1 Introduction

Our today's goal is to try to minimize energy functions of the form

$$E(f) = \sum_{p \in P} c(p, f(p)) + \lambda \sum_{(p,q) \in N} d(f(p), f(q))$$

Recall that P is the set of all pixels in the image, N is the neighborhood relation (i.e. $(p, q) \in N$ if pixels p and q are neighbors, usually in the 4-neighborhood).

A labeling function $f : P \to L$ that assigns every pixel p a label $f(p)$ from a (finite) set L of labels. We want to find a labeling f that minimizes the energy $E(f)$. The first term in the definition of E quantifies how "close" is the labeling f to the original image and is called the *data* term. The second term is the *smoothness* term. The relative importance of those two terms is given by the constant λ.

2 The Ising model

In the simplest case, we have only two possible labels, 0 and 1. In this case, the energy function looks as follows:

$$E(f) = \sum_{p \in P} c(p, f(p)) \ + \ \lambda \left|\{(p,q) \in N \mid f(p) \neq f(q)\}\right| .$$

This is the so-called *Ising model*.

Let us define

$$\begin{aligned} C_0 &= \{p \in P \mid f(p) = 0\} \\ C_1 &= \{p \in P \mid f(p) = 1\} \\ c_0(p) &= c(p, 0) \\ c_1(p) &= c(p, 1) \end{aligned}$$

Then we can write

$$E(f) = \sum_{p \in C_0} c_0(p) + \sum_{p \in C_1} c_1(p) + \lambda \left|\{(p,q) \in N \mid p \in C_0 \wedge q \in C_1\}\right| .$$

We show how to find an exact minimum of such simple function. We reduce the problem of minimizing E to the problem of finding a minimum s-t cut in a (weighted) undirected graph. So, given an energy minimization problem, we build a graph G so that

- There is an one to one correspondence between cuts in G and labelings of P and
- The energy of a labeling is an increasing function of the cost of its corresponding cut. In fact, the energy of a labeling turns out be equal to the cost of the corresponding cut.

Clearly, if we can efficiently find a min-cut in G (Éva Tardos will have a guest lecture on max flows and min cuts), we can convert this cut to its corresponding labeling that minimizes the energy E.

The graph G will contain a vertex for each pixel, plus two special nodes s and t. There will be an edge between any pair of adjacent pixels with weight λ. These are called *n-links*. For each pixel p, there will be an edge from s to p with weight $c_1(p)$ and an edge from p to t with cost $c_0(p)$. These edges are the *d-links*.

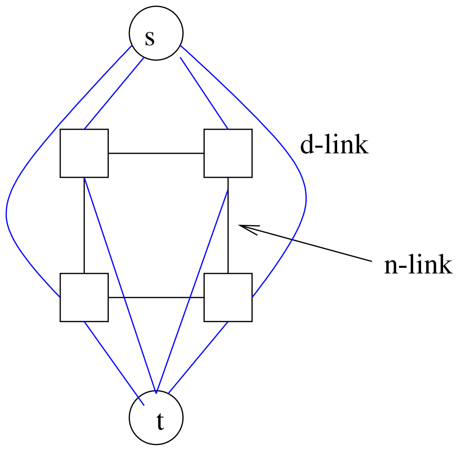


Figure 1: Reduction to minimum cut.

Any partition of the vertex set into two sets S, T, $S \cap T = \emptyset$, $S \cup T = V$, such that $s \in S$ and $t \in T$ is an s-t cut. The cost of such a cut is simply the sum of weights of edges that have one end in S and the other end in T. There is an easy one to one correspondence between cuts and labelings: simply label all pixels that are in S by 0 and all pixels that are in T by 1. Thus in our notation, $S = C_0 \cup \{s\}$, $T = C_1 \cup \{t\}$.

Now we need to show that the cost of a cut is equal to the cost of its corresponding labeling. Consider a cut (S,T). Its cost is

$$\begin{aligned}
\sum_{\substack{(p,q)\in E \\ p\in S, q\in T}} w(p,q) &= \sum_{p\in C_0} w(p,t) + \sum_{p\in C_0} w(p,t) + \sum_{\substack{(p,q)\in N \\ p\in C_0, q\in C_1}} \lambda \\
&= \sum_{p\in C_0} c_0(p) + \sum_{p\in C_1} c_1(p) + \lambda \,|\{(p,q)\in N \mid p\in C_0 \wedge q \in C_1\}| \\
&= E(f)
\end{aligned}$$

Since in the Ising model we can find the optimal labeling exactly, we can evaluate the performance of other minimization techniques. For example, one can take a noisy binary image and try to restore it using both an exact algorithm, and say, simulated annealing. Usually one finds out that annealing performs badly.[1]

[1]To be honest, one can make simulated annealing to look arbitrarily badly. The hard task is to make it look good. One needs to do a lot of parameter tuning and usually ends up setting the initial temperature a very low, thus making annealing be almost like greedy.

