NDMI011: Combinatorics and Graph Theory 1

Lecture #12
Error correcting codes

Irena Penev

1 A motivating example

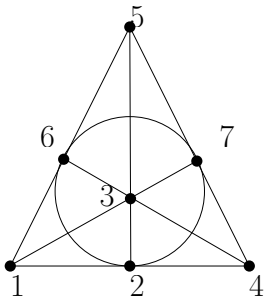
Let us suppose a sender wishes to send a message (say, a sequence of 1's and 0's) to a receiver. If the communication channel is unreliable or noisy, the message may get corrupted. For instance, the sender may send 1011, and the receiver may get 1001.[1] In this case, the receiver has no chance of spotting and fixing the error.

One way to address this problem might be to agree to triple each bit (i.e. each 1 or 0); so, instead of 1011, we would send 111000111111. Suppose just one error occurred, and the receiver received 111000110111. Because the receiver knows he was supposed to get a sequence of tripled 1's and 0's, he can confidently say that there was an error in the boxed triple: 111000$\boxed{110}$111. The receiver knows that the boxed triple should have been either 000 or 111, and the latter (i.e. 111) is more likely because it is more likely that only one error occurred than that two errors did. So, the receiver guesses that the message sent was 111000111111, which corresponds to 1011. On the other hand, if more than one error occurs in a triple corresponding to one bit, then the receiver will either fail to detect the error or will correct it incorrectly. For instance, if the receiver receives 111000100111, then he will incorrectly guess that the sender sent 111000000111, which corresponds to 1001.

Here is another way to address the same problem. Consider the Fano plane, represented below.[2]

[1] Here, errors are shown in red, to facilitate reading. However, the receiver does not see this: he simply receives a string of 1's and 0's, uncolored.

[2] We saw the Fano plane in Lecture 3. Here, points are relabeled (relative to what we had in Lecture 3), and the names of lines are omitted. We still have seven lines, represented by the six line segments and the circle. (Each line has exactly three points.)



We now form 16 row vectors of length seven as follows: we take all possible incidence vectors of lines of the Fano plane,[3] the incidence vectors of the complements of the lines of the Fano plane,[4] plus the vectors $(0,0,0,0,0,0,0)$ and $(1,1,1,1,1,1,1)$. Let $\mathcal{H}$ be the set of these 16 vectors. Now, these vectors have the following two properties:

- any two distinct vectors in $\mathcal{H}$ differ in at least three places/coordinates;
- for any vector $\mathbf{w}$ of 1's and 0's of length 7, there exists a unique vector $\mathbf{h} \in \mathcal{H}$ such that $\mathbf{w}$ and $\mathbf{h}$ differ in at most one place/coordinate.

This means that if a sender sends a vector from $\mathcal{H}$, and at most one error is made during transmission, the receiver can correctly guess which vector was sent.[5]

How do we use $\mathcal{H}$? First, note that there are precisely 16 strings of 1's and 0's of length four (indeed, these are simply the integers $0, 1, \ldots, 15$ written in binary code). So, we can set up a bijection π between the set of these 16 strings and the set $\mathcal{H}$. Now, suppose we wish to transmit a string of 1's and 0's of length $4n$, for some positive integer n. We divide such a string into n consecutive blocks of length four, and instead of sending these blocks, we send (consecutively) the n vectors from $\mathcal{H}$ that correspond to them. The advantage of this is that if, during transmission, at most one error is made in each vector, the receiver will be able to spot it and correct it, and then to read off (using π^{-1}) the sender's original $4n$-bit message.

Note that, if we use $\mathcal{H}$, then instead of sending $4n$ bits (the number of bits in our original message), we send $7n$ bits. If data is expensive, then this is clearly an improvement over tripling each bit (where we would send $3n$ bits for each n-bit message). We remark that $\mathcal{H}$ is a type of "Hamming code," sometimes called the *Hamming(7,4) code* (because the original 4 bits are converted into 7 bits).

[3]For example, the incidence vector of the line $\{1,2,4\}$ is $(1,1,0,1,0,0,0)$.

[4]For example, the incidence vector of the complement of the line $\{1,2,4\}$ is $(0,0,1,0,1,1,1)$.

[5]Indeed, the receiver simply chooses the unique vector from $\mathcal{H}$ that differs in at most one coordinate from the vector that the receiver received.

