Hypersolid User Guide

Laurent Oliversen · · Build 2026-09-09

Contents

This web application is a tool for exploring polytopes and analyzing multidimensional datasets, all powered by an intuitive 3D interface.
This guide explains the mathematical concepts and features of the Hypersolid application.

You can download this documentation in PDF format:
https://hypersolid.app/docs/hypersolid-user-guide-en-light.pdf

1. Getting Started

1.1. How the Interface Is Organized

The application is organized into tabs, listed in the top bar.

On the left, the working tabs:

- `Workspace` : the 3D scene and its settings panels. This is the heart of the application (part 2 and sections 3.1, 4.2, 4.3 of this guide).
- `Data` : data import and preprocessing (part 3).
- `Stats` : descriptive and inferential statistics (section 4.1).
- `Analysis` : a drop-down menu gathering every analysis module — `Advanced Stats` , `Hy-Personality` , `Factor` , `Cluster` , `Sup ML` , `Forecast` , `Hy-Performance` , `MMM` , `Optimization` and `Hyperion` (part 5).
- `Export` : every extraction (part 6).

On the right, the service tabs: `User Guide` (this document), `Upgrade` , `Contact` .

1.2. The Workspace Floating Panel

In the `Workspace` tab, all settings are gathered in a floating panel that can be moved by grabbing its header, and folded away with the collapse button. It is divided into five sections, selectable through the tabs at the top of the panel:

- `Solid` : polytope generation and style (sections 2.1 and 2.2).
- `Dataset` : reference datasets, custom points, point and label style (section 3.1).
- `Proj` : rotations, shears, projections and camera (section 2.3).
- `Exp` : exploration of the projected data — filters, orbitals, valences, scene markers (section 4.2).
- `Motion` : animation and interpolation (section 2.4).

Each section keeps its folders folded or unfolded as you navigate between them.

Some analysis tabs also offer a toggle to the scene view, which displays the 3D scene while keeping the module panel within reach.

1.3. User Modes

A selector, at the top right, chooses the **user mode**. The mode changes nothing in the computations: it simply hides the tabs and shapes outside its scope, to keep the interface light.

- `🔭 Explorer`: geometric and personality exploration. Gives access to the whole shape catalogue, as well as to the `Hy-Personality` and `Factor` tabs.
- `🧪 Data Scientist`: data analysis. Opens `Hyperion`, `Factor`, `Cluster`, `Sup ML`, `Optimization` and `Forecast`. The shape catalogue is restricted to the essential shapes, which serve as a visual landmark.
- `📊 Business Analyst`: performance management. Opens `Hyperion`, `Optimization`, `Forecast`, `Hy-Performance` and `MMM`. The shape catalogue is restricted as well.
- `⚡ Hypervision`: gathers the three previous ones, without restriction.

The `Workspace`, `Data`, `Stats`, `Advanced Stats` and `Export` tabs are visible in every mode.

1.4. Plans and Locking

Every feature belongs to a plan: `Guest`, `Free`, `Hyper`, `Pro`. Each plan strictly contains the previous one.

A feature that your plan does not cover is **not hidden**: it appears greyed out, suffixed with a padlock 🔒, and hovering it states the required plan. You therefore always see what the application can do, even without access to it.

The details of the plans — row and column quotas included — are found in the `Upgrade` tab.

2. Geometric Core

2.1. Solid Gen

The application generates geometric shapes defined by their vertices and edges in an N-dimensional space. The application supports the following shapes. Note that spheres are treated as polytopes for visualization purposes, even though they are technically smooth manifolds.
The generation and projection pipeline consists of the following steps:

- Polytope Generation (100% CPU Maths)
- ND-4D: normalization, rotation, shearing, projection (CPU Maths)
- 4D-3D: rotation, shearing, projection, normalization, scaling (CPU Maths)
- 3D-2D/1D: collapse onto a plane or a line (CPU Maths Filter)
- Scene Assembly (Construction of spheres, lines, and faces)
- Camera / GPU Rendering (Calculation of the final angle and screen display)

The order of transformations is indeed as follows: 1) rotation, 2) shearing, 3) projection.

The dimensional axes are denoted as follows:

- X, Y, Z for the usual 3D axes (right, up, and forward-oriented)
- W for the fourth dimension
- V5, V6, V7, V8, ... for all higher dimensions

2.1.1. Polygons (2D)

- `Triangle`: 2D simplex (3 vertices). The simplest regular polygon.
- `Square`: 2D hypercube (4 vertices). Regular polygon.
- `Rectangle`: polygon derived from the square, with sides of irregular length.
- `Pentagon`: regular polygon with 5 vertices.
- `Hexagon`: regular polygon with 6 vertices.
- `G2 Roots`: structure based on the vectors of the G2 root system.
- `Circle`: the 2D circle.
- `Ellipse`: the 2D ellipse.

Note: `Circle` and `Ellipse` are flat quadrics. A single circle has neither parallels nor meridians to draw, so they only expose the `Mesh` tracing mode and the `Resolution` setting.

2.1.2. Polyhedra (3D)

- `Tetrahedron`: 3D simplex (4 vertices). The simplest regular polyhedron.
- `Cube` (Hexahedron): 3D hypercube (8 vertices). Regular hexahedron.
- `Cuboid` (rectangular parallelepiped): hypercube with edges of irregular length.
- `Octahedron`: 3D orthoplex (6 vertices). Dual of the Cube.
- `Dodecahedron`: regular polyhedron with 12 pentagonal faces.
- `Icosahedron`: regular polyhedron with 20 triangular faces.
- `Cuboctahedron`: Archimedean solid with 8 triangular faces and 6 square faces.
- `Rhomb. Dodeca.` (Rhombic Dodecahedron, or Granatohedron): Catalan solid (dual of the Cuboctahedron) with 12 rhombic faces.
- `Icosidodecahedron`: Archimedean solid with 20 triangular faces and 12 pentagonal faces.
- `Rhomb. Triaconta.` (Rhombic Triacontahedron): Catalan solid (dual of the Icosidodecahedron) with 30 rhombic faces.
- `Sphere`: the usual 3D sphere. Treated as a polytope for visualization.
- `Ellipsoid`: the 3D ellipsoid, a sphere stretched along the axes.
- `Torus`: torus of revolution. This is not a sphere: it has parallels, meridians and `Villarceau` circles, but neither hypermeridians nor shells.

Note: `Sphere`, `Ellipsoid` and `Torus` are quadrics and, as such, have the tracing settings described in 2.1.6.

2.1.3. Polychora (4D)

- `Pentachoron` (5-Cell): 4D simplex (5 vertices). The simplest 4D polychoron.
- `Tesseract` (8-Cell): 4D hypercube (16 vertices). The 4D analogue of the cube.
- `4-Box`: 4D hypercuboid (16 vertices). The 4D analogue of the cube with edges of irregular length.
- `Hexadecachoron` (16-Cell): 4D orthoplex (8 vertices). Dual of the Tesseract.
- `Icositetrachoron` (24-Cell): 4D polychoron with no 3D analogue. Self-dual.
- `Hecatonicosachoron` (120-Cell): 4D polychoron with 120 dodecahedral cells.
- `Hexacosichoron` (600-Cell): 4D polychoron with 600 tetrahedral cells.
- `Glome` (3-Sphere): 4D analogue of a sphere.
- `3-Ellipsoid` (4D Ellipsoid): 4D analogue of an ellipsoid.
- `Toratope`: the family of tori in dimension 4. This is where the family is at its richest: besides the Clifford torus (T^2, flat), it counts four members that are 3-manifolds in $\mathbb{R}^4$ with no analogue in any other dimension. The member displayed is chosen with `Torus Type` (see 2.1.6).

2.1.4. N-Polytopes (N > 4)

For all generalized polytopes, a `Dimension` slider allows choosing the number of spatial axes.

- `N-Simplex` : generalized simplex in N dimensions (N+1 vertices).
- `N-Cube` : generalized hypercube in N dimensions (2^N vertices). Coordinates are permutations of (±1, ..., ±1).
- `N-Box` : generalized hypercuboid in N dimensions (2^N vertices).
- `N-Orthoplex` : generalized cross-polytope in N dimensions (2N vertices). Coordinates are permutations of (±1, 0, ..., 0).
- `N-Demicube` : polytope obtained from a hypercube by removing every other vertex.
- `E•n Roots` (En roots): structures based on the vectors of the En root system (base for generating Gosset polytopes). Limited to dimension 8, since the exceptional root systems stop at E8.
- `Gosset K•21` : exceptional Gosset polytopes up to 8 dimensions.
- `N-Sphere` : spheres in N dimensions.
- `N-Ellipsoid` : ellipsoid in N dimensions.
- `N-Torus` : $T^n \subset \mathbb{R}^{2n}$, one family of circles per factor. The flat torus has no radial stratification, and therefore exposes no shells.

2.1.5. Special

This category gathers structures that belong to no regular family.

- `Elysion Vault (3D)` : pointed 3D shape obtained from 4 pyramids, used as a logo by Laurent Oliversen, creator of Hypersolid. Two specific settings, `Vault Gap X` and `Vault Gap Y` , modify the spacing between the pyramids.
- `Elysion Cell (3D)` : right pentagonal prism — the motif of the Elysion 120-Cell custom tiling. Two specific settings: `Cell Height` (height of the cell) and `Cell Gap` (spacing between cells).
- `Complex Hopf (4D)` : the Hopf fibration over $\mathbb{C}$, that is $S^3 \to S^2$. Built in dimension 4, whatever the position of the `Dimension` slider.
- `Quatern. Hopf (8D)` : the Hopf fibration over the quaternions $\mathbb{H}$, $S^7 \to S^4$. Built in dimension 8.
- `Octon. Hopf (16D)` : the Hopf fibration over the octonions $\mathbb{O}$, $S^{15} \to S^8$. Built in dimension 16.

Note: unlike the spheres that one can *trace* as a fibration, these last three shapes **are** the fibration. They are always built at the dimension of their algebra ($\mathbb{C}$→4, $\mathbb{H}$→8, $\mathbb{O}$→16).

2.1.6. Quadric Settings (spheres, ellipsoids, tori)

Quadric shapes — spheres, ellipsoids, tori and fibrations — are not described by a list of vertices but by families of curves. They share the following group of settings.

Tracing mode

- `Tracing Mode` : representation mode. The available bases depend on the shape:
  - `Mesh` : representation based on parallels, meridians and hypermeridians.
  - `Stratified` : nesting of lower-rank spheres; the interlocking is another way of representing the extra dimension.
  - `Hybrid` (dimension > 4): combines mesh and stratification.
  - `Hopf Fibration` : partition of the glome into interlaced great circles that do not touch. The entry is renamed after the fibration actually traced: `Hopf Fibration` in dimension 4, `Quaternionic Hopf` in dimension 8, `Octonionic Hopf` in dimension 16, and `Pseudo-Hopf` for ellipsoids — because it is then not a true fibration, but an a posteriori deformation of one.
- `Use Edges` : switches the representation from curves to edges. Each family is then built with edges and vertices, which allows vertex localization and color management, at the cost of the curvature of the circles.
- `Resolution` : number of vertices per circle. Controls the fineness of the tracing.

Circle families

Three sliders count the circles traced. **Their names change with the selected shape**, because what they count is not the same object:

Shape	Slider 1	Slider 2	Slider 3
Spheres, ellipsoids, glomes	`Parallels`	`Meridians`	`Hypermeridians`
`Torus` (3D torus of revolution)	`Parallels`	`Meridians`	`Villarceau`
Toratope / N-Torus — Clifford	`Ring 1`	`Ring 2`	`Extra Rings`
Toratope — Tiger	`Ring 1`	`Ring 2`	`Tube Circles`
Toratope — Spheritorus	`Parallels`	`Meridians`	`Hypermeridians`
Toratope — Torisphere	`Parallels`	`Meridians`	`Tube Circles`
Toratope — Ditorus	`Outer Ring`	`Inner Ring`	`Tube Circles`
$\mathbb{H}$ and $\mathbb{O}$ fibrations	`Fiber Density 1`	`Fiber Density 2`	`Fiber Density 3`

On the quaternionic and octonionic fibrations, the three sliders do not count distinct families: each one drives an arbitrary subset of the fiber directions, and the split changes with the `Fiber Mode` . The numbered labels are therefore the only honest ones.

Stratification

- `Shells` : number of shells (lower-rank spheres) making up the sections, for stratified modes.

- `Shell Distance` : distribution of the shells within the envelope. The larger the value, the more the inner shells pack against the outer shell; the smaller it is, the further the shells move apart from one another, contracting towards the center.

Hybrid mode (dimension > 4)

- `Suspension Depth` : dimensional stretching then polar contraction. This factor models the number of dimensions represented by an interconnection (hypermeridian) of glomes.
- `Glomes` : number of glomes per dimensional suspension.
- `Susp. Hyper Mode` : representation mode of the interconnections between the glomes of a suspension — `Radial` , `Peripheral` or `Helical` .

Hypermeridians and fibers

- `Hypermeridian Mode` : representation of the hypermeridians in `Mesh` mode — `Magnetic` , `Interlaced` , `Tilted` , `Equatorial` or `Polar` .
- `Alternate Hyper-Modes` : available for `Mesh` tracing in dimension > 4. Alternates the hypermeridian representation mode depending on the dimension.
- `Fiber Mode` : selection of the fiber sampling pattern, in Hopf mode. The modes offered depend on the shape and the dimension.

Tori

- `Torus Type` : member of the torus family to build. `Clifford` is flat and exists in every dimension; the other four — `Tiger` , `Spheritorus` , `Torisphere` , `Ditorus` — are 3-manifolds in $\mathbb{R}^4$ and are therefore only offered in dimension 4. Bringing `Tube Thickness` back to 0 on a Tiger yields exactly the Clifford torus: it really is the same family.
- `Tube Thickness` : thickness of the tube, expressed as the ratio r/(R+r) and normalized so that the figure keeps its overall size. The slider stops at the *horn* of the shape, that is, where the hole closes up.

Stretching

Stretchable shapes (ellipse, ellipsoids, boxes, tori) have the `Scale X` , `Scale Y` , `Scale Z` and `Scale W` settings, which stretch the figure in the desired dimensions. Beyond dimension 4, these are joined by:

- `Scale V Base` : stretching in the 5th dimension. Serves as a reference value for all dimensions > 4.
- `Scale V Factor` : exponential stretch factor in dimensions > 5.

Note: only quadrics traced in mesh mode (`Mesh`) are eligible for tessellation.

2.1.7. Graph Operators

Four operators transform the vertex/edge graph of the selected solid. Their eligibility depends on the exact shape: truncation requires a regular vertex figure, alternation a bipartite graph, and the compound a solid that is not centrally symmetric.

- `Truncation` : cuts each vertex at ratio t along its edges. 0 = original solid; 0.5 = rectified, that is, at the edge midpoints (for example cube → cuboctahedron). Reserved for exactly regular shapes.
- `Alternate` (Coxeter alternation): removes every other vertex — cube → tetrahedron, tesseract → 16-cell, n-cube → n-demicube. Takes precedence over `Truncation` .
- `Compound` : superimposes on the solid its central inversion (v → −v) — tetrahedron → stella octangula, 5-cell → compound of two 5-cells. Takes precedence over `Truncation` .
- `Positive Coords` : translates each axis so that its minimum sits at 0. The origin of the scene then lands in a corner of the solid instead of at its center (for example tesseract ±1 → 0/2).

To these are added two general settings of the `Generator` group:

- `Type` : category of shapes offered in the `Shape` dropdown — `All` , `2D` , `3D` , `4D` , `ND (5+)` and `Special` . The last two are only offered if the subscribed plan gives access to higher dimensions.
- `Solid Scale` : uniform scale of the solid.

2.1.8. Tessellation

2.1.8.1. Tiling Types

`Enable Tessellation` turns tiling on. To fill space by replicating polytopes, you can then choose between different arrangement modes through `Tiling Type` :

- `Prism`
- `Honeycomb` — offered only for eligible shapes
- `Swarm`
- The custom nets specific to the selected shape (see 2.1.8.5)

2.1.8.2. Prism

Prisms are defined by a base polytope replicated in a single direction. The corresponding vertices of each replica are connected to one another.

- `Nb Slices` : number of replicas
- `Flat Base` : the polytope serving as the prism base is flattened in the selected plane — `None` , `XY` , `XZ` or `YZ`
- `Tile Rotation` : each replica rotates in a plane orthogonal to the shear direction; rotation proportional to the distance to the origin

- `Rotation Factor`: amplitude of the rotation, as a function of the distance to the origin
- `Slice Shear X`, `Slice Shear Y`, `Slice Shear Z`: spacing between the replicas, along the X, Y and Z axes

2.1.8.3. Honeycomb

Only certain shapes are eligible for "honeycomb" tiling, such as cubes or hexagons. Unlike the other arrangements (prisms and swarms), these tilings are compact and fill space perfectly without gaps.

