USING ANALYTICAL PROCEDURES FOR CONTINUOUS AUDITING AND FRAUD DETECTION

Pamela Jerskey, CISA, CFE
Boston College

This study describes various sampling methods to apply during an audit engagement or for continuous auditing. Various software applications can be used to detect auditor-specified exceptions from a large population. The following analysis introduces an analytical approach which flags current invoices based on a prior period distribution reasonableness model.

Directed Sampling

Directed sampling is typically used when performing audits requiring a detailed review of supporting documentation to test whether a control can be relied on. Typically the sample is selected from a representative population using an appropriate error rate and confidence level. A random sample ensures that all items in the population have a non-biased chance of being selected. The sampling risk is determined by the size of the sample. The larger the sample, the smaller the sampling risk. For instance, in a standard expense audit, the auditor would select a sample of documents to review for certain control criteria. Given a population of 5,000 invoices for one year, the auditor would randomly select 85 invoices to review. This sample, which would fall within a 90% confidence level and 3% error rate, is applied to the entire population.

Another directed sampling approach is to stratify the population for a given period of time. For example, using frequency analysis, we can determine how many invoices fall within certain ranges (i.e., less than $500; ge $500 and lt $5,000; ge $5,000 and lt $20,000, etc.). Depending on how many invoices fall within particular categories, we may decide only to review invoices that fall between $5,000 and $20,000. Within this range of values, a random sample could be selected. In another test, an auditor may decide to use the same initial population, summarize the information by vendor, and only examine documentation for the five highest paid vendors. An auditor could also run a program that looks for duplicate payments and decide to examine all documentation that fit this sampling technique.

Additional directed sampling techniques use Digital Analysis[1] . These programs, designed by Mark Nigrini, Ph.D., are based on statistical theory developed by a physicist (Benford's Law) to determine abnormal recurrence of digits, digit combinations, and specific numbers.

[1] The Use of Benford's Law as an Aid in Analytical Procedures, Mark J. Nigrini and Linda J. Mittermaier, Auditing: A Journal of Practice & Theory, Vol. 16, No.2, Fall 1997.

Analytical Sampling

In contrast, analytical sampling procedures typically look for interrelationships and patterns of certain variables. Various software can be employed for data analysis. In the programs described below, we use SAS® (SAS Institute, Inc.). Various analytical methods can be used to determine which data to reviewed for: (1) compliance with policy; (2) abnormalities; or (3) inefficiencies. For instance, data mining tools apply statistical and artificial intelligence techniques (i.e., algorithms) to discover patterns and irregularities in large volumes of data. Data mining techniques can be used for (1) predictions such as spotting fraudulent data; (2) estimations such as department spending patterns; (3) pattern analysis such as associations, links and groups of items; and (4) segmentation analysis such as group similarities.

The Reasonableness Model

We propose a supplementary approach or alternate testing method which models the distribution of amounts by some appropriate distribution which we shall call "the reasonableness model" (RM). Two distributions used in RM are the normal distribution and the log-normal distribution. For instance, the reasonableness model could look at current invoice amounts by a vendor, account number, or commodity and compare that amount with prior periods (i.e., quarterly, yearly, etc.) to determine if the invoice amount is out of an expected range of values.

Specifically, we first summarized prior period [quarterly, yearly, etc.] amounts that belong to the same vendor [account number, subcode, etc.]$\{\mathbf{a_i}\}$. We then compute its mean **M** and standard deviation **Std** by vendor [account number, subcode, etc.]. Mean **M** is the average of all the variables. Standard deviation **S** is a measure of how much the data vary from the mean **M** (see Figure 1). This gives us a basis for comparing any value **V** [current year amount] to see how likely it is to be consistent with the given data set $\{\mathbf{a_i}\}$ [prior period amounts].

