9

THE LAPLACIAN OPERATOR

A landmark geometry paper in 1966 by Mark Kac posed an intriguing question that continues to inspire theoretical and applied research in geometry CITE: "Can you hear the shape of a drum?"

The paper described a straightforward thought experiment also considered by past mathematicians including Bers CITE and Weyl CITE. A mathematician (and/or a musician) taps the surface of a drum in a dark room, and the audience hears the sound it makes. In particular, the audience hears the *spectrum* of the drum, i.e., the frequencies at which its surface vibrates in response to the initial impulse. The question is whether the audience—having heard but not seen the drum—can reconstruct its shape.

Audio and shape are intricately correlated. At the simplest level, when a cellist shortens the length of the instrument's strings by placing down a finger, the pitch rises; this action links the one-dimensional geometry of the string to the sound it makes when it vibrates. More broadly, it is remarkably difficult to encounter two objects that make the same sound when they are dropped on the ground.[1]

While Kac's question quickly was answered negatively—that is, there exist *isospectral* drums (see §REF)—it turns out that the vibration frequencies and patterns of a shape tell a great deal about its geometry. Indeed, while the original counterexample involved 16-dimensional tori CITE, it took nearly 30 years to identify two-dimensional examples of isospectral drums CITE. Thanks to the distinguishing properties of shape that can be gathered by computing vibration frequencies, spectral methods for shape analysis form the core of state-of-the-art algorithms that analyze the similarities and differences between geometric figures, and they inform our understanding of geometric domains as diverse as graphs, surfaces, and manifolds.

In this chapter, we aim to make more precise the notion of "vibration modes" appearing in the spectral problem through use of the Laplacian operator, which we introduce on progressively more complicated domains. From there, we show how the Laplacian can be discretized for computational pipelines and applied to a wide variety of geometric data processing problems. This chapter is and should be lengthy: Laplacian-based algorithms are common, effective tools for shape analysis but also draw upon mathematical ideas usually not encountered in undergraduate computer science.

JS: todo: triple check that compactness is mentioned whenever we talk about eigenfunctions

9.1 ONE-DIMENSIONAL WAVE EQUATION

Consider the physical system illustrated in Figure REF. Here, we have a set of $n+1$ particles each of mass m on the interval of length ℓ, attached by springs with constant k and rest length zero. The particles are spaced distance $h = \ell/n$ apart, and particles 0 and n are fixed in place. Suppose at time $t = 0$ each particle i is displaced horizontally by signed distance u_i from its rest position $\ell i/n$. Our goal is to track their motion over time.

Suppose $i \in \{1, \ldots, n-1\}$. Then, applying Newton's second law $F = ma$, we know

$$m\frac{d^2u_i}{dt^2} = k(u_{i+1} - u_i) + k(u_{i-1} - u_i) = k(u_{i+1} + u_{i-1} - 2u_i). \tag{9.1}$$

We can attempt to predict a "continuum limit" of behavior for this system as $n \to \infty$ by noticing that the right-hand side of this expression approximates a second derivative. Suppose $u(x,t)$ gives

[1] This observation comes with extensive empirical evidence gathered by the author.

the displacement of the particle that started at position x when $t = 0$. Then, as $n \to \infty$ the behavior of this many-particle system resembles the solution of the partial differential equation (PDE)

$$\frac{\partial^2 u(x,t)}{\partial t^2} = c^2 \frac{\partial^2 u(x,t)}{\partial x^2}. \tag{9.2}$$

Here, we have lazily absorbed all physical constants into a single (positive) value c^2. This PDE is known as the *wave equation*, roughly predicting the behavior of a compression as it propagates its way along a springy one-dimensional object like a Slinky. Figure REF illustrates some solutions of this equation in one dimension.

Figure REF shows a different illustration of the wave equation, where $u(x,t)$ is plotted as a height above position $x \in [0,1]$. A Taylor series argument shows that this interpretation agrees with (9.2) to first order; see REF for discussion. This roughly inspires the "Can you hear the shape of a drum?" question, although the reality is that realistically simulating the audio signal from even an idealized drum surface would involve challenging nonlinear PDE.

Momentarily returning to the finite-dimensional problem (9.1), suppose we stack the unknown u values into a vector-valued function $\mathbf{u}(t) \in \mathbb{R}^{n-1}$. Then, we have $\mathbf{u}''(t) = -c^2 L\mathbf{u}(t)$ where

$$L = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 \end{pmatrix}. \tag{9.3}$$

One can check that L is positive semidefinite (see exercise 9.1.) and hence has a full set of orthonormal eigenvectors $\{\boldsymbol{\phi}_k\}_k$ with corresponding eigenvalues $\{\lambda_k\}_k$. If we decompose $\mathbf{u}(t) = \sum_k u^k(t)\boldsymbol{\phi}_k$, then we find:

$$\sum_k (u^k)''(t)\boldsymbol{\phi}_k = \mathbf{u}''(t) = -c^2 L\mathbf{u}(t) = -c^2 \sum_k \lambda_k u^k(t)\boldsymbol{\phi}_k.$$

Assuming distinct eigenvalues, we have $(u^k)''(t) = -c^2\lambda_k u^k(t)$. A standard result in ordinary differential equations (ODE) shows as a consequence that

$$u^k(t) = a^k \sin\left(c\sqrt{\lambda_k}t\right) + b^k \cos\left(c\sqrt{\lambda_k}t\right). \tag{9.4}$$

From this perspective, the values $\sqrt{\lambda_k}$ specify the *frequency* components of the vibration.

We can carry out a completely analogous set of computations for the continuum PDE (9.2). For simplicity, we will assume $c = 1$ in our calculations; simple scaling arguments can take care of the general case. Define a linear operator $\mathcal{L}$ given by

$$\mathcal{L}[\cdot] : u \mapsto -\frac{\partial^2 u}{\partial x^2}.$$

This operator takes functions u to other functions $-u''$. In some sense, $\mathcal{L}$ can be considered a *positive semidefinite* operator under the inner product $\langle u, v\rangle := \int_0^\ell u(x)v(x)\,dx$ for functions $u, v : [0,\ell] \to \mathbb{R}$ with $u(0) = v(0) = u(\ell) = v(\ell) = 0$:

$$\begin{aligned} \langle u, \mathcal{L}[u]\rangle &= \int_0^\ell u(x)\cdot -u''(x)\,dx \text{ by definition of } \mathcal{L} \text{ and } \langle\cdot,\cdot\rangle \\ &= \int_0^\ell u'(x)^2\,dx \text{ by integration by parts} \\ &\geq 0. \end{aligned}$$

We can also think of $\mathcal{L}$ as symmetric:

$$\begin{aligned}\langle u, \mathcal{L}[v]\rangle &= \int_0^\ell u(x)\cdot -v''(x)\,dx \text{ by definition of } \mathcal{L} \text{ and } \langle\cdot,\cdot\rangle \\ &= \int_0^\ell u'(x)v'(x)\,dx \text{ by integration by parts} \\ &= \int_0^\ell u''(x)v(x)\,dx \text{ by integration by parts a second time} \\ &= \langle v, \mathcal{L}[u]\rangle.\end{aligned}$$

Hence, taking analogy to positive semidefinite, symmetric matrices, it comes as no surprise that $\mathcal{L}$ admits a sequence of orthonormal eigenfunctions of the form

$$\phi_k(x) = \sqrt{\frac{2}{\ell}}\sin\left(\frac{\pi k x}{\ell}\right),$$

with eigenvalues

$$\lambda_k = \left(\frac{\pi k}{\ell}\right)^2. \tag{9.5}$$

The functions ϕ_k actually span $L_2([0,\ell])$ with the prescribed (zero Dirichlet) boundary conditions, just as the eigenfunctions of the matrix L above span $\mathbb{R}^n$. While this particular sequence is easily recognized as a Fourier series, this property can also be motivated using the Hilbert–Schmidt Theorem, an infinite-dimensional analog of the spectral theorem (Theorem 2.1) showing that the eigenfunctions of a self-adjoint operator like $\mathcal{L}$ span the corresponding linear space under certain assumptions. Further deepening the connection between L and $\mathcal{L}$, readers familiar with discrete-time signal processing may recognize the eigenvectors of the matrix L as the discrete Fourier transform (DFT) basis.

From here, the solution (9.4) to the wave equation holds without change from the finite-dimensional case. More importantly for our discussion, the eigenvalues in (9.5) imply a key property: You *can* hear the shape of a line segment! Given a list of the resonant frequencies λ_k of a vibrating one-dimensional object, we can find the length ℓ of the segment. Even a single eigenvalue will suffice in this case: we can recover $\ell = \pi/\sqrt{\lambda_1}$.

Stepping back, we derived a spatial derivative operator $\mathcal{L}[\cdot]$ so that the wave equation can be written $u_{tt} = -\mathcal{L}[u]$. Then, we found that the eigenfunctions of $\mathcal{L}$ provide an explicit expression for the solution of the wave equation decomposing as a sum of simple sine waves. The corresponding eigenvalues of $\mathcal{L}[\cdot]$ completely describe the shape we are considering, in this case a line segment of length ℓ.

9.2 REGIONS IN $\mathbb{R}^n$

9.2.1 *Wave Equation on* $\mathbb{R}^n$

As illustrated in Figure REF, we now consider the case wave equation on a compact domain $\Omega \subseteq \mathbb{R}^n$ whose boundary $\partial\Omega \subset \mathbb{R}^n$ is a smooth $(n-1)$-dimensional submanifold. This case encapsulates the case of a vibrating drum head, when $n = 2$.

