Failure Factors of Software Projects at a Global Outsourcing Marketplace

Magne Jørgensen
Simula Research Laboratory and University of Oslo.
P.O.Box 134, NO-1325 LYSAKER, Norway.
E-mail: magnej@simula.no

Abstract: The presented study aims at a better understanding of when and why software projects fail. The analysis is based on a data set of 785,325 small-scale software projects at a global outsourcing marketplace. A binary logistic regression model relying solely on information known at the time of a project's start-up correctly predicted 74% of the project failures and 67% of the non-failures. The model-predicted failure probability corresponded well with the actual frequencies of failures for most levels of failure risk. The model suggests that the factors connected to the strongest reduction in the risk of failure are related to previous collaboration between the client and the provider and a low failure rate of previous projects completed by the provider. We found the characteristics of the client to be almost as important as those of the provider in explaining project failures and that the risk of project failure increased with an increased client emphasis on low price and with an increased project size. The identified relationships seem to be reasonable stable across the studied project size categories, which indicate that the results may potentially be applicable to larger projects than the small-scale outsourcing projects dominating this data set.

Index Terms: Outsourcing, Project failures, Risk management

1. INTRODUCTION

A great deal of resources are spent on software projects that fail to deliver useful functionality. For example, the proportion of started and then cancelled projects, sometimes termed "aborted" or "abandoned" projects, is reported to be 9% (Sauer, Gemino et al. 2007), 11% (Tichy and Bascom 2008), and 11.5% (El Emam and Koru 2008). Several non-peer reviewed reports claim a much higher proportion of cancelled software projects, but may be less reliable or less representative of the population of software projects. The frequently cited Standish Group Chaos Report (1995), for example, claims that as many as 31% of all software projects get cancelled. The low reliability of that report is discussed in several sources (Jørgensen and Moløkken-Østvold 2006; Eveleens and Verhoef 2010). While the cancellation rates described in the Standish Group Chaos Reports and similar non-peer reviewed surveys are likely to be exaggerated, there is no doubt that the proportion of cancelled projects is substantial.

The definition of a failed project in software surveys typically includes both cancelled projects and projects completed with a very poor product or process quality. Consequently, the reported failure rates appear higher than the corresponding cancellation rates. Exactly how much higher depends on the failure criteria used. For example, El Emam and Koru (El Emam and Koru 2008) categorized a project as having failed if it received a score of "poor" or "fair" in four out of five of the following performance criteria: user satisfaction, ability to meet budget targets, ability to meet schedule targets, product quality and staff productivity. This definition led to a failure rate of more than twice the cancellation rate for the same set of projects, i.e., a failure rate of 26% for the data set reporting a cancellation rate of 11.5%. Defining every project that does not deliver the specified product, is over budget, or is not on time as a failure, as is the case in several reports, typically amounts to 50-80% of all software projects being failures. For an overview of software failure surveys see (Hashmi and Stevrin 2009).

The challenge of defining project failures meaningfully is further illustrated in (Boehm 2000), where Barry Boehm makes the reasonable claim that not all cancellations should be considered to be failures. There may, for example, be good reasons for cancelling a well-managed project if the project's original assumptions of usefulness are no longer valid. In that case, the failure would clearly be to continue a project that is no longer needed instead of canceling it. A similar problem may occur when a project is interpreted as a failure because it delivers something other than what was originally specified or expected. There are development processes, e.g., agile methods, in which requirements are meant to evolve as part of the learning process and, clearly, it would be meaningless to define the learning process leading to change in requirements as indicating a failure. Finally, there may be differences in the failure perspectives of different project stakeholders, which also lead to different interpretations of whether a project has failed or not (Agarwal and Rathod 2006).

In spite of the problems with providing a commonly accepted definition of project cancellations and failures, there is little doubt that there are many situations where resources can be saved by reducing the number of projects that do not deliver anything, deliver a product much later than expected, or deliver a product that is not useful at all for the client. Not only is the direct waste of project resources likely to be substantial, but also the indirect waste such as lost business opportunities.

The importance of reducing the waste of resources on project failures motivates the high number of studies concerning the reasons for project failures, and methods to reduce failure rates. This includes several studies surveying what the stakeholders, such as the software developers, project managers, clients and users, perceive are the main failure and success factors of software projects. For lists of such factors see for example (Linberg 1999; Schmidt, Lyytinen et al. 2001; Charette 2005; Fabriek, Brand et al. 2008; Verner, Sampson et al. 2008; Al-Ahmad, Al-Fagih et al. 2009).

The study presented in this paper differs, to our knowledge, from previous studies on project failures on the following characteristics: i) It focuses on the effects of the potential failure factors known at the time of the project start-up, ii) It uses only observational data and not subjective assessments of project characteristics, and iii) It focuses on the projects of a global outsourcing marketplace. While there are other studies that use project data to predict failures based on project characteristics (Wohlin and Andrews 2001; Cerpa, Bardeen et al. 2010), these studies do not restrict the prediction of the risk of project failure to observable variables known at the project start-up. The use of the variables known at start-up makes the model more useful for practical settings. By the time it is possible to know that the project management is poor or the problem complexity is higher than expected, it may already be too late to take the proper action to avoid project problems and reduce the risk of failure. Prediction models using variables known at the project start-up may, amongst others, allow a client to get input related to an expected increase in the failure risk when emphasizing a lower price rather than higher skills when selecting a provider for the project. On the other hand, restricting a prediction model to variables observed at the project start-up means that the model will not be as accurate as models that include project failure variables known at much later stages in the development process.

The goal of our study is not only to predict the risk of project failure at the time of project start-up, but also to better understand when we have a context that is more likely to result in project failure. This improved understanding may be used to avoid situations with a high risk of failure, or if that is not possible, to give risk reducing actions a high priority in high-risk situations.

The remaining part of this paper is organized as follows: Section 2 describes the project database we used to build the model of project failure. Section 3 defines the model variables, describes the process of building the model, evaluates the accuracy of the prediction model and discusses the factors connected with higher or lower risks of project failure. Section 4 contains a discussion of the limitations of the analyses. Section 5 concludes.

2 The Project Database

The dataset we used to develop the prediction model consists of 785,325 small-scale software projects. The clients and providers of these projects have been using the services of vWorker.com (now merged with freelancer.com), which is a web-based global marketplace that connects software clients and providers. The providers are typically single software developers or smaller outsourcing companies located in low-cost countries, but include developers and companies from high-cost countries as well. There are also a few larger companies that use this marketplace, in addition to other channels, to find work for their employees. The services offered by the vWorker marketplace include:

- The means for clients to search for and invite project bids from providers with appropriate skills, e.g., Java, php, and SQL.

- Support for providers to place bids on a project.
- Arrangements that ensure that the provider is paid when the work is completed, and that the client does not have to pay if the work is of too low quality.
- Processes for managing disagreements/negotiations between the clients and the providers regarding payments or quality of work (arbitration processes).
- Skill tests of the providers.
- Evaluations of provider performances from previous projects through the presentation of aggregated and project specific information about client satisfaction and project cancellations.
- Evaluations of client performances from previous projects through the presentation of aggregated and project specific information about providers' satisfaction and project cancellations.

Typically, the providers and the clients never physically meet and conduct all of their communication through the functionality provided by vWorker.com, or other internet-based communication.

