[simula . research laboratory]

INF 5500

Collection and Evaluation of Practice-based Evidence

Magne Jørgensen
magnej@simula.no

[simula . research laboratory]

Learning goals of this lecture:

- Better insight in learning problems, as means to better know when we can trust practice-based experience
- Better ability in identifying, collecting and evaluating relevant practice-based evidence

Recommended reading:

- http://web.cs.wpi.edu/~jburge/thesis/kematrix.html
- Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method, Magne Jørgensen & Dag Sjøberg

Experience vs expertise and skill

- *"Yet in nearly every study of experts carried out within the judgement and decision-making approach, experience has been shown to be unrelated to the empirical accuracy of expert judgements"* (Hammond 1996, p. 278).
- The amount of "deliberate practice", i.e., activities especially designed to improve specific aspects of an individual's performance seems to be more closely related to skill than amount of experience (Ericsson, Krampe et al. 1993).

"What we learn from history is that people don't learn from history."
(George Bernard Shaw)

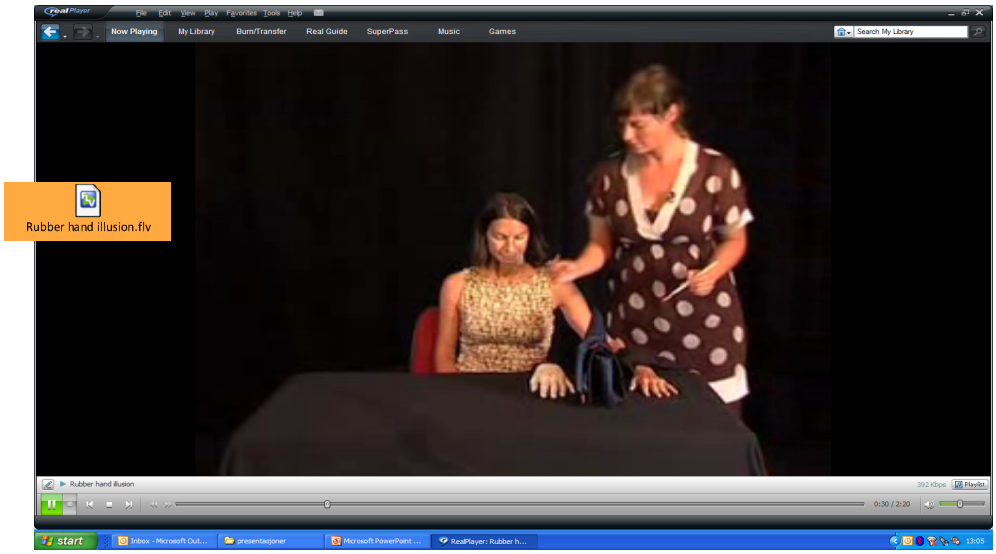
Experience vs expertise and skill

- The reasons why the quality of professionals' judgements may not improve much through experience are according to (Brehmer 1980):
  - We try to confirm theories, rather than reject incorrect hypotheses.
  - The fact that we are able to find a rule is sufficient to believe that we have a valid rule even though we have no experience indicating that the rule is valid. In other words, the confidence in own knowledge increases with the ability to find rules regardless of the validation of these rules.
  - In cases where we act on the experience based judgement there will be a number of additional factors that prevent us from detecting that our judgement is incorrect, e.g. self-fulfilling prophesies.
  - We tend to prefer deterministic rules even if the relationships between variables are probabilistic. If we find no deterministic rules, we tend to assume that there is no rule at all and start guessing.

Exercise

- Studies repeatedly shows that the actively managed mutual funds are not more profitable than their reference index (see for example www.ub.uit.no/munin/bitstream/10037/2118/1/thesis.pdf).
  - In addition, there is no (or a slightly negative) correlation between previous and future performance. This means that there are no persistence in the performance, either. Otherwise, we could select only the ”good” funds.
  - The model explaining the performance best is a pure ”by chance” model (random walk).
- Why, do you think, so many people do not see this and instead follow more profitable ways of investing their money, e.g., by buying so-called index funds?
  - In other words: Why do most people think the ”experts” managing the mutual funds have the skill we should be looking for, i.e., better predictions of the stock market than the reference index, when the reality is that they clearly don't have it?

