The Steiner problem on the data cube

Jimmy

September 12, 2020

This section is devoted to defining and solving a variant of the Steiner problem on a cell complex related to a binary feature matrix. The solution may be used to organize a data set and perhaps to interrogate its meaning.

Let S be a subset of the vertex set of the n-dimensional cube C and let F be the set of faces of C. Assume that each $f \in F$ contains at least one $s \in S$, or else replace C with a sub-cube by removing redundant axes. Let D denote an arbitrary subgraph of the containment directed graph on the set of standard facets of C. D is called a *description of* S if for every $s \in S$ and $f \in F$ such that $s \in f$, there is a directed path in D from s to f. We would like to prescribe a definition of *minimality* for descriptions of S for which a minimal element D_{min} can be computed effectively.

The notion of minimality most faithful to the original geometric Steiner problem is as follows. The barycentric subdivision of the standard cube complex structure of the boundary ∂C is a simplicial complex B with one vertex for each facet f of C (excluding the top-dimensional facet C itself) and a 1-cell for each pair of facets f_1, f_2 for which $f_1 \subset f_2$. There is a standard isomorphism of the geometric realization of B with ∂C for which each simplex of B is embedded with a linear map to $C \subset \mathbb{R}^n$. Each description D of S is realized as a subset of $C_1(B)$, and hence D is embedded as a graph with straight edges in ∂C. Note that the length of 1-cell corresponding to (f_1, f_2) is $\sqrt{\dim f_2 - \dim f_1}$, provided that C is defined to have side-length 2. Define $L(D)$ to be the total length of D in ∂C. D is called *geometrically minimal* if D realizes the minimum value of L among all descriptions of S.

I have not been able to find an algorithm that provably computes geometrically minimal descriptions. I have tried a few different methods, most involving binding *concepts* or *biclusters* as defined in the Formal Concept Analysis literature. A difficulty with my approach so far is that a search space defined by concept binding is readily navigated with algorithms whose local steps go in the right direction, but no L-related global structure is apparent that would provide stopping criteria. This approach may still yet be successful, especially for the generation of heuristics.

However in this section I suggest an alternative, closely related problem, for which sufficient global structure is available that a homological, linear approach substantially succeeds.

First we replace the set of descriptions D of S with the set of 1-chains $c \in C_1(B)$ such that $[c] = [c_0] \in H_1(B, A)$, where $c_0 := \sum_{e \in \text{edges } D_0} 1 \cdot e \in C_1(B)$ and $A := S \sqcup F$. (Throughout we assume integral coefficients, $\mathbb{Z}$.) I call these *description chains*. Note that the long exact sequence in homology for the pair (B, A) yields, with $H_1(B) = 0$, an embedding $H_1(B, A) \subset H_0(A)$, with respect to which:

$$[c_0] = (\overbrace{|F|, ..., |F|}^{|S| \text{ times}}, \overbrace{-|S|, ..., -|S|}^{|F| \text{ times}})$$

The equation $[c] = [c_0]$ has several consequences.

1. Before, there was a binding process that used a concept, i.e. a subset $\bar{S} \subset S$ and a subset $\bar{F} \subset F$ which together represent a maximal bipartite subgraph of D_0, to replace D_0 with a simpler description D with one additional node, the facet labelling the concept. The binding process can still be used to produce a new chain c from c_0, with the caveat that the weights of the resulting chain are now not necessarily equal to 1. The weights record the number of "strands" of D_0 which route through a given 1-cell belonging to the chain.

2. Because of (1), we cannot force the binding process for multiple concepts to be commutative. Previously there were a few different ways to define a simultanous binding, without regard for concept order, by specifying exactly which edges to include as a function of the concept set to be bound. Now, it is perhaps more helpful to use the language of homologies of strands, rather than concept binding, to describe navigation through the search space. The non-commutativity is explained through a simple example. Given 3 strands, once portions of strands 1 and 2 are bound by the application of a homology, the bound portion of 2 is no longer available for binding to strand 3, even if there is a homology of that portion of 2 to a portion of 3.

3. The search space is a sublattice $L \subset Z$, where $Z := Z_1(B, A) \subset C_1(B)/C_1(A)$. Namely L is the homology class $[c_0] := (c_0 + C_1(A)) + \partial(C_2(B)/C_2(A))$.

In practice we are not interested in chains c with negative cofficients, so the real search space is only the portion of L which lies in the positive generalized quadrant.
Define the following quadratic form Q on Z:

$$Q(c) := \sum_{e \in \text{ basis of } C_1(B)/C_1(A)} |e|^2 \cdot (\text{coefficient}(e, c))^2$$

where $|e|^2 = (\dim f_2 - \dim f_1)$ for (f_1, f_2) representing e. Note that $C_1(B)/C_1(A)$ has a basis which is nearly identical to the basis of $C_1(B)$; one must remove only the degenerate 1-chains lying in A.
A description chain $c \in L$ is called *minimal* if it realizes the infimum value of Q on L.
The problem of finding a minimal description chain can be cast as a special case of the classical Closest Vector Problem for lattices, namely the closest vector of L to 0 with respect to the norm Q, now thought of as the distance to 0. There seem to be several algorithms which yield the closest vector in some cases.
There are more efficacious algorithms for the approximate version of the Closest Vector Problem with respect to a pre-defined tolerance γ, i.e. that yield a vector at most γ from the chosen vector. This would likely be sufficient to produce useful, if not truly minimal, descriptions of S.