The characteristics of the data set of the vWorker projects that we used in this study include:

- Project data registered between May 2001 and October 2012.
- Number of projects: 785,325.
- Number of bids placed: 4,791,067.
- Mean number of bids per project: 6.1.
- Proportion of cancelled projects: 11.1%.
- Proportion of projects that were either cancelled or the provider received a client satisfaction rating of "poor" or worse: 14.0% (these are the projects we categorized as failed, see Section 3). Notice that a cancelled project either had no satisfaction score or a satisfaction score of "poor" or worse.
- Average provider pass rate on skill tests: 64%.
- Number of different provider nationalities: 187.
- The ten largest provider countries (sorted by decreasing number of projects): India, US, Romania, Pakistan, UK, Russia, Ukraine, Canada, Bangladesh, and the Philippines. Some of the providers located in high-cost countries seemed to use developers from low-cost countries, i.e., the provider country is not always as it seems from the vWorker presentation of a country.
- Number of different client nationalities: 177.
- The ten largest client countries (sorted by decreasing number of projects): US (with more than 50% of the projects), UK, Australia, Canada, India, Germany, the Netherlands, Israel, Sweden, and France.
- Proportion of projects where the client and provider are located in different countries: 90%.
- Proportion of projects where the client and provider have collaborated previously using vWorker: 43%.
- Price range of projects: 1–30,000 USD, with a mean of 146 USD, i.e., most projects are very small. Nearly all projects are based on a fixed-price contract between the client and the provider.

As can be seen from the above information, this data set includes a high number of projects, but the projects are, on average, very small. Where there is, we believe, value in understanding failure factors in smaller software projects, the small size of the projects clearly affects the generalizability of the results, e.g., to other outsoucing context. While there are arguments providing some support of a generalization of the findings from our data set to larger projects, which we will discuss in Section 4, such generalizations will rest on assumptions hard to fully validate as part of our study. In short, our data do mainly generalize to similar global outsourcing marketplaces.

3 Modelling Project Failure

We use project data from 2001 to 2008 to build a binary logistic regression model (Wonnacott and Wonnacott 1990) of project failure, and refer to this data set as the learning data set. The learning data set includes 437,278 projects, which is 56% of the total project data. The data from 2009 to 2012, which included 348,047 projects, is termed the evaluation data set, and is used to assess the prediction accuracy of the regression model. The evaluation data set is also used to assess the robustness of the model, i.e., to see how the model coefficients (or odds ratios) change from a model based on the learning data set (2001–2008) to one based on the evaluation data set (2009–2012). This chronological splitting of the data to build and evaluate the model reflects the fact that it is only realistic to use projects completed before the ones to be predicted to build a prediction model.

The remaining part of this section describes the variables used in the model (Section 3.1), the binary logistic regression model developed and its fit to the learning data set (Section 3.2), the analysis leading to the chosen cut-off point for predicting project failures (Section 3.3), the evaluation of the prediction accuracy of the model when applied to the evaluation data set (Section 3.4), the use of the model to explain project failure (Section 3.5), and a comparison of our results to the results of related studies (Section 3.6).

3.1 The Variables of the Model

We emphasized the inclusion of variables expected to have a direct or indirect causal connection to project failure. The emphasis on causally connected variables is motivated by our wish to use the model not only to predict, but also to understand essential relationships in software outsourcing. In addition, a model emphasizing causally relevant variables may be more robust with respect to over-fitting to the learning data set. The variable we try to predict through our model, i.e., the dependent variable, is the outcome of the project in terms of whether or not the project failed to deliver the expected functionality. This variable is defined in Section 3.1.1. The variables used to predict project failure, i.e., the independent variables, are related to the following four aspects of the projects:

- The skill of the provider, as indicated by the mean client satisfaction score from previously started projects, failure rate of previously started projects, and pass rate on skill tests at the time of the current project's start-up (for definitions of the provider skill variables see Section 3.1.2). We hypothesize that increased provider skill reduces the likelihood of project failure.
- The skill of the client, as indicated by provider satisfaction scores from previously started projects and failure rate of previously started projects (for definitions of the client skill variables see Section 3.1.3). We hypothesize that increased client skill reduces the likelihood of project failure. Including the skill variables of both the provider and client, we hoped, would enable us to compare the relative importance of the provider and client skills in predicting and explaining project failure.
- Collaboration aspects, as indicated by previous collaborations between the client and the provider, geographical distance between the client and the provider, and the client's focus on low price when selecting between providers for the current project (for the definitions of the collaboration characteristic variables see Section 3.1.4). We hypothesize that previous collaborations, a geographical closeness between the client and provider regions, and less focus on low price during the selection of a provider decreases the likelihood of project failure. In addition, we hypothesize that there are geographical regions of providers and clients with an increased likelihood of project failure.
- Project size, as indicated by the bid price of the selected provider for the current project (for a definition of the project size variable see Section 3.1.5). We hypothesize that the failure rate increases with the increased size of a project.

For most of the variables, we transfer a ratio or ordinal scale measurement into a nominal scale (category-based) variable. We do this to ease the analyses, to ease the interpretation of the odds ratios and to enable the inclusion of the category value "no value." The inclusion of the value "no value" is essential for several of the variables, e.g., to separate those clients and providers with no previous project experience (and consequently no failure rate) from those with high or low failure rates on previous projects. Notice that these categories are not categories of missing or unknown values, but rather the introduction of dummy variables telling whether a client or provider have received satisfaction scores or not, started any project or not, or completed skill tests or not.

A nominal scaled variable with broad categories may also be useful in a situation where smaller changes in the variable values do not necessarily reflect changes in the underlying phenomenon we want to model. This may be the case for several of the included variables, such as those related to the satisfaction levels of the providers and clients. We typically use the median value of a ratio or ordinal scale variable as the threshold value for the separation into a "low" or "high" category value due to skewed distributions and the goal to include an equal number of observations in each category. The threshold value for determining the "low" and "high" categories is based on the learning sample projects only, to preserve the chronologic separation of the learning and evaluation data set.

3.1.1 Project Failure

We categorized a project as failed when it was started, but not completed (a cancelled project), or had a client satisfaction score of 3 ("poor") or lower (more on this score in Section 3.1.2). Otherwise a project was categorized as not failed. The threshold client satisfaction score of 3 was selected because the marketplace itself (vWorker) instructs the clients to use this value to indicate a project failure. The majority of the projects (about 80%) that are categorized as failed are cancelled projects. Most cancelled projects have no client satisfaction score, but when one was available it was always 3 or lower.

TABLE 1: Project Failure Variable

Name	Values
ProjectFailure	1 = project is cancelled or has a client satisfaction score of 3 or lower 0 = project is completed and has a client satisfaction score of 4 or higher

The failure rate of the projects, applying the above definition, was 14% both for the learning and the learning data set. Our binary definition of project failure simplifies, in our opinion, the analysis and classification of the outcome of a project, as opposed to a continuous variable with values reflecting more or less failure. Our definition has, however, its limitations given the goal of reflecting common interpretations of what a project failure means. As discussed in Section 1, there may be projects where the cancellations have nothing to do with the project performance of the provider or the client. There may also be projects that would and should have been categorized as failures had there been better testing by the client at the time of submitting the provider satisfaction score.

The above challenges of defining project failures meaningfully, the non-deterministic (probabilistic) nature of project failure, and of having incomplete information available about the provider, client, and project at the time of start-up, suggest that we should not expect a model with accurate predictions of what we define as project failure.

3.1.2 Provider Skill

Our provider skill measures are based on the clients' satisfaction scores from previously started projects, the failure rates of previously started projects, and the pass rates on skill tests completed previously to the project. These measures, which relate to a provider's skill at the time of a project start-up, are defined in expressions (1) – (3).

(1) Mean satisfaction score provider = ΣClient satisfaction score on projects previous to the current projecs/Total number of previous projects with satisfaction scores, where satisfaction scores range from 1 ("horrible") to 10 ("excellent").

The mean satisfaction scores assess the providers' performances in completing the projects, as perceived by the clients. An examination of the client's comments with the scores indicates that even the values of 8 ("good") and 9 ("very good") are sometimes used when the client is quite dissatisfied, i.e., the textual and numerical data seems to give different viewpoints and suggest a lower than desired reliability of the data. The satisfactin scores are highly skewed towards the higher values and motivate the use of median-based categories.

(2) Failure rate provider = Number of failed projects previous to the current project/Total number of previous projects, where project failure is defined (as described in Section 3.1.1) as a project that is either cancelled or has a client satisfaction score of 3 ("poor") or less.

