A gentle introduction to sheaves on graphs

Jakob Hansen

Abstract

This document is an introduction to the language and theory of cellular sheaves on graphs with an eye toward engineering and other applications. No familiarity with topology or commutative algebra is assumed. However, a basic familiarity with graphs, particularly spectral graph theory, and a working knowledge of linear algebra from an abstract perspective (e.g. abstract vector spaces, quotients, direct sums and tensor products, inner products) are necessary.

1 Motivation: connections and relationships

Networks are everywhere. The patterns of connectivity between different entities (e.g., people, computers, ideas, neurons) have important consequences for the properties of systems constructed of these entities. This is the foundational insight behind network science and its allied fields, and it has proven immensely useful.

But sometimes, just knowing that two things are connected is not enough. We need to know something about the *relationship* implied by the connection. Sheaves offer a mathematically rigorous for approaching problems like this. They describe the local relationships associated to data on a network, and prescribe how those local relationships combine to give global structure to the data. Sheaves have a sophisticated and deep algebraic theory, as well as a nascent spectral theory, both of which offer powerful tools for understanding networks and systems.

Some of the things that sheaves can do:

- Collate high-dimensional data associated with nodes of a network
- Describe complex relationships between different nodes
- Provide multilayer and multiresolution views of network structures
- Implement sophisticated types of consensus dynamics
- Represent various engineering and physical systems.

I aim here to give an accessible introduction to the theory of sheaves on graphs, with hints at its potential applications.

2 What is a sheaf?

Sheaves are the canonical mathematical structure for attaching data to spaces. In algebraic topology and geometry, this can take many sophisticated, subtle forms, but we will deal with a simple version adapted to graphs and simplicial complexes. For our purposes, a graph is a collection of vertices and edges, together with an *incidence relation*, which tells us when a given vertex is incident to a given edge. That is:

Definition 2.1 (Graphs)**.** A *graph* G consists of a set V of vertices and a set E of edges, with an incidence relation $\trianglelefteq$, where we write $v \trianglelefteq e$ if v is one of the endpoints of the edge e. There may be more than one edge joining a given pair of vertices.

A sheaf is a way of specifying data spaces associated to edges and vertices of a graph, with extra information telling us how the data over different parts of the graph should be related. Our data will be vector valued, so our theory will involve linear algebra. We will deal only with real vector spaces that carry an inner product, although this restriction is not necessary. Most of the theory works for vector spaces over any field, unless an inner product is required. This restriction means that there is little lost by considering every vector space in sight to be isomorphic to $\mathbb{R}^n$ for some n, with the standard inner product.

Definition 2.2 (Sheaves)**.** Let G be a graph. A sheaf $\mathcal{F}$ on G consists of a vector space $\mathcal{F}(v)$ for each vertex v of G, a vector space $\mathcal{F}(e)$ for each edge e of G, and a linear transformation $\mathcal{F}_{v \trianglelefteq e} : \mathcal{F}(v) \to \mathcal{F}(e)$ for each incident vertex-edge pair $v \trianglelefteq e$.

The vector spaces $\mathcal{F}(v)$ and $\mathcal{F}(e)$ are called the *stalks* of $\mathcal{F}$ over v or e. The linear maps $\mathcal{F}_{v \trianglelefteq e}$ are sometimes called *restriction maps*. In agricultural terminology, a sheaf is a collection of stalks of grain bound together by twine; in mathematical terminology, a sheaf is a collection of stalks of data bound together by linear maps. The script letter $\mathcal{F}$ is frequently used to denote a sheaf; it is short for *faisceau*, the French word for sheaf.

The maps $\mathcal{F}_{v \trianglelefteq e}$ encode consistency requirements for our data. Whenever consistency can be verified locally, we have a sheaf structure. The simplest example of this is checking whether a real-valued function on the vertices of a connected graph is constant. This consistency condition can be verified locally: one only needs to check that the function is constant across each edge. Sheaf theory vastly generalizes this sort of local consistency checking.

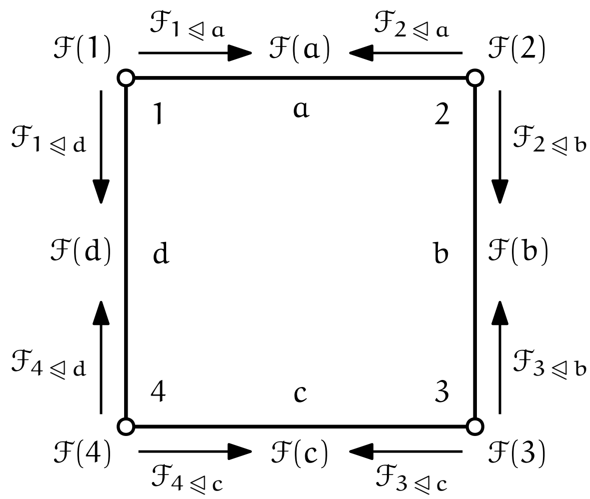

Figure 1: An illustration of the structure of a sheaf over a graph.

Definition 2.3 (Sections)**.** Let $\mathcal{F}$ be a sheaf on a graph G, and let W be a subset of the vertices of G. A *section* of $\mathcal{F}$ over W is an choice of vectors $x_v \in \mathcal{F}(v)$ for each $v \in W$, such that whenever v and v' are vertices incident to an edge e, we have $\mathcal{F}_{v \trianglelefteq e}(x_v) = \mathcal{F}_{v' \trianglelefteq e}(x_{v'})$.

Note that there is a natural way to add any two sections of $\mathcal{F}$ over W, simply by adding their values in each stalk. Because the restriction maps are linear, the sum of two sections is again a section. Similarly, there is a natural scalar multiplication on sections, computed stalkwise. This means that the sections of $\mathcal{F}$ over W form a vector space. This space is variously denoted $\mathcal{F}(W)$, $\Gamma(W;\mathcal{F})$, or $H^0(W;\mathcal{F})$. The space $\Gamma(G;\mathcal{F})$ of sections of $\mathcal{F}$ over all vertices in G is called the space of *global sections* of $\mathcal{F}$. Further, note that a global section (x_v) uniquely determines values x_e on the edges, by $x_e = \mathcal{F}_{v \trianglelefteq e}(x_v) = \mathcal{F}_{v' \trianglelefteq e}(x_{v'})$.

Exercise. Verify these claims, that is, show that $\Gamma(W;\mathcal{F})$ satisfies the axioms of a vector space, with the specified addition and scalar multiplication operations.

Remark. The term *section* comes from "cross section." One often thinks of a sheaf as having a horizontal portion given by the underlying graph, with a vertical dimension added by the stalks. A section consists of choosing a consistent copy of the graph within the stalks of the sheaf, or a cross section of the vertical part of the sheaf.

Example. Let V be a vector space. The *constant sheaf* $\underline{V}$ on G has the stalks $\underline{V}(v) = V$ for all vertices v and $\underline{V}(e) = V$ for all edges e, and the restriction maps $\mathcal{F}_{v \trianglelefteq e}$ all equal to the identity. For any subset of vertices W, $\Gamma(W;\underline{V})$ consists of the *locally constant*

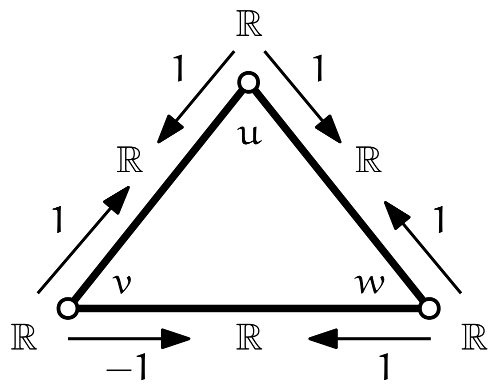

Figure 2: A sheaf with no nontrivial global sections.

functions on W with values in V. (A function is locally constant if it is constant on each connected component of the subgraph induced by W.)

Example. As a subexample of the previous example, let $V = \mathbb{R}$. The constant sheaf $\underline{\mathbb{R}}$ plays a special role in the theory of sheaves. Its relationship to a sheaf $\mathcal{F}$ determines the existence of global sections of $\mathcal{F}$.

Every sheaf has at least one global section: the constant section which is zero everywhere. (This is a consequence of the fact that the space of global sections of a sheaf is a vector space, and every vector space has a zero element.) However, many sheaves have no other global sections.

Example. Consider the sheaf in Figure 2. Suppose there is a global section taking value x_u on the vertex u. The condition to be a global section then means $x_v = x_u$ due to the upper left edge, and similarly $x_w = x_u$ due to the upper right edge. But the bottom edge forces $x_w = -x_v$, meaning $x_u = -x_u$, which can only be satisfied if all three values are zero.

In fact, having nontrivial global sections is a very special property. If G is a connected graph with at least one cycle and all stalks have the same dimension, almost any choice of restriction maps yields a sheaf with no global sections. One consequence of this fact is that if we try to build a sheaf from noisy real-world measurements, the sheaf is vanishingly unlikely to have global sections, and we will need either to denoise our data, or to work with approximate global sections.

Definition 2.4 (Cochains)**.** Let $\mathcal{F}$ be a sheaf over a graph G. There are two spaces of *cochains* of $\mathcal{F}$. A zero-dimensional cochain (or 0-cochain) is a choice of a vector $x_v \in \mathcal{F}(v)$ for every vertex v; a one-dimensional cochain (or 1-cochain) is a choice of $x_e \in$

$\mathcal{F}(e)$ for every edge. The space of zero-dimensional cochains is denoted $C^0(G;\mathcal{F})$, and the space of one-dimensional cochains is denoted $C^1(G;\mathcal{F})$.

In mathematical terms, the space of 0-cochains is the direct sum of the stalks over vertices, and the space of 1-cochains is the direct sum of the stalks over edges. That is,

$$C^0(G;\mathcal{F}) = \bigoplus_{v\in V} \mathcal{F}(v), \quad C^1(G;\mathcal{F}) = \bigoplus_{e\in E} \mathcal{F}(e).$$

The space of 0-cochains differs from the space of global sections because we do not impose any consistency requirements on our choices of vectors in each stalk. Therefore, the space of global sections is a subspace of the space of 0-cochains. In fact, the space of global sections is the kernel of a particular linear map from the space of 0-cochains to the space of 1-cochains. This map is called the *coboundary map*, and is defined as follows: First choose an orientation for the edges of G. This means selecting a direction for each edge, and yields an *incidence number* $[v : e] \in \{\pm 1\}$ for each incident vertex-edge pair $v \trianglelefteq e$ so that each edge has both a positive and a negative incidence number associated to it. This is done by letting $[v : e] = +1$ if e is directed toward v and -1 if it is directed away from it.

