Coding Theory and Linguistics

Matilde Marcolli

CS101: Mathematical and Computational Linguistics

Winter 2015

Error-correcting codes

- *Alphabet*: finite set A with $\#A = q \geq 2$.
- *Code*: subset $C \subset A^n$, *length* $n = n(C) \geq 1$.
- *Code words*: elements $x = (a_1, \ldots, a_n) \in C$.
- *Code language*: $\mathcal{W}_C = \cup_{m \geq 1} \mathcal{W}_{C,m}$, words $w = x_1, \ldots, x_m$; $x_i \in C$.
- ω-*language*: Λ_C, infinite words $w = x_1, \ldots, x_m, \ldots$; $x_i \in C$.
- Special case: $A = \mathbb{F}_q$, *linear codes*: $C \subset \mathbb{F}_q^n$ linear subspace
- in general: *unstructured codes*

- $k = k(C) := \log_q \#C$ and $[k] = [k(C)]$ integer part of $k(C)$

$$q^{[k]} \leq \#C = q^k < q^{[k]+1}$$

- *Hamming distance*: $x = (a_i)$ and $y = (b_i)$ in C

$$d((a_i), (b_i)) := \#\{i \in (1, \ldots, n) \mid a_i \neq b_i\}$$

- *Minimal distance* $d = d(C)$ of the code

$$d(C) := \min \{d(a, b) \mid a, b \in C, a \neq b\}$$

Code parameters

- $R = k/n =$ *transmission rate* of the code
- $\delta = d/n =$ *relative minimum distance* of the code

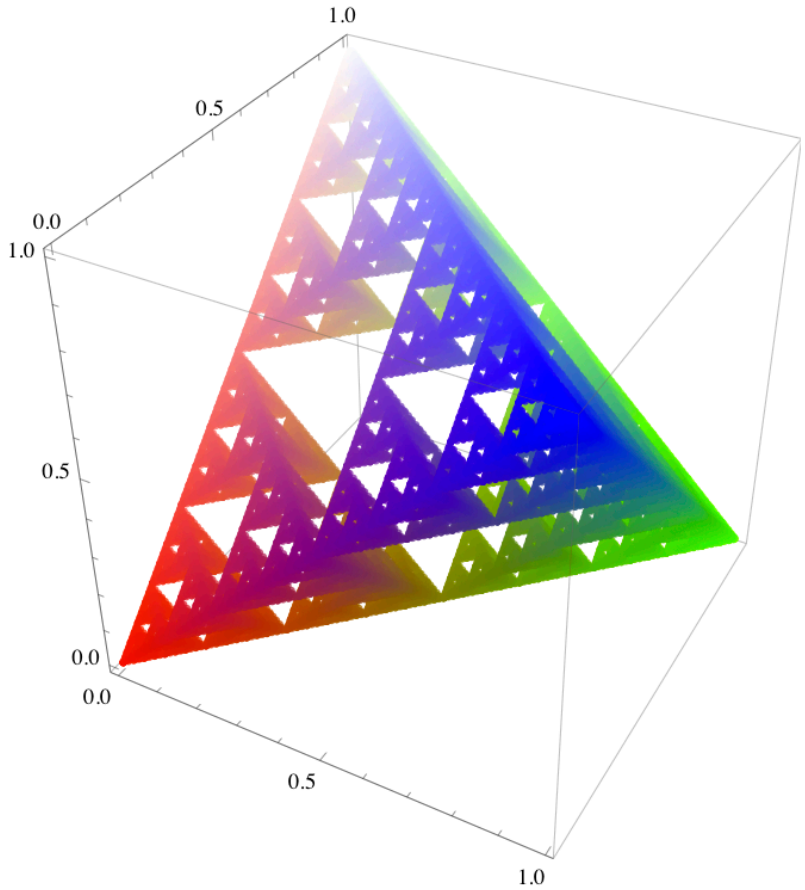
Small R: fewer code words, easier decoding, but longer encoding signal; small δ: too many code words close to received one, more difficult decoding