Learning problem 1: We see what we expect to see



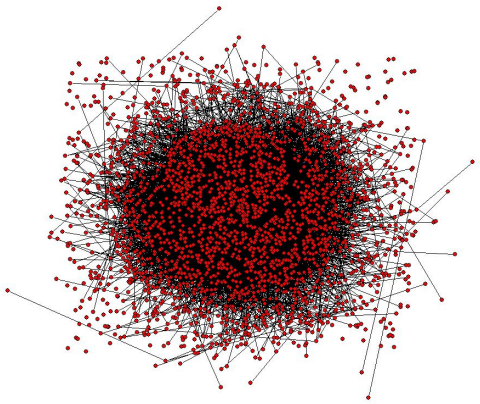
[simula . research laboratory]

Learning problem 2: “We won” - “they lost”

- We sincerely believe that we succeeded because we are skilled and failed becaue we had bad luck.
- The need for a high level of self-esteem makes learning sometimes difficult.
- Example:
  - Software developers systematically point at reasons outside their control to explain failures, and reasons the control as reasons for success.

Learning problem 3: Lack of the total picture

- **Local interpretation:** In a company, most project leaders agreed on that the most important reason for overruns was lack of clear and precise requirements.
- An analysis of the projects suggested the opposite. The advantage of vague requirements (increase of flexibility) was larger than the disadvantage of the lack of clarity.
- Exercise: Why didn’t the project leaders discover this?



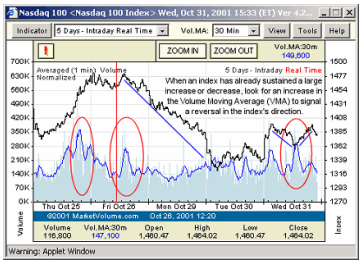
Learning problem 4: Superficial Learning

- Most people stop when they have believed they have found the direct causes, and do not look for indirect and contributory reasons.
  - A reason for problem failure is, for example, frequently ”unexpected events”.
  - BUT, unexpected events are quite common and should not be unexpected.
  - The important cause may be why they weren’t sufficiently prepared for unexpected events.
- Children are in many ways good learning examles for deeper learning.

[simula . research laboratory]

Learning problem 5: We see patterns were there are none

- HOT HAND?
  - ”*Basketball players and fans alike tend to believe that a player’s chance of hitting a shot are greater following a hit than following a miss on the previous shot. However, detailed analyses of the shooting records of [reference to several studies and a controlled shooting experiment] provided no evidence for a positive correlation between the outcomes of successive shots.*” (Gilovich, COGNITIVE PSYCHOLOGY 17, 295-314, 1985)
- Frequently the same problem in IT-projects. If B follows A two times in a row, we have a rule.
- Stock market analysis is heavily based on finding patters where there are none.



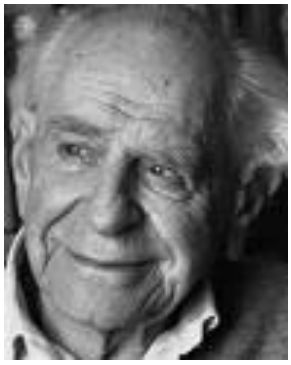
[simula . research laboratory]

Learning problem 6: Hindsight bias

- In a survey we gave the software professionals real and invented project outcomes. Regardless of the version they received, most of them thought that the outcomes were as expected.
- We do this, even when we (at least on behalf of others) are aware of the hindsight bias effect

Learning problem 7: Falsification

- Several studies show that we tend to confirm what we believe and are very poor at looking for and emphasizing non-conforming evidence.
- The consequence is that we may end up believing strongly in incorrect or strongly uncertain knowledge.



