

Avaliação de Riscos em Projetos de Software usando Modelagem de Redes Bayesianas
Risk Evaluation in Software Project Using Bayesian Network Modeling
Evaluación de riesgos en proyectos de software mediante el modelado de redes bayesianas

Recebido: 11/11/2020 | Revisado: 19/11/2020 | Aceito: 23/11/2020 | Publicado: 27/11/2020

Adalberto Ramos Cassia

ORCID: <https://orcid.org/0000-0001-8741-8837>

Universidade Nove de Julho, Brasil

E-mail: adalb2002@gmail.com

André Felipe Librantz

ORCID: <https://orcid.org/0000-0001-8599-9009>

Universidade Nove de Julho, Brasil

E-mail: librantz@uni9.pro.br

Ivanir Costa

ORCID: <https://orcid.org/0000-0003-4890-8133>

Universidade Nove de Julho, Brasil

Email: ivanirc@uni9.pro.br

Mauro de Mesquita Spinola

ORCID: <https://orcid.org/0000-0002-5147-9395>

Universidade de São Paulo, Brasil

E-mail: mauro.spinola@usp.br

Erika Midori Kinjo

ORCID: <https://orcid.org/0000-0002-6348-6162>

Universidade Nove de Julho, Brasil

E-mail: midori.kinjo@gmail.com

Resumo

Os eventos de risco de projeto geralmente são influenciados um pelo outro e raramente agem de forma independente. Nesse contexto, são necessários métodos eficazes para identificar, modelar e analisar esses riscos. O objetivo deste artigo é aplicar a análise de risco em um projeto de desenvolvimento de software, com base no modelo do Software Engineering Institute (SEI), usando o modelo Bayes para calcular as probabilidades de riscos e também a

estrutura de cálculo Noisy-OR para atribuir pesos iniciais à rede de fatores de riscos que influenciam o projeto. Dessa forma, espera-se aumentar as chances de sucesso do processo de análise de risco. Os resultados obtidos pelas técnicas adotadas demonstram ser promissores no auxílio ao processo de tomada de decisão pelos gerentes de projetos de desenvolvimento de software.

Palavras-chave: Projeto de software; Análise de risco; Modelagem bayesiana; Noisy-OR.

Abstract

Project risk events are often influenced by each other and rarely act independently. In this context, effective methods to identify, model and analyze these risks are necessary. The objective of this article is to apply the risk analysis in a software development project, based on the model of the Software Engineering Institute (SEI), using the Bayes model to calculate the event probabilities and also the Noisy-OR calculation structure to assign the initial weights of the network of factors that influence the project. In this way, it is expected to increase the chances of success of the risk analysis process. The results obtained by the techniques adopted prove to be promising in assisting the process of decision making by the managers of software development projects.

Keywords: Software project; Risk analysis; Bayesian modeling; Noisy-OR.

Resumen

Los eventos de riesgo del proyecto a menudo están influenciados entre sí y rara vez actúan de forma independiente. En este contexto, se necesitan métodos efectivos para identificar, modelar y analizar estos riesgos. El propósito de este artículo es aplicar el análisis de riesgo a un proyecto de desarrollo de software, basado en el modelo del Software Engineering Institute (SEI), utilizando el modelo de Bayes para calcular las probabilidades de riesgo y también la estructura de cálculo Noisy-OR. Asignar pesos iniciales a la red de factores de riesgo que influyen en el proyecto. Por lo tanto, se espera que aumente las posibilidades de éxito del proceso de análisis de riesgos. Los resultados obtenidos por las técnicas adoptadas resultan prometedores para ayudar en la toma de decisiones por parte de los responsables de proyectos de desarrollo de software.

Palabras clave: Proyecto de software; Análisis de riesgo; Modelado bayesiano; Ruidoso-OR.

1. Introdução

Em uma economia baseada em informações, a indústria de software tem desempenhado um papel significativo, contribuindo para o crescimento das atividades produtivas em todos os setores. Todavia, surgiram novos desafios no que diz respeito ao gerenciamento de projetos criados para o desenvolvimento de software, realizando um esforço sistemático para gerenciar os riscos associados aos projetos de software relevantes (Qu & Wang, 2015), já que o risco é representado pelo potencial de gerar resultado indesejado na forma de incidentes, eventos ou ocorrências (Badurdeen et al. 2014), sendo essencial que sejam conhecidos e mensurados, monitorados e atenuados os seus efeitos negativos (Shao, Feng, & Hu, 2016; Firdose & Rao, 2018; Li et al. 2019; Sundararajan, Bhasi, & Vijayaraghavan, 2019).

O uso de métodos e técnicas matemáticas e estatísticas pode contribuir para uma avaliação mais precisa do nível, grau ou impacto do risco, tornando o gerenciamento de riscos uma postura sólida com base em aspectos metodológicos consistentes que podem ser viáveis para a execução de projetos relevantes com riscos minimizados (Sharma et al. 2013; Sarigiannidis & Chatzoglou, 2014; Sundararajan, Bhasi, & Vijayaraghavan, 2014; Kumar & Yadav, 2015; Rodriguez, Ortega, & Concepcion, 2016; Organ & Stapleton, 2016).

Apesar desse avanço significativo no mundo acadêmico o tratamento de riscos em projetos de software ainda tem potencial para se obter maior precisão, melhorar o tratamento de variabilidade, melhorar a incorporação de fatores críticos de sucesso e melhorar a estrutura de priorização desses Fatores de Risco (FR) para garantir seu gerenciamento.

Para contribuir na melhoria da precisão da medição de riscos em projetos de software, o presente artigo tem como objetivo apresentar um modelo para mensuração de riscos integrando a estrutura Noisy-OR para atribuir pesos e o modelo Bayesiano para calcular a probabilidade de cada nó a partir de uma rede de eventos que influenciam o risco dos projetos, pois de acordo com Albuquerque, Lucena, & Barros (2020) o modelo Bayesiano é uma ferramenta poderosa de estimativa e medição utilizados em probabilidades condicionais (Albuquerque, Lucena, & Barros, 2020).