We can extend the (unitless) wave equation $u_{tt} = u_{xx}$ using similar arguments to those appearing in §9.1, yielding the following coupled system for the wave height $u(\mathbf{x}, t)$ as a function of location $\mathbf{x} \in \Omega$ and time $t \geq 0$:

$$\begin{aligned}\frac{\partial^2 u(\mathbf{x},t)}{\partial t^2} &= \textstyle\sum_i \frac{\partial^2 u(\mathbf{x},t)}{\partial (x^i)^2} \\ u|_{\partial\Omega} &\equiv 0\end{aligned} \tag{9.6}$$

Here, the second condition expresses *Dirichlet* boundary conditions, that is, that the vibrating drum head is clamped onto a flat, nonmoving solid object on the boundary $\partial\Omega$ for all time.

The same argument as the previous section shows you can construct solutions to this more generic wave equation using analogous formulas to (9.4); exercise 9.3. works this formula out in detail. This motivates finding the eigenvalues of the operator appearing on the right-hand side of (9.6), which takes the place of $\mathcal{L}$ from the §9.1:

$$\Delta[\cdot] : u(\mathbf{x}, t) \mapsto -\sum_i \frac{\partial^2 u(\mathbf{x}, t)}{\partial (x^i)^2}, \tag{9.7}$$

with boundary conditions $u|_{\partial\Omega} \equiv 0$. This operator, which takes twice-differentiable functions on Ω to their second derivatives summed over i, is known as the *Laplacian operator* on Ω. Often the Laplacian is notated $\Delta := -\nabla \cdot \nabla$, where ∇ is the gradient operator, invoking the formal relationship

$$\Delta = -\sum_i \frac{\partial^2}{\partial (x^i)^2} = -\underbrace{\begin{pmatrix} \frac{\partial}{\partial x^1} & \cdots & \frac{\partial}{\partial x^n} \end{pmatrix}}_{\text{“}\nabla^\top\text{”}} \underbrace{\begin{pmatrix} \frac{\partial}{\partial x^1} \\ \vdots \\ \frac{\partial}{\partial x^n} \end{pmatrix}}_{\nabla} .$$

Our definition of Δ makes a controversial choice: **We have included a negative sign.** There is disagreement in the literature—and sometimes among different sections of the same research paper—as to the "proper" sign of the Laplacian operator. We choose choose our sign to make analogy to positive definite matrices appearing in the computational pipeline.

A suspicious aspect of (9.6) is the fact that the sum is over coordinates i: It is not clear from this expression that if we rotate a drum by an angle that is not a multiple of 90° whether its vibration modes might change. Of course we know that this is not the case, and indeed the following proposition verifies invariance of Δ to rigid motion:

Proposition 9.1. *Suppose* $g(\mathbf{x}) := f(R\mathbf{x} + \mathbf{t})$, *where* $R^\top R = RR^\top = I_{n\times n}$. *Then,* $\Delta g(\mathbf{x}) = [\Delta f](R\mathbf{x} + \mathbf{t})$.

Proof. It suffices to show that the Laplacian is translation- and rotation-invariant independently; composing these properties leads to the formula in the proposition. Checking invariance to translation by a vector $\mathbf{t}$ is a short computation. Define $\mathbf{y} := R\mathbf{x} + \mathbf{t}$. By the chain rule, we can write

$$\frac{\partial}{\partial x^i} = \sum_j \frac{\partial y^j}{\partial x^i} \frac{\partial}{\partial y^j} = \sum_j R_i^j \frac{\partial}{\partial y^j}.$$

Hence,

$$\nabla_{\mathbf{x}} = -\sum_i \frac{\partial^2}{\partial (x^i)^2} = -\left(\sum_j R_i^j \frac{\partial}{\partial y^j}\right) \cdot \left(\sum_j R_i^j \frac{\partial}{\partial y^j}\right) = -\sum_j \frac{\partial^2}{\partial (y^j)^2},$$

by orthogonality of R, as needed. □

9.2.2 *Dirichlet Energy and Harmonic Functions*

The Laplacian operator appears in countless applications, many of which (superficially) have little to do with physics. For instance, in interpolation, suppose we are given the values of a function $u : \partial\Omega \to \mathbb{R}$ of a function u on the boundary $\partial\Omega$ that we wish to interpolate to the interior of Ω. A potential *nonparametric* approach to this regression-style problem is to find a smooth function u agrees with the given boundary data. In this case, we need to quantify what it needs for a function to be smooth.

One reasonable proxy for smoothness is to seek a function whose gradient has a relatively small norm at all points in the domain; after all, in some sense the smoothest possible function is the constant function, whose gradient equals zero. This idea is formalized using the *Dirichlet energy* of the function u, which measures the total norm of the gradient ∇u:

$$E[u] := \frac{1}{2}\int_\Omega \|\nabla u(\mathbf{x})\|_2^2\, d\mathbf{x}. \tag{9.8}$$

Then, our interpolation task can be written

$$\begin{array}{rl} \min_{u(\mathbf{x}):\Omega\to\mathbb{R}} & E[u] \\ \text{subject to} & u|_{\partial\Omega} \text{ prescribed.} \end{array} \tag{9.9}$$

We can apply integration by parts (in this case the Divergence Theorem) to give an alternative formula for $E[u]$:

$$\begin{aligned} E[u] &= \frac{1}{2}\int_\Omega \|\nabla u(\mathbf{x})\|_2^2\, d\mathbf{x} \\ &= \frac{1}{2}\int_\Omega \nabla u(\mathbf{x})\cdot\nabla u(\mathbf{x})\, d\mathbf{x} \\ &= \frac{1}{2}\oint_{\partial\Omega} u(\mathbf{x})\nabla u(\mathbf{x})\cdot\mathbf{n}(\mathbf{x})\, d\mathbf{x} + \frac{1}{2}\int_\Omega u(x)\Delta u(x)\, d\mathbf{x}. \end{aligned} \tag{9.10}$$

For any function $w(\mathbf{x})$ satisfying $w|_{\partial\Omega}\equiv 0$, since u locally minimizes $E[\cdot]$ with fixed boundary conditions we must have $\frac{d}{dh}E[u+hw]|_{h=0}=0$. Carrying this argument through shows that $u(\mathbf{x})$ must satisfy $\Delta u(\mathbf{x})\equiv 0$ in the interior of Ω, a famous PDE known as the *Dirichlet equation*. Solutions to this PDE are known as *harmonic functions* and satisfy countless elegant properties.

Example 9.1 (Mean value property). *JS: Informally derive mean value property here.*

Example 9.2 (Greens functions). *JS: Give explicit solution to Dirichlet problem in $\mathbb{R}^n$ here.*

9.2.3 *Laplacian Eigenvalues on $\mathbb{R}^n$*

Consider the "unit sphere" of functions on Ω, given by those functions $u(\mathbf{x})$ that satisfy $\int_\Omega u(\mathbf{x})^2\, d\mathbf{x} = 1$; we restrict ourselves to this set because scaling a function by a constant does not affect its qualitative structure. Using Lagrange multipliers, we can consider the set of stationary points of the Dirichlet energy $E[\cdot]$ in (9.8) on the sphere as critical points of the functional Λ over functions $u(\mathbf{x}):\Omega\to\mathbb{R}$ and Lagrange multipliers $\lambda\in\mathbb{R}$

$$\Lambda[u;\lambda] := \frac{1}{2}\int_\Omega \|\nabla u(\mathbf{x})\|_2^2\, d\mathbf{x} + \lambda\frac{1}{2}\left[1 - \int_\Omega u(\mathbf{x})^2\, d\mathbf{x}\right]$$

Take another arbitrary differentiable function $v(\mathbf{x})$. Assuming u is a stationary point, the directional derivative of Λ in the $v(\cdot)$ direction vanishes:

$$\begin{aligned} 0 &= \frac{d}{dh}\Lambda[u+hv;\lambda]\bigg|_{h=0} \\ &= \left[\frac{1}{2}\int_\Omega \frac{d}{dh}\|\nabla u(\mathbf{x}) + h\nabla v(\mathbf{x})\|_2^2\, d\mathbf{x} - \lambda\frac{1}{2}\int_\Omega \frac{d}{dh}(u(\mathbf{x})+hv(\mathbf{x}))^2\, d\mathbf{x})\right]\bigg|_{h=0} \\ &= \int_\Omega \nabla u(\mathbf{x})\cdot\nabla v(\mathbf{x})\, d\mathbf{x} - \lambda\int u(\mathbf{x})v(\mathbf{x})\, d\mathbf{x} \\ &= \int_\Omega v(\mathbf{x})(\Delta u(\mathbf{x}) - \lambda u(\mathbf{x}))\, d\mathbf{x} + \oint_{\partial\Omega} v(\mathbf{x})\nabla u(\mathbf{x})\cdot\mathbf{n}(\mathbf{x})\, d\mathbf{x}. \end{aligned} \tag{9.11}$$

This this expression vanishes for *any* choice of v, in the interior of Ω we deduce that u satisfies the eigenvalue relationship $\Delta u(\mathbf{x}) = \lambda u(\mathbf{x})$, that is, u is an eigenfunction of Δ. This justifies thinking of Laplacian eigenfunctions as "locally smoothest" functions over Ω as measured by the Dirichlet energy.

Depending on assumptions about u we can derive two different boundary conditions for our eigenvalue problem:

- Natural (Neumann) boundary conditions: If we make no assumptions on the behavior of $u(\mathbf{x})$ at $\partial\Omega$, the expression (9.11) above actually implies boundary conditions on $u(\mathbf{x})$ nonetheless, namely that $\nabla u(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0$ for all boundary points $\mathbf{x} \in \partial\Omega$. These are known as "natural" boundary conditions because they arose from differentiating Λ rather than any explicit conditions on $u(\cdot)$. Physically these conditions correspond to a vibrating sheet of metal floating in space without a clamped boundary.

