Topologically Equivariant Artist Model

Hannah Aizenman
Mikael Vejdemo-Johansson
Thomas Caswell
Michael Grossberg
Department of Computer Science, The Graduate Center

September 5, 2025

About

Me

- PhD candidate in Computer Science
- Matplotlib Community Manager & Core Developer
- socials/github: story645

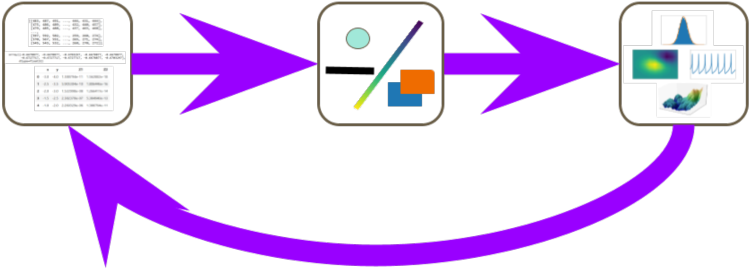
Project

- Funded by Chan Zuckerberg Initiative EOSS 1 & 3

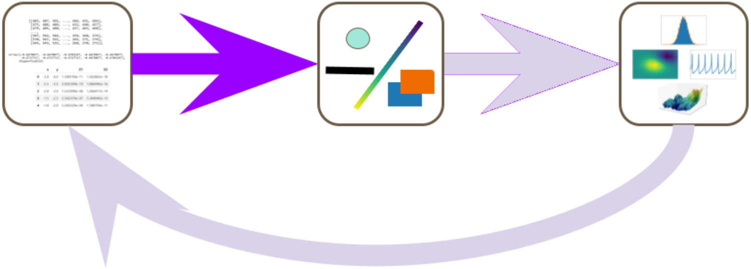
Goals

- uniform API for arbitrary datasets in arbitrary data containers
- generalizable methodology for expressing structure preservation
- framework that translates into coding specifications

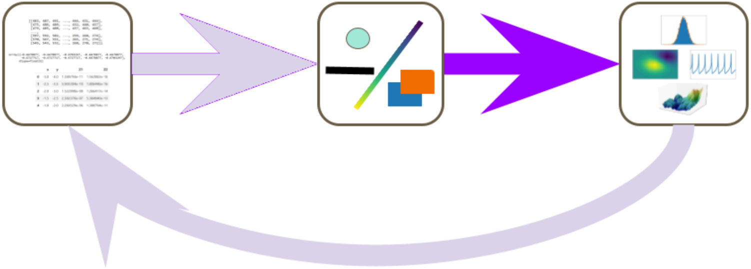
What do visualization libraries do?



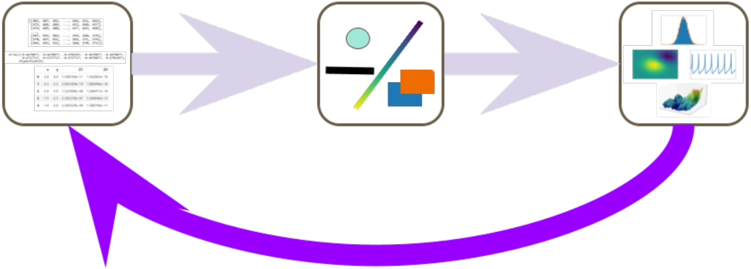
What do visualization libraries do?



What do visualization libraries do?



What do visualization libraries do?



Matplotlib

How do we express structure?

field is a set of values of the same type, e.g. one column of a table or the pixels of an image

topology is the connectivity and relative positioning of elements in a dataset [1].

Structure Preservation: Equivalent scales and transforms

Definition

[2] An **action** of $G = (G, \circ, e)$ on X is a function $act : G \times X \to X$. An action has the properties of identity $act(e, x) = x$ for all $x \in X$ and associativity $act(g, act(f, x)) = act(f \circ g, x)$ for $f, g \in G$.

Fields: Equivariance

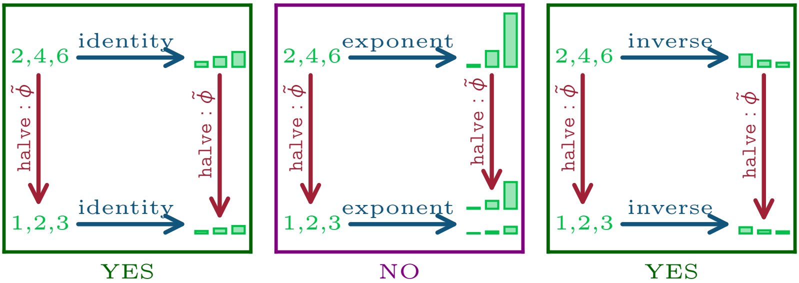
Given a group G that acts on both the input X and the output Y of a function $f : X \to Y$

Definition

A function f is **equivariant** when $f(act(g, x)) = act(g, f(x))$ for all g in G and for all x in X [3]

Fields: Equivariance

Is $\mathit{data} \mapsto \square$ equivariant w.r.t. scaling?



Fields: Homomorphism

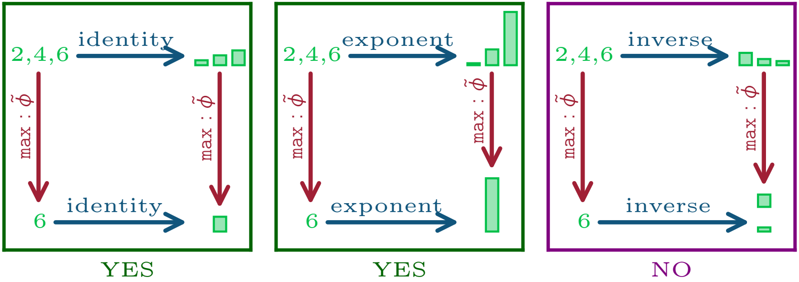
Given the function $f : X \to Y$, with operators $(X, \circ)$ and $(Y, *)$

Definition

A function f is **homomorphic** when $f(x_1 \circ x_2) = f(x_1) * f(x_2)$ and preserves identities $f(I_x) = I_y$ all $x, y \in X$ [2]

Fields: Homomorphism

Is $\mathit{data} \mapsto \square$ homomorphic w.r.t. partial order?

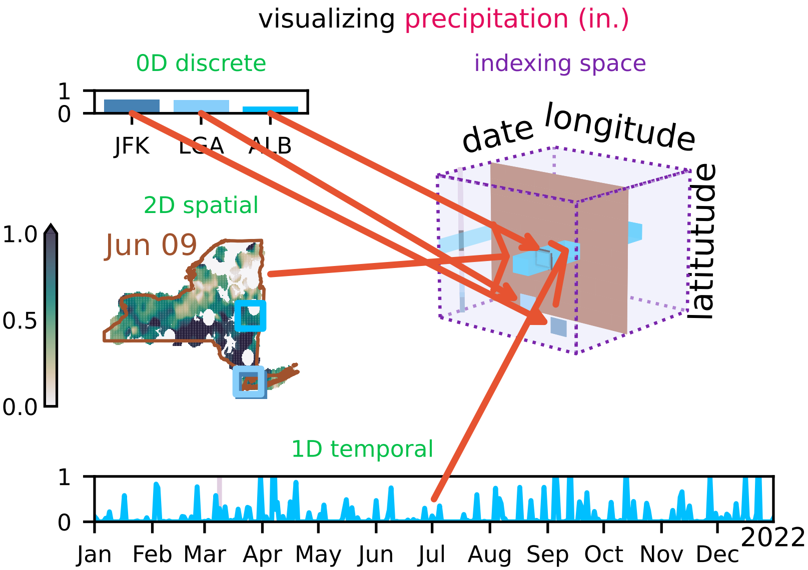


Topology: Homeomorphism

Definition

A function f is a *homeomorphism* if it is bijective, continuous, and has a continuous inverse function f^{-1}.

