7 Group Testing and Error-Correcting Codes

7.1 Group Testing

During the Second World War the United States was interested in weeding out all syphilitic soldiers called up for the army. However, syphilis testing back then was expensive and testing every soldier individually would have been very costly and inefficient. A basic breakdown of a test is: 1) Draw sample from a given individual, 2) Perform required tests, and 3) Determine presence or absence of syphilis.

If there are n soldiers, this method of testing leads to n tests. If a significant portion of the soldiers were infected then the method of individual testing would be reasonable. The goal however, is to achieve effective testing in the more likely scenario where it does not make sense to test n (say $n = 100,000$) people to get k (say $k = 10$) positives.

Let's say that it was believed that there is only one soldier infected, then one could mix the samples of half of the soldiers and with a single test determined in which half the infected soldier is, proceeding with a binary search we could pinpoint the infected individual in $\log n$ tests. If instead of one, one believes that there are at most k infected people, then one could simply run k consecutive binary searches and detect all of the infected individuals in $k \log n$ tests. Which would still be potentially much less than n.

For this method to work one would need to observe the outcome of the previous tests before designing the next test, meaning that the samples have to be prepared adaptively. This is often not practical, if each test takes time to run, then it is much more efficient to run them in parallel (at the same time). This means that one has to non-adaptively design T tests (meaning subsets of the n individuals) from which it is possible to detect the infected individuals, provided there are at most k of them. Constructing these sets is the main problem in (Combinatorial) Group testing, introduced by Robert Dorfman [Dor43] with essentially the motivation described above.[31]

Let A_i be a subset of $[T] = \{1, \ldots, T\}$ that indicates the tests for which soldier i participates. Consider $\mathbb{A}$ the family of n such sets $\mathbb{A} = \{A_1, \ldots, A_n\}$. We say that $\mathbb{A}$ satisfies the k-disjunct property if no set in $\mathbb{A}$ is contained in the union of k other sets in $\mathbb{A}$. A test set designed in such a way will succeed at identifying the (at most k) infected individuals – the set of infected tests is also a subset of $[T]$ and it will be the union of the A_i's that correspond to the infected soldiers. If the set of infected tests contains a certain A_i then this can only be explained by the soldier i being infected (provided that there are at most k infected people).

Theorem 7.1 *Given n and k, there exists a family $\mathbb{A}$ satisfying the k-disjunct property for a number of tests*

$$T = \mathcal{O}\left(k^2 \log n\right).$$

Proof. We will use the probabilistic method. We will show that, for $T = Ck^2 \log n$ (where C is a universal constant), by drawing the family $\mathbb{A}$ from a (well-chosen) distribution gives a $k-$disjunct family with positive probability, meaning that such a family must exist (otherwise the probability would be zero).

[31] in fact, our description for the motivation of Group Testing very much follows the description in [Dor43].

Let $0 \leq p \leq 1$ and let $\mathbb{A}$ be a collection of n random (independently drawn) subsets of $[T]$. The distribution for a random set A is such that each $t \in [T]$ belongs to A with probability p (and independently of the other elements).

Consider $k+1$ independent draws of this random variable, $A_0, \ldots, A_k$. The probability that A_0 is contained in the union of A_1 through A_k is given by

$$\Pr\left[A_0 \subseteq (A_1 \cup \cdots \cup A_k)\right] = \left(1 - p(1-p)^k\right)^T.$$

This is minimized for $p = \frac{1}{k+1}$. For this choice of p, we have

$$1 - p(1-p)^k = 1 - \frac{1}{k+1}\left(1 - \frac{1}{k+1}\right)^k$$

Given that there are n such sets, there are $(k+1)\binom{n}{k+1}$ different ways of picking a set and k others to test whether the first is contained in the union of the other k. Hence, using a union bound argument, the probability that $\mathbb{A}$ is k-disjunct can be bounded as

$$\Pr[k-\text{disjunct}] \geq 1 - (k+1)\binom{n}{k+1}\left(1 - \frac{1}{k+1}\left(1 - \frac{1}{k+1}\right)^k\right)^T.$$

In order to show that one of the elements in $\mathbb{A}$ is k-disjunct we show that this probability is strictly positive. That is equivalent to

$$\left(1 - \frac{1}{k+1}\left(1 - \frac{1}{k+1}\right)^k\right)^T \leq \frac{1}{(k+1)\binom{n}{k+1}}.$$

