JAIST Repository
https://dspace.jaist.ac.jp/

Title	リスクマネジメントにおける機械学習と知識創造の統合アプローチ―機械参加型（machine-in-the-loop）プロセスの提案―
Author(s)	森，俊樹
Citation	
Issue Date	2020-06
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/16726
Rights	
Description	指導教員：内平　直志，先端科学技術研究科，博士

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

博 士 論 文

リスクマネジメントにおける
機械学習と知識創造の統合アプローチ

―機械参加型（machine-in-the-loop）プロセスの提案―

森 俊樹

主指導教員　内平 直志

北陸先端科学技術大学院大学

先端科学技術研究科［知識科学］

令和 2 年 6 月

Abstract

In a knowledge society where knowledge workers become core competence in economy, business and industrial environment has been drastically changing with the increasing diversification of customer needs in global markets and the rapid technological changes of the Internet, machine learning, and artificial intelligence (AI). In order to establish a sustainable competitive advantage in such a situation, manufacturers are urged to build the dynamic capability to correspond to unexpected changes by further enhancing project risk management. However, despite of the existence of the standardized risk management process and methods, it is observed that managers often struggle with the effective application of project risk management in practice.

In this study, we assume that the essential challenge of project risk management is "the difficulty of making decisions including trade-offs at the right time for various uncertain events and conditions within limited time, cost, and resources." We provide a new explanation of the difficulty from the point of view of transaction cost theory and prospect theory. Then, we propose "machine-in-the-loop" risk management framework, which uses complementary relationship between human and machine learning models.

Furthermore, we examine a machine learning technique that may support the proposed framework. In general, there is a trade-off relationship that a simple machine learning model with higher interpretability has lower prediction accuracy, while a complex machine learning model with higher prediction accuracy has lower interpretability. In this study, we propose a new machine learning technique called SNB (superposed naive Bayes), which uses a two-step approach, i.e., firstly builds a naive Bayes ensemble via stochastic boosting, and then transforms it into a simple naive Bayes model by linear approximation. The proposed model can provide an effective way for balancing the trade-off between accuracy and interpretability.

Keywords: Project Risk Management, Machine Learning, Knowledge Management, Transaction Cost, Cognitive Bias

概要

本格的な知識社会、すなわち「知識が中核の資源となり、知識労働者が中核の働き手となった社会」を迎えて、企業を取り巻く環境の変化はますます激しくなっている。こうした状況の中で企業が持続的競争優位を確立するためには、変化に対応して柔軟に適応していく能力の構築が必要であり、プロジェクト・リスクマネジメントの一層の強化が不可欠である。しかしながら、リスクマネジメントのプロセスや手法は概ね標準化されているにも関わらず、その効果的な実践や定着化は必ずしも容易ではないという実態がある。従来、リスクマネジメントへのナレッジマネジメントの応用や人工知能（AI）・機械学習の応用などがそれぞれ個別に論じられてきたが、リスクマネジメントの本質的な課題に対して十分に踏み込めているとは言い難い。

本研究では、プロジェクト・リスクマネジメントの本質的な課題に対して、取引コスト理論およびプロスペクト理論の観点から考察を行った。不確実な状況下では、トレードオフを伴う意思決定の難しさに加えて、取引コストや認知バイアスの影響が作用することによる合理的な判断からの偏りが生じて、リスクマネジメントの形骸化などのさまざまな実践上の困難性につながっていると考えられる。その具体的な解決アプローチとして、人間の気づきと機械学習の相補性に着目した機械参加型（machine-in-the-loop）リスクマネジメントを提案する。

Machine-in-the-loop の実現においては、機械学習モデルの予測・推定結果を人間が解釈して意思決定に反映させる必要があり、モデルの解釈可能性が重要となる。既存の機械学習モデルにおいて、解釈可能性に優れた単純なモデルは予測精度が低くなる一方、予測精度が高い複雑なモデルは解釈可能性が低下するというトレードオフ関係が存在した。そこで、本研究では、新しい機械学習モデル SNB（superposed naive Bayes）を提案する。SNB は、ナイーブベイズの集団学習モデルを線形近似して単一のナイーブベイズに逆変換することにより、予測精度と解釈可能性の両立を実現している。

リスクマネジメントの本質的課題に対する理論的考察、その解決策としての機械参加型（machine-in-the-loop）リスクマネジメントの提案、予測精度と解釈可能性を両立した新しい機械学習モデル（SNB）の構築において、本研究の新規性がある。

目次

第 1 章 序章 11
1.1. 研究の背景 11
1.2. 研究の目的とリサーチクエスチョン 12
1.3. 研究の方法 14
1.4. 用語の定義 16
1.5. 論文の構成 20
第 2 章 先行研究の検討 23
2.1. はじめに 23
2.2. プロジェクトマネジメントとリスクマネジメント 24
2.2.1. リスクマネジメントの難しさ 27
2.3. ナレッジマネジメントと知識創造 30
2.3.1. リスクマネジメントへのナレッジマネジメントの適用 31
2.4. 人工知能（AI）と機械学習 33
2.4.1. リスクマネジメントへの機械学習の適用 34
2.5. 機械学習とナレッジマネジメントの統合 36
2.6. 機械学習の解釈可能性 39
2.7. 本研究の位置付け 43
第 3 章 リスクマネジメントの現状と課題 45
3.1. はじめに 45
3.2. 製品開発組織におけるリスクマネジメント 45
3.3. インタビュー目的と方法 48
3.4. 帰納的テーマティック・アナリシス法による分析 52
3.5. まとめ 65
第 4 章 リスクマネジメントの困難性の理論的考察 67
4.1. はじめに 67
4.2. リスクマネジメントの本質的な課題 67
4.3. 取引コスト理論による考察 70
4.4. プロスペクト理論による考察 71

4.5. TA のハイブリッドアプローチによる仮説の妥当性検証 75
4.6. まとめ 78
第 5 章 機械参加型リスクマネジメントの提案と評価 81
5.1. はじめに 81
5.2. 機械参加型（machine-in-the-loop）意思決定プロセス 81
5.3. 機械参加型リスクマネジメントの提案 84
5.4. 機械参加型リスクマネジメントの継続的改善 94
5.5. 演繹的 TA による提案アプローチ適用部門の評価 98
5.6. まとめ 104
第 6 章 予測精度と解釈可能性を両立した機械学習モデルの提案と評価 107
6.1. はじめに 107
6.2. 予測精度と解釈可能性の両立 107
6.3. 高精度かつ解釈容易な機械学習モデルの構築 110
6.3.1. ナイーブベイズ分類器 110
6.3.2. TAN（tree augmented naive Bayes） 111
6.3.3. ナイーブベイズの集団学習モデル 113
6.3.4. SNB（superposed naive Bayes） 115
6.4. 実験・評価方法の概要 119
6.4.1. 評価対象の機械学習モデル 119
6.4.2. データセットの概要 122
6.4.3. 予測精度の評価方法 125
6.4.4. 解釈可能性の評価方法 127
6.5. 実験・評価結果 128
6.5.1. 予測精度の評価結果 128
6.5.2. 解釈可能性の評価結果 131
6.5.3. 予測精度と解釈可能性のトレードオフ分析 134
6.6. まとめ 135
第 7 章 考察 137
7.1. 機械参加型リスクマネジメントの導入と展開 137
7.2. 未知のリスクへの対応 139

第 8 章 結論 141
8.1. リサーチクエスチョンへの回答 141
8.2. 理論的含意 143
8.3. 実務的含意 144
8.4. 本研究の限界 145
8.5. 将来研究への示唆 146
参考文献 149
付録 161
A1：リスクマネジメントの課題に関するインタビュー調査 161
A2：提案アプローチの効果に関するインタビュー調査 173
A3： RLR、SNB、RF+PDP の解釈可能性の比較分析 177
謝辞 187
研究業績リスト 189

図目次

図 1.1：本博士論文の研究ストラテジー 15
図 1.2：論文構成と研究ストラテジーとの対応関係 22
図 2.1：先行研究の検討範囲 23
図 2.2：不確実性のライフサイクル 29
図 2.3：組織的知識創造プロセス（SECI モデル） 31
図 2.4：モデルと効用に基づくエージェント 33
図 2.5：ナイーブベイズによるプロジェクト異常予測モデル 35
図 2.6：機械学習モデルによる意思決定支援と組織学習 38
図 2.7：知識創造とデータマイニングの 2 サイクルモデル 39
図 2.8：machine-in-the-loop 文書作成支援システム 42
図 3.1：製品開発組織におけるリスクマネジメント・プロセス 46
図 3.2：質問項目の変遷 51
図 4.1：リスクマネジメントの単純化した意思決定モデル 69
図 4.2：プロスペクト理論の価値関数 72
図 4.3：二重プロセス理論に基づく意思決定モデル 73
図 4.4：プロジェクト・リスクマネジメントの本質的な課題 75
図 5.1：機械参加型（machine-in-the-loop）意思決定プロセス 83
図 5.2：機械参加型（machine-in-the-loop）リスクマネジメント 85
図 5.3：ナイーブベイズ予測モデルの例 87
図 5.4：ナイーブベイズ予測モデルの工程別 ROC 曲線 88
図 5.5：進行中プロジェクトの工程別リスク推移（途中経過） 90
図 5.6：進行中プロジェクトの工程別リスク推移（最終結果） 92
図 5.7：ナイーブベイズ予測モデルにおける説明変数の重要度 93
図 5.8：プロジェクト間の類似度のクラスター分析 94
図 5.9：プロジェクト側と機械学習側の知識更新の 2 サイクル構造 95
図 5.10：機械参加型リスクマネジメントの適用効果の仮説 98
図 6.1：予測精度と解釈可能性を両立する方法 108
図 6.2：さまざまなベイズ分類器の構造 112

図 6.3：TAN からナイーブベイズへの変換 113
図 6.4：ナイーブベイズ・アンサンブル 115
図 6.5：WoE の重ね合わせによる SNB の生成 116
図 6.6：ナイーブベイズの WoE と SNB の WoE の比較 118
図 6.7：実験におけるモデル構築と性能評価の概要 125
図 6.8：Scott-Knott 検定の二重適用によるランク付けアプローチ 126
図 6.9：Reversed fractional rank（RFR）の計算例 127
図 6.10： 2 回目の Scott-Knott 検定の結果 131
図 6.11：予測精度と解釈可能性の比較 134

表目次

表 1.1：企業の世界時価総額ランキングの推移 12
表 2.1：リスクマネジメント・プロセスの比較 26
表 2.2：プロジェクトリスクのアンケート調査項目 34
表 3.1：インタビュー（第 1 回）の対象者 49
表 3.2：インタビュー（第 1 回）の質問項目 50
表 3.3：質問項目への Yes/No 回答の結果 50
表 3.4：リスクマネジメントに対する実務者の課題意識（その 1） 54
表 3.5：リスクマネジメントに対する実務者の課題意識（その 2） 55
表 3.6：抽出したカテゴリーとプロセスの対応付け 65
表 4.1：リスクマネジメントの実践上の難しさとその分類 77
表 5.1：人間と機械学習の相補関係 81
表 5.2：例題のプロジェクトデータに含まれる変数 86
表 5.3：失敗プロジェクト比率（実績）の工程別推移 89
表 5.4：失敗確率トップ 10 プロジェクトのランキング推移 90
表 5.5：混同行列による予測結果のサマリー 92
表 5.6：インタビュー（第 2 回）の対象者 98
表 5.7：インタビュー（第 2 回）の質問項目 99
表 5.8：コードの一覧、および、各切片データとの対応関係 101
表 6.1：評価対象の機械学習モデル 121
表 6.2：NASA MDP データセットの概要 123
表 6.3：JIT データセットの概要 124
表 6.4：５×５分割交差検証による AUC の平均と標準偏差 129
表 6.5：１回目の Scott-Knott 検定の結果 130
表 6.6：解釈可能性の評価結果 133

第1章 序章

1.1. 研究の背景

ポスト資本主義社会として“知識社会”の到来が予言されてから長い年月が経ち、現代が知識社会のまっただ中にあることは疑う余地のない事実であろう。知識社会とは「知識が中核の資源となり、知識労働者が中核の働き手となった社会」であり、Drucker（2002=2002）によると以下の 3 つの特質をもつ。

- 知識は資金よりも容易に移動するがゆえに、いかなる境界もない社会となる。
- 万人に教育の機会が与えられるがゆえに、上方への移動が自由な社会となる。
- 万人が生産手段としての知識を手に入れ、しかも万人が勝てるわけではないがゆえに、成功と失敗の並存する社会となる。

すなわち、知識社会は、組織にとっても個人にとっても高度に競争的な社会となる。

知識社会への移行に伴い、グローバル化による市場環境の変化、顧客ニーズの多様化、デジタル化などによる急激な技術変化、政治状況や社会状況の変化など、企業を取り巻く環境の変化はますます激しくなり、不確実性を増している。特に、近年のインターネットの爆発的な普及と機械学習・人工知能（AI: artificial intelligence）の技術の急速な進化は、知識社会への転換をますます加速し、変化のスピードを早めている。表 1.1 は、企業の世界時価総額ランキングの推移である。この表から、環境の変化に伴い上位の企業がめまぐるしく入れ替わってきたことが読み取れる。

こうした状況の中で企業が持続的競争優位を確立するためには、変化に対応して柔軟に適応していく能力の構築が必要であり、戦略や事業運営におけるプロジェクトマネジメントの一層の強化が不可欠である。しかしながら、変化の激しい環境の中でプロジェクトを計画通りに進めることはますます困難になっている。IT プロジェクトの実態に関してアンケート調査を実施したところ、1238 件中の 47.2%が“失敗”[1]だった（西村ほか 2018）。また、プロジェクトマネジメントの成熟度に関する調査によると、さまざまな領域の中でリスクマネジメントの成熟度が最も低く、全体のボトルネックになっていることがわかった（Ibbs and Kwak 2000; Grant and Pennypacker 2006）。

---

[1] 「スケジュール」「コスト」「満足度」の内、1 つでも条件を満たさないプロジェクトを“失敗”と定義した.

表 1.1：企業の世界時価総額ランキングの推移[2]

順位	1995年	2000年	2005年
1	エスコム	ゼネラル・エレクトリック	ゼネラル・エレクトリック
2	NTT	エクソンモービル	エクソンモービル
3	ゼネラル・エレクトリック	ファイザー	マイクロソフト
4	AT&T	シスコシステムズ	シティグループ
5	エクソンモービル	シティグループ	BP
6	コカ・コーラ	ウォルマート・ストアーズ	ロイヤル・ダッチ・シェル
7	メルク	ボーダフォングループ	プロクター&ギャンブル
8	トヨタ自動車	マイクロソフト	ウォルマート・ストアーズ
9	ロシュ・ホールディングス	AIG	トヨタ自動車
10	アルトリア・グループ	メルク	バンク・オブ・アメリカ

順位	2010年	2015年	2019年（現在）
1	エクソンモービル	アップル	アップル
2	中国石油天然気	アルファベット（Google）	アマゾン・ドット・コム
3	アップル	マイクロソフト	アルファベット（Google）
4	BHPビリトン	バークシャー・ハサウェイ	マイクロソフト
5	マイクロソフト	エクソンモービル	フェイスブック
6	中国工商銀行	アマゾン・ドット・コム	アリババ
7	中国建設銀行	フェイスブック	バークシャー・ハサウェイ
8	ロイヤル・ダッチ・シェル	ゼネラル・エレクトリック	ＪＰモルガン・チェース
9	ネスレ	ジョンソン&ジョンソン	エクソンモービル
10	中国移動	ウェルズ・ファーゴ	ジョンソン&ジョンソン

1.2. 研究の目的とリサーチクエスチョン

本博士論文では、プロジェクト活動に対する組織レベルのリスクマネジメント、すなわち、“プロジェクト・リスクマネジメント”を研究の対象とする。リスクマネジメントという概念は非常に幅広く、自然災害や予期せぬ事故などのハザード・リスクマネジメント、株価や為替変動などに対する金融リスクマネジメント、法令や規程に関するコンプライアンス・リスクマネジメント、企業活動におけるプロジェクト以外の定常業務に関するオペレーショナル・リスクマネジメントなども存在するが、これらは今回の研究の対象には含めない。以下、本博士論文においては、特に断りのない限

[2] 1995 年～2015 年：https://www.sc.mufg.jp/products/sp/intro201712/index.html（参照 2019-07-28）
2019 年：https://www.rakuten-sec.co.jp/web/special/foreign_marketcap_ranking/（参照 2019-07-28）

り、リスクマネジメントは暗黙的に“プロジェクト・リスクマネジメント”を指すこととする。

本研究は、なぜ、プロジェクト・リスクマネジメントの効果的な実践は難しいかという問題意識からスタートしている。企業を取り巻く環境の変化がますます激しくなる中、プロジェクト・リスクマネジメントの必要性や重要性は広く認識されているが、プロセスが形骸化して正しく運用されず、リスクが後から大きな問題として顕在化するなどの失敗が繰り返されている。そこには、他のマネジメント領域とは異なる、プロジェクト・リスクマネジメント特有の本質的な難しさや課題が存在しているのではないかというのが本研究の問いである。すなわち、本研究のリサーチクエスチョンは以下のようになる。

リサーチクエスチョン

RQ1：プロジェクト・リスクマネジメントの本質的な課題は何か？

RQ2：その課題に対応するには、どのような枠組みが有効か？

RQ3：その枠組みを実現する上で、最も重要な技術要素は何か？

プロジェクト・リスクマネジメントは、技術経営の主要な課題の１つであり、多くの先行研究がある。具体的には、（１）標準的なリスクマネジメント・プロセスに関する研究、（２）個別リスクの分類や体系化に関する研究、（３）リスク分析手法や支援ツールに関する研究、（４）リスクマネジメントの実践事例の報告、（５）リスクマネジメントと他の技術領域との統合に関する研究、などがある。本研究は、これらの中では（５）の研究の１つに分類される。（５）の従来研究では、リスクマネジメントへのナレッジマネジメントの応用や人工知能（AI）・機械学習の応用などがそれぞれ個別に論じられてきたが、本研究では、両者の相補性に着目して、機械学習と知識創造の統合アプローチを示す。また、本研究の枠組みに適した新しい機械学習モデルを提案、公開データセットを用いて既存アルゴリズムとのベンチマーク評価を実施し、その有効性を明らかにする。

本研究の最終ゴールは、本博士論文の研究成果を実際のプロジェクト・リスクマネジメントに応用することで、プロジェクトの成功確率を高めることにある。

1.3. 研究の方法

本博士論文の研究ストラテジーを図 1.1 に示す。前半部分の質的調査研究では、テーマティック・アナリシス法（TA: Thematic Analysis）を採用する。ここでは、Boyatzis（1998）による TA を土屋（2016）が解説した方法に沿って進める。TA は質的分析手法の 1 つであり、質的データの中にパターンを見出すための体系的なプロセスである。質的データを切片化してラベル付けし、類似するラベルをグルーピングするという手順は、他の分析手法、例えば、グラウンデッド・セオリー・アプローチ（GTA: Grounded Theory Approach; Strauss and Corbin 1998; 戈木 2013）などとあまり差異がないように思える。しかしながら、TA は、GTA のような厳格な“方法論”ではなく、むしろ、柔軟な“分析手法”とみなすことができる。すなわち、研究者の哲学的立ち位置を問題とせずにどのような立場であっても用いることができる（土屋 2016）。また、TA には、既存の理論や仮説に基づく“演繹的分析手法”、生データからテーマを生成する“帰納的分析手法”、帰納的分析手法と演繹的分析手法を組み合わせた“ハイブリッドアプローチ”などの多様な分析手法があり、研究の目的に合わせて研究者自身が自由に手法を選択できるという特徴をもつ（土屋 2016）。

本研究では、まず、現状のリスクマネジメントに関する実務者のインタビュー調査の結果から、帰納的分析手法に基づく TA（帰納的 TA）を実施し、リスクマネジメントの実践上の効果や難しさを明らかにする。続いて、取引コスト理論（Williamson 1981; 菊澤 2016）およびプロスペクト理論（Kahneman 2011=2014; 友野 2006）を用いて本質的な課題についての理論的考察を行った上で、帰納的 TA の結果に立ち戻って仮説の妥当性を検証する（ハイブリッドアプローチ）。さらに、課題解決に向けた新たな枠組みとして機械参加型（machine-in-the-loop）リスクマネジメントを提案し、実開発部門に適用、適用部門の実務者に対してインタビュー調査を実施し、演繹的分析手法に基づく TA（演繹的 TA）で分析して適用効果を検証する。

本博士論文の後半部分では、工学などで通常用いられる形成的アプローチ（synthetic approach）を採用する。すなわち、機械参加型リスクマネジメントの実現に向けて、新しい機械学習モデル SNB（superposed naive Bayes; Mori and Uchihira 2019）を提案し、予測精度と解釈可能性の 2 軸でその有効性を評価する。まず、予測精度に関しては、既にいくつかの客観的な評価方法が確立しており、かつ、実験に利用可能な公開デー

タセットも存在するため、統計的手法に基づく実験・評価が可能となる。今回は、Ghotra et al.（2015）を参考にして高度にコントロールされた定量的実験を実施する。一方、解釈可能性については、まだ客観的な評価方法が確立していないため、今回は、Lipton（2016）の評価基準に基づく定性的アセスメントを実施する。予測精度と解釈可能性のトレードオフ分析の方法は、本研究の理論的貢献の 1 つと考えられる。

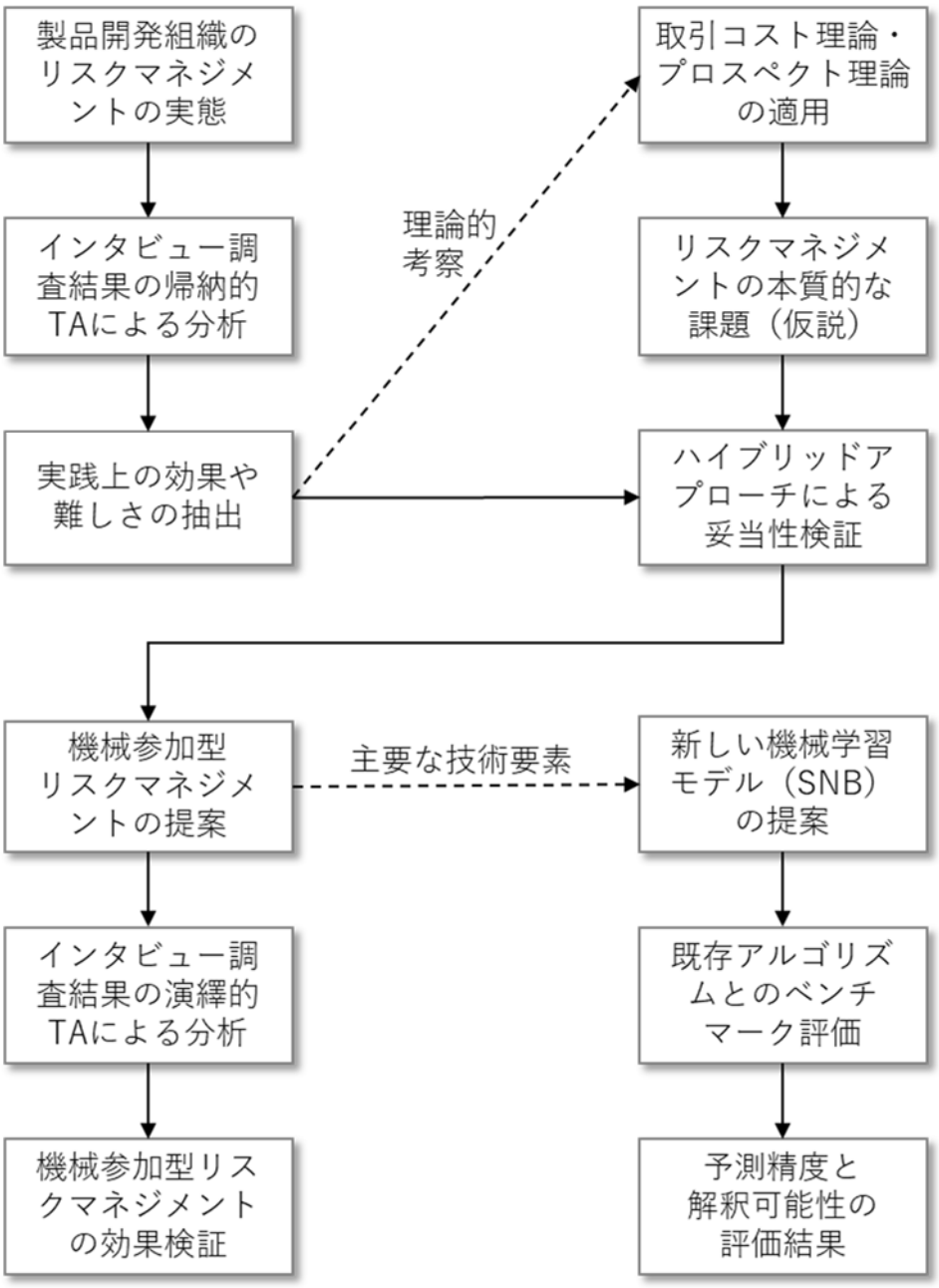


図 1.1：本博士論文の研究ストラテジー

1.4. 用語の定義

本博士論文は、プロジェクトマネジメント、ナレッジマネジメント、人工知能（AI）・機械学習という 3 つの異なる分野をまたがっている。以下、それぞれの分野における基本的な用語を定義する。

まず、プロジェクトマネジメント、および、リスクマネジメントに関して、Project Management Institute（2017=2018）、Smith and Merritt（2002=2003）、Cleden（2009）などを参考にして下記のように定義する。

プロジェクト（project）

独自のプロダクト、サービス、所産を創造するために実施する有期性のある業務。狭義のプロジェクトは、後述のプログラムやポートフォリオとは区別される。一方、広義のプロジェクトは、狭義のプロジェクト、プログラム、ポートフォリオを含んだ総称的概念である。

プログラム（program）

個別のプロジェクトでは達成できない価値を創造するために、整合的にマネジメントされる（狭義の）プロジェクトの集合体。

（プロジェクト）ポートフォリオ（project portfolio）

一体化してマネジメントされる（狭義の）プロジェクト、プログラム、および定常業務の集合体。ポートフォリオ内のプロジェクトやプログラムは、必ずしも相互に依存している必要はないという点がプログラムとは異なっている。

本博士論文においては、特に断りのない限り、プロジェクトという用語は“広義のプロジェクト”を指すものとする。

プロジェクトマネジメント（project management）

プロジェクトの目標を達成するために、知識・スキル・ツール・技法をプロジェクト活動へ適用すること。

（プロジェクト）タスク（project task）

プロジェクトの目標を達成するために必要な全作業を階層的に表した WBS（work breakdown structure）の一要素。担当者と期限が割り当てられ、プロジェクト計画の中に組み込まれる。

（プロジェクト）リスク（project risk）

発生が不確実な事象または状態であり、もし発生した場合、プロジェクトに有害な影響を与えるもの。プロジェクトに予想外の利益をもたらすものもリスクとして扱う場合があるが、本博士論文はプロジェクトの成功確率の向上を目的としていることから、成功の阻害要因となるリスクに限定する。リスクには、既知のリスクと未知のリスクがある。

既知のリスク

過去のデータなどを用いて将来起こることが予測可能な事象または状態のこと。確率的な枠組みで取り扱うことができる。known-unknowns（既知の未知）と呼ばれることもある（Cleden 2009; Ramasesh and Browning 2014）。

未知のリスク

過去のデータや知識がまったく役に立たず確率的に予測不可能、あるいは、何が起こるのかさえ予測できない事象または状態のこと。経済学の分野では、不確実性（uncertainty）とも呼ばれる（Knight [1921] 2006）。また、unknown-unknowns（未知の未知）と呼ばれることもある（Cleden 2009; Ramasesh and Browning 2014）。

（プロジェクト）リスクマネジメント（project risk management）

プロジェクトマネジメントの活動の一部であり、プロジェクト失敗の予兆となるリスクを早期に捉え、コントロールすること。一般に、リスクマネジメントという用語は、企業経営、コンプライアンス、金融、投資、環境、自然災害、安全など幅広い分野で使用されているが、本博士論文においては、プロジェクトマネジメントの分野に限定する。

続いて、ナレッジマネジメントに関して、Nonaka and Takeuchi（1995=1996）、Davenport and Prusak（1998=2000）、Dixon （2000=2003）、内平（2010）、Girard and Girard（2015）、人工知能学会編（2017）などを参考にして下記のように定義する。

データ（data）

何事かに関する事実の集合であり、明示的な意味は与えられておらず、１つ１つの事実の間の関係付けもされていない。

情報（information）

データと異なり明示的な意味（関連性や目的）をもっており、送り手と受け手をもつ。すなわち、情報の受け手に何らかの変化を与えることを意図して送り手によりつくられたもの。

知識（knowledge）

正当化された真なる信念（justified true belief）。知識の所有者の中で、所有者の価値観、過去の経験、課題・問題意識、現在の状況認識と結びついているもので、新しい情報に対して所有者の解釈・判断・行動を生み出すもの。知識には、暗黙知と形式知がある。

暗黙知（tacit knowledge）

個人の信念・直観・ノウハウなど、特定状況に依存する個人的な知識であり、そのままの形では他人への伝達が難しい。

形式知（explicit knowledge）

文章・図表・数字・数式など、自然言語や形式言語によって他人への伝達が可能な知識。

ナレッジマネジメント（knowledge management）

企業経営における管理領域の１つであり、知識の収集、蓄積、更新、分配、共有、創出などを含む一連の知識操作プロセスから構成される。その理論的基礎として、暗

黙知と形式知の相互変換による知識創造のプロセスモデルがある。

（組織的）知識創造（organizational knowledge creation）

組織成員が創り出した知識を、組織全体で製品やサービスあるいは業務システムに具現化するプロセス。人間の知識は暗黙知と形式知の社会的相互作用を通じて創造され拡大されるという前提に基づき、個人の暗黙知からグループの暗黙知を創造する“共同化”（socialization）、暗黙知から形式知を創造する“表出化”（externalization）、個別の形式知から体系的な形式知を創造する“連結化”（combination）、形式知から暗黙知を創造する“内面化”（internalization）という 4 種類の知識変換をスパイラルに繰り返すことによって、組織的かつ連続的な知識の創造と増幅を実現する。4 つの知識変換モードの頭文字をとって SECI モデルとも呼ばれている。

最後に、人工知能（AI）、および、機械学習に関して、人工知能学会編（2017）、Russell and Norvig（2003=2008）、Bishop（2006=2012）、Hastie et al.（2009=2014）、Davenport and Harris（2007=2008）、Arnott and Pervan（2005; 2014）などを参考にして下記のように定義する。

人工知能（AI: artificial intelligence）

認識、推論、判断など、人間と同じ知的な処理能力をもつコンピュータシステム。要素技術として、画像認識、音声認識、自然言語処理などのインタフェース技術、探索、知識表現、推論、学習などの汎用問題解決技術、対象分野ごとのオントロジー（語彙体系や基本ルール）構築技術などを含む。

機械学習（machine learning）

人工知能（AI）を実現するための要素技術の 1 つであり、人間が自然に行っている学習能力と同様の機能をコンピュータで実現しようとする技術・手法。大きくは、入力と出力の関係を学習する“教師あり学習”、データに内在する本質的な構造を抽出する“教師なし学習”、試行錯誤を通じて報酬を最大化するような行動を学習する“強化学習”の 3 種類に分類される。

データマイニング（data mining）

大量のデータから有用な知識を取り出すための一連のプロセス。データ収集と選択、データの前処理、データ変換、知識発見（knowledge discovery）アルゴリズムの適用、得られた知識の解釈と評価などのステップから構成される。機械学習とデータマイニングは技術や手法において多くの共通性があり、交差する部分も大きいが、その目的は若干異なる。すなわち、機械学習の目的がデータに内在する特徴に基づく予測であるのに対して、データマイニングは主に知識発見に重点を置いている。

意思決定支援システム（DSS: decision support system）

半構造化された、あるいは、構造化されていない意思決定問題におけるマネジャーの判断を支援するシステム。

ビジネスインテリジェンス（BI：business intelligence）

データに基づいてビジネスの実態や業績を把握し分析するための技術やプロセスの集合であり、その構成要素として、データウェアハウス、データ視覚化、データマイニング、経営意思決定支援、などを含む。

プロジェクトマネジメント、ナレッジマネジメント、人工知能（AI）・機械学習はそれぞれ独立な分野であるが､その理論的基盤や応用面では相互に密接に関連している。

1.5. 論文の構成

本研究では、インタビュー調査の結果からプロジェクト・リスクマネジメントの本質的な課題について理論的考察を行った上で、具体的な解決アプローチとして機械参加型（machine-in-the-loop）リスクマネジメントの概念を示し、その適用効果を検証する。さらに、新しい機械学習モデル SNB（superposed naive Bayes）を提案し、公開データセットを用いて予測精度と解釈可能性の両立性を評価する。以下に本博士論文の構成を示す。

第 1 章　序章

本研究の背景、本研究の目的とリサーチクエスチョン、基本的な用語の定義、および論文の構成を示す。

第 2 章　先行研究の検討

リスクマネジメントとその難しさ、リスクマネジメントへのナレッジマネジメントおよび機械学習の適用、機械学習とナレッジマネジメントの統合などに関する先行研究の検討を行い、本博士論文の位置付けを明確にする。

第 3 章　リスクマネジメントの現状と課題

現状のリスクマネジメントに関する実務者のインタビュー調査を実施し、帰納的分析手法に基づくテーマティック・アナリシス法（TA: Thematic Analysis）で分析して、リスクマネジメントの実践上の効果や難しさを洗い出す。

第 4 章　リスクマネジメントの困難性の理論的考察

前章で抽出したリスクマネジメントの実践上の難しさに対して、取引コスト理論およびプロスペクト理論を適用して本質的な課題の理論的考察を行う。さらに、TA のハイブリッドアプローチを適用して仮説の妥当性を検証する。

第 5 章　機械参加型リスクマネジメントの提案と評価

本質的な課題への対応として、機械参加型（machine-in-the-loop）リスクマネジメントを提案する。適用先部門の実務者に対してインタビュー調査を実施し、演繹的分析手法に基づく TA（演繹的 TA）で分析して before / after の効果を検証する。

第 6 章　予測精度と解釈可能性を両立した機械学習モデルの提案と評価

機械学習と知識創造の統合アプローチを実現するには、機械学習モデルの解釈可能性が重要である。ここでは、予測精度と解釈可能性を両立した新しい機械学習モデル SNB（superposed naive Bayes）を提案し、公開データセットを用いて既存アルゴリズムとのベンチマーク評価を実施する。

第 7 章　考察

上記結果に関して、いくつかの観点から考察を行う。

第 8 章　結論

リサーチクエスチョンに対する回答と本研究の理論的含意と実務的含意をまとめ、将来研究への示唆を述べる。

本博士論文の主要な研究成果は第 3 章、第 4 章、第 5 章、第 6 章であり、それぞれが研究ストラテジーに対して図 1.2 のように対応している。

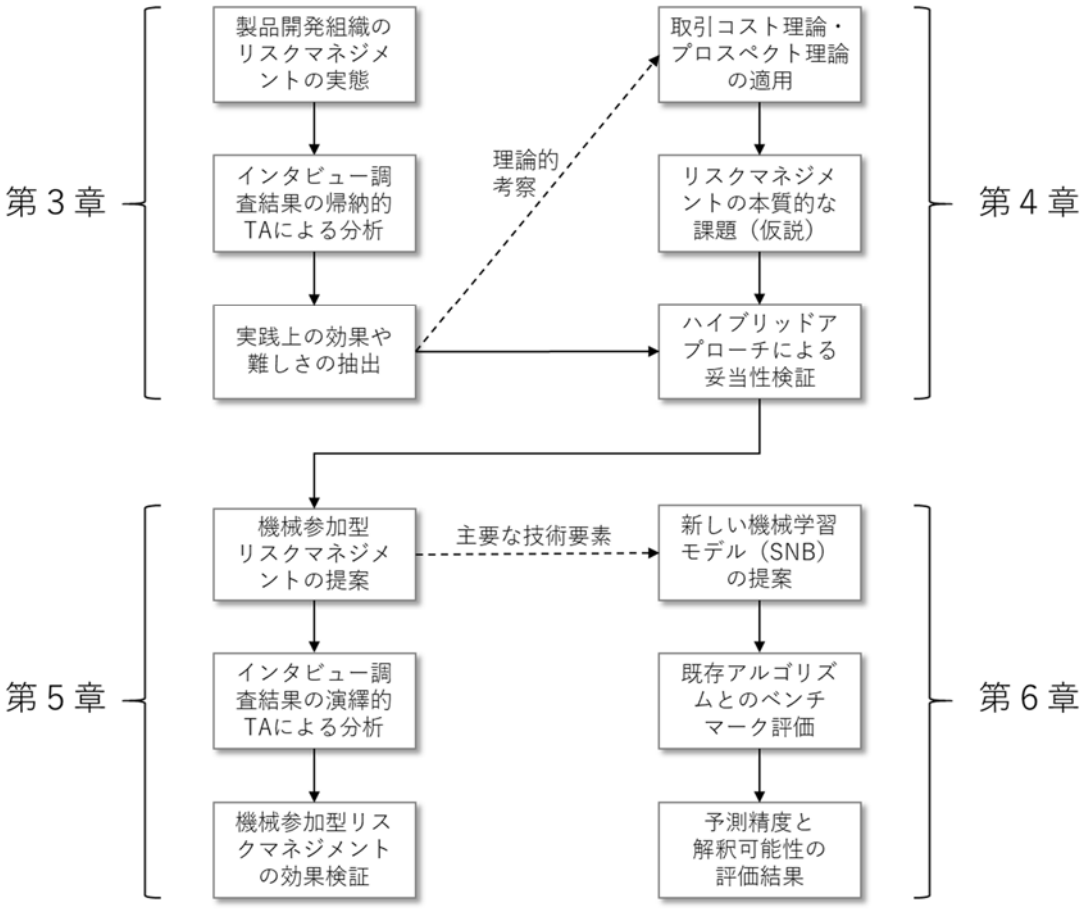


図 1.2：論文構成と研究ストラテジーとの対応関係

第2章 先行研究の検討

2.1. はじめに

本章では、（１）プロジェクトマネジメントとリスクマネジメント、（２）ナレッジマネジメントと知識創造、（３）人工知能（AI）と機械学習という３つの異なる分野、およびそれらの交差領域について先行研究の検討を行い、本研究の位置付けを明確にする。図 2.1 は、上記３つの研究分野の関係を表している。まず、2.2 節では、領域 A、すなわちプロジェクトマネジメントおよびリスクマネジメントの概要や標準について述べる。2.2.1 節では、さらにリスクマネジメントの実践上の難しさについて掘り下げる。2.3 節および 2.3.1 節では、それぞれ領域 B と領域 D、すなわちナレッジマネジメント・知識創造とそのリスクマネジメントへの適用について述べる。2.4 節および 2.4.1 節では、それぞれ領域 C と領域 E、すなわち人工知能（AI）・機械学習とそのリスクマネジメントへの適用について説明する。2.5 節では、領域 F、すなわち機械学習とナレッジマネジメントの統合について述べる。2.6 節では、さらに機械学習の解釈可能性について調査する。最後に、2.7 節にて、本研究の位置付け（領域 G に相当）を示し、先行研究との相違点を明らかにする。