Language of a Code

- strings of code words $\mathcal{W}_C = \cup_{m \geq 1} \mathcal{W}_C^m$
- ω-language Λ_C of code C, infinite sequences of code words
- Λ_C fractal in $[0,1]^n$ hypercube
- Hausdorff dimension $\dim_H(\Lambda_C) = R(C)$ rate of code
- min distance $d(C)$: threshold dim, lower dim slices (all directions parallel to coord axes) of Λ_C empty or singletons; higher dim some sections of positive Hausdorff dim

Example: unstructured $[3, 2, 2]_2$ code

$C = \{(0, 0, 0), (0, 1, 1), (1, 0, 1), (1, 1, 0)\}$



M. Marcolli, C. Perez, *Codes as fractals and noncommutative spaces*, Mathematics in Computer Science, Vol.6 (2012) N.3, 199-215 (SURF 2010)

ω-language and complexity

- Yu.I. Manin, M. Marcolli, *Error-correcting codes and phase transitions*, Mathematics in Computer Science, Vol.5 (2011) 133–170.

• Entropy of language $\mathcal{W}_C$, generating function:

$$s_C(m) = \#\mathcal{W}_{C,m}, \qquad G_C(t) = \sum_m s_C(m) t^m$$

Entropy: $\mathcal{S}_C = -\log_q \rho(G_C(t))$ with $\rho =$ radius of convergence

• $G_C(q^{-s}) = Z_C(s)$ partition function is generating function of language structure functions

• Entropy of language is code rate $R = R(C)$

- complexity $\mathcal{K}_{T_{\mathcal{U}}}(w)$ of *words* in a language
- for infinite words in ω-language Λ_C complexity

$$\kappa(w) = \liminf_{w_n \to w} \frac{\mathcal{K}(w_n)}{\ell(w_n)}$$

- Levin (semi)measure

$$\kappa(w) = \liminf_{w_n \to w} \frac{-\log_q \mu_{\mathcal{U}}(w_n)}{\ell(w_n)}$$

universal enumerable semi-measure $\mu_{\mathcal{U}}$

- bounds uniform Bernoulli measure μ on Λ_C

$$\kappa(x) \leq \lim \frac{-\log_q \mu(w)}{\ell(w)} = R(C)$$

achieved on full measure subset

The space of code parameters:

- Optimization problem: increase R and δ... how good are codes?
- $Codes_q$ = set of all codes C on an alphabet $\#A = q$
- function $cp : Codes_q \to [0,1]^2 \cap \mathbb{Q}^2$ to code parameters
$cp : C \mapsto (R(C), \delta(C))$
- the function $C \mapsto (R(C), \delta(C))$ is a *total recursive map*
- *Multiplicity* of a code point (R, δ) is $\#cp^{-1}(R, \delta)$

  - M.A. Tsfasman, S.G. Vladut, *Algebraic-geometric codes*, Mathematics and its Applications (Soviet Series), Vol. 58, Kluwer Academic Publishers, 1991.

Spoiling operations on codes: C an $[n, k, d]_q$ code

- $C_1 := C *_i f \subset A^{n+1}$

$$(a_1, \ldots, a_{n+1}) \in C_1 \text{ iff } (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n) \in C\,,$$

and $a_i = f(a_1, \ldots, a_{i-1}, a_{i+1} \ldots, a_n)$
C_1 an $[n+1, k, d]_q$ code (f constant function)

- $C_2 := C*_i \subset A^{n-1}$

$$(a_1, \ldots, , a_{i-1}, a_{i+1}, \ldots, a_n) \in C_2$$

$$\text{iff } \exists b \in A,\ (a_1, \ldots, a_{i-1}, b, a_{i+1}, \ldots, a_n) \in C$$

C_2 an $[n-1, k, d]_q$ code

- $C_3 := C(a, i) \subset C \subset A^n$

$$(a_1, \ldots, a_n) \in C_3 \text{ iff } a_i = a.$$

C_3 an $[n-1, k-1 \leq k' < k, d' \geq d]_q$ code

Asymptotic bound in the space of code parameters

- $V_q \subset [0,1]^2$: all code points $(R, \delta) = cp(C)$, $C \in Codes_q$
- U_q: set of limit points of V_q
- Isolated code points: $V_q \setminus (V_q \cap U_q)$