The coboundary map $\delta : C^0(G;\mathcal{F}) \to C^1(G;\mathcal{F})$ is given on vertex stalks by $\delta(x_v) = \sum_{v \trianglelefteq e}[v : e]\mathcal{F}_{v\trianglelefteq e}(x_v)$, and extends by linearity to all of $C^0(G;\mathcal{F})$. When necessary, we will disambiguate the sheaf to which δ pertains by a superscript: $\delta^{\mathcal{F}}$. We can also define δ by its output on each edge stalk: $(\delta x)_e = \mathcal{F}_{v\trianglelefteq e}x_v - \mathcal{F}_{u\trianglelefteq e}x_u$, where e is an edge oriented from u to v.

Example. The coboundary map of the constant sheaf $\underline{\mathbb{R}}$ on G, given in matrix form, is the transpose of the (signed) incidence matrix of G.

Example. Consider again the sheaf in Figure 2. We can represent $C^0(G;\mathcal{F})$ by $\mathbb{R}^3$, and we can also represent $C^1(G;\mathcal{F})$ by $\mathbb{R}^3$. If we orient each edge so that it is negatively incident to the vertex with higher number, the coboundary map is given by the matrix

$$\delta = \left[\begin{array}{c|c|c} 1 & -1 & 0 \\ \hline 1 & 0 & -1 \\ \hline 0 & -1 & -1 \end{array}\right].$$

Note that this matrix has a trivial nullspace, which reflects the fact that the only global section of this sheaf is the zero section.

Remark. The term "cochain" comes from algebraic topology. One of its major tools, homology, deals with understanding "chains" in a topological space, or formal sums of geometrically simple subspaces. For technical reasons, it is often useful to dualize this theory, producing cohomology, which studies cochains, or linear functionals on the space of chains. The cochains of sheaves on graphs are algebraically similar to these cohomological cochains, which is what provoked this use of terminology. Indeed, sheaf cohomology was a major motivation for the development of sheaf theory.

This cohomological connection is what leads to the common terminology of the *zeroth cohomology* (or *degree-zero cohomology*) for $H^0(G;\mathcal{F}) = \ker\delta$. We also consider the cokernel of δ, which is the quotient space $C^1(G;\mathcal{F})/(\operatorname{im}\delta)$. This space is denoted $H^1(G;\mathcal{F})$, and is called the *first* (or *degree-one*) *cohomology* of the sheaf $\mathcal{F}$. As a quotient space, its direct interpretation is less natural than that of the space of global sections of $\mathcal{F}$. However, the space $\operatorname{im}\delta$ is analogous to the cut space of a graph. The cut space of a graph is the space of all real-valued functions on edges spanned by indicator functions of cut sets. A quick argument shows that this is equal to the space $\operatorname{im} B^T$, where B is the signed incidence matrix of the graph. This is the matrix B with columns indexed by edges of the graph and rows indexed by vertices of the graph, where $B_{ve} = [v : e]$. The coboundary matrix is a sort of generalized incidence matrix, so we can interpret $\operatorname{im}\delta$ as the space generated by "cuts" of the data communication structure described by $\mathcal{F}$.

The incidence matrix is most famous in spectral graph theory as a building block for the graph Laplacian: $L = BB^T$. In order to use spectral methods with sheaves, we build a Laplacian out of the coboundary map. To do this, we use the inner product on the stalks (and hence on the spaces of cochains) to construct an adjoint δ^* to δ. If we think of δ as simply a block matrix by identifying our vector spaces with the standard $\mathbb{R}^n$, this adjoint δ^* is simply the transpose δ^T.

Definition 2.5 (Sheaf Laplacian)**.** If $\mathcal{F}$ is a sheaf on a graph G with coboundary map δ, the *sheaf Laplacian* of $\mathcal{F}$ is $L_{\mathcal{F}} = \delta^*\delta$.

The Laplacian $L_{\mathcal{F}}$ is a linear map from $C^0(G;\mathcal{F})$ to itself. It is positive semidefinite by construction: $\langle x, L_{\mathcal{F}}x\rangle = \langle x, \delta^*\delta x\rangle = \langle \delta x, \delta x\rangle \geqslant 0$. Further, its kernel is equal to $H^0(G;\mathcal{F})$, since $\ker L_{\mathcal{F}} = \ker\delta^*\delta = \ker\delta$.

Proposition 2.1 (Structure of the sheaf Laplacian)**.** *If $\mathcal{F}$ is a sheaf on a graph G, the matrix of the sheaf Laplacian $L_{\mathcal{F}}$ has a symmetric block structure, with the columns divided into one partition for each vertex of G. If v_1 and v_2 are distinct and both incident to the edge e, then the entry in the (v_2, v_1) block is $-\mathcal{F}^*_{v_2 \trianglelefteq e}\mathcal{F}_{v_1 \trianglelefteq e}$, and the entry in the (v_1, v_1) block is $\sum_{v_1 \trianglelefteq e} \mathcal{F}^*_{v_1 \trianglelefteq e}\mathcal{F}_{v_1 \trianglelefteq e}$.*

Exercise. Show this is true by writing down the block matrices for δ and δ^* and calcu-

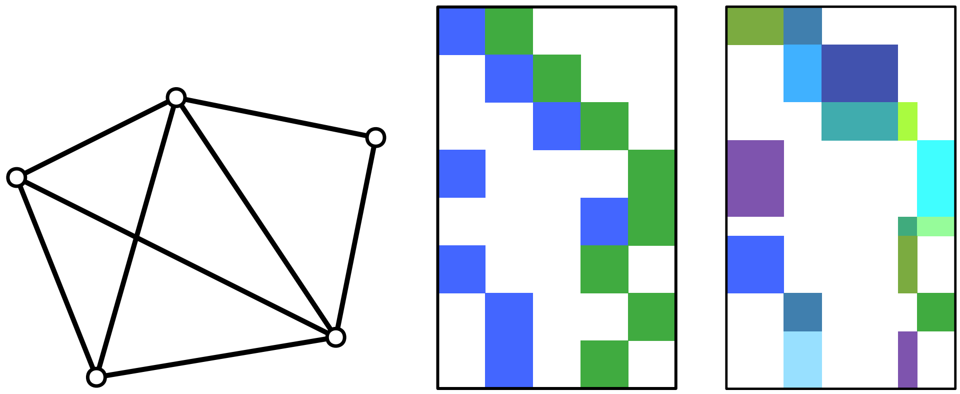

Figure 3: Illustration of a graph incidence matrix and a sheaf coboundary matrix.

lating. Use this to show that

$$(L_{\mathcal{F}}x)_v = \sum_{u,v \trianglelefteq e} \mathcal{F}^*_{v \trianglelefteq e}(\mathcal{F}_{v \trianglelefteq e}x_v - \mathcal{F}_{u \trianglelefteq e}).$$

One way to think about cellular sheaves is as a generalization of incidence matrices and Laplacians of graphs. The coboundary operator for a sheaf looks like an incidence matrix, but with arbitrary block entries instead of just ± 1. See a suggestive illustration in Figure 3. TODO: Make this figure better

3 Operations on sheaves

3.1 Sheaf morphisms

When we define mathematical objects, we usually want to know how to define relationships between them. Often the objects themselves are less important than the maps we can build between them. Linear algebra is a prime example of the supremacy of maps over objects. Vector spaces, as objects, exist so that we can study linear transformations. Linear maps are far more interesting than vector spaces themselves. We can take this to an extreme and forget all the internal structure of our objects, looking instead at the structure of the relationships between different objects of the same type. This is the point of view of category theory, which serves as a deep organizing principle for much of mathematics. A category is formally a collection of objects together with morphisms between those objects.

The same is true for sheaves. Sheaves have more structure than vector spaces, enough to be interesting on their own, but relationships between sheaves are still essential. This is why we define mappings or *morphisms* between sheaves. All we need to deal with to define maps between two sheaves on the same graph is linear algebra.

Definition 3.1 (Sheaf morphisms)**.** Let $\mathcal{F}$ and $\mathcal{G}$ be two sheaves on a graph G. A *morphism* or *map* $\varphi : \mathcal{F} \to \mathcal{G}$ consists of a linear map $\varphi_v : \mathcal{F}(v) \to \mathcal{G}(v)$ for each vertex v of G and a linear map $\varphi_e : \mathcal{F}(e) \to \mathcal{G}(e)$ for each edge e, such that these linear maps *commute* with the restriction maps. That is, for each incident vertex-edge pair (v, e), we have $\varphi_e \circ \mathcal{F}_{v \trianglelefteq e} = \mathcal{G}_{v \trianglelefteq e} \circ \varphi_v$.

This condition is easy to visualize in a diagram showing the relevant vector spaces and maps between them. The diagram *commutes* if any two paths along the arrows represent equal maps.

$$\begin{tikzcd} \mathcal{F}(v) \arrow[r, "\varphi_v"] \arrow[d, "\mathcal{F}_{ve}"] & \mathcal{G}(v) \arrow[d, "\mathcal{G}_{ve}"] \\ \mathcal{F}(e) \arrow[r, "\varphi_e"] & \mathcal{G}(e) \end{tikzcd}$$

A sheaf morphism can be represented by a large commuting diagram with many squares like the one above. The fact that the two sheaves are over the same graph means that the structure of a morphism is easy to define. The individual maps defining a sheaf morphism assemble into two maps $\varphi^0 : C^0(G;\mathcal{F}) \to C^0(G;\mathcal{G})$ and $\varphi^1 : C^1(G;\mathcal{F}) \to C^1(G;\mathcal{G})$, and the fact that the maps on stalks commute with restriction maps means that the amalgamated maps commute with the coboundary maps: $\delta^{\mathcal{G}} \circ \varphi^0 = \varphi^1 \circ \delta^{\mathcal{F}}$.

