Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Números Muy Normales

Tesis de Licenciatura en Ciencias de la Computación

Lucas Puterman

Director: Verónica Becher
Codirector: Olivier Carton
Buenos Aires, 2019

NÚMEROS MUY NORMALES

En esta tesis nos proponemos estudiar la noción de supernormalidad definida por Zeev Rudnick hace unos años. Lo poco que se conoce sobre esta noción no está publicado. Benjamin Weiss del Einstein Institute of Math de Hebrew University dio el 16 de Junio de 2010 una una conferencia en el Institute for Advanced Study in Princeton titulada "Random-like behavior in deterministic systems" donde describe la noción de supernormalidad, a la que llama Poisson generic (ver https://video.ias.edu/pseudo2010/weiss). En este video Weiss afirma que la mayoría de los números reales son supernormales y que la supernormalidad es más fuerte que la noción clásica de normalidad, es decir, que si un número es supernormal, entonces es normal pero no al revés. También afirma que el ejemplo más famoso de número normal, el número de Champernowne, no es supernormal. Y deja abierto el problema de dar una construcción explícita de un número supernormal. En esta tesis nos proponemos dar la demostración completa de que el número binario de Champernowne no es supernormal.

Palabras claves: Normalidad, Supernormalidad, Champernowne, Poisson, Pseudoaleatoreidad.

VERY NORMAL NUMBERS

In this thesis we aim to study the notion of supernormality defined by Zeev Rudnick a few years ago. The few things known about supernormality is not published. Benjamin Weiss from the Einstein Institute of Math de Hebrew University gave on June 16th, 2010 a lecture on “Random-like behavior in deterministic systems” where the notion of supernormal sequences is described under the name of Poisson generic sequences. (See https://video.ias.edu/pseudo2010/weiss). In this lecture, Weiss claims that almost every real number is supernormal and that the notion of supernormality is stronger than the classical notion of normaility. Which means that if a number is supernormal then it is normal, but not the other way around. Weiss also states that the most famous example of a normal number, the Champernowne number, is not supernormal. And finally he leaves open the problem of giving an explicit construction of a supernormal number. In this thesis we give the complete proof that the binary Champernowne number is not supernormal.

Keywords: Normality, Supernormality, Champernowne, Poisson, Pseudorandomness.

CONTENTS

1. INTRODUCTION

1.1 Normality

1.2 Supernormality

Definition 1.2.1. Let $\lambda > 0$ be a real number. Let $k \in_{\geq 0}$.

$$a_k(\lambda) = \frac{e^{-\lambda}\lambda^k}{k!}$$

Let x be a sequence over a binary alphabet. For u, w words, $|w|_u$ is the number of occurrences of u in w. Let $A_{k,n}$ be the frequency of occurrence of words of length n that occur exactly k times in the first $\lfloor \lambda(2^n + n - 1) \rfloor$ symbols of x, or what is to say:

$$A_{k,n}(\lambda) = \frac{\#\{w : |w| = n, |x[1...\lfloor \lambda(2^n + n - 1)\rfloor]|_w = k\}}{2^n}$$

Definition 1.2.2. x is $\lambda-$supernormal if

$$\lim_{n\to\infty} A_{k,n}(\lambda) = a_k(\lambda)$$

for each integer $k \geq 0$.

Definition 1.2.3. x is supernormal if it is $\lambda-$supernormal $\forall \lambda \in \mathbb{R}$

2. SUPERNORMALITY IS STRONGER THAN NORMALITY

2.1 Supernormality implies normality (Proof by Olivier Carton)

Fact 2.1.1.

$$\sum_{k\geq 0} a_k(\lambda) = e^{-\lambda}\sum_{k\geq 0}\frac{\lambda^k}{k!} = e^{-\lambda}e^{\lambda} = 1$$

Fact 2.1.2.

$$\sum_{k\geq 0} ka_k(\lambda) = e^{-\lambda}\sum_{k\geq 1}\frac{\lambda^k}{(k-1)!} = \lambda e^{-\lambda}\sum_{k\geq 0}\frac{\lambda^k}{k!} = \lambda$$

Fact 2.1.3.

$$\sum_{k\geq 0} A_{k,n} = 1$$

Fact 2.1.4.