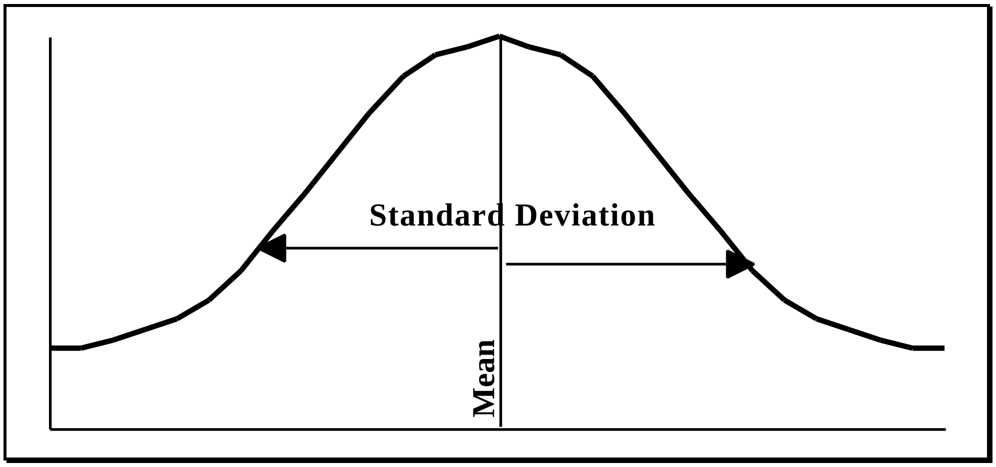

Figure 1

To flag any suspiciously large value **V** [current year amount] or suspiciously small value **V** , we consider **V > M+ (k*Std)** or **V < M - (k*Std)**, where **k** is a parameter used to determine the trade-off between sensitivity and false alarm rate. (For example, **k**=2 or **k**=3 are reasonable choices.) Under RM, the higher the value of **k** the lower the probability of finding an amount differing from the mean **M** by more than **(k*Std)** and therefore the more suspicious that amount.

Consider the following example. Payments made to Vendor XYZ for 1996 are: $25, $50, $65, $75, $400, $500, and $800. If we use **k**=3, Table 1 shows the computed values for 1996:

	1996 Data	
Sum = $\{a_i\}$	1915	
Mean	273.6	
Std	299.7	
3*Std	899.2	
M + (3*Std)	1173	Upper Limit of Sensitivity Rate
M - (3*Std)	-626	Lower Limit of Sensitivity Rate

Table 1

Consider next year (1997) payments made to Vendor XYZ are $2, $5, $50, $100, $500, $1000, $3000, $6000, and $8000. If we apply the formula **V > M+ (3*Std)** or **V < M - (3*Std)**, $3000, $6000, and $8000 are flagged for review since they fall outside the sensitivity rate (see Table 2).

1997 Data Exceptions
Normal-Dist
2
5
50
100
500
1000
3000
6000
8000

Table 2

Perhaps more appropriate for the data under consideration is the log-normal distribution (See Table 3) because all the data is positive and it is more scale-invariant. Before computing the mean, we also excluded ten percent of the outlying amounts which might skew the data. Then we compute the mean **M** and standard deviation **Std** using as our data set the values **$\{\log a_i\}$** . The test for an improbably high value **V** is then whether **log V > M + (k*Std)** and the test for an improbably low value **V** is whether **log V < M - (k*Std)**.

1996 Data			1997 Data	
$	Log		$	Log
25	1.398		2	0.301
50	1.699		5	0.699
65	1.813		50	1.699
75	1.875		100	2.000
400	2.602		500	2.699
500	2.699		1000	3.000
800	2.903		3000	3.477
			6000	3.778
			8000	3.903

Table 3

If we use **k**=3, Table 4 shows the computed values for 1996 using the log function:

	1996 Data	
Sum = $\{a_i\}$	14.99	
Mean	2.141	
Std	0.582	
3*Std	1.745	
M + (3*Std)	3.887	Upper Limit of Sensitivity Rate
M - (3*Std)	0.396	Lower Limit of Sensitivity Rate

Table 4

Consider next year (1997) payments made to Vendor XYZ are $2, $5, $50, $100, $500, $1000, $3000, $6000, and $8000. If we apply the formula **log V > M+ (3*Std)** or **log V < M - (3*Std)**, $2 (log=0.301) and $8000 (log=3.903) are flagged for review since they fall outside the sensitivity rate. The following table shows the comparisons:

1997 Data Exceptions	
Normal-Dist	Log-Dist
2	0.301
5	0.699
50	1.699
100	2.000
500	2.699
1000	3.000
3000	3.477
6000	3.778
8000	3.903

Table 5

Since **k** is an arbitrary number defined by the auditor to determine the trade-off between sensitivity and false alarm rate, there may be other exceptions that are not captured when the assigned **k** value is too high. Conversely, assigning a **k** value that is too low may produce too many exceptions. Therefore, we revised the algorithm to solve for **k**. Solving for **k** also lets the auditor review the range of **k** values, allowing the auditor to select for the highest **k** values to determine the exceptions that are worth following up on. For the next example, we will continue to use the log values as described in Table 2 and 3 and apply the formula **k = abs(M - log V) / Std.** If standard deviation is zero, which means that all the prior period numbers are equal, all current invoices should be flagged. To do this, we assigned the value of .0001 to Std to enable the corresponding invoice to be flagged.