Note that $\left(1 - \frac{1}{k+1}\right)^k \to e^{-1}\frac{1}{1-\frac{1}{k+1}} = e^{-1}\frac{k+1}{k}$, as $k \to \infty$. Thus, we only need

$$T \geq \frac{\log\left((k+1)\binom{n}{k+1}\right)}{-\log\left(1 - \frac{1}{k+1}e^{-1}\frac{k+1}{k}\right)} = \frac{\log\left(k\binom{n}{k+1}\right)}{-\log\left(1-(ek)^{-1}\right)} = \mathcal{O}(k^2\log(n/k)),$$

where the last inequality uses the fact that $\log\left(\binom{n}{k+1}\right) = \mathcal{O}\left(k\log\left(\frac{n}{k}\right)\right)$ due to Stirling's formula and the Taylor expansion $-\log(1-x^{-1})^{-1} = \mathcal{O}(x)$ □

This argument simply shows the existence of a family satisfying the k-disjunt property. However, it is easy to see that by having T slightly larger one can ensure that the probability that the random family satisfies the desired property can be made very close to 1.

Remarkably, the existence proof presented here is actually very close to the best known lower bound.

Theorem 7.2 *Given n and k, if there exists a family $\mathbb{A}$ of subsets of $[T]$ satisfying the k-disjunct property, then*

$$T = \Omega\left(\frac{k^2\log n}{\log k}\right).$$

Proof.

Fix a u such that $0 < u < \frac{T}{2}$; later it will be fixed to $u := \left\lceil (T-k)/\binom{k-1}{2} \right\rceil$. We start by constructing a few auxiliary family of sets. Let

$$\mathbb{A}_0 = \{A \in \mathbb{A} : |A| < u\},$$

and let $\mathbb{A}_1 \subseteq \mathbb{A}$ denote the family of sets in $\mathbb{A}$ that contain their own unique u-subset,

$$\mathbb{A}_1 := \left\{A \in \mathbb{A} : \exists F \subseteq A : |F| = u \text{ and, for all other } A' \in \mathbb{A},\ F \not\subseteq A'\right\}.$$

We will procede by giving an upper bound to $\mathbb{A}_0 \cup \mathbb{A}_1$. For that, we will need a couple of auxiliary family of sets. Let $\mathbb{F}$ denote the family of sets F in the definition of $\mathbb{A}_1$. More precisely,

$$\mathbb{F} := \{F \in [T] : |F| = u \text{ and } \exists! A \in \mathbb{A} : F \subseteq A\}.$$

By construction $|\mathbb{A}_1| \leq |\mathbb{F}|$

Also, let $\mathbb{B}$ be the family of subsets of $[T]$ of size u that contain an element of $\mathbb{A}_0$,

$$\mathbb{B} = \{B \subseteq [T] : |B| = u \text{ and } \exists A \in \mathbb{A}_0 \text{ such that } A \subseteq B\}.$$

We now prove that $|\mathbb{A}_0| \leq |\mathbb{B}|$. Let $\mathbb{B}'$ denote the family of subsets of $[T]$ of size u that are not in $\mathbb{B}$,

$$\mathbb{B}' = \left\{B' \subseteq [T] : |B'| = u \text{ and } B' \notin \mathbb{B}\right\}.$$

By construction of $\mathbb{A}_0$ and $\mathbb{B}$, no set in $\mathbb{B}'$ contains a set in $\mathbb{A}_0$ nor does a set in $\mathbb{A}_0$ contain a set in $\mathbb{B}'$. Also, both $\mathbb{A}_0$ and $\mathbb{B}'$ are antichains (or Sperner family), meaning that no pair of sets in each family contains each other. This implies that $\mathbb{A}_0 \cup \mathbb{B}'$ is an antichain containing only sets with u or less elements. The Lubell-Yamamoto-Meshalkin inequality [Yam54] directly implies that (as long as $u < \frac{T}{2}$) the largest antichain whose sets contain at most u elements is the family of subsets of $[T]$ of size u. This means that

$$|\mathbb{A}_0| + \left|\mathbb{B}'\right| = \left|\mathbb{A}_0 \cup \mathbb{B}'\right| \leq \binom{T}{u} = \left|\mathbb{B} \cup \mathbb{B}'\right| = |\mathbb{B}| + \left|\mathbb{B}'\right|.$$

This implies that $|\mathbb{A}_0| \leq |\mathbb{B}|$.

