

Alan Turing's algorithm for producing normal numbers

Verónica Becher

Universidad de Buenos Aires

Computable 90 Conference
The National Museum of Computing / London Mathematical Society
16–18 September 2026

A Note on Neural Networks

A.N. Turing

of steps. When this figure has been calculated and written down as on the figure of the K in section 1 is finished. Hence

What does K do in the K th section? Now let K be the D.N. of K . What does K do in the K th section? It must test whether K is satisfactory giving a very

On the other hand the verdict cannot be K . For if it were, then in the K th section of its motion K would be bound to compute the first $R(K) + 1$ figures of the sequence

It may be noted that the above sequence computed by the machine with K as its D.N. and to write down the sequence by

It may also be noted that the instructions for calculating the $R(K)$ th would amount to "calculate the first $R(K)$ figures computed by K and write down the $R(K)$ th".

It is clear that the above sequence is not a sequence of figures but a sequence of instructions for calculating figures. It is clear that the above sequence is not a sequence of figures but a sequence of instructions for calculating figures. It is clear that the above sequence is not a sequence of figures but a sequence of instructions for calculating figures.

[illegible][illegible][illegible]

10. Example
 Let $V = \mathbb{R}^n$ and let $T: V \rightarrow V$ be a linear transformation.
 (1) T is linear. (2) T is invertible. (3) T is symmetric.
 (4) T is orthogonal. (5) T is normal. (6) T is self-adjoint.
 (7) T is unitary. (8) T is Hermitian. (9) T is real symmetric.
 (10) T is real antisymmetric. (11) T is real symmetric and antisymmetric.
 (12) T is real symmetric and Hermitian. (13) T is real symmetric and unitary.
 (14) T is real symmetric and orthogonal. (15) T is real symmetric and self-adjoint.
 (16) T is real symmetric and normal. (17) T is real symmetric and invertible.
 (18) T is real symmetric and non-invertible. (19) T is real symmetric and zero.
 (20) T is real symmetric and identity. (21) T is real symmetric and scalar multiple of identity.

[illegible]

A Note on Normal Numbers

Although it is known that almost all numbers are normal ¹⁾ no example of a normal number has ever been given. I propose to shew how normal numbers may be constructed and to prove that almost all numbers are normal constructively

Consider the R -figure integers in the scale of t ($t \geq 2$). If γ is any sequence of figures in that scale we denote by $N(t, \gamma, n, R)$ the number of these in which γ occurs exactly n times. Then it can be proved without difficulty that

$$\frac{\sum_{n=1}^R n N(t, \gamma, n, R)}{\sum_{n=1}^R N(t, \gamma, n, R)} = \frac{R - r + 1}{R} t^{-r}$$

where $\ell(\gamma) = r$ is the length of the sequence γ : it is also possible ²⁾ to prove that

A Note on Normal Numbers

Although it is known that almost all numbers are normal⁽¹⁾, no example of a normal number has ever been given. I propose to show how normal numbers may be constructed and to prove that almost all numbers are normal constructively.

Consider the N -figure integers in the scale of t ($t \geq 2$). If γ is any sequence of figures in that scale we denote by $N(\gamma, \gamma, N)$ the number of those in which γ occurs exactly γ times. Thus it can be proved without difficulty that

$$\frac{\sum_{\gamma=1}^N N(\gamma, \gamma, N)}{N} = \frac{N}{t^N} = t^{-N}$$

where $\frac{N}{t^N}$ is the length of the sequence γ ; it is also possible to prove that

$$\sum_{\gamma=1}^N N(\gamma, \gamma, N) \leq t^N \quad \text{and} \quad \frac{N}{t^N} \leq t^N \quad (1)$$

Let N be a real number and $S(\gamma, \gamma, N)$ the number of occurrences of γ in the first N figures after the decimal point of the expression of N in the scale of t . N is said to be normal if

$$N^{-1} S(\gamma, \gamma, N) \rightarrow t^{-N}$$

where $N \rightarrow \infty$.

Now consider some of a finite number of open intervals with rational end points. These can be enumerated constructively. We take a particular constructive enumeration; let K_n be the n -th

set of intervals in the enumeration, then we have

Proposition 1

We can find a constructive⁽²⁾ function $K(N, n)$ of two integral variables, such that

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

$$K(N, n) \leq K(N, n+1) \quad \text{and} \quad K(N, n) \leq K(N, n+1)$$

then if $K(N, n) > 2 - K(N, n) > 1 - \frac{1}{K(N, n)}$

$C(N, n)$ is to be defined as follows

$$C(N, n) = \frac{1}{2}$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

$$C(N, n) = \frac{1}{2} \quad \text{if } N \leq 2$$

A Note on Normal Numbers, Alan M. Turing

Written presumably in 1936–1937.

A Note on Normal Numbers, Alan M. Turing

Written presumably in 1936–1937.