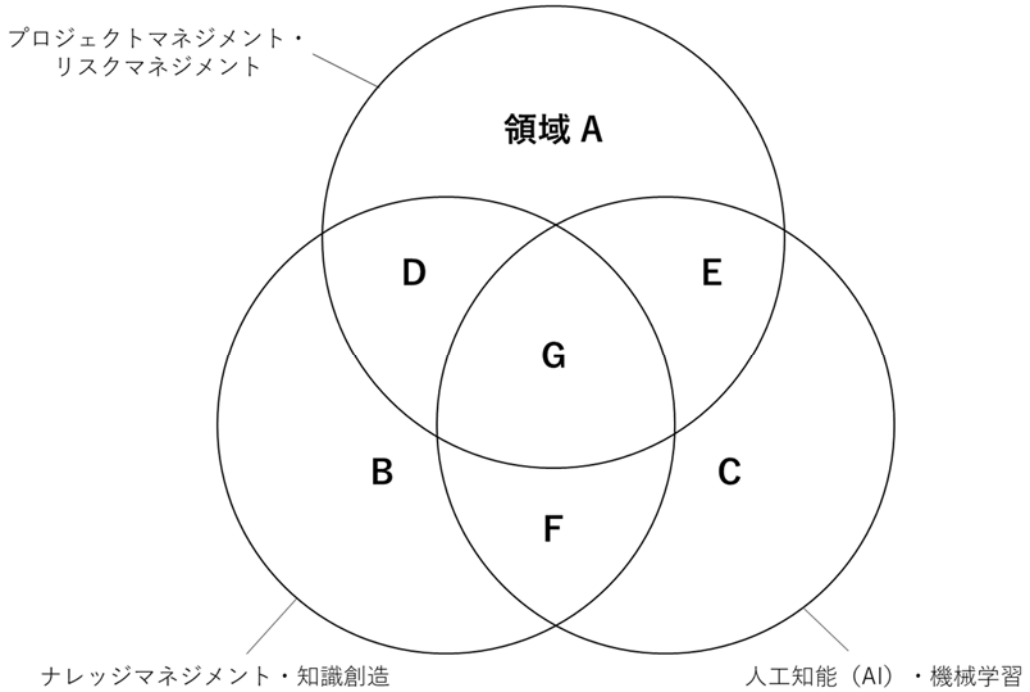


図 2.1：先行研究の検討範囲

2.2. プロジェクトマネジメントとリスクマネジメント

プロジェクトマネジメントの概念が確立する以前は、プロジェクトの運営は勘や経験に頼った属人的なものだったが、さまざまな活動の知見やノウハウが集約されることにより、プロジェクトマネジメントの概念・プロセス・手法は徐々に体系化されてきた。現在、プロジェクトマネジメントの標準的な知識体系としては、PMBOK（project management body of knowledge）ガイド（Project Management Institute 2017=2018）、P2M 標準ガイドブック（日本プロジェクトマネジメント協会 2014）、国際規格 ISO21500 シリーズ（ISO21500:2012; ISO21503:2017; ISO21504:2015）などがある。

PMBOK ガイドは、米国 PM（project management）学会が策定したガイドラインであり、1987 年にホワイトペーパーが作成され、1996 年に初版が発行された。最新版は、2017 年発行の第 6 版である。PMBOK ガイドは基本的にプロセスベースの体系であり、各プロセスは、インプット、ツールと技法、アウトプットの組み合わせとして記述される。第 6 版は 49 個のプロセスを含み、それらは以下に示す 10 個の知識エリアに分類されている。その内の 1 つにリスクマネジメントが含まれる。

1. プロジェクト統合マネジメント
2. プロジェクト・スコープマネジメント
3. プロジェクト・スケジュールマネジメント
4. プロジェクト・コストマネジメント
5. プロジェクト品質マネジメント
6. プロジェクト資源マネジメント
7. プロジェクト・コミュニケーション・マネジメント
8. プロジェクト・リスクマネジメント
9. プロジェクト調達マネジメント
10. プロジェクト・ステークホルダー・マネジメント

Royer（2001=2002）は、PMBOK ガイドに準拠して、立上げ、計画、遂行、コントロール、終結の各フェーズに対応したリスクマネジメントの具体的な進め方を説明し、ドキュメントの書式や、リスク監査のチェック項目、リスクデータベースのスキーマなどを示している。

P2M 標準ガイドブックは、(財) エンジニアリング振興協会が経済産業省の委託事業として 2001 年に発行した、日本発のプログラム＆プロジェクトマネジメントのガイドラインであり、2002 年以来、日本プロジェクトマネジメント協会（PMAJ）が普及を担当している。P2M 標準ガイドブックでは、プロジェクトを「繰り返しのない個別性と完了の期限を有する有期性を特徴とする活動」、プログラムを「組織戦略の実現などの目的達成のために複数のプロジェクトを有機的に組み合わせた統合的な活動」と定義し、その上で、プログラムを計画し実行するプログラムマネジメントと、プログラムを構成する個々のプロジェクトを確実に遂行するためのプロジェクトマネジメントの手法、および、その関連知識を統合的に取り扱っている。

ISO21500 シリーズは、PMBOK ガイドを初めとするさまざまなプロジェクトマネジメントの知識体系の要点を取り込んでいる。背景として、それ以前は各国の異なる団体が独自のプロジェクトマネジメント規格を定めていたが、2007 年以降、国際標準化に向けた動きが活発となり、ISO21500 シリーズが策定された。ISO21500 シリーズには、現在、"ISO21500:2012 Guidance on project management"、"ISO21503:2017 Project, programme and portfolio management -- Guidance on programme management" 、"ISO21504:2015 Project, programme and portfolio management -- Guidance on portfolio management" という 3 種類の規格が含まれており、それぞれプロジェクトマネジメント、プログラムマネジメント、ポートフォリオマネジメントに対応している。

上記ガイドラインを含め、さまざまな書籍・文献において固有のリスクマネジメント・プロセスが定義されている。一例として、以下に PMBOK ガイドにおけるリスクマネジメント・プロセスを示す。

1. リスクマネジメントの計画

   プロジェクトのリスクマネジメント活動を実行する方法を定義する。

2. リスクの特定

   どのリスクがプロジェクトに影響を与えるかを見定め、その特性を文書化する。リスクの特定はプロジェクト全期間にわたり繰り返し実行すべきプロセスであり、すべてのプロジェクト関係者が活動に参加することが望ましい。

3. リスクの定性的分析

   リスクの発生確率と影響度の査定に基づいて、この後の分析や処置のためにリスクの優先度付けを行う。

4. リスクの定量的分析

特定したリスクがプロジェクト目標全体に与える影響を数値により分析する。

5. リスク対応の計画

プロジェクト目標に対する好機を高め脅威を減少させるための選択肢と方策を策定する。リスク対応の主な戦略としては、回避、転嫁、軽減、受容などがある。

6. リスク対応策の実行、リスクの監視

リスク対応計画を実行し、特定したリスクを追跡して、残存リスクを監視する。また、新たなリスクの特定やリスクマネジメント・プロセスの有効性の評価なども行う。

表 2.1 に、主なリスクマネジメント・プロセスの比較を示す。各参考文献の中で用いられている表現をそのまま記載した。各ステップの粒度や名称は若干異なっているが、“リスクの特定”、“リスク分析”、“対応計画策定”、“リスクの監視”という基本的な作業の流れについては大きくは変わらない。

表 2.1：リスクマネジメント・プロセスの比較

参考文献	マネジメント計画	リスクの特定	リスク分析	対応計画策定	リスクの監視	振り返り
PMBOK (2017=2018)	・リスクマネジメントの計画	・リスクの特定	・リスクの定性的分析 ・リスクの定量的分析	・リスク対応の計画	・リスク対応策の実行 ・リスクの監視	
P2M (2014)	・方針策定	・リスクの特定	・リスクの分析・評価	・リスク対応策の策定	・対応策実施と監視・評価	・リスク教訓の整理
ISO21500 (2012)		・Identify risks	・Assess risks	・Treat risks	・Control risks	
Smith and Merritt (2002=2003)		・プロジェクトリスクの特定	・リスクの分析 ・リスクの優先付けとリスクマップの作成	・ターゲットリスクの解決計画立案	・プロジェクトリスクの監視	
Boehm (1991)		・Risk identification	・Risk analysis ・Risk prioritization	・Risk-management planning	・Risk resolution ・Risk monitoring	

リスクマネジメント・プロセスの各ステップを支援するために、これまで、さまざまな技法やツールが提案されてきた（Project Management Institute 2017=2018; Boehm 1991; Carbone and Tippett 2004; Smith and Merritt 2002=2003）。例えば、“リスクの特定”では、チェックリスト、デルファイ法、“リスク分析”では、リスクマトリックス、FMEA（failure mode and effects analysis）、デシジョンツリー、モンテカルロ・シミュレーション、“リスクの監視”では、リスクマネジメント・ダッシュボードなどが挙げられる。

2.2.1. リスクマネジメントの難しさ

リスクマネジメント・プロセスは概ね標準化されており、基本的な技法や支援ツールもほぼ出そろっている一方、リスクマネジメントの効果的な実践や定着化は必ずしも容易ではないという実態がある。本博士論文では、リスクマネジメントの標準的なプロセスや技法などの“理想形”と開発現場における“実践”とのギャップを、リスクマネジメントの“実践上の難しさ”と呼ぶことにする。

木野（2000）は、PMBOK とその他のリスクマネジメント体系を比較分析し、リスクマネジメントの実践上の課題について考察した。主な課題として以下を挙げている。

- リスクの特定方法としてチェックリストは有効であるが、汎用性を考慮し過ぎるとプロジェクト固有の状況に対応できず、リスクの特定作業における漏れが発生する可能性がある。
- リスクの定量的分析は「発生確率が予測しにくい」「リスクが顕在化したときの影響が多岐にわたる」「分析に労力をかける余裕がない」などの理由から実プロジェクトであまり実践されていない。
- リスクマネジメント・プロセスは他のマネジメント・プロセスと独立しており、プロジェクトの進捗に合わせて繰り返し適用する必要があるため、作業負荷が非常に高い。

Bannerman（2008）は、リスクマネジメントの理論と実践には大きなギャップが存在すると指摘しており、その理由の１つとして「リスクの認識は、ステークホルダーの価値観や組織文化などにより異なる」「マネジャーは、リスクの発生確率よりもその潜在的な損失の大きさに関心をもつ傾向がある」「ステークホルダーは、自分の責任範囲外のリスクに誘導しがちである」など、ヒューマン・ファクターの影響を挙げている。

また、プロセスの形骸化の危険性にも触れて、「リスクマネジメントは“型どおりに実施すればよい”（cookie cutter solution）プロセスではなく、その実践には高度なスキルと判断力と粘り強さを要する」と述べている。

Kutsch and Hall（2010）は、リスクマネジメントにおける意図的な“無関心”（irrelevance）について調査し、以下の 4 種類に分類した。

- 非話題性（untopicality）

  マネジャーなどの直感により過去の経験や好みに合致しないリスクを意図的に無視すること。一時しのぎとして、必ずしも重要ではないが自明なリスクや対処しやすいリスクを挙げておく場合もある。

- 非決定性（undecidability）

  リスクの根拠が弱いため、他のステークホルダーからリスクであると認めてもらえないこと。内部の人間関係による“隠された意図”（hidden agenda）が存在する場合もある。

- タブー（taboo）

  知ってはいけない、触れてはいけないなど、社会的に強要された無関心。重要なステークホルダーに余計な心配を与えてプロジェクトが中止になったら困る場合の配慮なども該当する。

- 態度の保留（suspension of belief）

  リスクが自然に解消するのを期待して、リスクが顕在化するまであえてリソースを費やさないこと。今やるべきことに追われて、リスク対応の優先度が下がっていることなどが原因。

さらに、マネジャー・クラスの実務者 18 名にインタビューし、上記バイアスの存在と影響を確認した。本論文では、“無関心”（irrelevance）は、通常のリスクマネジメントでは対応が困難であると結論付けている。

De Bakker et al.（2010）は、「リスクマネジメントは IT プロジェクトの成功に寄与するか？」という疑問に答えるために先行研究のメタアナリシスを実施した。1997 年から 2009 年までの学会誌論文 29 件を、過去のプロジェクトの結果を分析して主要なリスク要因を特定する“評価”（evaluation）アプローチと、現在のプロジェクトに対してリスクマネジメントの意思決定に必要な情報を収集・生成するためのプロセスを構築する“管理”（management）アプローチに分類し、それぞれに対して分析・考察を行っ

た。評価アプローチで得られるリスク要因の知識は有益だが、その後の具体的な行動に結びつかない可能性がある。一方、管理アプローチはプロジェクトの成功に直接的に寄与するが、プロセスの形骸化などの実践上の課題がある。本来、評価アプローチの成果は管理アプローチの中で活用されるべきだが、調査対象の先行研究においては両者の統合に触れたものはほとんどないことが判明した。

Cleden（2009）は、通常のリスクマネジメントが対象としている既知のリスクではなく、未知のリスクへの対応について論じている。プロジェクトの不確実性を完全になくすことは難しく、問題解決戦略（problem-solving strategies）、知識戦略（knowledge strategies）、予見戦略（anticipation strategies）、回復戦略（resilience strategies）、学習戦略（learning strategies）など、不確実性のライフサイクル（図 2.2）に応じた多面的なアプローチが必要であると主張している。

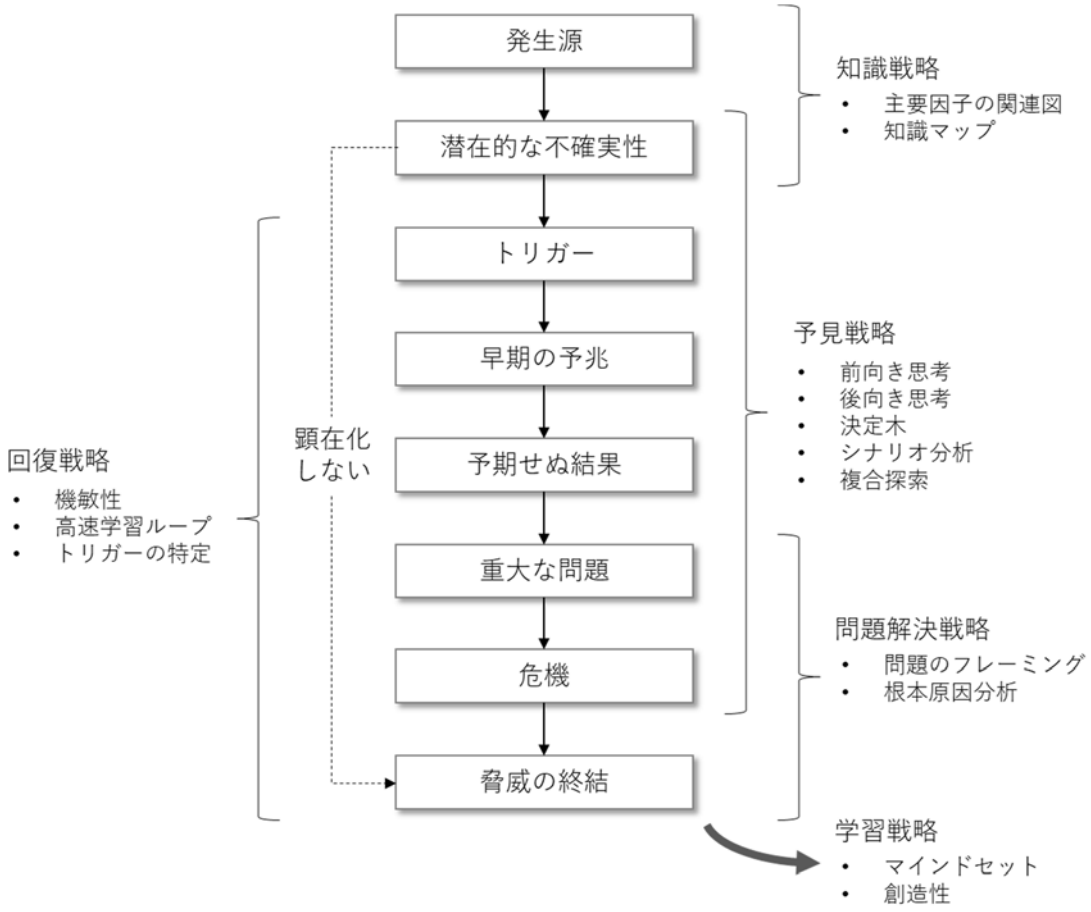


図 2.2：不確実性のライフサイクル（出典：Cleden 2009, p.18 に加筆）

2.3. ナレッジマネジメントと知識創造

ナレッジマネジメントは、組織目標を達成するために、人間・技術・知識コンテンツを組み合わせて知識を実務的に活用する試みである（Davenport and Prusak 1998=2000）。ナレッジマネジメントの中核的な理論の 1 つとして、組織的知識創造理論 （Nonaka and Takeuchi 1995=1996）がある。組織的知識創造理論では、知識を“正当化された真なる信念”（justified true belief）と定義し、「個人や組織の相互作用によりダイナミックに変化・進化していくもの」と捉えている。知識には、経験や勘に基づいており言葉などで表現が難しい暗黙知と、主に文章・図・数式などによって説明・表現が可能な形式知が存在する。組織的知識創造プロセスは、個人の知識を組織的に共有し、より高次の知識を生み出すことを目的としており、図 2.3 に示す SECI モデルが有名である。SECI モデルは、知識変換モードとして以下の 4 つのフェーズをもつ。

1. 共同化（Socialization）：暗黙知から暗黙知へ

   経験を共有することによって、メンタルモデルや技能などの暗黙知を創造するプロセス。暗黙知を共有する鍵は共体験であり、経験をなんらかの形で共有しないかぎり、他人の思考プロセスに入り込むことは難しい。

2. 表出化（Externalization）：暗黙知から形式知へ

   暗黙知を明確なコンセプト（概念）に表すプロセス。暗黙知がメタファー、アナロジー、コンセプト、仮説、モデルなどの形をとりながら次第に形式知として明示的になっていく。

3. 連結化（Combination）：形式知から形式知へ

   コンセプト（概念）を組み合わせてひとつの知識体系を創り出すプロセス。異なった形式知を組み合わせて新たな形式知を創り出す。

4. 内面化（Internalization）：形式知から暗黙知へ

   形式知を暗黙知へ体化するプロセス。行動による学習と密接に関連している。形式知が新たな個人へと内面化されることで、その個人と所属する組織の知的資産となる。

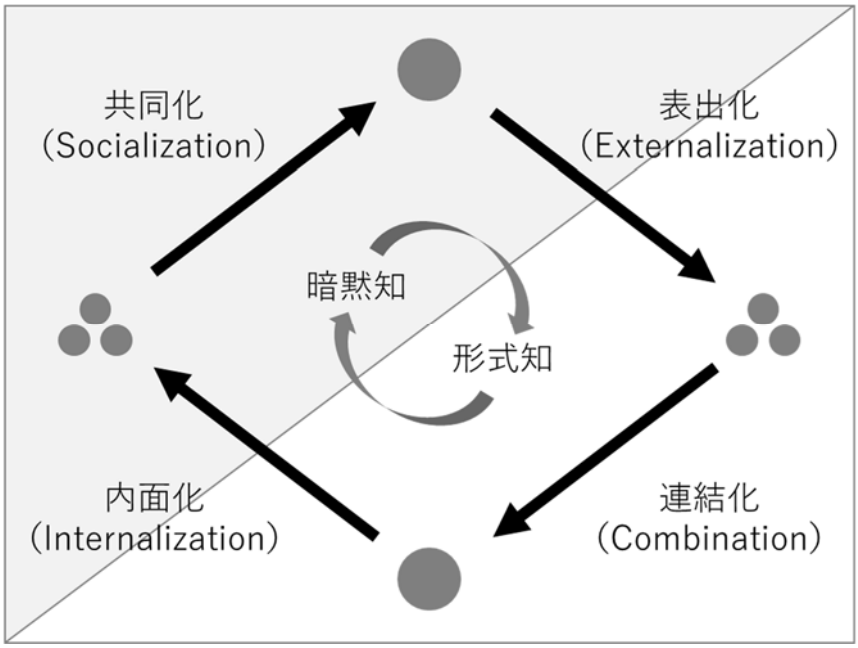


図 2.3：組織的知識創造プロセス（SECI モデル）

（出典：Nonaka and Takeuchi 1995=1996 を参照し筆者作成）

2.3.1. リスクマネジメントへのナレッジマネジメントの適用

プロジェクトリスクを組織で蓄積・活用すべき知識の一種とみなすことにより、リスクマネジメントへのナレッジマネジメントの適用が可能となる。Massingham（2010）は、リスクマネジメントとナレッジマネジメントという 2 つの独立した分野をまたがった新しい研究分野として“知識リスクマネジメント”（KRM: Knowledge risk management）の重要性を強調している。リスク評価の主観性に着目して、従来の発生度、影響度に加えて、知的資本（intellectual capital）に関する“個人”の特性、知識移転の障害（knowledge transfer barriers）に関する“知識”の特性、吸収能力（absorptive capacity）に関する“組織”の特性という 3 種類に認知的制約を考慮したリスク評価の枠組みを提案している。

Alhawari et al.（2012）は、知識獲得や知識共有などに起因するリスクマネジメント・プロセスの課題を改善するために、ナレッジマネジメントの技術要素を取り込んだ新たな枠組みを提案した。Knowledge-Based Risk Management framework（RiskManIT）は、ナレッジマネジメントに関連する新たなプロセスとして、KE（基本知識要素; knowledge essentials）、KBRC（知識に基づくリスク獲得; knowledge-based risk capture）、

KBRD（知識に基づくリスク発見; knowledge-based risk discovery）、KBREx（知識に基づくリスク説明; knowledge-based risk examination）、KBRS（知識に基づくリスク共有; knowledge-based risk sharing）、KBRE（知識に基づくリスク評価; knowledge-based risk evaluation）、KBRR（知識に基づくリスク貯蔵; knowledge-based risk repository）、KBREdu（知識に基づくリスク教育; knowledge-based risk education）を含む。

畑村（2000）は、失敗を「人間が関わって行う 1 つの行為が、はじめに定めた目標を達成できないこと」と定義した。その上で、失敗の特性を理解して知識化することで不必要な失敗を繰り返さないようにすること、さらに、その知識を活用して新たな創造へとつなげることを目標として“失敗学”を提唱した。失敗情報は、事象、経過、原因、対処、総括、知識化の 6 項目で整理され、再利用しやすい形でデータベースに蓄積される。濱口（2009）は、失敗学のエッセンスをリスクマネジメントへ展開する方法について論じている。特に、過度のマニュアル化による管理の形骸化の問題に着目し、上位概念への抽象化と下位概念への具体化を繰り返し行うことによってリスク情報を水平展開していく創造的アプローチの重要性を強調している。

内田（2016）は、プロジェクトリスクに関する知識を「認知バイアスにより間違った意味解釈を行う可能性のある知識」と捉えた上で、その知識移転について論じている。ここで、認知バイアスとは、「ある対象を評価する際に、自分の利害や希望に沿った方向に考えが歪められたり対象の目立ちやすい特徴に引きずられて、他の特徴についての評価が歪められる現象」（友野 2006）を指す。プロジェクト失敗に関する知識抽出において、分析者の視点を意図的にコントロールすることによっての先入観の混入を抑止する原因分析手法を提案している。

リスクマネジメントにナレッジマネジメントを適用したこれらの先行研究では、リスク評価の主観性の問題に対して、概念的枠組みやプロセスなどの形式的な側面からの対処を試みているが、リスクの本質である不確実性の定量的・確率的側面には十分に踏み込めていない。

2.4. 人工知能（AI）と機械学習

人工知能（AI）は、認識、推論、判断など、人間と同じ知的な処理能力をもつコンピュータシステムであり、その要素技術として、画像認識、音声認識、自然言語処理などのインタフェース技術、探索、知識表現、推論、学習などの汎用問題解決技術、対象分野ごとのオントロジー（語彙体系や基本ルール）構築技術などを含む（Russell and Norvig 2003=2008; 人工知能学会編 2017）。これらはすべて知的エージェントの構成要素となる。図 2.4 は、モデルと効用に基づくエージェントの構成を示している。

機械学習は、人工知能（AI）を実現するための主要な要素技術の１つであり、計算機アルゴリズムを通じてデータの中の規則性を自動的に見つけ出し、さらにその規則性を使って予測や推定などを行う（Bishop 2006=2012）。機械学習の手法は、入力と出力の関係を学習する“教師あり学習”、データに内在する本質的な構造を抽出する“教師なし学習”、試行錯誤を通じて報酬を最大化する行動パターンを学習する“強化学習”の３種類に分類される。この内、現在、最も実用化が進んでいるのは教師あり学習であり、本博士論文においても教師あり学習を中心に議論する。

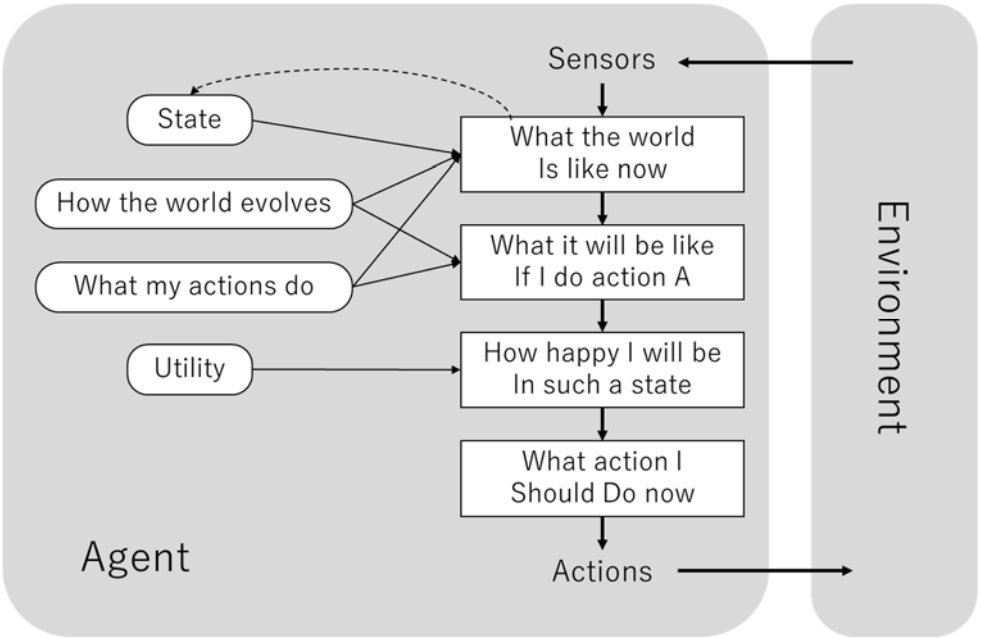


図 2.4：モデルと効用に基づくエージェント

（出典：Russell and Norvig 2003=2008, p.52 に加筆）

2.4.1. リスクマネジメントへの機械学習の適用

リスクマネジメントの難しさに対応するための有望なアプローチとして、近年発展が目覚ましい機械学習の応用がある。Takagi et al.（2005）は、表 2.2 に示した 5 つの観点、22 項目に関して、プロジェクトマネジャー32 名に対するアンケート調査を実施し、リスクの高いソフトウェア開発プロジェクトの特徴付けを行った。アンケート調査結果からロジスティック回帰モデルを作成し、2.3「暗黙の要件の不十分な見積り」、2.5「見積りに対するステークホルダーのコミットメント不足」、3.3「作業成果物の分解不足」、3.5「プロジェクト監視・制御の不十分な計画」の 4 つがプロジェクト失敗の主要因であることを特定した。

Lee et al.（2009）は、大規模エンジニアリング・プロジェクトを対象としたベイジアンネットワークによるリスクマネジメントの枠組みを示した。先行研究調査と有識

表 2.2：プロジェクトリスクのアンケート調査項目
（出典：Takagi et al. 2005 を参照し筆者修正）

1. 要件	1.1 あいまいな要件
	1.2 要件の不十分な説明
	1.3 要件の誤解
	1.4 顧客とプロジェクトメンバー間の要件に関するコミットメント不足
	1.5 頻繁な要件変更
2. 見積り	2.1 見積りの重要性の認識不足
	2.2 見積り手法の不十分なスキル・知識
	2.3 暗黙の要件の不十分な見積り
	2.4 技術的な問題の不十分な見積り
	2.5 見積りに対するステークホルダーのコミットメント不足
3. 計画	3.1 プロジェクト計画のマネジメントレビューの不足
	3.2 責任の割り当て不足
	3.3 作業成果物の分解不足
	3.4 プロジェクトレビューのマイルストーン未設定
	3.5 プロジェクト監視・制御の不十分な計画
	3.6 プロジェクト計画に対するプロジェクトメンバーのコミットメント不足
4. チーム編成	4.1 スキルと経験の不足
	4.2 不十分なリソース割り当て
	4.3 チームの士気が低い
5. プロジェクト管理活動	5.1 プロジェクトマネジャーのリソース管理の不足
	5.2 不適切なプロジェクト監視・制御
	5.3 プロジェクトの客観的な追跡に必要なデータの不足

者インタビューから 26 個のリスク項目を抽出し、その発生度と影響度について韓国の大手造船会社 11 社の 252 名に対する大規模アンケート調査を実施した。大企業向けと中小企業向けの 2 種類のベイジアンネットワーク・モデルを構築し、それに基づいて納期遅延や予算超過などに影響する主要なリスク項目を特定すると共に、主要リスクの感度分析や IF-THEN 分析などを行った。

Mori et al.（2013）、森ら（2013; 2014）は、プロジェクトの属性データ、およびプロジェクト計画時や進行中に収集される開発規模、工数、レビュー結果、工程遅延情報などの各種プロセスデータを入力として、プロジェクト進行中に動的にプロジェクト失敗確率を推定できるナイーブベイズ予測モデルを提案した。提案モデルは、図 2.5 に示すように、新たなデータにより事前確率が逐次更新されるベイズ更新の仕組みを用いてプロジェクト失敗確率を計算している。利用可能なデータが増えるほど、すなわち後工程になるほどモデルの予測精度が高まることが期待される。

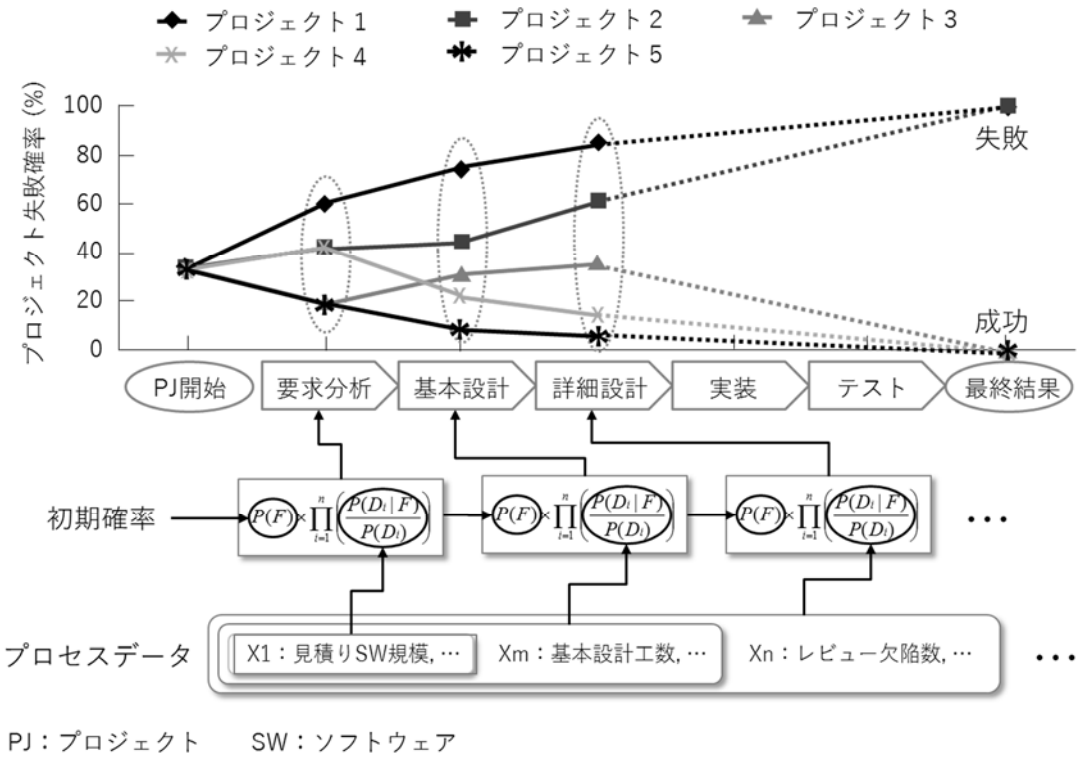


図 2.5：ナイーブベイズによるプロジェクト異常予測モデル

（出典：森ら 2014 に加筆）

Zhang and Tsai（2003）は、機械学習のソフトウェア工学への適用可能性を調査した。学習問題として再定義可能な作業の候補として、ソフトウェア部品の再利用、ラピッドプロトタイプピング、要求工学、リバースエンジニアリング、仕様の妥当性確認、テストオラクルの生成、テストの妥当性基準、ソフトウェア欠陥予測、プロジェクト工数予測、などを挙げている。

Mendes et al.（2018）は、ソフトウェア開発における意思決定支援においてベイジアンネットワークに基づくモデルを利用している。Web ベースのツールを用いてステークホルダーの知識を引き出し、ベイジアンネットワーク・モデルを半自動的に構築して総合的な価値評価を行う VALUE フレームワークを提案、フィンランドの大手情報通信会社において適用・評価を実施した。

大島・内平（2018）は、ソフトウェア開発におけるプロジェクトマネジメントに関する知識の分類モデルを提案している。形式知化またはシステム化の可否、人工知能（AI）による代替または補完の可能性によって知識を分類し、プロジェクトマネジメントへの人工知能（AI）活用の具体的な方策につなげている。

リスクマネジメントへの機械学習の適用は、過去のさまざまな知見をデータで裏付けたり、今まで知られていなかった新たな気付きを与えてくれるという意味で、非常に有益である。しかしながら、上記先行研究における機械学習の予測モデルはあくまで取得したデータからの帰納的推論であり、収集データの“外側”にある事象を予測することは難しい。

2.5. 機械学習とナレッジマネジメントの統合

人工知能（AI）・機械学習の技術は、近年、金融（Bahrammirzaee 2010）、マーケティング（Ngai et al. 2009）、製造（Li et al. 2017）、医用（Jiang et al. 2017）、農業（Kamilaris and Prenafeta-Boldu 2018）、法律（Surden 2014; 新田・佐藤 2019）、特許（Aristodemou and Tietze 2018）など、非常に幅広い分野で応用されている。人工知能（AI）・機械学習とナレッジマネジメントは相補的な関係にあり、広い意味においては、人間が関わるあらゆる応用分野でその統合が必要になると考えられる。ただし、それでは対象範囲が広すぎて全体を網羅することが難しいため、本博士論文では人間の意思決定支援に限定して関連論文を調査する。

意思決定支援システム（DSS: decision support system）の歴史は古く、その誕生は 1960 年代にさかのぼる。DSS は「半構造化された、あるいは、構造化されていない意思決定問題におけるマネジャーの判断を支援するシステム」と定義され、personal DSS（PDSS）、group support system（GSS）、negotiation support system（NSS）、knowledge management-based DSS（KMDSS）、intelligent DSS（iDSS）に分類される[3]（Arnott and Pervan 2005; 2014）。このうち、人工知能（AI）・機械学習とナレッジマネジメントの統合にもっとも関係するのは、intelligent DSS（iDSS）と knowledge management-based DSS（KMDSS）である。

Intelligent DSS（iDSS）は、人工知能（AI）・機械学習技術を応用した意思決定支援システムである。Merkert et al.（2015）は、1994 年から 2013 年までに発行された iDSS に関連する 52 件の論文（311 件から選択）を調査し、人工知能（AI）・機械学習の適用状況や意思決定支援への貢献について調べた。その結果、従来の研究は DSS および機械学習の技術的な側面に焦点を当てたものが大部分であり、意思決定の組織的な側面やヒューマン・ファクターを含めた検討は今後の課題であると述べている。

Bohanec et al.（2017）は、マーケティングの意思決定者が DSS の提案にしばしば懐疑的で、自身のメンタルモデルに従うことが多いため、組織効率を本質的に改善するためにはメンタルモデル自体を変える必要があると主張する。メンタルモデルは組織に深く根差した価値観を反映し、個人の行動を無意識的に制限すると考えられる。そこで、ある企業の B2B（business-to-business）事業の売上予測で試行して、解釈可能な機械学習モデルによる意思決定支援がメンタルモデルの変化を伴うような組織レベルの学習（図 2.6 のダブルループ学習）を促進することを確認した。

Knowledge management-based DSS（KMDSS）は、ナレッジマネジメントの実践、すなわち、個人や組織がもつ知識の蓄積、検索、移転、活用を通じて意思決定を支援する枠組みである。Shaw et al.（2001）は、大量の顧客データに基づくマーケティングの意思決定を支援するために、データマイニングの技術とナレッジマネジメントを組み合わせた統合フレームワークを示した。データマイニングによって発見された新たな知識は、ナレッジマネジメントによって系統的に管理され、最終的にマーケティング

[3] Arnott and Pervan（2005; 2014）の分類には executive information sysytem（EIS）、business intelligence（BI）、data warehouse なども含まれているが、本博士論文においては DSS の分類から除外した.

