EE5585 Data Compression February 19, 2013

Lecture 9

Instructor: Arya Mazumdar *Scribe: Surya Ramachandran*

Rate distortion

The theory of quantization (Lossy compression) is given by Rate Distortion. Consider that we have a binary file, X. The file can take any n-length binary string, where n is any large integer. Assuming that we do not have any information about the statistics, the file can take any value.

$$X = \{0, 1\}^n$$

If it is uniform, and every bit is independant and takes values 1 and 0 with equal probability of 1/2, then it cannot be compressed any further. The limit of the compression is given by the binary entropy function of half.

$$\text{h}(1/2) = 1$$

Note: H(X) represents the Entropy of a general random variable; h(x) represents the binary entropy function of bernoulli random variables.

Hence, the file cannot be compressed any further without any loss. In this case, we introduce a distortion in reconstructing the file to further compress it. For binary strings, the distortion measure we use is Hamming Distortion. Let us consider two files-

10101**01**010
10101**10**010

These two files do not agree in two positions. Hence, the distance between them is 2. This distance is called the Hamming distance, a valid metric on the space of all n length binary strings. Hence, it has the following properties:

- It is always non-negative
- It is symmetric
- It satisfies the triangle inequality

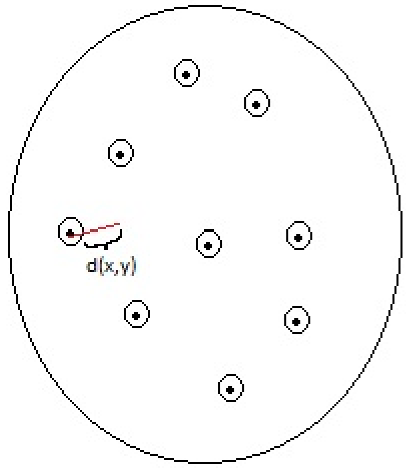
Code

A code, or source code C is a set of n length strings.

$$C \subseteq \{0, 1\}^n$$

C represents the centers or cluster points in the subspace. Whenever we are given a file which is a part of the set, we try to map it to the nearest center. Then, we do a binary encoding of the center. The number of codewords in the set is represented by $|C|$. To have $|C|$ codewords, we require $\log |C|$ bits. To encode the entire space, we require n bits, because the size of the space is 2^n. But the size of $|C|$ is much lesser than 2^n, because of which we require fewer bits to encode, due to which we achieve compression. However, while reconstructing, although we know the center point, there is no way of knowing exactly which point in the space got mapped to that center. This gives rise to the distortion. A code C achieves distortion Dn, where $0 \leq D \leq 1$, if for all $y \in \{0, 1\}^n$, $\exists x \in C$ such that $d(x, y) \leq nD$. Therefore, for any given point, we can find one codeword in the set, such that the point is atmost distance nD away.

Figure 1: Subspace of points



Let $M(n, D)$ be the size of the smallest code that recieves distortion nD. The compression ratio is given by $\frac{\log M(n,D)}{n}$.

$$R(D) = \lim_{n \to +\infty} \frac{\log M(n, D)}{n}$$

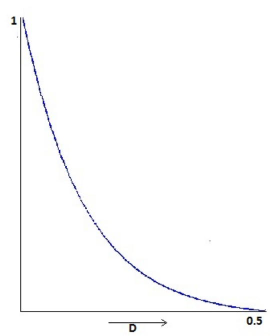
Assuming that this limit exists, to achieve a normalized distortion D, $R(D)$ is the max level of compression that can be done. D is called the *worst case distortion function*, because there is no probability associated with it.

Theorem

$$R(D) = 1 - h(D)$$

When D is $1/2$, the distortion is $n/2$. We can take two n length code words-

Figure 2: Rate Distortion



111...11
000...00

Now whatever binary string is given, the distance between the string and one of these codewords will be less than or equal to $n/2$. We can map the string to the code with the smaller distance. Hence, the worst case distortion is $n/2$. The given string is compressed to just one bit, (either 0 or 1). We can prove the theorem with the help of two claims.