Unpublished until 1992, when it appeared in the Collected Works, volume Pure Mathematics, edited by John L. Britton. An editorial note says that the proof of the second theorem is inadequate and speculates that the theorem could be false.

A Note on Normal Numbers, Alan M. Turing

Written presumably in 1936–1937.

Unpublished until 1992, when it appeared in the Collected Works, volume Pure Mathematics, edited by John L. Britton. An editorial note says that the proof of the second theorem is inadequate and speculates that the theorem could be false.

Reconstructed, corrected and completed in 2007

Becher, Figueira and Picchi, *Theoretical Computer Science* 377, 126-138.

Expansion of a real number in an integer base

For a real number x , its expansion in an integer base $b \geq 2$ is a sequence of integers $a_1, a_2, \dots$, where each $a_n \in \{0, \dots, b-1\}$ such that

$$x - \lfloor x \rfloor = \sum_{n \geq 1} a_n b^{-n} = 0.a_1 a_2 a_3 \dots$$

For example, π written in base 10 is 3.14159..., so its expansion is 1, 4, 1, 5, 9, ...

For rationals with two representations, we choose the one ending with 0s.

Normal Numbers

Defined by Émile Borel in 1909:

Normal Numbers

Defined by Émile Borel in 1909:

A real number is *normal to an integer base ≥ 2* if its expansion in that base is evenly balanced: all blocks of digits of the same length occur with the same limiting frequency.

Normal Numbers

Defined by Émile Borel in 1909:

A real number is *normal to an integer base ≥ 2* if its expansion in that base is evenly balanced: all blocks of digits of the same length occur with the same limiting frequency.

For instance, if a number is normal to base 2, each of the digits '0' and '1' occurs, in the limit, half of the time; each of the blocks '00', '01', '10' and '11' occurs one fourth of the time, and so on.

Normal Numbers

Defined by Émile Borel in 1909:

A real number is *normal to an integer base ≥ 2* if its expansion in that base is evenly balanced: all blocks of digits of the same length occur with the same limiting frequency.

A real number that is normal to every integer base ≥ 2 is called *absolutely normal*, or just *normal*.

Counterexamples

0.1010010001000010000010000... is not normal to base 2.

Counterexamples

0.1010010001000010000010000... is not normal to base 2.

0.1010101010101010101010101... is not normal to base 2.

Counterexamples

$0.1010010001000010000010000\dots$ is not normal to base 2.

$0.1010101010101010101010101\dots$ is not normal to base 2.

Rationals are not normal to any base.

Existence

Theorem (Borel 1909)

Almost all real numbers are normal.

Existence

Theorem (Borel 1909)

Almost all real numbers are normal.

Borel asked for an explicit example.

A Note on Normal Numbers

Although it is known that almost all numbers are normal ¹⁾ no example of a normal number has ever been given. I propose to shew how normal numbers may be constructed and to prove that almost all numbers are normal constructively

Consider the R -figure integers in the scale of t ($t \geq 2$).

If γ is any sequence of figures in that scale we denote by $N(t, \gamma, n, R)$ the number of these in which γ occurs exactly n times. Then it can be proved without difficulty that

$$\frac{\sum_{n=2}^R n N(t, \gamma, n, R)}{\sum_{n=1}^R N(t, \gamma, n, R)} = \frac{R-r+1}{R} t^{-r}$$

where $\ell(\gamma) = r$ is the length of the sequence γ : it is also possible ²⁾ to prove that

$$\sum_{n=Rt^{-r}}^R N(t, \gamma, n, R) < 2 t^R e^{-K^2 t^r / 4R}$$

$$|n - Rt^{-r}| > K$$

provided

$$\frac{K t^r}{R} < \frac{3}{4} \quad (1)$$

Turing's Note on Normal Numbers

Turing's Theorem 1

Borel's theorem that almost all numbers are normal, constructively.

Turing's Note on Normal Numbers

Turing's Theorem 1

Borel's theorem that almost all numbers are normal, constructively.

Turing's Theorem 2

An algorithm for producing normal numbers.

Turing's Note on Normal Numbers

Turing's Theorem 1

Borel's theorem that almost all numbers are normal, constructively.

Turing's Theorem 2

An algorithm for producing normal numbers.

Turing's First Page of the Handwritten Manuscript

His own appraisal of his work.

Turing's Theorem 1

Theorem 1

We can find a constructive³⁾ function $c(K, n)$ of two integral variables, such that

$$\widehat{K}_{c(K, n+1)} \leq \widehat{K}_{c(K, n)}$$

and $n \widehat{K}_{c(K, n)} > 1 - \frac{1}{K}$ for each K, n

and $\widehat{K}_{(K)} = \prod_{n=1}^{\infty} \widehat{K}_{c(K, n)}$ consists entirely of normal numbers for
each K .

Constructive means Computable

The class of *computable functions* consists of the partial functions $f: \mathbb{N}^t \rightarrow \mathbb{N}$ for which some Turing machine, on input $(x_1, \dots, x_t)$ (each x_i in binary), halts with $f(x_1, \dots, x_t)$ (in binary) exactly when this value is defined.

