Lab 5: Testing Benford's Law

The objective of this lab is to:

- Learn about Benford's law and apply it
- Gain some independence in figuring out how to do things with Excel
- Apply the Kolmogorov-Smirnov test to see whether data complies with Benford's law
- Use some simple functions in VBA

Benford's Law

These notes summarize the fine Wikipedia article on Benford's law,

http://en.wikipedia.org/wiki/Benford's_law

"Benford's Law" describes the frequency distribution of the leading digits in many sources of data. It was articulated by Frank Benford in 1938, and had been previously observed by Simon Newcomb in 1881.

A set of numbers is said to satisfy Benford's Law if the leading digit $d \in \{1, 2, \ldots, 9\}$ occurs with probability

$$\Pr[d] = \log_{10}(d+1) - \log_{10} d = \log_{10}\left(1 + \frac{1}{d}\right).$$

Benford noted that this law applies to many sources of data. He is reported to have tested it on several different domains, including the surface area of rivers, the sizes of US populations, numerical values of physical constants, molecular weights, and even numbers contained in an issue of *Reader's Digest.*

In 1972, Hal Varian suggested this law could be used to detect possible fraud in lists of data submitted in support of public planning decisions and, in 1999, Mark Nigrini applied similar ideas to forensic accounting.

1. Obtain the Data Set

We will use financial transactions from the UK Atomic Energy Authority.

Visit the web page

http://www.gov.uk/government/publications/ukaea-financial-transactions-april-2013

and download the CSV file for the **March 2015** financial transactions (click the "Download CSV 361KB" link on the page). An archived version of the file can be found here if there are issues with the site.

Comma-separated values (CSV) files store tabular data in a plain text file, with each line representing a row and each column separated by a comma. Excel is able to read these files when they are properly formatted without any extra steps to import them.

Open the downloaded CSV file in Excel and **immediately save it as an Excel Macro-Enabled Workbook (.xlsm) file.** If we keep it as a CSV file, our formatting, formulas and functions will be lost if we close and reopen Excel (as CSV files only store plain text). So, make sure you are saving your file as XLSM.

2. Add a Column for the Leading Digits

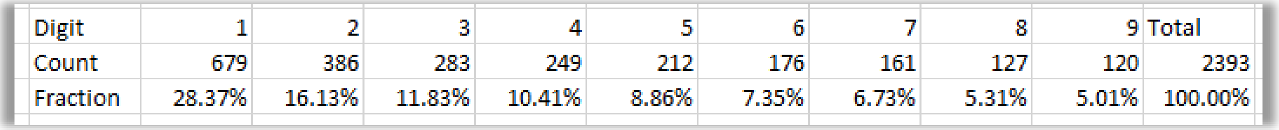
We are interested in the "Amount" values in column H. Add a column, J, that gives the leading digit of the figures in column H. You will want to use a combination of the LEFT function, which gives the left-most character, and the ABS function, which gives the absolute value. Make sure you apply these functions in the correct order. The goal is to remove any minus signs and return only the first digit. If you are unfamiliar with these functions, look up their documentation on the Microsoft Office Support Site: https://support.office.com

There should be at least one figure that is less than 1 and greater than -1, and so Excel writes it as 0.xxx and reports its leading digit to be 0. Change your formula to account for these entries by multiplying each amount by a power of 100 to shift the decimal point before computing the leading digit.

3. Count the Number of Occurrences for Each Leading Digit

Make a table in the spreadsheet that has the numbers 1, 2, … 9 in one row and has the count of the number of times each of these values occurs as a leading digit. You could use 9 different formulas with "COUNTIF", e.g. COUNTIF(…., "=1"), COUNTIF(…., "=2"), etc, but that is grotesque. Instead, use a single formula COUNTIF($…:$…, "=" & …) and copy it to all 9 locations. Replace the … with the correct cell references. You should be able to copy/drag this formula into each column without making any manual changes. ***Why do we need the $ in front of the column letters in this formula?***

Total the numbers and add another row giving the fraction of occurrences with that leading digit. You should now have something like this:



Digit	1	2	3	4	5	6	7	8	9	Total
Count	679	386	283	249	212	176	161	127	120	2393
Fraction	28.37%	16.13%	11.83%	10.41%	8.86%	7.35%	6.73%	5.31%	5.01%	100.00%

