Latent Trait Analysis for Risk Management of Complex Information Technology Projects

Tsuyoshi Idé, Sinem Güven, Ee-Ea Jan
IBM Research, T. J. Watson Research Center
{tide, sguven, ejan}@us.ibm.com

Sergey Makogon, Alejandro Venegas
IBM Global Technology Services
{smakogon@us, avenegas@cl}.ibm.com

***Abstract*—Recent years have seen a major increase in the application of predictive analytics to the service delivery domain as more and more service providers rely on such analytics for proactive risk management. At the pre-contract stage, identifying potential project risks accurately is of vital importance since it allows service providers to avoid profit erosion through proactive risk management. This paper describes a data-driven approach to project failure prediction of complex information technology (IT) projects. We introduce a novel theoretical framework of Latent Trait Analysis (LTA), whose original form was first developed in psychometrics. We take as the input questionnaire data of risk assessment reviews in the quality assurance (QA) process of IT projects before contract signing, and attempt to predict the project health in the delivery phase after contract signing. The idea is to explicitly capture the human cognitive process through LTA, and estimate the latent project failure tendency hidden behind the questionnaire answers collected by QA experts. Using real QA data of an IT service provider, we demonstrate that our approach outperforms existing approaches in project failure prediction while providing practical information on the usefulness of individual question items.**

I. Introduction

Recent years have seen a major increase in the application of predictive analytics to the service delivery domain as more and more service providers rely on such analytics for proactive risk management. In order to effectively manage project risks while ensuring high quality service delivery, service providers typically mandate a Quality Assurance (QA) process, where QA experts iteratively conduct risk assessment reviews in the solution design phase (prior to contract signature) to check the feasibility and the potential profitability of projects. At the pre-contract stage, identifying potential project risks accurately is of vital importance since it allows service providers to avoid profit erosion through proactive risk management.

Within the QA process, pre-defined and standardized questionnaires are used to cover various aspects of the business and technical risk factors for a given project. For each risk assessment question (or risk factor), qualified QA experts determine the answer (or risk level) based on their observations and expertise. For example, a risk factor of "Lack of awareness of customer requirements" may be encoded into five levels, where 1 represents "no risk," and 5 represents "exceptionally high risk." If, at least, some of the risk factors exhibit high risk levels, it is straightforward for QA experts to direct the solution design team to take risk mitigation actions before moving on to contract signature stage. As illustrated by Fig. 1, human experts serve as an "intelligent sensor" that translates the situation of complex IT projects into numerical values through a cognitive process.

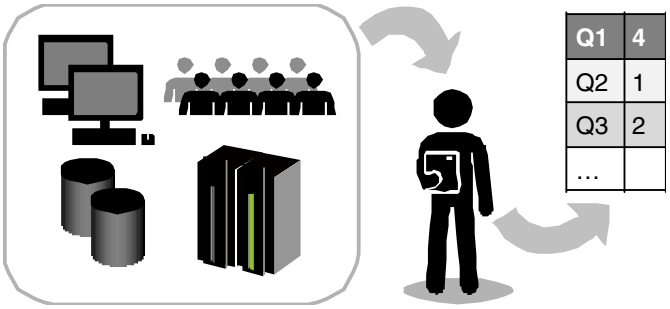

Fig. 1. A surveyor (Quality Assurance expert) is an intelligent sensor that transforms the latent risk of an IT project into integer risk levels.

The contents of the questionnaire are developed by QA experts based on their experience. Although QA reviews, based on human judgment alone, are generally effective when performed iteratively, there are still practical limitations when they are used to predict the risk of project failure after contract signature. *First*, due to the dependence of individual assessment questions, the total summation over them does not necessarily represent the true risk. For example, a slight indication of risk in one factor may be an indication of a serious trouble when observed simultaneously with a slight indication of risk in another factor. Thus, *second*, it is hard to define a practical indicator for project failure prediction that takes a balance between the prediction accuracy and interpretability. In particular, there are few practical approaches to quantitatively measure the usefulness of each of the question items in terms of project failure prediction.

Because of these limitations, growing attention has been paid to data-driven predictive approaches to accurately detect such subtle indications of future IT project troubles that would not be apparent to the human experts. For instance, Mojsilović et al. [1] proposed a logistic regression framework to quantify contractual risk in strategic outsourcing engagements. Ratakonda et al. [2] used the QA questionnaire data to understand typical patterns of project troubles with the aid of decision trees. Ray et al. [3] also used the QA data and proposed a framework to calculate the likelihood of individual risk factors using a k-nearest neighbor (k-NN) method. Figure 2 illustrates a typical contract risk management process based on QA questionnaire data. The project health indicator after contract signature, y, is predicted based on the pre-contract QA questionnaire data, $\boldsymbol{x}$.