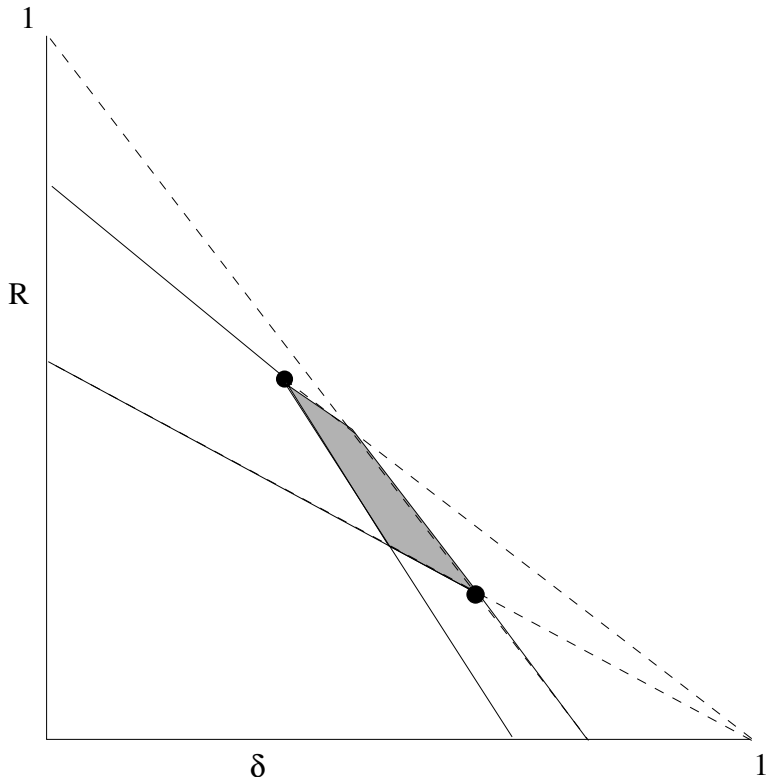
- Fact: existence of Asymptotic Bound:

U_q consists of all points below graph of a function

$$U_q = \{(R, \delta) \in [0,1]^2 \mid R \leq \alpha_q(\delta)\}$$

- Yu.I.Manin, *What is the maximum number of points on a curve over $\mathbb{F}_2$?* J. Fac. Sci. Tokyo, IA, Vol. 28 (1981), 715–720.

Method for establishing asymptotic bound: controlling quadrangles



$R = \alpha_q(\delta)$ continuous decreasing function with $\alpha_q(0) = 1$ and $\alpha_q(\delta) = 0$ for $\delta \in [\frac{q-1}{q}, 1]$; has inverse function on $[0, (q-1)/q]$; U_q union of all lower cones of points in $\Gamma_q = \{R = \alpha_q(\delta)\}$

Code points and multiplicities

- Set of code points of infinite multiplicity
$U_q \cap V_q = \{(R, \delta) \in [0,1]^2 \cap \mathbb{Q}^2 \mid R \leq \alpha_q(\delta)\}$ below the asymptotic bound

- Code points of finite multiplicity all above the asymptotic bound
$V_q \setminus (U_q \cap V_q)$ and isolated (open neighborhood containing (R, δ) as unique code point)

- again based on using spoiling operations on codes

  - Yu.I. Manin, *What is the maximum number of points on a curve over $\mathbb{F}_2$?* J. Fac. Sci. Tokyo, IA, Vol. 28 (1981), 715–720.
  - Yu.I. Manin, M. Marcolli, *Error-correcting codes and phase transitions*, Mathematics in Computer Science, Vol.5 (2011) 133–170.