$$A_{k,n} = 0 \ \textit{if}\ k > \lambda 2^n$$

Let $\epsilon > 0$ be chosen.
Let k_0 be chosen such that

$$\frac{1}{\lambda}\sum_{k>k_0} ka_k(\lambda) < \frac{\epsilon}{2}$$

Let n_0 be chosen such that

$$\forall i, n \ s.t\ 0 \leq i \leq k_0 - 1 \wedge n > n_0 \wedge |A_{i,n}(\lambda) - a_i(\lambda)| \leq \frac{\epsilon\lambda}{2k_0(k_0+1)}$$

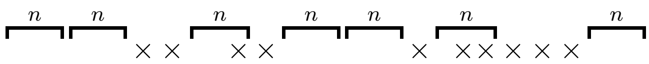
$$Rq? = \sum_{i=0}^{k_0-1} ka_k(\lambda) = \lambda - \sum_{k\geq k_0} ka_k(\lambda) \geq (\lambda - \frac{\epsilon}{2})$$

Consider the positions from 1 to $\lfloor \lambda 2^n \rfloor$ in x. A position is "blamed" if the word of length n starting at that position occurs more than k_0 times in $x\lfloor \lambda(2^n + n - 1)\rfloor$.

The number of non-blamed positions is

$$\begin{aligned}
\sum_{i=0}^{k-1} iA_{i,n}(\lambda)2^n &\geq 2^n\sum_{i=0}^{k-1} i(a_i(\lambda) - \frac{\epsilon\lambda}{2k_0(k_0+1)}) \\
&\geq 2^n\sum_{i=0}^{k-1} ia_i(\lambda) - \frac{2^n\epsilon\lambda}{2} \\
&\geq 2^n\lambda(1 - \frac{\epsilon}{2}) - \frac{2^n\epsilon\lambda}{2} \\
&= 2^n\lambda(1-\epsilon)
\end{aligned}$$

We cover the positions of $1 \dots \lambda 2^n$ by blocks of length such that blocks do not start at a blamed position. However a block may contain blamed positions.



The number of left positions is less than $\epsilon\lambda 2^n$. For a fixed word w, the number of bad blocks is o(number of possible blocks). Since each block can occur at most $k_0 - 1$ times then it is ok. Hot spot lemma is needed to conclude.

2.2 Normality does not imply supernormality

We aim to prove that normality does not imply supernormality. For this we prove that the Champernowne sequence, proven to be normal in [BC18], fails to be supernormal by proving that it is not 1-supernormal. The strategy followed to achieve this is similar to the ones used in [BCC19] and [PS19].

Let $X(n)$ be the concatenation of all words of length n over the alphabet with two simbols $\mathcal{A} = \{0, 1\}$ in lexicographic order. It is clear that $X(n)$ has length $n2^n$. Then, for example:

$$X(2) = 00\;01\;10\;11$$

Note that spaces were added for reading convenience.

Let the Champernowne sequence be the concatenation of $X(n)$ for $n = 1, 2, \dots$ Then the first symbols of the Champernowne sequence are:

$$\mathit{champ} = 0\;1\;00\;01\;10\;11\;000\;001\;010\;011\;100\;101\;110\;111\;0000\;0001\;\dots$$

By the definition of λ-supernormality, if we take $\lambda = 1$, then we need to prove that

$$\lim_{n\to\infty} A_{k,n}(1) = a_k(1)$$

$$\lim_{n\to\infty} \frac{\#\{w : |w| = n, |x[1...(2^n + n - 1)]|_w = k\}}{2^n} \neq \frac{e^{-1}}{k!}$$

First, let's see that given d, we define $k = 2^d$, if we take $n \geq d + k + 1$ then the whole block $X(k)$ covered in the first 2^n symbols of Champernowne.

