Proof of Optimality of Huffman Codes

CSC373 Spring 2009

1 Problem

You are given an alphabet A and a frequency function $f : A \to (0, 1)$ such that $\sum_x f(x) = 1$. Find a binary tree T with $|A|$ leaves (each leaf corresponding to a unique symbol) that minimizes

$$\mathrm{ABL}(T) = \sum_{\text{leaves of } T} f(x)\mathrm{depth}(x)$$

Such a tree is called *optimal*.

2 Algorithm

HUF(A, f)
If $|A| = 1$ then return a single vertex.
Let w and y be the symbols with the lowest frequencies.
Let $A' = A \setminus \{w, y\} + \{z\}$.
Let $f'(x) = f(x)$ for all $x \in A' \setminus \{z\}$, and let $f'(z) = f(w) + f(y)$.
$T' = \mathrm{HUF}(A', f')$.
Create T from T' by adding w and y as children of z.
return T

3 Proof

Lemma 1 *Let T be a tree for some f and A, and let y and w be two leaves. Let T' be the tree obtained from T by swapping y and w. Then $ABL(T') - ABL(T) = (f(y) - f(w))(depth(w, T) - depth(y, T))$.*

Proof

$$\begin{aligned}
\mathrm{ABL}(T') - \mathrm{ABL}(T) &= f(y)\mathrm{depth}(w, T) + f(w)\mathrm{depth}(y, T) - f(w)\mathrm{depth}(w, T) - f(y)\mathrm{depth}(y, T) \\
&= f(y)(\mathrm{depth}(w, T) - \mathrm{depth}(y, T)) + f(w)(\mathrm{depth}(y, T) - \mathrm{depth}(w, T)) \\
&= (f(y) - f(w))(\mathrm{depth}(w, T) - \mathrm{depth}(y, T))
\end{aligned}$$

Lemma 2 *There exists an optimal tree such that the two symbols with the lowest frequencies are siblings.*

Proof Let T be an optimal tree. Let w and y be two symbols with the lowest frequencies. If there is more than one symbol that has the lowest frequency, then take two that have the biggest depth. If w and y are siblings, then we are done. Otherwise, suppose without loss of generality, that $\mathrm{depth}(w, T) \geq \mathrm{depth}(y, T)$. We have three cases:

- w has a sibling z. Let T' be the tree created from T by swapping z and y, and thus making w and y siblings. By applying Lemma 1, we get that $\mathrm{ABL}(T') \leq \mathrm{ABL}(T)$. Since T is optimal, there cannot be another tree with a smaller cost, and so $\mathrm{ABL}(T') = \mathrm{ABL}(T)$. Thus T' is also optimal.

- w is an only child. Create T' by removing w's leaf and assigning w to its old parent. T' is cheaper than T, contradiction the optimality of T. Hence, this case is not possible.

- There exists a node z at a depth bigger then w. Create T' by swapping w and z. By our choice of w, $f(w) < f(z)$, so, applying Lemma 1, we have that T' is cheaper than T, a contradiction. Hence, this case is not possible.

Theorem 3 *The algorithm HUF(A, f) computes an optimal tree for frequencies f and alphabet A.*

Proof The proof is by induction on the size of the alphabet. The induction hypothesis is that for all A with $|A| = n$ and for all frequencies f, $\mathrm{HUF}(A, f)$ computes the optimal tree.

In the base case ($n = 1$), the tree is only one vertex and the cost is zero, which is the smallest possible.

For the general case, assume that the induction hypothesis holds for $n - 1$. That is, T' is optimal for A' and f'. First, let us show the following:

$$
\begin{aligned}
\mathrm{ABL}(T) &= \Big(\sum_{x \in A \setminus \{w,y\}} f(x)\mathrm{depth}(x, T)\Big) + f(w)\mathrm{depth}(w, T) + f(y)\mathrm{depth}(y, T) \\
&= \Big(\sum_{x \in A \setminus \{w,y\}} f(x)\mathrm{depth}(x, T)\Big) + (f(w) + f(y))(\mathrm{depth}(z, T') + 1) \\
&= \Big(\sum_{x \in A \setminus \{w,y\}} f(x)\mathrm{depth}(x, T)\Big) + f'(z)\mathrm{depth}(z, T') + f(w) + f(y) \\
&= \Big(\sum_{x \in A'} f'(x)\mathrm{depth}(x, T')\Big) + f(w) + f(y) \\
&= \mathrm{ABL}(T') + f(w) + f(y)
\end{aligned}
$$

Now, assume for the sake of contradiction that T is not optimal, and let Z be an optimal tree that has w and y as siblings (this exists by the above lemma). Let Z' be the tree obtained from Z by removing w and y. We can view Z' as a tree for the alphabet A' and frequency function f'. We can then repeat the calculation above and get $\mathrm{ABL}(Z) = \mathrm{ABL}(Z') + f(w) + f(y)$. So, $\mathrm{ABL}(T') = \mathrm{ABL}(T) - f(w) - f(y) > \mathrm{ABL}(Z) - f(w) - f(y) = \mathrm{ABL}(Z')$. Since T' is optimal for A' and f', this is a contradiction.