Benford's law

Haack Christina , Kersch Julie , Van der Weken Lynn

June 1, 2015

Contents

1 Introduction

BENFORD's law, also called the significant-digit law or the first-digit law, refers to the occurrence of leading digits in many real-life sources of data, like the length of rivers, the cost of gas bills, population sizes, and so on.

Simon NEWCOMB, an American mathematician and astronomer, was probably the first one to discover BENFORD's law. In 1881, NEWCOMB noticed that the first pages in the logarithmic tables were more referred to than the last ones. Apparently, people did more calculations using numbers that began with lower digits than with higher ones. NEWCOMB then found a formula that matched his observations quite well.

Frank Albert BENFORD, an American electrical engineer and physicist, rediscovered and generalized this phenomenon in 1938. He collected an enormous set of data, including atomic weights, baseball statistics and street addresses, in order to analyse this unexpected disparity.

One would maybe expect that the distribution of first- digit frequencies for each of the nine possible digits would be one-ninth (11.11%). However, about 30.1% of the numbers will start with 1.

As an example, one can easily see that the first one- hundred Fibonacci numbers have this property:

Fibonacci numbers $F_1, F_2, \ldots, F_{100}$				
1	10946	165580141	2504730781961	37889062373143906
1	17711	267914296	4052739537881	61305790721611591
2	28657	433494437	6557470319842	99194853094755497
3	46368	701408733	10610209857723	160500643816367088
5	75025	1134903170	17167680177565	259695496911122585
8	121393	1836311903	27777890035288	420196140727489673
13	196418	2971215073	44945570212853	679891637638612258
21	317811	4807526976	72723460248141	1100087778366101931
34	514229	7778742049	117669030460994	1779979416004714189
55	832040	12586269025	190392490709135	2880067194370816120
89	1346269	20365011074	308061521170129	4660046610375530309
144	2178309	32951280099	498454011879264	7540113804746346429
233	3524578	53316291173	806515533049393	12200160415121876738
377	5702887	86267571272	1304969544928657	19740274219868223167
610	9227465	139583862445	2111485077978050	31940434634990099905
987	14930352	225851433717	3416454622906707	51680708854858323072
1597	24157817	365435296162	5527939700884757	83621143489848422977
2584	39088169	591286729879	8944394323791464	135301852344706746049
4181	63245986	956722026041	14472334024676221	218922995834555169026
6765	165580141	1548008755920	23416728348467685	354224848179261915075

In contrary, the first one-hundred prime numbers do not follow this law:

Prime numbers $p_1, p_2, \dots, p_{100}$

2	31	73	127	179	233	283	353	419	467
3	37	79	131	181	239	293	359	421	479
5	41	83	137	191	241	307	367	431	487
7	43	89	139	193	251	311	373	433	491
11	47	97	149	197	257	313	379	439	499
13	53	101	151	199	263	317	383	443	503
17	59	103	157	211	269	331	389	449	509
19	61	107	163	223	271	337	397	457	521
23	67	109	167	227	277	347	401	461	523
29	71	113	173	229	281	349	409	463	541

2 Introductive example

Let us first have a look at the probability $P_n(d)$, which is the probability that $d \in \{1, ..., 9\}$ is the first digit of the n first natural numbers ($n \in \mathbb{N}_{>0}$). Fix $n = 9$. Then the probability $P_9(d) = \frac{1}{9}$, $\forall d \in \{1, ..., 9\}$. This is the simplest and clearest case.

Suppose $n > 9$. In this case the probability to have d as a first digit is not constant. Let $1 \leqslant n \leqslant 99$ and observe the case where $d = 1$: $P_1(1) = 1$ and $P_2(1) = \frac{1}{2}$. The probability falls until $n = 9$, where $P_9(1) = \frac{1}{9}$. For $n = 10$ to $n = 19$ the probability rises to $P_{19}(1) = \frac{11}{19}$. The probability falls again for $19 < n \leqslant 99$ to $P_{99}(1) = \frac{11}{99} = \frac{1}{9}$. So one can see that the probability $P_n(1)$ does not settle down . The cases for $d = 2, ..., 9$ are similar. We conclude that $(P_n(d))_{n \in \mathbb{N}_{>0}}$ is a divergent series.

