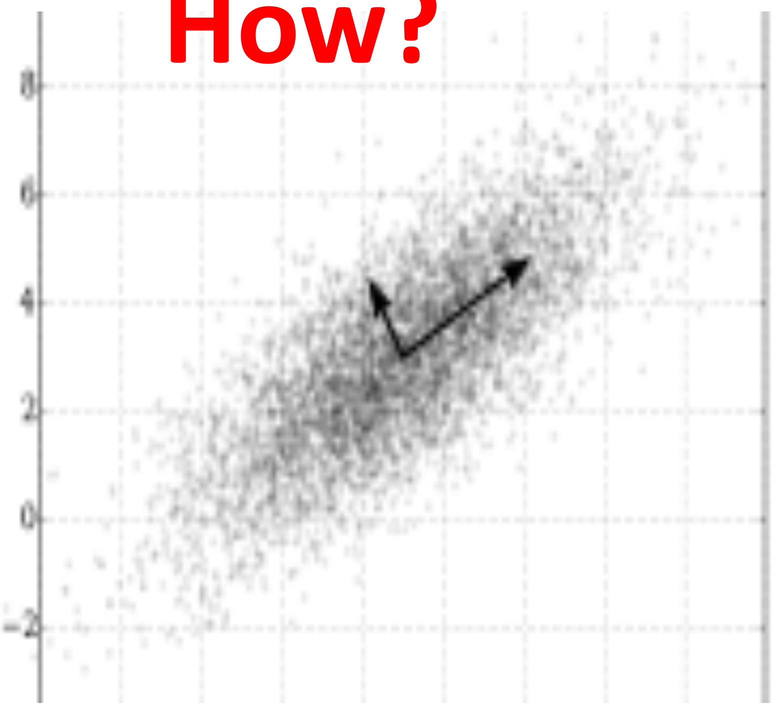
Dimension Reduction



The number of principal components is less than or equal to the number of original variables.

Geometrically, we want to find two axis directions of the elliptic curve. They are called **principal axes.**

How?



Note: the x-value and y-value of the data are correlated.
Their correlation are reflected by the covariance matrix of the data.

- **Principal component analysis** (**PCA**) is a procedure to find an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated.
- **Procedure of find the principal directions:**
- Step 1: Find the covariance matrix of the data directly (Note: one can first **standardize data**:
  - Find the mean μ_1 of the x-value & the mean μ2 of the y-value.
  - Subtract all x-value by μ_1 and subtract all y-value by μ_2. *Geometrically, move the x-axis and y-axis to the data center*.)
- Step 2: Find eigenvalues and eigenvectors of the covariance matrix of the data.
- Step 3: Order the eigenvalues from largest to smallest. The eigenvector corresponding to the largest eigenvalue is called the 1st principal axis. So on and so forth.
- Step 4: Form the rotation matrix using the corresponding O.N. eigenvectors.

Why do some people use X^TX for PCA? And others just use data matrix X instead?

Recall: $$\mathrm{cov}(X,Y) = \mathrm{E}\left[(X - \mathrm{E}[X])(Y - \mathrm{E}[Y])\right]$$

If each variable has a finite set of equal-probability values, x_i and y_i respectively for I = 1,…,n, then the covariance can be equivalently written in terms of the means E(X) and E(Y) as

$$\mathrm{cov}(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - E(X))(y_i - E(Y)).$$

If the data is standardized, then E(X) =E(y) = 0, then

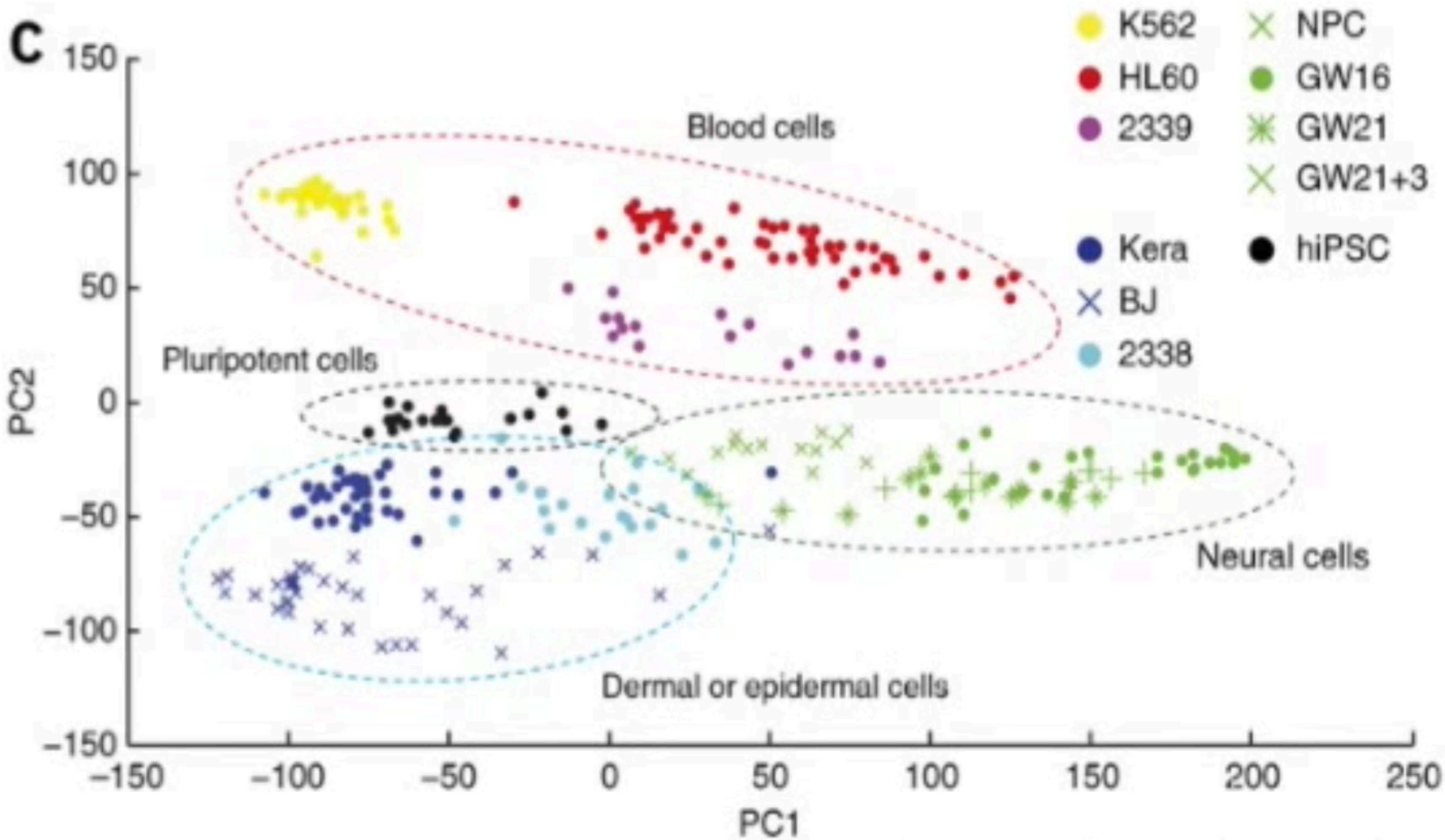
$\mathrm{cov}(X, Y) = (1/n)\, X^TY.$

The covariance matrix will be $(1/n)\mathbf{X}^T\mathbf{X}$.

Eigenvalues of $(1/n)\ \mathbf{X}^T\mathbf{X}$ will be just certain uniform scale of $\mathbf{X}^T\mathbf{X}$. It would not affect the ascent order the eigenvalues if people use $\mathbf{X}^T\mathbf{X}$.

If one uses the data matrix **X** directly, then one has to do a SVD. For details, see SVD next. ($X = USV^T$, and $X^T X = VDV^T$, So V is the one.)

Principal Component Analysis (PCA) has a lots of applications in Biotechnology.



Pollen et al. Nature Biotechnology 201

Switch to

- **Singular Value Decomposition (SVD)**

Singular Value Decomposition (SVD)

- What is SVD?
- It is a generalization of the notion of eigenvectors from *square* matrices to *any kind of matrix*.
- If A is a diagonalizable *square* matrix, then $A = PDP^{-1}$.
- If A is symmetric, then $A = PDP^T$. Where $P^T=P^{-1}$ (i.e. $PP^T=I$)

Here, D = diagonal matrix with eigenvalues on the diagonal
And P = Each column is the corresponding eigenvector

- X =(real) N × D matrix, then

$$\boxed{\underbrace{\mathbf{X}}_{N\times D} = \underbrace{\mathbf{U}}_{N\times N} \underbrace{\mathbf{S}}_{N\times D} \underbrace{\mathbf{V}^T}_{D\times D}}$$

The columns of U are called the left singular vectors, and the columns of V are called the right singular vectors of X.

U=N×N orthogonal matrix. i.e. $U^TU= UU^T=I$
V=D×D orthogonal matrix. i.e. $V^T V = VV^T = I$), and
S = N × D matrix containing the r = min(N,D) **singular values** $\sigma^i \geq 0$ on the main diagonal, with 0s filling the rest of the matrix.

How to decompose X? What are the singular values?

- Since X^TX is symmetric, so there exist an orthogonal matrix V such that

  $X^T X = VDV^T$, where $V = \text{evec}(X^T X)$

  eigenvectors

- But XX^T is also symmetric, so there exist an orthogonal matrix U such that

  $X X^T = UD_1U^T$, where $U= \text{evec}(XX^T)$.

- How D and D_1 are related? **Claim: XX^T and X^TX have the same *nonzero* eigenvalues!**
  - Let eval(XX^T)= the set of non zero eigenvalues of XX^T
  - and eval($X^T X$) =the set of non zero eigenvalues of $X^T X$, we can show eval(XX^T) = eval($X^T X$).

Theorem: XX^T and $X^T X$ have the same none zero eigenvalues.

- That is to show eval(XX^T) = eval($X^T X$).
- Work out details with the students on the board.
- Now you order the eigenvalues of XX^T, in the same way as you order the eigenvalues of $X^T X$ in an ascent order. Since all the eigenvalues are ≥0, then following the non zero eigenvalues there should be all zero eigenvalues.
- Now you find the U with the column putting eigenvectors corresponding to the ordered eigenvalues. Then U will be unique.
- Now $X^T X = VDV^T = VD^{1/2}D^{1/2}V$, we want to stick the U in the middle. But U is NxN. We have to be careful to make the dimension match. So we change $D^{1/2}$ to $S = N \times D$ matrix containing the $r = \min(N,D)$ singular values $\sigma^i \geq 0$, where σ^i is the sqaure root the corresponding eigenvalue. Then stick U^TU, which is an identity matrix, in the middle. Please see the figure on the next slide.

Effective computation for SVD

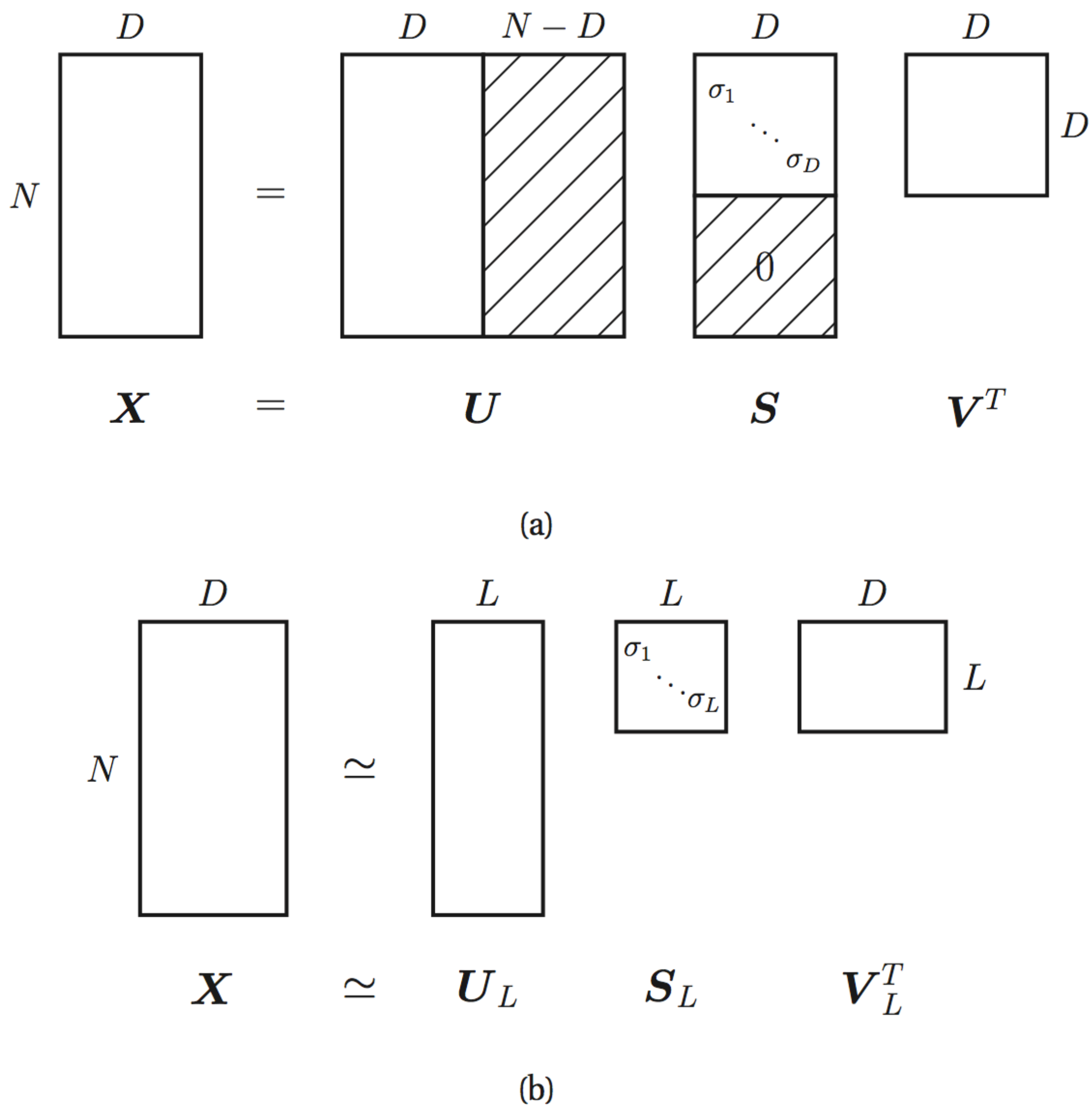
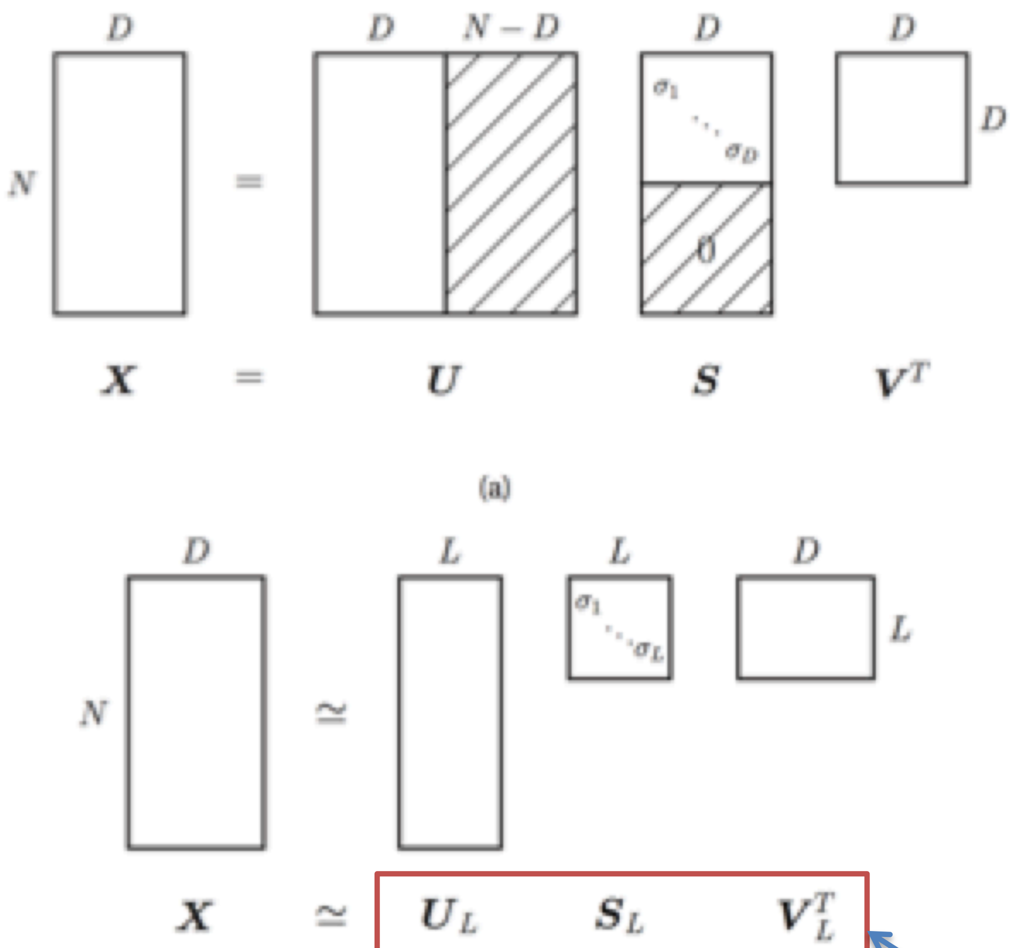