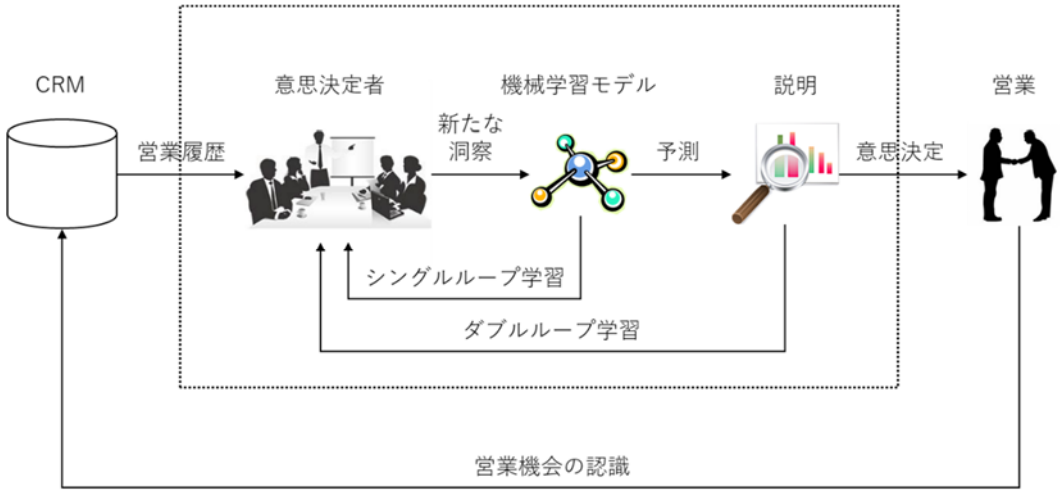


図 2.6：機械学習モデルによる意思決定支援と組織学習

（出典：Bohanec et al. 2017 に加筆）

戦略に結び付けられる。効果的な顧客関係管理（CRM: customer relationship management）を実現するには、ナレッジマネジメントとデータマイニングの両方の技術が必要であると述べている。

Herschel and Jones（2005）は、ナレッジマネジメントとビジネスインテリジェンスの関係について論じている。ビジネスインテリジェンスが形式知のみを取り扱っているのに対して、ナレッジマネジメントは暗黙知と形式知の両方を対象としていることから、ビジネスインテリジェンスはナレッジマネジメントの一部として統合されるべきであり、その定常的・構造的な情報処理プロセスを受けもつ一方、組織の知識創造には非定常的・非構造的な意味形成のプロセスも必要であり、両者のバランスをとるには組織文化やリーダーシップなどにも踏み込む必要があると述べている。

Wang and Wang（2008）は、データマイニングで得られた知識を行動に結びつけるために対象領域のビジネスに固有の知識が不可欠との認識から、ビジネス従事者（business insider）中心の知識創造サイクルとデータ分析担当者（data miner）中心のデータマイニング・サイクルをもつ 2 サイクルモデル（図 2.7）を提案している。

西原（廣瀬）（2019）は、人工知能（AI）がさらに発展した時代における組織的知識創造理論（Nonaka and Takeuchi 1995=1996）の役割について考察し、「意味を付与する」、「価値を創る」、「コンセプトを物語る」など、人間の創造性や社会的知性を必要とする作業については単純に人工知能（AI）に置き換えることは難しいと結論付けている。

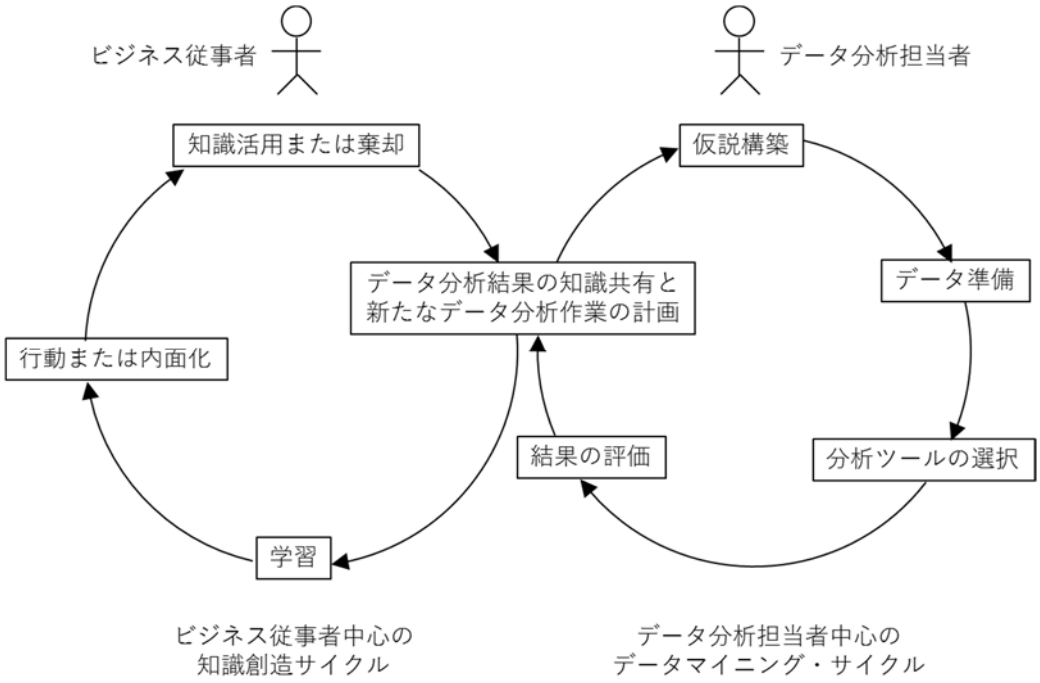


図 2.7：知識創造とデータマイニングの 2 サイクルモデル

（出典：Wang and Wang 2008 に加筆）

機械学習とナレッジマネジメントは相補的な関係にあり、人間の意思決定支援において両者の連携は不可欠である。しかしながら、その統合を効果的に実現するには、ブラックボックスの機械学習モデルでは不十分であり、機械学習の解釈可能性が重要となる。

2.6. 機械学習の解釈可能性

従来の機械学習の研究、特に教師あり学習においては、モデルの予測精度をいかに高めるかが議論の中心だった。しかしながら、近年、さまざまな分野で機械学習の実用化が進むにつれて、ブラックボックス・モデルの欠点や人間とのインタラクションの重要性が認識されるようになり、機械学習の解釈可能性（interpretability）が注目されている。そうした動きに関連して、アメリカ国防高等研究計画局（DARPA: Defense Advanced Research Projects Agency）の説明可能 AI（XAI: explainable artificial intelligence; Gunning 2017）など、さまざまな大型プロジェクトが立ち上がっている。

機械学習の解釈可能性の研究はまだ日が浅く、概念や用語が十分に確立していない

（Dosilovic et al. 2018）。解釈可能性は、説明可能性（explainability）や理解可能性（comprehensibility, understandability）などの用語とも密接に関連し、さまざまな文献で異なる定義がなされている。例えば、Ribeiro et al.（2016）は、解釈可能性（interpretability）を「入力変数と応答の間の定性的な理解を与える能力」と定義している。Doshi-Velez and Kim（2017）は、解釈可能性を「人間に理解可能な用語で説明または提示できる能力」と定義している。Kulesza et al.（2015）は、説明可能性（explainability）を「各予測に対する学習システムの根拠をエンドユーザーに正確に説明できる能力」と定義している。Martens et al.（2011）は、理解可能性（comprehensibility）を「予測モデルの心的適合（mental fit）」と定義している。これらの定義はそれぞれ少しずつ異なっているが、重なり合う部分も大きい。本博士論文においては、これらの概念を総称して“解釈可能性”（interpretability）と呼ぶことにする。

Lipton（2016）は、解釈可能性を、1 つの単純な概念ではなく複数の異なる概念の組み合わせと考え、予測モデルの内部動作に関する“透明性”（transparency）と外部出力に関する“事後解釈性”（post-hoc interpretability）に大別した。前者は、さらにモデルの透明性（模倣可能性：simulatability）、コンポーネントの透明性（分解可能性：decomposability）、アルゴリズムの透明性（algorithmic transparency）に分類される。一方、後者は、モデルの内部動作に踏み込まずに事後的に有用な情報を得ることが目的であり、テキストによる説明（text explanations）、可視化（visualization）、局所的説明（local explanations）、事例による説明（explanation by example）など、さまざまな方法が存在する。

予測モデルの解釈可能性を改善する第 1 のアプローチは、予測モデルそのものの透明性を高めることである。それには、決定木、ロジスティック回帰、ベイジアンネットワークなどの比較的解釈可能性に優れた予測モデルを採用する方法と、サポートベクターマシン、ニューラルネットワーク、集団学習（ensemble learning）などの予測精度は高いが複雑なモデルを単純で解釈容易なモデルに変換する方法がある。例えば、Baesens et al.（2003）は、与信リスクの評価においてニューラルネットワークの予測モデルからルールを抽出してデシジョンテーブルに変換する方法を示した。Martens et al.（2007）は、同じく与信リスクの評価においてサポートベクターマシンから解釈容易なルールを抽出した。Van Assche and Blockeel（2007）は、バギング、ランダムフォレストなどの集団学習モデルから単純な決定木を生成する方法を提案した。

予測モデルの解釈可能性を改善する第２のアプローチは、予測モデルの事後解釈性を高めることである。例えば、部分従属プロット（PDP: partial dependence plots）は、ブラックボックスの予測モデルに対してある１つの変数が変化したときの予測値の平均的な反応を可視化している（Goldstein et al. 2015）。Ribeiro et al.（2016）は、注目する入力値または予測値の周辺を局所的に近似したモデルを生成することで予測モデルの種類に依存せずに事後的な解釈を行う手法（LIME: local interpretable model-agnostic explanations）を提案した。Selvaraju et al.（2017）は、深層学習（deep learning; LeCun et al. 2015）を含む畳み込みニューラルネットワーク（CNN: convolutional neural network）を対象として、注目するクラスへの影響が大きい画像箇所を勾配の平均化によって特定しヒートマップなどで可視化する Grad-CAM（gradient-weighted class activation mapping）を提案した。Guidotti et al.（2018）は、ブラックボックスの予測モデルの透明性および事後解釈性を高めるためのさまざまな手法について、網羅的なサーベイを実施している。

一方で、予測モデルの解釈可能性を客観的に評価することは非常に難しい。もし、同じ種類のモデル同士を比較するのであれば、例えば、回帰モデルの項数、決定木のノード数など、モデルの“大きさ”が解釈可能性の評価指標になりえる。しかしながら、“大きさ”に基づく指標はモデルの種類に依存するため、異なる種類のモデル同士の比較には使えない。実験による解釈可能性の評価に関して、いくつかの研究（Allahyari and Lavesson 2011; Huysmans et al. 2011）が報告されているが、使用するデータセットや被験者の経験などのバイアスの影響を受けやすいとの指摘もある（Freitas 2014）。Doshi-Velez and Kim（2017）は、解釈可能性の評価方法を（１）人間の実際の作業で評価する“応用立脚型評価”（application-grounded evaluation）、（２）単純化した作業で実験する“人間立脚型評価”（human-grounded evaluation）、（３）代理作業を実験以外の方法で評価する“機能立脚型評価”（functionally-grounded evaluation）の３つに分類し、それぞれの課題と今後の研究の方向性をまとめている。

機械学習の応用面に目を向けると、例えば、医用分野における機械学習の応用では、人間の高度な専門知識に基づく意思決定、まれにしか発生しない事象を含む少ないデータセットなど、いわゆるビッグデータの世界とは異なる状況が存在する。そのような状況に対応して、機械学習のモデル構築プロセスへの人間の関与をさらに深めた“人間参加型”（human-in-the-loop）機械学習が提案されている（Holzinger 2016; Robert et al.

2016)。人間の参加は、自動化された機械学習モデルを監視して誤りを防止するだけでなく、NP 困難の組み合わせ問題などの実世界の複雑な課題に対して、機械学習単独の場合よりも一層効率的に解決できる可能性を示唆する（Holzinger et al. 2017)。

また、“人間参加型”（human-in-the-loop）機械学習とは逆の発想として、機械が人間の活動を支援する“機械参加型”（machine-in-the-loop）のシステムも提案されている。Clark et al.（2018）は、machine-in-the-loop の文書作成支援システム（図 2.8）を試作し、ユーザー実験による評価を実施した。提案システムは、人間が必要な文脈を提示することで機械から提言を受け取るなど、人間が常に操作の結果をコントロールする。ただし、machine-in-the-loop の概念は発表されてから日が浅く、その定義やアーキテクチャーについては発展途上であり、まだ十分に成熟しているとは言い難い。

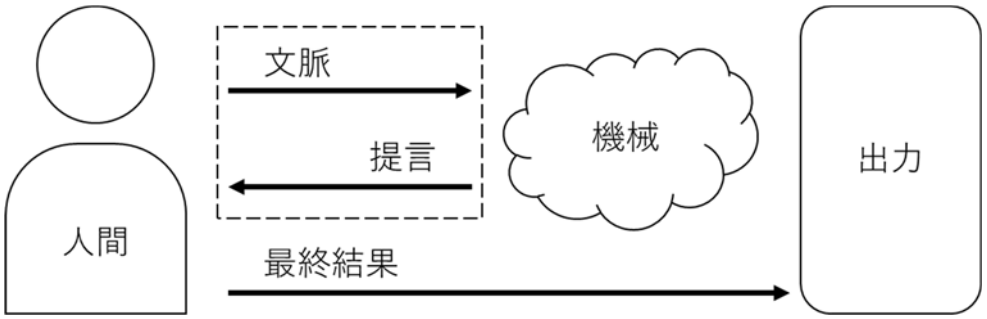


図 2.8：machine-in-the-loop 文書作成支援システム
（出典：Clark et al. 2018 に加筆）

このように説明可能 AI（XAI）の研究は、近年、大変な盛り上がりをみせているが、Miller（2019）は、その大部分の研究は「何がよい説明か？」についての理論的な裏付けが乏しいと指摘している。すなわち、現状の XAI に関する研究の多くは以下の 4 つの観点が不足しており、人間の“説明”行為に関する哲学、心理学、認知科学、社会学などの過去の知見をもっと活用すべきであると主張している。

1. 人間の説明は、対比的（contrastive）である。
2. 人間は、説明を（偏った方法で）選択する。
3. 人間にとって、確率よりも因果的理解の方が重要である。
4. 人間の説明は、社会的（social）である。

2.7. 本研究の位置付け

リスクマネジメント・プロセスは概ね標準化されており（2.2 節)、基本的な技法や支援ツールもほぼ出そろっている一方、リスクマネジメントの効果的な実践や定着化は必ずしも容易ではないという実態がある（2.2.1 節)。

リスクマネジメントの難しさへの対応としてナレッジマネジメントを適用した先行研究が存在する。それらはリスク評価の主観性の問題に対してフレームワークやプロセスなどの仕組み的な面からの対処を試みているが、リスクの本質である不確実性の定量的・確率的側面には十分に踏み込めていない（2.3.1 節)。

リスクマネジメントの難しさに対応するためのもう 1 つのアプローチは機械学習の応用である。それらは、過去のさまざまな知見をデータで裏付けたり、今まで知られていなかった新たな気付きを与えてくれるという意味で非常に有益であるが、あくまで過去に取得したデータからの帰納的推論であり、収集データの“外側”にある事象を予測することは難しい（2.4.1 節)。

機械学習とナレッジマネジメントは相補的な関係にあると考えられており、その統合に関して、人間の意思決定支援を中心に多くの先行研究が存在する。両者の効果的な統合を実現するには、ブラックボックスの機械学習モデルでは不十分であり、機械学習の解釈可能性が重要となる（2.5 節)。

機械学習の解釈可能性は、現在、さまざまな領域で活発に研究が行われているが、その概念や客観的な評価方法は発展途上であり、まだ十分に確立していない(2.6 節)。

本研究では、リスクマネジメントの困難性への対応として機械学習とナレッジマネジメントの統合アプローチを提案し、その有効性を検証する（図 2.1 の領域 G に相当)。ただし、統合アプローチの有効性が個別アプローチの有効性を単純に足し合わせたものと同じでは意味がない。リスクマネジメントの本質的な課題についての理論的な考察を示した上で、その解決に向けた具体的な枠組みを提案する。このようなリスクマネジメント、ナレッジマネジメント、機械学習の交差領域（図 2.1 の領域 G）においては、筆者の知る限り、関連する先行研究は存在しない。

本研究では、さらに、提案アプローチに適した新しい機械学習モデルとして、SNB（superposed naive Bayes）を提案する。SNB は、予測精度と解釈可能性の両立を特徴としており、その有効性を実証するために実開発プロジェクトの公開データセットを

用いて既存アルゴリズムとのベンチマーク評価を行った。予測精度と解釈可能性のトレードオフに関する実証的評価においても、本研究の新規性を主張できると考える。

第3章 リスクマネジメントの現状と課題

3.1. はじめに

本章では、製品開発組織におけるリスクマネジメントの現状と課題について実務者へのインタビュー調査を実施し、帰納的テーマティック・アナリシス法（TA: Thematic Analysis）を用いた分析によりリスクマネジメントの実践上の課題を明らかにする。3.2 節では、製品開発組織における一般的なリスクマネジメント・プロセスについて概観する。3.3 節では、製品開発部門のマネジャー・クラスの実務者 7 名に対して実施したインタビュー調査の目的と方法を述べる。3.4 節では、インタビュー調査で得られた質的データに対して帰納的 TA による分析を実施し、その結果をまとめる。

3.2. 製品開発組織におけるリスクマネジメント

製品開発を取り巻く環境の不確実性が増す中で、プロジェクト・リスクマネジメントの重要性は広く認識されており、さまざまな取り組みが行われている。多くの製品開発組織では PMBOK や国際規格などを参考として部門内の規程や標準プロセスが定められ、プロジェクトを対象とするリスクマネジメントが実施されている。製品開発組織におけるリスクマネジメントの特徴として、必ず、全体のプロジェクトマネジメントの進行に同期する形でリスクマネジメントが実行されるという点が挙げられる。

図 3.1 は、プロジェクトマネジメントに同期したリスクマネジメント・プロセスを表している。図の左側は、一般的なプロジェクトマネジメント・プロセスであり、立上げ、計画、実行、コントロール、終結の 5 つのステップからなる。コントロールから計画へのフィードバック･ループは､是正措置に基づく計画の修正に対応している。一方、図の右側は、リスクマネジメントの中核ステップであるリスクの特定、リスク分析、対応計画策定を表す。本来のリスクマネジメント・プロセスは、マネジメント計画、リスクの監視、振り返りなどのステップを含んでいるが、それぞれ、プロジェクトマネジメント・プロセスの立上げ、コントロール、終結へ統合可能であるため、図から省いている。 2 つのプロセスをつなぐ矢印は、プロセス間の制御の移行を示している。

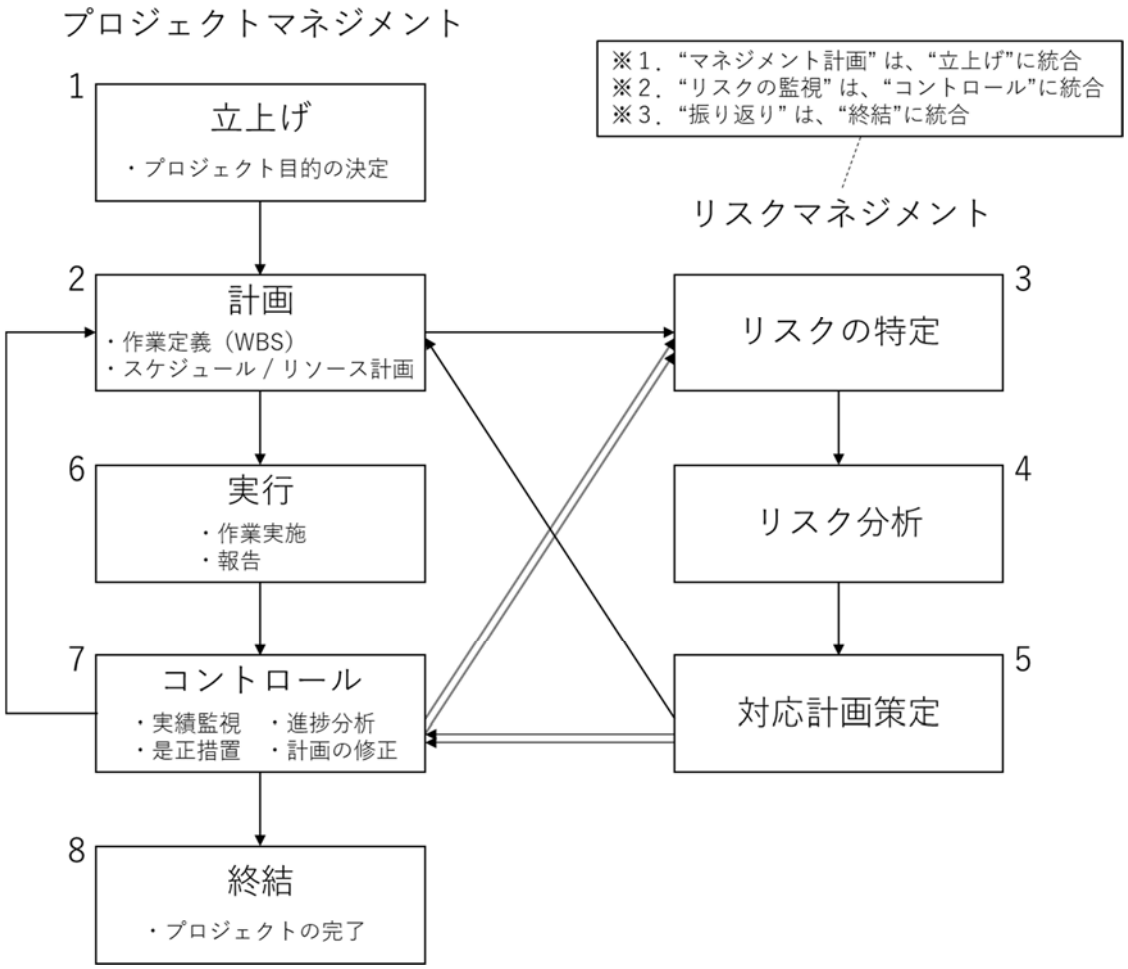


図 3.1：製品開発組織におけるリスクマネジメント・プロセス

以下、各ステップの中身を詳しく述べる。

1. まず、プロジェクトマネジメントの“立上げ”のステップでは、プロジェクトの目的・目標を明確化した上で、活動のスコープを定める。その際、何を「やらない」かを決定することも重要である。そして、主要メンバーのアサインなど、プロジェクト体制を確立して、前提条件や制約条件を洗い出し、プロジェクトマネジメントの全体計画を実施する。リスクマネジメントのプロセスや帳票[4]、リスク区分、役割分担などを決定するリスクマネジメント計画も、本ステップで実行される。
2. 続く“計画”のステップでは、プロジェクトに含まれる作業を分解して WBS（work

[4] 通常は、組織標準のプロセスや帳票などが存在するので、それをそのまま使用するか、またはテーラリングして使用するかを決定すればよい.

breakdown structure）を作成し、それに基づいてスケジュール計画やリソース計画、コスト計画などを策定する。WBS は、プロジェクトで実行することが“確定”したタスクの階層的リストであり、すべての活動計画の基本になる。

3. プロジェクトマネジメント・プロセスと並行して、リスクマネジメント・プロセスが起動する。すなわち、WBS をインプットとして、“リスクの特定”が実行される。プロジェクトメンバーが主体となり、可能であれば経験豊富な有識者（ベテラン技術者など）を加えて、プロジェクトで発生する可能性があるリスクをなるべく多く洗い出す。このステップにおける主要なツールはブレインストーミングであり、なるべく限定せずに想定できるリスクは何でも議論の場に乗せることが重要である。典型的なリスクのカテゴリーとしては、例えば、顧客関係、要求仕様、プロジェクト体制、メンバーのスキル、技術的な課題、調達品の品質などがある。特定したリスクは、リスク帳票に記録しておく。
4. “リスク分析”では、帳票に記録されたリスクに対して、その発生確率と影響度を評価し、対応計画策定のためのリスクの優先度付けを行う。その際、発生確率と影響度を 2 次元で評価したリスクマップや、FMEA（failure mode and effects analysis）などのツールが使用される。こうした分析は、定性的リスク分析と呼ばれる。また、一部の先行組織では、モンテカルロ・シミュレーションなどを用いた定量的リスク分析が実施されることもある。ただし、一般に、定量的リスク分析は非常に労力がかかるため、重要な案件など、特別な場合の評価に限定しているケースが多い。
5. リスクマネジメントの“対応計画策定”では、リスクの優先度に基づいて、すぐに対応策を実施するべきか、または、判断を保留して経過観察するべきかを決定する。リスク対応計画としては、例えば、リスクの回避、影響の軽減、第三者への移転などがある。不測の事態に備えて、代替案を検討したり予備の予算やスケジュールを確保することも重要である。対応計画策定の結果はプロジェクト計画にフィードバックされ、追加タスクまたは修正タスクとして WBS に反映される。
6. プロジェクトマネジメント・プロセスに戻ると、確定した WBS に基づいてスケジュールが作成され、順次タスクが実行される。各タスクの担当者は、プロジェクトが定めたルールにしたがって作業の進捗や結果をプロジェクトリーダーに報告する。WBS に反映されたリスク対応策についても、他のタスクと同様に担当者や

期限が設定されて順次実行される。

7. 続く“コントロール”のステップでは、報告に基づいて作業の実績を監視する。計画との差異を分析した上で、必要に応じて是正措置を検討する。結果は“計画”のステップにフィードバックされて、追加タスクまたは修正タスクとして WBS に反映される。なお、リスクはプロジェクトの進行にともなって時々刻々と変化するものであるため、継続的な“リスクの監視”が必要となる。また、リスクマネジメントは最初の1回だけ実施すればよいプロセスではなく、リスクの変化に応じて、プロジェクト期間中、繰り返し実行する必要がある。図中の“コントロール”から“リスクの特定”への複数の矢印、および、“対応計画策定”から“コントロール”への複数の矢印は、リスクマネジメント・プロセスの繰り返し実行の可能性を示している。
8. 最後の“終結”のステップでは、プロジェクト全体の実行結果を整理し、メンバーによる振り返りを実施する。そこには、リスクマネジメントの振り返りも含まれる。プロジェクトの知見と反省点を将来のプロジェクトに伝えることは、継続的な組織改善という観点からも非常に重要である。

3.3. インタビュー目的と方法

製品開発組織におけるリスクマネジメントの課題を調査するために、製品開発部門のマネジャー・クラスの実務者7名へのインタビュー調査を実施した。インタビュー対象者は、プロジェクトマネジメントの知識および実務に精通していること、部門横断的にさまざまなプロジェクトに関わっていることなどを基準として､社会インフラ、電力、半導体など、さまざまな事業領域から選定した。いわゆる合目的的サンプリング（purposive sampling）である。表 3.1 にインタビュー対象者の一覧を示す。本来、製品開発のリスクマネジメントに関わる情報は機密性が高く情報の公開が困難であるが、各インタビュー対象者には匿名性の確保と特定の組織や製品に関わる情報については非公開とすることを条件に了承を得た。また、各製品開発部門はそれぞれ固有のリスクマネジメント・プロセスをもっているが、細部の違いを無視すれば前節で示したプロセスと概ね類似していることを付け加えておく。

表 3.1：インタビュー（第 1 回）の対象者

対象者	実務年数	事業領域	インタビュー日時	場所
A	20年以上	社会インフラ	2019/01/24（木）14:15-15:00	X 事業所
B	10〜19年	社会インフラ	2019/01/28（月）13:30-14:30	Y 事業所
C	20年以上	半導体	2019/02/04（月）15:10-16:00	Z 事業所
D	10〜19年	電力	2019/02/05（火）11:00-11:45	Y 事業所
E	20年以上	社会インフラ	2019/02/06（水）09:00-09:45	X 事業所
F	10〜19年	社会インフラ	2019/02/07（木）17:30-18:15	X 事業所
G	20年以上	電力	2019/02/13（水）13:10-14:00	Y 事業所

インタビューは、あらかじめ用意したスクリプトに沿って開始し、適宜、気になる点について自由に意見を述べてもらうという“半構造化インタビュー”の形式を採用した。各インタビュー対象者には、インタビューの冒頭において、インタビューの目的、研究の方法、インタビュー内容の公開方法と注意点、過去の経験に基づく率直な意見を聞きたい旨、などを伝えた。

インタビュー・スクリプトに含まれる質問項目の最終版を表 3.2 に示す。ここで、質問項目中の“有識者”とは、対象の製品開発において豊富な知見をもっており、レビュー会議や認定会議などの製品開発における重要な意思決定の場への出席が求められるような管理者またはベテラン技術者のことを指している。質問項目はインタビュー対象者に対して事前にメールで送付した。その上で、インタビュー時に、まず、各質問項目に対して Yes/No で回答してもらい[5]、続いて「なぜ、そのように思うか？」と尋ねることによって、回答の背景にある思いや考えを聞き出すようにした。このような方法をとった理由として、リスクマネジメントの概念は非常に幅広く、各インタビュー対象者の役割や関心事もそれぞれ異なるため、漠然と質問すると発散してしまい核心部分にたどり着けない可能性を考慮した。一方、インタビュー結果を特定の答えに誘導してしまうのではないかとの懸念もあったが、表 3.2 の質問項目自体がかなり抽象的であるため、インタビュー結果の方向性を必要以上に限定しないと判断した。

[5] Q9、Q10、Q11 を除く.

表 3.2：インタビュー（第 1 回）の質問項目

Q1	リスクマネジメントは重要か？
Q2	リスクマネジメントは難しいか？
Q3	リスクマネジメントにおいて有識者の知見は必要か？
Q4	リスクマネジメントにおいて有識者の知見の活用は難しいか？
Q5	有識者の知見の中に思い込みやバイアス（偏見）はあるか？
Q6	リスクマネジメントにおいて異なる立場間の対立が生じることはあるか？
Q7	リスクマネジメントにおいてデータの活用は必要か？
Q8	リスクマネジメントにおいてデータの活用は難しいか？
Q9	有識者の知見とデータの活用はどのように補完し合えるか？
Q10	リスクマネジメントの本質的な課題は何か？
Q11	リスクマネジメントの有効性を高めるには何が必要か？

表 3.3 は、各質問項目に対するインタビュー対象者の Yes/No 回答の結果を示している。ただし、Q9、Q10、Q11 は、Yes/No を尋ねる種類の質問ではないため、ここには含めていない。“Yes->No”は、最初の回答は肯定的だが、すぐ後に否定的な内容の補足が続いたことを意味する。“No->Yes”は、逆に、最初の回答は否定的だが、すぐ後の補足が肯定的だった場合である。斜線は、その質問に対する回答が得られていないことを示す。

表 3.3：質問項目への Yes/No 回答の結果

	A	B	C	D	E	F	G
Q1	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Q2	Yes	Yes	Yes	Yes	No->Yes	Yes	No->Yes
Q3	Yes	Yes	Yes->No	Yes	Yes->No	Yes->No	Yes
Q4	Yes	Yes	Yes	No	Yes	Yes	Yes
Q5		Yes	Yes	No	Yes	Yes	Yes
Q6		Yes	Yes	Yes	Yes	No	Yes
Q7	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Q8	Yes	Yes	Yes		Yes	Yes	Yes

表 3.3 の各質問項目の細かい表現や質問の順番などについては、直前のインタビューの結果をふまえて都度改良を加えている。図 3.2 に、各質問項目の変遷を示す。各質問項目は上から順番に実施され、括弧のついた質問項目は最終版とは異なる表現を用いていたことを意味する。図 3.2 から、インタビュー対象者 B または C で概ね質問項目の大枠が定まってきたこと、および、Q9 はなかなか質問の表現が固まらなかったことがわかる。

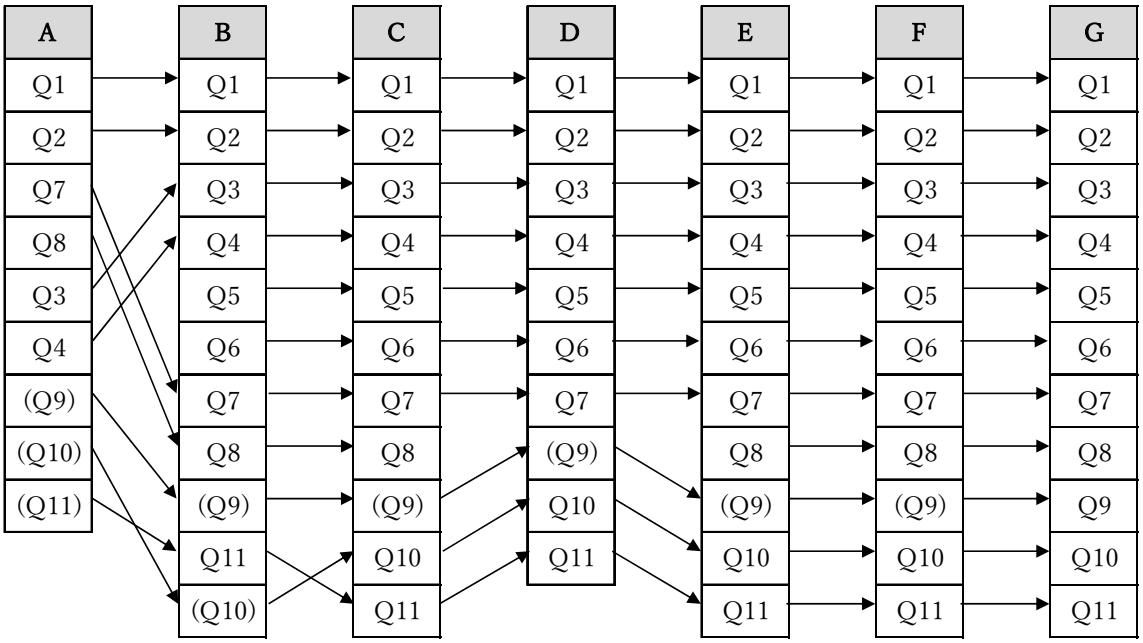


図 3.2：質問項目の変遷

すべてのインタビューの内容はボイスレコーダーで記録し、まず最初に、一語一句の正確な文字起こしを行った。その上で、内容に直接関係ない間投詞の削除、意味や文脈を変えない範囲での文章の修正（わかりやすい文章への書き換え、足りない語句の補充、など)、特定の個人や組織や製品に関わる情報の削除といった前処理を行い、分析用データとして整理して次のステップへ進んだ。

3.4. 帰納的テーマティック・アナリシス法による分析

前節で得られたインタビュー結果に対して、テーマティック・アナリシス法（TA: Thematic Analysis）を用いて分析した。TA は質的分析手法の 1 つであり、質的データの中にパターンを見出すための体系的なプロセスである。TA にはさまざまなバリエーションが存在するが、本研究では、Boyatzis（1998）による TA を土屋（2016）が解説した方法に沿って進めた。グラウンデッド・セオリー・アプローチ（GTA: Grounded Theory Approach; Strauss and Corbin 1998; 戈木 2013）などの厳格な“方法論”とは異なり、TA は、研究者の哲学的立ち位置に依存しない柔軟な“分析手法”として使用できる。また、TA には、既存の理論や仮説に基づく“演繹的分析手法”、生データからテーマを生成する“帰納的分析手法”、帰納的分析手法と演繹的分析手法を組み合わせた“ハイブリッドアプローチ”などの多様な分析手法があり、研究の目的に合わせて研究者自身が自由に手法を選択できるという特徴をもつ。本節では、生のインタビュー結果をインプットとした帰納的分析手法（帰納的 TA）を適用する。

土屋（2016）を参考に、以下の手順で帰納的 TA による分析を実施した。

1. **インタビュー・データを切片化する。**
   切片化の粒度（コーディングユニット）は、各質問項目に対する個々の回答とした。いわゆる、構造的コーディングである。ただし、1 つの回答の中で複数の話題に触れているものは、それぞれ別々の切片に分割した。
2. **切片データをコーディングする。**
   ここで、コーディングとは、生データである文字テキストデータに対して内容を代表する短い言葉（コード）をつける作業のこと。コーディングの品質向上のため、生データの内容を一度要約してからコードに変換した。
3. **関連するコードを階層的にまとめて、新たなコード（カテゴリー）を生成する。**
   その際、肯定的な例と否定的な例の両方を含んでいるのが、よいコード（カテゴリー）の条件とされる。
4. **コードをさらに抽象化して、その事象のパターンを説明するテーマを生成する。**
   コードの類似性と相違性の比較検討を繰り返すことで、1 つにまとめるべきコードのかたまりを見つける。

まず、インタビュー・データに対して、切片化（ステップ１）とコーディング（ステップ２）を行った（結果は付録 A1 に示す）。コーディングは筆者が単独で実施した。ただし、コードの信頼性を高めるために、最初のコーディングの後、元の切片に戻ってコードがデータの内容を正しく表しているかを確認し、不整合があった場合には再度コーディングを繰り返すという反復的な作業を行った。

続いて、得られた 90 件のコードを集約して 16 件のカテゴリーを生成し（ステップ３）、それらをさらに抽象化して全体を包括する 4 つのテーマにまとめた（ステップ４）。最初のテーマ「取り組むべき課題」は Q1 と Q2 と Q6 に、２番目のテーマ「知識活用の現状」は Q3 と Q4 と Q5 に、３番目のテーマ「データ活用の現状」は Q7 と Q8 と Q9 に、４番目のテーマ「将来への期待」は Q10 と Q11 にそれぞれ関連している。ここでも、ステップ２のコーディングのときと同様に、元のデータやコードに戻って内容の整合性をチェックし、不整合があった場合には再度集約やラベル付けを繰り返した。表 3.4 と表 3.5 に、生成したテーマとカテゴリーの一覧を示す。これらは、インタビュー対象者である製品開発部門のマネジャー・クラスの実務者が、現状のリスクマネジメントに対して抱いている率直な課題意識を反映していると考えられる。なお、各表の中の記述では、RM（リスクマネジメント）、PJ（プロジェクト）、DR（デザインレビュー）などの略語を使用している。

表 3.4：リスクマネジメントに対する実務者の課題意識（その 1）

テーマ	カテゴリー	コード
取り組むべき課題	リスクの早期検知	(A1) リスクの特定と予兆の早期検知が重要
		(G1) さまざまなリスクへの早期対策の重要性
		(A2) 未知のリスクを予測することの難しさ
		(B2-1) 見えている範囲外のリスク抽出の難しさ
	柔軟なリスク対応	(B1) 事前のリスク想定による柔軟な計画見直し
		(C1) RMで開発を安定化し環境変化に対応
		(B2-2) 自分では制御できないリスクの存在
		(C2) 複雑な因子の柔軟な制御が求められる
	対立の解消	(F6-1) 局所的な対立はあるが組織間の対立はない
		(B6) 管理側と現場側に危機感の温度差
		(D6) リスク対応への力の掛け具合に温度差
		(E6) 費用対効果と責任所在の認識ずれ
		(F6-2) リスク対応責任の押し付けに対する不満
		(G6) リソース配分での主観的判断や個人の思惑
		(C6) RMでの対立には人間の心理的要素が影響
	組織的な仕組み作り	(E1) 本来低減すべきリスクが放置されている現状
		(D2) RMの文化がないと継続的な改善が回らない
		(F2) RMを周知徹底することの難しさ
		(G2) リスクを組織的に是正することの難しさ
		(D1) 体系的な仕組み構築によるRMの底上げ
		(F1) RMへの組織的な取り組みの必要性
		(E2) リスク感度が高ければRMは難しくない
知識活用の現状	知識によるリスク特定	(E3-2) 同じパターンの失敗が繰り返し発生
		(A3) 過去の失敗経験や工夫点の再利用
		(B3) 有識者の知見活用による既知の問題の再発防止
		(D3) 有識者の知見を体系化して取り込む
		(F5-2) 未経験の分野では有識者の知見は当てにならない
		(G3) 既存顧客では知見は必要だが、新規は当てはまらない
		(A4) 新規開発では複数の視点からの気付きが必要
		(G5) 複数視点による客観的評価の必要性
	過去の知識の信頼性	(D5) 有識者の知見は概ね的確
		(B4) 有識者に依存し過ぎると見えなくなる部分もある
		(B5) 過去の知見は一般化しないと合致しない
		(C3) 過去の知見がかえって判断の柔軟性を欠落
		(C5) すべての知見には偏りが無意識的に混入
		(E5) 他人から聞いた情報をそのまま信じる傾向
		(F3-1) 有識者の提言が今の現場と乖離している可能性
		(F5-1) DR指摘における有識者の思い込みや偏り
		(D4) 有識者の経験をチェックリストで抽出
		(C4) 暗黙知を正確に形式知化するのは困難
		(G4) 複雑な事象での有識者の知見抽出の難しさ
	具体的な行動への誘導	(E4) RMにおける実践力や行動力の確立
		(F3-2) リスク対応が強制力をもつことの効果
		(E3-1) 有識者がたまに入っても効果なし
		(F4-1) 有識者の提言を受け取る側の問題
		(F4-2) 強制力をもつ指摘が的外れだと逆効果