Homeomorphic topology



Structure Preservation

- Bertin: equivalent data and visual field properties [4]
- Mackinlay: homomorphic fields and equivalent topology [5]
- Wilkinson: homomorphic scales and equivalent topology [1]
- Kindleman & Schieidegger: equivariant scales, invariant data representation, equivalent topology [6]

contribution

explicit homeomorphic topology, equivariant actions on fields and topology

Domain Specific Library: library assumes structure [19]

DATE	LATITUDE	LONGITUDE	PRCP (in.)	NAME
2021-01-01	43.1167	-77.6767	0.2205	ROCHESTER GTR INTL AP
2021-01-01	41.5	-73.9	0.0000	STONYKILL NEW YORK
2021-01-01	42.7431	-73.8092	0.2598	ALBANY AP
2021-01-01	43.8	-73.7	0.0000	SCHROON LAKE NEW YORK
2021-01-01	43.0078	-73.6511	0.0000	SARA NEW YORK
2021-01-01	40.7794	-73.8803	0.6614	NEW YORK LAGUARDIA AP
2021-01-01	40.6386	-73.7622	0.6299	NEW YORK JFK INTL AP
2021-01-01	43.1111	-76.1039	0.4094	SYRACUSE HANCOCK INTL AP
2021-01-01	40.7939	-73.1017	0.5984	ISLIP LI MACARTHUR AP
2021-01-01	43.35	-73.6167	0.1181	GLENS FALLS AP

ggplot[7]
Vega[10]
Altair[13]
Tableau [16]
[17, 18]

Domain Specific Library: library assumes structure [19]

DATE	LATITUDE	LONGITUDE	PRCP (in.)	NAME
2021-01-01	43.1167	-77.6767	0.2205	ROCHESTER GTR INTL AP
2021-01-01	41.5	-73.9	0.0000	STONYKILL NEW YORK
2021-01-01	42.7431	-73.8092	0.2598	ALBANY AP
2021-01-01	43.8	-73.7	0.0000	SCHROON LAKE NEW YORK
2021-01-01	43.0078	-73.6511	0.0000	SARA NEW YORK
2021-01-01	40.7794	-73.8803	0.6614	NEW YORK LAGUARDIA AP
2021-01-01	40.6386	-73.7622	0.6299	NEW YORK JFK INTL AP
2021-01-01	43.1111	-76.1039	0.4094	SYRACUSE HANCOCK INTL AP
2021-01-01	40.7939	-73.1017	0.5984	ISLIP LI MACARTHUR AP
2021-01-01	43.35	-73.6167	0.1181	GLENS FALLS AP

ggplot[7]
Vega[10]
Altair[13]
Tableau [16]
[17, 18]

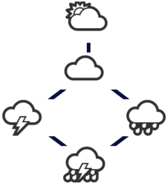
ImageJ[8]
ImagePlot[11]
Napari[14]

Domain Specific Library: library assumes structure [19]

DATE	LATITUDE	LONGITUDE	PRCP (in.)	NAME
2021-01-01	43.1167	-77.6767	0.2205	ROCHESTER GTR INTL AP
2021-01-01	41.5	-73.9	0.0000	STONYKILL NEW YORK
2021-01-01	42.7431	-73.8092	0.2598	ALBANY AP
2021-01-01	43.8	-73.7	0.0000	SCHROON LAKE NEW YORK
2021-01-01	43.0078	-73.6511	0.0000	SARA NEW YORK
2021-01-01	40.7794	-73.8803	0.6614	NEW YORK LAGUARDIA AP
2021-01-01	40.6386	-73.7622	0.6299	NEW YORK JFK INTL AP
2021-01-01	43.1111	-76.1039	0.4094	SYRACUSE HANCOCK INTL AP
2021-01-01	40.7939	-73.1017	0.5984	ISLIP LI MACARTHUR AP
2021-01-01	43.35	-73.6167	0.1181	GLENS FALLS AP

ggplot[7]
Vega[10]
Altair[13]
Tableau [16]
[17, 18]

ImageJ[8]
ImagePlot[11]
Napari[14]



Gephi[9]
Graphviz[12]
Networkx[15]

Building Block Library[29]: visual algorithms assume structure [30]

1. Matplotlib[20] $\rightarrow$ Seaborn[21], xarray [22]
2. D3 [23]
3. VTK [24, 25],MayaVi[26]$\rightarrow$ Titan[27], ParaView[28]

Design Composable Structure Preserving API

Fiber Bundles: Butler: "unified, dimension-independent framework" that expresses data as the mapping between continuity and fields [31, 32]

Simplicial Databases: Spivak: Add rich typing for fields to Bulter [33]

Category Theory Language: express constraints in specifications [34]

Sheaves: Ghrist "algebraic data structure" for representing data over topological spaces [35]

Expressive Types

$$\texttt{dataset} : \texttt{topology} \rightarrow \texttt{fields}$$

Fiber Bundle

Definition

A **fiber bundle** (E, K, π, F) is a structure with topological spaces E, F, K and bundle projection map $\pi : E \to K$ [36].

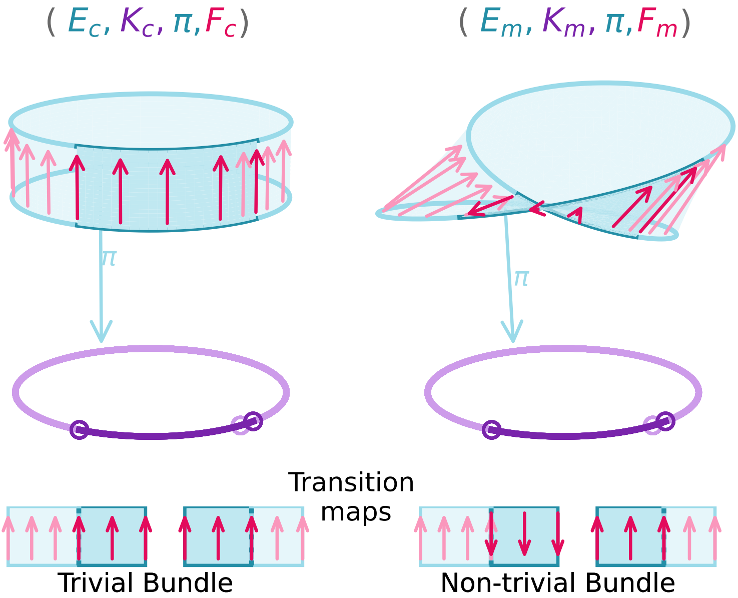
$$F \hookrightarrow E \xrightarrow{\pi} K \tag{1}$$

A continuous surjective map $\boldsymbol{\pi}$ is a **bundle projection** map when

1. all fibers in the bundle are isomorphic. Since all fibers are isomorphic $F \cong F_k$ for all points $k \in K$, there is a uniquely determined fiber space F given by the preimage of the projection π at any point k in the base space K: $F = \pi^{-1}(k)$.
2. each point k in the base space K has an open neighborhood U_k such that the total space E over the neighborhood is locally trivial.