2 Basic notions

An *alphabet* is some finite set of symbols $\Sigma = \{s_0, \ldots, s_m\}$. Often, our alphabet is a finite field $\mathbb{F}_q$, where q is prime power;[6] particularly often, our alphabet is $\mathbb{F}_2 = \mathbb{Z}_2$, which is simply the binary code (and we can do addition and multiplication modulo 2). A *word* of length n is a string (or row vector) of length n of symbols from our alphabet; Σ^n is the set of all words of length n using symbols from the alphabet Σ. A *code* is a subset C of Σ^n.[7] Elements of the code are *codewords*. Given words $\mathbf{x} = x_1 \ldots x_n$ and $\mathbf{y} = y_1 \ldots y_n$ in Σ^n,[8] the *Hamming distance* between $\mathbf{x}$ and $\mathbf{y}$, denoted by $d(\mathbf{x}, \mathbf{y})$, is the number of places in which $\mathbf{x}$ and $\mathbf{y}$ differ, i.e. $d(\mathbf{x}, \mathbf{y}) = |\{i \in \{1, \ldots, n\} \mid x_i \neq y_i\}|$. It is straightforward to check that the Hamming distance $d(\cdot, \cdot)$ is a "metric" on Σ^n, that is, that is satisfies the following three properties:[9]

- $d(x, y) = 0 \Leftrightarrow x = y$;
- $d(x, y) = d(y, x)$;
- $d(x, z) \leq d(x, y) + d(y, z)$.

The inequality from the third bullet point is referred to as the *triangle inequality*.

Codes are used as follows. A sender would like to send a message to a receiver, and for this, he uses some code $C \subseteq \Sigma^n$, where Σ is some alphabet. There is a bijection π (known both to the sender and the receiver) between all possible messages and the code C. Now, the sender encodes his message (i.e. turns it into a codeword in a code via the bijection) and sends it to the receiver. The receiver receives this codeword, but possibly with some errors. (If the sender sends the codeword x and the receiver receives the word $\widetilde{x}$, then $d(x, \widetilde{x})$ is the number of errors created during transmission.) The receiver corrects the errors (this is possible if the number of errors is small enough, where "small enough" depends on the code used), and then recovers the original message using π^{-1}.

In general, there are two competing goals for codes. On the one hand, we wish to send as many different messages as possible, using as few bits as

[6] Recall that, for a positive integer q, there is a field of size q if and only if q is a prime power (i.e. $q = p^n$, where p is a prime number and n is a positive integer). Furthermore, all finite fields of the same size are isomorphic. If q a prime power, then $\mathbb{F}_q$ is the unique (up to isomorphism) field of size q. Note that if p is a prime number, then $\mathbb{F}_p = \mathbb{Z}_p$ (but this is only true if p is prime!).

[7] So, in the example from section 1, we have $\Sigma = \mathbb{F}_2$, $n = 12$ (the original message had four bits, and so after we tripled each bit, we got 12 bits), and $C = \{w_1 \ldots w_{12} \in \Sigma^{12} \mid w_{3k-2} = w_{3k-1} = w_{3k} \forall k \in \{1, 2, 3, 4\}\}$.

[8] Here, we treat a string or length n and a row vector of length n as interchangeable. We use one or the other depending on convenience.

[9] Check this!

possible. On the other hand, we wish to maximize the number of errors that we can successfully correct.

Now, suppose Σ is an alphabet of size at least two, and $C \subseteq \Sigma^n$ is a code containing at least two codewords. Here are some parameters for the code C:

- the codeword *length* is n;
- the *size* of the alphabet is $q = |\Sigma|$;
- the *dimension* of C is $|C|$, instead of which we often consider the logarithm $k = \log_q |C|$;
- the *minimum distance* in C is $d = \min\{d(x, y) \mid x, y \in C, x \neq y\}$.

A code with these parameters is an $(n, k, d)_q$-*code*. Note that if at most $\lfloor \frac{d-1}{2} \rfloor$ errors are made during the transmission of a codeword, then the receiver can correctly spot and correct the errors by selecting the (unique) codeword with minimum Hamming distance from the word that he received.