Learning problem 8: A strong focus on learning may make things worse

- In particular, when the desire is not connected with the opportunities to learn
  - F. I. Steele: Organizational overlearning, Journal of Management Studies, 1971.
- Example: Governmental reports on the reasons for failed, mega-large IT-projects.
  - Interpretations based on highly incomplete argumentation
  - The causal chain is clearly too simplistic. There are, for example, many cases where the same chain led to success.
- Paradox: The learning itself frequently makes the learning less relevant.

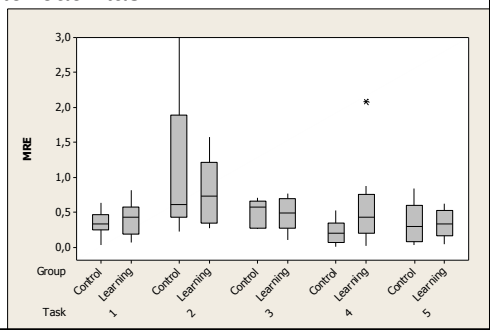
Results from a study

Design:

- 20 experienced software deveopers, randomly allocated a learning and a control group
- All of them estimated and complete the same five development tasks
- Those in the learning group, but not those in the control group were instructed to spend at least 30 minutes on the identification, analysis and summary of experience and learning after each task

Results:

- Those in the lerarning grop did not improve the estimation accuracy, and were more over-confident in the estimation accuracy. This may have been due over-estimation of how much they had learned.



OK, it's difficult to learn from experience. BUT, how should we collect reliable knowledge?

Guidelines: Check relevance, combine perspectives, triangulate of methods, be critical, design processes that go for the deeper cause-effect relationsships

- Check the relevance of the experience. Remember that:
  1. Relevance of knowledge and skill can be very narrow.
  2. Experience is not the same as knowledge. Preferably, to transfer from experience to relevant and reliable knowledge, the following conditions should be met by the persons's learning situation:
     - Learning-friendly conditions. Preferable situations where only few changes takes place and there are systematic effect measurement in place.
     - Unbiased interpretations. A person responsible for selecting a new tool is, as an illustration, not the best one to assess it's impact on quality and productivity.

How should we collect reliable knowledge?

- If unbiased, complete pictures from one person is difficult, try to collect information from more than one perspective, background and role.
  - Preferably, the informants should have formed their knowledge independent of each other.
- Example of knowledge collection technique:
  - Observations of on-the-job work
  - Interviews
  - Observations in controlled contexts with verbal protocols (thinking-aloud)
  - Study of written material (emails, experience reports, etc.)
  - Statistical modeling
  - Concept mapping
  - Sessions of analysis of cause-effects (Root Cause Analysis, Ishikawa, Post Mortem Analyses, …)

Types of cause (X) – effect (Y) relationships

- There is a *direct* causal link between X and the Y, i.e., X is a *direct reason* for Y.
- X leads to events that, in turn, lead to Y, i.e., X is an *indirect reason* for Y. If the events leading to Y started with X, we may call X the *root reason* or the *trigger reason*.
- The events actually leading to Y would have been harmless if X had not been present, i.e., X is an important *contributory reason*, or *necessary condition* for Y.
- The strength of Y always increases when X is present, i.e., X is a *deterministic* reason.
- The presence of X increases the probability of Y, i.e., X is a *probabilistic* reason.
- Manly the very high (or low) Y values are caused by X, i.e., X is mainly a *large effect* reason.

An example of data collection triangulation

- **Study**: Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method
- **Motivation**: How to collect practice-related experience that can enable reduced estimation error
- Experience collection methods:
  - Semi-structured interviews with employees in different roles
  - Examination of 68 written experience reports
  - Statistical analysis

Previous studies how a strong tendency to empasize direct reasons and reasons outside one's own control.

