Introduction to Differential Geometry and its Applications

Course Webpage

math143.github.io

Professor Weiqing Gu

Harvey Mudd College

TOPICS IN DIFFERENTIAL GEOMETRY

PROFESSOR WEIQING GU
SPRING 2018

Office: Shan 3420

Email: gu@g.hmc.edu

Office Phone: (909)621-8929

Fax: (909)621-8366

Office Hours: Tuesday 3:00pm - 4:00pm or by appointment

Course Meeting:

Monday, 2:45pm - 4:00pm

Wednesday, 2:45pm - 4:00pm

Course Location:

Harvey Mudd College Shan 2461

Graders/Tutoring Hours:

- Paul David, paul.david@cgu.edu
- Maggie Li, mli@g.hmc.edu

Tutoring Hours: TBD

Textbook:

All members of the class will be required to obtain the following texts:

- Information Geometry and its Applications by Shun-ichi Amari
- Riemannian Geometry by Do Carmo

Grading:

- 30% Homework
- 30% Midterm Project
- 40% Final Project

Course Requirements and Evaluation:

- ***Homework***

Homework will be assigned once a week and will be due every Monday before the lecture. Discussions and collaboration is encouraged. Solution hints also will be provided if necessary. But you must write up your own solution. Before tackling any problems, read the relevant section of the text and review your lecture notes. Consult with the faculty, the tutor (Paul David) or your classmates about matters that are unclear to you. NEVER LET YOURSELF FALL BEHIND.

Note:

I am trying my best to help you succeed in this course. Trust me, I am on your side. However, you must try your best. This is a very advanced and challenge course. It will greatly benefit you for your future graduate study or industry work. Though it is very hard at the beginning since there are many very high level new concepts, you will finally get used to them. By the time you are in your graduate school or your work, you will certainly be at the top of the "mountain" as a student or an employee with your talent ought to be. So let us work really hard now!

- ***Midterm/Final Projects***

The midterm and final projects are the largest components of the course. Each student will discover, explore, and attack a real world problem of your choosing. The detailed description and requirements for the midterm and final projects can be found under the "Mid/Final Project" tab.

- ***GitHub***

Students are expected to become comfortable with Github. Hence, each student is required to create a Github account for midterm and final project submission. If you already have a Github account, that's perfect. If not, please create a personal Github account and go over the tutorials online.

Classroom Policies:

- ***Attendence***

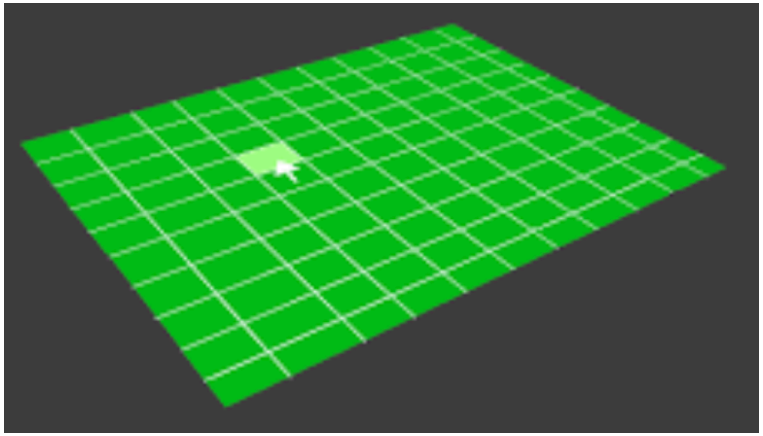
Attendence for each lecture is mandatory and is expected of all class members. if you're going to miss a lecture, it is neccessary for you to inform the instructor as soon as possible. You are also responsible for obtaining notes from another class member.

COURSE DESCRIPTION

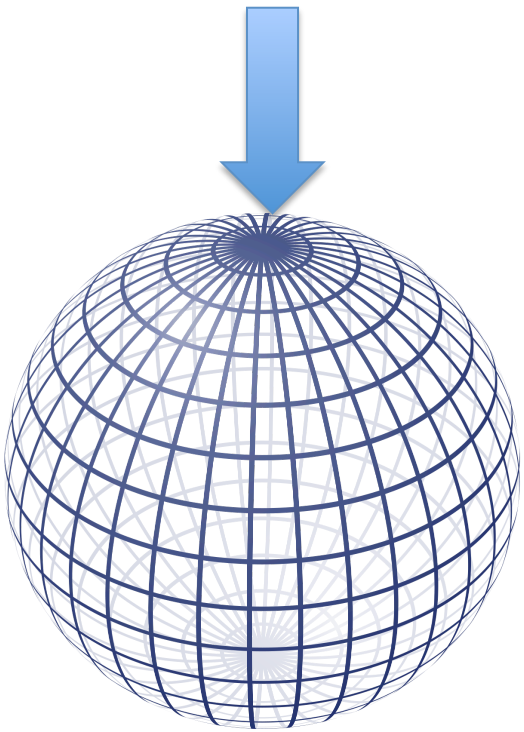
The objective of this course is to familiar with the students with the basic language and techniques of Riemannian Geometry and their new applications in big data analytics in addition to their classical applications in theoretical physics. The following are the possible topics in the courses. The order of the topics will depend on the students' interests in the class.

Topic 1: Curves and their Applications in Computer Vision
Topic 2: Surfaces and ISOMAP
Topic 3: Gauss Maps and their Applications
Topic 4: Convex Geometry and Convex Optimization in Machine Learning
Topic 5: Manifold and their Applications
Topic 6: Riemannian Metrics and Techniques to Select Appropriate Metrics
Topic 7: Tensors and their Applications
Topic 8: Connections and their Applications
Topic 9: Information Geometry and their Applications
Topic 10: Geodesics and their Applications
Topic 11: Curvatures and their applications

How to study Diff. Goe.?
Extend Flat to Curved



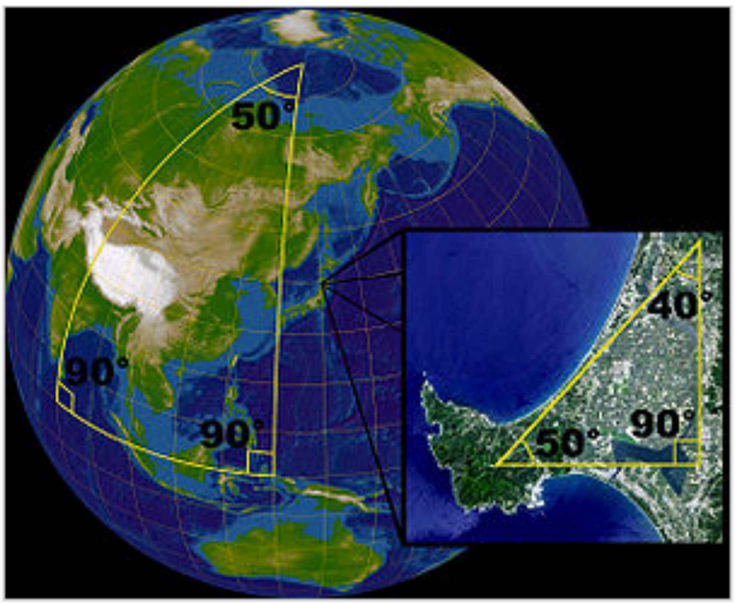
Vector Space:
Linear algebra



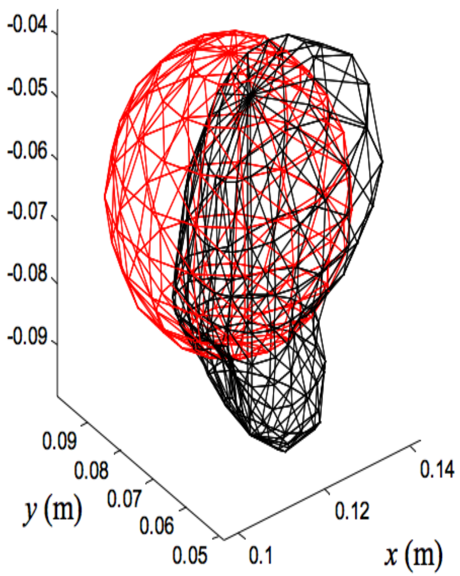
Manifold:
Differential Geometry

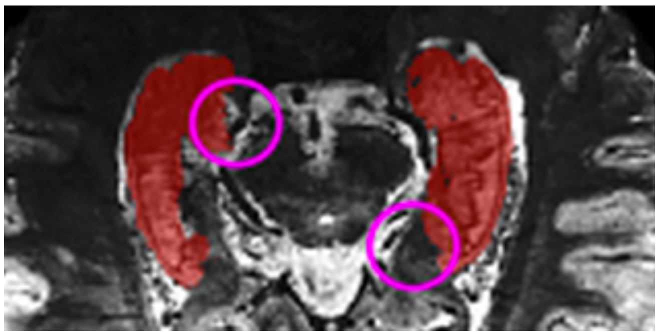
What is a manifold?

- An n-dimensional manifold locally "looks like" a piece of $\mathbf{R}^n$.
- Key features of a **manifold: curved**
- For examples, sphere and torus.



The sphere (surface of a ball) is a two-dimensional manifold since it can be represented by a collection of two-dimensional maps.





E.g. We can use manifolds to represent brain or breast cancer tumors.

Topological Mfld $\neq$ Differentiable mfld

Differentiable Manifolds

Physicist: A manifold is something which 'locally' look like a bit of n-dimensional Euclidean space $\mathbb{R}^n$.

Q: What do you mean "locally Look like"?

Mathematician:

$\mathbb{R}^n$

$\Downarrow$

V^n

$\Downarrow$

(Curves &) Surfaces → Parametrized, Regular ✓

$\Downarrow$

Manifolds

Using Key properties of $\mathbb{R}^n$ to define V^n.

Using the differentiability of $\mathbb{R}^2$, $\mathbb{R}^3$ and linear map to define a regular surface.

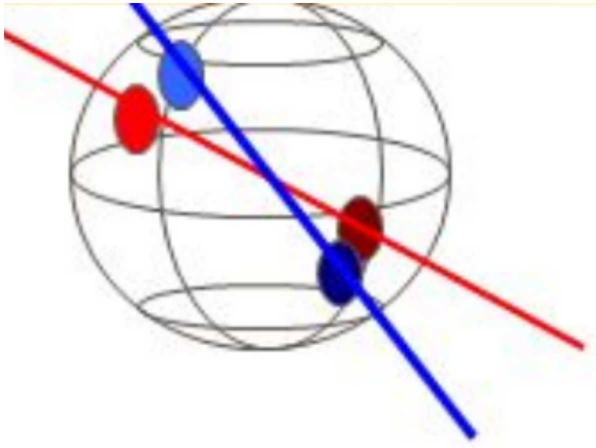
Using Key proposition of a regular surface to define manifolds

In this course mfld $=$ Differentiable mfld

Subtlety: The Sphere is not "the sphere you usually see."

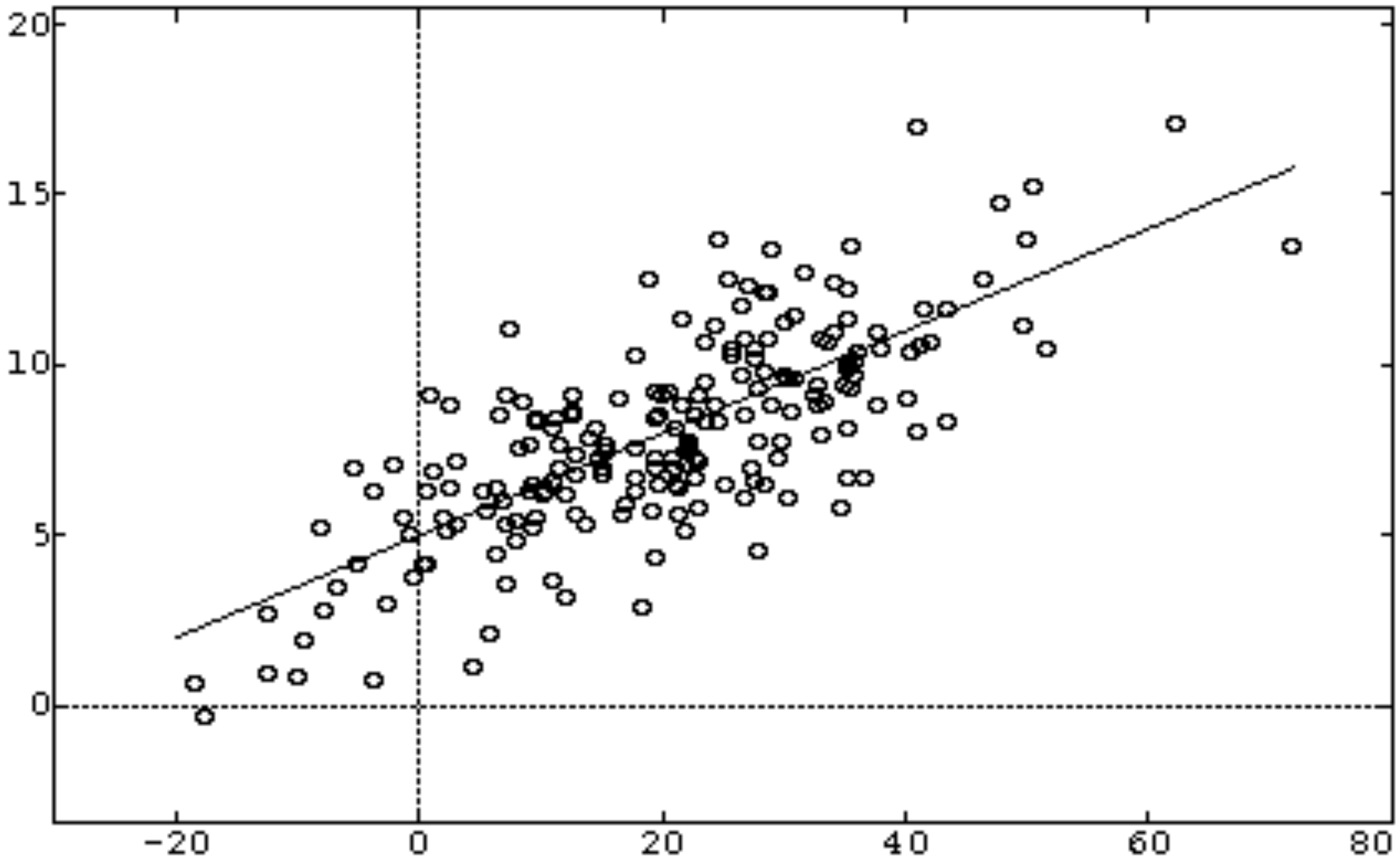
- **Example:** Consider the set of all possible oriented planes through origin.

G1R3 = G2R3

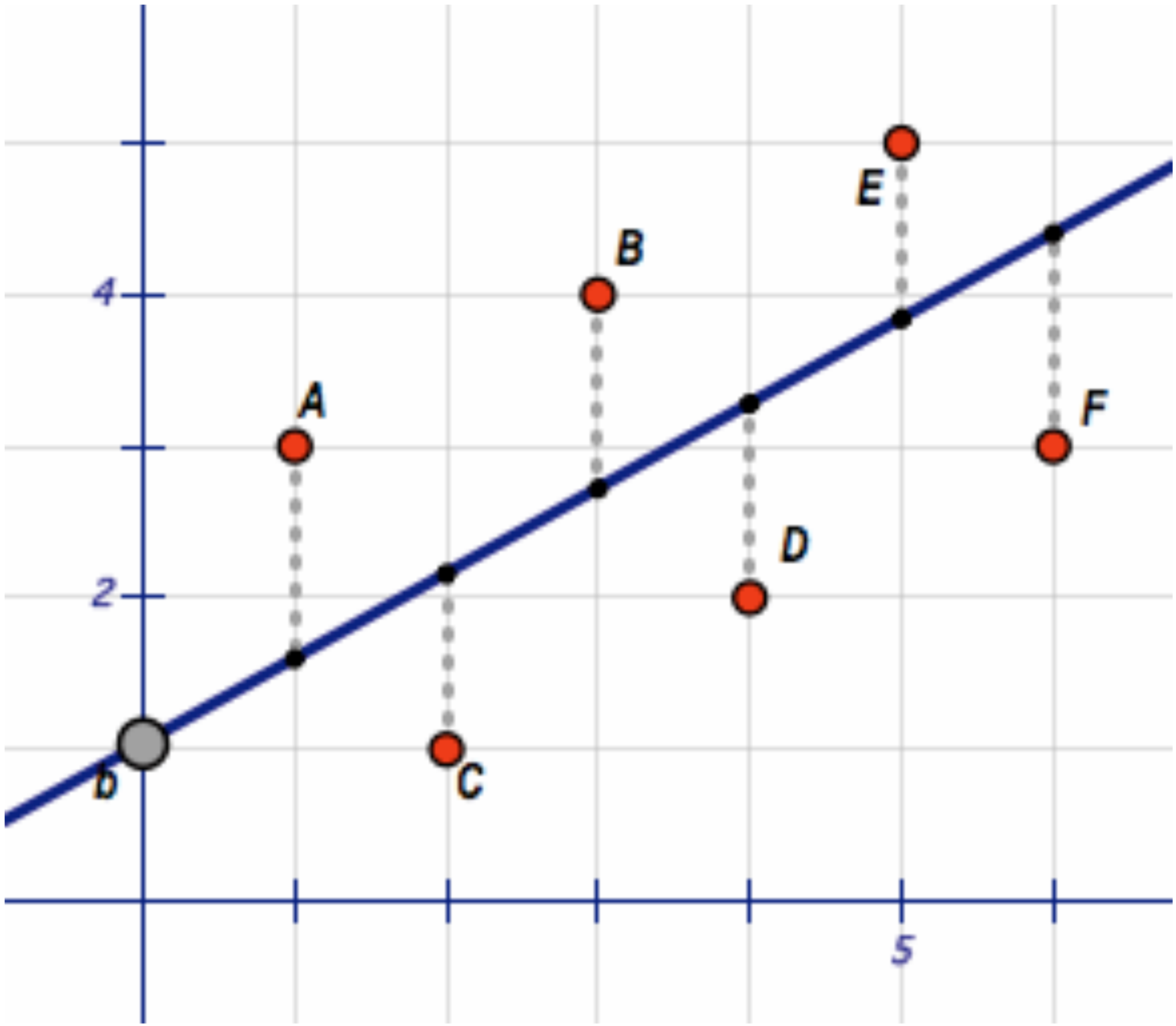


Recall: Linear Regression

Given some data: $D = \{x_i, y_i\}$



We change the view point: The line is changing to settle down to the line with the lowest cost.