Again, consider 1997 payments made to Vendor XYZ: $2, $5, $50, $100, $500, $1000, $3000, $6000, and $8000. The following values: $2 (k=3.163), $5 (k=2.479), $6000 (k=2.814) and $8000 (k=3.028) are the highest **k** values and should probably be reviewed. The following table shows the comparisons when calculating normal distribution, log-normal distribution and solving for the k value when using logs:

1997 Data Exceptions		
Normal-Dist	Log-Dist	k value
2	0.301	3.163
5	0.699	2.479
50	1.699	0.760
100	2.000	0.243
500	2.699	0.959
1000	3.000	1.476
3000	3.477	2.296
6000	3.778	2.814
8000	3.903	3.028

Table 6

In actual testing, we found many instances where there was only one invoice assigned to a vendor for the prior period. As a result, the mean equaled the amount of the invoice. Also, no standard deviation was calculated because applying the formula to compute the standard deviation for one sample yields an infinite value. A practical application would be to test vendors where there were at least 30 invoices for the prior period. To fully deploy the program, it is best to use as a base, values which span several years. If there is a requirement to test vendors where there is only one invoice for the prior period, we recommend writing another program to look at these exceptions because statistically, the larger the amount of prior data, the more reliable the reasonableness model will be.

To test current invoices against instances where there is only one invoice for the prior period, the program must be modified to check for this condition. When this condition occurs, set the standard deviation to equal the mean. Therefore, when standard deviation is equal to the mean, the resulting **k** value will vary depending on how large or small the prior value was. For instance, consider 1997 payments made to Vendor XYZ: $2, $5, $50, $100, $500, $1000, $3000, $6000, and $8000. If in the prior year only <u>one</u> invoice was paid to Vendor XYZ for $2 or $500 or $8000, the mean and standard deviation would equal the invoice amount. The following table shows results when solving for the k value (using logs) if there are a range of invoices in the prior year, as compared to solving for the k value (using logs) if there is only one invoice in the prior year:

		k value - no Std		
Inv Amt	k value w/Std	Inv=$2	Inv=$500	Inv=$8000
2	3.163	0.000	0.888	0.923
5	2.479	1.322	0.741	0.821
50	0.760	4.644	0.371	0.565
100	0.243	5.644	0.259	0.488
500	0.959	7.966	0.000	0.309
1000	1.476	8.966	0.112	0.231
3000	2.296	10.551	0.288	0.109
6000	2.814	11.551	0.400	0.032
8000	3.028	11.966	0.446	0.000

Table 7

The above assumptions can be changed to limit your flagged output. For instance, you could set standard deviation to equal the mean times an arbitrary value (< 1.00) to increase the sensitivity rate.

Another assumption that we made in the application is that if there are no prior period amounts paid to a particular vendor, there are no amounts flagged for the current period. This situation would occur, for instance, if there was a new vendor. Since the purpose of the program is to evaluate current amounts against prior history, there would be no reason to flag a new vendor.

After testing the program, we decided to eliminate several kinds of payments from our population since they fell under alternate approval structures within the University. For instance, we did not evaluate agency accounts because the college does not have direct spending control over these accounts. We also did not evaluate student refunds as they are automatically generated with approvals from the Financial Aid Department. Several other types of payments we eliminated were grants, capital expenditures, and student refunds since these also require several layers of approvals.

A future application of the reasonableness model is to apply it to "purchasing card" expenditures since there are minimal controls associated with this process. The purchasing card is used to purchase goods and services with a value of less than $1,000 for each item. The purchasing card concept is designed to delegate authority to purchase approved commodities directly from merchants similar to a credit card. The statement is automatically debited from University accounts by the bank. The card holder receives a statement and is responsible for reviewing the charges. Since Internal Audit will be reviewing these expenditures on an annual basis, we determined that we needed a methodology that would enable us to review any purchases that seemed "out of the ordinary". Since the purchasing card is deemed as one vendor and one subcode, we decided that different criteria are needed to determine sampling. Using the methodology described above, we will design the program to look at detailed transactions by merchant name and transaction amount. Since implementation of the purchasing card is a recent activity, we will continue to refine and modify the program to allow us to sample purchases.

I would like to acknowledge Professor R. J. Williams, Ph.D. from the College of Computer Science at Northeastern University for his advice and review.