表 3.5：リスクマネジメントに対する実務者の課題意識（その２）

テーマ	カテゴリー	コード
データ活用の現状	人間の気づきの支援	(A7-1) 予測モデルによるリスク予兆検知の有効性
		(A9-1) リスク特定で人間の気づきに依存している現状
		(A9-2) 未知のリスクに対するデータからの気づき
		(C9) 自分の感覚と突合せて新たな気づきを得る
	人間の判断の支援	(A8-2) 機械的に収集したデータから人間主体で考える
		(C7) 定性・定量両方の情報を使い人間が最終判断
		(D7) 有識者の経験を裏付けるバックデータの必要性
		(E7) データ分析による有識者知見の偏り補正
		(F7) 有識者の勘と経験の漏れをデータで補完
	合意形成や協調の促進	(A7-2) 客観的データによる議論の活性化
		(B7) 対立を埋める手段としてのデータ活用
		(G7) リスク対応の優先度付けでデータは有用
		(B8) 現場感覚と異なる基準を合意するのは難しい
		(C8) データがあっても理由が不明だと対策が打てない
		(E8) データが個人の感覚とずれたときの説得力
		(F8) 根本原因まで踏み込まないと受け入れられない
		(G9) 有識者知見と突き合わせた真因の深掘りが必要
	リスク要因のデータ化	(A8-1) PJ失敗要因の完全なデータ化は困難
		(A8-3) 人的要素の定量化の難しさ
		(G8-1) 定量化にはどうしても主観性が混入
		(G8-2) PJでは完全に同じ状況は起こり得ない
将来への期待	リスクの見える化	(A11-2) PJ状況を表す多様なデータの収集
		(A11-3) データ精度の確保は正しいPJ運営が前提
		(F10) 人間系で回っている部分がブラックボックス
	未知のリスクへの対応	(D11-1) 環境の変化の考慮して未知のリスクを洞察
		(D11-2) 過去に経験のないリスクにも踏み込むべき
	データ分析の高度化	(A11-1) リスク対策とその効果を予測モデルに反映
		(B11-1) 分析の専門家による第三者的リスク監査
		(C10-1) 因果関係を紐解いて施策効果を迅速に見極め
		(C11-1) 予測の高度化による意思決定支援への期待
		(G11) リソース制約を考慮したリスク優先度付け
		(G10-1) リスク優先度への組織内全一致の難しさ
	人材育成	(B11-2) 組織内でのリスク分析の基本スキルの育成
		(C11-3) 組織の視点では人材育成的側面も重要
		(G10-2) PJリーダーの力量不足を補う施策が必要
		(B10-1) 組織的なRMで皆が共通言語でしゃべれるように
		(B10-2) RMの基本を学べる現場向け実践ガイドの提供
	自律的なリスクマネジメントへ	(D10) RMが習慣付いて自然に回るような文化へ
		(E10) リスク感度を高めて一緒に拾うとPJは回り出す
		(E11) 良いチームはリスクに気付いたら自律的に動く
		(C10-2) 人間や組織の心理的要素をどう取り扱うか
		(C11-2) 人間的要素も含む統合マネジメントへ
		(F11-1) 強制力と負荷軽減のバランスが重要
		(F11-2) 最初のハードルを強制的に飛ばさせる

以下、テーマごとに分析結果の詳細を説明する。《》はテーマ、【】はカテゴリー、［］はコード、*イタリック体*はインタビュー・データを示す。

テーマ１：取り組むべき課題

リスクマネジメントで《取り組むべき課題》として、以下の４つが挙げられた。

第１は【リスクの早期検知】である。［さまざまなリスクへの早期対策の重要性］が叫ばれる一方、［未知のリスクを予測することの難しさ］も指摘されている。

- *リスクマネジメントは重要。特に、リスクの特定が一番大事であり、リスクが特定できていれば、それに対して対策も考えられる。リスクを特定して最初に手を打つということと、リスクの予兆が現れたのをとらえてすぐ手を打つということができないと、大規模プロジェクトではうまく行かない。(A1)*
- *リスクマネジメントでは、特に、リスクの抽出の部分が難しい。見えている範囲のリスクを洗い出すというのは、時間と経験もしくは適切な人がいれば可能だと思うが、そこから漏れているものをどうやって見つけ出すかというのはきわめて難しい。その漏れたリスクが、後々大きな影響になって出て来たり、本来それを計画に加えておくべきだったというような議論にもなる。(B2-1)*

第２は【柔軟なリスク対応】である。［事前のリスク想定による柔軟な計画見直し］が求められるが、［複雑な因子の柔軟な制御が求められる］という点で難易度が高い。

- *プロジェクトを進める中で、想像できないものが起きてしまうことはあり得るが、それでも、何も考えずに進めるよりも、考え得る範囲の中でどういったリスクがあって、それが起きたらどうするの？とか、やっぱり止めた方がいいんじゃない？とか、そういったことをプロジェクトの関係者と共有しておけば、計画の見直しなどにも結び付くと思う。(B1)*
- *リスクマネジメントの効果的な実践は、現時点では難しいと思っている。理由としては、顧客要求が変化する上に、かつ、開発進捗に与える影響要因が多い。これは、例えば、開発規模が大きくて非常に多くの開発チームが存在するとか、あるいは、色々なスキルをもったメンバーがいるとか、国内・海外の関係会社を含めて色々な人がいるとか、そういった色々な因子、要因があるので、そういったことを踏まえて計画遂行、予測、遅延対策といったことを柔軟に行う必*

要がある。これらの因子をパズルのように組み合わせてマネジメントして行かなければいけないので、そういった意味でも、技術的にも実際マネジメントを実行する上でも難しいと考えている。(C2)

第 3 は【対立の解消】である。[管理側と現場側に危機感の温度差]や[リスク対応への力の掛け具合に温度差]が存在する場合がある。このような[リスクマネジメントでの対立には人間の心理的要素が影響]していると考えられる。

- *リスクマネジメントにおける個人間や組織間の対立はあると思う。例えば、人的なリソースに関して、など。現場レベルで、プロジェクトリーダーやメンバーが「これぐらいの人数が必要なんじゃないか」と思っていても、組織の上の方は「その半分ぐらいで大丈夫だろう」とか、その危機感の認識の違いというのは現場側と上位組織側では乖離があると思う。結局、最終的にはお金に結びついていくのだろうが、人員の不足とか、スキルを持ってる人がいないとか、開発の期間が足りないとか、そういったことでの組織側と個人（プロジェクトリーダー）との対立は一般的にあると思う。(B6)*

第 4 は【組織的な仕組み作り】である。[本来低減すべきリスクが放置されている現状]があり、[体系的な仕組み構築によるリスクマネジメントの底上げ]が求められる。ただし、[リスクマネジメントの文化がないと継続的な改善が回らない]可能性がある。

- *リスクマネジメントは難しい。なぜなら、現状、文化がないから。まずプロジェクトが始まると、リスクの洗い出しを行うが、洗い出して終わりっていう現状がある。日々変化するリスクをウォッチして、それに追従して改善して、というところは習慣付いてないと回らないところがある。その習慣が付けば回って行って難しくなくなるのかもしれないが、現状は文化がないので回すのが難しい。(D2)*

テーマ 2 ：知識活用の現状

リスクマネジメントにおける《知識活用の現状》として、以下の 3 つが挙げられた。

第 1 は【知識によるリスク特定】である。[同じパターンの失敗が繰り返し発生]している現状がある。それに対して、[有識者の知見活用による既知の問題の再発防止]

が重要である。しかしながら、[未経験の分野では有識者の知見は当てにならない]との指摘もある。

- *過去、すごい火の車のプロジェクトに投入されたことがあり、人間関係とか、会社の仕組み的なところとか、生産管理部と品証の絡みとか、プロジェクトの実態みたいなものを経験できた。こんなことは他のプロジェクトでは起こらないだろうなと思っていたが、今でも、同じことがたまに起こる。その経験によって、こういうパターンのときはこうなるということが、なんとなく分かるようになった。(E3-2)*
- *有識者の知見は大変重要。過去の知見があるというのは、そういう失敗をしているということもあるし、そこを工夫したから上手く行ったということの両方があると思う。そういう知見をリスト化しておいて、次にプロジェクトが始まるときにそれを見ながらリスクを検討するというのはすごく有効だと思っている。未然防止リストという名前で、そういうリストを作っている。(A3)*

第2は【過去の知識の信頼性】である。[有識者の知見は概ね的確]であるという見解がある一方、[過去の知見がかえって判断の柔軟性を欠落]させる可能性や、[有識者の提言が今の現場と乖離している可能性」も指摘されている。その理由として、[暗黙知を正確に形式知化するのは困難]であることが影響していると考えられる。

- *理論と実際のバランスを持った有識者の知見は必要。ただし、あくまで必要条件の一つであり、有識者の知見があれば十分という訳でもないと考えている。有識者の知見というのは、ある意味、過去の話なので、場合によっては過去の知見が判断の柔軟性を欠落させるようなリスクもある。知見は知見として尊重しつつも、参考程度に留めておくべきかと思う。(C3)*
- *有識者の知見は、ある面では必要。有識者と言われているからにはきっと経験が豊富で、社内からも頼られているということなので、そういう人のコメントは、やはり、ある程度適切なものが多いと思う。ただし、それが全てではない。有識者からの提言にも足りない部分は絶対にあると思っている。例えば、大分経営寄りの方に行ってしまっていて、今の現場や現場の本当の声を知らない可能性がある。また、有識者には出来る人、優秀な人が多いが、現場のレベルはまちまちで、必ずしも皆が出来るわけではない。どうすればうまくやれるかと*

いう部分で、欠けている部分が生じる可能性がある。有識者の声を 100%信じるのは、危険だと思う。(F3-1)

- *リスクマネジメントにおける有識者の知見の活用は、ファースト・ステップとしてはトライするべきだと思う。ただし、有識者の知見を形式知化することは技術的な困難度が高いと思う。すなわち、有識者が経験したことというのは、具体的にどういう条件下で何をやってどのような効果があったのかということであり、そういったものを精度良く形式知化することは、すごく大変だと思う。有識者自身も、言葉にできる部分だけでなく、潜在意識というか自分でも気が付いてない、もしくは言語化できてない条件とかあった場合に、それを精度良く拾うというの難しい。(C4)*

第 3 は【具体的な行動への誘導】である。リスクを特定するだけでは意味がない。［リスクマネジメントにおける実践力や行動力の確立］が重要である。ただし、［強制力をもつ指摘が的外れだと逆効果］になることもある。

- *リスクには気付くのだけど、言いっ放しというのがリスクマネジメントの次の段階の難しさ。リスクに気付いて、チームで合意して、リスク管理票に入れて、それを管理するという流れや仕組みがない。普段からできるように躾られれば、実践力や行動力になっていくのだが。(E4)*

テーマ 3 ：データ活用の現状

リスクマネジメントにおける《データ活用の現状》として、以下の 4 つが挙げられた。

第 1 は【人間の気づきの支援】である。［リスク特定で人間の気づきに依存している現状］があり、データ分析の結果を［自分の感覚と突合せて新たな気づきを得る］ことができると有益である。

- *リスクマネジメントにおけるデータ活用について、今、データをうまく利用できているのは、予測モデルを使ってリスクの予兆をとらえるところ。予測モデルは結構当たるので、予測結果は客観的なデータとして扱うこともできると思っている。(A7-1)*
- *現状、過去にあった問題に対しては有識者の知見、新しい未知のものにはリス*

ク全体の観点リストを利用し、両方とも人間がみてここが怪しいねと言っている。それらに対して、有識者の知見とデータをうまく組み合わせて、例えば、データから今この辺が怪しいよとか言ってくれればうれしい。(A9-1)

第2は【人間の判断の支援】である。あくまで、[定性・定量両方の情報を使い人間が最終判断]することが重要。ただし、人間の判断には偏りもあるので、[有識者の勘と経験の漏れをデータで補完]できるとよい。

- *リスクマネジメントにおける判断の際、データは必要。人間の個人的な感覚だけで判断するのはまずい。ただし、データだけで判断するのではなくて、やはりプロジェクトリーダーが、データによる客観的な情報もちゃんと頭に入れながら、人間系でヒアリングした状況とかも踏まえて最終判断するべき。現状では、どうしても収集データの精度には限界があると思っている。ある程度データの精度には限界がある前提で、あくまで客観的な情報の一つとして関与するというスタンスでいる方が健全だと思っている。(C7)*
- *データによる客観的な判断は必要だと思う。データ分析の結果を見ると、人間のバイアスがかかって有識者が経験から「ここのところは怪しい」と決めつけていた部分に対して、それが外れたところも見えている。有識者に頼ったリスクアセスメントは、その人たちの経験の中でやってきたことがベースになって、リスクの有り無しや、その範囲や、対策を決めたりするのだが、その経験は10年以上も前のその人たちが現役バリバリだった頃の話というケースも多い。「こうすべきと言ってもらうのは有難いが、その頃とは納期も、お金も、人も、技術も何もかもが違う」ということを、よく若手の技術者から言われる。過去の蓄積された主観的な知識だけから施策を持って来られても困るので、ちゃんと現役レベルの状態を確認して、データを使って客観的に物事を決めて欲しい。(E7)*

第3は【合意形成や協調の促進】である。[リスク対応の優先度付けでデータは有用]であり、[対立を埋める手段としてのデータ活用]もある。その一方で、[データが個人の感覚とずれたときの説得力]は大きな課題であり、[データがあっても理由が不明だと対策が打てない]という問題もある。

- *リスクの重要度などを含めて、データの活用は必要だと思う。起こりやすいリスクだったら、当然、そこを優先的に処理しないといけない。でも、低コストで発生しても大した金額がかからないのであれば、ほっとけばいい、など。そういった優先度を付けるところで、データは必要となる。(G7)*
- *危ないということがデータの傾向として分かったとしても、そこを具体的にどうリカバーするかというところまで紐付けられないと、既に知っていることという受け取り方にしかされないような気がする。結局、そこで本当は何が問題になってるのかというところまで踏み込む必要がある。そうなると、データだけでは言い切れない。(F8)*

第4は【リスク要因のデータ化】である。[人的要素の定量化の難しさ]など、[プロジェクト失敗要因の完全なデータ化は困難]である。

- *プロジェクト体制とか、うまく稼働出来ているとか、チームワークがいいとか、ステークホルダーの人がちゃんとしているとか、要求仕様の決まり方が正しいとか、プロジェクトがうまく行くためには色々な要素が入っている。こういう要素がプロジェクトには多く含まれており、リスクの特定のところは、結局、人間が色々考えないと厳しいという気がする。機械的に取れるものは取って、プラス人間が考えるというのが一番よいような気がする。(A8-2)*
- *データの難しさは、プロジェクトの状況は必ずしも同じではないということ。昔データを取ったときの状況と今の状況とは、技術的にも、組織の体制的にも変わっているため、そのデータが示すことが必ずしも正しいとは言えない。そこは想定していくしかないのだが、新しい事象があったら変化するし、100%同じような状況は起こり得ないので、そういうところが難しい。しかし、そういうところが、リスクの本質でもある。(G8-2)*

テーマ4：将来への期待

リスクマネジメントに対する《将来への期待》として、以下の5つが挙げられた。

第1は【リスクの見える化】である。[人間系で回っている部分がブラックボックス]になっており、リスクの見える化が必要である。その際、正しいデータが必要となるが、[データ精度の確保は正しいプロジェクト運営が前提]である。

- *一つは、人間系で回ってるところが多いということ。もう一つは、リスクの見える化が現状出来ていないし、結構難しいのかなと思う。特にソフト系だと外からは本当に何も見えない。建築系で作った建物が倒れたなんてことは、まず日本じゃありえないけど、システム開発では頻繁にあったりする。ここの根本的な違いっていうのは何かな？っていうのが、その本質的な難しいところにつながるのかなと思う。見えづらい、ブラックボックス、でも中で色々動いている、というところが難しいのだと思う。(F10)*
- *データの精度も課題。データの精度が良くないと、結局、そのリスクが高いか低いかの判断を誤ってしまう。プロジェクト状況が正しくて、かつ、データがちゃんと出てきているということをセットで見た方がよい。プロジェクトが混乱していると、そもそも出てくるデータがうそかも知れない。ちゃんと進捗管理がされているとか、プロジェクトが正しく運営されているということと、データの精度が正しいということがペアで表現できないと、判断を間違えてしまう。(A11-3)*

第2は【未知のリスクへの対応】である。[過去に経験のないリスクにも踏み込むべき」であり、[環境の変化の考慮して未知のリスクを洞察]できるとよい。

- *今まであったことならば、経験や過去のデータの分析などで未然に抽出して防げると思うが、今まで経験したことのない事象やリスクは、なかなか過去のものから生み出しにくいと思う。環境や時代の変化などの情報から、未来に何がリスクとして出てくるのかが分かるというのが、リスクマネジメントの高度化なのだと思う。(D11-1)*

第3は【データ分析の高度化】である。[予測の高度化による意思決定支援への期待]がある。[リソース制約を考慮したリスク優先度付け]を行ったり、[因果関係を紐解いて施策効果を迅速に見極め]たい。

- *プロジェクト・リスクマネジメントの本質的な課題は、色々な事象があり因果関係が複雑な中で、何か対策を打ったときの費用と効果がどの位の大きさなのかを見極め、判断や対策を迅速に実施することが求められることだと思う。因果関係の解きほぐしと、費用対効果がリーズナブルかどうかという評価は、非*

常にパラメータも多いし、単純な式では表せない複雑な要素があるので、それをどうマネジメントするか、コントロールして行くかという部分が本質的な課題だと思う。(C10-1)

- *組織のリソースが決まっている中で、どのリスクから手を付けるのかを何らかの方法で順序付けできるとよい。組織として優先度付けの部分で失敗する場合が多いように思える。限られたリソースの中でどう優先度付けするかというのが、プロジェクトリーダーの力量でもある。今は人間系でやっているが、データを集めて、このパターンだとこのリスクが重要というのが自動的にでるとよい。(G11)*

第 4 は【人材育成】である。［PJ リーダーの力量不足を補う施策が必要］である。［組織的なリスクマネジメントで皆が共通言語でしゃべれるように］なることが望ましい。

- *プロジェクトリーダーの力量でリスクは変わってくると思う。一番重要なのはプロジェクトリーダーの力量だと思っている。力量が劣る、スキルが低いプロジェクトリーダーのときは、それをカバーするものが必要。プロジェクト開始前の体制を作るところをどうするかなど、どこからリスクマネジメントをするか、誰がリスクマネジメントをするかを考えないといけない。標準品でないソフトウェア開発だと、特に重要だと思う。ソフトウェア開発は、残念ながら、人の力量に依存するので。(G10-2)*
- *リスクマネジメントの教育は重要。一般論のリスクマネジメントというところと、組織でやっていることの周知徹底。そこがないと共通の言語でしゃべれない気がするので、教育と組織のプロセス周知はとても重要だと思う。それから、定性情報と定量情報をバランスよく使って、「これ危ないんじゃないの？」とか議論の材料にする活動を続けていくことが重要。教育をやって、プロセスを作って、それを周知徹底して、やった結果はどうだったかを振り返るという、組織的なリスクマネジメントの PDCA（Plan-Do-Check-Act）を、効果がすぐに出なくても粛々とやっていくことが重要だと思う。(B10-1)*

第 5 は【自律的なリスクマネジメントへ】の期待である。[良いチームはリスクに気付いたら自律的に動く]。[リスクマネジメントが習慣付いて自然に回るような文化へ]と変えて行きたい。

- *リスクがあって、それに気付いていても、自分たちの課題として見ていないことが問題。他の人の課題だから別に関係ないと思ってしまうと、結局、回り回って、自分たちの工程にしわ寄せが来たりすることもある。弱いチームは、他人事になっている。逆に、良いチームは、マネジャーや PMA（Project Management Assistant）がそこを全部引き取って、自分で動いて調整して、全体を同じ方向に向かせようとする。最終的には人に依存するが、そういう経験なり躾ができているかどうかが、すごく大きいと思う。(E11)*
- *相反しているが、ある程度の強制力は必要な一方、なるべく楽にして負荷をあまりかけないという、その両方のバランスが重要。優しくしていると絶対にやらないが、無理に強制して何か別の本質的な作業が滞ってしまうのも避けたい。リスクマネジメントの施策が自分たちの役に立っていると思ってもらい、リスクを正直に申告できる雰囲気にしたい。そういう姿勢がうまく伝わると、皆協力してくれると思うが、トップダウンで施策を展開すると、なかなかそうはならないところが難しい。(F11-1)*
- *リスクマネジメントの一番難しいところは、文化を作るところだと思う。皆、リスクマネジメントが必要だというのは頭では認識していると思うが、それを習慣づけてできるようにならないと、多分回らない、うまく行かない。そこがリスクマネジメントの本質、すなわち、一番必要なことだと思う。(D10)*

表 3.6 は、抽出した 4 つのテーマ、16 件のカテゴリーがリスクマネジメント・プロセスのどこに対応するかを示したものである。リスクの特定、リスク分析、対応計画策定が主にリスクマネジメント側で実行されるプロセスであるのに対して、対応策の実行は、プロジェクトの計画・実行・コントロールに関連するさまざまな調整を伴うプロジェクトマネジメント側のプロセスとみなすことができる。すなわち、リスクマネジメントにおける現場の課題の多くは、プロジェクトマネジメントとの境界領域で発生していることがわかる。

表 3.6：抽出したカテゴリーとプロセスの対応付け

プロセス / テーマ	リスクマネジメント		プロジェクトマネジメント
	リスクの特定	リスク分析・対応計画策定	対応策の実行
取り組むべき課題	・リスクの早期検知	・柔軟なリスク対応	・対立の解消 ・組織的な仕組み作り
知識活用の現状	・知識によるリスク特定 ・過去の知識の信頼性		・具体的な行動への誘導
データ活用の現状	・人間の気づきの支援 ・リスク要因のデータ化	・人間の判断の支援	・合意形成や協調の促進
将来への期待	・リスクの見える化 ・未知のリスクへの対応	・データ分析の高度化	・人材育成 ・自律的なRMへ

3.5. まとめ

本章では、まず、製品開発組織における一般的なリスクマネジメント・プロセスについて概観した上で、製品開発部門のマネジャー・クラスの実務者 7 名へのインタビュー調査を実施し、そのインタビュー・データを帰納的テーマティック・アナリシス法（TA: Thematic Analysis）を用いて分析した。その結果、11 件の質問項目に対して 90 件のインタビュー・データの切片およびコードが得られ、それが最終的に 16 件のカテゴリーと 4 つのテーマにまとめられた。

帰納的 TA による分析結果は、製品開発におけるリスクマネジメントの重要性と困難性の両面を含んでおり、先行研究におけるリスクマネジメントの実践上の課題を裏付けるものである。また、多くの課題はプロジェクトマネジメントとリスクマネジメントの境界領域で発生しており、単にリスクマネジメントのプロセスや技法だけの問題ではなく、対立の解消や合意形成などの人間的な要素がきわめて大きく影響していることがわかった。

第4章 リスクマネジメントの困難性の理論的考察

4.1. はじめに

本章では、前章において確認されたリスクマネジメントの“実践上の難しさ”[6]に対して、取引コスト理論およびプロスペクト理論に基づく理論的考察を示し、インタビュー・データの分析結果との対応付けにより仮説の妥当性を検証する。4.2 節では、リスクマネジメントと他のマネジメントの違いに着目することにより、リスクマネジメントの本質的な課題について考察する。4.3 節では、リスクマネジメントにおける利害関係者の限定合理性と取引コストについて深掘りする。4.4 節では、人間がもつ認知バイアスの影響についてプロスペクト理論を用いて考察する。4.5 節では、インタビュー・データの分析結果との対応付けにより、本章における仮説の妥当性を検証する。

4.2. リスクマネジメントの本質的な課題

前章のインタビュー・データの分析結果から、製品開発組織におけるリスクマネジメントの課題の多くはプロジェクトマネジメントとリスクマネジメントの境界領域で発生しており、かつ、単にリスクマネジメントのプロセスや技法だけの問題ではなくヒューマン・ファクターの影響がきわめて大きいことがわかった。このようなリスクマネジメントの“実践上の難しさ”に対応するには、表面に見えている課題だけでなく、その根本原因となる本質的な課題を明らかにする必要がある。

本節では、まず、リスクマネジメントと他のマネジメントの違いについて考察する。リスクマネジメントはプロジェクトマネジメントの一部だが、リスクという確率的な事象を扱っている点で他のマネジメントとは大きく性質が異なっている。すなわち、リスクは常に存在しているが、その存在が最初から認識されているとは限らないし、仮に認識されていたとしても必ずしも顕在化するとは限らないし、また、プロジェクト期間中ずっと同じ状態であるとも限らない。もし、リスクが“確定的”であり、必ず発生することが最初からわかっているならば、リスクへの対応は“タスク”としてあらかじめプロジェクト計画に組み込まれているべきであり、そもそもリスクマネジメン

---

[6] リスクマネジメントの標準的なプロセスや技法などの“理想形”と開発現場における“実践”とのギャップ.

トは不要なはずである。リスクの発生が“確率的”であり不確実性を伴うがゆえに、リスクマネジメントは必要なのだと言える。

プロジェクトにおいてどのタスクを実行すべきかの意思決定は、一般化して考えると、費用対効果の問題に帰着すると考えられる。すなわち、タスクの実行によって得られるトータルのベネフィットが実行にかかるコストを上回っているならば、そのタスクはプロジェクトでの実行の候補となる。しかしながら、大多数（ほぼすべて）のプロジェクトにおいては、時間やコストやリソースに関する“制約”が存在する。すべての候補タスクをプロジェクト内で実行することが不可能な場合には、“制約”の範囲内で必要なタスクを選択するというトレードオフ判断が必要となる。ここで、トレードオフとは、一方を追求すれば他方を犠牲にせざるを得ないという状態や関係を意味する。

リスクへの対応策など、不確実性を伴う事象に関する意思決定は、通常のタスク実行のトレードオフ判断と比べて、一層複雑かつ困難になることが予想される。そこで、本博士論文においては、リスクマネジメントの“実践上の難しさ”につながっている本質的な課題を「限られた時間やコストやリソースの中で、不確実さを伴うさまざまな事象や状態に対して、トレードオフを含む意思決定を適切なタイミングで実施することの難しさ」と仮定する。この不確実な状況下におけるトレードオフを伴う意思決定問題は、図 4.1 のようなデシジョンツリーで表現できる。これはある意味、最も単純な形でのリスクマネジメントの意思決定モデルとみなすことができる。ただし、より複雑にみえる現実のプロセスもこの単純化したモデルの組み合わせでかなりの部分を説明できる可能性があると考えている。図 4.1 より、リスク R へのコスト C の対応策を実行する場合の期待利得は、R が顕在化する確率を P_1、その際の損失を L_1 として $E_1 = -C - P_1L_1$、対応策を実行しない場合の期待利得は、R が顕在化する確率を P_2、その際の損失を L_2 として $E_2 = -P_2L_2$ と計算できるので、$E_1 > E_2$ すなわち $P_2L_2 - P_1L_1 > C$ のとき、リスク R への対応策を実行する方が期待利得が高い、つまりリスク R への対応策を実行する方が好ましいと判断できる。これは、既存のプロジェクト計画に対する新たなタスクの追加を意味し、しばしば元々の計画に含まれていたタスクの再調整を必要とする。

上記のような合理的な意思決定が可能となるには、まず「リスクの定量的評価が信頼できる」必要があり、その上で「人間は常に期待利得を高めるように行動する」と

いう人間の行動基準の仮定が必要となる。しかしながら、これまでのさまざまな事例が示すように、リスクマネジメントの実践において人間は必ずしも合理的に行動しているとは限らない。実際のプロジェクトでは、リスクを認識していても対応策をとらずにそのまま放置し、リスクが顕在化して問題（イシュー）になってから慌てて“火消し”に追われるというような状況がしばしば生じている。リスクマネジメントの実践におけるある種の行動パターンや方向性は、単にリスクの定量的評価の困難性などの技術的な要因だけで説明できるものではなく、人間の心理的特性など、異なる切り口からの分析も必要となる。そこで、本研究では、利害関係者間の取引コスト、および人間がもつ認知バイアスの影響に着目し、それらがリスクマネジメントにおける誤った判断や行動をさらに増幅する可能性があることを示す。

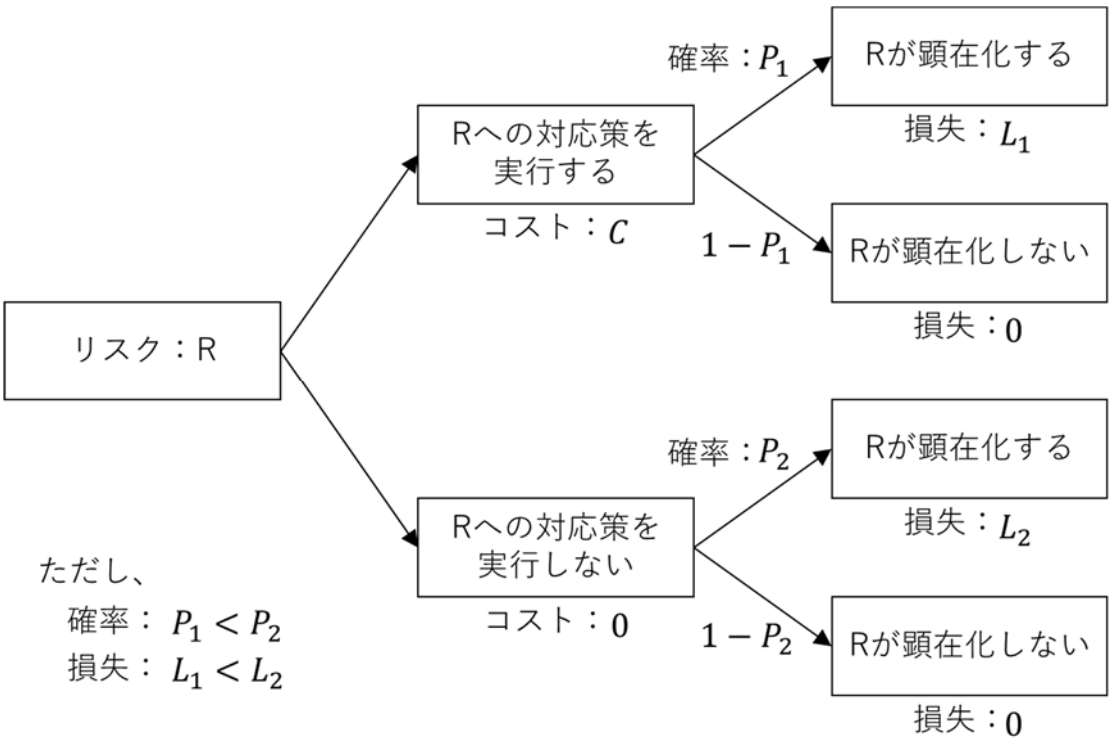


図 4.1：リスクマネジメントの単純化した意思決定モデル

4.3. 取引コスト理論による考察

取引コスト（transaction cost; Williamson 1981）とは「限定合理的な人間同士が取引をする際に発生するコスト」（菊澤 2016）である。ここで、限定合理性は「人間の情報収集・処理・伝達能力は限定されているため、限定された情報の中で意図的に合理的にしか行動できないこと」（菊澤 2016）を意味する。限定合理的な人間同士が取引をすると、互いに相手の不備につけ込んで機会主義的に自分に有利に取引を進めようとする可能性があるため、相互にだまされないように駆け引きが起こるなど、取引上の“手間”、すなわち取引コストが発生する。

リスクマネジメントにおいて、リスク対応策の実行はプロジェクトや組織の当初計画（ベースライン）に対して新たな作業を付加するものである。このような、既にある状態が選択されてしまっており、その状態からさらに別の状態へ移行するかどうかという選択的状況は、取引コスト理論における“非ゼロベースのケース”と呼ばれる（菊澤 2016, p.71-73）。非ゼロベースのケースでは、状態遷移によって影響を受ける各利害関係者の間で“取引”が発生する可能性がある。例えば、難度の高いプロジェクトにおいてスキルの高いメンバーを増員することは、プロジェクトリーダーにとっては必要なリスク対応策だろう。一方、組織の上位管理者はプロジェクトの難易度や優先度に関する異なる意見から、スキルの高いメンバーを別のプロジェクトにアサインしたいと考えているかもしれない。最終的な決定は利害関係者間、すなわちプロジェクトリーダーと組織の上位管理者の間の“取引”によって決まる。各利害関係者はそれぞれ独立した意思決定主体であり、かつ、一人の人間がプロジェクトや組織のすべての情報を知ることは実質困難であるため、“限定合理的”であると考えられる。限定合理的な人間同士の取引では、必ず“取引コスト”が発生する。すなわち、図 4.1 の意思決定モデルにおいて、取引コストを T とすると、リスク R への対応策を実行すべき条件は $P_2L_2 - P_1L_1 > C + T$ となり、リスク対応策の実行を回避する（元々の計画をなるべく変更しない）方向にシフトする。特に、チームやプロジェクトをまたがったリソース調整やプロジェクトの Go/Kill 判断など、利害関係者間の衝突（コンフリクト）が大きい場合には、その調整において多大な取引コストが発生することが予想され、リスク対応策の実行は一層困難となる。すなわち、非ゼロベースの状況においては、取引コストが大きいと、たとえ現状が非効率的であっても変化しない方が合理的であると

いう個別効率性と全体効率性が一致しないような現象が現れて、組織は「合理的に“失敗”するという不条理」（菊澤 2016）に陥る可能性がある。Kutsch and Hall（2010）の分類のうち、“非決定性”（undecidability）および“タブー”（taboo）には、この取引コストが強く影響している可能性がある。

取引コスト理論に基づく分析・考察は、リスクマネジメントの実践上の難しさへの効果的な対応方法について有益な示唆を与えてくれる。取引コスト理論によると、取引コストの大小には、資産特殊性、不確実性、取引頻度といった取引状況の特徴が関係している。このことから、リスクに関する個人の経験や知識の共有、不確実性の客観的な評価、柔軟な意思決定が取引コストの削減につながることが導かれる。さらに、すべての利害関係者は限定合理的であるという前提に立ち、組織の非効率性や不合理性を持続的に排除できるような「批判的合理的構造」（菊澤 2009）が備えられていることも重要である。

4.4. プロスペクト理論による考察

続いて、人間がもつ認知バイアスの影響について考察する。たとえ限定合理性が緩和されたとしても、不確実性が残る状況では必ずしも合理的な判断がなされるとは限らない。不確実性下における意思決定モデルの 1 つとして、プロスペクト理論（Kahneman 2011=2014; 友野 2006）がある。プロスペクト理論では、人間が確率や頻度についての判断を下すときに用いる一般的なヒューリスティクスを前提として、確率に対する人間の反応が線形でないこと、すなわち合理的判断からのバイアスが生じることを説明している。

プロスペクト理論から導かれる人間の価値評価の重要な特性として、“参照点依存性”および“損失回避性”がある。“参照点依存性”とは「価値は、参照点からの変化またはそれとの比較で測られ、絶対的な水準が価値を決定するのではない」（友野 2006）という性質である。また、“損失回避性”とは「損失は、同額の利得よりも強く評価される、つまり、同じ額の損失と利得があったならば、その損失がもたらす『不満足』は、同じ額の利得がもたらす『満足』よりも大きく感じられる」（友野 2006）という性質である。図 4.2 は、プロスペクト理論における価値関数のグラフを表している。“参照点依存性”によると、グラフの原点が参照点となり、価値（＝満足度）は参照点からの

相対的な利得または損失によって決まる。“損失回避性”によると、たとえ参照点からの利得と損失が同じであっても、価値に変換すると、プラスの価値よりマイナスの価値の方が大きいという非対称性を示す。

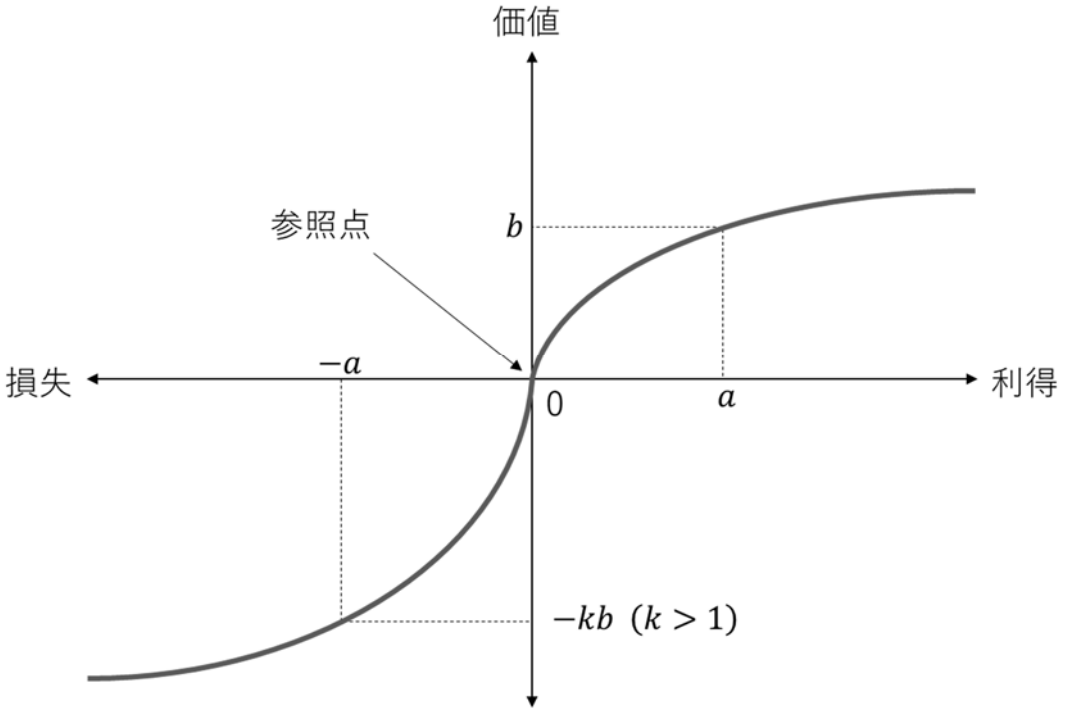


図 4.2：プロスペクト理論の価値関数

これらを再び図 4.1 の意思決定モデルに当てはめてみると、リスク対応策を含んでいない当初計画（ベースライン）が参照点となり、リスク R への対応策を実行すべき条件である $P_2L_2 - P_1L_1 > C + T$ の左辺は「利得」、右辺は「損失」とみなすことができる。損失は同額の利得よりも k 倍（$k > 1$）大きく評価されると仮定すると、同条件は $P_2L_2 - P_1L_1 > k(C + T)$ となり、リスク対応策の実行を回避する（元々の計画をなるべく変更しない）方向にさらにバイアス（偏り）がかかる。Kutsch and Hall（2010）の分類のうち、“非話題性”（untopicality）および“態度の保留”（suspension of belief）は、人間の認知バイアスの影響を強く受けている可能性がある。

