Forecasting of Software Development Work Effort: Evidence on Expert Judgment and Formal Models

Magne Jørgensen
Simula Research Laboratory
P.O.Box 134
NO-1325 LYSAKER, NORWAY
Email: magnej@simula.no
Fax: +47 67 82 82 01

<Version: March, 2007, Updated May 2007>

The review presented in this paper examines the evidence on use of expert judgment, formal models, and a combination of these two approaches when estimating (forecasting) software development work effort. Sixteen relevant studies were identified and reviewed. The review found that the average accuracy of expert judgment-based effort estimates was higher than the average accuracy of the models in ten of the sixteen studies. Two indicators of higher accuracy of judgment-based effort estimates were estimation models not calibrated to the organization using the model, and important contextual information possessed by the experts not included in the formal estimation models. Four of the reviewed studies evaluated effort estimates based on a combination of expert judgment and models. The mean estimation accuracy of the combination-based methods was similar to the best of that of the other estimation methods.

1. Introduction

Clients require effort and cost estimates of software projects as inputs to investment analyses. Similarly, project managers require effort estimates to enable planning and to control the software development work. Unfortunately, many software development effort estimates are quite inaccurate. A recent review of estimation accuracy studies indicates that software projects expend on average 30-40% more effort than is estimated (Moløkken-Østvold & Jørgensen, 2003). There seems to have been no substantial improvement in estimation accuracy over the years. Software projects experience severe delivery and management problems due to plans based on overoptimistic effort estimates. The negative effects of overoptimism are accentuated by (i) software bidding rounds where those companies that provide overoptimistic effort estimates are more likely to be selected, and (ii) overconfidence in the accuracy of the estimates; for example, 90% confidence effort prediction intervals only include the actual effort 60-70% of the time (Jørgensen, Teigen et al., 2004).

Software researchers have been addressing the problems of effort estimation for software development projects since at least the 1960s; see, e.g., Nelson (1966). Most of the research has focused on the construction of formal software effort estimation models. The early models were typically regression-based. Soon, however, more sophisticated effort estimation models appeared, for example models founded on case-based reasoning, classification and regression trees, simulation, neural networks, Bayesian statistics, lexical analyses of requirement specifications, genetic programming, linear programming, economic production models, soft computing, fuzzy logic modeling, statistical bootstrapping, and combinations of one or more of these models. A recent review (Jørgensen & Shepperd, 2007) identified 184 journal papers that introduced and evaluated formal models for software development effort estimation. Many of these studies describe the re-examination and improvement of previously proposed estimation methods. Several estimation models have been included in commercially promoted tools. A survey by Moores and Edwards (1992) found that 61% of the IT managers in the UK had heard about at least one of these software development effort estimation tools. The use of formal estimation models has also been promoted by software process improvement frameworks and in software engineering education readings.

In spite of the extensive research into estimation models, the high degree of availability of commercial estimation tools that implement the models, the awareness of these estimation tools, and the promotion of model-based estimation in software engineering textbooks, software engineers typically use their expert judgment to estimate effort (Heemstra & Kusters, 1991; Hihn & Habib-Agahi, 1991).

The limited use of models may be a sign of the irrational behaviour of software professionals. It may, on the other hand, be the case that expert judgment is just as accurate or has other advantages that render the current low use of effort estimation models rational. This leads to the research questions of this paper: i) Should we expect more accurate effort estimates when applying expert judgment or models? ii) When should software development effort estimates be based on expert judgment, on models, or on a combination of expert judgment and models?

Extending Jørgensen (2004), I review studies that compare the accuracy of software development effort estimates based on estimation models with those based on expert judgment and on a combination of these two approaches. The review process, limitations and results are included as Section 4. The factors examined in the review are derived from the discussion of the task of software development effort estimation in Section 2, and previous findings on the relative performance of model and judgment-based predictions are presented in Section 3. Section 5 provides concluding remarks about the implications of the findings of the review.

2. Software Development Effort Estimation

For the purpose of this review, I separate expert judgment and model-based effort estimates based on the type of mental process applied in the "quantification step", i.e., the step where an understanding of the software development estimation problem is translated into a quantitative measure of the required effort. I define judgment-based effort estimates to be based on a tacit (intuition-based) quantification step, and model-based effort estimates to be based on a deliberate (mechanical) quantification step; see, for example, Hogarth (2001) for an elaboration of the meaning of these terms. The quantification step is the final step of the process leading to an effort estimate for the total project or a project activity. If the final step is judgmental, the process is categorized as judgment-based. If the final step is mechanical, the process is categorized as model-based. There will be a range of quite different estimation processes belonging to each of the categories, i.e., neither expert judgment nor model-based effort estimation should be considered simply as "one method". When the outputs of two or more completed estimation processes are combined, we categorize the process as combination-based, and describe whether the combination step is judgmental or mechanical.

The term "expert" in this paper is used to denote all individuals with competence in estimating software development effort. In most studies, the expert is a software development professional, but we also use the term "expert" to denote, for example, a student with previous experience in effort estimation and the development of software for the type of task under consideration.

2.1 Expert Judgment-based Effort Estimation Processes

Most of the steps in the expert judgment-based effort estimation processes, e.g., the breaking down of the project into activities, may be explicit and can be reviewed readily. The quantification steps, however, are based on intuition to a significant degree, and are seldom based on explicit, analytical argumentation. This assessment of the quantification steps as being based on intuition is indicated both by a lack of analytical argumentation and by the frequent use of phrases such as "*I think that …*" and "*I feel that …*"; see for example the transcribed estimation team discussions in Jørgensen (2004). Similar results are reported in the software cost estimation study in Mukhopadhyay, Vicinanza et al. (1992): "… the verbal protocol contained little explicit information about the cognitive processes involved in selecting a source project [i.e., the selection of analogous projects as a basis for the estimate of the new project]." The poor understanding of the quantification step is also an indication that it is intuition-based. According to Brown and Siegler, psychological research on real-world quantitative expert estimation "has not culminated in any theory of estimation, not even in a coherent framework for thinking about the process" (Brown & Siegler, 1993)

2.2 Model-Based Effort Estimation Processes

There are many different types of software development effort estimation models available. Briand and Wieczorek (2002) categorize and describe many of these models. An example of a very simple "rule-of-thumb" estimation model is a model that contains, among other rules, the rule that a "small" program module with "high" complexity requires about 30 work-hours. However, a program module's size and degree of complexity are typically not known with high precision at the time of the estimation, and are typically based on expert judgment. The example illustrates that model-based effort estimation processes may rely very much on expert judgment-based input. As a consequence, model outputs may also be biased towards over-optimism or be impacted by the presence of irrelevant information.

More complex effort estimation models may be based on sophisticated analyses and dependencies between effort and other variables in sets of previously completed projects, and result in formulae of the following type:

$$Effort = a\ Size^{b} * Adjustment\ factor$$

The size variable can, for example, be a measure of the 'size of functionality,' derived from the requirements specified by the client or the estimated number of 'lines of code' to be programmed. The adjustment factor is typically derived from a weighted sum of the answers to questions relating to the complexity of the development, project member skills, and the tools used to support the development process. The adjustment factor may also include the input of a productivity factor, i.e., a measure of the historical productivity of similar projects.

Many estimation models assume that there are organization-independent and stable relationships between the variables, e.g., that parameters a and b in the above formula are approximately the same for all software development projects. Other estimation models recommend that core relationships be calibrated to the situation in which they are used. The difference in model calibration to the organization in which the model is used may be an important factor for estimation accuracy, and important for our review. The assumptions that many models make regarding situation-independent core relationships between size and effort may be a major cause of the inaccuracy of the estimation models. There is evidence to support the view that models calibrated to a particular organization, e.g., through deriving the model from the organization's own historical data only, may lead to an improvement in the estimation accuracy. This evidence is provided in Murali and Sankar (1997) and Jeffery, Ruhe et al. (2000), among other studies. To analyze how differences in the level of calibration to a particular organization affect the relative performance of models and expert judgment in the review, we use three categories of calibration level:

- *Low calibration (adjustment relative to a "nominal" project)*: The model assumes an organization-independent dependency between effort and other variables. The adjustment to the estimation situation at hand is done through standardized adjustment factors related to differences between the "nominal" (typical) project and the project to be estimated, e.g., add 20% to the total effort if the project applies a development method for the first time. No statistical analyses based on the organization's own historical project data are performed. Most commercial estimation models and several of the noncommercial estimation models are of this type.

- *Medium calibration (adjustment through the use of organization-specific productivity values):* Models in this category make assumptions similar to those in the low calibration category. The main difference is that some of the standardized adjustments relative to a "nominal" project are replaced with the use of organization-specific productivity values.
- *High calibration (estimation models derived from organization specific data only)*: Models in this category are generated from a dataset of projects that have previously been completed in the organization in which the model is supposed to be applied or in organizations with similar types of projects. There are many possible approaches to generating the models, e.g., regression analysis, case-based reasoning, or neural network development.

3. Prior Research

There are many studies on expert- and model-based judgment. In addition, there are numerous studies on related topics, such as intuition vs. analysis, and tacit vs. deliberate processes. In this section I have tried to present a set of representative results.

3.1 Clinical versus Statistical Prediction

In 1954, Meehl published his so-called "disturbing little book" *Clinical versus Statistical Prediction: A Theoretical Analysis and a Review of the Evidence* (Meehl, 1954). In it, Meehl summarizes twenty empirical studies and finds that clinicians (who provide expert judgments) are usually outperformed by actuarial methods (statistical prediction models). Meehl (1986) states, based on an updated set of reviewed studies, that "*When you are pushing 90 investigations, predicting everything from the outcomes of football games to the diagnosis of liver disease and when you can hardly come up with a half dozen studies showing even a weak tendency in favour of the clinician, it is time to draw a practical conclusion.*" A more recent meta-analysis, extending the studies summarized by Meehl, is provided in Grove, Zald et al. (2000). That study found that "*… mechanical predictions of human behaviors are equal or superior to clinical prediction methods for a wide range of circumstances.*"

Dawes, Faust and Meehl (1989) emphasize the following two factors that underlie the superiority of statistical models: i) Models are consistent; the same input always leads to the same conclusion, while experts are inconsistent. ii) Models ensure that variables contribute to a conclusion based on their actual predictive power and relationship to the criterion of interest. Experts have problems in distinguishing between valid and invalid variables, due, among other things, to poor and misleading feedback about the accuracy of judgment. The importance of the two factors is also supported by a substantial amount of independent empirical evidence, e.g., by studies on the "the dilution effect" in expert judgment (Wallner & Zimbelman, 2003).