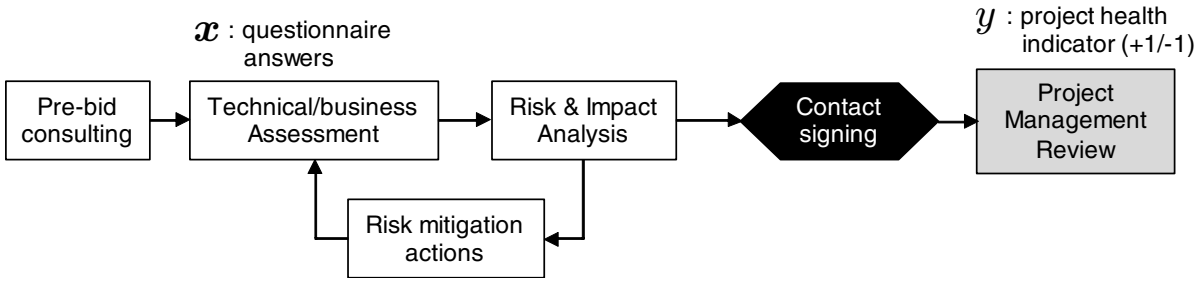

Fig. 2. Typical contract risk management process.

Although the previous work [1], [2], [3] report relatively high prediction accuracies, they do not pay full attention to the *generative model* of the assessment questionnaire data, and thus give only limited amount of information on the usefulness of the individual assessment questions. As the technology trend and business requirements change significantly over time, it is important for QA experts to have fully understood the usefulness of individual risk assessment questions in terms of their potential to detect risks. Due to the lack of a systematic approach to determine the minimum set of required questions that would cover all important risk factors, QA questionnaires tend to get more and more complex over time, resulting in prohibitive overhead.

Analyzing the generative model of individual assessment questions generally requires a model of the human cognition process (see Fig. 1). This is generally a challenging task given the complex intellectual processes of human being, which should be nonlinear and non-deterministic in nature. In this paper, we explicitly model the human cognition process using a theoretical model developed in psychometrics, and make use of it for project failure prediction. Thanks to the model that is tailored to understand the latent ability of human, our framework is capable of providing practical information on the usefulness of individual questions (risk factors) in a risk assessment questionnaire. The key ingredient of our framework is a probabilistic model called Latent Trait Analysis (LTA) [4]. LTA is a widely used technique to standardize academic achievement tests, such as SATs [5]. As is well known, the SAT score is not necessarily proportional to the summation of the number of correctly answered questions. This is very similar to the service delivery setting, where the sum of risk factors is not necessarily a good indicator of project failure (see Subsection II-B).

As the name suggests, LTA was originally proposed to quantitatively measure the ability of examinees, which is latent in the sense that it cannot be directly observed. As explained in a later section, LTA is based on a specific form of probability model that is tailored to capture the latent human ability most effectively. We know that questionnaire answers are affected by psychological perceptions. The numerical numbers of the risk levels are not ordinary statistical quantities that are typically assumed to follow the Gaussian distribution. The true failure tendency would be nonlinearly transformed by human cognition to give the directly observed values of questionnaire answers, as illustrated in Fig. 1. Just like SAT, which attempts to find the latent ability of examinees, our goal is to retrieve the latent failure tendency of complex IT projects with the aid of LTA. To the best of our knowledge, this is the first work to explicitly model the human cognition process in the area of project risk management.

The layout of the paper is as follows. Section II briefly explains the data we use in this study. Section III introduces the theory of LTA in the context of project risk management. Section IV extends the traditional LTA model to include multiple latent variables and explains how LTA is incorporated into a project failure prediction framework. Section V describes experimental results. Finally, Section VI concludes the paper.

II. Problem Setting

This section provides details on the data set used for project failure prediction, and formally states the problem.

A. Contractual risk mitigation process

As explained in Introduction, we focus on QA assessment data for project health prediction. Figure 2 shows a typical contract risk management process. A service provider starts with pre-bid consulting once a request for proposal is received from a potential customer. Next, QA experts of the service provider assess both technical and business aspects of the new deal by filling out pre-defined questionnaires, through which various risk factors are identified as risks. If those risks are judged as impactful to the profitability, or any other aspects of the contract, such as completeness of the technical solution, client satisfaction or delivery execution, corresponding risk mitigation actions are taken (through e.g., modification of solution, negotiation on service level agreements, adjustment of price, etc.) to eliminate the identified risk factors. Once the relevant risks are mitigated, the project moves on to the contract signing stage, which is followed by service delivery. In the service delivery phase, Project Management Reviews (PMR) are periodically conducted to check well-defined health metrics associated with the project.

B. Quality assurance questionnaire data

In this study, we take the QA assessment data as the input, and the project health indicator as the target. Although there are multiple questionnaires in the QA process, we focus on one type of questionnaire, which we call Contract Risk Assessment (CRA), containing 22 rather qualitative questions. Examples of the questions include:

- Service provider's relationship with the customer
- Experience in the planned solution

- Completeness of the cost case
- Feasibility of the schedule

Based on their experience and expertise, QA experts evaluate each of the factors and rate them using an integer number from 1 to 5, 1 representing "no risk," and 5 representing "exceptionally high risk."

Although projects typically go through multiple iterative assessment-mitigation cycles, in this study, we focus on the last CRA assessment as input to the predictive model, denoted by $\boldsymbol{x}^{(n)}$ for the n-th project in the training data set, as it best represents project risks right before contract signature.