(3) Skill test pass rate provider = Number of passed skill tests previous to the current project/Total number of skill tests taken previously to the current project, where the threshold for pass/no pass of a test (exam) is a proportion, set by vWorker, of the questions answered correctly.

As many as 28% fail to pass the test they participate in, which suggests that it is not easy to pass a vWorker test. There may be several problems with the quality and meaningfulness of the skill tests variable in the context of vWorker. For example, the outcome could be misleading if developers other than the one completing the test are those participating in your project. This may be the case when a company, rather than a single developer, is selected for a project. The problems with the skill tests suggest that we cannot expect very good results when using them, nor can we use the current data set to say much in general about the use of skill tests being meaningful in indicating the provider's skills in completing a project.

The category-based provider skill variables derived from expressions (1) – (3) are defined in Table 2.

TABLE 2: Provider Skill Variables

Name	Values
SatisfactionScoreProviderCat	Low = Provider has a mean client satisfaction score on previously started projects lower than the median of the mean client satisfaction scores (9.86) of all project providers. High = Provider has a mean client satisfaction score on previously started projects higher than or equal to the median of the mean client satisfaction scores (9.86) of all project providers. No Scores = Provider has no previous client satisfaction scores. About 9% of the projects start with a provider with no previous client satisfaction score.
FailureRateProviderCat	Low = Provider has a lower than median (5%) failure rate on previously started projects. High = Provider has an equal to or lower than median (5%) failure rate on previously started projects. No Projects = Provider has no previously started projects. About 8% of the projects are started with a provider with no previous project start-ups.
SkillTestPassRateProviderCat	Low = Provider has a pass rate on skill tests lower than the mean provider pass rate (72%) of all projects. High = Provider has a pass rate on skill tests higher than or equal to the median provider pass rate (72%) of all projects. No Tests = Provider has no completed skill tests. About 61% of the projects are started with a provider with no previous skill tests.

The following is an example illustrating the use of the variables. Assume that a project is started with Provider A. Provider A completed 30 projects before starting the current project, and failed in three of them. Two of these failures are related to cancelled projects and one to a project where the client gave a satisfaction score of 3 (“poor”). This gives a failure rate of 10% (3/30), which is higher than the median failure rate of 5%, and consequently gives a FailureRateProviderCat with value “High”. Provider A received client satisfaction scores for 20 of the 30 completed projects. Fifteen of the scores were 10 (max scores), three were 9, one was 8, and one was 3 (failed project). This gives Provider A a mean client satisfaction score of 9.40 [(15*10+3*9+8+3)/20], which is lower than the median of 9.86, and the SatisfactionScoreProviderCat value “Low”. Provider A has completed no tests, giving the SkillTestPassRateProviderCat value “No Tests”.

As can be seen in Table 2, the median skill values used as threshold values are better than those of the project population as a whole. The project failure rate of the learning data set is for example 14%, while our threshold value for “Low” and “High” failure rates is 5%. The reason for this is that the providers selected for many projects, who will be among those with the best skill scores, will have stronger impact on the median skill values (our threshold values) than the providers selected for fewer projects. Assume, for example, three projects where the selected providers (P1, P2, and P3) all have a failure rate from previous projects of 0%. The projects using P1 and P2 as providers are successful, while P3’s project fails. This will result in the failure rate of the three projects being 33%. If P3’s increased failure rate stops him from being selected for new projects (or causes him to be selected less frequently) the last project failure will not impact (or impact less) the failure rates of the selected providers. In short, providers with low failure rates and good client scores are selected much more often for future projects than those with higher failure rates and poorer client scores. This mechanism is frequently observed in project contexts and sometimes termed the Matthew or “the rich get richer” effect (Lin, Viswanathan et al. 2010).

3.1.3 Client Skill

Several studies suggest that not only a skilled provider, but also a skilled client is essential in avoiding project failures (Iacovou and Nakatsu 2008). Therefore, we included two measures related to the client’s skills. These measures are defined the same way as for the providers, see (equations (4) and (5) below.

(4) Mean satisfaction score client = ΣProvider satisfaction scores on projects previous to the current pro-

ject/Total number of previous projects with satisfaction scores, where the satisfaction scores, as before, range from 1 ("horrible") to 10 ("excellent").

The satisfaction scores assess the clients' performances for the projects, as perceived by the providers. An examination of the comments with the scores, which are also made available by the vWorker marketplace, indicates that the providers are hesitant to give negative comments and scores about the clients. Most providers give their clients the maximum satisfaction score of 10.

(5) Failure rate client = Number of failed projects previous to the current project/Total number of previous projects, where project failure, as before, is defined as a project that is either cancelled or has a client satisfaction score of 3 ("poor") or less.

The category-based client skill variables derived from expressions (4) and (5) are defined in Table 3.

TABLE 3: Client Skill Variables

Name	Values
SatisfactionScoreClientCat	Low = Client has a mean provider satisfaction score on previously started projects lower than the median of the mean provider satisfaction scores (10.0) of all clients. High = Client has a mean provider satisfaction score on previously started projects higher than or equal to the median of the mean provider satisfaction scores (10.0) of all clients. No Scores = Client has no previous provider satisfaction score. About 10% of the projects start with a client with no previous provider satisfaction score.
FailureRateClientCat	Low = Client has a lower than median (9%) failure rate on previously started projects. High = Client has an equal to or higher than median (9%) failure rate on previously started projects. No Projects = Provider has no previously started projects. About 10% of the projects are started with a client with no previous project start-ups.

The following is an example illustrating the use of the variables. Assume that a project is started with Client B as the client. Client B started five projects previously. None of the projects failed and all of the providers of the projects gave Client B maximum satisfaction scores (10). Consequently, the client failure rate is 0% at the time of the project start-up, which gives the FailureRateClientCat value "Low." The client's mean satisfaction score is 10.0, which gives the SatisfactionScoreClientCat value "High."

3.1.3 Collaboration Elements

The following measures, which we believe may be useful in indicating the potential risks and strengths of the collaboration between the client and the provider, are included in our analyses:

i) The client's collaboration with the provider on earlier occasions (PreviousCollaboration). Previous collaboration may be considered to be a test evaluating the skill of the provider, and that only providers delivering satisfactory quality are considered for repeated projects with the client. The presence of a previous collaboration with the selected provider is therefore a strong potential indicator of the reduced risk of project failure. The values of this measure are either "yes," when the client and the provider have collaborated previously within the vWorker marketplace, or "no".

ii) The client's focus on low price when selecting a provider (FocusLowPrice). It is known from studies on "the winner's curse" (Kern, Willcocks et al. 2002; Anandalingam and Lucas 2005; Jørgensen, Grimstad et al. 2005; Jørgensen 2013) that many bidders, and a focus on low price when selecting a provider, are likely to lead to the selection of an over-optimistic provider. It is also believed that over-optimistic estimates and plans increases the risk of project problems related to reduction in the quality of less fully specified elements, such as maintainability and user friendliness (Jørgensen and Sjøberg 2001). This means that it is possible that the risk of project failure is connected with the selection of a bid with a relatively low price. We define the focus on low price in the selection of a provider (FocusLowPrice) as follows:

(6) FocusLowPrice = (mean price of all bids – price of selected bid)/max(mean price of all bids, price of selected bid)

As can be seen from equation (6), FocusLowPrice measures the distance between the mean price of the bids and the price of the selected bid, and normalizes this to a value between -1 (minimum price focus) and 1 (maximum price focus) by dividing it by the maximum of the mean price and the price of the selected bid. In a situation where the selected bid is equal to the price of the mean bid (which is also the case when there is only one bid) the measure gives a value of 0.

iii) The failure rate of the provider region (FailureRateProviderRegion). Some geographical provider regions may have a higher failure rate than others. To test this we allocated each provider country a geographical region. A list of provider regions exemplified with the three countries with the most projects in that region, the region's percentage of projects in the role of provider, and the failure rate of the projects in the learning data set within each region is included in Table 4.

iv) The failure rate of the client region (FailureRateClientRegion). Some geographical client regions may have a higher failure rate than others. To test this we allocated each client country the same geographical regions as those of the providers. A list of client regions exemplified with the three countries with the most projects, the region's percentage of projects in the role of client, and the failure rate of the projects in the learning data set within each region is included in Table 4.

v) Geographical distance between client and provider (GeographicalDistance). It is sometimes observed that near-shore development, i.e., the selection of providers in the same or a nearby geographical region, has advantages when compared to outsourced software development (Westner and Strahringer 2010). This could for example be the consequence of a lesser difference in time zones, or lower cultural differences leading to better communication. We, therefore, included a variable indicating the geographical distance between the client and the provider. This variable has the value "same" if the client and the provider are within the same geographical region, "neighbor" if they are within neighboring regions, and "offshore" if they are in neither the "same" nor "neighbor" region. The neighboring regions are given in Table 4. Notice that purely geographically speaking, there are more neighboring regions than those listed in Table 4, and that we have only included those we perceived as geographically *and* culturally close enough to be meaningfully categorized as nearshore development. Other categorizations and opinions on nearshore are clearly possible.