- `Nb Slices X`, `Nb Slices Y`, `Nb Slices Z`: number of slices in the X, Y and Z directions

2D tilings (of the XY plane) have two additional settings to enter the third dimension (disjoint sets of slices in the Z direction):

- `Gap Z`: spaces the tiling slices along the Z axis
- `Nexus Z`: connects the corresponding vertices of replicas belonging to different Z slices

2.1.8.4. Swarm

Swarms are a generalization of prisms and honeycombs. They are the most flexible and customizable. They consist of disjoint slices of replicas in all directions, which can be connected by nexus (edges linking the corresponding vertices of each replica).
In terms of settings, the swarm combines those of prisms and honeycombs:

- `Nb Slices X`, `Nb Slices Y`, `Nb Slices Z`: number of slices in all directions (same as honeycomb)
- `Nexus X`, `Nexus Y`, `Nexus Z`: a connection system in all directions
- `Gap X`, `Gap Y`, `Gap Z`: spacing between slices is managed in all directions

To these are added the settings specific to prisms: `Slice Shear X`, `Slice Shear Y`, `Slice Shear Z`, `Flat Base`, `Tile Rotation` and `Rotation Factor`.

2.1.8.5. Custom Nets

Some polytopes have predefined nets, offered in `Tiling Type` when the corresponding shape is selected. For example:

For cubes:

- `Tesseract Net`: 3D cross made of N-cubes

For the Elysion Vault:

- `Elysion Seal`: circular assembly of Vaults
- `Elysion 2-Seal`: double circular assembly of Vaults
- `Elysion 3-Seal`: triple circular assembly of Vaults

2.2. Solid Style

The application uses an automatic coloring system to help identify dimensions.
Edges are colored based on their alignment with the main axes in the original high-dimensional space.
By default:

- X: Green
- Y: Blue
- Z: Orange
- W: Purple
- V5, V6, V7, V8, ...: Gray

Note that these colors are determined systematically for structures with edges parallel to the frame axes. This therefore applies to hypercubes. For figures with edges inclined in several dimensions, color management is more subtle.

The stylistic settings are organized into four folders: `Vertices`, `Edges`, `Faces` and `Background`.
Depth effects do not form a folder of their own: each family of objects carries its own, following its style settings.

2.2.1. Vertices

2.2.1.1. Basic Style

- `Show Vertices`: shows or hides the vertices.
- `Vertex Info`: makes the vertices clickable. A click opens the point information panel (see 6.5).
- `Vertex Size`: size of the vertices.
- `Vertex Shape`: shape of the marker — `Dot`, `Square`, `Diamond`, `Cross (+)` or `Cross (x)`.
- `Vertex Color`: color of the vertices.

2.2.1.2. Vertex Depth

Activating depth effects is done by checking the `Vertex Depth` box. You can then control the opacity ("Alpha") of geometric elements based on their depth in 4D/3D space. These commands are located just below the `Vertex Depth` box. Depending on the drawn object, depth is evaluated differently:

- `Depth Quantiles`: Discretizes distances from the origin into a set of bins (e.g., 10 quantiles), rendering them in distinct alpha layers via an X-Ray approach to avoid visual collisions (Z-Fighting).
- `Outer Alpha`: Base opacity for the outermost elements of the objects.
- `Inner Alpha`: Base opacity for the innermost elements (the core) of the objects.

Note: The system automatically creates a linear opacity gradient between outer and inner elements. If Outer Alpha < Inner Alpha, the gradient is disabled and the Outer Alpha value is applied uniformly to all depths.

- `Invert Alpha` : a mathematical toggle that inverts the opacity. Useful for hiding the exterior to observe only the solid core.
- `Alpha Cutoff` : a strict rendering threshold applied after the final alpha (and inversion) calculation. Any element whose final opacity falls below this value is entirely removed from the rendering pipeline (hidden), improving visual clarity and rendering performance.

2.2.2. Edges

Edges have the following settings:

- `Show Edges` : shows or hides the edges.
- `Edge Info` : makes the edges clickable, on the same principle as `Vertex Info` .
- `Edge Width` : thickness of the stroke.
- `Edge Default Color` : single color applied when per-axis coloring is disabled.

The `Edge Colors` checkbox allows defining a color per edge type. If unchecked, all edges carry the color defined by `Edge Default Color` .
If checked, the following setting allows managing colors for edges whose orientation is not parallel to the axes.

2.2.2.1. Color Tie Break

You can define the 5 colors associated with each frame axis:

- `X Axis` , `Y Axis` , `Z Axis` , `W Axis` , `Vi Axes (i > 4)` .

Depending on the object, these colors are used differently (typically for spheres).

The `Color Tie Break` list:
When an edge is not perfectly aligned with a single axis (e.g., the diagonal edges of a triangle), the color is determined by a tie-breaking rule (choice between 3 rules):

- `Vertex` : Color based on the index of the start/end vertex.
- `Edge` : Color based on the index of the edge in the list.
- `Hash` : Consistent random color based on the properties of the edge.

This rule applies to most polytopes. The results are hard to anticipate, but they generally allow distributing colors in different ways.

For spheres, other techniques are available:
For spheres/ellipsoids in Mesh/Stratified Edges mode:

- `Circle`: only; reproduces the same color-by-circle-type system used by Mesh/Stratified Circles tracings.

For spheres/ellipsoids in Hopf Fibration mode (complex, quaternionic, octonionic):

- `Fiber`: assigns a color per circle topology (associated with different imaginary numbers).
- `Fano`: coloring derived from the Fano plane, which encodes the multiplication table of the octonions. Reserved for the octonionic fibration.
- `Polar`: assigns a gradient of 3 colors (from the Axis X/Y/Z colors) to distinguish circles close to the north pole, south pole, and equator.
- `Bipolar`: assigns two gradients of 3 colors (from the Axis X/Y/Z/W/V colors) to distinguish circles close to the north pole, south pole, and equator.
- `Tripolar`: same with three gradients.
- `Quadripolar`: same with four gradients.

2.2.2.2. Sphere Outlines

Since spheres are generated differently, they use a specific coloring logic depending on their `Tracing Mode`:
Mesh Edges/Circles case: the mesh colors are defined by the colors in the Edges section:

- Parallels: color defined by `X Axis`,
- Meridians: color defined by `Y Axis`,
- Hypermeridians: color defined by `Z Axis`.

Hopf Fibration case
This advanced visualization uses the Hopf map to project the 3-Sphere (S^3) as a bundle of circles (called "fibers") over a 2-Sphere (S^2).

- Base: A 2-Sphere where each point corresponds to a full circle (fiber) in the 3-Sphere.
- Fibers: Disjoint circles filling 4D space. A scale of 12 fibers allows visualizing the structure without clutter.
- Coloring: complex Hopf fibers (4D) are colored based on their position on the base 2-Sphere (longitude/latitude), revealing the twisting structure. They use only the colors defined by `X Axis` and `Y Axis`. Quaternionic and octonionic Hopf fibers use color systems based on circle topology (either related to the mathematical structure or to their proximity to the poles).

Stratified Mode (equator & poles)
In this mode, the 3-Sphere is decomposed into a series of parallel 3D shells (a collection of classical spheres), in the same way a globe is sliced by latitude lines. The colors depend on the shell's position in the nesting.

- *Equator (Outer) Shell (outer equatorial shell = widest geometry)*: The "surface" of the 4D sphere. Represented by *X/Y*. Geometrically, this is the largest shell. The parallels and meridians of this sphere use the colors defined by `X Axis` and `Y Axis`.
- *Pole (Inner) Shell (inner polar shell = smallest geometry)*: The "core" of the 4D sphere. Represented by *Z/W*. Geometrically, these are points or very small circles. The parallels and meridians of this sphere use the colors defined by `Z Axis` and `W Axis`.
- *Intermediate shells*: the colors of non-extreme shells are interpolated between the previous two: this is therefore a gradient between the inner and outer shells.

Note: stratification artifact in perspective mode
Why do inner shells sometimes appear larger than outer shells?

When using Perspective Projection, objects closer to the 4D camera appear significantly larger. If the `4D Distance` is small (camera close to the object):

- An *intermediate shell* (which is geometrically smaller than the equator) happens to be closer to the camera.
- The perspective magnification can make this inner shell appear *visually larger* than the outer equatorial shell, potentially to the point of "swallowing" it.

The colors remain correct (as they are based on rank):

- The shell colored in *X/Y* is always the geometric equator (Rank 1).
- The shell colored in *Z/W* is always the geometric pole (Rank N).

If you see a Z/W shell appearing larger than an X/Y shell, it is a pure optical illusion caused by 4D perspective. Solution: modify the *4D Distance* or switch to *Orthographic Projection* to restore the intuitive visual hierarchy.

2.2.2.3. Edge Depth

Activating depth effects is done by checking the `Edge Depth` box.

Edges have the same settings as vertices to manage depth: `Depth Quantiles`, `Outer Alpha`, `Inner Alpha`, `Invert Alpha`, `Alpha Cutoff`. See the vertex depth settings for more details.

2.2.2.4. Depth in Spheres

A distinction is made between edges (straight lines) and shells (curved).

Objects with edges (polytopes and spheres using the Mesh Edges tracing mode)
Opacity is calculated by vertex/edge distance relative to the origin *in the projected 3D space* (not the original 4D space). Distances are sorted and divided into *N* quantile bins. Each bin receives an alpha value linearly interpolated between Outer Alpha and Inner Alpha:

Note on the depth of the 2/3-Sphere:
Even though all points of a 3-sphere are equidistant from the 4D origin by definition, the 4D→3D projection breaks this isotropy. In *perspective* projection, circles close to the W-pole (large W) are compressed towards the 3D origin, while equatorial circles (small W) are projected further away. In *orthographic* projection, a parallel near the north pole lies close to the 3D origin, while an equatorial parallel lies at a unit distance (the largest). The depth gradient on a Glome in *Mesh Edges* mode is thus real and physically significant: it reflects the 4D depth structure as seen through the projection.

Spheres with shells (Stratified Edges and Stratified Circles)
Depth is structurally evaluated based on the mathematical index of the shell (from 1 for the geometric equator to N for the pole). Since shells are naturally nested topologies, the alpha gradient is continuous and exact. Note: the `Depth Quantiles` setting does *not* apply to shells.

Hopf Fibration: shell-based depth (without edges)
The Hopf Fibration is rendered as a set of topologically nested *fiber circles* (closed loops), rather than discrete edges between vertices. Each circle belongs to a mathematical shell (a torus layer), numbered from the outermost torus to the innermost central fiber. The depth alpha is computed directly from the shell index; no 3D Z-distance sorting is needed. The formula is the same as the one used for Stratified/Shell objects. Since Hopf fibers are generated in a strict topological order (exterior → interior), the mathematical structure inherently encodes depth. This makes depth management exact, continuous, and perfectly consistent with the Inner/Outer Alpha controls, without any geometric heuristics.

Limitations - Mesh Circles only
In the *Mesh Circles* tracing mode for the 2/3-Sphere, all circle geometry is merged into a single combined mesh without a per-circle depth shader. Depth alpha does *not* apply to circles, only to vertices. In contrast, the *Mesh Edges* mode for Spheres *does support* depth because a GPU shader evaluates depth = distance per vertex at render time.

2.2.3. Faces

The `Show Faces` toggle overlays filled polygons onto the polytope's wireframe. The engine supports two fundamental modes for detecting and generating faces:

2.2.3.1. Face Detection Methods

The application uses specialized algorithmic generators to identify the true geometric faces of the polytope. The method used depends on the polytope family. The techniques used are:

- **Graph Cycle Detection (DFS)**: used for standard 3D/4D regular solids, the *N-Simplex* and the *N-Cube* (up to *N=5*). A Depth-First Search explores the edge graph to identify closed loops of 3, 4, or 5 edges.

- **Axes Combinatorial Search**: specific to the *N-Orthoplex* (N-Cross) family. It identifies triplets of vertices located on distinct orthogonal axes to form perfectly regular triangular faces. This mode is active up to *N=6*.
- **Parity Subspace Logic**: specific to the *N-Demicube* (*Dn Group*) family. It identifies the 3D cubic subspaces of the parent hypercube and extracts the tetrahedral faces that preserve the alternating parity of the vertices. This mode is active up to *N=5*.
- **Stochastic Sampling (in other cases)**: for complex exceptional polytopes (Gosset *E6, E7, E8*) or any object exceeding the dimension thresholds mentioned above, topological detection is automatically bypassed in favor of random sampling to maintain interactive performance.

2.2.3.2. Face Rendering

- `Show Faces`: shows or hides the faces.
- `Face Type`: rendering mode.
  - `Solid`: faces are triangulated and rendered as filled surfaces.
  - `Wireframe`: only the border edges and internal diagonals of the identified faces are drawn. This reveals the "tiling" of the polytope.
- `Wire Width`: thickness of the strokes, in `Wireframe` mode.
- `Face Color`: color of the faces.
- `Face Alpha`: transparency of the faces, when depth effects are disabled.

2.2.3.3. Face Sampling

For complex or high-dimensional objects where topological detection is too costly, the engine can stochastically sample pairs of vertices to form triangles (or quadrilaterals/pentagons).
In the following paragraph: a face is said to be *small* (respectively *large*) when the vertices defining it are, on average, at a distance *less than* (respectively *greater than*) a given value. The notion of distance refers to the Euclidean distance between a vertex and the polytope origin (the barycenter in most cases).

Sampling is enabled with `Face Sampling`. The associated settings are:

- `Large/Small Cutoff`: this parameter is a normalized distance used to classify faces into two categories: small or large. This threshold is between 0 and 1 because it represents the percentage of the largest distance observed among all faces. For example, a threshold of 0.5 means that the vertices of a face must be, on average, at a distance (to the origin) *greater than* (respectively *less than*) 0.5 of the maximum distance (the face considered "furthest") for the face to be considered *large* (respectively *small*).
- `Large Faces`: number of faces classified as *large* that will be retained (sampled) for coloring.
- `Small Faces`: number of faces classified as *small* that will be retained (sampled) for coloring.

Note for Regular Polytopes: For perfectly regular polytopes (Cubes, Tesseracts) where all vertices are equidistant from the barycenter, the "Small" population is mathematically empty because the normalized distance threshold is constant. In this case, only "Large" face sampling will produce results.

2.2.3.4. Face Depth

Activating depth effects is done by checking the `Face Depth` box.
Z-Sorting & Alpha (Painter's Algorithm)
To apply depth to faces, the engine dynamically reorganizes the faces from back to front at each frame. This enables depth-based transparency gradients:

- `Front Face Alpha` : opacity of the faces closest to the camera (the "skin").
- `Back Face Alpha` : opacity of the faces furthest from the camera (the "core").
- `Invert Face Alpha` : inverts the gradient, on the same principle as `Invert Alpha` for vertices and edges.

Example usage: setting `Back Face Alpha` = 1.0 and `Front Face Alpha` = 0.2 allows you to see the solid interior of the structure through a translucent outer layer.

Coverage by polytope type:
✅ Platonic Solids (3D): triangles, squares, pentagons
✅ Archimedean / Catalan (cuboctahedron, rhombic dodecahedron): mixed polygon types
✅ 4D Polychora: via graph cycle detection
✅ N-Simplex / N-Cube / N-Orthoplex: up to available resolution
❌ Sphere / Glome: Not applicable, rendered as circles
✅ Gosset K.21 dim=5 (D5): triangles + squares, manageable count
❌ Gosset K.21 dim≥6 (E6, E7, E8): Gosset polytopes E6, E7, E8 possess a number of 2-faces (planar faces) that grows exponentially. For E6, face calculation is fast, but projecting 2160 overlapping faces from 6D creates an unreadable opaque mesh with no visual benefit. For E7 and E8, the calculation would take several seconds and the result is just as unreadable. Only D5 (dim=5, ~40 vertices) exposes its faces, as a practical compromise between visual information and performance.

2.2.4. Background

This section allows displaying background or overlay elements.

2.2.4.1. Background Color and Gradient

- `Transparent` : removes the background. The scene is then rendered on a transparent background, which is useful for PNG exports meant to be overlaid.
- `Color` : solid background color, used when `Use Gradient` is unchecked.
- `Use Gradient` : replaces the solid color with a three-stop gradient.

The gradient is then adjusted with:

- `Gradient Type` : `Linear` or `Radial` .
- `Color 1 (Top)` , `Color 2 (Mid)` , `Color 3 (Bottom)` : the three colors of the gradient, from top to bottom.
- `Gradient Angle` : orientation of the linear gradient.
- `Mid Scale` : extent of the intermediate color. The larger the value, the more room the central band takes at the expense of the two extremes.
- `Mid H Position` / `Mid V Position` : horizontal and vertical position of the center of the intermediate color.
- `Smoothing` : softens the transitions between the three stops.

2.2.4.2. Halo

The halo is a diffuse light source painted into the background, independent of the gradient. It serves to detach the solid from the background without lighting the 3D scene.

- `Halo` : enables the halo.
- `Halo Color` : color of the halo.
- `Halo Intensity` : light intensity.
- `Halo Scale` : radius of the halo.
- `Halo H Position` / `Halo V Position` : horizontal and vertical position of the center of the halo.

2.2.4.3. Alignment Grid

The `Alignment Grid` option displays a grid tangent to the peripheral vertices, useful for adjusting alignments during rotations, for example.

2.2.4.4. Reference Axes and Planes

Location: `Workspace` tab, `Exp` section, `Scene Markers` folder. These markers describe the scene rather than the solid, but they accompany the geometry as well as the projected data.

Axes
Axes help with orientation in the spatial dimensions.

- `Show Scene Axes` : displays the axes of absolute 3D space (as seen by the camera), |X|, |Y|, and |Z|. They are enclosed in vertical bars to avoid confusion with the relative axes below.
- `Show Relative Axes` : displays the relative axes, which are bound to the polytope and any projected data points. When a shear or rotation is performed, these axes shift and rotate along with the polytope and the data. Note that they no longer represent absolute spatial coordinates, but rather the coordinates after projection/transformation: X, Y, Z, W, V5, V6, ..., V10.
- `Negative Extension` : extends the axes to the negative side as well.
- `Negative Arrow` : adds an arrow tip on the negative side of the axes.
- `Axis Length` : length of the axes (arrows).