- Dirichlet boundary conditions: Alternatively, if we assume from the start that $u|_{\partial\Omega} \equiv 0$, we can take $v|_{\partial\Omega} \equiv 0$ and safely ignore the boundary integral in (9.11) altogether, leading to the Dirichlet eigenvalue problem for Ω; this scenario corresponds to a vibrating drum head clipped to the boundary $\partial\Omega$ so that it does not move.

In either case, we are left with the Laplacian eigenvalue problem:

$$\boxed{\begin{array}{rl} \text{Find} & u(\mathbf{x}) : \Omega \to \mathbb{R} \text{ and } \lambda \in \mathbb{R} \\ \text{satisfying} & \Delta u(\mathbf{x}) = \lambda u(\mathbf{x})\ \forall \mathbf{x} \in \Omega \\ \text{subject to} & \text{one of} \begin{cases} u(\mathbf{x}) = 0\ \forall \mathbf{x} \in \partial\Omega & \text{Dirichlet boundary conditions} \\ \nabla u(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0\ \forall \mathbf{x} \in \partial\Omega & \text{Neumann boundary conditions.} \end{cases} \end{array}} \tag{9.12}$$

Since the Dirichlet energy is nonnegative, we automatically have $\lambda \geq 0$. The Dirichlet and Neumann eigenvalues of Ω most likely do not not coincide.

The spectral theorem for compact, self-adjoint operators shows that there is a sequence of eigenvalue-eigenfunction pairs $\{(\lambda_k, \phi_k(\cdot))\}_{k=1}^{\infty}$ for each Ω and choice of boundary conditions. These eigenvalues do not contain limit points and hence can be sorted as $0 \leq \lambda_1 \leq \lambda_2 \leq \cdots$ with $\lambda_k \to \infty$ as $k \to \infty$. Figure REF illustrates what Dirichlet and Neumann eigenfunctions look like on a two-dimensional domain Ω.

Finally, we can state the formal version of the "Can you hear the shape of a drum?" problem: **Do there exist two domains Ω with the same sequence of eigenvalues $\{\lambda_k\}_{k=1}^{\infty}$?**

As we have mentioned, this question has been answered negatively. As an example, Figure REF shows a pair of polygonal regions in $\mathbb{R}^2$ discovered by Gordon, Webb, and Wolpert with the same spectra [8, 9]. In fact, a clever origami-like construction can be used to construct many isospectral domains by leveraging certain symmetric relationships [5].

Example 9.3 (Constructing isospectral drums). *JS: It would be fun to illustrate the construction in [5]*

These counterexamples might dampen excitement about spectral geometry, but the reality is that the sequence of eigenvalues λ_k encodes a ton of information about geometry. As a simple example, a classical result from Weyl shows that Dirichlet eigenvalues "sense" the dimensionality of the domain Ω:

Theorem 9.1 (Weyl's Law). *Let $N(\lambda)$ be the number of Dirichlet eigenvalues of the Laplacian Δ for a domain $\Omega \subseteq \mathbb{R}^d$ less than or equal to λ. Then,*

$$\lim_{\lambda \to \infty} \frac{N(\lambda)}{\lambda^{d/2}} = (2\pi)^{-d} \omega_d \mathrm{vol}(\Omega), \tag{9.13}$$

where ω_d is the volume of the unit ball in $\mathbb{R}^d$.

JS: Just to make matters more confusing, maybe for fun mention the counterintuitive fact that any *finite* sequence of eigenvalues can be realized in the spectrum of a Laplacian.

9.3 GRAPH LAPLACIANS

We are slowly expanding our landscape to include more general and flexible Laplacian operators; we started with discrete one-dimensional problems, took a limit to smooth one-dimensional problems, and have expanded our consideration to higher-dimensional domains. We next push outward in the discrete landscape to show the matrix L in (9.3) as a special case of an interesting operator on graphs—even if they do not discretize a line.

Take $G = (V, E)$ to be an unweighted, undirected graph with edges E. We can think of the space of "functions" on G as the set of functions $f : V \to \mathbb{R}$, that is, an assignment of one scalar value $f(v)$ to each vertex $v \in V$. Unlike the previous section, in this case the set of functions is finite-dimensional and isomorphic to $\mathbb{R}^{|V|}$.

Given a function $f : V \to \mathbb{R}$, we can adapt the Dirichlet energy (9.8) to G to measure its smoothness along the graph G:

$$E[f] = \frac{1}{2} \sum_{(v,w)\in E} (f(v) - f(w))^2. \tag{9.14}$$

Suppose we arbitrarily orient the edges in E, that is, we define each edge as an *ordered* pair $(v, w) \in V \times V$. There is no need to double edges forward and backward here; just one orientation will suffice. Define a *differencing operator* $D \in \{-1, 0, 1\}^{|E|\times|V|}$ whose entries are given by:

$$D_{ev} := \begin{cases} -1 & \text{if } e = (v, w) \text{ for some } w \in V \\ 1 & \text{if } e = (w, v) \text{ for some } w \in V \\ 0 & \text{otherwise.} \end{cases} \tag{9.15}$$

In some sense, we can think of D as a "discrete gradient" operator, substituting for ∇ for domains in $\mathbb{R}^n$, measuring the change of a per-vertex function along each edge. In our particular application, the sign of $D\mathbf{f}$ is irrelevant since we oriented the edges arbitrarily; this value will be squared below.

If $\mathbf{f} \in \mathbb{R}^{|V|}$ gives the values $f(v)$ for each vertex in V, then the terms summed in (9.14) are exactly the rows of $D\mathbf{f}$. Hence, we have:

$$E[\mathbf{f}] = \frac{1}{2}\|D\mathbf{f}\|_2^2 = \frac{1}{2}\mathbf{f}^\top D^\top D\mathbf{f} := \mathbf{f}^\top L\mathbf{f}, \tag{9.16}$$

where we define $L \in \mathbb{R}^{|V|\times|V|}$ to be the matrix given by

$$L_{vw} = \begin{cases} -1 & \text{if } (v, w) \in E \text{ or } (w, v) \in E \\ \text{degree}(v) & \text{if } v = w \\ 0 & \text{otherwise.} \end{cases} \tag{9.17}$$

Reverse-engineering the connection between Dirichlet energies and Laplace operators suggested in (9.10), we *define* the matrix L to be the Laplacian operator of the graph G. The Laplacian of G encodes many of its structural properties; For instance, if $\chi \in \mathbb{R}^{|V|}$ is the 0-1 indicator of a connected component of G, then $D\chi = \mathbf{0}$; this argument indicates that the dimensionality of the null space of L is exactly the number of connected components.

We can also ask the same spectral question as in previous sections: **Given the eigenvalues of L, is the graph G fully determined?** Unfortunately, while once again it is difficult to encounter isospectral graphs, some examples exist. For instance, Figure REF shows a pair of *enneahedra* whose corresponding matrices L have the same spectra.

An elegant effort in mathematics and computer science has attempted to transfer classical insight about differential Laplacian operators to the discrete case. This discipline—known as *spectral graph theory*—has demonstrated that insight from PDE has strong bearing on how we process and understand discrete structures like graphs. The famous textbook [6] documents foundational results in this area. Beyond transferring ideas from PDE to graphs, e.g. defining analogs of heat diffusion and wave propagation, an elegant insight is that the matrix L also contains significant combinatorial information about the graph G without a clear smooth analog. Examples of both are below:

Example 9.4 (Semisupervised learning)**.** *The* graph-based semisupervised learning *problem seeks to infer information expressed as per-vertex labels on a graph $G = (V, E)$ based on labels in a subset $V_0 \subseteq V$. For instance, given the structure of a social network and the political leanings of a few individuals in the network, we might attempt to guess the political leanings of the remaining members of the network. Well-known techniques like [20] optimize a discrete version of the problem* (9.9)*: Given the values of some per-vertex function $\mathbf{f} \in \mathbb{R}^{|V|}$, the remaining function values are inferred by minimizing the discrete Dirichlet energy* (9.16)*. The resulting linear system encodes the assumption that a per-vertex signal on the graph changes slowly in regions densely connected by edges.*

Example 9.5 (Kirchoff's Theorem)**.** *As an example of a combinatorial result in spectral graph theory, take $\lambda_1, \ldots, \lambda_{|V|} \in \mathbb{R}_+$ to be the eigenvalues of L for a graph $G = (V, E)$; as argued above, we know $\lambda_1 = 0$. Then, the number of spanning trees of G is given by*

$$t(G) = \frac{1}{|V|} \prod_{k=2}^{|V|} \lambda_k.$$

That is, we can obtain the number of spanning trees in a graph by computing a minor of G, providing a surprising link between linear algebra and graph topology.

9.4 SUBMANIFOLD LAPLACIANS

Now that we have built up some intuition for Laplacians on simpler domains, we finally are ready to define the Laplacian associated to a submanifold. Although the formulas are somewhat more complicated, the properties we have seen in previous sections will carry through to this case as well. Similar to the previous section, we will motivate the Laplacian as the linear operator associated to the Dirichlet energy of a function; this underscores the intuition that the Laplacian is measuring quantities like smoothness.

