Applications of exp and geodesics:

- The mean shift algorithm

Let \( x_i \) be \( n \) points in \( \mathbb{R}^d \)

we wish to cluster. We do so by finding

the peak of a kernel density estimate

\[
    f(y) = \frac{1}{n} \sum_{i=1}^{n} \delta_k \left( \frac{||y - x_i||^2}{\epsilon^2} \right)
\]

where

\[
    \int K(u) \, du = 1
\]

(which means that \( f \) is a density)

Now notice that

\[
    \nabla_y f = - \frac{1}{n} \sum_{i=1}^{n} \left[ k'(\epsilon^2 \frac{||y - x_i||^2}{\epsilon^2}) \cdot \frac{y - x_i}{\epsilon^2} \right]
\]
So that:

\[
\frac{h^2}{2} \sum_{i=1}^{n} \frac{y_i - \frac{\sum k'(y_i - x_i)^2}{h^2}}{\sum k'(y_i - x_i)^2} = \mathbf{y} - \frac{\sum k'(y_i x_i)}{\sum k'(c)}
\]

Now we know that at the peak, \( \nabla_y f = 0 \)

So, AT THE PEAK.

\[
\hat{y} = \frac{\sum k'(c) x_i}{\sum k'(c)}
\]

This suggests the iteration

\[
y_{k+1} = y_k + \left[ \frac{\sum x_i k'(c)}{\sum k'(c)} - y_k \right]
\]

I mean shift vector
This iteration is effective, and leads to a very useful clusterer.

But what if the objects we wish to cluster are points on some Riemannian manifold?

Example: estimates of pose from different sources

\[ \epsilon \subseteq \text{SO}(3) \times \mathbb{R}^3 \]

\[ \downarrow \quad \text{Euclidean group} \]

\[ \epsilon = \text{SE}(3) \]
Example: Chromatic noise

Noise model: Intensity is correctly estimated, direction of RGB vector is not.

E.g. (I think) Color demosaicing mean shift on 2-sphere (in fact, hemisphere — paper says $C_{3,1}$, but this is nonsense).

Example: 3D Motion clustering.

A set of $3D$ pts in Orthographic views gives a rank 4 Data matrix, when motion is rigid.
Equivalently, the data matrix represents a point in \( G_{2F, 4} \), the point in subspace yields the camera motion and 3D geometry.

Now, if we have several independently moving rigid objects, IF we can segment the objects, above applies to each separately.

Traditional problem:

Segment AND obtain recon, motion
One strategy

- each group of 4 points yields an estimate of a subspace

- (hopefully) outliers dominate outliers - careful choice of groups can help here

- we then cluster to find segments

- clustering is on $G_{2F,4}$
Q: how can we do mean shift on a Riemannian manifold?

A: One algorithm is quite straightforward, but computations can get nasty.

Algorithm: \( x_i \in M \)

(represented in some set of coords, etc)

Define the density estimate

\[
f(y) = \frac{C}{n} \sum_i k \left( \frac{d(x_i, y)^2}{h^2} \right)
\]

Where:

- \( C \) normalizes
- \( d(x_i, y) \) is distance on manifold
Notice there is a minor technical difficulty here — the integral of a function depends on where you put it, — it's quite hard to simply shift f(x) around — which is why the normalization coast is at the front.

We need to define the gradient of a function on a Riemannian manifold

\[ \nabla f : = \text{vector such that} \]

\[ x f = \langle x, \nabla f \rangle \text{ for all } x \]
In coordinates

\[ X f = \sum_i x_i \frac{\partial f}{\partial u_i} \]

rep \( f \) in local coords

\[ \langle X, Y \rangle = \sum_{i,j} q_{ij} x_i y_j \]

so define \( q_{ij} = \frac{\partial}{\partial u_i} \)

\[ \sum_j q_{ij} q_{jk} = \delta^i_k \left\{ \begin{array}{ll} 1 & i = k \\ 0 & \text{otherwise} \end{array} \right. \]

then

\[ \nabla f = \left( \sum_j q_{ij} \frac{\partial f}{\partial u_j} \right) e_i \]

unit basis vector for \( i \)-th coord.
\[ \nabla f = \sum_{i} k'(i) \left[ \frac{\nabla d^2(x_i, y)}{h^2} \right] \]

*notice that this is a tangent vector if we can evaluate \( \nabla d^2(x_i, y) \) we're in business.

- We now need some facts about geodesics and the exponential map.
- Consider the geodesic leaving \( y \), in dirn \( T \).

\[ \text{length} \ exp_y(T) = \text{travel along section of this geodesic from } u=0 \text{ to } u=1 \]

we assume \( T \) is small enough that this exists
\[ \text{length} \left[ \exp (T) \right] = \int_{u=0}^{1} \langle T, T \rangle^{1/2} \, du. \]

(where the curve is given by ODE $\nabla T = 0$)

Now:

\[ T \langle T, T \rangle = \text{rate of change of } \langle T, T \rangle \text{ as we move along geodesic} \]

by properties of connections

\[ \rightarrow 2 \langle \nabla_{T} T, T \rangle = 0 \]

by properties of geodesics.
Notice
\[ \text{length} \left( \exp(\alpha T) \right) = \alpha \text{length} \left( \exp(T) \right) \]
(\(\alpha > 0\))

AND
\[
\|T\| = 1 \implies \text{length} \left( \exp(T) \right) = \int_0^1 du = 1
\]

So
\[ \text{length} \left( \exp(T) \right) = \|T\|. \]
Now consider the geodesic from $y$ to $x_i$.

