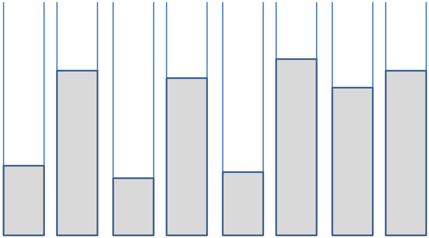
Minimum Pearson Distance Detection in the Presence of Unknown Slowly Varying Offset

Vitaly Skachek and Kees Schouhamer Immink

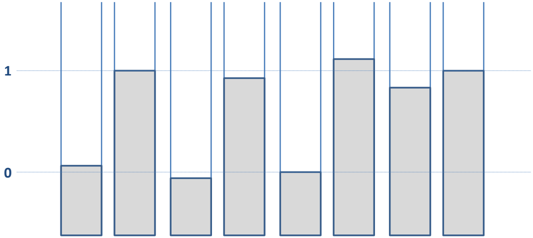
Barcelona, Spain

11 July 2016

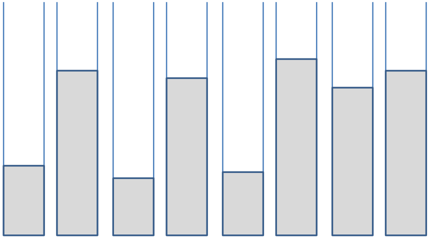
Data in NVM Memories



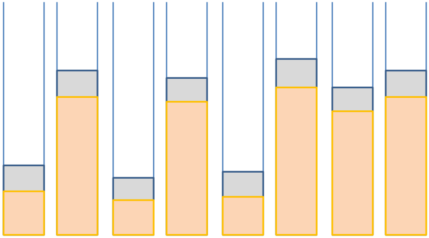
Data in NVM Memories



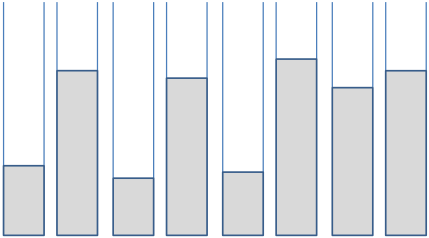
Data in NVM Memories



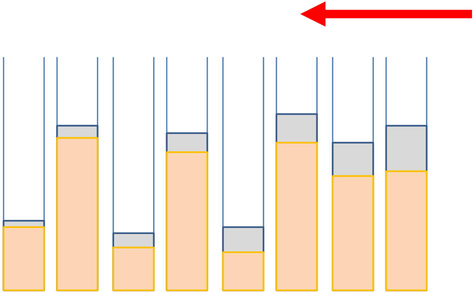
Uniform Leakage in NVM Memories



Data in NVM Memories



Slowly Varying Leakage in NVM Memories



Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.

Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.
- Codebook, S, of codewords $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{Q}^n$.

Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.
- Codebook, S, of codewords $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{Q}^n$.
- The read word $\mathbf{r} = (r_1, \ldots, r_n)$, $r_i \in \mathbb{R}$:

$$\mathbf{r} = a(\mathbf{x} + \boldsymbol{\nu}) + b\mathbf{1} + c\mathbf{s} \ ,$$

where $\mathbf{1} = (1, \ldots, 1)$, and $\mathbf{s} = (1, 2, \ldots, n)$.

Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.
- Codebook, S, of codewords $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{Q}^n$.
- The read word $\mathbf{r} = (r_1, \ldots, r_n)$, $r_i \in \mathbb{R}$:

$$\mathbf{r} = a(\mathbf{x} + \boldsymbol{\nu}) + b\mathbf{1} + c\mathbf{s} \; ,$$

  where $\mathbf{1} = (1, \ldots, 1)$, and $\mathbf{s} = (1, 2, \ldots, n)$.
- Unknown (positive) attenuation $a \in \mathbb{R}$, $a > 0$, and unknown varying offset, $b\mathbf{1} + c\mathbf{s}$, where b and $c \in \mathbb{R}$.

Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.
- Codebook, S, of codewords $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{Q}^n$.
- The read word $\mathbf{r} = (r_1, \ldots, r_n)$, $r_i \in \mathbb{R}$:

  $$\mathbf{r} = a(\mathbf{x} + \boldsymbol{\nu}) + b\mathbf{1} + c\mathbf{s} \; ,$$

  where $\mathbf{1} = (1, \ldots, 1)$, and $\mathbf{s} = (1, 2, \ldots, n)$.
- Unknown (positive) attenuation $a \in \mathbb{R}$, $a > 0$, and unknown varying offset, $b\mathbf{1} + c\mathbf{s}$, where b and $c \in \mathbb{R}$.
- Additive Gaussian noise $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$, $\nu_i \in \mathbb{R}$, with distribution $N(0, \sigma^2)$, where $\sigma^2 \in \mathbb{R}$ is the variance.

Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.
- Codebook, S, of codewords $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{Q}^n$.
- The read word $\mathbf{r} = (r_1, \ldots, r_n)$, $r_i \in \mathbb{R}$:

$$\mathbf{r} = a(\mathbf{x} + \boldsymbol{\nu}) + b\mathbf{1} + c\mathbf{s} \, ,$$

  where $\mathbf{1} = (1, \ldots, 1)$, and $\mathbf{s} = (1, 2, \ldots, n)$.
- Unknown (positive) attenuation $a \in \mathbb{R}$, $a > 0$, and unknown varying offset, $b\mathbf{1} + c\mathbf{s}$, where b and $c \in \mathbb{R}$.
- Additive Gaussian noise $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$, $\nu_i \in \mathbb{R}$, with distribution $N(0, \sigma^2)$, where $\sigma^2 \in \mathbb{R}$ is the variance.

K.A.S. Immink and J.H. Weber, "Minimum Pearson Distance Detection for Multi-Level Channels with Gain and/or Offset Mismatch," *IEEE Trans. Inform. Theory*, vol. IT-60, pp. 5966-5974, Oct. 2014.

Basic Settings

- Code alphabet $\mathcal{Q} = \{0, 1\}$.
- Codebook, S, of codewords $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{Q}^n$.
- The read word $\mathbf{r} = (r_1, \ldots, r_n)$, $r_i \in \mathbb{R}$:

$$\mathbf{r} = a(\mathbf{x} + \boldsymbol{\nu}) + b\mathbf{1} + c\mathbf{s} \ ,$$

  where $\mathbf{1} = (1, \ldots, 1)$, and $\mathbf{s} = (1, 2, \ldots, n)$.
- Unknown (positive) attenuation $a \in \mathbb{R}$, $a > 0$, and unknown varying offset, $b\mathbf{1} + c\mathbf{s}$, where b and $c \in \mathbb{R}$.
- Additive Gaussian noise $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$, $\nu_i \in \mathbb{R}$, with distribution $N(0, \sigma^2)$, where $\sigma^2 \in \mathbb{R}$ is the variance.

K.A.S. Immink and J.H. Weber, "Minimum Pearson Distance Detection for Multi-Level Channels with Gain and/or Offset Mismatch," *IEEE Trans. Inform. Theory*, vol. IT-60, pp. 5966-5974, Oct. 2014.

Minimum Euclidean Distance Detector

$$\mathbf{x}_{\mathrm{o}} = \arg\min_{\hat{\mathbf{x}} \in S} \delta_{\mathrm{e}}(\mathbf{r}, \hat{\mathbf{x}}) \ ,$$

where

$$\delta_{\mathrm{e}}(\mathbf{r}, \hat{\mathbf{x}}) = \sum_{i=1}^{n} (r_i - \hat{x}_i)^2 \ .$$

Minimum Euclidean Distance Detector

$$\mathbf{x}_o = \arg\min_{\hat{\mathbf{x}} \in S} \delta_e(\mathbf{r}, \hat{\mathbf{x}}) \ ,$$

where

$$\delta_e(\mathbf{r}, \hat{\mathbf{x}}) = \sum_{i=1}^{n} (r_i - \hat{x}_i)^2 \ .$$

We obtain:

$$\begin{aligned}\delta_e(\mathbf{r}, \hat{\mathbf{x}}) &= \sum_{i=1}^{n} (x'_i - \hat{x}_i)^2 + (b + ci)^2 \\ &+ 2b \sum_{i=1}^{n} x'_i + 2c \sum_{i=1}^{n} i x'_i - 2b \sum_{i=1}^{n} \hat{x}_i - 2c \sum_{i=1}^{n} i \hat{x}_i \ ,\end{aligned}$$

where $x'_i = a(x_i + \nu_i)$.