9.4.1 *Defining the Submanifold Laplacian*

Suppose $\mathcal{M}$ is a submanifold of $\mathbb{R}^n$, and take $f : \mathcal{M} \to \mathbb{R}$ to be a smooth function. Recall from Definition 5.1 that the *differential* $df_{\mathbf{p}} : T_{\mathbf{p}}\mathcal{M} \to \mathbb{R}$ of f at a point $\mathbf{p} \in \mathcal{M}$ is a linear operator, which maps tangent vectors $\mathbf{v} \in T_{\mathbf{p}}\mathcal{M}$ to the directional derivative $df_{\mathbf{p}}(\mathbf{v})$ of f in the $\mathbf{v}$ direction. We can give an alternative intuition for the differential as follows:

Proposition 9.2. *For each $\mathbf{p} \in \mathcal{M}$, there exists a unique vector $\nabla f(\mathbf{p}) \in T_{\mathbf{p}}\mathcal{M}$ so that $df_{\mathbf{p}}(\mathbf{v}) = \mathbf{v} \cdot \nabla f(\mathbf{p})$ for all $\mathbf{v} \in T_{\mathbf{p}}\mathcal{M}$.*

Proof. Take $\mathbf{b}_1, \ldots, \mathbf{b}_k \in T_{\mathbf{p}}\mathcal{M}$ to be a basis for the tangent space, and define $a_i := df_{\mathbf{p}}(\mathbf{b}_i)$. Define the inner product matrix $g_{ij} = \mathbf{b}_i \cdot \mathbf{b}_j$, with inverse matrix g^{ij}. We claim $\nabla f(\mathbf{p}) = \mathbf{x} := \sum_{ij} a_i g^{ij} \mathbf{b}_j$.

For a generic vector $\mathbf{v} \in T_{\mathbf{p}}\mathcal{M}$, write $\mathbf{v} = \sum_i v^i \mathbf{b}_i$. Then,

$$\begin{aligned}
\mathbf{v} \cdot \mathbf{x} &= \sum_i v^i \mathbf{b}_i \cdot \sum_{k\ell} a_k g^{k\ell} \mathbf{b}_\ell \text{ by definition of } \mathbf{v}, \mathbf{x} \\
&= \sum_{ik\ell} v^i a_k g^{k\ell} g_{i\ell} \text{ by definition } g_{ij} = \mathbf{b}_i \cdot \mathbf{b}_j \\
&= \sum_i v^i a_i \text{ since we multiplied } g \text{ by } g^{-1} \\
&= \sum_i v^i df_{\mathbf{p}}(\mathbf{b}_i) \text{ by definition of } a_i \\
&= df_{\mathbf{p}}\left(\sum_i v^i \mathbf{b}_i\right) \text{ by linearity} \\
&= df_{\mathbf{p}}(\mathbf{v}).
\end{aligned}$$

Hence, $\mathbf{x}$ suffices as a choice for $\nabla f(\mathbf{p})$. To show uniqueness, suppose $\nabla f(\mathbf{p}) = \sum_j c^j \mathbf{b}_j$. Then, if $\mathcal{M}$ is k-dimensional we have k conditions $\sum_j c^j g_{ij} = df_{\mathbf{p}}(\mathbf{b}_i)$, forming an invertible $k \times k$ linear system to solve for the c^i's. The chain of equalities above already verifies that the solution to this system is to take $c^j = \sum_i a_i g^{ij}$. **JS: note: I think this is proved in a previous chapter** □

The gradient ∇f is a generalization of the notion of a function gradient to submanifolds and is our first example of a *tangent vector field* on a surface; Figure REF illustrates a function and its corresponding gradient field. Before proceeding, we also should note that our definition of ∇f and proof of Proposition 9.2 were made much easier by the assumption that our manifolds are submanifolds of $\mathbb{R}^n$ rather than e.g. Riemannian manifolds, since we made use of dot products. One theme in more advanced treatments is that the differential $df_{\mathbf{p}}$ is a purely topological object, while $\nabla f(\mathbf{p})$ incorporates geometric information; this is the motivation behind introducing df before ∇f.

In any event, as long as we know how to integrate scalar functions along submanifolds $\mathcal{M} \subseteq \mathbb{R}^n$, we now can define the Dirichlet energy of a function $f : \mathcal{M} \to \mathbb{R}$ as

$$E[f] := \frac{1}{2} \int_{\mathcal{M}} \|\nabla f(\mathbf{p})\|_2^2 \, d\text{vol}(\mathbf{p}). \tag{9.18}$$

This formula generalizes (9.8) to curved domains using the definition of gradient explored above.

More broadly, we can define an *inner product* of smooth functions as follows:

$$\langle f, g \rangle_\nabla := \int_{\mathcal{M}} [\nabla f(\mathbf{p}) \cdot \nabla g(\mathbf{p}))] \, d\text{vol}(\mathbf{p}). \tag{9.19}$$

By construction, $E[f] = \frac{1}{2}\langle f, f \rangle_\nabla$.

Inspired by our discussion in previous sections, we can make an analogy to finite-dimensional linear algebra to motivate our next step. The functional (9.19) is in some sense a quadratic form. In finite-dimensional algebra, quadratic forms can be written as maps $(\mathbf{x}, \mathbf{y}) \mapsto \mathbf{x}^\top A\mathbf{y}$. This expression, however, can be parsed as $\mathbf{x} \cdot (A\mathbf{y})$, associating the linear map $\mathbf{y} \mapsto A\mathbf{y}$ to the quadratic form. We are in a similar scenario in (9.19): We have a quadratic form $(f, g) \mapsto \langle f, g \rangle_\nabla$, and the corresponding linear operator will be the Laplacian. We can state this result as follows:

Proposition 9.3. *Suppose $\mathcal{M}$ has no boundary. Then, for each smooth function $f : \mathcal{M} \to \mathbb{R}$ there is a smooth function Δf so that*

$$\langle f, g \rangle_\nabla = \int_{\mathcal{M}} [\Delta f](\mathbf{p}) g(\mathbf{p}) \, d\text{vol}(\mathbf{p}).$$

for all smooth $g : \mathcal{M} \to \mathbb{R}$.

The operator Δ mapping $f \mapsto \Delta f$ is known as the *Laplace–Beltrami operator.*

Careful proof of this statement, which follows from the Riesz Representation Theorem, requires machinery from functional analysis out of the scope of our discussion. As an alternative, however, we can come up with an explicit formula for the divergence of a vector field on $\mathcal{M}$ and use the definition $\Delta = -\nabla \cdot \nabla$ from previous sections. This explicit formula will satisfy the property above.

Suppose $\mathcal{M}$ is an m-dimensional submanifold of $\mathbb{R}^n$, and take $\mathbf{v} : \mathcal{M} \to \mathbb{R}^n$ to be a tangent vector field (i.e., $\mathbf{v}(\mathbf{p}) \in T_{\mathbf{p}}\mathcal{M}$). Our goal is to define a notion of divergence $\nabla \cdot \mathbf{v}$ that makes sense intrinsically to the tangent space. As illustrated in Figure REF, to do so in a geometric fashion, we wish to define divergence in a fashion that lifts the Divergence Theorem from calculus to curved objects.

For $\mathbf{p} \in \mathcal{M}$, take $\phi : \mathbb{R}^m \to \mathcal{M}$ to be a parameterization with $\phi(0) = \mathbf{p}$. Take $\bar{\mathbf{n}}_{m+1}, \ldots, \bar{\mathbf{n}}_n \in \mathbb{R}^n$ to be an orthonormal basis for the normal space $(T_{\mathbf{p}}\mathcal{M})^\perp$; then, as illustrated in Figure REF, we can extend ϕ to a parameterization of the region *around* $\mathbf{p}$ in ambient space $\mathbb{R}^n$ by taking

$$\bar{\phi}(x^1, \ldots, x^n) = \phi(x^1, \ldots, x^m) + \sum_{k=m+1}^{n} x^k \bar{\mathbf{n}}_k.$$

Take for small $r > 0$, consider the "cylindrical" shape above $\mathcal{M}$ via

$$C := \{\bar{\phi}(\mathbf{x}) : \|(x^1, \ldots, x^m)\|_2 \leq r \text{ and } x^k \in [0, \varepsilon] \text{ for } k \in \{m+1, \ldots, n\}\}.$$

Figure REF illustrates the set C. For small enough $r, \varepsilon > 0$, we can lift $\mathbf{v}$ to a vector field in the region C by taking $\bar{\mathbf{v}}(\mathbf{p}) := \mathbf{v}(\bar{\phi}^{-1}(\mathbf{p}))$; that is, we simply copy $\mathbf{v}$ to nearby points in the ambient space.

The field $\bar{\mathbf{v}}$ is a field in $\mathbb{R}^n$ rather than restricted to $\mathcal{M}$, so we can apply the classical divergence theorem to write

$$\oint_{\partial C} \bar{\mathbf{v}} \cdot \mathbf{n}_C \, dA = \int_C \nabla \cdot \bar{\mathbf{v}} \, dV. \tag{9.20}$$

On the right-hand side, the integrand $\nabla \cdot \bar{\mathbf{v}}$ can be simplified at $\mathbf{p}$. By definition of divergence in $\mathbb{R}^n$, if $\mathbf{e}_1, \ldots, \mathbf{e}_m \in T_{\mathbf{p}}\mathcal{M}$ forms any orthonormal basis for the tangent space of $\mathcal{M}$ at $\mathbf{p}$, then

$$\nabla \cdot \bar{\mathbf{v}}(\mathbf{p}) = \sum_{k=1}^{m} \mathbf{e}_k \cdot d\bar{\mathbf{v}}(\mathbf{e}_k) + \sum_{k=m+1}^{n} \bar{\mathbf{n}}_k \cdot d\bar{\mathbf{v}}(\mathbf{n}_k).$$

This formula does not involve anything intrinsic to $\mathcal{M}$ but is simply a complicated way of writing the familiar formula $\nabla \cdot \bar{\mathbf{v}} = \sum_k \mathbf{e}_k \cdot \frac{\partial \bar{\mathbf{v}}}{\partial x_k}$. But, by construction we have chosen to extend $\mathbf{v}$ to $\bar{\mathbf{v}}$ so that it does not change to first order as we leave $\mathcal{M}$. Hence the second sum vanishes:

$$\nabla \cdot \bar{\mathbf{v}}(\mathbf{p}) = \sum_{k=1}^{m} \mathbf{e}_k \cdot d\bar{\mathbf{v}}(\mathbf{e}_k).$$