Because sheaf morphisms are made out of linear maps, they share a number of properties with them. Every linear map has a kernel, and so does every sheaf morphism. What may be surprising is that the kernel of a sheaf morphism is a sheaf. The stalks are easy to define: $(\ker \varphi)(v) = \ker \varphi_v$ and $(\ker \varphi)(e) = \ker \varphi_e$. We can restrict $\mathcal{F}_{v \trianglelefteq e}$ to $\ker \varphi_v$, getting a map $\mathcal{F}_{v \trianglelefteq e}|_{\ker \varphi_v} : \ker \varphi_e \to \mathcal{F}(e)$. If the image of this map lies in $\ker \varphi_e$, we have produced a restriction map $\mathcal{F}'_{v \trianglelefteq e} : (\ker \varphi)(v) \to (\ker \varphi)(e)$. Suppose $x \in \ker \varphi_v$. Then $\varphi_v x = 0$, so $\mathcal{G}_{v \trianglelefteq e}\varphi_v x = 0$. Because $\mathcal{G}_{v \trianglelefteq e} \circ \varphi_v = \varphi_e \circ \mathcal{F}_{v \trianglelefteq e}$, we know $\varphi_e \mathcal{F}_{v \trianglelefteq e} x = 0$. But this means that $\mathcal{F}_{v \trianglelefteq e} x \in \ker \varphi_e$, as required.

Remark. The sort of argument used above is known as a *diagram chase*. They can become quite complicated, and proofs of this kind are usually best communicated by drawing a commutative diagram, and pointing at various objects in the diagram while discussing the fate of an element as it is transferred through various maps.

A similar diagram chasing argument shows that sheaf morphisms can be composed just like linear maps. Composing the stalkwise maps of two sheaf morphisms yields another sheaf morphism. Similarly, the image of a sheaf morphism is again a sheaf.

Exercise. Show that if $\varphi : \mathcal{F} \to \mathcal{G}$ and $\psi : \mathcal{G} \to \mathcal{H}$ are sheaf morphisms, then the collection of linear maps $(\psi_\sigma \circ \varphi_\sigma)_\sigma$ defines a sheaf morphism $\mathcal{F} \to \mathcal{H}$.

Exercise. Suppose $\varphi : \mathcal{F} \to \mathcal{G}$ is a sheaf morphism. Use an argument similar to the one that shows that $\ker \varphi$ is a sheaf to show that there is a naturally defined sheaf $\operatorname{im} \varphi$ whose stalks are $(\operatorname{im} \varphi)(\sigma) = \operatorname{im} \varphi_\sigma$ and whose restriction maps are induced by those of $\mathcal{G}$.

For every sheaf $\mathcal{F}$, there is an *identity morphism* $\mathrm{id} : \mathcal{F} \to \mathcal{F}$, which restricts to the identity map on each stalk. A sheaf morphism $\varphi : \mathcal{F} \to \mathcal{G}$ is an *isomorphism* if there exists a morphism $\psi : \mathcal{G} \to \mathcal{F}$ which is an inverse of φ; that is, $\varphi \circ \psi = \mathrm{id}$ and $\psi \circ \varphi = \mathrm{id}$. One way to think of a sheaf isomorphism is as representing a "change of basis" of the sheaf. From an algebraic perspective, two isomorphic sheaves behave identically.

Exercise. Show that a sheaf morphism $\varphi : \mathcal{F} \to \mathcal{G}$ is an isomorphism if and only if each component linear map φ_σ is invertible.

However, if we care about the inner product structure of the stalks of a sheaf, there is a finer distinction to be made. Invertible matrices preserve all algebraic properties of the vector spaces they map between, but they do not in general preserve the inner product structure. Similarly, sheaf isomorphisms do not automatically preserve the inner product structure. A unitary sheaf isomorphism is one where the maps on stalks are unitary maps of inner product spaces; that is, these maps must preserve the inner product.

Just as the space of linear maps from one vector space to another forms a vector space itself (consider the space of $m \times n$ matrices as the space of linear maps $\mathbb{R}^n \to \mathbb{R}^m$, for instance), so does the space of sheaf morphisms. The sum of two sheaf morphisms, taken by adding the component maps on stalks, is again a sheaf morphism. Scalar multiples of sheaf morphisms are again sheaf morphisms as well. We denote the space of sheaf morphisms $\mathcal{F} \to \mathcal{G}$ by $\mathrm{Hom}(\mathcal{F}, \mathcal{G})$. That is, $\varphi \in \mathrm{Hom}(\mathcal{F}, \mathcal{G})$ if it is a sheaf morphism $\phi : \mathcal{F} \to \mathcal{G}$.

Exercise. Show that $\mathrm{Hom}(\mathcal{F}, \mathcal{G})$ is a vector space under the operations described above.

Example. Let $\mathcal{F}$ be a sheaf on G, and consider the set of morphisms from the constant sheaf $\underline{\mathbb{R}}$ to $\mathcal{F}$. A linear map $\varphi_v : \mathbb{R} \to \mathcal{F}(v)$ is completely determined by $\varphi_v(1)$, so we may think of a morphism $\varphi : \underline{\mathbb{R}} \to \mathcal{F}$ as a collection of elements $x_v \in \mathcal{F}(v)$ and $x_e \in \mathcal{F}(e)$. The commutativity condition for being a sheaf morphism is that $\mathcal{F}_{ve} \circ \varphi_v = \varphi_e \circ \underline{\mathbb{R}}_{v \trianglelefteq e}$. Since $\underline{\mathbb{R}}_{v \trianglelefteq e}$ is the identity, we have $\mathcal{F}_{v \trianglelefteq e} x_v = x_e$. In other words, a morphism from $\underline{\mathbb{R}}$ to $\mathcal{F}$ is the same as a choice of $x_v \in \mathcal{F}(v)$ and $x_e \in \mathcal{F}(e)$ so that

$\mathcal{F}_{v \trianglelefteq e} x_v = x_e$ whenever $v \trianglelefteq e$. This is precisely the same information that we get from a global section of $\mathcal{F}$! Therefore, we have

Proposition 3.1 (Global sections are morphisms)**.** *If $\mathcal{F}$ is a sheaf on G, there is a natural isomorphism between the vector spaces* $\mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{F})$ *and* $\Gamma(G; \mathcal{F})$*.*

Proposition 3.2 (Global sections are functorial)**.** *Let $\varphi : \mathcal{F} \to \mathcal{G}$ be a sheaf morphism. This morphism induces a natural linear map $\Gamma(\varphi) : \Gamma(G; \mathcal{F}) \to \Gamma(G; \mathcal{G})$. Further, if $\psi : \mathcal{G} \to \mathcal{A}$ is another sheaf morphism, we have $\Gamma(\psi \circ \varphi) = \Gamma(\psi) \circ \Gamma(\varphi)$.*

Proof. Note that φ induces a map $\varphi^0 : C^0(G; \mathcal{F}) \to C^0(G; \mathcal{G})$ that commutes with the coboundaries of $\mathcal{F}$ and $\mathcal{G}$. In particular, then, if $x \in \ker \delta^{\mathcal{F}}$, then $\delta^{\mathcal{G}} \varphi^0(x) = \varphi^1 \delta^{\mathcal{F}} x = 0$, so $\varphi^0(x) \in \ker \delta^{\mathcal{G}}$. This means that φ^0 induces a map $\Gamma(\varphi) : \Gamma(G; \mathcal{F}) \to \Gamma(G; \mathcal{G})$ by restriction to $\ker \delta^{\mathcal{F}}$. It is obvious that $\psi^0 \circ \varphi^0 = (\psi \circ \varphi)^0$, and since $\Gamma(\psi)$,$\Gamma(\varphi)$, and $\Gamma(\psi \circ \varphi)$ are obtained by restriction of these maps, we get $\Gamma(\psi \circ \varphi) = \Gamma(\psi) \circ \Gamma(\varphi)$. □

Remark. The operation of taking global sections takes a sheaf and produces a vector space. The fact that this operation behaves nicely with respect to sheaf morphisms and linear maps means that taking global sections is *functorial*. It translates information from the category of sheaves to the category of vector spaces in a way that respects relationships between sheaves.

Remark. The isomorphism $\mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{F}) \simeq \Gamma(G; \mathcal{F})$ is *natural* in the sense that it commutes with sheaf morphisms. That is, given a sheaf morphism $\mathcal{F} \to \mathcal{G}$, we get morphisms $\mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{F}) \to \mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{G})$ and $\Gamma(G; \mathcal{F}) \to \Gamma(G; \mathcal{G})$, and the square

$$\begin{tikzcd} \mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{F}) \arrow[r, "\simeq"] \arrow[d] & \Gamma(G; \mathcal{F}) \arrow[d] \\ \mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{G}) \arrow[r, "\simeq"] & \Gamma(G; \mathcal{G}) \end{tikzcd}$$

commutes. This means that for every purpose which we can understand categorically, we can consider $\mathrm{Hom}(\underline{R}, \mathcal{F})$ and $\Gamma(G; \mathcal{F})$ to be "the same thing."

Exercise. What is the linear map $\mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{F}) \to \mathrm{Hom}(\underline{\mathbb{R}}, \mathcal{G})$ in the diagram above?

3.2 Homological Algebra

This section is motivated by a very natural question, but takes an unavoidable detour through some more sophisticated mathematical theory. It may be advisable to skip it on a first reading.

A sheaf morphism $\varphi : \mathcal{F} \to \mathcal{G}$ induces a morphism $\Gamma\varphi : H^0(G; \mathcal{F}) \to H^0(G; \mathcal{G})$. But even if the component maps of φ are all surjective, the induced linear transformation

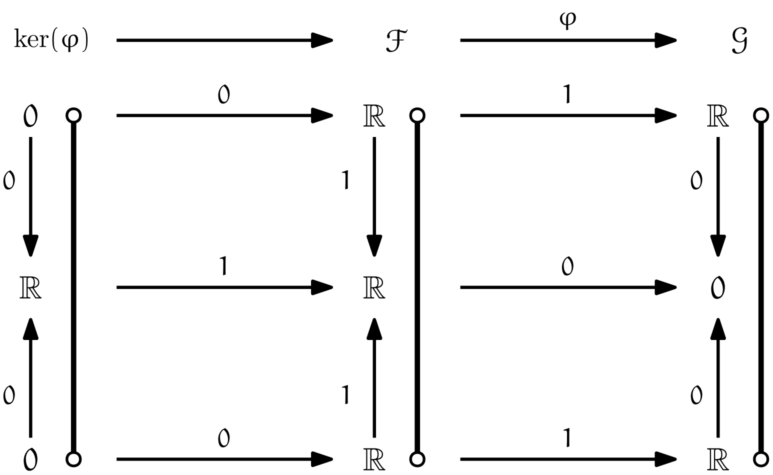

Figure 4: An exact sequence of sheaves with a non-exact sequence of global sections.