2. Revisão da literatura

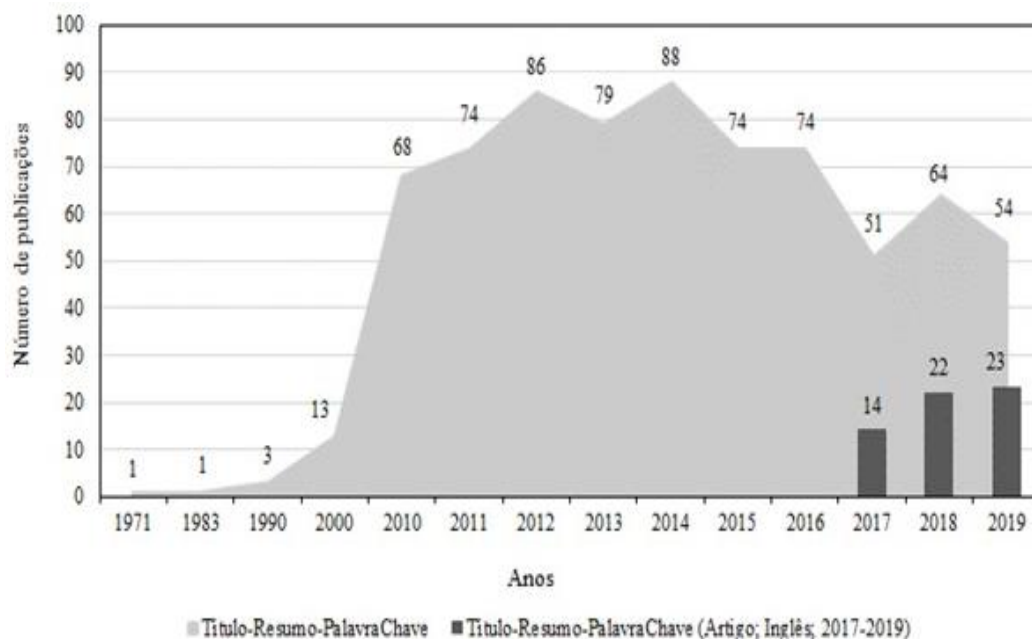
Para alcançar um conjunto de teorias, estudos e pesquisas com relevância na abordagem do assunto e com o objetivo de avançar em direção à fronteira do conhecimento

científico, em nível internacional, o estudo pesquisou nas bases científicas SCOPUS e identificou artigos com similaridade ao objeto tratado, utilizando o termo composto (“risk” e “software project”) como parâmetro de busca nos títulos, palavras-chave ou resumos, com 1.216 documentos identificados e catalogados na pesquisa.

Refinando-se a pesquisa com filtros para documentos ordenados por um número maior de citações, do tipo de artigo, escrito em inglês, período entre 2017 e 2019: (title-abs-key ("risk" e "software project")) e ((limite para (pubyear, 2019) ou limite para (pubyear, 2018) ou limite para (pubyear, 2017)) e ((limite para (doctype, "ar") e (limite para (idioma, "inglês"))), foram identificados 59 documentos considerados como estado da arte, demonstrando que o tema ainda é muito estudado na atualidade.

A Figura 1 mostra as trajetórias das publicações dos 730 documentos (área em cinza claro) e as publicações dos 59 documentos no período 2017-2019 (colunas em cinza escuro).

Figura 1 – Evolução do número de documentos identificados (1971-2019).



Fonte: Autores

Em relação às áreas de estudo, os maiores percentuais de documentos foram associados à Ciência da Computação e à Engenharia. Em relação aos países com os maiores números de publicações tem-se os Estados Unidos, China, Índia e Brasil, locais onde as atividades associadas ao desenvolvimento de software estão em ascensão.

Em relação a afiliação dos autores dos artigos publicados, há uma grande diversificação, com ênfase nos centros de referência mundiais em projetos de software, como

Estados Unidos, China e Brasil.

Os artigos científicos recuperados foram associados a uma ampla gama de periódicos, com alto nível de relevância no contexto internacional, destacando-se os periódicos: *Lecture Notes in Computer Science*, *ACM International Conference* e *Advances in Intelligent Systems and Computing*.

Para este estudo selecionou-se como relevantes 32 artigos científicos, acrescidos de outros documentos, que foram classificados e seus conteúdos avaliados, pois eram associados diretamente ao tema FR em projetos de software e que deram a sustentação teórica a este trabalho.

Riscos em projetos de software

A discussão acadêmica sobre os princípios e práticas de gerenciamento de riscos em projetos de software cresceu significativamente desde a publicação do estudo pioneiro de Boehm (Boehm, 1991), que definiu dois dos principais aspectos estruturais do risco no design de software: avaliação e controle. Neste contexto, estudos recentes abordam várias aplicações específicas sobre o gerenciamento de riscos em projetos de software (Sharma et al., 2013; Sundararajan, Bhasi, & Vijayaraghavan, 2014; Wanderley et al., 2015; Kumar & Yadav, 2015; Jiang, 2015; Song & Jiang, 2016; Khan et al., 2017).

Em seu estudo sobre controle de risco, Sundararajan, Bhasi, & Vijayaraghavan (2014), adicionou uma nova dimensão ao estudo de Boehm (1991), analisando os impactos que as mudanças nas equipes de projeto poderiam provocar no resultado, e apontou maneiras proposicionais de minimizar riscos ao considerar métodos ágeis de design de software. Wanderley et al. (2015), especificaram um modelo que poderia ser usado para medir riscos em vários projetos de software, demonstrando o aumento da qualidade das decisões relacionadas ao gerenciamento desses projetos.

Kumar & Yadav (2015), em uma abordagem inovadora, em concordância com os argumentos de Boehm (1991), propuseram um novo modelo para a mensuração de riscos em projetos de software, com maior variabilidade capturada por métodos estatísticos, incluindo o uso da rede Bayesiana. Para elaborar uma classificação dos tipos de risco, Song & Jiang (2016) contribuíram para a identificação de riscos em projetos de software com a proposição de um modelo que pudesse destacar os tipos de riscos mais importantes em projetos de software.