Other bounds in the space of code parameters

- singleton bound: $R + \delta \leq 1$
- Gilbert–Varshamov line: $R = \frac{1}{2}(1 - H_q(\delta))$

$$H_q(\delta) = \delta \log_q(q-1) - \delta \log_q \delta - (1-\delta) \log_q(1-\delta)$$

q-ary entropy (for linear codes GV line $R = 1 - H_q(\delta)$)

Statistics of codes and the Gilbert–Varshamov bound

Known *statistical* approach to the GV bound: *random codes*

Shannon Random Code Ensemble: ω-language with alphabet A; uniform Bernoulli measure on Λ_A; choose code words of C as independent random variables in this measure

Volume estimate:

$$q^{(H_q(\delta)-o(1))n} \leq Vol_q(n, d = n\delta) = \sum_{j=0}^{d} \binom{n}{j} (q-1)^j \leq q^{H_q(\delta)n}$$

Gives probability of parameter δ for SRCE meets the GV bound with probability exponentially (in n) near 1: expectation

$$\mathbb{E} \sim \binom{q^k}{2} Vol_q(n, d) q^{-n} \sim q^{n(H_q(\delta)-1+2R)+o(n)}$$

... a priori no good statistical description of the asymptotic bound

Estimates on the asymptotic bound

- Plotkin bound:

$$\alpha_q(\delta) = 0, \quad \delta \geq \frac{q-1}{q}$$

- singleton bound:

$$\alpha_q(\delta) \leq 1 - \delta$$

- Hamming bound:

$$\alpha_q(\delta) \leq 1 - H_q(\frac{\delta}{2})$$

- Gilbert–Varshamov bound:

$$\alpha_q(\delta) \geq 1 - H_q(\delta)$$

Computability question

• Note: only the asymptotic bound marks a significant change of behavior of codes across the curve (isolated and finite multiplicity/accumulation points and infinite multiplicity)

• in this sense it is very different from all the other bounds in the space of code parameters

• .... but no explicit expression for the curve $R = \alpha_q(\delta)$

• ... is the function $R = \alpha_q(\delta)$ computable?

- Yu.I. Manin, *A computability challenge: asymptotic bounds and isolated error-correcting codes*, arXiv:1107.4246

The asymptotic bound and Kolmogorov complexity

• the asymptotoc bound $R = \alpha_q(\delta)$ becomes computable given an oracle that can list codes by increasing Kolmogorov complexity

• given such an oracle: iterative (algorithmic) procedure for constructing the asymptotic bound

• ... it is at worst as "non-computable" as Kolmogorov complexity

• asymptotic bound can be realized as phase transition curve of a statistical mechanical system based on Kolmogorov complexity

- Yu.I. Manin, M. Marcolli, *Kolmogorov complexity and the asymptotic bound for error-correcting codes*, Journal of Differential Geometry, Vol.97 (2014) 91–108

Structural numbering for codes

• structural numbering of X: computable bijection $\nu_X : \mathbb{N} \to X$, principal homogeneous space over group of total recursive permutations $\sigma : \mathbb{N} \to \mathbb{N}$

• construct an enumeration $\nu_X : \mathbb{N} \to X$ for $X = Codes_q$ the space of q-ary codes

• $A = \{0, \ldots, q-1\}$ ordered, A^n lexicographically; computable total order ν_X:
(i) if $n_1 < n_2$ all $C \subset A^{n_1}$ before all $C' \subset A^{n_2}$;
(ii) $k_1 < k_2$ all $[n, k_1, d]_q$-codes before $[n, k_2, d']_q$-codes;
(iii) fixed n and q^k: lexicographic order of code words, concatenated into single word $w(C)$ (determines code): order all the $w(C)$ lexicographically

• also fixed enumeration $\nu_Y : \mathbb{N} \to Y$ of rational points $Y = [0,1]^2 \cap \mathbb{Q}^2$

• Kolmogorov ordering:
$\mathbf{K}_{T_{\mathcal{U}}}(x)$ = order x by growing Kolmogorov complexity $\mathcal{K}_{T_{\mathcal{U}}}(x)$

$$c_1\,\mathcal{K}_{T_{\mathcal{U}}}(x) \leq \mathbf{K}_{T_{\mathcal{U}}}(x) \leq c_2\mathcal{K}_{T_{\mathcal{U}}}(x)$$