Equivalently, it is the smallest class of partial functions $\mathbb{N}^t \rightarrow \mathbb{N}$ containing zero, successor and the projections, and closed under composition, primitive recursion and minimization.

The class of *total computable functions* consists of those defined on all of $\mathbb{N}^t$.

Turing's Theorem 1

There is a total computable function $c(k, n)$ of two integer arguments with values in finite sets of pairs of rational numbers in the unit interval such that

- For $c(k, n) = \{(a_1, b_1), \dots, (a_m, b_m)\}$ let $E_{c(k, n)} = \bigcup_{i=1}^m (a_i, b_i)$.
- $E_{c(k, n)}$ is included in $E_{c(k, n-1)}$.
- The measure of $E_{c(k, n)}$ is greater than $1 - 1/k$.
- For each k , $E(k) = \bigcap_n E_{c(k, n)}$ has measure exactly $1 - 1/k$ and consists entirely of normal numbers.

Turing's Theorem 1 “finite approximations”

The construction is uniform in the argument k .

Prune the unit interval by steps.

Step 0: $E_{c(k,0)}$ is the whole unit interval.

Step n : $E_{c(k,n)}$ results from removing from $E_{c(k,n-1)}$ the points that are **not** candidates to be normal, according to the inspection of an initial segment of their expansions.

Turing's Theorem 1 “finite approximations”

Total computable functions of the step n ,

initial segment size	linear
base	sublinear
block length	sublogarithmic
frequency tolerance	the technically largest, still converging to 0

$E_{c(k,n)}$ is the set of reals not discarded up to step n .

It is the union of **finitely many** intervals with rational endpoints.

Turing's Theorem 1 “finite approximations”

At the end, the construction discards
all rational numbers, because of their periodic structure,

Turing's Theorem 1 “finite approximations”

At the end, the construction discards

- all rational numbers, because of their periodic structure,
- all irrational numbers with an unbalanced expansion,

Turing's Theorem 1 “finite approximations”

At the end, the construction discards

- all rational numbers, because of their periodic structure,
- all irrational numbers with an unbalanced expansion,
- all normal numbers whose convergence to normality is too slow.

The set $E(k) = \bigcap_n E_{c(k,n)}$ consists entirely of normal numbers.
Its measure is exactly $1 - 1/k$.

Law of Large Numbers for blocks of digits

Fix a base $b \geq 2$. In most sequences of N digits (N large enough),

each single digit occurs about N/b times

each block of two digits occurs about N/b^2 times

each block of three digits occurs about N/b^3 times

...

Law of Large Numbers for blocks of digits

Fix a base $b \geq 2$. In most sequences of N digits (N large enough),

each single digit occurs about N/b times

each block of two digits occurs about N/b^2 times

each block of three digits occurs about N/b^3 times

...

For N large enough, in most sequences of N digits in base b every block of a given length ℓ occurs about N/b^ℓ times.

Equivalently, few sequences of N digits have some block, say of length ℓ , occurring far fewer, or far more, than N/b^ℓ times.

AN INTRODUCTION
TO THE
THEORY OF NUMBERS

BY

G. H. HARDY

*Sadleirian Professor of Pure Mathematics in the
University of Cambridge*

AND

E. M. WRIGHT

*Professor of Mathematics in the
University of Aberdeen*

OXFORD
AT THE CLARENDON PRESS
1938

Quantitative Law of Large Numbers

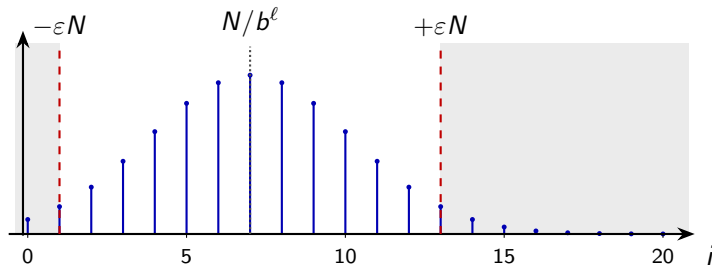
Lemma (extends Hardy & Wright 1938)

Let $b \geq 2, N, \ell$ be integers and ε a real with $7/N \leq \varepsilon \leq 1/b^\ell$.
Then for any block w of length ℓ ,

$$\sum_{|i - N/b^\ell| \geq N\varepsilon} \text{number of sequences of } N \text{ digits} \leq 2 b^N e^{-\varepsilon^2 N c_{b,\ell}},$$

with exactly i occurrences of w

for a positive constant $c_{b,\ell}$.



Turing's Theorem 2

Theorem 2

There is a rule whereby given an integer k and an infinite sequence of figures 0 and 1 (the p th figure in the sequence being $v(p)$) we can find a normal number $\alpha(k, v)$ in the interval $(0,1)$ and in such a way that for fixed k these numbers form a set of measure at least $1 - 2/k$, and so that the first n figures of v determine $\alpha(k, v)$ to within 2^{-n} .

