Computer-assisted Auditing for High-Volume Medical Coding

by Daniel T. Heinze, PhD; Peter Feller, MS; Jerry McCorkle, BA; and Mark Morsch, MS

Abstract

The volume of documents being processed by computer-assisted coding (CAC) has raised the bar regarding the need for audit methods suitable for production control and quality assurance. In this high-volume production environment, it becomes vitally important to adapt and implement techniques that have become a fundamental requirement for production operations management (POM). We present techniques and statistical methods that are developed and implemented for auditing the medical coding process and producing scores that accurately reflect the quality of the coding work, are comparable across time and between coders and auditors, and employ statistical methods for production control. The techniques and methods here described are patent pending and are implemented in the A-Life Medical, Inc. CoAudit™ system that is commercially available for auditing both computerized and human coding.

Introduction

The advent of CAC in high-volume environments demands the use of modern statistical production control and QA methods. Traditionally, coding has been done "manually" by human coders. Because the volume of medical documents being manually coded at any one location has been relatively small, quality assurance (QA) has primarily depended on the individual skills, training, and continuing education of the coders. In the field of medical coding, QA methods historically consist of an ad hoc review of some fixed number or percentage of the coders' work product with ad hoc or subjective scoring and evaluation of audit results. Audit results across time and between coders are, therefore, not mathematically comparable. Additionally, these methods do not scale to high-volume processing.

Although some aspects of QA and production control can be handled automatically, there is still the need for human audit of the coding work product. However, coding is a complex matter, and for some significant percentage of medical documents there will be a measurable diversity of opinion as to how they ought correctly to be coded. Further, the process is sufficiently complex that even the auditors are expected to make errors, though presumably at a lower error level than the coders or processes that are being audited. Consideration for both matters of opinion (subjective judgment) and error must be taken into account when devising a medical coding audit methodology.

Background

Faced with the problem of providing clients using the Actus® CAC application with a means to audit codes and perform production control, we embarked on a process of first soliciting client input and then developing consensus on a specification limit methodology for scoring around which we developed and/or applied the necessary methodologies for the following research issues.

Research Questions

Beginning from the description of how coding professionals audit one another, we address the following issues:

1. *Sample Selection*: Calculating the sample size for audits
2. *Specification and Control Limits*: Establishing and interpreting
    a. *Specification limits* that measure the acceptability of individual coded documents with a method for audit scoring that produces results suitable for incorporation in statistical QA and production control, but which are also designed so that the composite sample scores track the subjective judgment of human auditors when evaluating a computerized or human coding process to be acceptable, marginally acceptable, or unacceptable
    b. *Control limits* that measure the acceptability of a computerized or human coder
3. *Calibrating for Auditor Variability*: Adjusting the statistical methods to calibrate for auditor subjectivity and error so audit results can be meaningfully compared across time and between auditors, coders, and CAC

Methods

Corresponding to the research questions, the following methods are employed.

Sample Selection. Sample selection is governed first by identifying the population from which the sample will be drawn, second by applying some statistical method to determine the sample size, and third by selecting a random sample of the determined size from the population.

For purposes of code auditing, the population must be selected in accord with the objectives of the audit. Audit objectives should first be specified in terms of the target computerized or human coder to be audited with the population being then limited to codes produced by the target. The population may further be stratified according to subcharacteristics such as particular providers, procedures, or diagnoses. It may further be necessary to temporally limit the population to some period when the codes and guidelines for the audit objective were uniform.

Given a target population, the sample size must be calculated. We accept the sample size calculation employed by the Office of the Inspector General (OIG) and as described and implemented in the audit tool Rat-Stats as canonical for code audit purposes.[1] Considering the Rat-Stats function for Attribute Sample Size selection, we note that although the calculated sample size is guaranteed to be minimal, the confidence interval may be asymmetrical around the point estimate. With no harm to the accuracy or validity of the audit, we use a more basic calculation of sample size as given by many introductory statistics texts and also at the National Institute of Standards and Technology, resulting in a symmetric confidence interval at the possible cost of a slightly larger sample size.[2]

Specification and Control Limits. Two sets of performance limits are defined, the specification limits and the control limits. The specification limits are with respect to individual components of the production items under test. These can be judged as either correct or incorrect (pass/fail), and if incorrect (fail) then, optionally, as either of consequence or not of consequence. The control limits are the statistically defined limits that indicate whether the overall coding process under audit is in control or not. For the medical coding application, only the upper control limit is of true interest in that there is no adverse consequence if the process, as measured in terms of proportion of errors, falls below the lower control limit (in fact that is a good thing and indicates that the process is performing better than required or expected).