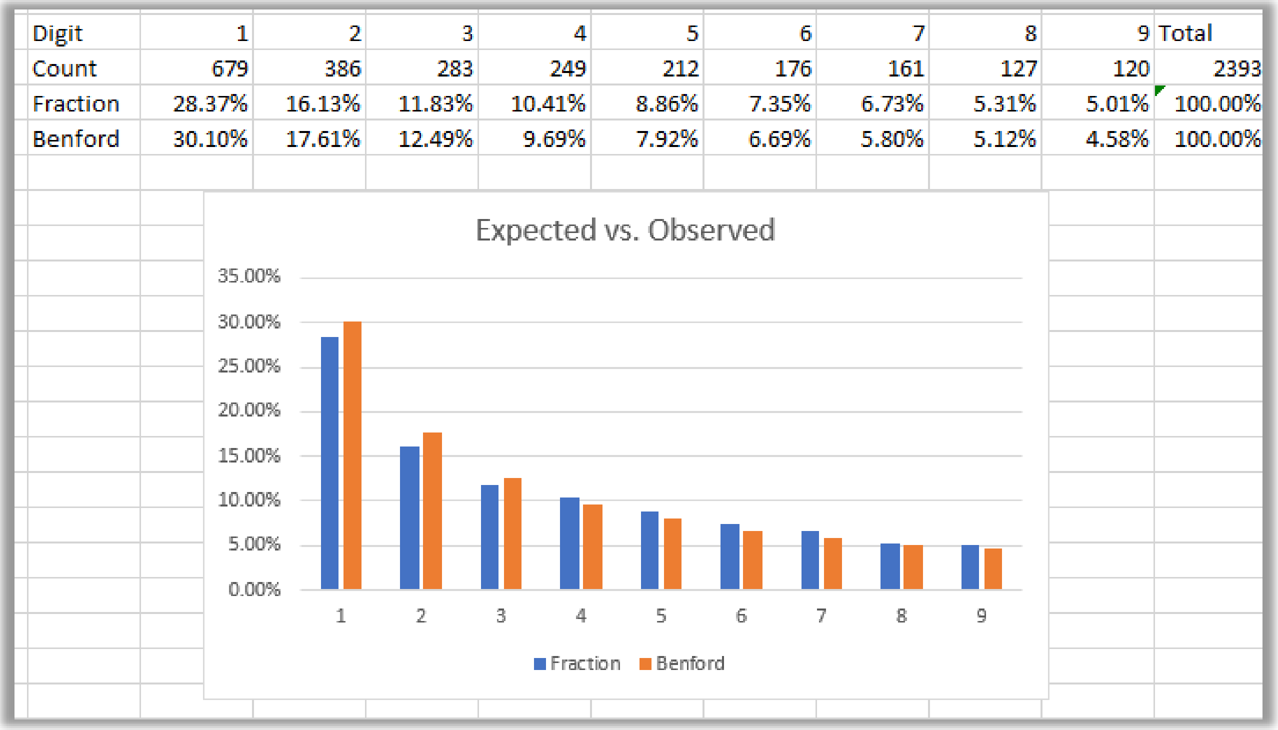
4. Add a Row to Compute the Expected Benford Rate

Add a row below the above table to give the expected fraction, using the equation below:

$$\Pr[d] = \log_{10}(d+1) - \log_{10} d = \log_{10}\left(1+\frac{1}{d}\right).$$

This is not an excel formula, so you must create one based on this formula. Recall that the **d** in this equation is the digit (1 to 9). Your formula should work correctly if pasted/dragged into any of the columns. ***How would you find out how to take the log to the base 10 in Excel if you do not know the function to use?***