- The geometry give immediately the Normal Equation for Least Square Approximation:

$$\theta = (X^T X)^{-1} X^T \vec{y}.$$

Extend a vector space to a manifold

The set of vectors starting at the origin in R^3 =

a **Vector Space.**

The set of “oriented planes” in R^3 **=**

a manifold $\mathbf{(G_2R^3)}$

Grassmannian and their applications

- Instead of considering the set of all vectors through origin, we consider set of all k-planes through origin. →
- Grassmann manifold (or Grassmannian).

- Example: Study big video data

Examples of manifolds

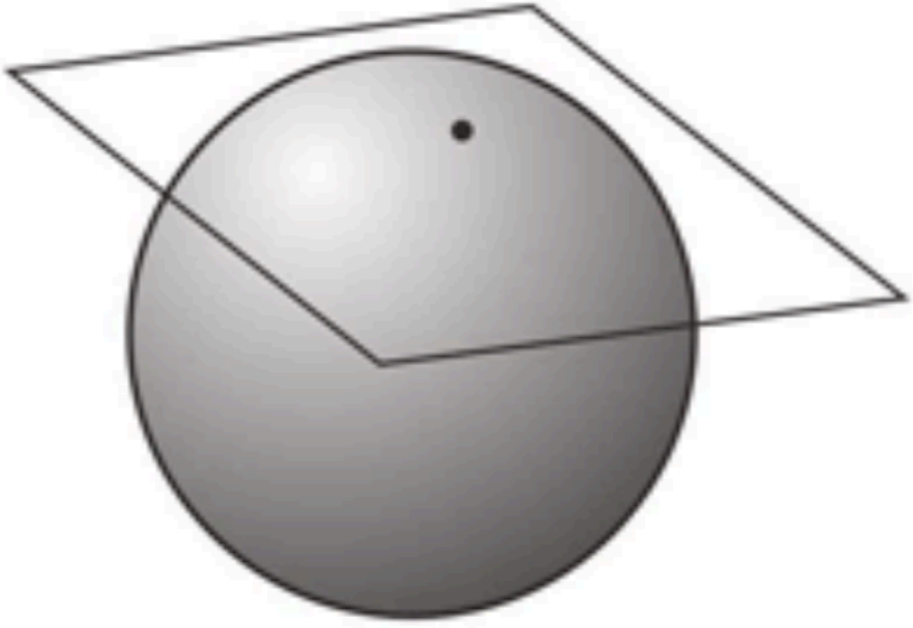
- The set of all rigid motions forms a manifold (in fact it is a Lie group).
- The set of all probability distributions forms a manifold. (called the statistical manifold).
- The set of Gaussian distributions forms a submanifold of the above.
- The set of all time series forms a manifold.
- The set of all curved shapes forms a manifold.
- The set of covariance matrices forms a manifold.
- Many many more….

Example: Video data--robust and fast extraction of foreground information

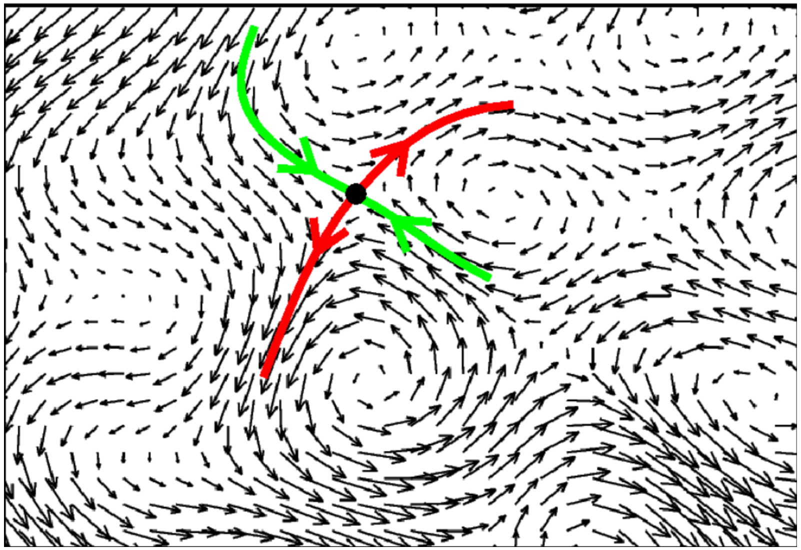
- https://sites.google.com/site/hejunzz/grasta

How to study (curved) manifolds?

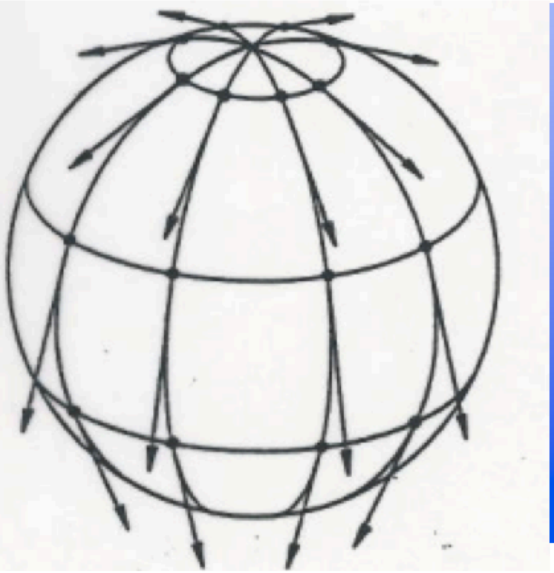
- Use tangent plane first
- Example: Tangent space to SO(3). More details Later.

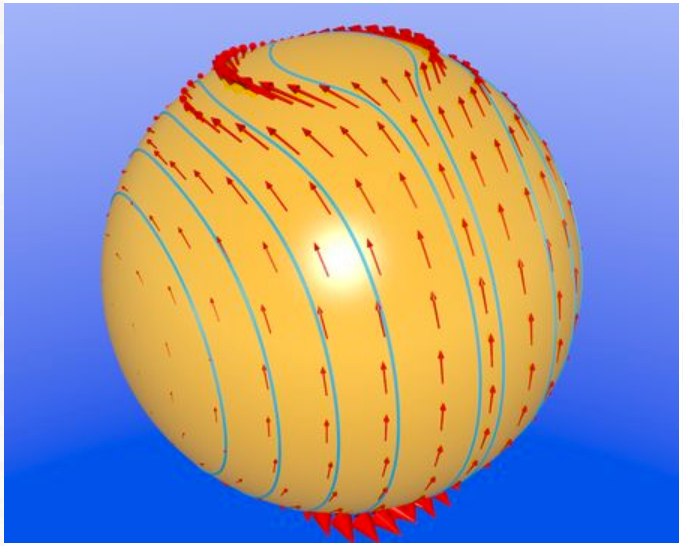


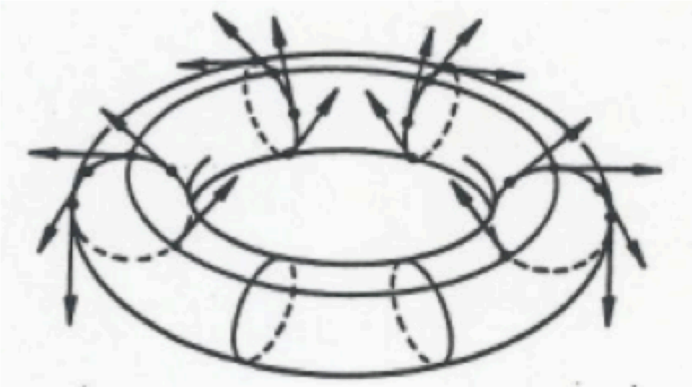
Then Use Vector Fields



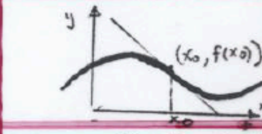
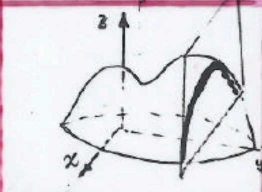
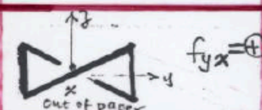
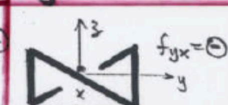
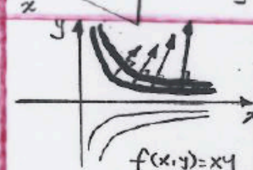
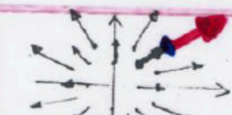
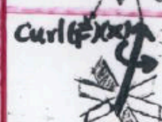
Extend the multi-V to manifolds







Recall all kind of derivatives in vector spaces

A	Big		Picture	of	Derivatives	(By Prof. Gu)
Type of funcns	Type of Derivatives	Notations	Pictures	Meanings	Remarks	Formulas
$f: \mathbb{R} \to \mathbb{R}$	Derivative of $f(x)$ at x_0	$\frac{df}{dx}$ or $f'(x)$		slope at $(x_0, f(x_0))$ of the curve		The tangent line at $(x_0, f(x_0))$: $\boxed{y = f(x_0) + f'(x_0)(x - x_0)}$
$f: \mathbb{R}^2 \to \mathbb{R}$ or $\mathbb{R}^n \to \mathbb{R}$	Partial derivatives w.r.t x or y	$\frac{\partial f}{\partial x} = f_x$ $\frac{\partial f}{\partial y} = f_y$		f_x = slope of graph in x-direction	Similarly, f_y = slope of graph in y-direction	(If exists) The tangent plane at $(a, b, f(a,b))$: $\boxed{z = f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)}$
$f: \mathbb{R}^2 \to \mathbb{R}$ or $\mathbb{R}^n \to \mathbb{R}$	Higher order partials: Here: 2^{nd} partials	$\frac{\partial^2 f}{\partial x^2} = f_{xx}$ $\frac{\partial^2 f}{\partial y^2} = f_{yy}$		f_{xx} = concavity in x direction $(f_y)_x$ = rate change of f_y as x increases	$f_{xx} > 0$: $f_{yy} < 0$:	Laplace's eqn: $\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} = 0$
$f: \mathbb{R}^2 \to \mathbb{R}$ or $\mathbb{R}^n \to \mathbb{R}$	Mixed partials	$\frac{\partial^2 f}{\partial x \partial y} = f_{yx}$			In the first picture, as x increases f_y increases from neg to positive $\Rightarrow f_{yx} = \oplus$	Spse $f(x, y)$ has continuous 1^{st}- and 2^{nd} partial deriv's $\Rightarrow \boxed{f_{xy} = f_{yx}}$
$f: \mathbb{R}^n \to \mathbb{R}$	Directional derivative of $\mathbb{R}$-valued funcn of 2-variable	$D_{\vec{u}} f(\vec{a})$		Rate change of f in the direction of v.	Any directional derivative is completely determined by the direction and gradient.	$\boxed{D_{\vec{u}} f(\vec{a}) = \nabla f(\vec{a}) \cdot \vec{u}}$ $\vec{u}$ (a unit vector!)
$f: \mathbb{R}^n \to \mathbb{R}$	Gradient of real valued function of 2 or 3 variables	∇f $\nabla f(\vec{a})$		$\frac{\nabla f(a,b)}{\lVert \nabla f(a,b) \rVert} \triangleq \vec{u}, a$ direction of steepest ascent. i.e. where $D_{\vec{u}} f(a)$ is maximized	If S is a surface given by: $f(x,y,z) = c$ then an equation for tangent plane to S at $\vec{x}_0$: $\nabla f(\vec{x}_0) \cdot (\vec{x} - \vec{x}_0) = 0$	$\nabla = \vec{i} \frac{\partial}{\partial x} + \vec{j} \frac{\partial}{\partial y} + \vec{k} \frac{\partial}{\partial z}$ $\nabla f = \vec{i} \frac{\partial f}{\partial x} + \vec{j} \frac{\partial f}{\partial y} + \vec{k} \frac{\partial f}{\partial z}$
$\vec{F}: \mathbb{R}^n \to \mathbb{R}^n$	Divergence of a vector field	$\mathrm{div}(\vec{F}) = \nabla \cdot \vec{F}$		Measurement of the "net mass flow" of $\vec{F}$ in or out at a pt.	or $\mathrm{div}\, \vec{F}$ = rate of expansion per unit volume.	$\mathrm{div}\, \vec{F} = \nabla \cdot \vec{F} = \frac{\partial F_1}{\partial x_1} + \frac{\partial F_2}{\partial x_2} + \cdots + \frac{\partial F_n}{\partial x_n}$
$\vec{F}: \mathbb{R}^3 \to \mathbb{R}^3$	Curl of a vector field	$\mathrm{curl}(\vec{F}) = \nabla \times \vec{F}$		A tiny twig or paddle at $x \in \mathbb{R}^3$ will spin around the axis in dir^n of vector curl $\vec{F}$ obeying R-H rule w/ angular velocity $\frac{1}{2} \mathrm{curl}\, \vec{F}(\vec{x})$ and angular speed $\frac{1}{2} \lvert \mathrm{curl}\, \vec{F}(\vec{x}) \rvert$ radians/sec.		$\mathrm{Curl}(\vec{F}) = \nabla \times \vec{F} = \begin{vmatrix} i & j & k \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ F_1 & F_2 & F_3 \end{vmatrix}$
$\vec{x}: \mathbb{R} \to \mathbb{R}^n$	Tangent to parametrized curve	$\dot{\vec{x}}(t)$		Velocity vector	$\dot{x}(t) = (\dot{x}_1(t), \cdots, \dot{x}_n(t))$	

Derivatives on Manifolds

- All kinds of derivatives in Multi-V

Extend
to manifolds

- Connections and Covariant Derivatives

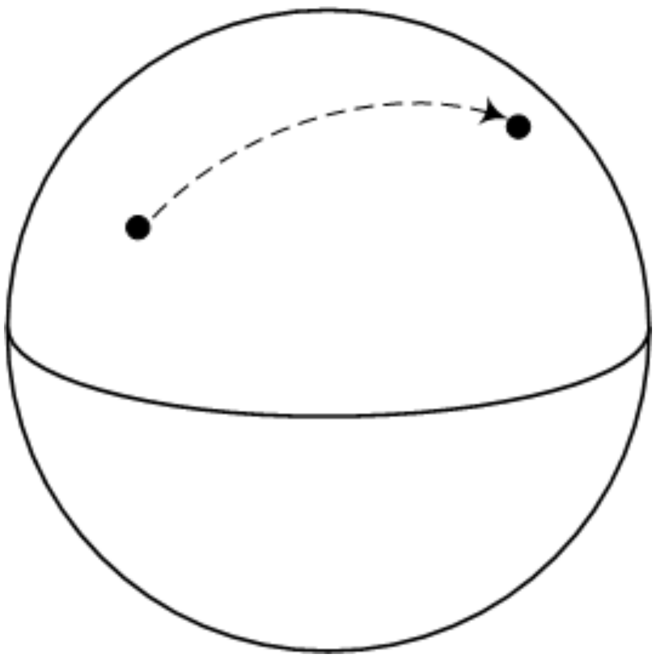
Important: Riemannian Metric (Not Euclidean metric!)

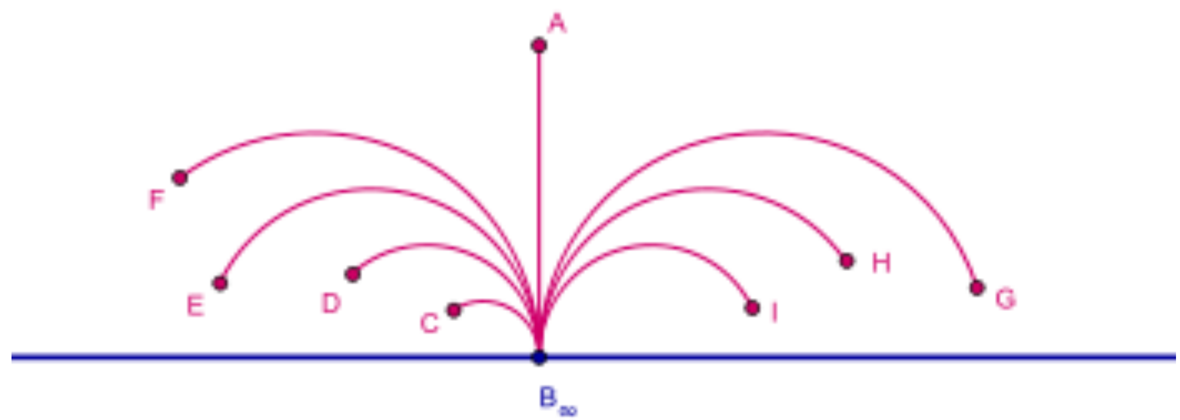
- Vector space + inner product = Euclidean space

Extend to manifolds

- Manifold + Riemannian metric = Riemannian manifold

What is Riemannian Measurements?