TABLE 4: Geographical Regions with Dominant Countries

Region	Three largest provider countries (decreasing values)	Percentage of projects	Failure rate as provider	Three largest client countries (decreasing values)	Percentage of projects	Failure rate as client	Neighbor region
Africa	South Africa, Tunisia, Nigeria	0.55%	22%	South Africa, Nigeria, Botswana	0.48%	18%	Middle East
East Asia	Philippines, China, Indonesia	5.12%	15%	Singapore, Malaysia, Hong Kong	2.28%	21%	South Asia
Eastern Europe	Romania, Russia, Ukraine	28.80%	11%	Romania, Poland, Russia	1.84%	17%	Western Europe
Latin America	Argentina, Brazil, Mexico	3.95%	12%	Brazil, Mexico, Puerto Rico	1.15%	17%	North America

Middle East	Egypt, Turkey, Israel	3.95%	15%	Israel, Turkey, Saudi Arabia	2.10%	20%	Africa
North America	United States, Canada (only two countries)	19.68%	15%	United States, Canada (only two countries)	62.2%	13%	Latin America
Oceania	Australia, New Zealand, Fiji	1.75%	12%	Australia, New Zealand, Fiji	5.99%	15%	None
South Asia	India, Pakistan, Bangladesh	27.04%	18%	India, Pakistan, Sri Lanka	2.05%	23%	East Asia
Western Europe	United Kingdom, Germany, Spain	9.13%	14%	United Kingdom, Germany, Netherlands	21.88%	15%	Eastern Europe

The category-based collaboration element variables used in the model are defined in Table 5.

TABLE 5: Collaboration Element Variables

Name	Values
PrevousCollaboration	“Yes” = at least one previous collaboration between the client and the provider. “No” = no previous collaboration.
FocusLowPriceCat	“Low” = Selection of a provider with a price that is the mean price of the bidding providers (FocusLowPrice <= 0) or higher. “Medium” = Selection of a provider with a price that is lower, but not more than 20% lower (FocusLowPrice between 0 and 0.2) than the mean price. “High” = Selection of a provider with a price that is at least 20% lower (FocusLowPrice >= 0.2) than the mean price. The FocusLowPrice-threshold of 0.2 divides the set of projects where the provider selects a project with a price lower than the mean price (FocusLowPrice < 0) into two equally sized project data sets.
FailureRateProviderRegionCat	“High” = provider region with relatively high failure rate. “Low” = other provider regions. We decided to include only those regions with an “unusually” high failure rate (as we perceived it from looking at Table 4) in the category “High.” We identified two provider regions with somewhat higher failure rates than the others: the regions Africa and South Asia. Note that the purpose of the chosen categorization is to see how information about the region may contribute to the accuracy and fit of the model, not to make claims about the skill differences of the providers from different regions. To do this, we would need more information about the representativeness of the providers included in our data set.
FailureRateClientRegionCat	“High” = client regions with relatively high failure rate. “Low” = other client regions. In a similar way to the categorization of the regions with a higher failure rate for the provider, we categorized the failure rates for the client depending on the region. After examining Table 4, we decided to categorize the regions of East Asia, the Middle East, and South Asia as regions with rela-

	tively high client failure rates. As before, this categorization is not meant to support claims about the quality of the clients from different regions.
GeographicalDistance	“Same” = same region for client and provider. “Neighbor” = provider is located in a neighboring region of the client. “Offshore” = all other client-provider region combinations.

3.1.4 Project Size

Larger projects may, on average, be more complex and more problematic than smaller ones. For a discussion of this, see (Jørgensen, Halkjelsvik et al. 2012). We included the price of the selected bid in the model as an indicator of the project size. The advantage of a size measurement based on the bid price is that it is known at the time of the project start-up and likely to be strongly correlated with the actual size of the project. The price may, however, not always be an accurate proxy of reasonable interpretations of the project size. A lower price may for example be a result of bids from a lower cost country or “strategic bidding,” e.g., to propose a very low bid price in order to get reference clients, thereby making it easier to get future clients. In spite of these limitations, we believe that the bid, in most cases, gives a sufficiently accurate indication of the size of the project for our purpose of modeling the risk of project failure. We use the logarithm of the bid as our model variable to avoid the strong influence of a few high bids.

TABLE 6: Project Size Variable

Name	Values
logProjectSize	log(Price), where Price is the amount in US dollars required by the selected provider to complete the project. One unit increase in logProjectSize (e.g., from log(Price) 2 to 3) corresponds to a ten times increase in project price.

3.2 The Binary Logistic Regression Model

We built a model of the risk of project failure based on a binary logistic regression, where the regression coefficients were estimated using a maximum likelihood estimation. The model output is a value between 0 and 1, and may be interpreted as the predicted probability of project failure, or at least as an indicator value for the risk of project failure. The odds ratio of a variable, which is our primary measure of the effect size of a variable, is calculated as the ratio of the odds of a project being a failure to the odds of a project being a non-failure given one unit increase of the variable, assuming all other variables are held constant. Categorical variables are represented by dummy variables with the value 1 when the value is present, otherwise with 0. For example, the categorical variable FailureRateProviderCat, which has the possible values of “High,” “Low,” and “No Projects,” is represented by the dummy variables “FailureRateProviderCat=Low” (which has the value 1 when the FailureRateProviderCat equals “Low,” otherwise 0) and “FailureRateProviderCat=No Projects” (which has the value 1 when the FailureRateProviderCat equals “No Projects,” otherwise 0). The third category, present when FailureRateProviderCat equals “High,” is implicitly covered since this value is present when both other dummy variables equal 0. The odds ratio of “FailureRateProvideCat=Low” then gives the increase or decrease in failure risk when, instead of selecting a provider with a “High” failure rate or “No Projects”, the client selects a provider with a “Low” failure rate. For example, an odds ratio of 0.6 for the variable “FailureRateProvider=Low” means that a provider with a low failure rate from previous projects reduces the likelihood of project failure to 60% of that for providers with “High” or “No Projects” failure rates, given the same values of all other variables included in the model. An odds ratio close to one for a variable indicates that different values of that variable do not increase or decrease the probability of project failure significantly. The more that an odds ratio deviates from 1, the stronger the importance of that variable in predicting the risk of project failure.

It may, as we will analyze in more detail later in this paper, not be straightforward to interpret the odds ratio in terms of probabilities. Such interpretation relies on a number of assumptions, for instance that the model is correctly specified and that the logit function, which is used by our regression model approach, is a good approximation of the probability function.

The resulting model from the binary logistic regression, using the learning dataset, is displayed in Table 7.