- `Axis Label Size`: size of the labels (X, Y, Z...).
- `Axis Arrow Size`: size of the axis arrowheads.
- `Axis Ticks`: displays unit tick marks (coordinates 1, 2, 3...) for the absolute scene axes and the relative data/polytope axes. The ticks differ per axis, since relative coordinates are transformed by spatial projection.
- `Axis Grid`: adds a grid in the planes carried by the axes, to read coordinates at a glance.
- `Axis Projection`: draws the orthogonal projection of each point onto the axes, as guide lines.

Planes

- `Plane |XY|`, `Plane |XZ|`, `Plane |YZ|`: displays the planes of the absolute frame (scene axes).
- `Plane Alpha`: opacity of the displayed planes.
- `Plane Occlusion`: the planes hide what lies behind them, instead of remaining translucent.
- `Plane Back Fade`: dims the back face of a plane when the camera moves to the other side, so that the plane does not conceal the scene.
- `Back Alpha`: residual opacity of that dimmed back face.

2.3. Projection

Projection is performed in this order:

- Projection of dimensions higher than 4 to the 4D space (X, Y, Z, W)
- Projection of the 4D space to 3D (the X, Y, Z coordinates incorporate the remnants of the W coordinate through perspective effects)
- Optionally, a collapse (orthographic projection) from 3D to 2D
- The whole is displayed in a 3D scene, through the camera

In the interface, settings are categorized by type of mathematical operation: rotation, shear, projection (with focal distance). But conceptually, dimensions are processed in cascade.

2.3.1. Views

This section allows applying specific mathematical visualizations. These rotation values reveal characteristic projections for each polytope.
The views below capture the following elements:

- Topological: Connectivity graph (vertices, edges)
- Combinatorial: Incidences between k-faces
- Metric: Dihedral angles, relative lengths
- Global: Periodicity, envelope, stratification

2.3.1.1. Unified Projection Mode

The three buttons below allow setting all projection modes (ND-4D, 4D-3D, and Camera) at once:

- `All Perspective` : unifies the entire pipeline (ND Projection, 4D Projection, and Camera) in *Perspective* mode. This guarantees a consistent focal depth throughout dimensional reduction.
- `All Ortho` : unifies the entire pipeline in *Orthographic* mode, revealing true geometric proportions without distortion.
- `Ortho-Perspective` : mixed mode. The dimensional reductions (ND→4D and 4D→3D) remain orthographic — the proportions of the polytope are therefore preserved — while only the 3D camera switches to perspective. The result is a scene that reads well in depth without the structure of the polytope being distorted by two successive stages of vanishing points.

2.3.1.2. Front and Petrie Projection

These two projections are respectively the simplest and the most complex to implement. They have a dedicated button for immediate access.

- `Front` : resets all rotation angles and shears (ND and 4D) to 0, and restores the camera to its default frontal orientation. This is the starting point of any visual analysis.
- `Petrie` : Configures rotations and projection to show the *Petrie Polygon* of the polytope. This is a specific orthographic projection where the projected polygon of the polytope forms a regular N-sided perimeter. The axis is aligned with the polygon that forms the longest Hamiltonian cycle zigzagging between all facets. This is the only view that reveals the global structure: it shows large-scale periodicity, which is invisible in local views. It remains unique in that it captures Hamiltonian periodicity, with no direct equivalent elsewhere.

2.3.1.3. Canonical Planes

Navigate between the different canonical planes by clicking the `◀ Prev` (previous) and `▶ Next` buttons. They include:

- `Vertex-first` : the axis passes through a vertex. Reveals local valence and the point symmetry group: how many edges radiate, and with what angular symmetry.
- `Edge-first` : the axis is aligned with an edge. Reveals the dihedral angles between the two (N-1)-facets sharing this edge - the "local curvature" of the object. Often overlooked, but irreplaceable for distinguishing polytopes with similar vertex-first views. Note: an edge bounds exactly 2 facets of codimension 1 in a convex regular polytope; in general this number depends on the local structure.
- `Cell-first` / `k-Facet-first` : the axis passes through the center of a cell or of a facet of dimension 3, 4, ..., N-1. Reveals concentric stratification: how cells/facets organize in successive layers around the center ("onion-like").
- `Schlegel` : corresponds to an (N-1)-facet-first view in perspective mode. It allows seeing all cells at once.

- `Equatorial` : corresponds to a Petrie projection followed by a 90° orbit in the vertical plane, as if looking at the Petrie edge-on. It allows seeing the stratification of facets around the barycenter.

Note: with the exception of the Schlegel view, all canonical plane views are only possible in orthographic projection mode.

2.3.1.4. Base Permutations

Permuting the base axes is particularly useful in factor analysis, to visualize the distribution of points along 3 simultaneous axes among the N available axes (X, Y, Z, W, V5, V6...). See the factor analysis section for more details.

Each pre-recorded view corresponds to a 90° rotation along one, two, or three axes. This visually allows replacing a base axis (X, Y, Z) with another available axis.
The 20 most common combinations (up to dimension 6) are offered. The first corresponds to the XYZ space view. The last translates the W.V5.V6 space view.

Navigate between the different factor spaces by clicking the `◄ Prev` (previous) and `► Next` buttons.

To better understand the rotations, it is recommended to display the relative axes (`Show Relative Axes`) which show the orientation of the polytope and of the overlaid data.

2.3.1.5. Collapse

Optional 2D/1D flattening step.
Dedicated to the Factor visualization tab, this final mathematical step allows compressing the fully projected 3D scene (Polytope, Zodiac/Astro/MBTI/Social data, and custom data) to a 2D plane or a 1D axis. This is accomplished by forcing one or two dimensions of the 3D output to zero immediately before rendering.

Plane Projections (2D):

- `XY` : Emulates a flattened view by flattening depth (z = 0).
- `XZ` : Flattens the vertical axis (y = 0).
- `YZ` : Flattens the horizontal axis (x = 0).

Axis Projections (1D):

- `X` , `Y` , `Z` : Flattens two dimensions simultaneously to project the entire structural cluster onto a single line. This is particularly useful for identifying correlations, variance, or standard deviations along a specific principal component.

Note: Since this projection is applied as a final filter on the 3D scene rather than in the original mathematical space, higher dimensions like W (4D depth) or V (5D+) cannot be isolated here - they are already folded into X, Y, and Z by the previous steps. Additionally, to maintain readability when points are mathematically collapsed onto identical spatial coordinates, text labels (Zodiac/Astro/MBTI/Social and custom) apply an intelligent spatial offset (derived from their original lost coordinate) to avoid visual overlapping.

2.3.1.6. Custom Views

Certain solids feature custom views (which can be modified). Navigate between views by clicking the `◀ Prev` (previous) and `▶ Next` buttons.

2.3.2. ND to 4D

This step reduces dimensions N > 4 to 4D (X, Y, Z, W).
The cascading pipeline of all higher dimensions is executed as follows:

- Initial rotation in all selected ND planes
- Chain of shears and projections (at each level of the dimensional cascade)

Note: by its non-destructive additive logic, shearing is much more effective than rotation for unfolding higher dimensions into the visible 3D space (projected on the first three axes X, Y, Z). This is because rotations in high dimensions can occur in planes orthogonal to the projection plane of each dimension to the next, compressing any visual unfolding attempt.

The `Lock Solid Rotation` checkbox allows locking the polytope and making it insensitive to rotations (note: the solid remains sensitive to shears and to the focal distance of projections). This is useful for analyzing data, revealing through rotation the relative axes of invisible dimensions (W, V5, V6...) without affecting the solid itself. If a cube or a sphere is displayed as a reference, it remains stationary during rotations, while the projected data (point cloud) continues to pivot with the relative axes.

2.3.2.1. ND Rotation

- `Circular Rotation`: performs a simultaneous rotation in XV5, YV6, ZV7, WV8, V5V9, V6V10.
- `Rotation Factor`: Controls the exponential decay of rotations for higher dimensions.
- `Rotation Plane` which contains a list of all additional rotation planes. Simply select one, then adjust the slider that appears just below it.

2.3.2.2. ND Shear

- `Circular Shear`: Adds a rotational geometric component (alternating directions along X, Y, Z) to the linear shear of 4D, creating spiral patterns for dimensions higher than 4.
- `Shear Factor`: Controls the exponential decay of shear offsets for higher dimensions.
- `Shear Plane`: selection of a specific plane to perform precise shearing along that plane only.

2.3.2.3. ND Projection

The mode is chosen with the `ND -> 4D` dropdown.

Perspective

Higher dimensions (D > 4) are scaled by a perspective factor. Objects with higher coordinates in these dimensions are projected closer to the 4D center, while applying a cumulative shear.

- `Base ND Distance`: determines the reference distance (from the camera) for the "perspective" of higher dimensions. The higher the value, the closer the outer layers are to the origin (distance from the camera increases, perspective decreases).
- `ND Distance Factor`: adjusts the exponential scale of the perspective distance across dimensions. A high value results in uniform spacing between dimensions. A low value causes wider spacing of higher dimensions.

By manipulating `Base ND Distance` and `ND Distance Factor` together, it is possible to customize the global perspective, as well as the spacing or clustering between dimensions (compression/dilation feel).

Orthographic

Parallel reduction to 4D. The perspective distance and depth accumulation are strictly ignored (W remains unchanged). The structure is revealed purely through linear shear. Focal distance concepts are ignored. Only the shear parameters remain to unfold the structure.

Oblique

Oblique projection is not offered in dimensions higher than 4 because it becomes very complex to configure. An orthographic projection accompanied by a rotation or a shear (circular here) is equivalent to an oblique projection.

2.3.3. 4D to 3D

The projection step aims to project 4D points (x, y, z, w) into the 3D space (x', y', z'). The W coordinate represents the "4D depth".
The processing begins with an optional linear 4D shear (`Shear` folder, `XW`, `YW`, `ZW` sliders) to the true 3D coordinates, based on the W depth. This allows tilting the 4D object into our physical dimension before the final pass of the perspective or orthographic camera.

2.3.3.1. 4D Rotation

Polytope rotation can be performed in all 3D and 4D planes. The `Rotation` folder exposes one slider per plane: `XY`, `XZ`, `YZ`, `XW`, `YW`, `ZW`. The last three only appear from dimension 4 onwards.

Note: the names of these sliders follow the active axis layout (see 4.3.3). Under a layout that renames the axes, `XY` becomes the corresponding pair.

2.3.3.2. 4D Shear

Polytope shearing can be performed in all 4D planes. The `Shear` folder exposes one slider per plane: `XW`, `YW`, `ZW`. Each one shifts the 3D position based on the 4D depth (W).

2.3.3.3. 4D Projection

Perspective
Standard 4D perspective. Objects with a higher w (further in 4D) appear smaller and converge towards the center.
The mode is chosen with the `4D -> 3D` dropdown. The `4D Distance` adjusts the focal length of the 4D camera. Smaller values create stronger perspective distortion.

Orthographic
Projection parallel to W. The W coordinate is completely ignored for point positions in 3D. Useful for preserving real geometric sizes.
Important: in the absence of rotation or shear, it therefore becomes impossible to visualize the 4D components. You should use the rotation and shear sliders to unfold the structure.

Oblique
Unfolds the 4D structure by shifting points based on their W coordinate along a specific 2D angle on the screen. Geometrically, oblique projection is equivalent to an orthographic projection combined with a shear, but it is offered as a distinct simplified mode, typically to reproduce Cavalier/Cabinet perspectives.

- `4D Angle`: controls the direction of the 4D shift in the 3D XY plane.
- `4D Scale`: controls the amplitude of the shift (how far the 4D depth is "extracted").

Note: manual `XW` / `YW` / `ZW` shearing is ignored in 4D Oblique mode since the oblique angle and scale fully define the 4D shear. The shears are therefore redundant.

2.3.4. Camera

The final phase renders the 3D wireframe on your 2D screen. Note that the 2D/1D projection in the Factor tab is performed before the camera rendering. This is why we can still orbit around the polytope and the collapsed points.

2.3.4.1. Camera Type

The type is chosen with the `Type` dropdown of the `Camera` folder.

- `Perspective`: standard 3D camera with vanishing points. Objects far from the camera appear smaller.
- `Orthographic`: parallel projection. Objects keep their size regardless of distance. Ideal for technical views.

- `Oblique` (cavalier/cabinet): a special mode combining orthographic projection with Z-axis (depth) shear. Oblique provides a cavalier perspective where the front face retains its true shape, while depth is represented by diagonal lines. Two additional parameters appear:
- `Angle` : direction of the axis simulating depth.
- `Scale` : amplitude of the shift simulating depth.

2.3.4.2. Viewport

Viewport

- `Roll` : rotates the entire view (camera roll) around the line of sight.

Note: there is no camera shearing. This would introduce unwanted tilting to the polytopes.

2.3.4.3. Camera Monitoring

These monitors are located at the bottom of the `Camera` folder and display the current camera coordinates:

- `Orbit Phi` : orbit in the horizontal plane.
- `Orbit Theta` : orbit in the vertical plane.
- `Zoom` : current zoom level.

This data is purely informative (non-editable). To modify the camera, simply use the mouse or trackpad.

The folder finally exposes a diagnostic toggle:

- `Polytope Logs` : writes to the browser console the details of the generation and projection of the current polytope (number of vertices, edges, faces, angles applied). Useful for reporting unexpected behavior.

2.4. Motion

Only for the Hyper plan.
The application provides tools to visualize high-dimensional geometry through motion.

2.4.1. Dynamic Effects

This folder gathers the "living" effects of the scene: pulsing glow on the selected point, valence glow, and animated camera moves.

- `Dynamic` : master switch for these effects. Unchecked, the scene stays static and camera moves are instantaneous.
- `Pulse Speed` : speed of the light pulse on the selected point.

- `Pulse Radius` : maximum radius of the pulse, expressed as a multiple of the size of the selected point. The minimum radius is the point size itself.

2.4.2. Orbit (automatic camera moves)

Some actions move the camera on their own: a double-click on a point, arriving on a canonical plane, a change of base permutation. This folder decides what triggers these moves and how they unfold.

Triggers

- `Best Plan Selection` : a double-click on a point (and on the intermediate steps of `Mid Point States`) rotates the axes towards its *Best Cos² Plan*. Can be combined with `Magnet Selection` ; with both unchecked, the double-click does nothing any more.
- `Magnet Selection` : orients the camera towards the selected point, centering it as closely as possible. When an arrival on a *Best Plan* occurs, this setting replaces the front view; on its own, the double-click simply brings the camera onto the point.

Trajectory of the moves

By default an automatic move goes straight to the target. The settings below add motion to it, to make the transition legible rather than abrupt.

- `Orbit Adjustment` : master switch for the six sliders below. Unchecked, no extra path is added.
- `Pre-Motion Phi` : extra phi path (in degrees) travelled *before* the move settles. 360 corresponds to a full turn. The sign is drawn at random (±) at each gesture.
- `Pre-Motion Theta` : same for theta.
- `Pre-Motion Zoom` : zoom kick applied along the way, before settling. Positive = closer, negative = further away.
- `Post-Motion Phi` : orbit offset (in degrees) added *after* the move, so that the arrival is not perfectly centered. Sign drawn at random.
- `Post-Motion Theta` : same for theta.
- `Post-Motion Zoom` : adjustment of the arrival zoom.

Scope

The `Scope` subfolder designates the gestures to which these extra paths apply:

- `Best Projection Plan` : the *Best Contrib* / *Best Cos² Plan* buttons.
- `Canonical Plan` : the `◄ Prev` / `► Next` buttons of the canonical planes.
- `Base Permutation` : the `◄ Prev` / `► Next` buttons of the base permutations.
- `Double-Click` : the double-click on a point, and the intermediate steps.

Note: these extra paths are only visible in `Dynamic` mode.

2.4.3. Auto-Rotate

Automatically rotates the polytope in up to three simultaneous planes.

- `Auto-Rotate` : enables automatic rotation.
- `Rotation Speed` : controls the angular velocity. Supports negative values to reverse the direction.
- `Rotation Plane 1` , `Rotation Plane 2` , `Rotation Plane 3` : the rotation planes to activate (XY, XZ, XW, YZ, YW, ZW, or the ND planes for N > 4). The list offered depends on the dimension of the current solid.

2.4.4. Auto-Shear

Oscillates the shear factors harmonically to make higher dimensions "pulse" in the view.

- `Auto-Shear` : enables automatic shearing.
- `Shear Amplitude` : amplitude of the oscillation.
- `Shear Speed` : frequency of the oscillation.
- `Shear Plane 1` , `Shear Plane 2` , `Shear Plane 3` : the shear planes to activate (XW, YW, ZW, or the ND planes for N > 4).

2.4.5. Interpolation

Performs a smooth transition between global state presets.

- `Interpolate` : enables the interpolation chain.
- `Start State` : the initial state of the transition — `Front` or `Petrie` .
- `Mid State` : an optional intermediate state — `None` , `Front` or `Petrie` . Set to `None` , the transition goes directly from Start to End.
- `End State` : the final target state — `Front` or `Petrie` .
- `Mid Point States` : list of point labels, comma-separated (matching by inclusion). Each point becomes an intermediate step: the camera rotates the axes to center it, exactly like a double-click. These steps are inserted into the chain.

