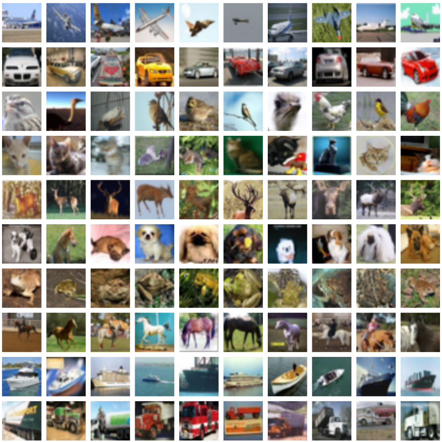
Introduction

Geometry of Data

August 23, 2023

CIFAR-10

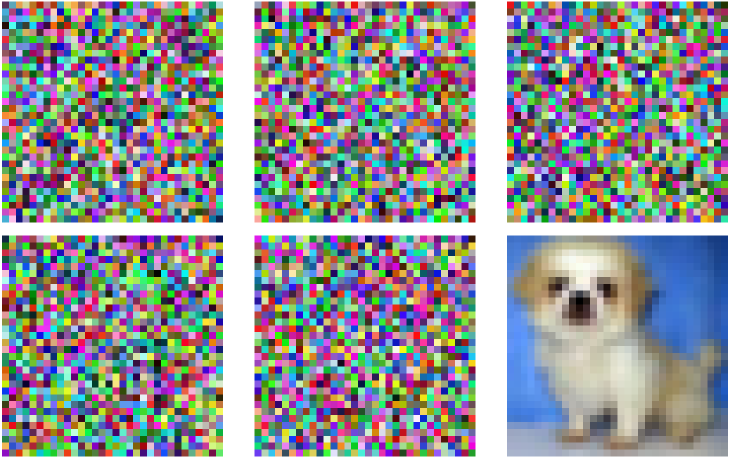
airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck



$32 \times 32 \times 3 = 3{,}072$ dimensions

10 classes

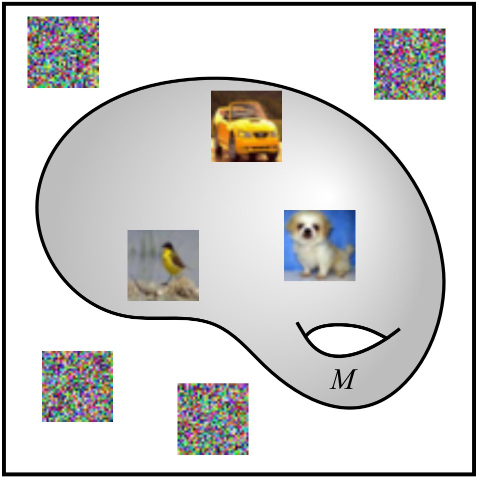
Uniform Random Images



just kidding!

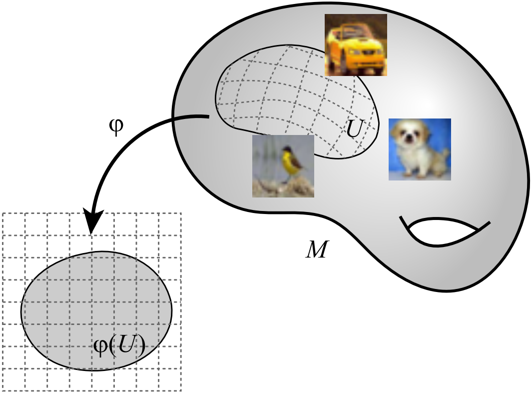
Manifold Hypothesis

Real data lie near lower-dimensional manifolds

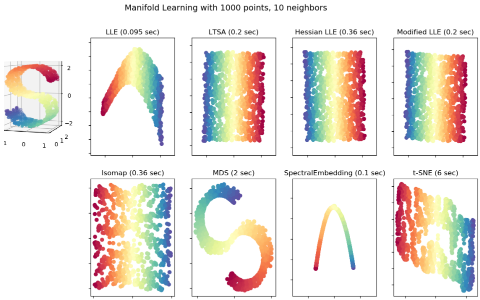


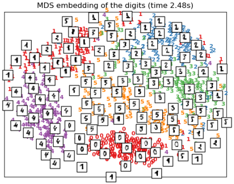
Manifold Learning

- Learn a model/representation for the data manifold
- Often involves finding a flat coordinate chart