TABLE 1
Questionnaire-Based Studies on Reasons for Software Estimation Error

Study	Population	Study Design	Results
Phan et al. [2]	Software professionals (80% of them were project managers or developers) in 191 organizations.	Four pre-defined categories: Long duration, over-optimism, poor analysis and design, and frequent changes.	The two most important reasons were "unrealistic over-optimism[2]" and "frequent changes".
Van Genuchten [3]	Project managers responsible for the estimation of 160 activities in six development projects within one department.	Pre-defined classification of reasons for error. The six project managers marked one (or more) of these for each activity.	Most frequent reasons were "more time spent on other work than planned" and "complexity of application underestimated".
Lederer and Prasad [4]	Estimation responsible (mainly project managers and developers) personnel in 112 organizations.	Pre-defined list of reasons where general importance for estimation error was marked with a value from 1 to 5.	Most important reasons were "frequent requests for changes by users", "users lack of understanding of their own requirements", and "overlooked tasks".
Standish Group - 1994[3]	"IT executive managers" (mainly project managers?) from 365 organizations.	Pre-defined classification of reasons.	The three most important reasons for estimation over-runs were "lack of user input", "incomplete requirements and specifications", and, "changing requirements and specifications".
Subramanian and Breslawski [5]	Project managers in different companies representing 45 projects.	Reasons classified by the authors based on responses from the project managers.	Most important reasons were "requirement change/addition/deletion", "programmer or team member experience, turnover", and, "design changes, scope, complexity".

The personnel interviewed

- The manager of the technical personnel (M-Tech).
- The manager of the human-computer-interaction personnel (M-HCI).
- The manager of the graphic design personnel (M-Graph).
- The most senior project manager (PM-Sen). This project manager was frequently used to review other project managers' estimates.
- Two project managers with technical background (PM-Tech1 and PM-Tech2).
- A project manager with human computer interaction background (PM-HCI).
- A project manager with graphic design background (PM-Graph).

The interviews

Results:

- The responses depended very much on the reasons provided
- General managers provided more general reasons.
- Little critique of own role, e.g., the project managers did not think their project management ability was a problem, while the general managers thought this.
- Only on respondent mentioned ”contributory reasons”.
- The chain of reasons were not well explained and mainly based on beliefs.
- All reasons were described deterministically, in spite of that a probabilistic description would have been more correct in most contexts.

Interviews are well suited to get access to indirect reasons, but may need special attention to get to the deep-level causes.

Subject	Reasons
M-Tech *(Manager of the software developers)*	No systematic feedback to enable learning (→→). Insufficient time on estimation and planning (→→), leads to overlooked tasks (→).
M-HCI *(Manager of the HCI personnel)*	Lack of processes enabling learning from experience (→→). Insufficient focus on HCI in the estimation process (→→). Lack of client realism in HCI-requirements (→→). Poor project planning (→→). Poor project management (→→).
M-Graph *(Manager of the graphical designer personnel)*	Project managers are not skilled in planning multi-disciplinary projects (→→), which leads to insufficient focus on graphic design in the estimation process (→→), and inefficient allocation and use of graphic design resources (→). No systematic feedback to enable learning (→→). Insufficient tool support for project management (→→). Poor project management (→→). Customer requirements difficult to interpret (→→).
PM-Sen *(Senior project manager with extensive experience from project bidding and planning)*	Insufficient focus on the project manager role (→→), leads to insufficient training and feedback (→→). Insufficient standardization of planning and development processes (→→). The experience database of previous projects is not used (→→). Inefficient allocation of project resources (→→).
PM-Tech1 *(Project manager with technical background)*	Clients unable to deliver a good requirement specification (→→), leads to unplanned re-work (→). Lack of requirement change control processes (→→). Insufficient time spent on estimation and planning (→→). Not sufficient focus on learning from experience (→→).
PM-Tech2 *(Project manager with technical background)*	Projects are frequently different from earlier projects (→→), leads to lack of relevant experience when estimating (→), because of lack of checklists (→) and experience database (→). Incomplete requirement specifications (→→).
PM-HCI *(Project manager with HCI background)*	HCI is involved too late (→→), which leads to unrealistic expectations by clients (→→), and unplanned activities (→). Project manager has insufficient knowledge about HCI (→→). Not sufficient focus on learning from experience (→→).
PM-Graph[4] *(Project manager with graphic designer background)*	Insufficient focus on graphic design in the estimation process (→→). No systematic feedback to enable learning (→→). Estimate strongly impacted by price-to-win (→→). Lack of justification of estimates (→→).

