Introduction to Machine Learning - COS 324

Homework Assignment 7 Solutions

I. Compute the entropy of the following distributions:

(i) The distribution on integers from one to $n \geq 2$, where i has probability proportional to 2^{-i} (scaled such that all probabilities sum up to one). Stated equivalently, for this distribution it holds that

$$\frac{\Pr[i]}{\Pr[i+1]} = 2$$

Solution: The probability of i is $\Pr[i] = \frac{2^{-i}}{\sum_{j=1}^{n} 2^{-j}} = \frac{2^{-i}}{1-2^{-n}}$ $(i = 1, \ldots, n)$. Therefore the entropy of this distribution is

$$\begin{aligned}
\text{Entropy} = -\sum_{i=1}^{n} \Pr[i] \log_2 \Pr[i] &= -\sum_{i=1}^{n} \frac{2^{-i}}{1-2^{-n}} \log_2 \frac{2^{-i}}{1-2^{-n}} \\
&= \frac{1}{1-2^{-n}} \sum_{i=1}^{n} 2^{-i} \left(i + \log_2(1-2^{-n})\right) \\
&= \frac{1}{1-2^{-n}} \sum_{i=1}^{n} 2^{-i} \left(i + \log_2(1-2^{-n})\right) \\
&= \frac{1}{1-2^{-n}} \sum_{i=1}^{n} 2^{-i} i + \frac{\log_2(1-2^{-n})}{1-2^{-n}} \sum_{i=1}^{n} 2^{-i} \\
&= \frac{1}{1-2^{-n}} \sum_{i=1}^{n} 2^{-i} i + \frac{\log_2(1-2^{-n})}{1-2^{-n}} (1-2^{-n}) \\
&= \frac{1}{1-2^{-n}} S + \log_2(1-2^{-n}),
\end{aligned}$$

where

$$S = \sum_{i=1}^{n} 2^{-i} i. \tag{1}$$

Note that

$$2S = \sum_{i=1}^{n} 2^{-(i-1)} i = \sum_{i=0}^{n-1} 2^{-i}(i+1). \tag{2}$$

Subtracting (2) by (1), we get

$$S = 2^{-0}(0+1) + \sum_{i=1}^{n-1} 2^{-i} - 2^{-n}n = 2 - 2^{-(n-1)} - 2^{-n}n.$$

Therefore Entropy $= \frac{1}{1-2^{-n}}(2 - 2^{-(n-1)} - 2^{-n}n) + \log_2(1-2^{-n}) = 2 - \frac{n}{2^n-1} + \log_2(1-2^{-n})$.

(ii) The uniform distribution on all binary strings of length n, with exactly k ones.

Solution: This is a uniform distribution over $\binom{n}{k}$ elements. Its entropy is $-\sum_{i=1}^{\binom{n}{k}} \frac{1}{\binom{n}{k}} \log_2 \frac{1}{\binom{n}{k}} = -\binom{n}{k}\frac{1}{\binom{n}{k}} \log_2 \frac{1}{\binom{n}{k}} = \log_2 \binom{n}{k}$.

II. In this exercise we show that entropy is a lower bound on lossless compression. Suppose files are sequences of m bits, of which $m \cdot p$ are 1 and $m \cdot (1-p)$ are 0. Here $p \in (0,1)$ is some fraction.

(i) Give an expression for the total number of distinct files.

Answer: $\binom{m}{mp}$.

(ii) Let N be the number computed in the previous part. Show that

$$\lim_{m\to\infty} \frac{1}{m} \log N = H(X_p),$$

where X_p is a Bernoulli random variable with parameter p.

You may use Stirling's approximation:

$$n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n.$$

Proof: Using Stirling's approximation:

$$\begin{aligned}
\lim_{m\to\infty} \frac{1}{m} \log N &= \lim_{m\to\infty} \frac{1}{m} \log \binom{m}{mp} = \lim_{m\to\infty} \frac{1}{m} \log \frac{m!}{(mp)!(m(1-p))!} \\
&= \lim_{m\to\infty} \frac{1}{m} \log \frac{\sqrt{2\pi m}\left(\frac{m}{e}\right)^m}{\sqrt{2\pi mp}\left(\frac{mp}{e}\right)^{mp} \sqrt{2\pi m(1-p)}\left(\frac{m(1-p)}{e}\right)^{m(1-p)}} \\
&= \lim_{m\to\infty} \frac{1}{m} \log \frac{1}{\sqrt{2\pi mp(1-p)}p^{mp}(1-p)^{m(1-p)}} \\
&= -\lim_{m\to\infty} \frac{\log \sqrt{2\pi mp(1-p)} + mp\log p + m(1-p)\log(1-p)}{m}
\end{aligned}$$

$$= -\lim_{m\to\infty}\left(\frac{\log\sqrt{2\pi mp(1-p)}}{m} + p\log p + (1-p)\log(1-p)\right)$$

$$= p\log p + (1-p)\log(1-p) = H(X_p). \quad \square$$

(iii) Imagine a file compression algorithm that, given any file of length m, compresses it to $\tilde{m}$ bits. Show that if $\tilde{m} < m\cdot(H(X_p)-\varepsilon)$ for some $\varepsilon > 0$, then it must necessarily be a lossy compression; meaning that two different files must correspond to the same compressed file.

Proof: There are $2^{\tilde{m}}$ files of length $\tilde{m}$. It suffices to show $2^{\tilde{m}} < N = \binom{m}{mp}$ for sufficiently large m. We have

$$2^{\tilde{m}} < N = \binom{m}{mp} \iff \tilde{m} < \log N \Longleftarrow m(H(X_p)-\varepsilon) < \log N$$

$$\iff H(X_p) - \varepsilon < \frac{1}{m}\log N.$$

The last inequality holds for sufficiently large m because the RHS has limit $H(X_p)$ when $m \to \infty$. $\square$

III. Let $\varepsilon, \delta > 0$ be two given parameters. Using the fundamental theorem of statistical learning, compute an upper bound on the number of examples needed to learn a binary decision tree with k nodes over n variables, that will attain generalization error at most ε with probability $1-\delta$.

Solution: The number of decision trees with k nodes over n variables is at most $n^k(2k+1)!$. (See lecture notes 13.) Thus the sample complexity (in the realizable setting) is $O\left(\frac{\log\left(n^k(2k+1)!)/\delta\right)}{\varepsilon}\right) = O\left(\frac{k\log n + k\log k + \log\delta^{-1}}{\varepsilon}\right)$.