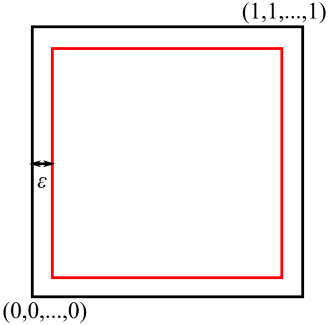
Manifold Learning





From scikit-learn.org

Volumes in High Dimensions



What is the volume of the unit d-cube shrunk by some small amount in each dimension?

$$V = (1 - 2\epsilon)^d$$

Approaches 0 as $d \to \infty$

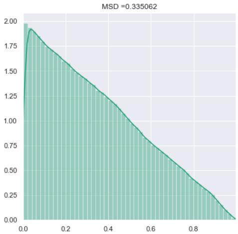
Example: $256 \times 256 \times 3$ images, $\epsilon = \frac{1}{256}$

$$V \approx 2.0 \times 10^{-670}$$

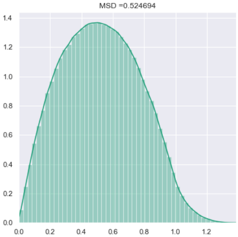
Distances in High Dimensions

Sample two points uniformly from the unit d-cube:
$X, Y \sim \mathrm{Unif}([0,1]^d)$

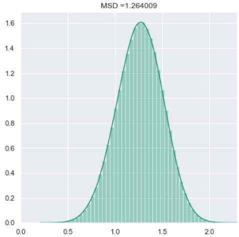
What is the distribution of the distance between them?
$D = \|X - Y\|$