Turing's Theorem 2

There is an algorithm that, given an integer k and an infinite sequence ν of zeros and ones, produces a normal number $\alpha(k, \nu)$ in the unit interval, expressed in base 2.

Turing's Theorem 2

There is an algorithm that, given an integer k and an infinite sequence ν of zeros and ones, produces a normal number $\alpha(k, \nu)$ in the unit interval, expressed in base 2.

In order to write down the first n digits of $\alpha(k, \nu)$ the algorithm requires at most the first n digits of ν .

Turing's Theorem 2

There is an algorithm that, given an integer k and an infinite sequence ν of zeros and ones, produces a normal number $\alpha(k, \nu)$ in the unit interval, expressed in base 2.

In order to write down the first n digits of $\alpha(k, \nu)$ the algorithm requires at most the first n digits of ν .

For a fixed k these numbers $\alpha(k, \nu)$ form a set of measure at least $1 - 2/k$.

Theorem 2: Algorithm “follow the measure”

It works by steps.

Theorem 2: Algorithm “follow the measure”

It works by steps.

Start with the unit interval.

Theorem 2: Algorithm “follow the measure”

It works by steps.

Start with the unit interval.

At each step, divide the current interval into two halves, and choose the half that still contains candidates in large enough measure.

Theorem 2: Algorithm “follow the measure”

It works by steps.

Start with the unit interval.

At each step, divide the current interval into two halves, and choose the half that still contains candidates in large enough measure.

If both halves do, use the current bit of the oracle ν to decide (this will happen infinitely often).

The output $\alpha(k, \nu)$ is the trace of the left/right choice at each step.

Turing's Algorithm

With each integer n we associate an interval of the form

$\left(\frac{m_n}{2^n}, \frac{m_n+1}{2^n}\right)$ whose intersection with $E_{(K)}$ is of positive measure .
and given m_n we obtain m_{n+1} as follows. Put

$$m_n E_{(K,n)} \cap \left(\frac{m_n}{2^n}, \frac{2m_n+1}{2^{n+1}}\right) = a_{n,m}$$

$$m_n E_{(K,n)} \cap \left(\frac{2m_n+1}{2^{n+1}}, \frac{m_n+1}{2^n}\right) = b_{n,m}$$

and let r_n be the smallest m for which either $a_{n,m} < K^{-1} 2^{-2n}$
or $b_{n,m} < K^{-1} 2^{-2n}$ or both $a_{n,m} > \frac{1}{K(K+n+1)}$ and $b_{n,m} > \frac{1}{K(K+n+1)}$

There exists such an r_n for $a_{n,m}$ and $b_{n,m}$ decrease either to 0

or to some positive number. In the case where $a_{n,r_n} < K^{-1} 2^{-2n}$ we
put $m_{n+1} = 2m_n + 1$: if $a_{n,r_n} > K^{-1} 2^{-2n}$ but $b_{n,r_n} < K^{-1} 2^{-2n}$

we put $m_{n+1} = 2m_n$, and in the third case we put $m_{n+1} = 2m_n$

or $m_{n+1} = 2m_n + 1$ according as $v(u) = 0$ or 1. For each n the
interval $\left(\frac{m_n}{2^n}, \frac{m_n+1}{2^{n+1}}\right)$ includes normal numbers in positive measure.

The intersection of these intervals contains only one number.

which must be normal.

Turing's algorithm

- Input: Positive integer k and infinite sequence ν of zeros and ones.
- Algorithm: Nested, decreasing, dyadic intervals $I_0, I_1, I_2, \dots$.
At step n , I_n is the left half, or the right half, of I_{n-1} .
When the choice is left, the n -th bit of $\alpha(k, \nu)$ in base 2 is 0; and when the choice is right it is 1.
- Output: $\alpha(k, \nu)$ is the trace of the left/right choice.

Correctness of the Algorithm

From Theorem 1, $E(k) = \bigcap_n E_{c(k,n)}$. The algorithm determines $I_0, I_1, I_2, \dots$

Invariant: $I_n \cap E(k)$ has positive measure.

Correctness of the Algorithm

From Theorem 1, $E(k) = \bigcap_n E_{c(k,n)}$. The algorithm determines $I_0, I_1, I_2, \dots$

Invariant: $I_n \cap E(k)$ has positive measure.

Threshold: $T(k, n)$ is a lower bound on the measure of $I_n \cap E_{c(k,n)}$,
 $T(k, n) = T(k, n-1)/2 - \text{measure}(E_{c(k,n-1)} \setminus E_{c(k,n)})/2$.

Output: $\alpha(k, \nu) = \bigcap_n I_n$.

Hence, $\alpha(k, \nu) \in \bigcap_n E_{c(k,n)}$, so it is normal.

Correctness of the Algorithm

From Theorem 1, $E(k) = \bigcap_n E_{c(k,n)}$. The algorithm determines $I_0, I_1, I_2, \dots$

Invariant: $I_n \cap E(k)$ has positive measure. 