Local triviality means $E|_U = U \times F$.

Fiber Bundle



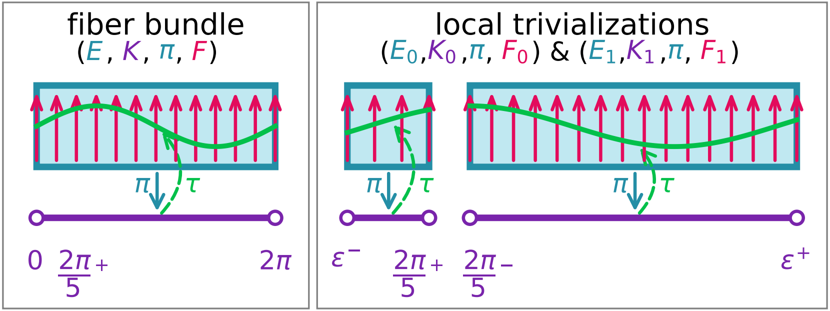
Fiber Bundle: section

Definition

A **section** $\tau : K \to E$ over a fiber bundle is a smooth right inverse of $\pi(\tau(k)) = k$ for all $k \in K$

$$\begin{tikzcd} F \arrow[r, hook] & E \arrow[d, two heads, "\pi"'] \\ & K \arrow[u, bend right, dashed, "\tau"'] \end{tikzcd} \tag{2}$$

Fiber Bundle: section



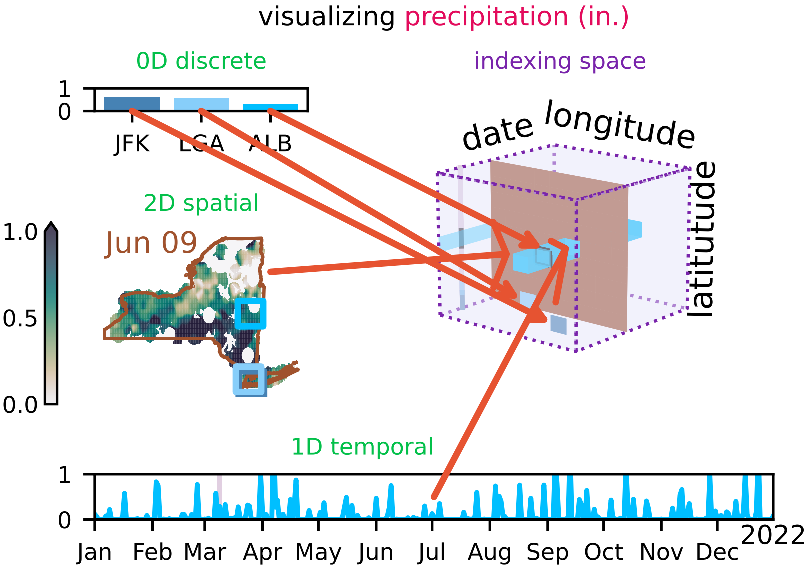
Base space: topology

The base space of a fiber bundle is a quotient topology[37]. Given a set X and a function $\mathcal{N} : X \to 2^{2^X}$ that assigns to any $x \in X$ a non-empty collection of subsets $\mathcal{N}(x)$, where each element of $\mathcal{N}(x)$ is a *neighborhood of* x, then X with $\mathcal{N}$ is a topological space and $\mathcal{N}$ is a neighborhood *topology* if for each x in X: [38]

Definition

1. if N is a neighborhood $N \in \mathcal{N}(x)$ of x then $x \in N$
2. every superset of a neighborhood of x is a neighborhood of x; therefore a union of a neighborhood and adjacent points in X is also a neighborhood of x
3. the intersection of any two neighborhoods of x is a neighborhood of x
4. any neighborhood N of x contains a neighborhood $M \subset N$ of x such that N is a neighborhood of each of the points in M

Why neighborhood topology?



Definition

An **category** $\mathcal{C}$ consists of the following *data*:

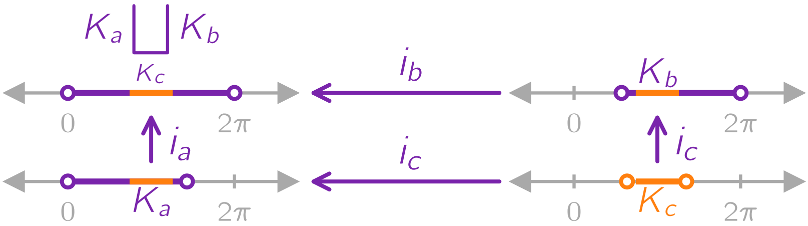
1. a collection of *objects* $X \in \mathbf{ob}(\mathcal{C})$
2. for every pair of objects $X, Y \in \mathbf{ob}(\mathcal{C})$, a set of *morphisms* $X \xrightarrow{f} Y \in Hom_{\mathcal{C}}(X, Y)$
3. for every object X, a distinct *identity morphism* $X \xrightarrow{id_x} X$ in $Hom_{\mathcal{C}}(X, X)$
4. a *composition function* $f \in Hom_{\mathcal{C}}(X, Y) \times g \in Hom_{\mathcal{C}}(Y, Z) \to g \circ f \in Hom_{\mathcal{C}}(X, Z)$

such that

1. *unitality:* for every morphism $X \xrightarrow{f} Y$, $f \circ id_x = f = id_y \circ f$
2. *associativity:* if any three morphisms f, g, h are composable,

$$X \xrightarrow{f} Y \xrightarrow{g} Z \xrightarrow{h} W$$

Add records → Add to base space



Fiber space: Fields

types and schema	record	table
$\begin{tikzcd} \mathcal{U}_\sigma \arrow[r, dotted] \arrow[d, two heads, "\pi_\sigma"'] \arrow[dr, dotted] & \mathcal{U} \arrow[d, two heads, "\pi"] \\ C \arrow[r, "\sigma"'] & \mathtt{DT} \end{tikzcd}$	$\begin{tikzcd} \mathcal{U}_\sigma \arrow[d, two heads, "\pi_\sigma"'] \\ C \arrow[u, dashed, bend right, "r"'] \end{tikzcd}$	$\begin{tikzcd} F \coloneqq \Gamma^\pi(\sigma) \arrow[d, two heads, "\pi"'] \\ K \arrow[u, dashed, bend right, "\tau"'] \end{tikzcd}$

- $\mathcal{U}$ Space of all possible values, e.g. $\mathbb{R} \times \mathbb{R}^+$
- C Field names, e.g. *TEMP*, *PRCP*
- $\mathtt{DT}$ Data Type, e.g. `float`, `PosFloat`

Types and Schema

A *schema* consists of a pair (C, σ) where C is the set of field names and $\sigma : C \to \mathbf{DT}$ is a function from field name to field data type[33] The function σ is composed with π such that $\pi^{-1}(\sigma(C)) \subseteq \mathscr{U}$; this composition induces a domain bundle $\pi_\sigma : \mathscr{U}_\sigma \to C$ that associates a field name $c \in C$ with its corresponding domain $\pi_\sigma^{-1}(C) \subseteq \mathscr{U}_\sigma$.