• Parameters map: $f : X \to Y$ with $X = Codes_q$, $Y = [0,1]^2 \cap \mathbb{Q}^2$ and $f = cp : C \mapsto (R(C), \delta(C))$ code parameters

• total recursive map $f = cp : Codes_q \to [0,1]^2 \cap \mathbb{Q}^2$

• total recursive function $f : X \to Y \Rightarrow \forall y \in f(X), \exists x \in X$, $y = f(x)$ and $\exists$ computable $c = c(f, \nu_X, \nu_Y) > 0$

$$\mathcal{K}_{T_{\mathcal{U}}}(x) \leq c \cdot \nu_Y^{-1}(y)$$

• meaning: when increasing Kolmogorov ordering of x also increasing structural ordering of $f(x)$

Algorithmic construction of the asymptotic bound = by successive approximations separate out subsets of $f(X) \subset Y$ that have finite and infinite multiplicity

Multiplicities:

- take $F(x) = (f(x), n(x))$ with

$$n(x) = \#\{x' \,|\, \nu_X^{-1}(x') \leq \nu_X^{-1}(x),\ f(x') = f(x)\}$$

total recursive function $\Rightarrow$ $F(X) \subset Y \times \mathbb{Z}^+$ enumerable

- $X_m := \{x \in X \,|\, n(x) = m\}$ and $Y_m := f(X_m) \subset Y$ enumerable
- multiplicities: $mult(y) := \#f^{-1}(y)$

$$Y_\infty \subset \cdots f(X_{m+1}) \subset f(X_m) \subset \cdots \subset f(X_1) = f(X)$$

$Y_\infty = \cap_m f(X_m)$ and $Y_{fin} = f(X) \setminus Y_\infty$

Complexity counting:

- for $x \in X_1$ and $y = f(x)$: complexity

$$K_{T_{\mathcal{U}}}(x) \leq c \cdot \nu_Y^{-1}(y)$$

- $y \in Y_\infty$ and $m \geq 1$: $\exists$ unique $x_m \in X$, $y = f(x_m)$, $n(x_m) = m$ and $c = c(f, u, v, \nu_X, \nu_Y) > 0$

$$K_{T_{\mathcal{U}}}(x_m) \leq c \cdot \nu_Y^{-1}(y)\, m\, \log(\nu_Y^{-1}(y) m)$$

- both of these complexity estimates follow from general formulae for Kolmogorov complexity under composition of total recursive functions
- again the meaning is that increasing Kolmogorov ordering of x_m also increases structural ordering of Y and multiplicities m

Oracle mediated recursive construction of Y_∞ and Y_{fin}

- Choose sequence (N_m, m), $m \geq 1$, $N_{m+1} > N_m$
- Step 1: $A_1 =$ list $y \in f(X)$ with $\nu_Y^{-1}(y) \leq N_1$; $B_1 = \emptyset$
- Step $m+1$: Given A_m and B_m, list $y \in f(X)$ with $\nu_Y^{-1}(y) \leq N_{m+1}$; $A_{m+1} =$ elements in this list for which $\exists\, x \in X$, $y = f(x)$, $n(x) = m+1$; $B_{m+1} =$ remaining elements in the list
- Note: at this step invoke oracle: produce list of $x \in X$ with explicitly bounded complexity

$$K_{T_\mathcal{U}}(x_m) \leq c \cdot \nu_Y^{-1}(y)\, m\, \log(\nu_Y^{-1}(y) m)$$

to ensure that this x with $n(x) = m+1$ appears in this list (if it exists)

- obtain $A_m \cup B_m \subset A_{m+1} \cup B_{m+1}$, union is all $f(X)$
- $B_m \subset B_{m+1}$ and $Y_{fin} = \cup_m B_m$
- $Y_\infty = \cup_{m \geq 1}(\cap_{n \geq 0} A_{m+n})$
- from A_m to A_{m+1} first add all new y with $N_m < \nu_Y^{-1}(y) \leq N_{m+1}$ then subtract those that have no more elements in the fiber $f^{-1}(y)$: these will be in B_{m+1}
- B_m is m-th step approximation of set of isolated code points
- A_m successively approximates the region of code-points below the asymptotic bound