プロスペクト理論によると、認知バイアスの元となるヒューリスティクスによる判断は“二重プロセス理論”（Kahneman 2011=2014; McAfee and Brynjolfsson 2017=2018; 友野 2006）におけるシステム I によって直感的に行なわれていると想定される。ここで、

二重プロセスとは人間が持っている２つの情報処理システムのことであり、１つは、直感的、連想的、迅速、自動的、感情的、並列処理、労力がかからない等の特徴をもつシステムで、システムⅠと呼ばれ、もう１つは、分析的、統制的、直列処理、規則支配的、労力を要するといった特徴で表わされるシステムであり、システムⅡと呼ばれる。システムⅠとシステムⅡは相補的な関係にあり、システムⅠが問題に対して素早く答えを見付ける一方､システムⅡはシステムⅠが素早く決定したことを監視して、必要に応じてそれを承認したり、修正や変更を加える役割をもつ。しかしながら、システムⅡがシステムⅠを必ずしも修正できない場面が数多く存在し、その代表的なものの１つがリスクマネジメントにおける意思決定であると考えられる。認知バイアスに起因するリスクマネジメントの難しさに対応するには、システムⅠとシステムⅡをいかに効果的に連携させるかが重要となる。

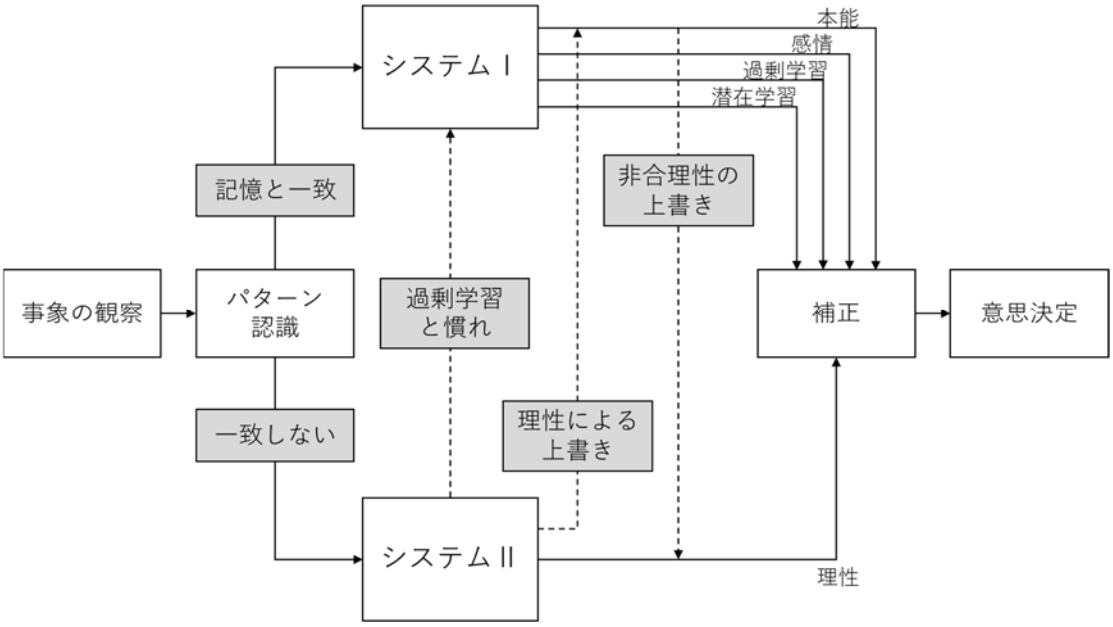


図 4.3：二重プロセス理論に基づく意思決定モデル

（出典：Croskerry 2009 を参照し筆者修正）

図 4.3 は、Croskerry（2009）が示した二重プロセス理論に基づく意思決定のモデルである。元々は医療分野における診断推論のモデルだが、リスクマネジメントの意思決定にも当てはまると考えられる。人間がある事象を観察すると、まずパターン認識

が行われ、過去の記憶と一致した場合にはシステムⅠが優勢になり、一致しない場合にはシステムⅡが優勢になる。システムⅠは、本能、感情、過去の学習結果などを並列処理して直観的な判断結果を生成する一方、システムⅡは、理性に基づいて合理的な判断結果を出力し、相互のインタラクションによる補正を経て最終的な意思決定が行われる。ただし、過剰学習や慣れの影響でシステムⅡからシステムⅠに制御が移行したり、システムⅠの判断をシステムⅡが取り消したり、逆にシステムⅡの判断をシステムⅠが上書きするなど、2つのシステムは頻繁に切り換わる。Croskerry（2009）は、最適な意思決定を行うためには、2つのシステムを適切なバランスで“混合”することが重要であると述べている。

二重プロセスに対する別のアプローチとして、上記のようにシステムⅡを意図的に起動してシステムⅠをモニタリングするのではなく、システムⅠがもつさまざまな特性をむしろ積極的に活用するという方法も考えられる。このアプローチは、“ナッジ”（nudge）と呼ばれる。ナッジには「軽く肘でつつく」という意味があるが、ここでは「選択を禁じることも経済的なインセンティブを大きく変えることもなく、人々の行動を予測可能な形で変える選択アーキテクチャーのあらゆる要素」（Thaler and Sunstein 2008=2009）と定義される。良い選択アーキテクチャーを設計するための原則として、Thaler and Sunstein（2008=2009）は以下の6項目を挙げている。

1. デフォルト

   デフォルト・オプションをうまく活用する。与えられた選択にデフォルト・オプションがあると、人間はそれが自分にとって良いかどうかに関係なく現状維持バイアスからその選択肢を無意識的に選ぶ傾向がある。

2. エラーを予測する

   人間は必ずミスをするという前提のもと、エラーをあらかじめ予測してその予防手段をシステムの設計に埋め込む。（同様の仕組みは、工場などの製造ラインの改善においてはポカヨケと呼ばれている。）

3. フィードバックを与える

   人間のパフォーマンスを向上させる最善の方法はフィードバックを与えること。操作がうまくできているかミスをしているかの適切なフィードバックを与える。

4. マッピング（選択と幸福度の対応関係）を理解する

   例えば、数値情報をなるべく利用実態に即した単位に置き換えるなど、さまざまな選択肢に関する情報を理解しやすく提示し、適切な選択肢を選ぶことが可能となるように仕向ける。

5. 複雑な選択を体系化する

   選択肢を絞り込むための適切な手段を提供する。選択肢の数が少なくてよく理解している場合には、すべての選択肢の特性を調べて必要に応じてトレードオフを検討することもできるが、選択肢が多くなると“属性値による排除”などの単純化戦略が採用される傾向がある。

6. インセンティブ

   「誰が利用するか」「誰が選ぶか」「誰がコストを支払うか」「誰が便益を得るか」を考慮した上で、適切なインセンティブの設計を行う。

4.5. TA のハイブリッドアプローチによる仮説の妥当性検証

図 4.4 は、プロジェクト・リスクマネジメントの実践上の難しさに対する仮説を図示したものである。すなわち、プロジェクト・リスクマネジメントの標準的なプロセスや技法などの“理想形”と、現場におけるリスクマネジメントの“実践”との間には大きなギャップがあり、それがリスクマネジメントの形骸化など、さまざまな実践上の

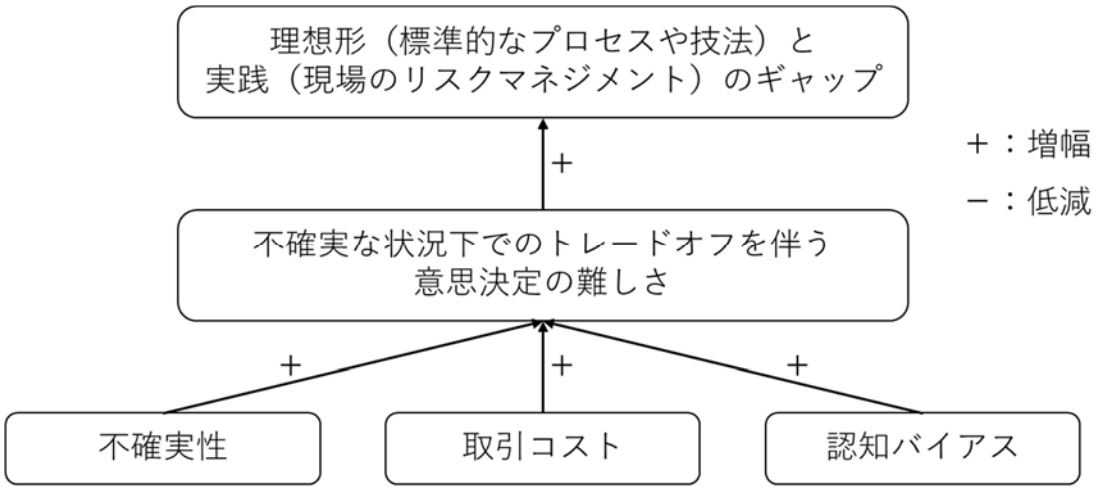


図 4.4：プロジェクト・リスクマネジメントの本質的な課題

難しさにつながっていた。そこで、本博士論文では、「限られた時間やコストやリソースの中で、不確実さを伴うさまざまな事象や状態に対して、トレードオフを含む意思決定を適切なタイミングで実施することの難しさ」、すなわち「不確実な状況下でのトレードオフを伴う意思決定の難しさ」をプロジェクト・リスクマネジメントの本質的な課題とみなし、さらに取引コストと認知バイアスの影響がその難しさを増幅していると仮定した

上記仮説について、帰納的 TA の分析結果（3.4 節）を用いたハイブリッドアプローチにより妥当性を検証した。ここで、TA のハイブリッドアプローチとは、帰納的分析によって得られた結果を他の理論や仮説を用いて再分析あるいは解釈する方法である。帰納的 TA により抽出された 90 件のコードには、大きく分けて、リスクマネジメントの"効果"に関するものと、リスクマネジメントの"難しさ"に関するものが混在する。本分析には主に後者が必要なため、まず、リスクマネジメントの"難しさ"に関連する 41 件のコードを選別した。続いて、各コードを（1）取引コストの影響あり、（2）認知バイアスの影響あり、（3）その他、の 3 パターンに分類した。取引コストの影響と認知バイアスの影響には相互作用[7]があり、どちらの影響が強いかを完全に切り分けることは難しいが、もし、最初の認識が偏っているならば認知バイアスの影響、(仮に認識が正しくても)他者を説得できないことが問題ならば取引コストの影響とみなした。

分類の結果を表 4.1 に示す。（1）取引コストの影響あり（TC）は 14 件、（2）認知バイアスの影響あり（CB）は 17 件、（3）その他は 10 件となった。リスクマネジメントの"難しさ"に関するコード全 41 件のうち、31 件（75.6%）は取引コストまたは認知バイアスの影響を受けており、本博士論文の仮説および理論的考察を裏付けていると考えられる。（3）その他に属する残りの 10 件のコードは、主に、リスクマネジメントの組織ルーティン化（OR）、または、未知のリスクへの対応（UK）のいずれかに関連していた。ただし、前者には、柔軟な意思決定を必要とする部分との切り分けが含まれるため、上記の本質的な課題とも密接に関連している。一方、後者は知識やデータだけで解決する問題ではなく、別の切り口も含めた多面的なアプローチが必要となる。

[7] 取引コストを意識することで認知バイアスが増幅したり、逆に、認知バイアスがあることで取引コストが増大するケースは、しばしば生じる.

表 4.1：リスクマネジメントの実践上の難しさとその分類

テーマ	カテゴリー	コード	TC	CB	OR	UK
取り組むべき課題	リスクの早期検知	(A2) 未知のリスクを予測することの難しさ				○
		(B2-1) 見えている範囲外のリスク抽出の難しさ				○
	柔軟なリスク対応	(B2-2) 自分では制御できないリスクの存在	○			
		(C2) 複雑な因子の柔軟な制御が求められる	○			
	対立の解消	(B6) 管理側と現場側に危機感の温度差	○			
		(D6) リスク対応への力の掛け具合に温度差	○			
		(E6) 費用対効果と責任所在の認識ずれ	○			
		(F6-2) リスク対応責任の押し付けに対する不満	○			
		(G6) リソース配分での主観的判断や個人の思惑	○			
		(C6) RMでの対立には人間の心理的要素が影響	○			
	組織的な仕組み作り	(E1) 本来低減すべきリスクが放置されている現状			○	
		(D2) RMの文化がないと継続的な改善が回らない			○	
		(F2) RMを周知徹底することの難しさ			○	
		(G2) リスクを組織的に是正することの難しさ	○			
知識活用の現状	知識によるリスク特定	(E3-2) 同じパターンの失敗が繰り返し発生		○		
		(F5-2) 未経験の分野では有識者の知見は当てにならない		○		
		(G3) 既存顧客では知見は必要だが、新規は当てはまらない		○		
	過去の知識の信頼性	(B4) 有識者に依存し過ぎると見えなくなる部分もある		○		
		(B5) 過去の知見は一般化しないと合致しない		○		
		(C3) 過去の知見がかえって判断の柔軟性を欠落		○		
		(C5) すべての知見には偏りが無意識的に混入		○		
		(E5) 他人から聞いた情報をそのまま信じる傾向		○		
		(F3-1) 有識者の提言が今の現場と乖離している可能性		○		
		(F5-1) DR指摘における有識者の思い込みや偏り		○		
		(C4) 暗黙知を正確に形式知化するのは困難		○		
		(G4) 複雑な事象での有識者の知見抽出の難しさ		○		
	具体的な行動への誘導	(E3-1) 有識者がたまに入っても効果なし			○	
		(F4-1) 有識者の提言を受け取る側の問題		○		
		(F4-2) 強制力をもつ指摘が的外れだと逆効果		○		
データ活用の現状	人間の気づきの支援	(A9-1) リスク特定で人間の気づきに依存している現状		○		
	合意形成や協調の促進	(B8) 現場感覚と異なる基準を合意するのは難しい	○			
		(C8) データがあっても理由が不明だと対策が打てない	○			
		(E8) データが個人の感覚とずれたときの説得力	○			
		(F8) 根本原因まで踏み込まないと受け入れられない	○			
	リスク要因のデータ化	(A8-1) PJ失敗要因の完全なデータ化は困難				○
		(A8-3) 人的要素の定量化の難しさ		○		
		(G8-1) 定量化にはどうしても主観性が混入		○		
		(G8-2) PJでは完全に同じ状況は起こり得ない				○
将来への期待	リスクの見える化	(A11-3) データ精度の確保は正しいPJ運営が前提			○	
		(F10) 人間系で回っている部分がブラックボックス				○
	データ分析の高度化	(G10-1) リスク優先度への組織内全一致の難しさ	○			

※ TC は「取引コストの影響あり」、CB は「認知バイアスの影響あり」、OR は「組織ルーティン化に関連」、UK は「未知のリスクに関連」を示す。

4.6. まとめ

本章では、リスクマネジメントの本質的な課題を「限られた時間やコストやリソースの中で、不確実さを伴うさまざまな事象や状態に対して、トレードオフを含む意思決定を適切なタイミングで実施することの難しさ」と捉えて、デシジョンツリーを用いて不確実な状況下での意思決定のトレードオフ関係を分析した。しかしながら、人間は必ずしも合理的に判断し行動しているとは限らず、利害関係者間の取引コストや人間がもつ認知バイアスの影響についても考慮する必要がある。

“取引コスト”は、限定合理的な人間同士の取引（≒駆け引き）において発生するコストであり、リスクマネジメントにおける意思決定の硬直性に影響している。取引コストの大小は、資産特殊性、不確実性、取引頻度といった取引状況の特徴に関連し、特に、チームやプロジェクトをまたがったリソース調整やプロジェクトの Go/Kill 判断など、利害関係者間の衝突（コンフリクト）が大きい場合には取引コストも増大する。限定合理性および取引コストの観点においてリスクマネジメントの実践上の難しさに対応するには、リスクに関する認識や判断、知識を組織的に共有し、その非効率性や不合理性を継続的に排除する仕組みが必要となる。

一方、人間がもつ認知バイアスの影響はプロスペクト理論を用いて分析できる。プロスペクト理論では、人間が確率や頻度についての判断を下すときに合理的判断からバイアス（偏り）が生じやすくなる状況を理論的に説明している。プロスペクト理論から導かれる人間の価値評価の重要な特性として“参照点依存性”および“損失回避性”があり、これらもまた、リスクマネジメントの硬直性に大きな影響を与えている。

認知バイアスの元となるヒューリスティクスは、“二重プロセス理論”におけるシステムⅠによって直感的に行なわれている。二重プロセスとは人間が持っている２つの情報処理システムのことであり、直感的なシステムⅠと理性的なシステムⅡにより構成される。システムⅠとシステムⅡは相補的な関係にあり、システムⅡはシステムⅠを監視して必要に応じて補正する役割をもつが、リスクマネジメントにおける意思決定ではしばしばその補正機能が働かない場面がある。すなわち、認知バイアスに起因するリスクマネジメントの困難性に対処するには、システムⅠとシステムⅡをいかに効果的に連携させるかが重要となる。

帰納的 TA の分析結果（3.4 節）を用いたハイブリッドアプローチにより上記仮説の

妥当性を検証したところ、リスクマネジメントの"難しさ"に関するコード全 41 件のうち、31 件（75.6%）は取引コストまたは認知バイアスの影響を受けていることがわかった。一方、残りの 10 件は、主に、リスクマネジメントの組織ルーティン化、または、未知のリスクへの対応のいずれかに関連していた。前者は主に柔軟な意思決定を必要とする部分との切り分けの難しさであり、上記の本質的な課題と密接に関連しているが、後者は別の切り口も含めた多面的なアプローチが必要である。

第5章 機械参加型リスクマネジメントの提案と評価

5.1. はじめに

本章では、リスクマネジメントの本質的な課題への対応策として、機械参加型（machine-in-the-loop）リスクマネジメントを提案する。5.2 節では、人間と機械学習の協調による機械参加型の意思決定プロセスを説明する。5.3 節では、機械参加型リスクマネジメントの具体的なイメージを提示する。5.4 節では、機械参加型リスクマネジメントの継続的進化について論じる。5.5 節では、演繹的テーマティック・アナリシス法（TA: Thematic Analysis）によるインタビュー・データの分析を実施し、提案アプローチの妥当性を評価する。

5.2. 機械参加型（machine-in-the-loop）意思決定プロセス

本節では、人間と機械学習の協調によってシステムⅠとシステムⅡを効果的に連携した意思決定の枠組みを検討する。表 5.1 に示すように、人間と機械学習は相補的な関係にある。すなわち、人間は、機械学習の予測結果を用いて人間がもつ先入観やバイアス（偏り）を排除し、合理的な意思決定につなげることができる。一方、機械学習は、人間からの気づきやフィードバック（現実との矛盾点や経験・感覚との違いなど）を得ることにより、データに不足している背景知識などを補完し、新たな変化や想定外の事象への対応力を高めることができる。

表 5.1：人間と機械学習の相補関係

	人間	機械学習
強み	暗黙知により、現実とモデルとのギャップの気づきが得られる。	モデルに基づく客観的な判断・意思決定が可能。
弱み	先入観やバイアス（偏り）の影響を受けやすい。	過去データからの帰納的推論であり、想定外の変化に弱い。

人間の意思決定プロセスを一般化すると、(1)環境からの入力情報に基づく現状把握や気づきに始まり、(2)仮説を立てて可能な行動を計画し、(3)複数の選択肢を評価して、(4)意思決定して行動し、(5)環境からのフィードバック結果を確認して(必要に応じて)知識を更新する、という一連の流れになる。人間の日常的な思考を支配するシステムⅠは、上記手順を踏みつつもさまざまなショートカットを用意して意思決定を短絡化しようとする。例えば、特定の状況と結びついた単純化した行動ルールの設定、個人の嗜好に基づく主観的な選択肢の評価、などが挙げられる。これらは、意思決定プロセスを大幅に効率化する一方、思考の偏りの温床にもなるため、客観的かつ論理的なシステムⅡを意図的に起動して、定常的にシステムⅠをモニタリングする仕掛けが必要となる。本博士論文においては、ここに機械学習モデルの活用を試みる。

一般に、人間が機械学習モデルの予測・推定結果を活用するには、以下の 3 つのパターンが考えられる。

A) 気づきの支援
   - 予測・推定結果を人間の気づきに活用

B) 人間が解釈して利用
   - 予測・推定結果を人間が解釈して意思決定

C) 機械的に利用
   - 予測・推定結果から自動的に意思決定

このうち、C)は機械が主役であり人間が介在する余地はないが、A)、B)についてはあくまで人間が主役となる。そこで、人間の意思決定プロセスを中心に置き、それと機械学習モデルが相互作用する“機械参加型(machine-in-the-loop)意思決定プロセス”が考えられる(図 5.1)。図の左側は人間の意思決定プロセスを表し、図の右側は機械学習モデルの予測･推定とモデル更新のサイクルである。実線の矢印は制御フロー、点線の矢印はデータフローを表す。機械学習から人間への矢印(データフロー)は、新たな情報の提示による気づきの支援(A)や予測･推定結果に基づく判断材料の提供(B)を示している。特に、後者(B)においては、予測・推定結果を人間が解釈して意思決定に反映させる必要があり、機械学習モデルの解釈可能性がきわめて重要とな

る。一方、人間から機械学習への矢印（データフロー）は、意思決定・行動の結果や予測・推定と現実とのギャップを機械学習モデルにフィードバックすることによるモデル更新を示し、機械学習モデルの継続的な予測精度向上に貢献する。

このような人間と機械学習の協調の枠組みは、システムⅠとシステムⅡを効果的な連携を促進する。すなわち、人間の意思決定プロセスはデフォルトではシステムⅠが起動しているが、機械学習の介入によりシステムⅡが喚起され、人間と機械学習が協調した合理的な意思決定が促される。その結果、人間の意思決定における認知バイアスの影響の低減が期待できる。Bengio（2019）は、「人工知能（AI）・機械学習は、現状、システムⅠの役割しか果たせておらず、今後、システムⅡの能力を獲得していく必要がある」と述べている。しかし、これはあくまで、C）“機械が主役の世界”での話であり、A)、B）“人間が主役の世界”においては、必ずしも、機械がシステムⅡの能力も獲得せずとも、人間の中にあるシステムⅡを喚起するという役割を果たすことは十分に可能であるというのが、本博士論文の主張である。

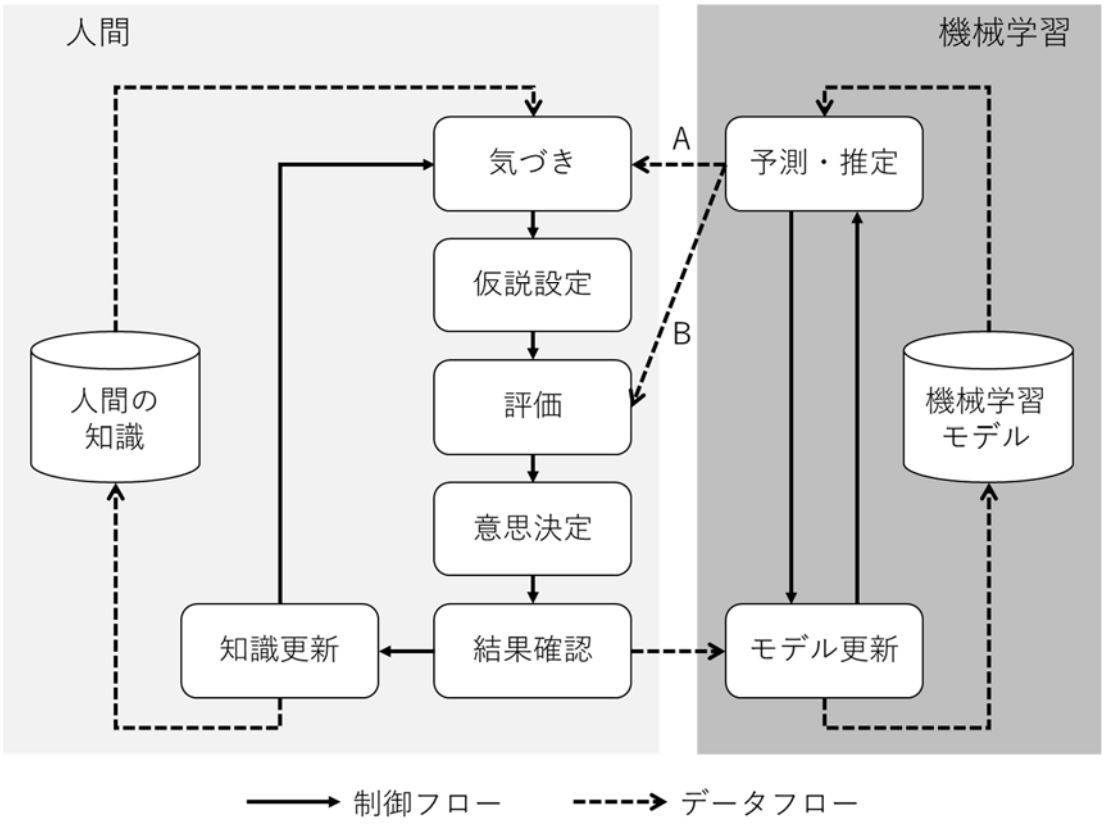


図 5.1：機械参加型（machine-in-the-loop）意思決定プロセス

機械参加型意思決定プロセスは、限定合理的な利害関係者間の合意形成にも役立つと考えられる。すなわち、同一の機械学習モデルを利用する machine-in-the-loop の環境の中に複数の限定合理的な人間が存在する場合、最初は、各人間の知識および機械学習モデルにはさまざまなバイアス（偏り）や不整合が存在する可能性がある。しかしながら、機械学習モデルに対する行動結果のフィードバックやドメイン知識の教示、および機械学習モデルからの継続的なバイアス補正を通じて、徐々に machine-in-the-loop の環境の中に共通的な知識が蓄積・共有されていくことが期待される。これは、限定合理的な利害関係者間の合意形成に役立つと同時に、組織内の有識者がもつ知識・ノウハウの蓄積や継承にもつながり、若手・中堅実務者の人材育成という観点からも有効である。

5.3. 機械参加型リスクマネジメントの提案

製品開発組織におけるリスクマネジメントは、一連の意思決定プロセスとみなすことができる。すなわち、“リスクの特定”では「プロジェクトにどのようなリスクがあるか？」、“リスク分析”では「リスクの発生確率や影響度はどの程度か？」、“対応計画策定”では「リスクに対して、いつ、どのような対策をとるか？」、“リスクの監視”では「プロジェクトのリスクはどのように変化しているか？」、“振り返り”では「将来のプロジェクトに向けて、何を残すべきか？」について、それぞれ意思決定を行う。これらの意思決定のステップに対して機械学習モデルの構築・活用のステップを同期させて、現実（リアル）の世界と仮想（データ）の世界を対応付けることにより、リスクマネジメントにおける machine-in-the-loop の枠組みを実現することができる（図 5.2）。本博士論文においては、これを“機械参加型（machine-in-the-loop）リスクマネジメント”と定義する。

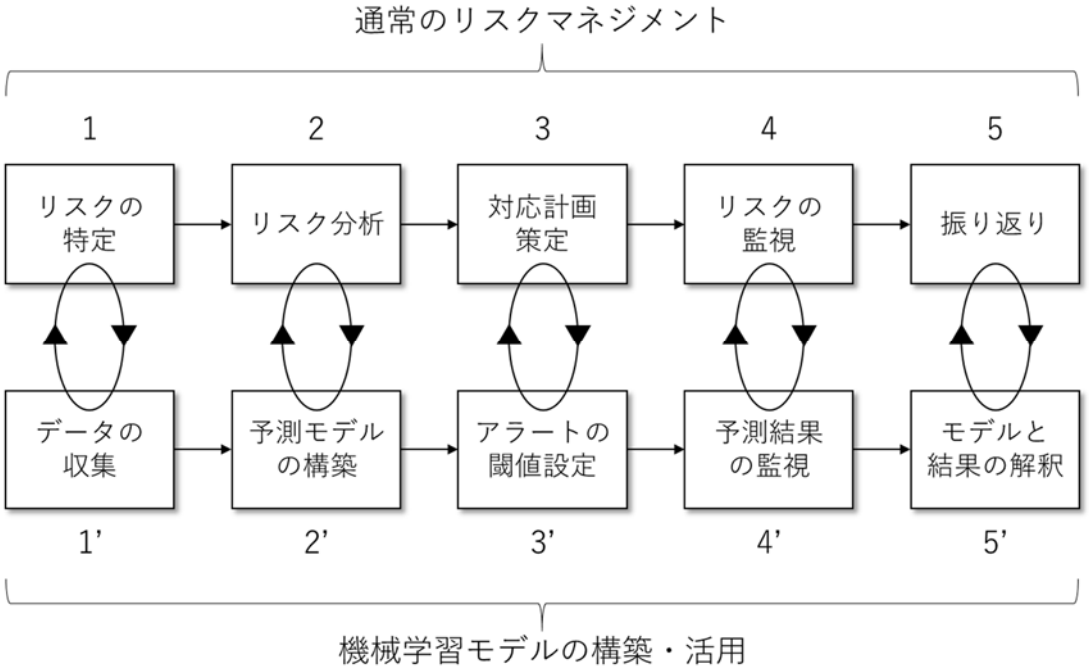


図 5.2：機械参加型（machine-in-the-loop）リスクマネジメント

以下、図 5.2 の各ステップについて具体的な例題を用いて説明する。

まず、1. “リスクの特定”に対応して、機械学習側ではモデル構築のための 1’.“データ収集”を行う。その際、実世界でのモデルの解釈を容易にするために、リスク要因と測定データをひも付けて、実際のプロジェクトの状況とモデルの振り舞いの乖離をなるべく減らすことが望ましい。表 5.2 は、例題のプロジェクトデータ[8]に含まれる変数を示している。X1 から X23 は説明変数であり、Y は目的変数である。本例題のデータは全部で 200 件あり、そのうちの 150 件を学習データ、残りの 50 件をテストデータとして使用する。一般に、プロジェクト・リスクマネジメントで使用されるデータはビッグデータにはほど遠く、それほど数が多くない場合があり、また一部のデータには欠損値が多く含まれるなど、必ずしもデータ品質がよいとは言えない場合がある。機械参加型リスクマネジメントでは、こうした特性に適した機械学習モデルを選択する必要がある。

[8] 実データを加工して生成した架空のデータであり、特定の組織には依存していない.

表 5.2：例題のプロジェクトデータに含まれる変数

ID	変数名	説明	分類	工程
X1	product_type	製品分類 = {AAA, BBB, CCC, DDD, EEE}	離散	要求仕様化 (RD)
X2	dev_type	開発分類 = {X, Y, Z}	離散	要求仕様化 (RD)
X3	rank	開発ランク = {A, B}	離散	要求仕様化 (RD)
X4	RD_effort	要求仕様化工数 [人H]	連続	要求仕様化 (RD)
X5	RD_review_time	要求仕様化レビュー時間 [H]	連続	要求仕様化 (RD)
X6	RD_defects	要求仕様化欠陥数 [件]	連続	要求仕様化 (RD)
X7	BD_effort	基本設計工数 [人H]	連続	基本設計 (BD)
X8	BD_review_time	基本設計レビュー時間 [H]	連続	基本設計 (BD)
X9	BD_defects	基本設計欠陥数 [件]	連続	基本設計 (BD)
X10	BD_delay	基本設計遅延日数 [日]	連続	基本設計 (BD)
X11	DD_effort	詳細設計工数 [人H]	連続	詳細設計 (DD)
X12	DD_review_time	詳細設計レビュー時間 [H]	連続	詳細設計 (DD)
X13	DD_defects	詳細設計欠陥数 [件]	連続	詳細設計 (DD)
X14	DD_delay	詳細設計遅延日数 [日]	連続	詳細設計 (DD)
X15	SLOC	ソースコードの総ステップ数 [LOC]	連続	実装 (CD)
X16	reuse_rate	ソースコードの再利用率 [%]	連続	実装 (CD)
X17	UT_effort	単体テスト工数 [人H]	連続	実装 (CD)
X18	UT_defects	単体テスト欠陥数 [件]	連続	実装 (CD)
X19	IT_effort	結合テスト工数 [人H]	連続	テスト (Test)
X20	IT_defects	結合テスト欠陥数 [件]	連続	テスト (Test)
X21	ST_effort	総合テスト工数 [人H]	連続	テスト (Test)
X22	ST_defects	総合テスト欠陥数 [件]	連続	テスト (Test)
X23	dev_delay	開発遅延日数 [日]	連続	テスト (Test)
Y	PJ_outcome	プロジェクト成功(S) / 失敗(F)	離散	PJ終了時点 (End)

続いて、2. “リスク分析”に対応して、機械学習側では 2’. “予測モデルの構築”を行う。ここでは、機械学習モデルとしてナイーブベイズ分類器（Domingos and Pazzani 1997; Lewis 1998）を採用する。ナイーブベイズは、ベイズの定理に基づいた比較的単純な機械学習モデルであり、計算効率がよくノイズに対して頑健という特徴がある。変数間に条件付き独立性の仮定を置くことで確率計算を単純化し、新たな情報追加による事前確率の逐次更新アルゴリズムを実現している。このような確率更新のメカニズムは“ベイズ更新”と呼ばれ、リスクマネジメントにおける人間の思考プロセスに比較的近いと考えられている。データに含まれる欠損値も、ベイズ更新において新たな情報追加がなかったとみなすことで矛盾なく処理できる。

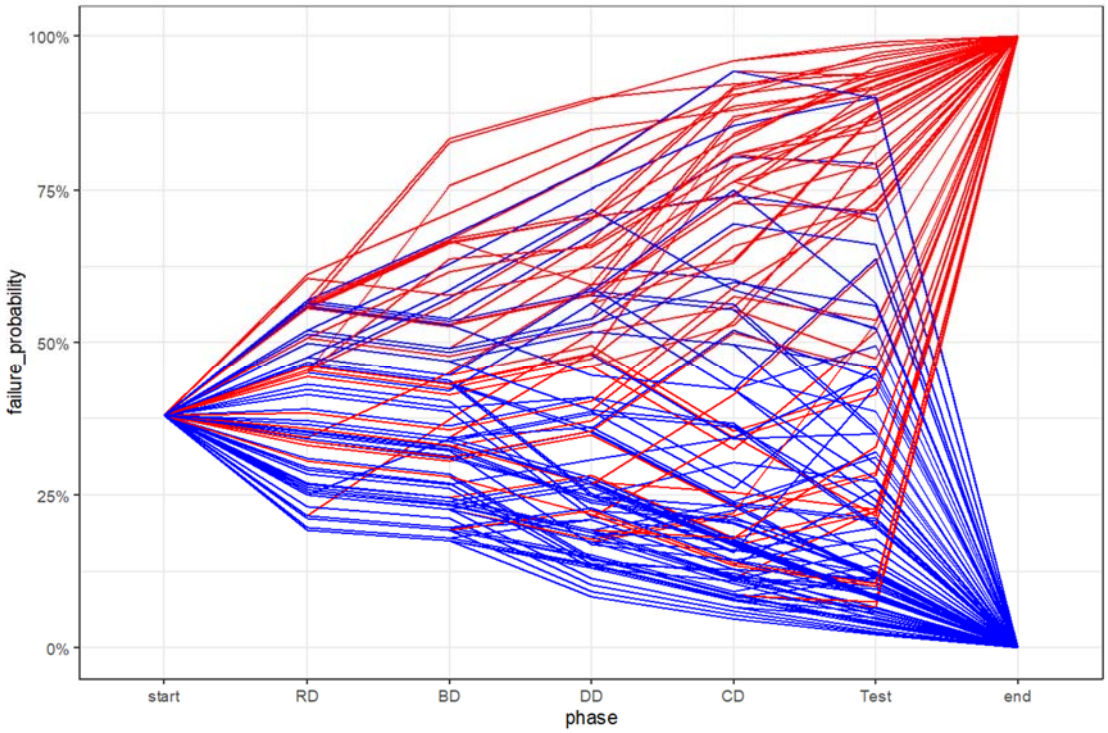


図 5.3：ナイーブベイズ予測モデルの例

表 5.2 の学習データ 150 件に対して、工程別リスク推移を示すナイーブベイズ予測モデルを構築すると図 5.3 のようになる。ここで、X 軸はプロジェクトの工程を表しており、左から右へいくほど後の工程になる。start はプロジェクトの開始時点、end は終了時点である。Y 軸はプロジェクトの失敗確率を表しており、各折れ線は個々のプロジェクトに対応している。赤い折れ線は失敗したプロジェクトの失敗確率の推移を表し、end において失敗確率が 100%となる。一方、青い折れ線は成功したプロジェクトの失敗確率の推移を表し、end において失敗確率が 0%となる。図 5.3 から、個々のプロジェクトの失敗確率が逐次更新されていく様子がわかる。

ナイーブベイズ予測モデルは、後工程になるほど新たな情報が追加されるため、一般に、予測精度の段階的向上が期待される。図 5.4 は、要求仕様化（RD）、基本設計（BD）、詳細設計（DD）、実装（CD）、テスト（Test）における ROC 曲線を表している。ここで、ROC（Receiver Operating Characteristic）曲線とは、閾値をさまざまな値に変化させたときの偽陽性率（X 軸）と真陽性率（Y 軸）のプロットであり、その下側の面積は AUC（area under the ROC curve; Fawcett 2006）と呼ばれる。AUC は 0 から 1 までの範囲の値をとり、値が大きいほど予測精度が高いことを意味する。RD、BD、

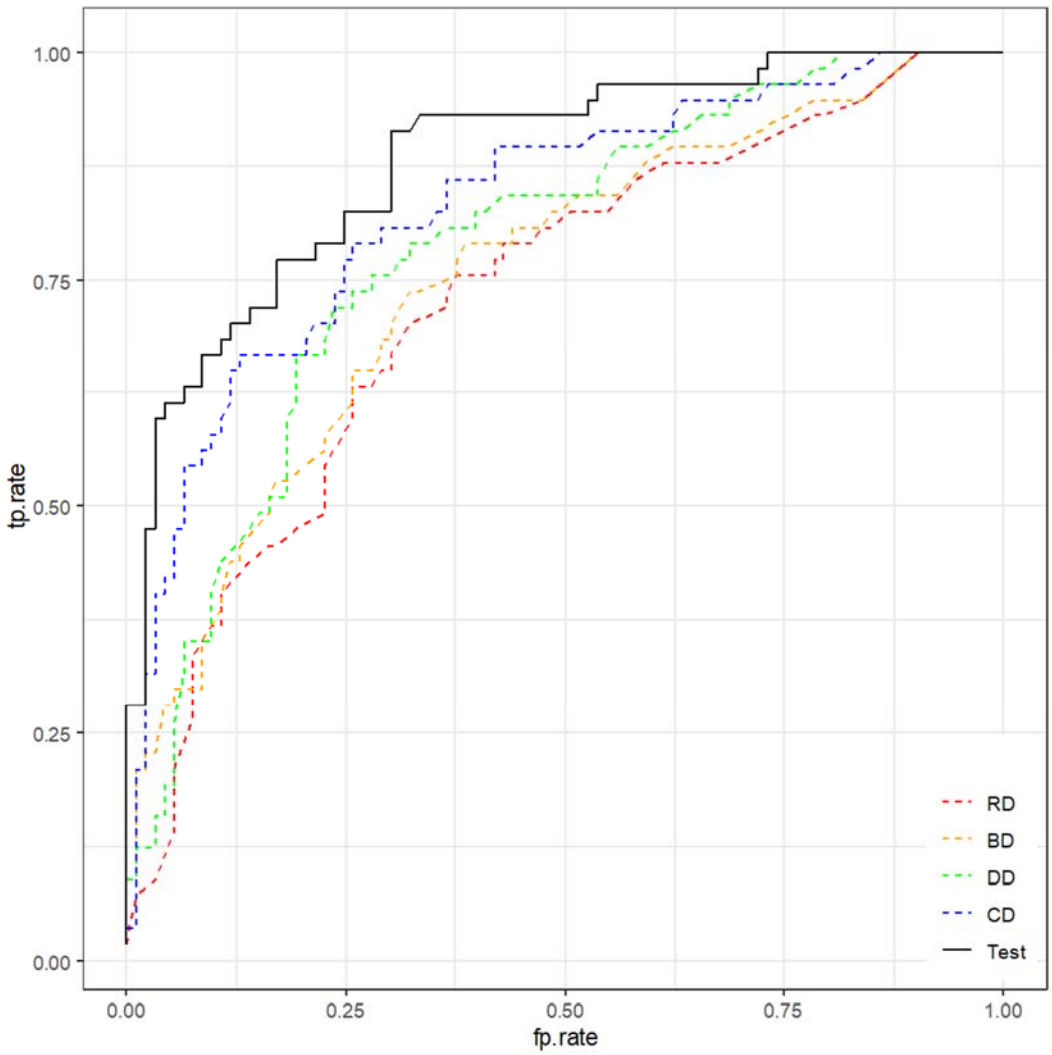


図 5.4：ナイーブベイズ予測モデルの工程別 ROC 曲線

DD、CD、Test における AUC は、それぞれ、0.729、0.754、0.784、0.831、0.883 となり、後工程になるにしたがって、徐々に精度[9]が向上していくことがわかる。

3. “対応計画策定”において、機械学習側では 3’. “アラートの閾値設定”を行う。すなわち、学習データに対する予測モデルの当てはめ結果などを参考として、予測結果がどのようになったらアラートを上げるのかをあらかじめ設定しておく。原則として、アラートを発するセンサーの感度を高くすれば、失敗の見逃しは減るが誤検知率が高くなるため、もし、高コストのリスク対応策を実施する場合には費用対効果の面で考

[9] 学習データへの当てはめ精度.