After contract signature, metrics, such as project health indicator, are tracked on periodically, and QA experts review the progress of the plan based on various additional information sources including interviews with the delivery team. Typically, the project health indicator includes multiple sub-indicators such as financial, technical, and project management status. For simplicity, we focus on the financial health indicator (represented by one of A, B, C, D based on business definitions), since financial health is known to be a dominant indicator of project failures. If the indicator falls in either C or D, the project is considered to be (financially) troubled. For the purposes of our study, we take the PMR with the worst financial health indicator as the target, which is denoted by y. For simplicity, we assume y takes the value of $+1$ when troubled (*i.e.* C or D), and -1 otherwise.

C. Characterizing quality assurance questionnaire data

In the QA data set, several hundred projects are present, only several tens of which have the troubled status. Note that, first, the distribution of project size is long-tailed. Some of the projects are very big, and thus, the heterogeneity over the projects is huge. In fact, the median is several times smaller than the mean project size. Due to the huge variety in the complexity of the projects, quantifying the risk factors is very challenging. To the best of our knowledge, questionnaire-based quantification is the only practical approach known so far to data-driven project failure prediction.

Second, since we focus on the questionnaire answers in the last assessment cycle, the majority of risk factors have been already mitigated, and thus, if any, there are only slight indications of risks in $\boldsymbol{x}^{(n)}$ that would not be readily comprehensible to human experts. Figure 3 summarizes the distribution of the risk levels for each of the (A, B, C, D) PMR ratings. We see that about 60% of the items is answered as 1 (no risk) irrespective of the PMR ratings. More interestingly, there is no clear trend that troubled projects get larger total risk levels. Figure 4 depicts the estimated distribution of the total risk level defined simply by the summation over 22 questionnaire answers, calculated for several hundred projects. The RBF (radial basis function) kernel with the bandwidth 2 was used for density estimation. The distribution of the D-rated projects is almost the same as the B-rated ones. These features of the data clearly show the importance of the heterogeneity and the correlation among the individual risk factors to discriminate between healthy and troubled projects.

Third, the data set is highly imbalanced in the sense that the majority of samples is about healthy projects. This renders

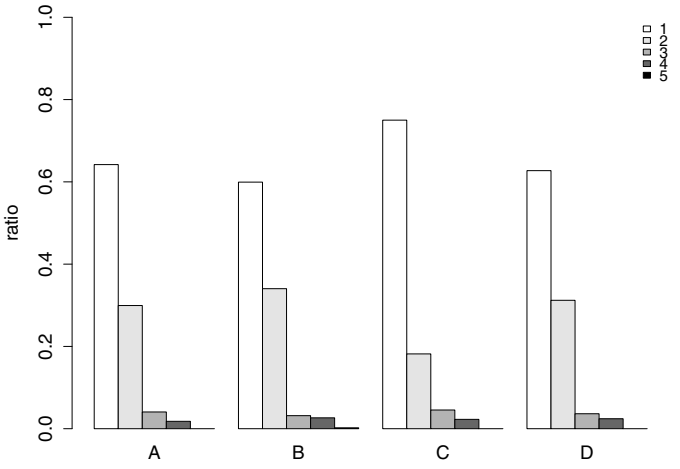

Fig. 3. Distribution of risk revels for each PMR health rating.

naive applications of existing machine learning technologies inappropriate for binary classification. In the previous work [2], [6], some data set selection seems to have been performed to ensure the applicability of binary classifiers. In our study, we do not make any arbitrary pre-selection of samples.

D. Problem statement

Now, we formally state the problem setting. We are given a training data set:

$$\mathcal{D} = \{(\boldsymbol{x}^{(n)}, y^{(n)}) \mid n = 1, 2, \ldots, N\}, \tag{1}$$

where $\boldsymbol{x}^{(n)}$ is an M-dimensional vector representing the CRA questionnaire answers ($M = 22$), and $y^{(n)}$ is the PMR health rating. Each of the dimensions of $\boldsymbol{x}^{(n)}$ takes an integer value in the predefined risk levels, while $y^{(n)}$ takes either of $+1$ or -1 (troubled or healthy). The number of projects is denoted by N. In the CRA data, N is on the order of several hundred.

Our problem is to develop a framework of quantifying the usefulness of each of the M inputs in terms of predictability of y. Note, first, that an indicator for the usefulness must be readily understood by users (QA experts and project managers who are not necessarily experts in analytics). For example, we could use some sort of linear model and might take the regression coefficients as the measure of the usefulness. However, such approaches are not useful in the present context since they are generally unbounded real numbers that can even be negative.

III. Latent Trait Analysis for Project Risk Analysis

This section explains a new framework of latent trait analysis for contract risk analysis.

A. Probability distribution for at-risk answers

Imagine that we have a latent variable representing the tendency of project failure, and denote it by a scalar variable θ.