Interpolation chain

Clicking `Interpolation` first moves the current view to `Start State` , then to `Mid State` (if enabled) and to any steps of `Mid Point States` , before settling on `End State` .

Pace of the transition

- `Ease In/Out` : enables a smooth transition, on an "S-curve", rather than a constant speed.
- `Duration (s)` : duration of each segment of the transition.
- `Start Pause (s)` : dwell time on the start state before setting off again.
- `Mid Pause (s)` : dwell time on each intermediate step.

- `End Pause (s)`: dwell time on the arrival state.

2.4.6. Oscillation

`Oscillate`: if oscillation is enabled, the interpolation does not stop at `End State`. It reverses the sequence and loops indefinitely between Start, Mid and End. The start, mid and end pauses apply on each pass.

2.5. A Few Projection Examples

2.5.1. Tesseract

Tesseract
The tesseract has a large number of famous projections.
The *two centered hexagons* view (available in the `Custom Views` of the Projection panel):

- Rotation: (XY, XZ, YZ) = (30°, 35.26°, 135°)
- 4D perspective projection with an orthographic camera projection
- Use the 4D distance to adjust the spacing between the two hexagons

The *double oblique perspective* view (available in the `Custom Views` of the Projection panel):

- 4D oblique projection with an oblique camera projection

3. Data Ingestion

3.1. Dataset

Location: `Workspace` tab, `Dataset` section.

3.1.1. Synthetic Data

These datasets are random or open-source. They allow you to test the application without having to import data.

- `5D Data` : data with 5 pseudo-random numerical variables
- `12D Data` : data with 12 pseudo-random numerical variables
- `Titanic Data` : open-source data
- `Titanic Categ. Data` : same as above, but with discretized data; allows testing MCAs
- `Titanic Mixed Data` : same as above, but with mixed data; allows testing FAMDs
- `MMM Data` : data to test the forecast and the MMM
- `Hy-Perf Single Data` : data to test the Hy-Performance module, with a single effect
- `Hy-Perf Multi Data` : data to test the Hy-Performance module, with multiple effects
- `Clear Imported Data` : empties the current dataset and returns the scene to its initial state

3.1.2. Reference Data

These reference datasets are purely cosmetic and serve as visual landmarks. Each one is a drop-down menu whose first entry, `None` , turns the display off.

- `Zodiac Data` : `Astro` (the 12 signs), `Malgovert` (the 13 psychic clusters from the eponymous novel), `Elysion 120-C Loop` and `Elysion 120-C H3`
- `Semiology Data` : `12 Circuits` and `6 Circuits` , the neurological circuits that condition behavior, as well as `HBDI` (Herrmann's four brain quadrants)
- `MBTI-DISC-SS Data` : the behavioral frameworks used in business — `MBTI (3D)` , `MBTI (4D)` , `Social Styles (3D)` , `DISC (3D)` and `DISC x SS (4D)` , the last one crossing DISC and Social Styles on four axes
- `Task Management Data` : `Management Styles` and `Managerial Grid` (the Blake and Mouton grid)
- `Clear Selection` : returns the four menus to `None` in one go. The button only appears if at least one dataset is selected.

Fiction fact: in the Zodiac category there is a dataset called "Malgovert" which corresponds to the psychic classification of the thirteen Zodiac clusters in the Malgovert series by Laurent Oliversen. It is somewhat like a blood-type classification, but with three variables. The points are defined by 3 coordinates representing the sensitivity of the zodiacal profiles to the 3 elysian serums. The spatial position of these clusters is actually used in the fiction.

3.1.3. Custom Points

This section allows you to manually enter up to 10 spatial coordinates (X, Y, Z, W, V5, ..., V10). The resulting point is displayed overlaid on the polytope. You can attach a label to it, duplicate it or delete it.

3.2. Points & Labels

Location: `Workspace` tab, `Dataset` section, `Points` and `Labels` folders.

These settings govern the appearance of every data point projected into the scene, whatever its origin, as well as that of their labels.

3.2.1. Point Families

The application distinguishes six families of points, which are displayed and colored independently:

Family	Origin
Imported	the rows of the imported dataset
Reference	the reference datasets (3.1.2)
Factor Individuals	the individuals projected in the factor plane
Factor Numeric Features	the projected numerical variables
Factor Categorical Features	the projected qualitative categories
Cluster	the centroids produced by clustering

3.2.2. Display

The `Points / Display` folder exposes one checkbox per family: `Show Imported Pts`, `Show Reference Pts`, `Show Fact Indiv Pts`, `Show Fact Num Pts`, `Show Fact Categ Pts`, `Show Cluster Pts`.

The `Labels / Display` folder does the same for the labels: `Show Imported Lbls`, `Show Reference Lbls`, `Show Fact Indiv Lbls`, `Show Fact Num Lbls`, `Show Fact Categ Lbls`, `Show Cluster Lbls`.

Displaying the points without the labels is the usual setting on large volumes; the reverse — labels only — is useful for reading a dense factor plane.

3.2.3. Point Style

- `Point Shape`: shape of the marker — `Dot`, `Square`, `Diamond`, `Cross (+)` or `Cross (x)`.
- `Point Size`: base size of the points.

3.2.4. Encoding by Variable

Three visual channels can be driven by a column of the dataset, rather than being fixed. This is the way to make the scene carry additional information without adding an axis.

- `Variable Point Size`: the column that drives point size. `Variable Max Size` sets the size reached by the maximum value.
- `Variable Point Color`: the column that drives point color.
- `Variable Point Opacity`: the column that drives opacity. Opacity naturally encodes a notion of certainty or inclusion: favor uncertainty metrics (relative standard error, absolute value of the margin, silhouette).
  - `Opacity Invert`: reverses the mapping, so that a low value gives an opaque point — for example so that low uncertainty stays solid.
  - `Min Opacity`: opacity floor applied to the minimum value. It is never zero, so that a point never disappears silently.
- `Hide Var From Space`: removes from the computation of spatial coordinates the variable used for the encoding. Without this option, the variable would be counted twice — once as an axis, once as a visual channel.

3.2.5. Color Gradient

The `Points / Gradient` folder defines the color ramp used by `Variable Point Color`. An `i` monitor at the top of the folder recalls the variable currently being encoded.

- `Diverging Colors`: uses a diverging ramp (negative → middle → positive) anchored on 0, instead of a sequential min → max ramp. This is the right choice as soon as zero means something — a gap, a margin, an uplift.
- `Seq Min` / `Seq Max`: the two colors of the sequential ramp.
- `Div Neg` / `Div Mid` / `Div Pos`: the three colors of the diverging ramp.

3.2.6. Fixed Colors

When no variable drives the color, each family carries its own color.

- `Points / Color` : `Reference Point` , `Data Point` , `Factor Indiv Point` , `Factor Feature Point` .
- `Labels / Color` : `Reference Label` , `Data Label` , `Factor Indiv Label` , `Factor Feature Label` .

3.2.7. Label Style

- `Label Size` : size of the text.
- `Label Angle` : orientation of the label relative to the point.
- `Label Offset` : distance between the label and its point. Increasing this gap is the simplest way to unstack overlapping labels.

3.3. Data Import

Location: `Data` tab.

3.3.1. Import CSV

You can load data in CSV with the `Upload CSV` button. Depending on the subscribed plan, data is limited to a certain number of rows and columns (indicated in the interface). The `Clear Data` button empties the imported dataset.

The expected structures are:

- UTF-8 CSV format
- The first row corresponds to headers
- Separator: comma, semicolon, tab, or pipe "|"
- The following column headers are reserved for specific uses: "label" (point name), "segment" (for inferential statistics) and "target" (for supervised machine learning)

By default, if columns x, y, z, w, v5, v6, v7... are present in the file, they are automatically assigned to the corresponding spatial coordinates.
If no header matches the above convention, then the numerical variables are assigned in order of appearance in the file: the first numerical column is assigned to X, the second to Y, the third to Z, etc.
In case of a mix (example: a, x, y, z, b, c), then the recognized headers (x, y, z) are assigned to the 3 spatial coordinates, and the unknown headers (a, b, c) are assigned to the following spatial coordinates in this order: a -> w, b -> v5, c -> v6.

3.3.2. Role Mapping

This section allows defining the columns that will play a specific role for analytics. Some roles are interchangeable in certain scenarios.
The columns that can play roles are:

- `label`: name that will be displayed above the point on the projected scene
- `color`: point and label color
- `segment`: grouping column used for descriptive or exploratory statistics (profiling)
- `target`: dependent variable column to be explained, used for variable selection (correlation) and for supervised algorithms (LDA, Sup ML)
- `treatment`: binary column for calculating incremental effects and uplifts between two groups (profiling, Hy-Performance); examples: clinical trials, marketing campaigns...
- `flag_test`: binary column (0, 1) to define the rows that will be used to train supervised models and those for the testing phases

Once the roles are assigned, click `Apply Roles`.

3.3.3. Axes Mapping

You can assign numerical variables to the spatial axes of your choice.
Simply enter a Variable / Axis combination and click `Add`.

The `Merge T|C columns on same axes` option is only available for Hy-Performance data. It allows pooling variables suffixed with _T and those with _C. For example, instead of having two axes to display NB_T and NB_C, the option displays a single NB axis that merges the two populations (T and C). To distinguish the two populations, you must either display the labels, or go to `Data Style` and check the `Colorize Treatment` box (`Class Style` section).

3.3.4. Sampling

On a large file, it is often preferable to work on a sample first: the scene stays fluid and the models fit in a few seconds.

- `Size Mode`: how the size is expressed — `Number` (a number of rows) or `Proportion` (a fraction of the file).
- `Size Value`: the desired size, expressed according to the chosen mode.
- `Sample Mode`: sampling method — `Random` (random draw), `Head` (the first rows) or `Tail` (the last ones). `Head` and `Tail` are useful on chronological data, where the order of the file carries meaning.
- `Sample`: applies the sampling.
- `Clear`: restores the complete file.

3.3.5. Test Split

If the `flag_test` column is not defined, it can be created and filled automatically at random by setting a test percentage (`Test Size`) and clicking `Random Split` . The `Counts` monitor displays the resulting split between training and test. `Clear` removes the split.

3.3.6. Preview

A data preview is available in the tab.
In the header are two sections.

Counting

- `Total rows` : number of rows in the imported dataset
- `Total columns` : number of columns in the imported dataset
- `Numeric columns` : number of numerical columns
- `Categorical columns` : number of categorical columns
- `Rows retained` : number of rows retained according to the user's plan (pricing)
- `Spatial dim retained` : number of spatial columns retained (according to the user's plan) and thus visualized in the scene

Reserved columns
This list of tags refers to the reserved roles. It helps identify the columns playing a special role: uid, label, color, target, segment, treatment, flag_test

Column typing
For each column, a tag indicates the type of the column: numerical, categorical or boolean.

3.4. Preprocessing

Location: `Data` tab.

This section allows performing a number of transformations that are sometimes required to run algorithms (factorial projections, clustering, Sup ML).

The `Keep Original Column` checkbox allows keeping the original column alongside the transformed one: the source is kept and the transformation is written into a new `<col>_t` column, which `Undo` removes. Unchecked, the transformation overwrites the column in place. Note: keeping the original is not always desirable, as it introduces information redundancy.

A `Select Column` field allows choosing the column to transform; each entry there is suffixed with its type and its activation state. Each transformation is applied and canceled by means of `Apply` and `Undo` buttons.

Finally, a `Re-Apply in Inference` checkbox allows keeping the settings of these steps to reuse them later on new data; a necessary step when using trained models (Factor, Clustering, Sup ML) on a transformed dataset.

The `Clear Preprocessing` button cancels in one go every transformation applied to the dataset.

Order of the steps
The transformations are applied in a fixed order, which is that of the sections below: typing → deactivation → imputation → winsorization → discretization → category binning → one-hot encoding. The parameters are always estimated on the training rows only (`flag_test` other than 1), then applied to every row.

3.4.1. Column Typing

Three buttons are available to change the format of the columns.

- `Make Numeric` : transforms the column into numerical data; projects spatially in the scene
- `Make String` : transforms the column into qualitative data; does not project
- `Make Bool` : transforms the column into binary data (true/false, yes/no, 0/1); does not project in the scene; corresponds to indicator or "dummy" variables

String and bool data are very similar in their implication within the different features. The only distinction comes from the fact that a bool variable will indeed be used in linear and logistic regression models (Sup ML), whereas string data will be ignored.

Using or ignoring the column:

- `Activate` : allows the variable in the scene and the various modeling modules
- `Deactivate` : excludes the column

3.4.2. Imputation

This section allows filling empty data.

- `Num Imputation` : value used to fill empty numerical data — `-1` , `0` , `mean` (mean of the non-empty data) or `median` (median of the non-empty data)
- `Categ Imputation` : string used to fill empty categorical data

Each of the two has its own `Apply` and `Undo` buttons.

3.4.3. Winsorization

This section allows capping extreme numerical values.

- `Tail Cut %` : percentage capped on high values; a value of 2% means that all values above the 98th percentile will be capped.
- `Two-Sided` : also caps extreme low or negative values

3.4.4. Discretization

This section allows discretizing numerical data into bins. They thus become categorical (qualitative). Discretization is performed into quantiles.

- `Nb Bins` : number of bins (quantiles)

3.4.5. Category Binning

This section allows grouping categories that have low counts and induce sparsity, and therefore instability. It is the equivalent of extreme low values for qualitative data.

- `Rare Category Label` : label of the merged category
- `Use Frequency` : uses count proportions rather than absolute counts
- `Count Threshold` : threshold, in number of occurrences, below which a category is deemed rare. Used when `Use Frequency` is unchecked.
- `Freq Threshold` : the same threshold, expressed as a frequency. Used when `Use Frequency` is checked.

3.4.6. One-Hot Encoding

This section allows transforming a categorical variable into indicator variables. Each category of the variable becomes a binary column (bool, 0/1).

- `Keep all categories` : keeps all categories and transforms them into binaries; if the option is unchecked, the last category is dropped to avoid multicollinearity.

4. Data Exploration

4.1. Stats

Location: `Stats` tab.

4.1.1. Descriptive Statistics

Contains descriptive analyses of the data imported into the application:

- Summary table of dispersion metrics: mean, standard deviation, min and max values, quartiles (Q1, Q2, Q3), coefficient of variation (CV), semi-interquartile range (SIQR = (Q3-Q1)/2), midhinge (MH = (Q1+Q3)/2), trimean (TM = (Q2 + MH)/2), quartile coefficient of dispersion (QCD = SIQR/MH)
- Boxplots: for each numerical variable, with the option to display a distinct scale per variable (`Different Scales`), or a common scale for all variables (`Same Scale`).
- Category Distribution: count of the categories present in each categorical variable, with a list to choose the variable

4.1.2. Inferential Statistics

You can perform quick statistical tests: comparing means or proportions, as well as sample size estimations.

4.1.2.1. Settings

- `Use Case` : mean comparison, proportion comparison, statistical power analysis
- `Value (numeric)` : for mean comparisons; variable acting as the numerical measure
- `Outcome (0/1)` : for proportion comparisons; binary success column (bool, 0/1, yes/no...) that feeds the numerator of the proportion
- `Group (category)` : variable acting as the categorical group
- `Variable A` and `Variable B` : the two variables to compare, when the use case opposes two columns rather than two groups of a single column

Settings specific to power analysis:

- `Design` : data type for the analysis: means or proportions
- `Solve for` : target value of the analysis: sample size or target power
- `Cohen's d` : desired effect size value (for two means)

- `Proportion 1` : value of the first proportion
- `Proportion 2` : value of the second proportion
- `N per Group` : size of each group to determine power
- `Target Power (1−β)` : power threshold deemed sufficient. It materializes the green zone of the gauge, and it is the target used when solving for sample size.

Common settings:

- `Method` : parametric (based on known statistical distributions) or non-parametric (robust to outliers)
- `Alpha` : significance level (type I error) to reject the null hypothesis
- `Alternative` : one-sided (`Greater (A > B)` = right; `Less (A < B)` = left) or two-sided (`Two-Sided`)

The `Run Testing` button launches the test; `Clear` resets the results.

4.1.2.2. Associated Statistics

The results are presented in two areas:

- On the left, the test metrics and some descriptive statistics
- On the right, gauges for power, significance and effect size; as well as a representation of the confidence interval

4.2. Data Exp (Data Exploration)

Location: `Workspace` tab, `Data Exp` panel.

4.2.1. Selection Mode (orbit and selection)

This folder decides what the mouse (or trackpad) does in the scene, and serves as the starting point for building manual groups.

- `Orbit` : default mode. The gesture rotates and moves the scene.
- `Select` : the gesture draws a rectangle that captures the projected points it covers.
- `Tag Selection` : designates the native group whose points you want to capture — the same groups as those of the valence system. Moving this menu selects nothing by itself.
- `Select Tag Points` : captures in one go every point carrying the tag chosen above, replacing the current selection. The interaction mode is not changed: you stay in orbit or in selection. The button can be clicked again as many times as you like, which makes it the way back after refining a selection by hand.

Three read-only monitors report the current state:

- `Current Mode` : `Orbit` or `Select` . The selection count stays visible even after returning to orbit.
- `Selection` : number of points currently selected.
- `Selection as Group` : shows `Active` as soon as the selection has been materialized as a group variable.

Saving a selection as a group is done in the `Manual Groups` block, just below the folder.

4.2.2. Data Scaling

Scaling reads as a pipeline of two successive stages: first the columns, then the rows. Each stage is a drop-down menu, because its options are competing answers to a single question — only one is applied.

