Benford's Law

Ethan Bolker

May 2015

1 Benford's Law

2 A Python program

```python
# http://rosettacode.org/wiki/Benford%27s_law#Python

from __future__ import division
from itertools import islice, count
from collections import Counter
from math import log10
from random import randint

expected = [log10(1+1/d) for d in range(1,10)]

def fib():
    a,b = 1,1
    while True:
        yield a
        a,b = b,a+b

# powers of 3 as a test sequence
def power_of_threes():
    return (3**k for k in count(0))

def heads(s):
    for a in s: yield int(str(a)[0])

def show_dist(title, s):
    c = Counter(s)
    size = sum(c.values())
    res = [c[d]/size for d in range(1,10)]

    print("\n%s Benfords deviation" % title)
    for r, e in zip(res, expected):
        print("%5.1f%% %5.1f%%  %5.1f%%" % (r*100., e*100., abs(r - e)*100.))

def rand1000():
    while True: yield randint(1,9999)

if __name__ == '__main__':
    show_dist("fibbed", islice(heads(fib()), 1000))
    show_dist("threes", islice(heads(power_of_threes()), 1000))

    # just to show that not all kind-of-random sets behave like that
    show_dist("random", islice(heads(rand1000()), 10000))
```

with its outputn

```
PS C:\eb\python\benford> python .\benford.py

fibbed Benfords deviation
 30.1%  30.1%    0.0%
 17.7%  17.6%    0.1%
 12.5%  12.5%    0.0%
```

```
  9.6%   9.7%    0.1%
  8.0%   7.9%    0.1%
  6.7%   6.7%    0.0%
  5.6%   5.8%    0.2%
  5.3%   5.1%    0.2%
  4.5%   4.6%    0.1%

threes Benfords deviation
 30.0%  30.1%    0.1%
 17.7%  17.6%    0.1%
 12.3%  12.5%    0.2%
  9.8%   9.7%    0.1%
  7.9%   7.9%    0.0%
  6.6%   6.7%    0.1%
  5.9%   5.8%    0.1%
  5.2%   5.1%    0.1%
  4.6%   4.6%    0.0%

random Benfords deviation
 11.2%  30.1%   18.9%
 11.4%  17.6%    6.2%
 10.6%  12.5%    1.9%
 11.1%   9.7%    1.4%
 11.1%   7.9%    3.2%
 11.6%   6.7%    4.9%
 10.8%   5.8%    5.0%
 10.9%   5.1%    5.8%
 11.3%   4.6%    6.7%
```

Python idioms to learn:

- `from x import y`
- `from __future__`
- `a, b = b, a+b`
- `zip`
- `count`
- `Counter`
- `yield`
- `islice`
- formatted output

3 Experiments

- Deviation should be signed.
- Find tables of values online – perhaps wikipedia. Read them and test for Benford.
- Does the table of primes less than N (for large N)follow Benford's law?
- Do the values of the sequence $\lfloor \log(n) \rfloor$ for $n < N$ (for large N) follow Benford's law?
- Does the set of integers less than N (for large N) follow Benford's law?

4 Mathematica

This is the Mathematica implementation from rosettacode.

```
fibdata = Array[First@IntegerDigits@Fibonacci@# &, 1000];
Table[{d, N@Count[fibdata, d]/Length@fibdata, Log10[1. + 1/d]}, {d, 1,
    9}] // Grid
```

I haven't a clue how to read this, and don't think it's a useful exercise in this course to try.

Here is the LaTeX source for this document. You can cut it from the pdf and use it to start your answers. I used the `\jobname` *macro for the source file name, so you can call your file by any name you like.*

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Benford's Law
% Math 480 Spring 2015
%

\documentclass[10pt]{article}
\usepackage[textheight=10in]{geometry}

\usepackage{verbatim}
\usepackage{amsmath}
\usepackage{amsfonts} % to get \mathbb letters