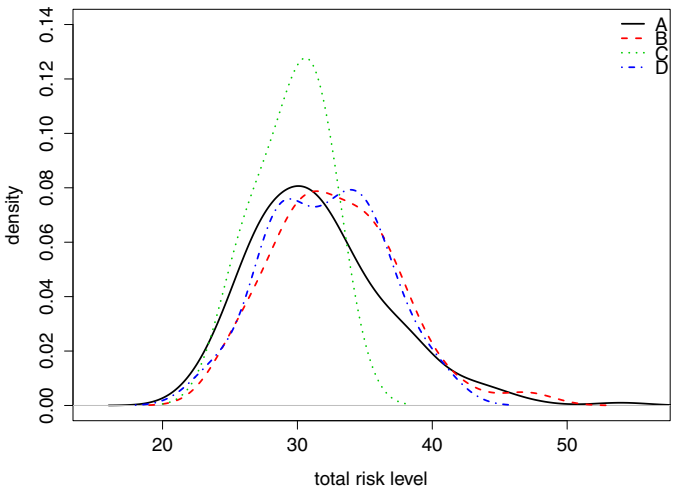

Fig. 4. Distribution of total risk level for each PMR health rating.

A project takes an assessment test consisting of M questions to yield a binary value. For simplicity, we restrict ourselves to the single grade question, where 1 represents at-risk, and 0 represents no-risk. This simplification is clearly justified by the highly skewed distribution shown in Fig. 3. An "answer sheet" is represented by an M-dimensional binary vector $\boldsymbol{x} \in \{0,1\}^M$.

For each question, say the i-th question, we model the probability of at-risk as a modified version of logistic function:

$$P(\theta, a_i, b_i, c_i) \equiv c_i + \frac{1-c_i}{1+\mathrm{e}^{-a_i(\theta-b_i)}}, \tag{2}$$

where a_i, b_i, c_i are the model parameters of the i-th question and are usually called the discrimination, difficulty, and guessing parameters, respectively. The guessing parameter must satisfy the condition of $0 \le c_i \le 1$.

Figure 5 depicts P as a function of θ. Since P is a monotonically increasing function of θ as long as the discrimination parameter is positive, we see that the more value of θ a project has, the more likely it is for the QA expert to choose the at-risk option, $x_i = 1$. Notice that the nonlinear curve naturally captures the bias of human that they tend to be overly optimistic on the lower risk side while overly cautious on the higher risk side.

To get more intuition of this model, consider the limit of $\theta \to \pm\infty$. If the latent project failure variable goes to positive infinity, P takes the value of 1. This means that QA experts will choose the at-risk option if a project is evidently in trouble. If it goes to negative infinity, P goes to c_i. This means that for a given risk assessment question, the QA experts may choose the option of at-risk even if a project is completely healthy and its latent failure tendency is infinitely small. In our context, the parameter c_i represents the possibility that QA experts use a random guess to fill out the questionnaire, or simply make a mistake in doing so.

For the difficulty parameter b_i, it is obvious that it plays a role of a threshold of risk. The probability P takes a value near 1 when $\theta - b_i$ is large. Thus when b_i is large, only very risky projects having a large θ are allowed to take $x_i = 1$. Thus the LTA model allows *automated threshold tuning* for the individual question items.

By putting all together, the probability of an answer pattern $\boldsymbol{x}$, which contains M answers given by a project having the latent failure tendency θ is given by

$$p(\boldsymbol{x} \,|\theta, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \prod_{i=1}^{M} P(\theta, a_i, b_i, c_i)^{\delta(x_i,1)} [1 - P(\theta, a_i, b_i, c_i)]^{\delta(x_i,0)}, \tag{3}$$

where δ represents Kronecker's delta, and x_i is the answer to the i-th question. Also, we defined $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$ as $(a_1, \ldots, a_M)^\top$, $(b_1, \ldots, b_M)^\top$, and $(c_1, \ldots, c_M)^\top$, respectively.

Although the model (2) seemingly looks like a modified version of logistic regression, the problem setting is totally different from supervised learning. The latent failure tendency θ is an unobserved latent variable, and what we can know is only how the QA experts answered the risk assessment questions for each of the projects. We are attempting to estimate θ as well as the model parameters, $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$, based on a collection of answer sheet from the N projects. Thus the problem falls in the category of unsupervised learning.

B. Maximum a posteriori estimation for LTA Parameters

To capture the dispersion of the latent failure tendency, the LTA model assumes the standard Gaussian distribution as the prior distribution for θ:

$$f(\theta \mid \gamma, \omega) = \sqrt{\frac{\gamma}{2}} \exp\left\{-\frac{\gamma}{2}(\theta - \omega)^2\right\}, \tag{4}$$

where γ and ω are thought of as given constants for now.

Following the Bayesian learning framework, given the data $\mathcal{D}$, the unknown model parameters $\{\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}\}$ are given as the MAP (maximum a posteriori) solution that maximizes the log marginal likelihood:

$$\max_{\boldsymbol{a},\boldsymbol{b},\boldsymbol{c}} L(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \,|\mathcal{D}, \omega, \gamma) \quad \text{subject to} \quad 0 \le c_i \le 1 \quad (i = 1, \ldots, M), \tag{5}$$

where

$$L(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \mid \mathcal{D}, \omega, \gamma) \equiv \sum_{n=1}^{N} \ln \int_{-\infty}^{\infty} \mathrm{d}\theta^{(n)} \; f(\theta^{(n)}) \; p(\boldsymbol{x}^{(n)} | \theta^{(n)}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}). \tag{6}$$

Here $\theta^{(n)}$ is the latent trait (or failure tendency) of the n-th project.