Stage 1 — `Column Scaling` (per variable)

- `None` : no transformation.
- `Standard` : centering and reduction. The mean is removed, then division by the standard deviation. This is the classical choice as soon as variables do not share the same unit.
- `Robust` : variant resistant to extreme values. Centering is done on the median and dispersion on the interquartile range, so that a handful of outliers no longer compresses the rest of the cloud.

Stage 2 — `Row Scaling` (per point)

- `None` : no transformation.
- `Unit Norm` : each point is brought back to a unit distance from the origin. Only its *direction* remains; the magnitude is erased.
- `Global Quantile Norm` : same principle, but the reference distance is a quantile of the set of distances rather than the norm of each point. The points then keep their relative gaps, while being brought to a common scale.
    - `Quantile` : the quantile used as reference. Only appears for this mode.

`After Projection`

This setting only appears if a `Row Scaling` is active. It defines when row scaling takes place:

- Box unchecked: in the initial N-dimensional space, before any rotation, shear or projection.
- Box checked: after the projection cascade, on the resulting 3D space.

It is better to scale *before* the projection to keep as much information as possible during the projections. Nevertheless, a posteriori normalization allows obtaining a spread on the unit sphere of all projected points, which greatly simplifies visual exploration — even if a lot of information has been lost in the process.

`Data Zoom`

This slider spaces out or contracts all projected points. It acts as a multiplier of all coordinates of the final 3D space, after scaling. It is therefore not a true normalization, but a visual zoom: the underlying ND coordinates do not change, and it consequently neither brings points into nor takes them out of an orbital (see 4.2.4).

4.2.3. Filtering

The filtering section allows displaying the projected points that meet certain conditions.

- `Filter` : enables filtering. Unchecked, all points remain visible whatever the criteria entered.
- `Inter-Conditions` : nature of the combination between the three groups of criteria below — `All (AND)` (intersection) or `Any (OR)` (union). AND means that every group filled in must be satisfied simultaneously to display a point; OR that a single one is enough.

Each group also has its own internal condition, which combines its criteria with one another: `Label Condition` , `Group Condition` and `Axis Condition` , each settable to `AND` or `OR` .

The criteria are not applied continuously: you must click `Apply` to validate them, and `Clear` to reset everything. The `Status` monitor reports the current filter and the number of points retained.

4.2.3.1. Search Labels

The first group concerns the labels attached to the points:

- `Case Sensitive` : perform searches without considering case. "LABEL" is perceived as "label" or "Label".
- `Whole Word` : perform searches considering the entire label. If checked, the string "Lab" will not match the string "Label".
- `Show` : manual input of a string included in the label to determine the points to display; you can enter several different words separated by a comma
- `Hide` : manual input of a string included in the label to determine the points to hide; you can enter several different words separated by a comma

4.2.3.2. Group Display

This group filters on column roles and computed groups. Each menu lists the available categories; leaving a menu on `None` amounts to not filtering on it.

- `Target` : categories of the target column.
- `Treatment` : categories of the treatment column.
- `Segment` : categories of the segmentation column.
- `Cluster` : the clusters, if a clustering has occurred.

- `Personality` : the families of personality points — micro and macro semiology features, HBDI, DISC profiles and features, Social Styles, Management Styles. See 5.3.

4.2.3.3. Axis Value Filtering

This group concerns the properties of the projected points. It requires selecting one to three axes: `Axis 1`, `Axis 2`, `Axis 3`. If you do not wish to populate them all, you can leave them on `None`.

- `Pipeline Stage` : the stage of the pipeline at which the coordinates are read — original ND coordinates, standardized ND, normalized ND or projected 3D. This is an important point: filtering on the 3D coordinates amounts to filtering on what you see, filtering on the ND coordinates amounts to filtering on the real data.

You can then choose which values to filter: by sign, by absolute value, or by factorial metric.

- `Axis 1 Sign`, `Axis 2 Sign`, `Axis 3 Sign` : filter the points whose coordinates are, as you choose, `Positive` or `Negative` on the corresponding axis. `None` disables the criterion for that axis.
- `Use Coords` : allow filtering on coordinates
- `Coords Ineq` : the direction of the comparison — `≥`, `>`, `≤`, `<` or `=`.
- `Absolute Coords` : linked to the previous checkbox; absolute value that a point's coordinates must satisfy on all previously defined axes (if AND condition) or on at least one of these axes (OR condition)
- `Use Contrib` : allow filtering on factorial contributions
- `Contrib Ineq` : the direction of the comparison for the contribution.
- `Contribution` : same as for the coordinates, but applies to the value of the point's factorial contribution on the axes defined in the previous lists; only applies to "Factor" points, that is, points projected via a PCA or an MCA, which participated in forming the factorial axes.
- `Use Cos2` : allow filtering on factorial squared cosines
- `Equality Tolerance` : tolerance margin applied to the `=` comparison. Without it, a strict equality on floating-point numbers would practically never retain any point.
- `Cos² Ineq` : the direction of the comparison for the squared cosine.
- `Cos²` : same as for the coordinates, but applies to the value of the point's factorial squared cosine on the axes defined in the previous lists; only applies to "Factor" points, that is, points projected via a PCA or an MCA, whether or not they participated in forming the factorial axes.

4.2.3.4. Combined Filtering

Combining the two previous rules
You can define conditions on the strings and on the axes. In this case, the AND/OR condition applies to the whole.
For example, with an AND condition, a point must respect the entered strings and respect the values on all defined axes. With an OR condition, a point must respect the entered strings or respect the values on one of the defined axes.

Influence of automatic projection planes (canonical and factorial)

When filters are active, if you click on an automatic projection plane, namely the canonical planes (e.g., vertex-first, edge-first, etc. Cf. Views) or the factorial planes (`Best Contrib Plan` and `Best Cos² Plan`), then the resulting axes are sent to the filtering axes.

This allows scanning different canonical and factorial spaces while updating the filters based on the new axes displayed on the screen.

4.2.4. Orbital

The orbital is a multidimensional filtering system based on distance. Its name is not an official mathematical term, but a reference to atomistic physics-chemistry, which defines the orbital as a zone of space within which an electron can be observed with a certain probability level. Here, the goal is to highlight the proportion of the population located within the zone delimited by the orbital.

In the application, by default, an orbital will hide the points close to the origin and leave only the peripheral points (the outliers). You can reverse the masking logic (`Outlier Mode`) and display only the central points (inliers).

Important remark:

the orbital acts in the ND space, before the projection and the `Data Scaling`. Thus, data scaling cannot take a point out of or into the orbital. An outlier has no reason to become an inlier because you zoom on the scale of the data.

There is nevertheless a way to fit the size of the solid to the dataset: use `Solid Scale`. This parameter comes into play at the moment the solid is created, and therefore the orbital that derives from it.

4.2.4.1. Orbital Types

The orbital is displayed with `Show Orbital`, and its type is chosen in the `Shape` menu. Three orbitals are proposed:

- `Gaussian Ellipsoid`: a multidimensional ellipsoid based on normal distributions. Two ways to size it: `Confidence (%)`, a Gaussian confidence level based on the Mahalanobis distance, or `k (σ)`, a number of standard deviations. The `Stat Fit` checkbox switches between the two readings.
- `Box`: an n-rectangle whose lengths are defined by percentiles in each dimension; the `Quantile X`, `Quantile Y`, `Quantile Z`, `Quantile W` and `Quantile V` sliders allow adjusting these lengths.
- `Quantile Ellipsoid`: the ellipsoid inscribed within the previous n-rectangle; the `Outer Ellipsoid` option allows using the ellipsoid circumscribed around the n-rectangle.

The `Inside | Outside` monitor permanently displays the number of points located inside and outside the orbital — this is the direct reading of the masked proportion.

In terms of usage, the Gaussian ellipsoid is easy to handle (a single `Confidence (%)` slider), but is only appropriate for point clouds with a Gaussian tendency (normal distribution in each dimension).
The "box" is a robust variant of the previous ellipsoid. But it tends to become very strict (many hidden points) for high quantile values. The quantile ellipsoid is a much softer variant of the box; it only hides points that are distant in several simultaneous directions. But its circumscribed variant (outer orbital) is very restrictive (even more than the box). You have to lower the thresholds to avoid hiding the entire cloud.

From a masking proportion perspective, we have roughly the following values:

- In an n-dimensional space, the 90th quantile n-rectangle in all dimensions encompasses approximately (0.9^n) % of the population (72% in 3D, 65% in 4D, 60% in 5D..., 35% in 10D).
- The ellipsoid inscribed in the previous n-rectangle only covers a rapidly decreasing fraction: in 3D, the ellipsoid orbital occupies 52% of the rectangle, in 4D 30%... and in 10D barely 0.25%!
- The ellipsoid circumscribed around the n-rectangle reverses the trend. In 3D, the orbital occupies almost 2.7 times more space, in 4D 1.2 times more, in 5D 8.3 times more... and it grows faster and faster!

From most tolerant to most strict: inscribed ellipsoid < n-rectangle < circumscribed ellipsoid.

4.2.4.2. Comparison between Orbital and Filtering

What is the difference between the orbital and filtering based on absolute coordinates?
Both objects serve to filter the visualization, but each has its nuances.
Number of filtered dimensions
Filtering based on `Absolute Coords` only applies to (at most) three axes defined in the filtering axes. The orbital, on the other hand, applies to all possible dimensions simultaneously.
Boolean condition
Coordinate filtering can use the AND or OR condition, whereas the orbital only uses an AND principle by construction.
Definition of filtering thresholds
Coordinate filtering requires formally stating the coordinate threshold value that will be checked on the filtering axes. The orbital works instead by a quantile logic (proportion of points having a coordinate less than or equal to the threshold) or probabilistically (Gaussian confidence interval based on the Mahalanobis distance).

4.2.5. Decision Boundary

This option allows displaying the decision boundary of binary logistic regression or the predictive hyperplane of linear regression.

- `Show Boundary` : displays the hyperplane or the boundary.
- `Boundary Alpha` : adjusts the transparency of the hyperplane / decision boundary.
- `Resolution` : adjusts the mesh fineness of the hyperplane / decision boundary.

- `Iso Levels` : in classification, overlays three nested probability shells (p = 0.25 / 0.5 / 0.75) instead of the single boundary at 0.5. This reads the slope of the decision, and not only its location.
- `PD Feature A` / `PD Feature B` : the two variables carried by the axes of the partial dependence surface. All other variables are held at their training median.
- `Residual Sticks` : draws, for each point, a vertical segment between its observed value and the value predicted by the model. The length of the segment is the residual.
- `Residual Alpha` : transparency of these segments.

Note: the decision boundary is only displayed if the target variable is binary; the prediction hyperplane is only displayed if the number of predictors (explanatory variables) is greater than or equal to 2. The degenerate case of the line (simple regression) is ignored.

4.2.6. Hy-Perf Cube

Allows quickly changing the axes displayed on the scene to analyze Hy-Performance datasets.
`Cube Preset` : these presets allow quickly mapping certain spatial axes to the transformed dataset variables (budget, uplift, ROI, etc.).

For further customization, you need to map the axes manually in Dataset / Axes Mapping.

4.2.7. Valence

The Valence logic (connectivity links) is based on a proximity analysis in space: each point is linked to its nearest neighbors, and the color of the line says which neighborhood rank it corresponds to.

- `Show Lines` : displays the valence lines.
- `Selection Only` : draws the lines only from the selected points or vertices, which then act alone as anchors. This is the setting to use as soon as the cloud is dense.

Point classification
The system groups all displayed points into three main families to filter link types:

- Reference: points from presets (Zodiac, Malgovert, DISC, MBTI, etc.).
- Data: points imported via CSV or added manually.
- Factor: points calculated by Factorial Analysis (Individuals or Categories).

Overall logic: Data points are the mobile and analyzed elements, while Reference points, Factor points and the Polytope serve as fixed landmarks (anchors). We thus look at how user data orients itself relative to the application's internal structure.

Valence Info

If `Valence Info` is checked, additional information is added to the side panel when a point is clicked: you can view the various neighbors of a point as well as their respective distances, for the different projection hypotheses (ND, 3D, with or without normalization).
The `Use Inverse Dist` checkbox allows displaying, not the Euclidean distances, but the weightings by the inverse of the distances. Example: with 3 neighbors, the value displayed for a given neighbor is the inverse of the distance divided by the sum (over the 3 neighbors) of the inverse distances.

Neighborhood rank

The `K Neighbors` parameter defines the number of ranks (closest distances) to display per point; it is not the exact number of links displayed (see "Ties" below).

4.2.7.1. Pipeline Stage

The concept of neighborhood depends on the projection stage. As a reminder, the projection includes all these stages:

- ND-4D: ND-standardization, ND-normalization (optional), rotation, shear, projection
- 4D-3D: rotation, shear, projection, 3D-Normalization (optional), scaling (optional)
- 3D-2D/1D: collapse

A neighborhood study can be performed in the initial ND scene before spatial projection. This can include or exclude a normalization step. Similarly, the neighborhood can be determined after projection, in the resulting 3D scene visible on screen. This scene can also include a normalization.

In summary, neighborhoods can be considered:
The `Pipeline Stage` menu designates the stage retained:

- `ND` : valence is calculated on the initial data, ND space without normalization
- `ND Stand` : valence is calculated on centered-reduced ("standardized") ND data, before projection
- `ND Norm` : valence is calculated on normalized ND data, before projection
- `3D` : valence is calculated on projected 3D data, before 3D normalization and before scaling
- `3D Scaled` : valence is calculated on projected data, after any 3D normalizations and scalings

4.2.7.2. Valence Directions

A connection is established between anchor points and neighborhood points. For each anchor point, we look for the neighborhood points. The closest neighbors to the anchor point are called "rank 1". The next neighbors are rank 2, etc.

For each point P, all other points are sorted by Euclidean distance. Points at the same distance (with a 10^-5 tolerance) are grouped into a single rank. With K = 2, the two closest distance ranks are kept - this can generate more than 2 lines if several points share the same distance.

There are two different algorithms depending on the nature of the connection:
Mutual Links (Internal to a group): anchors and neighbors belong to the same point group
Examples: Reference->Reference, Data->Data, Polytope->Polytope

- Calculation: for two points I and J, we calculate J's position in I's

neighbor ranking (rank_{IJ}) and conversely (rank_{JI}).

- Condition: the link is drawn if the average rank (rank_{IJ} + rank_{JI}) / 2 is

below the K Neighbors parameter.

- Logic: allows showing only strong and reciprocal relations within the same population.

Unidirectional Links: anchors and neighbors are different groups
Examples for: Data->Reference, Data->Factor, Zodiac->Polytope.

- Calculation: We take each "Anchor" point (e.g., a Data point) and look for its

nearest neighbors in the "Neighbor" group (e.g., the Polytope vertices).

- Condition: The link is drawn if the neighbor's rank is below K Neighbors.
- Logic: This is an attraction logic. We see which reference points or which facets of the solid the data "tends" towards.

In the valence rules block, located below the folder, you can add all the desired links between the groups acting as anchors and the groups acting as neighborhoods.

Handling Ties
When a rank contains more than one neighbor (a tie), the corresponding lines can be drawn dashed (click `Ties Dash`) instead of solid. This visually signals ambiguous neighborhoods. A line is dashed if either of its ends has a tie at its respective rank level.

- Tolerance: If two points are at an almost identical distance (10^(-5)), they

receive the same rank.

- Visual: If a rank contains several points (tied), the lines are displayed

dashed (if the Dashed Ties option is active in the Style panel).

Color code and distances

- Dimensions: Distances are calculated on the final coordinates (x, y, z)

after all transformations (ND-4D-3D). This means that zoom and 4D
rotations directly impact the valence links.

- Colors: each proximity level (neighbor rank) can be identified by a customizable color

4.2.8. Valence and Orbital Style

Valence style (`Valence Metrics / Style` folder)

- `Line Width` : thickness of the valence lines.
- `Ties Dash` : draws dashed the lines corresponding to a rank containing several tied neighbors (see "Handling Ties" above).
- `Rank 1` , `Rank 2` , `Rank 3` , `Rank 4+` : the four colors associated with the neighborhood ranks.

Orbital style (`Orbital / Style` folder)

- `Orbital Width` : thickness of the orbital's outline.
- `Center Origin` : centers the orbital on the origin of the frame rather than on the barycenter of the cloud.
- `Size Constraints` : enables the three bounds below. They are purely aesthetic and have no mathematical effect: they only prevent an orbital from becoming illegible because it is too small or too large on screen.
    - `Absolute Size Min` : absolute minimum size.
    - `Size Ratio Min` and `Size Ratio Max` : bounds of the size relative to that of the solid.
- `3D Color` , `4D Color` , `ND Color` : colors of the orbital according to the space in which it is computed.
- `Alpha` : transparency. To hide an orbital without disabling it, simply set its alpha to zero.

4.2.9. Trajectory

Links the points of a same group chronologically, to read a movement rather than a frozen cloud.

- `Show Trajectories` : draws one chronological polyline per group, through the projected points.
- `Time Column` : the column that gives the chronological order.
- `Group Column` : the column that defines the groups; each category gives a trajectory.
- `Trajectory Opacity` : transparency of the polylines.
- `Time Gradient` : colors each path from oldest to most recent (dark → light), so that the direction of travel is legible.

4.2.10. Dimension Fading

On a cloud projected from a high-dimensional space, many points end up superimposed on screen while being far apart in the dimensions that are not represented. Fading makes this hidden mass visible by playing on opacity.