$\Gamma\varphi$ might not be. That is, $\mathcal{G}$ might have global sections that do not come from global sections of $\mathcal{F}$. This is a bit surprising: when a linear map $T : \mathbb{R}^n \to \mathbb{R}^m$ is surjective, we can always solve an equation of the form $T(x) = y$.

Example. Consider the two sheaves $\mathcal{F}$ and $\mathcal{G}$ in Figure 4. The sheaf $\mathcal{F}$ is the constant sheaf $\underline{\mathbb{R}}$, while $\mathcal{G}$ has vertex stalks $\mathbb{R}$ and edge stalk 0. There is an easily constructed sheaf morphism $\varphi : \mathcal{F} \to \mathcal{G}$ which is the identity on vertex stalks and the zero map on the edge stalk. Note that $H^0(G; \mathcal{F})$ is one-dimensional, consisting of constant functions, while $H^0(G; \mathcal{G}) \simeq C^0(G; \mathcal{G})$ is two-dimensional. Thus $\Gamma\varphi : H^0(G; \mathcal{F}) \to H^0(G; \mathcal{G})$ cannot be surjective.

Homological algebra is one approach to understanding why this breaks down: why is it not always possible to "lift" a global section over a surjective sheaf morphism $\mathcal{F} \to \mathcal{G}$?

We first observe that this strangeness only happens with surjective sheaf morphisms. An injective sheaf morphism induces an injective map on global sections. Somehow, injectivity is a local condition, while surjectivity requires us to integrate global information.

Exercise. Show that if $\varphi : \mathcal{F} \to \mathcal{G}$ is an injective morphism—that is, φ_σ is an injective linear map for each vertex or edge σ—then $\Gamma\varphi$ is an injective linear map $H^0(G; \mathcal{F}) \to H^0(G; \mathcal{G})$.

The central concept of homological algebra is the *exact sequence* of maps. This is a series of sheaf morphisms or linear transformations

$$\cdots \longrightarrow \mathcal{F}_{k-1} \xrightarrow{\varphi_{k-1}} \mathcal{F}_k \xrightarrow{\varphi_k} \mathcal{F}_{k+1} \xrightarrow{\varphi_{k+1}} \cdots$$

where the image of one morphism is equal to the kernel of the subsequent morphism. That is, $\operatorname{im} \varphi_{i-1} = \ker \varphi_i$ for all i. A special sort of such morphism is a *short exact sequence*. This has five terms, two of which are zero.

$$\underline{0} \longrightarrow \mathcal{H} \xrightarrow{i} \mathcal{F} \xrightarrow{q} \mathcal{G} \longrightarrow \underline{0}$$

Such a sequence is very special. It must be the case that $\mathcal{H} = \ker q$ and $\mathcal{G} = \mathcal{F}/\mathcal{H}$. Short exact sequences therefore encode a sheaf $\mathcal{F}$, a subsheaf $\mathcal{H}$ thereof, and the quotient $\mathcal{G} = \mathcal{F}/\mathcal{H}$.

We can take the global sections of each sheaf in a short exact sequence of sheaves, and we get a sequence

$$0 \longrightarrow H^0(G;\mathcal{H}) \xrightarrow{i} H^0(G;\mathcal{F}) \xrightarrow{q} H^0(G;\mathcal{G}) \longrightarrow 0.$$

Unfortunately, this sequence is no longer exact in general. In particular, the condition $\operatorname{im} q = \ker(0) = H^0(G;\mathcal{G})$ does not hold. This is exactly the problem we noticed before, that a surjective sheaf morphism does not necessarily turn into a surjective morphism on global sections.

Homological algebra comes to the rescue by extending this sequence in a way that makes it exact. We get what is called the *long exact sequence* on the cohomology of the sheaves in the original sequence:

$$0 \longrightarrow H^0(G;\mathcal{H}) \xrightarrow{i} H^0(G;\mathcal{F}) \xrightarrow{q} H^0(G;\mathcal{G}) \xrightarrow{d} H^1(G;\mathcal{H}) \xrightarrow{i} \cdots$$

This now says that $\operatorname{im} q = \ker d$. If we know d, we can see which sections of $\mathcal{G}$ lift to sections of $\mathcal{F}$ by checking if they lie in $\ker d$. In particular, if $H^1(G;\mathcal{H})$ is the zero vector space, $\ker d$ must in fact be all of $H^0(G;\mathcal{G})$, so every section of $\mathcal{G}$ can be lifted to a section of $\mathcal{F}$. In short:

Proposition 3.3. *Suppose $\varphi : \mathcal{F} \to \mathcal{G}$ is a surjective sheaf morphism, and $H^1(G;\ker\varphi) = 0$. Then any global section of $\mathcal{G}$ lifts to a global section of $\mathcal{F}$.*

In fact, sections lift precisely when d is the zero map. We say that the *connecting morphism* d measures the failure of q to induce a surjective map on sections. The existence of the map d relies on the quotienting performed in the construction of H^1: it is only well-defined as a map to equivalence classes.

Example. Continuing the simple example in Figure 4, the kernel of φ is the sheaf $\ker(\varphi)$ with vertex stalks 0 and edge stalk $\mathbb{R}$. Then $H^0(G;\ker(\varphi)) = 0$ and $H^1(G;\ker(\varphi)) \simeq \mathbb{R}$, which illustrates the possibility that $\Gamma\phi$ may not be surjective.

This result gives one interpretation for the first cohomology of a sheaf, in terms of what it says about relationships between different sheaves. While these concepts may seem even more hopelessly abstract than the already-abstract notions of sheaves and morphisms, the reader should rest assured that the exposition here is far simpler than nearly any other explanation of sheaf cohomology or homological algebra.

These tools are leveraged in [9, 10] to build a sheaf-based theory of signal sampling. The signals one desires to sample are sections s of a sheaf $\mathcal{F}$, and but we are only able to observe the sections $\varphi(s)$ of some sheaf $\mathcal{G}$ induced by a sheaf morphism $\varphi : \mathcal{F} \to \mathcal{G}$. When $H^1(\ker \varphi)$ is zero, sampling makes sense: every sampled signal corresponds to at least one actual signal. When $H^0(\ker \varphi)$ is zero, perfect reconstruction of signals is possible.

Homological algebra is a vast subject, with branching realms of application, and this discussion only scratches the surface of it. The interested reader might consult a book like [2], but a warning is in order: a typical prerequisite for reading and understanding a book on homological algebra is to have read and understood a book on homological algebra.

3.3 Sheaves and graph homomorphisms

There are several different notions of graph homomorphism. The definition we will use is more permissive than some. For our purposes, a graph homomorphism f from $G = (V_G, E_G)$ to $H = (V_H, E_H)$ is given by a map $f_V : V_G \to V_H$ and a map $f_E : E_G \to E_H \cup V_H$ such that if v is incident to e, then either $f_V(v)$ is incident to $f_E(e)$ or $f_V(v) = f_E(e)$. This means that an edge (or indeed an entire induced subgraph) can be collapsed onto a single vertex. A competing definition of graph homomorphism is a map on vertices that preserves the adjacency relation, which is not well suited to our purposes. The explicit treatment of edges in our notion of homomorphism is necessary for use with sheaves, since both edges and vertices carry data.

Remark. This notion of graph homomorphism is what we get when we consider graphs as *cell complexes* or *simplicial complexes*, and as such it extends to higher-dimensional structures than graphs.

If $f : G \to H$ is a graph homomorphism, we can transfer sheaves on G or H across f to get sheaves on H or G. These operations are known as the pushforward and pullback of sheaves across f.

Definition 3.2 (Pullback)**.** Let $f : G \to H$ be a graph homomorphism, and let $\mathcal{F}$ be a sheaf on H. The *pullback* of $\mathcal{F}$ over f, $f^*\mathcal{F}$, is a graph on G with stalks $(f^*\mathcal{F})(\sigma) = \mathcal{F}(f(\sigma))$ and restriction maps $(f^*\mathcal{F})_{v \trianglelefteq e} = \mathcal{F}_{f(v) \trianglelefteq f(e)}$.

We can lift cochains of $\mathcal{F}$ on H to cochains of $f^*\mathcal{F}$ on G, and in fact sections of $\mathcal{F}$ lift to sections of $f^*\mathcal{F}$.

Proposition 3.4. *There exist linear maps* $(f^*)^i : C^i(H;\mathcal{F}) \to C^i(G; f^*\mathcal{F})$ *that commute with the coboundary maps of* $\mathcal{F}$ *and* $f^*\mathcal{F}$. *That is,* $\delta^{f^*\mathcal{F}} \circ (f^*)^0 = (f^*)^1 \circ \delta^{\mathcal{F}}$.

Proof. For any vertex v of H, there is an obvious map $\mathcal{F}(v) \to f^*\mathcal{F}(v')$ whenever $f(v') = v$. We can assemble these into a map $\mathcal{F}(v) \to \bigoplus_{f(v')=v} f^*\mathcal{F}(v')$. Then combining these maps over vertices v of H, we get a map $(f^*)^0 : C^0(H;\mathcal{F}) = \bigoplus_{v\in H} \mathcal{F}(v) \to \bigoplus_{v\in H} \left(\bigoplus_{f(v')=v} f^*\mathcal{F}(v')\right) = \bigoplus_{v'\in G} f^*\mathcal{F}(v') = C^0(G; f^*\mathcal{F})$. A completely analogous argument furnishes a map $(f^*)^1 : C^1(H;\mathcal{F}) \to C^1(G; f^*\mathcal{F})$. Another, perhaps simpler, way to think of this map is as letting $((f^*)^i x)_\sigma = x_{f(\sigma)}$.