Because $\mathbb{A}$ satisfies the k-disjunct property, no two sets in $\mathbb{A}$ can contain eachother. This implies that the families $\mathbb{B}$ and $\mathbb{F}$ of sets of size u are disjoint which implies that

$$|\mathbb{A}_0 \cup \mathbb{A}_1| = |\mathbb{A}_0| + |\mathbb{A}_1| \leq |\mathbb{B}| + |\mathbb{F}| \leq \binom{T}{u}.$$

Let $\mathbb{A}_2 := \mathbb{A} \setminus (\mathbb{A}_0 \cup \mathbb{A}_1)$. We want to show that if $A \in \mathbb{A}_2$ and $A_1, \ldots, A_j \in \mathbb{A}$ we have

$$\left|A \setminus \bigcup_{i=1}^{j} A_i\right| > u(k-j). \tag{64}$$

This is readily shown by noting that if (64) did not hold then one could find $B_{j+1}, \ldots, B_k$ subsets of A of size t such that $A \setminus \bigcup_{i=1}^{j} A_i \subseteq \bigcup_{i=j+1}^{k} B_i$. Since A as no unique subsets of size t there must exist

$A_{j+1}, \ldots, A_k \in \mathbb{A}$ such that $B_i \subseteq A_i$ for $i = j+1, \ldots, k$. This would imply that $A \subseteq \bigcup_{i=1}^{k} A_i$ which would contradict the k-disjunct property.

If $|\mathbb{A}_2| > k$ then we can take $A_0, A_1, \ldots, A_k$ distinct elements of $\mathbb{A}_2$. For this choice and any $j = 0, \ldots, k$

$$\left| A_j \setminus \bigcup_{0 \leq i < j} A_i \right| \geq 1 + u(k-j).$$

This means that

$$\left| \bigcup_{j=0}^{k} A_j \right| = \sum_{j=0,\ldots,k} \left| A_j \setminus \bigcup_{0 \leq i < j} A_i \right| \geq \sum_{j=0,\ldots,k} [1 + u(k-j)] = 1 + k + u\binom{k+1}{2}.$$

Since all sets in $\mathbb{A}$ are subsets of $[T]$ we must have $1 + k + u\binom{k+1}{2} \leq \left| \bigcup_{j=0}^{k} A_j \right| \leq T$. On the other hand, taking

$$u := \left\lceil (T-k) / \binom{k+1}{2} \right\rceil$$

gives a contradition (note that this choice of u is smaller than $\frac{T}{2}$ as long as $k > 2$). This implies that $|\mathbb{A}_2| \leq k$ which means that

$$n = |\mathbb{A}| = |\mathbb{A}_0| + |\mathbb{A}_1| + |\mathbb{A}_2| \leq k + \binom{T}{u} = k + \binom{T}{\left\lceil (T-k)/\binom{k+1}{2} \right\rceil}.$$

This means that

$$\log n \leq \log \left(k + \binom{T}{\left\lceil (T-k)/\binom{k+1}{2} \right\rceil} \right) = O\left(\frac{T}{k^2} \log k \right),$$

which concludes the proof of the theorem.

□

We essentially borrowed the proof of Theorem 7.2 from [Fur96]. We warn the reader however that the notation in [Fur96] is drasticly different than ours, T corresponds to the number of people and n to the number of tests.

There is another upper bound, incomparable to the one in Theorem 7.1 that is known.

Theorem 7.3 *Given n and k, there exists a family $\mathbb{A}$ satisfying the k-disjunct property for a number of tests*

$$T = \mathcal{O}\left(k^2 \left(\frac{\log n}{\log k} \right)^2 \right).$$

The proof of this Theorem uses ideas of Coding Theory (in particular Reed-Solomon codes) so we will defer it for next section, after a crash course on coding theory.

The following Corollary follows immediately.

Corollary 7.4 *Given n and k, there exists a family $\mathbb{A}$ satisfying the k-disjunct property for a number of tests*

$$T = \mathcal{O}\left(\frac{k^2 \log n}{\log k} \min\left\{\log k, \frac{\log n}{\log k}\right\}\right).$$

While the upper bound in Corollary 7.4 and the lower bound in Theorem 7.2 are quite close, there is still a gap. This gap was recently closed and Theorem 7.2 was shown to be optimal [DVPS14] (original I was not aware of this reference and closing this gap was posed as an open problem).