- `Enable Dim Fading`: enables fading.
- `Opaque threshold`: threshold below which a point stays fully opaque.
- `Min alpha`: opacity floor of the points that are furthest away in the hidden dimensions.
- `Show hidden-mass`: displays the hidden-mass indicator, that is, the share of the cloud that the current projection does not render.

4.2.11. Scene Markers

The `Scene Markers` folder (reference axes and planes) is described in 2.2.4.4, along with the other scenery elements of the scene.

4.2.12. Analysis Module Overlays

Three folders only appear once the corresponding module has been run. They project its result into the 3D scene.

`Association Rule Edge` — see 5.2.4

- `Show Rule Edges`: draws an edge between the items linked by an association rule.
- `Edge Min Lift`: hides the edges whose lift is below this threshold. This filter is purely visual and remains independent from the `Min Lift` used when extracting the rules.

`MMM Surface & Map` — see 5.10

- `Fit Landscape`: displays the hyperparameter search grid (θ × ec) of a channel, colored by fit score, in place of the time series. Requires having run a local MMM beforehand.
- `Landscape Channel`: the channel whose fit landscape is displayed.
- `Collinearity Map`: displays the channels as points — proximity reflects the co-evolution of the transformed spends, color reflects the VIF. A high VIF signals collinearity, and therefore unreliable attribution.

`Optimization 3D View` — see 5.7

- `3D View`: what the scene represents — `None`, `Selection Plane` or `Feasibility Polytope` (the polytope of admissible solutions).

4.3. Data Style

Location: `Workspace` tab, `Exp` section, `Class Style` and `Layouts` folders.

The style of the points and labels themselves (size, shape, families displayed, encoding by variable) is described in 3.2. This section deals with what comes on top: coloring by class, shape by class, and the way axes and numbers are named and displayed.

4.3.1. Class Style (coloring by class)

Five roles can be used to color the points. They are not exclusive: they are stacked in order of priority, and the first one that applies to a point wins.

Priority	Setting	Source of the color
1	`Colorize User Groups`	the groups drawn by hand with the lasso
2	`Colorize Cluster`	the cluster categories
3	`Colorize Target`	the categories of the target column
4	`Colorize Treatment`	the categories of the treatment column
5	`Colorize Segment`	the categories of the segmentation column

A point not covered by one level falls back to the next: a point with no manual group takes the color of its cluster, and so on.

Two settings reuse palettes already defined elsewhere, rather than creating a sixth one:

- `Use Valence Colors` : assigns to color slots 1 to 4 the valence palette (ranks 1 to 4+).
- `Use Orbital Colors` : assigns to slots 5 to 7 the orbital palette (3D / 4D / ND) — or to slots 1 to 3 if the valence colors are disabled.

Shape by class

- `Shape By` : encodes a class role by the *shape* of the point, which is orthogonal to color — two qualitative variables can therefore be read at once. Your choice of: `None` , `Manual Group` , `Cluster` , `Target` , `Treatment` or `Segment` . The first category keeps the shape defined by `Point Shape` ; the following ones receive distinct glyphs.
- `Category Shapes` : monitor recalling the category → shape mapping actually used.

4.3.2. Valence Colors

For valence, each line connects two points, A and B, and its color translates the neighborhood rank. Point A might rank B as its closest neighbor (Rank 1), while B might rank A differently (e.g., Rank 2). To reflect this bidirectional relationship, the displayed color is determined by the integer part of the average of both ranks: Displayed Rank = $\lfloor (\text{rankA}\rightarrow\text{B} + \text{rankB}\rightarrow\text{A}) / 2 \rfloor$
This guarantees that the color reflects mutual agreement: two points considering each other close get a strong color (Rank 1), while asymmetrical relationships are naturally degraded.

4.3.3. Layouts

A *layout* renames the spatial axes and adapts their colors, so that the scene speaks the vocabulary of the framework under study rather than that of mathematics. The computation is in no way modified: only the dressing changes.

- `Layout` : the active layout.
  - `Default` : standard display, axes X, Y, Z, W, V5...
  - `Imported Data` : the axes take the names of the imported columns assigned to them.
  - `Malgovert` : alternative display used by the author in the context of the novel Malgovert. The X, Y and Z coordinates are renamed K, L and S. They stand for the Kappa (Cocytus), Lambda (Lethe) and Styx (Sigma) serums in the fictional universe of Malgovert. The colors of the corresponding axes are modified to reflect the serum colors in the novel: blue for Cocytus (K), silver for Lethe (L) and green for Styx (S).
  - `MBTI 4D` , `DISC 3D` , `SS 3D` , `DISC-SS 4D` , `Task Management` : alternative displays that take up the labels of the behavioral frameworks used in business.
- `Label Chars` : number of characters kept for the axis names of a layout. For example, in the `SS 3D` layout, the X and Z axes become "Assertiveness" and "Responsiveness"; setting the slider to 2, the labels become "As" and "Re".
- `Number Decimals` : number of decimals displayed for numbers, in every table and every panel of the application.
- `Local Settings Reset` : resets the display settings kept locally by the browser, and restores the default values.

5. Analysis

5.1. Running a Module

No analysis module recomputes on its own: the settings are laid down first, then a button launches the computation. Every module follows the same convention.

- `Run <Module>` : runs the computation **in the browser**. This is the default route: the result is immediate and the 3D scene updates right after.
- `Run Large <Module>` : delegates the same computation to the **remote Python engine**. This variant only appears for the modules concerned and on the plans that grant access to it. It is useful as soon as the data volume makes the local computation too slow, or the requested method exceeds what the browser can hold in memory.
- `Clear` : erases the module's results and removes its points from the scene.

The buttons carry the name of the module: `Run Correlation` , `Run Profiling` , `Run Association` and `Run Large Association` , `Run Factor` , `Run Clustering` , `Run Estimator` and `Run Large Estimator` , `Run Forecast` and `Run Large Forecast` , `Run Performance` , `Run MMM` and `Run Large MMM` , `Map Personality` , `Run Testing` , `Run` for optimization.

A module run on data that has changed since stays displayed as it is: it is up to you to run it again. The reminder is especially worth it after a preprocessing step (part 3.4) or a resampling.

5.2. Advanced Stats

This section complements the Stats tab. Here, the analyses cover several pairs of columns in parallel. The tab is subdivided into three parts: correlations (bivariate), profiling (over- or under-representation of certain attributes within clusters) and association rules.

5.2.1. Thresholds for Statistical Tests

You can define the desired statistical tests as well as the threshold values used to colorize the metrics. These thresholds apply to the correlation and profiling charts.

- `Num Testing` : statistical test desired for numerical correlations (Pearson, Spearman, Kendall)
- `Cat Testing` : statistical test(s) desired for categorical dependencies (chi2 only, or a mix of chi2 and Fisher's exact test for 2x2 contingency tables)

- `Mixed Testing` : statistical test used to analyze the gaps between central values of two groups (ANOVA F, which generalizes Student's T test, and Kruskal-Wallis, which generalizes the Mann-Whitney U test)
- `Alpha` : threshold value below which p-values turn green in the tables.
- `Num Effect Threshold` : threshold value above which numerical effect sizes (in absolute value) turn green; applies to R2, Spearman's rho squared, and Kendall's tau
- `Cat Effect Threshold` : threshold value above which categorical effect sizes (in absolute value) turn green; applies to Cramer's V, Cohen's h and Wilcoxon's r
- `OR Effect Threshold` : threshold value above which ratio and log-ratio metrics (in absolute value) turn green; applies to (log-)odds ratios, (log-)lifts and Weights of Evidence
- `Mix Effect Threshold` : threshold value above which categorical effect sizes (in absolute value) turn green; applies to Cohen's d, Hedges' g, Glass's Delta, Cliff's delta, Eta squared (H test) and Omega squared (ANOVA)

5.2.2. Correlations

5.2.2.1. Settings

Data to analyze (optional and cumulative):

- `Target` : target variable used to determine the most discriminating variables
- `Imported Data` : data imported in CSV
- `Factor Data` : data from a factorial projection
- `Cluster Data` : cluster labels from a clustering
- `Score Data` : scores computed by a Sup ML estimator

5.2.2.2. Associated Statistics

Table of correlation metrics with the target
Bivariate dependence statistical tests (Pearson and equivalents, chi-square, ANOVA, etc.) with the p-value and the associated effect sizes (R2, omega squared, Cramer's V, etc.)

Table of correlation metrics between explanatory variables
Same as above, but between the explanatory variables

Effect size matrices (intra-variable correlation)
Correlation matrices (effect size) between variables.

Effect size bars vs Target
Intensity of the correlation between the explanatory variables and the target variable. Reflects the discriminating power.

Correlation matrices of the scene

Correlation matrices between spatial data or data projected by a factorial method.

For these correlation matrices, all metrics are offered:

- `Dot Product` (normalized dot product): sum of the component-wise products, divided by the product of the norms; dot(u, v) = (u·v) / (||u|| × ||v||) (cosine similarity); range: [-1, 1]. 1 = aligned, 0 = orthogonal, -1 = opposite
- `Pearson` : Pearson's linear correlation coefficient, from -1 to 1; the coefficient is close to zero when there is no correlation; $r(u, v) = \Sigma(u_i - \bar{u})(v_i - \bar{v}) / (\sigma_u \times \sigma_v)$ where $\bar{u}$, $\bar{v}$ are the means and σ are the standard deviations of the coordinate components.
- `Similarity` (similarity index): ratio of the distance of a pair to the maximum distance of the dataset; sim(u, v) = 1 - d(u, v) / maxDist where d is the Euclidean distance and maxDist is the largest distance among the active points; equals 0 when two points are superimposed, equals 1 when the two points are the most distant of all pairs

Note: the diagonal self-comparisons of the matrix are left empty.

5.2.3. Profiling

This section is an in-depth description of the characteristic values found in groups of interest (segment, target, cluster, score quantile), with or without treatment.

5.2.3.1. Settings

Data to analyze (optional and cumulative):

- `Imported Data` : data imported in CSV
- `Factor Data` : data from a factorial projection
- `Cluster Data` : cluster labels from a clustering
- `Score Data` : scores computed by a Sup ML estimator

The settings specific to profiling are:

- `Group Variable` : variable used to define the groups to profile; corresponds to one of the variables of interest (segment, target, cluster, score quantile)
- `Use Complement` : to compute the denominators of the ratios, either the total population is used, or the population complementary to the group under study (= the "rest" of the population = the other groups). If this option is checked, the complement is used; this highlights the metrics of small groups.
- `Moment Metric` : variable brought forward for the display of numerical profiling metrics
- `Lift Metric` : variable brought forward for the display of categorical profiling metrics

5.2.3.2. Associated Statistics

Profiling table
Five tabs centralize all the metrics:

- `Profile` : general summary; combines part of the lift and moment views
- `Moment` : focus on the numerical descriptive variables
- `Lift` : focus on the categorical descriptive variables
- `Lift Net` : focus on the categorical descriptive variables (1 row per category) in the presence of a treatment
- `Info Value` : focus on the categorical descriptive variables (1 row per variable) with or without a treatment; it complements the Lift / Lift Net view with aggregated indicators

Profiling bars
A graphical version of the previous results.
The charts are grouped by profiling metric. One row per variable, with the option to sort by group.

- Positive bars (one color per group) reflect categories or high values that are characteristic (typical, over-represented) of the groups concerned.
- Negative bars (one color per group) reflect categories or low values that are characteristic (under-represented) of the groups concerned.

5.2.4. Association

Association rules determine sets of categories (qualitative variables) called "antecedents" that appear at the same time as a category of interest called the "consequent". The rules only work between categorical variables.

5.2.4.1. Settings

The settings specific to association rules are:

- `Group Variable` : optional variable used to partition the rules: either the rules are global, or they are subdivided according to every category of the group variable
- `Group as Consequent` : rather than partitioning the rules, the group variable is this time used to directly define the Consequent values.
- `Min Support` : minimum support to retain a rule; this is the proportion of rows that contain the whole set of antecedent categories; recommended 1-5%
- `Min Confidence` : minimum confidence to retain a rule; this is the proportion of rows (among the rows having all the antecedents) for which the consequent appears; it filters out unreliable rules, even though it is not sufficient (the lift below is needed to discard rules that are reliable but of no interest); recommended 60-80%

- `Min Lift` : minimum over-representation to retain a rule; a rule with a lift of 1 is a random selection; a rule with a high lift indicates that the consequent is indeed more likely to occur when the antecedents appear (a useful rule); a negative lift is precisely how you find the antecedents that push the consequent away (the opposite goal); recommended 1.2
- `Max Antecedent Size` : number of antecedents used to define a rule; the larger this number, the more the combinatorics explode
- `Max Consequent Size` : number of consequents used to define a rule; the larger this number, the more the combinatorics explode
- `Top N` : visual filter to keep only the first N rows (but the rows can be ordered or filtered independently of this value)

The `Run Association` button launches the extraction in the browser, `Run Large Association` delegates it to the Python engine (see 5.1), and `Clear` resets the results.

5.2.4.2. Associated Statistics

The results table lists the rules as well as the associated metrics (support, confidence, lift, etc.).
The ∧ symbol means intersection (logical AND).

5.3. Hy-Personality

Here is the "rigorous" method to determine which Zodiac signs resemble you the most!

A personality questionnaire built into the application can be filled in online. It combines themes present in several frameworks used in business (Social Style, DISC, MBTI) as well as the lexical descriptions of the astrological zodiac.

Beware, this is not a psychological/psychometric analysis protocol. It is an illustrative example showing the value of factorial projections applied to questionnaires of all kinds. The goal here is to show visually the lexical resemblances between different famous frameworks.

Filling in the questionnaire and projecting the results thus allows you to visualize the concepts or attributes shared between your answers and the profiles present in the different frameworks. This approach relies on a PCA or an MCA of the answers of the different profiles to the 18 questions of the form. It is a dataset provided by the application. It is free, but cannot be modified in this interface. The answers you add, on the other hand, are customizable.

Procedure: answer the 18 questions by moving the slider left or right (5-degree scale) according to the answer that suits you best. If you hesitate, you can leave the slider in the middle (balanced or neutral answer).
Once finished, enter a label and click `Submit` to project the results into the factorial hyperspace!

5.3.1. Data Sources

The questionnaire can come from different sources:

- `Reference Points`: data already built into the application (behavioral frameworks and zodiac)
- `User Form`: online questionnaire displayed on screen
- `CSV`: questionnaire imported in CSV format (via the `Form Import` section of the current tab, or the import section of the `Data` tab)

The `Form Import` section offers `Upload CSV` to load a file of answers, and `Clear All Respondents` to erase all imported respondents at once.

5.3.2. Projection Techniques

The answers to the questionnaire (from all sources) are projected using a PCA or an MCA already trained with the application's reference data.

- `Method`: `PCA`, `MCA` or `MFA` (multiple factor analysis, when the questions are grouped into thematic blocks)
- `Keep All Factors`: keeps all factors (this also includes the statistical noise in the explanation of the variances)
- `Nb Factors`: if the previous option is unchecked, you can define here the number of factors to keep; recommended between 3 and 8

The `Map Personality` button launches the projection; `Clear` resets the results.

5.3.3. Reference Points Filtering

The `Reference Filtering` section allows choosing which reference points are displayed. Each category is subdivided between *Features*, which correspond to the description criteria, and *Profiles*, which are their outcome — a combination of several criteria.

The available checkboxes are:

- `Semio Micro Features` and `Semio Macro Features`: the semiology circuits, at the fine then the aggregated level
- `HBDI Features`: Herrmann's brain quadrants
- `MBTI Profiles` and `MBTI Features`
- `DISC Profiles` and `DISC Features`
- `SS Profiles` and `SS Features`: Social Styles
- `MS Profiles` and `MS Features`: Management Styles
- `MG Profiles`: Managerial Grid (the Blake and Mouton grid)
- `Zodiac Profiles` and `Zodiac Features`

Example: the *Extraversion* and *Introversion* points, which define the orientation of energy, are *MBTI Features*, whereas *INTP* is an *MBTI Profile* defined by the presence of the "I" category (for Introvert). Likewise, *Fire* is a *Zodiac Feature*, whereas *Aries* is a *Zodiac Profile*.

5.3.4. Associated Statistics

Radar of answers by profile
Each vertex of the radar represents a question (integer values from -2 to 2). Two profiles are superimposed in this radar to visualize the overall resemblance of their answers to the questionnaire.

Answers of all respondents by question
The bar chart gives the answer of all respondents to the selected question.

Correlation matrices of the scene
Correlation matrices between raw answers to the personality questionnaire (integer values from -2 to 2), and also between projected answers (factorial coordinates).

5.3.5. Example of use: inverted zodiac

Go to *Modeling / Personality*
In the *Personality Map* panel, leave the default settings:

- *Method*: PCA
- *Reference Points*: checked
- *User Form*: checked
- *CSV*: unchecked

Fill in the questionnaire, give a name and click *Submit*. You are then switched to Workspace.

Go to the *Dataset* section of the Workspace panel, *Points / Display* folder, and check only:

- *Show Reference Pts*: these are the points already in the application
- *Show Fact Indiv Pts*: these are the answers to the online form

Go to the *Exp* section of the Workspace panel.

