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***

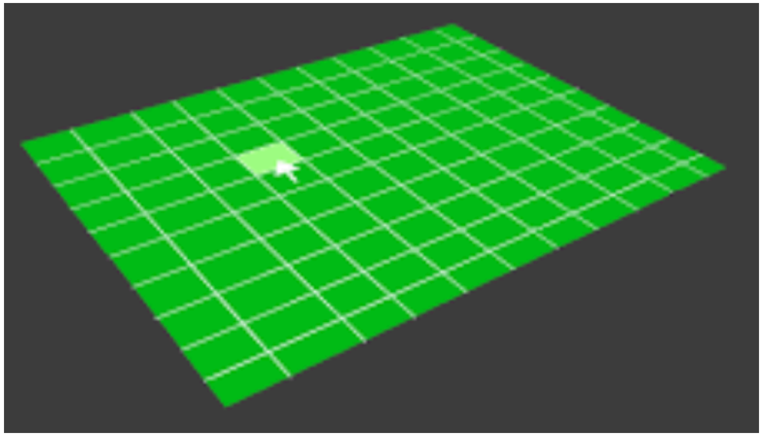
Attendence for each lecture is mandatory and is expected of all class members. if you're going to miss a lecture, it is neccessary for you to inform the instructor as soon as possible. You are also responsible for obtaining notes from another class member.

COURSE DESCRIPTION

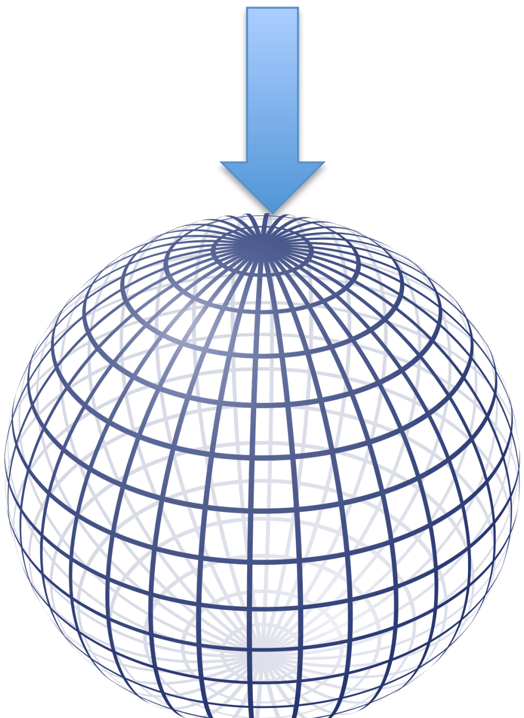
The objective of this course is to familiar with the students with the basic language and techniques of Riemannian Geometry and their new applications in big data analytics in addition to their classical applications in theoretical physics. The following are the possible topics in the courses. The order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



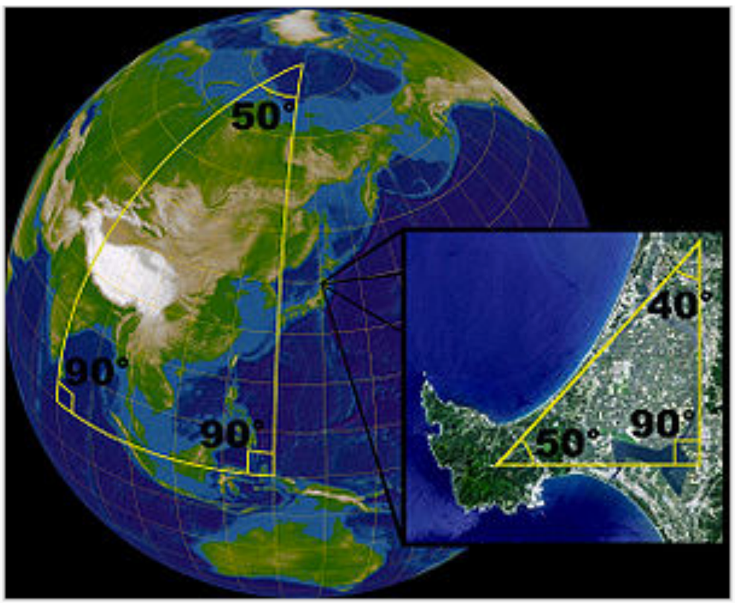
Vector Space:
Linear algebra



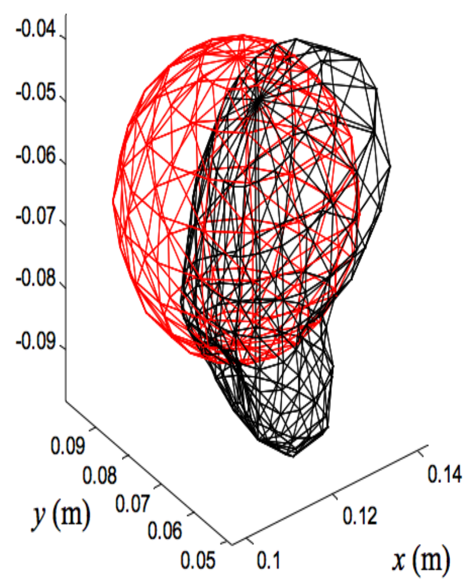
Manifold:
Differential Geometry

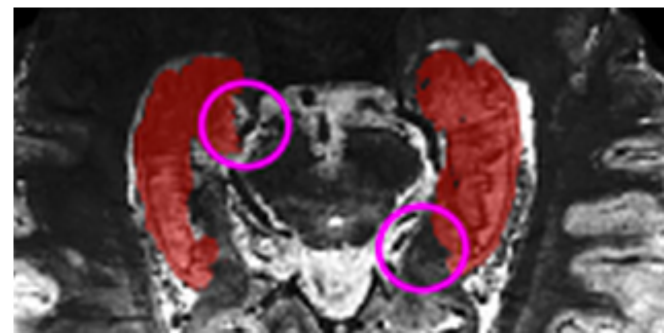
What is a manifold?

- 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

1.

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.

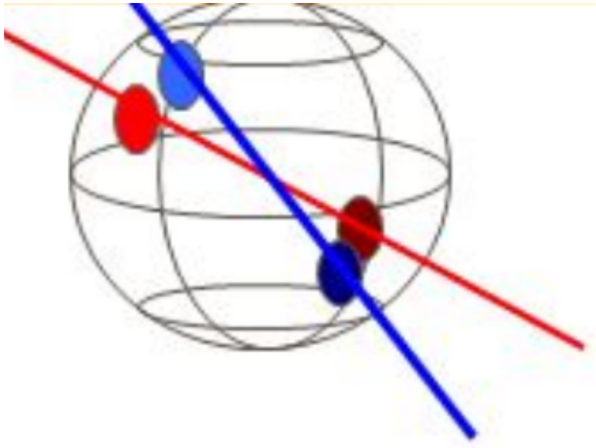
Using Key proposition of a regular surface to define manifolds

In this course mfld $=$ Differentiable mfld

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

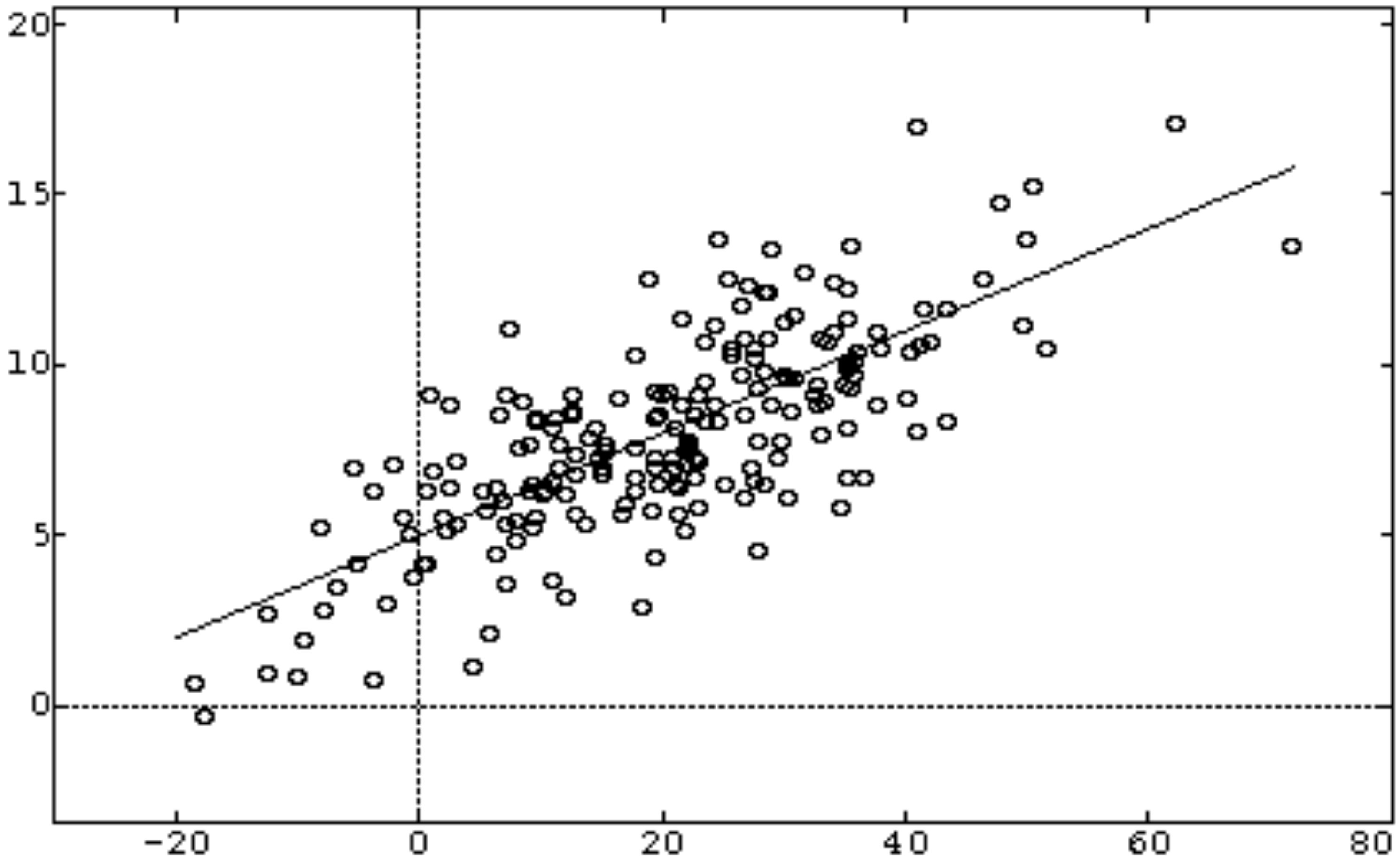
- **Example:** Consider the set of all possible oriented planes through origin.