Transferring Meehl's recommendation "*it is time to draw a practical conclusion*" naïvely to the context of software effort estimation, we are exhorted to use models and to stop using expert judgment when estimating software development effort. There are, however, at least two issues that may make this type of conclusion premature in the context of software development:

- The performance of an estimation model depends on the properties of the relationships it attempts to model. In the domain of software development effort estimation, the validity of basic model assumptions, e.g., the stability of an effort-size relationship (Kitchenham 1992; Dolado 2001), are contentious, and may have lower validity than the essential assumptions made when using models in other domains. In medicine, for example, the assumption of stable underlying (biology-based) relationships may be more plausible than in software development contexts where the technology, the types of software produced, and the production methods, change frequently.
- Dawes, Faust and Meehl (1989) specify that a condition for a *fair* comparison is that both the model and the expert base their predictions on the same input data. This condition is typically not met in the field settings for software development effort estimation. In fact, to meet this condition we may have to remove some of the information typically used by the experts in field settings, i.e., create a situation that in many ways would be perceived as *unfair*, and deviate from the natural setting of effort estimation.

3.2 Contextual Information

It may be possible to include most contextual information in a model. When software effort estimation models typically choose to include only a few variables, and potentially not all variables are important, the reasons are of a practical nature:

- a large number of variables can easily lead to over-fitting and lower accuracy when there are small data sets to learn from;
- models need to be simple if their users are to understand them;
- the development of a proper model may be too complex or take too much effort; and
- variables with the potential to become important are many, and in most cases are not important.

For example, the most important input in software development effort estimation situations is a textual description of the requirements to be met by the software system, together with oral information collected at meetings with the clients. This textual and oral information contains a great deal of knowledge that it is scarcely practical to provide as an input to a model, e.g., highly specific information that enables the developers to understand the steps needed to perform the programming tasks or the importance of a particular requirement for a particular client. The aggregated and translated model version of this textual and oral information which is provided as an input to estimation models can hardly be said to be “the same data,” and our context may consequently be different from that assumed by Dawes, Faust and Meehl (1989). Another example of the important contextual information typically possessed by the experts but not necessarily easily transferred to a model is very specific information (so-called “broken leg” cues) about the software developers allocated to the task. The experts may, for example, possess a lot of information about the differences in productivity among the developers, which may be huge, or may know that one of the developers has successfully solved a very similar task earlier. However, this additional information possessed by the experts does not always lead to a more accurate judgment. For example, the presence of information of lesser relevance may easily have strong, unwanted impacts on judgment-based software development effort estimates (Jørgensen & Sjøberg, 2004), and the total effect of more contextual information on the experts’ judgments is not obvious.

It may be of particular relevance for the review in this paper to examine previous studies on the performance of expert and model predictions in situations where the experts possess additional (contextual) information, i.e., comparisons which are closer to real-life situations in software development effort estimation. The search for studies of this type resulted mainly in forecasting studies. Several researchers in forecasting seem to question the generality of the finding that models are more accurate than experts. Lawrence and O’Connor (1996), for example, observe that many of the studies that report the superiority of model-based judgment seem to be based on an environment where the important variables are well-established, prespecified and not autocorrelated, and where there is little contextual information that only the expert possesses; i.e., that the results are based on environments that favour model-based judgment more than many real-life forecasting environments do.

Findings suggesting that there are forecasting situations that may benefit from expert judgment include:

- Judgment-based forecasts were more accurate than statistical models in situations that contained a substantial amount of contextual information (Webby & O'Connor, 1996; Goodwin, 2000).
- Judgment-based forecasts were better in unstable, changing situations, while the models performed better during periods of stability (Sanders & Ritzman, 1991).
- A combination of model- and expert-based judgment was frequently better than either alone (Blattberg & Hoch, 1990; Goodwin, 2000).

However, there are also findings that indicate the opposite, e.g., that the inclusion of irrelevant information leads to the superiority of model-based judgment (Whitecotton, Sanders et al., 1998). The existence of situations where the benefits of contextual information are large enough to compensate for judgmental inconsistency and improper weighting emphasize that a comparison of expert- and model-based effort estimation accuracy needs to take into account the amount and nature of the contextual information.

3.3 Expertise

A limitation of many studies comparing expert judgment and models is that they are based on the *average* performance of a set of experts who are chosen more or less arbitrarily, and not, for example, on the performance of the *best* experts. The value of studying the performance of university students in conducting a complex task in a domain where they have little experience is not always obvious. Not surprisingly, there are several authors that question many of the results on the basis of a lack of ecological validity; see, for example, Bolger and Wright (1994).

Shanteau (1992) emphasizes that the characteristics of a task play an important role in the performance and learning of experts. Software development effort estimation has characteristics of both poor and good expert performance. While the characteristics “some errors expected” and “problem decomposable” may lead to good expert performance, “decisions about behaviour” and “unique task” may lead to poorer expert performance. It is consequently difficult to decide, based on Shanteau’s work, how much experts are able to learn and improve with increased experience in real-life software development effort estimation contexts, i.e., how the level of estimation expertise is connected with the amount of experience.

In most situations in which software development effort is estimated, there are several competing estimation models and several expert estimators to select from or to combine. The selection of the model and expert is typically expert judgment-based. Selecting improper models or experts may lead to very inaccurate predictions, and hence, the process by which an estimation method is selected may be essential for this review. Hogarth (2005) makes a similar point when he examines the trade-off between biased, intuition-based judgments and the risk involved in selecting or executing analytical rules. Analytical errors are more likely when the analytical complexity, as perceived by the person selecting the rule, is high. So far, there has been no study in the context of software development on experts’ ability to select proper models and experts, and only a few studies on formal strategies for selecting estimation models and experts; see, e.g., Shepperd and Kadoda (2001). An important issue for the review is, consequently, whether the risk of selecting very inaccurate estimation methods is higher when selecting a model or when selecting an expert. It may, for example, be the case that complex effort estimation models are sometimes the most accurate, but are also connected with the most inaccurate estimates, due to over-fitting to one type of situation.

Finally, expertise in using estimation models and expertise in applying expert judgment may have different side-effects regarding the actual work performed. There may, for example, be a stronger effect of “self-fulfilling prophecies” when applying expertise in making judgments compared to expertise in using models; i.e., people’s ownership and commitment related to expert judgment may be stronger than that to model output. We were unable to find any studies on this side-effect of using different types of models for effort estimation. However, there are related findings, e.g., findings on the positive effect of effort estimate accountability on estimation accuracy in software development contexts (Lederer & Prasad, 2000).

4. The Review

4.1 The Review Process

The review process aims to identify and analyse empirical studies that compare expert judgment-based and model-based software development effort estimation. The identification of relevant studies is based on an examination of software development effort estimation journal papers identified in a recent review (Jørgensen & Shepperd, 2007). That review constitutes currently, as far as we know, the most complete list of journal papers on software development effort estimation, and can be accessed at www.simula.no\BESTweb. Potentially relevant papers presented at conferences were identified through a manual inspection of the studies resulting from a search in the library database Inspec for papers including the terms (‘*effort estimation*’ OR ‘*cost estimation*’) AND ‘*software development*’ (last search conducted February 2006). In spite of this fairly comprehensive search for relevant papers, there may still be missing papers which are relevant. As an illustration, when we contacted the authors of the reviewed papers, one of them made us aware of a relevant paper not found by our search.

In total, seventeen relevant papers were identified. One of the papers was excluded, namely, Pengelly (1995), due to incomplete information about how the estimates were derived, which left sixteen papers for review. The sixteen studies are reviewed with respect to important contextual factors, i.e., the factors identified in the discussion in Sections 2 and 3. The main design factors and results reviewed for each study are as follows:

DESIGN FACTORS

- Study design
- Estimation method selection process
- Estimation models
- Calibration level
- Model use expertise and degree of mechanical use of model
- Expert judgment process
- Expert judgment estimation expertise
- Possible motivational biases in estimation situation
- Estimation input
- Contextual information
- Estimation complexity
- Fairness limitations
- Other design issues

RESULTS:

- Accuracy

- Variance
- Other results

The factors are explained and applied in Appendix A.

Sixteen is a small number of studies when attempting to analyze how the numerous design factors potentially affect the estimation accuracy of models and expert judgments differently. In addition, since none of the reviewed studies were explicitly designed to identify *when* we could expect expert judgment or models to perform better, much information about several of the factors is missing. When our interpretation of factor values is based, to a large extent, on a qualified guess, we have described this interpretation as "probable". We sent the results of our review to the authors of each of the sixteen studies and urged them to inform us of any incorrect classifications and interpretations regarding their own study. Authors representing thirteen of the sixteen papers responded. The authors' responses led only to minor corrections.

The main evaluation measure in this review is estimation accuracy, i.e., the deviation between the estimated and actual effort. This should not be taken to imply that we think that other measures, e.g., flexibility in the use of the method, the cost of the estimation process, or the ease of understanding the basis of the estimates, are unimportant. The reasons for not emphasizing these factors are that they deserve reviews on their own and (the practical reason) that none of the studies reported criteria for any comparison other than accuracy.