n	1	9	19	99	199	999	1999	9999	19999
$P_n(1)$	$\frac{1}{1}$	$\frac{1}{9}$	$\frac{11}{19}$	$\frac{11}{99}$	$\frac{111}{199}$	$\frac{111}{999}$	$\frac{1111}{1999}$	$\frac{1111}{9999}$	$\frac{11111}{19999}$

n	8	9	89	99	899	999	8999	9999	89999
$P_n(9)$	0	$\frac{1}{9}$	$\frac{1}{89}$	$\frac{11}{99}$	$\frac{11}{899}$	$\frac{111}{999}$	$\frac{111}{8999}$	$\frac{1111}{9999}$	$\frac{1111}{89999}$

By comparing the two tables one can observe that the probability $P_n(1) \geqslant \frac{1}{9}$ and the probability $P_n(9) \leqslant \frac{1}{9}$ for any $n > 9$. So we can conclude that in general $P_n(1) \gg P_n(9)$.

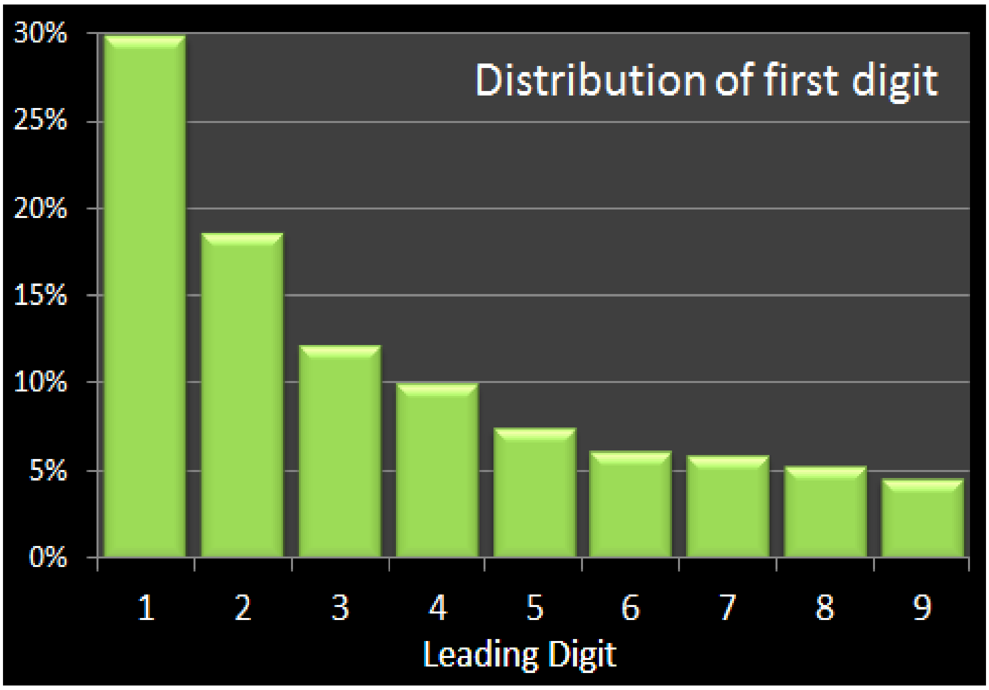
3 The Formula

A set of numbers is said to satisfy BENFORD's law if the leading digit $d \in \{1, ..., 9\}$ occurs with probability:

$$P(d) = \log_{10}(d + 1) - \log_{10}(d) = \log_{10}(\frac{d+1}{d}) = \log_{10}(1 + \frac{1}{d}).$$

The table below shows the probability for all d to be the leading digit:

d	$P(d)$
1	0.3010
2	0.1761
3	0.1249
4	0.0969
5	0.0792
6	0.0669
7	0.0580
8	0.0512
9	0.0458
Sum	1.0000



[1]

[1]Reference[?]

The distribution of first digits, according to Benford's law. Each bar represents a digit, and the height of the bar is the percentage of numbers that start with that digit.