Por outro lado, a norma ISO/IEC (2009) especifica que o gerenciamento de riscos

envolve os seguintes elementos:

- a) Identificação de Eventos,
- b) Evolução de Eventos,
- c) Classificação de prioridade de risco, seguida pela sincronização e aplicação do custo efetivo dos resultados à depreciação,
- d) Monitoramento de Eventos e,
- e) Controle, envolvendo a probabilidade e/ou impacto de eventos indesejados.

As redes Bayesianas são modelos gráficos probabilísticos que permitem o estudo de problemas nos quais existe dependência entre as variáveis caracterizadas como causa e efeito (Bernardo & Smith, 2000). Métodos e estruturas consolidados podem apoiar a melhoria dos resultados obtidos com a aplicação de redes Bayesianas para o tratamento de probabilidades de risco, entre elas a estrutura Noisy-OR, que é uma estrutura de computação booleana que pode ser utilizada como um modelo para captura não determinística das interações entre causas e efeitos (Pearl, 1988).

Com a abordagem sintonizada de estudos anteriores que enfatizavam os FR e proposição de modelos para mensuração e avaliação de riscos em projetos de software, o presente estudo, em consonância com as proposições iniciais de Boehm (1991) e em linha com o estudo de Kumar & Yadav (2015), apresenta uma proposta para a linearização de um modelo para medir FR associados a projetos de software na busca de contribuir para a melhoria da qualidade da análise dos aspectos associados ao risco em projetos de software.

Fatores de risco em projetos de software

As orientações para o gerenciamento de riscos são fornecidas pelas áreas estratégicas da organização e seu gerenciamento eficaz deve ser aplicado em todos os três níveis (estratégico, programa e projeto), de maneira articulada e que precisam ser integradas aos processos, atividades e rotinas da organização (HM Treasury, 2004). O gerenciamento de riscos deve ser estruturado em níveis, nos quais o nível estratégico lida com decisões estratégicas, o nível do programa lida com decisões que transformam estratégias em ações e o de projeto implementa essas ações.

O Software Engineering Institute (SEI) especifica os FR em programas de software que serão considerados neste estudo como FR em projetos de software, corroborados pelos conceitos contidos no livro, que são: Ambiente, Preparação, Objetivos, Resiliência, Execução e Resultado. Esses FR são detalhados no livro "The Orange Book: Management of Risk -

Principles and Concepts", publicado por HM Treasury em 2004 (Leopoldino & Borenstein, 2011; Kutsch et al. 2013; Liu et al. 2013; Iamandi et al. 2015; Librantz et al. 2016).

Estudos recentes que abordam o gerenciamento de riscos e a incorporação de FR, mostram elementos que podem melhorar a precisão das estimativas e a capacidade preditiva de seus modelos na avaliação e na viabilidade do projeto (Agrawal, Singh, & Sharma, 2016; Janjua, Jaafar, & Lai, 2016; Jenko & Roblek, 2016; Wu, Dai, & Zhu, 2016, & Cao & Zhang, 2016).

Dentre os métodos quantitativos utilizados no tratamento de riscos em projetos, a rede Bayesiana tem sido considerada uma das principais ferramentas de análise (Ykhlef & Algawiaz, 2014; Reed & Knight, 2013; Dutra & Santos, 2014; Islam, Mouratidis, & Weippl, 2014; Kumar & Yadav, 2015; Firdose & Rao, 2016).

Neste estudo, os fatores foram apresentados à especialistas ou gerentes de projeto que determinaram a relação e os graus de influência entre eles. A Figura 2 mostra um grafo orientado onde os nós são os FR de projeto de software (elipses com seus nomes) e as arestas as ligações (links) entre eles.

Figura 2 – FR em projetos de software e seus relacionamentos.



Fonte: Adaptado de Alberts & Dorofee (2009).

Na busca de sustentação teórica para esta pesquisa foram analisados artigos selecionados nas bases acadêmicas que corroboram este trabalho e que foram indicados no estudo de Librantz et al. (2016), que demonstram o real interesse no estudo e avaliação dos seis FR (Figura 1) em projetos de software e listados na Quadro 1.

Quadro 1 – Fatores de Risco em Projetos de Software e autores da literatura.

Fator de Risco	Autores
Ambiente	Kumar & Yadav (2015); Song & Jiang (2016); Jenko & Roblek (2016); Janjua, Jaafar, & Lai (2016)
Preparação	Sharma et al. (2013); Organ & Stapleton (2016)
Objetivos	Dutra & Santos (2014); Bista & Dongol (2015); Agrawal, Singh, & Sharma (2016)
Resiliência	Persson & Schlichter (2015)
Execução	Ykelef & Algawiaz (2013); Sundararajan. Bhasi, & Vijayaraghavan (2014); Herteliu & Despa (2014); Danculescu (2014); Hussain, Mkpojiogu, & Kamal (2016)
Resultado	Penha et al. (2014); Wanderley et al. (2015); Wu, Dai, & Zhu (2016)

Fonte: Autores

3. Metodologia

Este estudo analisa o comportamento de FR com o uso de modelos matemáticos e estatísticos e apresenta um experimento, elaborado para a validação do incremento da qualidade na avaliação de riscos em projeto de software.

Os passos metodológicos aplicados no estudo foram:

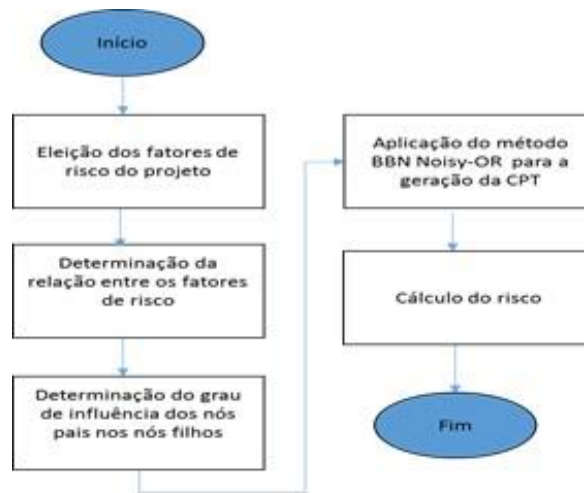
- 1) Revisão da literatura – Busca de literatura que são associadas com os assuntos abordados nesse estudo;
- 2) Identificação dos FR em projetos de software – A partir dos artigos avaliados os FR de projetos de software são identificados, sendo que a opção adotada foi o uso da estrutura desenvolvida pelo SEI;
- 3) Relação de dependência - Identificação do modelo de relacionamento entre os FR;
- 4) Aplicação dos modelos de Bayes e Noisy-OR;
- 5) Cálculo do risco - Após as aplicações dos modelos Bayes e Noisy-OR, são realizados cálculos de fator e peso;
- 6) Simulação – É realizada com as estruturas de relacionamentos e pesos definidos para uma simulação; e
- 7) Análise dos Resultados e das implicações - Com base nos resultados dos

cálculos, a análise dos resultados é realizada e, em seguida, as implicações são explicadas.