To handle the constraint on the guessing parameter, the method of barrier function can be used. Specifically, we replace the marginal likelihood L with the following objective function

$$\tilde{L}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \mid \mathcal{D}, \omega, \gamma) \equiv L(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \mid \mathcal{D}, \omega, \gamma) + \mu_1 \sum_{i=1}^{M} \ln c_i + \mu_2 \sum_{i=1}^{M} \ln(1 - c_i) \tag{7}$$

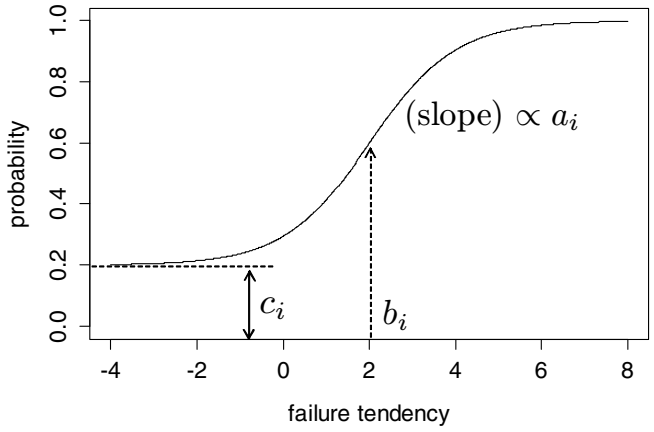

Fig. 5. Item Characteristic Curve.

and solve the unconstrained optimization problem using the gradient method combined with line search for the step width [7] and the Gauss-Hermite quadrature. See [8] for details. Typically, the solution is not very sensitive to the choice of the coefficients μ_1 and μ_2. We set $\mu \equiv \mu_1 = \mu_2$.

C. Estimating the latent failure tendency

Once the MAP solution for the LTA parameters $\hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{c}}$ is obtained, the predictive distribution of the latent failure tendency θ for an arbitrary $\boldsymbol{x}$ is given by

$$p(\theta \mid \boldsymbol{x}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{c}}) \propto p(\theta \mid \gamma, \omega)\, p(\boldsymbol{x} \mid \theta, \boldsymbol{x}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{c}}) \tag{8}$$

This is readily given by Bayes' theorem. Unfortunately, the prior distribution is not conjugate to $p(\boldsymbol{x} \mid \theta, \boldsymbol{x}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{c}})$ and thus an analytic expression is hard to obtain. We thus attempt to make point estimation for θ by choosing the value of the maximum probability density, namely,

$$\hat{\theta} = \arg\max_{\theta} p(\theta \mid \boldsymbol{x}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{c}}). \tag{9}$$

To solve this, we consider the equation:

$$\frac{\partial}{\partial \theta} \ln p(\theta \mid \boldsymbol{x}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{c}}) = 0.$$

After some algebra, this leads to the following fixed-point equation

$$\gamma(\theta - \omega) = \sum_{i=1}^{M} \frac{\hat{a}_i(1-\hat{c}_i)}{1+\mathrm{e}^{-\hat{\phi}_i}} \left\{ \frac{\delta(x_i,1)}{\hat{c}_i + \mathrm{e}^{\hat{\phi}_i}} + \frac{\delta(x_i,0)}{1-\hat{c}_i} \right\}, \tag{10}$$

where $\hat{\phi}_i \equiv \hat{a}_i(\theta - \hat{b}_i)$. This equation is solved using any numerical solver, or simply using iterative substitution between the left- and right-hand sides.

IV. Project Failure Prediction Framework

This section describes how LTA is incorporated into the project failure prediction framework.

A. Multiple latent variable model

Solving Eq. (9), a point estimate of $\hat{\theta}$ is calculated for an arbitrary $\boldsymbol{x}$. Since θ is introduced as the latent failure tendency, one straightforward approach to predict y is to use $\hat{\theta}$ as a surrogate of $\boldsymbol{x}$. When a new questionnaire answer $\boldsymbol{x}$ comes in, we translate it into θ, and perform classification of y in the space of θ. The classification can be done by k-nearest neighbor (k-NN) method.

One practical issue of this approach is that in general questionnaires are designed to include multiple categories that are loosely coupled with each other, and thus reducing $\boldsymbol{x}$ to a single scalar variable θ may have a risk of oversimplification. In the case of QA questionnaire, it typically includes categories such as the goodness of relationship among different parties (customer, subcontractors, internal teams, etc.) and the feasibility of technical solutions themselves. Although these categories are not completely independent, handling them as groups leads to better interpretability in practice.

To capture such a hierarchical structure, we extend the original LTA framework to include multiple latent variables. We partition the M question items into several disjoint groups $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_G$, and, instead of Eq. (3), we assume

$$p(\boldsymbol{x} \mid \theta, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \prod_{g=1}^{G} P_g(\boldsymbol{x} \mid \theta_g, \boldsymbol{a}_g, \boldsymbol{b}_g, \boldsymbol{c}_g), \tag{11}$$

where the g-th term is defined as

$$P_g(\boldsymbol{x} \mid \theta_g, \boldsymbol{a}_g, \boldsymbol{b}_g, \boldsymbol{c}_g) \equiv \prod_{l \in \mathcal{M}_g} P(\theta_g, a_{g,l}, b_{g,l}, c_{g,l})^{\delta(x_l,1)} \times [1 - P(\theta_g, a_{g,l}, b_{g,l}, c_{g,l})]^{\delta(x_l,0)} \tag{12}$$

The probability of answering at-risk, $P(\cdot,\cdot,\cdot,\cdot)$, has been defined by Eq. (2).