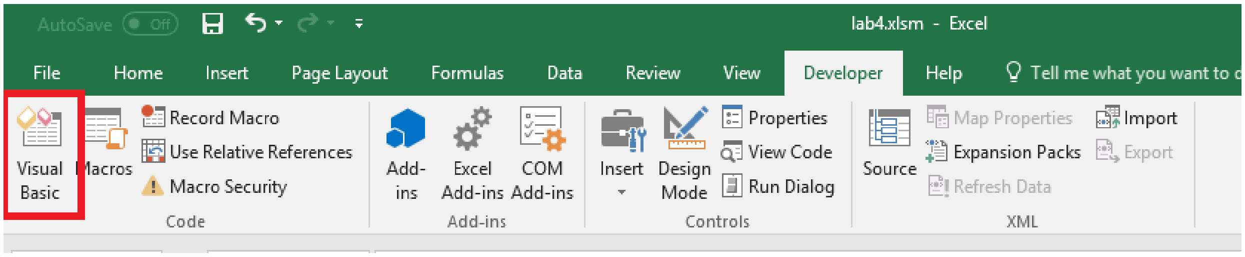
Also, add a bar chart to display the expected Benford rate vs. the observed rate. You should now have something like this:



Digit	1	2	3	4	5	6	7	8	9	Total
Count	679	386	283	249	212	176	161	127	120	2393
Fraction	28.37%	16.13%	11.83%	10.41%	8.86%	7.35%	6.73%	5.31%	5.01%	100.00%
Benford	30.10%	17.61%	12.49%	9.69%	7.92%	6.69%	5.80%	5.12%	4.58%	100.00%

5. Make a Benford and Fraction Function in VBA

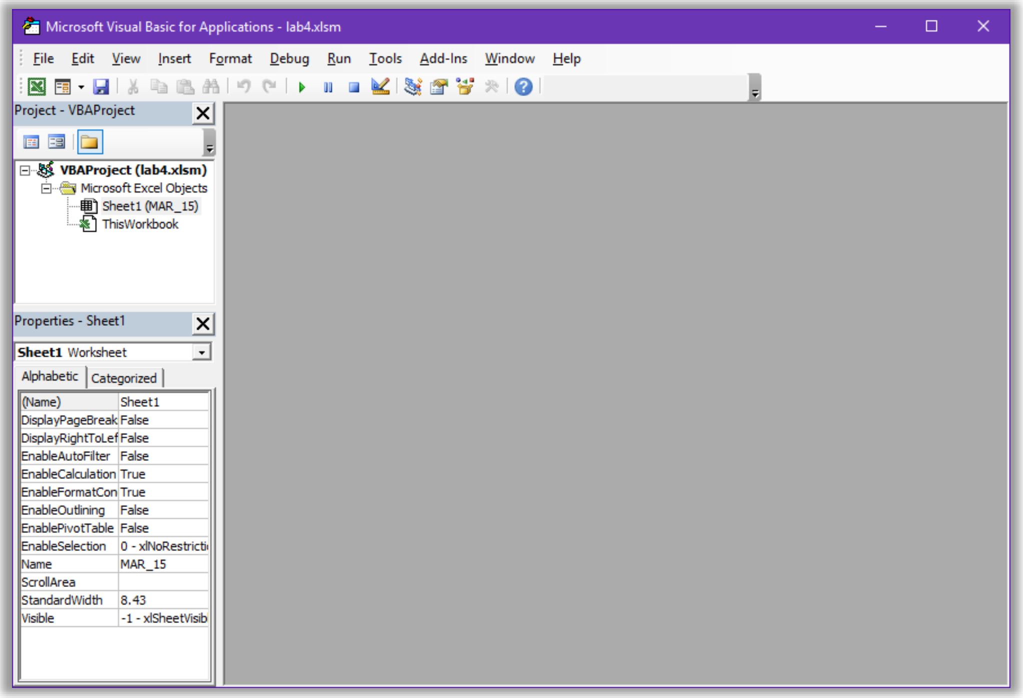
To get some practice creating VBA functions, we will make a VBA function to do fraction and Benford distribution calculations we did in parts 3 and 4. Open the Visual Basics IDE by clicking the follow button in the Developer Tab:



If your Developer tab is missing, following the instructions in the Week 4 lecture notes to expose it, or watch the following video guides (sound is not required):