Although it employs the divergence on $\mathbb{R}^n$, this expression does not depend on the rate of change of $\mathbf{v}$ in the direction of the extension to C! Further, by rotation-invariance of divergence in $\mathbb{R}^n$ (again, nothing involving $\mathcal{M}$) our particular choice of the $\mathbf{e}_k$ basis did not matter. Hence, we can reasonably define

Definition 9.1 (Divergence). *The divergence of a tangent vector field* $\mathbf{v}$ *to a submanifold* $\mathcal{M} \subseteq \mathbb{R}^n$ *at point* $\mathbf{p} \in \mathcal{M}$ *is given by*

$$\nabla \cdot \mathbf{v}(\mathbf{p}) := \sum_{k=1}^{m} \mathbf{e}_k \cdot d\mathbf{v}(\mathbf{e}_k),$$

where $\mathbf{e}_1, \ldots, \mathbf{e}_m \in T_{\mathbf{p}}\mathcal{M}$ *form an orthonormal basis.*

So far, we have shown for small $r, \varepsilon > 0$ that the right-hand side of (9.20) is roughly $\nabla \cdot \mathbf{v}(\mathbf{p})$ scaled by $\operatorname{vol} C$. We can simplify the left-hand side as well. As illustrated in Figure REF, decompose C into three components: A small patch $C_0 := C \cap \mathcal{M}$ on $\mathcal{M}$, a small patch "lifted" from $\mathcal{M}$ by height ε, and an uncapped cylindrical boundary, the extrusion of $\partial C_0 \subset \mathcal{M}$. As $r, \varepsilon \to 0$, the integral terms for the surface patch and its lifted counterpart vanishes, since $\bar{n}$ is a normal and $\bar{v}$ is tangent to $\mathcal{M}$. What we are left with is a boundary integral along ∂C_0:

$$\oint_{\partial C} \bar{\mathbf{v}} \cdot \mathbf{n}_C \, dA \approx \varepsilon^{n-m} \oint_{\partial C_0} \mathbf{v} \cdot \mathbf{n}_{C_0} \, d\ell.$$

Similarly, note $\operatorname{vol} C \approx \varepsilon^{n-m} \text{Area}\, C_0$. Hence, we have motivated (in an admittedly sketchy fashion) the limit:

$$\nabla \cdot \mathbf{v} = \lim_{r \to 0} \frac{1}{\text{Area}\, C_0(r)} \oint_{\partial C_0} \mathbf{v} \cdot \mathbf{n}_{C_0} \, d\ell. \tag{9.21}$$

This expression justifies our intution for divergence as a "flux density" of $\mathbf{v}$ along $\mathcal{M}$.

Figure REF shows a typical schematic motivating the proof behind a key property of divergence of a tangent vector field $\mathbf{v}$ on $\mathcal{M}$:

$$\int_{\mathcal{M}} \nabla \cdot \mathbf{v}(\mathbf{p}) \, dA(\mathbf{p}) = \oint_{\partial \mathcal{M}} \mathbf{v}(\mathbf{p}) \cdot \mathbf{n}_{\partial \mathcal{M}}(\mathbf{p}) \, d\ell(\mathbf{p}). \tag{9.22}$$

This property, known as the *divergence theorem*, is illustrated in Figure REF. In some sense, it can be obtained by partitioning $\mathcal{M}$ into a set of infinitesimally small patches C_0 and summing hte result of (9.21). Here, $\mathbf{n}_{\partial \mathcal{M}}(\mathbf{p}) \in T_{\mathbf{p}}\mathcal{M}$ denotes the normal to the boundary $\partial \mathcal{M}$ in the tangent plane.

At this point, we have *finally* justified the formula $\Delta = -\nabla \cdot \nabla$, since we have on hand both the divergence and gradient operator for tangent vector fields. Exercise 9.6. double-checks—now with a calculation rather than an analytical argument—that Proposition 9.3 holds, closing the loop in our discussion.

9.4.2 *Submanifold Laplacian Eigenfunctions*

The motivation for the Laplacian eigenfunction problem in §9.2.3 holds with little change when we extend from regions in $\mathbb{R}^n$ to submanifolds. Since we defined a Dirichlet energy in the previous section, we can motivate the Dirichlet and Neumann eigenvalue problems variationally, equipping them with a similar interpretation. From a physical perspective, we can think of the eigenfunctions satisfying $\Delta \phi_k = \lambda_k u_k$ as vibration modes of a curved domain made of extremely thin material in the case of surfaces. As in the flat case, Laplace–Beltrami eigenfunctions are orthogonal with respect to the inner product of functions on a surface and span the space of square-integrable functions.

9.4.3 *Properties of the Laplace–Beltrami Operator*

Spectral geometry is rich with theoretical properties for the Laplace–Beltrami operator of a (sub)manifold. Cataloging and proving all the critical properties of this operator easily could occupy an entire textbook—and indeed several such textbooks exist CITE. Below we list off some of the properties that have motivated computational applications or are used to check the performance of discretized Laplace–Beltrami operators against their smooth counterparts:

Example 9.6 (Heat equation). *Beginning with the 1D wave equation motivated in §9.1, the manifold Laplacian appears in countless PDEs governing physical phenomena. Specializing Δ to flat Euclidean domains recovers well-known physical PDEs encountered at the introductory level; extending them to curved domains like submanifolds of $\mathbb{R}^3$ helps explain dynamics when they propagate along curves, bent surfaces, and so on.*

Perhaps the simplest dynamical PDE to come from the Laplace operator is the heat equation*:*

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = -\alpha\Delta u(\mathbf{x},t), \tag{9.23}$$

subject to initial conditions specifying $u(\mathbf{x},0)$ and boundary conditions as discussed in §9.2.3. Here, the Laplacian Δ is understood to act only on the manifold variable $\mathbf{x}$. This PDE determines how an initial distribution of heat $u(\mathbf{x},0)$ diffuses along a submanifold $\mathcal{M}$ as a function of time t; the constant $\alpha > 0$ determines the diffusivity of the (isotropic) material.

Take $\{(\lambda_k,\phi_k)\}_{k=1}^{\infty}$ to be the set of Laplace-Beltrami eigenfunctions satisfying $\Delta\phi_k = \lambda_k\phi_k$, and take $u_0(\mathbf{x}) := u(\mathbf{x},0)$ to be the initial conditions. We can write u_0 in the eigenfunction basis as $u_0(\mathbf{x}) = \sum_{k=1}^{\infty} u_0^k\phi_k(\mathbf{x})$, where $u_0^k = \langle u_0,\phi_k\rangle$. Then, following an argument similar to the derivation of (9.4), we can solve (9.23) in closed-form as

$$u(\mathbf{x},t) = \sum_{k=1}^{\infty} u_0^k e^{-\alpha t\lambda_k}\phi_k(\mathbf{x}). \tag{9.24}$$

This formula motivates a key object at the intersection of PDE and geometry, the heat kernel *$\mathcal{H}_t(\mathbf{x},\mathbf{y})$, which satisfies*

$$\frac{\partial \mathcal{H}_t(\mathbf{x},\mathbf{y})}{\partial t} = -\Delta_{\mathbf{x}}\mathcal{H}_t(\mathbf{x},\mathbf{y}). \tag{9.25}$$

and approaches a δ function as $t\to 0$. If our domain is compact, we can write

$$\mathcal{H}_t(\mathbf{x},\mathbf{y}) := \sum_{k=1}^{\infty} e^{-\lambda_k t}\phi_k(\mathbf{x})\phi_k(\mathbf{y}). \tag{9.26}$$

From this definition, (9.24) for $\alpha = 1$ implies

$$u(\mathbf{x},t) = \int \mathcal{H}_t(\mathbf{x},\mathbf{y})u_0(\mathbf{y})\,d\mathbf{y}, \tag{9.27}$$

recalling the well-known fact that the heat equation in $\mathbb{R}^n$ can be solved via convolution against a Gaussian kernel. Indeed, the heat kernel associated to $\mathbb{R}^n$ is given by

$$\mathcal{H}_t(\mathbf{x},\mathbf{y}) = \frac{1}{(4\pi t)^{n/2}}e^{-\|\mathbf{x}-\mathbf{y}\|_2^2/4t}, \tag{9.28}$$

although due to non-compactness of $\mathbb{R}^n$ the series formula (9.26) is irrelevant.