Figure 12.8 (a) SVD decomposition of non-square matrices $\mathbf{X} = \mathbf{USV}^T$. The shaded parts of $\mathbf{S}$, and all the off-diagonal terms, are zero. The shaded entries in $\mathbf{U}$ and $\mathbf{S}$ are not computed in the economy-sized version, since they are not needed. (b) Truncated SVD approximation of rank L.

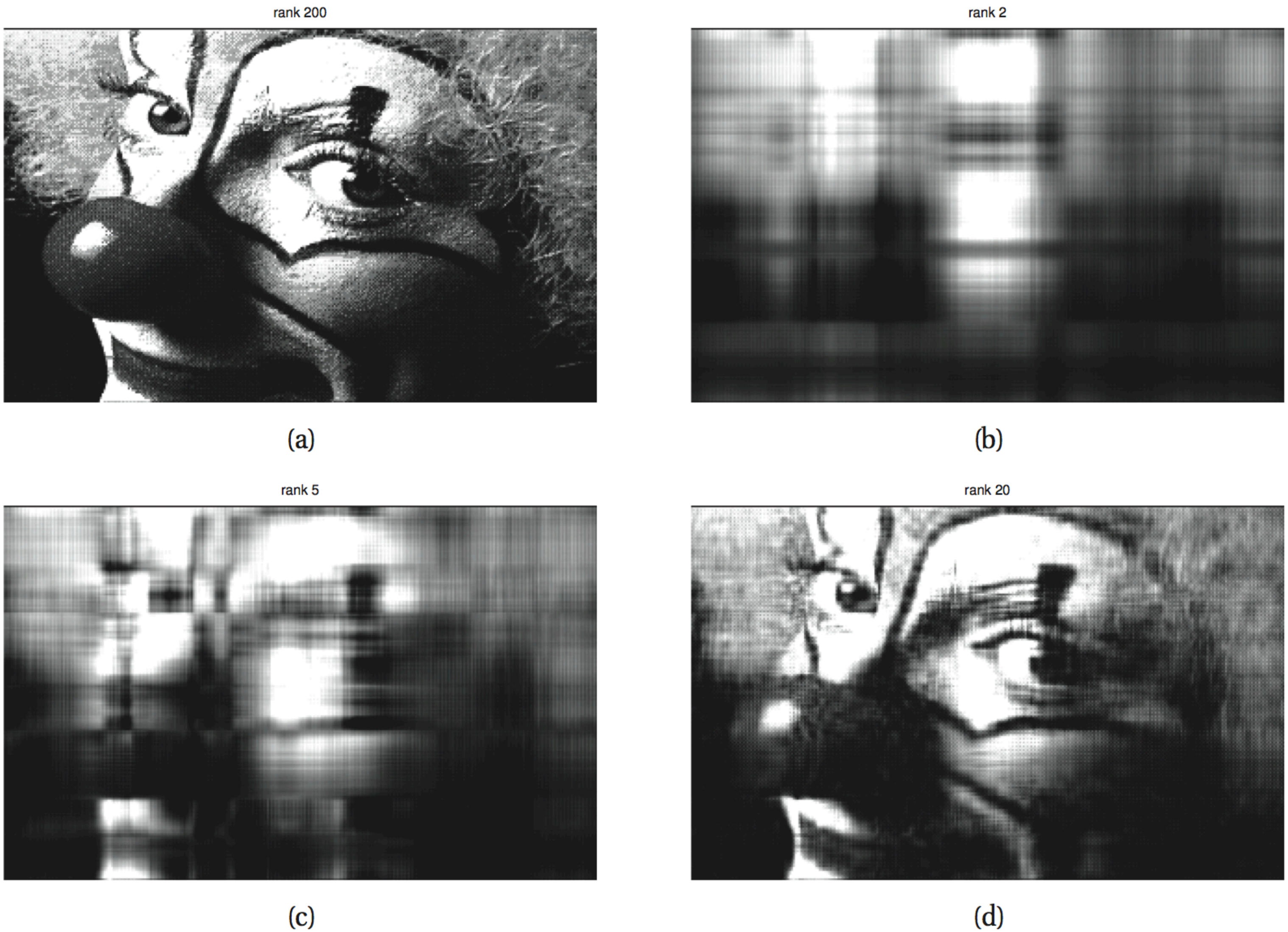
Application: SVD as a dimension-reduction technique



- Example: Data compression
- Rank the eigenvalues in descent order. Set all the small eigenvalues (equivalently singular values) to zero, you will compress the data.

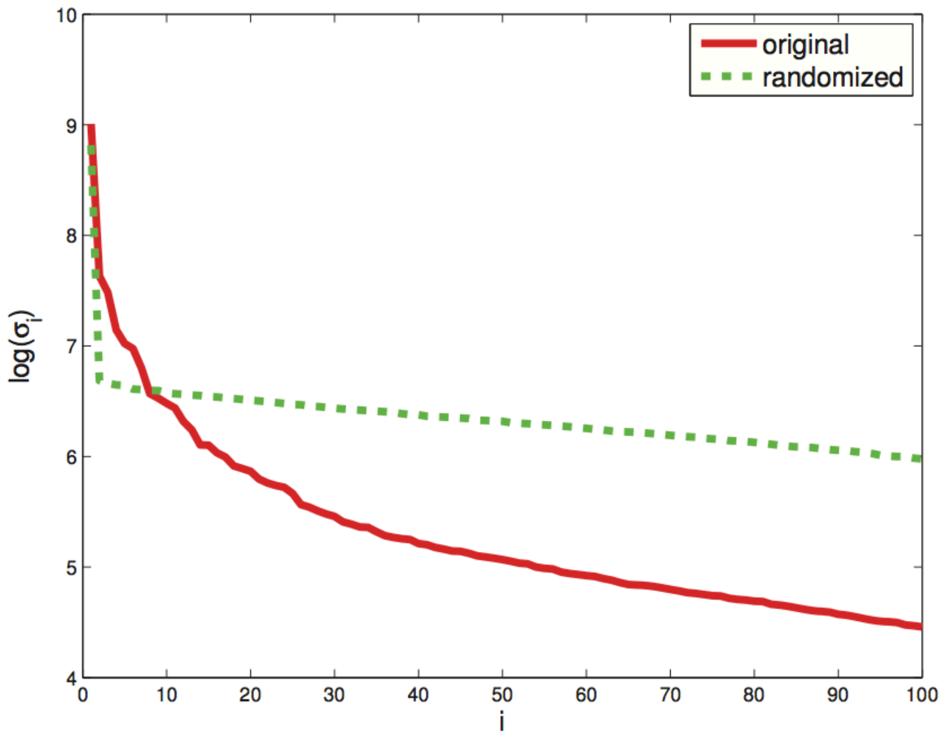
The error in this approximation is $||\mathbf{X} - \mathbf{X}_L||_F \approx \sigma_{L+1}$

Consider the 200 × 320 pixel image. This has 64,000 numbers in it. We see that a rank 20 approximation, with only (200 + 320 + 1) × 20 = 10, 420 numbers is a very good approximation.



Rank = 20 means setting all $\sigma_L = 0$ for $L \geq 21$. Error is about σ_{21}.

Figure 12.9 Low rank approximations to an image. Top left: The original image is of size 200 × 320, so has rank 200. Subsequent images have ranks 2, 5, and 20. Figure generated by `svdImageDemo`.



12.10 First 50 log singular values for the clown image (solid red line), and for a data
:d by randomly shuffling the pixels (dotted green line). Figure generated by `svdImageDemo`.

Data Compression

- If your boss wants to compress a data set, you may ask him/her what would be the threshold of an error acceptable. (Since naturally, compressing data will loss information, it is reasonable to ask the threshold of an acceptable error.)
- If your boss say 5%, you perform an SVD on your data, and find the singular value $\sigma_{L,}$ Say L = 30, which is close to 5% and then set all $\sigma_L = 0$ for $L \geq 31$. You know after the data compress, your error is about σ_{31} which is about 5%.

$$X \simeq \boxed{U_L \quad S_L \quad V_L^T}$$

These sizes of those matrices are much smaller.

Recall: **Correlation**

- Correlation of two random variables are defined by "normalizing" the covariance of the two random variables.
- If we have a random vector, then we can define a covariance matrix.
- Covariance matrix is symmetric matrix, and in fact it is semi positive definite matrix.
- So the covariance matrix can be diagonalized, with eigenvalues being none negative; which is the base for PCA.

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

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

Exercise

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

If **x** is a d-dimensional random vector, its **covariance matrix** is defined to be the following symmetric, semi positive definite matrix:

$$\mathrm{cov}[\mathbf{x}] \triangleq \mathbb{E}\left[(\mathbf{x}-\mathbb{E}[\mathbf{x}])(\mathbf{x}-\mathbb{E}[\mathbf{x}])^T\right]$$

Ofen denoted by Σ

$$= \begin{pmatrix} \mathrm{var}[X_1] & \mathrm{cov}[X_1,X_2] & \cdots & \mathrm{cov}[X_1,X_d] \\ \mathrm{cov}[X_2,X_1] & \mathrm{var}[X_2] & \cdots & \mathrm{cov}[X_2,X_d] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{cov}[X_d,X_1] & \mathrm{cov}[X_d,X_2] & \cdots & \mathrm{var}[X_d] \end{pmatrix}$$

correlation coefficient & correlation matrix

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

$$\mathrm{corr}\left[X,Y\right] \triangleq \frac{\mathrm{cov}\left[X,Y\right]}{\sqrt{\mathrm{var}\left[X\right]\mathrm{var}\left[Y\right]}}$$

- If X and Y are indep., then cov [X, Y] = 0; say X and Y are uncorrelated.

- A **correlation matrix** of a random vector has the form:

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

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

Example of Correlation Coefficients

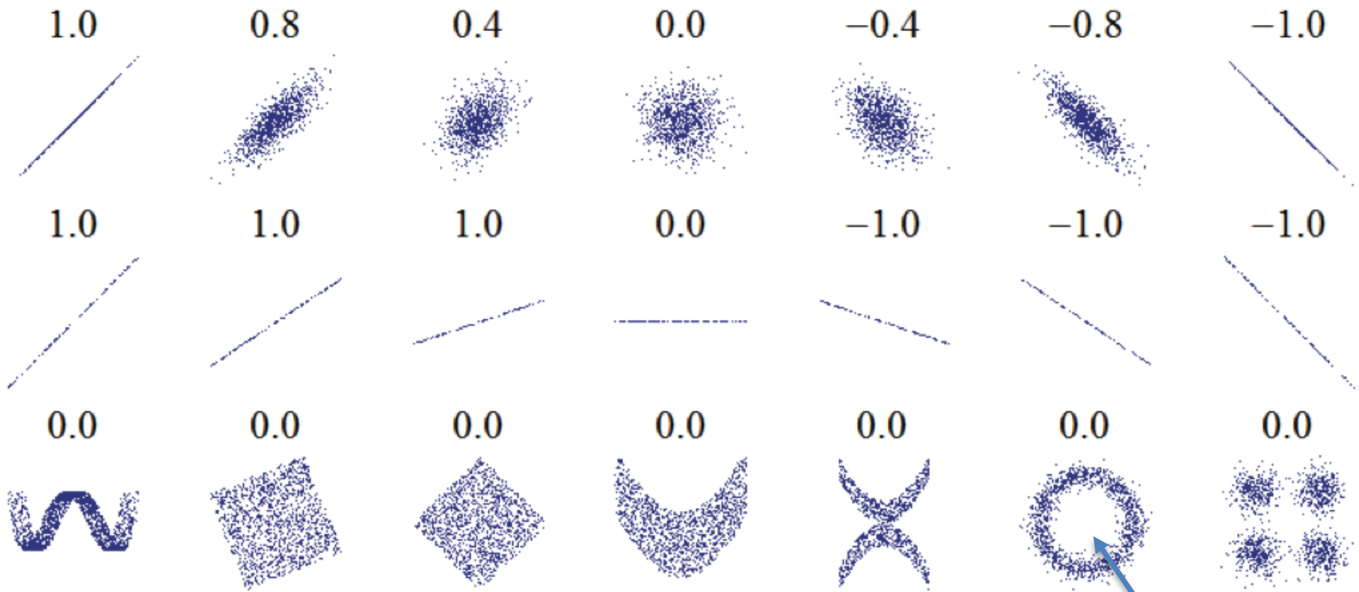


Figure 2.12 Several sets of (x, y) points, with the correlation coefficient of x and y for each set. Note that the correlation reflects the noisiness and direction of a linear relationship (top row), but not the slope of that relationship (middle), nor many aspects of nonlinear relationships (bottom). 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`

E.g. It did not detect the data living close to a circle.

