

Lecture Notes, October 20, 2010: PROOF OF OPTIMALITY OF HUFFMAN'S PROCEDURE

<http://www.isi.ee.ethz.ch/teaching/courses/it1>

Recall Huffman's procedure:

Suppose that the source emits the symbol $x \in \mathcal{X}$ with probability $p_X(x)$.

- If $|\mathcal{X}| = 1$ assign the empty string to the one symbol in \mathcal{X} . In this case the construction is complete. (**exit**).
- If $|\mathcal{X}| > 1$ choose two symbols $x', x'' \in \mathcal{X}$ of least probability, and consider a new source with alphabet $\tilde{\mathcal{X}} = \mathcal{X} \setminus \{x', x''\} \cup \{\xi\}$ where ξ is some symbol that does not appear in \mathcal{X} . Set

$$p_{\tilde{X}}(\tilde{x}) = \begin{cases} p_X(x') + p_X(x'') & \text{if } \tilde{x} = \xi \\ p_X(\tilde{x}) & \text{if } \tilde{x} \in \mathcal{X} \end{cases}.$$

- Design an optimal code for the source $\tilde{\mathcal{X}}$.
- Append the string 1 to the string representing ξ , and set that to be the codeword corresponding to x' . Similarly append the string 0 to the string representing ξ , and set that to be the codeword corresponding to x'' . Delete the codeword corresponding to ξ from the codebook.

Proof of optimality: The proof is by induction on the cardinality of \mathcal{X} . The procedure is clearly optimal when $|\mathcal{X}| = 1$ because the empty string has length 0. Assume now that it is optimal for $|\mathcal{X}| = m$ and we shall prove that it then follows that it is optimal for $|\mathcal{X}| = m + 1$. Without loss in generality let $\mathcal{X} = \{1, \dots, m + 1\}$ with corresponding probabilities

$$p_1 \geq \dots \geq p_{m+1} > 0.$$

(We are assuming that symbols that have zero probability need not be encoded.)

Let $L^*(p_1, \dots, p_{m+1})$ denote the expected length of the best prefix free code for a source with probabilities (p_1, \dots, p_{m+1}) .

Step 1: We claim that there exists a prefix-free code C for a source with probabilities p_1, \dots, p_{m+1} that satisfies $L(C) = L^*(p_1, \dots, p_{m+1})$ (optimal) and that additionally satisfies that the codewords corresponding to m and the codeword corresponding to $m + 1$ are siblings.

We show this as follows. Let l_{\max} denote the greatest length among the lengths of the codewords of some optimal code C^* . Note that there must be at least two codewords of length l_{\max} for otherwise we could trim the tree by replacing the codeword of length l_{\max} with its father, and thus reduce the expected length of the code.

Now note that if $p_i \geq p_j$ and $l_i \geq l_j$ then exchanging the codewords corresponding to i and j cannot increase the expected length of the code, and we thus deduce that there exists an optimal codebook that assigns both to m and to $m + 1$ codewords of length l_{\max} .

Consider now the set of all the symbols that are mapped to codewords of the maximal length l_{\max} . We have established that there is no loss in optimality in assuming that this set contains m and $m+1$. Exchanging the codewords corresponding to any two symbols that are mapped to codewords of equal length does not change the expected length. By performing such operations among the codewords of maximal length we deduce that there exists an optimal codebook that assigns m and $m+1$ sibling codewords of maximal length, thus establishing our claim.

Step 2: By Step 1 of the proof, we may restrict our code search to codes that assign to m and to $m+1$ codewords that are siblings. Define for any such code C a corresponding code \tilde{C} by choosing the first $m-1$ codewords the same as in code C and the m -th codeword as the m -th codeword of C (or equivalently as the $m+1$ -st codeword of C) without the last bit. Note that then the expected length $L(C)$ satisfies $L(C) = p_m + p_{m+1} + L(\tilde{C})$ where $L(\tilde{C})$ is the expected length of the code \tilde{C} when used for a source which emits the symbols $1, \dots, m$ with probabilities $p_1, \dots, p_{m-1}, p_m + p_{m+1}$. Therefore in order to minimize $L(C)$ one has to minimize the expected length $L(\tilde{C})$ of the corresponding code \tilde{C} , and consequently,

$$L^*(p_1, \dots, p_{m+1}) = p_m + p_{m+1} + L^*(p_1, \dots, p_{m-1}, p_m + p_{m+1}).$$

Step 3: By the induction hypothesis, Huffman's procedure results in a code of expected length $p_m + p_{m+1} + L^*(p_1, \dots, p_{m-1}, p_m + p_{m+1})$ and is hence, by Step 2, optimal. \square