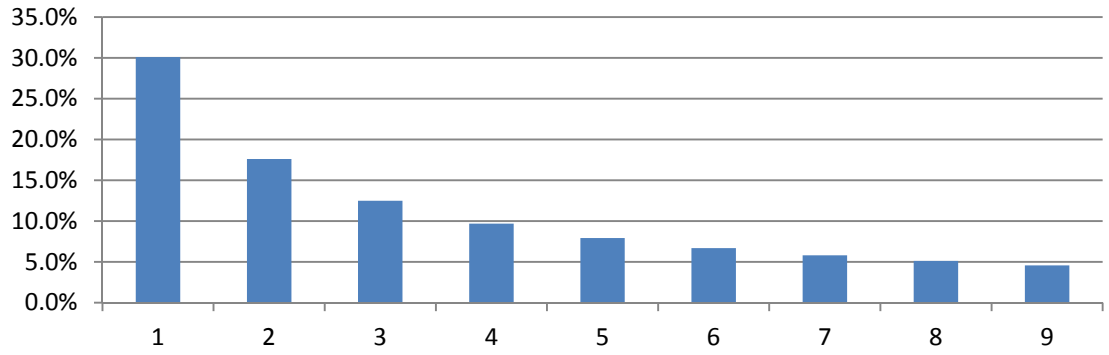
What I Learned This Month: Benford's Law

Scott Chapman
American Electric Power

Consider a set of measurements of natural features, such as the size of lakes in North America. It doesn't matter if you measure them by acre-feet of water stored, surface area in square miles or shoreline in kilometers, just measure them all the same way. You might intuitively expect that if you looked at the first digit of those measurements you'd find that each digit was used about the same number of times. But that would be incorrect. In general what you would find is that more measurements start with the number 1 than any other number. And the number 2 starts more measurements than any of the other remaining numbers. And so on. The distribution will look approximately like this:



It turns out that this phenomenon was first noticed in the late 1800s and was later re-discovered and formally expressed by physicist Frank Benford in the 1930s, and hence it was named Benford's Law. I just learned about it now. It turns out that Benford's Law applies pretty universally to any set of natural measurements that are not artificially constrained and span at least a few orders of magnitude. For example, the population of cities in Ohio probably follows Benford's Law, but the population of voting districts probably doesn't because the voting districts are artificially constructed to contain somewhat similar populations. (Or perhaps to address particular political requirements, but that's beyond the scope of my knowledge.)

So what does this have to do with IT? Well, it turns out that Benford's Law generally applies to all sorts of things outside of just the size of lakes or the distance of stars from Earth. Financial auditors apparently use this to look for financial fraud because, for example, credit card transactions follow Benford's Law. If you go to http://testingbenfordslaw.com/ you will find that it applies to various measurements of IT data as well, such as file sizes in the Linux source tree.

Upon stumbling across this during some recent lunch hour reading, I almost immediately thought: does this apply to my performance data? So I wrote a little SAS macro to do a Benford's analysis on a given set of data for a given value. I did a little testing and found that the things I tested generally followed Benford's

Law. This included things like I/O counts and CPU and elapsed time for batch jobs and CICS transactions. I thought that was kind of cool!

It also turns out that Benford's Law can be extended to an arbitrary number of leading digits and using two (or possibly even three) leading digits is usually more useful than using just a single digit. I've looked at a number of sets of data where the single-digit analysis looks completely benign, but the two digit analysis shows some spikes which may bear further investigation.

I think that is part of the potential value of applying Benford's Law to our performance data—it might encourage you to investigate your applications in ways that might yield new and interesting information. If you have a set of data that generally follows Benford's Law, but there are some spikes around certain values, it may be useful to look into the transactions or jobs or whatever it is that's generating those spikes. Those spikes might not be a problem, or they might point you towards some sort of artificial limit that you were previously unaware of, such as transactions that get cancelled after 30 seconds of elapsed time or something like that.

One thing I have discovered is that you need a reasonably sized set of data that is from a long enough time period to give you sufficient samples, but not so long that the interesting spikes get overwhelmed by all the mundane data.

The beauty of this particular type of pathology detection is that it's not dependent on any sort of history: in theory, you could do this analysis in real-time as transactions are flowing through the system. Again, this sort of analysis doesn't tell you that you really do have a problem, but it might give you a hint that something is amiss. Finding possible problems without any historical context sounds really cool to me!

Another idea I had was that it might be worthwhile to do the analysis on your performance or stress test results. If the transaction response times are not following Benford's Law, that might be an indication that the test had some issues or is somehow not realistic.

Note that I haven't actually used this to find an actual problem or analyze a performance/stress test, but it sounds like a good idea. Of course, it's highly likely that one of my readers has tried this. If you've successfully (or not) used Benford's Law for performance analysis, I'd like to hear from you! Or if you think this is a bad idea, then certainly I'd like to hear about that too. As always, you can reach me via email at sachapman@aep.com.