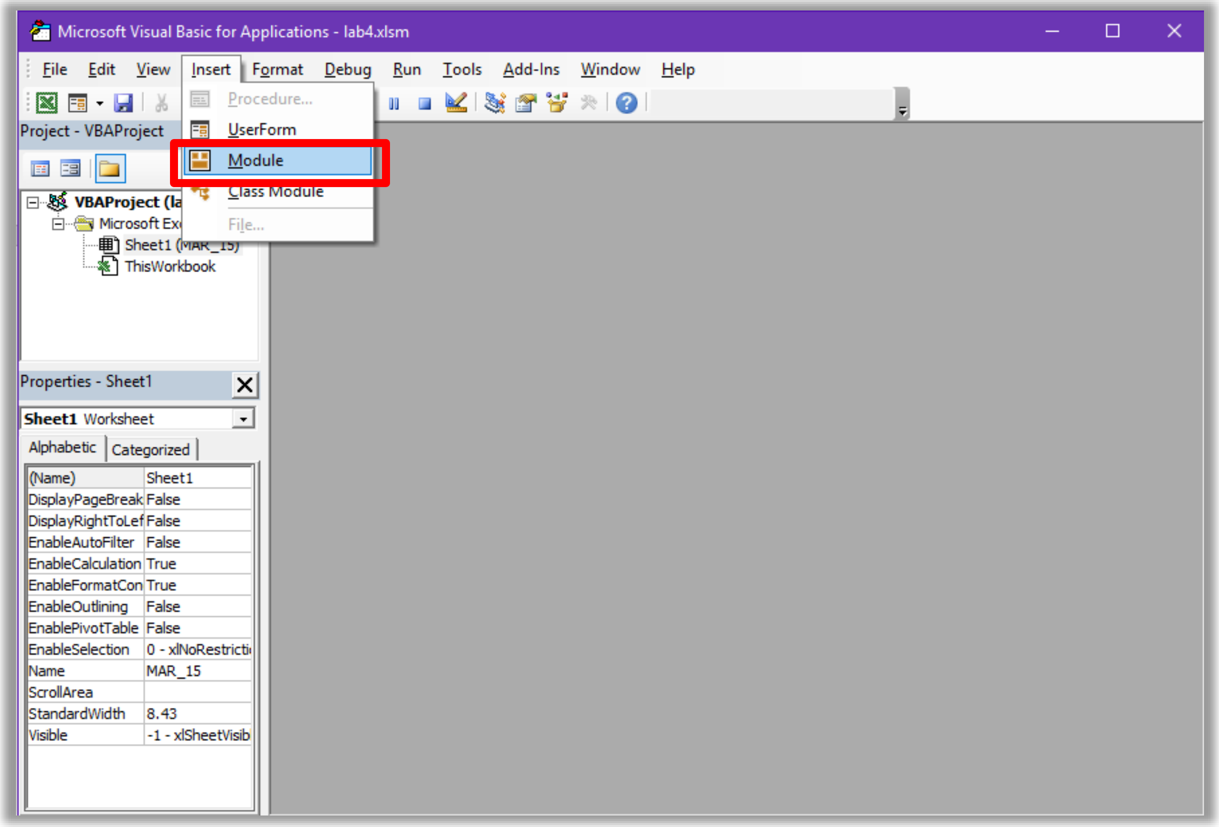
- How to Add the Developer Tab in Excel 2016 for Windows (YouTube Video)
- How to Add the Developer Tab in Excel 2016 for MacOS (YouTube Video)

Once you have the Visual Basics IDE open, it should look something like this:



(Name)	Sheet1
DisplayPageBreak	False
DisplayRightToLef	False
EnableAutoFilter	False
EnableCalculation	True
EnableFormatCon	True
EnableOutlining	False
EnablePivotTable	False
EnableSelection	0 - xlNoRestricti
Name	MAR_15
ScrollArea	
StandardWidth	8.43
Visible	-1 - xlSheetVisib