Review

Diagonalization of Symmetric Matrices

Every symmetric matrix can be diagonalized over $\mathbb{R}$!

Moreover, there exists an orthogonal matrix P such that $P^{-1}AP = P^TAP = D$, a diagonal matrix.

WHY?

Diagonalization of Symmetric Matrices

Key 1

Theorem: *All the roots of the characteristic polynomial of a real symmetric matrix are real numbers.*

Key 2

If the symmetric matrix A has an eigenvalue λ of multiplicity k, then the solution space of the homogeneous system $(\lambda I_n - A)\mathbf{x} = 0$ has dimension k. This means that there exist k linearly independent eigenvectors of A associated with the eigenvalue λ.

Diagonalization of Symmetric Matrices

Key 3

Theorem : *If A is a symmetric $n \times n$ matrix, then eigenvectors that belong to distinct eigenvalues of A are orthogonal.*

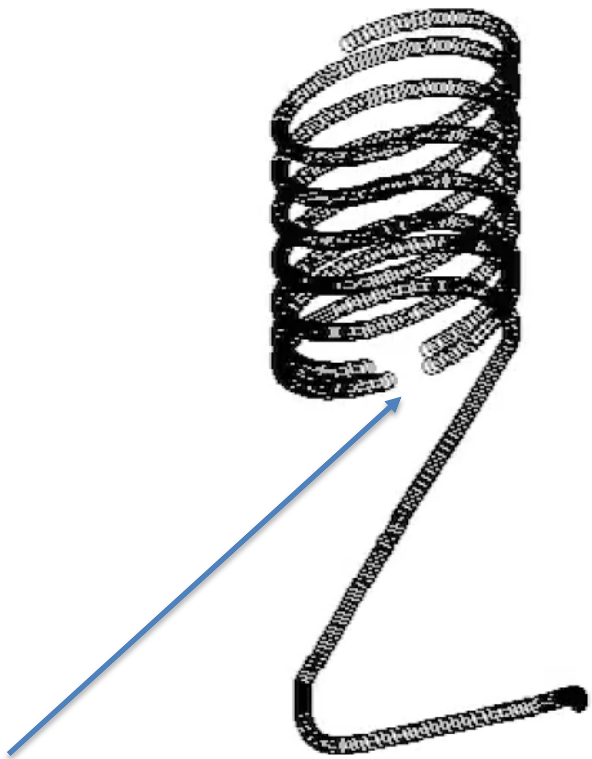
Key 4

By the Gram-Schmidt process we can choose an orthonormal basis for this solution space. Thus we obtain a set of k orthonormal eigenvectors associated with the eigenvalue λ. Since eigenvectors associated with distinct eigenvalues are orthogonal, if we form the set of all eigenvectors we get an orthonormal set. Hence, the matrix P whose columns are the eigenvectors is orthogonal.

Next

- Curves
- Frenet frame (*a classical example of moving frames)*
- Frenet formula
- Curvature and Torsion
- Fundamental Theorem of the Local Theory of Curves

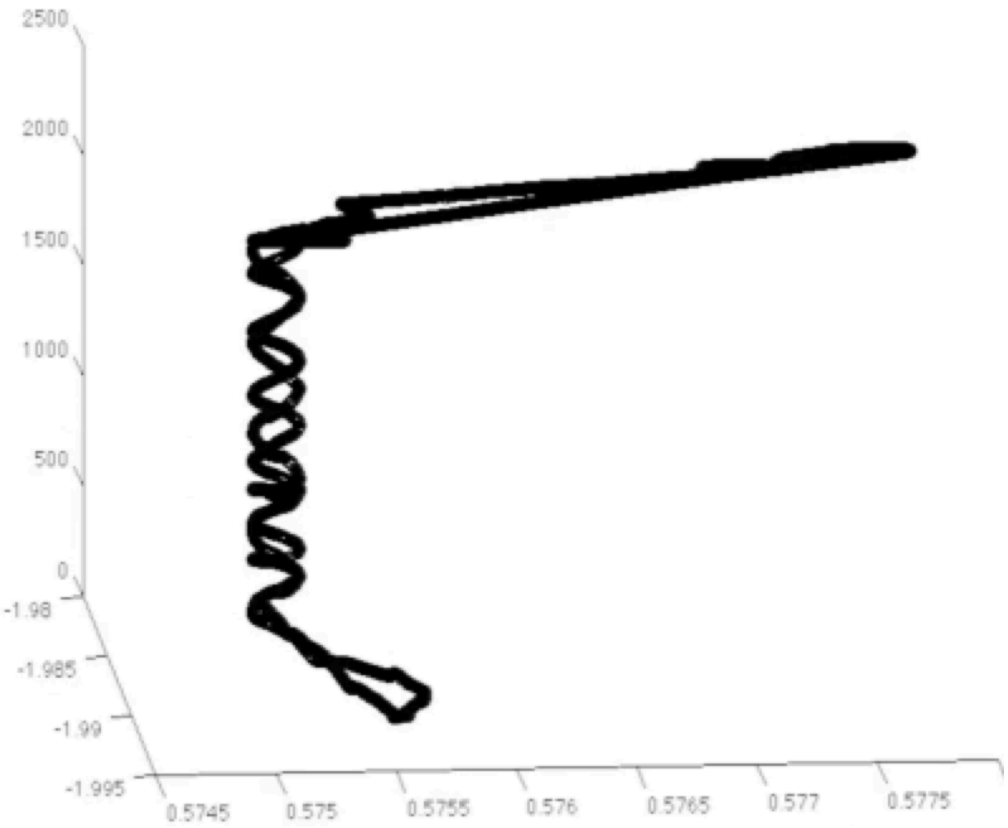
How to fill the missing data here?

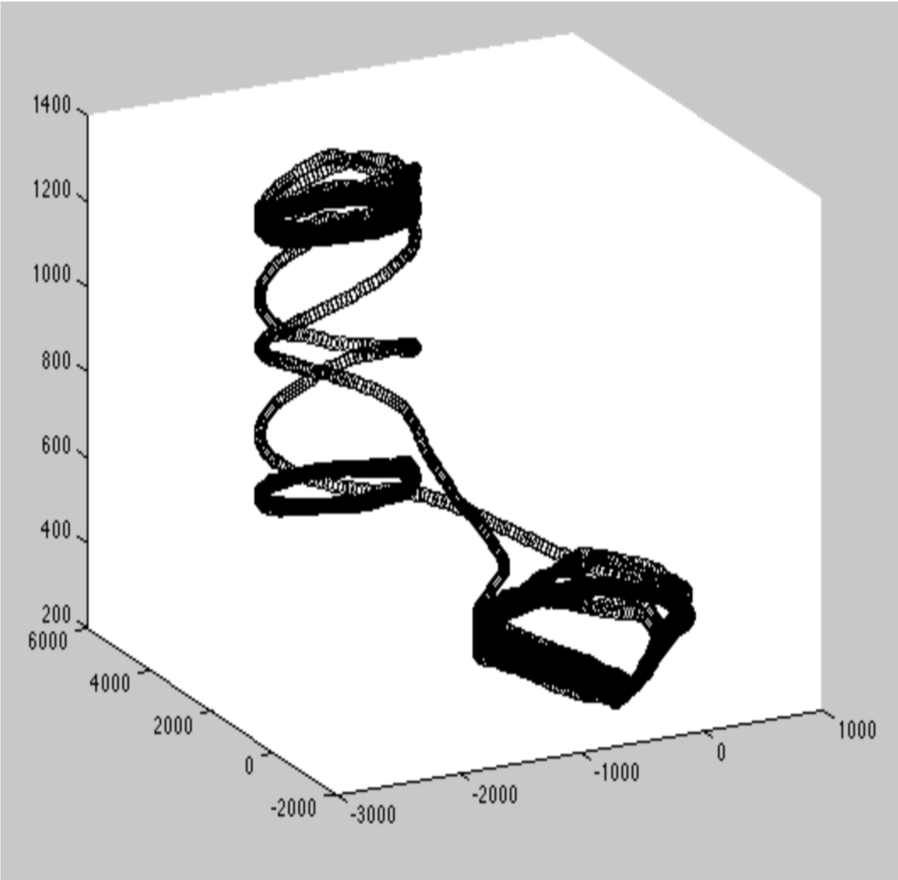


- This is just because of missing data.
- We can confirm it by the dynamics and kinematics of an UAV.

- Linear interpolation here may not make sense.
- We need to use other parts of the trajectory to predict how this missing part should look like.
- We need mathematically describe the trajectory.
- What kind of curve best describe the trajectory?

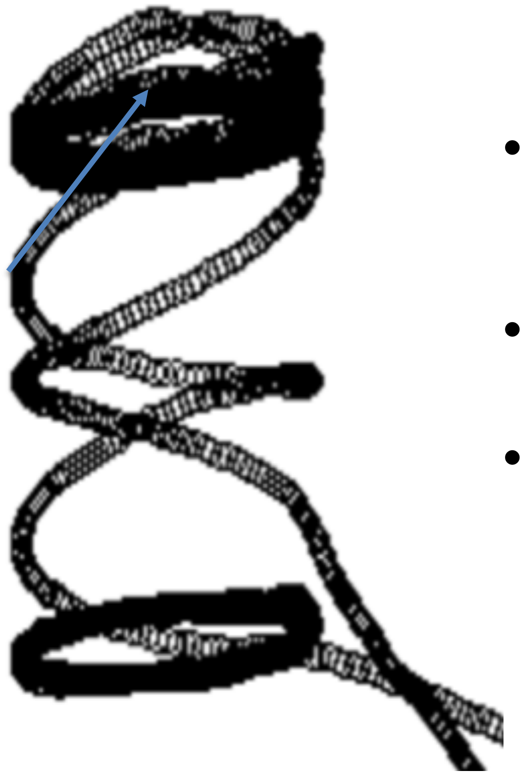
Here are the trajectory of Student1 and Student2



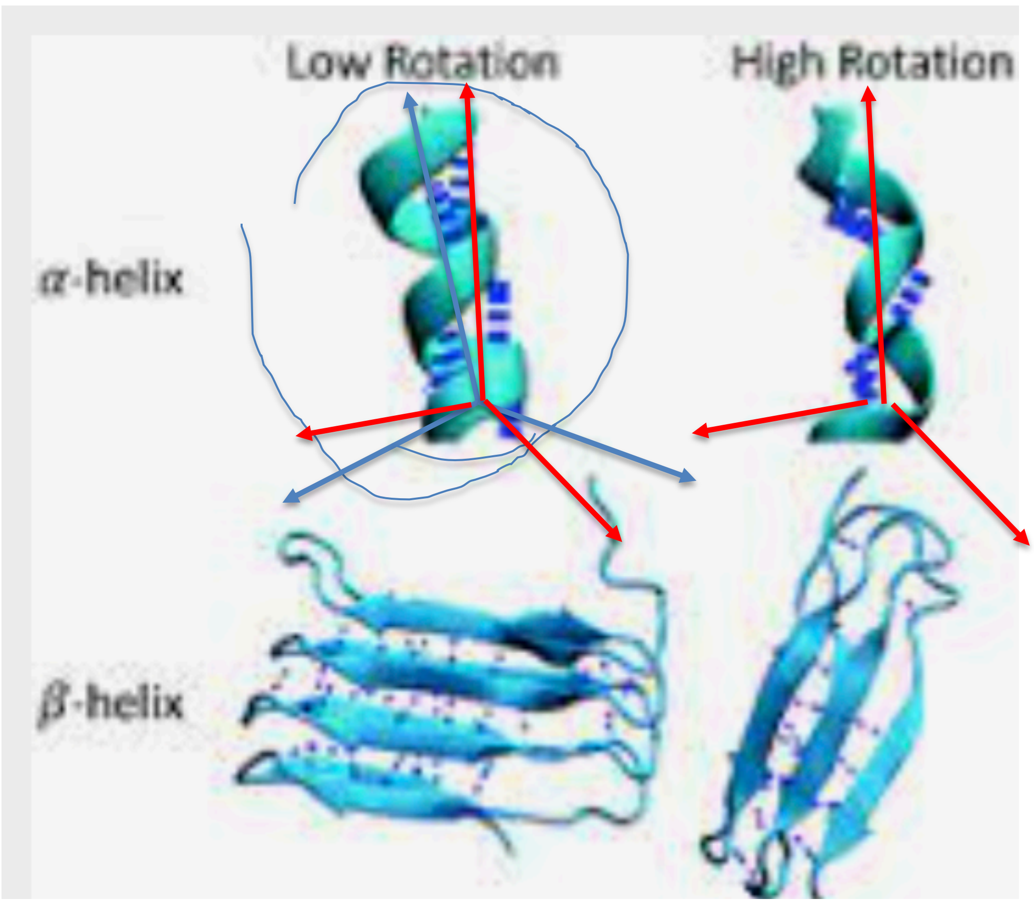


Q: What are the differences between the instructor's trajectory and that of the students?

Compare with the instructor's trajectory with that of the student



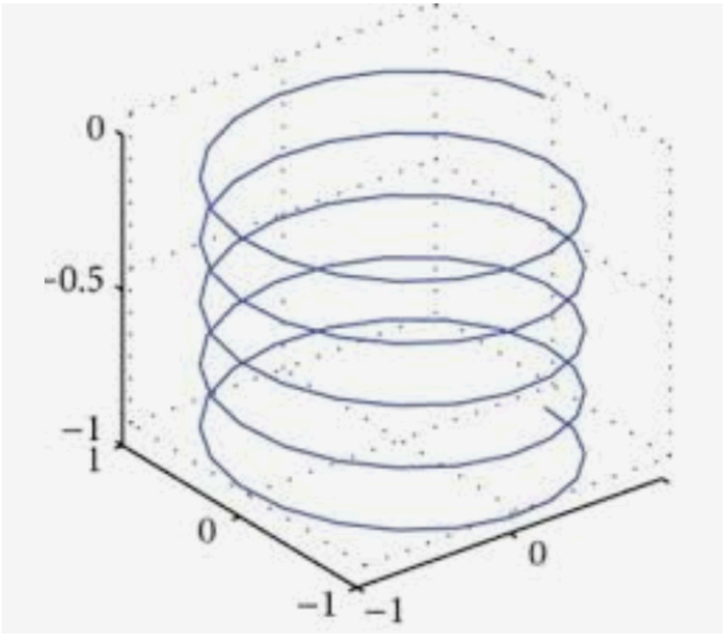
- What kind of differences you have seen?
- How to describe the dissimilarity?
- Need non-Euclidean metrics.