TABLE 7: Results from the Logistic Regression Model

Predictor variable	Coeffi- cient	p-value	Odds ratio	95% confidence interval	
				Lower	Upper
Constant	-2.90	0.00			
SatisfactionScoreProviderCat=Low	0.35	0.00	1.42	1.39	1.45
SatisfactionScoreProviderCat=No Scores	0.91	0.00	2.49	2.33	2.67
FailureRateProviderCat=Low	-0.66	0.00	0.52	0.51	0.53
FailRateProviderCat=No Projects	-0.34	0.00	0.71	0.67	0.76
SkillTestPassRateProviderCat=Low	0.07	0.00	1.07	1.02	1.12
SkillTestPassRateProviderCat=No Tests	0.58	0.00	1.79	1.74	1.85
SatisfactionScoreClientCat=Low	0.18	0.00	1.20	1.17	1.23
SatisfactionScoreClientCat=No Scores	0.25	0.00	1.28	1.23	1.33
FailureRateClientCat=Low	-0.64	0.00	0.53	0.52	0.54
FailureRateClientCat=No Projects	-0.63	0.00	0.53	0.51	0.56
PreviousCollaboration=Yes	-1.74	0.00	0.17	0.17	0.18
FocusLowPriceCat=Low	-0.19	0.00	0.83	0.81	0.85
FocusLowPriceCat=Medium	-0.08	0.00	0.92	0.89	0.95
FailureRateProviderRegionCat=High	0.27	0.00	1.31	1.28	1.33
FailureRateClientRegionCat=High	0.42	0.00	1.53	1.48	1.58
GeographicalDistance=Neighbor	-0.07	0.02	0.93	0.90	0.97
GeographicalDistance=Offshore	0.02	0.10	1.02	1.00	1.05
logProjectSize	0.71	0.00	2.03	1.99	2.06

The model fit is indicated by a Somer's D of 0.58 and a Goodman-Kruskal Gamma of 0.59. There are 79% concordant pairs, i.e., there were 79% of all possible pairs of projects where one project failed and the other did not fail and the model gave the failed project the higher probability of failure. The level of model fit is, we believe, promising for the building of a prediction model and to identify failure factors.

Given the probabilistic nature of the output of the model, i.e., that a higher model output value predicts a *higher risk* of project failure and *not* deterministically that the project will be a failure, a meaningful analysis of model fit may in our context be to compare the model risk predictions and the actual failure frequencies. We do this for selected intervals of predicted failure risks in Table 8.

TABLE 8: Risk Predictions vs Actual Failure Frequency (Learning Data Set)

Predicted risk (probability interval)	Number of projects	Actual failure frequency
0.0-0.1	181630	0.03
0.1-0.2	88018	0.12
0.2-0.3	63126	0.20
0.3-0.4	46980	0.28
0.4-0.5	29978	0.36
0.5-0.6	14071	0.44
0.6-0.7	4732	0.50
0.7-0.8	900	0.52
0.8-0.9	57	0.67
0.9-1.0	0	No data

As can be seen in Table 8, an increase in the predicted risk is closely connected with an increase in failure frequency. It can also be seen that the model, when interpreting the risk values as probabilities, slightly overestimates the failure frequency for the higher risk values. For example, when the model gives values in the range of 0.4 to 0.5, the corresponding project failure frequency is 0.36, which is slightly too low to be included in the interval. There are several possible reasons for this overestimation of actual frequencies. It may be that the logit probability function does not model the actual project failure probability function well, or that there are challenges related to the relatively low proportion of failures compared to non-failures (King and Zeng 2001). While the model's overestimation of probabilities for some probability intervals is not critical for our evaluation of the model's prediction accuracy, it implies that we should be careful when interpreting the model's output values and the odds ratios in terms of probabilities. The model coefficients and odds ratios are strongly connected to the probability of project failure, but cannot be expected to reflect the exact increase or decrease in failure probability.

3.3 The Cutoff Value for Predicting Project Failure

To find an appropriate cutoff-value for the output of our prediction model, i.e., a threshold value for when to predict a project failure or non-failure, we calculated the ROC (Receiver Operating Characteristics) curve for the learning data set as follows:

1. The coefficients shown in Table 7 were used to calculate, for each project, the model-predicted project failure value (ProjectFailure).
2. For a set of selected cutoff-values covering the interval
3. between 0.0 and 1.0, see Figure 1, we predicted that a project would fail if the model-based project failure value was higher than the selected cutoff-value, otherwise it would not fail. If, for example, the model for a particular project gave a ProjectFailure-value of 0.4 and the cutoff-value was 0.3, we would predict a project failure.
4. The predicted failure was compared with the actual outcome, failure or non-failure, for each project. A prediction was said to be a "True Positive" (TP) if the model predicted a failure and the project actually failed, a "False Negative" (FN) if the model predicted a non-failure and the project actually failed, a "False Positive" (FP) if the model predicted a failure and the project actually did not fail, and a "True Negative" (TN) if the model predicted a non-failure and the project actually did not fail.
5. For each cutoff-value the "Sensitivity" (the proportion of failures correctly identified as failures) (=TP/(TP+FN)), and the "Specificity" (the proportion of non-failures correctly identified as non-failures) (=TN/(TN+FP)) were calculated.
6. For each cutoff-value we displayed the "Sensitivity" and "1-Specificity", see Figure 1. The better the model, the higher the Sensitivity (correct failure predictions) and the lower the "1-Specificity" (incorrect non-failure predictions).

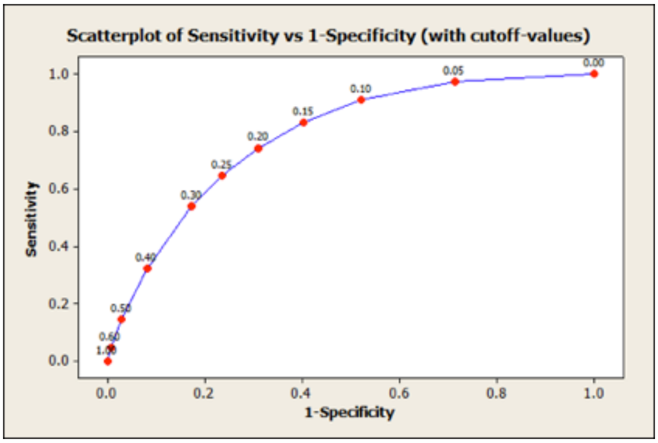

Figure 1. The ROC Curve of the Model for Different Cutoff-Values

As can be seen in the ROC-curve in Figure 1, there is a trade-off between being good at detecting project failures (high Sensitivity) and being good at avoiding predicting failures when a project is not failing (low 1-Specificity). The area under the ROC curve, which is typically termed the AOC, is a value between 0 and 1, where a value of 1 indicates a perfect fit and a value of 0.5 indicates that the model is no better than random predictions. The AOC of our model is about 0.8, which suggests a good fit of the model. For more on the properties of the AOC and its relationships to other measures of fit, see (Fawcett 2006).

Our main use of the ROC-curve is to support the selection of a cutoff value by determining which output value of the model should be used to predict whether a project will be a failure or not. The default cutoff value chosen by many statistical tools is 0.5, but as can be seen from Figure 1 this would not be a good choice in our case. The model's ability to predict non-failures would be very good for a cutoff value of 0.5 (97% of the non-failures would be predicted correctly), but the ability to predict project failures would be poor since only 15% of the failures would be predicted correctly. In other words, even though the total accuracy, which would be 85% when measured as the ratio of correct predictions to the total number of predictions, would be high for the default cutoff value of 0.5, it would not be the cutoff-value we would choose to predict project failures.

Exactly which cutoff-value to choose is a matter of cost function: How much is the cost of failing to predict a project failure compared to the cost of failing to predict a project non-failure? For the purpose of evaluating the prediction accuracy of our model we assumed that the cost of incorrect predictions is about the same for failures and non-failures, which led us to choose the cutoff-value of 0.2. A cutoff-value of 0.2 implies, for the learning data set, that we would correctly identify 74% of the failures (Sensitivity) and 69% of the non-failures (Specificity). The accuracy of the predictions using the cutoff-value of 0.2 is about 0.7 (about 70% of the predictions will be correct). Note that this is a substantially lower accuracy than the 85% correct predictions for the default cutoff-value of 0.5, in spite of being a more meaningful prediction model.