Applying the formula to our example in the introduction:

	d	1	2	3	4	5	6	7	8	9
$\#\{D_1 = d\}$	**Fibonacci**	30	18	13	9	8	6	5	7	4
	Prime	25	19	19	20	8	2	4	2	1
	$10^2 \cdot \log(1+d^{-1})$	30.10	17.60	12.49	9.691	7.918	6.694	5.799	5.115	4.575

4 Some properties of Benford's law

Benford's law needs data that are neither totally random nor overly constrained, but rather lie somewhere in between. In fact, it applies most accurately to data that are distributed smoothly across many orders of magnitude.

4.1 Proper Probability Space

The proper probability space is part of a mathematical explanation of Benford's law by the American mathematician Theodore P. Hill, which delivers a convincing explanation of the recurrence of Benford's law in natural phenomenons.

The definition of the proper probability space is a very important key for the understanding and the study of Benford's law. In the case of constructing such a space, a quickly appearing problem is the fact that the event space is not a Borel set[2]of $\mathbb{R}$. Example: All real numbers, whose mantissae belong to a definite Borel set, are never included in a single interval. So for example the mantissa[3] of every number in $[10, 20)$ is in $[1, 2)$. But, considering the fact that every number in $[1, 2), [100, 200), [1000, 2000), ...$

[2]**Borel sets** are named after the French mathematician Emile Borel (7 January 1871 - 3 February 1956). A **Borel set** is any set in a topological space that can be formed from open sets (or, equivalently, from closed sets) through the operations of countable union, countable intersection, and relative complement.

Borel sets are important in measure theory, since any measure defined on the open sets of a space, or on the closed sets of a space, must also be defined on all Borel sets of that space. Any measure defined on the Borel sets is called a Borel measure. Borel sets and the associated Borel hierarchy also play a fudamental role in descriptive set theory. In some contexts, Borel sets are defined to be generated by the compact sets of the topological space, rather than the open sets. [?]

[3]the decimal part of a logarithm to base 10 as distinguished from the integral part. [?]

has also its mantissa in $[1, 2)$ it becomes evident that a probability cannot be devoted to a single interval if only one suits to the corresponding set of mantissae.

Anyway, by considering the following example it becomes evident that the set of real numbers $\bigcup_{n=-\infty}^{\infty} B \cdot 10^n$ includes all positive numbers whose mantissae belong to B_1, where B is a Borel of $[1, 10)$. An observer might assign a probability at this set by only looking at the mantissae. Consequently for a measurable space combined with a kind of Benford-experiment, where just positive numbers are sampled, an natural definition is $(\mathbb{R}^+, M)$.

The main properties of the event space M, called the mantissa algebra, are explained in the following lemma:

Lemma 1. *1. Every non-empty set in M is infinitive with accumulation points at $+\infty$ and at 0, i.e. in any set $S \subset M$ it is constantly possible to find arbitrary small or arbitrary large non-zero numbers.*

2. M is self-similar in the sense that $\forall m \in N, S \subset M \rightarrow 10^m \cdot S \subset M$.

3. M is closed under scalar multiplication ($\forall s > 0, S \subset M \rightarrow s \cdot S \subset M$).

4.2 Scale invariance

Groups of numbers that follow Benford's law are scale invariant:

Scale invariance means that any universal law should be independent of units. One can measure data using a range of different scales (feet/metres, pounds/dollars, gallons/ millilitres, etc.); so, if the digit frequency law is true for some scale, it must be true for all of them.

So this means that if one multiplies all numbers by an arbitrary constant, then the distribution of first-digit frequencies remains unchanged.

Since one is interested in the distribution of first significant digits, it makes sense to express numbers in scientific notation $x \cdot 10^m$ where $1 \leqslant x < 10$ $m \in \mathbb{Z}$, for all numbers except zero. The first significant digit d is then simply the first digit of x. One can easily derive a scale invariant distribution for d once one has found a scale invariant distribution for x.