Fact 2.2.1. *$\forall n, n \geq 2$ it follows that $\sum_{i=1}^{n} i2^i < 2^{n+log(n)+1}$*

Fact 2.2.2. *$X(k)$ accounts for half of the total amount of symbols in the first 2^{d+k+1} symbols of Chamernowne.*

Proof. $2^{d+k+1} = 2^{d+k}2$ and $X(k)$ has length $k2^k = 2^d 2^k = 2^{d+k}$ □

We prove that Champernowne is not supernormal by showing that the frequence of words of length $k+d+1$ that do not occur in the first 2^{k+d+1} symbols of the Champernowne sequence is higher that the expected frequency if it were supernormal. To accomplish this, we exhaustively look how many different words of length 2^{k+d+1} are there within $X(k)$ and give an upper bound for the amount of different words that can appear in the first 2^{k+d+1} symbols of the Champernowne sequence.

Now, let's take a look at what the words of length $k+d+1$ that occur in $X(k)$ look like. There are four different cases that can happen of how a word x is formed with elements from $X(k)$. In the following analysis, u, v and w are consecutive words of length k in $X(k)$:

- Case 1:

$$x = u_1u_2 \dots u_k \quad v_1v_2 \dots v_dv_{d+1}$$

  Which means it is the occurrence modulo 0 for a given word u of length k in $X(k)$ plus the remaining $d+1$ symbols which are taken from the next word.

- Case 2:

$$x = u_{k-d-1} \dots u_k \quad v_1v_2 \dots v_k$$

  Which is the case where the word of length $k+d+1$ is formed from the last $d+1$ symbols of a word and the whole k symbols of the next word.

- Case 3:

$$x = u_{n+1}u_{n+2} \dots u_k \quad v_1v_2 \dots v_{d+n+1}$$

  with $n \in \{1, 2, \dots, k-d-2\}$.

  Which is the case where the $k+d+1$ symbols are taken from two words of length k and none of the words is complete.

- Case 4:

$$x = u_{k-d-1+n}u_{k-d+n} \dots u_k \quad v_1v_2 \dots v_k \quad w_1w_2 \dots w_{d+1-n}$$

  with $n \in \{1, 2, \dots, d\}$ Which is the case where the word of length $k+d+1$ is formed by a full word, and the extra $d+1$ symbols are taken from both the end of the previous word and the beggining of the next one.

- Case 5:

  In the previous cases we considered every word u such that there is a word v and in case 4 two words v and w that come after u in $X(k)$. This is not true for the las two words of $X(k)$ so we consider them a different case.

2.2.1 Case Analysis

For the simplicity of the proof we define the function $next(w)$ that is used repetidely in the case analysis.

Definition 2.2.1. Let $next(w) : \mathcal{A}^n \to \mathcal{A}^n$ be $|w|$ times 0 if w only consists of $1s$ and the word that comes after w in lexicographic order in any other case.

Case 1

$$x = u_1u_2 \dots u_k \quad v_1v_2 \dots v_dv_{d+1}$$

This case accounts for the occurrence modulo 0 for a given word u of length k in $X(k)$ plus the remaining $d+1$ symbols which are taken from the next word. As an example, some of the words of length $k+d+1$ formed from $X(k)$ taking $k = 8, d = 3$ are shown between brackets:

(00000000 0000) 0001

(00000001 0000) 0010

(00000010 0000) 0011

$$\vdots$$

$$(00001110 \quad 0000)\ 1111$$

$$(00001111 \quad 0001)\ 0000$$

$$(00010000 \quad 0001)\ 0001$$

$$\vdots$$

$$(11111110 \quad 1111)\ 1111$$

There are two important things to notice here. The first one is that as the words of length $k+d+1$ are formed by a full word of length k followed by the first $d+1$ symbols from the next word, in almost every case the first $d+1$ symbols are equal to the last $d+1$. The only way for this not to happen, is when the last $k-d-1$ symbols from the first word u are all $1s$, which means that the next word in lexicographic order v consists of $next(v_1v_2\dots v_{d+1})$ concatenated with $next(v_{d+2}v_{d+3}\dots v_k)$.

The second important thing to notice is that as $X(k)$ is the concatenation of all words of length k, all words of length k occur one time in an alligned position modulo k. This means that the first k symbols of x takes every possible configuration.

These two facts leave two possible schemes for what a word x of case 1 may look like:

$$\underbrace{A}_{d+1} \quad \underbrace{B}_{k-d-1} \quad A$$

$$\underbrace{A}_{d+1} \quad \underbrace{11\dots1}_{k-d-1} \quad next(A)$$

For the first scheme we have:

$$2^{d+1}(2^{k-d-1}-1)$$

$$2\cdot 2^d(\frac{2^k}{2\cdot 2^d}-1)$$

$$2^k-\frac{1}{2\cdot 2^d}$$

$2^k-\frac{1}{2\cdot 2^d}$ different words.