3.4 The Prediction Accuracy of the Model

To evaluate the prediction accuracy of the model we applied the model developed with the learning data set (projects from 2001–2008) to the evaluation data (projects from 2009–2012). The evaluation data set contains 347,667 projects and we used the cutoff-value of 0.2 chosen in Section 3.3. This gives the following accuracy prediction results:

- True Positives (TP): 34,987 (failed projects predicted to be failures)
- False Negatives (FN): 11,777 (failed projects incorrectly predicted to be non-failures)
- False Positives (TP): 97,935 (non-failed projects incorrectly predicted to be failures)
- True Negatives (TN): 202,968 (non-failed projects correctly predicted to be non-failures)
- Sensitivity (TP/(TP+FN): 0.75 (proportion of correctly predicted failed projects)
- Specificity (TN/(FP+TN): 0.67 (proportion of correctly predicted non-failed projects)
- Accuracy ((TP+TN)/(TP+FN+FP+TN)): 0.68 (proportion of correct predictions)

Comparing the accuracy values for the evaluation data set with the corresponding values for the learning set we see that the sensitivity (0.74 vs. 0.75), specificity (0.69 vs. 0.67), and accuracy (0.70 vs. 0.68) are very similar. Table 9 compares the model probability with the actual frequency of failure, similarly to what we did in Table 8 for the learning data set.

TABLE 9: Risk Prediction vs. Actual Failure Frequency (Evaluation Data Set)

Predicted risk (probability interval)	Number of projects	Actual failure frequency
0-0.1	144480	0.03
0.1-0.2	70265	0.11
0.2-0.3	49347	0.19
0.3-0.4	38640	0.25
0.4-0.5	26225	0.32
0.5-0.6	13294	0.39
0.6-0.7	4481	0.46

0.7-0.8	886	0.53
0.8-0.9	49	0.63
0.9-1.0	No data	No data

Comparing Tables 8 and 9, we see that the correspondence between predicted risk and actual failure frequency of the evaluation data set is almost as good as with the learning data set. In addition, the tendency towards overestimating the higher failure frequencies remains.

As can be seen in Table 9, a higher model output value is strongly connected to an increase in actual failure frequency. For example, while it is very unlikely (3% likely) to observe a project failure when the model gives a risk value in the interval of 0.0–0.1, it is about ten times more likely (32% vs. 3%) to observe a project failure when the model gives a value in the interval 0.4–0.5.

3.5 The Failure Factors

A good understanding of how different factors contribute to the higher risk of project failure may be useful for risk reduction or other purposes. To support this type of understanding we included all project data into one data set, which resulted in the coefficient and odds ratios displayed in Table 10. As can be seen, the odds ratios are very similar to those for the learning data set only as displayed in Table 7.

TABLE 10: Results from the Logistic Regression Model (All Data)

Predictor variable	Coefficient	p-value	Odds ratio	95% confidence interval	
				Lower	Upper
Constant	-2.68	0.00			
SatisfactionScoreProviderCat=Low	0.31	0.00	1.36	1.34	1.39
SatisfactionScoreProviderCat=No Scores	0.82	0.00	2.26	2.15	2.38
FailureRateProviderCat=Low	-0.68	0.00	0.51	0.50	0.52
FailureRateProviderCat=No Projects	-0.28	0.00	0.76	0.72	0.80
SkillTestPassRateProviderCat=Low	0.15	0.00	1.17	1.14	1.20
SkillTestPassRateProviderCat=No Tests	0.51	0.00	1.67	1.64	1.70
SatisfactionScoreClientCat=Low	0.18	0.00	1.20	1.18	1.22
SatisfactionScoreClientCat=No Scores	0.25	0.00	1.28	1.24	1.32
FailureRateClientCat=Low	-0.67	0.00	0.51	0.50	0.52
FailureRateClientCat=No Projects	-0.64	0.00	0.53	0.51	0.55
PreviousCollaboration=Yes	-1.78	0.00	0.17	0.17	0.17
FocusLowPriceCat=Low	-0.16	0.00	0.85	0.84	0.87
FocusLowPriceCat=Medium	-0.07	0.00	0.93	0.91	0.95
FailureRateProviderRegionCat=High	0.25	0.00	1.28	1.26	1.30
FailureRateClientRegionCat=High	0.44	0.00	1.55	1.51	1.58
GeographicalDistance=Neighbor	-0.08	0.00	0.92	0.90	0.95
GeographicalDistance=Offshore	-0.01	0.18	0.99	0.97	1.01
logProjectSize	0.64	0.00	1.91	1.88	1.93

Table 10 suggests, among other things, the following relationships:

- Selecting a provider with a relatively low client satisfaction score from previous projects increases the probability of project failure by 36%, when compared to selecting a provider with no or a high previous provider satisfaction score. Even more striking is the strong increase (126%) in the risk of project failure when selecting among the providers with no previous client satisfaction score. Two potential explanations for the latter result are: i) Those without previous client satisfaction scores have little or no previous project experience, and as a consequence, lower skill and ii) Those without previous client satisfaction scores have previous project experience, but the client has chosen not to evaluate them, rather than give them a low score. A further analysis of those without any previous client satisfaction scores shows that most of them (86%) have no previous project experience, but it also shows that providers *with* previous project experience and no previous client evaluations have a much higher than average failure rate (27%). This suggests that not only is there an effect of less experience on project failure, but also that clients are less likely to give the provider a score when dissatisfied. Instead of giving a very low score, the client may avoid giving a satisfaction score altogether.

- Selecting a provider with a relatively low failure rate on previous projects reduces the probability of project failure to almost half (51%). Selecting a provider with no previous failure rate information also reduces the likelihood of project failure (76%). Consequently, it is essential to avoid providers with high failure rates from previous projects to reduce the project failure risk, and it is better to select a provider with no rather than one with high failure rate.
- It is a 17% higher risk of project failure when selecting a provider with the lowest test pass rates. Interestingly, the providers who had not participated in the tests (which are the majority of the providers) have a 76% higher probability of project failure. In other words, the voluntary skill tests seem to be avoided by the less skilled providers, and it is the lack of tests, rather than the test scores themselves, that seems to be the most useful indicator of project failures in this context. See also our discussion in Section 3.1.2 on the limitations on skill tests in the studied context.
- A provider's satisfaction with the client on previous projects is a good predictor of project failure. A client among those with lowest provider satisfaction scores has a 20% higher probability of project failure. A client with no previous provider satisfaction scores has 28% higher probability of failure.
- Clients with low failure rates from previous projects have a much lower (51%) risk of project failure. This value suggests that the failure rate of the client on previous projects is almost as good a predictor of project failure as the failure rate of the provider. This may, with some care, be interpreted to suggest that the skill of the client is almost as important as the skill of the provider to avoid project failure. Note that the odds ratios for this, and the other variables, assume constant values in the other variables, i.e., this is the effect of the client's previous failure rate on the likelihood of project failure for the same values of provider skill, collaboration elements, and project size.
- If the client and the provider have collaborated previously, the risk of project failure is only 17% of otherwise. Previous collaboration seems, therefore, to be the factor with strongest risk reducing effect in this data set. Previous collaboration may be considered as a highly relevant test of the provider's ability to deliver on time with quality, and communicate well with the client. Only providers scoring well on a previous collaboration are likely to be re-selected as providers. In short, this result suggests that there is nothing better than the use of highly realistic tests, such as real projects, to evaluate the skill when selecting a provider. In addition, there are elements of increased trust that could explain the positive effect of previous collaboration.
- Clients who chose a provider with a price equal to or higher than the mean bid reduced the risk to 85%, compared to those with a stronger focus on low price, i.e., those selecting a provider with a price lower than the mean bid price. As before, this risk reduction assumes the same characteristics of collaboration and same level of provider skill, i.e., it measures the negative effect of the focus of low price alone. As suggested in (Jørgensen 2013), this could be related to the negative effect of "the winner's curse" or that a too low price makes a, perhaps otherwise skilled, provider perform poorly.
- If a provider is from one of the geographical regions with previously high project failure rates, the risk of project failure increases by 28%. The effect is even stronger for the client regions connected with high failure rates, where we observe an increase in risk of 55%. Further analysis shows that the project failure rates are especially high, between 43% and 86% higher than the average failure rate, when combining a client from a high failure rate client region with a provider from a high failure rate provider region.
- The use of offshore providers has been a much discussed reason for project failure (Leidner and Kayworth 2006) . It was therefore surprising that we found no significant increase (p=0.18) in project failures between offshore (providers outside the same or neighboring region) and other projects. This should not be used to conclude that such differences are not essential in other outsourcing contexts, just that the data does not support any such connection in our context.
- The size of the project, as indicated by the bid price, has a large effect on the risk of project failure. The analysis suggests that the risk of project failure almost doubles (91% higher) when the size increases by a factor of ten. We should be careful when using this result in generalizing project sizes outside the sizes of the studied projects. It may be the case that what is considered to be a large and complex project is very much dependent on the size of the previously completed projects.