If a distribution for x is scale-invariant, then the distribution of $y = \log_{10}(x)$ should remain unchanged when one adds a constant value c to y, because one would be multiplying x by some constant $a \in \mathbb{R}$, and

$$\log_{10}(ax) = \log_{10}(a) + \log_{10}(x) = \log_{10}(a) + y = c + y.$$

Now, the only probability distribution on $y \in [0, 1[$ that will remain unchanged after the addition of an arbitrary constant to y, is the uniform distribution.

So y is uniformly distributed between $\log_{10}(1) = 0$ and $\log_{10}(10) = 1$. In order to find the probability that d is 1, one has to evaluate:

$$P(d = 1) = P(1 \leqslant x < 2) = P(0 \leqslant y < \log_{10}(2))$$

To find this one calculates the integral :

$$\int_0^{\log_{10}(2)} 1dy = \log_{10}(2)$$

which is approximately 0.301. In general:

$$P(d = n) = P(n \leqslant x < n+1) = P(\log_{10}(n) \leqslant y < \log_{10}(n+1)), n \in \{1, ..., 9\}$$

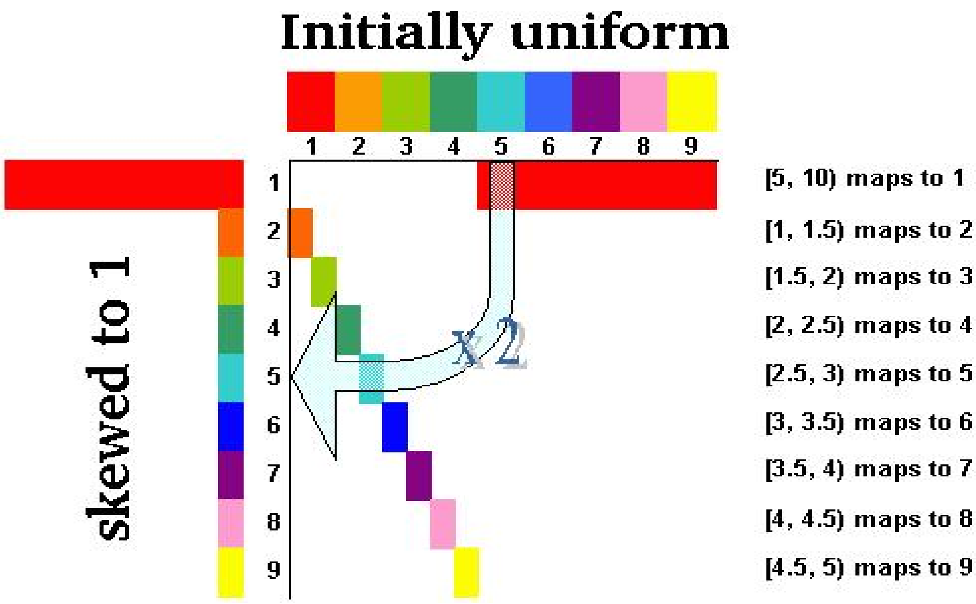
and this is given by

$$\int_{\log 10(n)}^{\log_{10}(n+1)} 1dy = \log_{10}(n+1) - \log_{10}(n) = \log_{10}(\frac{n+1}{n})$$

The expression $\log_{10}(\frac{n+1}{n})$ was exactly the formula given by Newcomb and later by Benford for the proportion of numbers whose first digit is n.

Now, one can show that equally likely digits are not scale invariant:

Proof. Suppose that each of the digits $1, 2, ..., 9$ are equally likely to appear as the first significant digits in any number. If the first significant digit is 1, then multiplying by 2 will yield a new first digit of 2 or 3 with equal probability. But if the first significant digit is 5, 6, $7, 8$ or 9 the new first digit must be 1. It turns out that in the new set of accounts, a first digit of 1 is much more likely than any other first digit, as one can see in the picture below:



[4]

The original uniform distribution is now heavily skewed towards the digit 1. So if the numbers are scale invariant, they will not be uniformly distributed. □

4.3 Base invariance

Base invariance can be defined as a more differential hypothesis that also induces BENFORD's law. In a general significant digit law the basic motivation for a deduction of base invariance is the intuitive assumption that any reasonable kind of such law should be as valid for base 10 as for other integral bases. The idea is that if a certain law should emerge in observing data tests in base 10, then this law should also appear by using another base.