Corresponding to this partition, we also assume that the prior distribution is partitioned accordingly:

$$f_G(\theta \mid \gamma, \omega) = \prod_{g=1}^{G} f(\theta_g \mid \gamma, \omega), \tag{13}$$

where $f(\theta_g \mid \gamma, \omega)$ on the right hand side is defined by Eq. (4). We use common hyper-parameters for each partition for simplicity.

Equations (11) and (13) suggests that the model assumes statistical independence between the partitioned groups. This assumption drastically simplifies the formulation as compared to the partial credit model [8], which also introduces multiple ability parameters in the model. Unlike academic tests, it is often the case in QA questionnaire that the number of the projects N is on the same order of the number of question items M. The traditional partial credit model is known to require a huge amount of data for stable parameter estimation, and thus inappropriate in our case.

Thanks to the independence assumption, to find the latent failure tendency, we simply solve Eqs. (5) and (9) independently for each g, ending up with a point-estimated G-dimensional latent variable $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \ldots, \hat{\theta}_G)^\top$.

B. Failure perdition

Based on the multiple latent variable model, a point estimation for the G-dimensional latent failure tendency $\hat{\boldsymbol{\theta}}$ is obtained for an arbitrary $\boldsymbol{x}$. As stated in the previous subsection, to predict y for a new project having a questionnaire answer $\boldsymbol{x}$, we simply apply k-NN method for failure prediction. Specifically, we first pick k samples that are closest to the estimated $\hat{\boldsymbol{\theta}}$ in the latent failure tendency space. For the distance metric, we use the Euclidean distance

$$d(\boldsymbol{x}, \boldsymbol{x}^{(n)}) = \sum_{g=1}^{G} w_g \left(\hat{\theta}_g - \hat{\theta}_g^{(n)}\right)^2 \tag{14}$$

where $\hat{\theta}_g^{(n)}$ is the point-estimated latent failure tendency for the g-th group of $\boldsymbol{x}^{(n)}$. We included w_g as the weight for the group g to handle prior knowledge on the relative importance between different item groups. Once k-nearest neighbor samples are identified in $\mathcal{D}$, we check the values of y of the selected samples. If the decision score

$$a(\boldsymbol{x}) \equiv \ln \frac{N_{+1}^k}{N_{-1}^k}, \tag{15}$$

where N_y^k is the number of samples of the class of y in the NNs, is greater than a threshold, then we classify the instance into the $y = +1$ (troubled) class. The threshold is typically optimized using leave-one-out (LOO) cross validation (CV).

V. Experiment

This section presents results of experimental evaluation of our project failure prediction framework. The data we use is the CRA questionnaire data having $M = 22$ question items over several hundred contracts as described in Section II.

A. Learning LTA model

Based on the predefined subcategories of CRA, we partitioned the 22 question items into four groups ($G = 4$), which respectively correspond to (1) communication issues with the client, (2) the well-definedness of the project scope, (3) the feasibility of the delivery plan, and (4) project management issues related to subcontractors and internal teams. To solve Eqs. (5) and (9), we used the `ltm` package [9] in R. The hyper-parameters were fixed as $\omega = 0, \gamma = 1$. As previously mentioned, no pre-selection of sample was made. As a result, the data is highly imbalanced in the sense that the majority of the projects are healthy ($y = -1$).

Figure 6 shows ICCs for the group of $g = 3$, which contains from 12th through 17th risk assessment questions. We drew $P(\theta_g, a_i, b_i, c_i)$ with the solid lines as well as $[1 - P(\theta_g, a_i, b_i, c_i)]$ with the dashed lines. We clearly see that the 17th question is hardly useful to discriminate between the troubled and healthy statuses. This question is actually a very formal question on service pricing, and is expected to be less dependent on the quality of delivery plans. We also see that 12th and 15th question items are understood as less sensitive indicators of project failure in the sense that they turn on only for those evidently at-risk. In contrast, the 14th and 16th questions are useful to pick up subtle indication of project failure. These questions ask straight about how much clear and realistic the project plan is, and they are likely to effectively capture the risk of future project failure. In this way, the ICCs provide very useful information on questionnaire design.

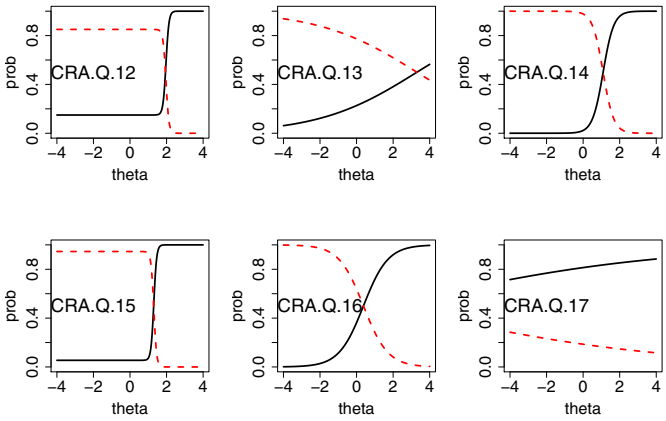

Fig. 6. Examples of item characteristic curves. The solid lines represent the probability answering at-risk P while the dashed lines represents the probability of answering no-risk $(1 - P)$.

