Statistical Tools for Supporting Software Testing

Ed Brandt (Refis)
Alessandro Di Bucchianico (TU/e)

12th Dutch Testing Day
November 17, 2006
Outline

- Basics of Reliability Analysis
  - Case of Rijkswaterstaat (Dutch Directorate for Public Works and Water Management)

- How to Apply Reliability models
  - Statistical Best Practices
  - Tool by LaQuSo and Refis
The weakest link?
The main question:

How to determine the failure probability of software?
Determine failure probability by:
- Risk analysis
- Code analysis
- Testing
- Reliability analysis
Test results

- 200 bugs found, of which
  - 198 solved
  - 1 work around
  - 1 minor bug still open

- 1000 bugs found, of which
  - 998 solved
  - 1 work around
  - 1 minor bug still open
Definition of “Reliability”

"The probability that an item will perform a required function without failure under stated conditions for a stated period of time"
Reliability growth
Reliability growth
Reliability growth curve
These curves do not occur at random.
Results

Reliability for 1 year = 0.9014
Results

Expected number of defects in production = 5 or 6
How to Apply Reliability Models
Step 1: Data collection

- document data sources
- record data collection procedures
- clean-up data
Step 2: Determine Growth

- formal tests (Laplace, MIL-HB, LRT,...)
Step 3: Initial Model Selection
Step 4: Estimate Models

- ML
  - Parameters
  - Likelihood

- LS
  - Cumulative number of defects
  - Time

© TU/e - Refis
Step 5: Validate Model

- goodness-of-fit plots (TTT-plot, u-plot, QQ-plot, residual plots)
- goodness-of-fit tests (Kolmogorov, Cramer-von Mises,...)
Step 6: Interpret Model

- use confidence bounds to learn about precision
Refis LaQuSo Tool
Conclusions

- statistics can be a useful additional method to evaluate software reliability
- a tool is being developed for software reliability analyses that incorporates best practices ("coding standards") from statistics