Commutativity will be satisfied if for every edge e of H, we have $(\delta^{f^*\mathcal{F}}(f^*)^0 x)_e = ((f^*)^1 \delta^{\mathcal{F}} x)_e$. The right hand side is equal to $(\delta^{\mathcal{F}} x)_{f(e)}$, while the left hand side is equal to

$$(f^*\mathcal{F})_{h(e)\trianglelefteq e}((f^*)^0 x)_{h(e)} - (f^*\mathcal{F})_{t(e)\trianglelefteq e}((f^*)^0 x)_{t(e)}$$
$$= \mathcal{F}_{f(h(e))\trianglelefteq f(e)} x_{f(h(e))} - \mathcal{F}_{f(t(e))\trianglelefteq f(e)} x_{f(t(e))} = (\delta^{\mathcal{F}} x)_{f(e)}.$$

□

If $x \in C^0(H;\mathcal{F})$, $f^*x \in C^0(G; f^*\mathcal{F})$, and if $\delta^{\mathcal{F}} x = 0$, then $\delta^{f^*\mathcal{F}} f^* x = f^* \delta^{\mathcal{F}} x = 0$. Thus, if x is a section of $\mathcal{F}$, f^*x is a section of $f^*\mathcal{F}$. Not every section of $f^*\mathcal{F}$ necessarily arises in this way, of course, but this property is a major reason for constructing $f^*\mathcal{F}$.

The pushforward of a sheaf is a little bit more complicated to define.

Definition 3.3 (Pushforward)**.** Let $f : G \to H$ be a graph homomorphism, and let $\mathcal{F}$ be a sheaf on G. The *pushforward* of $\mathcal{F}$ over f, $f_*\mathcal{F}$, is a sheaf on H with stalks $f_*\mathcal{F}(\sigma) = \Gamma(f^{-1}(\sigma);\mathcal{F})$.

The restriction maps are slightly more complicated to specify. Given an element $x \in \Gamma(f^{-1}(v);\mathcal{F})$ and an edge e incident to v, every edge $e' \in f^{-1}(e)$ is incident to a unique vertex $v(e') \in f^{-1}(v)$. The local section x has a value $x_{v(e')}$ at each of these vertices, and hence we can define $(f_*\mathcal{F})_{v\trianglelefteq e}(x) = \sum_{e'\in f^{-1}(e)} \mathcal{F}_{v(e')\trianglelefteq e'}(x_{v(e')})$.

If a graph homomorphism sends edges only to edges, the pushforward is easier to define. In this case, $f_*\mathcal{F}(\sigma) = \bigoplus_{f(\tau)=\sigma} \mathcal{F}(\tau)$ and $(f^*\mathcal{F})_{v\trianglelefteq e} = \bigoplus_{f(\sigma)=v, f(\tau)=e} \mathcal{F}_{\sigma\trianglelefteq\tau}$.

Sections of $\mathcal{F}$ over G push forward to sections of $f_*\mathcal{F}$ over H. In fact, these two sheaves have the same space of global sections.

Proposition 3.5. *There is an isomorphism* $f_* : H^0(G;\mathcal{F}) \to H^0(H; f_*\mathcal{F})$.

Proof. We will prove this in the case that f does not send any edge to a vertex, although it is true in general. This means that $f_*\mathcal{F}(v) \simeq \bigoplus_{f(u)=v} \mathcal{F}(u)$ for every vertex v of H, and similarly for edges. Let x be a section of $\mathcal{F}$ and define $(f_*x)_v = \bigoplus_{f(u)=v} x_u$. This is clearly an injective map into $C^0(H; f_*\mathcal{F})$. Note that if $v, v' \trianglelefteq e$,

$$f_*\mathcal{F}_{v \trianglelefteq e}(f_*x)_v = \bigoplus_{\substack{f(u)=v \\ f(h)=e}} \mathcal{F}_{u \trianglelefteq e} x_u$$

and

$$f_*\mathcal{F}_{v' \trianglelefteq e}(f_*x)_{v'} = \bigoplus_{\substack{f(u')=v' \\ f(h)=e}} \mathcal{F}_{u' \trianglelefteq e} x_{u'}.$$

For each edge h of G with $f(h) = e$, the corresponding terms in the sum are equal, because this is the condition for x to be a section of $\mathcal{F}$. Thus f_*x is a section of $f_*\mathcal{F}$. Conversely, this same equation shows that the condition for some y to be a section of $f_*\mathcal{F}$ is precisely the requirement that it be in the image of f_* $\square$

There are two operations which combine two sheaves on the same graph into a single sheaf.

Definition 3.4 (Direct sum)**.** If $\mathcal{F}$ and $\mathcal{G}$ are sheaves on a graph G, their *direct sum* $\mathcal{F} \oplus \mathcal{G}$ is the sheaf with stalks $(\mathcal{F} \oplus \mathcal{G})(v) = \mathcal{F}(v) \oplus \mathcal{G}(v)$, $(\mathcal{F} \oplus \mathcal{G})(e) = \mathcal{F}(e) \oplus \mathcal{G}(e)$ and restriction maps $(\mathcal{F} \oplus \mathcal{G})_{v \trianglelefteq e} = \mathcal{F}_{v \trianglelefteq e} \oplus \mathcal{G}_{v \trianglelefteq e}$.

One should think of the direct sum of two sheaves as combining two different data structures over the graph G in an independent way. Neither influences the other. A section of $\mathcal{F} \oplus \mathcal{G}$ is equivalent to a pair of a section of $\mathcal{F}$ and a section of $\mathcal{G}$.

Example. The zero sheaf $\underline{\mathbf{0}}$, which assigns the zero vector space to each vertex and edge, is an identity for the direct sum. That is, for any sheaf $\mathcal{F}$, $\mathcal{F} \simeq \underline{\mathbf{0}} \oplus \mathcal{F}$.

Definition 3.5 (Tensor product)**.** If $\mathcal{F}$ and $\mathcal{G}$ are sheaves on a graph G, their *tensor product* $\mathcal{F} \otimes \mathcal{G}$ is the sheaf with stalks $(\mathcal{F} \otimes \mathcal{G})(v) = \mathcal{F}(v) \otimes \mathcal{G}(v)$, $(\mathcal{F} \otimes \mathcal{G})(e) = \mathcal{F}(e) \otimes \mathcal{G}(e)$ and restriction maps $(\mathcal{F} \otimes \mathcal{G})_{v \trianglelefteq e} = \mathcal{F}_{v \trianglelefteq e} \otimes \mathcal{G}_{v \trianglelefteq e}$.

The tensor product is harder to interpret. Words like "twisting" and "multiplication" tend to come up when trying to describe this operation. One way to think of this is that it is possible to "add" and "multiply" sheaves, and these operations satisfy at least a distributive property: $\mathcal{F} \otimes (\mathcal{G} \oplus \mathcal{H}) \simeq (\mathcal{F} \otimes \mathcal{G}) \oplus (\mathcal{F} \otimes \mathcal{H})$.

Example. The constant sheaf $\underline{\mathbb{R}}$ is an identity of sorts for the tensor product. That is, for any sheaf $\mathcal{F}$, $\mathcal{F} \simeq \underline{\mathbb{R}} \otimes \mathcal{F}$.

Exercise. Show why $\underline{\mathbb{R}} \otimes \mathcal{F} \simeq \mathcal{F}$ for any sheaf $\mathcal{F}$.

One can naturally identify the coboundary map of $\mathcal{F} \oplus \mathcal{G}$ with the direct sum of the respective coboundary maps; the analogous statement does not hold for the coboundary map of $\mathcal{F} \otimes \mathcal{G}$. (This is because direct sums of vector spaces do not distribute over tensor products.)

Why care about these operations? One reason to like pushforwards is that they correspond naturally to a sort of abstraction or modularization process. Suppose we have a sheaf on a graph G that represents some system, made of atomic pieces. We might want to view this system "from farther away," by treating subsystems—represented by induced subgraphs of the graph—as black boxes connected together. The resulting graph G' of the zoomed-out system has one vertex for each subsystem, with the same edges between systems. This means that there is a homomorphism $G \to G'$, which sends all vertices and edges in a subsystem to the vertex representing them in G'; the pushforward over this homomorphism represents the black-boxed system.

The exact semantics of pushforwards and other sheaf operations will depend on the choices made in modeling the system. They are tools that offer formal ways to represent things we might do to systems—ways to combine or transform them.

3.4 Product graphs

There are several notions of the product of two graphs. Perhaps the most intuitive is the Cartesian product $G \square H$. This is the graph with vertex set $V(G) \times V(H)$, and an edge between (v_G, v_H) and (v'_G, v'_H) if either $v_G = v'_G$ and $v_H \sim v'_H$ or $v_G \sim v'_G$ and $v_H = v'_H$. That is, the edge set is $E(G) \times V(H) \cup V(G) \times E(H)$. (See Figure 5.) This graph carries two projection homomorphisms $\pi_G : G \square H \to G$ and $\pi_H : G \square H \to H$.

If G and H carry sheaves $\mathcal{F}$ and $\mathcal{G}$, there is a natural sheaf on $G \square H$ given by $\mathcal{F} \boxtimes \mathcal{G} = \pi_G^* \mathcal{F} \otimes \pi_H^* \mathcal{G}$. More concretely, $\mathcal{F} \boxtimes \mathcal{G}(v_G, v_H) = \mathcal{F}(v_G) \otimes \mathcal{G}(v_H)$, and the stalk associated with the edge between (v_G, v_H) and (v'_G, v_H) is $\mathcal{F}(v_G \sim v'_G) \otimes \mathcal{G}(v_H)$, and similarly for the edge between (v_G, v_H) and (v_G, v'_H). The restriction map from (v_G, v_H) to $(v_G \sim v'_G, v_H)$ is $\mathcal{F}_{v_G \trianglelefteq (v_G \sim v'_G)} \otimes \mathrm{Id}_{\mathcal{G}(v_H)}$.

If $\mathcal{F}$ and $\mathcal{G}$ are the constant sheaf $\underline{\mathbb{R}}$ on their respective graphs, then $\mathcal{F} \boxtimes \mathcal{G}$ is the constant sheaf $\underline{\mathbb{R}}$ on $G \square H$.