$G^1R_3 = G^2R_3 = RP^2$

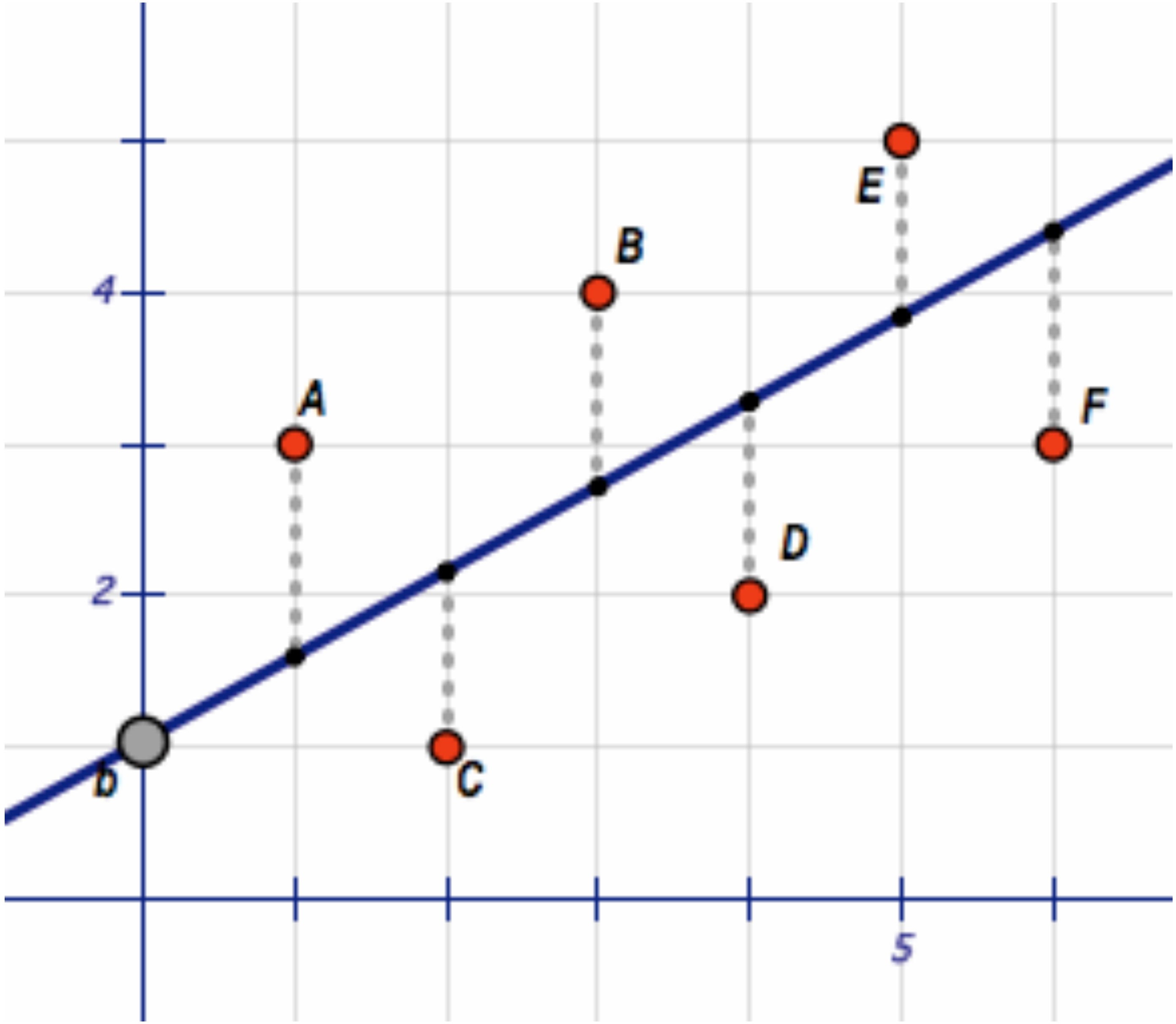


Recall: Linear Regression

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 (G_2R^3)

Grassmannians and their applications

- Instead of considering the set of all vectors through origin, we consider set of all k-planes through the origin in R^n. $\rightarrow$
- Grassmann manifolds $\mathbf{G_k R^n}$ (or Grassmannian).
- *So what?*
- *Example: Study big video data*

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

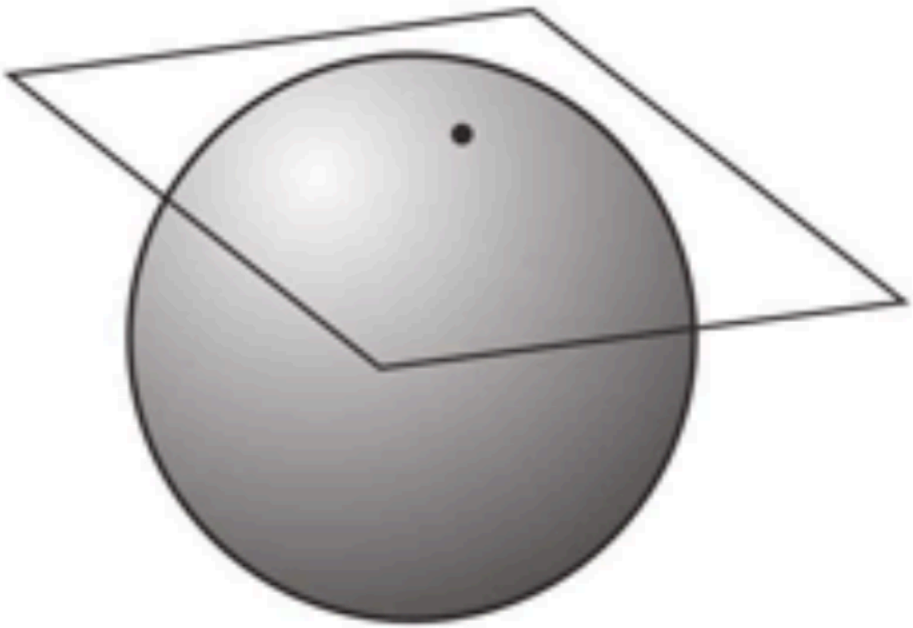
- https://sites.google.com/site/hejunzz/grasta

Examples of manifolds

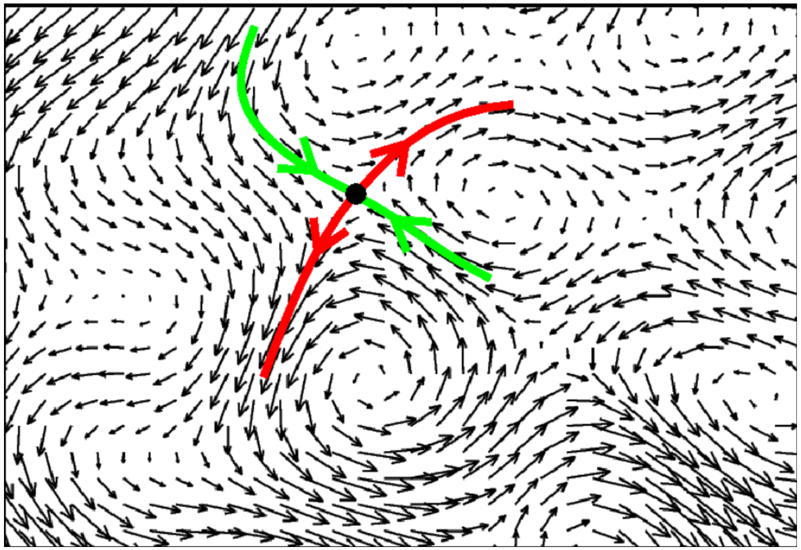
- The set of all rigid motions forms a manifold (in fact it is a Lie group).
- The set of all probability distributions forms a manifold. (called the statistical manifold).
- The set of Gaussian distributions forms a submanifold of the above.
- The set of all time series forms a manifold.
- The set of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

How to study (curved) manifolds?

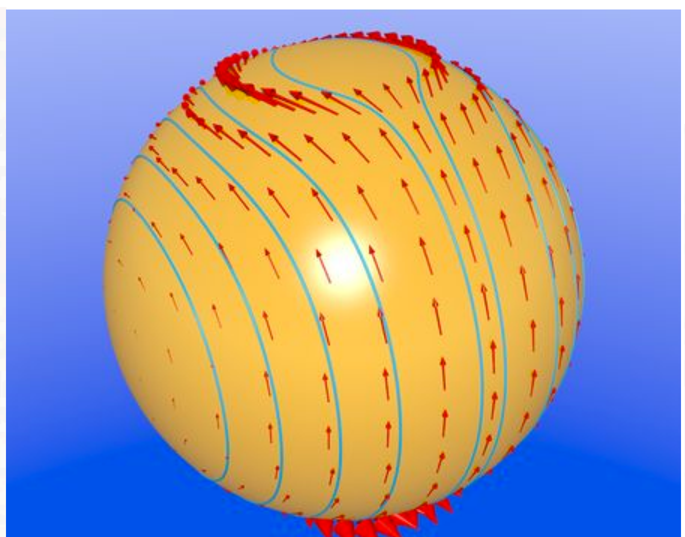
- Use tangent plane first
- Example: Tangent space to SO(3). More details Later.