Pearson Distance

$$\delta(\mathbf{r}, \hat{\mathbf{x}}) = 1 - \rho_{\mathbf{r},\hat{\mathbf{x}}} \ ,$$

Pearson Distance

$$\delta(\mathbf{r}, \hat{\mathbf{x}}) = 1 - \rho_{\mathbf{r},\hat{\mathbf{x}}} \ ,$$

where

$$\rho_{\mathbf{r},\hat{\mathbf{x}}} = \frac{\sum_{i=1}^{n}(r_i - \overline{r})(\hat{x}_i - \overline{\hat{x}})}{\sigma_r \sigma_{\hat{x}}}$$

is *the Pearson correlation coefficient*,

Pearson Distance

$$\delta(\mathbf{r}, \hat{\mathbf{x}}) = 1 - \rho_{\mathbf{r},\hat{\mathbf{x}}} \ ,$$

where

$$\rho_{\mathbf{r},\hat{\mathbf{x}}} = \frac{\sum_{i=1}^{n} (r_i - \overline{r})(\hat{x}_i - \overline{\hat{x}})}{\sigma_r \sigma_{\hat{x}}}$$

is *the Pearson correlation coefficient*,

$$\overline{\hat{x}} = \frac{1}{n} \sum_{i=1}^{n} \hat{x}_i$$

is the *average symbol value* of $\hat{\mathbf{x}}$, and

$$\sigma_{\hat{x}}^2 = \sum_{i=1}^{n} (\hat{x}_i - \overline{\hat{x}})^2$$

is the (unnormalized) *symbol value variance* of $\hat{\mathbf{x}}$.

Minimum Pearson Distance Detector

$$\mathbf{x}_o = \arg\min_{\hat{\mathbf{x}} \in S} \delta(\mathbf{r}, \hat{\mathbf{x}}) \ .$$

Minimum Pearson Distance Detector

$$\mathbf{x}_o = \arg\min_{\hat{\mathbf{x}} \in S} \delta(\mathbf{r}, \hat{\mathbf{x}}) \; .$$

We obtain:

$$\begin{aligned} \delta(\mathbf{r}, \hat{\mathbf{x}}) &= 1 - \frac{1}{\sigma_r \sigma_{\hat{x}}} \sum_{i=1}^{n} x'_i (\hat{x}_i - \overline{\hat{x}}) \\ &- \frac{1}{\sigma_r \sigma_{\hat{x}}} \sum_{i=1}^{n} (b' + ci)(\hat{x}_i - \overline{\hat{x}}) \; , \end{aligned}$$

where $x'_i = a(x_i + \nu_i)$ and $b' = b - \overline{r}$.

Minimization of Pearson Distance

The relevant $(b, c, \hat{\mathbf{x}})$-dependent term of $\delta(\mathbf{r}, \hat{\mathbf{x}})$ equals

$$\sum_{i=1}^{n}(b' + ci)(\hat{x}_i - \overline{\hat{x}}) = b' \sum_{i=1}^{n}(\hat{x}_i - \overline{\hat{x}}) + c \sum_{i=1}^{n} i(\hat{x}_i - \overline{\hat{x}}) \; .$$

Minimization of Pearson Distance

The relevant $(b, c, \hat{\mathbf{x}})$-dependent term of $\delta(\mathbf{r}, \hat{\mathbf{x}})$ equals

$$\sum_{i=1}^{n}(b' + ci)(\hat{x}_i - \overline{\hat{x}}) = b' \sum_{i=1}^{n}(\hat{x}_i - \overline{\hat{x}}) + c \sum_{i=1}^{n} i(\hat{x}_i - \overline{\hat{x}}) \ .$$

The first term is zero since

$$\sum_{i=1}^{n}(\hat{x}_i - \overline{\hat{x}}) = \sum_{i=1}^{n} \hat{x}_i - n\overline{\hat{x}} = 0 \ .$$

Minimization of Pearson Distance

The relevant $(b, c, \hat{\mathbf{x}})$-dependent term of $\delta(\mathbf{r}, \hat{\mathbf{x}})$ equals

$$\sum_{i=1}^{n}(b' + ci)(\hat{x}_i - \overline{\hat{x}}) = b' \sum_{i=1}^{n}(\hat{x}_i - \overline{\hat{x}}) + c \sum_{i=1}^{n} i(\hat{x}_i - \overline{\hat{x}}) .$$

