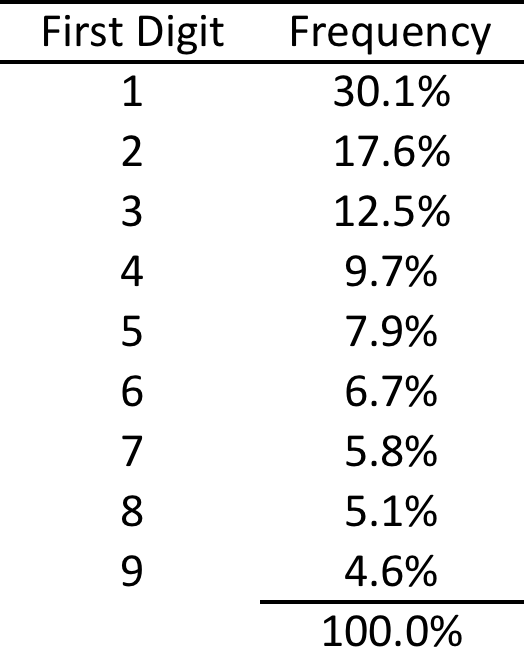
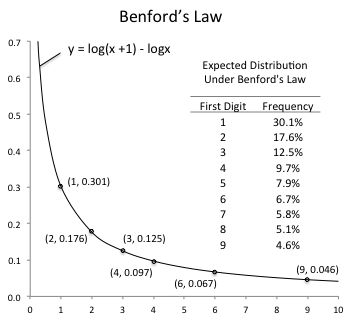
Benford’s Law

**1. Background.** Benford’s Law, also called the First-Digit Law, refers to the frequency distribution of digits in many (but not all) real-life sources of data. In this distribution, 1 occurs as the leading digit about 30% of the time, while larger digits occur in that position less frequently, for example 9 as the first digit occurs less than 5% of the time.

This result has been found to apply to a wide variety of data sets, including electricity bills, street addresses, stock prices, population numbers, death rates, lengths of rivers, physical and mathematical constants, and processes described by power laws, which are very common in nature. It tends to be most accurate when values are distributed across multiple orders of magnitude.

The discovery of Benford’s Law goes back to 1881, when the American astronomer Simon Newcomb noticed that in logarithm tables used at that time to perform calculations the earlier pages, which contained numbers that started with 1, were much more worn than the other pages. Newcomb’s published result is the first known instance of this observation and includes a distribution on the second digit as well. Newcomb proposed a law that the probability of a single number N being the first digit of a number was equal to log(N + 1) – logN.



The phenomenon was again noted in 1938 by the physicist Frank Benford, who tested it on data from 20 different domains. His data set included the surface area of 335 rivers, the sizes of 3259 US populations, 104 physical constants, 1800 molecular weights, 500 entries from a mathematical handbook, 30 numbers contained in an issue of Reader’s Digest, the street addresses of the first 342 persons listed in American Men of Science, and 418 death rates. The total number of observations used in the paper was 20,229. Newcomb’s observation became known as Benford’s Law.

Benford’s Law, although widely known, continues to defy easy explanation. If we were to use a random number generator to generate integers in the range 100 to 999, and then look at how often the numbers begin with 1, how often with 2, and so on, we would expect to see each digit about one-ninth (11.1%) of the time. But with real world data, we see a very different distribution.

To explore this phenomenon, let’s compare two sequences of numbers, one that grows linearly and one that grows geometrically. Starting with the number 1 and adding 0.2 to it repeatedly produces this linear sequence:

1.0, 1.2, 1.4, 1.6, 1.8, 2.0, 2.2, 2.4, 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 3.8, . . . 9.0, 9.2, 9.4, 9.6, 9.8

In this sequence there are five numbers that begin with 1, five numbers that begin with 2, five numbers that begin with 3, and so on.

But consider what happens when we go from 1 to 10 using an exponential sequence instead, *e.g.* starting with 1 and multiplying successive numbers by 1.05 until we get to 10:

1.00, 1.05, 1.10, 1.16, 1.22, 1.28, 1.34, 1.41, 1.48, 1.55, 1.63, 1.71, 1.80, 1.89, 1.98, 2.08, 2.18, 2.29, 2.41, 2.53, 2.65, 2.79, 2.93, 3.07, 3.23, 3.39, 3.56, 3.73, 3.92, 4.12, 4.32, 4.54, 4.76, 5.00, 5.25, 5.52, 5.79, 6.08, 6.39, 6.70, 7.04, 7.39, 7.76, 8.15, 8.56, 8.99, 9.43, 9.91.

In this sequence there are 15 numbers that begin with 1 (31%), 8 numbers that begin with 2 (17%), and so on. There are only 2 numbers that begin with 9 (4%). This distribution of digits is well predicted by Benford’s Law.