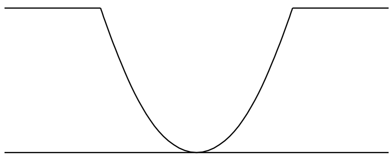
2.1 Generalizations of the Ising model

What if one needs to use more than two labels? There are some spacial cases for which the energy can be minimized efficiently. One of them is the case when the labels are integers $\{0, 1, \ldots, k-1\}$, for some $k > 0$ and the energy function is

$$E(f) = \sum_{c(p,f(p))} + \lambda \sum_{(p,q)\in N} |f(p) - f(q)|.$$

In fact, such energy functions can also be minimized using a min-cut computation (1998 Roy, later fixed by BVZ, IG).

The problem with this function is that it doesn't behave well on sharp edges in images. Energy functions that empirically do better on edges usually have the d function in the smoothness term that looks like this:



Energy functions that behave well at sharp edges and are called *discontinuity preserving.* Of course, minimizing discontinuity preserving energy functions is hard. If somebody invents a new energy function, tries it and gets poor results, there may be two reasons: (i) the function itself is bad or (ii) we do no tknow how to come even close to optimum. Usually it is hard to tell which case we're in, and thus people keep inventing more and more fancy energy functions that nobody knows how to minimize.

Next we focus on the simplest useful generelization of the Ising model for which we can find (provably) good approximate solutions fast.

3 The Potts model

In the Potts model, we have a finite set of labels L and the energy function is of the form

$$E(f) = \sum_{p\in P} c(p, f(p)) + \lambda \sum_{(p,q)\in N} d(f(p), f(q))$$

where

$$d(\alpha, \beta) = T[\alpha \neq \beta]$$

I.e., $d(\alpha, \beta)$ is 0 if $\alpha = \beta$ and 1 otherwise. The Ising model is a special case of Potts when $|L| = 2$.

In a way similar to what we did for the Ising model, we can reduce the problem of minimizing the energy function to a generalization of the s-t cut problem, called the MULTIWAY CUT.

The input to the MULTIWAY CUT[2] problem is an undirected graph $G = (V, e)$ with weights on the edges, and a set of vertices $T \subseteq V$. We call vertices in T *terminals.* The goal is to partition

[2] also known as MULTITERMINAL CUT

V into $|T|$ disjoint sets S_u (one set for each $u \in T$), such that for each $u \in T$, $u \in S_u$ and the cost of the partition is minimized. We define the cost of a partition $S_{u_1}, \ldots, S_{u_k}$ to be the sum of costs of all edges that have endpoints in different sets.

But now we have a problem: unlike the 2-way cut (aka s-t cut), the k-way cut is NP-hard for $k \geq 3$. Worse than that, the multiway cut is NP-hard even if we restrict the input graph to be a grid. Hence we cannot hope to find an exact minimum of the energy in the Potts model.

Theorem. Minimizing the energy function in the Potts model is NP hard.

Facing this cruel fact, what can we do? We can (i) invent heuristics that seem to work well on some inputs but nobody can prove anything about them (ii) try to find an approximation algorithm. In the next two lectures we will discuss an aproximation algorithm for minimizing the energy in the the Potts model based on multiway cuts. For now, let us briefly mention a quite powerful local search heuristic for minimizing the Potts energy.

An α-β *swap* is an operation that relabels some pixels labeled α with the label β and some pixels labeled β to α. The good news is, that given a pair of labels, we can efficiently compute the best α-β swap, that decreases the energy by the greatest amount – simply ignore all pixels not labeled α, β and run the optimal energy minimization algorithm for the Ising model. The local search algorithm itself is really simple:

1. Start with an arbitrary initial labeling.

2. Repeatedly pick two labels, say α and β, find the optimal α-β swap. If it decreases the overall energy, do it. If for no pair of labels α, β the swap can improve the energy, or the improvements are sufficiently small, stop.

Note that in every step of the algorithm, the energy decreases. Once the algorithm stops, it produces a solution that is a local optimum with respect to α-β swap operations.