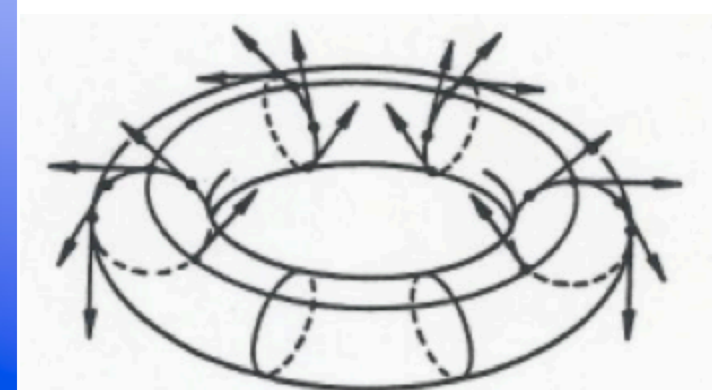


Then Use Vector Fields

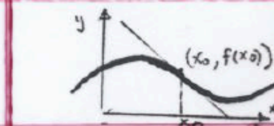
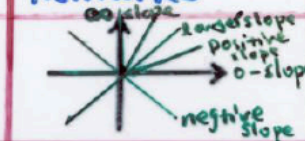
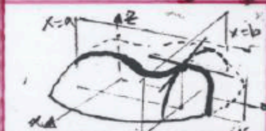
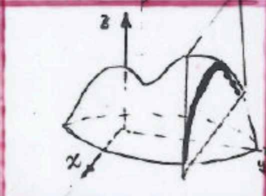
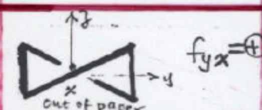
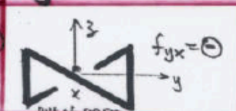
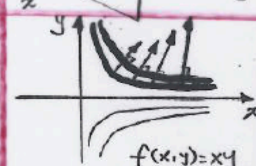
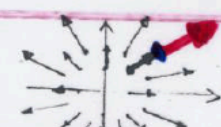
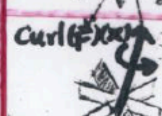


Extend the multi-V
to manifolds





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: $\boxed{f_{xy}=f_{yx}}$
$f:\mathbb{R}^n\to\mathbb{R}$	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 $\nabla f(\vec a)$		$\frac{\nabla f(a,b)}{\|\nabla f(a,b)\|}=\vec u$, 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 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 $\frac{1}{2}\|\mathrm{curl}\,\vec F(\vec x)\|$ radians/sec.		$\mathrm{Curl}(\vec F)=\nabla\times\vec F=\begin{vmatrix}\vec i & \vec j & \vec k\\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z}\\ F_1 & F_2 & F_3\end{vmatrix}$
$\vec x:\mathbb{R}\to\mathbb{R}^n$	Tangent to parametrized curve	$\dot{\vec x}(t)$		Velocity vector	$\dot{\vec x}(t)=(\dot x_1(t),\dots,\dot x_n(t))$	

Derivatives on Manifolds

- All kinds of derivatives in Multi-V

Extend
to manifolds

- Connections and Covariant Derivatives

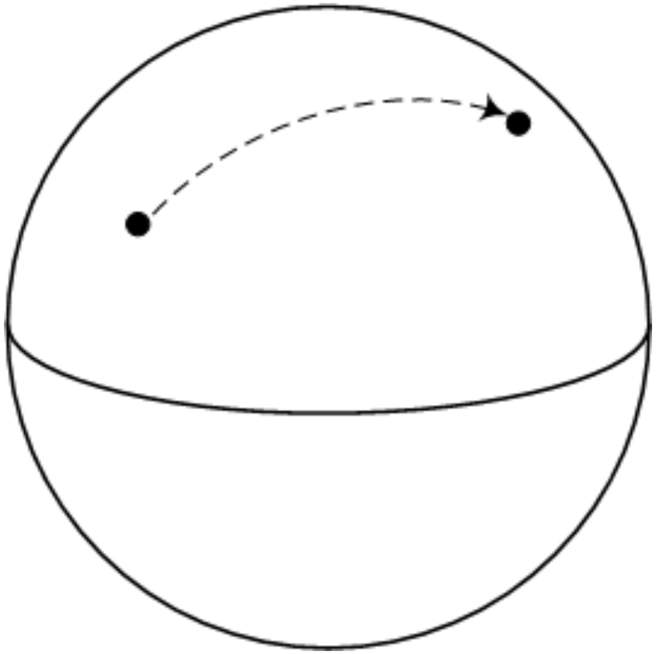
Important: Riemannian Metric (Not Euclidean metric!)

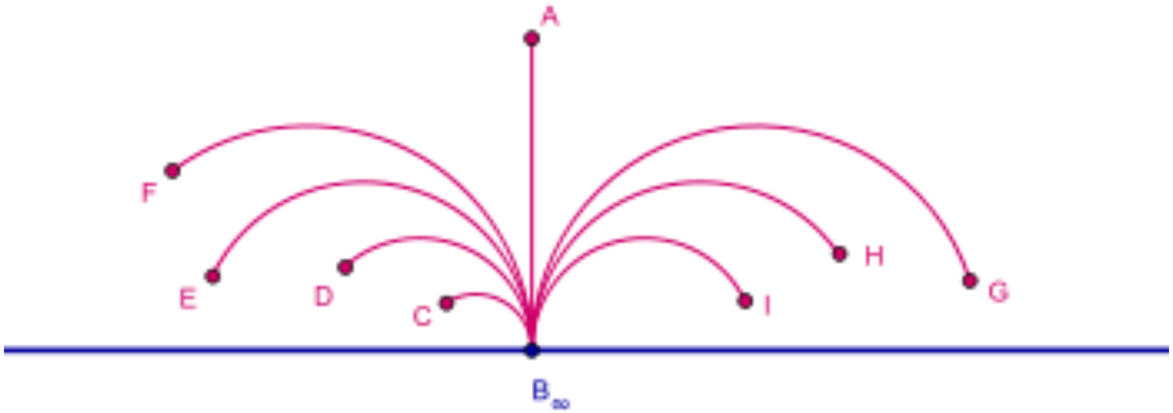
- Vector space + inner product = Euclidean space

Extend to manifolds

- Manifold + Riemannian metric = Riemannian manifold

What are Riemannian Measurements?





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

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. {All Rigid Motions} = manifold. Each element is a frame!
- 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

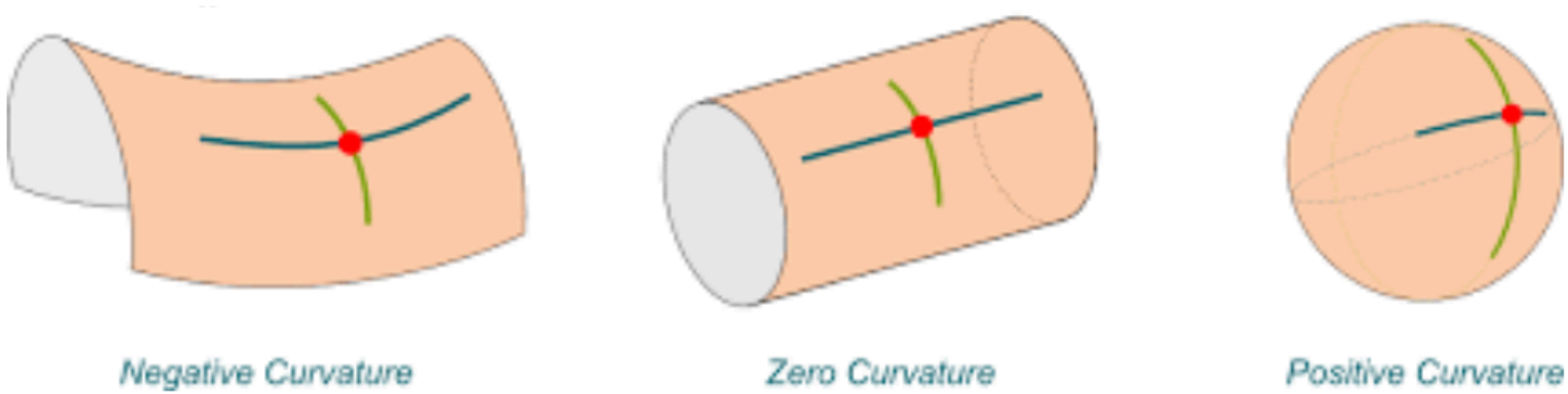
- All kinds of integration in Multi-V

Extend
to manifolds

- Integrations on manifolds

Curvatures and applications

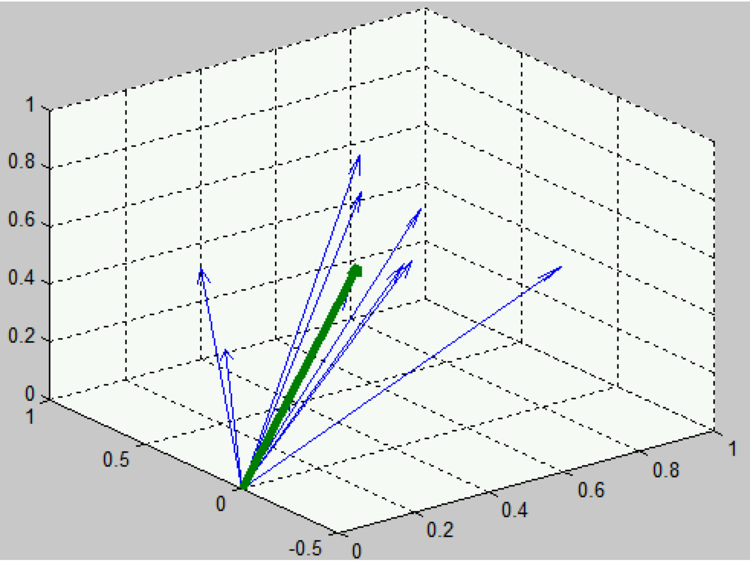
- As Key data features
- Face recognition
- Model various problems using curvature including network congestion.
- Many many more…



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?