Remark 7.5 *We note that the lower bounds established in Theorem 7.2 are not an artifact of the requirement of the sets being k-disjunct. For the measurements taken in Group Testing to uniquely determine a group of k infected individuals it must be that the there are no two subfamilies of at most k sets in $\mathbb{A}$ that have the same union. If $\mathbb{A}$ is not $k-1$-disjunct then there exists a subfamily of $k-1$ sets that contains another set A, which implies that the union of that subfamily is the same as the union of the same subfamily together with A. This means that a measurement system that is able to uniquely determine a group of k infected individuals must be $k-1$-disjunct.*

7.2 Some Coding Theory and the proof of Theorem 7.3

In this section we (very) briefly introduce error-correcting codes and use Reed-Solomon codes to prove Theorem 7.3. We direct the reader to [GRS15] for more on the subject.

Lets say Alice wants to send a message to Bob but they can only communicate through a channel that erases or replaces some of the letters in Alice's message. If Alice and Bob are communicating with an alphabet Σ and can send messages with lenght N they can pre-decide a set of allowed messages (or codewords) such that even if a certain number of elements of the codeword gets erased or replaced there is no risk for the codeword sent to be confused with another codeword. The set C of codewords (which is a subset of Σ^N) is called the codebook and N is the blocklenght.

If every two codewords in the codebook differs in at least d coordinates, then there is no risk of confusion with either up to $d-1$ erasures or up to $\lfloor \frac{d-1}{2} \rfloor$ replacements. We will be interested in codebooks that are a subset of a finite field, meanign that we will take Σ to be $\mathbb{F}_q$ for q a prime power and C to be a linear subspace of $\mathbb{F}_q$.

The dimension of the code is given by

$$m = \log_q |C|,$$

and the rate of the code by

$$R = \frac{m}{N}.$$

Given two code words c_1, c_2 the Hamming distance $\Delta(c_1, c_2)$ is the number of entries where they differ. The distance of a code is defined as

$$d = \min_{c_1 \neq c_2 \in C} \Delta(c_1, c_2).$$

For linear codes, it is the same as the minimum weight

$$\omega(C) = \min_{c \in C \setminus \{0\}} \Delta(c).$$

We say that a linear code C is a $[N, m, d]_q$ code (where N is the blocklenght, m the dimension, d the distance, and $\mathbb{F}^q$ the alphabet.

One of the main goals of the theory of error-correcting codes is to understand the possible values of rates, distance, and q for which codes exist. We simply briefly mention a few of the bounds and refer the reader to [GRS15]. An important parameter is given by the entropy function:

$$H_q(x) = x\frac{\log(q-1)}{\log q} - x\frac{\log x}{\log q} - (1-x)\frac{\log(1-x)}{\log q}.$$

- Hamming bound follows essentially by noting that if a code has distance d then balls of radius $\lfloor \frac{d-1}{2} \rfloor$ centered at codewords cannot intersect. It says that

$$R \leq 1 - H_q\left(\frac{1}{2}\frac{d}{N}\right) + o(1)$$

- Another particularly simple bound is Singleton bound (it can be easily proven by noting that the first $n + d + 2$ of two codewords need to differ in at least 2 coordinates)

$$R \leq 1 - \frac{d}{N} + o(1).$$

There are probabilistic constructions of codes that, for any $\epsilon > 0$, satisfy

$$R \geq 1 - H_q\left(\frac{d}{N}\right) - \epsilon.$$

This means that R^* the best rate achievable satisties

$$R^* \geq 1 - H_q\left(\frac{d}{N}\right), \tag{65}$$

known as the GilbertVarshamov (GV) bound [Gil52, Var57]. Even for $q = 2$ (corresponding to binary codes) it is not known whether this bound is tight or not, nor are there deterministic constructions achieving this Rate. This motivates the following problem.

Open Problem 7.1 *1. Construct an explicit (deterministic) binary code ($q = 2$) satisfying the GV bound* (65).

2. Is the GV bound tight for binary codes ($q = 2$)?

7.2.1 Boolean Classification

A related problem is that of Boolean Classification [AABS15]. Let us restrict our attention to In error-correcting codes one wants to build a linear codebook that does not contain a codeword with weight $\leq d - 1$. In other words, one wants a linear codebook C that does intersect $B(d-1) = \{x \in \{0,1\}^n : 0 < \Delta(x) \leq d-1\}$ the pinched Hamming ball of radius d (recall that $\Delta(d)$ is the Hamming weight of x, meaning the number of non-zero entries). In the Boolean Classification problem one is willing to confuse two codewords as long as they are sufficiently close (as this is likely to mean they are

in the same group, and so they are the same from the point of view of classification). The objective then becomes understanding what is the largest possible rate of a codebook that avoids an Annulus $A(a,b) = \{x \in \{0,1\}^n : a \leq \Delta(x) \leq b\}$. We refer the reader to [AABS15] for more details. Let us call that rate

$$R_A^*(a,b,n).$$

Note that $R_A^*(1,d-1,n)$ corresponds to the optimal rate for a binary error-correcting code, conjectured to be $1 - H_q\left(\frac{d}{N}\right)$ (The GV bound).