We found that all the included variables, except the one related to "offshore," had a significant effect on project failure. In particular, we found it interesting to observe how much the characteristics of the client and the previous collaborations between the client and the provider affected the risk of failure. This suggests that the responsibility of the client in explaining project failure is strong and that previous collaboration is the best skill test and indicator of a low risk of failure in future projects. We also found that a client can reduce the risk of project failure substantially through less focus on low price and more focus on provider skill when selecting a provider.

3.6 Related Work

In several cases our results are, in spite of the differences in typical project size and variety of outsourcing context, consistent with those of previous studies related to project failures:

- The cancellation rate of 11.1% is about the same as in other peer-reviewed studies on this topic (see references in the introduction).
- The skill of the provider and the quality of the previous collaboration between the client and the provider are essential for explaining project failure (McManus and Wood-Harper 2007; Nakatsu and Iacovou 2009) .
- The role of the client is essential to explain project performance (Maglyas, Nikula et al. 2010). The role of the client in leading to cancelled software projects is exemplified in (Ahonen and Savolainen 2010).
- The risk of project failure increases with increased project size (Flyvbjerg, Skamris Holm et al. 2004).

Our results extend and to some extent contradict previous relevant work with respect to:

- The finding of a strong increase in the risk of project failure in situations where the client selected a provider with a lower than average bid price. This is a topic not much studied in software project contexts, but see our studies related to this in (Jørgensen, Grimstad et al. 2005; Jørgensen 2013).
- The finding that offshore projects did not have a significantly higher failure rate than projects conducted in the same or neighboring geographical region. This seems to be in opposition to what is reported in (Westner and Strahringer 2010) and the perception of many software managers (Nakatsu and Iacovou 2009). We do not currently know much about the reason for this difference in findings. It is possible that the providers and clients involved in the studied type of outsourcing (those using the golobal outsourcing marketplace for small projects) are different from the type of providers and clients included in other studies. We followed-up with analyses of whether there were some client regions where offshore collaborations were especially risky, i.e., whether our general analysis hid some interesting results present on a more detailed level. The analyses gave that there may be regions where the failure rate increased when collaborating offshore, but that clients in the largest client regions, (North America, Western Europe, Oceania, and South Asia) had on average no increased project risk in such cases. Interestingly, the highest failure rates were found when both the client and the provider were from South Asia, mainly India, which suggests that the difference in culture may not be the main explanation for the higher risk of project failure in that region. More studies may be needed to get more robust and context dependent results on the effect of offshore collaborations on project failure risk.

4 Limitations and Discussions

Most of the projects included in our analysis are small. While there is value in knowing the success factor of smaller-sized projects, which to some extent can be said to be the building blocks of larger projects, the results would be of greater value if they were likely to generalize to larger projects. As an input to the generalization discussion we examined the extent to which the odds ratios would change for different project size sub-sets, including "very small" (1-100 USD), "small" (100-1000 USD), and "medium" (1000 and up to 30,000 USD) sized projects. The median price of the projects in the category of medium sized projects is about 2000 USD. The same variables and the same binary logistic regression were used for each of the sub-sets of project data. The main finding was that the model fit (Somer's D) and the odds ratios for the different sub-sets of data were similar for most ofthe model variables, see Table 11. Interesting observations include that performing poorly on the skill tests (SkillTestPassRateProviderCat=Low) is a better predictor of increased risk of failure for the smallest projects and that less focus on low price when selecting a provider (FocusLowPriceCat=Low or Medium) is a better predictor of decreased of failure for the larger projects. In general, however, that odds ratio give support to the belief that the importance of the included variables to explain risk of project failure is likely to remain reasonably stable with increasing project size, at least in the context of the global outsourcing marketplace studied. In spite of this result, we can hardly use this analysis of project sub-sets to make strong claims about the relevance for large projects, especially outside an outsourcing context similar to the one that we have studied. The size-independence within our data set may, nevertheless, suggest that several of the results may generalize to projects somewhat larger than those included in the current analysis.

TABLE 11: Odds Ratios for Project Size Categories (All data, number in italics when p>0.1)

Predictor variable	0-100 USD (D = 0.57)	100-1000 USD (D = 0.55)	1000-3000 USD (D = 0.56)
SatisfactionScoreProviderCat=Low	1.32	1.42	1.40
SatisfactionScoreProviderCat=No Scores	2.19	2.40	1.86
FailureRateProviderCat=Low	0.51	0.50	0.53
FailureRateProviderCat=No Projects	0.73	0.72	0.73
SkillTestPassRateProviderCat=Low	1.22	1.09	*1.04*
SkillTestPassRateProviderCat=No Tests	1.68	1.67	1.39
SatisfactionScoreClientCat=Low	1.23	1.15	1.11
SatisfactionScoreClientCat=No Scores	1.33	1.25	*1.13*
FailureRateClientCat=Low	0.51	0.52	0.46
FailureRateClientCat=No Projects	0.51	0.55	0.55
PreviousCollaboration=Yes	0.14	0.21	0.19
FocusLowPriceCat=Low	0.95	0.73	0.65
FocusLowPriceCat=Medium	*0.98*	0.84	0.84
FailureRateProviderRegionCat=High	1.34	1.20	1.21
FailureRateClientRegionCat=High	1.61	1.42	1.28
GeographicalDistance=Neighbor	0.95	0.90	0.81
GeographicalDistance=Offshore	*1.00*	*0.98*	*0.92*
logProjectSize	1.94	1.91	1.48

Several of the smaller projects are not traditional software project, but rather the development of simple scripts or technical support. The observation that the odds ratios remain stable when excluding this suggests that the failure factors are not very different in different types of projects, i.e., that we would not get different results when trying to separate the analysis into subgroups of projects. We conducted the regression analysis on several other sub-groups of projects, e.g., based on type of skill needed, but did not find much difference in odds ratios.

Several of the variables, e.g., the project failure variable, are not easy to define in a manner that makes them objectively measurable and fully consistent with the variety of interpretations. The relatively good prediction accuracy and high robustness of our model, however, give some support in that the model variables approach the phenomena we tried to understand.

We tried to emphasize variables causally related to the risk of project failure, but have only partly succeeded in doing so. Low client skill as indicated by project failure rate is, for example, hardly the direct reason for the failure of a new project, but more of a characteristic that is correlated with the competence of the client, and that increases the likelihood of failure risk actions and non-actions by the client. Perhaps even more problematic is our use of the geographical region to explain project failure. Potentially, there are causal links between the software provider and client cultures in a geographical region, and therefore the higher risk of project failure, but these links are most likely very complex, context dependent, and currently not well understood. Further work should try to better understand the underlying mechanisms of each of the identified failure factors in order to get closer to the ideal of the cause-effect model.