Key is using Riemannian metrics and gradient decent along geodesics on a manifold.

Key ideas: Moving frames & Parallel Transport

- Basis or Orthonormal basis(frames) in R^n

Extend
to manifolds

- Moving Frames on Manifold

We will study the Key Concepts about differentiation on manifold

- When the moving frames changes with time, it traces out a "curve" of moving frames.
- E.g. Rigid Motions
- You can express the derivatives of the moving frame in the frame itself.
- Key: Christoffel symbols.

$$\Gamma^{\lambda}_{\mu\nu} = \frac{1}{2} g^{\lambda\beta} \left(\frac{\partial g_{\mu\beta}}{\partial x^{\nu}} + \frac{\partial g_{\nu\beta}}{\partial x^{\mu}} + \frac{\partial g_{\mu\nu}}{\partial x^{\beta}} \right)$$

We also have gradient descent on manifold for Machine Learning!

- Gradient descent and newton’s algorithms in Euclidean Spaces

Extend
to manifolds

- Gradient descent and newton’s algorithms in on manifolds

Integration on Manifolds

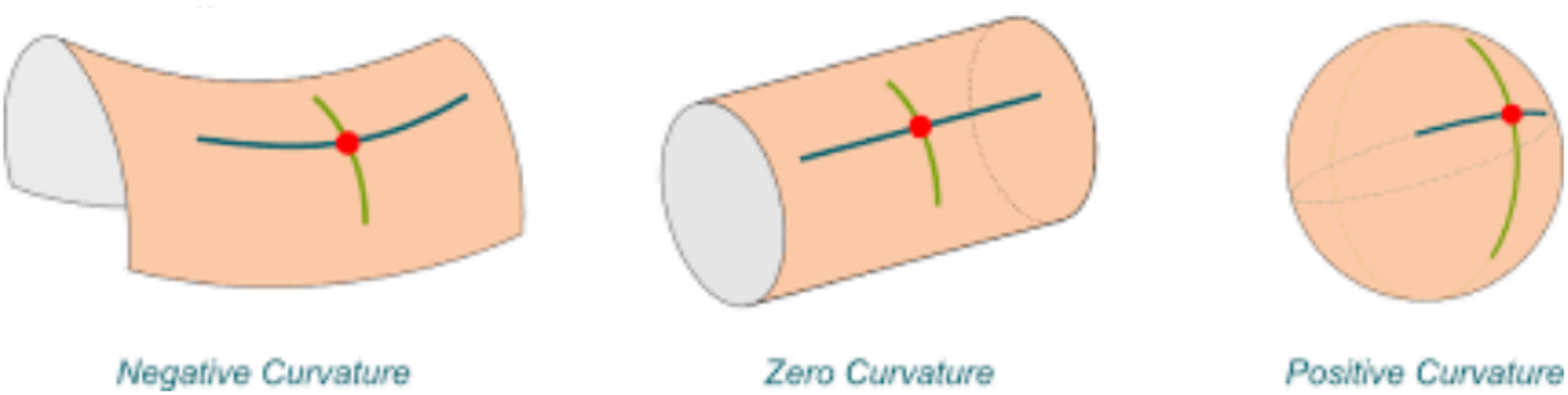
- All kinds of integration in Multi-V

Extend
to manifolds

- Integrations on manifolds

Curvatures and applications

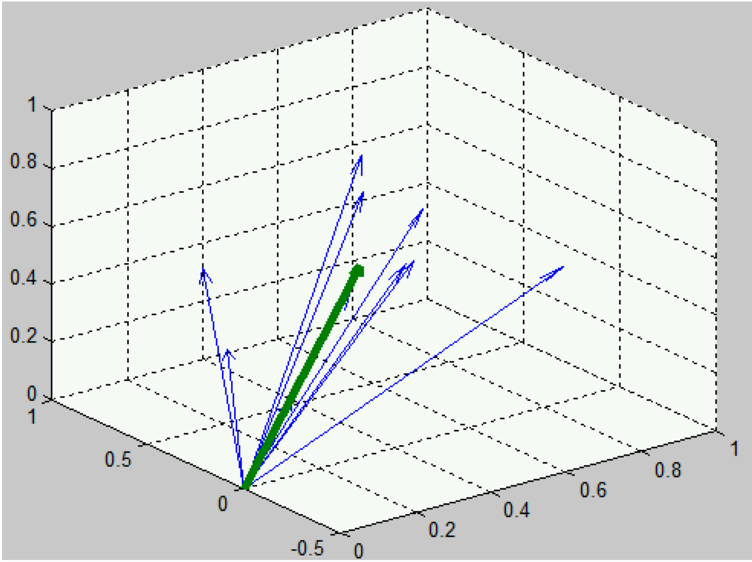
- As Key data features
- Face recognition
- Model various problems using curvature including network congestion.
- Many many more…



We will study curves first.

Even for curves, there are a lots of applications, say in computer vision!

Q: How to find the "mean shape" of the following shapes?



Recall: we can find mean vector in R^n

How to get this kind of "curve-shape" Data?

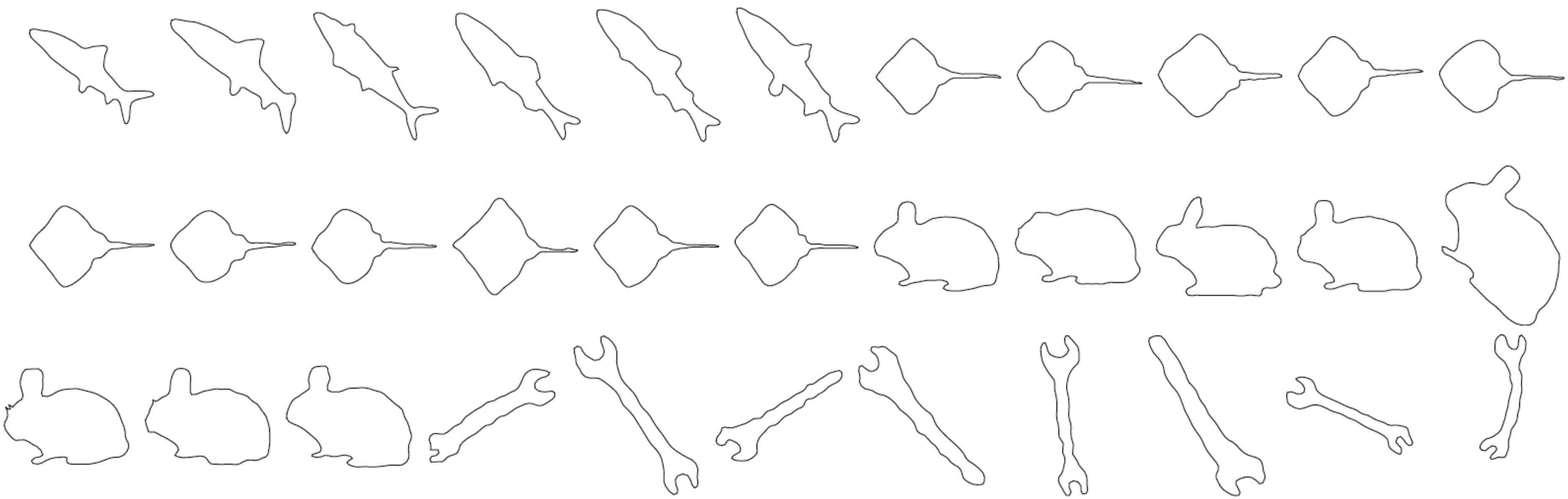
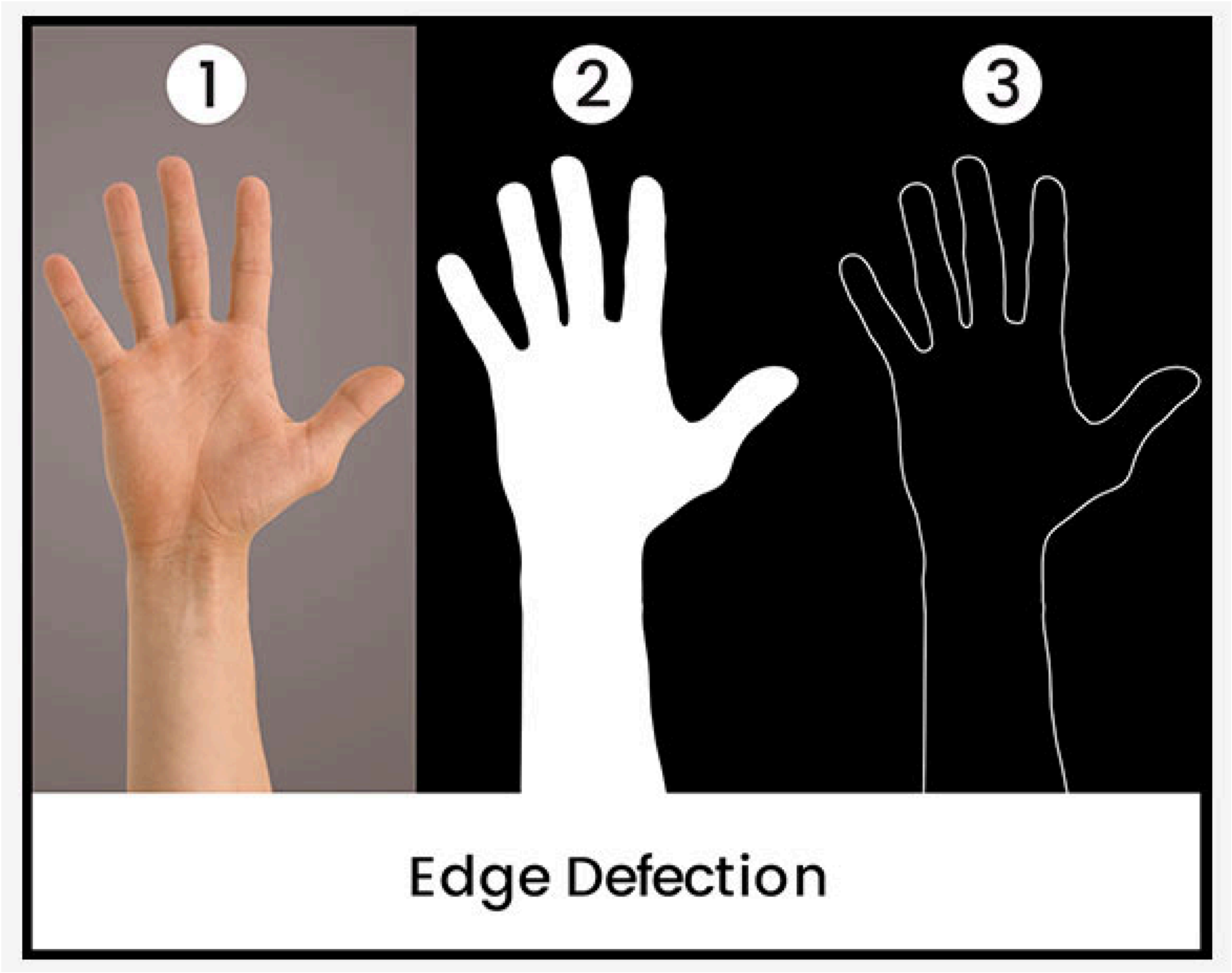


Figure 1: Selection of shapes from the dataset [14].

[14] B. Kimia. Computer vision group at lems at brown university, database of 99 binary shapes. https://vision.lems.brown.edu/content/available-software-and-databases, 2015.

You can build these kinds of data too!



Shortest “path” between given two shapes

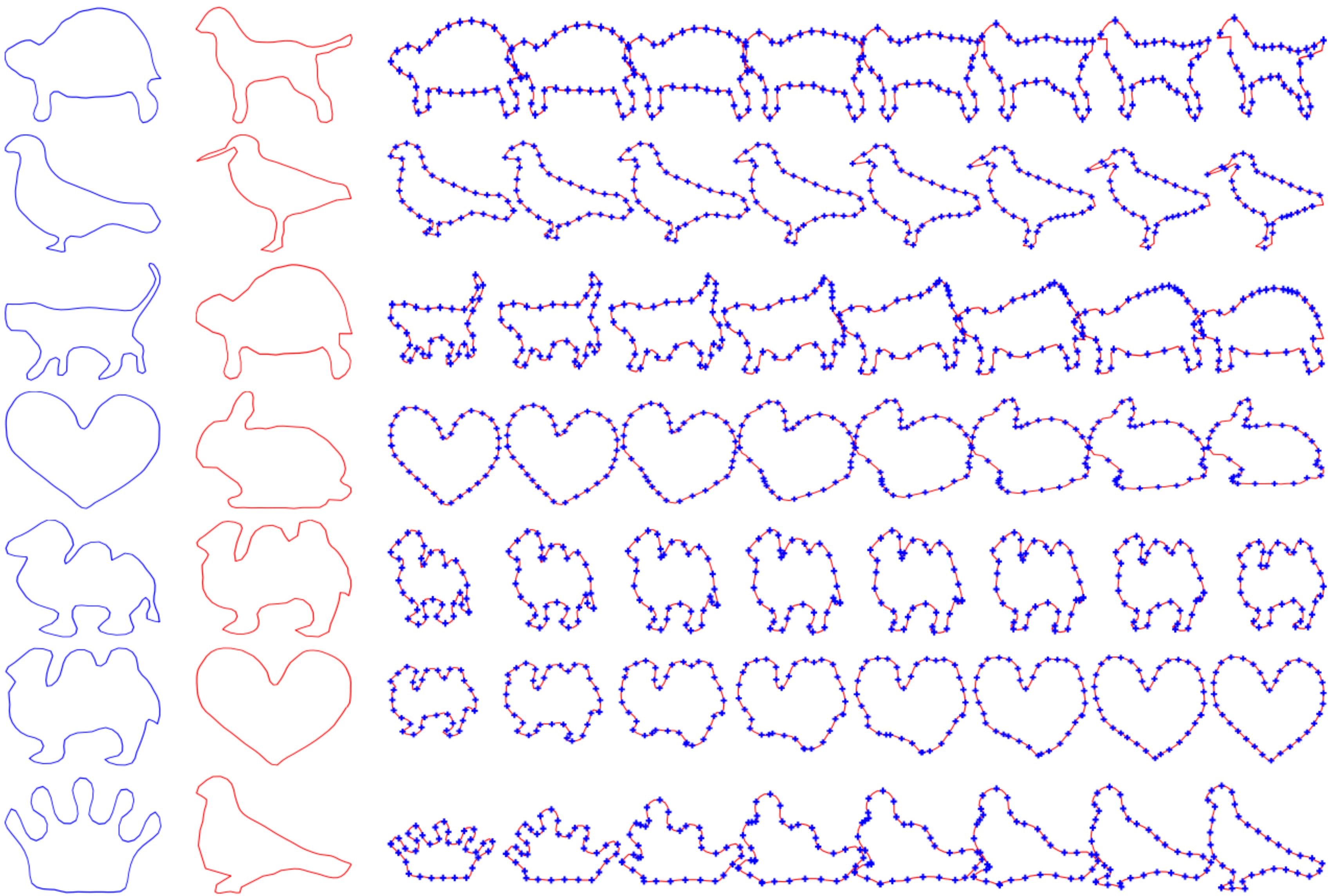


Figure 2. Row-wise geodesic paths in $\mathcal{C}$ between the pair of curves shown to the left.