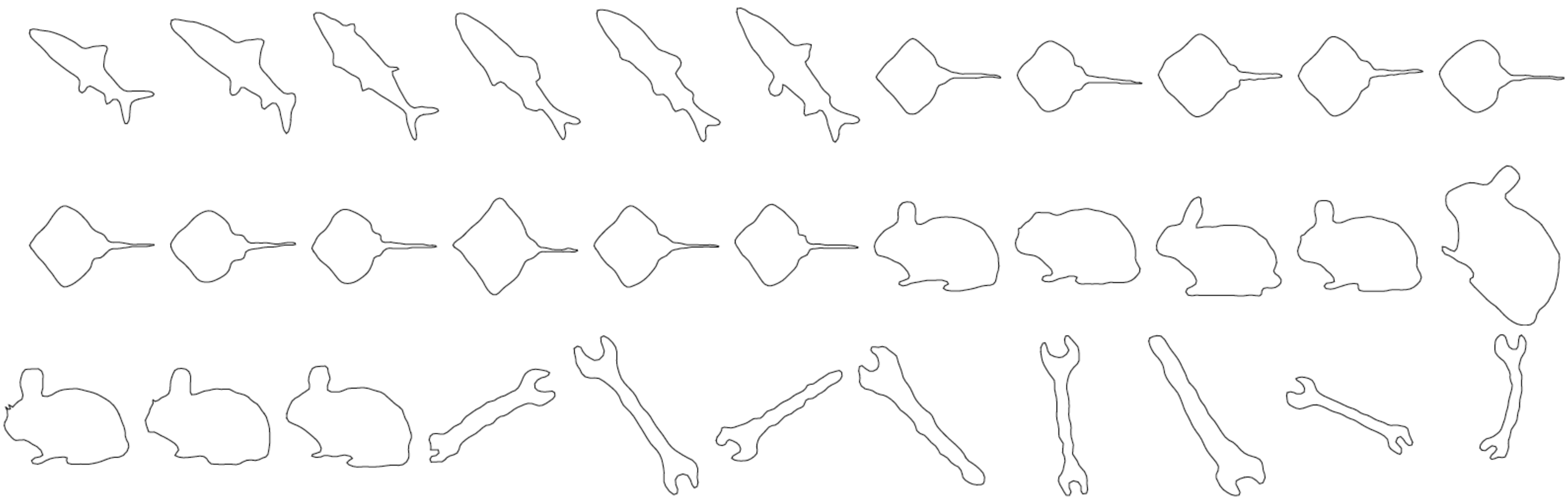
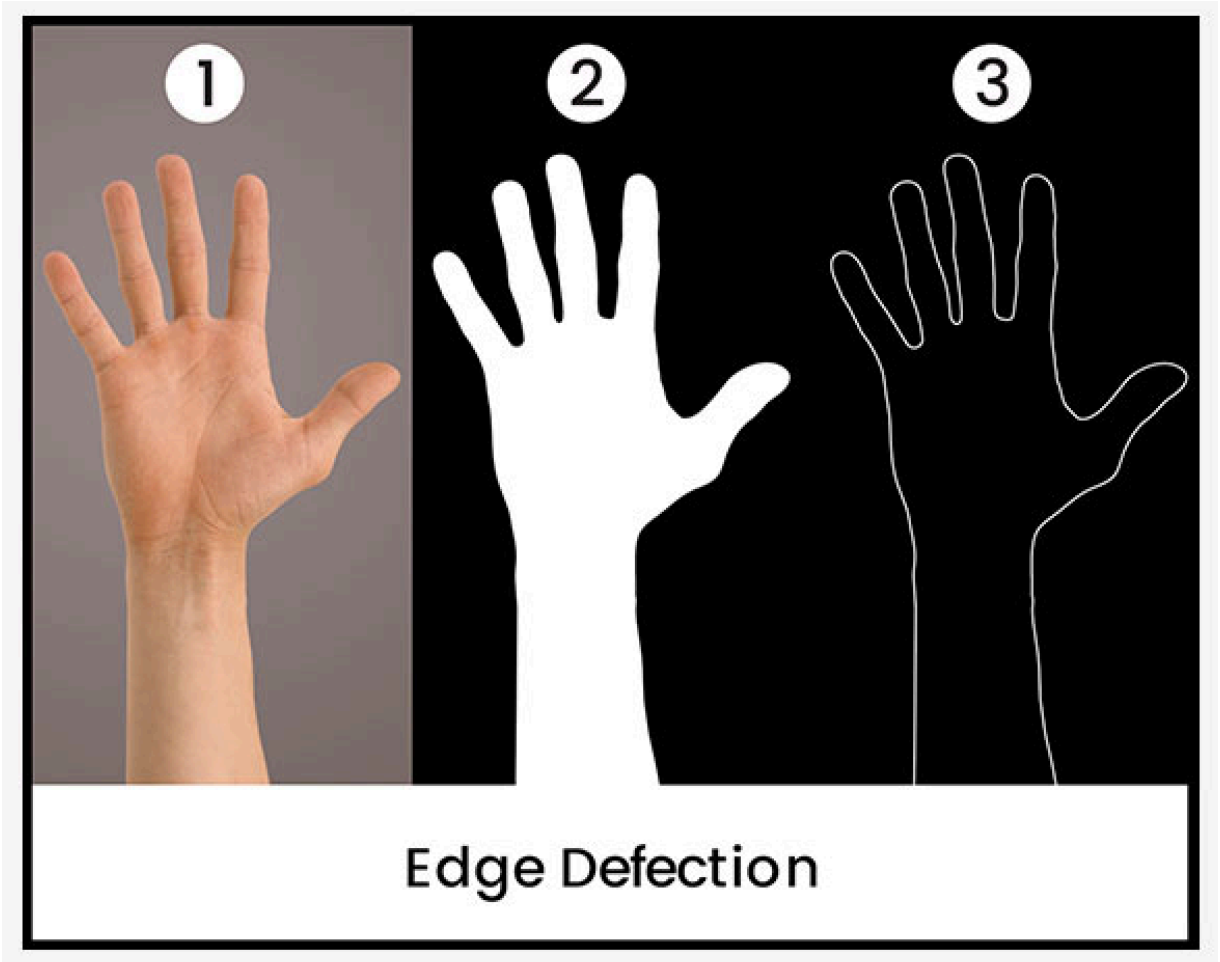


Figure 1: Selection of shapes from the dataset [14].

[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!



The set of all curve shapes form a manifold!

Shortest "path" between given two shapes

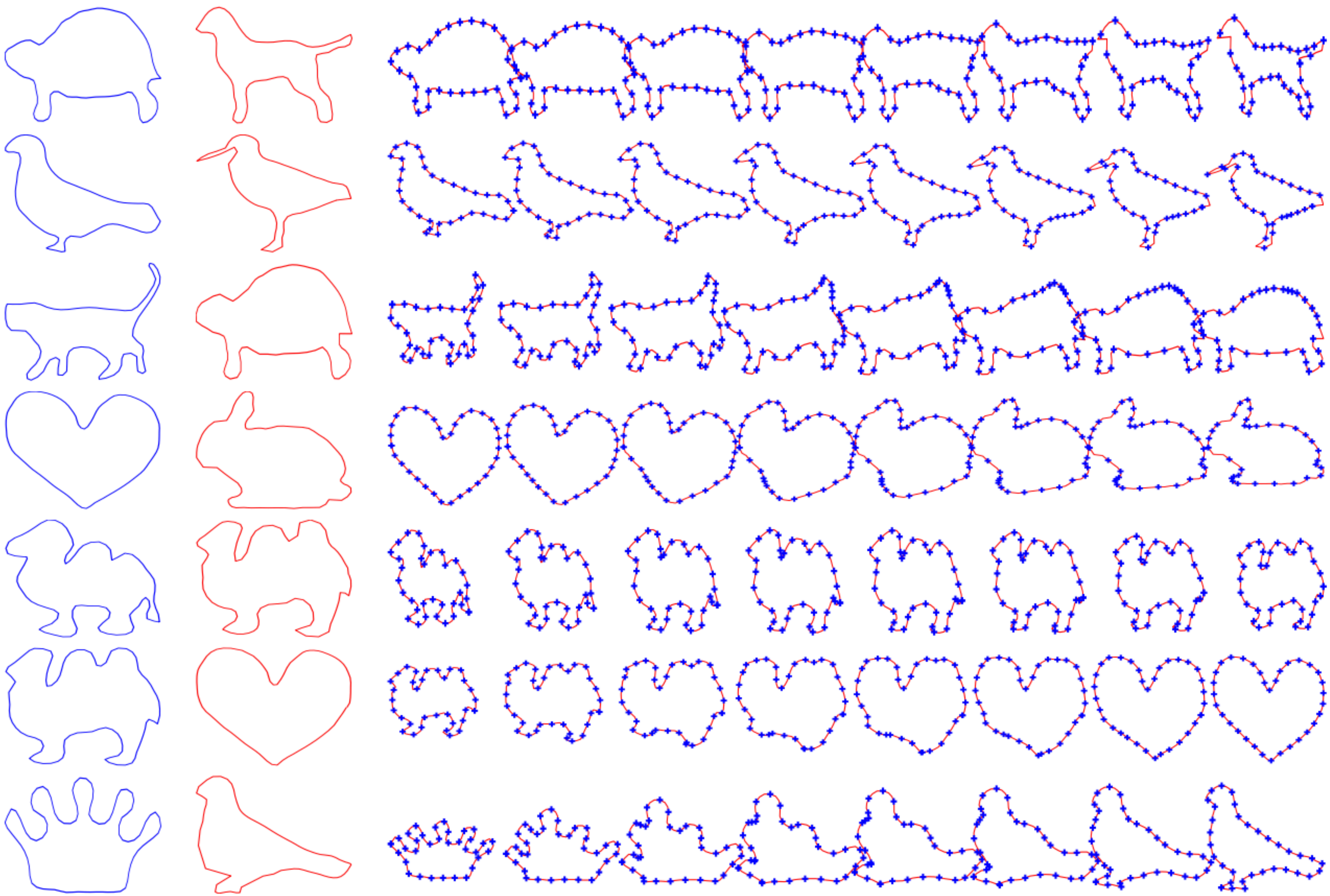


Figure 2. Row-wise geodesic paths in $\mathcal{C}$ between the pair of curves shown to the left.

We also perform "clustering" on the manifold of the curve shapes.