The building of a sound binary logistic regression model is not unproblematic either. Proper interpretation of such models is based on numerous assumptions that are hard to meet in the context of software development. We showed that we must be careful not to interpret the outputs of the model as exact probabilities, but rather as risk values strongly correlated with probabilities.

To further test the robustness of the model we examined how the odds ratios changed when removing variables. If the odds ratios changed very much there could for example be problems related to a high degree of multicollinearity. We found, however, that the odds ratios were quite stable to the removal of variables from the model. This increases our belief in the validity and robustness of the model.

We include several observations with the same client or the same provider. This may violate the assumption of independence among the observations. To check whether this type of dependence would impact the results we repeated the analysis with only one, randomly selected, observation per client and per provider. The remaining data set had about 60,000 projects. The odds ratios were almost the same as in the data set with all client and provider observations included.

To further test the robustness of the analysis we separated the data set into one sub-set per year. We found the odds ratios, i.e., the reasons for project failure, and the failure rate to be almost the same for all included years. The only exepction was the first year, which included only few observations and had a lower failure rate.

5 Conclusions

We report in this paper that the risk of project failures, when defined as projects that have been cancelled or completed with very dissatisfied clients, can be reasonably accurately predicted by using variables known at the time of the project start-up. The prediction model, which is mainly valid in the context of small-scale projects on outsourcing marketplaces, seems quite robust with respect to changes in time and project size. The robustness of the model, the prediction accuracy, and the high number of projects analyzed make us believe that it models relationships between the included variables and the risk of project failure meaningfully. Main implications of the results include:

- The client may substantially reduce the risk of project failure by emphasizing good provider skills rather than low price.
- The best way of ensuring the selection of a skilled provider is to base the skills assessment on previous collaboration, or historical data about the failure rate of the provider from previous projects.
- A skilled client seems to be almost as important for avoiding project failure as a skilled provider.

We plan to follow up with studies on the identified failure factors, such as the client skill, in contexts with larger projects.

Acknowledgement

Thanks to the founder and former CEO of vWorker.com, Mr. Ian Ippolito, and Dr. Matt Barrie at freelancer.com who generously provided the project data.

References

The Standish Group: Chaos. 1995. West Yarmouth, MA, The Standish Group.

Agarwal, N. and Rathod, U. 2006. Defining 'success' for software projects: An exploratory revelation. *International Journal of Project Management* **24**(4): 358-370.

Ahonen, J. J. and Savolainen, P. 2010. Software engineering projects may fail before they are started: Post-mortem analysis of five cancelled projects. *Journal of Systems and Software* **83**(11): 2175-2187.

Al-Ahmad, W., Al-Fagih, K., Khanfar, K., Alsamara, K., Abuleil, S. and Abu-Salem, H. 2009. A Taxonomy of an IT Project Failure: Root Causes'. *International Management Review* **5**(1): 93-105.

Anandalingam, G. and Lucas, H. C. J. 2005. The Winner's Curse in High Tec. *IEEE Computer* **38**(3): 96-97.

Boehm, B. 2000. Project termination doesn't equal project failure. *Computer* **33**(9): 94-96.

Cerpa, N., Bardeen, M., Kitchenham, B. and Verner, J. 2010. Evaluating logistic regression models to estimate software project outcomes. *Information and Software Technology* **52**(9): 934-944.

Charette, R. N. 2005. Why software fails. *IEEE spectrum* **42**(9): 36.

El Emam, K. and Koru, G. 2008. A replicated survey of IT software project failures. *IEEE Software*(September/October): 85-90.

Eveleens, J. L. and Verhoef, C. 2010. The rise and fall of the Chaos report figures. *IEEE Software*(January/February): 30-36.

Fabriek, M., Brand, M. v. d., Brinkkemper, S., Harmsen, F. and Helms, R. 2008. *Reasons for success and failure in offshore software development projects*. European conference on information systems: 446-457.

Fawcett, T. 2006. An introduction to ROC analysis. *Pattern recognition letters* **27**(8): 861-874.

Flyvbjerg, B., Skamris Holm, M. K. and Buhl, S. L. 2004. What causes cost overrun in transport infrastructure projects? *Transport Review* **24**(1): 3-18.

Hashmi, M. T. and Stevrin, P. 2009. High IT Failure Rate: A Management Prospect. *Blekinge Tekniska Hogskola Sektionen for Management*.

Iacovou, C. L. and Nakatsu, R. 2008. A risk profile of offshore-outsourced development projects. *Communications of the ACM* **51**(6): 89-94.

Jørgensen, M. 2013. *A Strong Focus on Low Price When Selecting Software Providers Increases the Likelihood of Failure in Software Outsourcing Projects*. EASE, Porto de Galinhas.

Jørgensen, M., Grimstad, S. and Ieee Comp, S. O. C. 2005. *Over-optimism in software development projects: "The winner's curse"*. 15th International Conference on Electronics, Communications and Computers (CONIELECOMP 2005), Puebla, MEXICO, Ieee Computer Soc: 280-285.

Jørgensen, M., Halkjelsvik, T. and Kitchenham, B. 2012. How does project size affect cost estimation error? Statistical artifacts and methodological challenges. *International Journal of Project Management*.

Jørgensen, M. and Moløkken-Østvold, K. 2006. How large are software cost overruns? A review of the 1994 CHAOS report. *Information and Software Technology* **48**(4): 297-301.

Jørgensen, M. and Sjøberg, D. I. K. 2001. Impact of effort estimates on software project work. *Information and Software Technology* **43**(15): 939-948.

Kern, T., Willcocks and van Heck, E. 2002. The winner's curse in IT outsourcing: strategies for avoiding relational trauma. *California Management Review* **44**(2): 47-69.

King, G. and Zeng, L. 2001. Logistic regression in rare events data. *Political analysis* **9**(2): 137-163.

Leidner, D. E. and Kayworth, T. 2006. Review: A review of culture in information systems research: Toward a theory of information technology culture conflict. *MIS quarterly* **30**(2): 357-399.

Lin, M., Viswanathan, S. and Agarwal, R. 2010. An Empirical Study of Online Software Outsourcing: Signals under Different Contract Regimes. *Available at SSRN 1694385*.

Linberg, K. R. 1999. Software developer perceptions about software project failure: a case study. *Journal of Systems and Software* **49**(2): 177-192.

Maglyas, A., Nikula, U. and Smolander, K. 2010. *Comparison of two models of success prediction in software development projects*. Software Engineering Conference (CEE-SECR), 2010 6th Central and Eastern European, IEEE: 43-49.

McManus, J. and Wood-Harper, T. 2007. Understanding the sources of information systems project failure. *Management services* **51**(3): 38-43.

Nakatsu, R. T. and Iacovou, C. L. 2009. A comparative study of important risk factors involved in offshore and domestic outsourcing of software development projects: A two-panel Delphi study. *Information & Management* **46**(1): 57-68.

Sauer, C., Gemino, A. and Reich, B. H. 2007. The impact of size and volatility - On IT project performance. *Communications of the Acm* **50**(11): 79-84.

Schmidt, R., Lyytinen, K., Keil, M. and Cule, P. 2001. Identifying software project risks: An international Delphi study. *Journal of management information systems* **17**(4): 5-36.

Tichy, L. and Bascom, T. 2008. The business end of IT project failure. *Mortage Banking* **68**(6): 28.

Verner, J., Sampson, J. and Cerpa, N. 2008. *What factors lead to software project failure?* Research Challenges in Information Science, 2008. RCIS 2008. Second International Conference on, IEEE: 71-80.

Westner, M. K. and Strahringer, S. 2010. The current state of IS offshoring in Germany: Project characteristics and success patterns. *Journal of Information Technology Management* **21**(1): 49.

Wohlin, C. and Andrews, A. A. 2001. Assessing project success using subjective evaluation factors. *Software Quality Journal* **9**(1): 43-70.

Wonnacott, T. H. W. and Wonnacott, R. J. 1990. *Introductory Statistics*, John Wiley & Sons.