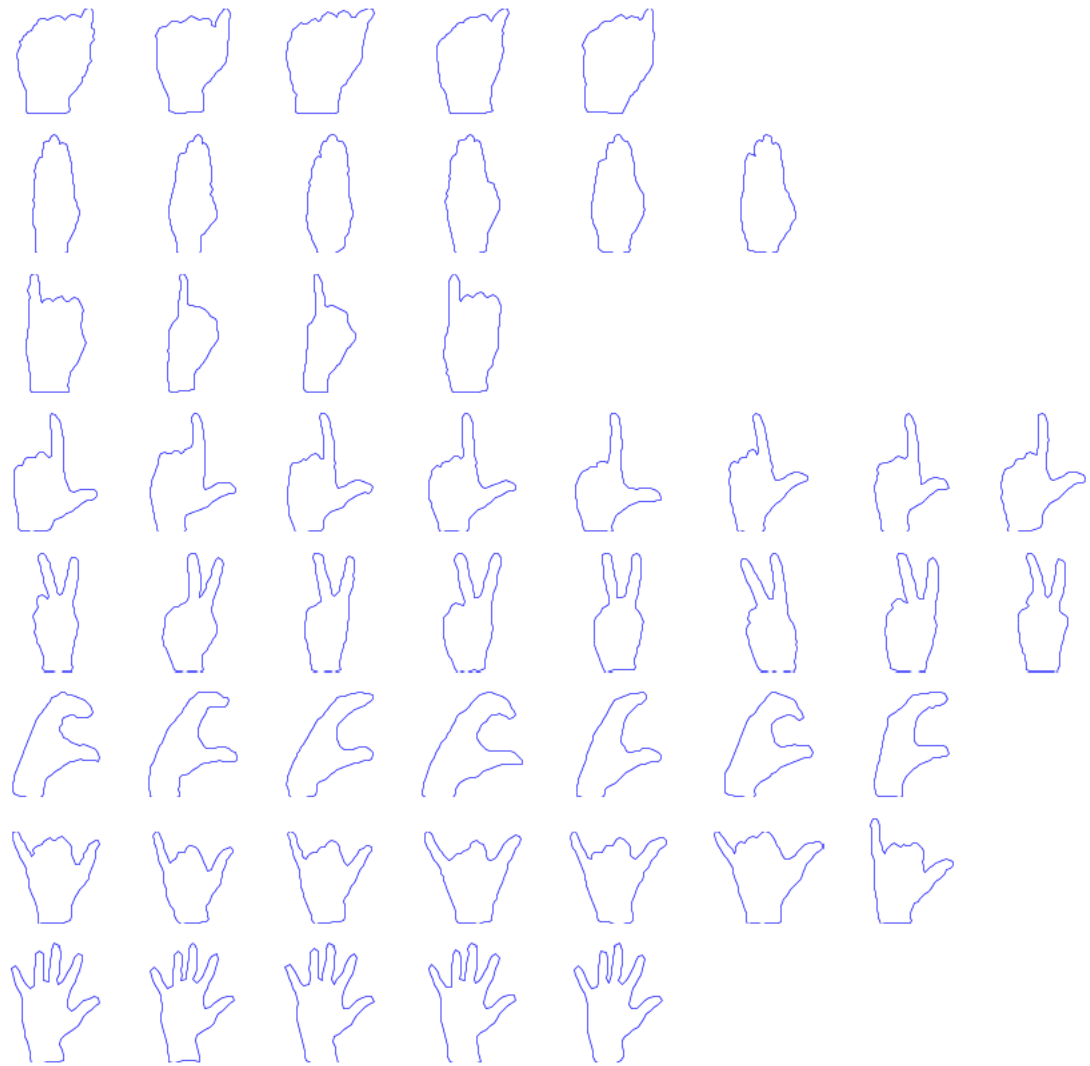
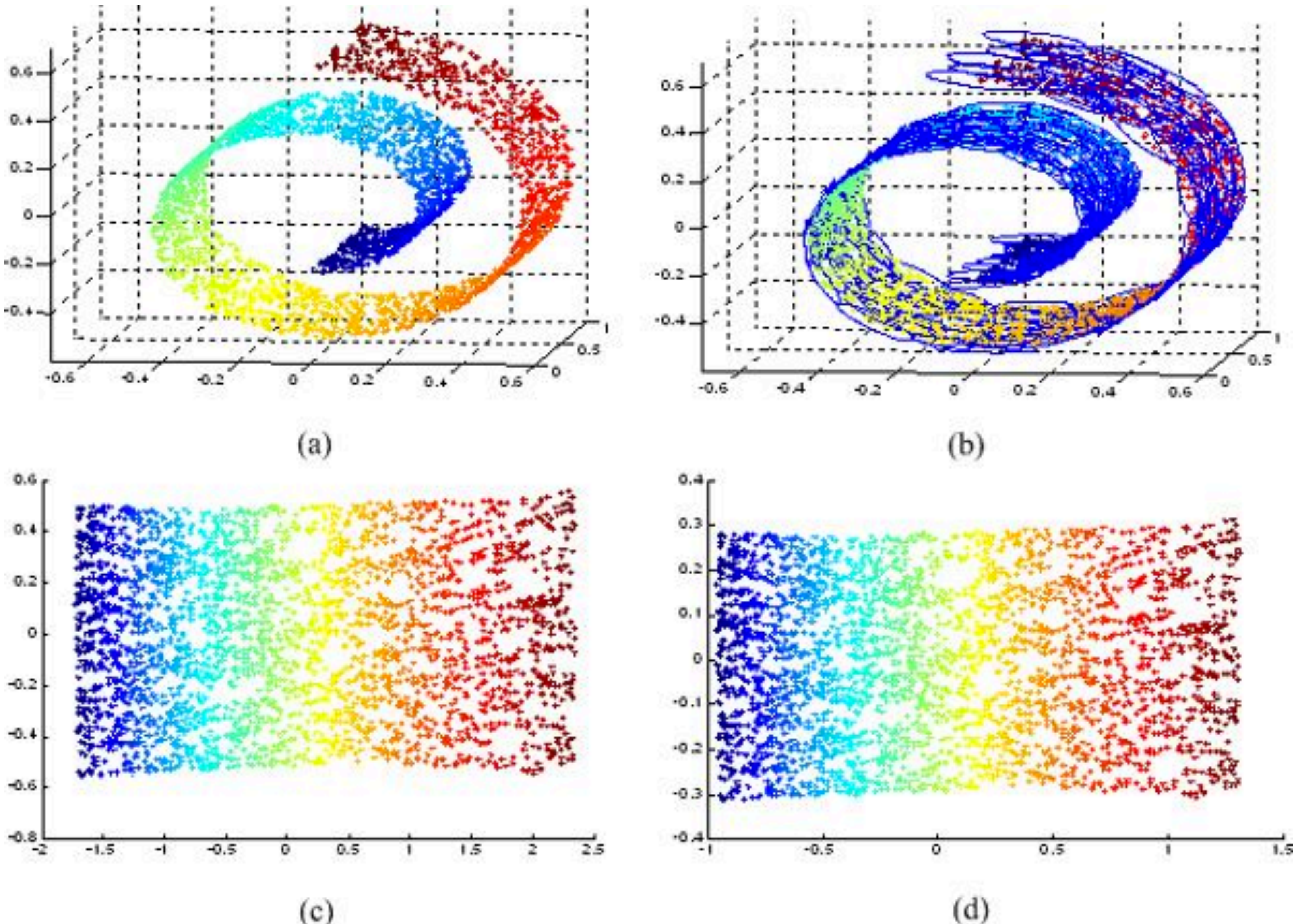


FIGURE 3. Eight clusters obtained using a hierarchical clustering algorithm using the elastic geodesic distance with DP alignment as metric.

Then we will study surfaces



(a) (b) (c) (d)

- Recall: We turned key properties in $\mathbf{R}^n$ to define abstract vector space.
- We will turned the key properties for regular surface in $\mathbf{R}^3$ to define abstract manifold.
- Later we will turned the key properties of a tangent bundle to define abstract vector bundle (only if time permits).

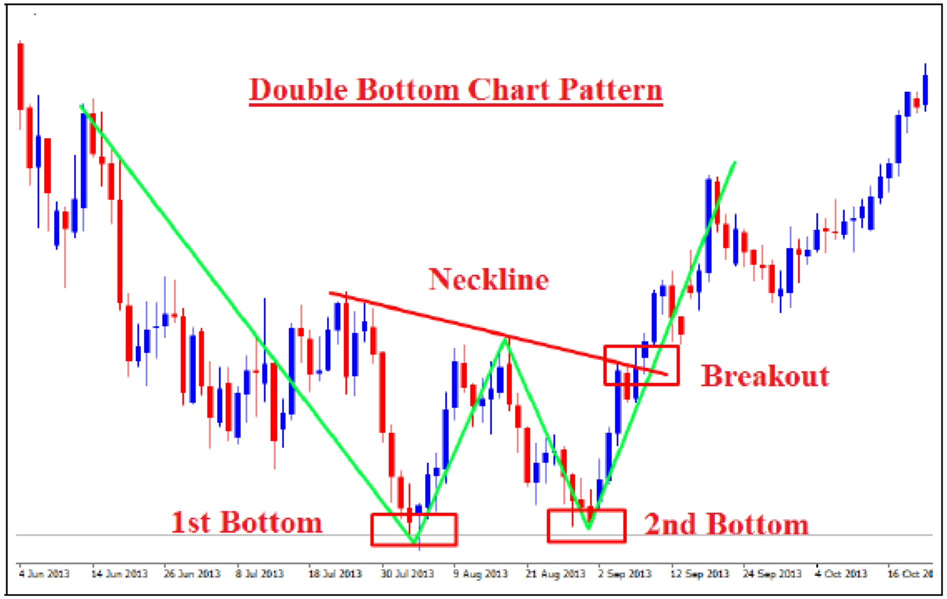
Extend Shortest Distance to Geodesics

- In Euclidean space
- On a sphere
- On a manifold of curve shapes

Why is Differential Geometry Powerful?

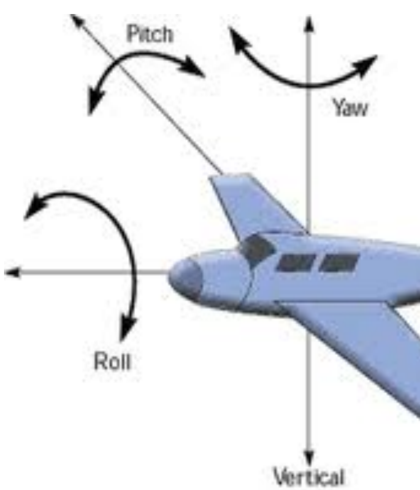
- Manifolds provide clever ways to model real world, big data and theoretic physics problems.
- Help to identify best metrics or distance functions for the problems at hand.
- Captures big data in low dimensional intrinsic spaces.
- Provide theoretical background for machine learning.

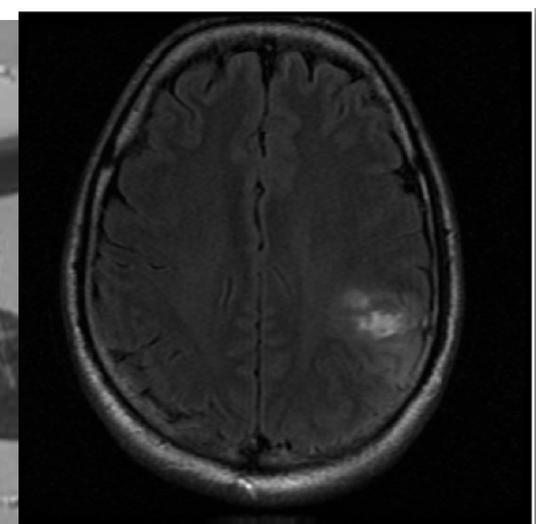
Examples of Applications of Differential Geometry



Which sensors to use to detect components of the chemicals?

UAV Anomaly Detection

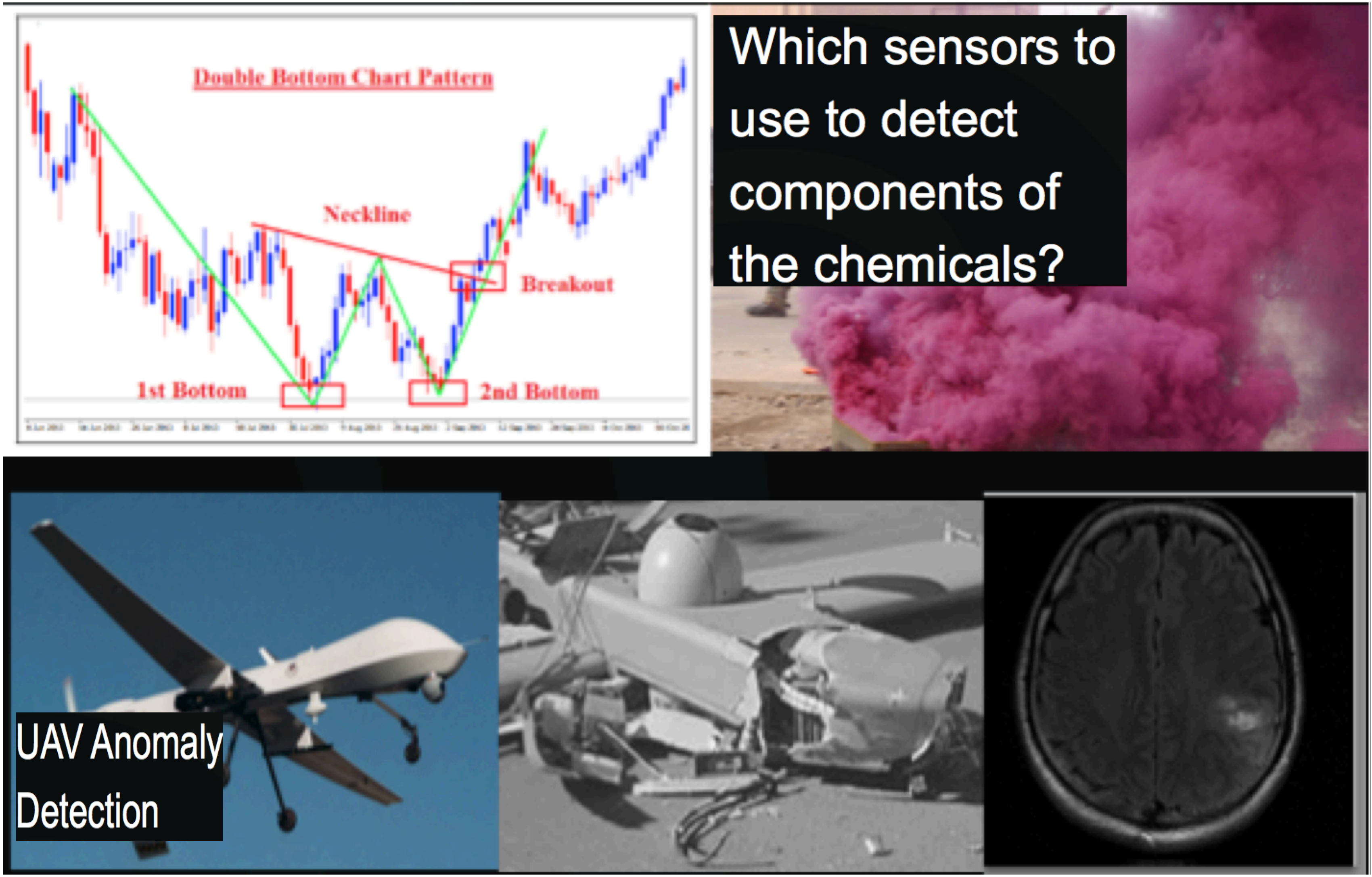




Detecting UAV Anomalies Using Manifold of Rigid Motions.

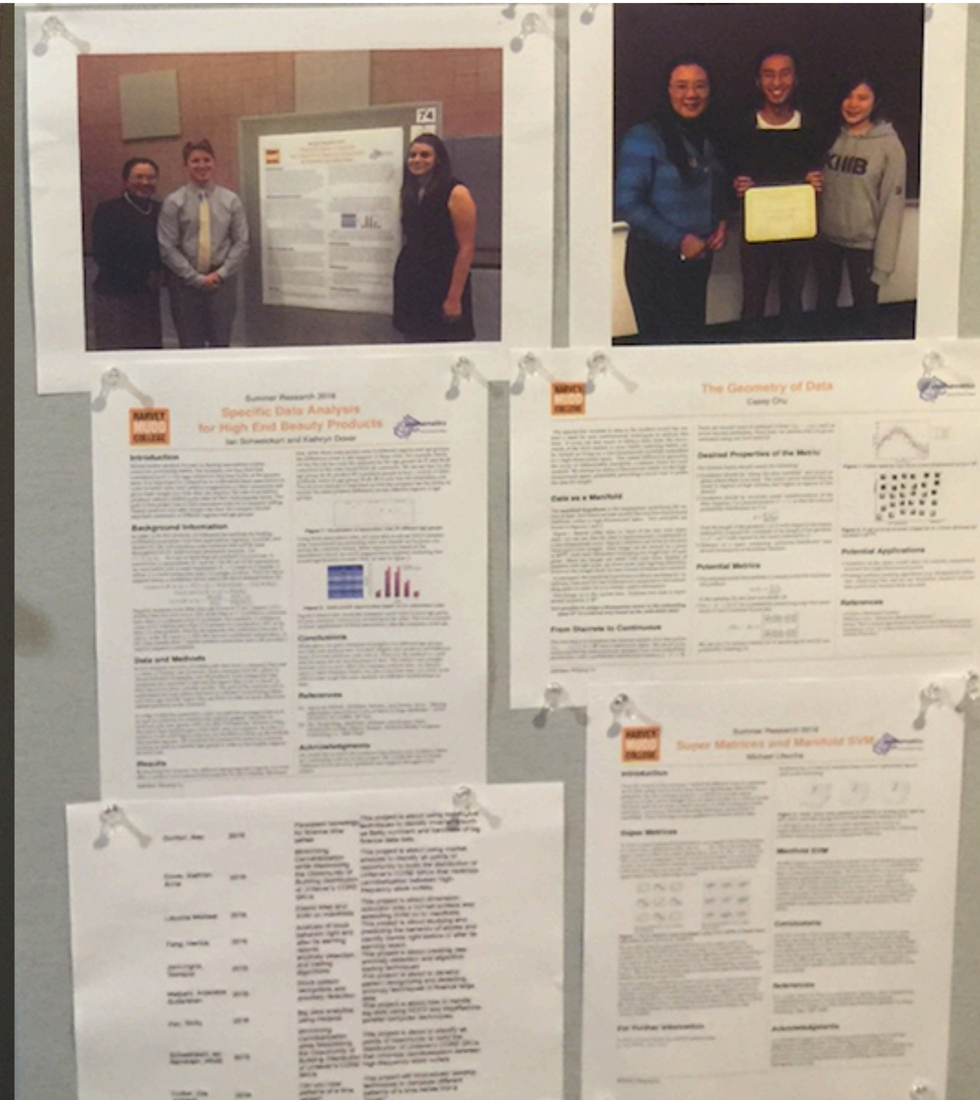
Selecting Chemical Sensors using Stiefel Manifolds.

Find patterns in stock data & features in brain images.