Cálculo do risco

O algoritmo do processo proposto para cálculo de riscos está representado na Figura 3. Observa-se neste algoritmo que, após a seleção dos FR, as relações entre esses fatores e os graus de influência necessários para a aplicação da estrutura e geração de Noisy-OR da rede Bayesiana, culmina no cálculo dos respectivos riscos.

Figura 3 – Processo de cálculo de riscos (algoritmo) de projeto de software.



Fonte: Autores

Para realizar o teste do modelo proposto, aplicou-se um experimento empírico contendo dados fornecidos por especialistas ou gerentes em projetos de software. Para atribuir os pesos iniciais dos nós pertencentes à rede, foi utilizada a estrutura Noisy-OR, onde: a) Uma série de cálculos foi aplicada à rede de nós (FR); b) Pesos probabilísticos (valores) foram atribuídos no formato booleano (Verdadeiro ou Falso).

Posteriormente, o princípio do modelo Bayesiano foi aplicado para calcular a ocorrência de cada nó (fator de risco) da rede no projeto. Observa-se no algoritmo mostrado na Figura 3 que, antes de aplicar os modelos Noisy-OR e Bayesiano para calcular os riscos, era necessário eleger os FR, a determinação das relações entre esses fatores e os graus de influência.

Um cálculo sistemático de riscos é baseado em um pequeno conjunto de fatores,

conhecidos como “direcionadores”, que influenciam fortemente o resultado.

Esse conjunto de "direcionadores" pode ser usado para calcular os pontos fortes e fracos do projeto atual e formar a base para a análise de risco subsequente. Após a eleição dos FR, é necessário determinar o grau de influência dos nós pais nos nós filhos.

O Quadro 2 apresenta os valores referentes aos graus de influência que são distribuídos em porcentagem.

Quadro 2 – Graus de influência de nós Pais nos nós Filhos da rede Bayesiana.

Nó Filho	Nó Pai	% de influência	Nó Filho	Nó Pai	% de influência
Ambiente	-	-	Preparação	Objetivos	45,00
Execução	Preparação	35,00	Resiliência	Ambiente	55,00
Execução	Resiliência	10,00	Resiliência	Preparação	25,00
Execução	Ambiente	55,00	Resiliência	Objetivos	20,00
Objetivos	Ambiente	40,00	Resultado	Execução	40,00
Preparação	Ambiente	55,00	Resultado	Ambiente	35,00
			Resultado	Preparação	25,00

Fonte: Autores

Aplicação do modelo bayesiano

Conforme declarado nos estudos de risco de projeto (Ykhlef & Algawiaz, 2014; Kumar & Yadav, 2015), uma rede Bayesiana consiste em um conjunto de variáveis e um conjunto de arestas que ligam as variáveis; as variáveis e arestas formam um gráfico direcionado sem ciclos: para cada variável A que tem como pais B1, ..., Bn, existe uma tabela P (A | B1, ..., Bn).

Esta técnica permite descrever a relação de causa e efeito e, por isso, apresenta resultados animadores no que se refere diagnóstico, previsão e classificação de falhas e análise de riscos (Librantz et al., 2020). Entre as aplicações desta técnica encontram-se desenvolvimento de novos produtos (Chin et al, 2009), diagnóstico de infecções (Sareen, Sood, & Gupta, 2017), identificação e controle de riscos no transporte aeronáutico (Martins & Maturana, 2013). Um dos desafios práticos ao construir modelos RB para suporte à decisão e avaliação de risco é definir a tabelas de probabilidade condicional necessária (CPTs) (Fenton,

Noguchi, & Neil, 2019). A CPT consiste em um conjunto de distribuições de probabilidade indexadas pelas possíveis combinações de estados pais. Por exemplo, considerando que todas as variáveis em um modelo são binárias e uma variável X possui 3 pais, a tabela de probabilidade condicional que corresponde a X terá $2^3 = 8$ possibilidades de distribuições (Zagorecki, & Druzdzal, 2013).

O uso do modelo Noisy-OR fornece condições ajustadas aos modelos com um número grande de variáveis e sem a necessidade de usar muitos valores de probabilidades para a rede de análise (Perreault et al. 2016). O Quadro 3 apresenta, para fins ilustrativos, uma CPT (Conditional Probability Table) gerada pelo cálculo booleano Noisy-OR.

Quadro 3 – CPT gerada pelo cálculo booleano Noisy-OR.