Formulas—Sample Size and Control Limits

The following defines the parameters and formulas for selecting an unrestricted random sample $fpc \cdot n$ from a population of size N. Defect number x is recalculated to provide X which is the defect number modified to account for the expected subjectivity and error of the auditor according to the formula $X = x - (CV \cdot P \cdot fpc \cdot n)$. The rational for this formula is that if the error level of the auditor is CV and

the auditee is expected to make proportion P errors, then the number of correct auditee codes that were incorrectly judged to be errors by the auditor is $CV \cdot P \cdot fpc \cdot n$, which should be subtracted from the raw defect number x.

CV is the expected or observed judgment subjectivity/error proportion of the auditor.
CL is the desired confidence level as a percent where $CL \le 100 \cdot (1 - CV)$ is preferred.
Z is the area under the tails of the distribution for the desired CL.
H is the half width of the desired confidence interval where $H \ge (CV/2)$.
$H \ge (CV/2) + 0.005$ is preferred.
P is the expected auditee proportion of errors.
N is the size of the population of documents to be sampled.
n is the unadjusted sample size where $n = (Z^2 \cdot P \cdot (1-P)) / H^2$.
fpc is the finite population correction factor where $fpc = \sqrt{(N-n)/(N-1)}$.
$fpc \cdot n$ is the finite population adjusted sample size.
x is the observed defect/error number.
X is the defect/error number adjusted for the auditor error rate where $X = x - (CV \cdot P \cdot fpc \cdot n)$.
e is the sample proportion of defects where $e = x / fpc \cdot n$.
E is the adjusted sample proportion of defects where $E = X / fpc \cdot n$.
UCL is the upper control limit where $UCL = P + Z \cdot \sqrt{P \cdot (1-P) / fpc \cdot n}$.
LCL is the lower control limit where $LCL = P - Z \cdot \sqrt{P \cdot (1-P) / fpc \cdot n}$.

Specific matters of guidance in using the formulae are:

1. CV, the expected or observed auditor subjectivity and error, conforms to $CV \ge 0.03$.
2. The half-width H of the desired confidence interval should be greater than $CV/2$, the error proportion of the auditor, i.e. no matter how large the sample, we cannot be more confident of our audit results than we are of our auditor. Increasing the sample size, which is the practical effect of decreasing H, will not truly improve precision once $H = (CV/2)$. $H \ge (CV/2) + 0.005$ is recommended.
3. $CL \ge 100 \cdot (1 - CV)$ because, similar to H, we cannot expect to achieve a confidence level in the audit that is greater than the maximum accuracy that the auditor can achieve.

Formulas—Specification Limits

In the CoAudit implementation diagnoses and findings are coded using the U.S. Department of Health and Human Services International Classification of Diseases, 9th Clinical Modification (ICD-9-CM), and procedures and level of service are coded using the American Medical Association Current Procedural Terminology (CPT). Other coding systems may be substituted. The following defines the core scoring method for the audit coding of individual documents:

1. Diagnosis and findings codes each receive a weight of 1 and are judged as correct or incorrect (pass/fail).
2. Diagnosis and findings codes may further be judged as of consequence or of no consequence.
3. Procedure and level of service codes each receive a weight of 2 and are judged as correct or incorrect (pass/fail).
4. The modifier codes associated with a procedure or level of service code each have a weight of 1 and are judged as correct or incorrect (pass/fail). Note that modifier codes are optional,

and so there may correctly be no modifier codes and so no modifier code score for any given procedure or level of service code.

5. All modifier code scores are considered to be of consequence.
6. The relational links between diagnosis or findings codes and procedure or level of service codes whereby a particular diagnosis or findings code is indicated as the support for particular procedure or level of service code each receive a weight of 1 and are judged as correct or incorrect (pass/fail).
7. All procedure and level of service codes must be linked to at least one diagnosis or findings code.
8. Links are all judged to be of consequence.
9. The ranked order in which procedure and level of service codes appears relative to other procedure codes and/or the level of service code receives a weight of 1 and is judged as correct or incorrect (pass/fail).
10. In the preferred implementation, ranked order of the procedure and level of service codes is always judged to be of consequence.
11. The unit value of a procedure code receives a weight of 1 and is judged correct or incorrect (pass/fail).
12. The unit value of a procedure code is always judged to be of consequence.
13. The document score $d = 100 - (ModCnt / TotCnt) \cdot 100$ where:

$$ModCnt = \sum_{i=1}^{\max(yc,yo)} ECPTpos_i + ECPTcode_i + ECPTu_i + ECPTm_i + ECPTl_i + \sum_{j=1}^{\max(zc,zo)} EICDcode_j \cdot ICDc_j$$

$$TotCnt = \sum_{i=1}^{\max(yc,yo)} wCPTpos_i + wCPTcode_i + (wCPTu_i \cdot CPTu_i) + (wCPTm_i \cdot CPTm_i) + (wCPTl_i \cdot \max(CPTlc_i, CPTlo_i) + \sum_{j=1}^{\max(zc,zo)} wICDcode_i \cdot ICDc_i$$