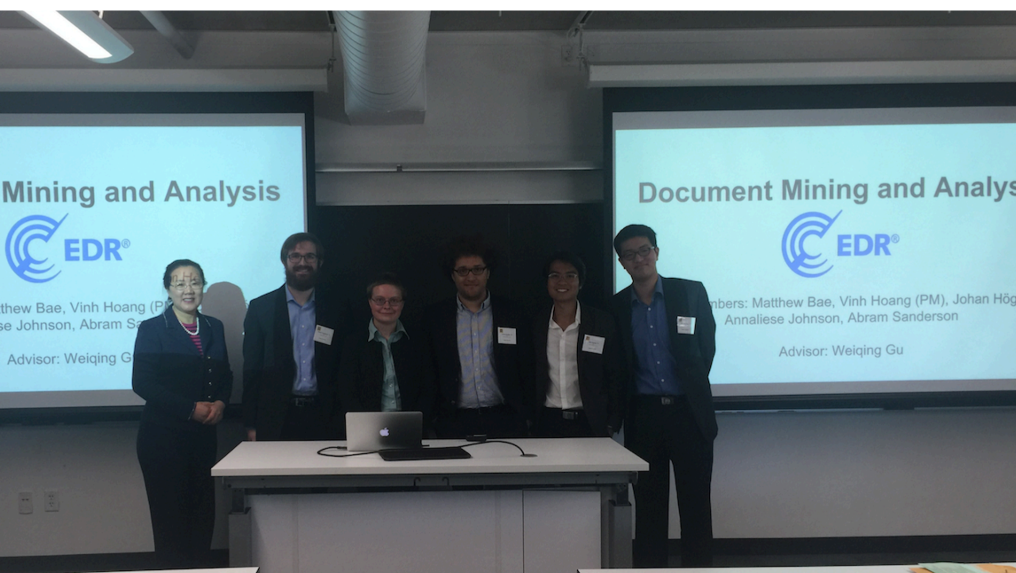
Manifold is powerful in capturing large data use only its small intrinsic dimension!

Big Data Summer Researches



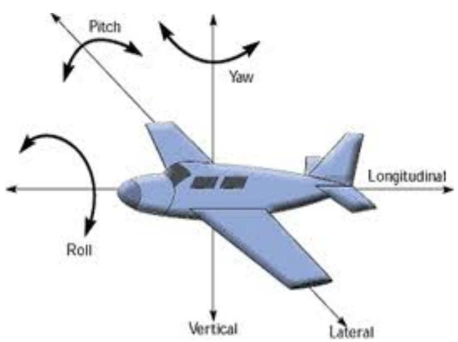
Applications in Math Clinics.

Example: EDR clinic



Example1: Problem Statement

- Unmanned Aerial Vehicle is not really unmanned! At least 3 operators behind each UAV.
- Why? Because operators have to constantly monitor the UAV in case of anomalies to avoid damaging and destroying the UAV.
- How to reduce the operator-to-vehicle ratio?
- Need to develop UAV anomaly detection and auto alarm techniques.



- W. Gu et al., *The Intersection of Robust Intelligence and Trust in Autonomous Systems*, AAAI 2014: 28th AAAI Conference on Artificial Intelligence.
- W. Gu et al., *UAV Obstacle Avoidance Via Users Geometric Intuition*, submitted to 11th ACM/IEEE International Conference on Human-Robot Interaction 2016.

Example1 conti: Big Data Challenge

Human behaviors | UAV-Health & Status | Lost GPS or Communications | Environmental conditions | … | Cyber attacks

Example: The causes of this mishap

1) Engine overheat: *coolant line leaking*

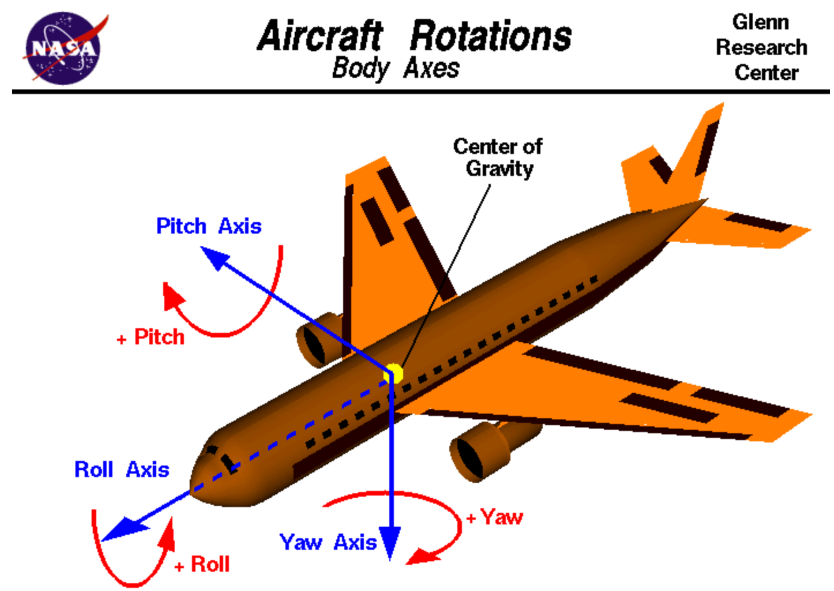
2) Lost control: *human error*

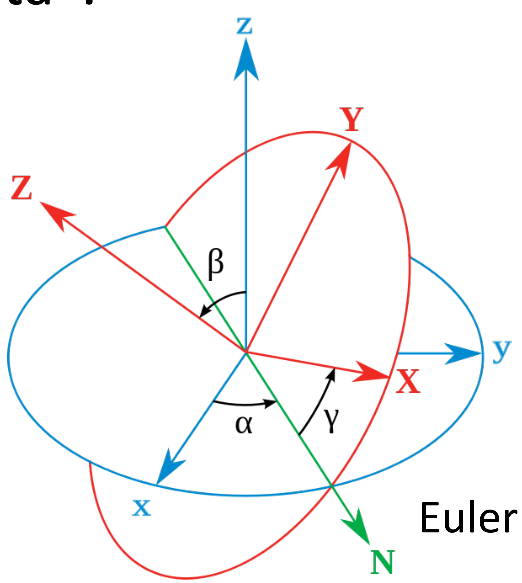
Lessen Learned: Many mishaps resulted from **combined causes** but ***no metric*** *for a combination of anomaly behaviors!*

E.g. No model for combinationa affect of weather & human's throttle control for UAV's behavior.

How manifold is used in large nonlinear UAV dynamical data analysis?

Note: Each type of "data" would has an effect on the UAV. We transform all UAV dynamical data to "manifold valued data".





Euler Angles

$$SO(3) = \{M \in M_{3\times 3}(\mathbb{R}) \mid MM^T = I,\ \det(M) = 1\}$$

$\hat{\mathbf{\Omega}}^b = Q^T \dot{Q}$ *(UAV body angular velocity. Q in SO(3)).*

$$\begin{bmatrix} \Omega_1^b \\ \Omega_2^b \\ \Omega_3^b \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

$$\mathbf{\Omega}^b = B^{-1}\dot{\mathbf{\Theta}}$$

$$B = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}$$

Mathematically Modeled Causes-&-Effects

Use quaternion multiplication to avoid gimbal lock

- Each cause could either rotate the body frames of the UAV or translate it to a different location.
- It is easy to represent a translation of an UAV by a vector in R^3.
- We use quaternion numbers to represent rotations of the UAV.
- Let $q \in \mathbb{H}$ be a unit quaternion. Define $R_q(p) \doteq qp\bar{q}$ where $p \in \mathbb{R}^3 \cong \mathrm{Im}(\mathbb{H})$
- Composition of two rotations is just multiplying q_1 & q_2.

- Handle the dynamics of moving frames of the UAV using tensor operations. (See example below.)

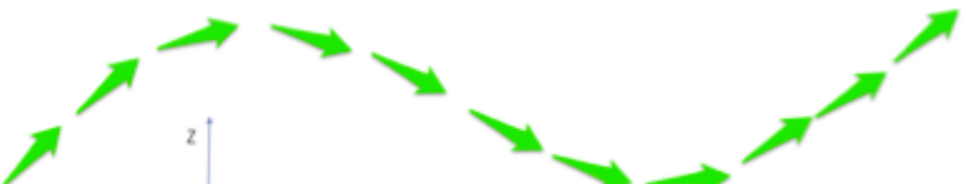
$$
\begin{aligned}
L(\alpha, \mathbf{v}) &= (I - \mathbf{v} \otimes \mathbf{v}^*) \cos\alpha + \hat{\mathbf{v}} \sin\alpha + \mathbf{v} \otimes \mathbf{v}^* \\
(\mathbf{v} \otimes \mathbf{w}^*)\mathbf{u} &= (\mathbf{w} \cdot \mathbf{u})\mathbf{v}, \ \forall\, \mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^3
\end{aligned}
$$

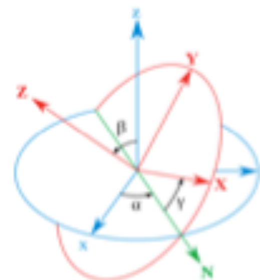
$$
\hat{\mathbf{v}} = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}
$$

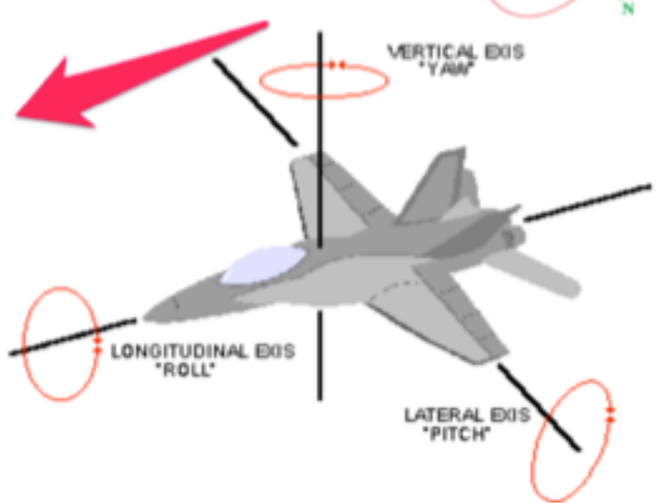
$\mathbf{a} \times \mathbf{b} = \hat{\mathbf{a}}\mathbf{b}$

***Key*: L could act on moving frames.**

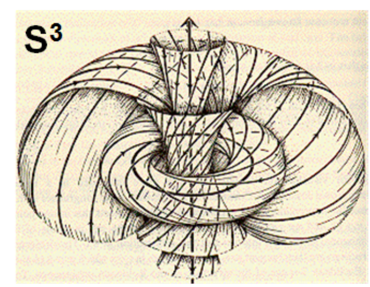
- *Each rotation is Represented by q, a unit vector on S^3*
  Since $R_q = R_{-q}$, now The manifold is RP(3).







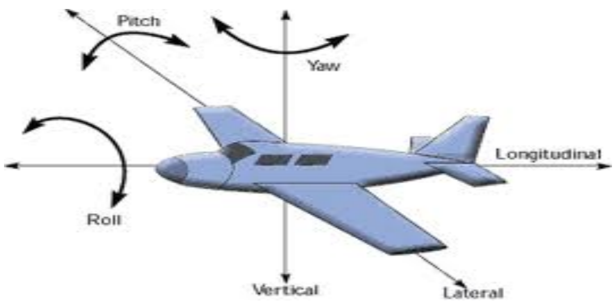
Euler's Rotation Theorem: In R^3 any displacement of a rigid body such that a point on the rigid body remains fixed, is equivalent to a single rotation about some axis that runs through the fixed point.

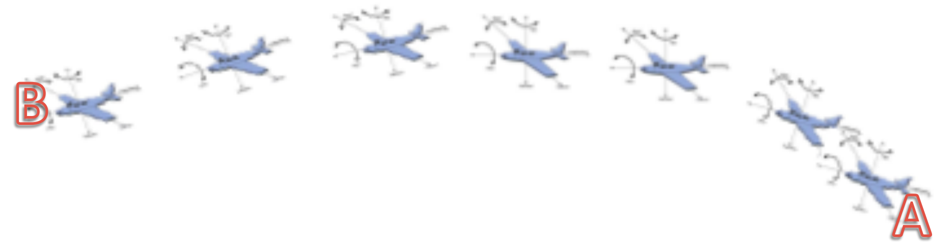


By identify antipodal points

M = SO(3) = RP^3

Dimension reduction use manifold





- The sensor data transformed to pitch, roll , yaw, and positions for an UAVs
- **M** = the set of data in SO(3) “x” $\mathbf{R}^3$ transformed from of 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.

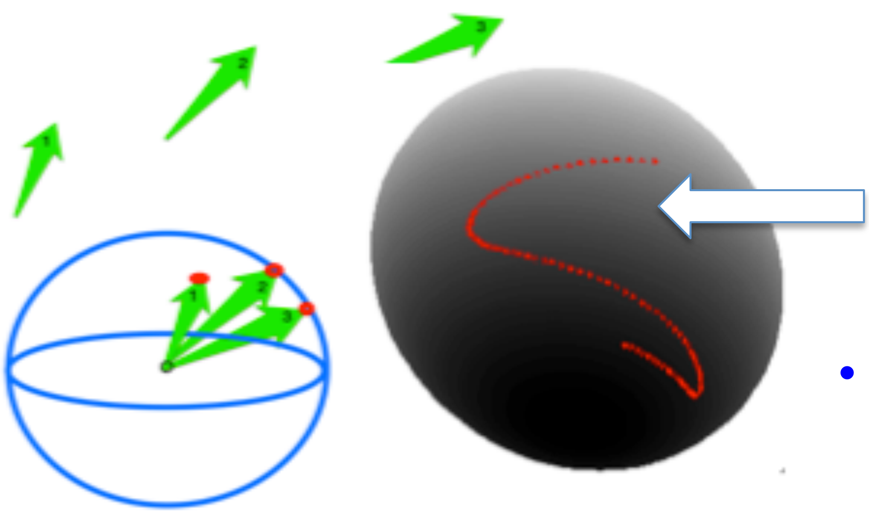
As a UAV flying, it’s behaviors trace out a curve on TM.

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**.*

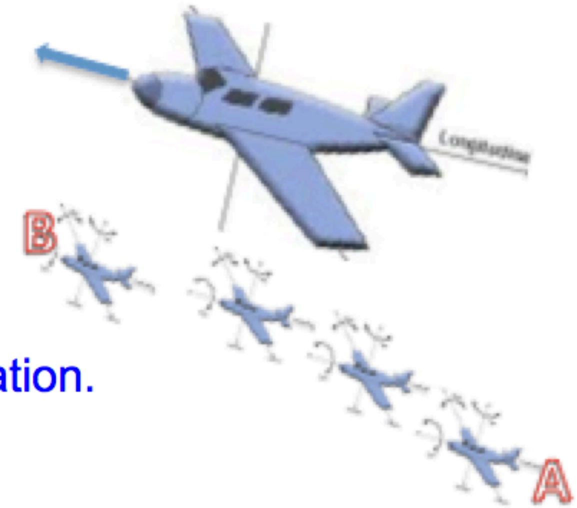
- ***Key: Created a dimension-reduction technique for nonlinear data.***

So What? How Do They Work to Detect Anomalies?

A Simplified Example for Concept Understanding

1. How do the "headings" of the following flight path look like?

- Only consider UAV heading Directions (depend on time)



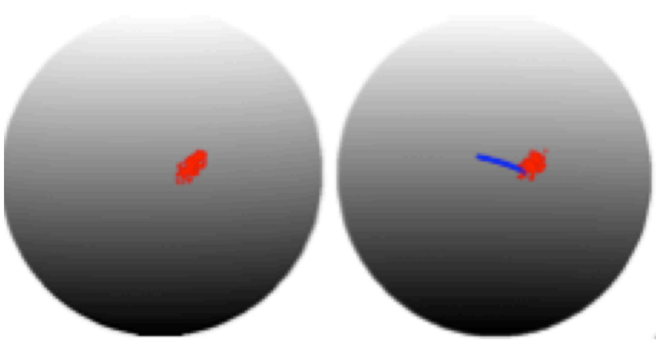
2. Mission Objective: Transport a missal from site *A* to site *B* as soon as possible.