- In *Data Scaling*, set *Row Scaling* to *Unit Norm* and *Data Zoom* to around 2 or 3 depending on how it feels.
- In *Valence*, check *Show Lines*, then *Valence Info* and *Use Inverse Dist*, and set *Pipeline Stage* to *ND*.
- Still in Valence, below the K Neighbors slider, check only *Factor-Reference*

Just below, in the *Filtering* section, check Filter then, in the *Personality* list, choose *Zodiac Profiles*.

From now on, clicking the point corresponding to your answer opens a side panel with statistics. You can consult the *ND Valence* and *ND-Norm Valence* charts, which state your "proportion" of each Zodiac sign (seen as your nearest neighbors in terms of answers).

To get a better sense of the distances, at the very bottom of the panel, after the valences, click the *Best Cos² Plan* button. This displays the space in which your answer is spatially best represented. The nearest neighbors then tend to move closer visually, which is more consistent.

5.4. Factor

This tab allows applying dimensionality reduction methods to the points imported in CSV, or to the datasets already built in (synthetic data, answers to the personality questionnaire).
Simply choose the number of factorial axes retained and the technique used.

5.4.1. Factorial Axes

- `Keep All Factors` : keeps all possible factorial axes; depending on the method, this corresponds either to the number of numerical variables (PCA), to the number of categories of the categorical variables (MCA), to the sum of the two previous ones (FAMD), or to the number of classes - 1 (LDA).
- `Nb Factors` : allows choosing the number of axes to retain
- `Inference` : uses, on new data, the projection matrices already calibrated by a previous factorial analysis

5.4.2. Factorial Techniques

- `PCA` : principal component analysis; suited to continuous numerical data
- `MCA` : multiple correspondence analysis; suited to categorical data
- `FAMD` : factor analysis of mixed data; suited to data mixing numerical and categorical variables
- `MFA` : multiple factor analysis; suited to data organized in thematic blocks, each block being weighted so that none dominates the projection merely because of its number of variables
- `LDA` : Fisher's linear discriminant analysis; suited to numerical data including a target variable to separate groups
- `MCA+LDA` : discriminant analysis on qualitative data (Saporta's method); allows performing an LDA on categorical data
- `FAMD+LDA` : discriminant analysis on mixed data (a variant of Saporta's method); allows performing an LDA on mixed data

Scope of the computation and overlays

- `Process Individuals` : projects the individuals (the rows) into the factorial space.

- `Process Features` : projects the variables (the columns). The two checkboxes are independent: on a large dataset, computing only one of the two lightens the scene appreciably.
- `Correlation Sphere` : displays a unit spherical mesh in the space of the variables. A numerical variable close to the surface ($\Sigma\cos^2 \approx 1$) is well rendered by the three displayed axes; a variable close to the center is not. Requires the numerical variable points to be visible (PCA, FAMD).
- `Partial Stars` : links a category to its partial components, in multiple factor analysis — `selected` (for the selected point only), `all` (for all) or `off` .
- `Target` : the target variable, required by the discriminant methods (LDA and its variants).
- `Inference` : applies an already trained factorial model to new rows, instead of refitting it.

The `Run Factor` button launches the projection.

Once the data is projected, it can be visualized in the Workspace tab, and is displayed on top of the polytope that serves as a visual anchor. It is recommended to use a simple cube (or even a tesseract for 4D analysis) or a sphere to make exploration easier.

5.4.3. Associated Statistics

In the Factor tab, you will find a number of statistics relating to the factorial method:

- Contribution of the individuals (the rows) to the axes: a point far along an axis will strongly "contribute" to that axis
- Contribution of the features (the columns) to the axes: same, but from the point of view of the variables
- Squared cosine of the individuals: a point with a high squared cosine is well represented on that axis; its projected coordinate remains faithful
- Squared cosine of the features: same, but from the point of view of the variables

Note that the factorial axes have a correspondence with the spatial axes:

- 1st factorial axis (PCA_1, MCA_1) corresponds to the X axis
- 2nd factorial axis (PCA_2, MCA_2) corresponds to the Y axis

And so on with the Z, W, V5, V6 axes, etc.
Thus, when analyzing contributions or squared cosines, the value of the drop-down list corresponds to the factorial axis. This is why they are displayed in the form "X (PCA_1), Y (PCA_2)" or "X (MCA_1), Y (MCA_2)".
But if the dataset contains spatial coordinates (also named X, Y, Z), then you end up in situations where the X, Y, Z axes (of the dataset before projection) contribute to the X axis (which is in fact the first factorial axis, after projection). You must therefore keep in mind what belongs to the input dataset and what belongs to the result of the factorial projection in the target space (and which may share the same letters).

5.5. Clustering

Clustering allows grouping points by spatial proximity.

5.5.1. Data Sources

The data source to cluster carries nuances that enable or disable certain options.
Numerical data:

- `Imported Data` : raw imported data (CSV); only the numerical variables of the CSV are retained
- `Spatial Data` : data visually projected in the scene: this includes the previous numerical data as well as the reference points and the manually added points (Workspace / Dataset tab); it is therefore an enriched version of the previous one, but it does not support the *Inference* mode (see below)

Factorial data:

- `Factor Individuals` : projected data of the individuals (rows)
- `Factor Features` : projected data of the features (columns)

Personality data (Hy-Personality):

- `Personality Form` : projected data of the respondents to the personality questionnaire

5.5.2. Methods

Clustering methods

- `K-Means` : the simple method to go fast
- `Hybrid` : the complete method that combines K-Means and HCA (hierarchical cluster analysis) and potentially a second K-Means for "peripheral cleaning"
- `Partition` : a particular case that consists in assigning, to each spatial point, the nearest reference point; this therefore requires reference points to be displayed (either the points defined in `Dataset` , or the projected points of the `Hy-Personality` questionnaire)

Options specific to the Spatial Data and Personality Form sources

- `Add Reference Points` : if the source is *Spatial Data*, then this option allows adding the reference points added manually (Workspace / Dataset tab); if the source is *Personality Form*, then this option allows adding the projected points of type MBTI, DISC, SS, Zodiac.
- `Reference as Center` : uses the reference points as seeds (initial points) for the first K-Means; it necessarily disables the choice of the number of *Centers*

Options specific to the Imported Data and Factor Data (Individuals or Features) sources

- `Inference` : application of an already trained clustering model to new data

Common options

- `Standardize` : unit scaling of the numerical data

5.5.3. Pre-KMeans

The first stage consists in applying a K-Means method to the initial dataset.

- `N Init` : number of different attempts to determine the best partitioning
- `Use Sqrt(N) Centers` : uses the square root of the number of points as the number of centroids to perform the partitioning
- `N Centers` : if the square-root rule is unchecked, then this slider sets the desired number of centroids. In general, a number between the cube root and the square root of the initial size (number of rows) is a good starting point
- `Filter Outliers` : allows assigning isolated points to a separate class; based on the Mahalanobis distance
- `Outlier Confidence %` : confidence level used to define the outliers

5.5.4. HCA (Hierarchical Cluster Analysis)

The second step is a hierarchical cluster analysis (Ward's method) applied to the centroids of the Pre-KMeans.

- `HCA Auto-Cutoff` : automatically determines the number of final clusters
- `Cutoff Metric` : metric used to automatically determine the number of clusters; can be based on the silhouette coefficient or on the acceleration curve of the dendrogram (the elbow on the between-class inertia curve)
- `Min Nb Clusters` : minimum number of clusters desired (allows searching for other optima in the previous method)
- `Max Nb Clusters` : maximum number of clusters desired (allows restricting the search for optima in the previous method)
- `Nb Clusters` : if the Auto-Cutoff mode is not checked, then this slider determines the number of final clusters

5.5.5. Final Cleaning

Depending on the spatial distribution of the data, HCA methods can suffer from questionable allocations for the points located at the periphery of the clusters.

- `Post-KMeans` : Voronoi reassignment after the HCA. A K-Means performed on the barycenters of the HCA clusters reassigns the peripheral points to the geometrically most intuitive cluster, which corrects the boundary cells.

- `Merging Threshold %` : at the very end, this percentage allows merging the clusters that do not have enough individuals (proportion set by this slider)

The `Run Clustering` button launches the algorithm; the `Clear` button resets the results.

5.5.6. Cluster Ellipsoids

A covariance envelope can be drawn around each cluster, in the 3D scene. This is the most direct visual reading of the quality of a segmentation: two overlapping envelopes signal ambiguous clusters.

- `Show Ellipsoids` : displays one translucent covariance shell per cluster.
- `k·σ` : half-extent of the shell, expressed in standard deviations.
- `Opacity` : transparency of the shells.

5.5.7. Associated Statistics

Quality metrics
The quality metrics displayed are: silhouette coefficient, Davies-Bouldin Index (DBI), Calinski-Harabasz Index, Dunn Index

Charts
The following section displays the dendrogram of the HCA, and the acceleration curves of the between-class inertia. Then comes the distribution of the population by cluster, the boxplots by spatial variable, and the contingency matrices with the variables of interest (Segment, Target, Treatment).

5.6. Sup ML (Supervised Machine Learning)

Sup ML allows training regression (linear) and classification models.

5.6.1. Settings

- `Source` : selection of the data to model: either the raw imported data, or the data projected by a factorial method (on the individuals only)
- `Target` : target variable to explain
- `Estimator` : choice of model: linear regression or logistic regression
- `Score Bins` : number of quantiles used to discretize the predicted score; useful to run profiling analyses on the scores (Stats tab, Profiling section)
- `Inference` : application of an already trained model to new data
- `Cluster Predictor` : uses the clustering classes (if they exist) as additional explanatory variables

The `Run Estimator` button launches the algorithm; the `Clear` button resets the results.

5.6.2. Associated Statistics

The results are available in two tabs, one per sample (Train, Test).

Linear regression
The quality metrics displayed are: R2, MAE, RMSE.
A panel displays the coefficients of the regression.

Logistic regression
The quality metrics displayed are: Balanced Accuracy, Macro Precision, Macro F1, ROC AUC, Gini.
A panel displays the confusion matrix, then the coefficients of the regression (and the associated odds ratios).
Finally, two curves are offered: ROC and lift (cumulative gains).

Scores and score quantiles
The model's scores are discretized into score quantiles. These quantiles can be used as a group variable in profiling.

5.7. Optimization

This module looks, among the individuals of the dataset, for those that maximize or minimize a target variable, under the constraints defined by the active filters. The result can be projected into the scene (see 4.2.12).

5.7.1. Settings

- `Target` : the variable to optimize.
- `Direction` : the direction of the optimization — `maximize` or `minimize` .
- `Individual ID` : the column identifying each individual in the results.
- `Error Margin` : tolerance margin allowed around the optimum. Widening this margin brings back a set of near-optimal solutions rather than a single point, which is often more usable.
- `Seed` : seed of the random generator. With a fixed seed, two runs give the same result — indispensable for comparing two configurations.

The `Run` button launches the solving; `Clear` resets the results.

5.7.2. Associated Statistics

The results list the individuals retained, the value reached by the target, and the gap to the theoretical optimum. The associated 3D view (`Optimization 3D View` , see 4.2.12) allows displaying either the selection plane or the polytope of admissible solutions.

5.8. Forecast

The time series forecasting model relies on a Holt-Winters exponential smoothing.

5.8.1. Settings

- `Time Column` : time variable (can be a simple index)
- `Target` : target variable to predict over time
- `Seasonal Period` : number of periods that define a season
- `Seasonality` : type of seasonality: additive or multiplicative
- `Auto-fit α/β/γ` : automatically determines the exponential smoothing coefficients, by a grid search minimizing the in-sample error. Unchecked, the three coefficients are set by hand:
  - `Alpha (level)` : smoothing of the level (the height of the curve)
  - `Beta (trend)` : smoothing of the trend (the slope)
  - `Gamma (season)` : smoothing of the seasonal component (the periodicity)
- `Back-Test Holdout` : number of points removed from the training set to test the model
- `Inference Horizon` : number of additional points to predict (data after the imported periods)

The `Run Forecast` button launches the computation in the browser. `Run Large Forecast` delegates the computation to the remote Python engine: this is the route to take on large volumes, where the local computation becomes too slow.

5.8.2. Associated Statistics

Metrics
The results consist of visualization curves as well as quality metrics (in the training and back-test samples): MAE (mean absolute error), RMSE (root mean squared error), MAPE (mean absolute percentage error).

Charts
The charts display the actual, fitted and predicted curves.
A decomposition of the 3 pillars of the built function is also available: the level (the height), the trend (the slope), the season (the periodicity).

5.9. Hy-Performance

Feature coming soon.
Goal: study the effectiveness of a treatment (clinical trial or marketing action) on a population, by means of an A/B test integrating statistical evaluations.

5.9.1. Data Sources

Granular data or aggregated data

- `Keep All Factors` : keeps al
- `Nb Factors` : allows
- `Inference` : uses, on

5.9.2. Settings

Type of analysis: absolute or incremental

- `Control Group` : checked, each action is evaluated by a target-versus-control test. Unchecked, the estimation is done single-arm (`SOLO_` prefix), by confidence interval — this is the mode to use when the design did not provide for a control group.
- `Alpha` : significance level (type I error) to reject the null hypothesis
- `Alternative` : one-sided (`Greater` = right; `Less` = left) or two-sided (`Two-sided`)
- `Multi-Test Correction` : correction for multiple tests — `None (per action)` tests each action in isolation; `BH-FDR` re-arbitrates the whole set of significance indicators on the Benjamini-Hochberg threshold, computed on the entire tested family. This second option controls the false discovery rate at level alpha, and its decision propagates to the incremental amounts.
- `Expected Ratio (T/C)` : safeguard against sample imbalance. Here you declare the target/control ratio planned by the design (1 = 50/50, 9 = 90/10). A chi-square test per action warns when the split actually observed departs from it. See 5.9.4 for the details. The value 0 disables the control.

Analyzability thresholds (reliability of the measures and volume-related instability)

- `Analyz Min Conv` : minimum threshold to consider an action analyzable
- `Max Relative Prec (%)` : maximum tolerated value of the coefficient of variation, to consider an action analyzable
- `Vol Increase Method` : method used to define the stability of the statistical tests when volume increases: you can choose to increase the size of the controls (Controls Only) or of both the controls and the treated (All Groups)
- `Vol Increase Factor` : enlargement factor of the groups to test stability with respect to volume
- `Amount Formula Switch` : number of converted (in each group) that selects the type of formulas used for the uplift computations; when a group has fewer converted than the figure stated here, then the volume-based formulas are used, otherwise the amount-based formulas are used

Incremental amount computations

- `Inlier Salvage` : method for recovering/preserving low-volume actions whose mean precision is sufficient
- `Inlier Min Conv` : minimum number of converted for an action to be "salvageable"

- `Inlier Max CV` : maximum tolerated value of the coefficient of variation, to consider a low-volume action "salvageable"

ROI handling

- `Breakeven KO → 0` : actions that do not reach break-even (the amounts of the treated converted do not cover the budget) are replaced by zero (rather than by negative values)
- `Isolate Non-Zero IA` : if the option is checked, then significant actions are isolated and are not grouped over time; otherwise, only ROI+ actions (positive ROI) are isolated, but the other actions are indeed grouped over time.

5.9.3. Associated Statistics

PERIOD
Period of aggregation for the groups of actions and the financial KPIs: Q1, Q2, Q3, Q4, H1, H2, FY

AGG_TYPE
Is the aggregate evaluated on actions (before grouping, coded "BG") or on groups of actions (after grouping, coded "AG")

YEAR
Year of the action

HALF_YEAR
Semester of the action

QUARTER
Quarter of the action

MONTH
Month of the action

WEEK
Week of the action

DAY
Day of the action

STRATEGY
Strategy followed by the action

FLAG_OUT
Remove from analysis

LEVEL1
Typology of the action (1st level category)

LEVEL2
Typology of the action (2nd level category)

LEVEL3
Typology of the action (3rd level category)

LEVEL4
Typology of the action (4th level category)

SEGMENT
LEVEL1 - LEVEL2 - LEVEL3 - LEVEL4

ACTION_GROUP
(Month) - Strategy - Segment - ROI Flag - (ROI Trend)

Month and Trend are only IApearing for certain ROI flags and to_group flag

MARGIN
Margin Profit (in % of outstanding)

AMOUNT_FACTOR
Factor used to assess the profit from margin and amount: PROFIT = AMOUNT *AMOUNT_FACTOR*
MARGIN
Useful when margins depend on two multiplied variables (eg. monthly subscription price and number of months subscribed)

UNIT_COST
Unit communication cost

BUDGET
Budget of the action (unit cost * nb targets)

ANALYZABLE_BUDGET
Part of the budget (in €) that is analyzable (ROI_FLAG is not "NA")

ANALYZABLE_BUDGET_RATIO
Part of the budget (in %) that is analyzable (ROI_FLAG is not "NA")

POSITIVE_ROI_BUDGET
Part of the budget (in €) that is profitable (ROI_FLAG is "ROI+")

POSITIVE_ROI_BUDGET_RATIO
Part of the budget (in %) that is profitable (ROI_FLAG is "ROI+")

POS_ROI_ANA_BUDGET_RATIO
Part of the analyzable budget (in %) that is profitable (ROI_FLAG is "ROI+")

NB_T
Nb of Targets

NB_C
Nb of Controls

CG_SIZE
Control Group Size (%)

NB_CONV_T
Nb of Converted Targets

NB_CONV_C
Nb of Converted Controls

CONV_RATE_T
Conversion Rate Target = Conv Rate Target

CONV_RATE_C
Conversion Rate Control = Conv Rate Control

UPLIFT_NB
Uplift in volume (delta rates in percentage points)

UPLIFT_NB_RATIO
Uplift in Volume Ratio (%)

AMOUNT_T
Amount (Total Converted Amount) Target

AMOUNT_C
Amount (Total Converted Amount) Control

SQUARED_AMOUNT_T
Squared Amount (Total Squared Converted Amount) Target

SQUARED_AMOUNT_C
Squared Amount (Total Squared Converted Amount) Control

WEIGHTED_SQUARED_AMOUNT_C
Scaled Squared Amount Control

PROFIT_A
Profit All Customers (Targets + Controls)

PROFIT_T
Profit Target

PROFIT_C
Profit Control

AVG_AMOUNT_A
Average Amount per Customer

AVG_AMOUNT_T
Average Amount per Target

AVG_AMOUNT_C
Average Amount per Control

AVG_AMOUNT_CONV_A
Average Amount per Converted Customer

AVG_AMOUNT_CONV_T
Average Amount per Converted Target

AVG_AMOUNT_CONV_C
Average Amount per Converted Control

VAR_AMOUNT_T
Variance of the Amount Target

VAR_AMOUNT_C
Variance of the Amount Control

VAR_AMOUNT_CONV_T
Variance of the Converted Amount Target

VAR_AMOUNT_CONV_C
Variance of the Converted Amount Control

UPLIFT_AMOUNT_RATIO
Uplift in Amount Ratio (%)

UPLIFT_TOTAL_RATIO
Total Uplift Ratio (%)

CI_VALID_T
Is the Confidence Interval of RR valid?