Before being generalized to other bases the definition of mantissa algebra M was used. By that the notation M_b will indicate the mantissa algebra in an uncertain base b (thus $M_{10} = M$). That means: All the theorems, properties and definitions are practically identical, except that b replaces 10. So for example $\log_b$ replaces log in the probability distributions.

Definition 1. *A probability measure P on $(\mathbb{R}^+, M_b)$ is base invariant if*

$$\forall m \in \mathbb{N}, \forall S \in M_b, P(S) = P(S^{\frac{1}{m}})$$

At first sight it might not be very easy to understand why this definition

[4]Reference[?]

has a relevance to base invariance. Therefore we have to consider :

$$S = \bigcup_{q=-\infty}^{\infty} [b^y, b^x) \cdot b^q$$

Whereby $x, y \in [0, 1)$ (representation of any left-closed interval of $[1, b)$ can be represented in the form $[b^y, b^x)$)

4.3.1 Base invariance in data tests

Usually data tests are expressed in 8 different bases from 3 to 10. For investigating bases greater than 10 is usually required a much more complicated computer routine. It should be mentioned that the base 2 also has a special function because in base 2 the first digit is always 1.

The following table shows the Chi-square, the distance and the p-value for the first digits of each of the converted datasets. Here the distance and the Chi-squares [5] don't have the same meaning according the base (there are $b - 1$ classes for the first digit in base b, so they tend to be naturally lower in the low bases).

By taking that into account, the base invariance is verified if the average distance $|f_i - e_i|$ was considered to be the order of 10^{-3} on all the classes and all the bases.

Base	Chi-square	Distance	p-value
3	0.1	$2.7 \cdot 10^{-6}$	74%
4	5.7	$7.1 \cdot 10^{-5}$	6%
5	13.4	$1.4 \cdot 10^{-4}$	0%
6	13.9	$1.3 \cdot 10^{-4}$	1%
7	14.1	$1.1 \cdot 10^{-4}$	1%
8	20.4	$1.9 \cdot 10^{-4}$	0%
9	12.9	$8.1 \cdot 10^{-5}$	8%
10	6.1	$3.5 \cdot 10^{-5}$	64%

In the figure above the Chi-squares seem not to be significant. The excellent fit in base 3 might be due to the fact that there exist only two classes. Altough the p-values[6] are bad as usual, the overall base invariance is verified

[5] In the probability theory and statistics, the **Chi-squared distribution** (also Chi-square or χ -distribution) with k degrees of freedom is the distribution of a sum of squares of k independent standard normal random variables. [**?**]

[6] The p-value is a function of the observed sample results. A statistic that is often used for testing a statistical hypothesis. The so-called significance level of test, traditionally 5% or 1% and denoted as α is a threshold value, which is chosen before performing the

quite well because the distance[7] is considered.

4.4 Sum invariance

Sum invariance means that the sum of all entries with leading digit d is constant for various d. For instance, in a sample of integers that follow the first-digit law, there are lots of 1, a bit less of 2, and so on until 9. So the sum of all the 1s should be roughly equal to that of 2s to that of 3s, etc. One should also consider second and third significant digits, and so on, e.g.: the expected sum of all entries starting with 2.7182 is equal to that of all entries starting with 3.1415.

Informally, one can say that a distribution is sum-invariant if for any natural number n, the expected sum of the mantissae of all entries starting with that fixed n-tuple of significant digits is the same as that for any other n-tuple. Here, on considers the mantissa function

$$M : \mathbb{R}^+ \to [1, 10)$$

such that

$$x \mapsto M(x) = r$$

where r is the unique number in $[1, 10)$ with $x = r \cdot 10^n$ for some n in $\mathbb{Z}$. For example, $M(5) = M(0.05) = 5$.