Example 9.7 (Other physical PDEs)**.** *The heat equation in Example 9.6 is one of many linear PDEs from physics that can be written in terms of the Laplacian operator. Some others include:*

$$\begin{aligned} \textit{Wave equation:}\quad & \frac{\partial^2 u(\mathbf{x},t)}{\partial t^2} = -\Delta u(\mathbf{x},t) \\ \textit{Schrödinger equation:}\quad & \frac{\partial \psi(\mathbf{x},t)}{\partial t} = -i\Delta\psi(\mathbf{x},t) \end{aligned}$$

Here we have chosen to write unitless *versions of these equations that neglect important physical constants but are sufficient for shape analysis. On compact domains, eigenfunction-based analogs of the formulas in Example 9.6 can be used to derive corresponding solution and kernel formulas.*

Example 9.8 (Weyl's Law)**.** *Weyl's Law in Theorem 9.1 holds without change for curved submanifold domains, if we interpret d as the intrinsic dimensionality of the domain.*

Example 9.9 (Mean curvature normal)**.** *Recall from §5.5.1 that we can interpret the mean curvature-weighted normal vector $H\mathbf{n}$ of a surface as the first variation of surface area. For an embedded surface $\mathcal{M} \in \mathbb{R}^3$, take $\mathbf{p} \in \mathcal{M}$ and consider a local parameterization $g : U \subseteq \mathbb{R}^2 \to \mathcal{M}$ with $g(\mathbf{0}) = \mathbf{p}$ and $dg_{\mathbf{0}}(\mathbf{v}) = \mathbf{v}$—that is, g is an isometry at $\mathbf{0}$. Then, we have*

$$\Delta_{\mathbb{R}^2} g = -\frac{\partial^2 g}{\partial x^2} - \frac{\partial^2 g}{\partial y^2},$$

by definition. ***JS: then what?***

Example 9.10 (Curvature, geodesics, and heat diffusion)**.** *A number of intrinsic quantities we have already discussed are hiding in computations using the Laplacian operator. For example,* Varadhan's theorem *states that you can recover geodesic distances from the heat kernel via the relationship* CITE

$$d(\mathbf{x},\mathbf{y}) = \lim_{t\to 0} \sqrt{-4t \log \mathcal{H}_t(\mathbf{x},\mathbf{y})}. \tag{9.29}$$

It is a straightforward exercise to check this formula for the heat kernel of the plane, which is a Gaussian distribution. Similarly, suppose we write a Taylor series for $\mathcal{H}_t(\mathbf{x},\mathbf{x})$ *about* $t = 0$*:*

$$\mathcal{H}_t(\mathbf{x},\mathbf{x}) = (4\pi t)^{-1/2} \sum_{k=0}^{\infty} a_k t^k.$$

Here, the coefficient is needed to account for the blowup of $\mathcal{H}_t$ *as* $t \to 0$*. Then, we find* $a_0 = 1$ *and* $a_1 = K(\mathbf{x})/6$*, where K is Gaussian curvature* CITE.

These formulas and many others are direct consequences that the Laplacian operator completely determines the metric structure *of a manifold: If we know the manifold and its Laplacian operator, this information is enough to recover all its intrinsic properties.*

9.5 DISCRETIZING THE LAPLACIAN

Inspired by the Laplacian's integral role in theoretical analysis, myriad algorithms for understanding meshes, point clouds, and other domains build up discretized Laplacian operators to probe properties of a shape. For this reason, considerable effort has been put into computational approximation of the Laplacian, the heat kernel, vibration modes, and related objects. In more detail, suppose we represent a surface or geometric domain with k points in $\mathbb{R}^n$. Then, one reasonable model might be to discretize the set of functions as vectors in $\mathbb{R}^k$, i.e., an assignment of a scalar value per point in the domain. Then, by linearity, we can think of the Laplacian as a $k \times k$ matrix whose action approximates that of computing the sum of second derivatives (up to sign). Our task is to propose reasonable formulas for the entries of this matrix.

We already have encountered one discrete Laplacian matrix, the graph Laplacian in §9.3. This particular operator attempts to capture Laplacian-like behavior when analyzing graphs, which are not required to have manifold structure. A rudimentary Laplacian matrix for a 3D triangle mesh might be the graph Laplacian of the mesh edges, but this operator does not incorporate edge lengths or other geometric information. In contrast, our goal in this section is to formulate Laplacians that incorporate geometric rather than purely combinatoral structure.

In theory, the Laplacian of a manifold has many important properties. Following CITE and previous sections, these include the following:

- **Symmetry:** The Laplacian is a self-adjoint operator, that is, $\langle u, \Delta v\rangle = \langle \Delta u, v\rangle$ for all u, v.
- **Local support:** The Laplacian is built from derivatives, so we do not expect it to involve values of a function outside a small neighborhood.
- **Linear precision:** The Laplacian of a linear function is zero.
- **Maximum principle:** Functions u satisfying $\Delta u = 0$ have no local maxima away from the boundary of the domain.
- **Definiteness:** The Laplacian is a positive semidefinite operator, that is, $\langle \Delta u, u\rangle \geq 0$ for all u.

Each of these properties can be easily translated onto an analogous condition on our $k \times k$ matrix. A fundamental result proved in CITE about Laplacian operators on meshes is that there is "no free lunch," that is, it is *impossible* to construct a discrete Laplacian satisfying all the conditions above at once.

The "no free lunch" theorem is a curious observation that partially explains the lack of consensus on Laplacian formulas in the applied community. It also imparts a certain degree of freedom: There is no universal best Laplacian operator for every task but rather an engineer must *choose* the Laplacian whose properties are most favorable for a given application.

In any event, below we summarize a few methods for computing Laplacian operators in practice; we will encounter one or two more, e.g., when we discuss exterior calculus in future chapters. Our goal is to motivate some formulas commonly encountered in practice, and to introduce the finite element method as a general strategy for computing with differential operators.

9.5.1 *Triangle Meshes*

We begin by building up a specific Laplace operator associated to manifold triangle meshes, as defined in §REF. The demand for this Laplacian stems from applications in computer graphics, computer vision, and geometry processing, which process 3D surfaces and use Laplacians for tasks like correspondence and retrieval.

9.5.1.1 *Integration by Parts*

Our goal initially will be to introduce Laplacian operators for triangle meshes (V, E, F), with or without boundary. As a reminder, $V = \{\mathbf{v}_i\}_{i=1}^n$ is a collection of vertex positions, and E and F group the vertices in V into edges and triangular faces, respectively. We will identify the space of functions on our surface with $\mathbb{R}^n$, thought of as one value per vertex on the mesh.

Why is it so hard to approximate the Laplacian operator on a triangulated surface? The challenges in many ways resemble those when approximating curvature, introduced in Chapter 5. In particular, Laplacians are *second-order* operators—they compute second derivatives of functions. But, triangulated surfaces are roughly *first-order* objects, composed of flat faces that meet along sharp edges.

As one of many ways to motivate the ubiquitous cotangent Laplacian formula for triangulated surfaces as well as a key method in numerical partial differential equations (PDE), we will derive a discrete Laplacian using the *finite element method (FEM)*. The advantages of FEM for triangulated surfaces resemble the structure-preserving curvature discretizations in §5.7.2: By integrating against the Laplacian rather than evaluating it directly, we can significantly reduce the complexity of our computation.

To begin, for functions $u, v : \mathcal{M} \to \mathbb{R}$, we note the following integration by parts formula:

$$\int_{\mathcal{M}} u \Delta v \, dA = \int_{\mathcal{M}} \nabla u \cdot \nabla v \, dA - \oint_{\partial \mathcal{M}} u \nabla v \cdot \hat{\mathbf{n}} \, d\ell, \tag{9.30}$$

where $\hat{\mathbf{n}}$ is the outward-pointing unit normal to the boundary in the tangent plane to the surface. This formula is sometimes known as Green's first identity, and it is a consequence of the divergence theorem (or Stokes' theorem). Since we will be dealing with triangle meshes with flat facets, it is actually sufficient to use the planar version of this formula rather than the more advanced version on a curved surface, but it holds identically in either case.

We rarely step back to appreciate the remarkable power of integration by parts, but (9.30) is a truly remarkable fact. Suppose we are interested in the Laplacian Δv of a function $v : \mathcal{M} \to \mathbb{R}$. Typically we think of Δv as another function $\Delta v : \mathcal{M} \to \mathbb{R}$ obtained by differentiating v twice. But if our only use for Δv is to integrate it against other functions $u : \mathcal{M} \to \mathbb{R}$, the right-hand side of (9.30) indicates that we only need the *first derivatives* of u.

To muddy the waters even more, for a point $\mathbf{p} \in \mathcal{M}$, consider a sequence of positive functions $f_k(\mathbf{x})$ given by $f_k(\mathbf{x}) := \exp(-k \cdot d(\mathbf{x}, \mathbf{p})^2)$.[2] Then, we can compute a sequence of weighted average values:

$$a_k := \frac{\langle \Delta v, f_k \rangle}{\langle 1, f_k \rangle}.$$

Since a_k is the weighted average of Δv in smaller and smaller neighborhoods, under suitable regularity assumptions, we have $a_k \to \Delta v(\mathbf{p})$ as $k \to \infty$. But again by (9.30), each a_k is computable from first derivatives alone! Informally, this argument indicates we can use inner products $\langle \cdot, \Delta u \rangle$ to "sense" values of Δu without ever evaluating a Laplacian explicitly.

Helping formalize some of the intuition above, below we define the L^2 dual of a function, which is a function*al* defined by computing inner products:

Definition 9.2 (L^2 dual of a function)**.** *Given a function $f : \mathcal{M} \to \mathbb{R}$, we define its dual $\mathcal{L}_f : L^2(\mathcal{M}) \to \mathbb{R}$ to be the operator that inputs a function g and returns its integral against f:*

$$\mathcal{L}_f[g] := \langle f, g \rangle = \int_{\mathcal{M}} f(\mathbf{x}) g(\mathbf{x}) \, dA(\mathbf{x}). \tag{9.31}$$

Roughly using constructions similar to that of a_k above, given access to the functional $\mathcal{L}_f$ we can reconstruct the values of a function f.

From (9.30) we have:

$$\mathcal{L}_{\Delta v}[u] = \int_{\mathcal{M}} \nabla u \cdot \nabla v \, dA - \oint_{\partial \mathcal{M}} u \nabla v \cdot \hat{\mathbf{n}} \, d\ell. \tag{9.32}$$

In a sense, this formula allows us to use the Laplacian operator without ever evaluating it at an isolated point.

