Software Quality Engineering:

Testing, Quality Assurance, and Quantifiable Improvement

Jeff Tian, tian@engr.smu.edu
www.engr.smu.edu/~tian/SQEbook

Chapter 21. Risk Identification for Quantifiable Quality Improvement

- Basic Ideas and Concepts

- Traditional Statistical Techniques

- Newer/More Effective Techniques

- Tree-Based Analysis of ODC Data

Risk Identification: Why?

- Observations and empirical evidences:
  - ▷ 80:20 rule: non-uniform distribution:
    - – 20% of the modules/parts/etc. contribute to
    - – 80% of the defects/effort/etc.
  - ▷ implication: non-uniform attention
    - – risk identification
    - – risk management/resolution
- Risk Identification in SQE:
  - ▷ 80:20 rule as implicit hypothesis
  - ▷ focus: techniques and applications

Risk Identification: How?

- Qualitative and subjective techniques:
  - ▷ Causal analysis
  - ▷ Delphi and other subjective methods

- Traditional statistical techniques:
  - ▷ Correlation analysis
  - ▷ Regression models:
    – linear, non-linear, logistic, etc.

- Newer (more effective) techniques:
  - ▷ Statistical: PCA, DA, TBM
  - ▷ AI-based: NN, OSR
  - ▷ Focus of our Chapter.

Risk Identification: Where?

- 80% or target:
  - ▷ Mostly quality or defect (most of our examples also)
  - ▷ Effort and other external metrics
  - ▷ Typically directly related to goal
  - ▷ Resultant improvement

- 20% or contributor:
  - ▷ 20%: risk identification!
  - ▷ Understand the link
  - ▷ Control the contributor:
    - – corrections/defect removal/etc.
    - – future planning/improvement
    - – remedial vs preventive actions

Traditional Technique: Correlation

- Terminology:
  - ▷ r.v.: random variables
  - ▷ i.v.: independent (random) variable
    - also called predictor (variable)
  - ▷ d.v.: dependent (random) variable
    - also called response (variable)
  - ▷ observations and distribution

- Statistical distributions:
  - ▷ 1d: normal, exponential, binomial, etc.
  - ▷ 2d: independent vs. correlated
  - ▷ covariance, correlation (coefficient)

Traditional Technique: Correlation

- Correlation coefficient:
  - ▷ ranges between -1 and 1
  - ▷ positive: move in same direction
  - ▷ negative: move in opposite direction
  - ▷ 0: not correlated (independent)

- Correlation analysis:
  - ▷ use correlation coefficient
  - ▷ linear (Pearson) correlation vs. non-parametric (Spearman) correlation
  - ▷ based on measurement type/distribution:
    - – non-normal distribution
    - – ordinal measurement etc.

Traditional Technique: Correlation

- Correlation analysis: applications
  - ▷ understand general relationship
    – e.g., complexity-defect correlation
  - ▷ risk identification also
  - ▷ cross validation (metrics etc.)

- Correlation analysis: assessment
  - ▷ only partially successful
  - ▷ low correlation, then what?
  - ▷ data skew: 0-defect example
  - ▷ uniform treatment of data

⇒ Other risk identification techniques needed.

Traditional Technique: Regression

- Regression models:
  - ▷ as generalized correlation analysis
  - ▷ n i.v. combined to predict 1 d.v.
  - ▷ forms of prediction formula
    $\Rightarrow$ diff. types of regression models

- Types of regression models:
  - ▷ linear: linear function

$$y = \alpha_0 + \alpha_1 x_1 + ... + \alpha_n x_n + \epsilon$$

  - ▷ log-linear: linear after log-transformation
  - ▷ non-linear: non-linear function
  - ▷ logistic: represent presence/absence of categorical variables

Traditional Technique: Regression

- Regression analysis: applications
  - ▷ similar to correlation analysis
  - ▷ multiple attribute data

- Regression analysis: assessment
  - ▷ only partially successful
  - ▷ similar to correlation analysis
  - ▷ often marginally better (R-sqr vs c.c.)
  - ▷ same kind of problems
  - ▷ data transformation problem
  - ▷ synthesized metrics $\sim$ regression model?

$\Rightarrow$ Other risk identification techniques needed.

New Techniques

- New statistical techniques:
  - ▷ PCA: principal component analysis
  - ▷ DA: discriminant analysis
  - ▷ TBM: tree-based modeling
- AI-based new techniques:
  - ▷ NN: artificial neural networks.
  - ▷ OSR: optimal set reduction.
  - ▷ Abductive-reasoning, etc.
- Focus of our Chapter.

New Techniques: PCA & DA

- Not really new techniques,
  but rather new applications in SE.

- PCA: principal component analysis
  - ▷ Idea of linear transformation.
  - ▷ PCA to reduce dimensionality.
  - ▷ Effectively combined with DA and other techniques (NN later).