Threshold: $T(k, n)$ is a lower bound on the measure of $I_n \cap E_{c(k,n)}$,
 $T(k, n) = T(k, n-1)/2 - \text{measure}(E_{c(k,n-1)} \setminus E_{c(k,n)})/2$.

Output: $\alpha(k, \nu) = \bigcap_n I_n$.

Hence, $\alpha(k, \nu) \in \bigcap_n E_{c(k,n)}$, so it is normal.

Correctness of the Reconstructed Algorithm

Define $e : \mathbb{N} \rightarrow \mathbb{N}$; $e(n)$ is exponential in n . Redefine $c(k, n)$ so that $E_{c(k, e(n+1))} \subseteq E_{c(k, e(n))}$ and let $E(k) = \bigcap_n E_{c(k, e(n))}$. The algorithm determines $I_0, I_1, I_2, \dots$

Invariant: $I_n \cap E(k)$ has positive measure.



Correctness of the Reconstructed Algorithm

Define $e : \mathbb{N} \rightarrow \mathbb{N}$; $e(n)$ is exponential in n . Redefine $c(k, n)$ so that $E_{c(k, e(n+1))} \subseteq E_{c(k, e(n))}$ and let $E(k) = \bigcap_n E_{c(k, e(n))}$. The algorithm determines $I_0, I_1, I_2, \dots$

Invariant: $I_n \cap E(k)$ has positive measure. 

Threshold: $T(k, n)$ is a lower bound on the measure of $I_n \cap E_{c(k, e(n))}$,
 $T(k, n) = T(k, n-1)/2 - \text{measure}(E_{c(k, e(n-1))} \setminus E_{c(k, e(n))})/2$.

Output: $\alpha(k, \nu) = \bigcap_n I_n$

Hence, $\alpha(k, \nu) \in \bigcap_n E_{c(k, e(n))}$. An extra argument implies that $\alpha(k, \nu)$ is normal (Piatetskii-Shapiro 1951).

Turing's Theorem 2 (reconstruction comments)

A number x is normal to base b iff for every block w , of any length ℓ ,

$$\lim_{N \rightarrow \infty} \frac{\text{\#occurrences of } w \text{ in the first } N \text{ digits of } x}{N} = b^{-\ell}.$$

Turing's Theorem 2 (reconstruction comments)

A number x is normal to base b iff for every block w , of any length ℓ ,

$$\lim_{N \rightarrow \infty} \frac{\text{\#occurrences of } w \text{ in the first } N \text{ digits of } x}{N} = b^{-\ell}.$$

Our reconstruction, however, only controls frequencies at the checkpoints $N = e(n)$, with $e(n)$ exponential in n . So it yields x with

$$\lim_{n \rightarrow \infty} \frac{\text{\#occurrences of } w \text{ in the first } e(n) \text{ digits of } x}{e(n)} = b^{-\ell}.$$

Turing's Theorem 2 (reconstruction comments)

A number x is normal to base b iff for every block w , of any length ℓ ,

$$\lim_{N \rightarrow \infty} \frac{\# \text{occurrences of } w \text{ in the first } N \text{ digits of } x}{N} = b^{-\ell}.$$

Our reconstruction, however, only controls frequencies at the checkpoints $N = e(n)$, with $e(n)$ exponential in n . So it yields x with

$$\lim_{n \rightarrow \infty} \frac{\# \text{occurrences of } w \text{ in the first } e(n) \text{ digits of } x}{e(n)} = b^{-\ell}.$$

Between checkpoints the frequency can drift by at most the constant factor $C = \sup_n e(n+1)/e(n)$; so, infinitely many lengths ℓ , for every block w of length ℓ ,

$$\limsup_{N \rightarrow \infty} \frac{\# \text{occurrences of } w \text{ in the first } N \text{ digits of } x}{N} \leq C b^{-\ell}.$$

By Piatetskii-Shapiro's theorem this implies x is normal.

Turing's Theorem 2 (reconstruction comments)

Read literally, Turing's proof does not establish the theorem as stated.

But the scheme of Turing's algorithm is correct; with checkpoints at every length the invariant cannot be proved; with exponentially spaced checkpoints it can (because the measure of the set of candidates discarded at step n decays doubly exponentially in n).

Piatetskii-Shapiro's theorem closes that gap.

The first application of computable numbers

In the seminal paper “On Computable Numbers ...” Turing defines that a real number x is computable if there is a Turing machine that outputs each of the digits in the expansion of x in some base, one digit after the other.

Equivalently, a real number x is computable if there is a total computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n)$ is the n -th digit in the expansion of x in some base.

The first application of computable numbers

In the seminal paper “On Computable Numbers ...” Turing defines that a real number x is computable if there is a Turing machine that outputs each of the digits in the expansion of x in some base, one digit after the other.