9.5.1.2 *Galerkin Finite Element Method (FEM)*

The space of functions $f(\mathbf{x})$ on a triangle mesh surface—or nearly any other domain—is of course infinite-dimensional: We must store the value of $f(\mathbf{x})$ at every point $\mathbf{x}$, whether $\mathbf{x}$ is a vertex, on an edge, or in the interior of a triangle. Computationally, it is impossible to $f(\mathbf{x})$ for an infinite number of points $\mathbf{x}$, so we are forced to make an approximation. One common strategy is to assume we can approximate $f(\mathbf{x})$ in a *basis*, i.e., $f(\mathbf{x}) \approx \sum_{k=1}^{K} v^k \psi_k(\mathbf{x})$ for a set of coefficients v^k and a set of basis functions $\psi_1(\cdot), \ldots, \psi_K(\cdot)$. **JS: does an earlier chapter do PL bases?**

Suppose we wish to solve the Laplace equation $\Delta f = g$. Unfortunately, even if we write $f(\cdot)$ in the basis of ψ_k's, there is no guarantee that g can be written in the same finite-dimensional basis—similarly if we swap f and g. Working out exact solutions to Laplace's equation on a triangle mesh is typically not possible, so instead we might attempt to *approximate* f and g in the same basis, by restricting to functions $f(\mathbf{x}) \approx \sum_{k=1}^{K} v^k \psi_k(\mathbf{x})$ and $g(\mathbf{x}) = \sum_{k=1}^{K} w^k \psi_k(\mathbf{x})$. We can no longer enforce equality, so instead we want the approximation

$$g(\mathbf{x}) \approx \Delta f(\mathbf{x}) \iff \sum_{k=1}^{K} w^k \psi_k(\mathbf{x}) \approx \sum_{k=1}^{K} v^k \Delta \psi_k(\mathbf{x}).$$

We have K unknowns in our problem, but ∞ relationships—one for each $\mathbf{x}$ on the meshed surface. Our task is to write a tractable problem whose solution gives a reasonable choice of v^k and w^k values that approximate solutions of the Laplace equation.

[2] Technically this function is not differentiable at the cut locus of the surface, but this is a fairly easy bug to repair! Advanced students might recognize this argument as integrating against a Dirac δ-function.

Take $\ell \in \{1, \ldots, K\}$. Integrating both sides against a *test function* $\psi_\ell(\cdot)$, we have

$$\begin{aligned}\int \psi_\ell(\mathbf{x}) \sum_k w^k \psi_k(\mathbf{x})\, dA(\mathbf{x}) &\approx \int \psi_\ell(\mathbf{x}) \Delta \sum_k v^k \psi_k(\mathbf{x}) \\ \implies \sum_k w^k \langle \psi_\ell, \psi_k \rangle &\approx \sum_k v^k \langle \psi_\ell, \Delta\psi_k \rangle \\ &= \text{boundary terms} + \sum_k v^k \int \nabla\psi_\ell(\mathbf{x}) \cdot \nabla\psi_k(\mathbf{x})\, dA(\mathbf{x}) \text{ by (9.30).} \end{aligned} \tag{9.33}$$

Our strategy here is following the logic used to derive $\mathcal{L}_{\Delta v}$ in (9.32). We choose to disregard boundary terms since for the simplest versions of Laplace's equation we will find that they are irrelevant: Typically we either prescribe values of $f(\cdot)$ on the boundary of the domain (Dirichlet conditions) or will choose boundary conditions where the boundary integral vanishes (Neumann conditions $\nabla f \cdot \hat{\mathbf{n}} = 0$); see (9.12) for an example.

Given our basis of functions ψ_k, we can define two matrices M and L as follows:

$$\begin{aligned}\text{Mass matrix: } M_{\ell k} &:= \langle \psi_\ell, \psi_k \rangle = \int \psi_\ell(\mathbf{x}) \psi_k(\mathbf{x})\, dA(\mathbf{x}) \\ \text{Stiffness matrix: } L_{\ell k} &:= \int \nabla\psi_\ell(\mathbf{x}) \cdot \nabla\psi_k(\mathbf{x})\, dA(\mathbf{x}).\end{aligned}$$

Substituting these matrices into (9.33) and accounting for different boundary conditions above in the Laplace equation, we can write:

$$\begin{aligned}\text{Neumann conditions: } & L\mathbf{v} \approx M\mathbf{w} \\ \text{Dirichlet conditions: } & \begin{cases} L_{\text{inner}}\mathbf{v} \approx M_{\text{inner}}\mathbf{w} \\ \mathbf{v}_{\text{boundary}} \text{ prescribed.} \end{cases}\end{aligned}$$

Here, the subscript "inner" refers to the subset of rows of a matrix corresponding to non-boundary vertices of the mesh, anticipating a construction we will make in §9.5.1.3 in which there is one basis function ψ_k per vertex in the mesh.

Replacing $\approx$ with $=$ in the linear expressions above gives the *Galerkin finite element method (FEM)* strategy for solving Laplace's equation. For example, if we want to approximate solutions to the Laplace equation with Neumann conditions, we can do so by writing $\mathbf{v} = L^{-1}M\mathbf{w}$. While it is outside of our discussion, the general theory of the finite element method provides conditions under which this approximation is convergent as we add more and more basis functions to our domain; see CITE for discussion. This expression also suggests a reasonable approximation of the Laplacian operator as $M^{-1}L$.

It is worth pausing to note the broad strategy above, which comprises one of the major approaches to solving PDEs. The basic strategy in Galerkin FEM is to write our input data and unknowns in some function basis $\{\psi_k\}_{k=1}^K$. We can no longer solve our PDE exactly in this basis, but if we integrate both sides against $\psi_1, \ldots, \psi_K$ we end up with a $k \times k$ invertible linear system that we can solve to approximate our unknowns. A slightly more general version of the same approach chooses a different set of ψ's for the test functions and for approximating the unknowns, but this strategy is less common in geometry processing.

A second variational perspective on Galerkin FEM also provides some insight into its behavior. For simplicity, we will only consider the Laplace equation $\Delta f = g$ with Neumann boundary conditions. Recall from our discussion in §9.2.2 that we can think of solutions of the Laplace equation with zero right-hand side as solutions to an optimization problem (9.9). Similarly, we obtain the Laplace equation $\Delta f = g$ if we optimize the following objective:

$$E_g[f] := \int \left[\frac{1}{2} \|\nabla f(\mathbf{x})\|_2^2 - f(\mathbf{x}) g(\mathbf{x}) \right] dA(\mathbf{x}).$$

Omitting boundary conditions in this optimization problem actually leads $f(\cdot)$ to satisfy Neumann conditions at optimality.JS: add as homework problem

Suppose rather than optimizing over all possible functions $f(\mathbf{x})$ we restrict to functions that can be written $f(\mathbf{x}) = \sum_k v^k \psi_k(\mathbf{x})$ and that we approximate f, g in the ψ_k basis as above. In this case,

$$E_g[f] = \int \left[\frac{1}{2} \left\| \sum_k v^k \nabla \psi_k(\mathbf{x}) \right\|_2^2 - \sum_{k\ell} v^k w^\ell \psi_k(\mathbf{x}) \psi_\ell(\mathbf{x}) \right] dA(\mathbf{x}) = \frac{1}{2} \mathbf{v}^\top L \mathbf{v} - \mathbf{v}^\top M \mathbf{w}.$$

Minimizing $E_g[\cdot]$ in this restricted space (that is, setting the gradient with respect to $\mathbf{v}$ to $\mathbf{0}$) leads to the linear system $L\mathbf{v} = M\mathbf{w}$ as above.

The variational perspective above is valuable for a few reasons. In some sense, we can think of Galerkin FEM as optimizing the same objective as the infinite-dimensional Dirichlet energy (in the case of the Laplace equation), just in a restricted basis of functions. Moreover, this perspective roughly suggests that the Dirichlet energy of the discrete solution will be *higher* than the true solution to the PDE, thanks to the fact that we have in effect added a constraint to a minimization problem.

9.5.1.3 *Piecewise-Linear Elements and the Cotangent Laplacian*

Recall our broad goal in §9.5.1 is to develop Laplacian operators on triangle meshes. The previous section reveals a strategy for deriving *many* such operators. In particular, each choice of basis $\psi_1, \ldots, \psi_K$ leads to a different Laplacian operator $M^{-1}L$. Here, we outline one particularly common choice of functional bases on triangle meshes and show that it leads to a familiar formula from Chapter 5.

Our discussion in several past sections leads to a common theme. It might appear that the Laplacian operator Δ requires taking *second* derivatives of functions, since the form of the Laplacian we are most familiar with on $\mathbb{R}^n$ looks like $\sum_k \partial^2/\partial x_k^2$. But thanks to integration by parts, this is not the case! For example, in the previous section we found that we can approximate the Laplacian as $M^{-1}L$, where the only derivatives of the basis functions ψ_k needed are *first* derivatives to construct L. Out of laziness—and for deep reasons in numerical analysis—it can be preferable to choose a function basis $\psi_1, \ldots, \psi_K$ that admits *just enough* derivatives to construct the operators we need. This is exactly what we will do in this section.

Suppose we are given a triangle mesh (V, E, F) with vertices V, edges E, and triangles F. Probably the most intuitive way to represent a function is to use a vector $\mathbf{f} \in \mathbb{R}^{|V|}$, with one value per vertex of the mesh. In this case, our task is to interpret this vector as a function the meshed domain, i.e. to "fill in" the values of our function in the interiors of the triangles and along the edges.

Consider a single triangle as in Figure REF. If we are given values of a function on the three vertices of the triangle, perhaps the only reasonable way to interpolate the function within the interior of the triangle is to use an *affine* expression; in the plane of the triangle we can write $f(\mathbf{x}) = \mathbf{q}^\top \mathbf{x} + f_0$ for some vector $\mathbf{q} \in \mathbb{R}^2$ and some constant f_0. This interpolant is exactly the barycentric interpolation strategy familiar in computer graphics for shading triangles given vertex colors.