Open Problem 7.2 *It is conjectured in [AABS15] (Conjecture 3 in [AABS15]) that the optimal rate in this case is given by*

$$R_A^*(\alpha n, \beta n, n) = \alpha + (1-\alpha) R_A^*\left(1, \beta n, (1-\alpha)\right) + o(1),$$

where $o(1)$ goes to zero as n goes to infinity.

This is established in [AABS15] for $\beta \geq 2\alpha$ but open in general.

7.2.2 The proof of Theorem 7.3

Reed-Solomon codes[RS60] are $[n, m, n-m+1]_q$ codes, for $m \leq n \leq q$. They meet the Singleton bound, the drawback is that they have very large q ($q > n$). We'll use their existence to prove Theorem 7.3

Proof. [of Theorem 7.3]

We will construct a family $\mathbb{A}$ of sets achieving the upper bound in Theorem 7.3. We will do this by using a Reed-Solomon code $[q, m, q-m+1]_q$. This code has q^m codewords. To each codework c we will correspond a binary vector a of length q^2 where the i-th q-block of a is the indicator of the value of $c(i)$. This means that a is a vector with exactly q ones (and a total of q^2 entries)[32]. We construct the family $\mathbb{A}$ for $T = q^2$ and $n = q^m$ (meaning q^m subsets of $\left[q^2\right]$) by constructing, for each codeword c, the set of non-zero entries of the corresponding binary vector a.

These sets have the following properties,

$$\min_{j \in [n]} |A_j| = q,$$

and

$$\max_{j_1 \neq j_2 \in [n]} |A_{j_1} \cap A_{j_2}| = q - \min_{c_1 \neq c_2 \in C} \Delta(c_1, c_2) \leq q - (q-m+1) = m-1.$$

This readily implies that $\mathbb{A}$ is k-disjunct for

$$k = \left\lfloor \frac{q-1}{m-1} \right\rfloor,$$

because the union of $\left\lfloor \frac{q-1}{m-1} \right\rfloor$ sets can only contain $(m-1)\left\lfloor \frac{q-1}{m-1} \right\rfloor < q$ elements of another set.

Now we pick $q \approx 2k\frac{\log n}{\log k}$ (q has to be a prime but there is always a prime between this number and its double by Bertrand's postulate (see [**?**] for a particularly nice proof)). Then $m = \frac{\log n}{\log q}$ (it can be taken to be the ceiling of this quantity and then n gets updated accordingly by adding dummy sets).

[32] This is precisely the idea of code concatenation [GRS15]

This would give us a family (for large enough parameters) that is k-disjunct for

$$\begin{aligned}\left\lfloor \frac{q-1}{m-1} \right\rfloor &\geq \left\lfloor \frac{2k\frac{\log n}{\log k} - 1}{\frac{\log n}{\log q} + 1 - 1} \right\rfloor \\ &= \left\lfloor 2k\frac{\log q}{\log k} - \frac{\log q}{\log n} \right\rfloor \\ &\geq k.\end{aligned}$$

Noting that

$$T \approx \left(2k\frac{\log n}{\log k}\right)^2.$$

concludes the proof. □

7.3 In terms of linear Bernoulli algebra

We can describe the process above in terms of something similar to a sparse linear system. let 1_{A_i} be the $t - dimensional$ indicator vector of A_i, $1_{i:n}$ be the (unknown) $n-$dimensional vector of infected soldiers and $1_{t:T}$ the $T-$dimensional vector of infected (positive) tests. Then

$$\begin{bmatrix} | & & | \\ 1_{A_1} & \cdots & 1_{A_n} \\ | & & | \end{bmatrix} \otimes \begin{bmatrix} | \\ | \\ 1_{i:n} \\ | \\ | \end{bmatrix} = \begin{bmatrix} | \\ 1_{t:T} \\ | \end{bmatrix},$$

where $\otimes$ is matrix-vector multiplication in the Bernoulli algebra, basically the only thing that is different from the standard matrix-vector multiplications is that the addition operation is replaced by binary "or", meaning $1 \oplus 1 = 1$.

