Re: Randomness of digits within pi

Source: http://sci.tech−archive.net/Archive/sci.math/2007−11/msg02332.html

- *From*: David Bernier <david250@xxxxxxxxxxxx>
- *Date*: Sun, 11 Nov 2007 18:22:50 −0500

jonas.thornvall@xxxxxxxxxxx wrote:

> On 11 Nov, 13:48, David Bernier <david...@xxxxxxxxxxxx> wrote:
>
> > David Bernier wrote:
> >
> > > jonas.thornv...@xxxxxxxxxxx wrote:
> > >
> > > > On 9 Nov, 17:58,
> > > > gr...@xxxxxxxxxxxxxxxx (Daniel Grubb)
> > > > wrote:
> > >
> > > [...]
> > >
> > > > > Here's a program that can
> > > > > calculate pi to 33 million
> > > > > places fairly
> > > > > quickly
> > > > > on most modern machines.
> > > > > http://www.geocities.com/hjsmithh/Pi/Super_Pi.html
> > >
> > > > Ok i admit i am being lazy anyone who can
> > > > distribute it as file on
> > > > bittorent?
> > > > Someone must have run the program and
> > > > burned it down?
> > >
> > > You might want to send an e−mail to Xavier Gourdon, the
> > > author
> > > of the program Pifast ...
> > > I used Xavier Gourdon's PiFast43.exe program to compute
> > > pi to 1 billion decimals. It took about 12 to 25 hours on a PC
> > > with an Athlon XP 2200 32−bit processor and 1 GB of
> > > RAM.
> > > The resulting file, at 1.28 GB, can be opened with Notepad
> > > to see the first 50 or 100 lines or so, but it is so large that
> > > it takes minutes to go down 2 pages.
> > > I believe it would it would be easy to access the file
> > > using C or C++ with fopen(), fscanf(), fclose().

> I seem to remember that the file has line numbers,
> line feeds/CR (the Windows CR+LF line breaks) and
> some information about the start and finish of the
> computation at the beginning of the file.
> Cf.:
> <http://numbers.computation.free.fr/Constants/PiProgram/pifast.html>

> I wrote a program in C to read the 1.28 GB file, so as to ignore all
> characters that are spaces, line feeds,
> carriage returns, whitespace, text, "3." and the line numbers (each line
> had 50 digits).
>
> Up to 800 million decimals after the point, I got a distribution of:
> 0s: 79991897
> 1s: 79997003
> 2s: 80003316
> 3s: 79989651
> 4s: 80016073
> 5s: 79996120
> 6s: 80004148
> 7s: 79995109
> 8s: 80002933
> 9s: 80003750
>
> This agrees with Yasumasa Kanada's data here:
> <ftp://pi.super−computing.org/pub/pi/pi.all.freq.3b>

> But i hope you people *of any should understand* that counting
> standalone decimal digits will tell you nothing about any pattern in
> the string?

Well, the frequency of each digit should not stray too far from 10%.
I wanted to know if my 800 million decimals were right. That's
the real reason I counted the number of 0s, 1s, ... 9s in my file
and compared that with Kanada's results.

> It only tell you that each standalone decimal digit is uniform
> distributed, and that really say nothing.
> I know about diehard, and statistic tests, but if you really looking
> for pinning down an anomaly to show that a expanding function do not
> output uniform distribution of output you should look for anomalies of
> longer digit lengths i understand that only around 350 billions or was
> it 3,5 billions values of pi is known.

I think Kanada and his collaborators have passed the 10^12 decimal
digit mark.

Actually, the Diehard battery (for base 10) has "runs tests" where
digits are grouped in threes to represent a number from 0.000 to
0.999 . The base 10 version of the runs tests (and other tests) was done for
10^9 decimals of pi and the results reported in George Marsaglia's
article which he mentioned in this thread :
< http://interstat.statjournals.net/YEAR/2005/articles/0510005.pdf >

So you propose a repeated digits test. It could be done.
But the first 10^9 decimals of pi didn't produce really suspicious results,
from looking at Marsaglia's article.

> I would pick a value of say one 100 000 to 10 millions to look for in
> the data and try to find anomalies, because if you find an anomaly of
> that size it would certainly say something about the function itself.
>
> So why can't you make a search for 111111,222222,... to 999999 and
> 1111111,2222222,7777777... to 9999999 if you already programmed a
> short snippet.

I think Marsaglia's "Monkey tests" are at least as good. But the first
billion digits of pi passed the "Monkey tests".

I have no ideas for new tests. I compressed a one million−byte
file with the first million decimals, and it was 0.2% larger than
the critical value of 1000000* log(10)/log(256) = 415,241.01 bytes.

David Bernier
.