Effort-based Re-estimation During Software Projects

Stephen MacDonell, AUT
Martin Shepperd, Bournemouth Uni
Agenda

• Rationale for re-estimation
• Industry data and analysis approach
• Results of analysis to date
• Outcomes and limitations
• Conclusions and next steps
• Preliminary insights…
Rationale and background

• Accurate estimation is a challenge!
  – Estimation is not (always) rational
  – Managers tend to be optimists
  – There has been a reluctance to move from early estimates
  – Global models, built based on unstable product factors, are widely used
Rationale and background (ctd)
Rationale and background (ctd)

• Alternatively we could (should?):
  – Use local models, based on process/resource factors
  – Harness growing certainty in data
  – Leverage managers’ expertise
  – Compare with the plan during (not just after) the project and then re-estimate
Industry data set

- We had access to one data set:
  - Software developed for a large test equipment manufacturer
  - Single organisation, multi-national
  - Sixteen development projects over an 18 month period
  - Effort range: 500-7800 person-hours
  - Consistency in technology, process, people
Industry data set (ctd)

• For each of the sixteen projects:
  – Effort for each phase had an original estimate (OE) and many had an adjusted, current estimate (CE)
  – Actual effort expended was also recorded at the project phase level
  – There was high confidence in the accuracy of the recorded effort data
Feasibility analysis

- Waterfall-like process, dominated by planning (PP), design (DES), implementation (IMP) and testing (TEST)
- Model fitting of effort per phase based mainly on process measures using least-squares linear regression
- Note: the entire data set was used – main aim was to assess feasibility
Model fitting of effort per phase

• Focused on design, implementation and testing phases (median 77% of project effort):
  – Design effort from planning effort
  – Implementation effort from design effort
  – Testing effort from design effort
  – Testing effort from implementation effort
Model fitting of effort per phase (ctd)

• Each model was built with and without a dummy variable indicating the intended deployment environment – runtime or non-runtime

• Three baseline models also built –
  (a) ‘predicting’ zero for every phase;
  (b) taking the mean phase effort;
  (c) taking the median phase effort
Model fitting of effort per phase (ctd)

- We also built simple combined models – the mean of the regression value and the manager’s estimates (OE and CE)
- Each model was assessed using sum of error and sum of absolute error indicators, and compared to the error of manager estimates
Results against OE (sum of error)

- Minimal improvements in fitting design effort (DES) based on planning effort (PP)
- Substantial improvements in fitting implementation (IMP) from DES, and testing effort (TEST) using DES or IMP (14%, 21% and 21% respectively)
- For specific project phases, fitting both IMP and TEST from DES resulted in improved values in 19 of 32 cases
Results against OE (sum of absolute error)

- Managers’ original estimates were more than 17,000 person-hours out.
- Regression models reduced error to just over 6,000 person-hours.
- Models produced improved values in 29 of 48 cases.
- Again, there were minimal gains in fitting DES using PP values.
Results against CE (sum of error)

• Managers’ current estimates were generally worse than the originals
• In particular, managers significantly underestimated DES and IMP effort
• Our models avoided gross errors (reducing error by 6,500 person-hours), but led to improved phase values in fewer than half the cases
Results against CE (sum of absolute error)

• Managers’ estimates outperformed the regression models in fitting DES using PP

• However, an improvement of more than 3,000 person-hours of effort was achieved in fitting IMP and TEST, with 20 of 32 phase values improved
Overall results of feasibility test

- In minimizing sum of error, the multivariate regression models were most effective.
- In minimizing sum of absolute error, the combined regression/manager approach worked best.
- Modelling implementation and testing effort using design effort appears to be particularly fruitful.
- In this case there was little gained in fitting design effort from planning effort.
Limitations

• This was a specific data set – general applicability of the results is unknown
• The whole data set was used for fitting and assessment of accuracy
• We were unable to utilize manager knowledge about other factors
• Clearly this does not address the ongoing need for early estimates
Conclusions and next steps

- Managers’ estimates can be improved upon using simple models based on prior-phase effort data.
- Use of multiple methods appears fruitful.
- Next steps:
  - predicting projects in sequence;
  - predicting projects using a moving sample;
  - combining product and process factors.
Predicting projects in sequence: preliminary outcomes

• All observations in a ‘growing’ data set…
  – Against OE, sum of error:
    15% reduction, improved 9 of 22 predictions
  – Against OE, sum of absolute error:
    11% reduction, improved 12 of 22 predictions
  – Against CE, sum of error:
    15% reduction, improved 9 of 22 predictions
  – Against CE, sum of absolute error:
    10% reduction, improved 12 of 22 predictions
Predicting projects in sequence: preliminary outcomes (ctd)

- Moving window using last five projects...
  - Against OE, sum of error:
    24% reduction, improved 8 of 22 predictions
  - Against OE, sum of absolute error:
    14% reduction, improved 14 of 22 predictions
  - Against CE, sum of error:
    24% reduction, improved 8 of 22 predictions
  - Against CE, sum of absolute error:
    13% reduction, improved 14 of 22 predictions