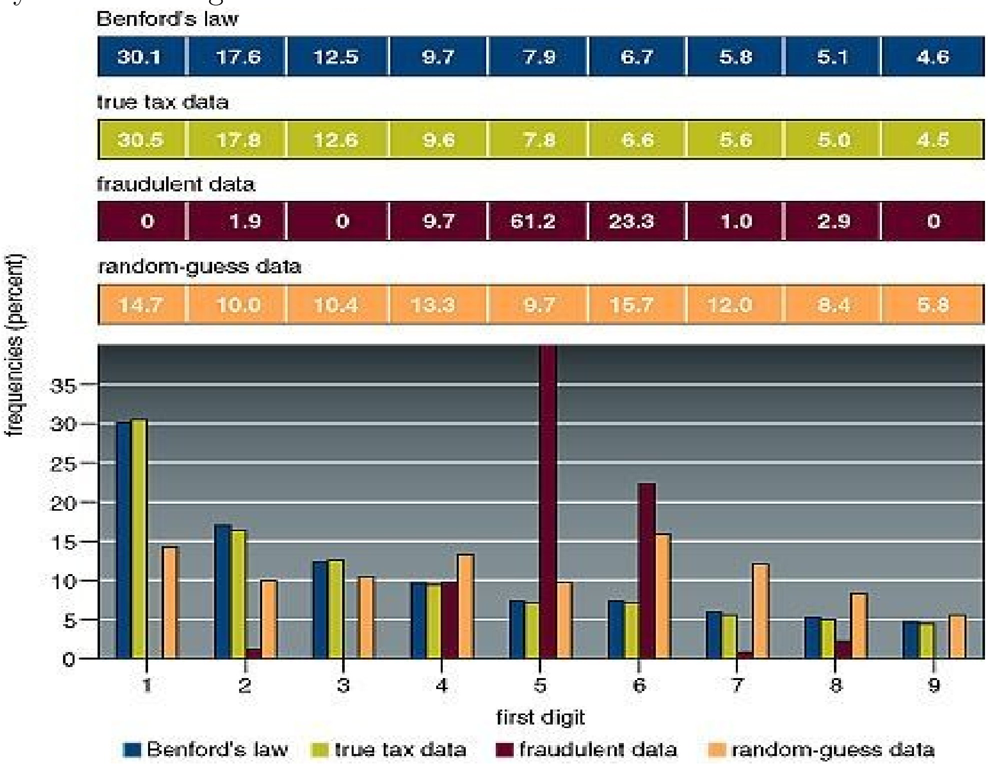
5 Utility of Benford's law

BENFORD's law can be used to detect fraudulent or random-guess data in applications like election campaign finance, toxic gas emission or income tax returns like on the graphic below. Here the first significant digits of true tax data follow BENFORD'S law closely. Fraudulent data taken from a study of cash disbursement and payroll in business do not follow BENFORD's law. Likewise, data taken from the author's study of 743 freshmen's responses to a request to write down a six-digit number at random do not follow the

test. If the p-value is smaller than or equal to the significance level (α), the suggestion can be made that the observed data do not lead to the assumption that the null hypothesis is true. In this case the hypothesis must be rejected and the alternative hypothesis is true. When the p-value is calculated correctly, such a test is guaranteed control for the fact that the Type I error rate is not greater than α. [**?**]

[7] In mathematics, a distance function or metric function generalizes the concept of the physical distance. A metric is a function that concretely describes what consequences it has for elements of some space to be far away from or closed to each other. In most cases, "distance from A to B" is interchangeable with distance between B and A.[**?**]

law. Although these are very specific examples, in general, fraudulent or randomly chosen data appear to have far fewer numbers starting with 1 and many more starting with 6 than do true data.



6 Conclusion

Benford's law was discovered by the American mathematician and astronomer Simon Newcomb in 1881 and was made famous in 1938 by physicist Frank Benford. After observing sets of naturally occurring numbers, Benford discovered as the first number in a list, the phenomenon of a surprising pattern in the occurrence frequency of the digits one through nine. In essence can be said that referring to the frequency distribution of digits in diverse life data, 1 occurs as the leading digit in almost 33% of the time. Oppositely, larger numbers in that position occur less frequently, so 9 (with 4.58%) as the first digit less than 5% of the time.