Record

Definition

A **record** is a function $r : C \to \mathscr{U}_\sigma$ and the set of records on π_σ is denoted $\Gamma^\pi(\sigma)$. Records must return an object of type $\sigma(C) \in \mathbf{DT}$ for each field $c \in C$.

Tables are sections $\tau : K \to \Gamma^\pi(\sigma)$ from an indexing space K to the set of all possible records $\Gamma^\pi(\sigma)$ on the schema bundle

Fiber TYPE F

We define the fiber space F to be the space of all possible data records

$$F \coloneqq \{r : C \to \mathscr{U}_\sigma \mid \pi_\sigma(r(C)) = C \textit{ for all } C \in C\} \tag{3}$$

such that the preimage of a point is the corresponding data type domain $\pi^{-1}(k) = F_k = \mathscr{U}_{\sigma_k}$.

Fiber category F

The fiber category has a single object F of an arbitrary type and morphisms on the fiber object $\tilde{\phi} \in Hom(F, F)$.
The fiber category $\mathcal{F}$ is also equipped with a bifunctor $\otimes : \mathcal{F} \times \mathcal{F} \to \mathcal{F}$ for combining fiber types.

Fiber morphism $\tilde{\phi}$

scale	operators	sample constraint
nominal	$=, \neq$	$\tau(k_1) \neq \tau(k_2) \implies \tilde{\phi}(\tau(k_1)) \neq \tilde{\phi}(\tau(k_2))$
ordinal	$<, \leqslant, \geqslant, >$	$\tau(k_1) \leqslant \tau(k_2) \implies \tilde{\phi}(\tau(k_1)) \leqslant \tilde{\phi}(\tau(k_2))$
interval	$+, -$	$\tilde{\phi}(\tau(k) + C) = \tilde{\phi}(\tau(k)) + C$
ratio	$*, /$	$\tilde{\phi}(\tau(k) * C) = \tilde{\phi}(\tau(k)) * C$

Constraints as definition: Dates

- year: $\{y \in \mathbb{I} | 1992 \leqslant y \leqslant 2025\}$
- month: $\{m \in \mathbb{I} | 1 \leqslant m \leqslant 12\}$
- day: $\{d \in \mathbb{I} | 1 \leqslant d \leqslant 31\}$
- date: $\otimes : F_{year} \times F_{month} \times F_{day} \rightarrow F_{date}$

where the composition $\otimes$ includes the constrain to only return dates that have the right number of days for each month.

Expand records → multiply fiber space

$$
\begin{tikzcd}
\overset{F_a \times_{F_c} Fb}{\text{59°F,Oct 02,0.07in.}} \arrow[r, "\pi_b"] \arrow[d, "\pi_a"] & \overset{F_b}{\text{Oct 02,0.07in.}} \arrow[d, "\pi_c"] \\
\underset{F_a}{\text{59°F,Oct 02}} \arrow[r, "\pi_c"] & \underset{F_c}{\text{Oct 02}}
\end{tikzcd}
$$

local sections

We define section functions locally:

$$\Gamma(U, E\restriction_U) \coloneqq \left\{\tau : U \to E\restriction_U \;\middle|\; \pi(\tau(k)) = k \textit{ for all } k \in U\right\} \tag{4}$$

such that each section function $\tau : k \mapsto r$ maps from each point $k \in U$ to a corresponding record in the fiber space $r \in F_k$ over that point.

This encoding can be translated into a set of signatures $\{\texttt{data-subset} : \texttt{topology} \to \texttt{fields}$ s.t. $\texttt{data-subset} \subset \texttt{dataset}\}$.

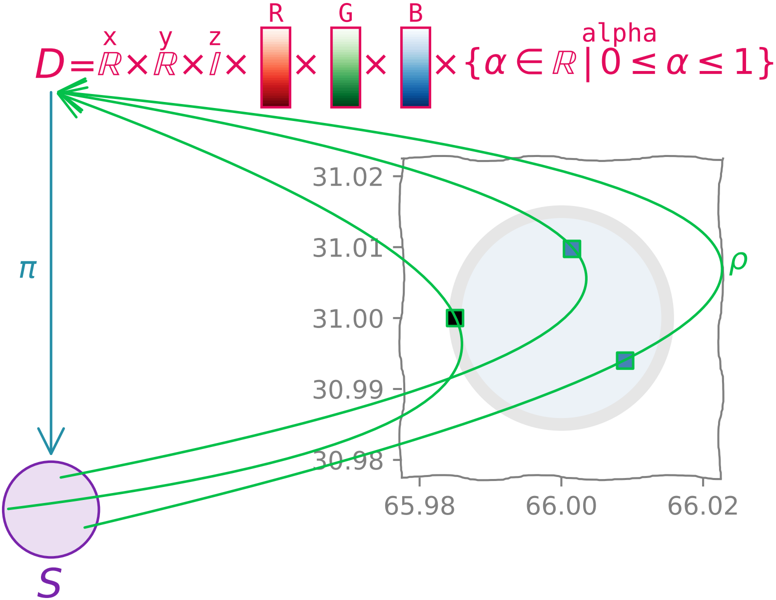
global sections

When a bundle is trivial $E = K \times F$, we can defined a global sections $\tau : K \to F \in \Gamma(K, F)$ which we translate into a data signature of the form $\texttt{dataset} : \texttt{topology} \to \texttt{field}$ where $\tau = \texttt{dataset}$, $K = \texttt{topology}$ and $F = \texttt{fields}$

Graphic bundle and section

$$D \hookrightarrow H \xrightarrow{\pi} \twoheadrightarrow S \tag{5}$$

$$\begin{aligned} \Gamma(W, H{\restriction}_W) \coloneqq & \\ & \left\{ \rho : W \to H{\restriction}_W \;\middle|\; \pi(\rho(s)) = s \textit{ for all } s \in W \right\} \end{aligned} \tag{6}$$


R
G
B
alpha
x
y
z
D = ℝ × ℝ × I × × × × {α ∈ ℝ | 0 ≤ α ≤ 1}
π
ρ
S
31.02
31.01
31.00
30.99
30.98
65.98
66.00
66.02

Definition

[43, 44] A **functor** is a map $F : \mathcal{C} \to \mathcal{D}$, which means it is a function between objects $F : \mathbf{ob}(\mathcal{C}) \mapsto \mathbf{ob}(\mathcal{D})$ and that for every morphism $f \in Hom(C_1, C_2)$ there is a corresponding function $F : Hom(C1, C2) \mapsto Hom(F(C_1), F(C_2))$. A **functor** must satisfy the properties

- *identity*: $F(id_C(C)) = id_D(F(C))$
- *composition*: $F(g) \circ F(f) = F(g \circ f)$ for any composable morphisms $C_1 \xrightarrow{f} C_2$, $C_2 \xrightarrow{g} C_3$

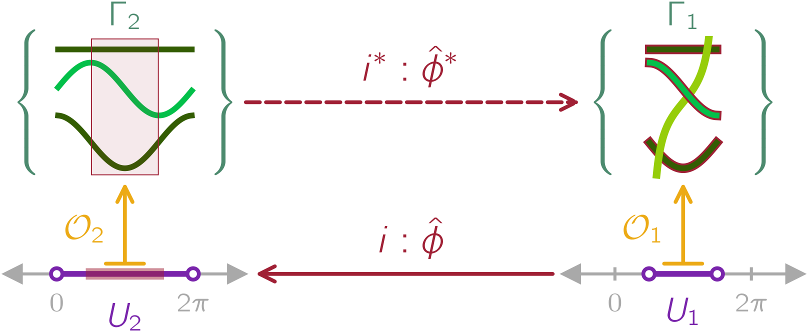
$F(C) \in \mathbf{ob}(\mathcal{D})$ denotes the object to which an object C is mapped, and $F(f) \in Hom(F_(C_1), F_(C_2))$ denotes the morphism that f is mapped to.