The Experience Reports

- Experience reports from 68 projects/tasks
- Classification scheme for the reasons for accurate and inaccurate estimates
- Includes measures of estimation accuracy per project/task

Id.	Reason	Reported in Project	Mean MRE	Mean RE	Proportion of Over Median Large Projects
1	Unexpected events and overlooked tasks (→)	5, 8, 10, 11, 15, 21, 25, 26, 30, 31, 35, 43, 47, 49, 50, 51, 52, 58, 60, 61, 62, 63, 64, 65, and, 66	0.32	0.32	60%
2	Change requests from clients or "functionality creep" (→)	5, 7, 9, 14, 15, 16, 18, 22, 23, 31, 47, 48, 61, and, 67	0.35	0.32	71%
3	Simpler task or more skilled developer than expected (→)	13, 34, 36, 42, 57, and, 59	0.54	-0.54	17%
4	Resource allocation problem (→→)	8, 28, 43, and, 47	0.32	0.32	50%
5	Poor requirement specification or problems with communication with the client (→→)	4, 8, 18, 22, 25, 26, 31, 43, 44, 45, 48, 54, 59, 63, and, 67	0.42	-0.26	73%
6	Too little effort on estimation work (→→)	63	0.70	0.70	100%
7	High priority on quality, cost accuracy not of high importance (→→)	17, 18, 22, and, 30	0.32	0.33	75%
8	More reuse than expected from other projects (→)	4, and, 57	0.61	-0.61	50%

Experience reports

- Mainly direct reasons were reported.
- Success was described as due to the respondents' own skill and choices, failures were attributed events outside their control.
- Some obvious reasons were not reported, e.g., reasons related to the "political estimation games".
- A more structured process for experience reporting may have led to more reliable reports.

Statistical analysis

- MRE = 0,14 + 0,13 Company Role + 0,13 Participation + 0,13 Client Priority, (p=0.03) (p=0.08) (p=0.07) (p=0.09)
- RE = 0,12 - 0,29 Company Role + 0,27 Previous Accuracy (p=0.05) (p=0.004) (p=0.01)
  - Company Role: The project was estimated by a software developer = 1. The project was estimated by a project manager = 0.
  - Participation: The estimator estimated the work of others = 1. The estimator participated in the estimated project = 0.
  - Client Priority: The client prioritized time-to-delivery= 1. The client had other project priorities than time-to-delivery, i.e., cost or quality = 0.
  - Previous Accuracy: The estimator believed that he/she had estimated similar tasks with an average error of 20 percent or more = 1; less than 20 percent error = 0.

The Results Summarized

Interviews	Experience Reports	Statistical analysis of MRE
No systematic feedback to enable learning	Unexpected events and overlooked tasks	Project estimated by a software developer (as opposed to a project manager)
Poor project planning and management	Change requests from clients or "functionality creep"	Project estimated by a person not participating in the project
Poor requirement specification	Simpler task or more skilled developer than expected (reason for effort under-run)	Client prioritizes time-to-delivery, not cost or quality

- Different respondents and collection methods lead to different results.

Excercise

- Assume that your task is to analyse whether your company should introduce pair-programming. You know a couple of other companies that have used pair-programming and want to interview them about their experience, i.e., you want to get practice-based evidence about pair-programming relevant for you own company.
- Outline the design of the interview? (including preparation, selection of respondents, questions and request for other material that could be used to quality assure the interview-based responses – method triangulation)