- DA: discriminant analysis
  - ▷ Discriminant function
  - ▷ Risk id as a classification problem
  - ▷ Combine with other techniques

New Techniques: PCA & DA

- PCA: why?
  - ▷ Correlated i.v.'s $\Rightarrow$ unstable models
  - ▷ Extreme case:
    linearly dependent $\Rightarrow$ singularity
  - ▷ linear transformation (PCA) $\Rightarrow$
    uncorrelated PCs (or domain metrics)

- PCA: how?
  - ▷ Covariance matrix: Σ
  - ▷ Solve $|\Sigma - \Lambda| = 0$ to obtain eigenvalues λ_j along the diagonal for the diagonal matrix Λ
  - ▷ λ_j's in decreasing value
  - ▷ Decomposition: $\Sigma = C^T \Lambda C$
  - ▷ C: matrix of eigenvectors
    (transformation used)

New Techniques: PCA & DA

- Obtaining PCA results:
  - ▷ Transformation: $D = ZT$, where
    - – Z is the original data matrix
    - – T is the transformation matrix
  - ▷ Λ, C, T calculated by various statistical packages/tools

- PCA result interpretation/usage:
  - ▷ Eigenvalues $\approx$ explained variance.
  - ▷ First few (3-5) principal components (PCs) explain most of the variance.
  - ▷ Uncorrelated PCs
    $\Rightarrow$ good/stable (linear/other) models

- PCA example: Table 21.1 (p.357)

New Techniques: PCA & DA

- DA: how?
  - ▷ Define discriminant function.
  - ▷ Classify into G_1 and G_2
    - G_1: not fault-prune
    - G_2: fault-prune
  - ▷ Definitions: Section 21.3.1 (p.357).
  - ▷ Other/similar definitions possible.
  - ▷ Minimize misclassification rate in model fitting and in prediction.
  - ▷ Good results (Khoshgoftaar et al., 1996).

- PCA&DA: Summary and Observations:
  - ▷ Positive/encouraging results, but,
  - ▷ Much processing/transformation needed.
  - ▷ Much statistics knowledge.
  - ▷ Difficulty in data/result interpretation.

New Technique: NN

- NN or ANN: artificial neural networks
  - ▷ Inspired by biological computation
  - ▷ Neuron: basic computational unit
    – different functions
  - ▷ Connection: neural network
  - ▷ Input/output/hidden layers

- NN applications:
  - ▷ AI and AI problem solving
  - ▷ In SQE: defect/risk identification

New Technique: NN

- Computation at a neuron: 2 stages
  - ▷ Weighted sum of input: $h = \sum_{1}^{n} x_i$
    (may include constant)
  - ▷ Then activation function $y = g(h)$
    – threshold, piecewise-linear,
    – Gaussian, sigmoid (below), etc.
    $$y = \frac{1}{1 + e^{-\beta x}}$$
  - ▷ Illustration: Fig 21.1 (p.358)

- Overall computation:
  - ▷ Layers of neurons
  - ▷ Input layer: raw data feed
  - ▷ Other layers: computation at n neurons
  - ▷ Objective: minimize prediction error at the output layer

New Technique: NN

- NN algorithm: backward propagation
  - ▷ Fig 21.2 (p.359)
    (actually algorithm ideas, not exact)
  - ▷ Trace through steps
  - ▷ Error: deviance (sum of error sqr)

- NN study (Khoshgoftaar and Szabo, 1996):
  - ▷ Table 21.2 (p.359)
  - ▷ NN superior to linear regression.
  - ▷ NN+PCA superior to NN on raw data.

New Technique: TBM

- TBM: tree-based modeling
  - ▷ Similar to decision trees
  - ▷ But data-based (derived from data)
  - ▷ Preserves tree advantages:
    - – easy to understand/interpret
    - – both numerical and categorical data
    - – partition ⇒ non-uniform treatment

- TBM applications:
  - ▷ Main: defect analysis
    TBDMs (tree-based defect models)
  - ▷ Past: psychology, SE-Amadeus, etc.
  - ▷ Reliability: TBRMs (Ch.22)

- TBM: both risk identification and characterization.

New Technique: TBM

- TBM for risk identification:
  - ▷ Assumption (in traditional techniques):
    - – linear relation
    - – uniformly valid result
  - ▷ Reality of defect distribution:
    - – isolated pocket
    - – different types of metrics
    - – correlation/dependency in metrics
    - – qualitative differences
  - ▷ Need new risk id. techniques.

- TBM for risk characterization:
  - ▷ Identified, then what?
  - ▷ Result interpretation.
  - ▷ Remedial/corrective actions.
  - ▷ Extrapolation to new product/release.
  - ▷ TBDMs appropriate.

New Technique: TBM

- TBDMs: tree-based defect models using tree-based modeling (TBM) technique

- Decision trees:
  - ▷ multiple/multi-stage decisions
  - ▷ may be context-sensitive
  - ▷ natural to the decision process
  - ▷ applications in many problems
    - – decision making & problem solving
    - – decision analysis/optimization

- Tree-based models:
  - ▷ reverse process of decision trees
  - ▷ data $\Rightarrow$ tree
  - ▷ idea of decision extraction
  - ▷ generalization of “decision”

New Technique: TBM

- Technique: tree-based modeling
  - ▷ Tree: nodes=data-set, edges=decision.
  - ▷ Data attributes:
    - – 1 response & n predictor variables.
  - ▷ Construction: recursive partitioning.
  - ▷ Usage: relating response to predictors
    - – $Y = Tree(X_1, \ldots, X_n)$
    - – understanding vs. predicting
    - – identification and characterization
  - ▷ Works for mixed-types of data.
  - ▷ Tree growing and pruning.