Definition

A **presheaf** $F : \mathcal{C}^{op} \rightarrow \boldsymbol{Set}$ is a contravariant functor from an object in an arbitrary category to an object in the category $\boldsymbol{Set}$[36].

The presheaf is contravariant when for every arbitrary morphism between input base spaces $\hat{\phi} : U_1 \rightarrow U_2$ there exists a corresponding pullback function between the sets of sections $\hat{\phi}^* : \Gamma(U_2, E\restriction_{U_2}) \rightarrow \Gamma(U_1, E\restriction_{U_c1})$.

topological equivariance: presheaf



Container invariance: sheaf

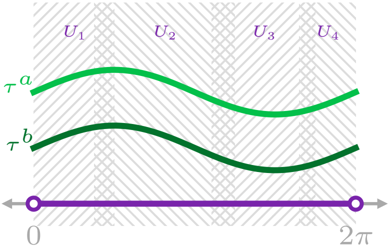
Definition

[45, 36] A **sheaf** is a presheaf that satisfies the following two axioms

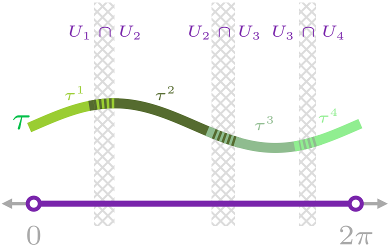
- *locality* two sections in a sheaf are equal $\tau^a = \tau^b$ when they evaluate to the same values $\tau^a|_{U_i} = \tau^b|_{U_i}$ over the open cover $\bigcup_{i \in I} U_i \subset U$ (indexed by I).
- *gluing* the union of sections defined over subspaces $\tau^i \in \Gamma(U_i, E|_{U_i})$ is equivalent to a section defined over the whole space $\tau|_{U_i} = \tau^i$ for all $i \in I$ if all pairs of sections agree on overlaps $\tau^i|_{U_i \cap U_j} = \tau^j|_{U_i \cap U_j}$

Container invariance: sheaf

locality: $\tau^a = \tau^b$



gluing: $\tau|_{U_i} = \tau^i$



What about needing more than one record?

The sheaf over a very small region surrounding a point k is called a *stalk*[46]

$$\mathcal{O}_{K,E}\restriction_k \coloneqq \lim_{U \ni k} \Gamma(U, E\restriction_U) \tag{7}$$

where the fiber is contained inside the stalk $F_k \subset \mathcal{O}_{K,E}\restriction_k$. The *germ* is the section evaluated at a point in the stalk $\tau(k) \in \mathcal{O}_{K,E}\restriction_k$ and is the data.

Homeomorphism

We define the mapping ξ to be a surjective continuous map:

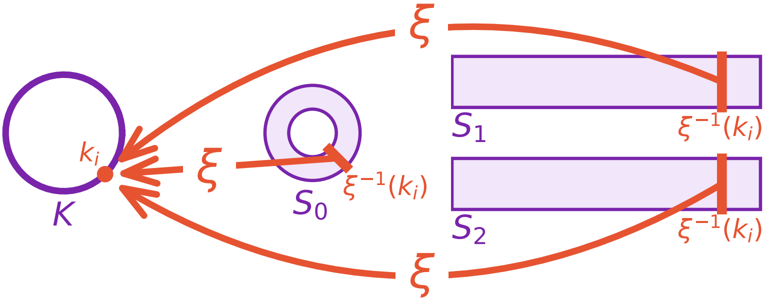
$$\xi : W \rightarrow U \tag{8}$$

between a graphic subspace $W \subseteq S$ and data subspace $U \subseteq K$.
The set of points in graphic space that correspond to each point in data space is

$$\xi^{-1}(k) = \{s | \xi(s) = k \forall k \in K, s \in S\} \tag{9}$$

such that every point in a graphic space has a corresponding point in data space.

Homeomorphism



Pushforward: Graphic for Data

Definition

Given a sheaf $\mathcal{O}_{S,H}$ on S, the **pushforward** sheaf $\xi_*\mathcal{O}_{S,H}$ on K is defined as

$$\xi_*(\mathcal{O}_{S,H})(U) = \mathcal{O}_{S,H}(\xi^{-1}(U)) \tag{10}$$

for all opensets $U \subset K$[46].

This provides a way to look up which graphic corresponds with a data index

$$\xi_*\rho(k) = \rho\restriction_{\xi^{-1}(k)} \tag{11}$$

such that $\xi_*\rho(k))(s) = \rho(s)$ for all $s \in \xi^{-1}(k)$.

Pullback: Data at graphic

Definition

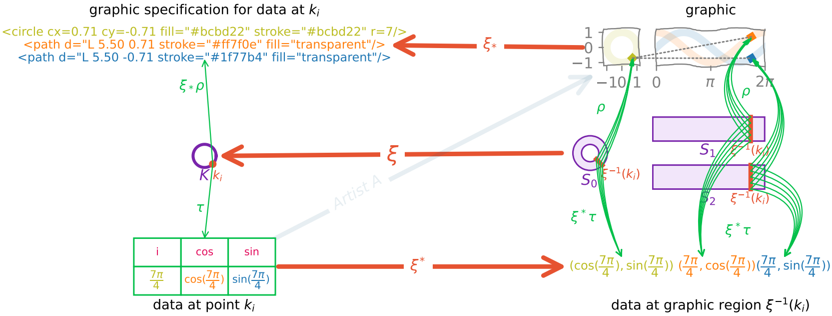
[46] Given a sheaf $\mathcal{O}_{K,E}$ on K, the **pullback** sheaf $\xi^*\mathcal{O}_{K,E}$ on S is defined as the sheaf associated to the presheaf $\xi^*(\mathcal{O}_{K,E})(W) = \mathcal{O}_{K,E}(\xi(W))$ for $\xi(W) \in K$.

This provides a way to then look up what data values correspond with a graphic index

$$\xi^*\tau(s) = \tau(\xi(s)) = \tau(k) \tag{12}$$

As ξ is surjective, there are many points $s \in W \subseteq S$ in the graphic space that correspond to a single point $\xi(s) = k$.

Homeomorphism



Definition

Given two functors $F, G : \mathcal{C} \rightarrow \mathcal{D}$, a **natural transformation** $\alpha : F \rightarrow G$ is a function which assigns to each object c of $\mathcal{C}$ a morphism $\alpha_c : F(input) \rightarrow G(c), G(c) \in \mathcal{D}$, in such a way that for every morphism $f : c \rightarrow c', c' \in \mathcal{C}$, the morphisms in $\mathcal{D}$ commute such that $\alpha'_c(F(f)(F(c))) = G(f)(\alpha_c(F(c)))$. When this holds, α_c is *natural* in c.[41].