Nó	Valor booleano	Prob. Priori	Nó	Valor booleano	Prob
Ambiente	F	50,00%	Resiliência	FFF	0,00%
Ambiente	T	50,00%	Resiliência	FFT	55,00%
Execução	FFF	0,00%	Resiliência	FTF	25,00%
Execução	FFT	55,00%	Resiliência	FTT	66,25%
Execução	FTF	35,00%	Resiliência	TFF	20,00%
Execução	FTT	70,75%	Resiliência	TFT	64,00%
Execução	TFF	10,00%	Resiliência	TTF	40,00%
Execução	TFT	59,50%	Resiliência	TTT	73,00%
Execução	TTF	41,50%	Resultado	FFF	0,00%
Execução	TTT	73,68%	Resultado	FFT	40,00%
Objetivos	F	0,00%	Resultado	FTF	35,00%
Objetivos	T	40,00%	Resultado	FTT	61,00%
Preparação	FF	0,00%	Resultado	TFF	25,00%
Preparação	FT	55,00%	Resultado	TFT	55,00%
Preparação	TF	45,00%	Resultado	TTF	51,25%
Preparação	TT	75,25%	Resultado	TTT	70,75%

Fonte: Autores

A definição formal da especificação geral do Modelo Bayesiano é apresentada pela

equação 1 utilizada para o cálculo de probabilidades na análise de risco.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Embora o modelo Bayesiano possa ser usado para modelar domínios com o reconhecimento de influências causais, reduzindo o número de probabilidades necessárias para especificar relacionamentos no domínio, a especificação dessas probabilidades necessárias pode se tornar proibitiva no caso de grandes redes (Perreault et al., 2016), e é por isso que a estrutura Noisy-OR aparece, neste estudo, como um componente relevante.

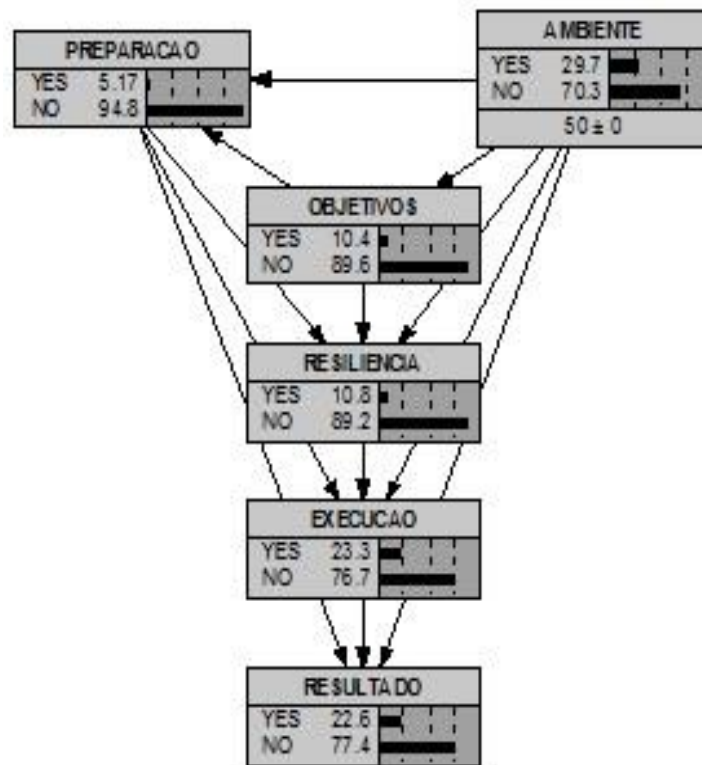
A realização do experimento empírico se deu em dois passos sequenciais: o cálculo dos riscos e a simulação do modelo para o comportamento da avaliação de riscos.

3. Resultados e Discussão

Para os testes relacionados ao modelo proposto, foram realizados os cálculos de risco para cada um dos seis fatores (Figura 2), considerando-se os modelos Bayesiano e Noisy-OR.

A Figura 4 apresenta a rede Bayesiana usada para cada fator de risco em projetos de software. A implementação foi realizada com o suporte do software Netica para MS-Windows (Norsys, 2017), e a tabela de pesos inicial da rede foi carregada e, em seguida, foram calculados os pesos de cada fator de risco do projeto.

Figura 4 – Rede Bayesiana usada para modelar riscos em projetos de software.



Fonte: Autores (ferramenta Netica).

Simulação

Baseado no fator "Ambiente", que tem influência em todos os outros fatores, a variação dos pesos foi realizada e o comportamento da rede foi analisado após essas variações. A variação foi causada pelo valor booleano "Yes" do nó "Ambiente" entre intervalos de 10 a 100 por cento. O Quadro 4 apresenta o resultado da simulação.

Quadro 4 – Simulação da variação do valor Booleano Yes para o nó Ambiente.

Nó	Variação de Valor Booleano "Yes" de 10% a 100%									
	10	20	30	40	50	60	70	80	90	100
Ambiente	10	20	30	40	50	60	70	80	90	100
Preparação	35,4	37,8	40,2	42,6	45,0	47,5	49,9	52,3	54,7	57,1
Objetivos	58,0	56,0	54,0	52,0	50,0	48,0	46,0	44,0	42,0	40,0
Resiliência	37,6	38,5	39,4	40,3	41,2	42,1	43,0	43,9	44,8	45,7
Execução	29,3	31,0	32,8	34,5	36,2	38,0	39,7	41,5	43,2	44,9
Resultado	22,9	25,9	28,9	31,9	34,9	37,9	40,9	43,9	46,9	49,9

Fonte: Autores

Após a simulação da variação do valor booleano "Yes" do fator "Ambiente", uma análise de cenário foi realizada marcando o booleano "Yes" aos fatores (nós): "Ambiente", "Preparação", "Objetivos", "Resiliência" e "Execução", e então foi verificado quanto foi mudado nos valores Booleanos do fator "Resultado". O Quadro 5 apresenta o resultado dessa análise.

Quadro 5 – Análise comparativa da influência no fator "Resultado" na relação com os outros fatores da rede Bayesiana para projetos de software.

(% Variação de cada fator para 100% comparado ao fator "Resultado")	
Variação do Valor Booleano "Yes" para 100%	
Fator=100%	Impacto no Fator "Resultado" (%)
Ambiente	49,90
Preparação	59,10
Objetivos	44,60
Resiliência	52,30
Execução	63,20

Fonte: Autores

Baseado nesta informação, é possível verificar que o fator "Execução" teve grande influência no fator "Resultado", com 63.20%, e o fator "Objetivos" foi o fator com menor influência com 44.60%.

Implicações gerenciais

Conforme apresentado neste estudo, o gerenciamento de riscos consiste em uma alternativa destinada a enfrentar o problema dos ambientes cada vez mais complexos das empresas e que necessitam de tomadas de decisões gerenciais de qualidade (Herrmann, 2015). Por outro lado, o gerenciamento de riscos em projetos de software envolve a tomada de decisões, sob incerteza, mas com base em dados sistemáticos e objetivos, a fim de melhorar as chances de sucesso nas compensações (Ye et al. 2014; Kumar, Latha, & Praynlin, 2014).