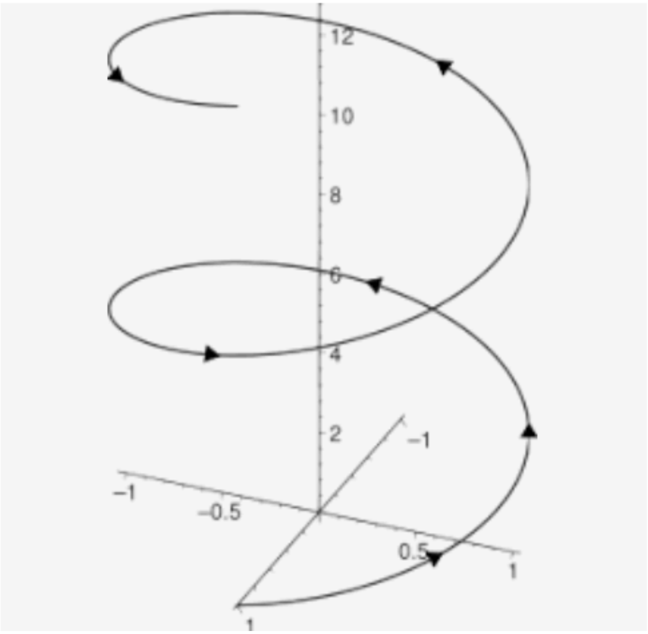


$R = (v1, v2, v3)$
with $R^T R = I$
$B = (u1, u2, u3)$
with $B^T B = I$

$XR = B$, want $X^T X = I$
$X = B R^{-1}$
$X = B R^T$

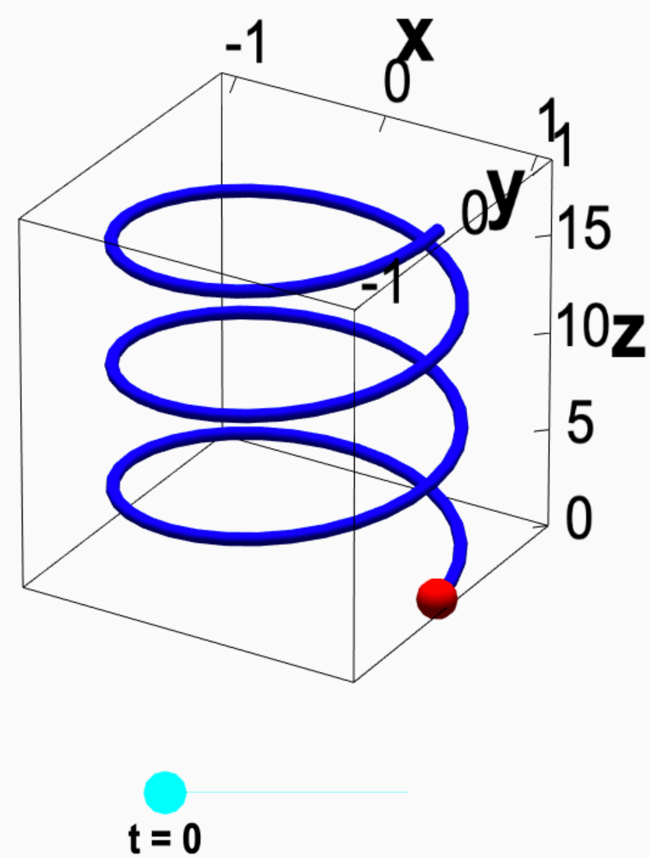
Recall: **Helix Curve**





- How to describe a helix curve?
- They could have different orientations!

You could think of a curve $\mathbf{c} : \mathbf{R} \to \mathbf{R}^3$ as being a wire. For example, $\mathbf{c}(t) = (\cos t, \sin t, t)$, for $0 \leq t \leq 6\pi$, is the parametrization of a helix. You can view it as a slinky or a spring.



Parametrized helix. The vector-valued function $\mathbf{c}(t) = (\cos t, \sin t, t)$ parametrizes a helix, shown in blue. This helix is the image of the interval $[0, 6\pi]$ (shown in cyan) under the mapping of $\mathbf{c}$. For each value of t, the red point represents the vector $\mathbf{c}(t)$. As you change t by moving the cyan point along the interval $[0, 6\pi]$, the red point traces out the helix.

In general: Parametrized Curve

Parametrized and Regular Curves

Definition

A *parametrized differentiable curve* is a differentiable map $\alpha : I \to \mathbb{R}^3$ of an open interval $I = (a, b)$ of the real line $\mathbb{R}$ into $\mathbb{R}^3$.

$t \rightarrow (x(t), y(t), z(t))$

Definition

A parametrized differentiable curve $\alpha : I \to \mathbb{R}^3$ is said to be *regular* if $\alpha'(t) \neq 0$ for all $t \in I$.

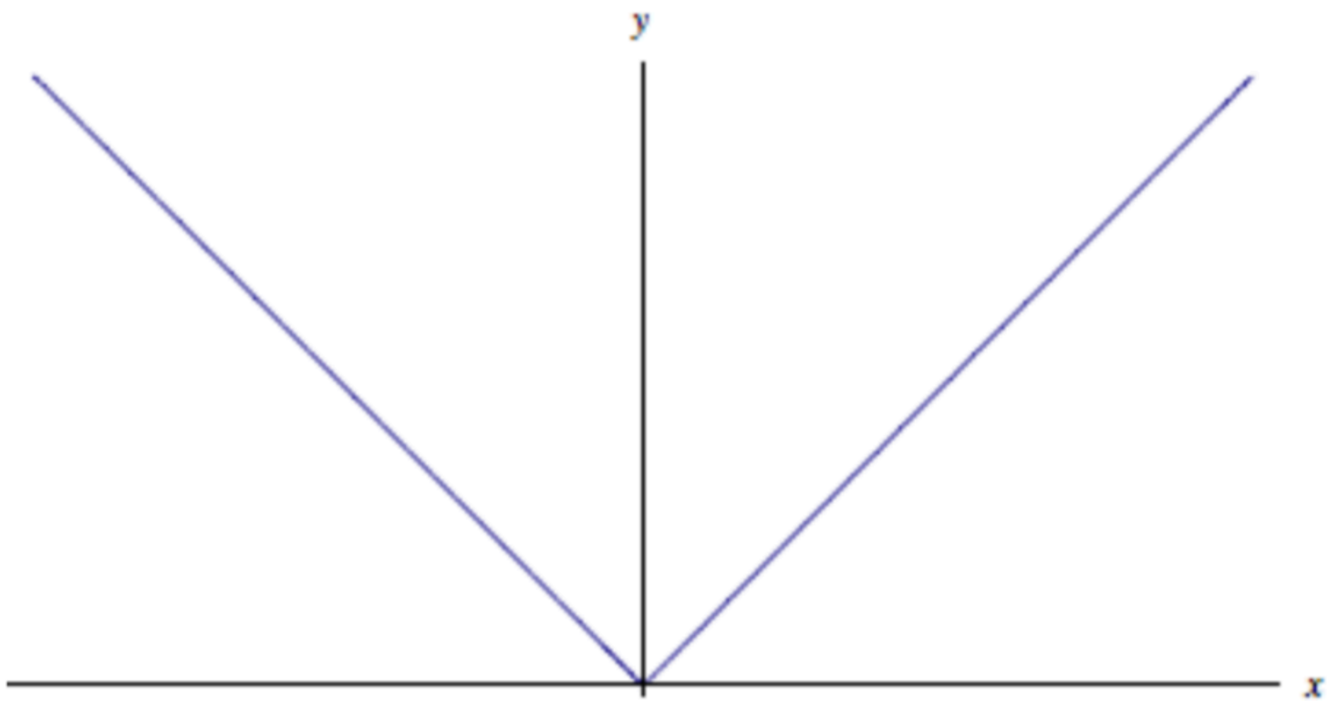
Definition

We say that $s \in I$ is a *singular point of order 1* if $\alpha''(s) = 0$ (in this context, the points where $\alpha'(s) = 0$ are called singular points of order 0).

Examples

Example 1

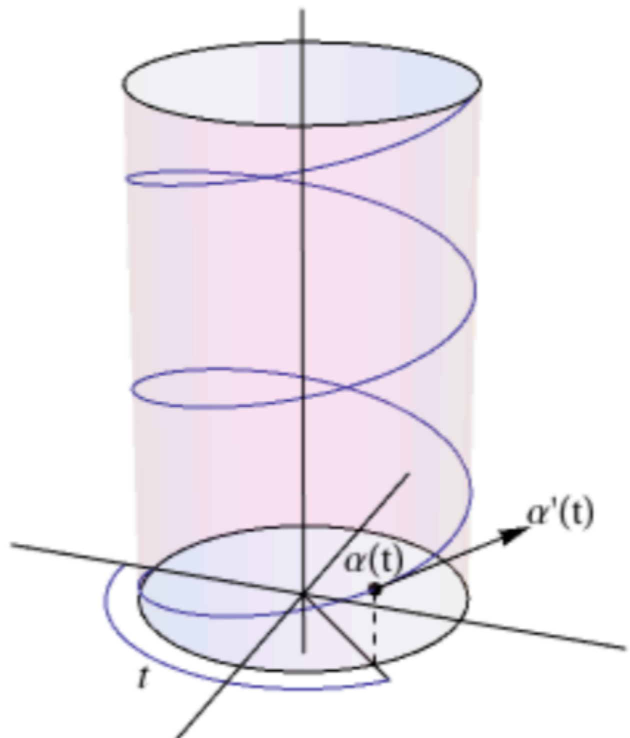
The map $\alpha : \mathbb{R} \to \mathbb{R}^2$ given by $\alpha(t) = (t, |t|), t \in \mathbb{R}$ (not differentiable).



Examples

Example 2

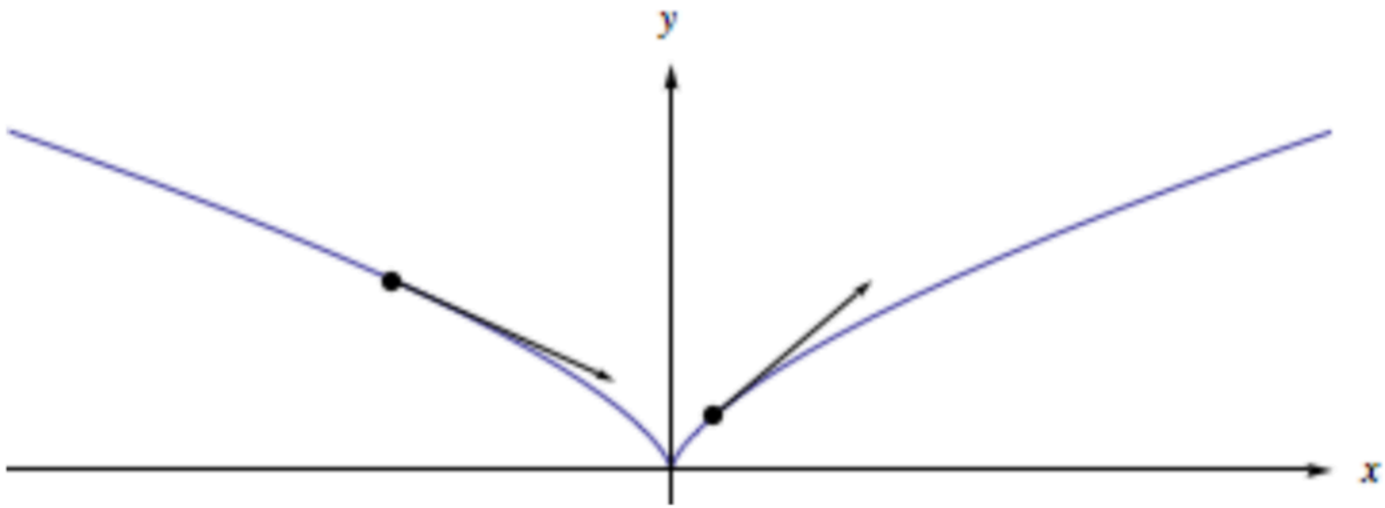
A helix of pitch $2\pi b$ on the cylinder $x^2 + y^2 = a^2$.