Remark. The notation $\mathcal{F} \boxtimes \mathcal{G}$ is adapted from the topological world, where the most important product is the Cartesian product, and this "outer product" of sheaves is frequently useful.

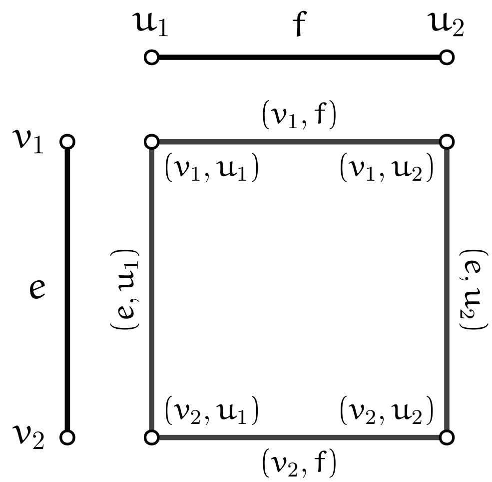

Figure 5: A simple Cartesian product graph.

4 Spectral sheaf theory

The sheaf Laplacian as defined in Section 2 is an invariant of sheaves on a labeled graph with specified stalks and orthonormal bases. Unlike the graph Laplacian, however, it is not a complete invariant. That is, there are non-isomorphic sheaves on the same graph which have the same sheaf Laplacian. A trivial reason for this is that the sheaf Laplacian does not record very much about the structure of the sheaf over the edges of the graph, but even more generally, there is too much room for redundant realizations of the same matrix structure.

Example. The two sheaves in Figure 6 both have sheaf Laplacian

$$\begin{bmatrix} 2 & -2 \\ -2 & 4 \end{bmatrix}$$

but are not isomorphic.

Remark. One important fact about sheaves is that the algebraic structure can make parts of the underlying graph "invisible." We can effectively delete a vertex or edge by assigning it the zero vector space. This is an extension of the way that one may delete edges from a graph by setting their weights to zero. Note that this then allows for "half-open" edges that are incident to only one vertex.

Although the sheaf Laplacian is not a complete invariant of sheaves, it does convey a lot of information about the sheaf. One obvious piece of data preserved by the Laplacian is the space of global sections, since $\ker L_{\mathcal{F}} = \ker \delta$. The smallest nontrivial

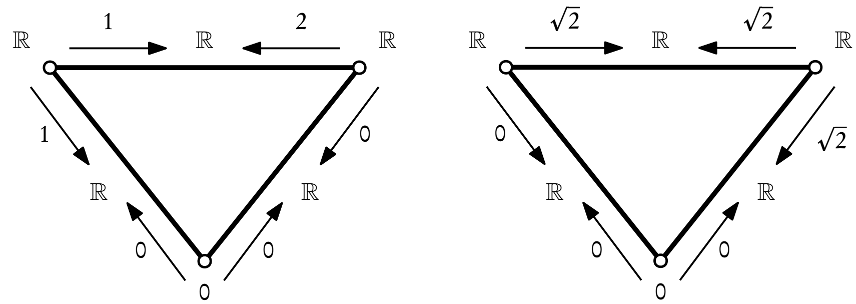

Figure 6: Two nonisomorphic sheaves with identical sheaf Laplacians

eigenvalue of a sheaf Laplacian gives us information about how close the sheaf is to having more global sections. This is a consequence of the Courant-Fischer theorem, which states that for an $n \times n$ symmetric matrix A with eigenvalues $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \lambda_n$,

$$\lambda_k = \min_{\dim Y = k} \left(\max_{y \in Y, y \neq 0} \frac{\langle y, Ay \rangle}{\langle y, y \rangle} \right).$$

Applying this to the sheaf Laplacian, with $k = \dim(H^0(G;\mathcal{F})) + 1$, we see that Y must contain $H^0(G;\mathcal{F})$ as well as the eigenvector of $L_\mathcal{F}$ corresponding to λ_k, which minimizes

$$\frac{\langle y, L_\mathcal{F} y \rangle}{\langle y, y \rangle} = \frac{\|\delta y\|^2}{\|y\|^2}$$

subject to the constraint that y is orthogonal to $H^0(G;\mathcal{F})$.

The spectrum of the sheaf Laplacian interacts in interesting ways with the sheaf operations described above. These often generalize results known for the spectra of graph Laplacians.

4.1 Basic facts about sheaf Laplacians

Proposition 4.1. *Let $\underline{V}$ be the constant sheaf on a graph G. The sheaf Laplacian of $\underline{V}$, with respect to an orthonormal basis of V, is $L_G \otimes \mathrm{id}_V$, where L_G is the graph Laplacian of G.*

A common question in spectral graph theory is how the spectrum of a graph changes as we manipulate the graph. These tend to be the easiest results to extend to spectral sheaf theory, since the analogous theorem statements are usually immediately evident.

A useful concept when studying the spectra of related matrices is the interlacing of eigenvalues.

Definition 4.1. Let A, B be $n \times n$ matrices with real spectra, and let $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n$ be the eigenvalues of A and $\mu_1 \leqslant \mu_2 \leqslant \cdots \leqslant \mu_n$ be the eigenvalues of B. We say the eigenvalues of A are $(\mathbf{p},\mathbf{q})$**-interlaced** with the eigenvalues of B if for all k, $\lambda_{k-p} \leqslant \mu_k \leqslant \lambda_{k+q}$. (We let $\lambda_k = \lambda_1$ for $k < 1$ and $\lambda_k = \lambda_n$ for $k > n$.)

We now show that the eigenvalues of sheaf Laplacians are interlaced after deleting a subgraph.

Theorem 4.2 (Eigenvalue interlacing for the sheaf Laplacian). *Let $\mathcal{F}$ be a sheaf on a graph G, C a collection of edges of G, and let $H = G \setminus C$ denote G with the edges in C removed. Let $\mathcal{G}$ be the sheaf with the same vertex stalks as $\mathcal{F}$ but with all edge stalks over edges not in C set to zero. Then the eigenvalues of L_H are $(t,0)$-interlaced with the eigenvalues of L_G, where $t = \operatorname{codim} H^0(G;\mathcal{G}) = \dim C^0(G;\mathcal{F}) - \dim H^0(G;\mathcal{G})$.*

Proof. This is a standard sort of argument in spectral graph theory. We use Rayleigh quotients to bound eigenvalues in terms of the quadratic form defined by the Laplacian. Notice that $L_H = L_G - L_C$, where L_C is the Laplacian of $\mathcal{G}$. By the Courant-Fischer theorem, we have

$$\begin{aligned}
\mu_k &= \min_{\dim Y = k} \left(\max_{y \in Y, y \neq 0} \frac{\langle y, L_G y\rangle - \langle y, L_C y\rangle}{\langle y, y\rangle} \right) \\
&\geqslant \min_{\dim Y = k} \left(\max_{y \in Y \cap H^0(G;\mathcal{G}), y \neq 0} \frac{\langle y, L_G y\rangle}{\langle y, y\rangle} \right) \\
&\geqslant \min_{\dim Y = k-t} \left(\max_{y \in Y, y \neq 0} \frac{\langle y, L_G y\rangle}{\langle y, y\rangle} \right) = \lambda_{k-t}
\end{aligned}$$

and

$$\begin{aligned}
\lambda_k &= \min_{\dim Y = k} \left(\max_{y \in Y, y \neq 0} \frac{\langle y, L_G y\rangle}{\langle y, y\rangle} \right) \\
&\geqslant \min_{\dim Y = k} \left(\max_{y \in Y, y \neq 0} \frac{\langle y, L_G y\rangle - \langle y, L_C y\rangle}{\langle y, y\rangle} \right) = \mu_k.
\end{aligned}$$

□

Definition 4.2 (Covering map). A graph morphism $f : G \to H$ is called a *covering map* if for every $v \in H$, the number of vertices u with $f(u) = v$ is constant, there are no edges that map to vertices, and the number of edges incident to $u \in G$ is the same as the number of edges incident to $f(u) \in H$. (See Figure 7.)

Proposition 4.3. *Let $f : G \to H$ be a covering map of graphs. If $\mathcal{F}$ is a sheaf on H, the spectrum of $L_{f^*\mathcal{F}}$ contains the spectrum of $L_{\mathcal{F}}$.*

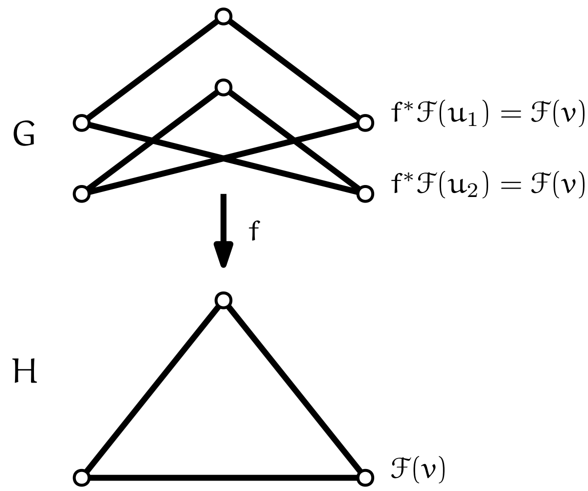

Figure 7: A covering map of graphs and the sheaf pullback

Proof. Consider the map $f^* : C^i(H;\mathcal{F}) \to C^i(G; f^*\mathcal{F})$ given by $(f^*x)_v = x_{f(v)}$. If x is an eigenvector of $L_\mathcal{F}$ with eigenvalue λ, we have

$$(L_{f^*\mathcal{F}} f^* x)_v = \sum_{v \trianglelefteq e, w \trianglelefteq e} ((f^*\mathcal{F})^*_{ve}(f^*\mathcal{F})_{ve}(f^*x)_v - (f^*\mathcal{F})^*_{ve}(f^*\mathcal{F})_{we} f^* x_w)$$

$$= \sum_{f(v) \trianglelefteq e, f(w) \trianglelefteq e} \mathcal{F}^*_{f(v)e}\mathcal{F}_{f(v)e}x_{f(v)} - \mathcal{F}^*_{f(v)e}\mathcal{F}_{f(w)e}x_{f(w)} = (L_\mathcal{F} x)_{f(v)} = \lambda x_{f(v)} = \lambda (f^*x)_v.$$

Since every eigenvector of $L_\mathcal{F}$ produces a corresponding eigenvector of $L_{f^*\mathcal{F}}$ with the same eigenvalue, the spectrum of $L_{f^*\mathcal{F}}$ contains the spectrum of $L_\mathcal{F}$. □

Note that this strengthens the result that sections of $\mathcal{F}$ lift to sections of $f^*\mathcal{F}$ to state that when f is a covering map, eigenvectors of $L_\mathcal{F}$ lift to eigenvectors of $L_{f^*\mathcal{F}}$.