Nesse sentido, o uso de técnicas matemáticas e estatísticas destinadas a apoiar o gerenciamento de riscos, com abordagem de modelos como a rede Bayesiana e a estrutura Noisy-OR, e considerando-se o modelo especificado pelo SEI para projetos de software, pode contribuir para melhorar a qualidade das decisões dos gestores das empresas. E dessa forma influencia positivamente nos aspectos econômicos importantes, como produtividade e rentabilidade (Ye et al. 2014; Shankari & Thirumalaiselvi, 2014).

A melhoria na precisão preditiva da probabilidade de ocorrência de falhas fornece uma base sólida para o gerenciamento eficaz dos riscos associados ao projeto (Qu & Yang, 2016), e simplifica o tratamento de questões relacionadas ao conjunto de fatores usados na avaliação de riscos (Perreault et al., 2016), bem como, permite focar a atenção em possíveis pontos de falha no processo de desenvolvimento do projeto e elaborações preventivas (Nasar, Johri, & Chanda, 2014; Organ & Stapleton, 2016).

O gerenciamento de riscos, com ênfase em fatores críticos e usando as técnicas apresentadas neste estudo, tende a melhorar o processo de gerenciamento do próprio projeto, o que contribui para o lançamento de produtos de software de alta qualidade, com prazos, custos acordados, e favorecendo o sucesso da organização (Rana et al. 2016; Chevers & Grant, 2017). Por outro lado, os sistemas de informação resultantes desses projetos, se realizados de acordo com as especificações confiáveis, favorecem o fluxo de informações ágeis e as necessidades de satisfação funcional que são frequentemente vitais para aumentar a capacidade competitiva da organização (Kumar, Latha, & Praynlin, 2014).

4. Considerações Finais

Neste estudo, uma abordagem BBN (Bayesian Belief Networks) foi proposta para a estimativa de risco em projetos de software. O modelo utilizado para a realização do experimento empírico teve, como base estrutural as especificações propostas pelo SEI e contou com a aplicação da abordagem de rede Bayesiana associada à estrutura Noisy-OR.

Esta abordagem combinada e apresentada neste trabalho, forneceu dados probabilísticos sobre os FR de projeto de software, que pode apoiar o processo de tomada de decisão de forma quantitativa e corroborada pela literatura acadêmica.

O processo proposto amplia a capacidade e a qualidade do cálculo de risco dos modelos existentes, aumentando as melhorias nas estimativas a partir da conjugação da rede Bayesiana e da estrutura Noisy-OR. Isso permite que os responsáveis pelo gerenciamento de projetos de software sejam mais seguros no desempenho de suas funções de avaliação de riscos em projetos, expandindo as alternativas, minimizando as ameaças e aprimorando seus planos de contingência.

Para futuras pesquisas, o impacto de cada fator pode ser incluído com base nesse modelo.

Referências

Agrawal, R., Singh, D., & Sharma, A. (2016). Prioritizing and optimizing risk factors in agile software development. In: 2016 Ninth international Conference on Contemporary Computing (IC3). doi:10.1109/IC3.2016.7880232

Alberts, C., & Dorofee, A. A. (2009). Framework for Categorizing Key Drivers of Risk. Software Engineering Institute (CMU/SEI-2009-TR-007), Carnegie Mellon University, Pittsburgh. Recuperado de: https://resources.sei.cmu.edu/asset_files/TechnicalReport/2009.

Albuquerque, M. A. de, Lucena, S. L. L. de, & Barros, K. N. N. de O. (2020). Comparison of classic and Bayesian model for data on perinatal deaths at ISEA, Campina Grande-PB. Research, Society and Development, 9(8), e464985477. Doi:10.33448/rsd-v9i8.5477

Badurdeen, F., Shuaib, M., Wijekoon, K., Brown, A., Faulkne, W., Amundson, J., Jawahir, I. S., Goldsby, T. J., Iyengar, D., & Boden, B. (2014). Quantitative modeling and analysis of

supply chain risks using bayesian theory. *Journal of Manufacturing Technology Management*, 25(5), 631-654. Doi:10.1108/JMTM-10-2012-0097

Bernardo, J. M., & Smith, A. F. M. (2000). *Bayesian theory*. John Wiley & Sons, Ltd. ISBN 0471924164.

Bista, R., & Dongol, D. A. (2015). Model for software risk management. *Technia - International Journal of Computing Science and Communication Technologies*, 7(2), 1028-1035. ISSN:0974-3375.

Boehm, B. W. (1991). Software risk management: Principles and practices. *Journals & Magazines, IEEE Software*, 8(1), 32-41. Recuperado de: <https://ieeexplore.ieee.org/document/62930>.

Cao, W., & Zhang, X. (2016). Supply chain risk assessment based on support vector machine. *Revista Ibérica de Sistemas e Tecnologias de Informação (RISTI)*, E5, 310-322. ISSN:1646-9895.

Chevers, D. A., & Grant, G. (2017). Developer's views on information systems quality and success in canadian software development firms. *Journal of Information Systems and Technology Management: JISTEM*, 14(1), 3-20. doi:10.4301/S1807-17752017000100001

Chin, K-S., Tang, D-W., Yang, J-B., Wong, S. Y., & Wang, H. (2009). Assessing New Product Development Project Risk by Bayesian Network with a Systematic Probability Generation Methodology. *Expert Systems with Applications*, 36(6), 9879-9890. doi:10.1016/j.eswa.2009.02.019

Danciulescu, A. (2014). Risk management an important tool in ict sme's in Romania. *Journal of Information Systems & Operations Management*, p. 1-13. Recuperado de: <https://www.academia.edu/9042447>.

Dutra, E., & Santos, G. (2014). Risks in initiatives to improve software processes: an investigation in the Brazilian context. In: *XII Workshop on Thesis and Dissertation in Software Quality*, 59-64. Recuperado de: sbqs.sbc.org.br/2014.