For the second scheme we substract one to the cases due to the fact that the last word of length k in $X(k)$ has its continuation outside $X(k)$:

$$2^{d+1}-1$$

$$2\cdot 2^d-1$$

$2\cdot 2^d-1$ different words.

Counting the whole case together we have $2^k-\frac{1}{2\cdot 2^d}+2\cdot 2^d-1$ which is less than $2^k-2\cdot 2^d$ different words.

Case 2

$$x = u_{k-d-1} \dots u_k \quad v_1 v_2 \dots v_k$$

This case accounts for the occurrence modulo 0 for a given word u of length k in $X(k)$ plus the remaining $d+1$ symbols which are taken from the previous word. This means that in this case the word of length $k+d+1$ corresponding to the first word of $X(k)$ does not have a corresponding word inside $X(k)$. As an example, some of the words of length $k+d+1$ formed from $X(k)$ taking $k=8, d=3$ are shown between brackets:

0000 (0000 00000001)

0000 (0001 00000010)

$\vdots$

0000 (1111 10000000)

1000 (0000 10000001)

$\vdots$

1111 (1110 11111111)

As in the previous case, as $X(k)$ is the concatenation of all words of length k, all words of length k occur one time in an alligned position modulo k. This means that the last k symbols of x take every possible configuration. The other important thing to notice is that as the first $d+1$ symbols of x come from the word u which occups exactly before v in lexicographic order, then:

$$next(u_{k-d-1}u_{k-d} \dots u_k) = u_{v-d-1}v_{k-d} \dots v_k$$

This leaves only one possible scheme for what a word x of case 2 may look like:

$$\underbrace{A}_{d+1} \quad \underbrace{B}_{k-d-1} \quad next(A)$$

This scheme gives us the following amount of different words that may occur:

$$(2^{d+1}2^{k-d}) - 1$$

$$2 \cdot 2^d(\frac{2^k}{2^d}) - 1$$

$$2 \cdot 2^k - 1$$

$2 \cdot 2^k - 1$ different words which is less than $2 \cdot 2^k$ diffrent words.

Case 3

$$x = u_{n+1}u_{n+2}\dots u_k \quad v_1v_2\dots v_{d+n+1}$$

with $n \in \{1, 2, \dots, k-d-2\}$.

This case accounts for when the $k+d+1$ symbols are taken from two words of length k and none of the words is complete. As an example, some of the words of length $k+d+1$ formed from $X(k)$ taking $k=8, d=3$ are shown between brackets Some extra spaces are added within u and v to make clear the scheme explained later. Taking $n=1$.

```
0 (0000 000    0 0000) 001
0 (0000 001    0 0000) 010
        ⋮
0 (0001 110    0 0001) 111
0 (0001 111    0 0010) 000
0 (0010 000    0 0010) 001
        ⋮
1 (1111 101    1 1111) 110
0 (1111 110    1 1111) 111
```

Taking $n = k-d-2 = 3$

```
000 (0000 0    000 0000) 1
000 (0000 1    000 0001) 0
000 (0001 0    000 0001) 1
000 (0001 1    000 0010) 0
        ⋮
111 (1110 1    111 1111) 0
111 (1111 0    111 1111) 1
```

In this case, it also happens that as $X(k)$ is the concatenation of all words of length k, for each value of n, all words of length k take the u position once, except the last of the words of length k in $X(k)$.

It is important to notice that for a given value of n, the first n symbols of u are not be considered to form x. This means that it can be interepreted that the symbols from u that are considered are, the first $d+1$ symbols after n which are called A and the remaining $k-d-1-n$ symbols which are be called B.

Now, if we divide the $n+d+1$ symbols that are used from v to form x into the first n symbols which are called C and the remaining $d+1$ symbols, it is possible to see that these $d+1$ symbols are always equal to the symbols from A except for the case where B

$= 11\dots1$ as they account for the same indexes of u and v and v comes immediately after u in lexicographic order.