Examples

Example 3

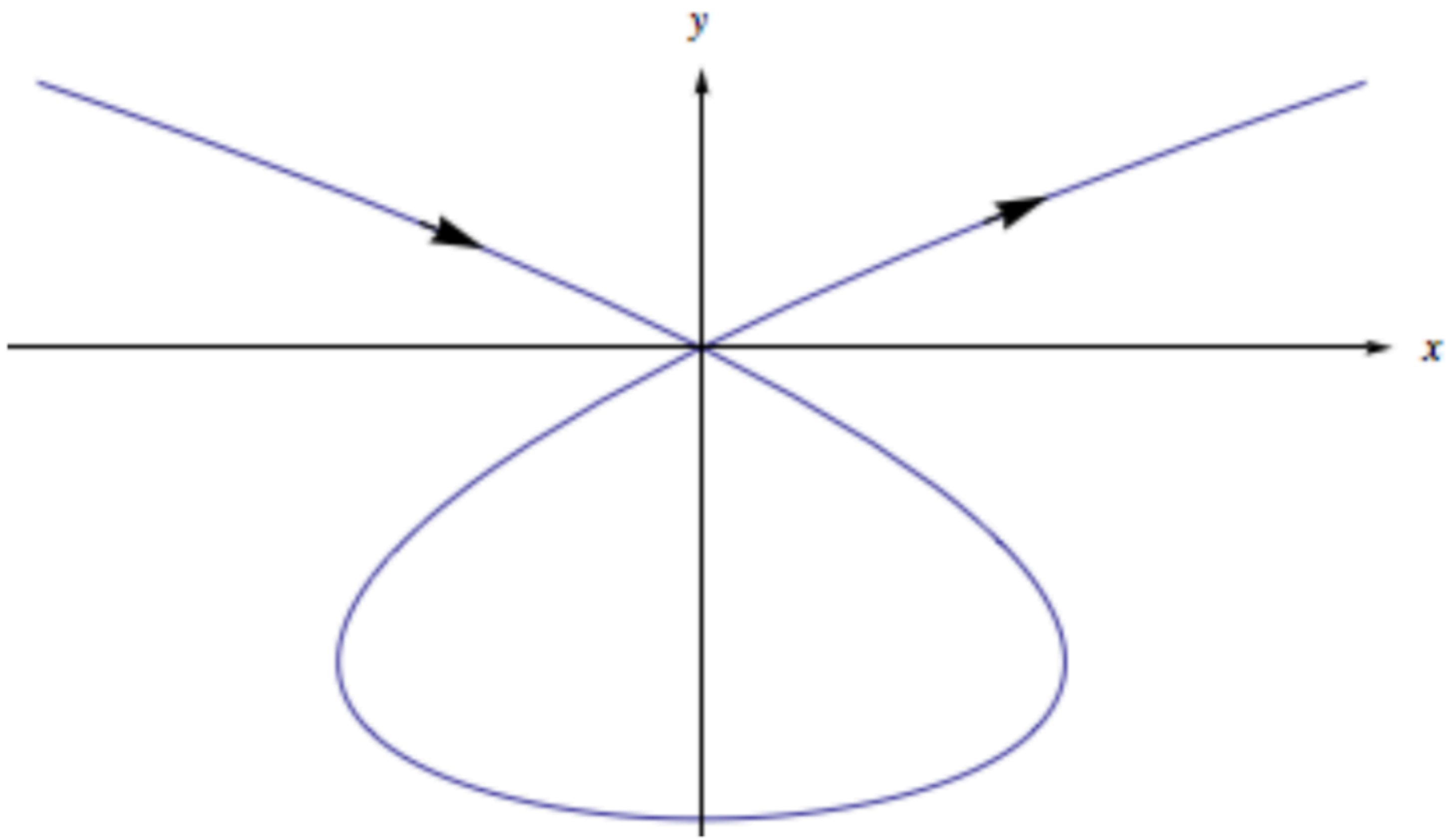
The map $\alpha : \mathbb{R} \to \mathbb{R}^2$ given by $\alpha(t) = (t^3, t^2), t \in \mathbb{R}$.



Examples

Example 4

The map $\alpha : \mathbb{R} \to \mathbb{R}^2$ given by $\alpha(t) = (t^3 - 4t, t^2 - 4), t \in \mathbb{R}$.



Arc Length of a Curve

Definition

Given $t \in I$, the *arc length* of a regular parametrized curve $\alpha : I \to \mathbb{R}^3$, from the point t_0, is by definition

$$s(t) = \int_{t_0}^{t} \|\alpha'(t)\| \, dt,$$

where

$$\|\alpha'(t)\| = \sqrt{(x'(t))^2 + (y'(t))^2 + (z'(t))^2}$$

is the length of the vector $\alpha'(t)$.

Definition

A parametrized curve $\alpha : I \to \mathbb{R}^3$ is said to be parametrized by arc length if $\|\alpha'(t)\| = 1$ (that is, if α has unit speed) for all $t \in I$.

Parametrization by Arc Length

Proposition (Geometric meaning of above definition)

A curve $\alpha : I \to \mathbb{R}^3$ is parametrized by arc length if and only if the parameter t is the arc length of α measured from some point.

Proof.

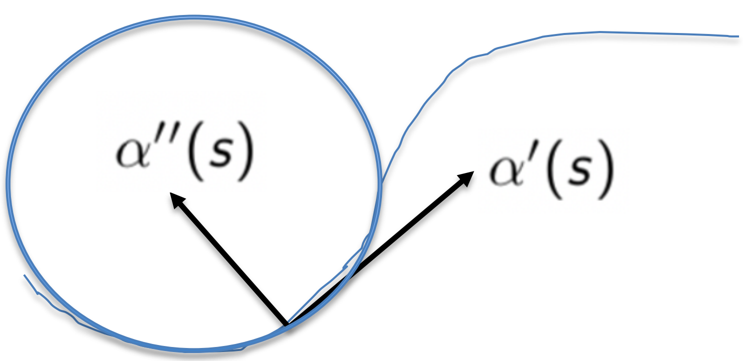
$\square$

Proposition (Advantages of $\|\alpha'(s)\| = 1$)

Let $\alpha : I \to \mathbb{R}^3$ be a curve parametrized by arc length. Then $\alpha''(s)$ is orthogonal to $\alpha'(s)$ for all $s \in I$.

Proof.

$\square$



$$\|\alpha'(s)\| = 1$$

$$\|\alpha'(s)\|^2 = 1$$

$\alpha'(s)$ dot $\alpha'(s)$ = 1

Take both sides

Reparametrization by Arc Length

Example

Consider the helix $\alpha : \mathbb{R} \to \mathbb{R}^3$ given by $\alpha(t) = (\cos t, \sin t, t)$.

$\alpha'(t)$ = (-sint, cost, 1)

Calculate the length of the curve, you get sqrt(2).
Now reparametrize: s = t/sqrt(2),
Now the curve in s will be having unit speed, therefore it is parametrized by arclength.

- From now on, we are going to assume curves are parametrized by arc length.

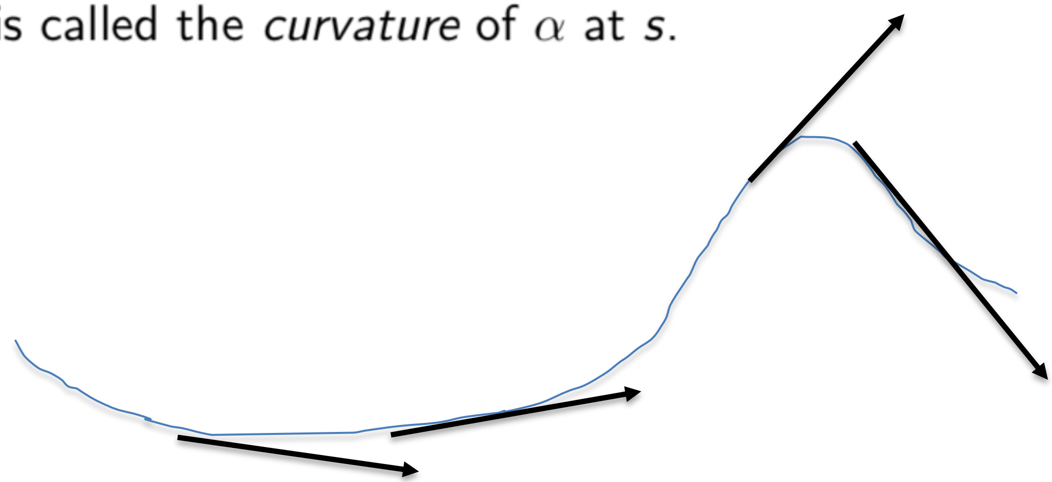
Curvature

Geometric Meaning

Let $\alpha : I = (a, b) \to \mathbb{R}^3$ be a curve parametrized by arc length s. Since the tangent vector $\alpha'(s)$ has unit length, the norm $\|\alpha''(s)\|$ of the second derivative measures the rate of change of the angle which neighboring tangents make with the tangent at s. $\|\alpha''(s)\|$ gives, therefore, a measure of how rapidly the curve pulls away from the tangent line at s, in a neighborhood of s.

Definition

Let $\alpha : I \to \mathbb{R}^3$ be a curve parametrized by arc length $s \in I$. The number $\|\alpha''(s)\| = k(s)$ is called the *curvature* of α at s.

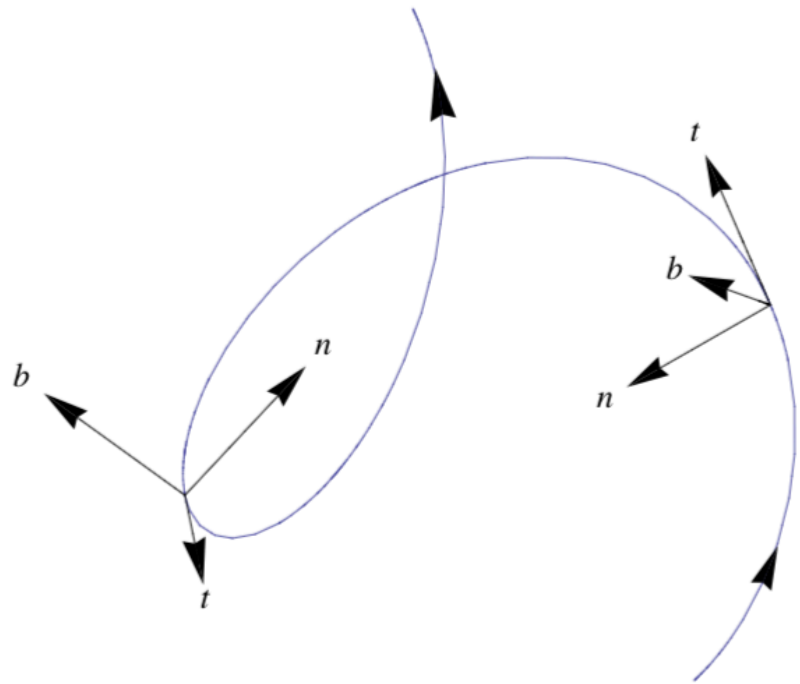


$(\alpha'(s))' = \alpha''(s)$

Torsion

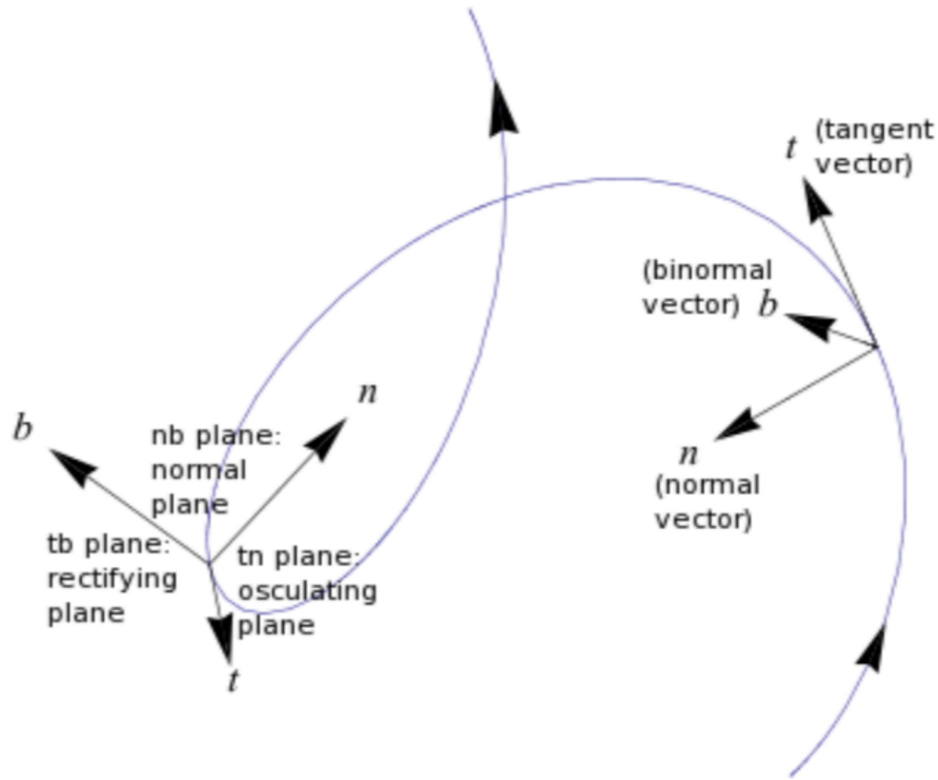
Geometric Meaning

Since $b(s)$ is a unit vector, the length $\|b'(s)\|$ measures the rate of change of the neighboring osculating planes with the osculating plane at s; that is $b'(s)$ measures how rapidly the curve pulls away from the osculating plane at s, in a neighborhood of s.



Frenet Frame

$$\alpha'(s) \stackrel{\triangle}{=} t(s)$$
$$\alpha''(s) = k(s)n(s)$$
$$t(s) \wedge n(s) = b(s)$$



Frenet Formulas

$$\begin{cases} t' = kn, \\ n' = -kt - \tau b, \\ b' = \tau n \end{cases}$$

Fundamental Theorem of the Local Theory of Curves

Theorem

Given differentiable functions $k(s) > 0$ and $\tau(s), s \in I$, there exists a regular parametrized curve $\alpha : I \to \mathbb{R}^3$ such that s is the arc length, $k(s)$ is the curvature, and $\tau(s)$ is the torsion of α Moreover, any other curve $\overline{\alpha}$ satisfying the same conditions differs from α by a rigid motion; that is, there exists an orthogonal map ρ of $\mathbb{R}^3$, with positive determinant, and a vector c such that $\overline{\alpha} = \rho \circ \alpha + c$.

Proof of uniqueness.

Claim: arc length, curvature, and torsion are invariant under the rigid motion. $\square$

Note: Homework will be given in the lecture.

Homework problems

- Problem A

Let $\alpha(t)$ be a parametrized curve which does not pass through the origin. If $\alpha(t_0)$ is the point of the trace of α closest to the origin and $\alpha'(t_0) \neq 0$, show that the position vector $\alpha(t_0)$ is orthogonal to $\alpha'(t_0)$.

Homework problems

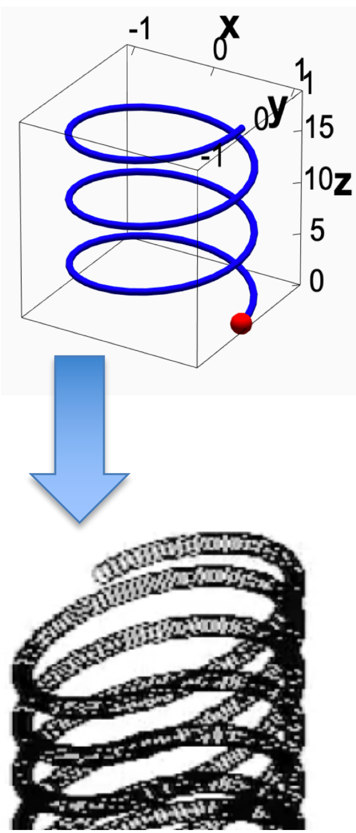
- Problem B

Show that the set of rigid motions forms a group.

Creative activity - Extra Credit

- How to create a transformation from the data on some helix to the data of the instructor's trajectory?
- Review different operators in R^2, e.g. we have shear map below. Here we want to shear a curve! For more info:

https://en.wikipedia.org/wiki/Transformation_matrix



For shear mapping (visually similar to slanting), there are two possibilities.