4.2 Review Limitations

The review work revealed several factors limiting the validity of the results of the studies, including the following:

- *Lack of information about the expert judgment-based process*. Most studies do not describe the expert judgment-based estimation process. This means that while there are many different models evaluated, expert judgment is lumped into one category. This is particularly unfortunate, given the potentially large differences between unstructured, unaided expert judgment and expert judgment supported by a well-structured estimation process, detailed checklists, proper feedback, and historical data.
- *Different estimation methods were used on different estimation tasks in field studies.* The study reported in Grimstad and Jørgensen (2007) exemplifies how a comparison of model-based and expert judgment-based estimation in field settings can be biased by the use of expert judgment in situations where it is not possible to use estimation models. A straightforward comparison of the accuracy of effort estimations for projects that applied both estimation models and expert judgment yielded the result that using models led to significantly more accurate estimates. However, it was also observed that the estimation model was seldom used at an early stage of the project, and was never used when the estimator had no experience with similar projects. Both these situations are, however, connected with a higher than average estimation complexity. When only comparing estimation tasks with similar estimation complexities, model-based and expert judgment-based estimates were found to be accurate to the same degree. Unfortunately, none of the other field studies in our review perform this kind of analysis. The results reported in Grimstad and Jørgensen (2007) suggest that it is likely that the expert judgment-based performance is better, in actual fact, than is reported in the reviewed field studies, but more evidence is needed to confirm our conjecture.
- *Imprecise use of terminology.* Few of the reviewed studies reported that steps had been taken to ensure that the term 'estimate' was used with the same meaning when using models and expert judgment. If models are more likely to provide the most likely effort and experts are more likely to provide the planned or

budgeted effort, this may mean that expert judgment-based estimates are, in actual fact, less accurate than is reported in situations where a tendency towards over-optimism is present. However, the overall effect on the results of the review of using estimation terminology imprecisely is not well understood.

- *Different "loss functions" of models and experts.* None of the reviewed studies analyzed the "loss functions" of the estimation methods, and it is difficult to draw conclusions about the impact of this issue from the results of our review. If expert judgments are, consciously or unconsciously, based on more appropriate and flexible loss functions than the loss function of the estimation models, the reported accuracy results may provide an overly negative view of the experts' performance. For example, while most software effort estimation models are based on the assumption that over- and underestimation are equally bad, judgment-based effort estimates may be based on an assumption that effort estimates that are too high would lead to inefficient development work, and should be avoided more than estimates that are too low.
- *Estimation accuracy affected by effort management*. A strong belief in an effort estimate may lead to a stronger belief in the plan that is made and a greater commitment to following the plan. If this belief depends on the estimate's correspondence with an expert's gut feeling regarding the correctness of the estimate, the results may be biased in favour of the expert. Consequently, it may be the ability to better work to the estimate or plan that leads to a better expert judgment performance, and not a stronger skill in estimating accurately.
- *Experts estimating in groups*. Software companies frequently assign the task of estimating effort to groups. This may be the rule rather than the exception when the projects are large. However, only one of the reviewed studies enabled a comparison of the output of models with the output from a group of experts.
- *Unpublished results.* The effect of unpublished results is unknown. It may, for example, be the case that several of the studies where self-developed estimation models are evaluated and are found to yield less accurate estimates than the experts are not published.

These limitations mean that the results of the review should be interpreted carefully, and that better-designed studies are needed to deliver robust results about when to apply model-based and when to apply expert judgment-based effort estimates. Such studies should include proper descriptions of all the design factors outlined in Section 4.1, and aim at a better understanding of when and why one method is more accurate in one particular context.

In spite of the strong limitations of the reviewed studies, I believe that it is worthwhile to summarize the available evidence. To know the current state of our knowledge is of value, even if the review should show that our knowledge is modest due to study design limitations.

4.3 Results

In this section I try to answer the research questions stated in the introduction:

- Did models or expert judgment lead to the most accurate estimates? (Section 4.3.1)
- When did the estimation models, the expert judgments, and the combination of these two approaches each lead to the most accurate estimates? (Section 4.3.2)

Details of the review data are provided as Appendix A.

4.3.1 Which Estimation Method Yields the Most Accurate Effort Estimates?

The reviewed studies report the accuracy results differently. Hence, it is not possible to summarize the results as simply “method X leads, on average, to an A% improvement in estimation accuracy”. Instead, we have described the accuracy results as reported by the study itself in Appendix A, and produced in Table 1a simple categorization of whether a study reported that the models or the experts had the best estimation accuracy. The comparison is made relative to the *most accurate*, *average*, and *least accurate* performance of the models and the experts. Not all studies report data that allow all variants of comparisons; e.g., most studies report only the average accuracy of the experts. When a study evaluates only one estimation model or expert, the accuracy of that model or expert is categorized as the average model or expert accuracy in Table 1. The studies are sorted chronologically, i.e., Study 1 was conducted before Study 2, etc. Table 1 shows, for example, that there were only two studies (Studies 2 and 12) that enabled a comparison of the most accurate model and the most accurate expert, and that both of these studies found that the most accurate expert was more accurate than the most accurate model.

Table 1: Experts vs Models

	Expert More Accurate	**Model More Accurate**
Most Accurate Model vs Most Accurate Expert	Studies 2 and 12	No studies
Most Accurate Model vs Average Accuracy of Experts	Study 6	Studies 1, 2, 7, 9, 11, 12, and 14
Most Accurate Model vs Least Accurate Expert	No studies	Studies 2 and 12
Average Accuracy of Models vs Most Accurate Expert	Studies 2 and 12	No studies
Average Accuracy of Models vs Average Accuracy of Experts	Studies 1, 2, 3, 5, 6, 7, 9, 10, 11, and 13	Studies 4, 8, 12, 14, 15, and 16
Average Accuracy of Models vs Least Accurate Expert	No studies	Studies 2 and 12
Least Accurate Model vs Most Accurate Expert	Studies 2 and 12	No studies
Least Accurate Model vs Average Accuracy of Experts	Studies 1, 2, 6, 7, 9, and 11	Studies 12, and 14
Least Accurate Model vs Least Accurate Expert	No studies	Studies 2 and 12

The principal finding that may be derived from Table 1 is that the review does not support the view that we should replace expert judgment with models in software development effort estimation situations. On the other hand, neither does it support the view that software development effort estimation models are useless. A comparison of the average accuracy of the models with the average accuracy of the experts shows that ten studies found increased accuracy with the use of expert judgment and six with the use of estimation models.

The unit in Table 1 is the study. The studies vary considerably, however, in the number of observations included. This means that, although Study 1 has only 14 observations and Study 9 has 140, they both have the same weight in Table 1. To test whether a change of study unit would make a difference, we weighted the estimation accuracy of the twelve studies reporting the MAPE (Studies 1, 2, 6, 7, 8, 9, 10, 11, 12, 13, 15, and 16) in accordance with the number of observations included in the study. This resulted in a weighted MAPE of the experts which was slightly better than that of the models (99% vs 107%). The four studies which were not part of this analysis (Studies 3, 4, 5, and 14) included two studies in favor of models and two in favor of expert judgment. The high values of the weighted MAPEs of both the experts and the models are largely due to a few laboratory studies with a high number of observations and lacking most of the information available in many real-life estimation contexts.. Removing the laboratory study with the most inaccurate estimates (Study 2), for example, reduced the weighted MAPE to 78% for both the expert and the model-based effort estimates. A typical value of the MAPE for effort estimation in field settings is, as reported in the introduction, 30-40%.

The field studies (Studies 3, 4, 5, 8, 10, and 16) have the most observations, and may have the greatest external validity . Of the field studies, three are in favour of using models and three in favour of using expert judgment; none of them reported large differences in accuracy related to the use of models and expert judgment in estimating software development effort, i.e., the general result here is that there were no large difference between models and experts. Only the three smallest field studies reported the MAPE, and for this reason, we have not included the weighted MAPE for the field studies alone.

A possible objection to the results in Table 1 is that the models are not mechanically used, i.e., the use is better described as “expert judgment in disguise”. If this is the case, the review merely compares one type of expert judgment with another. This possibility is difficult to exclude for some of the reviewed studies. Eight of the studies (Studies 2, 6, 9, 10, 11, 12, 14, and 15), however, describe a rather mechanical use of the models, i.e., the model users had limited or no opportunity to adjust the input to yield a model output in accordance with their “gut feeling”. A comparison of the average accuracy of the experts and models for that subset of studies shows that the expert judgment led to more accurate effort estimates in five of these eight studies, i.e., the degree of mechanical use of the models seems not to explain the lack of model superiority in our review. The model users had previous experience in the use of models in all of these eight studies.

In eight of the studies, the model builder and evaluators are the same (Studies 6, 7, 9, 10, 11, 12, 13, and 14). In these studies, the vested interest of showing benefit from the model may be higher than in the other studies. An analysis of the results shows that in spite of this vested interest, the average accuracy of the experts was better than that of the self-developed models in five out of the eight studies.

Interestingly, the recent studies are more frequently in favour of using models than the early studies. However, it is too early to see whether this is a trend due to estimation models having improved over the years or is only due to a random variation in the study design and the types of models evaluated.

Assume that we were able to select the best model. On this assumption, Table 1 suggests that the use of this model is likely to lead to more accurate estimates than the judgments of either the average or the least accurate experts, but not more accurate estimates than the judgments of the best expert. Now assume that we are unskilled in model selection and select the least accurate model. In this case, Table 1 suggests that only the judgments of the least accurate experts are less accurate than the output of this model. The ability to select the best models has been little studied in the context of software development and may deserve more attention. The results reported in MacDonell and Shepperd (2003) suggest that using formal rules (e.g., the rule-based induction algorithms) to select the best model does not yield the desired result.