The first term is zero since

$$\sum_{i=1}^{n}(\hat{x}_i - \overline{\hat{x}}) = \sum_{i=1}^{n} \hat{x}_i - n\overline{\hat{x}} = 0 .$$

The second term is zero if all codewords, $\hat{\mathbf{x}} \in S$, satisfy

$$\sum_{i=1}^{n} i\hat{x}_i = \overline{\hat{x}} \sum_{i=1}^{n} i = \frac{1}{2}n(n+1)\overline{\hat{x}} .$$

Minimization of Pearson Distance

Principal Condition

$$2\sum_{i=1}^{n} i\hat{x}_i = (n+1)\sum_{i=1}^{n} \hat{x}_i \ .$$

Minimization of Pearson Distance

Principal Condition

$$2\sum_{i=1}^{n} i\hat{x}_i = (n+1)\sum_{i=1}^{n} \hat{x}_i \ .$$

The remaining term is

$$1 - \frac{1}{\sigma_r \sigma_{\hat{x}}} \sum_{i=1}^{n} x_i'(\hat{x}_i - \bar{\hat{x}}) \ ,$$

and it is independent of a, b, and c.

Minimization of Pearson Distance

Principal Condition

$$2\sum_{i=1}^{n} i\hat{x}_i = (n+1)\sum_{i=1}^{n}\hat{x}_i \ .$$

The remaining term is

$$1 - \frac{1}{\sigma_r \sigma_{\hat{x}}}\sum_{i=1}^{n} x'_i(\hat{x}_i - \bar{\hat{x}}) \ ,$$

and it is independent of a, b, and c.

Conclusion

Minimum Pearson distance detector is (a, b, c)-immune.

Properties of Codewords

Principal Condition Rewritten

$$\sum_{i=1}^{n} \left(i - \frac{n+1}{2} \right) \hat{x}_i = 0 \; .$$

Properties of Codewords

Principal Condition Rewritten

$$\sum_{i=1}^{n} \left(i - \frac{n+1}{2} \right) \hat{x}_i = 0 \ .$$

Properties

- The inverse of a codeword is a codeword.

Properties of Codewords

Principal Condition Rewritten

$$\sum_{i=1}^{n} \left(i - \frac{n+1}{2} \right) \hat{x}_i = 0 \ .$$

Properties

- The inverse of a codeword is a codeword.
- The reverse of a codeword is a codeword

Properties of Codewords

Principal Condition Rewritten

$$\sum_{i=1}^{n} \left(i - \frac{n+1}{2} \right) \hat{x}_i = 0 \; .$$

Properties

- The inverse of a codeword is a codeword.
- The reverse of a codeword is a codeword
- Let n is odd, and $\mathbf{x} \in S$. Assume that $\tilde{\mathbf{x}}$ agrees with $\mathbf{x}$ on all $\tilde{x}_i$, $i \neq (n+1)/2$, and $\tilde{x}_{(n+1)/2} = 1 - \hat{x}_{(n+1)/2}$. Then, $\tilde{\mathbf{x}} \in S$. The minimum distance of S equals unity.

Properties of Codewords

Principal Condition Rewritten

$$\sum_{i=1}^{n} \left(i - \frac{n+1}{2} \right) \hat{x}_i = 0 \, .$$

Properties

- The inverse of a codeword is a codeword.
- The reverse of a codeword is a codeword
- Let n is odd, and $\mathbf{x} \in S$. Assume that $\tilde{\mathbf{x}}$ agrees with $\mathbf{x}$ on all $\tilde{x}_i$, $i \neq (n+1)/2$, and $\tilde{x}_{(n+1)/2} = 1 - \hat{x}_{(n+1)/2}$. Then, $\tilde{\mathbf{x}} \in S$. The minimum distance of S equals unity.
- If n is even, any $\mathbf{x} \in S$ contains an even number of ones.

Counting using Generating Functions

Define a bi-variate generating function

$$h_n(x, y) = (1 + xy)(1 + xy^2) \ldots (1 + xy^n) .$$

Counting using Generating Functions

Define a bi-variate generating function

$$h_n(x, y) = (1 + xy)(1 + xy^2) \ldots (1 + xy^n) \ .$$

The coefficient of $x^{i_0} y^{j_0}$ equals the number of sequences that satisfy the conditions

$$\sum_{i=1}^{n} x_i = i_0 \text{ and } \sum_{i=1}^{n} i x_i = j_0 \ .$$

Counting using Generating Functions

Define a bi-variate generating function

$$h_n(x, y) = (1 + xy)(1 + xy^2) \dots (1 + xy^n) \; .$$

The coefficient of $x^{i_0} y^{j_0}$ equals the number of sequences that satisfy the conditions

$$\sum_{i=1}^{n} x_i = i_0 \text{ and } \sum_{i=1}^{n} i x_i = j_0 \; .$$

- The number $N_{\mathrm{dc}^2}(n)$ of dc^2-balanced length-n codewords is given by the coefficient of $x^{n/2} y^{\frac{n(n+1)}{4}}$.