慮が必要となる。逆に、ノイズを拾わないようにセンサーの感度を低くしてアラートの精度を高めれば、誤検知率は低くなるが失敗の見逃しは多くなる。どのような閾値でどのような対応をとるべきかは、リスクマネジメントの戦略やリスク対応計画のオプション（選択肢）にも関連してくる。

図 5.3 の例において、予測での「失敗確率 50%以上」、および、「失敗確率 70%以上」を閾値の候補とする。表 5.3 は、学習データへの予測モデルの当てはめ結果における各閾値での失敗プロジェクト比率（実績）の工程別推移を表している。「失敗確率 50%以上」はいわゆる高感度のセンサーに相当し、RD、BD、DD などの上流工程の段階から多くのアラートが上がっている。元々の（何も情報がないときの）失敗プロジェクト比率が 38%であることから、「失敗確率 50%以上」のアラートによりプロジェクト失敗のリスクが大幅に高まっていると判断できる。例えば、要監視対象として状況確認の頻度を増やすなどの対応が考えられる。「失敗確率 70%以上」はいわゆる高精度のアラートに相当し、感度は下がるが検知したプロジェクトの失敗の確度は一層高くなる。コストはかかるが高い効果が見込める施策、例えば、プロジェクトマネジメントオフィス（PMO; Block and Frame 1998=2002）からのサポートメンバーの投入なども選択肢に入ってくる。

こうした失敗確率の閾値に基づくアラートの他に、失敗確率の変動が大きいプロジェクトのリストアップや、失敗確率トップ X などのランキングの監視も有効である。表 5.4 は、Test フェーズにおいて失敗確率がトップ 10 だったプロジェクトの工程別ランキング推移を示している。

表 5.3：失敗プロジェクト比率（実績）の工程別推移

		RD	BD	DD	CD	Test
失敗確率50%以上	該当するPJ数	33	33	47	50	49
	失敗PJ数	23	23	30	38	39
	失敗PJの比率 [%]	69.7%	69.7%	63.8%	76.0%	79.6%
失敗確率70%以上	該当するPJ数	0	5	14	32	38
	失敗PJ数	0	5	10	27	34
	失敗PJの比率 [%]	0.0%	100.0%	71.4%	84.4%	89.5%

表 5.4：失敗確率トップ 10 プロジェクトのランキング推移

PJ名	RD	BD	DD	CD	Test	最終結果
PJ073	23	2	2	1	1	失敗
PJ121	23	2	2	1	2	失敗
PJ004	23	33	17	9	3	失敗
PJ058	9	7	11	6	4	失敗
PJ130	50	22	15	7	5	失敗
PJ032	27	34	20	17	6	失敗
PJ050	6	1	1	5	7	失敗
PJ042	14	26	30	13	8	失敗
PJ128	1	5	5	8	9	失敗
PJ023	9	7	7	4	10	失敗

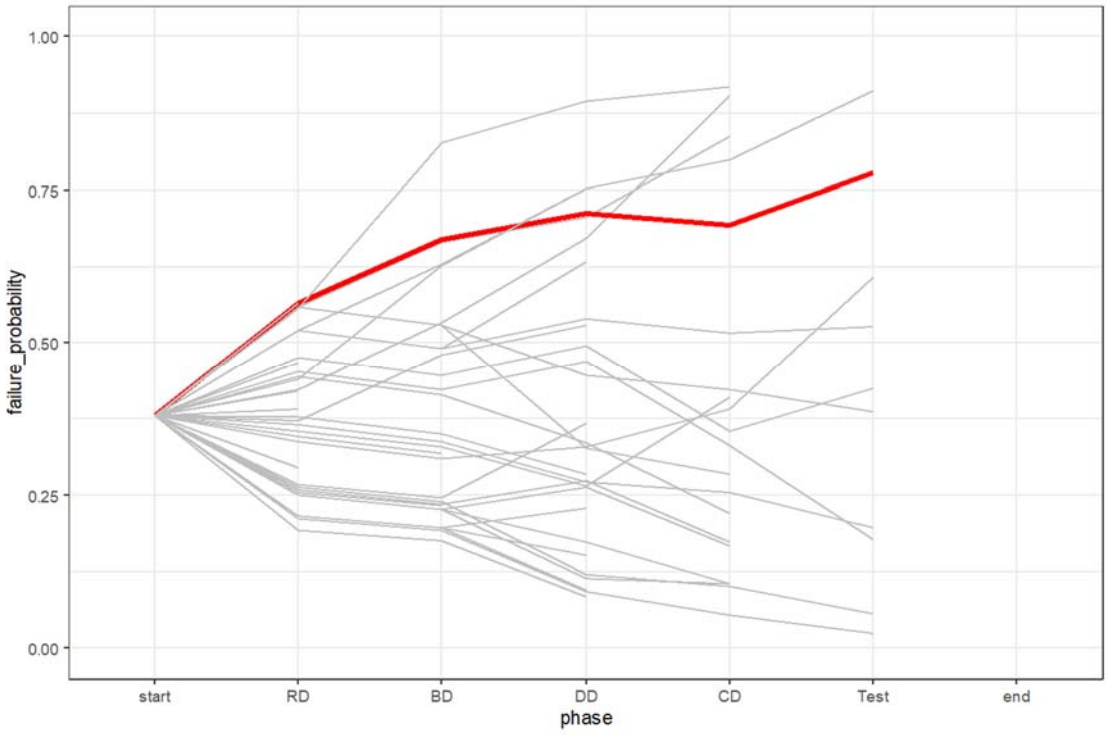


図 5.5：進行中プロジェクトの工程別リスク推移（途中経過）

4. “リスクの監視”に対応して、機械学習側ではテストデータの 4’. “予測結果の監視”を行う。図 5.5 は、150 件の完了プロジェクトで学習したナイーブベイズ予測モデルを、残り 50 件のプロジェクト（テストデータ）に対して適用した結果である。これらは進行中のプロジェクトであり、各プロジェクトの最新工程までの失敗確率が示されている。赤色の折れ線を、現在注目しているプロジェクトと仮定すると、このプロジェクトは RD フェーズから失敗確率が 50%を超えており、その後も徐々に失敗確率が高まって最終的には 70%を上回るなど、相対的にかなりリスキーなプロジェクトと判断できる。継続的な監視を行うと共に、プロジェクトの実態をメンバーからヒアリングした上で、より積極的なリスク対応策の実施が必要となる可能性がある。

最後の 5. “振り返り”において、機械学習側では 5’. “モデルと結果の解釈”や考察を行う。図 5.6 は、図 5.5 の予測の最終結果である。赤い折れ線は失敗したプロジェクトの失敗確率の推移を表し、青い折れ線は成功したプロジェクトの失敗確率の推移を表す。Test 工程で失敗確率が 50%を超えたプロジェクトを“失敗”とみなしたとき、予測が当たったプロジェクトを実線、外れたプロジェクトを点線で表している。予測結果のサマリーは、表 5.5 の混同行列（confusion matrix）で表される。正解率（accuracy）、再現率（recall）、適合率（precision）、F 値（F-measure）は、それぞれ、0.80、0.67、0.57、0.62 となる。ここで、“失敗”と予測したが実際には“成功”だったケース（図 5.6 の青色の点線）は、第 1 種の過誤、または、偽陽性（FP: false positive）と呼ばれ、主に再現率に影響する。一方、“成功”と予測したが実際には“失敗”だったケース（図 5.6 の赤色の点線）は、第 2 種の過誤、または、偽陰性（FN: false negative）と呼ばれ、主に適合率に影響する。リスク対応策の費用対効果を改善するには再現率を上昇させる必要があり、失敗の見逃しが問題になる場合には適合率を上昇させる必要がある。一般に、再現率と適合率はトレードオフの関係にあり、リスクマネジメントの戦略オプションを考慮しつつ、全体最適を図っていく必要がある。

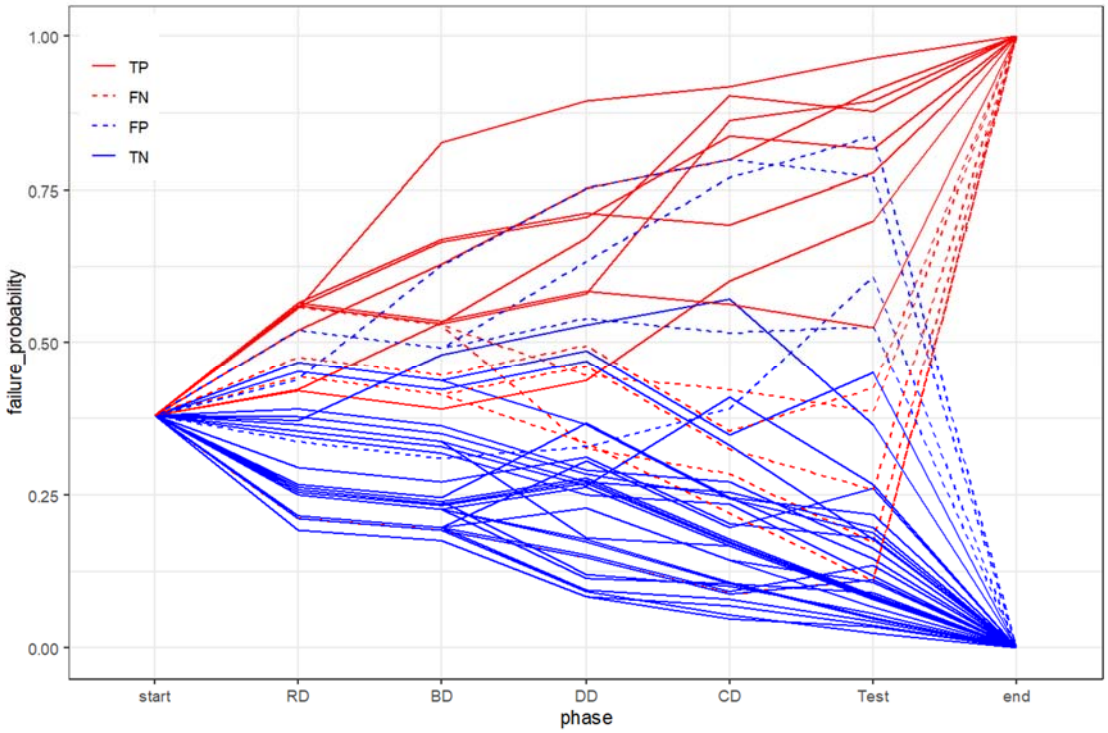


図 5.6：進行中プロジェクトの工程別リスク推移（最終結果）

表 5.5：混同行列による予測結果のサマリー

		予測	
		失敗	成功
実績	失敗	8	6
	成功	4	32

正解率	= (8+32)/(8+4+6+32)	= 0.80
適合率	= 8/(8+4)	= 0.67
再現率	= 8/(8+6)	= 0.57
F値	= 2*0.67*0.57/(0.67+0.57)	= 0.62

予測が外れたケースに対する分析は、予測モデル改善の重要な機会となる。その際、ナイーブベイズは、比較的単純な構造をもつホワイトボックスの機械学習モデルであるため、モデルの中間・最終結果を用いてさまざまな分析が可能である。図 5.7 は、BIC（Bayesian information criteria; Schwarz 1978）から算出した説明変数の重要度（Mori and Uchihira 2019）であり、縦軸の数値が大きいほど失敗への影響が大きい。グラフから、後工程になるほど失敗への影響が大きい変数が増えていることがわかる。また、product_type、dev_type、rank などの属性情報、RD、BD、UT、IT、ST における各工数、BD、DD における各工程遅延は、重要度が 0 であり予測精度に寄与していないことも読み取れる。こうした考察は将来のデータ収集や予測モデルの改善に有効である。

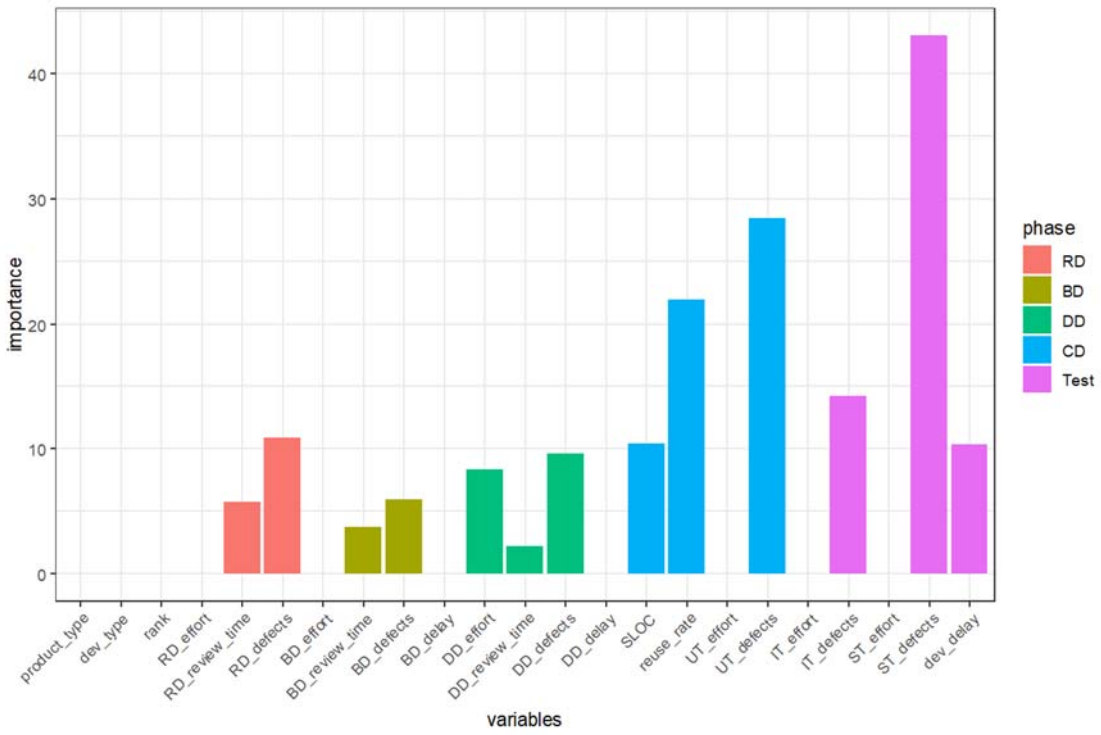


図 5.7：ナイーブベイズ予測モデルにおける説明変数の重要度

また、ナイーブベイズでは、モデルの中間生成物を抽出して加工することで、変数や事例の類似度を分析することもできる。図 5.8 は、予測結果におけるプロジェクト間の類似度を算出してクラスター分析を行ったものである。縦軸は、プロジェクト間の“距離”[10]を表し、距離が近いほど予測結果が似ていることを意味する。赤色のボックスは、予測結果が似ているにもかかわらず実際の結果が異なっていたプロジェクト群を示している。例えば、これらにおいて実際の結果の違いがどこから生じたのかなどを調べることにより、プロジェクト失敗の本質的なメカニズムの理解や新たなデータ収集のヒントなどが得られる可能性がある。

[10] WoE（weights of evidence）ベクトルのコサイン類似度.

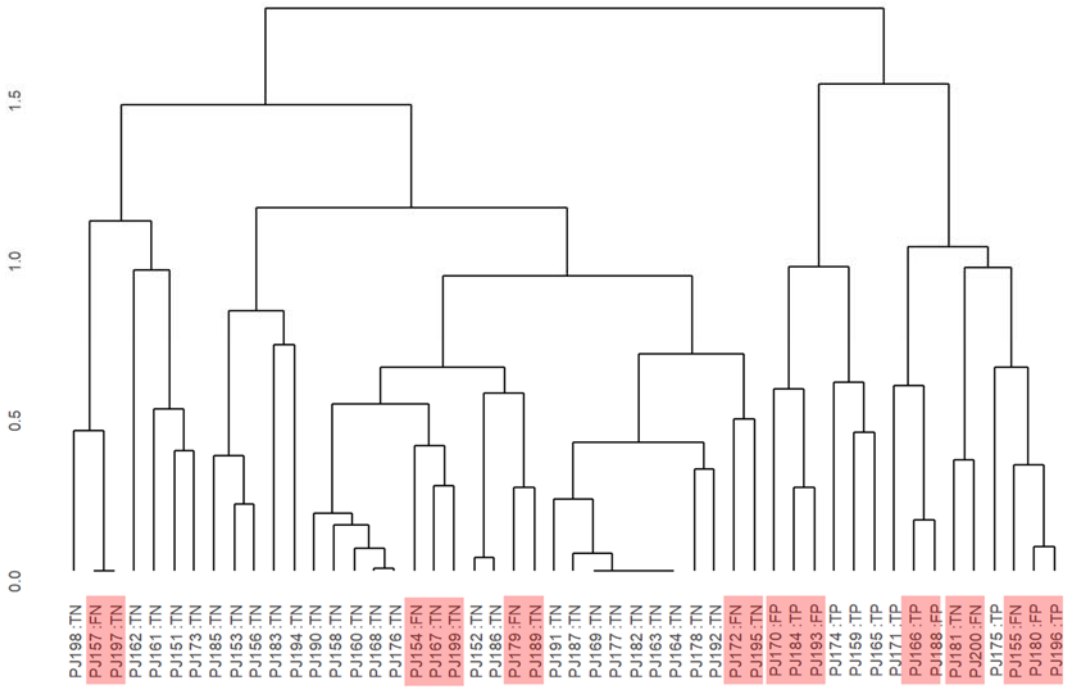


図 5.8：プロジェクト間の類似度のクラスター分析

このように、機械参加型リスクマネジメントでは、機械学習モデルがリスクマネジメントにおける人間の合理的な意思決定を支援すると同時に、機械学習モデルも人間の知識を積極的に取り入れることによって継続的に進化していくことが可能となる。図 5.2 の上下のステップをつなぐリンクは、さまざまな場面におけるプロジェクト実務者とデータ分析者の間の相互連携の重要性を示している。

5.4. 機械参加型リスクマネジメントの継続的改善

機械参加型リスクマネジメントは、リスクマネジメントを行うプロジェクト実務者（以下、プロジェクト側）と機械学習を活用するデータ分析者（以下、機械学習側）の双方がそれぞれ不完全な知識に基づいて行動していることを前提としている。すなわち、機械参加型リスクマネジメントの実践においては、プロジェクト側のリスク知識の更新と機械学習側のモデル化知識の更新を並行かつ継続的に実施することが重要である。これらの更新には、組織的知識創造プロセス （SECI モデル; Nonaka and Takeuchi 1995=1996）が適用可能となる。ただし、リスクという不確実性を伴う事象や

状態の知識を取り扱うには、プロジェクト側と機械学習側がそれぞれ独立に知識更新を行うだけでは十分ではない。プロジェクト側では、リスクに関する知識の取り扱いにおいて確率や統計による定量的な評価が不可欠だが、残念ながら人間の直観は確率や統計の処理をあまり得意としていない。もし、リスクに関する知識の変換が主に人間の直感によってなされ、それが客観的な視点から十分にチェックされない場合、認知バイアスの影響で知識の増幅が妨げられ、表面的な知識を単に共有するだけの活動へとスパイラルダウンする恐れがある。また、機械学習側においても、データが表しているのは現実のプロジェクトのごく一部であり、モデルはその範囲内での内挿に過ぎない。プロジェクト側から現場の知識を積極的に取り込まなければ、そもそも実用的なモデルにたどり着けない可能性がある。すなわち、プロジェクト側のリスク知識更新と機械学習側のモデル化知識更新は、相互に密接に連携しながらスパイラルアップしていくことが期待される。

図 5.9 は、機械参加型リスクマネジメントにおけるプロジェクト側と機械学習側の知識更新のイメージを示している。プロジェクト側のリスク知識更新のサイクルと機械学習側のモデル化知識更新のサイクルが相互に連携しながら、それぞれ SECI モデルが回るという 2 サイクル構造となる。一方のサイクルが他方のサイクルの駆動力となり、互いの知識を補完しながらスパイラルアップしていく。以下、SECI モデルの各知識変換モードで想定されるプロジェクト側と機械学習側の相互学習の例を示す。

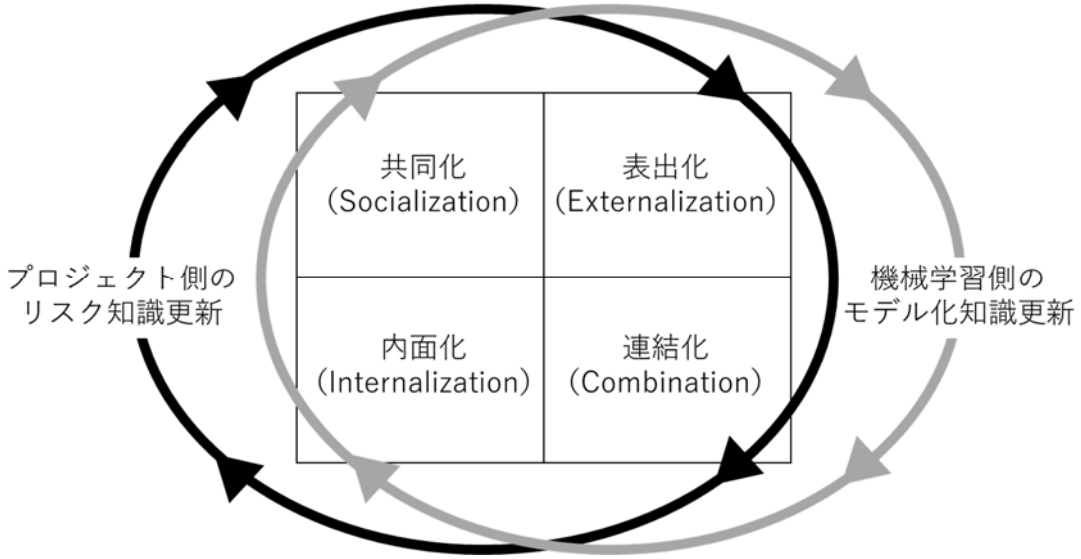


図 5.9：プロジェクト側と機械学習側の知識更新の 2 サイクル構造

共同化（Socialization）：暗黙知から暗黙知へ

プロジェクト側は、プロジェクト活動における成功や失敗などのさまざまな経験をメンバー間で共有する際、機械学習モデルの予測結果との一致点や相違点に着目することにより、外部的かつ客観的な議論の軸を得ることができる。機械学習側は、プロジェクト活動のさまざまな側面についての生データを収集・蓄積する際、プロジェクト側の日常的な行動を観察することにより、新たなデータ収集のヒントを得ることができる。ここで、生データには、情報システムなどに記録されたデータ、プロジェクト活動をモニタリングして得られたデータ、各種成果物から抽出・変換されたデータなどが含まれる。

表出化（Externalization）：暗黙知から形式知へ

プロジェクト側は、各個人の経験の中に埋もれているリスクに関する知識を抽出して形式知化する際、機械学習モデルの構築・活用への参画を通じて、プロジェクトの特性などに関する思い込みやバイアス（偏り）を減らすことができる。これは一種のリフレーミング、すなわち、「ある枠組み（フレーム）で見ている事象に対して、その枠組みをはずして違う枠組みで見ること」の効果である。ここで、リフレーミング（Bandler and Grinder 1981=1988）とは、主に神経言語プログラミング（NLP: neuro-linguistic programming）の分野で用いられる用語であり、人間のもっている心理的枠組み(フレーム)の変換によって人間の反応や行動の変革を引き起こすことを意味する。また、機械学習側は、予測モデルのプロトタイプ構築において、収集・蓄積した生データを加工して説明変数や目的変数の候補を選択する際、プロジェクト側のもつ現場の実践的な知識を活用することにより変数の探索を効率的に行うことができる。

連結化（Combination）：形式知から形式知へ

プロジェクト側は、例えば、失敗事例データベースの構築など、形式知化されたリスクに関する個々の知識を組み合わせて全体として矛盾がないように体系化する際、機械学習モデルの予測結果との突き合わせにより知識体系の矛盾点を見つけたり、リフレーミングによる新しい知識の獲得などが期待できる。また、機械学習側は、構築した予測モデルをさまざまな角度から分析し、他の概念と結合して継続的にブラッシュアップしていく際、プロジェクト側から単なる性能面だけでない実用的な視点での

フィードバックを得ることができる。

内面化（Internalization）：形式知から暗黙知へ

プロジェクト側は、体系化された知識をプロジェクト活動の現場へと展開する際、機械学習モデルの構築・活用を通じて得られた客観的な裏付けに基づき、自身の知識・行動への確信や他者への説得性を高めることができる。また、機械学習側は、予測モデルを解釈・理解してモデル化知識として内面化する際、プロジェクト側から表面的な分析だけでは見えないデータ間の依存関係や順序関係などの背景知識を得ることができる。こうした解釈・理解は次の反復でのデータ収集に活かすことができる。

このようにプロジェクト側のリスク知識更新のサイクルと機械学習側のモデル化知識更新のサイクルが相互に連携しながら反復して回ることで、機械参加型リスクマネジメントが継続的に進化していく。

機械参加型リスクマネジメントの適用効果として、以下の仮説が立てられる。

従来は、リスクマネジメントの本質的な課題である不確実な状況下でのトレードオフを伴う意思決定問題に加えて、取引コストや認知バイアスの影響が作用することにより合理的な判断からの偏りが増幅し、リスクマネジメントの形骸化などのさまざまな実践上の困難性につながっていたと考えられる。これに対して、機械参加型リスクマネジメントの適用により、（1）不確実さの低減（過去データに基づく予測）による意思決定の合理化、（2）利害関係者間の合意形成による取引コストの低減、（3）システムⅠとシステムⅡの連携による認知バイアスの影響の緩和、が期待できる（図5.10）。すなわち、図 4.1 の意思決定モデルにおいて、新たな取引コストを T'、損失にかかる係数を k' とすると、リスク R への対応策を実行すべき条件は $P_2L_2 - P_1L_1 > k'(C + T')$ となる。ただし、 $T > T' > 0,\ k > k' > 1$ である。このように、提案アプローチはリスクマネジメントの実践上の困難性への有効な対応策となることが期待できる。

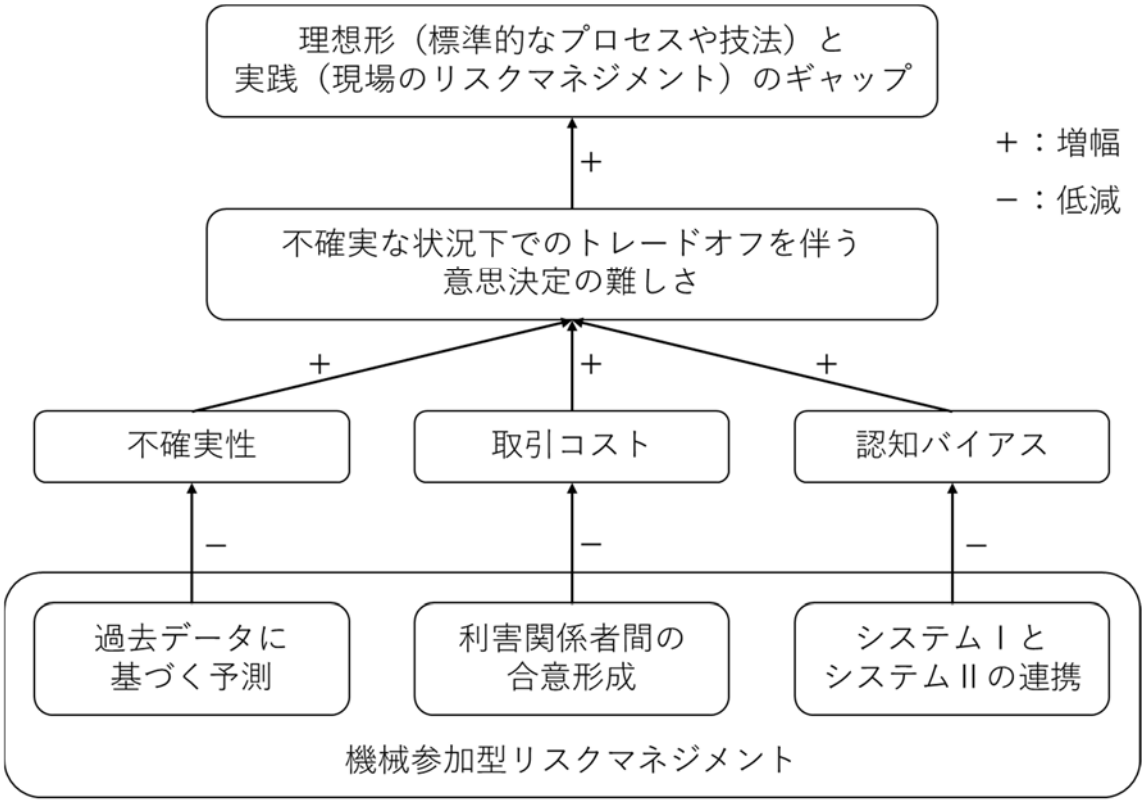


図 5.10：機械参加型リスクマネジメントの適用効果の仮説

5.5. 演繹的 TA による提案アプローチ適用部門の評価

機械参加型リスクマネジメントを社会インフラ系の実製品開発部門に適用した。適用期間は 2017 年 4 月から現在（2020 年 3 月）であり、現在もまだ、適用が続いている。提案アプローチの有効性を確認するために、適用部門の実務者 2 名へのインタビュー調査を実施した。表 5.6 にインタビュー対象者の一覧を示す。各インタビュー対象者には匿名性の確保と特定の組織や製品に関わる情報については非公開とすることを条件に了承を得た。

表 5.6：インタビュー（第 2 回）の対象者

対象者	実務年数	事業領域	インタビュー日時	場所
H	20年以上	社会インフラ	2020/02/27（木）11:00-11:30	X 事業所
I	20年以上	社会インフラ	2020/02/27（木）14:00-14:30	X 事業所

インタビューは、あらかじめ用意したスクリプトに沿って開始し、適宜、気になる点について自由に意見を述べてもらうという“半構造化インタビュー”の形式を採用した。各インタビュー対象者には、インタビューの冒頭において、インタビューの目的、研究の方法、インタビュー内容の公開方法と注意点、適用効果について率直な意見を聞きたい旨、などを伝えた。

インタビュー・スクリプトの質問項目を表 5.7 に示す。ここで、before は機械参加型リスクマネジメントを適用する前の状況、after は現状、および、想定される近い将来の姿[11]を指している。質問項目はインタビュー対象者に対して事前にメールで送付した。インタビュー時には、まず、Q1 から Q3 に対して Yes/No で回答してもらい、続けて「なぜ、そのように思うか？」と尋ねることによって、回答の背景にある思いや考えを聞き出すようにした。その上で、最後に一般的な質問を投げかけた。

表 5.7：インタビュー（第 2 回）の質問項目

Q1	before/afterで、プロジェクト失敗の要因がより明確になったか？
Q2	before/afterで、プロジェクトリスクについて第三者への説明が容易になったか？
Q3	before/afterで、プロジェクトリスクに関する自身の想定や思い込みが変化したか？
Q4	before/afterで、その他、何か気づいたことはあるか？

すべてのインタビューの内容はボイスレコーダーで記録し、まず最初に、一語一句の正確な文字起こしを行った。その上で、内容に直接関係ない間投詞の削除、意味や文脈を変えない範囲での文章の修正（わかりやすい文章への書き換え、足りない語句の補充、など)、特定の個人や組織や製品に関わる情報の削除といった前処理を行い、分析用データとしてまとめた。

インタビュー結果は、テーマティック・アナリシス法（TA: Thematic Analysis）の演繹的分析手法を用いて分析した。演繹的分析手法は、既存の理論や先行研究結果の仮説を用いて質的データを分析する方法である（土屋 2016)。以下に、本博士論文で実施した演繹的 TA の手順を示す。帰納的 TA では、特に前提を置かず、ボトムアップ的にコーディングを行うのに対して、演繹的 TA では、トップダウンでコードを設定して、そこに生データを当てはめるという点が異なっている。

[11] 機械参加型リスクマネジメントの適用が現在進行形であるため.

1. **インタビュー・データを切片化する。**
   切片化の粒度（コーディングユニット）は、各質問項目に対する個々の回答とした。いわゆる、構造的コーディングである。ただし、1 つの回答の中で複数の話題に触れているものは、それぞれ別々の切片に分割した。
2. **切片データをコーディングする。**
   既存の理論や仮説からあらかじめ初期コードを設定しておき、切片データに対して内容を代表していると思われるコードを当てはめる。その際、各コードには、肯定的な例と否定的な例の両方が含まれる可能性がある。
3. **必要に応じて、新たなコード（カテゴリー）を追加する。**
   一部の切片データは、初期コードに当てはまらない可能性もある。その際は、必要に応じて、新たなコードを追加したり複数のコードをまとめてカテゴリー化する。

まず、インタビュー・データの切片化（ステップ 1）を行い、24 件の切片データが得られた。続いて、各切片データのコーディング（ステップ 2、ステップ 3）を実施した。ここでは、初期コードとして、前節で示した機械参加型リスクマネジメントの適用効果より、（1）不確実さやあいまいさの低減、（2）利害関係者間の合意形成、（3）データによる認知バイアスの補正、を設定した。コーディング作業は筆者を含めて計 2 名で実施した。切片データが初期コードの内容を直接的には語っていない場合においても、語りの背景や意図を推察して可能な限りもっとも関連性の高いコードを割り当てる方針とした。コードの信頼性を高めるために、最初のコーディングの後、元の切片に戻ってコードがデータの内容を正しく表しているかを確認し、不整合があった場合には再度コーディングを繰り返すという反復的な作業を行った。表 5.8 に、最終的なコードの一覧、および、各切片データとの対応関係を示す。なお、詳細なコーディング作業の過程および結果については、付録 A2 に掲載している。

表 5.8：コードの一覧、および、各切片データとの対応関係

コード	対応する切片番号	
	肯定的 (P)	否定的 (N)
不確実さやあいまいさの低減	H1-2、H4-1、H4-7、I1-1、I1-2、I4-1	H4-2、H4-3
利害関係者間の合意形成	H2-1、H2-2、I4-2、I4-3	H4-4、H4-5、H4-6、I3-1、I3-2、I4-6
データによる認知バイアスの補正	H1-1、I2、I4-4、I4-5	H3、I4-7

演繹的 TA の結果から、前節で示した機械参加型リスクマネジメントの適用効果に関する３つの仮説、（１）不確実さの低減による意思決定の合理化、（２）利害関係者間の合意形成による取引コストの低減、（３）システムⅠとシステムⅡの連携による認知バイアスの影響の緩和、の妥当性が検証された。表 5.8 の“肯定的”な切片データは、after での具体的な効果や期待を語っており、その裏付けとなる。一方で、表 5.8 には“否定的”な切片データも含まれる。ただし、これらは仮説そのものに対する否定というよりも、むしろ、データ収集・蓄積・前処理の負荷（H4-2、H4-3）、組織マインド醸成の難しさ（H4-4、H4-5、H4-6）、データの独り歩き[12]への懐疑（I3-1、I3-2、I4-6）、失敗根本原因の定量化の困難性（H3）、リスクをタブー視する集団心理（I4-7）など、目指す姿への移行過程における課題や阻害要因について言及していると考えられる。提案アプローチの適用効果を一層高めるには、今後、これらの課題や阻害要因の緩和、解消にも継続的に取り組んでいく必要がある。

以下、各コードに対応する主要なインタビュー・データを掲載する。

コード１：不確実さやあいまいさの低減

- *before/after で、プロジェクト失敗の要因がより根拠のあるわかりやすさになったと思う。以前は、経験則だった。規模が大きいとか、初号機の後のやつはヤバいよねとか、漠然とした経験値でヤバそうな匂いがするプロジェクトがうわさになっている感じだった。それが今は、定量的にやはりそうかっていう根拠みたいなものに変わってきている。(I1-1:肯定的)*

[12] 実態と乖離した過剰なデータ依存.