It is of equal importance to select good experts, since the least accurate expert performed worse than the models in each study. Research results suggest that it is possible, to some extent, to select among the best estimation experts by emphasizing relevant experience from very similar projects (Jørgensen & Sjøberg, 2002; Jørgensen, 2004), e.g., based on whether the estimators recall close analogies or not. Another means to identify the most accurate experts is to use their previous estimation accuracies to predict the future accuracy. In Jørgensen, Faugli et al. (2007) it is reported that, among twenty experienced software professionals with similar skill levels and backgrounds, the correlation between the estimation accuracy of previous and future programming tasks was 0.40, and that using the previous estimation errors to predict the most overoptimistic estimator (out of two) for future tasks would yield a 68% success rate.

An evaluation of effort estimates *combining* the inputs from experts and models is included in only four of the studies (Studies 1, 12, 13 and 14). All studies except Study 1 combined expert judgment-based estimates with estimates from models with a high levels of calibration. Study 1 evaluated the judgmental combination of expert judgment and two models with a low level of calibration. In that study, the combined estimate was as accurate as the best model and slightly better than the expert judgment-based estimate. In Study 12, the experts judgmentally combined the models' and their own judgment-based effort estimates. This combination led to an improvement in accuracy compared to the use of either models or expert judgment alone. Study 13 found that expert judgment-based effort estimates were slightly better than those based on a mechanical combination of estimation methods. Study 14 found that expert judgment, regression analysis-based models, and case-based reasoning-based models complemented each other well, i.e., when one method was not very accurate, it was likely that at least one of the other models was significantly more accurate. A simple average of the three methods improved the accuracy compared to the best individual method, i.e., the regression-based method. The details of the results for the combination-based estimates are included in Appendix A.

4.3.2 When to Use Expert Judgment and Models

Table 2 compares the average accuracy of the model-based estimates with the average accuracy of the expert judgment-based estimates for each study relative to the model calibration levels: low, medium and high, as described in Section 2.4. Some studies provide "mixed evidence", e.g., Study 2 found that one model with a low level of calibration was more accurate, and another with the same level of calibration was less accurate, than the average accuracy of the experts. Note that some of the studies do not report enough information for us to decide on the calibration level of the models, and so are not included in Table 2. When the level of calibration is not reported, we only reported our assessment (qualified guess) in Table 2 when this assessment was confirmed by one of the authors of the paper reporting the study. One study may provide more than one result.

Table 2: Evidence on the Relationship between accuracy and the level of model calibration

	Low Calibration	**Medium Calibration**	**High Calibration**
The model is less accurate than the average expert	Studies 1, 5, 6, and 7	Study 9	Studies 6, and10
The model is more accurate than the average expert	Study 14	Studies 8, and 15	Studies 9, 12, 14, and 16
"Mixed evidence"	Study 2	No studies	Studies 7, 11, and 13

Table 2 suggests a weak connection between how well models perform relative to experts and the level of model calibration, i.e., models should be calibrated to the situation in which it is used to compete with expert judgment. The studies which provide counterevidence of the connection between the calibration level and performance are Studies 2 and 14 . A discussion with the author of Study 14 suggests that a possible reason for the model's performing well in spite of the low calibration may have been that the set of projects that led to the construction of the estimation model was similar to the set of projects on which the model was applied, i.e., that the model was reasonably well-calibrated to the organizational context "by accident". The "mixed evidence" of the models with a low level of calibration in Study 2 is caused mainly by one expert who provided extremely inaccurate estimates, which does not provide strong counterevidence for the proposed connection. Interestingly, Table 2 suggests that the proportion of studies evaluating models with high calibration is higher for the most recent studies, i.e., there seems to have been a shift from general estimation models towards more situation-tailored models. This may explain the trend of improved model accuracy over the years that is suggested by Table 1.

The level of contextual information, i.e., the amount of information possessed only by the experts, was derived from the study design description. The authors of the papers describing the study were given the opportunity to correct our assessment of the contextual information. Table 3 summarizes this information and compares the average accuracy of the models with the average accuracy of the experts for each study.

Table 3: Evidence on the Relationship between accuracy and theexistence of contextual information

	Same information given to models and expert	**Experts provided with more information than the models**
The model is less accurate than the average expert	Studies 2, 6, and 11	Studies 1, 3, 5, 7, 9, 10, and 13
The model is more accurate than the average expert	Study 12	Studies 4, 8, 14, 15, and 16

As can be seen, the majority of the studies were based on providing different inputs to the experts than to the models, which is what actually happens in real life software development contexts. Only four studies provided the same information to the models and the experts. Hence, it is difficult to draw conclusions about the importance of contextual information for the relative estimation performance of experts and models based on Table 3 alone. It is interesting to note that in three of the four studies the experts were more accurate than the models, even when they possessed the same information.

The importance of contextual information for the accuracy of the expert judgment-based effort estimates may be better illustrated by a comparison of the average accuracy (MAPE) of expert estimation-based effort estimates in the studies where the experts did not have contextual information (Studies 2, 6, 11 and 12), and the subset of the other studies that reported the MAPE (Studies 7, 8, 9, 10, 13, 14, 15, and 16). When the experts were given the same input as the models, the average MAPE is 157%. When the experts are given additional contextual information, the average MAPE is 36%. The two groups of studies may not be completely comparable, i.e., there may be differences in the estimation complexity, but the big difference in accuracy nevertheless suggests that the performance of the experts improves substantially with contextual information.

Few of the studies report results regarding the accuracy by type of estimation task. In fact, only two studies (Studies 3 and 8) report this type of information, and then only related to the size of the projects to be estimated, stating that larger projects are typically more difficult to estimate. The results of these two studies imply that the main benefit of estimation models is to avoid large overruns in situations known to induce a strong degree of overoptimism. This evidence is weak at present, but fits with common sense, which indicates that models are less affected by wishful thinking than software professionals are.

5. Concluding Remarks

In the reviewed studies, the models failed to systematically perform better than the experts when estimating the effort required to complete software development tasks. Possible reasons for this include:

- The experts have natural advantages in that they typically possess more information and are more flexible in how the information (or lack of information) is processed.
- It may be difficult to build accurate software development effort estimation models. In particular, the lack of stable relationships and the use of small learning data sets may easily lead to models' being overfitted to the available data. A tendency towards model overfitting may explain why the level of calibration to the organization when using the model seems to matter so much in relation to the estimation accuracy.

The models' ability to weight variables more correctly, to reduce biases, and to produce consistent estimates may consequently have been insufficient to compensate for the low quality of the models and their inability to use all of the relevant contextual information. The software development community is, consequently, still in a position where the evidence supports neither a replacement of models with expert judgment, nor a replacement of expert judgment with models. If, as suggested in MacDonell and Shepperd (2003), there is a high degree of independence between estimates based on common effort estimation models and expert judgment, and it is difficult to devise rules for selecting the most accurate estimation method, the solution seems to be to use a combination of models and experts.

Based on the modest evidence to date, two conditions for producing more accurate expert judgment-based effort estimates seem to be that the models are not calibrated to the organization using them, and that the experts possess important contextual information not included in the formal models and apply it efficiently. The use of models, either alone or in combination with expert judgment, may be particularly useful when i) there are situational biases that are believed to lead to a strong bias towards overoptimism; ii) the amount of contextual information possessed by the experts is low; and iii) the models are calibrated to the organization using them. Two of the reviewed studies evaluated a mechanical combination, and two studies a judgmental combination, of expert judgment and models. The results from these four studies suggest that combined estimates lead to accuracy levels similar to the best of the other estimation methods, regardless of type of combination.

So far, there have been two different types of responses to our findings. Most researchers outside the software engineering community seem to find it surprising that the models are not better than the experts, while most software engineering researchers and practitioners seem to find it surprising that the experts would not be even better in comparison with the models. Hopefully, our results will lead to more studies in domains similar to software development, leading to a better understanding of when to use a model and when to use expert judgment. There are still many important unanswered questions and claims with little or no evidence.

Acknowledgement: Thanks to Scott Armstrong, Fred Collopy, Jason Dana, Robin Dawes, Robin Hogarth, and Michael Roy for useful comments on drafts of this paper.

References

Anda, B., Benestad, H. C., & S. E. Hove. (2005). *A multiple-case study of effort estimation based on use case points*. ISESE 2005 (Fourth International Symposium on Empirical Software Engineering), Noosa, Australia, IEEE Computer Society: 407–416

Atkinson, K. & Shepperd, M. (1994). *Using function points to find cost analogies*. European Software Cost Modelling Meeting, Ivrea, Italy.

Bergeron, F. & St-Arnaud, J. Y. (1992). Estimation of information systems development efforts: a pilot study. *Information and Management*, *22* (4), 239-254.

Blattberg, R. C. & Hoch, S. J. (1990). Database models and managerial intuition: 50% model + 50% manager. *Management Science*, *36* (8), 887-899.

Bolger, F. & Wright, G. (1994). Assessing the quality of expert judgment: Issues and analysis. *Decision Support Systems*, *11* (1), 1-24.

Briand, L. C. & Wieczorek I. (2002). Resource estimation in software engineering. In J. J. Marcinak, *Encyclopedia of software engineering* (pp. 1160-1196). New York, John Wiley & Sons.

Brown, N. R. & Siegler R. S. (1993). Metrics and mappings: A framework for understanding real-world quantitative estimation. *Psychological Review*, *100* (3), 511-534.

Dawes, R. M., Faust, D., & Meehl, P. E. (1989). Clinical versus actuarial judgment. *Science*, *243*, 1668-1674.

Dolado, J. J. (2001). On the problem of the software cost function. *Information and Software Technology*, 43 (1), 61-72.

Goodwin, P. (2000). Improving the Voluntary Integration of Statistical Forecasts and Judgment. *International Journal of Forecasting*, *16* (1), 85-99.

Grimstad, S. & Jørgensen M. (2007). A Framework for the Analysis of Software Cost. In preparation.

Grove, W. M., Zald, D. H., Lebow BS, Snitz BE, & Nelson C.. (2000). Clinical versus Mechanical Prediction: A Meta-Analysis. *Psychological Assessment*, *12* (1), 19-30.