Extending this basis to the mesh leads to the *piecewise linear* or (informally) *hat* basis for functions on a triangle mesh. In particular, given a vector of values $\mathbf{f} \in \mathbb{R}^{|V|}$, we can interpret $\mathbf{f}$ as a function along the surface as follows:

$$f(\mathbf{x}) = \sum_{v \in V} f^v h_v(\mathbf{x}), \tag{9.34}$$

where h_v is affine in the interior of every triangle in T and at the vertices satisfies:

$$h_v(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} = \text{position of } v \\ 0 & \text{otherwise.} \end{cases}$$

A hat function h_v is illustrated in Figure REF.

As illustrated in Figure REF, since the hat functions h_v are affine in each triangle, their gradients ∇h_v are *piecewise-constant*; that is, they can be written as one vector per triangle in the mesh. This makes deriving the stiffness matrix L with entries $L_{ij} = \int \nabla h_i \cdot \nabla h_j$ a fairly straightforward proposition, in which we sum one constant value per face of the mesh. **JS: add derivation here. for now see 6.838 slides since I don't want to draw pictures**

The derivation above leads us to the celebrated *cotangent Laplacian* formula:

$$L_{vw} = \frac{1}{2} \cdot \begin{cases} \sum_{i\sim k}(\cot \alpha_{ik} + \cot \beta_{ik}) & \text{if } i = j \\ -(\cot \alpha_{ij} + \cot \beta_{ij}) & \text{if } i \sim j \\ 0 & \text{otherwise.} \end{cases} \tag{9.35}$$

JS: check sign to make sure this is positive semidefinite Here, $\sim$ denotes adjacency on a triangle mesh, and α_{ik}, β_{ik} are the two angles opposite edge (i, k) as in Figure REF; for boundary edges we omit the β_{ij} term. The astute reader may recognize this expression, as it already appeared in (5.22), the gradient of mesh surface area with respect to vertex positions. This is not a coincidence; **JS: refer to an exercise**.

It is somewhat more difficult to work out elements of the mass matrix M for the hat basis. In particular, since the hat functions are piecewise linear, the products $h_v(\mathbf{x})h_w(\mathbf{x})$ are piecewise quadratic. **JS: do the integral** This leads us to the formula

$$M_{ij} = \begin{cases} \frac{1}{6} \cdot \text{area of one-ring} & \text{if } i = j \\ \frac{1}{12} \cdot \text{area of adjacent triangles} & \text{if } i \sim j \\ 0 & \text{otherwise.} \end{cases} \tag{9.36}$$

Aside (Mass lumping). *A common practice is* mass lumping, *in which M is approximated by a diagonal matrix. In particular, notice that the integral of a function f written in the hat basis can be computed as $\mathbf{1}^\top M\mathbf{f}$. Grouping this expression as $(M\mathbf{1})^\top \mathbf{f}$ helps us interpret the vector $M\mathbf{1}$ of row sums as* area weights, *whose entries give the contribution of each vertex to integrating a function on the surface. A common lumped approximation is to write M as a diagonal matrix whose elements are*

$$M_{ii} = \frac{1}{3} \cdot \textit{area of one-ring around vertex } i,$$

yielding the barycentric area weights *matrix. Other choices are illustrated in Figure REF. These different choices of M often do not affect convergence of FEM and can counteract oversmoothing associated to using a non-diagonal M matrix. On the other hand, CITE proves that lumped approximations actually throw away some geometric information about the mesh.*

Aside (Higher-order FEM). ***JS: There are lots of other FEM bases out there. Higher-order elements (e.g., piecewise quadratic) are nice and gaining popularity.***

Aside (Other triangle mesh Laplacians). ***JS: forward reference to DEC chapter, intrinsic Delaunay papers, SGP 2020 paper, etc.***

Aside (Convergence of FEM approximations). ***JS: cite some stuff***

9.5.2 *Tetrahedral Meshes*

JS: omit in 6.838, borrow formula from Alec's thesis, basically the same as the derivations above just harder to draw

9.5.3 *Polygonal and Nonmanifold Domains*

JS: omit in 6.838, mention a few SIGGRAPH/SGP papers

9.5.4 *Point Clouds*

Applications in machine learning require Laplacians of higher-dimensional manifolds. For example, a typical model is that a dataset is some sample drawn from a low-dimensional manifold embedded in higher-dimensional space; this is in some sense a curved version of the reasoning by linear methods like principal component analysis (PCA).

If we are just given a sampling of data points from a manifold $\mathcal{M} \subseteq \mathbb{R}^n$, the methods above for constructing Laplacians are no longer relevant: We lack a mesh joining our points together. We might attempt to extract a mesh from such a point cloud, but this process can be unstable and is difficult to carry out for higher-order simplices (intrinsic dimension ≥ 4). Instead, the typical strategy is to extract a Laplacian operator directly from a point cloud.

Belkin and Niyogi one of the best-known point cloud Laplacian operators in their seminal work CITE. The basic idea is to construct a graph Laplacian similar to the ones we encountered in §9.3, but now we are going to weight the terms in (9.14) with a factor that depends on distance.

As input, suppose we are given a collection of points $\mathbf{x}_1, \ldots, \mathbf{x}_K \in \mathbb{R}^n$; our assumption is that the $\mathbf{x}$'s are approximately sampled from a lower-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^n$. Our goal is to derive a symmetric, positive semidefinite matrix $L \in \mathbb{R}^{K \times K}$ whose action approximates that of the Laplacian operator of $\mathcal{M}$ acting on functions whose values are sampled at the $\mathbf{x}_k$'s.

A useful relationship between intrinsic and extrinsic geometry motivates Belkin and Niyogi's Laplacian. In particular, recall the heat kernel $\mathcal{H}_t$ defined in (9.26). A Taylor series argument reveals the following approximation of $\mathcal{H}_t$ when t is small:

$$\mathcal{H}_t(\mathbf{x}, \mathbf{y}) = (4\pi t)^{-m/2} e^{-\|\mathbf{x}-\mathbf{y}\|_2^2/4t} (\phi(\mathbf{x}, \mathbf{y}) + O(t)), \tag{9.37}$$

where m is the dimension of the manifold and $\phi(\mathbf{x}, \mathbf{y})$ is a smooth function that satisfies $\phi(\mathbf{x}, \mathbf{x}) = 1$. Hence, when $\mathbf{x}$ and $\mathbf{y}$ are close, we can write CITE

$$\mathcal{H}_t(\mathbf{x}, \mathbf{y}) \approx (4\pi t)^{-m/2} e^{-\|\mathbf{x}-\mathbf{y}\|_2^2/4t}.$$

Returning to the heat equation $u_t = -\Delta u$, we then have the following approximate formula:

$$\Delta f(\mathbf{x}) \approx \frac{1}{t} \left[f(\mathbf{x}) - (4\pi t)^{-m/2} \int_{\mathcal{M}} e^{-\|\mathbf{x}-\mathbf{y}\|_2^2/4t} f(\mathbf{y}) \, d\mathrm{Vol}(\mathbf{y}) \right], \tag{9.38}$$

with equality as $t \to 0$.

Formula (9.38) motivates the following Laplace operator approximation on our point cloud $\mathbf{x}_1, \ldots, \mathbf{x}_k$:

$$L\mathbf{f} \approx \frac{1}{t} \left[f^i - \frac{1}{K} (4\pi t)^{-m/2} \sum_{\|\mathbf{x}_j - \mathbf{x}_i\|_2 < \varepsilon} e^{-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/4t} f^j \right], \tag{9.39}$$

where $f^i = f(\mathbf{x}_i)$. This Laplacian corresponds to adding the following weights to edges in equation (9.14):

$$W_{ij} = \begin{cases} e^{-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/4t} & \text{if } \|\mathbf{x}_i - \mathbf{x}_j\|_2 < \varepsilon \\ 0 & \text{otherwise.} \end{cases}$$

This point cloud Laplacian L is an extremely popular choice, but it comes with some drawbacks that can make it difficult to use with confidence in practice. Most importantly, rarely do we know the "intrinsic dimension" of the dataset sampled by the $\mathbf{x}_i$'s, so the parameter m must be chosen based on cross validation or another procedure. Similarly, the diffusion time t is a difficult parameter to choose: If t is too large then the approximation (9.38) becomes inaccurate, but as $t \to 0$ we suffer from the fact that the $\mathbf{x}_i$'s are not sampled densely enough to capture local behavior.

9.6 APPLICATIONS OF THE LAPLACIAN

9.7 OPERATOR-BASED GEOMETRY PROCESSING

JS: might need to be its own chapter—can drop in replacements for the Laplacian that are sensitive to different aspects of geometry. for now read our survey chapter.

9.8 EXERCISES

9.1. Show that L in (9.3) is positive semidefinite.

9.2. **JS: wave equation in 1d is the only PDE whose two solutions are shifting left and right**

9.3. **JS: solve 2d wave equation in eigenfunctions**

9.4. Consider the following variational problem intended for smoothing an input function $\bar{u}$:

$$\min_u \frac{1}{2}\int_\Omega (u(\mathbf{x}) - \bar{u}(\mathbf{x}))^2 \, d\mathbf{x} + \underbrace{\frac{\alpha}{2}\int_\Omega \|\nabla u(\mathbf{x})\|_2^2 \, d\mathbf{x}}_{E[u]}. \tag{9.40}$$

Here the parameter $\alpha \geq 0$ controls the amount of smoothing. Show that u satisfies the differential equation $u + \alpha\Delta u = \bar{u}$.

9.5. **JS: show Laplacian in $\mathbb{R}^n$ is rigid invariant**

9.6. **JS: show that div and grad on a submanifold do indeed give dirichlet energy**

9.7. **JS: show that laplacian agrees with pullback in exponential map**

9.8. **JS: prove no free lunch criteria**

9.9. **JS: start with laplace eigenvalue problem and derive Galerkin FEM formulation two ways (integrating both sides against a test function, and starting from a variational problem)**