Counting using Generating Functions

Define a bi-variate generating function

$$h_n(x, y) = (1 + xy)(1 + xy^2) \dots (1 + xy^n) .$$

The coefficient of $x^{i_0} y^{j_0}$ equals the number of sequences that satisfy the conditions

$$\sum_{i=1}^{n} x_i = i_0 \text{ and } \sum_{i=1}^{n} i x_i = j_0 .$$

- The number $N_{\mathrm{dc}^2}(n)$ of dc^2-balanced length-n codewords is given by the coefficient of $x^{n/2} y^{\frac{n(n+1)}{4}}$.
- The number $N(n)$ of desired length-n codewords is given by the sum of the coefficients of $x^i y^{\frac{i(n+1)}{2}}$, for $0 \leq i \leq n$.

Counting using Generating Functions (cont.)

Denote by $C_m(i,j)$ the coefficient of $x^i y^j$ in $h_m(x,y)$.

Counting using Generating Functions (cont.)

Denote by $C_m(i,j)$ the coefficient of $x^i y^j$ in $h_m(x,y)$.

Recursive Relation

For $m = 1, \ldots, n$, $i = 0, \ldots, m$, and $j = 0, \ldots, m(m+1)/2$,

$$C_m(i,j) = C_{m-1}(i,j) + C_{m-1}(i-1, j-m) \ ,$$

initial conditions $C_0(0,0) = 1$ and $C_0(i,j) = 0$ for any $(i,j) \neq (0,0)$.

Computational Results

Table : Size of codebook, $N(n)$, and $N_{dc^2}(n)$.

n	$N(n)$	$N_{dc^2}(n)$
4	4	2
5	8	0
6	8	0
7	20	0
8	18	8
9	52	0
10	48	0
11	152	0
12	138	58

Asymptotical Analysis

Define stochastic variables

$$s = x_1 + x_2 + \ldots + x_n \text{ and } p = x_1 + 2x_2 + \ldots + nx_n \ ,$$

where x_i, $1 \leq i \leq n$, are i.i.d. binary random variables.

Asymptotical Analysis

Define stochastic variables

$$s = x_1 + x_2 + \ldots + x_n \text{ and } p = x_1 + 2x_2 + \ldots + nx_n \ ,$$

where x_i, $1 \leq i \leq n$, are i.i.d. binary random variables.

$$E[x_i^2] = E[x_i] = 1/2 \quad \text{and} \quad E[x_i x_j] = 1/4 \ .$$

Asymptotical Analysis

Define stochastic variables

$$s = x_1 + x_2 + \ldots + x_n \text{ and } p = x_1 + 2x_2 + \ldots + nx_n ,$$

where x_i, $1 \leq i \leq n$, are i.i.d. binary random variables.

$$E[x_i^2] = E[x_i] = 1/2 \quad \text{and} \quad E[x_i x_j] = 1/4 .$$

If n is large, by the central limit theorem, the number of n-sequences, denoted by $\varphi(s, p)$, is given by

$$\varphi(s, p) \approx \frac{2^n}{2\pi\sigma_s\sigma_p\sqrt{1-\rho^2}} \cdot e^{-\frac{f(s,p)}{2(1-\rho^2)}} ,$$

where

$$f(s, p) = \left(\frac{s - \mu_s}{\sigma_s}\right)^2 + \left(\frac{p - \mu_p}{\sigma_p}\right)^2 - \frac{2\rho(s - \mu_s)(p - \mu_p)}{\sigma_s\sigma_p} .$$

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.
- σ_s^2 and σ_p^2 are the variance of s and p, respectively.

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.
- σ_s^2 and σ_p^2 are the variance of s and p, respectively.
- ρ is the linear correlation between s and p.