Heemstra, F. J. & Kusters, R. J. (1991). Function point analysis: Evaluation of a software cost estimation model. *European Journal of Information Systems*, *1*(4), 223-237.

Hihn, J. & Habib-Agahi, H. (1991). Cost estimation of software intensive projects: A survey of current practices. *International Conference on Software Engineering, Austin, TX, USA*. IEEE Comput. Soc. Press, Los Alamitos, CA, USA: 276-287.

Hogarth, R. M. (2001). *Educating Intution*. Chicago, University of Chicago Press.

Hogarth, R. M. (2005). Deciding Analytically or Trusting your Intuition? The Advantadges and Disadvantadges of Analytic and Intuitive Thought. In T. Betsch and S. Haberstroh. Mahwah,*The routines of decision making* (pp. 67-82). NJ, Erlbaum.

Jeffery, D. R., Ruhe, M. & Wieczorek I. (2000). A comparative study of two software development cost modeling techniques using multi-organizational and company-specific data. *Information and Software Technology*, *42* (14), 1009-1016.

Jørgensen, M. (1997). *An empirical evaluation of the MkII FPA estimation model*. Norwegian Informatics Conference, Voss, Norway, Tapir, Oslo: 7-18.

Jørgensen, M. (2004). A review of studies on expert estimation of software development effort. *Journal of Systems and Software*, *70* (1-2), 37-60.

Jørgensen, M. (2004). Top-down and bottom-up expert estimation of software development effort. *Information and Software Technology*, *46* (1), 3-16.

Jørgensen, M., Faugli, B., & Gruschke, T. (2007). Characteristics of software engineers with optimistic predictions. *Journal of Systems and Software*, to appear.

Jørgensen, M. & Shepperd, M. (2007). A Systematic Review of Software Development Cost Estimation Studies. *IEEE Transactions on Software Engineering*, 33(1), 33-53

Jørgensen, M. & Sjøberg, D. I. K. (2002). Impact of experience on maintenance skills. *Journal of Software Maintenance and Evolution: Research and practice, 14* (2), 123-146.

Jørgensen, M. & Sjøberg, D. I. K. (2004). The impact of customer expectation on software development effort estimates. *International Journal of Project Management*, *22*, 317-325.

Jørgensen, M., Teigen, K. H., & Kjetil Moløkken. (2004). Better sure than safe? Over-confidence in judgement based software development effort prediction intervals. *Journal of Systems and Software*, *70* (1-2), 79-93.

Kitchenham, B. (1992). Empirical studies of assumptions that underlie software cost-estimation models. *Information and Software Technology*, *34* (4), 211-218.

Kitchenham, B., Pfleeger, S. L., et al. (2002). An empirical study of maintenance and development estimation accuracy. *Journal of Systems and Software*, *64* (1), 57-77.

Kusters, R. J., van Genuchten, M.J.I. & Heemstra, F.J. (1990). Are software cost-estimation models accurate? *Information and Software Technology*, *32* (3), 187-190.

Lawrence, M. & O'Connor, M. (1996). Judgement or models: The importance of task differences. *Omega, International Journal of Management Science*, *24* (3), 245-254.

Lederer, A. L. & Prasad, J. (2000). Software management and cost estimating error. *Journal of Systems and Software*, *50* (1), 33-42.

MacDonell, S. G. & Shepperd, M. J. (2003). Combining techniques to optimize effort predictions in software project management. *Journal of Systems and Software*, *66* (2), 91-98.

Meehl, P. E. (1954). *Clinical versus Statistical Prediction: A Theoretical Analysis and a Review of the Evidence*. Minneapolis, US, University of Minnesota Press.

Meehl, P. E. (1986). Causes and effects of my disturbing little book. *Journal of Personality Assessment*, *50*, 370-375.

Moløkken-Østvold, K., & Jørgensen, M. (2003). A review of surveys on software effort estimation. *International Symposium on Empirical Software Engineering (ISESE 2003)*, Rome, Italy, IEEE Computer Society: 223-230.

Moores, T. T. & Edwards, J. S. (1992). Could large UK corporations and computing companies use software cost estimating tools? – A survey. *European Journal of Information Systems*, *1* (5), 311-319.

Mukhopadhyay, T., Vicinanza, S. S., & Prietula, M. J. (1992). Examining the feasibility of a case-based reasoning model for software effort estimation. *MIS Quarterly*, *16* (2), 155-171.

Murali, C. S. & Sankar C. S. (1997). Issues in estimating real-time data communications software projects. *Information and Software Technology*, *39* (6), 399-402.

Myrtveit, I. & Stensrud, E. (1999). A controlled experiment to assess the benefits of estimating with analogy and regression models. *IEEE Transactions on Software Engineering*, *25* (4), 510-525.

Nelson, E. A. (1966). *Management Handbook for the Estimation of Computer Programming Costs*. AD-A648750, Systems Development Corp.

Niessink, F. & van Vliet, H. (1997). *Predicting maintenance effort with function points*. International Conference on Software Maintenance, Bari, Italy, IEEE Comput. Soc, Los Alamitos, CA, USA: 32 - 39.

Ohlsson, N., Wohlin, C., et al. (1998). A project effort estimation study. *Information and Software Technology*, *40* (14), 831-839.

Pengelly, A. (1995). Performance of effort estimating techniques in current development environments. *Software Engineering Journal*, *10* (5), 162-170.

Ribu, K. (2001). *Estimating object-oriented software projects with uses cases*. Informatics, University of Oslo. MSc thesis.

Sanders, D. E. & Ritzman, L. P. (1991). On knowing when to switch from quantitative to judgemental forecasts. *International Journal of Forecasting*, *11* (6), 27-37.

Shanteau, J. (1992). Competence in experts: The role of task characteristics. *Organizational Behaviour and Human Decision Processes*, *53* (2), 252-266.

Shepperd, M. & Kadoda, G. (2001). Comparing software prediction techniques using simulation. *IEEE Transactions on Software Engineering*, *27* (11), 1014-1022.

Vicinanza, S. S., Mukhopadhyay, T., & Prietula, M. (1991). Software effort estimation: An exploratory study of expert performance. *Information Systems Research*, *2* (4), 243-262.

Walkerden, F. & Jeffery, R. (1999). An empirical study of analogy-based software effort estimation. *Empirical Software Engineering*, *4* (2), 135-158.

Wallner, W. S. & Zimbelman M. F. (2003). A cognitive footpring in archival data: Generalizing the dilution effort from laboratory to field settings. *Organizational Behavior and Human Decision Processes*, *91*, 254-268.

Webby, R. G. & O'Connor, M. J. (1996). Judgemental and Statistical Time Series Forecasting: A Review of the Literature. *International Journal of Forecasting*, *12* (1), 91-118.

Whitecotton, S. M., Sanders, D. E., & Norris K.B. (1998). Improving predictive accuracy with a combination of human intuition and mechanical decision aids. *Organizational Behaviour and Human Decision Processes*, *76* (3), 325-348.

Appendix A: Review of the studies

Template used in the review:

Factor	Description of factor
DESIGN ISSUES	
Study design	Description of the type of study, subjects and material, and the sequence of steps.
Estimation method selection process	Description of how the estimation method, e.g., a particular model or the estimation experts, was selected.
Estimation models	Name of the included estimation models.
Calibration level	“Low”, “medium” or “high”. See Section 2.4 for explanation.
Model use expertise	Description of the experience and/or expertise involved in using the estimation model.
Expert judgment process	Description of the steps involved in the expert judgment.
Expert judgment estimation expertise	Description of the experience and/or expertise of the expert in expert judgment-based effort estimation.
Possible motivational biases in estimation situation	Description of any elements potentially biasing the effort estimates.
Estimation input	Description of the type of information provided as input to the experts and the models.
Contextual information	Description of the information possessed by the experts, but not used as input to the estimation models.
Estimation complexity	Description of the difficulty of the estimation task.
Fairness limitations	Description of introduced study elements that, in comparison to a “natural” estimation setting, may favour either models or experts.
Other design issues	Issues of importance not covered by the above design factors.
RESULTS	
Accuracy	Estimation accuracy results as reported by the studies.
Variance	Variance in estimation accuracy among the set of experts and the set of models.
Other results	Issues of importance not covered by the above factors.

Study 1: Kusters, Genuchten et al. (1990)

Factor	Description of factor
DESIGN ISSUES	
Study design	Laboratory-based comparison. Fourteen software professionals (project leaders) and two estimation models. One project estimated. Sequence: 1) The project leaders received information about the project. 2) The project leaders estimated the effort required based on expert judgment. 3) The project leaders produced two model-based estimates, one for each of the two models. 4) The project leaders combined the three estimates, based on judgment, to give a final effort estimate.
Estimation method selection process	The model selection was based on three criteria: i) a tool supporting the model must exist; ii) the model must be developed specially for estimating software development projects; and iii) the model must be applicable at an early stage of the software development. Two out of four of the evaluated models satisfied these criteria. There is no description of how the project leaders were selected.
Estimation models	The two commercial estimation tools BYL and Estimacs.
Calibration level	Low (both models)
Model use expertise	Not described, but probably low. No training in use of the models reported.
Expert judgment process	Described as based on “knowledge and experience”.
Expert judgment estimation expertise	Subjects described as experienced project leaders. Expertise related to the project to be estimated not described.
Possible motivational biases in estimation situation	No biases described or likely from the description of the estimation situation.
Estimation input	A requirement specification (3 pages with text + 14 diagrams on dataflow).
Contextual information	Probably relevant contextual information present in the requirement specification.
Estimation complexity	Not described. Probably more difficult than in real life settings, where more information from clients and colleges can be collected when needed.
Fairness limitations	The experts were prevented from collecting the information they would typically use. Untrained in the use of the models.
Other design issues	The model was used by the estimators *after* they had produced an expert judgment-based effort estimate. This may have affected the input and the use of the models.
RESULTS	
Accuracy	The actual effort of the project was eight man-months. The mean expert judgment-based estimate was 28.4 man-months, the mean BYL estimate was 27.7 man-months, the mean Estimacs estimate was 48.5 man-months, and the mean final (judgment-based combination of expert and model-output) estimate was 27.7 man-months.
Variance	One of the models (Estimacs) yielded much less accurate estimates than the other estimation methods.
Other results	There was almost as high a variance in estimates for the same project data when applying the models as when applying expert judgment. This may indicate that the use of the models was far from “mechanical”. Eleven out of the 16 project leaders reported that they would use these models if they were available to them. The typical reason for use was the models’ usefulness as a means of communication and checklists.