B. Latent project failure tendency

Based on the learned LTA model shown in Fig. 6, we calculated the latent project tendency θ_g for each of the samples in $\mathcal{D}$ using Eq. (10). The result is shown in Fig. 7. Although Eq. (10) gives only point-estimated values $\{\theta_g^{(1)}, \ldots, \theta_g^{(N)}\}$, we performed kernel density estimation [10] to capture the overall trend of θ_g. The bandwidth of the RBF kernel [10] was chosen as 0.18 and 0.32 for healthy ($y = -1$) and troubled ($y = +1$) projects, respectively.

In the figure, we clearly see that troubled projects (denoted by the dashed curve) tend to have more value of the latent failure tendency than healthy projects. It is interesting to compare Fig. 7 with Fig. 4. As discussed in Subsection II-C, there is no clear proportional relationship between the total number of at-risk answers and the actual likelihood of project failure. The C-rated projects look even better than B-rated project in Fig. 4. Here, thanks to the nonlinear transformation by the LTA-based model, we conclude that the $\boldsymbol{\theta}$ can be a much better indicator of project failure.

C. Health indicator prediction

Finally, to validate the LTA model, we evaluated the prediction accuracy on the binarized PMR financial project health indicator y. The performance was evaluated by the F-value defined by

$$f = \frac{2r_1 r_2}{r_1 + r_2},$$

which is the harmonic mean between the prediction accuracy in the healthy projects, r_1, and the prediction accuracy in the troubled projects, r_2. Here we remind the readers that we are working on a highly imbalanced data set. For example, if only 5 projects are troubled out of 100 projects, it is easy to achieve 95% "overall accuracy" by always predicting $y = -1$ (healthy). Our metric is different from the overall accuracy. If

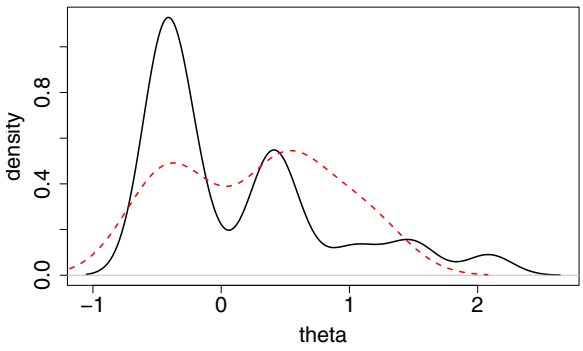

Fig. 7. Distribution of point-estimated latent project failure tendency for the group $g = 3$. The solid and dashed lines denote healthy and troubled projects, respectively.

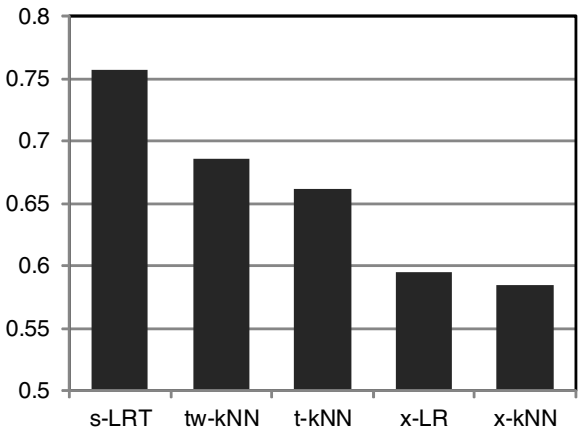

Fig. 8. Comparison of failure prediction accuracies.

a predictor simply ignores the minority class, the F-value will be zero.

To decide on the number of NNs (see Subsection IV-B), we used LOO CV. For example, to check hit or miss for the n-th sample in the training data set $\mathcal{D}$, we held out the sample from $\mathcal{D}$, and learnt the model from the remaining $N-1$ samples to make prediction for that sample. The threshold value for a for the k-NN classification is fixed as the ratio of healthy samples to troubled projects.

In addition to the proposed approach with $G = 4$, we evaluated other classification algorithms for comparison:

- `x-kNN`: A baseline method of k-NN classification in the $\boldsymbol{x}$ space [3], where Eq. (14) is replaced with

$$d_{\mathsf{A}}(\boldsymbol{x}, \boldsymbol{x}^{(n)}) = \sum_{i,j=1}^{M} \mathsf{A}_{i,j}(x_i - x_i^{(n)})(x_j - x_j^{(n)}) \quad (16)$$

  with $\mathsf{A}_{i,j} = \delta_{i,j}$. The original five-graded CRA risk levels were used as-is without binarization. The number of k was optimized via LOO CV.

- `x-LR`: The other baseline method using logistic regression (LR) [1] in the $\boldsymbol{x}$ space. The original five-graded CRA risk levels were used as-is without binarization. Upon training, bootstrap resampling was performed for the troubled projects to obtain the same sample size as the healthy projects. The decision threshold was optimized via LOO CV.