Q: *Dose the pilot have to constantly monitor the UAV?*

3. How to detect anomalies? (See figures below}.

4. Back-identify: a strong wind just started causing the deviation.

Figure L: Normal neighborhood of UAV headings in a specified direction;
Figure R: Deviation beyond bdry of normal nbhd considered anomalous



5. A warning auto issued

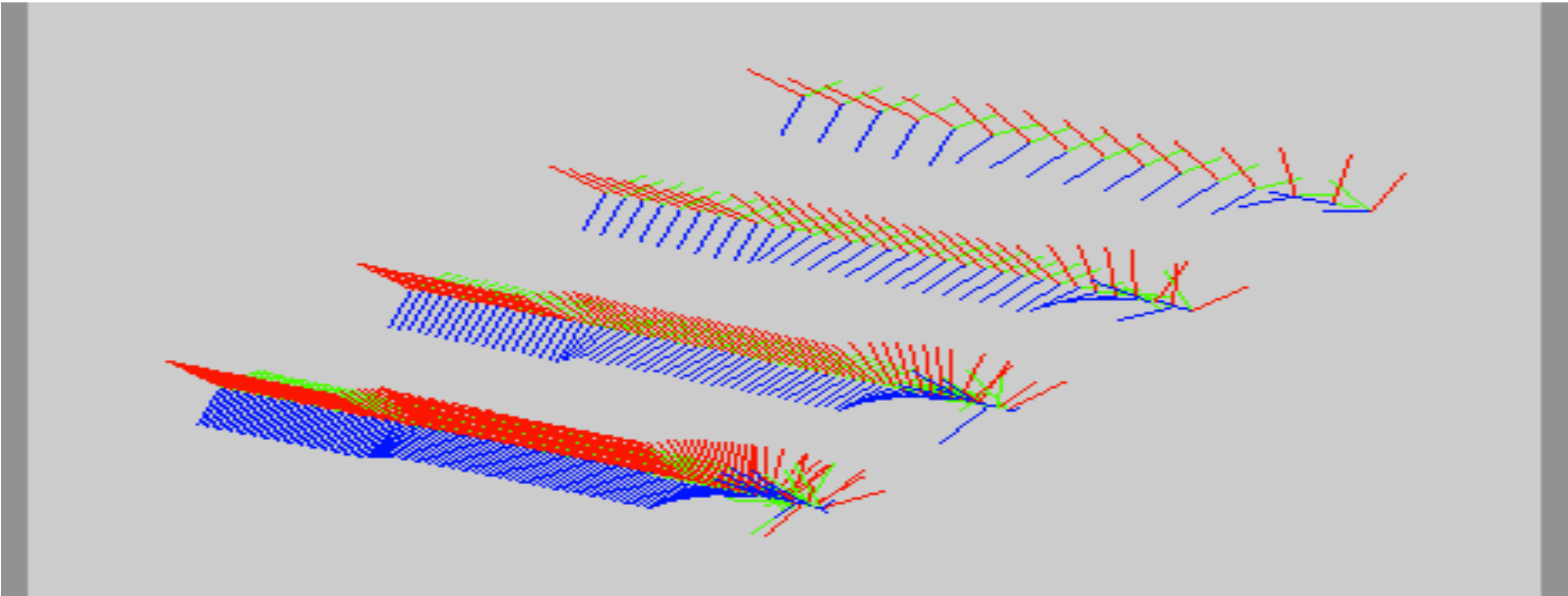
6. The operator corrected UAV's deviation from its mission path.

Note: Smaller the neighborhood, Less mission cost!

Triaged & compressed data use wavelet method

- First we extended wavelet method from R^n to manifold setting.

US_Air Manifold Data

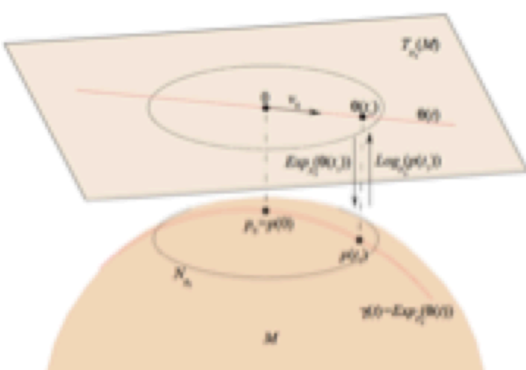


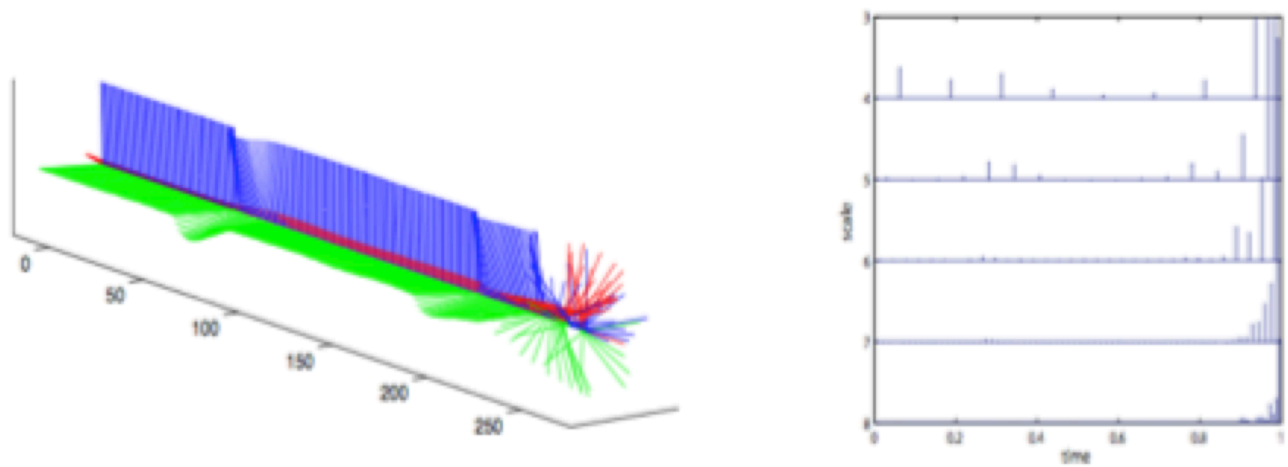
Example of using Frobenius norm on SO(3)

Used true data for the accident of US Air Flight 427

For example, we consider a dataset of orientations of a UAV as a function of time. The data themselves were converted from Pitch/Roll/Yaw to time series of orientations in SO(3). The raw data are illustrated in panel (a) of Figure 3 below and display mostly normal behavior, with two ‘bumps’ and then a catastrophic ‘swerve’ at the very end. This UAV crashed to the ground. The measurements of the metric function are displayed in panel (b) of Figure 3 as a function of scale and location. It is evident that the measurements are small except at the end of the time interval in question when the crash occurred.

We could use such a metric function to define thresholds and red flags to give warnings to the operators so that they could “correct” the UAV and avoid catastrophic situations.

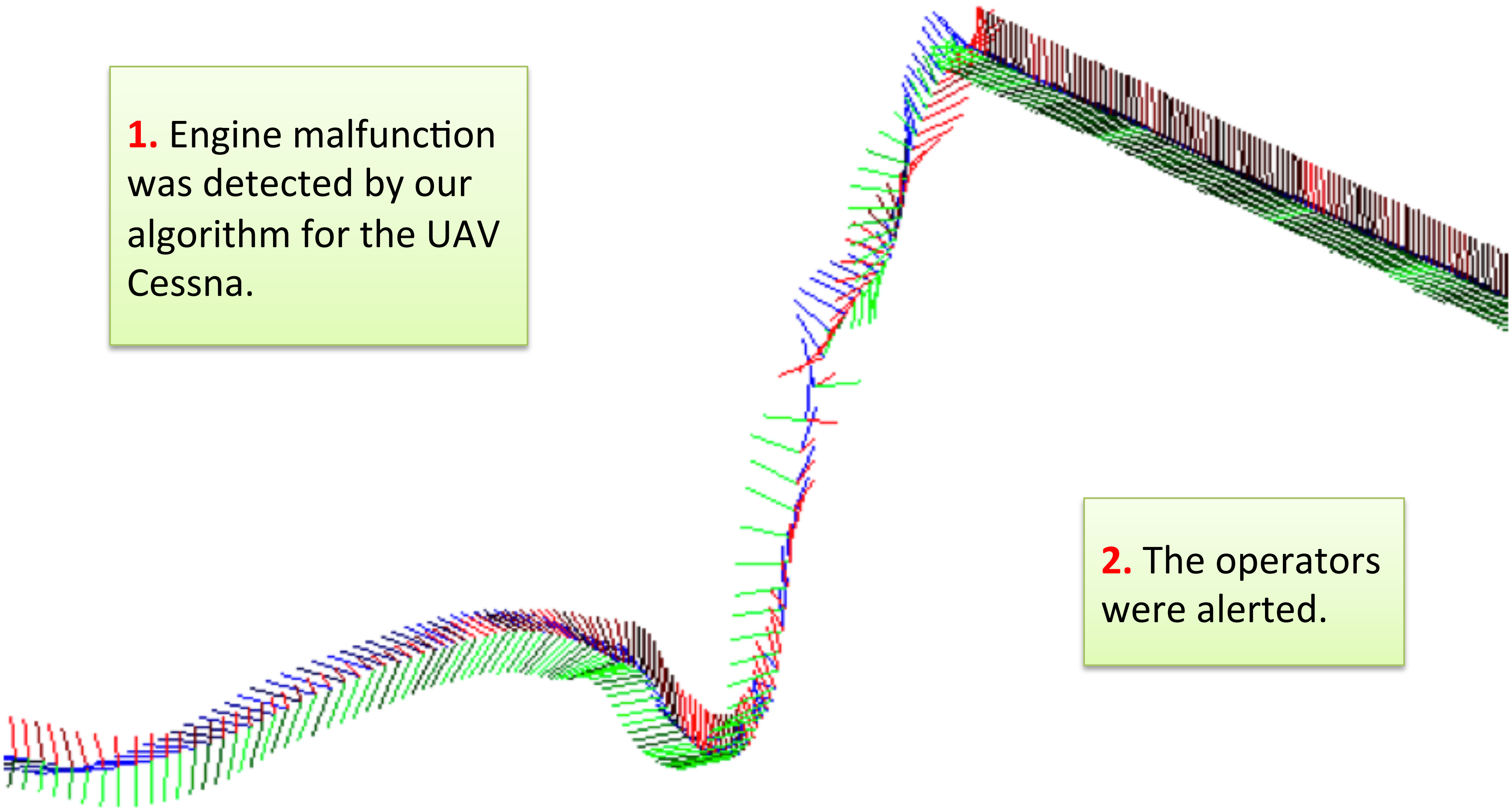




(a) (b)

Figure a: UAV orientation versus time; **Figure b:** Frobenius Norm of corresponding so(3)-wavelet coefficients

Result: Example of Anomaly Detection: *Engine malfunction*



1. Engine malfunction was detected by our algorithm for the UAV Cessna.

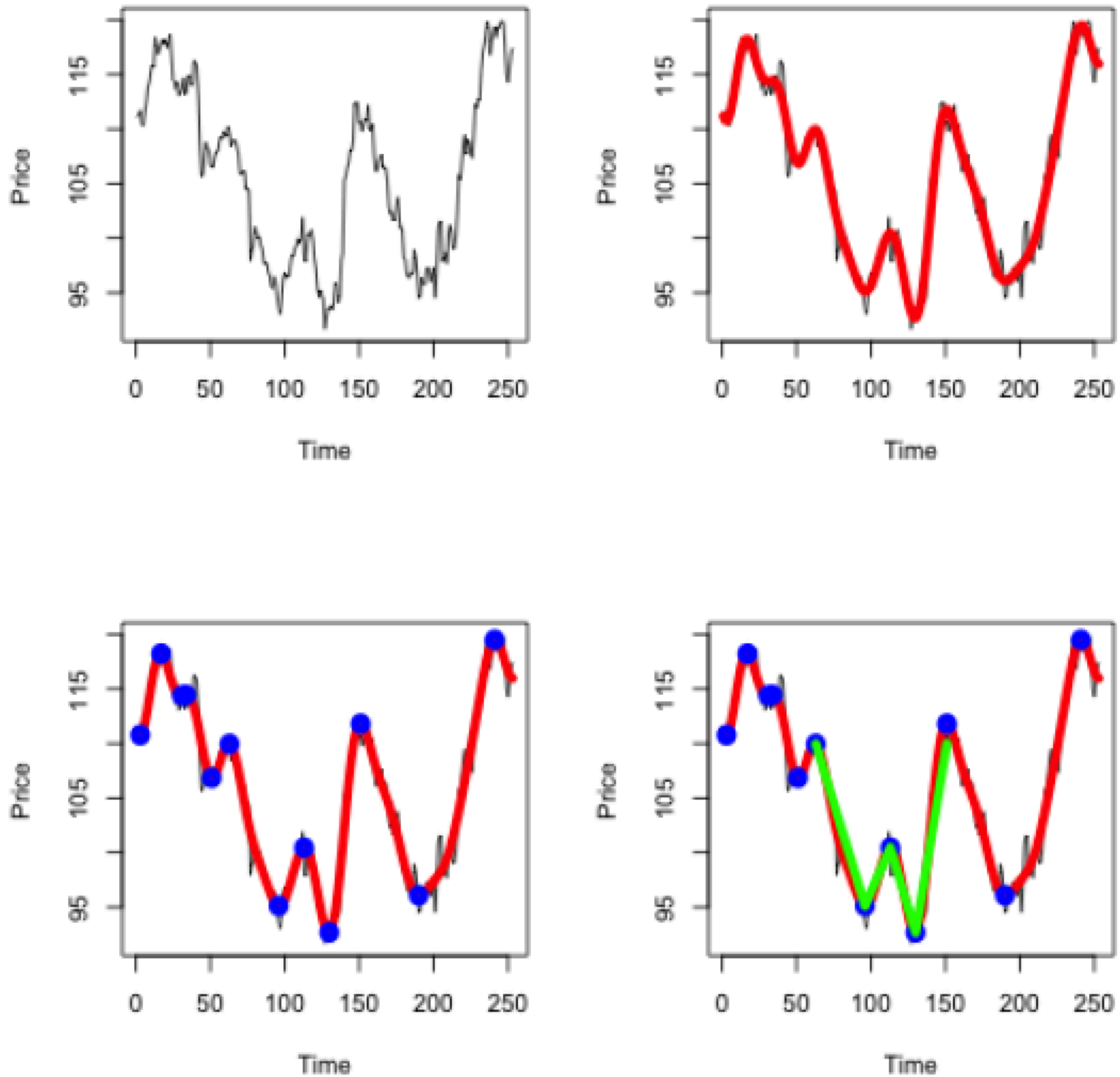
2. The operators were alerted.

3. The control was regained by a pilot with no ground collision and the UAV was safely landed.

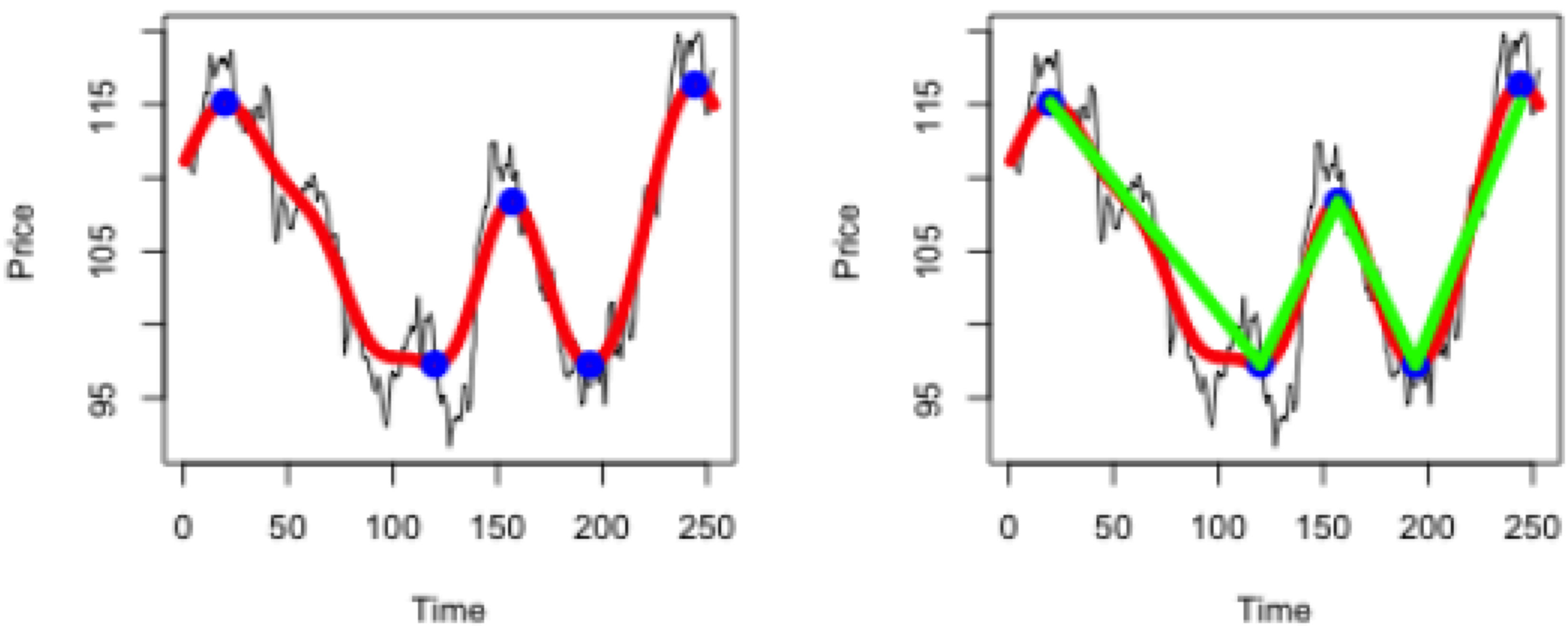
Example 2: Problem Statement

- There are large finance data available publically such as stock data.
- The challenge is how to teach a machine to auto recognize patterns in the stock chart.
- Then make predictions based on historical data and current economic environment and sensitivities and opinions of the investors.
- Finally select trading strategies automatically and make algorithm trading.

For pattern recognition , we first use Gaussian Process to fit the data.



With my undergraduate student, Kate Dover, we developed auto stock pattern recognition tools.