2.1 Some simple codes

The simplest code is the *total code* Σ^n, where Σ is an alphabet with $q = |\Sigma| \geq 2$ and n is a positive integer. The total code Σ^n is an $(n, n, 1)_q$ code.[10] If we use this code, we send little data, but we cannot correct even a single error!

The *repetition code* Rep_n of length n over the alphabet Σ (with $q = |\Sigma| \geq 2$) is the code $C = \{\underbrace{x \dots x}_{n} \mid x \in \Sigma\}$. It is an $(n, 1, n)_q$-code.[11] This code allows us to correct as many as $\lfloor \frac{n-1}{2} \rfloor$ errors, but it uses a lot of data.

Another simple example is the *parity code* C of length n (with $n \geq 2$) over the alphabet $\mathbb{F}_2$; it consists of all words of the form $w_1 \dots w_n$ with $w_1, \dots, w_n \in \mathbb{F}_2$ and $\sum_{i=1}^{n} w_i = 0$. Let us check that this is an $(n, n-1, 2)_2$-code. Obviously, the codeword length is n and the size of the alphabet is $q = 2$. Next, $|C| = 2^{n-1}$; this is because the first $n-1$ symbols of a codeword can be chosen arbitrarily (and there are 2^{n-1} ways of doing this), but the n-th symbol is uniquely determined by the previous $n-1$ ones (because the sum must be 0). So, $k = \log_q |C| = \log_2 2^{n-1} = n - 1$. Finally, it is obvious that two different words cannot have distance 1, for otherwise, the sum of symbols in one of them would be 1, a contradiction. On the other hand, both $\underbrace{0 \dots 0}_{n-2}00$ and $\underbrace{0 \dots 0}_{n-2}11$ are in our code, and the distance between them is 2. So, the minimum distance in our code is $d = 2$.

[10] Indeed, the size of the alphabet is q, the codeword length is n, and $k = \log_q |\Sigma^n| = \log_q q^n = n$. The minimum distance is $\Delta(\Sigma^n) = 1$ (indeed, recall that $|\Sigma| \geq 2$, and take two symbols $s_1, s_2 \in \Sigma$; then the distance between $s_1 \underbrace{s_1 \dots s_1}_{n-1}$ and $s_2 \underbrace{s_1 \dots s_1}_{n-1}$ is 1).

[11] Indeed, the size of the alphabet is q, and the codeword length is n. Further, $|C| = |\Sigma| = q$, and so $k = \log_q |C| = \log_q q = 1$. Finally, the distance between any two distinct words is precisely n.

2.2 The Hadamard code

Given vectors $\mathbf{a} = (a_1, \ldots, a_n)^T$ and $\mathbf{b} = (b_1, \ldots, b_n)^T$ in $\mathbb{R}^n$, the *standard inner product* (or *dot product*) of $\mathbf{a}$ and $\mathbf{b}$ is $\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i b_i$. Two vectors in $\mathbb{R}^n$ are *orthogonal* with respect to the dot product if their dot product is zero.

A *Hadamard matrix* of order n is an $n \times n$ matrix whose entries are all 1 or -1, and whose columns are pairwise orthogonal (with respect to the dot product). For example, the matrix

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

is Hadamard matrix of order 2. Furthermore, if H is an $n \times n$ Hadamard matrix, then

$$\begin{bmatrix} H & H \\ H & -H \end{bmatrix}$$

is a Hadamard matrix of order $2n$.[12]

Proposition 2.1. *Let H be a Hadamard matrix of order n. Then $HH^T = nI_n$.*[13] *Furthermore, H^T is also a Hadamard matrix of order n.*

Proof. Let us show that $H^T H = nI_n$. To simplify notation, set $H = \begin{bmatrix} \mathbf{h}_1 & \ldots & \mathbf{h}_n \end{bmatrix}$. For each $i \in \{1, \ldots, n\}$, the (i,i)-th entry of $H^T H$ is $\mathbf{h}_i \cdot \mathbf{h}_i$, which is equal to n because all entries of a Hadamard matrix are ± 1. On the other hand, for distinct $i, j \in \{1, \ldots, n\}$, the (i,j)-th entry of $H^T H$ is $\mathbf{h}_i \cdot \mathbf{h}_j$, which is equal to 0 since any two distinct columns of a Hadamard matrix are orthogonal. This proves that $H^T H = nI_n$.