Equivalently, a real number x is computable if there is a total computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n)$ is the n -th digit in the expansion of x in some base.

A measure-theoretic argument does not prove the existence of computable normal numbers: computable numbers form a countable, hence measure-zero, set.

The first application of computable numbers

In the seminal paper “On Computable Numbers ...” Turing defines that a real number x is computable if there is a Turing machine that outputs each of the digits in the expansion of x in some base, one digit after the other.

Equivalently, a real number x is computable if there is a total computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n)$ is the n -th digit in the expansion of x in some base.

A measure-theoretic argument does not prove the existence of computable normal numbers: computable numbers form a countable, hence measure-zero, set.

When ν is the expansion of a computable number (Turing puts all zeros), Turing’s algorithm yields a computable normal number.

The first application of computable numbers

In the seminal paper “On Computable Numbers ...” Turing defines that a real number x is computable if there is a Turing machine that outputs each of the digits in the expansion of x in some base, one digit after the other.

Equivalently, a real number x is computable if there is a total computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n)$ is the n -th digit in the expansion of x in some base.

A measure-theoretic argument does not prove the existence of computable normal numbers: computable numbers form a countable, hence measure-zero, set.

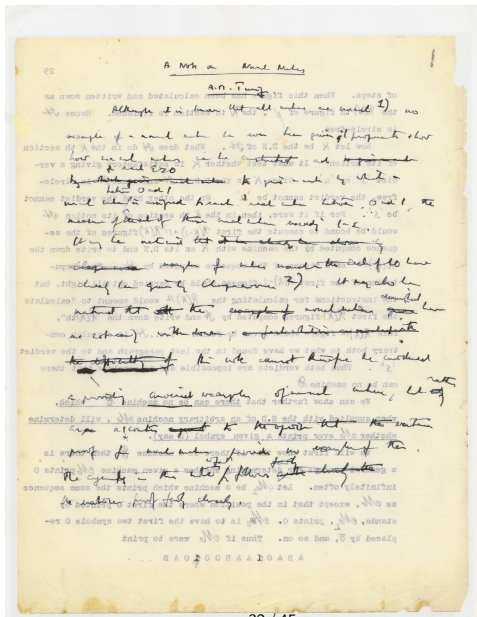
When ν is the expansion of a computable number (Turing puts all zeros), Turing’s algorithm yields a computable normal number.

Turing's Normal Numbers

As ν ranges over all binary sequences, the set of numbers $\alpha(k, \nu)$ that can be output by the algorithm has measure at least $1 - 2/k$.

Besides, the algorithm can be adapted to insert the bits of ν at fixed positions of the output sequence. Thus, there is a normal number in each Turing degree.

Turing's First Page of the Handwritten Manuscript



Turing's First Page of the Handwritten Manuscript

Motivation: *"No example of a normal number has ever been given."*

Turing cites Champernowne's 0.123456789101112131415... as an example of a normal number to base 10 (it has not been proved to be normal to any other base).

"It may also be natural that an example of a normal number be demonstrated as such and written down. This note cannot, therefore, be considered as providing convenient examples of normal numbers"

Turing's First Page of the Handwritten Manuscript

Motivation: *"No example of a normal number has ever been given."*

Turing cites Champernowne's 0.123456789101112131415... as an example of a normal number to base 10 (it has not been proved to be normal to any other base).

"It may also be natural that an example of a normal number be demonstrated as such and written down. This note cannot, therefore, be considered as providing convenient examples of normal numbers"

He was aware of the algorithm's computational complexity.

Turing's First Page of the Handwritten Manuscript

Motivation: *"No example of a normal number has ever been given."*

Turing cites Champernowne's 0.123456789101112131415... as an example of a normal number to base 10 (it has not been proved to be normal to any other base).

"It may also be natural that an example of a normal number be demonstrated as such and written down. This note cannot, therefore, be considered as providing convenient examples of normal numbers"

He was aware of the algorithm's computational complexity.

"...but rather, to counter [...] that the existence proof of normal numbers provides no example of them. The arguments in the note, in fact, follow the existence proof fairly closely."

Turing's First Page of the Handwritten Manuscript

Motivation: *"No example of a normal number has ever been given."*

Turing cites Champernowne's 0.123456789101112131415... as an example of a normal number to base 10 (it has not been proved to be normal to any other base).

"It may also be natural that an example of a normal number be demonstrated as such and written down. This note cannot, therefore, be considered as providing convenient examples of normal numbers"

He was aware of the algorithm's computational complexity.

"...but rather, to counter [...] that the existence proof of normal numbers provides no example of them. The arguments in the note, in fact, follow the existence proof fairly closely."

Which existence proof Turing refers to?

Speculation: Why did Turing put the note aside?

- His algorithm wasn't "convenient" (exponential complexity).
- The error in the induction to prove the algorithm's invariant.
- He moved to other problems.

Letter G.H. Hardy to Turing (AMT/D/5)

as from
Trin. Coll. Camb.