Study 2: Vicinanza, Mukhopadhyay et al. (1991)

Factor	Description of factor
DESIGN ISSUES	
Study design	Laboratory-based comparison. Five software professionals (S1, … , S5) and two models. Ten projects estimated. Sequence: 1) Software professionals asked to sort the importance of "cost factors", i.e., project and environmental attributes with relevance for the effort. 2) Presentation of the cost factors of a project. 3) The software professionals estimated the effort of a project while "verbalizing their thoughts". The same project was estimated using the two models. 4) Steps 2) and 3) were repeated for all ten projects.
Estimation method selection process	The model selection process was not described. The experts were selected from five different companies based on the following criteria: i) minimum of ten years of experience as software professional; ii) sufficient experience with project management and effort estimation; iii) a reputation among their peers as an individual who had consistently made accurate estimates.
Estimation models	The two noncommercial models COCOMO and Function Points
Calibration level	Low (both models)
Model use expertise	Not described. Probably high, since the authors were the model users. The format of the estimation input most likely led to a rather mechanical use of the models.
Expert judgment process	Different strategies were applied by different experts. S1, S3, S4 and S5 appeared to apply an algorithmic strategy, i.e., a strategy similar to that of many estimation models. S2 (the most accurate expert) applied an analogy-based strategy.
Expert judgment estimation expertise	Except for S1 (by far the least accurate estimator), all software professionals had some experience with projects similar to those estimated. S2 had the most relevant experience of similar projects.
Possible motivational biases in estimation situation	No biases described or likely from the description of the estimation situation.
Estimation input	Project data on model input format, i.e., no textual requirement specification. Thirty-seven project attributes were presented, including the actual size of the delivered systems (in lines of code and functionality).
Contextual information	None.
Estimation complexity	Not described. Probably substantially more difficult than in real life settings for the experts, due to the unfamiliar format of the estimation input and the lack of relevant contextual information.
Fairness limitations	The lack of contextual information means that the models have an advantage not present in field settings.
Other design issues	Both the expert and the estimation models received more precise input than in field settings, e.g., they received the actual, instead of the estimated, number of lines of code of the software to be produced. It is not clear how this affected the comparison of model and expert judgment.
RESULTS	
Accuracy	The estimation accuracy rank (measures as MAPE) was as follows: S2 (32.0%), S5 (65.5%), Function Point Model (107.4%), S4 (140.6%), S3 (146.1%), COCOMO Model (758.2%), S1 (1106.7%).
Variance	One of the experts (S1) and one of the models (COCOMO) produced extremely inaccurate effort estimates.
Other results	The extremely low accuracy of S1 may have been caused by his experience of another software domain (military software).

Study 3: Heemstra and Kusters (1991)

Factor	Description of factor
DESIGN ISSUES	
Study design	Field study based on questionnaires to 597 Dutch companies. Organizations applying estimation models were compared with those relying more on the use of expert judgment.
Estimation method selection process	Not described.
Estimation models	The non-commercial model Function Points.
Calibration level	Not described. Probably low or medium.
Model use expertise	Not described.
Expert judgment process	Not described.
Expert judgment estimation expertise	Not described.
Possible motivational biases in estimation situation	Not described. It is probable that many project estimates had motivational biases towards over-optimism.
Estimation input	Not described. Probably textual requirement specifications and meetings with the clients.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity.
Fairness limitations	Not reported whether there are other, unexamined, differences between the organizations applying models and those relying on expert judgment. The organizations applying models may, for example, be more mature and concerned about estimation accuracy. It is not known how this may affect the fairness of comparison.
Other design issues	The study makes comparisons at the level of organizations, not of projects. The relationship to the estimation method is therefore indirect, and causal relationships between the estimation method and the accuracy are difficult to establish.
RESULTS	
Accuracy	There were lower effort overruns in organizations that relied more on expert judgment. As an illustration, the proportion of low effort overruns (less than 10%) was: i) 66.7% for model users and 79.5% for nonmodel users on small projects, ii) 43.5% for model users and 51.4% for nonmodel users on medium-sized projects, iii) 27.7% for model users and 50% for nonmodel users on large projects, and iv) 33.3% for model users and 47.6% for nonmodel users on very large projects.
Variance	Not reported.
Other results	Very large projects seem to have smaller effort overruns (more than 50% overruns) in organizations that apply estimation models. As an illustration, while none of the very large projects in organizations that applied estimation models (n=21) had more than 100% effort overruns, the corresponding proportion in organizations that relied more on expert judgment was 9.5%.

Study 4: Lederer and Prasad (2000)

Factor	Description of factor
DESIGN ISSUES	
Study design	Field study based on questionnaires to 112 software organizations, requesting information about large projects. Different types of expert judgment compared with estimation models.
Estimation method selection process	Not described.
Estimation models	The following estimation practices were interpreted as the use of an estimation model: the use of complex statistical formula, a software package for estimating, or established standards.
Calibration level	Not described.
Model use expertise	Not described.
Expert judgment process	The following estimation practices were interpreted as the use of expert judgment: intuition, comparison with similar projects, the use of past projects based on personal memory, and guessing.
Expert judgment estimation expertise	Not described.
Possible motivational biases in estimation situation	Not described. Probably many projects with motivational biases towards over-optimism.
Estimation input	Not described. Probably textual requirement specifications and meetings with the clients.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity. The study focused on large projects. This probably implies a bias towards tasks of more than average complexity.
Fairness limitations	It is possible that organizations that use estimation models more frequently differ from the norm.
Other design issues	The study makes comparisons at the level of organizations, not of projects. The relationship to the estimation method is therefore indirect, and causal relationships between the estimation method and the accuracy are difficult to establish.
RESULTS	
Accuracy	No significant correlation was found between the use of estimation models and the proportion of overrun in an organization. A significant, positive correlation was found ($r=0.19$, $p<0.05$) between the use of expert judgment and the proportion of projects with an effort overrun. The results discouraged using intuition and guessing on large projects. On the other hand, the study reported no significant, positive effect on the proportion of overruns from the use of estimation models.
Variance	Not reported.
Other results	The only factor that was correlated with a lower proportion of effort overrun was the level of estimation accuracy accountability. The implication of this is summarized as follows: "*... the research suggests somewhat ironically that the most effective approach to improving the estimating accuracy may be to make estimators, developers and managers more accountable for the estimate, even though it may be impossible to direct them explicitly on how to produce a more accurate one.*"

Study 5: Bergeron and St-Arnaud (1992)

Factor	Description of factor
	DESIGN ISSUES
Study design	Field study based on questionnaires to 89 software professionals. Information about their last project that required more than 150 person-days for development was collected. Expert judgment was compared with estimation models.
Estimation method selection process	Not described.
Estimation models	Different estimation method categories are surveyed. Only one of these can, with high confidence, be categorized as model-based, i.e., the "algorithmic" category.
Calibration level	Not described. Probably low.
Model use expertise	Not described.
Expert judgment process	Different categories are surveyed. At least the following two belong to expert judgment: personal experience and expert judgment.
Expert judgment estimation expertise	On average, six years of estimation experience.
Possible motivational biases in estimation situation	Not described. Probably many projects with motivational biases towards over-optimism.
Estimation input	Not described. Probably textual requirement specifications and meetings with the clients.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity.
Fairness limitations	It is possible that projects that use estimation more frequently use models that differ from those used in other projects with respect to their estimation complexity.
Other design issues	The use of a method in each project is not binary (yes/no) but on a four point scale: 1 = not used, 2 = low importance, 3 = medium importance, 4 = high importance.
	RESULTS
Accuracy	No significant correlation was found between the use of expert judgment or algorithmic models and the absolute estimation error in early stages of the project (feasibility phase or preliminary phase). In the planning phase (functionality phase), the use of expert judgment was significantly correlated with a lower absolute estimation error (r=0.46, p=0.01), while the use of algorithmic models was significantly correlated with higher absolute estimation errors (r=0.27, p=0.08).
Variance	Not reported.
Other results	

Study 6: Mukhopadhyay, Vicinanza et al. (1992)

Factor	Description of factor
	DESIGN ISSUES
Study design	Laboratory-based comparison. One software professional, one model of the expert, and two estimation models. Five projects estimated. Sequence: 1) One software professional estimated ten projects while "thinking aloud" (this part of the experiment is part of Study 2). 2) The researchers constructed a model (Estor) of the expert using the "case-based reasoning form of analogical problem solving as a theoretical model". 3) The software professional estimated five projects similar to the previous ten projects. 4) The same five projects were estimated by applying Estor (the model of the expert), and the two models COCOMO and Function Points.
Estimation method selection process	The process for model selection was not described. The most accurate expert (S2) from Study 2 was selected as a reference expert.
Estimation models	Estor, COCOMO and Function Points
Calibration level	High (Estor), Low (COCOMO and Function Points)
Model use expertise	Not described. Probably high, since the paper authors were the model users. The format of the estimation input most likely led to a rather mechanical use of the models.
Expert judgment process	Analogy-based strategy.
Expert judgment estimation expertise	Some relevant expertise from similar projects.
Possible motivational biases in estimation situation	No biases described or likely from the description of the estimation situation.
Estimation input	Project data on COCOMO and Function Point estimation model input format, i.e., no textual requirement specification. Thirty-seven project attributes presented, including the actual size of the delivered systems (in lines of code and functionality).
Contextual information	None.
Estimation complexity	Not described. Probably more difficult than in real-life settings for the expert, due to the unfamiliar format of estimation input and the lack of relevant contextual information.
Fairness limitations	The lack of contextual information means that the models have an advantage not present in field settings.
Other design issues	Both the expert and the estimation models received more precise input than in field settings, e.g., they received the actual number of the lines of code of the software to be produced, instead of simply an estimate. It is not clear how this affects the comparison between the use of models and expert judgment.
	RESULTS
Accuracy	The estimation accuracy rank (measured as MAPE) was as follows: The expert (30.72%), The model of the expert – Estor (52.79%), Function Points (102.74%), COCOMO (618.99%).
Variance	One of the models (COCOMO) yielded very inaccurate results.
Other results	The verbal protocol contained little information about the verbal processes involved, and it was difficult to model the mental processes of the expert. The consequence was that the model of the expert (Estor) sometimes selected analogies which were different from those selected by the expert. The authors recommend the use of Estor as an expert *support* system.