Key: Need an appropriate distance function

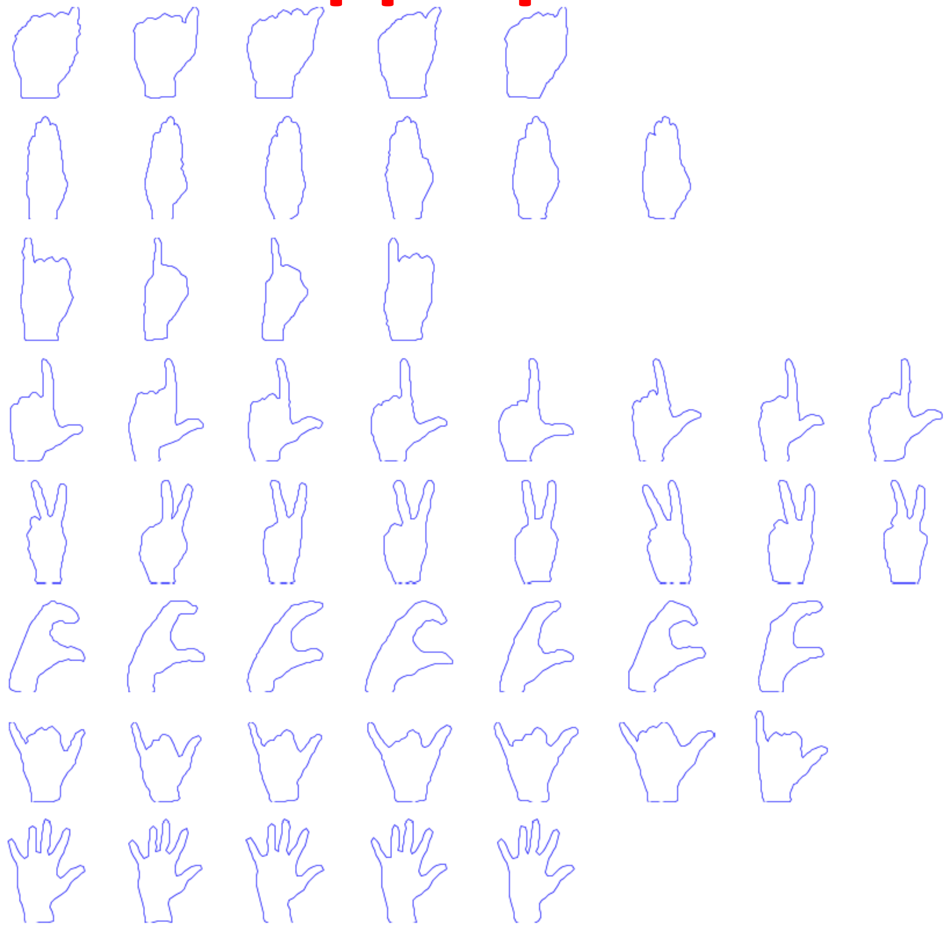
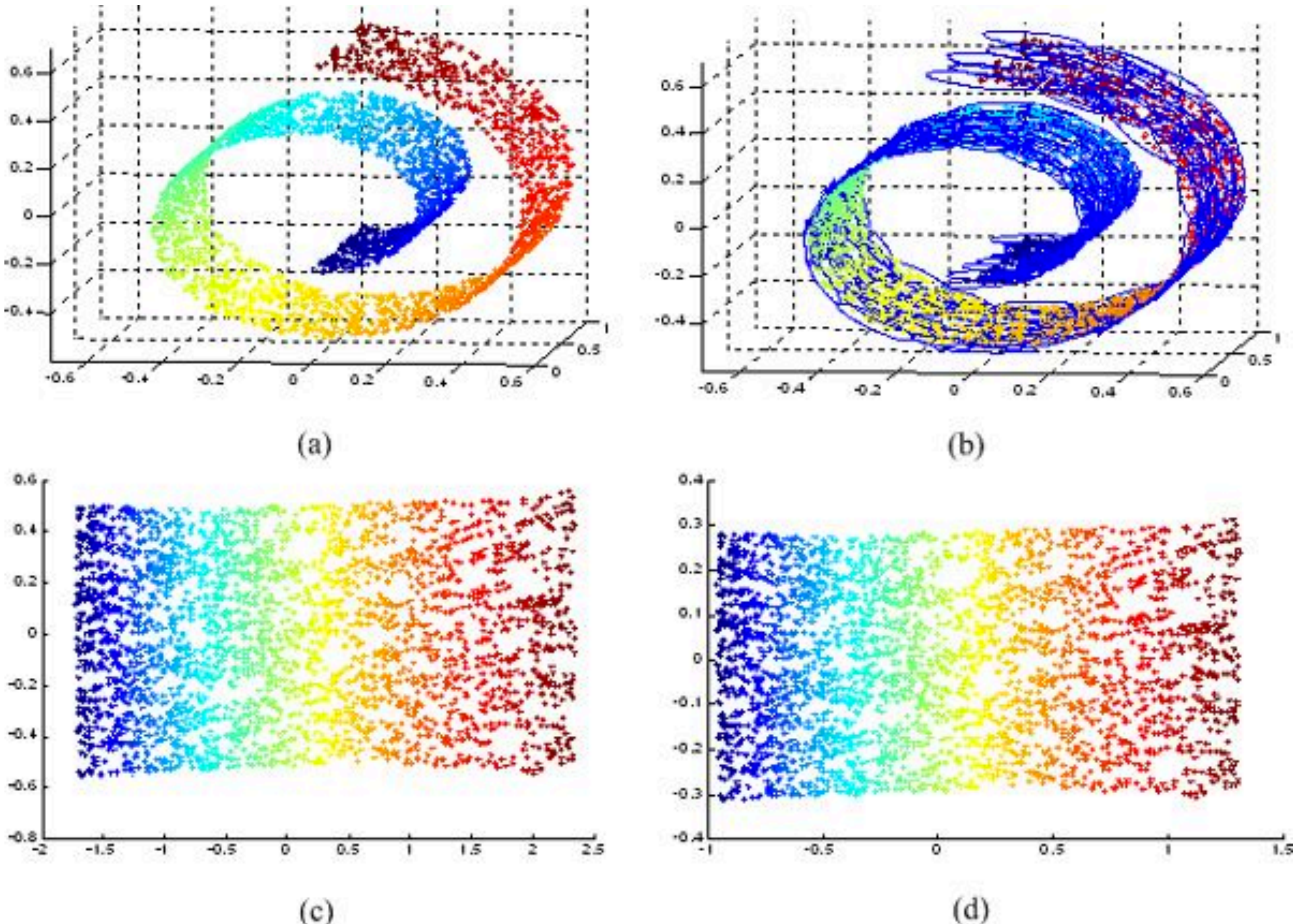


FIGURE 3. Eight clusters obtained using a hierarchical clustering algorithm using the elastic geodesic distance with DP alignment as metric.

Then we will study surfaces w/ many applications including dimension reduction



Then we will study manifolds: Mimic linear algebra to give rigorous definition of a manifold

- Recall: We turned key properties in $\mathbf{R}^n$ to define abstract vector space.
- **We will turned the key properties for regular surface in 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).

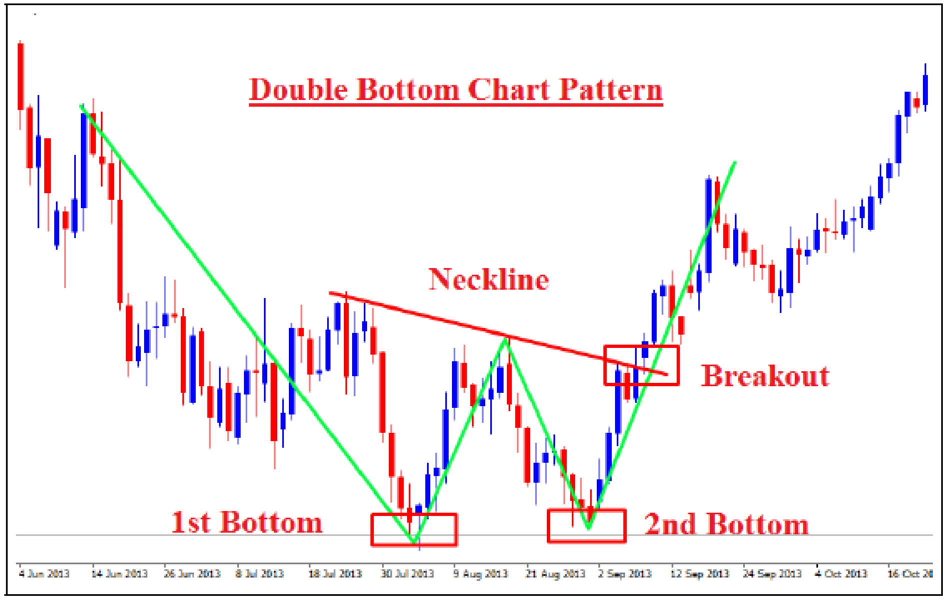
Extend Shortest Distance in R^n to Geodesics on manifolds

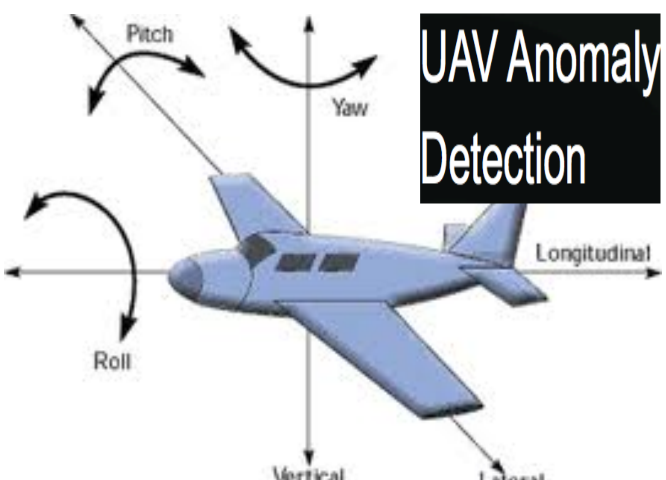
- In Euclidean space
- On a sphere
- On a manifold of curve shapes

Why is Differential Geometry Powerful?