Now, since $H^T H = nI_n$, we have that $(\frac{1}{n}H^T)H = I_n$; since $\frac{1}{n}H^T$ and H are square matrices whose product is the identity matrix, we know from Linear Algebra that $\frac{1}{n}H^T$ and H are both invertible and are each other's inverses. Consequently, $H(\frac{1}{n}H^T) = I_n$, and we deduce that $HH^T = nI_n$.

It remains to show that H^T is a Hadamard matrix. Since H is a Hadamard matrix of order n, we know that H^T is an $n \times n$ matrix, and that all entries of H^T are ± 1. It remains to show that the columns of H^T are pairwise orthogonal. To simplify notation, we set $H^T = \begin{bmatrix} \mathbf{a}_1 & \ldots & \mathbf{a}_n \end{bmatrix}$; note that this means that $\mathbf{a}_1^T, \ldots, \mathbf{a}_n^T$ are the rows of H (listed from top to bottom). Now, fix distinct $i, j \in \{1, \ldots, n\}$. Then the (i,j)-th entry of HH^T is $\mathbf{a}_i \cdot \mathbf{a}_j$. But we we already showed that $HH^T = nI_n$, and so (since $i \neq j$) the (i,j)-th entry of HH^T is 0. Thus, $\mathbf{a}_i \cdot \mathbf{a}_j = 0$. So, the columns of H^T are pairwise orthogonal, i.e. H^T is a Hadamard matrix. □

We now construct the Hadamard code as follows. Fix any Hadamard matrix H of order n. Then the Hadamard code associated with H consists

[12] Check this!

[13] As usual, I_n is the $n \times n$ identity matrix.

of all rows of H and all rows of $-H$. This code has $2n$ codewords.[14] It is easy to check that this is an $(n, 1 + \log_2 n, \frac{n}{2})_2$-code.[15]

3 The Singleton, Hamming, and Gilbert-Varshamov bounds

For positive integers n, d, q with $n \geq d$ and $q \geq 2$, let $A_q(n, d)$ be the maximum size of a code (i.e. the maximum possible number of codewords in a code) C with the following parameters:

- the size of the alphabet is q;
- the codeword length is n;
- the minimum distance is at least d.

The Singleton bound. *For all positive integers n, d, q such that $n \geq d$ and $q \geq 2$, we have that $A_q(n, d) \leq q^{n-d+1}$.*

Proof. We prove this by induction on n, keeping q fixed and allowing d to vary. More precisely, we fix positive integers n, d, q such that $n \geq d$ and $q \geq 2$, and we assume inductively that for all positive integers n', d' with $n' \geq d'$ and $n' < n$, we have that $A_q(n', d') \leq q^{n'-d'+1}$. We must show that $A_q(n, d) \leq q^{n-d+1}$.

Fix a code C over an alphabet Σ with $|\Sigma| = q$, and assume that the codeword length in C is n and that the minimum distance between codewords in C is at least d. We must show that $|C| \leq q^{n-d+1}$. If $d = 1$, then

$$|C| \quad \leq \quad |\Sigma^n| \quad = \quad q^n \quad = \quad q^{n-d+1},$$

and we are done. So from now on, we assume that $d \geq 2$.[16]

We now construct the code $\widetilde{C} \subseteq \Sigma^{n-d+1}$ as follows: $\widetilde{C}$ is the set of all words $w_1 \dots w_{n-d+1}$ in Σ^{n-d+1} for which there exist some $w_{n-d+2}, \dots, w_n \in \Sigma$ such that $w_1 \dots w_{n-d+1} w_{n-d+2} \dots w_n \in C$.[17] Let us check that $|\widetilde{C}| = |C|$. We define the function $f : C \to \widetilde{C}$ by setting $f(w_1 \dots w_{n-d+1} w_{n-d+2} \dots w_n) = w_1 \dots w_{n-d+1}$ for all $w_1 \dots w_{n-d+1} w_{n-d+2} \dots w_n \in C$; we will show that f is a bijection. By the construction of $\widetilde{C}$ and f, we have that f is onto $\widetilde{C}$. Now, fix codewords $\mathbf{w} = w_1 \dots w_n$ and $\mathbf{w}' = w'_1 \dots w'_n$ in C such that $f(\mathbf{w}) = f(\mathbf{w}')$; then $w_1 \dots w_{n-d+1} = w'_1 \dots w'_{n-d+1}$, and it follows that the

[14] For this, we must check that no two rows of H are the same, and that no row of H is equal to any row of $-H$. But this follows from the fact that, by Proposition 2.1, H^T is a Hadamard matrix (details?).