These leave two possible schemes for what a word x of case 3 may look like:

$$\underbrace{A}_{d+1} \quad \underbrace{B}_{k-d-1-n} \quad \underbrace{C}_{n} \quad A$$

$$\underbrace{A}_{d+1} \quad \underbrace{11\dots1}_{k-d-1-n} \quad \underbrace{C}_{n} \quad next(A)$$

Looking closely at the first scheme, it is possible to see, if we put together B and C which have length $k-d-1-n$ and n respectively, that we have the following scheme:

$$\underbrace{A}_{d+1} \quad \underbrace{B}_{k-d-1} \quad A$$

which is exactly the same one as in case 1. This means that all the possible words that can be formed following this scheme don't yield any new words.
The same thing happens with the second scheme when concatenating $11\dots1$ with C:

$$\underbrace{A}_{d+1} \quad \underbrace{11\dots1C}_{k-d-1} \quad next(A)$$

Which is a particular case of case 2.

This means that for case 3 there are no words that appear that should be taken into account as new words.

Case 4

$$x = u_{k-d-1+n}u_{k-d+n}\dots u_k \quad v_1v_2\dots v_k \quad w_1w_2\dots w_{d+1-n}$$

with $n \in \{1,2,\dots,d\}$

This case accounts for when the $k+d+1$ symbols are taken from three words of length k. The k symbols of v are used and the remaining $d+1$ symbols are taken from both the previous and the following words u and w. As an example, some of the words of length $k+d+1$ formed from $X(k)$ taking $k=8, d=3$ are shown between brackets Some extra spaces are added within u and v to make clear the scheme that is explained later.

Taking $n=1$.

0000000 (0	000 0000 1	000) 00010
0000000 (1	000 0001 0	000) 00011
	⋮	
0011111 (0	001 1111 1	001) 10000
0011111 (1	010 0000 0	010) 00001
	⋮	
1111110 (1	111 1111 0	111) 11111

Taking $n=2$.

000000 (00	00 0000 01	00) 000010

$$\begin{array}{lll}
000000\ (01 & 00\ 0000\ 10 & 00)\ 000011 \\
 & \vdots & \\
001111\ (01 & 00\ 1111\ 10 & 00)\ 111111 \\
001111\ (10 & 00\ 1111\ 11 & 01)\ 000000 \\
001111\ (11 & 01\ 0000\ 00 & 01)\ 000001 \\
 & \vdots & \\
111111\ (01 & 11\ 1111\ 10 & 11)\ 111111
\end{array}$$

In this case, it also happens that as $X(k)$ is the concatenation of all words of length k, for each value of n, all words of length k take the u position once, except the last of the words of length k in $X(k)$.

We call A the first n symbols of x which are taken from the end of v. The following $d+1-n$ symbols which are the first of v are called B and it happens that unless the remaining symbols of v are all 1s, they are be the same as the last $d+1-n$ of x because these symbols are the first $d+1-n$ from w. Now, we consider the remaining $k-d-1+n$ symbols from v as two blocks, one block C of length $k-d-1$ and the remaining n symbols which are exctly $next(A)$ as v is the next word in lexicographic order after u. This yields the two following schemes:

$$\underbrace{A}_{n} \quad \underbrace{B}_{d+1-n} \quad \underbrace{C}_{k-d+1} \quad next(A) \quad B$$

$$\underbrace{11\ldots10}_{n} \quad \underbrace{B}_{d+1-n} \quad \underbrace{11\ldots1}_{k-d+1} \quad next(A) \quad next(B)$$

For the first scheme we have:

$$\sum_{n=1}^{d}(2^n-1)(2^{d+1-n})(2^{k-d-1})$$

$$\sum_{n=1}^{d}(2^n-1)(\frac{2\cdot 2^d}{2^n})(\frac{2^k}{2\cdot 2^d})$$

$$2^k\sum_{n=1}^{d}\frac{2^n-1}{2^n}$$

For the second scheme we have:

$$\sum_{n=1}^{d}2^{d+1-n}$$

$$\sum_{n=1}^{d}2\frac{2^d}{2^{-n}}$$

$$2\cdot 2^d\sum_{n=1}^{d}2^{-n}$$

When putting them both together we get:

$$2^k \sum_{n=1}^{d} \frac{2^n - 1}{2^n} + 2 \cdot 2^d \sum_{n=1}^{d} 2^{-n} < d2^k + 2 \cdot 2^d$$