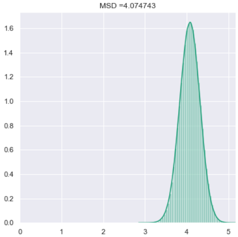
$d = 1$



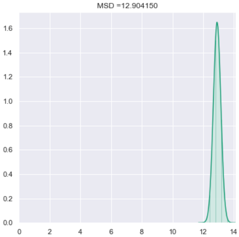
$d = 2$



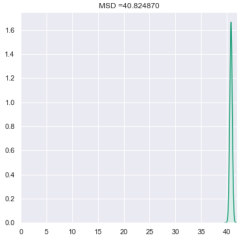
$d = 10$



$d = 100$



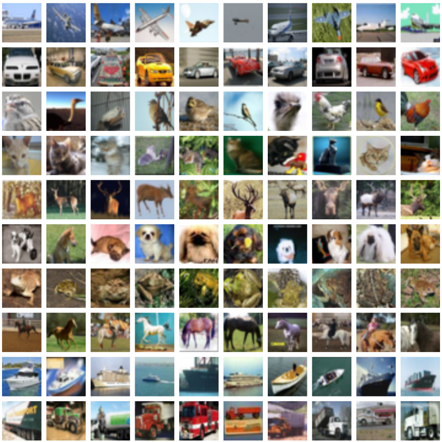
$d = 1{,}000$



$d = 10{,}000$

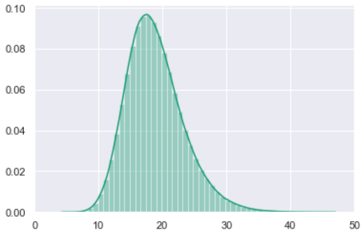
CIFAR-10

airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck

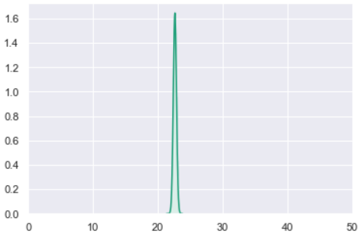


$32 \times 32 \times 3 = 3{,}072$ dimensions
10 classes

Distances in Real Data



CIFAR-10



$\mathrm{Unif}([0, 1]^{3072})$

Manifold-valued Data

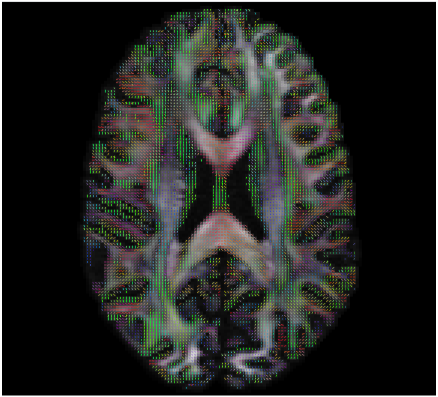
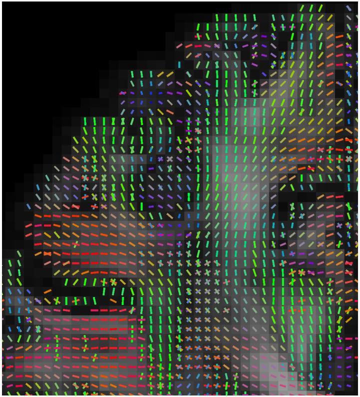
- Manifold already known, not learned
- Manifold arises from natural non-linear constraints on data
- Linear data analyses (in fact, vector space operations) violate these constraints

Directional Data

Data living on a circle (S^1) or sphere (S^2), etc.

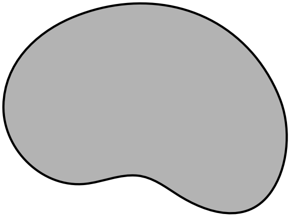
- Orientation of molecules in protein structure
- Direction of robot or autonomous vehicle
- Position on the earth
- Motion capture: orientation of joints
- Time (time of day, day of the year, etc.)

Directional Data: Diffusion MRI

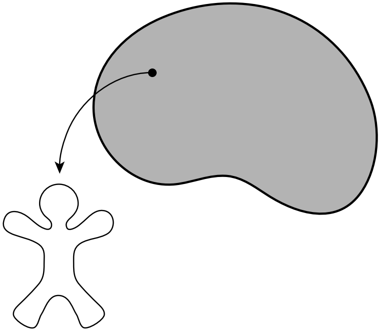
Voxel features are directions of axons in brain

Shape Manifolds



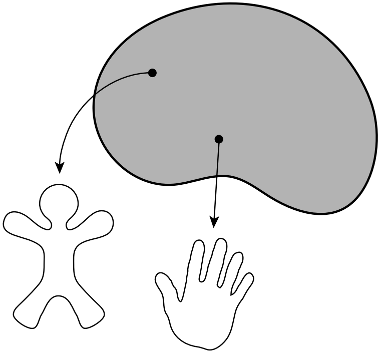
A shape is a point in a high-dimensional, nonlinear manifold, called a **shape space**.

Shape Manifolds



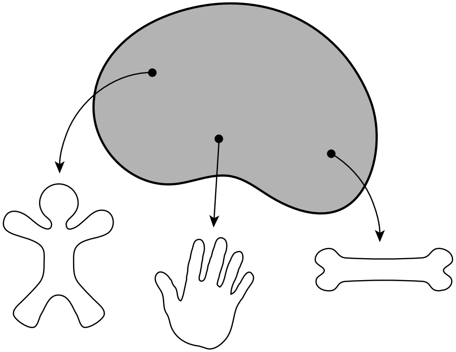
A shape is a point in a high-dimensional, nonlinear manifold, called a **shape space**.