Example 3: Topic modeling

Expert Opinion and Coherence Based Topic Modeling

Natchanon Suaysom and **Weiqing Gu**
Mathematics Department, Harvey Mudd College
301 Platt Blvd, Claremont, California 91711

Abstract

In this paper, we propose a novel algorithm that rearrange the topic assignment results obtained from topic modeling algorithms, including NMF and LDA. The effectiveness of the algorithm is measured by how much the results conform to expert opinion, which is a data structure we defined to represent the probability that a pair of highly correlated words appear together. In order to make sure that the internal structure does not get changed too much from the rearrangement, coherence, which is a well known metric for measuring the effectiveness of topic modeling, is used to control the balance of the internal structure. The final algorithm which takes into account both coherence and expert opinion is presented. An algorithm for obtaining expert opinion from training data is also developed. Finally we compare amount of adjustments needed to be done for each topic modeling method, NMF and LDA.

Introduction

In usual topic modeling, such as Latent Dirichlet Allocation (LDA) and Nonnegative Matrix Factorization (NMF), the algorithm would consist of preprocessing data, coming

balance the initial topics obtained from NMF and LDA automatically. In order to do so, we need to create certain data structures which would fit the purpose of this task. Tree and Directed Acyclic Graph (DAG) have received community attention to be the objects that capture opinion on words, (Lu and Zhai 2008), (Wei and Gulla 2010) (Li and McCallum 2006). In this paper we also created new data structure which combines TREE and DAG, we call the new structure TDAG, the reason we created such a data structure is we modeled expert opinions in Tree or in DAG and sometime both so we can integrate them. By doing so, we gain advantages in algorithmic sense that allows us to optimize a cost function, which captures the similarity between expert opinion and topic model results through each height of a tree, and allows the graph to have multiple roots. The advantage of it will be explained in Topic Assignment section.

Assuming that the opinion on words are arbitrary, then we used the structure of TDAG to develop an algorithm that will make the clustering able to reflect expert opinion as well as possible, while still keeps the essential information on words that are not in expert opinion. For example, for NMF $||A - WH||$ is still small (so that it still makes a good clustering

More Examples:

Kat's thesis

Math 197: Senior Thesis

Geometry of Machine Learning

Kathryn Dover

Introduction

There are many well developed machine learning methods used in pattern recognition that are based on probability but these methods are highly dependent on the numeric values in the dataset. The goal of this research is to use a geometric definition patterns that is invariant under size and rotation. Well known patterns in stock market data were defined in a geometric way so an algorithm could use these definitions to identify patterns and predict them.

The Dataset

Patterns in stock market prices were used because stock data is easily available and there are already well-defined patterns. This project looked at the double-bottom pattern (W), the double-top pattern (M) and the head and shoulder pattern. Example of these patterns are shown in Figure 1.

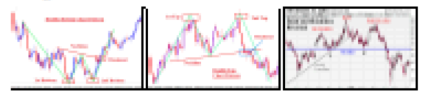


Figure 1: Examples of a double bottom, double top, and head and shoulder pattern (Investopedia, 2016).

Geometric Definitions

Given our dataset, let $\{Q_1, Q_2, Q_3\}$ denote local maxima and $\{P_1, P_2, P_3, P_4\}$ denote local minima of a certain subset of the data. Then we can define our patterns to be the vectors generated by these two sets, as seen in Figure 2.

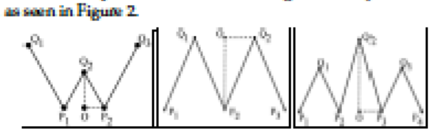


Figure 2: The vector representation for each shape.

The key to this geometric definition is finding a midpoint O and creating a basis with it. From this basis, the other vectors in the shape can be defined and compared to each other.

Fuzzy Shapes

We wanted to define a "fuzzy shape" that will be considered an approximate form of the standard shape. Let $\vec{v_1} \| \vec{v_2}$, $\vec{v_1} \perp \vec{v_2}$ imply that the vectors are almost parallel and almost perendicular, respectively. Thus, we can say that for a sequence to be a W, we would need the following criterion to be met:

1. $\overline{Q_1Q_3} \| \overline{P_1P_2}$
2. $\overline{OQ_2} \perp \overline{P_1P_2}$
3. $\overline{P_1Q_1} \| \overline{P_2Q_2}$
4. $\overline{P_1Q_2} \| \overline{P_2Q_3}$

Thus, we can define how far fuzzy shapes are from the standard shape by measuring how parallel or perpindicular they are to each other. Additionally, this check is not specific to the numeric values but on the vectors making it invariant under size and rotation.

Prediction

Let $M_\ell = \{\ell_1, \ell_2, \ell_3, \ell_4\}$ and $M_s = \{s_1, s_2, s_3, s_4\}$ be the consecutive lengths and slopes of a standard M. Then, for the ith M found in the dataset, let M_{ℓ_i} and M_{s_i} be defined similarly. Thus, we can find constants k_ℓ^i, k_s^i such that

$$(\ell_1, \ell_2, \ell_3, \ell_4) - k_\ell^i\left(\ell_1^i, \ell_2^i, \ell_3^i, \ell_4^i\right) = 0$$

$$(s_1, s_2, s_3, s_4) - k_s^i(s_1^i, s_2^i, s_3^i, s_4^i) = 0$$

If we save every k_ℓ^i and k_s^i along with the slope and length of the segment that follows each shape, we can make the following average for a rough prediction:

$$\ell' = \frac{k_\ell^1 \ell_5^1 + k_\ell^2 \ell_5^2 + \dots + k_\ell^n \ell_5^n}{n}$$

$$s' = \frac{k_s^1 s_5^1 + k_s^2 s_5^2 + \dots + k_s^n s_5^n}{n}$$

Implementing the Algorithm

In order to process the data and find the shapes using the geometric definitions, my algorithm goes through the following steps (the illustration is shown in Figure 3):

1. A Gaussian Process fits the data and the local extrema are identified.
2. The algorithm identifies which sequences of extrema satisfy the criterion for each shape.
3. When a shape is found, the categorizing constants k_ℓ^i and k_s^i are calculated and stored.

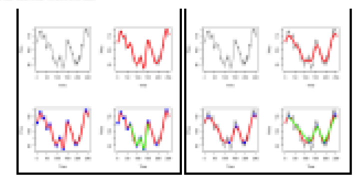


Figure 3: This shows how the data is fit and the shape is found using local extrema.

Results

The algorithm was run for variances 0.001 and 4 over the entire stock price histor for the companies Apple, Disney, Microsoft, Nike, Walmart and the NASDAQ index. From Figure 3, we can see that the algorithm was able to find the same shape. When I did algorithm Ws, I got that the expected slope and length after a W was -0.17109 and 15.54218. When I found Ws for the SP500 data, I checked the expected slopes and lengths based off of the constants I found in the previous data. Compared with the actual slopes and lengths in the data, the range in errors I got was (0.02, 3.52) for slopes, (5.04, 107.83) for lengths. Therefore, this averaging techinique is a decent indicator for what direction the stock moves after the shape, but not for how long it goes.

References

Investopedia, Staff. 2016. Investopedia. Available online at investopedia.com. URL http://www.investopedia.com/terms/d/doublebottom.asp.

Acknowledgments

I would like to thank my advisor, Weiqing Gu, for helping me through this project and Dagan Karp, for begin my second reader.

Advisor: Professor Weiqing Gu
Reader: Professor Dagan Karp

Example 4. Using the manifold of covariance matrices

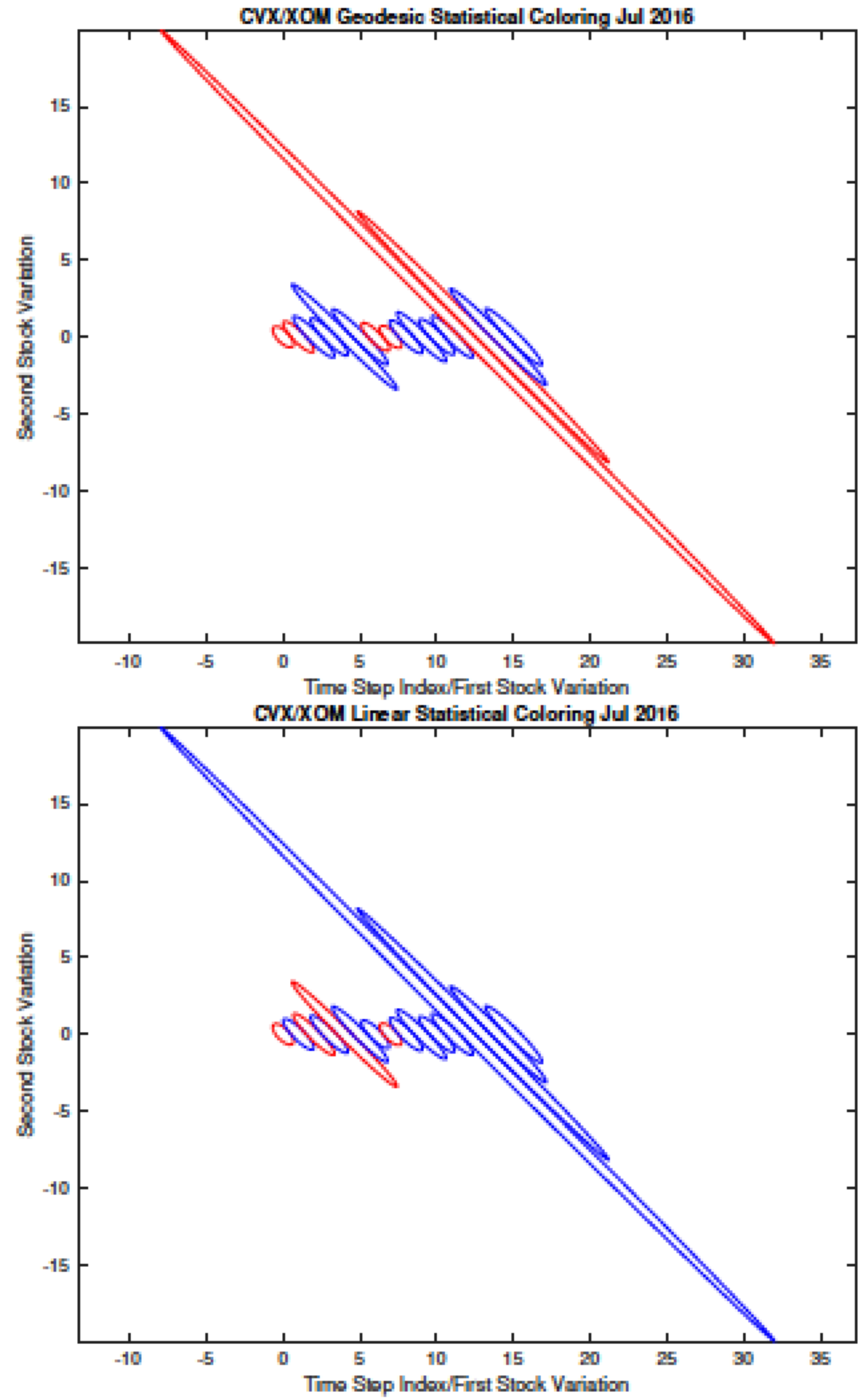
Anomaly Detection and Feature Extraction via the Manifold of Correlation Matrices

Paul David* Weiqing Gu†

Abstract

Correlation matrices are essential for performing statistical analyses on large sets of multidimensional data as well as for making predictions on unclassified data. Correlations can provide significant information regarding the relationship between multiple variables, but it has not been investigated until now the true relationship from one such matrix to another, especially for dynamical data. We present here for the first time a proof that the set of non-degenerate correlation matrices is a manifold embedded in the manifold of symmetric positive-definite matrices. This proof is obtained via a group action and shows how a natural Riemannian metric, and hence distances, can be obtained on this manifold. With this structure, we are able to use the technique of geodesic gradient descent for optimizing mean-squared distances between positive-definite correlation matrices of size n. While correlations have been studied in a time-invariant context such as [1, 2], understanding time-dependent sequences of correlations can aid in optimization and prediction of time-dependent random variables. In the context of machine learning, symmetric positive-definite matrices $SPD(n)$ as well as correlation matrices $Corr(n)$ have been shown to be important in many research fields. A few of these areas of application include diffusion tensor imaging [3–5], statistics for modeling Gaussian distributions [1, 6], and their role in classification of data sets that occur on non-linear spaces [1, 7–9]. Of fundamental importance to the aforementioned research is to find efficient ways averaging and optimizing $SPD(n)$-valued data, and to do so in ways that are reflective of the inherent manifold structure of $SPD(n)$.

Proved the set of correlation matrices forms a manifold. Then developed new techniques to do big data analytics on this new manifold including geodesic distance.



My thesis student, Casey Chu's work : developing a new metric and distance on data manifold.

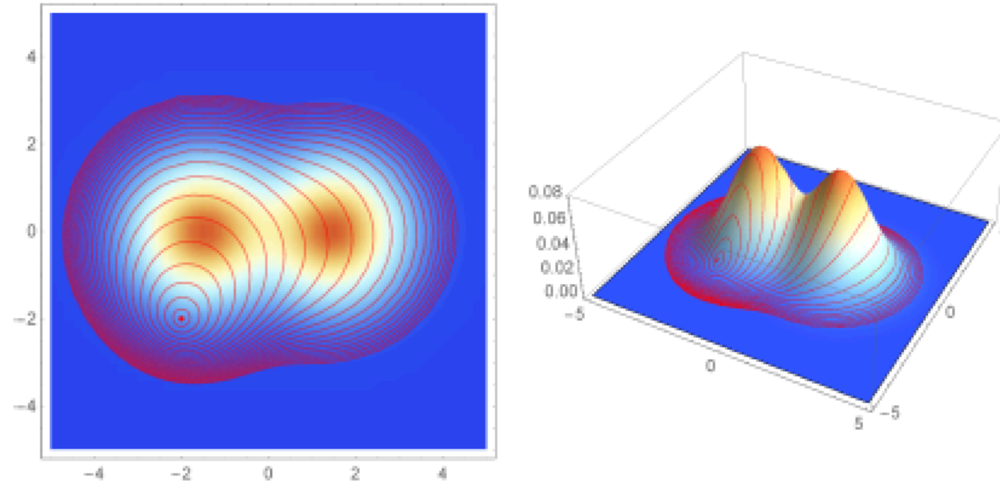


Figure 7.4 The contours represent points equidistant from $(-2, -2)$, under the scaled Euclidean metric ($\alpha = 1$) with a mixture of two normal distributions centered at $(\frac{3}{2}, 0)$ and $(-\frac{3}{2}, 0)$.

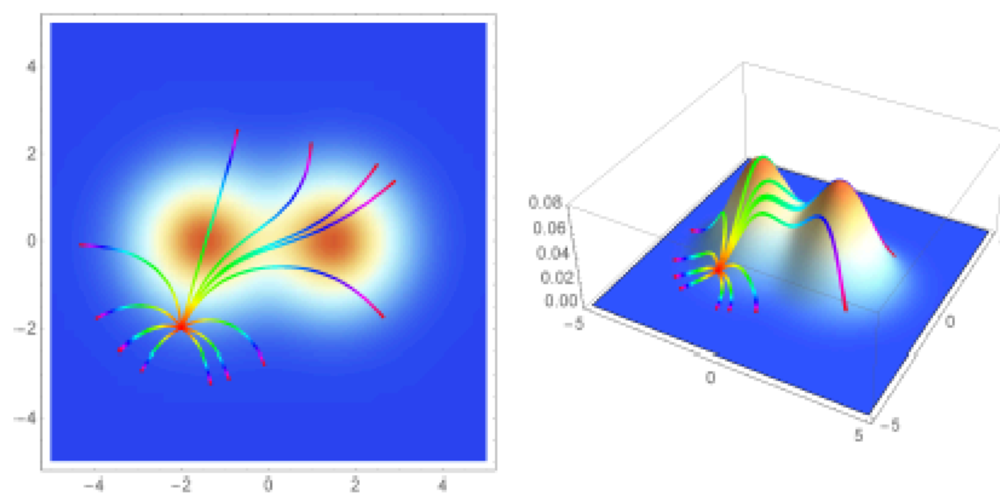


Figure 7.5 A selection of geodesics of the same length from $(-2, -2)$.

Hypergradient Descent Almost Converges

Conner DiPaolo* Weiqing Gu†

Harvey Mudd College
Claremont, CA, USA

Working in progress

Abstract

Recently, Baydin et al. [2] proposed the Hypergradient Descent (HGRAD) optimization algorithm to adaptively update the learning rate of traditional first-order optimization methods. This algorithm does not converge in the traditional sense, but under stronger assumptions we show HGRAD converges linearly for strongly convex objective functions. Specifically, the set of valid (α_0, β) pairs depends not only on the strong convexity constants but also on how close the initial $x^{(0)}$ is to the optimal x^*. In addition, our analysis shows how the sequence of learning rates $\alpha^{(t)}$ reveals information about the geometry of the objective surface simply as a byproduct of running the optimization algorithm. This is investigated empirically, and provides an interesting attack direction for looking at the geometry of neural network loss surfaces.

1 Introduction

Optimizers have provided efficient and correct methods for optimizing convex functions (for a comprehensive overview, see Boyd and Vandenberghe [4]). For unconstrained convex problems with reasonable size or strong structure, second order methods such as Newton's Method [4, Sec. 9.5] guarantee solutions in a couple dozen fast iterations. That said, many important *convex* problems in machine learning including regular logistic regression often have enough parameters to render an inverse Hessian computation intractable or undesirable. Even if we could solve these large and generic linear systems efficiently, the increased prevalence of *non-convex* problems such as deep learning breaks down the theoretical and empirical backbone of many fast second order methods. As such, machine learners are often forced to turn to necessarily slower first order methods such as Stochastic Gradient Descent [3], RMSProp [9], Ada-Delta [10], Adam [7] that have become increasingly tuned in an attempt to approach the performance of second order methods. All of these first order methods have some analog to the step size parameter α of the regular gradient descent update

$$x^{(t+1)} = x^{(t)} - \alpha \nabla f(x^{(t)}). \tag{1}$$

This parameter may change on a preset schedule or be fixed, but in general they are hard to pick and rely on past experience for best practices.