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.
- σ_s^2 and σ_p^2 are the variance of s and p, respectively.
- ρ is the linear correlation between s and p.

$$\mu_s = \frac{n}{2} \, , \qquad \sigma_s^2 = \frac{n}{4} \, ,$$

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.
- σ_s^2 and σ_p^2 are the variance of s and p, respectively.
- ρ is the linear correlation between s and p.

$$\mu_s = \tfrac{n}{2}\,, \qquad \sigma_s^2 = \tfrac{n}{4}\,,$$

$$\mu_p = \frac{n(n+1)}{4}\,, \qquad \sigma_p^2 = \frac{n(n+1)(2n+1)}{24}\,,$$

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.
- σ_s^2 and σ_p^2 are the variance of s and p, respectively.
- ρ is the linear correlation between s and p.

$$\mu_s = \frac{n}{2}\,, \qquad \sigma_s^2 = \frac{n}{4}\,,$$

$$\mu_p = \frac{n(n+1)}{4}\,, \qquad \sigma_p^2 = \frac{n(n+1)(2n+1)}{24}\,,$$

$$\rho^2 = \frac{3}{2} \cdot \frac{n+1}{2n+1}\,.$$

Asymptotical Analysis (cont.)

- μ_s and μ_p are the average of s and p, respectively.
- σ_s^2 and σ_p^2 are the variance of s and p, respectively.
- ρ is the linear correlation between s and p.

$$\mu_s = \tfrac{n}{2}\,, \qquad \sigma_s^2 = \tfrac{n}{4}\,,$$

$$\mu_p = \tfrac{n(n+1)}{4}\,, \qquad \sigma_p^2 = \tfrac{n(n+1)(2n+1)}{24}\,,$$

$$\rho^2 = \tfrac{3}{2} \cdot \tfrac{n+1}{2n+1}\,.$$

The number of dc^2-balanced codewords is:

$$N_{\mathrm{dc}^2}(n) \approx \varphi(\mu_s, \mu_p) \approx \frac{2^n}{2\pi\sigma_s\sigma_p\sqrt{1-\rho^2}},$$

and therefore

$$r_{\mathrm{dc}^2}(n) \approx 2\log_2 n - \log_2 \frac{4\sqrt{3}}{\pi}\,.$$

Redundancy Estimate

$$N(n) \approx N_{\mathrm{dc}^2}(n) \cdot \sum_{\substack{s=0 \\ s(n+1) \mod 2 = 0}}^{n} e^{-\frac{f\left(s, \frac{(n+1)s}{2}\right)}{2(1-\rho^2)}} .$$

Redundancy Estimate

$$N(n) \approx N_{\mathrm{dc}^2}(n) \cdot \sum_{\substack{s=0 \\ s(n+1) \mod 2=0}}^{n} e^{-\frac{f\left(s, \frac{(n+1)s}{2}\right)}{2(1-\rho^2)}} .$$

For n odd, $N(n) \approx \frac{2^n}{n^{3/2}}\sqrt{\frac{24}{\pi}}$.

Redundancy Estimate

$$N(n) \approx N_{\mathrm{dc}^2}(n) \cdot \sum_{\substack{s=0 \\ s(n+1) \mod 2=0}}^{n} e^{-\frac{f\left(s, \frac{(n+1)s}{2}\right)}{2(1-\rho^2)}} .$$

For n odd, $$N(n) \approx \frac{2^n}{n^{3/2}} \sqrt{\frac{24}{\pi}} .$$

For n even, $$N(n) \approx \frac{2^n}{n^{3/2}} \sqrt{\frac{6}{\pi}} .$$

Redundancy Estimate

$$N(n) \approx N_{\mathrm{dc}^2}(n) \cdot \sum_{\substack{s=0 \\ s(n+1) \mod 2=0}}^{n} e^{-\frac{f\left(s, \frac{(n+1)s}{2}\right)}{2(1-\rho^2)}} .$$

$$\text{For } n \text{ odd,} \quad N(n) \approx \frac{2^n}{n^{3/2}}\sqrt{\frac{24}{\pi}} .$$

$$\text{For } n \text{ even,} \quad N(n) \approx \frac{2^n}{n^{3/2}}\sqrt{\frac{6}{\pi}} .$$

Redundancy Estimate

$$r(n) = n - \log_2 N(n) \approx \frac{3}{2}\log_2 n + \alpha ,$$

where $\alpha = -1.467...$ for n odd, and $\alpha = -0.467...$ for n even.

Thank you!

Thank you!