Partition function for code complexity

$$Z(X,\beta) = \sum_{x \in X} \mathcal{K}_{T-\mathcal{U}}(x)^{-\beta}$$

weights elements by inverse complexity with $\beta =$ inverse temperature, thermodynamic parameter

- variant with prefix-free complexity $ZP(X,\beta) = \sum \mathcal{KP}(x)^{-\beta}$

- prefix-free complexity: intrinsic characterization by Levin in terms of maximality for all probabilities enumerable from below $p : X \to \mathbb{R}_+ \cup \{\infty\}$

$$\{(r,x) \,|\, r < p(x)\} \subset \mathbb{Q} \times X \quad \text{enumerable}$$

Convergence properties

- Kolmogorov complexity and Kolmogorov ordering

$$c_1 \mathbf{K}(x) \leq \mathcal{K}(x) \leq c_2 \mathbf{K}(x)$$

- convergence of $Z(X, \beta)$ controlled by series

$$\sum_{x \in X} \mathbf{K}_u(x)^{-\beta} = \sum_{n \geq 1} n^{-\beta} = \zeta(\beta)$$

- Partition function $Z(X, \beta)$ convergence for $\beta > 1$; phase transition at pole $\beta = 1$

Asymptotic bound as a phase transition

- $\delta = \beta_q(R)$ inverse of $\alpha_q(\delta)$ on $R \in [0, 1 - 1/q]$
- Fix $R \in \mathbb{Q} \cap (0, 1)$ and $\Delta \in \mathbb{Q} \cap (0, 1)$

$$Z(R, \Delta; \beta) = \sum_{C: R(C) = R; 1 - \Delta \leq \delta(C) \leq 1} K_u(C)^{-\beta + \delta(C) - 1}$$

- Phase transition at the asymptotic bound:
- $1 - \Delta > \beta_q(R)$: partition function $Z(R, \Delta; \beta)$ real analytic in β
- $1 - \Delta < \beta_q(R)$: partition function $Z(R, \Delta; \beta)$ real analytic for $\beta > \beta_q(R)$ and divergence for $\beta \to \beta_q(R)_+$

Application to Linguistics: Syntactic Parameters and Coding

- M. Marcolli, *Principles and Parameters: a coding theory perspective*, arXiv:1407.7169

• idea: assign a (binary or ternary) code to a family of languages and use position of code parameters with respect to the asymptotic bound to test relatedness

• N = number of syntactic parameters $\Pi = (\Pi_\ell)_{\ell=1}^N$
each Π_ℓ with values in $\mathbb{F}_2 = \{0, 1\}$
(or $\mathbb{F}_3 = \{-1, 0, +1\}$ if include parameters that are not set in certain languages)

• $\mathcal{F} = \{L_k\}_{k=1}^m$ a set of natural languages (language "family")

• Code $C = C(\mathcal{F})$ in $\mathbb{F}^N$ ($\mathbb{F}_2^N$ or $\mathbb{F}_3^N$) with m code words
$w_k = \Pi(L_k)$ string of syntactic parameters for the language L_k

Interpretation of Code Parameters

• $R = R(C)$ measures ratio between logarithmic size of number of languages in $\mathcal{F}$ and total number of parameters: how $\mathcal{F}$ distributed in the ambient $\mathbb{F}^N$

• $\delta = \delta(C)$ is the minimum, over all pairs of languages L_i, L_j in $\mathcal{F}$ of the relative Hamming distance

$$\delta(C(\mathcal{F})) = \min_{L_i \neq L_j \in \mathcal{F}} \delta_H(L_i, L_j)$$

$$\delta_H(L_i, L_j) = \frac{1}{N} \sum_{\ell=1}^{N} |\Pi_\ell(L_i) - \Pi_\ell(L_j)|$$

• code parameter δ used in Parameter Comparison Method for reconstruction of phylogenetic trees

Interpretation of Spoiling Operations

- first spoiling operation: effect of including one syntactic parameter in the list which is dependent on the other parameters
- second spoiling operation: forgetting one of the syntactic parameters
- third spoiling operation: forming subfamilies by considering languages that have a common value of one of the parameters

Parameters from Modularized Global Parameterization Method

- G. Longobardi, *Methods in parametric linguistics and cognitive history*, Linguistic Variation Yearbook, Vol.3 (2003) 101–138
- G. Longobardi, C. Guardiano, *Evidence for syntax as a signal of historical relatedness*, Lingua 119 (2009) 1679–1706.