Artist

$$
\begin{tikzcd}
\texttt{topology} \arrow[rr, bend left, "\texttt{data}"] \arrow[rr, bend right, "\texttt{graphic}"'] \arrow[rr, phantom, "\Downarrow\,\texttt{artist}"] & & \texttt{topology} \arrow[r] & \texttt{fields}
\end{tikzcd}
$$

How?

$$
\begin{tikzcd}
\Gamma(U, \xi_* H\restriction_U) & & \Gamma(W, ,)H\restriction_W \arrow[ll, "\xi_*"'] \\
\Gamma(U, E\restriction_U) \arrow[u, "Hom_{\mathcal{O}_K}"] \arrow[urr, "Hom_{\mathcal{O}_K, \mathcal{O}_S}"] \arrow[rr, "\xi^*"] & & \Gamma(W, \xi^* E\restriction_W) \arrow[u, "Hom_{\mathcal{O}_S}"']
\end{tikzcd}
\tag{13}
$$

Artist

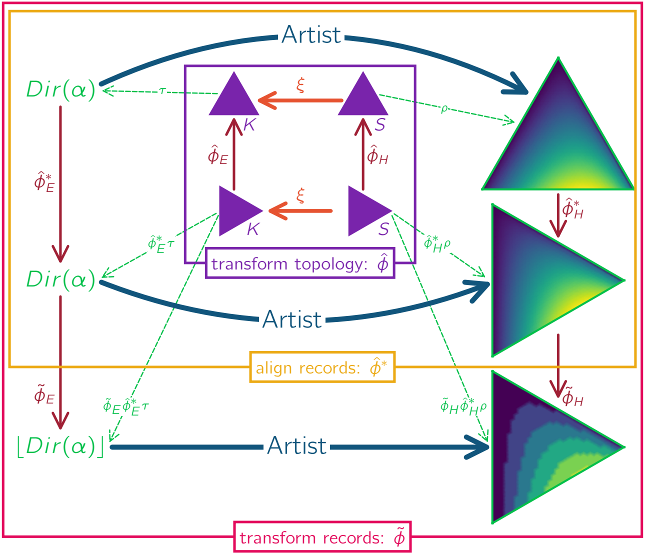
$$A : \Gamma(K, E) \rightarrow Im_A(S, H), Im_A(S, H) \subset \Gamma(S, H) \tag{14}$$

where:

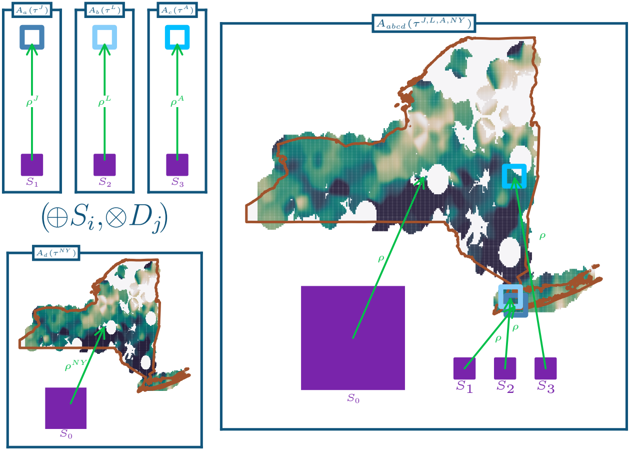
$$Im_A(S, H) \coloneqq \{\rho \mid \exists \tau \in \Gamma(K, E) \; s.t. \; A(\tau) = \rho, \; \xi(S) = K\} \tag{15}$$

$$
\begin{tikzcd}
\Gamma(K, E) \arrow[rrrr, "A"] \arrow[d, "\hat{\phi}^*{}_E"'] & & & & Im_A(S, H) \arrow[d, dotted, "\hat{\phi}^*{}_H"] \\
\Gamma(K', \hat{\phi}^*{}_E E) \arrow[dd, "\tilde{\phi}_E"'] & K & & S \arrow[ll, "\xi"'] & Im_A(S', \hat{\phi}^*{}_H H) \arrow[dd, dotted, "\tilde{\phi}_H"] \\
 & K' \arrow[u, "\hat{\phi}_E"] & & S' \arrow[ll, "\xi"'] \arrow[u, dotted, "\hat{\phi}_H"'] & \\
\Gamma(K', E') \arrow[rrrr, "A"] & & & & Im_A(S', H')
\end{tikzcd}
\tag{16}
$$

Visualizing the Dirichlet Distribution



$$\bigsqcup_i \prod_j A(Hom(K_i, F_j)) = A(Hom(\bigsqcup_i K_i, \prod_j F_j)) \implies$$



$$\bigsqcup_i \prod_j Hom(S_i, D_j)) = Hom(\bigsqcup_i S_i, \prod_j D_j)$$

Animation and Interactivity

pan, zoom, scroll sheaf: locality + gluing 12

selection and hover pushforward 11, pullback 12

brushing, linking, annotation composition of artists **??**

Testing if A is equivariant

M is a (scaler, vector) measurable component (e.g. color, position, shape, texture, rotation,) of the rendered visual element.

$$\begin{tikzcd}
\Gamma(U, E\restriction_U) \arrow[r, "A"] \arrow[d, "\eta"'] & Im_A(W, H\restriction_W) \arrow[r, "render"] \arrow[d, "\delta"] & visualization \arrow[dl, "measure"] \\
Hom(U, M) \arrow[r, "\xi^*"'] & Hom(W, M) &
\end{tikzcd}$$

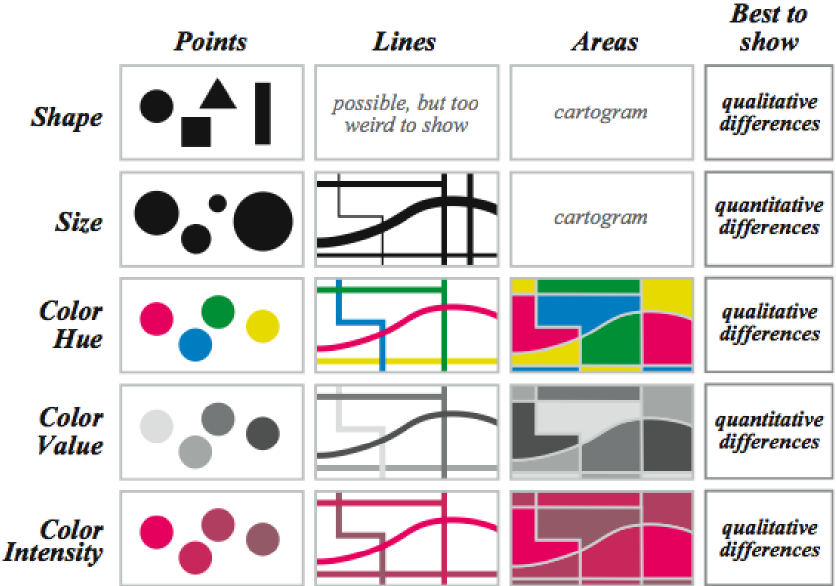
$$\text{input } \eta : \tau \mapsto (U \xrightarrow{\eta_\tau} M)$$