[15] Details?

[16] This implies that $n - d + 1 < n$; we will apply the induction hypothesis to $n - d + 1$.

[17] So, $\widetilde{C}$ is the set of all words that can be obtained by deleting the last $d - 1$ symbols of a codeword in C.

Hamming distance between $\mathbf{w}$ and $\mathbf{w}'$ is at most $d-1$.[18] Since the minimum distance in C is at least d, we conclude that $\mathbf{w} = \mathbf{w}'$, and it follows that f is one-to-one. Thus, $f : C \to \widetilde{C}$ is a bijection, and we deduce that $|\widetilde{C}| = |C|$.

Now, $\widetilde{C}$ is a code over Σ, with $|\Sigma| = q$, the length of codewords in $\widetilde{C}$ is $n-d+1 < n$,[19] and obviously, the minimum distance in $\widetilde{C}$ is at least 1. So, by the induction hypothesis, we have that

$$|\widetilde{C}| \;\leq\; A_q(n-d+1,1) \;\leq\; q^{(n-d+1)-1+1} \;=\; q^{n-d+1}.$$

Since $|\widetilde{C}| = |C|$, we deduce that $|C| \leq q^{n-d+1}$, which is what we needed to show. □

We now need some notation. Suppose n, t, q are positive integers and Σ is an alphabet of size q. For all $\mathbf{w} \in \Sigma^n$, we let $B_t^{\Sigma^n}(\mathbf{w})$ be the "combinatorial ball" of radius t around $\mathbf{w}$, i.e. $B_t^{\Sigma^n}(\mathbf{w})$ is the set of all words in Σ^n whose Hamming distance from $\mathbf{w}$ is at most t. When no confusion is possible, we write $B_t(\mathbf{w})$ instead of $B_t^{\Sigma^n}(\mathbf{w})$.

Proposition 3.1. *Let n, t, q be positive integers such that $n \geq t$ and $q \geq 2$, and let Σ be an alphabet of size q. Then $|B_t(\mathbf{w})| = \sum_{k=0}^{t} \binom{n}{k}(q-1)^k$ for all $\mathbf{w} \in \Sigma^n$.*

Proof. Fix a word $\mathbf{w} \in \Sigma^n$. We must show that the number of words in Σ^n at distance at most t from $\mathbf{w}$ is precisely $\sum_{k=0}^{t} \binom{n}{k}(q-1)^k$. Clearly, it suffices to show that for all $k \in \{0, \ldots, t\}$, the number of words in Σ^n at distance k from $\mathbf{w}$ is precisely $\binom{n}{k}(q-1)^k$. So, fix $k \in \{0, \ldots, t\}$. There are $\binom{n}{k}$ ways to choose the k places in which a word at Hamming distance k from $\mathbf{w}$ differs from $\mathbf{w}$. For each such choice, and for each of the k selected placed, we have $q-1$ ways of altering $\mathbf{w}$ in that place;[20] so, for all k places together, we get $(q-1)^k$ ways of altering $\mathbf{w}$. So, there are precisely $\binom{n}{k}(q-1)^k$ words in Σ^n at distance k from $\mathbf{w}$. □

The Hamming bound. *Let n, d, q be positive integers such that $n \geq d$ and $q \geq 2$, and let $t = \lfloor \frac{d-1}{2} \rfloor$. Then $A_q(n,d) \leq \frac{q^n}{\sum_{k=0}^{t} \binom{n}{k}(q-1)^k}$.*

Proof. Fix a code $C \subseteq \Sigma^n$, where Σ is an alphabet of size q, and assume that the minimum distance between codewords in C is at least d. We must show that $|C| \leq \frac{q^n}{\sum_{k=0}^{t} \binom{n}{k}(q-1)^k}$. Set $m = |C|$ and $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_m\}$. Since the minimum Hamming distance between codewords in C is at least d, and

[18] Indeed, $\mathbf{w}$ and $\mathbf{w}'$ are both of length n, and they coincide in their first $n-d+1$ places. So, they differ in at most $d-1$ places, i.e. their Hamming distance is at most $d-1$.