Study 7: Atkinson and Shepperd (1994)

Factor	Description of factor
DESIGN ISSUES	
Study design	Mix of laboratory and field settings. Estimates provided by software professionals in field settings within one company were compared with those of four models in laboratory settings. Twenty-one projects were estimated. Sequence of the laboratory part of the study (cross-validation): 1) The project to be estimated was removed from the data set 2) The remaining twenty projects were used as the basis for the selection of project analogies as input to three of the models. 3) The effort required for the project was estimated by all four models. 4) Steps 1)-3) were repeated for all projects.
Estimation method selection process	Not described.
Estimation models	Three self-developed analogy-based estimation models and the Function Points model.
Calibration level	High (analogy-based estimation models), low (Function Points)
Model use expertise	Not described. Probably high, at least for the self-developed models.
Expert judgment process	Not described.
Expert judgment estimation expertise	Not described.
Possible motivational biases in estimation situation	The expert judgment-based effort estimates were probably subject to typical field setting-biases, e.g., bias towards over-optimism.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about project requirement. The input to the models contained information about the properties of the completed software system, e.g., the actual size of the completed software.
Contextual information	The expert probably had a lot of contextual information not included in the models.
Estimation complexity	Not described.
Fairness limitations	The estimation models were based on software size information only known at the completion time, while the experts only possessed the information available at the start of the project. The cross-validation method implies that the models apply information from future as well as from previous projects, while the experts only apply information about previous projects. The experts' estimates (field use) may affect the actual use of effort, while this is not possible for the models (laboratory use).
Other design issues	
RESULTS	
Accuracy	The estimation accuracy rank (measured as MAPE) was as follows: the best analogy-based model (8%), the experts (39%), the other two analogy-based models (50% and 68%), and the Function Points model (99%).
Variance	Higher variance among the models, even when the low calibration model Function Points is excluded from the comparison.
Other results	

Study 8: Jørgensen (1997)

Factor	Description of factor
	DESIGN ISSUES
Study design	Field study within one company. Expert judgment compared with one estimation model. Twenty-six projects were estimated.
Estimation method selection process	Not described.
Estimation models	Function Points.
Calibration level	Medium.
Model use expertise	Probably high. The company had an estimation team devoted to the use of the model.
Expert judgment process	Not described.
Expert judgment estimation expertise	Not described.
Possible motivational biases in estimation situation	The expert judgment-based effort estimates were probably subject to typical field setting-biases, e.g., a bias towards over-optimism.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about project requirements. The input to the models contained the information necessary for using the Function Point estimation method.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity.
Fairness limitations	Unknown assignment of estimation method to projects of different estimation complexity. Not known how this may affect fairness of comparison.
Other design issues	It is unclear to what degree the actual use of the model was, in reality, a combination of model and expert judgment (see “Other results”)
	RESULTS
Accuracy	The model-based (Funtion Points) effort estimates were more accurate. The mean accuracy (MAPE) of the model was 10%, while that of the experts was 20%.
Variance	All the largest estimation errors (>30% deviation) were connected with the use of expert judgment. The most important benefit of using the model seemed to be the avoidance of large effort overruns.
Other results	A mechanical use of the Function Point estimation method, i.e., exactly as prescribed in the textbooks, yielded an estimation accuracy (MAPE) of 58%. This may either illustrate the benefits of using expert judgment as the input to the model or suggest that the field use was, in reality, a combination of expert judgment-based and model-based estimates.

Study 9: Niessink and van Vliet (1997)

Factor	**Description of factor**
	DESIGN ISSUES
Study design	A mixed field and laboratory study within one company. Expert judgment (field setting) was compared with estimation models (laboratory setting). One hundred and forty software maintenance projects estimated. Sequence: 1) Collection of information about one hundred and forty previously estimated and completed projects. 2) Development and cross validation-based evaluation of an analogy-based estimation model based on the projects. 3) Calibration and evaluation of the Function Point-based model. Not clear exactly how the data set was used to calibrate and evaluate the Function Point model.
Estimation method selection process	Not described.
Estimation models	Analogy and Function Point-based models.
Calibration level	High (analogy) and medium (Function Point)
Model use expertise	Probably high (the authors themselves). The study design most likely led to a mechanical use of the models.
Expert judgment process	Not described.
Expert judgment estimation expertise	Not described.
Possible motivational biases in estimation situation	The expert judgment-based effort estimates were probably subject to typical field setting-biases, e.g., bias towards over-optimism.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about the project requirements. The input to the models contained the information necessary for using the Function Point estimation method.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity.
Fairness limitations	The cross-validation method implies that the models apply information from the future, as well as from previous projects, while the experts only apply information about previous projects. The experts' estimates (field use) may affect the actual use of effort, while this is not possible for the models (laboratory use). The tool producing the analogy-based method develops and tests several analogy-based models and presents only the model with the highest accuracy.
Other design issues	
	RESULTS
Accuracy	The (best) analogy-based model (MAPE = 40%) was more accurate than the experts (MAPE = 57%). The least accurate estimates were produced by the Function Point-based estimation method (the best version had a MAPE of 71%).
Variance	Different alternative methods of calibrating the Function Point model to the data set were evaluated. Some of the alternatives were highly inaccurate.
Other results	

Study 10: Ohlsson, Wohlin et al. (1998)

Factor	**Description of factor**
DESIGN ISSUES	
Study design	Field study of student projects. Expert judgment was compared with one simple estimation model. The same project was estimated and executed by seven project teams, each with 17-19 members.
Estimation method selection process	Not described.
Estimation models	Estimated effort based on the effort of the same project the year before, i.e., similar to analogy-based models.
Calibration level	High.
Model use expertise	High (the authors themselves). The use of the very simple model was mechanical.
Expert judgment process	Described as being based on own "knowledge and experience". Probably based on group estimation processes. The students received the data from the previous year's project teams as support for their own estimates. This information probably acted as an estimation anchor.
Expert judgment estimation expertise	Probably low (students).
Possible motivational biases in estimation situation	Not described. Student projects are different from commercial projects in that they have a strong element of training. This means that the students may find it advantageous to spend more time on the project than is necessary in order to learn more, i.e., that an optimistic estimate of the effort is *higher* than the most likely effort and not lower, as in commercial projects.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about the project requirements. The input to the models contained the amount of effort used on the same project in the previous year.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity.
Fairness limitations	
Other design issues	Estimated effort seems to mean the same as planned effort in this study.
RESULTS	
Accuracy	The expert judgment-based estimates were, on average, slightly more accurate, with a MAPE of 15.8%, as compared to the model-based MAPE of 19.2%.
Variance	Similar variance in estimation error.
Other results	

Study 11: Walkerden and Jeffery (1999)

Factor	Description of factor
DESIGN ISSUES	
Study design	Laboratory-based comparison. Twenty-five students and two models with many variants of each. Two projects estimated. Sequence: 1) Construction of 25 subsets of 15 projects, i.e., 25 different training sets. 2) The expert receives a training subset. 3) The expert estimates the effort required for two of the remaining four projects. Each estimate should start with a selection of the two projects most similar to the one to be estimated. 4) The model-based effort estimates were calculated based on the same subset.
Estimation method selection process	Not described.
Estimation models	Two types of analogy-based estimation models (ANGEL and ACE) and a linear regression-based model.
Calibration level	High (all models)
Model use expertise	Probably high (the authors themselves). The study design most likely led to a mechanical use of the models.
Expert judgment process	The students were instructed to follow an analogy-based estimation process, i.e., to first select analogies, then estimate the effort.
Expert judgment estimation expertise	Low. Use of students and data in a format unfamiliar to software professionals.
Possible motivational biases in estimation situation	Not described.
Estimation input	Information in a model input format.
Contextual information	None.
Estimation complexity	Not described.
Fairness limitations	The lack of contextual information means that the models have an advantage not present in a field setting. Inexperienced estimators.
Other design issues	A model based on the experts' selection of analogies was also evaluated.
RESULTS	
Accuracy	The model based on the experts' selection of analogies was the most accurate (MAPE = 39%). One of the self-developed models was the second most accurate (MAPE = 46%). The experts achieved accuracy similar to, or better than, (MAPE = 56%) the rest of the model variants (MAPEs of 55%, 60%, 67%, 68%, 100, and 114%).
Variance	The linear regression model most frequently had the largest estimation error. All models had situations where they yielded the most accurate estimates and where they yielded the least accurate effort estimate.
Other results	Some of the students made simple errors leading to inaccurate estimates, e.g., that a project with a total effort of 250 person-hours and team size of 2 meant that each person would need 500 person-hours.