My thesis student Bo Zhang’s work

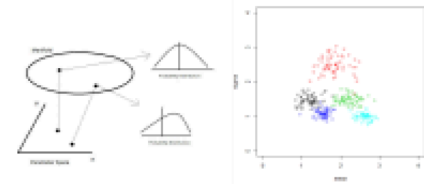
Math 197: Senior Thesis

Machine Learning on Statistical Manifold

Bo Zhang

Introduction

This senior thesis project generalizes some fundamental machine learning algorithms from the Euclidean space to the statistical manifold, an abstract space in which each point is a probability distribution. In this thesis, we adapt the *Support Vector Machine*, the *K-Means Clustering*, and the *Hierarchical Clustering Methods* to classifying and clustering probability distributions. we use various *statistical distances* as measures of the dissimilarity between probability distributions.



(a) Statistical Manifold. (b) Parameter Space Normal Distribution

Figure 1

Methods

Classification

We develop an analogy of the *optimal separating hyperplane* algorithm on the statistical manifold, in order to classify probability distributions. Let the training data be a sequence of distributions parametrized by $x_1, x_2, \ldots, x_n$ and a sequence of labels $y_i \in \{+1, -1\}$. Let $\mathrm{DF}(p_1, p_2)$ denote the statistical distance between two probability densities p_1 and p_2. We formulate the classification problem as an optimization problem below:

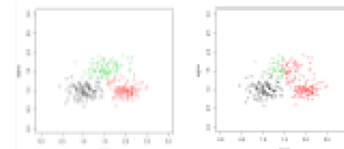
$$\begin{aligned}
&\max_{\theta_1,\theta_2,\beta_0,\beta^T} D \\
&D_i \geq D \\
&D_i = \min_t \mathrm{DF}(p(x;x_i), p(x;\theta(t))) \quad \textit{for each } i \\
&p(x;\theta(t)) = p(x;(1-t)\theta_1 + t\theta_2) \\
&y_i(\beta_0 + \beta^T x_i) \geq 0 \quad \textit{for each } i \\
&\beta_0 + \beta^T\theta_1 = 0 \\
&\beta_0 + \beta^T\theta_2 = 0
\end{aligned} \tag{1}$$

Clustering

We first generate distributions on the statistical manifold for the purpose of clustering. Below, we use the univariate normal distribution as an example to illustrate this generating process.

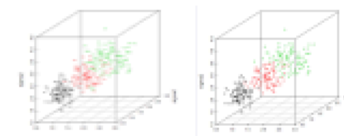
1. Select k pairs of parameters (μ_k, σ_k).
2. For each k, do the following t times:
   (a) Generate n samples from the univariate normal distribution $f(x; \mu_k, \sigma_k)$.
   (b) Reconstruct $f(x; \mu_k, \sigma_k)$ from these n samples and obtain unbiased estimates $(\hat{\mu}_k, \hat{\sigma}_k)$.

We apply modified versions of the Hierarchical Clustering Method and the K-Means Clustering method to clustering probability distributions. We compare the Euclidean-distance-based methods against the statistical-distance-based methods. Figure 2 illustrates the difference between the same Hierarchical Clustering Algorithm applied on the univariate normal distributions with three clusters using different metrics.



(a) Fisher-Rao distance (b) Euclidean distance

Figure 2



(a) Fisher-Rao distance (b) Euclidean distance

Figure 3

Results

We apply the clustering algorithms on simulated clusters and record the empirical results, as shown below.

Table 1: Univariate Normal Distribution Results: clustering when k = 3

Algorithm	Clustering Accuracy	
Hierarchical Clustering with Fisher-Rao Metric	0.905	0.005
Hierarchical Clustering with Euclidean Metric	0.858	0.007
K-Means Clustering with Fisher-Rao Metric	0.961	0.001
K-Means Clustering with Euclidean Metric	0.940	0.001

Table 2: Bivariate Normal Distribution Results: clustering when k = 3

Algorithm	Clustering Accuracy	
Hierarchical Clustering with Fisher-Rao Metric	0.860	0.008
Hierarchical Clustering with Euclidean Metric	0.716	0.012
K-Means Clustering with Fisher-Rao Metric	0.937	0.001
K-Means Clustering with Euclidean Metric	0.877	0.003

For Further Information

- Please feel free to contact me at bzhang@g.hmc.edu.
- You can download the full report and the poster at http://www.math.hmc.edu/~bzhang/thesis/.

Acknowledgments

I want to express my appreciation to the Department of Mathematics at Harvey Mudd College, my advisor Prof. Weiqing Gu, and reader Prof. Nicholas Pippenger for their generous help and constructive advice.

Advisor: Weiqing Gu
Reader: Nicholas Pippenger

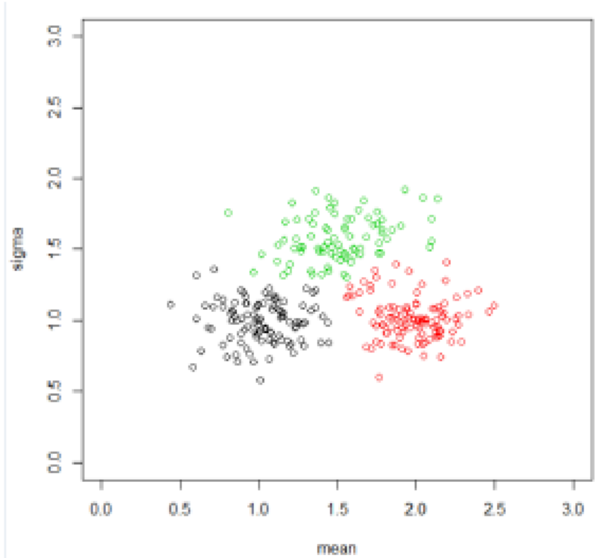
Methods

Classification

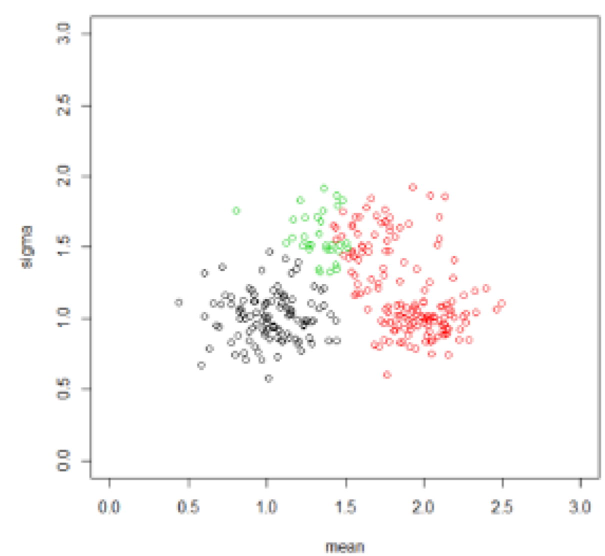
We develop an analogy of the *optimal separating hyperplane* algorithm on the statistical manifold, in order to classify probability distributions. Let the training data be a sequence of distributions parametrized by $\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}$ and a sequence of labels $y_i \in \{+1, -1\}$. Let $\mathrm{DF}(p_1, p_2)$ denote the statistical distance between two probability densities p_1 and p_2. We formulate the classification problem as an optimization problem below:

$$
\begin{aligned}
&\max_{\boldsymbol{\theta_1}, \boldsymbol{\theta_2}, \beta_0, \boldsymbol{\beta}^T} D \\
&D_i \geq D \\
&D_i = \min_t \mathrm{DF}(p(x; \mathbf{x_i}), p(x; \boldsymbol{\theta}(t))) \quad \textit{for each } i \\
&p(x; \boldsymbol{\theta}(t)) = p(x; (1-t)\boldsymbol{\theta_1} + t\boldsymbol{\theta_2}) \\
&y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{x_i}) \geq 0 \quad \textit{for each } i \\
&\beta_0 + \boldsymbol{\beta}^T \boldsymbol{\theta_1} = 0 \\
&\beta_0 + \boldsymbol{\beta}^T \boldsymbol{\theta_2} = 0
\end{aligned}
\tag{1}
$$

We apply modified versions of the Hierarchical Clustering Method and the K-Means Clustering method to clustering probability distributions. We compare the Euclidean-distance-based methods against the statistical-distance-based methods. Figure 2 illustrates the difference between the same Hierarchical Clustering Algorithm applied on the univariate normal distributions with three clusters using different metrics.



(a) Fisher-Rao distance.



(b) Euclidean distance.

Figure 2

My thesis student Cynthia Yan's work

Mathematics of Emergent Gravity Based on Quantum Entanglement

Cynthia Yan
(Advisor: Weiqing Gu)
Harvey Mudd College

October 6, 2017

Everywhere in this world, there are needs of data to decisions.

We apply our big data analytics to the problems everywhere from local community to international companies.

Keeping Green Grass Green Initiative

Weiqing Gu and Zhaocheng Yi

Introduction and Background

Conserving water has always been very important for California especially during drought seasons. The goal of this project is to create automatic recommendation irrigation rates for the purpose of minimizing the water but keeping the grass still green, including rate equals to zero meaning no irrigation is required. In this project, we took the lawns of Harvey Mudd College as an example to develop our algorithm for smart irrigation. We used big data analytics to develop a mathematical modeling and algorithm. Our algorithm is scalable and can be extended to different lawns whether owned by an individual or an organization as long as there is historical data available.

There are six meter-stations for the whole Harvey Mudd College campus. And there are four flow sensors in the main lawn area monitoring how much the water flow each day. Different colors show points of connections with numbers of meter stations in Figure 1. Figure 2 shows the irrigation management site of the Rain Bird Software for Landscape coefficient.

Figure 1: Left - Harvey Mudd College Campus Meter Stations

Figure 2: Right - Rain Bird Management Site

Methods

1. Obtained data from HMC Facility and Management Department
2. Processed data by cleaning and organizing them
3. Found the patterns of the irrigation based on historical data
4. Mathematically model relations between irrigation patterns and the weather data
5. Developed algorithms by taking historical data and weather data from online for the next five days to predict the amount needed to irrigate for the next day.
6. Finally, we designed an APP to auto notify the owner or lawn manager the exact amount of water for irrigation.

Results

Evapotranspiration calculates the amount of water needed for grass with a combination of plant factor and weather variables. It is one of the most important factors to determine the rates of irrigation. We analyzed the relationship between ET and Temperature and found that ET and Temperature are closely related, which can be demonstrated in Figure 3.

The amount of Maximum Irrigation is determined both by the schedule pattern and relationship with the temperature. We analyzed the irrigation data by their schedule pattern in each month. By dividing irrigation into different months, we potentially broke down how temperature influences irrigation amount and frequency. We found that irrigation models depend on seasons and irrigation amount patterns correlated with temperature. The irrigation pattern of May 2017 is shown in Figure 4 as an example. The purple dots on the x-axis indicate that no irrigation amount and it also rained that day. The blue bars mean irrigation with a certain amount and it also rained that day.

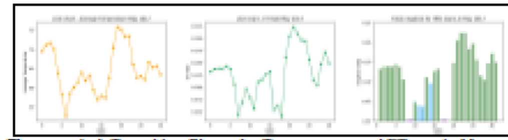


Figure 3: Left Two - Line Charts for Temperature and ET rate in May 2017

Figure 4: Right - Total Irrigation Pattern for HMC lawns in May 2017

The APP we designed is a Programmable Short Message Services (SMS) from Twilio, a communication platform that generates an Application Programming Interface (API) to exchange text and picture messages. The example message is shown in Figure 5. Gabriela Gamiz

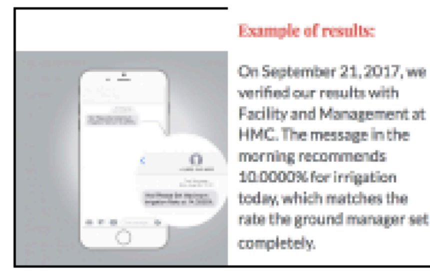


Figure 5: Sample Product Result

Conclusions

The advantage of our method is independent of sensors, ground pipelines, and wires, no matter if they are damaged by the ground animals. We can adjust our irrigation schedule according to special events managed by Harvey Mudd College. Later then, we can move ahead without wires for the sensors to avoid the trouble caused by ground animals. More importantly, in this project, we have developed techniques to optimize smart irrigation using Harvey Mudd College lawns as examples. Our techniques can be extended to different lawns regardless they are owned by individual homeowners or by organizations as long as there is historical irrigation data available. In addition, we are also going to use this example as a start point to integrate with Mudd students coursework to help the communities to make various optimal decisions based on the historical data available to produce predictive results for making such decisions.

Acknowledgments

Special thanks to Herrick Fang for setting up the APP, the deliverable of this project. Further thanks to Harvey Mudd College Facility and Management Department for the data. Additionaly, we would like to

For example, we also extend our data analytics to community services and helping people in poor neighborhoods

Minimizing the cost and maximizing the effectiveness of mission organization for food allocation to marginalizing communities

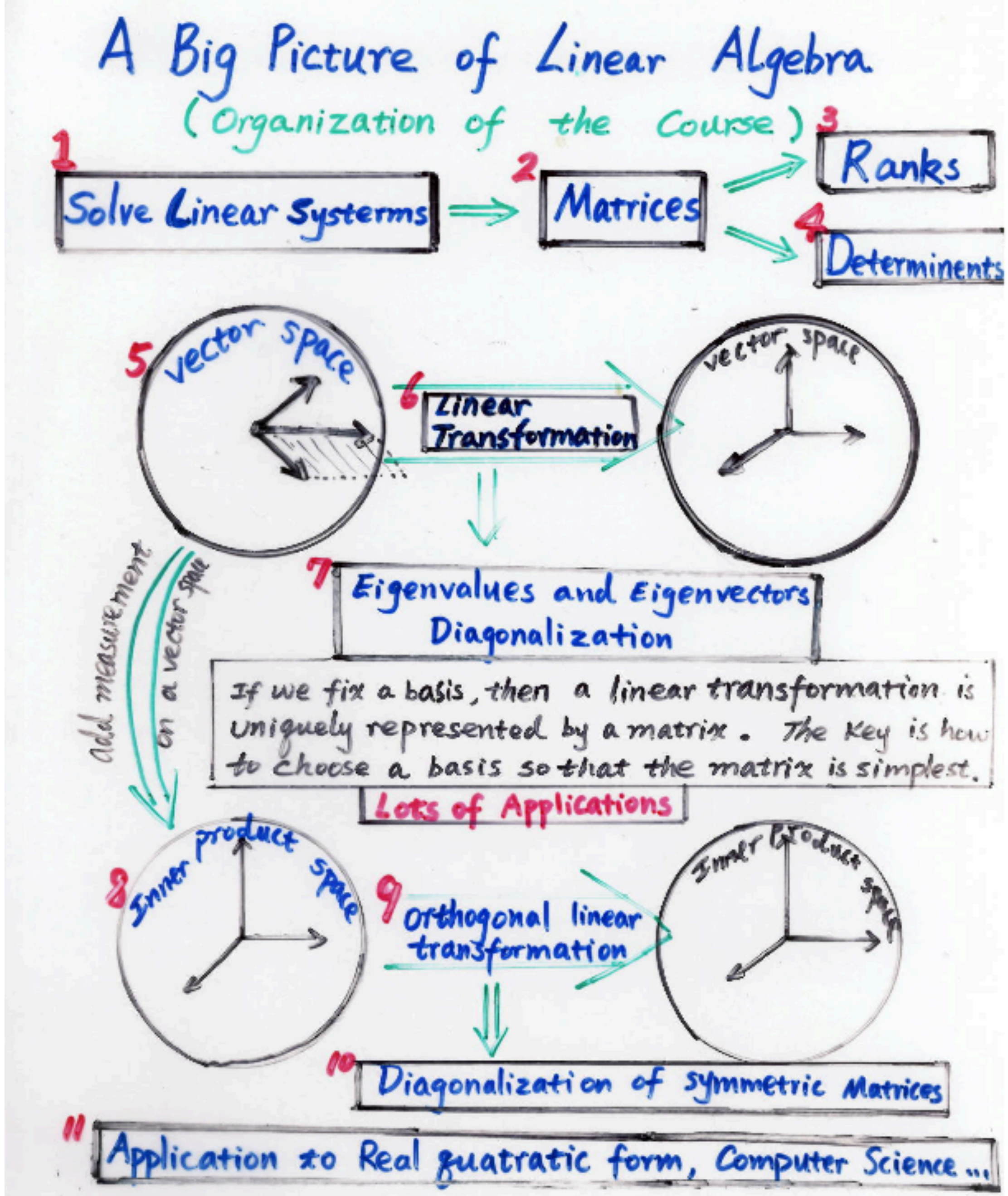
Matthew Simon
Advisor: Prof. Weiqing Gu
Oct 8, 2017

- Recall: We turned key properties in $\mathbf{R}^n$ to define abstract vector space.
- We will turned the key properties for regular surface in $\mathbf{R}^3$ to define abstract manifold.
- Later we will turned the key properties of a tangent bundle to define abstract vector bundle.

You must be very familiar with linear algebra so that be able to mimic its key ideas!

From Regular Surface to Abstract Manifold:

Mimic the linear algebra philosophy



Thank you!