\usepackage[utf8]{inputenc}
\DeclareFixedFont{\ttb}{T1}{txtt}{bx}{n}{9} % for bold
\DeclareFixedFont{\ttm}{T1}{txtt}{m}{n}{9}  % for normal
% Defining colors
\usepackage{color}
\definecolor{deepblue}{rgb}{0,0,0.5}
\definecolor{deepred}{rgb}{0.6,0,0}
\definecolor{deepgreen}{rgb}{0,0.5,0}

\usepackage{listings}

%Python style from
%http://tex.stackexchange.com/questions/199375/problem-with-listings-package-for-python-syntax-color
\newcommand\pythonstyle{\lstset{
  language=Python,
  backgroundcolor=\color{white}, %%%%%%%
  basicstyle=\ttm,
  keywordstyle=\ttb\color{deepblue},
  emph={MyClass,__init__},
  emphstyle=\ttb\color{deepred},
  stringstyle=\color{deepgreen},
  commentstyle=\color{red},  %%%%%%%%
  frame=tb,
  showstringspaces=false,
  numbers=left,numberstyle=\tiny,numbersep =5pt
}}

\usepackage{hyperref}

\begin{document}

\pythonstyle{}

%%%%%%%%%%%%%%%% start here %%%%%%%%%%%%%%%%
\begin{center}
\Large{
Benford's Law \\
Ethan Bolker \\
May 2015
}
\end{center}

\section{Benford's Law}
```

```
\section{A Python program}

\lstinputlisting{benford.py}

with its outputn

\begin{verbatim}
PS C:\eb\python\benford> python .\benford.py

fibbed Benfords deviation
 30.1%  30.1%    0.0%
 17.7%  17.6%    0.1%
 12.5%  12.5%    0.0%
  9.6%   9.7%    0.1%
  8.0%   7.9%    0.1%
  6.7%   6.7%    0.0%
  5.6%   5.8%    0.2%
  5.3%   5.1%    0.2%
  4.5%   4.6%    0.1%

threes Benfords deviation
 30.0%  30.1%    0.1%
 17.7%  17.6%    0.1%
 12.3%  12.5%    0.2%
  9.8%   9.7%    0.1%
  7.9%   7.9%    0.0%
  6.6%   6.7%    0.1%
  5.9%   5.8%    0.1%
  5.2%   5.1%    0.1%
  4.6%   4.6%    0.0%

random Benfords deviation
 11.2%  30.1%   18.9%
 11.4%  17.6%    6.2%
 10.6%  12.5%    1.9%
 11.1%   9.7%    1.4%
 11.1%   7.9%    3.2%
 11.6%   6.7%    4.9%
 10.8%   5.8%    5.0%
 10.9%   5.1%    5.8%
 11.3%   4.6%    6.7%
\end{verbatim}

Python idioms to learn:

\begin{itemize}
\item \lstinline|from x import y|
\item \lstinline|from __future__|
\item \lstinline|a, b = b, a+b|
\item \lstinline|zip|
\item \lstinline|count|
\item \lstinline|Counter|
\item \lstinline|yield|
\item \lstinline|islice|
\item formatted output
\end{itemize}

\section{Experiments}

\begin{itemize}
```

```
\item Deviation should be signed.
\item Find tables of values online -- perhaps wikipedia. Read them and
  test for Benford.
\item Does the table of primes less than $N$ (for large $N) $follow
  Benford's   law?
\item Do the values of the sequence $\lfloor \log(n) \rfloor$ for $n <
  N$ (for large $N$)  follow Benford's law?
\item Does the set of integers less than $N$ (for large $N$) follow
  Benford's law?
\end{itemize}

\section{Mathematica}

This is the Mathematica implementation from rosettacode.

\begin{verbatim}
fibdata = Array[First@IntegerDigits@Fibonacci@# &, 1000];
Table[{d, N@Count[fibdata, d]/Length@fibdata, Log10[1. + 1/d]}, {d, 1,
    9}] // Grid
\end{verbatim}

I haven't a clue how to read this, and don't think it's a useful
exercise in this course to try.

\newpage
\emph{
Here is the \LaTeX{} source for this document. You can cut it from the
pdf and use it to start your answers. I used the} \verb!\jobname!
\emph{macro for the source file name, so you can call your file by any
  name you like.}
\verbatiminput{\jobname}

\end{document}
```