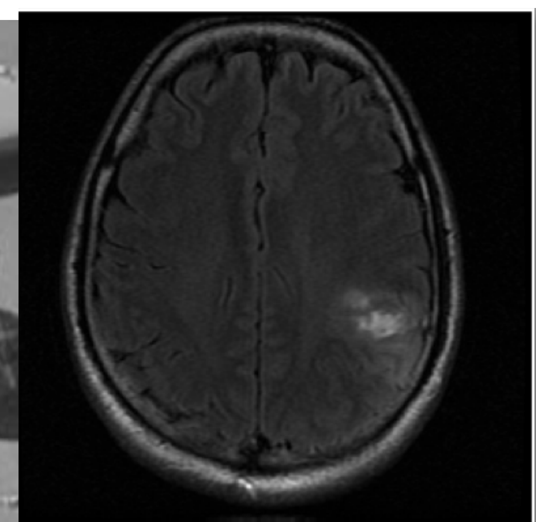
- Manifolds provide clever ways to model real world, big data and theoretic physics problems.
- Help to identify best metrics or distance functions for the problems at hand.
- Capture big data in a low dimensional intrinsic space.
- Provide theoretical background and powerful techniques for machine learning especially for nonlinear dynamical data.

Examples of Applications of Differential Geometry





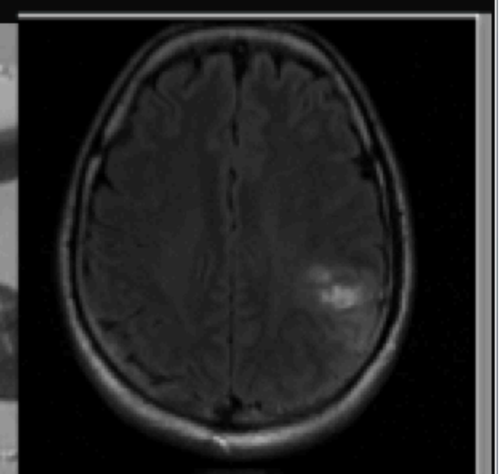
UAV Anomaly Detection



Detecting UAV Anomalies Using Manifold of Rigid Motions.

Selecting Chemical Sensors using Stiefel Manifolds.

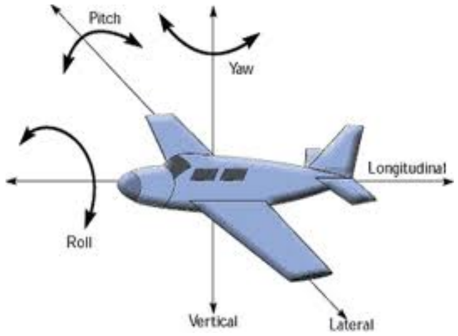
Find patterns in stock data & features in brain images.



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

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.



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*

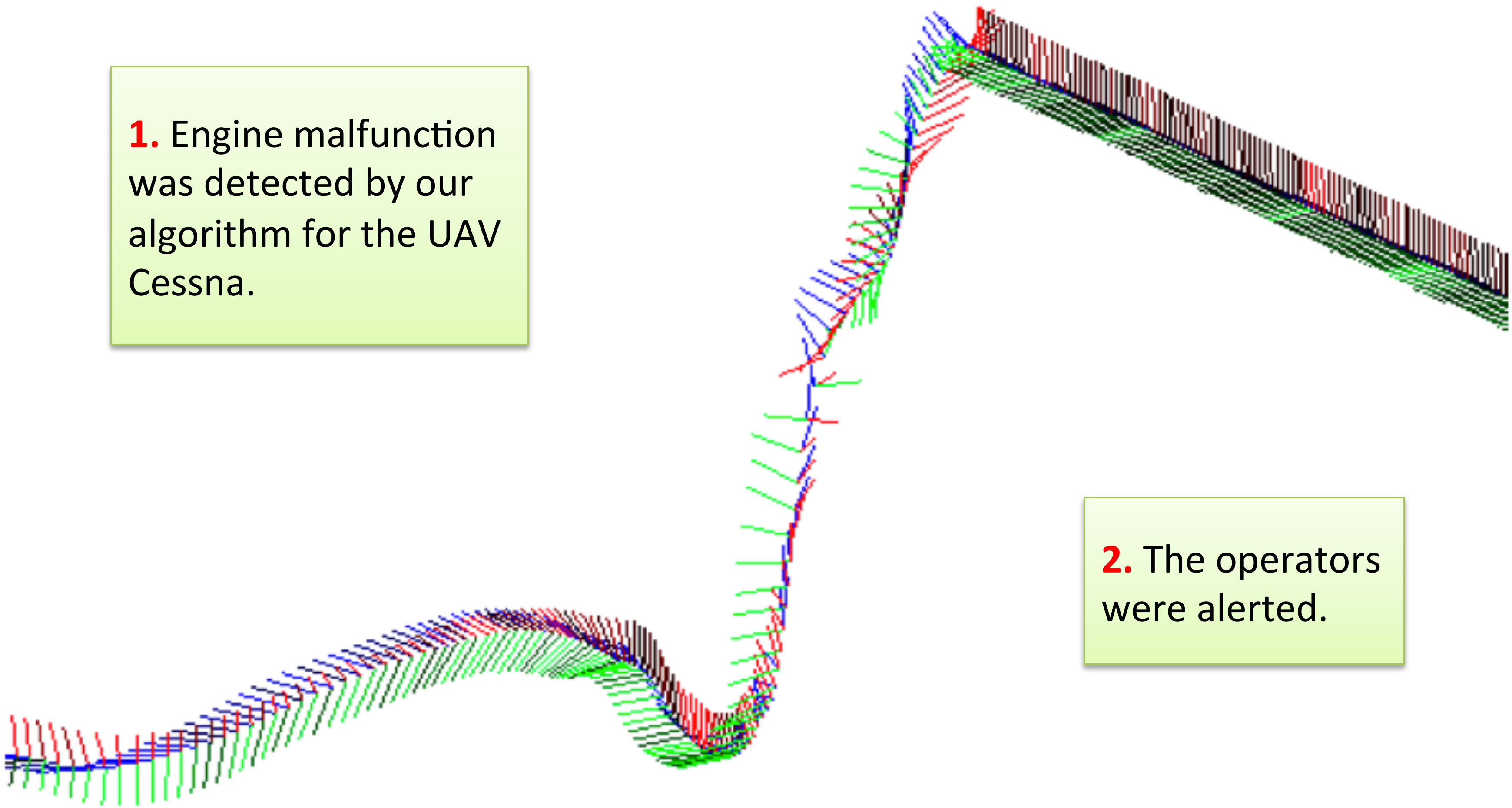
*2) **Lost control:** human error*

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.

Result: Example of Anomaly Detection: *Engine malfunction*

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



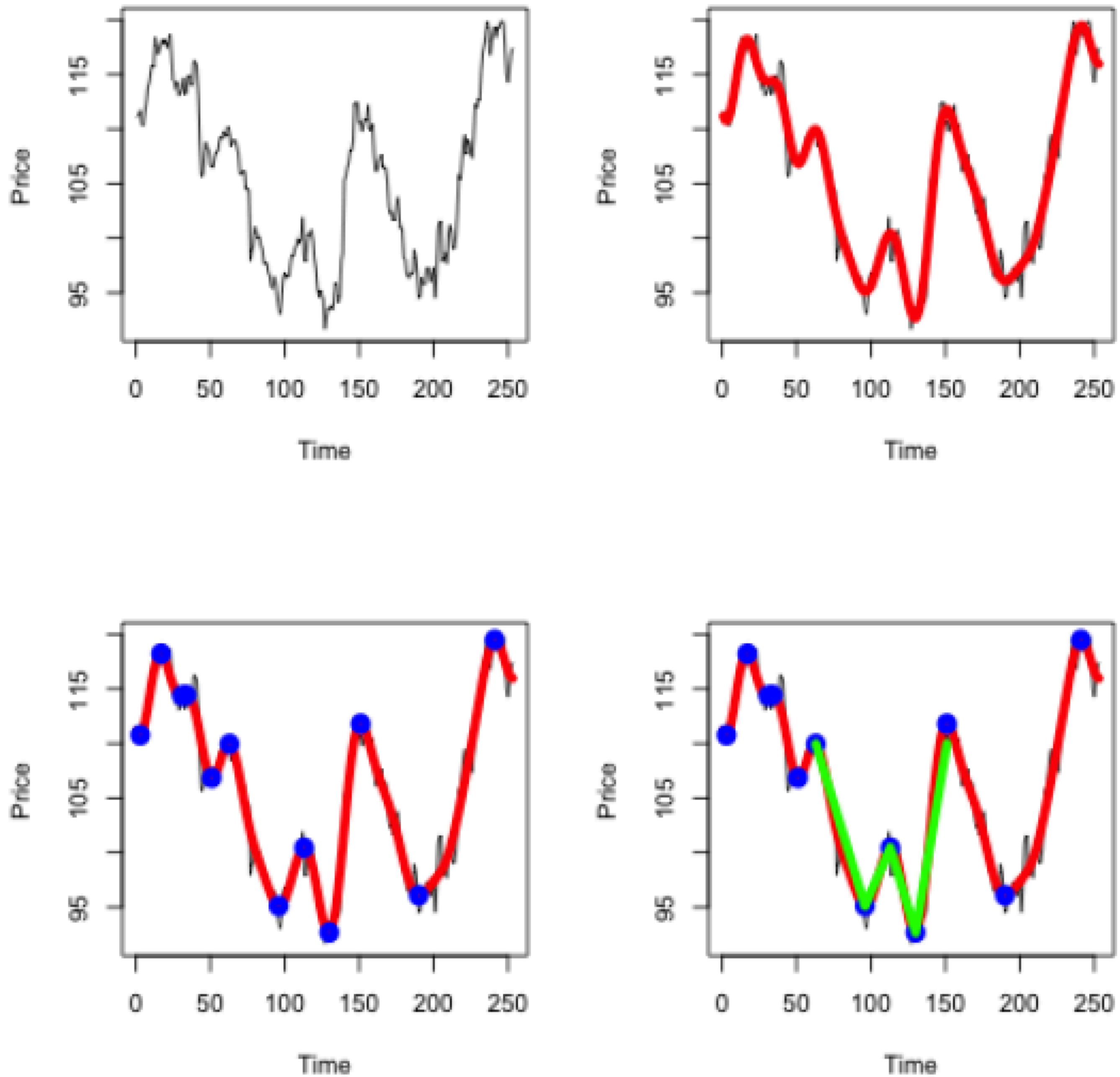
2. The operators were alerted.