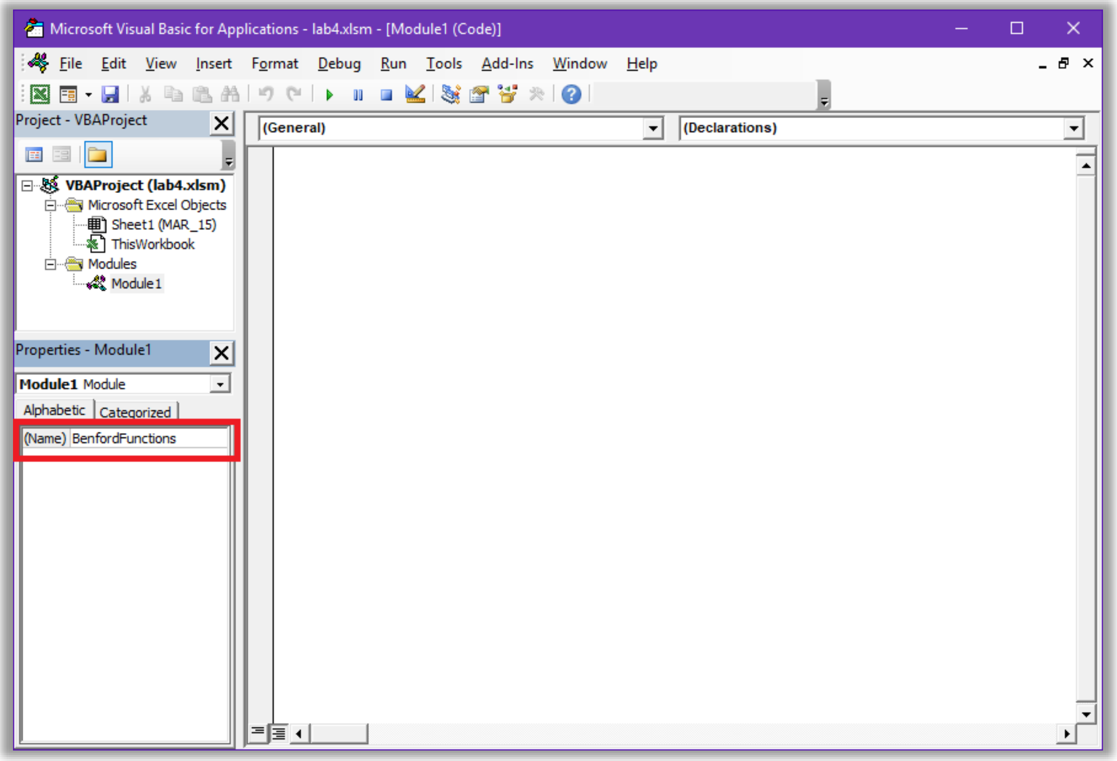
Insert a new module by clicking Insert -> Module



(Name)	Sheet1
DisplayPageBreak	False
DisplayRightToLef	False
EnableAutoFilter	False
EnableCalculation	True
EnableFormatCon	True
EnableOutlining	False
EnablePivotTable	False
EnableSelection	0 - xlNoRestricti
Name	MAR_15
ScrollArea	
StandardWidth	8.43
Visible	-1 - xlSheetVisib

If you are using the MacOS version of Excel, the interface for your Visual Basics IDE might be different.

Once you have a module inserted, rename it to “BenfordFunctions” using the properties window:



(Name) BenfordFunctions

Create two new functions with the following function headers and code:

```
Function fraction(count As Integer, total As Integer) As Double
    fraction = count / total
End Function

Function benford(digit As Integer) As Double
    benford = Log(1 + 1 / digit) / Log(10)
End Function
```

The fraction function takes a **count** from our table in step 3 and the **total** number of amounts from the same table. This function returns the fraction of occurrences based on the given **count** and **total**.

The Bedford function takes a **digit** (1 to 9) and returns the expected Benford distribution for that digit. Note, the Log function used here is the VBA Log function and not the Excel Log function. The VBA Log function is different from the Excel Log function in that it is a base e (natural) logarithm and not base 10 (common logarithm) as we need. To calculate the base 10 logarithm this equation is used:

$$log_b(X) = \frac{log_e(X)}{log_e(b)}$$

In this case, we want the base 10 logarithm, so the equation becomes:

$$log_{10}(X) = \frac{log_e(X)}{log_e(10)}$$

Questions

Record your answers to the following questions on paper or in a text document so you may show the TA later.

1. Read over the code of the **fraction** and **benford** functions and identify each part of the function as one of:
   - Function Keyword
   - Function Name
   - Parameters
     - Parameter Name
     - Parameter Type
   - Function Return Type
   - Function Body
   - End of Function Keyword

   If you are stuck or confused read over pages 29 to 38 in the ***Week 4 Getting Started With VBA*** lecture slides on the Week 4 tab of OWL.

