Open Access

Milestone 011 — Geometry should follow invariance, not aesthetics

Tyler J.H.G.

Abstract

Which geometry preserves the invariances a task actually needs?

Research period: September 2–3, 2026
Historical anchors: #194–#198

For a while I described the geometric branch as **"spheres all the way down."** If hyperspherical state was useful in one part of the model, perhaps representation, reasoning, hierarchy, and relationships could all use compatible spherical constructions.

Writing down their requirements exposed the problem. Hierarchy needs room for growth; scalar quantities need a notion of scale; relationships may need order-sensitive composition; a concept with several contextual realizations may need more than one direction. Those are different demands on a geometry.

I started evaluating each part separately: what must its state preserve, which changes should leave it equivalent, and which operations does the model need to perform on it?

1. Visual intuition

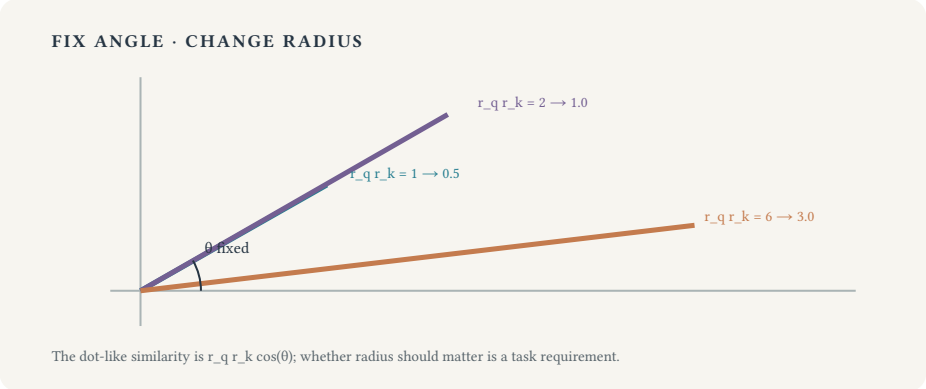

Figure 1: At a fixed angle, changing either radius changes the inner product.

Published Sep 29, 2026

2. A simple example: radius and direction

Suppose direction represents semantic identity and radius represents a structural quantity such as argument depth. For a nonzero vector, write

`x = r u`, with `||u|| = 1`.

We have assigned two meanings to the coordinates, but the metric still determines how they interact. In ordinary Euclidean geometry,

$ds^2 = dr^2 + r^2 \, d\Omega^2$.

Changing radius changes the scale of angular distance. Inner-product similarity also couples the quantities:

`q · k = r_q r_k cos(theta)`.

The same angle produces a different score when either magnitude changes. If semantic identity is supposed to remain invariant to the structural quantity, this coupling violates the intended requirement.

A product representation or explicit side channel makes independence part of the construction. If the quantities are intended to interact, a coupled or warped geometry may suit that requirement. Specifying the intended behavior gives us a way to choose between them.

> **Sticky note — product geometry:** give two independently varying factors separate geometries. Their independence is then explicit in the construction.

= The same test applied elsewhere

Hyperbolic geometry supplies room for exponential growth with distance from the center, making it a candidate for tree-like hierarchy. The relevant question is whether we need that growth and separation at the dimensions and distortion we can afford. Finite trees can also be represented in ordinary spaces, with different tradeoffs.

Subspace-valued representations offer a way to represent several persistent contextual realizations together. Grassmann geometry treats the subspace as the object, independently of the basis used to describe it. That property is useful when the consumer needs the subspace and respects basis invariance. A consumer that receives only one emitted vector may make the hidden subspace an elaborate parameterization of that vector.

Relation operators need to preserve order when the task depends on order. If applying A then B can differ from applying B then A, an operator family whose members always commute cannot represent both outcomes.

Each case supplies a condition to check before training. The architecture has to express the distinction that motivated it.

3. Research context

Hyperbolic representation learning already connects negative curvature with hierarchy. Grassmann and subspace methods supply machinery for objects whose

identity is a subspace. Geometric deep learning more broadly uses symmetry and invariance to constrain a model's mathematical structure.

Those connections made the literature useful for design. For an existing construction, I could ask which property made it appropriate for its task and whether my task had the same requirement. The answer determines which tools transfer and what still needs to be tested.

4. Try a small example

A toy calculation holds the angle fixed and varies the radii to show how scale changes the inner product.

```
import numpy as np
theta=np.pi/3
for rq,rk in [(1,1),(2,1),(2,3)]:
    print(rq,rk,'dot-like similarity=',rq*rk*np.cos(theta))
```

Saved output:

```
1 1 dot-like similarity= 0.5000000000000001
2 1 dot-like similarity= 1.0000000000000002
2 3 dot-like similarity= 3.000000000000001
```

5. Interpretation

Different parts of the system can require different invariances, growth laws, and operations. Deriving those requirements first makes the choice of geometry testable.

Mathematical compatibility establishes that a construction can express the intended behavior. Training then tests whether that construction improves the model under the available budget.

6. Sources and chronology

- Milestone notebook 011 — preserved chronological source
- Research chronology
- Publication dispositions and Atlas-to-article map

The matching Atlas theme is **Radius–angle coupling**. The earlier Visual Intuition Atlas notebook preserves the source visualization.

The next chronological article is **Milestone 012 — Natural-source task distinctions**.