Shape Manifolds



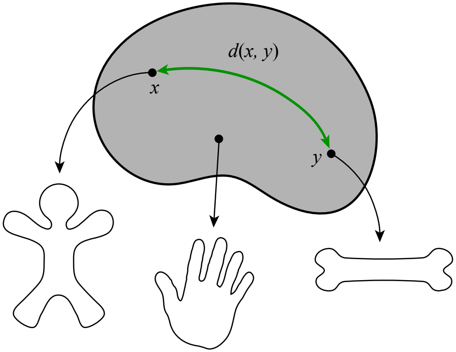
A shape is a point in a high-dimensional, nonlinear manifold, called a **shape space**.

Shape Manifolds



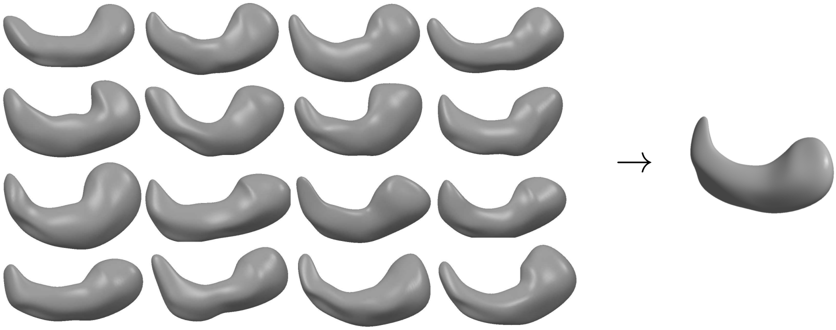
A shape is a point in a high-dimensional, nonlinear manifold, called a **shape space**.

Shape Manifolds

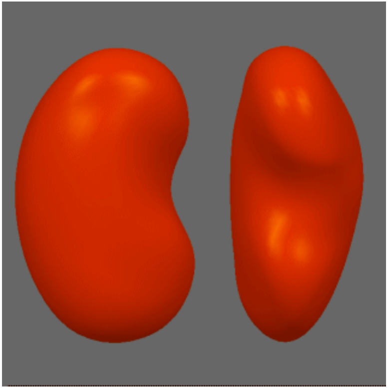


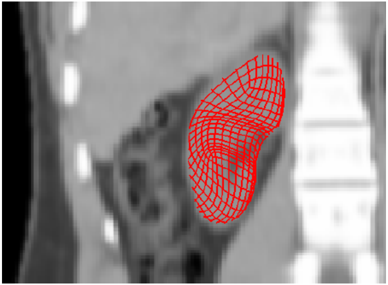
A metric space structure provides a comparison between two shapes.

Shape Statistics: Averages



Shape Statistics: Variability





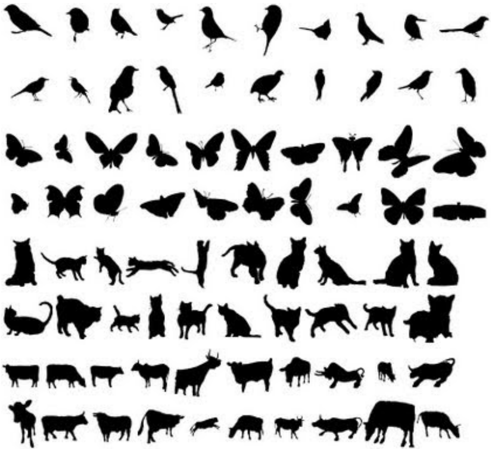
Shape priors in segmentation

Shape Application: Bird Identification

American Crow

Common Raven

Shape Statistics: Classification

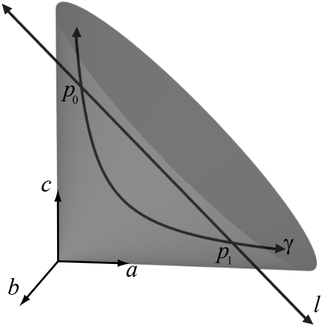


http://sites.google.com/site/xiangbai/animaldataset

Information Geometry

Parameters of a probability model live on manifolds

Example: covariance matrix of a 2D Gaussian distribution:



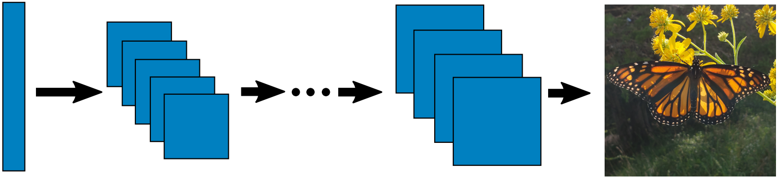
$\Sigma \in \mathrm{PD}(2)$ is of the form

$$\Sigma = \begin{pmatrix} a & b \\ b & c \end{pmatrix},$$

$$ac - b^2 > 0, \quad a > 0.$$

(positive-definite constraint)

Deep Generative Models



Input:

$z \in \mathbb{R}^d$

$z \sim N(0, I)$

$\xrightarrow{g = g_L \circ g_{L-1} \circ \cdots \circ g_1}$

Output:

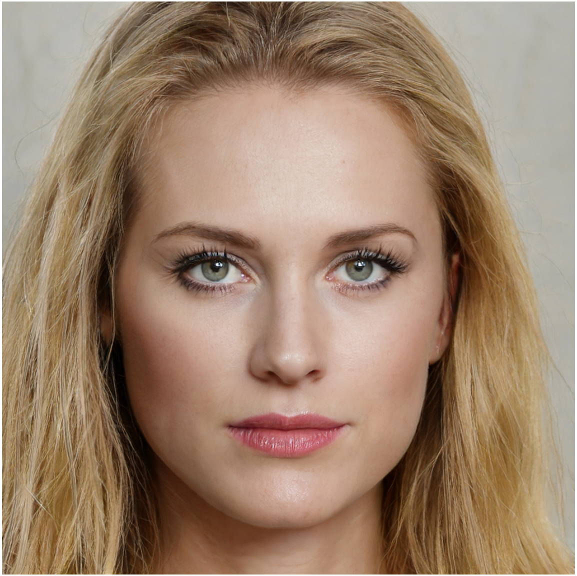
$x \in \mathbb{R}^D$

$d << D$

These are not real people

Karras et al., CVPR 2020, and `thispersondoesnotexist.com`

These are not real people



Karras et al., CVPR 2020, and `thispersondoesnotexist.com`

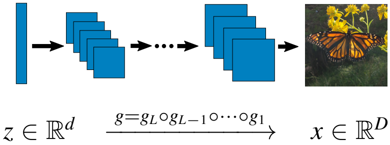
These are not real people

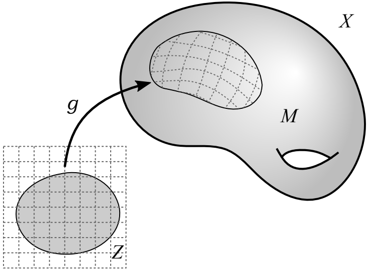
Karras et al., CVPR 2020, and `thispersondoesnotexist.com`

These are not real people

Karras et al., CVPR 2020, and `thispersondoesnotexist.com`

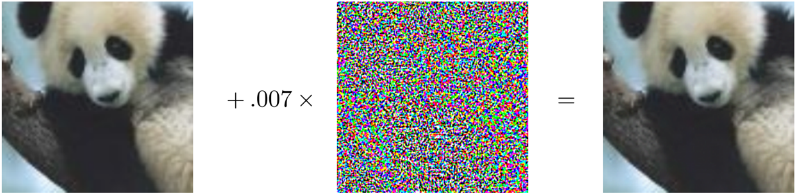
Generative Models as Immersed Manifolds





Shao, Kumar, Fletcher, The Riemannian Geometry of Deep Generative Models, DiffCVML 2018.

Adversarial Examples



$$x + .007 \times \text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)) = \boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$$

$\boldsymbol{x}$
"panda"
57.7% confidence

$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

Goodfellow et al. ICLR 2015