A shear parallel to the *x* axis has $x' = x + ky$ and $y' = y$. Written in matrix form, this becomes:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

A shear parallel to the *y* axis has $x' = x$ and $y' = y + kx$, which has matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ k & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Overview of Lecture 1

- Why we need nonlinear data analysis?
  - First starting with curves and their analysis
- Similarity measurements for nonlinear data
  - First a few examples: Arc-length, Geodesic length
- **Introduction to cell phone data**
- Introduction to rigid motion

Introduction of Cell Phone Data

- There are a lot of cell phone data sets available online
- For examples:
- 1. HMOG data set: http://www.cs.wm.edu/~qyang/hmog.html

Rotation Data

Rotation data is returned as a Euler angle, representing the number of degrees of difference between the device coordinate frame and the Earth coordinate frame.

Alpha

The rotation around the z axis. The `alpha` value is 0° when the top of the device is pointed directly north. As the device is rotated counter-clockwise, the `alpha` value increases.

Illustration of alpha in the device coordinate frame

Now rotation around x-axis

Beta

The rotation around the x axis. The `beta` value is 0° when the top and bottom of the device are equidistant from the surface of the earth. The value increases as the top of the device is tipped toward the surface of the earth.

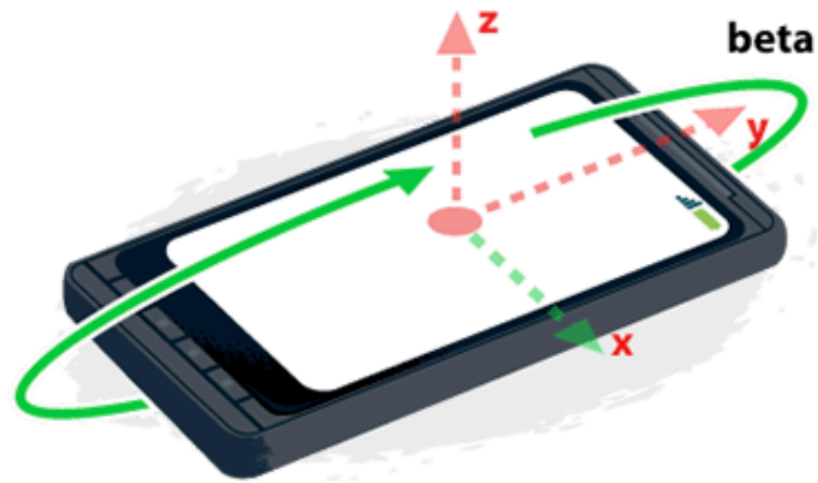


Illustration of beta in the device coordinate frame

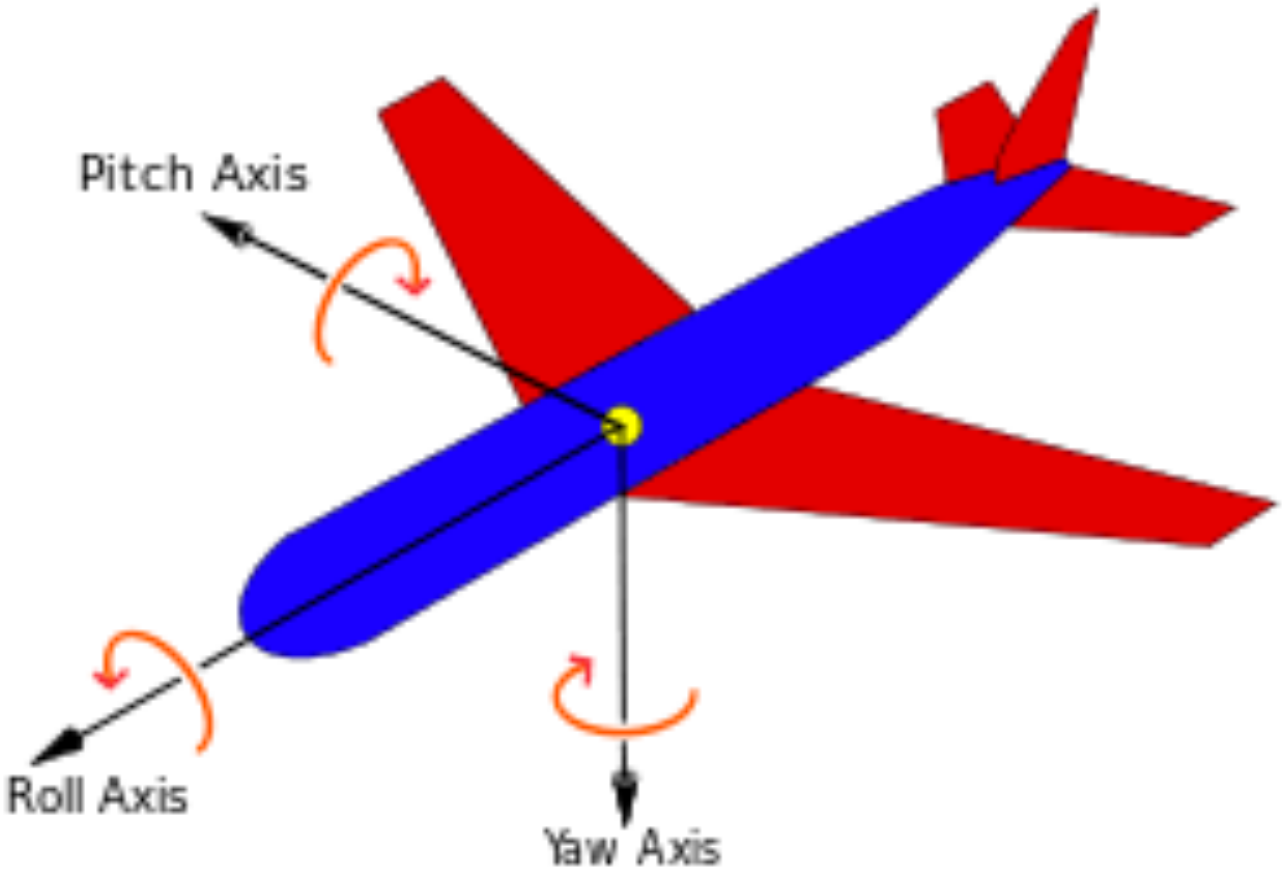
Now rotate around y -axis

Gamma

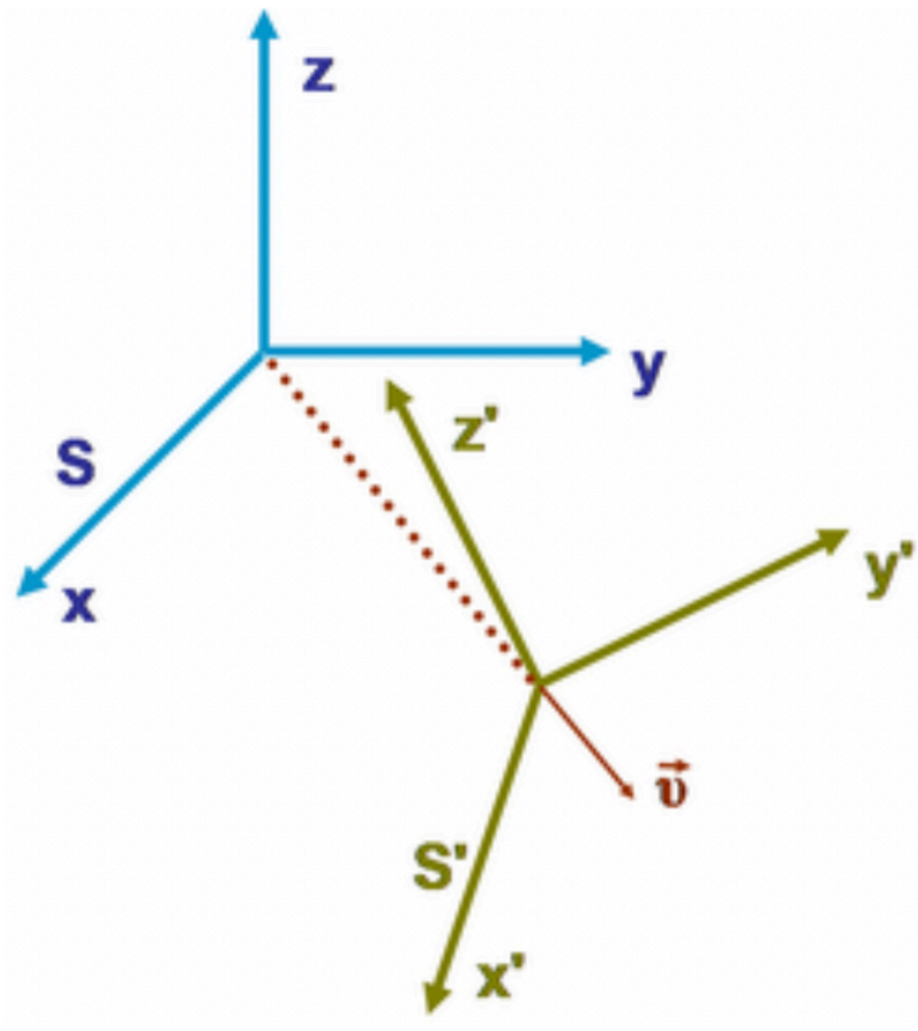
The rotation around the y axis. The `gamma` value is 0° when the left and right edges of the device are equidistant from the surface of the earth. The value increases as the right side is tipped towards the surface of the earth.

Illustration of gamma in the device coordinate frame

Pitch, Roll, and Yaw



Concept of Moving Frame



Real world Application

- **Using cell phone data to authenticate users.**
- Very hard problem and lots of math involved

H-MOG Data Set: A Multimodal Data Set for Evaluating Continuous Authentication Performance in Smartphones

Qing Yang, Ge Peng, David T. Nguyen, Xin Qi, Gang Zhou (*Colleget of William and Mary*)
Zdeňka Sitová (*New York Institute of Technology*; *Masaryk University*)
Paolo Gasti, Kiran S. Balagani (*New York Institute of Technology*)

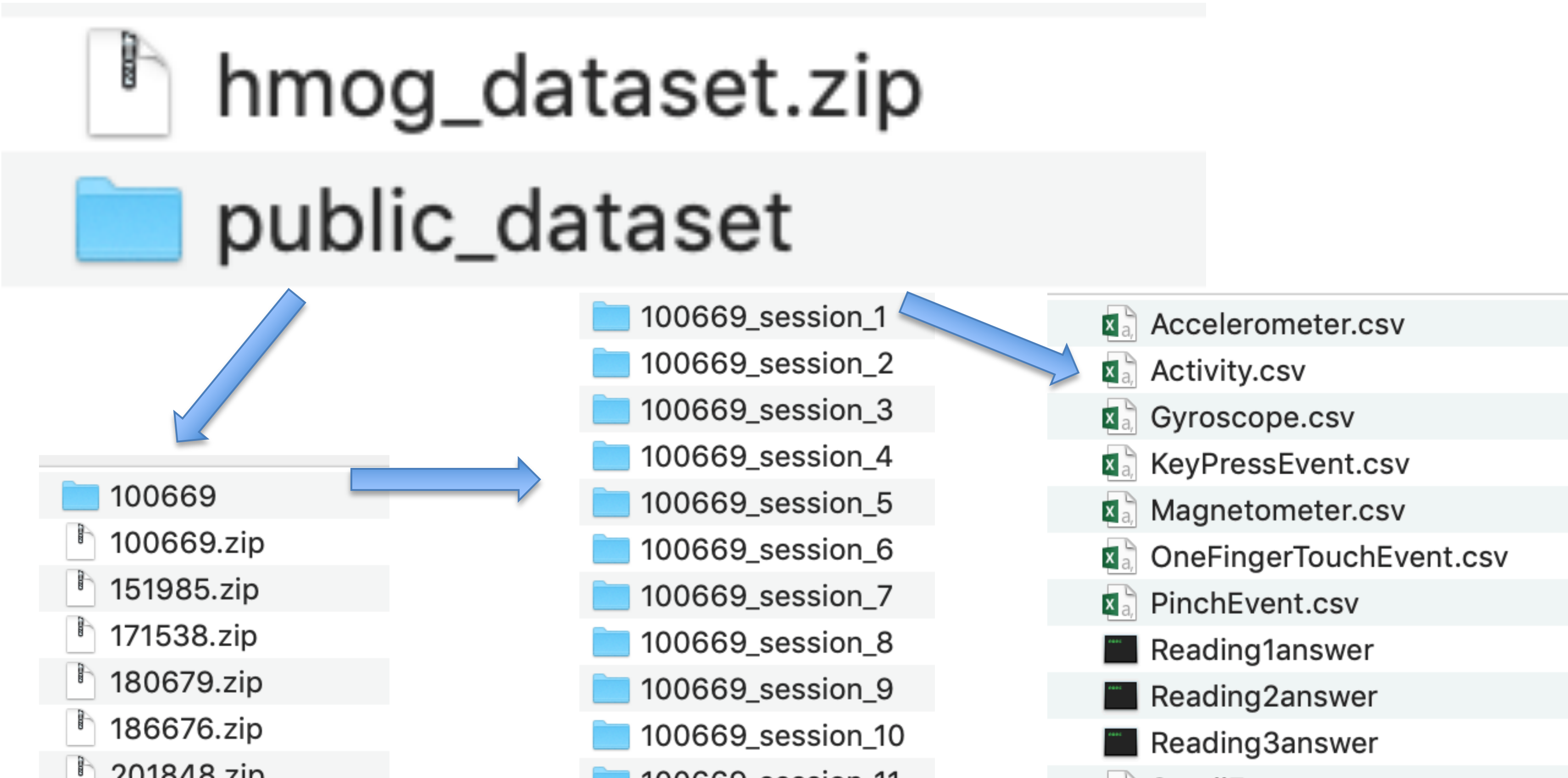
1. Introduction

We performed a large-scale user study to collect a wide spectrum of signals about user behaviors on smartphones, including touch, gesture, and pausality of the user, as well as movement and orientation of the phone. This dataset has been used to evaluate a continuous authentication modality named H-MOG in smartphones. A detailed description of this dataset and its application is in our poster paper (PDF) in ACM SenSys'14. The H-MOG paper using this dataset is published on IEEE Transactions on Information Forensics and Security (link on IEEE Xplore).