Fenton, N. E., Noguchi, T., & Neil, M. (2019). An Extension to the Noisy-OR Function to Resolve the 'Explaining Away' Deficiency for Practical Bayesian Network Problems. *IEEE transactions on knowledge and data engineering*, 31(1), 2019, 2441-2445. Recuperado de: <https://www.ieeexplore.ieee.org/document/8607084>.

Firdose, S., & Rao, L. M. (2016). 3LRM-3 Layer risk mitigation modelling of ICT software development projects. *International Journal of Electrical and Computer Engineering (IJECE)*, 6(1), 349-356. doi:10.1159/ijece.v6i1.9026

Firdose, S., & Rao, L. M. (2018). PORM: Predictive optimization of risk management to control uncertainty problems in software engineering. *International Journal of Electrical and Computer Engineering (IJECE)*, 8(6), 4735-4744. doi:10.1159/ijece.v8i6.pp4735-4744

Herrmann, J. W. (2015). *Engineering decision making and risk management*. New Jersey: John Wiley & Sons, Ltd. ISBN: 978-1-118-91933-0.

Herteliu, E., & Despa, M. L. (2014). Risk management in citizen oriented innovative software development projects. *Journal of Information Systems & Operations Management*, 8, 1-18.

HM Treasury. (2004). *The Orange Book, Management of Risk - Principles and Concepts*. UK, London: HM Treasury. Recuperado de: <https://erm.ncsu.edu/library/article/orange-book-principles>.

Hussain, A., Mkpojiogu, E., & Kamal, F. M. (2016). The role of requirements in the success or failure of software projects. *International Review of Management and Marketing*, 6(2016), 306-311. Recuperado de: <http://www.econjournal.com>

Iamandi, O., Popescu, S., Dragomir, M., & Morariu, C. (2015). A critical analysis of project management models and its potential risks in software development. *Calitatea*, 16(149), 55-61.

Islam, S., Mouratidis, H., & Weippi, E. (2014). An empirical study on the implementation and evaluation of a goal driven software development risk management model. *Information and*

Software Technology. 56(2), 117-133. doi:10.1016/j.infsof.2013.06.003

ISO/IEC. (2009). International standards organization standard on risk management. ISO 31000: Risk Management – Risk Assessment Techniques. Recuperado de: <https://www.iso.org/iso-31000-risk-management.html>.

Janjua, U. I., Jaafar, J., & Lai, F. W. (2016). Expert's opinions on software project effective risk management. In: IEEE Software 2016 3rd International Conference on Computer and Information Sciences (ICCOINS). Recuperado de: <https://ieeexplore.ieee.org>.

Jenko, A., & Roblek, M. (2016). A primary human critical success factors model for the ERP system implementation. Organizacija, 49(3), 145-160. doi:10.1515/orga-2016-0014

Jiang, R. (2015). A novel risk metric for staff turnover in a software project based on information entropy. Entropy, 17(5), 2834-2852. doi:10.3390/e17052834

Khan, A. A., Keung, J., Niazi, M., Hussain, S., & Ahmad, A. (2017). Systematic literature review and empirical investigation of barriers to process improvement in global software development: Client–vendor perspective. Information and Software Technology, 87, 180-205. doi:10.1016/j.infso.2017.03.006

Kumar, C., & Yadav, D. K. (2015). A probabilistic software risk assessment and estimation model for software projects. In: Procedia Computer Science: Eleventh International Multi-Conference on Information Processing-2015, 54, 353-361. doi:10.1016/j.procs.2015.06.041

Kumar, M., Latha, P., & Praynlin, E. (2014). Software effort estimation using genetic algorithm. International Journal on Information Sciences and Computing, 8(1), 9-16. doi:10.18000/ijisac.50142

Kutsch, E., Denyer, D., Hall, M., & Lee-kelley, E. (2013). Does risk matter? disengagement from risk management practices in information systems projects. European Journal of Information Systems, 22, 637-649. doi:10.1057/ejis.2012.6

Leopoldino, C. B., & Borenstein, D. (2011). Componentes de risco para a gestão de projetos

de software. READ - Revista Eletrônica de Administração (online), setembro-dezembro. doi:10.1590/S1413-23112011000300003

Li, X., Jiang, Q., Hsu, M. K., & Chen, Q. (2019). Support or risk? Software project risk assessment model based on rough set theory and backpropagation neural network. *Sustainability*, 11, 1-12. Doi:10.3390/su11174513

Librantz, A. F. H., Santos, F. C. R., Dias, C. G., Cunha, A. C. A., Costa, I., & Spinola M. M. (2016). AHP modelling and sensitivity analysis for evaluating the criticality of software programs. In: *Advances in Production Management Systems. Initiatives for a Sustainable World. APMS 2016, IFIP Advances in Information and Communication Technology*, Springer, Cham, 488. doi:10.1007/978-3-319-51133-7_30

Librantz, A. F. H., Costa, I., Spinola, M. M., Oliveira Neto, G., & Zerbinati, L. (2020). Risk assessment in software supply chains using the Bayesian method. *International Journal of Production Research*. doi: 10.1080/00207543.2020.1825860

Liu, J. Y., Yang, M., Klein, G., & Chen, H. (2013). Reducing user-related risks with user-developer partnering. *Journal of Computer Information Systems*, 54(1), 66-74. doi:10.1080/08874417.2013.11645672

Martins, M. R.; Maturana, M. C. (2013). Application of Bayesian Belief network to the human reliability analysis of na oil tanker operation focusing on collision accidents. *Reliability Engineering & System Safety*, 110, 89-109. doi:10.1016/j.res.2012.09.008

Nasar, M., Johri, P., & Chanda, U. (2014). Software testing resource allocation and release time problem. *International Journal of Modern Education and Computer Science*, 6(2), 48-55. doi: 10.5815/ijmecs.2014.02.07

Norsys. (2017). Netica for MS Windows. Norsys Software Corporation. Recuperado de: <https://www.norsys.com/download.html>.