This means that we are essentially solving a linear system (with this non-standard multiplication). Since the number of rows is $T = \mathcal{O}(k^2 \log(n/k))$ and the number or columns $n \gg T$ the system is underdetermined. Note that the unknown vector, $1_{i:n}$ has only k non-zero components, meaning it is $k-$sparse. Interestingly, despite the similarities with the setting of sparse recovery discussed in a previous lecture, in this case, $\tilde{\mathcal{O}}(k^2)$ measurements are needed, instead of $\tilde{\mathcal{O}}(k)$ as in the setting of Compressed Sensing.

7.3.1 Shannon Capacity

The goal Shannon Capacity is to measure the amount of information that can be sent through a noisy channel where some pairs of messages may be confused with eachother. Given a graph G (called the confusion graph) whose vertices correspond to messages and edges correspond to messages that may be confused with each other. A good example is the following: say one has a alphabet of five symbols $1, 2, 3, 4, 5$ and that each digit can be confused with the immediately before and after (and 1 and 5 can be confused with eachother). The confusion graph in this case is C_5, the cyclic graph

on 5 nodes. It is easy to see that one can at most send two messages of one digit each without confusion, this corresponds to the independence number of C_5, $\alpha(C_5) = 2$. The interesting question arises when asking how many different words of two digits can be sent, it is clear that one can send at least $\alpha(C_5)^2 = 4$ but the remarkable fact is that one can send 5 (for example: "11","23","35","54", or "42"). The confusion graph for the set of two digit words $C_5^{\oplus 2}$ can be described by a product of the original graph C_5 where for a graph G on n nodes $G^{\oplus 2}$ is a graph on n nodes where the vertices are indexed by pairs ij of vertices of G and

$$(ij, kl) \in E\left(G^{\oplus 2}\right)$$

if both a) $i = k$ or $i, k \in E$ and b) $j = l$ or $j, l \in E$ hold.

The above observation can then be written as $\alpha\left(C_5^{\oplus 2}\right) = 5$. This motivates the definition of Shannon Capacity [Sha56]

$$\theta_S(G) \sup_k \left(G^{\oplus k}\right)^{\frac{1}{k}}.$$

Lovasz, in a remarkable paper [Lov79], showed that $\theta_S(C_5) = \sqrt{5}$, but determining this quantity is an open problem for many graphs of interested [AL06], including C_7.

Open Problem 7.3 *What is the Shannon Capacity of the 7 cycle?*

7.3.2 The deletion channel

In many applications the erasures or errors suffered by the messages when sent through a channel are random, and not adversarial. There is a beautiful theory understanding the amount of information that can be sent by different types of noisy channels, we refer the reader to [CT] and references therein for more information.

A particularly challenging channel to understand is the deletion channel. The following open problem will envolve a particular version of it. Say we have to send a binary string "10010" through a deletion channel and the first and second bits get deleted, then the message receive would be "010" and the receiver would not know which bits were deleted. This is in contrast with the erasure channel where bits are erased but the receiver knows which bits are missing (in the case above the message received would be "??010"). We refer the reader to this survey on many of the interesting questions (and results) regarding the Deletion channel [Mit09].

A particularly interesting instance of the problem is the Trace Reconstruction problem, where the same message is sent multiple times and the goal of the receiver is to find exactly the original message sent from the many observed corrupted version of it. We will be interested in the following quantity: Draw a random binary string with n bits, suppose the channel has a deletion probability of $\frac{1}{2}$ for each bit (independently), define $\mathcal{D}\left(n; \frac{1}{2}\right)$ has the number of times the receiver needs to receive the message (with independent corruptions) so that she can decode the message exactly, with high probability. It is easy to see that $\mathcal{D}\left(n; \frac{1}{2}\right) \leq 2^n$, since roughly once in every 2^n times the whole message will go through the channel unharmed. It is possible to show (see [HMPW]) that $\mathcal{D}\left(n; \frac{1}{2}\right) \leq 2^{\sqrt{n}}$ but it is not known whether this bound is tight.

Open Problem 7.4 *1. What are the asymptotics of $\mathcal{D}\left(n; \frac{1}{2}\right)$?*

2. *An interesting aspect of the Deletion Channel is that different messages may have different difficulties of decoding. This motivates the following question: What are the two (distinct) binary sequences $x^{(2)}$ and $x^{(2)}$ that are more difficult to distinguish (let's say that the receiver knows that either $x^{(1)}$ or $x^{(2)}$ was sent but not which)?*

18.S096 Topics in Mathematics of Data Science
Fall 2015