Study 12: Myrtveit and Stensrud (1999)

Factor	**Description of factor**
	DESIGN ISSUES
Study design	Laboratory-based comparison. Sixty-eight software professionals and two models. Each software professional estimated one project from a set of forty-eight projects. Sequence: 1) Each software professional was allocated to a project to be estimated based on a random selection. 2) Model-based estimates (analogy and regression-based) were produced for the project, based on the set of forty-seven remaining projects. 3) The project's effort was estimated by applying expert judgment and using the information about the other forty-seven projects. 4) The project's effort was estimated by using the estimate derived by the analogy-based model as the input to the estimation process (judgment-based combination). 5) The project's effort was estimated by using the estimate derived by the regression-based model as the input to the estimation process (judgment-based combination).
Estimation method selection process	Not described.
Estimation models	Analogy-based model (ANGEL) and regression-based model.
Calibration level	High (both models).
Model use expertise	The people building the models and producing the model estimates were researchers experienced in the use of estimation models. The study design most likely led to a mechanical use of the models. The users of the output from the model in the judgment-based combination had received an introduction to the models, but were hardly experienced users.
Expert judgment process	Not described.
Expert judgment estimation expertise	All but seven had estimated projects before, but only ten had previously estimated projects similar to the type estimated in the experiment.
Possible motivational biases in estimation situation	Not described.
Estimation input	Information in a model format, i.e., quantitative factors describing the project.
Contextual information	Probably none.
Estimation complexity	High variation.
Fairness limitations	The lack of contextual information meant that the models have an advantage not present in field setting.
Other design issues	
	RESULTS
Accuracy	One of the judgmental combination-based estimates (Step 4) and one of the models (regression-based) were the most accurate (with MAPEs of 126% and 127% respectively). The other judgmental combination and the other model (analogy-based) were the second most accurate (with MAPEs of 136% and 154% respectively). Expert judgment was the least accurate (MAPE=243%). These results are based on the average of *all* subjects. The paper presents subcategories and adjustments of results which alter these results slightly, e.g., the expert judgment of the senior group experts was more accurate than the analogy-based model, but not that of the regression-based model. The average accuracy of the senior group was a MAPE of 94%, while that of the junior group was as high as MAPE=321%.
Variance	The inexperienced experts (junior group) had the most inaccurate effort estimates. The analogy-based model had a less accurate "worst case" than that of the experienced experts (senior group).
Other results	The experts, on average, believed that no benefit was derived from using a combination of expert judgment and the estimation models. This is in sharp contrast to the actual improvement in estimation accuracy.

Study 13: Kitchenham, Pfleeger, et al. (2002)

Factor	Description of factor
	DESIGN ISSUES
Study design	A mixed field and laboratory study within one company. Expert judgment (field setting) was compared with the average of estimation methods (field settings), and models (laboratory setting). One hundred and forty-five software maintenance projects were estimated.
Estimation method selection process	In the field setting, the project manager discussed with an independent estimator to determine which estimation method (of eight different estimation methods) might be best suited to the project. Usually, two estimation methods were used when projects were less than 200 work-hours, and three methods otherwise. The selection of the estimation model for the laboratory use was based on a comparison of the accuracy resulting from different model variants and data sets.
Estimation models	In the field setting, few projects applied an estimation model (too few for a meaningful analysis, other than the analysis of when models were used in combination with expert judgment). The models most used in the field setting were CA-Estimacs and Function Points. The model used in the laboratory setting was a Function Point-based model calibrated using regression analysis.
Calibration level	High (both in field setting models and with the laboratory models).
Model use expertise	Not described. Probably high (both field and laboratory use).
Expert judgment process	The process included the selection of an estimation method, a review of the estimate by people in different roles, and documentation of the basis of the estimate.
Expert judgment estimation expertise	Not described. Probably high.
Possible motivational biases in estimation situation	Not described. The bias towards over-optimism and self-fulfilling prophecy may be lower in maintenance projects compared with development project situations.
Estimation input	Not described. The expert judgment-based effort estimates were probably based on textual and oral information about project requirements. The input to the models contained the information necessary for using the models.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described. Probably a high variation in task size and complexity.
Fairness limitations	It is possible that the simplest projects were those applying expert judgment instead of the "average" method.
Other design issues	
	RESULTS
Accuracy	In the field setting, the expert judgment-based estimates were slightly more accurate than the "average" effort estimates (MAPE = 25% vs MAPE = 27%), with a significant difference (p=0.05) when the median instead of the mean APE is compared. Both types of field estimate were more accurate than those produced (in laboratory use) by the Function Point-based model (MAPE = 44%).
Variance	There was lower variance in the estimation error (lower proportion of estimates with 25% or less error) of the "average" compared to the expert judgment-based effort estimates. The highest variance occurred with the laboratory use of the Function Point-based model.
Other results	

Study 14: MacDonell and Shepperd (2003)

Factor	Description of factor
DESIGN ISSUES	
Study design	A mix of laboratory and field settings. The estimates provided by software professionals (the project managers) in field settings within one company were compared with those of two estimation models in laboratory settings. Twenty-six tasks were estimated. Sequence of the laboratory part of the study: 1) Data set divided into two variants of training sets (51 tasks in each set) and two variants of validation sets (26 tasks in each set). 2) Two regression and two case-based reasoning models were developed based on the training sets. 3) The effort of the tasks in the two validation sets were calculated by both models.
Estimation method selection process	The authors describe the selection of the models as being based on representative and contrasting concerns.
Estimation models	Two self-developed models. One was based on linear regression and the other was based on case-based reasoning, applying the tool ANGEL. The models are described as using straightforward model-building techniques.
Calibration level	High (both models)
Model use expertise	Not described. Probably high (self-developed models applied by the authors themselves). The study design most likely led to a mechanical use of the models.
Expert judgment process	Not described.
Expert judgment estimation expertise	Not described.
Possible motivational biases in estimation situation	The experts underestimated the tasks, i.e., not the typical tendency towards over-estimation. It is possible that there were motivational biases towards under-estimation in this particular field setting.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about the task requirements. The models had information available that was derived from the task requirements on a format suitable for model input.
Contextual information	The experts probably had a lot of contextual information that was not included in the models.
Estimation complexity	Not described.
Fairness limitations	The low accuracy of the expert judgment-based estimates is surprising in light of the reported accuracy in seemingly similar situations, e.g., in situations with effort estimation of extensions to an existing large system within one organization; see, for example, Study 13. This, together with the reported strong tendency towards under-estimation, may point to situational biases or low expertise of the expert estimators.
Other design issues	The study focuses on the combination of estimates, not on finding the best estimation method.
RESULTS	
Accuracy	The average of the estimates from the three estimation methods was the most accurate, with a sum absolute error of 214 and 226 work-hours for validation sets 1 and 2, respectively. The best single method-based estimation accuracy was achieved with the use of regression analysis (sum absolute error of 241 and 223), and case-based reasoning (sum absolute error of 254 and 360). The least accurate estimates were those based on expert judgment (sum absolute error of 296 and 410).
Variance	There was greater variance in accuracy among the expert judgment-based estimates than among the model-based estimates.
Other results	When one estimation method estimated inaccurately, one or both of the other methods tended to estimate significantly better. The authors tried to identify rules that could support the selection of the most accurate estimation method, but were not able to find any.

Study 15: Ribu (2001); Anda, Benestad et al. (2005)

Factor	**Description of factor**
	DESIGN ISSUES
Study design	A mix of field and laboratory studies. Expert judgment (field setting) was compared with a model (laboratory setting). Ten projects were estimated. Four of the projects were based on the same requirement specification.
Estimation method selection process	Not described.
Estimation models	Case Point-based model
Calibration level	Low
Model use expertise	Probably high. The study design was most likely led to a mechanical use of the models.
Expert judgment process	Not described. One of the expert estimates may have been supported by a model.
Expert judgment estimation expertise	Probably high.
Possible motivational biases in estimation situation	The expert judgment-based effort estimates were probably subject to typical field setting-biases, e.g., bias towards over-optimism.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about project requirements. The input to the models contained the information necessary for using the Use Case Point estimation model.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described.
Fairness limitations	
Other design issues	It is not reported whether, or to what extent, the models had essential information not available at the time of expert judgment-based estimation.
	RESULTS
Accuracy	The model-based estimates were more accurate than the expert judgment-based ones (MAPE = 0.23 vs MAPE = 0.34).
Variance	The expert judgment-based estimates had a stronger tendency towards strong over-optimism.
Other results	The results of this study deviate from those of the other studies in that the model was not calibrated to the organizations using them, but still produced the most accurate estimates. One reason for this may be that the model was based on projects quite similar to those in the study.

Study 16: Grimstad and Jørgensen (2007)

Factor	Description of factor
DESIGN ISSUES	
Study design	Field study. Expert judgment compared with model use. Eighteen projects estimated.
Estimation method selection process	Not described.
Estimation models	Simple rule-based model.
Calibration level	Medium.
Model use expertise	Probably high.
Expert judgment process	Not described.
Expert judgment estimation expertise	Probably high.
Possible motivational biases in estimation situation	The effort estimates were probably subject to typical field setting-biases, e.g., bias towards over-optimism.
Estimation input	The expert judgment-based effort estimates were probably based on textual and oral information about project requirements. The input to the models contained a judgment-based classification of the elements of the software to be developed.
Contextual information	Not described. Probably much of the contextual information in the requirement specification and meetings was not provided as input to the estimation models.
Estimation complexity	Not described.
Fairness limitations	Different types of projects were estimated using expert judgment and the models. Some of the effects of this difference are, however, adjusted for in the reporting of the results.
Other design issues	
RESULTS	
Accuracy	When not adjusting for differences in estimation complexity, the models yielded more accurate estimates than did the experts (MAPE = 0,07 vs MAPE = 0,18). When comparing only projects with similar estimation complexities, the estimation accuracy is similar, e.g., when comparing only projects where the estimator had some previous experience, the MAPEs for the model and expert judgment are MAPE = 0,07 and 0,10, respectively.
Variance	Similar variance in estimation accuracy.
Other results	Illustrates the importance of adjusting for potential biases in methods of selecting estimation methods.