Abstract:
We introduce hand movement, orientation, and grasp (HMOG), a set of behavioral features to continuously authenticate smartphone users. HMOG features unobtrusively capture subtle micro-movement and orientation dynamics resulting from how a user grasps, holds, and taps on the smartphone. We evaluated authentication and biometric key generation (BKG) performance of HMOG features on data collected from 100 subjects typing on a virtual keyboard. Data were collected under two conditions: 1) sitting and 2) walking. We achieved authentication equal error rates (EERs) as low as 7.16% (walking) and 10.05% (sitting) when we combined HMOG, tap, and keystroke features. We performed experiments to investigate why HMOG features perform well during walking. Our results suggest that this is due to the ability of HMOG features to capture distinctive body movements caused by walking, in addition to the hand-movement dynamics from taps. With BKG, we achieved the EERs of 15.1% using HMOG combined with taps. In comparison, BKG using tap, key hold, and swipe features had EERs between 25.7% and 34.2%. We also analyzed the energy consumption of HMOG feature extraction and computation. Our analysis shows that HMOG features extracted at a 16-Hz sensor sampling rate incurred a minor overhead of 7.9% without sacrificing authentication accuracy. Two points distinguish our work from current literature: 1) we present the results of a comprehensive evaluation of three types of features (HMOG, keystroke, and tap) and their combinations under the same experimental conditions and 2) we analyze the features from three perspectives (authentication, BKG, and energy consumption on smartphones).

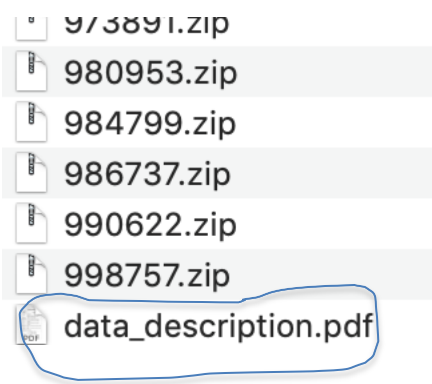
Download and understand the data set

- Please Download the data from the webpage
- You will get a zip file



What are those data sets? For example, what is gyroscope data?

- There is a read me at the end of the data set with all zip files of all user IDs.



Data Description

1. Activity.csv

Name	Description
ID	Composed as: SubjectID + Session_number + ContentID + Run-time determined Counter value
SubjectID	6 digits: ID of current subject
Session_number	1-24: session number for current subject
Start_time	Start time of current activity, in absolute timestamps
End_time	End time of current activity, in absolute timestamps
Relative_Start_time	Start time of current activity, relative to system boot
Relative_End_time	End time of current activity, relative to system boot

Gesture_scenario	1: Sit 2:Walk
TaskID	1, 7, 13, 19: Reading + Sitting 2, 8, 14, 20: Reading + Walking 3, 9, 15, 21: Writing + Sitting 4, 10, 16, 22: Writing + Walking 5, 11, 17, 23: Map + Sitting
ContentID	1: first sub-task 2: second sub-task 3: third sub-task

2. Accelerometer.csv

Name	Description
Systime	Absolute time-stamp
EventTime	Sensor event relative time-stamp
ActivityID	Belonged activity
X	Acceleration minus Gx on the x-axis
Y	Acceleration minus Gy on the y-axis
Z	Acceleration minus Gz on the z-axis
Phone_orientation	0: Portrait and no rotate 1: device rotated 90 degrees counter-clockwise 3: device rotated 90 degrees clockwise

3. Gyroscrope.csv

Name	Description
Systime	Absolute time-stamp
EventTime	Sensor event relative time-stamp
ActivityID	Belonged activity
X	Angular speed around the x-axis
Y	Angular speed around the y-axis
Z	Angular speed around the z-axis
Phone_orientation	0: Portrait and no rotate 1: device rotated 90 degrees counter-clockwise 3: device rotated 90 degrees clockwise

4. Magnetometer.csv

Name	Description
Systime	Absolute time-stamp
EventTime	Sensor event relative time-stamp
ActivityID	Belonged activity
X	Ambient magnetic field in the X axis in micro-Tesla (uT)
Y	Ambient magnetic field in the Y axis in micro-Tesla (uT)
Z	Ambient magnetic field in the Z axis in micro-Tesla (uT)
Phone_orientation	0: Portrait and no rotate 1: device rotated 90 degrees counter-clockwise 3: device rotated 90 degrees clockwise

5. TouchEvent.csv

Name	Description
Systime	Absolute time-stamp
EventTime	Sensor event relative time-stamp
ActivityID	Belonged activity
Pointer_count	1: Single touch 2: Multi-touch
PointerID	0: Single touch; or first pointer in multi-touch 1: Second pointer in multi-touch
ActionID	0 or 5: DOWN 1 or 6: UP 2: MOVE
X	Touch location in X coordination
Y	Touch location in Y coordination
Pressure	Touch pressure
Contact_size	Touch contact size
Phone_orientation	0: Portrait and no rotate 1: device rotated 90 degrees counter-clockwise 3: device rotated 90 degrees clockwise

Homework

- Please read the following paper:

Journals & Magazines > IEEE Transactions on Informat... > Volume: 11 Issue: 5

HMOG: New Behavioral Biometric Features for Continuous Authentication of Smartphone Users

- https://ieeexplore.ieee.org/document/7349202?arnumber=7349202

Overview of Lecture 1

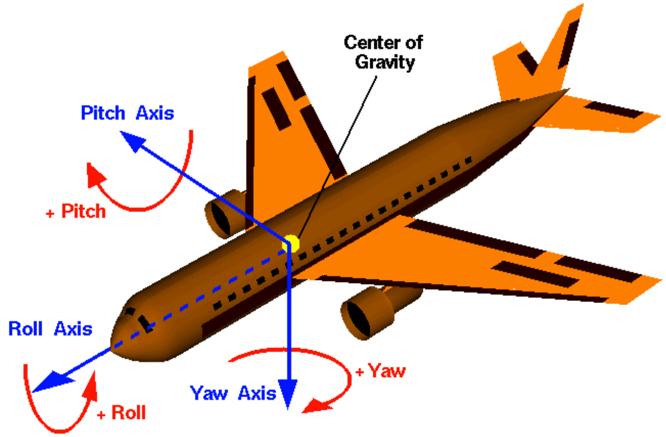
- Why we need nonlinear data analysis?
  - First starting with curves and their analysis
- Similarity measurements for nonlinear data
  - First a few examples: Arc-length, Geodesic length
- Introduction to cell phone data
- **Introduction to rigid motion**

Introduction to Rigid Motion

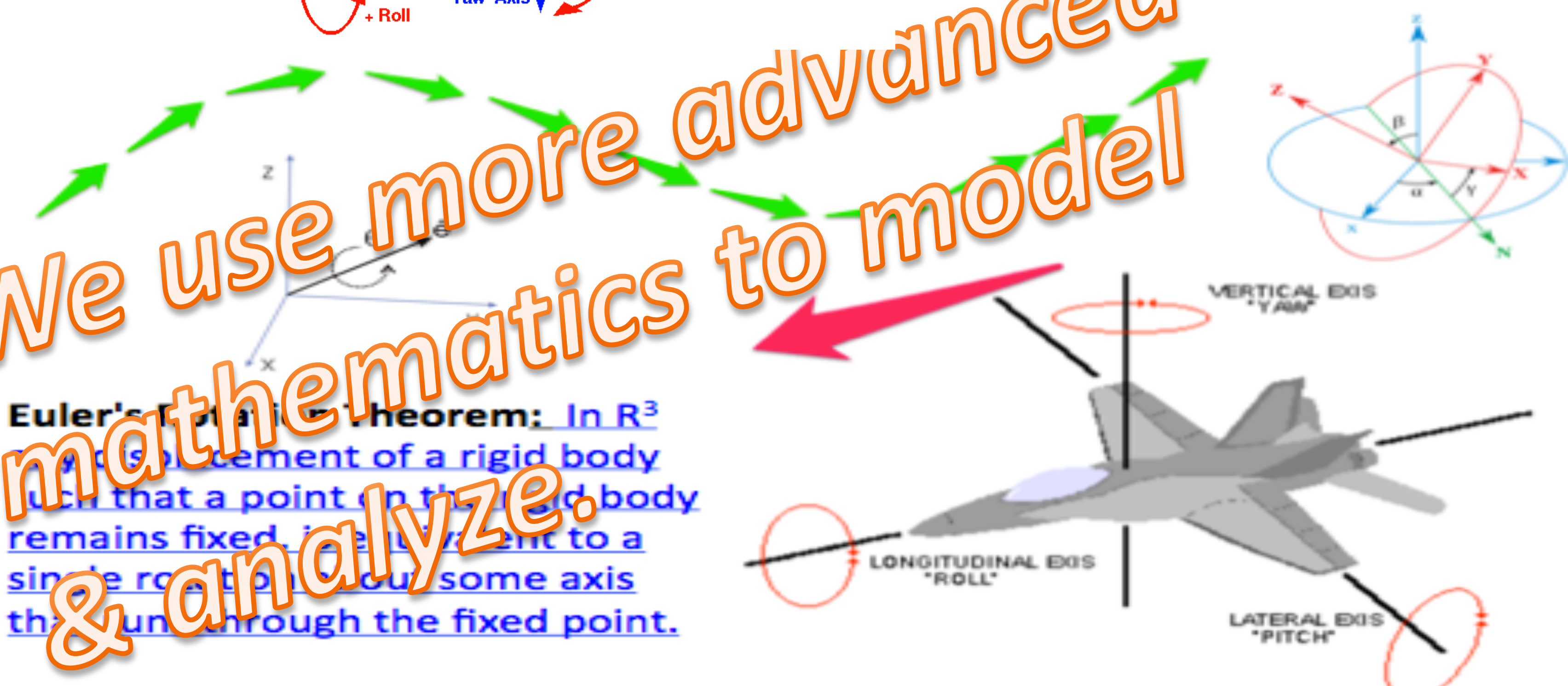
Details of Hard Math Behind UAV Data
(similar for cell phone data
or auto vehicle data)

- **Moving frames**
- **The set of orthonormal matrices**
- **The set of rotations in R^3**
- **Lie group SO(3)** **(details later**

Viewing an UAV as a point is not enough since it has more complicated dynamics such as pitch, roll, yaw and their angular velocities

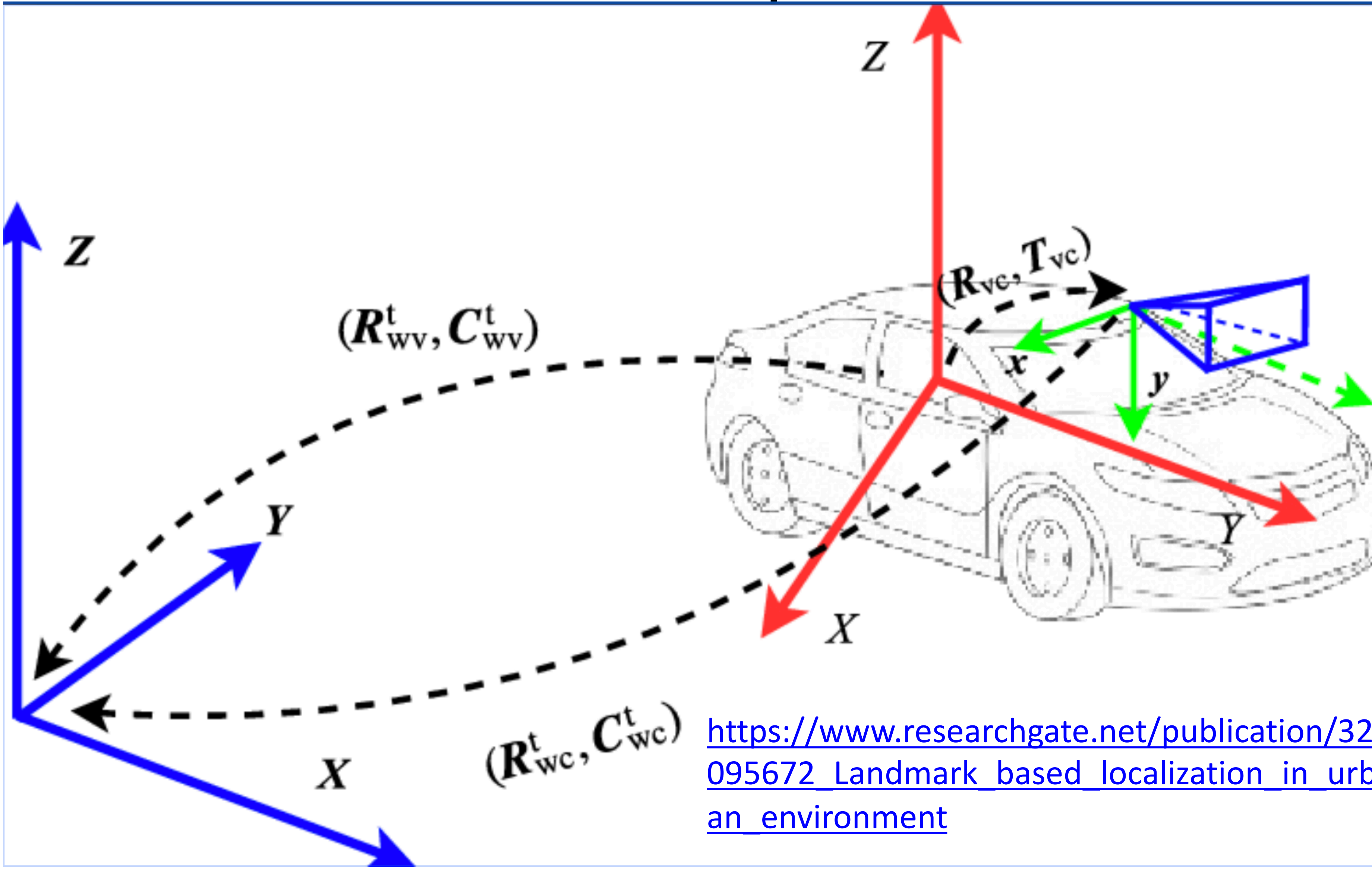


Details later!



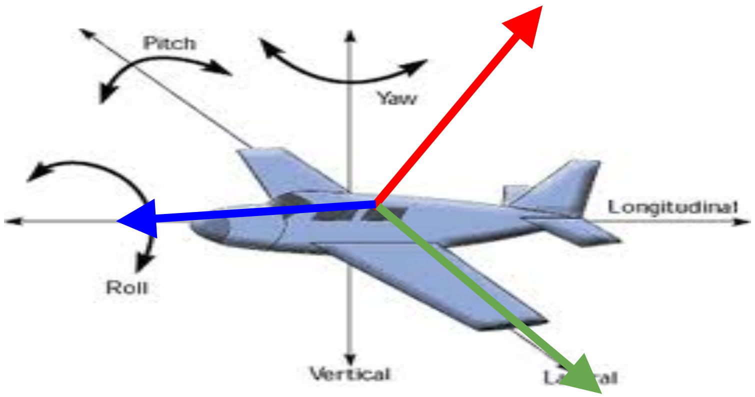
We use more advanced mathematics to model & analyze.