- Define $\nabla$ by $x_i = \exp_y(\nabla)$

- Notice: a small positive step along $\nabla$ moves me towards $x_i$.

$$d^2(x_i, y) = \|\nabla\|^2 = \left[\langle\nabla, \nabla\rangle^{1/2}\right]^2$$

There is a sign issue here.

$$\nabla d^2(x_i, y) = 2d(x_i, y) \cdot \nabla d(x_i, y)$$

$$= 2 \|\nabla\| \cdot \left[ -\frac{\nabla}{\|\nabla\|}\right] = -\nabla$$

Drain along geodesic away from $x_i$. 
The mean shift update now becomes

\[ m_k = -\sum_i k_i \cdot V_i \]

This is a tangent vector at \( y_k \)

and then we form

\[ y_{k+1} = \exp_y (m_k \cdot y_k) \]

Notice

- In \( \mathbb{R}^n \), this is just what we've used to
- For anything else, we need to be good at \( \exp \).
Some examples:

- Lie groups.

(a lie group is a group that is a differentiable manifold, where the group's action on itself $G \times G \to G$ is differentiable; they have lots of interesting properties; all our usual matrix groups are lie groups)

- Assume a matrix representation

- Define two MATRIX operators

$$e^M = \sum_{i=0}^{\infty} \frac{1}{i!} M^i$$
\[
\log(M) = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i} (M - \text{Id})^i
\]

You can verify that \( \log(e^M) = M \)
\( e^{\log M} = M \) etc.

These series may come with requirements on \( M \) for convergence!

One can now show (O'Neil, ch 11) that

\[
\exp(M) = X e^{[X^{-1} M]}
\]

Matrix: a point on group
If \( \exp(X) = Y \) then

\[
V = X \log \left[ X^{-1} Y \right]
\]

None of this is particularly attractive; but in special cases, we can do things

eg. \( \text{SO}(3) \) (3D rotations)

notice: if \( M(\phi) \) is \( \in \text{SO}(3) \)

then \( M^T M = 1d \)

so \( M^T \nabla M + \nabla \phi \left( \nabla M^T \right) M = 0 \)

now consider tangents at origin, \( M = 1d \)

\( \nabla \phi M + \nabla \phi M^T = 0 \)
We always work with the natural inner product on the matrix space.

So if \( u, v \) are tangent vectors represented as matrices, we use

\[
\langle u, v \rangle = \sum_{ij} u_{ij}^* v_{ij}^*
\]

\[= \text{Trace} \left( u^T v \right)\]
So tangent space at origin consists of \( N \text{ s.t. } N + N^T = 0 \) write

\[
[\omega]_x = \begin{bmatrix}
0 & -w_z & w_y \\
w_z & 0 & -w_x \\
-w_y & w_x & 0
\end{bmatrix}
\]

(here \( \omega = [w_x, w_y, w_z] \) is axis of rotation, \( \|\omega\| \) is magnitude)

then

\[
e_x = I_d + \frac{\sin \|\omega\|}{\|\omega\|} [\omega]_x + \frac{(1 - \cos \|\omega\|)}{\|\omega\|^2} [\omega]_x^2
\]

(called the Rodrigues formula - expand, and match terms \( \omega \) / Taylor series)
It's also straightforward to extract axis angle from a rotation. Consider \( M \in SO(3) \):

\[
Mv = v \quad \text{if } v \text{ is axis}
\]

Other two eigenvalues reveal angle of eigenvalues of 2D rotation:

\[
\begin{pmatrix}
\lambda_1, \lambda_2 = e^{\pm i\theta}
\end{pmatrix}
\]

This seems to work rather well for clustering pose.

(Subbarao + Meer paper)
Working on Grassmannians

(Earlier notes were a little sloppy; better here)

\[ G_{n,p} = \{ \text{set of } p \text{-dimensional flats through the origin in } \mathbb{R}^n \} \]

(which is the same as \( p - 1 \text{-d flats in } \mathbb{R}^{n-1} \))

Multiple representations available. We will parametrize with

\[ [Y] = n \times p \text{ matrix} \]

s.t. \( Y^T Y = I_d \).
Notice that multiple Y's can represent the same point on the Grassmannian.

- If \( M_{n \times n} \) is a rotation, then \( MM^T = 1_d \)

- \( YM \) represents the same point

Interpretation:

- \( n \) orthogonal basis for vectors for the space we wish to rep'n
- \( M_{n \times n} \), right action is a rotation of that basis w/in that space.
Tangent vectors:

- Easy argument (there's a harder argument in Edelman).

want \( \Delta \) st.

\[
(Y + 3\Delta)^T(Y + 3\Delta) = I_d
\]

to first order.

so

\[
Y^T\Delta = 0
\]

\( \in \mathbb{R}^{n \times p} \)

Tangent space is \( (n-p) \) dimensional, as it should be.
Metric

\[ \langle \Delta, \Gamma \rangle = \text{tr}(\Delta^T \Gamma) \] as before.

Then, there is a parametric form for \( \exp \)

\[ \exp_y(H) = [Y Y^T u^T] \begin{pmatrix} \cos \Sigma & -\sin \Sigma \\ \sin \Sigma & \cos \Sigma \end{pmatrix} V^T \]

where \( u \Sigma V^T \) is SVD of \( H \).

Proof: Edelman, p15, easy!

And

if

\[ M = \exp_y(H) \]

\[ H = A \sin^{-1}(s) B^T \]
Where

\[ A S D^T = M - YY^TM \]

\[ B C D^T = Y^TM \]

\[ C^TC + S^TS = Id \]

S, C diagonal.

(Subbarao + Meer - Haven't checked this!)