Claim 1

$$R(D) \geq 1 - \mathrm{h}(D)$$

Proof Consider a code of size M. For any given point, we can find a center which has distortion atmost nD from that point. Let us draw spheres of radius nD around every center, covering every point in the space.

$$\begin{aligned} M \times \sum_{i=0}^{nD} \binom{n}{i} &\geq 2^n \\ M &\geq \frac{2^n}{\sum_{i=0}^{nD} \binom{n}{i}} \end{aligned}$$

We know that,

$$\begin{aligned} \sum_{i=0}^{nD} \binom{n}{i} &\leq 2^{n\mathrm{h}(D)} \\ M &\geq 2^{n[1-\mathrm{h}(D)]} \\ \frac{\log M}{n} &\geq 1 - \mathrm{h}(D) \\ R(D) &\geq 1 - \mathrm{h}(D) \end{aligned}$$

Claim 2

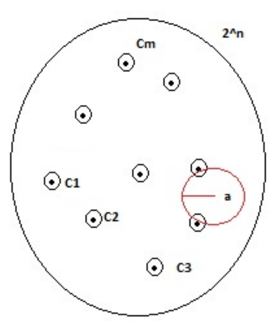
$$R(D) \leq 1 - h(D)$$

Proof: Choose a code C of size M.

$$M \equiv 2n \ln 2 \times \frac{2^n}{\sum_{i=0}^{nD} \binom{n}{i}}$$

This choice of code is random, uniform and independent of each other.

Figure 3: Random Coding



$$C = c_1, c_2, \ldots, c_m$$

A file is called a *Bad File* if there is no point within distance D to any point in C. Given a point a,

$$P(d(c_1, a) \leq nD) = \frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n}$$

The probability that a is bad w.r.t c_1 is

$$1 - \frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n}$$

Hence, the probaility that a is a bad file is

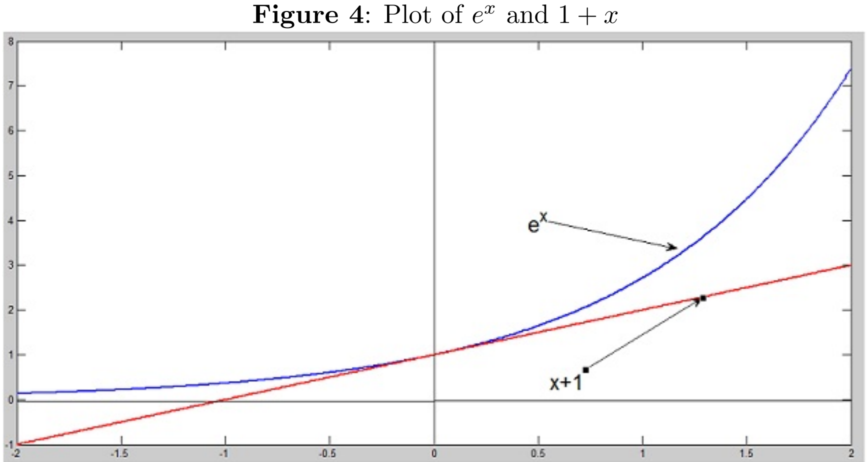
$$(1 - \frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n})^M$$

The average number of bad files is given by

$$2^n \times (1 - \frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n})^M$$

From the below graph, we can see that $e^x \geq 1 + x$ Hence, the average number of bad files is

Figure 4: Plot of e^x and $1 + x$



$$\leq 2^n \times (e^{-\frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n}})^M$$

This can be written as

$$e^{n \ln 2 - M \frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n}}$$

On expanding,

$$e^{n \ln 2 - 2n \ln 2 \frac{2^n}{\sum_{i=0}^{nD} \binom{n}{i}} \times \frac{\sum_{i=0}^{nD} \binom{n}{i}}{2^n}}$$

$$\begin{aligned} &= e^{-n \ln 2} \\ &= 2^{-n} \end{aligned}$$

2^{-n} is exponentially close to 0 and strictly less than 1. Therefore, the average number of bad files is less than 1. There has to be a code of size $2n \ln 2 \times \frac{2^n}{\sum_{i=0}^{nD} \binom{n}{i}}$ for which there are no bad files. There exists a code that achieves normalized distortion D, and has size $2n \ln 2 \times \frac{2^n}{\sum_{i=0}^{nD} \binom{n}{i}}$.