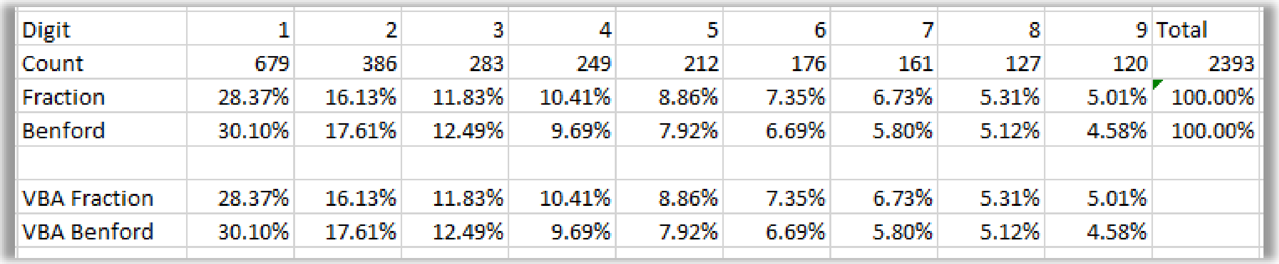
2. In the **fraction** function, why are **count** and **total** parameters **Integers** (whole numbers with no decimal places) while the return of the function is a **Double** (a real number with decimal places)?

3. If we wished to take the base 2 logarithm of the number X but can only use the base e logarithm, what equation could we use? Write the VBA code to calculate the base 2 logarithm of the variable X? You do not have to give a function header, or any extra statements just complete the next line:

   `myLog2Result = ` ______________________________________

6. Use your New Functions

Add two new rows under our table from step 4, that use our **Fraction** and **Benford** functions to calculate the values from step 3 and 4. The result should look like this:



Digit	1	2	3	4	5	6	7	8	9	Total
Count	679	386	283	249	212	176	161	127	120	2393
Fraction	28.37%	16.13%	11.83%	10.41%	8.86%	7.35%	6.73%	5.31%	5.01%	100.00%
Benford	30.10%	17.61%	12.49%	9.69%	7.92%	6.69%	5.80%	5.12%	4.58%	100.00%
VBA Fraction	28.37%	16.13%	11.83%	10.41%	8.86%	7.35%	6.73%	5.31%	5.01%	
VBA Benford	30.10%	17.61%	12.49%	9.69%	7.92%	6.69%	5.80%	5.12%	4.58%	

Ensure the values from your functions are the same as the values you received from your Excel formulas.

7. Use a Statistical Test to See Whether the Distribution is as Expected

Although the table above looks OK, we could ask how bad it must be before we reject it. The solution is to use an appropriate statistical test.

The Kolmogorov-Smirnov test (KS test) may be used to determine whether observations are drawn from some particular distribution. Here, we test the hypothesis that the leading digits follow Benford's law

$$H_0: F(d) = Benford\ Distribution(d)\ for\ all\ d \in \{1,2,\dots,9\}$$

at level $\alpha = 0.001$.

To perform the Kolmogorov-Smirnov test, we do the following:

Step 1. Compute the cumulative fraction of the observations occurring for each value of d.

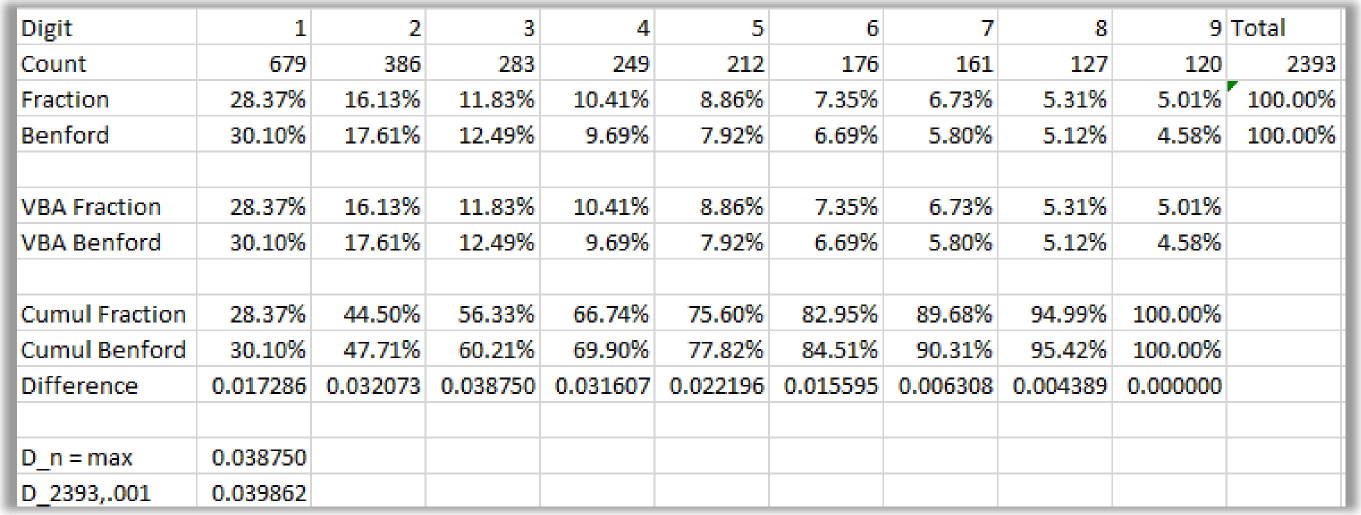
Step 2. Compute the Kolmogorov-Smirnov test statistic, D_n, which is the maximum absolute value difference between the observed cumulative distribution and the hypothesized cumulative distribution.