Cartesian products of sheaves have spectra completely determined by their factors. This is a generalization of a result for Cartesian products of graphs.

Proposition 4.4. *Let $\mathcal{F}$ be a sheaf on a graph G and $\mathcal{G}$ a sheaf on a graph H. The Laplacian spectrum of the sheaf $\mathcal{F} \boxtimes \mathcal{G}$ on $G \square H$ consists of all sums $\lambda + \mu$, where λ is an eigenvalue of $L_\mathcal{F}$ and μ is an eigenvalue of $L_\mathcal{G}$.*

Proof. Let's look at $C^0(G\square H; \mathcal{F} \boxtimes \mathcal{G})$ and $C^1(G\square H; \mathcal{F} \boxtimes \mathcal{G})$. Note that

$$C^0(G\square H; \mathcal{F} \boxtimes \mathcal{G}) = \bigoplus_{\substack{u \in V(G) \\ v \in V(H)}} \mathcal{F}(u) \otimes \mathcal{G}(v) \simeq \left(\bigoplus_{u \in V(G)} \mathcal{F}(u) \right) \otimes \left(\bigoplus_{u \in V(H)} \mathcal{G}(v) \right).$$

This is simply $C^0(G;\mathcal{F}) \otimes C^0(H;\mathcal{G})$. Similarly,

$$
\begin{aligned}
C^1(G\square H;\mathcal{F}\boxtimes\mathcal{G}) &\simeq \left(\bigoplus_{\substack{u\in V(G)\\ f\in E(H)}} \mathcal{F}(u)\otimes\mathcal{G}(f)\right) \oplus \left(\bigoplus_{\substack{e\in E(G)\\ v\in V(H)}} \mathcal{F}(e)\otimes\mathcal{G}(v)\right)\\
&\simeq \left(\bigoplus_{u\in V(G)} \mathcal{F}(u)\right)\otimes\left(\bigoplus_{f\in E(H)} \mathcal{G}(f)\right) \oplus \left(\bigoplus_{e\in e(G)} \mathcal{F}(e)\right)\otimes\left(\bigoplus_{v\in V(H)} \mathcal{G}(v)\right)\\
&= \left(C^0(G;\mathcal{F})\otimes C^1(H;\mathcal{G})\right) \oplus \left(C^1(G;\mathcal{F})\otimes C^0(H;\mathcal{G})\right).
\end{aligned}
$$

With a little work, best done in the privacy of one's own home, it can be shown that under this identification, the coboundary matrix of $\mathcal{F}\boxtimes\mathcal{G}$ is

$$
\delta_{\mathcal{F}\boxtimes\mathcal{G}} = \begin{bmatrix} \mathrm{id}_{C^0(G;\mathcal{F})}\otimes\delta_{\mathcal{G}} \\ \delta_{\mathcal{F}}\otimes\mathrm{id}_{C^0(H;\mathcal{G})} \end{bmatrix},
$$

from which we can conclude that $L_{\mathcal{F}\boxtimes\mathcal{G}} = \mathrm{id}_{C^0(G;\mathcal{F})}\otimes L_{\mathcal{G}} + L_{\mathcal{F}}\otimes\mathrm{id}_{C^0(H;\mathcal{G})}$. This means that if x is an eigenvector of $L_{\mathcal{F}}$ with eigenvalue λ and y is an eigenvector of $L_{\mathcal{G}}$ with eigenvalue μ, then $L_{\mathcal{F}\boxtimes\mathcal{G}}x\otimes y = \mathrm{id}_{C^0(G;\mathcal{F})}\otimes L_{\mathcal{G}}x\otimes y + L_{\mathcal{F}}\otimes\mathrm{id}_{C^0(H;\mathcal{G})}x\otimes y = \lambda x\otimes y + \mu x\otimes y = (\lambda+\mu)(x\otimes y)$. $\square$

A number of other interesting results related to sheaves and their Laplacians may be found in [7]. Among these are results on:

- spectra of pushforward sheaves
- a version of "effective resistance" for sheaves, treating them as a sort of generalization of an electrical network.
- spectral sparsification of sheaves, where a number of edges may be removed from a graph while keeping the sheaf Laplacian spectrum approximately the same.
- harmonic functions on sheaves, which are the nearest one can get to a section when certain values have been specified.

TODO: reproduce some of these results here

5 Directions for applications

It's finally time to talk about what this all might be good for. I will here sketch a few points of contact between sheaf theory and problems and phenomena we might see in the real world. These are not fully developed applications. No one (as far as I know) has built anything using them yet. But I think they are sufficiently varied and convincing to show that sheaves are a natural way to represent a number of different real-world situations.

5.1 Sensors

Here's a simple example of a sheaf that suggests itself in certain applications. Consider a set of sensors observing subsets of some domain, with fields of view that overlap. Say that sensor i can see some subset $U_i \subseteq X$. We build a graph G using the data of these subsets: add one vertex for each sensor, and an edge between sensor i and sensor j when $U_i \cap U_j \neq \varnothing$. We assume that what each sensor observes is a function $U_i \to \mathbb{R}^n$. Clearly, if sensor i and sensor j are looking at the same thing, the functions they see should agree on the intersection of U_i and U_j. This condition translates into a sheaf on G. For the vertex corresponding to sensor i, the stalk is $\mathcal{F}(i) = \{f : U_i \to \mathbb{R}^n\}$, the vector space of functions on U_i with values in $\mathbb{R}^n$. The stalk over the edge between i and j is $\mathcal{F}(i \sim j) = \{f : U_i \cap U_j \to \mathbb{R}^n\}$, the space of functions on the overlap of the two sensor domains. The restriction maps are then just restriction of functions to the common domain.

Remark. One might call this the *sheaf of functions subordinate to a cover*. It is one of the canonical examples of a sheaf, and this is where the term "restriction maps" originates—they abstract the notion of restriction of functions to a smaller domain.

Under the assumption that the sensors see the whole domain X, the global sections of this sheaf are precisely the functions $X \to \mathbb{R}^n$. Why is this a useful result? It allows us to check the conditions of consistency locally, without having to construct a global picture explicitly. If sensors can communicate according to the connections in the graph G, they can check consistency completely locally.

5.2 Diffusion

The existence of a Laplacian immediately suggests consideration of its associated differential equation. This is the Laplacian flow, or diffusion

$$\dot{x} = -L_{\mathcal{F}} x.$$

Since $L_{\mathcal{F}}$ is positive semidefinite with kernel equal to $H^0(G; \mathcal{F})$, the trajectories of this dynamical system converge to global sections of $\mathcal{F}$. With a little more work (write out the closed form solution in terms of the eigendecomposition of $L_{\mathcal{F}}$) it's possible to show that the limiting section is in fact the orthogonal projection of the initial condition onto $H^0(G;\mathcal{F})$. This could be combined with the sensing sheaf above to give a simple distributed algorithm for ensuring consistency of sensor observations: just run the diffusion algorithm until convergence.

5.2.1 Opinion dynamics

Graph Laplacian-based diffusions are often used as a building block for the construction or study of network dynamics. Sheaf Laplacians can add richness to these models while remaining analytically tractable. Opinion dynamics in social networks offer an illustrative example. Perhaps the simplest model of opinion dynamics posits a social network described by a graph G, where each agent has a one-dimensional space of opinions. The dynamics are described by the graph Laplacian flow on G, and eventually converge to consensus.

Replacing G with a sheaf on G allows us to model various more interesting aspects of the system:

- We can add extra dimensions to the opinion space by increasing the dimension of the vertex and edge stalks
- We can implement links that force opinions apart for certain agents by changing the sign of one of the restriction maps on an edge
- We can model the difference between an agent's privately-held opinion and the one communicated to others by changing edge stalks and restriction maps.

While these additions are still limited by the fact that the overall dynamics are linear, they introduce interesting behavior in such a system, and can all be analyzed within the sheaf framework. Further, the linear dynamics can be used as a building block for more sophisticated models. Some ideas in this direction are explored in [8].

5.3 Distributed consensus

Diffusion on graphs is also used to specify consensus dynamics on networks specifically designed for the purpose. Suppose we have a collection of agents connected according to the edges in a graph, and each has some internal state in some vector space V. We want them to communicate over the edges of the graph in order to come to an agreement on a value. We can represent this condition as saying we wish the agents to reach a global section of the constant sheaf $\underline{V}$. This can be achieved by having them follow the Laplacian flow associated to the sheaf.

This requires each agent to communicate its full state to all of its neighbors. This is perhaps an onerous requirement, but is not necessary. Sheaf theory can help us reduce the bandwidth requirements.

Definition 5.1 (Approximation to a sheaf)**.** Let G be a graph, and let $\mathcal{G}$ be a sheaf on G. We say that a sheaf $\mathcal{F}$ on G is an *approximation to* $\mathcal{G}$ if there exists a morphism $a : \mathcal{F} \to \mathcal{G}$ which is an isomorphism on vertex stalks, and which induces an isomorphism $\Gamma(G; \mathcal{G}) \to \Gamma(G; \mathcal{F})$.

If V is a vector space, we denote the constant sheaf with stalk V by $\underline{V}$, and say that $\mathcal{F}$ is an *approximation to the constant sheaf* if $\mathcal{F}$ is an approximation to $\underline{V}$.