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

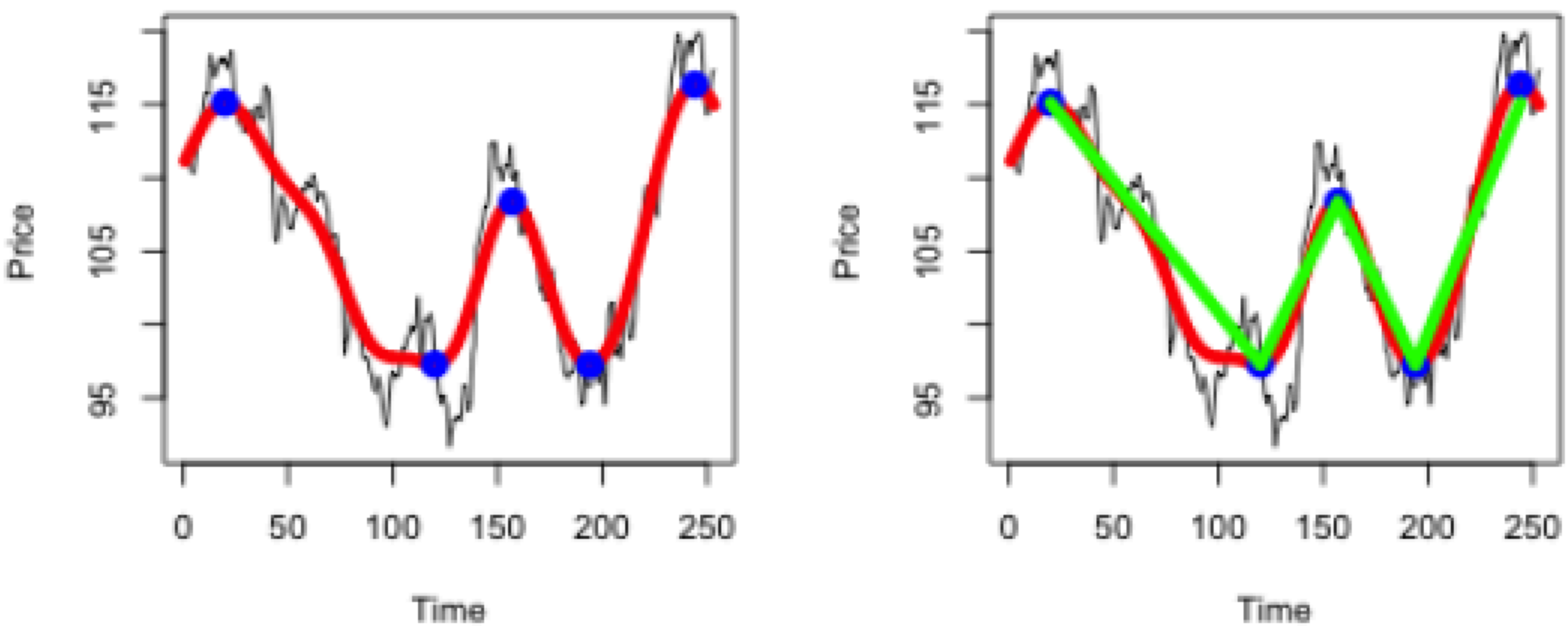
Example 2: Problem Statement

- 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.



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

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 $SPD(n)$.

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

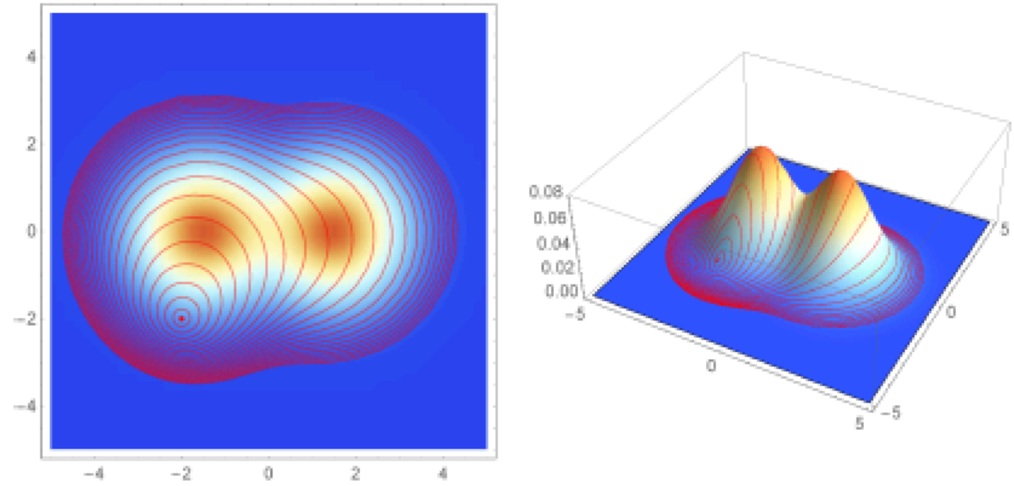


Figure 7.4 The contours represent points equidistant from $(-2,-2)$, under the scaled Euclidean metric ($\alpha = 1$) with a mixture of two normal distributions centered at $(\frac{3}{2}, 0)$ and $(-\frac{3}{2}, 0)$.

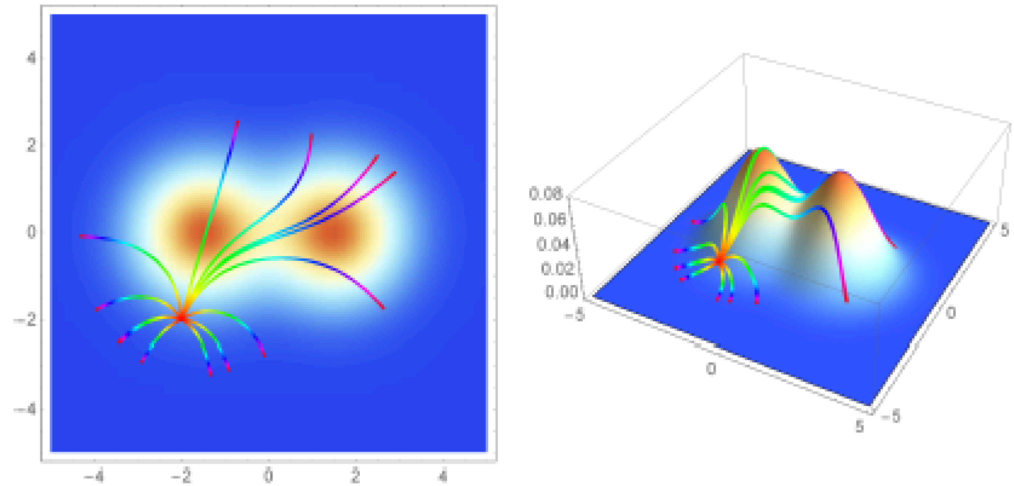


Figure 7.5 A selection of geodesics of the same length from $(-2,-2)$.

Hypergradient Descent Almost Converges

Working in progress

Conner DiPaolo* Weiqing Gu†

Harvey Mudd College
Claremont, CA, USA

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

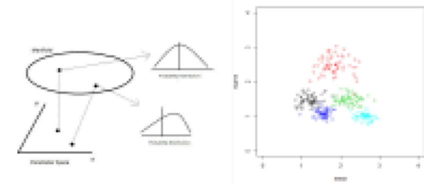
Math 197: Senior Thesis

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 $x_1, x_2, ..., x_n$ 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:

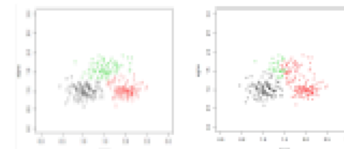
$$
\begin{aligned}
&\max_{\theta_1,\theta_2,\beta_0,\beta^T} D \\
&D_i \geq D \\
&D_i = \min_t \mathrm{DF}(p(x;x_i), p(x;\theta(t))) \quad \textit{for each } i \\
&p(x;\theta(t)) = p(x;(1-t)\theta_1 + t\theta_2) \\
&y_i(\beta_0 + \beta^T x_i) \geq 0 \quad \textit{for each } i \\
&\beta_0 + \beta^T\theta_1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

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.

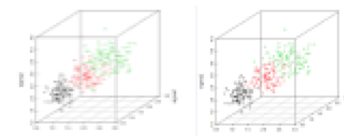
1. Select k pairs of parameters (μ_k, σ_k).
2. For each k, do the following t times:
   (a) Generate n samples from the univariate normal distribution $f(x; \mu_k, \sigma_k)$.
   (b) Reconstruct $f(x; \mu_k, \sigma_k)$ from these n samples and obtain unbiased estimates $(\hat{\mu}_k, \hat{\sigma}_k)$.