- *開発の途中でわかるようになった。以前は、規模が大きくて怪しいなど、最初だけしかわからなかった。そして、わからないねって言っている中で、やっぱりだめだったかとか、そうでもなかったかと言っていたのが、開発の途中でも設計変更の数とか、DR（デザインレビュー）の遅れの状態とかでわかるようになってきたというのが、すごい進化だと思う。(I1-2:肯定的)*
- *データに基づいた話ができるというところは、すごく有効だと思う。また、人が不足したり、リスクの匂いを感じられない人を補えるというところが、after としてはすごく良いと思う。ただし、前処理や、データが取り切れないということになると、やはりスタートしようとした時点でデータがないと始められないので、そこを如何に普段からできるようにするかということをやらないと、逆に作業が増えてしまう可能性もあるような気がする。(H4-3:否定的= データ収集・蓄積・前処理の負荷)*

コード 2 ：利害関係者間の合意形成

- *プロジェクトリスクについて第三者への説明は容易になったと思う。まだ、残念ながら説明変数が足りていないが、もっと色々なものが測れるようになったら、プロジェクト側に、例えば、品質保証について説明する際、こちら側が感じた感覚だけで説得するよりもデータに基づいて説明した方が、聞く側もやはりそういう風にデータに出ているのかと思ってもらえるし、説明する側もデータを使うことで客観的な話ができると思う。そういう意味では、やはりデータがあった方が絶対に説得しやすいと思っている。(H2-1:肯定的)*
- *リスク予測を組織内に展開していくための一番大きな壁は、やはりマインドなのかも知れない。何をやるにしても、結局、そこに行き着く気がする。その技術が必要だって思ってもらえたり、それがあると良いねという認識を皆にもってもらえるようになると、前向きにとらえてもらえる。今のままだと、一部の人たちだけが何かやってるねで終わってしまうかも知れない。そこの境目を乗り越えないといけないと思っている。(H4-5:否定的=組織マインド醸成の難しさ)*
- *before/after のマイナス面は、数字が独り歩きしたときの怖さ。まだ、自分たちがいる間は数字が独り歩きしないように、うまい使い方をしようとするが、他部門を見ていると、先頭で引っ張っているリーダーたちがいなくなった途端に*

活動が形骸化してしまうなど、本当に正しい運用を継続できるのか心配になることがある。仕組みだけ残って、魂が抜けるというイメージ。そこは、マイナス面というよりも、気をつけておかなきゃいけないことかと思う。自分たちは、まだ本格的に展開しているわけではないので、正直、マイナス面はよくわからない。ただし、怖いのは、そうなる可能性はリスクとして常にあるということ。（I4-6:否定的= データの独り歩きへの懐疑）

コード３：データによる認知バイアスの補正

- *before/after でプロジェクト失敗の要因は、どちらかと言えば、明確になったと思う。今までも感覚的にはすごく感じていたものが、データに基づいて確かめられるという面と、予想外に違うものが入っているような気がするという面がある。例えば、設計変更などはそこまでプロジェクト失敗に影響があるとは思っていなかった。（H1-1:肯定的）*
- *before/after で、事業の戦略会議に情報を出せるようになったのは非常に大きな成果だと思っている。今まではその会議に出ている人たちの経験でものごとを決定していたのが、客観的なデータとして情報を出せるようになったというところは非常に大きな変化だと思う。これはお前の考え方なのかって聞かれても、そうじゃないです、データが言ってるんですと答えられるようになった。それは今までにはなかった。（I2:肯定的）*
- *プロジェクトで火を噴いているところに何回も入った経験があるが、いろいろ悪いことが起きている中で、皆が一生懸命、何とかしようとしているときに、何か新しく悪いことが起きても、誰も見ようとしない。見なかったことにする。それは今言うな、今言ったらこっちが進まなくなる、今この歩みを止めたら終わりだから言うな、あまりうるさかったら切るぞ、という雰囲気になる。こうした一致団結モードをどうするかというのは非常に難しい問題。例え、定量的にリスクを測って提案しても、最後は、リスクがあっても仕方ない、目を閉じるみたいなデータの使われ方をすると、まったく意味がない。最後に判断するのは人間なので。（I4-7:否定的= リスクをタブー視する集団心理）*

5.6. まとめ

本章では、まず、人間の意思決定プロセスを中心に置き、それと機械学習モデルが相互作用する機械参加型（machine-in-the-loop）意思決定プロセスを提示した。人間の意思決定プロセスはデフォルトではシステムⅠが起動しているが、機械学習の介入によりシステムⅡが喚起され、人間と機械学習が協調した合理的な意思決定が促される。その結果、人間の意思決定における認知バイアスの影響の低減が期待できる。

プロジェクト・リスクマネジメントは一連の意思決定プロセスとみなすことにより、その各ステップに機械学習モデルの構築・活用のステップを対応付けた機械参加型（machine-in-the-loop）リスクマネジメントを提案した。この提案アプローチに適した機械学習モデルとして、ナイーブベイズ分類器が挙げられる。ナイーブベイズは、ベイズの定理に基づいた機械学習モデルであり、変数間に条件付き独立性の仮定を置くことで確率計算を単純化し、新たな情報追加による事前確率の逐次更新アルゴリズムを実現している。このような確率更新のメカニズムは“ベイズ更新”と呼ばれ、リスクマネジメントにおける人間の思考プロセスに比較的近いと考えられている。

機械参加型リスクマネジメントは、リスクマネジメントを行うプロジェクト実務者と機械学習を活用するデータ分析者の双方がそれぞれ不完全な知識に基づいて行動していることが前提となっている。すなわち、機械参加型リスクマネジメントの知識更新は、プロジェクト側のリスク知識更新のサイクルと機械学習側のモデル化知識更新のサイクルが相互に連携しながら、それぞれ SECI モデルを回すという 2 サイクル構造となる。一方のサイクルが他方のサイクルの駆動力となり、互いの知識を補完しながらスパイラルアップしていくことで、機械参加型リスクマネジメントの継続的な進化が期待される。

機械参加型リスクマネジメントの適用効果として、（１）不確実さの低減による意思決定の合理化、（２）利害関係者間の合意形成による取引コストの低減、（３）システムⅠとシステムⅡの連携による認知バイアスの影響の緩和、などが想定される。そこで、提案アプローチの適用部門の実務者 2 名に対して、新たにインタビュー調査を実施した。インタビュー結果を演繹的テーマティック・アナリシス法（TA: Thematic Analysis）を用いて分析し、上記仮説の妥当性を検証した。その一方、データ収集・蓄積・前処理の負荷や組織マインド醸成の難しさなど、いくつかの阻害要因も明らかと

なり、その緩和や解消にも今後、継続的に取り組んでいく必要がある。

第6章 予測精度と解釈可能性を両立した機械学習モデルの提案と評価

6.1. はじめに

本章では、人間と機械学習モデルの協調について機械学習モデルの側から検討する。6.2 節では、予測精度と解釈可能性のバランスについて考察する。6.3 節では、新たに高精度かつ解釈容易な機械学習モデルを構築する。新しいモデルのベースとなるナイーブベイズ分類器（6.3.1 節）、ナイーブベイズの拡張モデルである TAN（tree augmented naive Bayes）（6.3.2 節）、ナイーブベイズの集団学習モデル（6.3.3 節）、提案手法である SNB（superposed naive Bayes）（6.3.4 節）をそれぞれ説明する。

続いて、提案手法（中間モデルも含む）と主要な機械学習モデルを予測精度と解釈可能性の 2 つの観点から比較する。6.4 節では、実験・評価方法の概要として、比較対象の機械学習モデル（6.4.1 節）、実験で使用するデータセット（6.4.2 節）、予測精度の実験・評価方法（6.4.3 節）、解釈可能性の評価方法（6.4.4 節）について説明する。6.5 節では、予測精度の実験・評価結果（6.5.1 節）、解釈可能性の評価結果（6.5.2 節）、予測精度と解釈可能性のトレードオフ分析の結果（6.5.3 節）を示す。

6.2. 予測精度と解釈可能性の両立

人間と機械学習が効果的に連携するには、予測精度が高く、かつ、解釈可能性に優れた機械学習モデルが必要となる。しかしながら、一般に、機械学習モデルの予測精度と解釈可能性はトレードオフの関係にある。すなわち、解釈可能性に優れた機械学習モデル（例えば、決定木、ルールベース学習器、線形モデル、など）は、通常、予測精度があまり高くなく、逆に、予測精度が高い機械学習モデル（例えば、サポートベクターマシン、集団学習、深層学習、など）は、総じて解釈可能性の面で劣る。ここで、前者はホワイトボックス・モデル、後者はブラックボックス・モデルと呼ばれる。機械学習モデルの予測精度と解釈可能性はどちらも重要な性質であり、機械参加型（machine-in-the-loop）プロセスにおいては、両者をバランスよく備えた機械学習モデルを使用することが望ましい。

予測精度と解釈可能性を両立する方法について、Guidotti et al.（2018）は、ブラックボックス・モデルをリバースエンジニアリングして解釈可能性を高める方法（以下、本博士論文においては、“リバースエンジニアリング法”と呼ぶ）と、最初から予測精度を高めたホワイトボックス・モデルを設計する方法（以下、本博士論文においては、“ホワイトボックス設計法”と呼ぶ）に大別した。前者は、さらに解釈の目的に応じて、モデル全体を説明する“モデル説明型”、特定の結果を解釈する“結果説明型”、入力の変化に対する感度などを調べる“モデル検査型”に分類される。これらの関係は、図 6.1 のようにまとめられる。“リバースエンジニアリング法”と“ホワイトボックス設計法”は、Lipton（2016）の“事後解釈性”（post-hoc interpretability）と“透明性”（transparency）に概ね対応している。

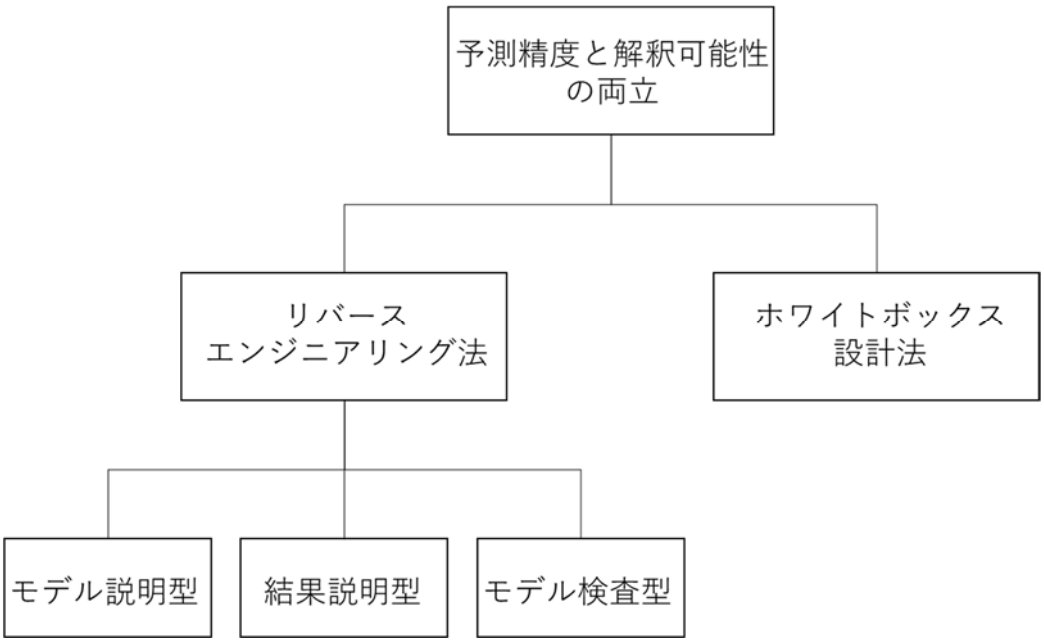


図 6.1：予測精度と解釈可能性を両立する方法
（出典：Guidotti et al. 2018 を参照し筆者修正）

“リバースエンジニアリング法”が適しているのは、そのベースとなるブラックボックス・モデルの性能面の優位性がきわだっている場合である。例えば、画像認識における深層学習は、その正確さやスピードにおいて、もはや人間を凌駕するほどの性能に達している（LeCun et al. 2015）。このような領域では、ブラックボックス・モデル

の恩恵を最大限に享受できる“リバースエンジニアリング法”の選択が妥当となる。一方、“ホワイトボックス設計法”が適しているのは、ブラックボックス・モデルの予測精度面での優位性がそれほど大きくなく、人間との協調が一層重視される場合である。いわゆるビッグデータや精度の高いデータが使えず、かつ、専門家の知識や判断に大きく依存するような領域が主に該当する。リバースエンジニアリングはブラックボックス・モデルのある側面を切り出しているので、モデル説明、結果説明、モデル検査など、解釈の目的に応じてそれぞれ異なる手法を用いる必要があるのに対して、ホワイトボックス・モデルは、モデル構造そのものが“透明”（transparent）であるため、モデルの予測結果の根拠をより直接的に解釈できる。

本博士論文が対象としているリスクマネジメントの領域は、これまで述べてきたように、現場の有識者が多くの有益な情報を暗黙知としてもっており、それを引き出して活用していく必要がある。その一方、リスクという不確実性を伴う事象の知識を取り扱う上で、有識者の暗黙知には先入観やバイアス（偏り）が含まれており、客観的なデータによる補正も必要である。ただし、組織やプロジェクトで収集されるデータの多くは、現状、ビッグデータにはほど遠く、また一部のデータは人手で入力されていたり欠損値が多く含まれるなど、必ずしもデータ品質がよいとは言えない場合がある。こうした状況では、たとえ高精度なブラックボックス・モデルを適用したとしても、その効果を最大限に発揮することは期待できない。したがって、提案アプローチ、すなわちリスクマネジメントにおける知識創造と機械学習の統合アプローチにおいては、予測精度と解釈可能性を両立する方法として“ホワイトボックス設計法”が適していると考えられる。

“ホワイトボックス設計法”においては、例えば、決定木、ルールベース学習器、線形モデルなど、単純で解釈容易な機械学習モデルをベースとして、その高精度な拡張モデルを設計するというアプローチが一般的である。そのようなベースモデルの候補の１つとして、ナイーブベイズ分類器がある。本博士論文では、次節以降、“ホワイトボックス設計法”の立場から、ナイーブベイズ分類器をベースモデルとした高精度な拡張モデルを設計する。

6.3. 高精度かつ解釈容易な機械学習モデルの構築

6.3.1. ナイーブベイズ分類器

ナイーブベイズ分類器（Domingos and Pazzani 1997; Lewis 1998）は、ベイズの定理に基づく単純な機械学習モデルであり、説明変数間に条件付き独立性の仮定を置くことで確率計算を大幅に効率化している。ここで、Y を 2 値の目的変数（1：真、0：偽）、X_1 から X_d を離散値または連続値の説明変数とする。ベイズの定理および条件付き独立性の仮定に基づき、事後確率 $P(Y=1 \mid X_1,\cdots,X_d)$ は、事前確率 $P(Y=1)$ と尤度 $P(X_i \mid Y=1)$ の積を定数 $P(X_1,\cdots,X_d)$ で割ったものとして、次のように表される。

$$P(Y=1 \mid X_1,\cdots,X_d) = \frac{P(Y=1)P(X_1,\cdots,X_d \mid Y=1)}{P(X_1,\cdots,X_d)} = \frac{P(Y=1)\prod_{i=1}^{d} P(X_i \mid Y=1)}{P(X_1,\cdots,X_d)} \quad (1)$$

同様に、事後確率$P(Y=0 \mid X_1,\cdots,X_d)$は、以下のようになる。

$$P(Y=0 \mid X_1,\cdots,X_d) = \frac{P(Y=0)P(X_1,\cdots,X_d \mid Y=0)}{P(X_1,\cdots,X_d)} = \frac{P(Y=0)\prod_{i=1}^{d} P(X_i \mid Y=0)}{P(X_1,\cdots,X_d)} \quad (2)$$

式（1）と式（2）から、$Y=1$ の場合の対数オッズは以下のとおり。

$$W(X) = W_0 + \sum_{i=1}^{d} W_i(X_i) \quad (3)$$

ただし、

$$W(X) = log\frac{P(Y=1 \mid X_1,\cdots,X_d)}{P(Y=0 \mid X_1,\cdots,X_d)}, \quad W_0 = log\frac{P(Y=1)}{P(Y=0)}, \quad W_i(X_i) = log\frac{P(X_i \mid Y=1)}{P(X_i \mid Y=0)}$$

式（3）は線形加法モデルであり、$W_i(X_i)$ は weight of evidence（WoE; Good 1985; Madigan et al. 1997）と呼ばれる。$W_i(X_i)$ の値が正の場合、X_i は $Y=1$ の仮説を支持するエビデンスとなり、逆に負の場合、$Y=0$ の仮説を支持するエビデンスとなる。すなわち、WoE の総和である $W(X)$ の値が大きいほど、$Y=1$ となる（事後）確率が高くなる。ただし、$W(X)$ は確率値ではないので、シグモイド関数などを用いて確率値に変換する必要がある。ナイーブベイズによる $Y=1$ の確率の推定値 $P_{nb}(X)$ は、$W(X)$ のシグモイド変換 $sigmoid$ により以下のように表される。

$$P_{nb}(X) = sigmoid\big(c_0 + c_1 W(X)\big) = \frac{1}{1 + exp\Big(-\big(c_0 + c_1 W(X)\big)\Big)} \quad (4)$$

ただし、パラメータ c_0, c_1 は、学習データの尤度が最大、すなわちデータへの当てはまりが最良となるように調整される。

ナイーブベイズは単純な機械学習モデルであるため、予測結果が安定しており、ノイズに対して頑健で、かつ計算効率がよいという特徴がある。説明変数間の条件付き独立性の仮定はしばしば現実とはかけ離れているが、多くの実世界の応用においてナイーブベイズは十分に実用的な性能を示すことが知られている（Domingos and Pazzani 1997; Menzies et al. 2007; Rish 2001; Zang 2004）。Rish（2001）は、ナイーブベイズの予測精度が変数間の依存関係の強さに直接的には関連しないことをモンテカルロ・シミュレーションの実験で確認した。Zhang（2004）は、変数間の依存関係が強い場合においても､それらが各クラスに均等に分布していたり互いに打ち消しあっているならば、ナイーブベイズの予測結果は依然として最適になり得ることを証明した。しかしながら、上記の条件が成立しない場合においては、ナイーブベイズの予測精度は他のより洗練された機械学習モデルと比べて必ずしも高いとは言えない。

別の側面として、ナイーブベイズにおける確率計算の過程は人間の思考プロセスと似ているため、モデルの解釈が容易であるという特徴がある（Kononenko 1993; Kotsiantis 2007）。特に WoE による方法は、医用診断（Heckerman et al. 1992; Madigan et al. 1997）や資源探査（Agterberg et al. 1990; Bonham-Carter et al. 1988）などの解釈可能性が重視される幅広い分野で利用されている。

6.3.2. TAN（tree augmented naive Bayes）

ナイーブベイズは、構造が単純なため計算効率がよく、予測結果が安定しており、ノイズに対して頑健で、モデルの解釈が容易であるという特徴があるが、その一方、一般的な予測精度は必ずしも高いとは言えない。ただし、ナイーブベイズの説明変数間の条件付き独立性の仮定はかなり強い制約であるため、その制約を緩めることで予測精度の改善が期待できる。

Tree augmented naive Bayes（TAN; Friedman et al. 1997）は説明変数間に木構造の依存関係を許容したナイーブベイズの拡張モデルであり、各説明変数は目的変数以外の他の１つの説明変数だけに依存することが可能である。図 6.2 に、さまざまなベイズ分類器の変数間依存関係の構造を示す。図 6.2（a）はナイーブベイズであり、各説明変

数（X_1, X_2, X_3）は目的変数 Y だけに依存しており、説明変数間の依存関係は存在しない。図 6.2（b）は TAN であり、Y からの依存関係を除いた各説明変数間の依存関係が木構造（親が 1 つだけ）となっている。一方、図 6.2（c）は、説明変数 X_3 が複数の説明変数（X_1, X_2）に依存しているため、TAN ではなく一般のベイジアンネットワークとなる。TAN の木構造は、各枝の重みを自己相互情報量（pairwise mutual information）としたときの最大重み全域木（maximum weighted spanning tree）として決定される。Friedman et al. (1997) は、TAN について「計算の効率性と頑健さを維持しつつ、予測精度においてナイーブベイズを大きく上回る」と報告している。

TAN モデルは、ナイーブベイズに特有の欠損値の取り扱い、すなわち確率計算の過程で欠損値は無視されるという性質を利用することにより、等価なナイーブベイズ・モデルに変換できる。図 6.3 に、TAN からナイーブベイズへのモデル変換の例を示す。図 6.3（a）は説明変数 X_3 が目的変数 Y と説明変数 X_1 に依存している TAN モデルであり、この X_3 に対して式 (5) を適用し 2 つの変数 X_3' と X_3'' に分割することにより、図 6.3（b）のナイーブベイズ・モデルに変換することができる。また、変換後のナイーブベイズ・モデルから、必要に応じて、元の TAN モデルの構造を復元することも可能である。

$$X_3' = \begin{cases} X_3, & if\ X_1 = x_{11} \\ NA, & if\ X_1 \neq x_{11} \end{cases}, \quad X_3'' = \begin{cases} X_3, & if\ X_1 = x_{12} \\ NA, & if\ X_1 \neq x_{12} \end{cases} \tag{5}$$

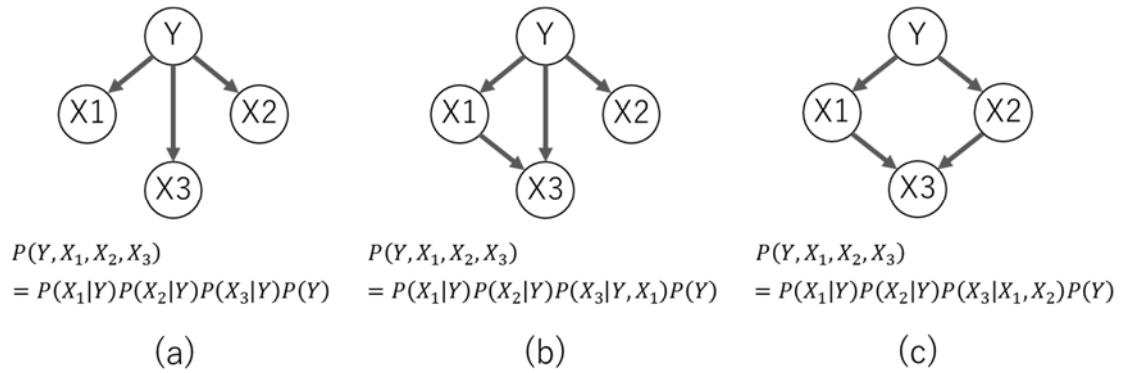
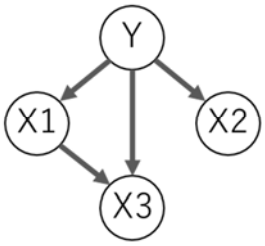


図 6.2：さまざまなベイズ分類器の構造

（出典：Mori and Uchihira 2019）

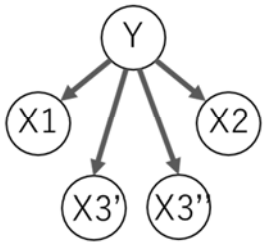
X_1	X_2	X_3	Y
x_{11}	x_{21}	x_{31}	1
x_{11}	x_{22}	x_{31}	1
x_{11}	x_{21}	x_{31}	0
x_{11}	x_{22}	x_{32}	0
x_{12}	x_{21}	x_{32}	1
x_{12}	x_{22}	x_{32}	1
x_{12}	x_{21}	x_{32}	0
x_{12}	x_{22}	x_{31}	0



$$P(Y, X_1, X_2, X_3) = P(X_1|Y)P(X_2|Y)P(X_3|Y, X_1)P(Y)$$

(a)

X_1	X_2	X_3'	X_3''	Y
x_{11}	x_{21}	x_{31}	NA	1
x_{11}	x_{22}	x_{31}	NA	1
x_{11}	x_{21}	x_{31}	NA	0
x_{11}	x_{22}	x_{32}	NA	0
x_{12}	x_{21}	NA	x_{32}	1
x_{12}	x_{22}	NA	x_{32}	1
x_{12}	x_{21}	NA	x_{32}	0
x_{12}	x_{22}	NA	x_{31}	0



$$P(Y, X_1, X_2, X_3) = P(X_1|Y)P(X_2|Y)P(X_3'|Y)P(X_3''|Y)P(Y)$$

(b)

図 6.3：TAN からナイーブベイズへの変換

（出典：Mori and Uchihira 2019）

6.3.3. ナイーブベイズの集団学習モデル

集団学習モデルは、単体のモデルよりも高い予測精度を得るために、複数のモデルの予測結果を集約する手法である。これまで、さまざまな集団学習モデルが提案されているが、もっとも代表的な手法としてはバギング（bagging）系の手法とブースティング（boosting）系の手法がある。

バギング（bagging; Breiman 1996）は、元データから無作為復元抽出を繰り返し、それらに適合する単体の予測モデル（通常は決定木）を多数生成して、個々の予測結果を投票または平均化することによって全体としての予測値を得る手法である。バギングの予測精度は、一般に、構成要素である単体のモデルの予測精度よりも高くなる。

バギングの拡張モデルであるランダムフォレスト（random forest; Breiman 2001）は、説明変数についてもランダム選択を行うことで、個々のモデルのバリアンス（予測値のばらつき）を意図的に高め、全体としての予測精度をさらに向上させている。

一方、ブースティング（boosting; Freund and Schapire 1997）は、直前の予測結果に基づいて各データの重みを逐次的に更新し、それに対応する予測モデル（決定木など）を順次生成することで集団学習モデルを構築する手法である。バギングと同様に、個々の予測結果の集約（加重平均など）により全体としての予測値を得る。AdaBoost は、Freund and Schapire（1997）によって提案された最初の実用的なブースティング・アルゴリズムであり、各反復において誤った予測結果の重みが増加し正しい予測結果の重みが減少するように、各データの重みを順次更新する。AdaBoost は、しばしば予測精度を劇的に改善するが、過学習しやすいという欠点がある。そこで、Friedman（2002）は、元データから無作為非復元抽出したデータに単体モデルを当てはめ、その重みの計算に勾配降下法を用いる確率勾配ブースティング（stochastic gradient boosting）を提案している。

集団学習モデルは、そのベースとなる予測モデルに決定木を用いることが多い。その理由として、集団学習にはバリアンスを低減する性質があり、不安定な予測モデルの過学習を防ぐことで全体の予測精度を向上させるため、決定木などの低バイアス（予測値の偏り）・高バリアンスの予測モデルが適している。一方、ナイーブベイズは高バイアス・低バリアンスの安定した予測モデルであるため、集団学習の恩恵を十分に受けることができないと考えられてきた（Bauer and Kohavi 1999）。しかしながら、主にバリアンスの低減が目的のバギング系の手法と異なり、ブースティング系の手法はバイアスとバリアンスの両方に対する効果が期待できるとの報告もある（Webb 2000）。これは、ブースティングがナイーブベイズの弱点である偏りの大きさ（高バイアス）を補完する可能性があることを意味している。

そこで、ここでは、ナイーブベイズをベース予測モデルとして確率勾配降下法により AdaBoost を拡張した、新しいナイーブベイズの集団学習モデルを提案する。本博士論文では、以降、ナイーブベイズ・アンサンブル（naive Bayes ensemble; NBE）と呼ぶ。図 6.4 に、提案したアルゴリズムの概要を示す。元の AdaBoost とは、ステップ 4a の元データの無作為非復元抽出、ステップ 4c および 4e の勾配降下法によるパラメータ計算、ステップ 4d の out-of-bag 推定値（Breiman 2001）の計算（予測結果の集約に利

用）などが異なっている。ブースティングの適用により、元のナイーブベイズと比較して予測精度の向上が期待できる一方、集団学習モデルに共通の課題として解釈可能性は低下する。

1. Initialize sample weights $w_i = 1/n$, $i = 1, 2, \cdots, n$.
2. Build a naive Bayes classifier $nb_1(X)$ from $X = \{x_1, x_2, \cdots, x_n\}$.
3. Set $\beta_1 \leftarrow 1$.
4. For $t = 2$ to m:
   a. Draw a random subsample X_a of size ηn $(0 < \eta < 1)$ from X without replacement.
   b. Build $nb_t(X_a)$ with a threshold s_t so as to minimize $e_t = \frac{\sum_{x_i \in X_a} w_i I\left(y_i \neq C_{nb}^{(t)}(x_i|s_t)\right)}{\sum_{x_i \in X_a} w_i}$.
   c. Compute $\alpha_t = \nu \log((1 - e_t)/e_t)$, where ν $(0 < \nu < 1)$ is a shrinkage parameter.
   d. Compute $\beta_t = \nu \log((1 - e_{oob})/e_{oob})$, where $e_{oob} = \frac{\sum_{x_i \in X - X_a} w_i I\left(y_i \neq C_{nb}^{(t)}(x_i|s_t)\right)}{\sum_{x_i \in X - X_a} w_i}$.
   e. Set $w_i \leftarrow w_i \, exp\left(\alpha_t \, I\left(y_i \neq C_{nb}^{(t)}(x_i|s_t)\right)\right)$, $i = 1, 2, \cdots, n$.
5. Output $P_{nbe}(x) = \frac{\sum_{t=1}^{m} \beta_t P_{nb}^{(t)}(x)}{\sum_{t=1}^{m} \beta_t}$.

図 6.4：ナイーブベイズ・アンサンブル（出典：Mori and Uchihira 2019）

6.3.4. SNB（superposed naive Bayes）

前節のナイーブベイズ・アンサンブル（NBE）は、元のナイーブベイズと比較して予測精度の向上が期待できる一方、そのトレードオフとして解釈可能性は低下する。そこで、本博士論文では、NBE を線形近似して単一のナイーブベイズに変換する方法、superposed naive Bayes（SNB）を提案する。すなわち、SNB の構築は、NBE を構築する第 1 のステップと、NBE を線形近似して単一のナイーブベイズに変換する第 2 のステップからなる 2 段階アプローチとなる。第 1 のステップは主に予測精度の向上に寄与し、第 2 のステップは主に解釈可能性の向上に寄与する。ナイーブベイズは一種の線形加法モデルであるため、線形近似された NBE からナイーブベイズへの変換は WoE

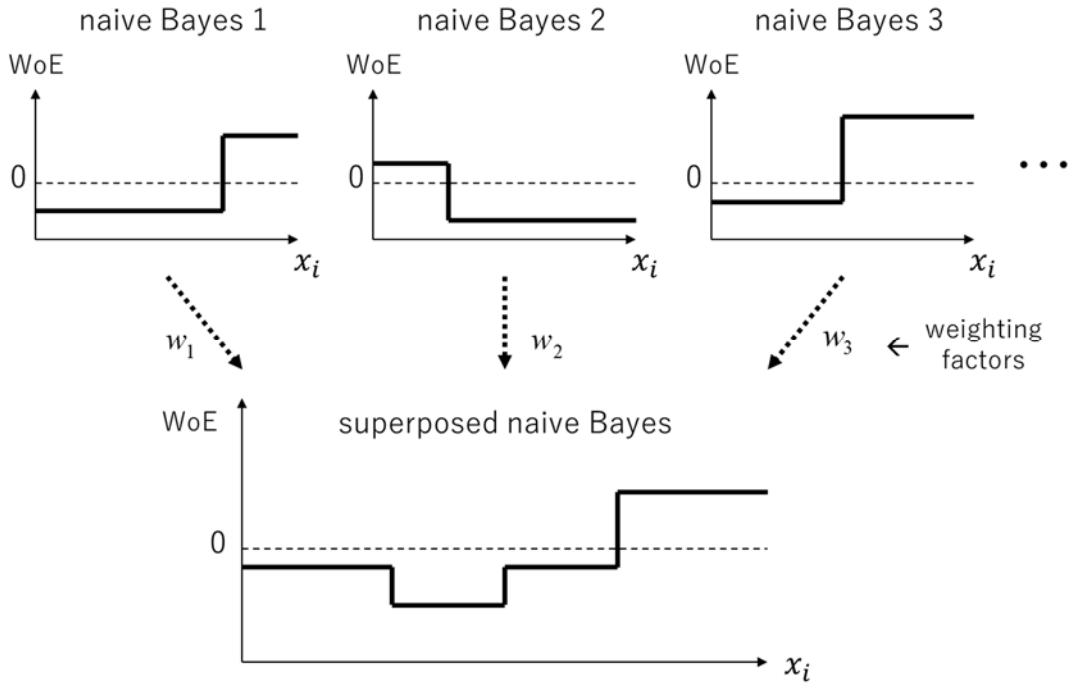


図 6.5：WoE の重ね合わせによる SNB の生成

（出典：Mori and Uchihira 2019 に加筆）

（weight of evidence）の重ね合わせとして表現できる。図 6.5 に、WoE の重ね合わせによる SNB 生成のイメージを示す。

WoE の重ね合わせを行うには、まず、NBE の非線形項を線形近似する必要がある。式（3）と式（4）から、NBE の予測結果は以下にように表される。

$$P_{nbe}(X) = \frac{\sum_{t=1}^{m} \beta_t P_{nb}^{(t)}(X)}{\sum_{t=1}^{m} \beta_t} = \frac{\sum_{t=1}^{m} \beta_t \, sigmoid\left(c_0^{(t)} + c_1^{(t)} W^{(t)}(X)\right)}{\sum_{t=1}^{m} \beta_t}$$

$$= \frac{\sum_{t=1}^{m} \beta_t \, sigmoid\left(c_0^{(t)} + c_1^{(t)} \sum_{i=0}^{d} W_i^{(t)}(X)\right)}{\sum_{t=1}^{m} \beta_t} \quad (6)$$

ここで、線形近似として $sigmoid\left(c_0^{(t)} + c_1^{(t)} W^{(t)}(X)\right) \cong a_t + b_t W^{(t)}(X)$ を仮定すると、式（6）は SNB の予測結果を表す以下の式に変換できる。

$$P_{snb}(X) = W_0' + \sum_{i=1}^{d} W_i' \quad (7)$$

ただし、

$$W_0' = \frac{\sum_{t=1}^{m} \beta_t \left(a_t + b_t W_0^{(t)}(X)\right)}{\sum_{t=1}^{m} \beta_t}, \qquad W_i' = \frac{\sum_{t=1}^{m} \beta_t b_t W_i^{(t)}(X)}{\sum_{t=1}^{m} \beta_t}$$

上記シグモイド関数の線形近似においては、テイラー近似を使用する。すなわち、$sigmoid\left(c_0^{(t)} + c_1^{(t)} W^{(t)}(X)\right)$ の 1 次近似は、$c_0^{(t)}$ の周りのテイラー展開として以下のように導出される。

$$sigmoid\left(c_0^{(t)} + c_1^{(t)} W^{(t)}(X)\right) = \sum_{n=0}^{\infty} \frac{sigmoid^{(n)}\left(c_0^{(t)}\right)}{n!}\left(c_1^{(t)} W^{(t)}(X)\right) \cong a_t + b_t W^{(t)}(X) \quad (8)$$

ただし、

$$a_t = \frac{1}{1 + exp\left(-c_0^{(t)}\right)}, \qquad b_t = \frac{c_1^{(t)} exp\left(-c_0^{(t)}\right)}{\left(1 + exp\left(-c_0^{(t)}\right)\right)^2}$$

式（3）と式（7）の比較からわかるように、SNB はナイーブベイズと同じ構造をもっている。すなわち、SNB は重ね合わせた WoE である W_i' $(i = 1, 2, \cdots, d)$ をもつ単一のナイーブベイズとみなすことができる。ただし、SNB の WoE は元のナイーブベイズの WoE よりもかなり複雑な形状をしており、その複雑さの中に NBE のモデル構造の特徴が反映されていると考えられる。

図 6.6 は、ナイーブベイズの WoE（上側）と SNB の WoE（下側）を比較したものである。それぞれのグラフを解釈すると、ナイーブベイズ（上側）では、*LOC_EXECUTABLE* < 70 ではリスクを低減する方向（*WoE* = -0.5）に作用するが、70 付近を上回ると一転、急激なリスク増加因子（*WoE* = 1.5）に切り換わる。一方、SNB（下側）では、WoE の値の変化がスムージングされており、*LOC_EXECUTABLE* > 10 辺りから徐々にリスクが上昇していく。そして、*WoE* = 0 の直線と交差する値、すなわち、*LOC_EXECUTABLE* = 40 付近において、リスク低減傾向（*WoE* < 0）からリスク増加傾向（*WoE* > 0）に緩やかに切り換わる。

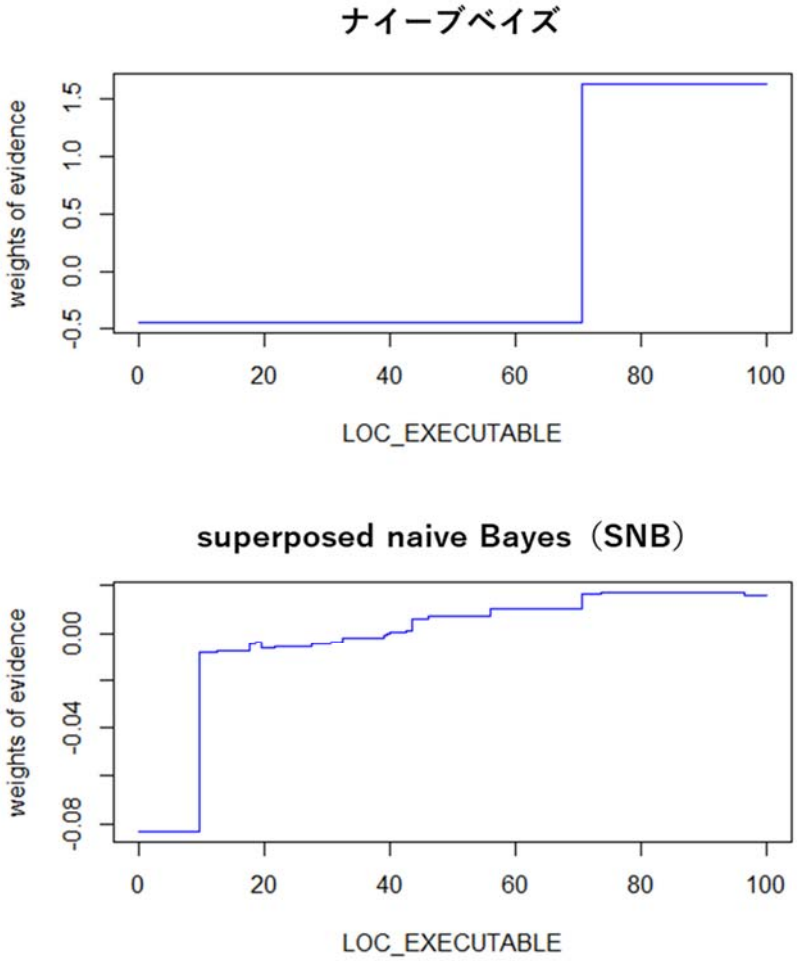


図 6.6：ナイーブベイズの WoE と SNB の WoE の比較

提案手法（SNB）におけるナイーブベイズの集団学習モデルに対するテイラー近似のアイデアは、Ridgeway et al.（1998）に触発されたものである。主な違いとして、Ridgeway et al.（1998）はブースティングの結果を集約する結合関数に対してテイラー近似を適用しているが、SNB では個々のナイーブベイズの予測結果に対するシグモイド補正の部分にテイラー近似を適用している。また、Ridgeway et al.（1998）はオリジナルの AdaBoost を使用しているが、SNB では確率的勾配降下法などを使用した拡張 AdaBoost を採用することで、さらなる予測精度の向上を実現している。

6.4. 実験・評価方法の概要

6.4.1. 評価対象の機械学習モデル

表 6.1 は、実験で評価対象とした機械学習モデル（提案手法も含む）の一覧である。１列目はカテゴリー、２列目は機械学習モデルの名称、３列目は以後使用する略称、４列目は実験で使用した実装を示す。最初の 14 個は、さまざまなアルゴリズムを用いた主要な機械学習モデルであり、機械学習ソフトウェア Weka（Waikato Environment for Knowledge Analysis; Witten et al. 2011）の環境から利用可能である。残りの６つは、提案手法およびその中間モデルであり、Java と R で実装されている。