A summary of the mathematical properties described in the present work could be:

1. Introduction with the context already mentioned above, including tables showing that the fist one-hundred Fibonacci numbers follow Ben-

FORD's law and in contrary the first one-hundred prime numbers do not. (**??**)

2. An introductive example which by comparing these two tables to the conclusion that in general $P_n(1) \gg P_n(9)$. (**??**)

3. Probability formula, graphic and table illustrating the distribution of first digits, according to BENFORD's law. (**??**)

4. As some properties of BENFORD's law are mentioned:

   (a) The proper probability space by Theodore P. HILL. (**??**)

   (b) Scale invariance, including a table and mathematical derivations and proofs showing where the distribution of first digits of a data set is scale invariant (e.g. independent of the units the data are expressed in or not). (**??**)

   (c) Base invariance manifesting the assumption that if a certain law should appear in base 10, then this law should also appear if another base was used (inclusive table and a short explanation why in regular data tests base invariance is expressed in different bases from 3 to 10 in **4.3.1**). (**??**)

   (d) Sum invariance indicating that the sum of all entries with leading digit d is constant for all various d. (**??**)

5. A brief declaration of the worldwide utility of BENFORD's law, respective its use for the detection of random-guess or fraudulent data. (**??**)

Ultimately we have tried to give a short and understandable insight into this leading digit phenomenon. BENFORD's law unquestionably applies to many situations in the real world and is therefore an interesting object of study.

References

[1] http://en.wikipedia.org/wiki/Chi-squared_distribution

[2] http://www.yourdictionary.com/mantissa

[3] http://en.wikipedia.org/wiki/P-value

[4] http://en.wikipedia.org/wiki/Distance

[5] http://digitalcommons.calpoly.edu/cgi/viewcontent.cgi?article=1040&context=rgp_rsr

[6] http://projecteuclid.org/download/pdfview_1/euclid.ps/1311860830

[7] http://wwwf.imperial.ac.uk/~nadams/classificationgroup/Benfords-Law.pdf

[8] http://www.sciencedirect.com/science/article/pii/S0165489614000262

[9] http://www.benfords-law.com/

[10] Jon Walthoe. Looking out for number one

[11] A. Berger and T.P. Hill. A basic theory of Benford's law

[12] Hans Humenberger, Das "Benford-Gesetz Warum ist die Eins als führende Ziffer von Zahlen bevorzugt?, Materialien für einen realitätsbezogenen Mathematikunterricht -Band 6,Wien, 2000