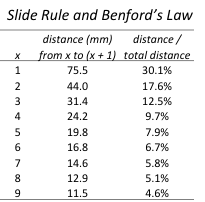
As another example, consider a culture of 1000 bacteria for which the population doubles every day. Every few hours, for 30 days, you count the number of bacteria and write that number on a list. By the end of 30 days, your list of numbers will follow Benford’s Law quite accurately.

Why? It is instructive to take one day at a time.

* On the first day, the number of bacteria is increases from 1000 to 2000. The first digit is 1 for the whole day.
* On the second day, there are 2000 bacteria increasing to 4000. The first digit is 2 for about 14 1/2 hours and 3 for about 9 1/2 hours.
* On the third day, there are 4000 bacteria increasing to 8000. The first digit increases from 4 to 7, with progressively less time per digit.
* The next day, there are 8000 bacteria increasing to 16,000. The leading digit passes rapidly though 8 and 9 in a few hours, but once there are 10,000 bacteria, the first digit will be 1 for a whole 24 hours, until the number of bacteria gets to 20,000.
* Roughly, the leading digit is 1 for one day out of three.

From this simple example, it can be seen that the first digit is 1 with the highest probability, and 9 with the lowest. The longer we continue this experiment, the more closely the distribution of leading digits approaches the Benford distribution.

Another way to think about it: An exponentially growing quantity is equivalent to moving at a constant speed rightward on a log scale. One place to find a log scale is on an old-fashioned slide rule. As shown in the table below, the linear distance from one to two is 30% of the total distance from 1 to 10 on a log scale. In fact, all the leading digits 1-9 are distributed according to Benford’s Law.



On a Post slide rule, the Benford Distribution can in fact be read directly by comparing the D (log) and L (linear) scales. 1 to 2 on the log scale corresponds to 0 to 0.30 on the linear scale, *i.e.* the leading digit is 1 30% of the log scale, and so on. In fact, the slide rule represents Newcomb’s logarithm tables, hence Benford’s Law, *exactly.*

In the absence of a slide rule, a log scale can be visualized by starting with linear scale from 0 to 1 in uniform 0.1 increments. On one side of he scale, label these increments 0, 0.1, 0.2, 0.3 . . . 1.0. On the other side, label the same increments 100 = 1, 100.1 = 1.26, 100.2 = 1.58, 100.3 = 2.00 . . . 101.0  = 10.00. Again, for 0 to 0.3 on the linear scale, the leading digit on the log scale is 1.

For further challenge,

* Analyze the distribution for the second digit.

**2. Write and Test Program.** Write a program that reads a file of integers and shows the distribution of the leading digit. To test your program, create two files containing the linear and exponential sequences described earlier, multiplied by 100 to allow integer arithmetic (convince yourself that multiplying by a constant does not alter the distribution):

* BenfordLinear.txt: 100 to 980 with an additive increment of 20.
* BenfordExponential.txt: 100 to 991 with a multiplicative ratio of 1.05.

Your output should look like this for these two cases (The first two lines in each case are the prompt to the user followed by the response of the user with an input filename.):

*Linear data:*

Let’s count those leading digits...

input file name? BenfordLinear.txt

Digit Count Percent

1 5 11.11

2 5 11.11

3 5 11.11

4 5 11.11

5 5 11.11

6 5 11.11

7 5 11.11

8 5 11.11

9 5 11.11

Total 45 100.00

*Exponential data:*

Let’s count those leading digits...

input file name? BenfordExponential.txt

Digit Count Percent

1 15 31.25

2 8 16.67

3 6 12.50

4 4 8.33

5 4 8.33

6 3 6.25

7 3 6.25

8 3 6.25

9 2 4.17

Total 48 100.00

**3. Locate and analyze at least one real-world data set.** Note that real world data can on occasion contain entries which are zero. **Your program should detect and exclude those cases**, *e.g.* if two zero entries are inserted into your BenfordExponential.txt file, your output should be:

*Exponential data with two zeros:*

input file name? BenfordExponential.txt

excluding 2 zeros

Digit Count Percent

1 15 31.25

2 8 16.67

3 6 12.50

4 4 8.33

5 4 8.33

6 3 6.25

7 3 6.25

8 3 6.25

9 2 4.17

Total 48 100.00

The choice of real-world data is up to you. Benford’s original compilation can be found at: http://mathworld.wolfram.com/BenfordsLaw.html.

**4. Grading Rubric.**

Submission will be HyperGrade and Hardcopy

**HyperGrade**

All test cases pass 10

At least 3 pass 7

Otherwise 5

**Hardcopy (please organize to fit neatly on one page)**

Analysis of real-world data set. Include description and size of data set, cite source, show console output, **add critical comments on agreement or disagreement with Benford’s Law**.

Analysis of real-world data 10

\_\_

Total possible 20 **Homework category**