CI_VALID_C

CI_SE_NB_T
Standard Error of the Resp Rate Confidence Interval

CI_SE_NB_C

CI_LOWER_NB_T
Lower bound of the Resp Rate Confidence Interval

CI_LOWER_NB_C

CI_UPPER_NB_T
Upper bound of the Resp Rate Confidence Interval

CI_UPPER_NB_C

CI_SE_AMOUNT_(CONV)_T
Standard Error of the Average Amount Confidence Interval *[1]

CI_SE_AMOUNT_(CONV)_C

CI_LOWER_AMOUNT_(CONV)_T
Lower bound of the Average Amount Confidence Interval *[1]

CI_LOWER_AMOUNT_(CONV)_C

CI_UPPER_AMOUNT_(CONV)_T
Upper bound of the Average Amount Confidence Interval *[1]

CI_UPPER_AMOUNT_(CONV)_C

VOLUME_INCREASE_FACTOR
("VOL_INC", see *[2])
Strategy to increase volumes: it impacts NB, NB_CONV, CONV_RATE.

Note: this metric is not visible in the output. It's only an intermediate within the code.

Z_STAT
Z Statistic

Z_NORM_PROBA
Cumulative probability of the Z-Stat regarding the Normal distrib

Z_NORM_P_VALUE
P-Value of the Z-Stat regarding the Normal distrib

ODDS_RATIO
Odds Ratio of the CONVancing contingency table

Z_HYPERGEOM_PROBA
Cumulative probability of the Z-Stat regarding the Hypergeometric distrib

Z_HYPERGEOM_P_VALUE
P-Value of the Z-Stat regarding the Hypergeometric distrib

P_VALUE_NB
Retained P-value for volume significance

SS_NB
Is the uplift in volume significant?

T_STAT_(CONV)
T Statistic *[1]

T_NORM_PROBA_(CONV)
Cumulative probability of the T-Stat regarding the Normal distrib *[1]

T_NORM_P_VALUE_(CONV)
P-Value of the T-Stat regarding the Normal distrib *[1]

T_STUD_PROBA_(CONV)
Cumulative probability of the T-Stat regarding the Student distrib *[1]

T_STUD_P_VALUE_(CONV)
P-Value of the T-Stat regarding the Student distrib *[1]

P_VALUE_AMOUNT_(CONV)
Retained P-value for amount significance *[1]

SS_AMOUNT_(CONV)
Is the uplift in amount significant? *[1]

SS_DOUBLE_(CONV)
Means "Significant in both amount and nb" *[1]

SS_ZERO_(CONV)
Means "Not Significant in neither amount nor nb" *[1]

SS_DIV_(CONV)
Means "Significant in amount (or nb) but not in nb (or amount)" *[1]

RAW_ANALYZABILITY
Is the action / action-group roughly analyzable regarding the CG existence and the Nb of Conversions?

CI_LOWER_NB_C_WITH_CONV_RATE_T
Theoretical value of the Control's lower Conf Interval bound if we assume that CONV_RATE_C = CONV_RATE_T

TRUE_ZERO_CONTROL
Is the zero number of Converted Control a reality or an issue due to low volume?

NET_ANALYZABILITY
Is the action / action-group deCONVitely analyzable regarding the Conversions and the statistical significance stability?

NOTE FOR MULTI-EFFECT

- If one effect is analyzable then the global effect is analyzable

IA_PC_AMOUNT
Credit: additional Amount per customer based on amounts
Insurance: additional probability per customer based on amounts

IA_PC_NB
Credit: additional Amount per customer based on volumes
Insurance: additional probability per customer based on volumes

IA_COHERENCE_FLAG
Credit: retained formula for additional Amount
Insurance: retained formula for additional probability

"2" means amount-based formula
"1" means volume-based formula
"0" means unsignificant (IA = 0)

The "New" method is also named "coherence rule": a action is coherent if, when increasing UPLIFT_NB, we do not decrease AVG_AMOUNT_CONV_T.

The "Old" method is also named "retake rule": if we are not significant in amount, then we might "retake" the IA from the significance in nb.

IA_PC
Credit: retained additional amount per customer
Insurance: retained additional probability per customer

IA
Credit: retained additional Amount
Insurance: retained additional probability

ROI_PC
ROI per customer

Note for multi-effect
> The global-effect-ROI-PC is calculated by computing the underlying effect-IA-PC (multiplied by their respective margin/lifetime) and by substracting the unit cost once! → it's not the simple sum of the underlying effect-ROI-PC!

- If the global effect is NOT analyzable, then the ROI is cut to zero

ADD_PROFIT_PC
Additional Profit per customer

ROI
Total ROI

ADD_PROFIT
Total additional Profit

ROI_PU
ROI per unit invested

ROI_PU_ANA
ROI per unit invested and analyzable

ROI_MAX_PC
Theoretical maximum ROI per customer (if control never get Converted)

ROI_MAX
Theoretical maximum ROI (if control never get Converted)

ROI_FLAG
Typology of ROI/IA (situations of profitability)

TO_GROUP
Is the action groupable?
We want to group the actions that are not analyzable or not significant

ROI_AMOUNT_WITHOUT_SIGNIF
Theoretical ROI (based on amounts) without significance

ROI_NB_WITHOUT_SIGNIF
Theoretical ROI (based on volumes) without significance

ROI_TREND
Has the action (groupable) a positive or negative trend of ROI?

5.9.4. Methodology Guide

Hy-Perf is designed to *refuse to conclude* when the data does not allow a conclusion. This section explains the statistical safeguards, so as to read the results correctly - and their silences.

5.9.4.1. Analyzability: when the tool refuses to conclude

Each action must pass decoupled checks before an uplift or a ROI is retained:

- **Estimability** (`Analyz Min Conv`, default 10): below ~10 conversions, rates and means are simply not estimable - the action is marked NA.

- **Precision** (`Max Relative Prec (%)` , default 20%): the relative standard error of the measured means must stay below the threshold; a "significant" but very imprecise effect is not actionable.
- **Formula routing** (`Amount Formula Switch` , default 50): below 50 conversions per group, the amount formulas are too unstable - the engine switches to the volume formulas.

An NA is not a bug: it is the tool telling you that the sample cannot answer the question yet. NA actions are re-tested under a simulated volume increase (`Vol Increase Method` / `Vol Increase Factor`): some are "salvageable" - to be run again later with more volume - rather than dead ends. The budgets of non-analyzable actions are excluded from the analyzable KPIs instead of being counted silently.

5.9.4.2. Multiplicity and the alpha = 0.1 default

Each action x effect x segment is tested at `Alpha` = 0.1 by default, with no correction between tests. This default is deliberate: the product philosophy is one decision per action, not mass discovery. But it is permissive for budget decisions: test 100 actions with no real effect and about 10 will still come out "significant" by chance. The dashboard displays this expected number of false positives next to the aggregates.
When you screen a large catalogue of actions and the aggregate picture prevails, switch `Multiplicity` to BH-FDR (Statistical Inference panel): each significance flag is re-gated on the Benjamini-Hochberg cutoff, controlling the expected share of false discoveries at alpha, and propagates through to the IA and ROI.

5.9.4.3. Winner's curse on aggregates

Aggregated KPIs (analyzable budgets, total incremental amounts, bucket ROI) are built from the actions that passed the significance filter. This selection is optimistic by construction: among noisy measures, those that pass are more often the ones that got lucky upwards. The dashboard therefore brackets the aggregate: total ROI **with** the significance filter vs **without** (ROI_AMOUNT_WITHOUT_SIGNIF / ROI_NB_WITHOUT_SIGNIF columns) - the truth generally lies between the two.

5.9.4.4. SRM — declaring the expected target/control ratio

If your design assigns target and control according to a known ratio (e.g. 90/10), declare it through the `Expected Ratio (T/C)` control (`srm_expected_ratio` ; 0 = disabled). The engine then runs a sample-ratio-mismatch chi-square test per action (SRM_CHI2 / SRM_P_VALUE / SRM_WARNING columns): a real split that deviates significantly from the declared ratio most often signals a contaminated control group, the most frequent silent killer of A/B analyses in CRM. Without a declared ratio, the control stays disabled and never raises a false alarm.

5.10. MMM

The MMM (Marketing Mix Modeling) is a powerful tool that allows evaluating the contribution of investment levers to a time-based target variable.

5.10.1. Settings

- `Time Column` : time variable (can be a simple index)
- `Target` : target variable to predict over time

The investment channels

This section allows telling the model which columns represent the marketing levers. This assignment is indispensable for the model to work correctly.

Baseline

The baseline is the "natural" contribution to the target variable when all other investments stop.

- `Trend` : adds a slope to the baseline
- `Seasonal Period` : to specify the seasonality

Adstock

- `Carryover (max lag)` : geometric adstock window — number of past periods over which a spend keeps acting; carryover effect (geometric law) of an investment over several periods after it took place
- `Max Delay` : maximum waiting time / delay (in periods) between the moment of the investment and the appearance of its effect on the target

Ridge

- `L2 Penalty` : strength of the ridge regularization to handle multicollinearity; it pulls the contribution of the channels towards zero to remove useless noise
- `Non-Negative media` : constraint forbidding media from having a negative contribution

ROI Prior

- `Prior Weight` : global strength for the ROI priors; this value is applied by default on each channel (when the value of the channel weight is < 0)
- `ROI - <channel>` : value of the theoretical ROI (contribution) for each channel (optional)
- `W - <channel>` : strength of the channel's ROI prior; this value replaces the Prior Weight value when it is >= 0; when the value is < 0, then Prior Weight serves as the replacement value

Back-Test

- `Holdout` : number of points removed from the training set to test the model

- `Reliability Bootstrap` : number of samples used to establish the 90% confidence interval of the contributions

5.10.2. Associated Statistics

Metrics
The results consist of visualization curves as well as quality metrics (in the training and back-test samples): R2, MAE (mean absolute error), RMSE (root mean squared error), MAPE (mean absolute percentage error).

Charts
The first chart represents the slices of contributions.
The second table exposes the contributions of each channel, as well as a few associated metrics (adstock, delay, total investment, variance inflation factor, etc.).
The third chart displays the response curves (saturation curves) that represent the evolution of the contribution as a function of the budget. Note the presence of a particular point on each curve: the elbow, that is, the moment when the curve starts to plateau rapidly.

5.10.3. Methodology Guide

Hypersolid's MMM runs *entirely in your browser*: no cloud, no data transfer, no MMM software to install - a complete Marketing-Mix Model on local data. This autonomy comes with a duty of honesty: this section states what the model needs, what its numbers mean, and when NOT to believe them.

5.10.3.1. Data prerequisites

- **At least 52 regular periods** - weekly is ideal, 104 weeks is comfortable. The engine warns below 52 points and refuses duplicated timestamps.
- **Regular cadence**: the rows are treated as equidistant; gaps or mixed frequencies distort the carryover - a pre-flight warning is raised when the cadence looks irregular.
- **Variable spend per channel**: a channel that spends the same amount every period is unidentifiable (blocking error). Flighting - alternating spend on and off - is what makes attribution possible.
- **A clean target**: a consistent KPI (sales, conversions, revenue) with no unexplained regime change; add control columns for known shocks (price, distribution, promotions).
- **Seasonality covered at least twice**: a seasonality of period 52 requires 2+ years of data, otherwise the seasonal dummies eat the media signal (a warning is raised, along with the one on the parameters/observations ratio).

5.10.3.2. Reading the results

For business profiles:

- **ROI** = modeled contribution / spend over the analysis window. This is a model-based attribution, not an experiment: the best coherent explanation of your history, not a measured causal fact.

- **robust ✓** means that the channel's contribution stayed above zero across the bootstrap resamplings. It is a stability indicator, NOT a statistical significance test.
- **The run notes matter as much as the numbers**: they signal non-convergence, unidentified saturation (dashed curves), collinearity and data quality problems. A run covered in notes is a run not to be believed.
- For data profiles:
- Lead with the **holdout MAPE**: the backtest is chronological and leak-free (transforms, standardization and priors are recalibrated on the training head only). The in-sample R2 is the most flattering metric - a mere sanity check.
- **The confidence intervals** are block bootstrap intervals conditional on the calibrated transforms: calibration uncertainty is not included, so the real uncertainty is wider than the displayed interval.
- **VIF > 10** on a channel means that its individual share cannot be reliably separated from the correlated channels - the combined contribution may be right, the split between them is not.

5.10.3.3. Three structural limits

- **Attribution without experimentation**: a regression MMM reads correlation patterns over time. When you have experimental measurements (geo-tests, lift studies), inject them as ROI priors (`Prior Weight`, per-channel `ROI` / `W` controls) - that is exactly their role, and it is the frequentist equivalent of Meridian's Bayesian priors.
- **Edge bias**: the adstock ramps up at the start of the series, and the carryover generated after the last period is never credited - the ROI is slightly underestimated for channels with a strong carryover.
- **Shape of the saturation**: the Hill slope is fixed at 1, so the response curves are concave and saturating, never S-shaped. When the calibration pushes a channel's half-saturation to the top of its search grid, the curve is drawn **dashed** with a hollow elbow: **the saturation is not identified for that channel - do not read a budget allocation elbow on an indicative curve**.

5.10.3.4. When NOT to believe the model

- Fewer than ~52 points, irregular cadence, or a run covered in warnings.
- The channel you care about has a dashed response curve (unidentified saturation) or a VIF > 10.
- The holdout MAPE is far above the train MAPE (overfitting), or the solver reports a non-convergence.
- Spend concentrated on a single burst, or two channels always active together - no model can separate them.
- The target went through a structural shock (pandemic, pricing overhaul) with no control column to cover it.

5.10.3.5. Validation and reproducibility

The engine is benchmarked against Meridian's public simulated dataset: the channel rankings are compatible in the sense of the confidence intervals, and the contribution levels fall within the Bayesian band as soon as weak ROI priors are used (the gap without priors is documented, with its causes). A seeded sensitivity analysis verifies that the ROI ranking is stable under resampled noise. Every result is stamped with an `engineVersion` and a `configHash`: same data, same config, same version ⇒ same output, verifiable.

5.11. Hyperion

6. Import / Export

Location: `Export` tab. The analytical extractions are also accessible from each analysis tab.

6.1. Solid

- `View (PNG)`: exports the current visualization image, in PNG format. The capture reproduces exactly the displayed scene, background included — if `Transparent` is checked in the background settings (see 2.2.4.1), the PNG is produced on a transparent background.
- `Copy Rotation`: copies to the clipboard the current rotation state of the solid (all planes, ND included), in a form reusable as a custom view.

6.2. Config

You can export and later re-import the workspace configurations, that is, the whole set of settings of its panels.

Export

- `Export All Config`: exports all settings in JSON format.

You can also export only a sub-part:

- `Export Solid Config`: generation and style settings applied to the solid.
- `Export Projection Config`: rotation, shear and projection settings.
- `Export Data Config`: style settings applied to the projected data.

Import

- `Import Config`: re-imports a previously exported configuration file.

6.3. Scene

- `Projected Data (CSV)`: exports the projected spatial data in CSV format.

6.4. Analytics

Location: `Export` tab, or at the top right of each analytical tab (Stats, Factor, Clustering...).

You will find all possible extractions, organized by module:

- Stats: descriptive statistics, correlation, profiling, association
- Hy-Personality: metrics related to the projected responses of the personality form
- Factor: contributions, squared cosines and variances related to the factorial projection
- Cluster: clustering metrics, curves and contingency tables
- Sup ML: metrics, coefficients and curves of the regression/classification models
- Forecast / Time Series: metrics of the exponential smoothing (Holt-Winters)
- Hy-Performance: the tables of actions, strategies and duels
- MMM: contributions, slice curves and saturation curves of the MMM

6.5. PDF Recap

- `Charts Recap (PDF)`: gathers into a single PDF the charts produced by the analysis modules during the session. Serves as a report to share, without having to export each chart separately.

6.6. Info Points

When you click on a point in the scene, a number of data items are displayed in the right side panel. At the very bottom of this panel there is a button to download this data.