$$\text{output } \delta : \rho \mapsto (W \xrightarrow{\delta_\rho} M)$$

$$\eta_\tau(k) = \delta_\rho(s) \text{ for all } \xi(s) = k, k \in K, s \in S$$

Visual Space: Specialized data space

$P \hookrightarrow V \xrightarrow{\pi} K$



Encoding

$$\nu : F_k \to P_k \tag{17}$$

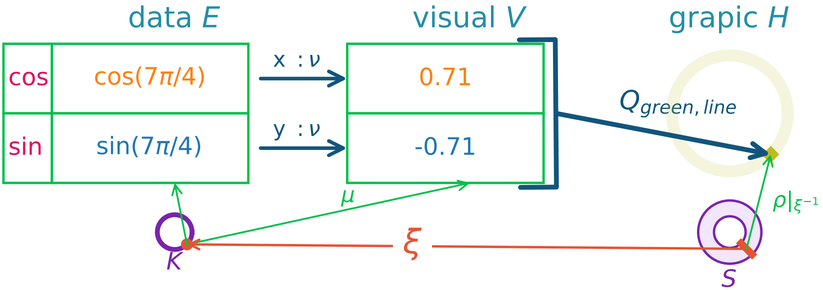
$$\begin{tikzcd}
F_k \arrow[r, "\nu"] \arrow[rr, bend right, dashed, "\nu''"'] & P_k \coloneqq F'_k \arrow[r, "\nu'"] & P'_k
\end{tikzcd} \tag{18}$$

Composition

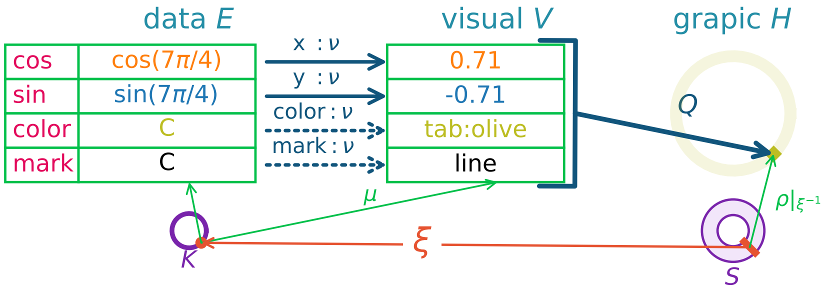
$$Q : \Gamma(K, V) \to \Gamma(S, H) \tag{19}$$

$$\begin{tikzcd}
\Gamma(K, V) \arrow[r, "\nu"] \arrow[rr, bend right, dashed, "Q'"] & \Gamma(K, V') \arrow[r, "Q"] & \Gamma(S, H)
\end{tikzcd} \tag{20}$$

Construction Stages



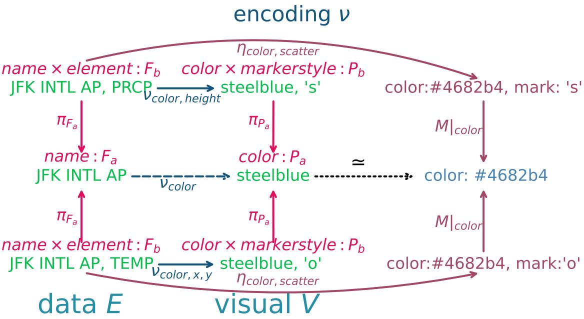
Construction Stages: composition



Verify encoder

$$
\begin{tikzcd}
F_k^a \times F_k^b \arrow[r, "\nu_{ab}"] \arrow[d, "\pi_a"'] \arrow[rr, bend left, "\eta_{ab}"] & P_k^a \times P_k^b \arrow[d, "\pi_a"] & M_k^{ab} \arrow[d, "M\restriction_a"] \\
F_k^a \arrow[r, dashed, "\nu_a"] & P_k^a \arrow[r, dotted, "\simeq"] & M_k^a \\
F_k^a \times F_k^c \arrow[r, "\nu_{ac}"] \arrow[u, "\pi_a"] \arrow[rr, bend right, "\eta_{ac}"'] & P_k^a \times P_k^c \arrow[u, "\pi_a"'] & M_k^{ac} \arrow[u, "M\restriction_a"']
\end{tikzcd}
\tag{21}
$$

Verify encoder



Verify compositor

$$
\begin{tikzcd}
\Gamma(K, V^a \times V^b) \arrow[r, "Q_{ab}"] \arrow[d, "\pi_a"'] & Im_{\underset{ab}{A}}(S, H) \arrow[d, "M\restriction_a \circ \delta_{ab}"] \\
\Gamma(K, V^a) \arrow[r, dotted, "\simeq"] & Hom(K, M^a) \\
\Gamma(K, V^a \times V^c) \arrow[r, "Q_{ac}"] \arrow[u, "\pi_a"] & Im_{\underset{ac}{A}}(S, H) \arrow[u, "M\restriction_a \circ \delta_{ac}"']
\end{tikzcd}
\tag{22}
$$

Verify compositor

composition Q

$$
\begin{tikzcd}[column sep=huge]
\begin{array}{c} \mathit{color} \times \mathit{markerstyle} : P_b \\ \text{steelblue, 's'} \end{array} \arrow[r, "Q_{square}"] \arrow[d, "\pi_{P_a}"'] & \square \arrow[d, "M|_{P_a} \circ \delta_{P_a,\, square}"] \\
\begin{array}{c} \mathit{color} : P_a \\ \text{steelblue} \end{array} \arrow[r, dashed, "\simeq"] & \text{color: \#4682b4} \\
\begin{array}{c} \mathit{color} \times \mathit{markerstyle} : P_b \\ \text{steelblue, 'o'} \end{array} \arrow[r, "Q_{circle}"] \arrow[u, "\pi_{P_a}"] & \bigcirc \arrow[u, "M|_{P_a} \circ \delta_{P_a,\, circle}"']
\end{tikzcd}
$$

visual V | graphic H

Implementation Choices: $A_K = A_S$

$$\begin{tikzcd}
\Gamma(W, \xi^* E\restriction_W) \arrow[r, "\nu_S"] \arrow[rr, bend left, "A_S"] & \Gamma(W, \xi^* V\restriction_W) \arrow[r, "Q_S"] & Im_A(W, H\restriction_W) \arrow[d, "\xi_*"] \\
\Gamma(U, E\restriction_U) \arrow[u, "\xi^*"] \arrow[r, "\nu_K"] \arrow[rr, bend right, "A_K"'] & \Gamma(U, V\restriction_U) \arrow[u, "\xi^*"] \arrow[r, "Q_K"] & Im_A(U, \xi_* H\restriction_U)
\end{tikzcd}$$

References I

[1] L. Wilkinson. *The Grammar of Graphics*. 2nd ed. Statistics and Computing. New York: Springer-Verlag New York, Inc., 2005.

[2] R. P. Grimaldi. *Discrete and Combinatorial Mathematics, 5/e*. Pearson Education, 2006.

[3] A. M. Pitts. *Nominal Sets: Names and Symmetry in Computer Science*. USA: Cambridge University Press, 2013.