- `t-kNN`: The proposed k-NN classification approach in the $\boldsymbol{\theta}$ space with a uniform weight ($w_g = 1$ for $g = 1, \ldots, 4$).

- `tw-kNN`: The proposed k-NN classification approach in the $\boldsymbol{\theta}$ space with a tuned weight (see the text).

For the weight for `tw-kNN`, we used a simple 0-1 weighting. Specifically, we picked one of the four groups, and simply turned off the weight for the selected group, leaving the other weights unchanged as one.

Figure 8 shows the comparison of the best classification accuracies. The prediction accuracies of the baseline methods are as low as 0.59, which is much lower than previously reported accuracies [2], [3]. As suggested in Subsection II-C, our extensive experimental study shows that the task of failure project prediction is so challenging that achieving a practical accuracy is very hard when simply using existing classification approaches as long as we do not make any pre-selection of samples. Figure 8 also shows that the proposed approaches significantly improve the accuracies of the baseline methods. In `x-kNN` and `x-LR`, the distance between instances is calculated by disregarding the distinction between the ordinal scale and the ratio scale. In our method, the ordinal scale of questionnaire answers is translated into the ratio scale of the latent project failure tendency in the way that it naturally captures the human cognitive process. The result of Fig. 8 clearly shows the translation is critical for prediction.

For `tw-kNN`, the best accuracy was achieved when $w_1 = 0$ while $w_2 = w_3 = w_4 = 1$. It is interesting to see that `tw-kNN`, which omits the $g = 1$ group, gives a better performance than the full model of `t-kNN`. The omitted group is about communication issues with the client. One might think that this does not make sense because smooth and candid communication is obviously a prerequisite for successful projects. This is actually a tricky part of QA questionnaire analysis. As shown in Fig. 2, the contract risk management process is an *iterative* process. If a project has an obvious problem in such an important point as customer relationship, the management team is likely to take immediate actions including replacement of the project manager. Otherwise, the project will not be approved for contract signing. As the result, questions on customer relationship are not very useful for data-driven analysis, which is performed based on the last CRA assessment.

The fact that the approach with a tuned weighting over individual question items produced a better performance suggests the importance of non-uniform weighting on different variables. It is expected to achieve further improvement by optimizing the weighting factors. In fact this is the case. Figure 8 also shows our preliminary result of `s-LRT`, which solves an optimization problem for the distance metric in Eq. (16). The improvement was significant. However, due to space limitations, we leave detailed discussion to a separated paper [11].

VI. Concluding remarks

We have presented a new framework of contract risk management in the solution design phase of IT projects. We introduced a novel approach based on latent trait analysis to model the complex human recognition process of project risk assessment. We extended the traditional LTA to include multiple latent variables to handle naturally defined groups of question items in the IT project assessment.

From a mathematical perspective, our framework translates the ordinal scale of graded answers into the ratio scale, so that the notion of Euclidean (or even Riemannian) distance is well-defined. We demonstrated that our framework achieves a much better prediction performance of project failure while providing practically useful information on the usefulness of individual question items in the form of item characteristic curve.

For future work, it would be interesting to further explore the distance weight optimization approach briefly introduced in the last section. Li et al. [6] recently proposed the use of metric learning for k-NN, which is essentially to optimize A in Eq. (16), in the context of contractual financial risk prediction. Although their problem setting is different from ours in that they leverage extra information regarding contract similarities, it would be interesting to consider how it is integrated into our framework.

References

[1] A. Mojsilović, B. Ray, R. Lawrence, and S. Takriti, "A logistic regression framework for information technology outsourcing lifecycle management," *Computers & Operations Research*, vol. 34, no. 12, pp. 3609–3627, Dec. 2007.

[2] K. Ratakonda, R. Williams, J. Bisceglia, R. Taylor, and J. Graham, "Identifying trouble patterns in complex it services engagements," *IBM Journal of Research and Development*, vol. 54, no. 2, pp. 5:1–5:9, March 2010.

[3] B. K. Ray, S. Tao, A. Olkhovets, and D. Subramanian, "A decision analysis approach to financial risk management in strategic outsourcing contracts," *EURO Journal on Decision Processes*, pp. 1–17, 2013.

[4] M. Wilson, *Constructing Measures*. Psychology Press, 2004.

[5] Wikipedia, "http://en.wikipedia.org/wiki/SAT."

[6] Z. Li, S. Tao, and H. Xiong, "Déjà vu: Assessing similarity between service contracts for risk prediction," in *Proc. 2014 IEEE International Conference on Services Computing*, 2014, pp. 147–154.

[7] W. Murray and M. H. Wright, "Line search procedures for the logarithmic barrier function," *SIAM Journal on Optimization*, vol. 4, no. 2, pp. 229–246., 1995.

[8] F. B. Baker and S.-H. Kim, *Item Response Theory: Parameter Estimation Techniques* , 2nd ed. CRC Press, 2004.

[9] D. Rizopoulos, "ltm: An R package for latent variable modeling and item response theory analyses," *Journal of Statistical Software*, vol. 17, no. 5, pp. 1–25, 2006.

[10] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* , 2nd ed. Springer, 2009.

[11] T. Idé and A. Dhurandhar, "Informative prediction based on ordinal questionnaire data," submitted.