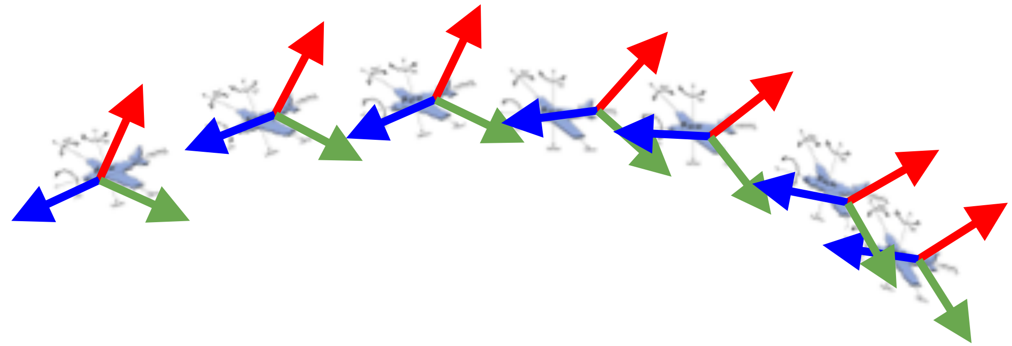
How to set a good coordinate system to model a problem?



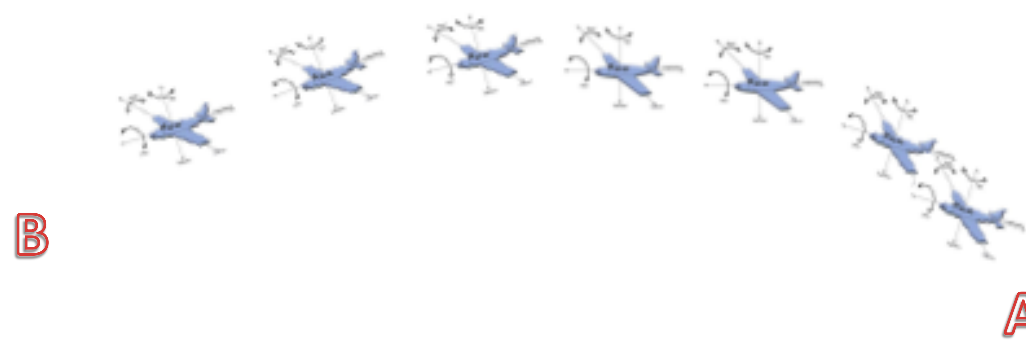
https://www.researchgate.net/publication/320095672_Landmark_based_localization_in_urban_environment

For Example: we want a computer to mathematically understand a pilot's manual flight control skill. Then we can compare between good controls and poor controls.





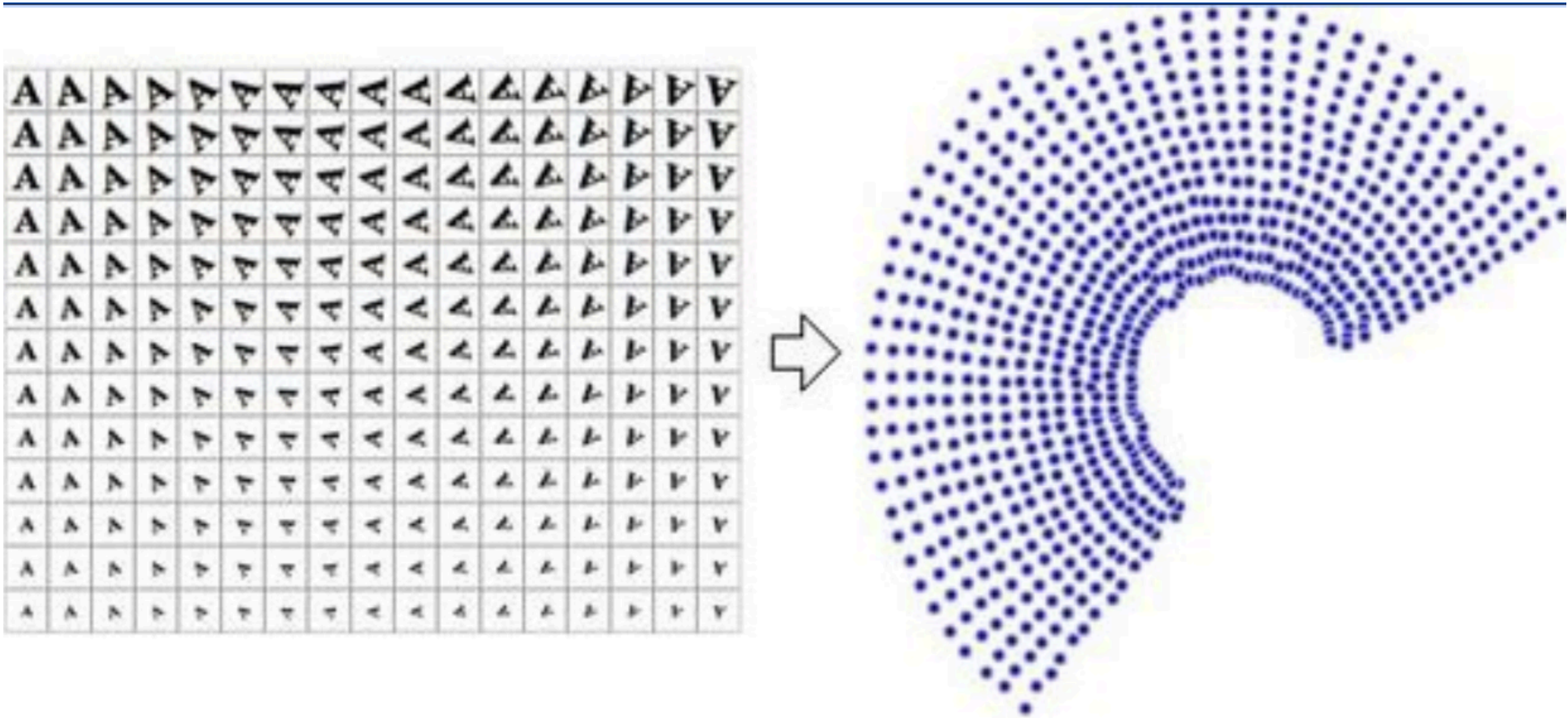
Key: This kind of mathematics captures dynamical behaviors of any UAVs



References

- https://datascience.stackexchange.com/questions/5694/dimensionality-and-manifold
- https://github.com/VivekPa/IntroNeuralNetworks
- https://developers.google.com/web/fundamentals/native-hardware/device-orientation/
- https://ieeexplore.ieee.org/document/7349202?arnumber=7349202

Back up slides

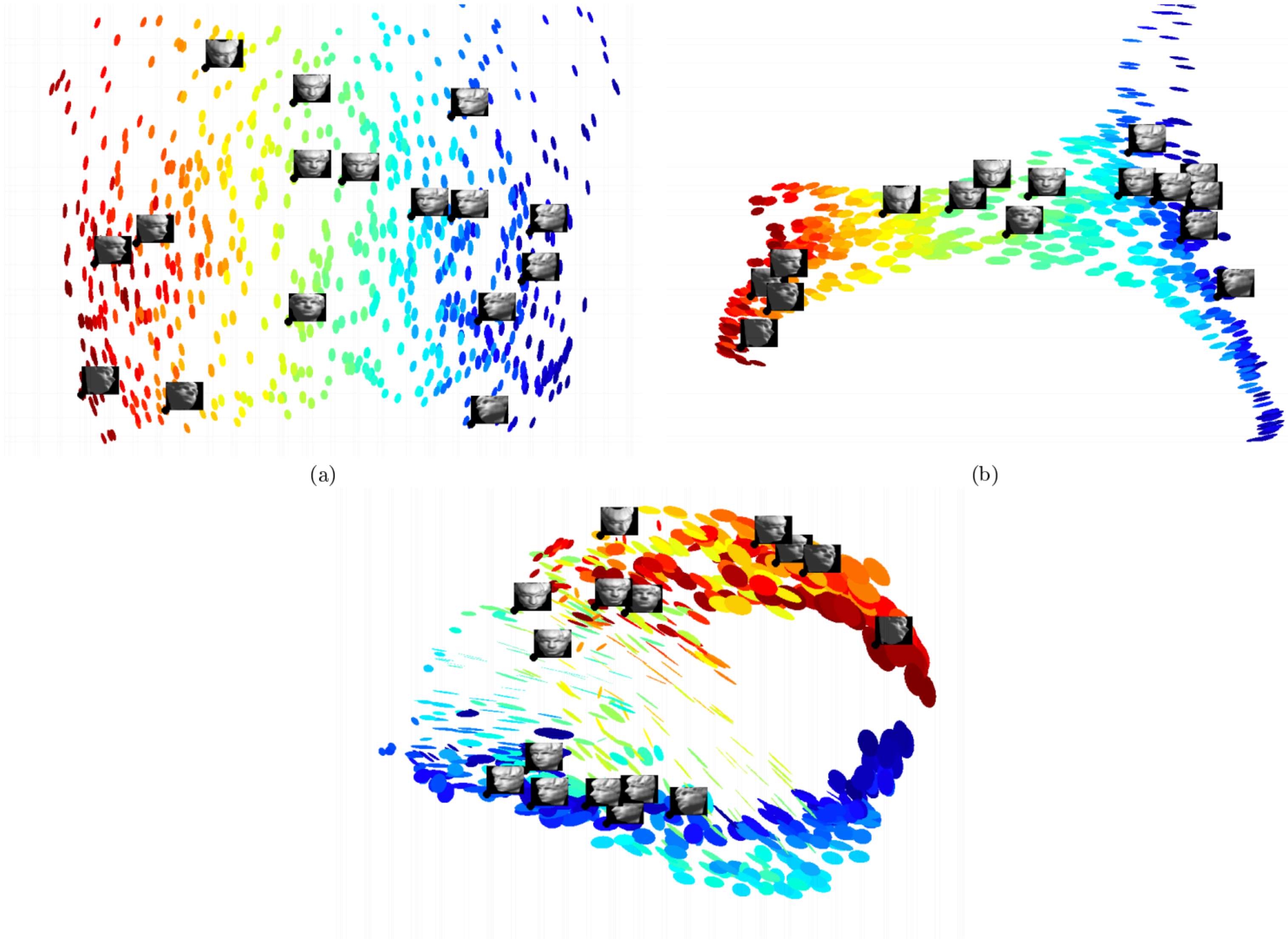


- Metric Learning and Manifolds: Preserving the Intrinsic Geometry
- https://www.stat.washington.edu/mmp/geometry/reading-group17/html/RMetric.pdf

Abstract

A variety of algorithms exist for performing non-linear dimension reduction, but these algorithms do not preserve the original geometry of the data except in special cases. In general, in the low-dimensional representations obtained, distances are distorted, as well as angles, areas, etc. This paper proposes a generic method to estimate the distortion incurred at each point of an embedding, and subsequently to "correct" distances and other intrinsic geometric quantities back to their original values (up to sampling noise).

Our approach is based on augmenting the output of an embedding algorithm with geometric information embodied in the Riemannian metric of the manifold. The Riemannian metric allows one to compute geometric quantities (such as angle, length, or volume) for any coordinate system or embedding of the manifold. In this work, we provide an algorithm for estimating the Riemannian metric from data, consider its consistency, and demonstrate the uses of our approach in a variety of examples.



(a) (b) (c)

Figure 5: Two-dimensional visualization of the faces manifold, along with embedding. The color corresponds to the left-right motion of the faces. The embeddings shown are: (a) Isomap, (b) LTSA, and Diffusion Maps ($\lambda = 1$) (c) . Note the very elongated ellipses at the top and bottom of the LTSA embedding, indicating the distortions that occurred there.

Find a paper to read which does the analysis using HMOG data

- Read
- Give a 1-2 page summary

HMOG: New Behavioral Biometric Features for Continuous Authentication of Smartphone Users

Zdeňka Sitová, Jaroslav Šeděnka, Qing Yang, Ge Peng, Gang Zhou, *Senior Member, IEEE*, Paolo Gasti, *Member, IEEE*, and Kiran S. Balagani, *Member, IEEE*

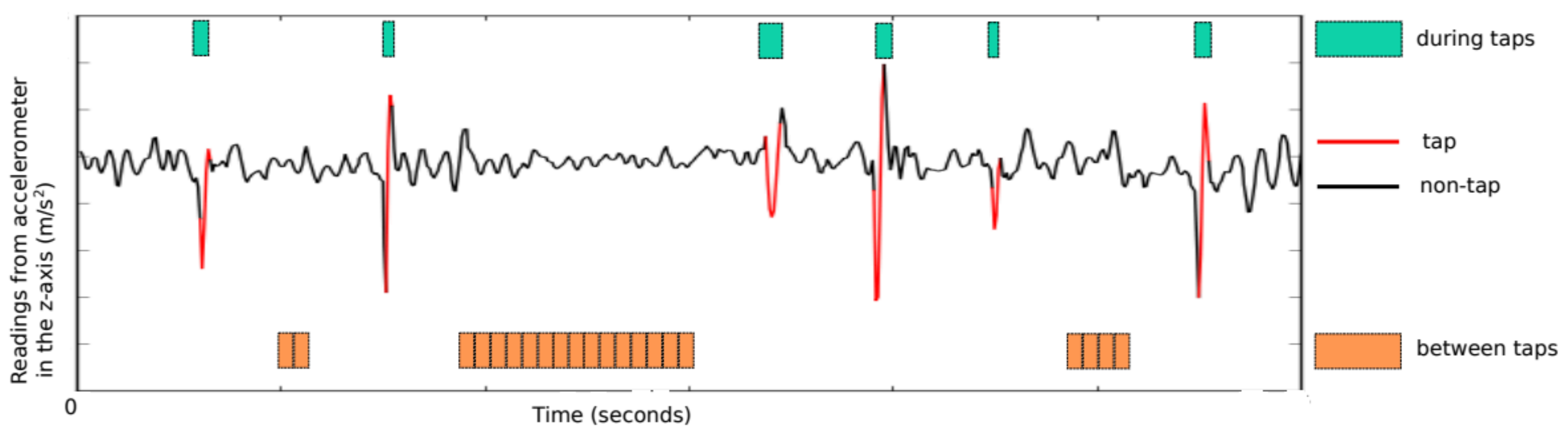


Fig. 8. HMOG features extracted *during* and *between* taps. The figure shows a sample of readings from the z-axis of accelerometer in sitting condition.

Techniques in Geometric Analysis:

Example

Assume that all normals of a parametrized curve pass through a fixed point. Prove that the trace of the curve is contained in a circle.

Homework: Rewrite all the proofs in the example.