[13] https://plus.maths.org/content/looking-out-number-one

[14] http://en.wikipedia.org/wiki/Borel_set

[15] http://econ.ucsb.edu/~doug/240a/Benford%20Law.html

[16] http://www.math.unt.edu/~allaart/papers/invar.pdf

[17] http://www.rexswain.com/benford.html

7 Annex

7.1 Version 1

```java
package Benfordslaw;

      import java.util.Scanner;

public class Benford2 {

          public static void main(String[] args){

             Scanner scan = new Scanner(System.in);
             int length; // length of the numerical sequence

             System.out.println("Enter the length of the numerical sequence :");
             length = scan.nextInt();
             System.out.println();


              int[] occurences = new int[9]; // frequencies of leading digits
              double[] digits = new double[length]; // sequence of numbers

              System.out.println("Enter a sequence of numbers : ");

              createSequenceOfNumbers(digits,scan);
              System.out.println();
              System.out.println("We will test Benfords law by using the following
sequence of numbers :");
              print(digits);
              System.out.println();


            System.out.println("occurences of each leading digit : ");

              countOccurences(digits, occurences);
              print(occurences);
              System.out.println();

              System.out.println("Histogram:");
              for (int i = 0; i <9 ; i++){
                   System.out.print(i+1 +": ");
                   for (int n =0; n<occurences[i]; n++){

                   System.out.print("*");
              }
              System.out.println();}
              System.out.println();



              System.out.println("Test of Benfords law : ");
              System.out.println();
              test(occurences, digits.length);


          } // print-methods for doubles and integers
```

```
private static void print(int[] array) { //prints an array of integer values
                for (int j = 0; j <array.length; ++j) {
                    System.out.print(array[j] + " ");
                }
                System.out.println();
            }

            private static void print(double[] array) {  // prints an array of double
values



                for (int j = 0; j <array.length; ++j) {
                    System.out.print(array[j] + " ");

                }
                System.out.println();
            }

           public static int leadingDigit(double digit){ // returns the number which is
nearest to the left of digit (computes the leading digit)
               if (digit == 0) {
                   return 0;
               } else if (digit < 0){
                   digit = -digit;
               }
               while (digit < 1){
                   digit *= 10;
               }

               while (digit >= 10){
                   digit  /= 10;
               }

               return (int)digit;
           }

           public static void createSequenceOfNumbers(double[] digits,Scanner x){ //
method creates a sequence of numbers which can be entered by the user

             double inputNumber;

               for (int i = 0; i < digits.length; i++){

                   inputNumber = x.nextDouble(); // input of a number by the user
                   digits[i]=inputNumber;         // each input number will be stored in
an array of type double
               }
           }
```

```
  public static void countOccurences(double[] digits, int[] occurrences) { // for each
number in digits[] this method determines the amount of occurrences where i+1 is a
leading digit in the sequence of numbers
               for (int i = 0; i < digits.length; ++i){                              //
computes an array of counts for the occurrences of each leading digit (0-9)
                   int k = leadingDigit(digits[i]);
                   ++occurrences[k-1];



               }
           }



           public static void test(int[] occurrences, int size) { // this method
compares the frequencies with the predictions of Benfords law
               System.out.println("We compare the frequencies with the predictions of
Benfords law :");
               System.out.println();
               for (int i = 0; i < 9; ++i){
                   System.out.print( (i+1) + " : " + 100*(occurrences[i]/(double)size) +
"%, ");
                   System.out.println( " Benford: "  + Math.log10(1 + 1.0/(i+1))*100 +
"%");
               }
           }



}
```

7.2 Version 2

```
package Benfordslaw;


        import java.io.*;
        import java.util.*;

        public class Benfordslaw {

           public static void main(String[] args) {

                 Scanner scan;
          try{

              scan = new Scanner(new File("fibonnacci.txt"));

              analyze(countFrequencies(scan));

           }catch(FileNotFoundException e){

                 e.printStackTrace();
           }
           }

           public static int leadingDigit(int number) { // returns the leading digit (leftmost digit) of a
number
              int leadingDigit = Math.abs(number);
              while (leadingDigit >= 10) {
                 leadingDigit = leadingDigit / 10;
              }
              return leadingDigit;
          }

           public static int firstDigitOf(String s) { // returns the first nonzero digit of a string
                                                 // returns 0 if the string does not contain any digits
                 char c;

                 for (int i = 0;i<s.length();i++){
                 c=s.charAt(i);
                 if(Character.isDigit(c)&&(c >= '1' && c <= '9')){


                     return c - '0';
              }
                 }
                 return 0;
           }
```

```
public static int[] countFrequencies(Scanner s) { // reads in integers from a text file
      int [] frequencies = new int[10];             // counts the frequencies of   each leading
digit (from 0 to 9)
   int digit = 0 ;
   while (s.hasNext()) {

      digit = firstDigitOf(s.next());
      frequencies[leadingDigit(digit)]++;
   }
   return frequencies;
}

public static int computeSum(int[] numbers) { // returns the sum of integers of a given
array
   int sum = 0;
   for (int i = 0; i<numbers.length; i++) {
      sum = sum + numbers[i];
   }
   return sum;
}




public static void analyze(int[] frequencies) { // prints frequency distribution of each
leading digit
   System.out.println();

   int total = computeSum(frequencies) - frequencies[0];
   double percent;
   System.out.println("Digit Count Percent");
   for (int i = 1; i < frequencies.length; i++) {
      percent = frequencies[i] * 100.0 / total;
      System.out.printf("%5d %5d %6.2f\n", i, frequencies[i], percent);

   }
```

```
    System.out.printf("Total %5d %6.2f\n", total, 100.0);
            System.out.println();
            System.out.println("predictions by Benfords Law :"); // comparison with the
predictions of Benfords Law
            for(int i = 1;i<frequencies.length;i++){

              System.out.println(i+" :  "+Math.log10(1 + 1.0/(i+1))*100+" %");

            }
          }


}
```