$$\begin{aligned}
\frac{\log M}{n} &= \frac{1}{n} \log \frac{2n \ln 2 \times 2^n}{\sum_{i=0}^{nD} \binom{n}{i}} \\
\sum_{i=0}^{nD} \binom{n}{i} &\geq \frac{2^{n\mathrm{h}(D)}}{\sqrt{8n(1-D)D}} \\
\frac{\log M}{n} &\leq \frac{1}{n} \log \frac{2^n 2n \ln 2 \sqrt{8n(1-D)D}}{2^{n\mathrm{h}(D)}} \\
&= 1 - \mathrm{h}(D) + \frac{1}{n} \log(2n \ln 2 \sqrt{8n(1-D)D})
\end{aligned}$$

When $n \to \infty$, the last term tends to 0. There exists a code that at a limit of $n \to \infty$, achieves the distortion $1 - \mathrm{h}(D)$. For the smallest code, this function is the rate distortion function.

$$R(D) \leq 1 - \mathrm{h}(D)$$

Hence, this proves Claim 2. Since both Claim 1 and 2 are valid, our Theorem $R(D) = 1 - \mathrm{h}(D)$ is proved.

We proved this theorem for the *worst case distortion*. Now, we shall look at *average case distortion*. In this case, there are certain probability distributions and statistics among the file. Let us consider an n length string, where each bit is a independent Bernoulli random variable. Code C achieves average distortion nD, if

$$E[d(X, C)] \leq nD$$

The distance $d(X, C)$ is the minimum of all the distances $d(X, Y)$, where $Y \in C$.

$$R_{av}(D) = \lim_{n \to +\infty} \frac{M_{av}(n, D)}{n}$$

Theorem

$$R_{av}(D) = 1 - \mathrm{h}(D)$$

For a large file, it does not matter whether it is the worst case distortion or average distortion. The same rate of compression is achievable.

Claim 1

$$R_{av}(D) \leq 1 - \mathrm{h}(D)$$

We have proved that

$$R(D) \leq 1 - \mathrm{h}(D)$$

for the worst case distortion. Because this holds for the worst case distortion D, it has to achieve an average distortion of at most D. This is called the *Direct part/ Achievability.*

Claim 2

$$R_{av}(D) \geq 1 - \mathrm{h}(D)$$

This is called the *Converse/Impossibility.*

Proof Suppose a code achieves average distortion nD.

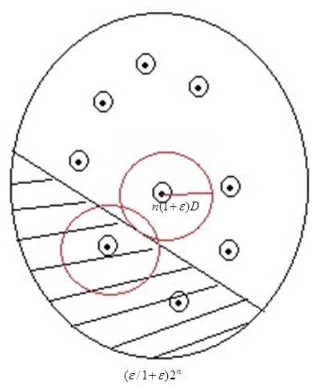
$$E(d(X, C) \leq nD$$

By *Markov's Inequality,*

$$
\begin{aligned}
P(d(X,C) > n(1+\epsilon)D) &\leq \frac{E[d(X,C)]}{n(1+\epsilon)D} \\
&\leq \frac{1}{1+\epsilon} \\
P(d(X,C) \leq n(1+\epsilon)D) &\geq \frac{\epsilon}{1+\epsilon}
\end{aligned}
$$

Hence, at least $\frac{\epsilon}{1+\epsilon}$ fraction of all points in $\{0,1\}^n$ has distortion less than or equal to $n(1+\epsilon)D$ for C.

Figure 5: Illustration of average distortion



If we draw spheres around the points in C, with radius $n(1+\epsilon)D$, this fraction of the points has to be covered.

$$
\begin{aligned}
|C| \sum_{i=0}^{n(1+\epsilon)D} \binom{n}{i} &\geq \frac{\epsilon}{1+\epsilon} 2^n \\
|C| &\geq \frac{\epsilon}{1+\epsilon} \frac{2^n}{\sum_{i=0}^{n(1+\epsilon)D} \binom{n}{i}} \\
\frac{\log |C|}{n} &\geq \frac{1}{n} \log \frac{\epsilon}{1+\epsilon} + 1 - \mathrm{h}((1+\epsilon)D)
\end{aligned}
$$

As $n \to \infty$,

$$
\frac{log|C|}{n} \geq 1 - \mathrm{h}((1+\epsilon)D)
$$

for any $\epsilon > 0$

$$
\begin{aligned}
R_{av}(D) &\geq 1 - \mathrm{h}((1+\epsilon)D) \\
R_{av}(D) &\geq 1 - \mathrm{h}(D)
\end{aligned}
$$

Therefore, $R_{av}(D) = 1 - \mathrm{h}(D)$ This function is called the *Rate Distortion function.* It can be defined for any source.