June 1

Dear Turing

I have just come across your letter (March 28), which I seem to have put aside for reflection and forgotten.

I have a vague recollection that Borel says in one of his books that (Lebesgue had shown him a construction. Try Leçons sur la théorie de la croissance (including the appendices), or the probability book (written under his direction by a lot of people, but including one volume on arithmetical prob[ability, by himself). Also I seem to remember vaguely that, when Champernowne was doing his stuff, I had a hunt, but could find nothing satisfactory anywhere.

Now, of course, when I do write, I do so from London, where I have no books to refer to. But if I put it off till I return, I may forget again. Sorry to be so unsatisfactory. But my 'feeling' is that L. made a proof which never got published.

Yours sincerely,
G.H. Hardy

as from

Trin. Coll. Camb.

June 1

Dear Turing,

I have just come across your letter (March 28), which I seem to have put aside for reflection and forgotten.

I have a vague recollection that Borel says in one of his books that Lebesgue had shown him a construction. Try Leçons sur la théorie de la croissance (including the appendices), or the probability book (written under his direction by a lot of people, but including one volume on arithmetical prob[ability, by himself).

Also I seem to remember vaguely that, when Champernowne was doing his stuff, I had a hunt, but could find nothing satisfactory anywhere.

Now, of course, when I do write, I do so from London, where I have no books to refer to. But if I put it off till I return, I may forget again.

Sorry to be so unsatisfactory. But my 'feeling' is that L. made a proof which never got published.

Yours sincerely,

G.H. Hardy

1/2 late 30

SUR CERTAINES DÉMONSTRATIONS D'EXISTENCE ;

PAR M. H. LEBESGUE.

Dans une lettre, adressée à M. Borel, et qui accompagnait l'envoi de l'article précédent, M. Sierpinski se demandait si cet article devait être publié, s'il ne ferait pas double emploi avec une démonstration que j'avais indiquée à M. Borel et que celui-ci a signalée dans la deuxième édition de ses *Leçons sur la théorie des fonctions* (p. 198).

DÉMONSTRATION ÉLÉMENTAIRE DU THÉORÈME DE M. BOREL SUR LES NOMBRES ABSOLUMENT NORMAUX ET DÉTERMINATION EFFECTIVE D'UN TEL NOMBRE ;

PAR M. W. SIERPINSKI.

On appelle, d'après M. Borel, *simplement normal* par rapport à la base q ⁽¹⁾ tout nombre réel x dont la partie fractionnaire

(1) E. BOREL, *Leçons sur la théorie des fonctions*, p. 197, Paris, 1914.

Henri Lebesgue and Waław Sierpiński independently gave examples of normal numbers: *Bulletin de la Société Mathématique de France* 45, 1917, respectively on pages 132–144 and 125–132.

Speculation: Why did Turing put the note aside?

- His algorithm wasn't "convenient" (exponential complexity).
- The error in the induction to prove the algorithm's invariant.
- He moved to other problems.
- Turing was unaware of Lebesgue's and Sierpiński's work, and Hardy's reply gave him no priority to claim. 

Unrecognized priority

Lebesgue and Sierpiński each constructed their example in essentially the same way: they took the infimum of an uncountable set of numbers in $(0, 1)$. The set was chosen so that it consists entirely of numbers normal to all bases. Moreover, the infimum belongs to the set, so it is actually the minimum of the set.

But “taking the minimum of an uncountable set” **is not a finitary construction**.

In contrast, Turing’s algorithm gives finitarily defined normal numbers. It seems that G.H. Hardy missed the importance of this.

Algorithms Producing Normal Numbers

We say that an algorithm producing a real number has computational complexity $O(f(n))$ if it uses $O(f(n))$ elementary operations to produce the n -th digit.

- (Double) exponential time, Turing's algorithm, 1936–1937.
- Exponential time, Weyl criterion and exponential sums
Schmidt 1961/1962; M. Levin 1979
- Polynomial time
Lutz and Mayordomo 2013, Figueira and Nies 2013, Becher, Heiber and Slaman 2013.
- Nearly linear time
Lutz and Mayordomo, 2016, 2021.

Our Algorithm, nearly quadratic. Output in Base 10

0.4031290542003809132371428380827059102765116777624189775110896366...

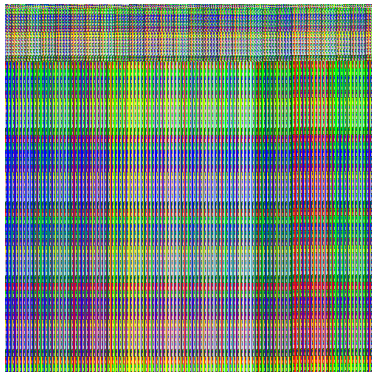


First 250000 digits output by the algorithm

Plotted in 500x500 pixels, 10 colors

Parameters $t_1 = 3$; $\epsilon_1 = 1$; $t_i = (3 \log(i)) + 3$; $\epsilon_i = 1/t_i$.