[4] J. Bertin. *Semiology of Graphics : Diagrams, Networks, Maps*. Redlands, Calif.: ESRI Press, 2011.

[5] J. Mackinlay. "Automatic Design of Graphical Presentations". Stanford, 1987.

[6] G. Kindlmann and C. Scheidegger. "An Algebraic Process for Visualization Design". In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (Dec. 2014), pp. 2181–2190.

References II

[7] H. Wickham. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

[8] C. A. Schneider, W. S. Rasband, and K. W. Eliceiri. "NIH Image to ImageJ: 25 Years of Image Analysis". In: *Nature Methods* 9.7 (July 2012), pp. 671–675.

[9] M. Bastian, S. Heymann, and M. Jacomy. "Gephi: An Open Source Software for Exploring and Manipulating Networks". In: International AAAI Conference on Weblogs and Social Media. Vol. 3. Proceedings of the International AAAI Conference on Web and Social Media. 2009.

[10] A. Satyanarayan, K. Wongsuphasawat, and J. Heer. "Declarative Interaction Design for Data Visualization". In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. Honolulu Hawaii USA: ACM, Oct. 2014, pp. 669–678.

References III

[11] S. Studies. *Culturevis/Imageplot*.

[12] J. Ellson et al. "Graphviz— Open Source Graph Drawing Tools". In: *Graph Drawing*. Ed. by P. Mutzel, M. Jünger, and S. Leipert. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 483–484.

[13] J. VanderPlas et al. "Altair: Interactive Statistical Visualizations for Python". In: *Journal of Open Source Software* 3.32 (Dec. 2018), p. 1057.

[14] N. Sofroniew et al. *Napari/Napari: 0.4.5rc1*. Version v0.4.5rc1. Zenodo, Feb. 2021.

References IV

[15] A. A. Hagberg, D. A. Schult, and P. J. Swart. "Exploring Network Structure, Dynamics, and Function Using NetworkX". In: *Proceedings of the 7th Python in Science Conference*. Ed. by G. Varoquaux, T. Vaught, and J. Millman. Pasadena, CA USA, 2008, pp. 11–15.

[16] C. Stolte, D. Tang, and P. Hanrahan. "Polaris: A System for Query, Analysis, and Visualization of Multidimensional Relational Databases". In: *IEEE Transactions on Visualization and Computer Graphics* 8.1 (Jan. 2002), pp. 52–65.

[17] P. Hanrahan. "VizQL: A Language for Query, Analysis and Visualization". In: *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*. SIGMOD '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 721.

References V

[18] J. Mackinlay, P. Hanrahan, and C. Stolte. "Show Me: Automatic Presentation for Visual Analysis". In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (Nov. 2007), pp. 1137–1144.

[19] J. Heer and M. Agrawala. "Software Design Patterns for Information Visualization". In: *IEEE Transactions on Visualization and Computer Graphics* 12.5 (2006), pp. 853–860.

[20] J. D. Hunter. "Matplotlib: A 2D Graphics Environment". In: *Computing in Science Engineering* 9.3 (May 2007), pp. 90–95.

[21] M. L. Waskom. "Seaborn: Statistical Data Visualization". In: *Journal of Open Source Software* 6.60 (2021), p. 3021.

[22] S. Hoyer and J. Hamman. "Xarray: ND Labeled Arrays and Datasets in Python". In: *Journal of Open Research Software* 5.1 (2017).

References VI

[23] M. Bostock, V. Ogievetsky, and J. Heer. "D³ Data-Driven Documents". In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (Dec. 2011), pp. 2301–2309.

[24] M. D. Hanwell et al. "The Visualization Toolkit (VTK): Rewriting the Rendering Code for Modern Graphics Cards". In: *SoftwareX* 1–2 (Sept. 2015), pp. 9–12.

[25] B. Geveci et al. "VTK". In: *The Architecture of Open Source Applications* 1 (2012), pp. 387–402.

[26] P. Ramachandran and G. Varoquaux. "Mayavi: 3D Visualization of Scientific Data". In: *Computing in Science Engineering* 13.2 (Mar. 2011), pp. 40–51.

[27] Brian Wylie and Jeffrey Baumes. "A Unified Toolkit for Information and Scientific Visualization". In: *Proc.SPIE.* Vol. 7243. Jan. 2009.

[28] J. Ahrens, B. Geveci, and C. Law. "Paraview: An End-User Tool for Large Data Visualization". In: *The visualization handbook* 717.8 (2005).

[29] K. Wongsuphasawat. "Navigating the Wide World of Data Visualization Libraries (on the Web)". Outlier Conf. 2021.

[30] M. Tory and T. Moller. "Rethinking Visualization: A High-Level Taxonomy". In: *IEEE Symposium on Information Visualization.* 2004, pp. 151–158.

[31] D. M. Butler and S. Bryson. "Vector-Bundle Classes Form Powerful Tool for Scientific Visualization". In: *Computers in Physics* 6.6 (1992), p. 576.

[32] D. M. Butler and M. H. Pendley. "A Visualization Model Based on the Mathematics of Fiber Bundles". In: *Computers in Physics* 3.5 (1989), p. 45.

[33] D. I. Spivak. *Simplicial Databases*. 2009. URL: https://arxiv.org/abs/0904.2012. Pre-published.

[34] V. Wiels and S. Easterbrook. "Management of Evolving Specifications Using Category Theory". In: *Proceedings 13th IEEE International Conference on Automated Software Engineering (Cat. No.98EX239)*. 1998, pp. 12–21.

[35] R. W. Ghrist. *Elementary Applied Topology*. Vol. 1. Createspace Seattle, 2014.

[36] E. Spanier. *Algebraic Topology*. McGraw-Hill Series in Higher Mathematics. Springer, 1989.

[37] J. R. Munkres. *Elements of Algebraic Topology*. Menlo Park, Calif: Addison-Wesley, 1984.

[38] R. Brown. *Topology and Groupoids*. 2006.

References IX

[39] F. W. Lawvere and S. H. Schanuel. *Conceptual Mathematics: A First Introduction to Categories*. Cambridge University Press, 2009.

[40] E. Riehl. *Category Theory in Context*. Dover Publications, Nov. 16, 2016.

[41] S. Mac Lane. *Categories for the Working Mathematician*. Vol. 5. Springer Science & Business Media, 2013.

[42] B. Fong and D. I. Spivak. *An Invitation to Applied Category Theory: Seven Sketches in Compositionality*. 1st ed. Cambridge University Press, July 2019.

[43] T.-D. Bradley. *What Is a Functor? Definitions and Examples, Part 2*. Math3ma. URL: https://www.math3ma.com/blog/what-is-a-functor-part-2 (visited on 10/07/2021).

References X

[44] T.-D. Bradley, J. Terilla, and T. Bryson. *Topology : A Categorical Approach*. MIT Press, 2020.

[45] M. J. Baker. *EuclideanSpace: Maths - Sheaf*. URL: https://www.euclideanspace.com/maths/topology/sheaf/ (visited on 01/06/2022).

[46] G. Harder and K. Diederich. *Lectures on Algebraic Geometry I: Sheaves, Cohomology of Sheaves, and Applications to Riemann Surfaces*. Aspects of Mathematics. Vieweg+Teubner Verlag, 2008.