yc is the number of post-audit procedure and/or level of service codes in the document
zc is the number of post-audit diagnosis and/or findings codes in the document
yo is the number of pre-audit procedure and/or level of service codes in the document
zo is the number of pre-audit diagnosis and/or findings codes in the document
$CPTu = 1$ if procedure code has units, else 0
$CPTm = 1$ if procedure code has modifier, else 0
$CPTlc =$ the post-audit number of links for the procedure code
$CPTlo =$ the pre-audit number of links for the procedure code
$ECPTl$ is the difference between the $\max(CPTlc, CPTlo)$ and the number of links that are identical (link to the same ICD-9 code) both pre-audit and post-audit
$ECPTpos = wCPTpos$ if post-audit rank order position of procedure code $\neq$ pre-audit position, else 0
$ECPTcode = wCPTcode$ if post-audit code $\neq$ pre-audit code, else 0
$ECPTu = wCPTu$ if post-audit unit $\neq$ pre-audit unit, else 0
$ECPTm = wCPTm$ if post-audit modifier $\neq$ pre-audit modifier, else 0
$EICDcode = wICDcode$ if post-audit code $\neq$ pre-audit code, else 0
$wCPTpos = 1$, the weight for procedure rank order
$wCPTcode = 2$, the weight for a procedure or level of service code
$wCPTu = 1$, the weight for a procedure unit
$wCPTm = 1$, the weight for a procedure modifier

$wCPTl = 1$, the weight for a procedure link
$wICDcode = 1$, the weight for a diagnosis or findings code
$ICDc = 1$ if the diagnosis or findings code audit change is of consequence, else 0

1. The sample score $s = \sum_{i=1}^{fpc \cdot n} d_i \; / \; fpc \cdot n$
2. The defect level $x = (s \cdot \; fpc \cdot n) / 100$

Calibrating for Auditor Variability

An initial CV can be established by making an educated estimate of the auditor's accuracy, but auditors should be periodically tested to provide a benchmark CV value. Without this calibration, audit results across time and between auditors will not be meaningfully comparable. The objective of the testing is to track the CV value of each auditor across time using standardized benchmark tests. The benchmark test consists of a set of coded documents for the auditor to audit. The benchmark test must conform to three principles:

1. From one test session to the next, a significant portion of the test (at least 50 percent in the preferred implementation) must consist of the same documents with the same codes as were present on the previous test. The remaining documents will be new. In the preferred implementation, the order of the documents from test to test will be randomized.
2. Over time, the test documents must be selected so as to reflect the distribution of encounter and document types that coders would be expected to work with under actual production conditions.
3. Test sessions must be separated by sufficient time and test size must be sufficiently large in order that auditors would not reasonably be expected to remember a significant percentage of their edits from one test session to the next.

Auditor scores on the benchmark tests consist of two parts. First, determine CV as calculated on the recurring documents from one test session to the next. Second, the relative variances between auditors who take the same test are calculated and may be used as a cross-check on the intra-auditor CV variance.

Results and Discussion

The CoAudit methodology is designed to correlate to the qualitative judgment of human auditors who may judge a coding process, as defined by the sample selection parameters, to be acceptable, marginally acceptable, or unacceptable. As such, the results of an audit are meaningful primarily when represented as a time series against the system control limits as in an X-bar chart. Because standards of acceptability may vary with time and between organizations, empirical tests must be performed periodically for calibration purposes, but $P = 0.1$, $H = 0.02$ and $CV = 0.03$ are recommended starting parameters. A process may have sample scores that are consistently in control (acceptable), occasionally out of control (marginally acceptable), or consistently out of control (unacceptable). As a starting point, monthly audits are recommended with more than two sample scores out of control in a year being considered unacceptable and requiring intervention to bring the system back in control. Statistical significance tests (e.g. χ -square) can be used to measure the effectiveness of interventions. At the time of writing, alpha tests on six coders indicate that the initial objectives have been met, but minor adjustments are expected as a result of beta testing.

Daniel Heinze, PhD, is the Chief Technology Officer of A-Life Medical, Inc. in San Diego, CA.

Peter Feller, MS, is a Senior Software Engineer at A-Life Medical, Inc. in San Diego, CA.

Jerry McCorkle, BA, is the Vice President of Client Services & Operations of A-Life Medical, Inc. in San Diego, CA.

Mark Morsch, MS, is the Vice President of NLP & Software Engineering at A-Life Medical, Inc. in San Diego, CA.

Notes

1. Department of Health and Human Services – Office of Inspector General Office of Audit Services. *Rat-Stats Companion Manual*. September 2001. Available online at http://oig.hhs.gov/organization/OAS/ratstats/ratstatsmanual.pdf, last accessed July 18, 2006.

2. National Institute of Standards and Technology. *Engineering Statistics Handbook*. June 2003. Available online at http://www.itl.nist.gov/div898/handbook/, last accessed July 18, 2006.