Step 3. Compare $D_n$with $D_{n,\alpha}$ (see below). If $D_n < D_{n,\alpha}$, then we conclude that we cannot reject H_0.

The Kolmogorov-Smirnov tables are standard and are available from several sources. We are interested in situations where there are many observations ($n > 50$), in which case we can use the following:

α	**0.001**	0.01	0.05	0.10	0.15	0.20
$D_{n,\alpha}$	$\mathbf{1.95/\sqrt{n}}$	$1.63/\sqrt{n}$	$1.36/\sqrt{n}$	$1.22/\sqrt{n}$	$1.14/\sqrt{n}$	$1.07/\sqrt{n}$

Following this procedure, your spreadsheet should appear as below, and since .038750 < .039862, we cannot reject the null hypothesis. That is, the observations are compatible with Benford's law at $\alpha = 0.001$.



Digit	1	2	3	4	5	6	7	8	9	Total
Count	679	386	283	249	212	176	161	127	120	2393
Fraction	28.37%	16.13%	11.83%	10.41%	8.86%	7.35%	6.73%	5.31%	5.01%	100.00%
Benford	30.10%	17.61%	12.49%	9.69%	7.92%	6.69%	5.80%	5.12%	4.58%	100.00%
VBA Fraction	28.37%	16.13%	11.83%	10.41%	8.86%	7.35%	6.73%	5.31%	5.01%	
VBA Benford	30.10%	17.61%	12.49%	9.69%	7.92%	6.69%	5.80%	5.12%	4.58%	
Cumul Fraction	28.37%	44.50%	56.33%	66.74%	75.60%	82.95%	89.68%	94.99%	100.00%	
Cumul Benford	30.10%	47.71%	60.21%	69.90%	77.82%	84.51%	90.31%	95.42%	100.00%	
Difference	0.017286	0.032073	0.038750	0.031607	0.022196	0.015595	0.006308	0.004389	0.000000	
D_n = max	0.038750									
D_2393,.001	0.039862									

If we cannot reject the null hypothesis and the observations seem to be compatible with Benford's law what does this say about the legitimacy of the financial transactions (if anything)?

8. Show the TA Your Work & Optional Tasks

If you have completed the lab or the lab is coming to an end put up your hand and have the TA check over your work.

Optional Tasks (not required for full lab mark)

Up to 2 bonus marks may be awarded for completing **all** optional tasks correctly.

If there is still time remaining in the lab and you wish explore Bedford's law and practice VBA some more, try these following **optional** tasks:

1. Use the website http://www.testingbenfordslaw.com to test Benford's law against several other data sets. Write down at least two data sets that you were most surprised to see follow Benford's law.

2. It would be useful to have a Log function in VBA to find the logarithm of a number in any base. Create such a function, named logx, that takes a number and a base using the formula given in step 5. Update your benford function to use your new logx function rather than the built-in log function.

3. Create a VBA function that calculates the value of $D_{n,\alpha}$ from step 7. Your function should take the sample size n as a parameter as well as the significance level α. Assume that α will always be a value in the table from step 7. You will likely need to use some IF statements and the SQR function to make this function.