Case 5

We consider the last to words of $X(k)$ as special cases. The last word of $X(k)$ does not apply to any of the cases since there are no words v and w inside $X(k)$ to consider the cases. Something similar occurs with the word previous to the last one and case 4. While it is true that we do know which words come immediately after $X(k)$, which are the first words of size $d+k+2$ in lexicographic order, as we are giving an upper bound, it is valid to consider all of these as different words to all of the ones considered in the previous cases. By doing this, we would have to consider $d+k+1$ new words for the last word and d words for the previous one. So this yields $2d+k+1$ words to consider.

2.2.2 Bounding words that appear in Champernowne

If the Champernowne sequence were supernormal then the expected frequency of words that appear at least one time in the first $2^n + n - 1$ symbols would be:

$$\lim_{n\to\infty} \frac{\#\{w : |w| = n, |champ[1...(2^n + n - 1)]|_w > 0\}}{2^n} = 1 - e^{-1}$$

Now, by Fact 2.2.2 we know that $X(k)$ accounts for half of the words of the first 2^{d+k+1} symbols of Champernowne. If we analize what happens with words of length $d+k+1$ using the bounds we have for the occurrences of different words within $X(k)$ and we assume that the remaining $2^{d+k} + d + k - 1$ symbols are all different, then

$$\lim_{d\to\infty} \frac{\#\{w : |w| = d+k+1, |champ[1 \dots 2^{d+k+1} + d + k + 1 - 1]|_w > 0\}}{2^{d+k+1}} =$$

$$\lim_{d\to\infty} \frac{\#\{w : |w| = d+k+1, |champ[1 \dots 2^{d+k+1} + d + k]|_w > 0\}}{2^{d+k+1}} <$$

$$\lim_{d\to\infty} \frac{\text{Case 1} + \text{Case 2} + \text{Case 4} + \text{Case 5} + \text{other half} + (d+k)}{2^{d+k+1}} =$$

$$\lim_{d\to\infty} \frac{(2^k - 2 \cdot 2^d) + (2 \cdot 2^k) + (d2^k + 2 \cdot 2^d) + (2d + k + 1) + (2^{d+k}) + (d+k)}{2^{d+k+1}} =$$

$$\lim_{d\to\infty} \frac{(2^k - 2k) + (2 \cdot 2^k) + (d2^k + 2k) + (3d + 2k + 1) + (k2^k)}{2k2^k} =$$

$$\lim_{d\to\infty} \frac{3 \cdot 2^k + d2^k + 3d + 2k + 1}{2k2^k} + \frac{1}{2} =$$

$$\lim_{d\to\infty} \frac{3}{2k} + \frac{d}{2k} + \frac{d}{k2^k} + \frac{1}{2 \cdot 2^k} + \frac{1}{2k2^k} + \frac{1}{2} = \frac{1}{2} < 1 - e^{-1}$$

Finally, if in the original sequence we consider that the n's such that $n = d + 2^d + 1$ are a subsequence of $n = 1, 2, \ldots$ then we can say that if

$$\lim_{n\to\infty} \frac{\#\{w : |w| = n, |champ[1...(2^n + n - 1)]|_w > 0\}}{2^n}$$

exists, then it is not $1 - e^{-1}$. This implies that Champernowne is not 1-supernormal which means it is not supernormal.

Corollary 2.2.0.1. *If x is a normal number it is not implied that x is supernormal.*

3. BIBLIOGRAPHY

BIBLIOGRAPHY

[BC18] Verónica Becher and Olivier Carton. *Normal Numbers and Computer Science*, pages 233–269. Springer International Publishing, Cham, 2018.

[BCC19] Veronica Becher, Olivier Carton, and Ignacio Cunningham. Low discrepancy sequences failing poissonian pair correlations. *Archiv der Mathematik*, 113, 04 2019.

[PS19] Ísabel Pirsic and Wolfgang Stockinger. The champernowne constant is not poissonian. *Funct. Approx. Comment. Math.*, 60(2):253–262, 03 2019.