Programmed by Martin Epsztejn, University of Buenos Aires



First 250000 digits of Champernowne

Plotted in 500x500 pixels, 10 colors

Borel's question is still unresolved

It is desirable to show that a *known* irrational number such as π or $\sqrt{2}$, is normal.

We would like an example with a simple mathematical definition which, in addition to being normal, provably has extra properties.

Borel's question is still unresolved

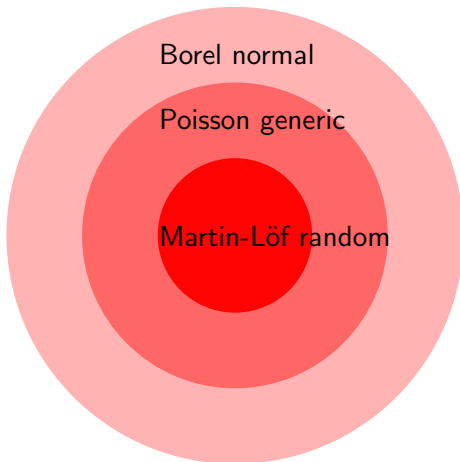
It is desirable to show that a *known* irrational number such as π or $\sqrt{2}$, is normal.

We would like an example with a simple mathematical definition which, in addition to being normal, provably has extra properties.

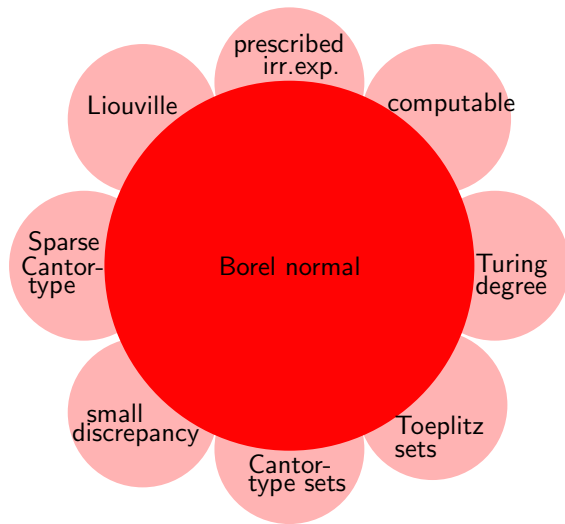
Conjecture (Borel 1951)

All algebraic irrational numbers are normal.

Normal and More



Normal and More



- Turing, A. M. A Note on Normal Numbers. *Collected Works of Alan M. Turing, volume on Pure Mathematics*, 1992, edited by J. L. Britton, 117-119. Notes of editor, 263–265. North Holland. Reprinted in *Alan Turing - his Work and Impact*, S B. Cooper and J. van Leeuwen editors, Elsevier, 2012.
- Turing, A.M. 1936. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society Series 2*, 42:230–265.
- Becher, V., Figueira, S., Picchi, R. 2007. Turing's unpublished algorithm for normal numbers. *Theoretical Computer Science* 377:126–138.
- Becher, V., Heiber, P., Slaman, T., 2013. A polynomial-time algorithm for computing absolutely normal numbers, *Information and Computation*, 232:1-9
- Borel, É. 1909. Les probabilités dénombrables et leurs applications arithmétiques. *Rendiconti del Circolo Matematico di Palermo* 27:247–271.
- Champernowne, D. 1933. The construction of decimals normal in the scale of ten. *Journal of the London Mathematical Society* 8: 254–260.
- Hardy, G.H., Wright, E.M. 1938 first edition. *An Introduction to the Theory of Numbers*. Oxford University Press.
- Lutz, J. and Mayordomo, E. 2021. Computing absolutely normal numbers in nearly linear time. *Information and Computation* 281:104746.
- Lebesgue, H. 1917. Sur certaines démonstrations d'existence. *Bulletin de la Société Mathématique de France* 45:132–144.
- Piatetskii-Shapiro, I.I. 1951 .On the law of distribution of the fractional parts of the exponential function, *Izv. Akad. Nauk SSSR Ser. Mat.*, 15(1):47–52.
- Sierpiński, W. 1917. Démonstration élémentaire du théorème de M. Borel sur les nombres absolument normaux et détermination effective d'un tel nombre. *Bulletin de la Société Mathématique de France* 45:125–132.

Turing's Note on Normal Numbers

A story of unrecognized scientific priority

Proved the existence of computable normal numbers.

Turing's Note on Normal Numbers

A story of unrecognized scientific priority

Proved the existence of computable normal numbers.

Gave the first algorithm.

Turing's Note on Normal Numbers

A story of unrecognized scientific priority

Proved the existence of computable normal numbers.

Gave the first algorithm.

In this note Turing showed that classical notions such as measure and continuity admit finite approximations, and they can be made computational. Turing was a pioneer of effective mathematics.

The End

Special acknowledgement to Turing's Digital Archive for the copy of the original manuscript, and the letter from Hardy to Turing.