Organ, J., & Stapleton, L. (2016). Technologist engagement with risk management practices during systems development? approaches, effectiveness and challenges. *AI & Society*, 31(3),

347-359. doi:10.1007/s00146-015-0597-4

Pearl, J. (1998). Probabilistic reasoning in intelligent systems: Networks of plausible inference. Palo Alto, Morgan Kaufmann Publishers.

Penha, R., Kniess, C. T., Bergman, D. R., & Biancolino, C. A. (2014). Emprego de técnicas de gerenciamento de riscos técnicos em uma empresa de desenvolvimento de softwares. *Revista Gestão & Tecnologia*. jan-abr, 14(1), 151-173. Recuperado de: <http://repositorio.uninove.br/xmlui/bitstream/handle/123456789/365/287-528-1-RV.pdf?sequence=1>.

Perreault, L., Strasser S., Thornton M., & Sheppard, J. W. (2016). A Noisy-OR model for continuous time Bayesian networks. In: Proceedings of the Twenty-Ninth International – Florida Artificial Intelligence Research Society Conference, 668-673. Recuperado de: <https://www.cs.montana.edu/sheppard/pubs/flairs-2016.pdf>

Persson, S., & Schlichter, B. R. (2015). Managing risk areas in software development offshoring:A CMMI Level 5 Case. *Journal of Information Technology Theory and Application*, 16(1), 5-24. Recuperado de: <https://aisel.aisnet.org/jitta/vol16/iss1/2>.

Qu, Y., & Wang. H. (2015). Analysis on chaotic characteristics of software project risk system. *International Journal of Security and Its Applications*, 9(2), 49-60. doi:10.14257/ijisia.2015.9.2.06

Qu, Y., & Yang, T. (2016). Research on occurrence frequency of IT projects risk based on fuzzy influence diagram. In: The Institute of Electrical and Electronics Engineers, Inc. (ICMLC) - Conference Proceedings. Piscataway, 166-171. doi: 10.1109/ICMLC.2016.7860895

Rana, R., Staron, M., Berger, C., Hansson, J., Nilsson, M., & Meding, W. (2016). Analyzing defect inflow distribution and applying bayesian inference method for software defect prediction in large software projects. *Journal of Systems and Software*, 117, 229-244. doi: 10.1016/j.jss.2016.02.015

Reed, A. H., & Knight, L. V. (2013). Project duration and risk factors on virtual projects. *The Journal of Computer Information Systems*, 54(1), 75-83. Doi:10.1080/08874417.2013.11645673

Rodriguez, A., Ortega, F., & Concepcion, R. (2016). A method for the evaluation of risk in IT projects. *Expert Systems with Applications*, 45, 273-285. doi:10.1016/j.eswa.2015.09.056

Sareen, S., Sood, S. K., & Gupta, S. K. (2017). Secure internet of things-based cloud framework to control zika virus outbreak. *International Journal of Technology Assessment in Health Care*. 33. doi:10.1017/S0266462317000113

Sarigiannidis, L., & Chatzoglou, P. D. (2014). Quality vs risk: An investigation of their relationship in software development projects. *International Journal of Project Management*, 32(6), 1073-1082. doi:10.1016/j.ijproman.2013.11.001

Shankari, H. K., & Thirumalaiselvi, R. (2014). Guidelines to improve software engineering process using artificial intelligence techniques. *I-Manager's Journal on Software Engineering*, 8(4), 33-43. doi:10.26634/jse.8.4.3050

Shao, Z., Feng, Y., & Hu, Q. (2016). Effectiveness of top management support in enterprise systems success: a contingency perspective of fit between leadership style and system life cycle. *European Journal of Information Systems*. 25(2), 131-153. doi: 10.1057/ejis.2015.6

Sharma, A., Basora, D., Chhillar, N., & Yadav, D. (2013). A comprehensive study of software risk management. *International Journal of Advanced Research in Computer Science*, 4(10), 61-67. Recuperado de: www.ijarcs.info.

Song, H., & Jiang, J. (2016). Risks identification in embedded software development: evidence from MVBC project survey. *Procedia Computer Science*, 91, 798-806. doi:10.1016/j.procs.2016.07.082

Sundararajan, S., Bhasi, M., & Vijayaraghavan, P. K. (2014). Case study on risk management practice in large offshore-outsourced agile software projects. *IET Software*, 8(6), 245-257. doi:10.1049/iet-sen.2013.0190

Sundararajan, S., Bhasi, M., & Vijayaraghavan, P. K. (2019). Variation of risk profile across software life cycle in IS outsourcing. *Software Quality Journal*, 27, 1563-1582. doi: 10.1007/s11219-019-09451-8

Wanderley, M., Menezes, J., Gusmão, C., & Lima, F. Proposal of risk management metrics for multiple project software development. *Procedia Computer Science*, 64, 1001-1009. doi:10.1016/j.procs.2015.08.619

Wu, D., Dai, Q., & Zhu, X. (2016). Measuring the Effect of Project Risks Based on Shapley Value for Project Risk Response. *Procedia Computer Science: Information Technology and Quantitative Management (ITQM-2016)*, 91, 774-778. doi:10.1016/j.procs.2016.07.076

Ye, Y., Jankovic, M., Kremer, G. E., & Bocquet, J. C. (2014). Managing uncertainty in potential supplier identification. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 28, 339-351. doi:10.1017/S0890060414000511

Ykhlef, M., & Algawiaz, D. (2014). A new strategic risk reduction. In: *IEEE Software 5th International Conference on Computer Science and Information Technology (CSIT)*, 179-183. Doi:10.1080/18756891.2014.960236

Zagorecki, A.; Druzdel, M. (2013). Knowledge Engineering for Bayesian Networks: How Common Are Noisy-MAX Distributions in Practice? *IEEE transactions on systems man cybernetics-systems*. 43(1), 186-195. Recuperado de: <https://www.pitt.edu/~druzdel/ftp/zagorecki13.pdf>.

Porcentagem de contribuição de cada autor no manuscrito

Adalberto Ramos Cassia – 25%

André Felipe Librantz – 25%

Ivanir Costa – 25%

Mauro de Mesquita Spinola - 15%

Erika Midori Kinjo – 10%