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

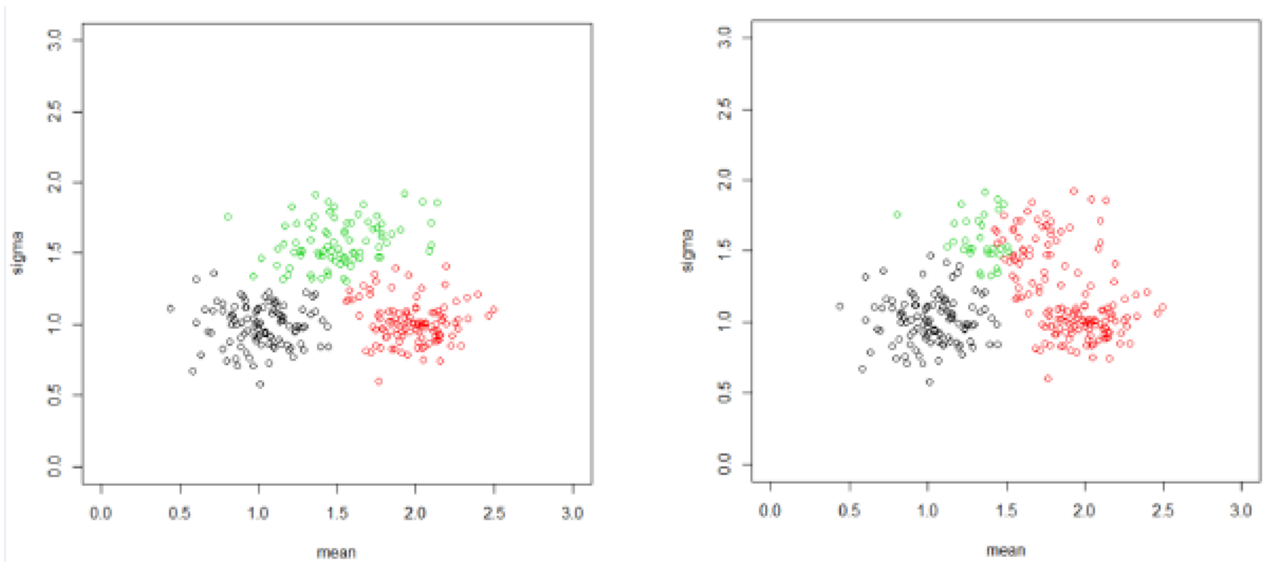
Methods

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

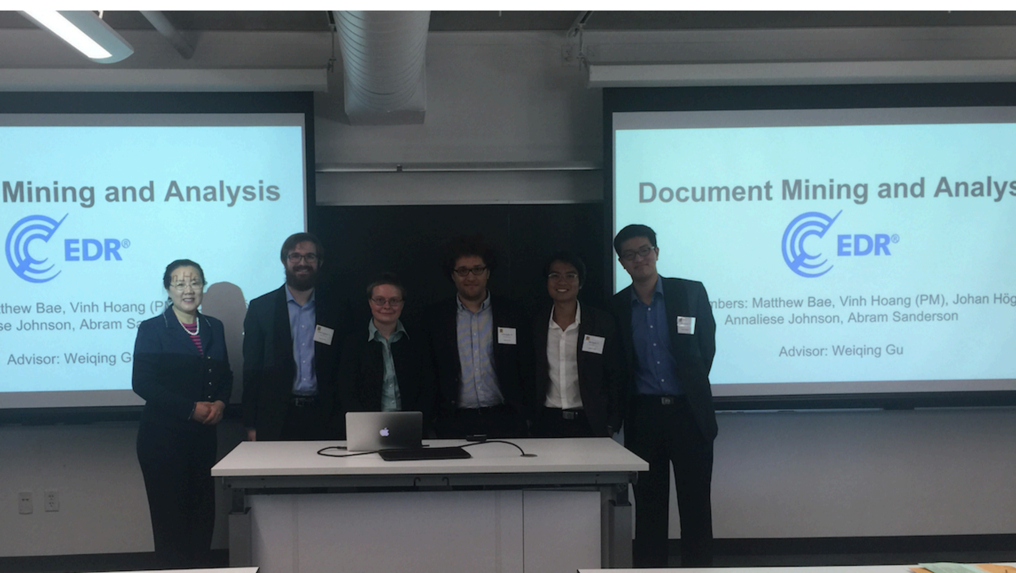
Mathematics of Emergent Gravity Based on Quantum Entanglement

Cynthia Yan
(Advisor: Weiqing Gu)
Harvey Mudd College

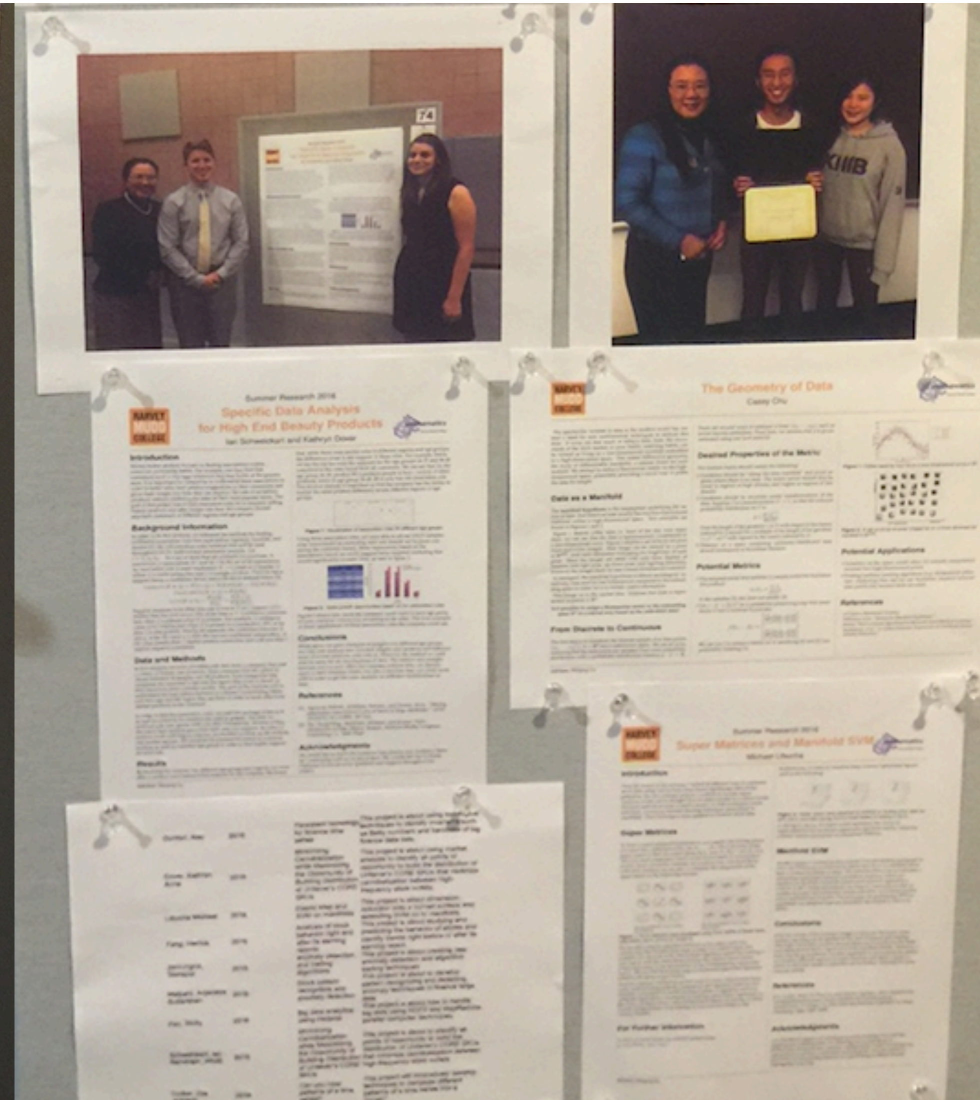
October 6, 2017

Also Applications in Math Clinics

Example: EDR clinic



Also Big Data Summer Researches at HMC

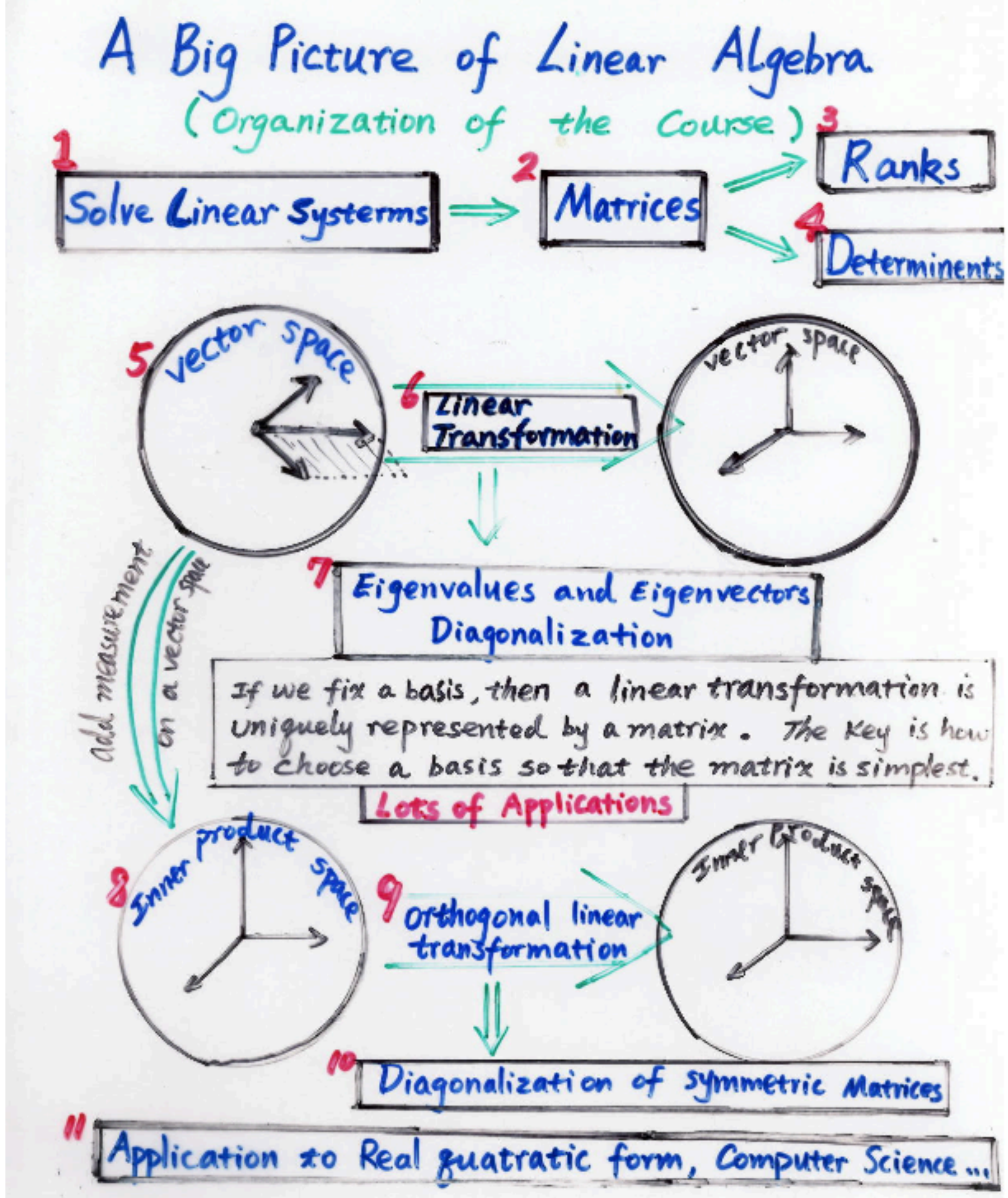


Now let's review some key ideas in linear algebra and mimic them to study Diff Geo!

- 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.

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!