Proposition 5.1. *If $\mathcal{F}$ is an approximation to $\underline{V}$, then it is isomorphic to a sheaf with vertex stalks V where the restriction maps $\mathcal{F}_{v \trianglelefteq e} : V \to \mathcal{F}(e)$ and $\mathcal{F}_{v' \trianglelefteq e} : V \to \mathcal{F}(e)$ are equal.*

Proof. Note that because $a : \underline{V} \to \mathcal{F}$ is an isomorphism on vertex stalks, $\mathcal{F}$ is clearly isomorphic to a sheaf with vertex stalks V. For every edge e we have the diagram

$$\begin{tikzcd}
V \arrow[r, "1"] \arrow[d, "1"] & V \arrow[d, "\mathcal{F}_{v \trianglelefteq e}"] \\
V \arrow[r, "a_e"] & \mathcal{F}(e) \\
V \arrow[r, "1"] \arrow[u, "1"] & V \arrow[u, "\mathcal{F}_{v' \trianglelefteq e}"]
\end{tikzcd},$$

and the only way it can commute is if $\mathcal{F}_{v \trianglelefteq e} = \mathcal{F}_{v' \trianglelefteq e} = a_e$.

□

The proof of this proposition shows that specifying an approximation to $\underline{V}$ is the same as specifying a morphism $a_e : V \to W_e$ for each edge e of G. Further, in order to produce an approximation to $\underline{V}$, the a_e must assemble to a map $a : C^1(G; \underline{V}) \to C^1(G; \mathcal{F}) = \bigoplus_{e \in E} W_e$ such that $\ker(a \circ \delta_{\underline{V}}) = \ker \delta_{\underline{V}}$. This holds if $\ker a$ is contained in a complement to $\operatorname{im} \delta$; equivalently, the projection map $\pi : C^1(G; \underline{V}) \to H^1(G; \underline{V})$ must be an isomorphism when restricted to $\ker a$.

This inspires a way to construct an approximation to the constant sheaf. Choose a subspace K_e of V for each edge e of G and define a_e to be the projection map $V \to V/K_e$. If $\bigoplus_{e \in E} K_e$ has the same dimension in $H^1(G; \underline{V})$ as in $C^1(G; \underline{V})$, then $a = \bigoplus_{e \in E} a_e$ defines the edge maps giving an approximation to $\underline{V}$. (The vertex maps may be taken to be the identity.)

A spectrally good approximation to the constant sheaf (satisfying a few other conditions) is analogous to an expander graph. It is possible that expander sheaves would allow for implementations of faster consensus algorithms for high-dimensional data on graphs by reducing the amount of communication needed to ensure convergence. Some exploration of these ideas in a language closer to that of traditional spectral graph theory is in [4]. Constructing such sheaf approximations and expanders is a surprisingly deep problem.

5.4 Distributed optimization

Graph consensus dynamics are often used as a component in more complex algorithms, like distributed optimization. The standard formulation casts an optimization problem with local objectives f_v corresponding to vertices with the goal of minimizing $\sum_v f_v(x)$. This is converted into a more naturally distributed problem by duplicating state variables: minimize $\sum_v f_v(x_v)$ subject to the constraint that $x_v = x_u$ for all vertices u and v.

Those who have developed a sense for the applicability of cellular sheaves[1] will immediately interpret this condition as saying that the collection of the x_v must form a section of the constant sheaf. Various approaches can then be used to design a distributed algorithm to solve the optimization problem. Typically these will combine a local optimization process, like gradient descent, with a consensus process, like local averaging. The natural question, then, is whether we can extend these algorithms to solve distributed optimization problems over other sheaves, and whether there might be any use for this.

The answer to both questions is yes. Sheaf diffusion is the key component. The sheaf Laplacian can serve as a drop-in replacement for the graph Laplacian in algorithms designed for standard distributed optimization. This straightforward generalization is discussed in [5].

When would you want to optimize over the sections of a non-constant sheaf? One situation where this might be useful comes from the sensor network example that began this section. A network of sensors observing different regions of a domain (or different aspects of some abstract domain) could use a properly constructed optimization problem to cooperatively improve their local estimates of the observed data.

5.5 Learning sheaves

From a network science perspective, it may be interesting to try to find a sheaf for which a given collection of signals is close to being sections. That is, we are given a number of vectors $x_i \in C^0(G; \mathcal{F})$, and wish to find both a graph G and a sheaf $\mathcal{F}$ on G so that

[1] Currently a somewhat rarefied group! The purpose of these notes is to expand it.

$\sum_i x_i^T L_{\mathcal{F}} x_i$ is small. It turns out that the space of sheaf Laplacians is a convex cone, so we can cast this as a convex optimization problem. Seeking to minimize $\sum_i x_i^T L_{\mathcal{F}} x_i$ alone leads to a minimum at $L_{\mathcal{F}} = 0$, so we need to add regularization terms. Details can be found in [6].

What does the learned $L_{\mathcal{F}}$ tell us? We can extract an underlying graph structure from the sparsity pattern of the Laplacian. These are connections implied by the data. Each edge further includes a description of the tendencies of the relationships in the data: an edge "pushes" the data on its incident vertices to lie in some algebraic relationship.

6 Where to go from here

6.1 Beyond graphs

In algebraic topology, graphs have higher-dimensional generalizations known as simplicial complexes and cell complexes. Graphs are one-dimensional cell complexes, and simple graphs are one-dimensional simplicial complexes. Sheaves can be defined on cell complexes and simplicial complexes by adding stalks for higher-dimensional cells and restriction maps for each incidence relation between cells. These give us a way to encode higher-order consistency conditions, but these do not appear in quite the way one might expect. The coboundary map extends to a sequence of coboundary maps between cochain spaces of adjacent cell degrees, but the significance of the kernel of higher coboundary maps is less obvious.

Even more generally, it is possible to define a sheaf over a partially-ordered set; this is the approach taken by Michael Robinson in his work on sheaves and data analytics. This case subsumes the cell complex case, since the face relations of a cell complex form a poset.

There are versions of sheaves for general topological spaces and even more exotic structures like sites, but these have the downside in applications that they are not typically amenable to computation. When computations are possible, they typically reduce to operations equivalent to those taken with cellular sheaves.

6.2 Beyond vector spaces

There is nothing in the definition of a sheaf that requires the stalks actually be vector spaces. One may construct a sheaf *valued in* any number of other categories. For instance, the stalks might be simply sets, and the restriction maps just functions. Or they might be groups, and the restriction maps group homomorphisms. The difficulty here is that the tools we have developed for sheaves of vector spaces do not have immediate analogues when the stalks are not vector spaces. One situation that allows some of the machinery developed for vector spaces is the case of sheaves of semimodules over a semiring. (A semiring is the analogue of a ring without the requirement that every element have an additive inverse.) These give a natural way to model nonnegativity constraints and directionality.

6.3 Turning things around

Some relationships we would like to model are more naturally expressed with maps going the opposite way. Rather than restriction maps going from vertex stalks to edge stalks, we want to consider extension maps going from edge stalks to vertex stalks. In category theory, it is common to take the *dual* of a construction by reversing the direction of all relevant morphisms. When we do this to a sheaf, we get what is called a *cosheaf*.

Cosheaves are somewhat less intuitive than sheaves. Rather than global sections, they have *cosections*, which are assignments to vertex stalks modulo an equivalence relation given by the edges. Similarly, rather than *co*homology, they have homology, and the cosections of a cosheaf are precisely the elements of the degree zero homology space H_0. This homology is computed by constructing a boundary map $\partial : C_1 \to C_0$.

We can think of the construction of a sheaf Laplacian as pairing a sheaf with its dual cosheaf; the boundary of the cosheaf is the adjoint of the coboundary of the sheaf, i.e., $\partial = \delta^*$. In some situations there is a natural applied interpretation of both a sheaf and its dual cosheaf.

6.4 Further reading

Most other resources about sheaves have a heavier topological and algebraic emphasis, and more prerequisites. The goal of these notes is to prepare the reader interested in applications to tackle the more sophisticated treatments.

Elementary Applied Topology, by Robert Ghrist [3]. This book serves as an introduction to algebraic topology and its applications to a wide variety of problems. The penultimate chapter discusses sheaves, but is not accessible without a good deal of the previous material in the book.

Topological Signal Processing, by Michael Robinson [11]. This book is written from a more explicit engineering point of view, with examples of specific applications.

Not everything in here is relevant to the study of sheaves on graphs, though, and it focuses less on networks. Robinson also has slides and video from a seminar on sheaves and data he presented for DARPA: **Tutorial on Sheaves in Data Analytics** [12].

Sheaves, Cosheaves, and Applications, by Justin Curry [1]. Justin's thesis is a perennial source of new insights, but is written from the most abstract perspective of any of these resources. It is not likely to be a fruitful read for anyone not comfortable with the material in Elementary Applied Topology.

Toward a Spectral Theory of Cellular Sheaves, by Jakob Hansen and Robert Ghrist [7]. This is an expository paper intended for an audience in applied topology. It contains more general versions of most of the results discussed in this introduction.

References

[1] J. Curry. *Sheaves, Cosheaves, and Applications*. PhD thesis, University of Pennsylvania, 2014.

[2] S. I. Gelfand and Y. I. Manin. *Methods of Homological Algebra*. Springer Monographs in Mathematics. Springer-Verlag, Berlin, second edition edition, 2003.

[3] R. Ghrist. *Elementary Applied Topology*. CreateSpace, https://www.math.upenn.edu/ ghrist/notes.html, 2014.

[4] J. Hansen. Expansion in Matrix-Weighted Graphs. *arXiv:2009.12008 [math]*, Sept. 2020.

[5] J. Hansen and R. Ghrist. Distributed optimization with sheaf homological constraints. In *57th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, 2019.

[6] J. Hansen and R. Ghrist. Learning sheaf Laplacians from smooth signals. In *Proceedings of ICASSP*, 2019.

[7] J. Hansen and R. Ghrist. Toward a spectral theory of cellular sheaves. *Journal of Applied and Computational Topology*, 3(4):315–358, Dec. 2019.

[8] J. Hansen and R. Ghrist. Opinion dynamics on discourse sheaves. *arXiv:2005.12798 [math]*, May 2020.

[9] M. Robinson. The Nyquist theorem for cellular sheaves. In *Proceedings of the 10th International Conference on Sampling Theory and Applications*, 2013.

[10] M. Robinson. A sheaf-theoretic perspective on sampling. *arXiv:1405.0324 [math]*, May 2014.

[11] M. Robinson. *Topological Signal Processing*. Mathematical Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[12] M. Robinson. Tutorial on Sheaves in Data Analytics. http://www.drmichaelrobinson.net/sheaftutorial/, 2015.