• Determiner Phrase Module:
- syntactic parameters dealing with person, number, gender (1–6)
- parameters of definiteness (7–16)
- parameters of countability (17–24)
- genitive structure (25–31)
- adjectival and relative modification (32–14)
- position and movement of the head noun (42–50)
- demonstratives and other determiners (51–50 and 6–63)
- possessive pronouns (56–59)

Simple Example:

• group of three languages $\mathcal{F} = \{\ell_1, \ell_2, \ell_3\}$: Italian, Spanish, French using first group of 6 parameters

• code $C = C(\mathcal{F})$

ℓ_1	1	1	1	0	1	1
ℓ_2	1	1	1	1	1	1
ℓ_3	1	1	1	0	1	0

• code parameters: $(R = \log_2(3)/6 = 0.2642, \delta = 1/6)$

• code parameters satisfy $R < 1 - H_2(\delta)$: below the Gilbert–Varshamov curve

Spoiling operations in this example:

- first spoiling operation:

first two parameters same value 1, so
$C = C' \star_1 f_1 = (C'' \star_2 f_2) \star_1 f_1$ with f_1 and f_2 constant equal to 1 and $C'' \subset \mathbb{F}_2^4$ without first two letters

- second spoiling operation:

conversely, $C'' = C' \star_2$ and $C' = C \star_1$

- third spoiling operation:

$C(0,4) = \{\ell_1, \ell_3\}$ and $C(1,6) = \{\ell_2, \ell_3\}$

What if languages are not in the same historical family?

Example: $\mathcal{F} = \{L_1, L_2, L_3\}$: Arabic, Wolof, Basque

• excluding parameters that are not set, or are entailed by other parameters, for these languages: left with 25 parameters from original list (number 1–5, 7, 10, 20–21, 25, 27–29, 31–32, 34, 37, 42, 50–53, 55–57)

• code $C = C(\mathcal{F})$

L_1	1	1	1	1	1	1	0	1	0	1	0	1	0	1	1	1	1	1	1	0	1	0	0	0	0
L_2	1	1	1	0	0	1	1	0	1	0	1	0	0	1	0	1	1	0	0	1	1	1	1	1	1
L_3	1	1	0	1	0	0	1	0	0	0	1	1	1	0	1	1	0	1	1	1	1	1	1	0	0

• code parameters: $\delta = 0.52$ and $R > 0$ violates Plotkin bound

$\Rightarrow$ isolated code above the asymptotic bound

Asymptotic bound and language relatedness

- For binary syntactic parameters: a code $C = C(\mathcal{F})$ violates the Plotkin bound if any pair $L_i \neq L_j$ of languages in $\mathcal{F}$ has $\delta_H(L_i, L_j) \geq 1/2$
- L_i and L_j differ in at least half of the parameters: it would not happen in a group of historically related languages
- but what about codes above the asymptotic bound that do not violate the Plotkin bound?
- Expect: $C = C(\mathcal{F})$ above the asymptotic bound
$\Rightarrow \mathcal{F}$ not a historical language family
(quantitative test of historical relatedness)

Why the asymptotic bound?

• Why look at position with respect to asymptotic bound as a test of historical relatedness? because it is the only true "bound" in the space of code parameters across which behavior truly changes

• codes below the asymptotic bound are *easily deformable* (as long as number of syntactic parameters is large)

• if think of language evolution as a process of parameter change, expect languages that have evolved in the same family to determine codes in this zone of the space of code parameters

• codes $C = C(\mathcal{F})$ above the asymptotic bound should be a clear sign that list of languages in $\mathcal{F}$ do *not* belong to same historical family

• though there can be codes $C = C(\mathcal{F})$ below the asymptotic bound that also don't come from historically related languages: converse implication does not hold