[19] We are using the fact that $d \geq 2$.

[20] Indeed, we can select any symbol from Σ, except the one that appears in the selected place in the word $\mathbf{w}$ itself. Since $|\Sigma| = q$, we have $q-1$ choices.

since $t = \lfloor \frac{d-1}{2} \rfloor$, we see that the combinatorial balls $B_t(\mathbf{c}_1), \ldots, B_t(\mathbf{c}_m)$ are pairwise disjoint.[21] We now compute:

$$\begin{aligned}
q^n &= |\Sigma^n| && \text{because } |\Sigma| = q \\
&\geq |\bigcup_{i=1}^{m} B_t(\mathbf{c}_i)| && \\
&= \sum_{i=1}^{m} |B_t(\mathbf{c}_i)| && \text{because } B_t(\mathbf{c}_1), \ldots, B_t(\mathbf{c}_m) \\
& && \text{are pairwise disjoint} \\
&= m \sum_{k=0}^{t} \tbinom{n}{k}(q-1)^k && \text{by Proposition 3.1} \\
&= |C| \sum_{k=0}^{t} \tbinom{n}{k}(q-1)^k && \text{because } m = |C|
\end{aligned}$$

This implies that $|C| \leq \frac{q^n}{\sum_{k=0}^{t} \binom{n}{k}(q-1)^k}$, which is what we needed to show. □

The Gilbert-Varshamov bound. *Let n, d, q be positive integers such that $n \geq d$ and $q \geq 2$. Then $A_q(n,d) \geq \frac{q^n}{\sum_{k=0}^{d-1} \binom{n}{k}(q-1)^k}$.*

Proof. Fix a code $C \subseteq \Sigma^n$, where Σ is some alphabet of size q, with minimum distance between codewords in C at least d, and with $|C| = A_q(n,d)$.[22] We must show that $|C| \geq \frac{q^n}{\sum_{k=0}^{d-1} \binom{n}{k}(q-1)^k}$.

Set $m = |C|$ and $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_m\}$.

Claim. $\Sigma^n = \bigcup_{i=1}^{m} B_{d-1}(\mathbf{c}_i)$.

Proof of the Claim. It is clear that $\bigcup_{i=1}^{m} B_{d-1}(\mathbf{c}_i) \subseteq \Sigma^n$. Suppose that $\bigcup_{i=1}^{m} B_{d-1}(\mathbf{c}_i) \subsetneq \Sigma^n$, and fix some $\mathbf{w} \in \Sigma^n \setminus \left(\bigcup_{i=1}^{m} B_{d-1}(\mathbf{c}_i) \right)$. Then $d(\mathbf{w}, \mathbf{c}_i) \geq d$ for all $i \in \{1, \ldots, m\}$. We now form a new code $\widetilde{C} := C \cup \{\mathbf{w}\}$; obviously, $\widetilde{C} \subseteq \Sigma^n$, with $|\Sigma| = q$, and by construction, the minimum distance in $\widetilde{C}$ is at least d. But now the fact that $|\widetilde{C}| = |C| + 1 = A_q(n,d) + 1$ contradicts the definition of $A_q(n,d)$. This proves the Claim. ■

[21] Note that we are using the triangle inequality for the Hamming distance here.

[22] Such a code C exists by the definition of $A_q(n,d)$.

We now compute:

$$\begin{aligned}
q^n &= |\Sigma^n| && \text{because } |\Sigma| = q \\
&= |\bigcup_{i=1}^{m} B_{d-1}(\mathbf{c}_i)| && \text{by the Claim} \\
&\leq \sum_{i=1}^{m} |B_{d-1}(\mathbf{c}_i)| \\
&= m \sum_{k=0}^{d-1} \tbinom{n}{k}(q-1)^k && \text{by Proposition 3.1} \\
&= |C| \sum_{k=0}^{d-1} \tbinom{n}{k}(q-1)^k && \text{because } m = |C|
\end{aligned}$$

It follows that $|C| \geq \frac{q^n}{\sum_{k=0}^{d-1} \binom{n}{k}(q-1)^k}$, which is what we needed to show. $\square$