- Algorithm: Fig 21.3 (p.360)
  - ▷ regression tree and example
  - ▷ classification tree: modify Step 3

New Technique: TBM

- TBDM example: Fig 21.4 (p.361)
  - ▷ IBM-NS: a commercial product.
  - ▷ 11 design/size/complexity metrics.
  - ▷ High-risk subsets: nodes rll and rr
    – characterization: Table 21.3 (p.361)
  - ▷ Design and control complexity as main predictors of high-risk.

- Key "selling" points:
  - ▷ intuitiveness and interpretation
    – compare to PCA, NN
  - ▷ quantitative & qualitative info.
  - ▷ hierarchy/importance/organization

New Technique: OSR

- OSR: optimal set reduction
  - ▷ pattern matching idea
  - ▷ clusters and cluster analysis
  - ▷ similar to TBM but different in:
    - – pattern extraction vs. partition
- OSR: technique
  - ▷ pattern extraction
  - ▷ algorithm sketch: Fig 21.5 (p.362)
  - ▷ organization/modeling results:
    - – no longer a tree, see example
    - – general subsets, may overlap
    - – illustration: Fig 21.6 (p.363)
- Details and some positive results:
  see Briand et al. (1992)

Risk Identification: Comparison

- Comparison: cost-benefit analysis
  $\approx$ comparing QA alternatives (Ch.17).

- Comparison area: benefit-related
  - ▷ accuracy
  - ▷ early availability and stability
  - ▷ constructive information and guidance for (quality) improvement

- Comparison area: cost-related
  - ▷ simplicity
  - ▷ ease of result interpretation
  - ▷ availability of tool support

Comparison: Accuracy

- Accuracy in assessment:
  - ▷ model fits data well
    - – use various goodness-of-fit measures
  - ▷ avoid over-fitting
  - ▷ cross validation by review etc.

- Accuracy in prediction:
  - ▷ over-fitting $\Rightarrow$ bad predictions
  - ▷ prediction: training and testing sets
    - – within project: jackknife
    - – across projects: extrapolate
  - ▷ minimize prediction errors

Comparison: Usefulness

- Early availability and stability
  - ▷ to be useful must be available early
  - ▷ focus on control/improvement
  - ▷ apply remedial/preventive actions early
  - ▷ track progress: stability

- constructive information and guidance
  - ▷ what: assessment/prediction
  - ▷ how to improve?
    - – constructive information
    - – guidance on what to do
  - ▷ example of TBRMs

Comparison: Usability

- Can't explain in a few words
  ⇒ difficulties with reception/deployment

- Simplicity & result interpretation?
  - ▷ technique easy to use/understand
  - ▷ what does it (the result) mean?
  - ▷ training effort involved
  - ▷ causal and other connections

- Tool and other support:
  - ▷ availability of easy-to-use tools
  - ▷ other support: process/personnel/etc.
  - ▷ direct impact on deployment

Summary & Recommendation

- Comparison summary and recommendation:
  - ▷ Summary: Table 21.4 (p.364)
  - ▷ Recommendation: TBM good balance.
  - ▷ Suite: Other technique with TBM.

- Lifecycle integration:
  - ▷ Process and data availability
    ⇒ inspection/testing/other QA data.
  - ▷ Experience/infrastructure/tools/etc. for implementation/technology transfer.
  - ▷ Similar techniques for other problems
    – e.g., identifying effort, schedule risks.
  - ▷ Tailoring to individual process/product

Tree-Based ODC Data Analysis

- Continuation of ODC analysis:
  - ▷ IBM Toronto data from ODC (Ch.20)
  - ▷ 1-way $\rightarrow$ 2-way $\rightarrow$ n-way analyses
    – combinatorial explosion
  - ▷ Better focus on n-1 linkage:
    – 1 response variable: impact
    – n (=6 here) predictor variables
  - ▷ ODC attributes in Table 20.6 (p.347)
    – all except "severity" used
    – impact-severity analysis already done: see Table 20.7 (p.351)

- Tree-based ODC modeling
  - ▷ Classification trees (instead of regression trees)
  - ▷ Change in distribution

Tree-Based ODC Data Analysis

- Result interpretation:
  - ▷ Overall result: Fig 21.7 (p.366)
  - ▷ Dominant impact: tree nodes.
  - ▷ Impact distribution: bars.
  - ▷ Confidence: frequency and cardinality.

- Impact distribution results:
  - ▷ Primary partition: defect trigger
  - ▷ High homogeneity of right subtree
  - ▷ Problem identification: left subtree
  - ▷ Distribution: Fig 21.8 (p.367)

- Usage of modeling results:
  - ▷ Passive tracking and correction
  - ▷ Active problem identification and quality control