- OneR（Holte 1993）は、単純なルールベースの学習器であり、予測に最小誤差変数を使用している。
- JRip は、RIPPER（Cohen 1995）の Weka 版実装であり、IREP（incremental reduced error pruning）アルゴリズムに基づいてルールを学習する。
- J48 は、情報エントロピーを使用した代表的な決定木アルゴリズムである C4.5（Quinlan 1993）の Weka 版実装である。
- NBTree（Kohavi 1996）は、決定木の末端の葉がナイーブベイズとなっているハイブリッド・モデルである。
- リッジロジスティック回帰（RLR; Le Cessie and Van Houwelingen 1992）は、リッジ推定量を使用した多項ロジスティック回帰モデルである。
- サポートベクターマシン（SVM; Burges 1998）は、２つのクラスを分離する最大マージン超平面を見つける二項分類器である。
- 多層パーセプトロン（MLP; Jain et al. 1996）は、入力層と出力層と１つ以上の隠れ層をもつフィードフォワード・ニューラルネットワークである。
- ナイーブベイズ分類器（Domingos and Pazzani 1997; Lewis 1998）は、もっとも単純な構造のベイジアンネットワークであり、各説明変数に対して正規分布を仮定する連続ナイーブベイズ（NBc）と各説明変数を事前に離散化する離散ナイーブベイズ（NBd）が存在する。
- Tree augmented naive Bayes（TAN; Friedman et al. 1997）は、説明変数間の条件付

き独立性の仮定を緩和したナイーブベイズの拡張モデルである。

- Averaged one-dependence estimators（AODE; Webb et al. 2005）は、すべての ODEs（one dependence estimators; 目的変数と 1 つの説明変数のみに依存するベイジアンネットワーク）の予測値を平均化することで全体としての予測値を得るベイジアン集団学習モデルである。
- Hidden naive Bayes（HNB; Jiang et al. 2009）は、各説明変数に対して、他のすべての変数からの影響を結合した“隠れ”親ノードをもつベイジアン予測モデルである。
- AdaBoost（AdaBst; Freund and Schapire 1997）は、反復ごとの予測結果に基づいて各データの重みを逐次更新するブースティングのもっとも代表的な実装である。本実験では、反復回数を 100 回とした。
- ランダムフォレスト（RF; Breiman 2001）は、元データの無作為復元抽出と説明変数のランダム選択を組み合わせたバギングの高度な拡張アルゴリズムである。本実験では、反復回数を 100 回とした。
- Weka 版とは別に、Java と R で離散ナイーブベイズ（NBd2）を実装している。これは、naive Bayes ensemble（NBE）、および、superposed naive Bayes（SNB）の生成に使用される。本実験では、NBE および SNB の反復回数を 100 回とした。
- Weka 版とは別に、Java と R で TAN（TAN2）を実装している。これは、TAN 版の naive Bayes ensemble（NBE2）、および、TAN 版の superposed naive Bayes（SNB2）の生成に使用される。本実験では、NBE2 および SNB2 の反復回数を 100 回とした。

表 6.1：評価対象の機械学習モデル（出典：Mori and Uchihira 2019）

Category	Technique	Abbr.	Impl.
Rule-based learners	OneR classifier	OneR	Weka
Rule-based learners	RIPPER	JRip	Weka
Decision trees	C4.5	J48	Weka
Decision trees	NBTree	NBTree	Weka
Regression models	Ridge logistic regression	LR	Weka
Support vector machines	Support vector machine	SVM	Weka
Neural networks	Multilayer perceptron	MLP	Weka
Bayesian learners	Gaussian naive Bayes	NBc	Weka
Bayesian learners	Discrete naive Bayes	NBd	Weka
Bayesian learners	Tree-augmented naive Bayes	TAN	Weka
Bayesian ensemble learners	Averaged one-dependence estimators	AODE	Weka
Bayesian learners	Hidden naive Bayes	HNB	Weka
Decision tree ensemble learners	AdaBoost	AdaBst	Weka
Decision tree ensemble learners	Random forest	RF	Weka
Bayesian learners	Discrete naive Bayes	NBd2	Java, R
Bayesian learners	Tree-augmented naive Bayes	TAN2	Java, R
Bayesian ensemble learners	Naive Bayes ensemble	NBE	Java, R
Bayesian ensemble learners	Naive Bayes ensemble + TAN	NBE2	Java, R
Bayesian ensemble learners	Superposed naive Bayes	SNB	Java, R
Bayesian ensemble learners	Superposed naive Bayes + TAN	SNB2	Java, R

6.4.2. データセットの概要

本研究では、ソフトウェア開発プロジェクトに関連する 2 つの異なるデータソースから選択した 13 の実データセットを使用した。1 つ目のデータソースは、 Shepperd et al.（2013）によって作成され、Menzies et al.（2016）によって提供されている NASA（アメリカ航空宇宙局）MDP（metrics data program）データセットのクリーン版であり、そこでは問題のあるデータ（例えば、矛盾する特徴量や非現実的な値をもつデータなど）や不要なデータ（例えば、定数値のみの特徴やまったく同一のデータなど）が削除されている。2 つ目のデータソースは、Kamei et al.（2013）によって提供された JIT（just-in-time）データセットであり、 有名なオープンソースプロジェクトから抽出されたさまざまな変更レベルのメトリクスが含まれている。

1 つ目のデータソース、MDP データセットの概要を表 6.2 に示す。ここでは、MDP データセットから、11 個のデータセット（MC2、KC3、MW1、CM1、PC1、PC2、PC3、PC4、PC5、MC1、JM1）を選択した。これらは元々、宇宙船用機器、地球周回衛星用の飛行ソフトウェア、リアルタイム予測地上システム、地上データ用のストレージ管理システムなど、さまざまな NASA の実運用システムから収集されたものである。各データセットは、ソフトウェアモジュールのソースコード行数、循環的複雑度など、品質やコストに影響するさまざまな属性値を含んでいる。

2 つ目のデータソース、JIT データセットの概要を表 6.3 に示す。ここでは、Bugzilla と Columba の 2 つのデータセットを選択した。 Bugzilla は、元々Mozilla プロジェクトによって開発された Web ベースのバグ管理システムであり、Columba は、Java で実装されたオープンソースの電子メールクライアントである。各データセットにはバイナリの目的変数と 14 個の数値的な説明変数が含まれている。このうち、LAn、LDn、LTn、NUCn の 4 変数は、説明変数間の相関を低くするためにあらかじめ正規化されている。

表 6.2：NASA MDP データセットの概要（出典：Lessmann et al. 2008）

		NASA MDP datasets (cleaned version)										
		MC2	KC3	MW1	CM1	PC1	PC2	PC3	PC4	PC5	MC1	JM1
LOC counts	LOC_total	X	X	X	X	X	X	X	X	X	X	X
	LOC_blank	X	X	X	X	X		X	X	X	X	X
	LOC_code_and_comment	X	X	X	X	X	X	X	X	X	X	X
	LOC_comments	X	X	X	X	X	X	X	X	X	X	X
	LOC_executable	X	X	X	X	X	X	X	X	X	X	X
	Number_of_lines	X	X	X	X	X	X	X	X	X	X	
Halstead attributes	content	X	X	X	X	X	X	X	X	X	X	X
	difficulty	X	X	X	X	X	X	X	X	X	X	X
	effort	X	X	X	X	X	X	X	X	X	X	X
	error_est	X	X	X	X	X	X	X	X	X	X	X
	length	X	X	X	X	X	X	X	X	X	X	X
	level	X	X	X	X	X	X	X	X	X	X	X
	prog_time	X	X	X	X	X	X	X	X	X	X	X
	volume	X	X	X	X	X	X	X	X	X	X	X
	num_operands	X	X	X	X	X	X	X	X	X	X	X
	num_operators	X	X	X	X	X	X	X	X	X	X	X
	num_unique_operands	X	X	X	X	X	X	X	X	X	X	X
	num_unique_operators	X	X	X	X	X	X	X	X	X	X	X
McCabe attributes	cyclomatic_complexity	X	X	X	X	X	X	X	X	X	X	X
	cyclomatic_density	X	X	X	X	X	X	X	X	X	X	
	design_complexity	X	X	X	X	X	X	X	X	X	X	X
	essential_complexity	X	X	X	X	X	X	X	X	X	X	X
Miscellaneous	branch_count	X	X	X	X	X	X	X	X	X	X	X
	call_pairs	X	X	X	X	X	X	X	X	X	X	
	condition_count	X	X	X	X	X	X	X	X	X	X	
	decision_count	X	X	X	X	X	X	X	X	X	X	
	decision_density	X	X	X	X	X	X	X	X			
	design_density	X	X	X	X	X	X	X	X	X	X	
	edge_count	X	X	X	X	X	X	X	X	X	X	
	essential_density	X	X	X	X	X	X	X	X	X	X	
	global_data_complexity	X	X							X	X	
	global_data_density	X	X							X	X	
	maintenance_severity	X	X	X	X	X	X	X	X	X	X	
	modified_condition_count	X	X	X	X	X	X	X	X	X	X	
	multiple_condition_count	X	X	X	X	X	X	X	X	X	X	
	node_count	X	X	X	X	X	X	X	X	X	X	
	normalized_cyclomatic_compl.	X	X	X	X	X	X	X	X	X	X	
	parameter_count	X	X	X	X	X	X	X	X	X	X	
	percent_comments	X	X	X	X	X	X	X	X	X	X	
Number of code attributes		39	39	37	37	37	36	37	37	38	38	21
Number of modules		125	194	253	327	705	745	1077	1287	1711	1988	7782
Number of defective modules		44	36	27	42	61	16	134	177	471	46	1672
Percentage of defective modules		35.2%	18.6%	10.7%	12.8%	8.7%	2.1%	12.4%	13.8%	27.5%	2.3%	21.5%

表 6.3：JIT データセットの概要（出典：Kamei et al. 2013）

Dimension	Name	Definition	Rationale	JIT datasets	
				Bugzilla	Columba
Diffusion	NS	Number of modified subsystems	Change modifying many subsystems are more likely to be defect-prone.	X	X
	ND	Number of modified directories	Changes that modify many directories are more likely to be defect-prone.	X	X
	NF	Number of modified files	Changes touching many files are more likely to be defect-prone.	X	X
	Entropy	Distribution of modified code across each file	Changes with high entropy are more likely to be defect-prone, because a developer will have to recall and track large numbers of scattered changes across each file.	X	X
Size	LAn	Lines of code added (normalized)	The more lines of code added, the more likely a defect is introduced.	X	X
	LDn	Lines of code deleted (normalized)	The more lines of code deleted, the higher the chance of a defect.	X	X
	LTn	Lines of code in a file before the change (normalized)	The larger a file, the more likely a change might introduce a defect.	X	X
Purpose	FIX	Whether or not the change is a defect fix	Fixing a defect means that an error was made in an earlier implementation, therefore it may indicate an area where errors are more likely.	X	X
History	NDEV	The number of developers that changed the modified file	The larger the NDEV, the more likely a defect is introduced, because files revised by many developers often contain different design thoughts and coding styles.	X	X
	AGE	The average time interval between the last and the current change	The lower the AGE (i.e., the more recent the last change), the more lilely a defect will be introduced.	X	X
	NUCn	The number of unique changes to the modified files (normalized)	The larger the NUC, the more likely a defect is introduced, because a developer will have to recall and track many previous changes.	X	X
Experience	EXP	Developer experience	More experienced developers are less likely to introduce a defect.	X	X
	REXP	Recent developer experience	A developer that has often modified the files in recent months is less likely to introduce a defect, because he/she will be more familiar with the recent developments in the system.	X	X
	SEXP	Developer experience on a subsystem	Developer that are familiar with the subsystems modified by a change are less likely to introduce a defect.	X	X
Number of metrics				14	14
Number of changes				4620	4455
Number of defective changes				1696	1361
Percentage of defective changes				36.7%	30.5%

6.4.3. 予測精度の評価方法

異なる種類の機械学習モデルに対する予測精度の評価方法として、Ghotra et al.（2015）と同様のアプローチを採用した。図 6.7 に、実験におけるモデル構築と性能評価の概要を示す。

機械学習モデル（図 6.7 の predictors）の予測精度は、5 分割交差検証を用いて評価した。すなわち、データセット（同 datasets）をランダムに 5 分割して、その内の 4 つ（80%）を訓練データ（同 training data）、残りの 1 つ（20%）をテストデータ（同 testing data）として使用し、このプロセスを 5 回繰り返す。さらに、異なる分割を用いて、この 5 分割交差検証を 5 回繰り返し、合計 25 回の反復を実施した。

予測精度の評価指標としては、ROC 曲線の AUC（area under the ROC curve; Fawcett 2006）を使用した。ROC 曲線は、閾値をさまざまな値に変化させたときの偽陽性率（X 軸）と真陽性率（Y 軸）のプロットであり、AUC はその下側の面積となる。AUC は 0 から 1 までの範囲の値をとり、値が大きいほど精度が高いことを意味する。また、AUC が 0.5 未満の場合は、ほぼランダム推定と変わらないことを意味する。

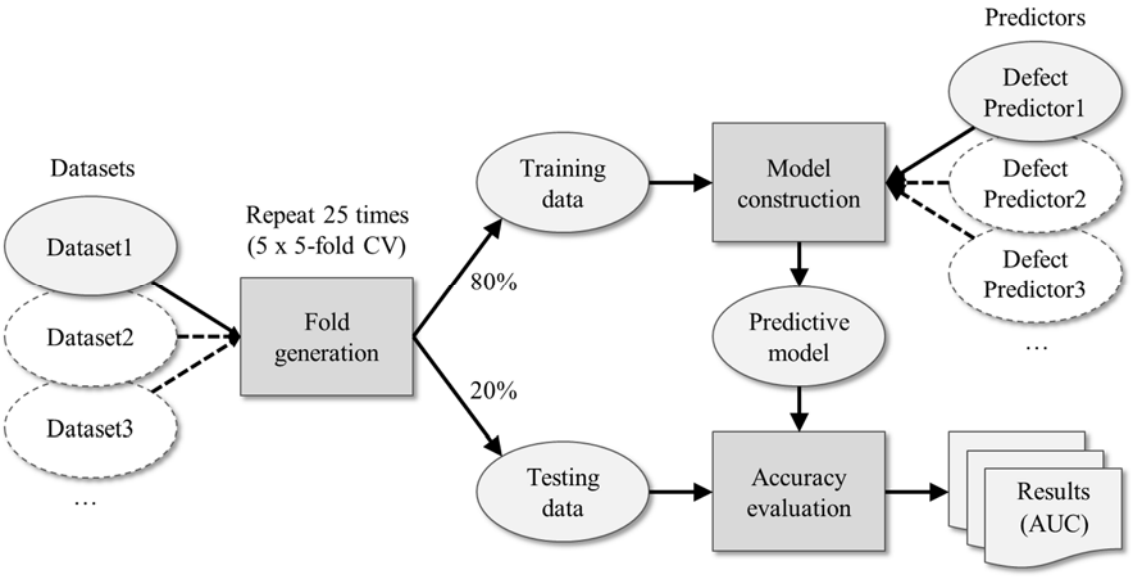


図 6.7：実験におけるモデル構築と性能評価の概要

（出典：Ghotra et al. 2015 に加筆）

続いて、ランダム分割やデータセットの違いを考慮した上での機械学習モデルの予測精度の統計的有意差を判定するために、Scott-Knott 検定（Jelihovschi et al. 2014）を行った。Scott-Knott 検定は、階層クラスター分析を用いて検定対象をグループ化するものである。図 6.8 は、Scott-Knott 検定の二重適用による機械学習モデルのランク付けアプローチを示している。まず、1 回目の Scott-Knott 検定を個別のデータセットごとに実施し、次に、最初の検定結果に対して 2 回目の Scott-Knott 検定を実施することで、統計的に有意な予測精度をもつグループごとの機械学習モデルのランク付けが示される。

得られたランクは、fractional rank（Bury and Wagner 2008）を逆順にしてアイテムの総数で割った reversed fractional rank（RFR）に変換した。ここで、fractional rank FR を同じグループ内の順序ランクの平均として、アイテムの総数を Nとすると、reversed fractional rank RFRは、以下の式で与えられる。

$$RFR = (N - FR + 1)/N \tag{9}$$

図 6.9 に、RFR の計算例を示す。RFR は 0 から 1 までの範囲の値をとり、値が大きいほどランクが上位であることを意味する。また、RFR の合計は常に $(N+1)/2$ に等しくなる。

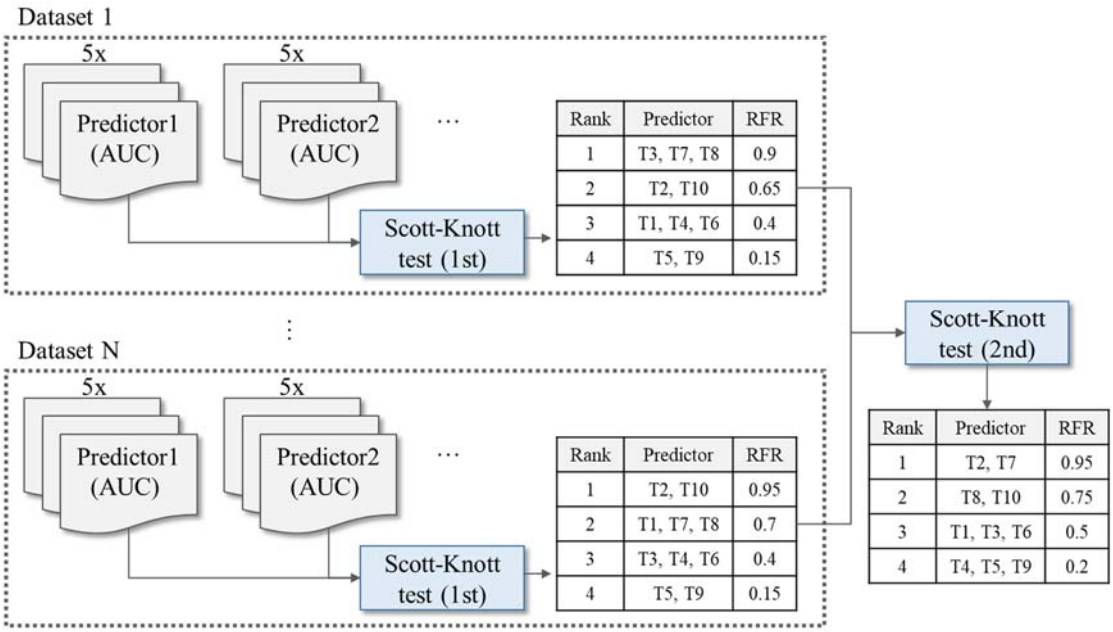


図 6.8：Scott-Knott 検定の二重適用によるランク付けアプローチ

（出典：Ghotra et al. 2015 に加筆）

Group	Item	Ordinal rank	Fractional rank	Reversed fractional rank (RFR)
Group 1	A	1	(1+2+3) / 3 = 2	(10-2+1) / 10 = 0.9
	B	2	2	0.9
	C	3	2	0.9
Group 2	D	4	(4+5) / 2 = 4.5	(10-4.5+1) / 10 = 0.65
	E	5	4.5	0.65
Group 3	F	6	(6+7+8) / 3 = 7	(10-7+1) / 10 = 0.4
	G	7	7	0.4
	H	8	7	0.4
Group 4	I	9	(9+10) / 2 = 9.5	(10-9.5+1) / 10 = 0.15
	J	10	9.5	0.15

図 6.9：Reversed fractional rank（RFR）の計算例

（出典：Mori and Uchihira 2019）

6.4.4. 解釈可能性の評価方法

機械学習モデルの解釈可能性を評価するには、本来、その評価指標を明確にする必要がある。しかしながら、異なる種類の機械学習モデルに対して解釈可能性を比較・評価することは非常に難しい。例えば、回帰モデルの項数、ルールベース学習器のルール数、決定木のノード数など、モデルの“大きさ”に基づく評価指標はすべて特定のモデルに依存するため、そのまま適用することができない。Freitas（2014）は、異なる種類の機械学習モデルの解釈可能性を完全に客観的な方法で測定することは不可能であると述べている。実験による解釈可能性の評価についても、使用するデータセットの特性や被験者の経験など、さまざまな要因によって容易にバイアスの影響を受ける可能性がある。したがって、本博士論文では、複数の評価基準を設定した上で、異なる種類の機械学習モデルの解釈可能性を“定性的”に評価することにする。

解釈可能性の評価基準には、Lipton（2016）の“透明性”（transparency）に関する分類、すなわち、モデルの透明性（模倣可能性：simulatability）、コンポーネントの透明性（分解可能性：decomposability）、アルゴリズムの透明性（algorithmic transparency）を採用する[13]。モデルの透明性は、ユーザーがモデル全体を一度に頭の中に入れて検討でき

[13]リスクマネジメントに特化した客観的な基準の設定は非常に難しく、今回は解釈可能性の一般的な基準を採用することとした.

るかどうかで判定する。コンポーネントの透明度は、モデルの各構成要素が言葉による説明などの直感的な説明を許容するかどうかで判定する。アルゴリズムの透明度は、ユーザーが学習アルゴリズムの動作を理解できるかどうかで判定する。これらに基づいて、機械学習モデルの解釈可能性[14]を評価するための以下の質問を設定する。

Q1. モデルの全体像は、ユーザーが完全に理解できる程度の単純さか？

Q2. モデルの各構成要素（入力、内部パラメータ、内部計算、など）は直感的に説明可能か？

Q3. アルゴリズムは、乱数などを使用しておらず確定的（非確率的）か？

6.5. 実験・評価結果

6.5.1. 予測精度の評価結果

6.4.3 節で述べた評価方法に基づいて、提案手法（NBE、NBE2、SNB、SNB2）と他の代表的な機械学習モデルの予測精度を比較した。表 6.4 は、13 データセット（6.4.2 節）に対する 20 モデル（6.4.1 節）の 5 × 5 分割交差検証による AUC の平均（上段）と標準偏差（下段）を示している。2 列目“MC2”から 14 列目“columba”は個別のデータセットを表し、15 列目“Mean”はすべてのデータセットの平均値（上段：AUC の平均の平均、下段：AUC の標準偏差の平均）、最終 16 列目“Rank”は“Mean”の上段（AUC の平均の平均）のランキングを示している。2 列目から 15 列目までの太字は、上段（AUC の平均）の上位 3 つを表す。表 6.4 からわかるように、AUC はモデル間、データセット間、および交差検証の繰り返しにおいても大きくばらついているため、全体平均で単純比較することはできず、Scott-Knott 検定の二重適用によるランク付けアプローチが必要となる。

表 6.5 は、1 回目の Scott-Knott 検定の結果を示している。表の各数値は、予測精度に対応する reversed fractional rank（RFR）である。2 列目“MC2”から 14 列目“columba”は個別のデータセットを表し、15 列目“Mean”はすべてのデータセットの RFR の平均値、最終 16 列目“Rank”は“Mean”のランキングを示している。2 列目から 15 列目まで

[14] 部分従属プロット（PDP; Friedman 2001）や LIME（Ribeiro et al. 2016）などの事後解釈性の支援ツールは使用しないことを前提とする.

表 6.4：5 × 5 分割交差検証による AUC の平均（上段）と標準偏差（下段）

（出典：Mori and Uchihira 2019）

	MC2	KC3	MW1	CM1	PC1	PC2	PC3	PC4	PC5	MC1	JM1	bugzilla	columba	Mean	Rank
OneR	0.583 (0.024)	0.554 (0.024)	0.536 (0.044)	0.534 (0.018)	0.548 (0.017)	0.499 (0.000)	0.546 (0.006)	0.626 (0.008)	0.571 (0.007)	0.508 (0.005)	0.540 (0.003)	0.576 (0.005)	0.549 (0.004)	0.552 (0.013)	20
Jrip	0.586 (0.033)	0.620 (0.054)	0.587 (0.021)	0.508 (0.018)	0.578 (0.022)	0.483 (0.005)	0.551 (0.024)	0.700 (0.014)	0.617 (0.013)	0.560 (0.013)	0.559 (0.005)	0.690 (0.007)	0.650 (0.013)	0.591 (0.019)	19
J48	0.608 (0.021)	0.569 (0.037)	0.503 (0.039)	0.567 (0.040)	0.679 (0.068)	0.544 (0.061)	0.635 (0.042)	0.745 (0.034)	0.678 (0.014)	0.587 (0.058)	0.624 (0.008)	0.748 (0.006)	0.669 (0.011)	0.627 (0.034)	18
NBTree	0.645 (0.034)	0.569 (0.079)	0.594 (0.072)	0.647 (0.050)	0.783 (0.043)	0.689 (0.050)	0.763 (0.012)	0.874 (0.016)	0.704 (0.012)	0.739 (0.032)	0.654 (0.004)	0.772 (0.002)	0.713 (0.005)	0.704 (0.031)	16
RLR	0.627 (0.048)	0.668 (0.064)	0.633 (0.027)	0.688 (0.045)	0.826 (0.019)	0.732 (0.047)	**0.820** (0.005)	0.900 (0.004)	0.741 (0.003)	0.726 (0.029)	0.673 (0.001)	0.751 (0.000)	0.724 (0.001)	0.732 (0.023)	11
SVM	0.606 (0.025)	**0.684** (0.020)	0.705 (0.008)	0.673 (0.052)	0.805 (0.021)	0.710 (0.036)	0.735 (0.032)	0.896 (0.003)	0.707 (0.001)	0.617 (0.009)	0.653 (0.001)	0.707 (0.005)	0.688 (0.004)	0.707 (0.017)	14
MLP	**0.719** (0.040)	0.612 (0.057)	0.609 (0.023)	0.614 (0.043)	0.756 (0.027)	0.683 (0.016)	0.772 (0.013)	0.881 (0.006)	0.722 (0.007)	0.702 (0.020)	0.653 (0.003)	0.729 (0.006)	0.722 (0.006)	0.705 (0.021)	15
NBc	0.707 (0.008)	0.649 (0.013)	0.695 (0.027)	0.665 (0.022)	0.750 (0.013)	0.714 (0.053)	0.737 (0.012)	0.814 (0.014)	0.694 (0.006)	0.691 (0.030)	0.633 (0.003)	0.676 (0.003)	0.661 (0.006)	0.699 (0.016)	17
NBd	0.627 (0.023)	0.607 (0.045)	0.692 (0.013)	0.689 (0.025)	0.803 (0.015)	0.792 (0.013)	0.766 (0.009)	0.811 (0.004)	0.726 (0.001)	0.729 (0.016)	0.668 (0.002)	0.731 (0.001)	0.732 (0.003)	0.721 (0.013)	13
TAN	0.615 (0.041)	0.606 (0.043)	0.687 (0.025)	0.716 (0.035)	0.837 (0.005)	**0.815** (0.018)	0.794 (0.008)	0.881 (0.002)	0.747 (0.006)	0.801 (0.013)	0.671 (0.004)	**0.774** (0.002)	0.743 (0.007)	0.745 (0.016)	9
AODE	0.643 (0.027)	0.608 (0.045)	0.703 (0.007)	**0.721** (0.014)	0.840 (0.008)	**0.818** (0.012)	0.795 (0.007)	0.875 (0.006)	0.728 (0.004)	0.810 (0.013)	0.670 (0.003)	0.765 (0.001)	0.745 (0.004)	0.748 (0.011)	8
HNB	0.618 (0.030)	0.605 (0.052)	0.678 (0.026)	**0.719** (0.014)	**0.845** (0.006)	0.810 (0.012)	0.799 (0.008)	0.874 (0.006)	0.747 (0.011)	0.795 (0.020)	**0.681** (0.002)	0.768 (0.003)	0.741 (0.005)	0.745 (0.015)	10
AdaBst	0.688 (0.028)	0.671 (0.037)	**0.709** (0.035)	**0.717** (0.031)	0.838 (0.015)	0.762 (0.016)	0.806 (0.009)	**0.917** (0.003)	0.738 (0.006)	0.832 (0.022)	**0.675** (0.003)	**0.779** (0.002)	0.747 (0.003)	0.760 (0.016)	4
RF	0.706 (0.016)	**0.722** (0.015)	**0.719** (0.025)	0.702 (0.029)	**0.868** (0.009)	0.785 (0.020)	**0.831** (0.005)	**0.936** (0.002)	**0.792** (0.006)	**0.883** (0.035)	**0.695** (0.005)	**0.819** (0.002)	**0.785** (0.002)	**0.788** (0.013)	1
NBd2	0.677 (0.012)	0.658 (0.024)	0.703 (0.005)	0.676 (0.010)	0.781 (0.007)	0.770 (0.013)	0.776 (0.004)	0.811 (0.004)	0.728 (0.001)	0.759 (0.015)	0.670 (0.001)	0.724 (0.002)	0.744 (0.001)	0.729 (0.008)	12
NBE	0.678 (0.013)	0.681 (0.020)	0.700 (0.016)	0.709 (0.010)	0.835 (0.007)	0.805 (0.015)	0.803 (0.005)	0.891 (0.003)	0.735 (0.001)	**0.839** (0.029)	0.674 (0.001)	0.768 (0.002)	**0.764** (0.001)	**0.760** (0.010)	3
SNB	0.678 (0.013)	0.673 (0.025)	**0.707** (0.014)	0.706 (0.004)	0.827 (0.008)	**0.815** (0.012)	0.801 (0.005)	0.885 (0.004)	0.735 (0.001)	0.821 (0.025)	0.674 (0.001)	0.770 (0.002)	**0.765** (0.002)	0.758 (0.009)	6
TAN2	0.691 (0.017)	0.683 (0.022)	0.703 (0.016)	0.688 (0.017)	0.822 (0.009)	0.760 (0.007)	0.798 (0.007)	0.886 (0.005)	0.751 (0.005)	0.784 (0.016)	0.672 (0.004)	0.757 (0.004)	0.753 (0.005)	0.750 (0.010)	7
NBE2	**0.714** (0.020)	**0.685** (0.020)	0.704 (0.024)	0.712 (0.018)	**0.851** (0.011)	0.777 (0.020)	0.808 (0.006)	**0.903** (0.006)	**0.754** (0.008)	**0.846** (0.016)	0.674 (0.004)	0.772 (0.004)	0.758 (0.007)	**0.766** (0.013)	2
SNB2	**0.707** (0.022)	0.679 (0.020)	0.696 (0.025)	0.716 (0.014)	0.839 (0.006)	0.778 (0.014)	**0.809** (0.008)	0.899 (0.006)	**0.752** (0.008)	0.796 (0.034)	0.674 (0.004)	0.772 (0.004)	0.756 (0.007)	0.760 (0.013)	5

表 6.5： 1 回目の Scott-Knott 検定の結果（出典：Mori and Uchihira 2019）

	MC2	KC3	MW1	CM1	PC1	PC2	PC3	PC4	PC5	MC1	JM1	bugzilla	columba	Mean	Rank
OneR	0.125	0.1	0.075	0.075	0.05	0.075	0.075	0.05	0.05	0.05	0.05	0.05	0.05	0.067	20
Jrip	0.125	0.325	0.225	0.075	0.1	0.075	0.075	0.1	0.1	0.125	0.1	0.15	0.1	0.129	19
J48	0.125	0.1	0.075	0.15	0.15	0.15	0.15	0.15	0.15	0.125	0.15	0.425	0.2	0.162	18
NBTree	0.375	0.1	0.225	0.4	0.375	0.225	0.375	0.5	0.275	0.425	0.3	0.75	0.3	0.356	17
RLR	0.375	**0.75**	0.225	0.4	**0.75**	0.35	**0.975**	0.8	0.825	0.425	0.775	0.425	0.375	0.573	11
SVM	0.125	**0.75**	**0.675**	0.4	0.375	0.35	0.225	0.8	0.275	0.2	0.3	0.2	0.25	0.379	14
MLP	**0.9**	0.325	0.225	0.2	0.225	0.225	0.375	0.5	0.425	0.275	0.3	0.325	0.375	0.360	16
NBc	**0.9**	**0.75**	**0.675**	0.4	0.225	0.35	0.225	0.25	0.2	0.275	0.2	0.1	0.15	0.362	15
NBd	0.375	0.325	**0.675**	0.4	0.375	0.6	0.375	0.25	0.425	0.425	0.5	0.325	0.45	0.423	13
TAN	0.375	0.325	**0.675**	**0.8**	**0.75**	**0.9**	0.7	0.5	0.825	0.675	0.5	0.75	0.6	0.644	8
AODE	0.375	0.325	**0.675**	**0.8**	**0.75**	**0.9**	0.7	0.5	0.425	0.675	0.5	0.75	0.6	0.613	10
HNB	0.375	0.325	**0.675**	**0.8**	**0.75**	**0.9**	0.7	0.5	0.825	0.675	0.95	0.55	0.6	0.663	7
AdaBst	0.65	**0.75**	**0.675**	**0.8**	**0.75**	0.6	0.7	0.95	0.6	0.9	0.775	0.95	0.6	0.746	5
RF	**0.9**	**0.75**	**0.675**	**0.8**	**0.75**	0.6	**0.975**	**1**	**1**	**1**	**1**	**1**	**1**	**0.881**	1
NBd2	0.65	**0.75**	**0.675**	0.4	0.375	0.6	0.375	0.25	0.425	0.425	0.5	0.25	0.6	0.483	12
NBE	0.65	**0.75**	**0.675**	**0.8**	**0.75**	**0.9**	0.7	0.8	0.6	0.9	0.775	0.75	0.925	0.767	3
SNB	0.65	**0.75**	**0.675**	**0.8**	**0.75**	**0.9**	0.7	0.5	0.6	0.675	0.775	0.75	0.925	0.727	6
TAN2	0.65	**0.75**	**0.675**	0.4	**0.75**	0.6	0.7	0.5	0.825	0.675	0.5	0.5	0.8	0.640	9
NBE2	**0.9**	**0.75**	**0.675**	**0.8**	**0.75**	0.6	0.7	0.8	0.825	0.9	0.775	0.75	0.8	0.771	2
SNB2	**0.9**	**0.75**	**0.675**	**0.8**	**0.75**	0.6	0.7	0.8	0.825	0.675	0.775	0.75	0.8	0.754	4

の太字は、最上位の RFR を表す。表 6.5 からわかるように、交差検証の繰り返しによる影響は除去されて、統計的に有意でない AUC のばらつきには同じ RFR が与えられているが、データセット間のばらつきは依然として残っている。

図 6.10 に、1 回目の Scott-Knott 検定で得られた RFR に対する 2 回目の Scott-Knott 検定の結果を示す。結果は平均 RFR にしたがって降順にソートされている。2 回目の Scott-Knott 検定は、20 個の機械学習モデルを 4 つのグループに分割した。1 番目のグループは、RF、NBE2、NBE、SNB2、AdaBst、SNB を含み、2 番目のグループは、HNB、TAN、TAN2、AODE、RLR を含み、3 番目のグループは、NBd2、NBd、SVM、NBc、MLP、NBTree を含み、4 番目のグループは、J48、JRip、OneR を含む。各グループの RFR はそれぞれ 0.875、0.6、0.325、0.1 となる。提案手法とその中間モデル、すなわち superposed naive Bayes（SNB、SNB2）とナイーブベイズ・アンサンブル（NBE、NBE2）は、ランダムフォレスト（RF）や AdaBoost（AdaBst）と共に最初のグループにランク付けされている。また、提案手法は他の主要なベイジアン学習器、すなわち、連続ナイーブベイズ（NBc）、離散ナイーブベイズ（NBd）、tree-augmented naive Bayes（TAN）、averaged one-dependence estimators（AODE）、hidden naive Bayes（HNB）などと比較して、統計的に有意な優れた予測精度を示している。

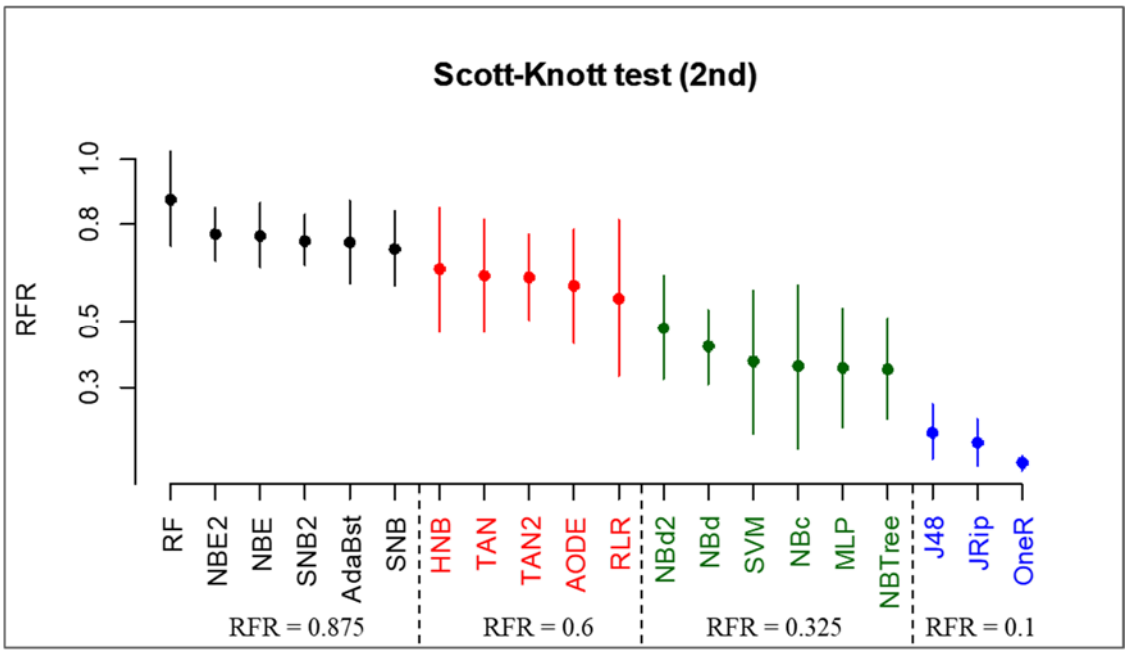


図 6.10： 2 回目の Scott-Knott 検定の結果（出典：Mori and Uchihira 2019）

6.5.2. 解釈可能性の評価結果

6.4.4 節で述べた評価方法に基づいて、提案手法（中間モデルも含む）と他の代表的な機械学習モデルの解釈可能性の定性評価を比較した。表 6.6 は、モデルの透明性、コンポーネントの透明性、アルゴリズムの透明性に関する 3 つの質問に対する各機械学習モデルの評価結果[15]をまとめたものである。機械学習モデルの解釈可能性は、質問への Yes の回答数によってランク付けし、reversed fractional rank（RFR）に変換した。以下に、各機械学習モデルに対する解釈可能性の評価結果の根拠を説明する。

- OneR や JRip などのルールベース学習器は、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいて透明性があると考えられる。
- J48 の各ノードは言葉での説明が可能であるため、コンポーネントレベルでは透明性がある。しかしながら、木の大きさはしばしばユーザーが理解可能なサイズよりもはるかに大きくなる可能性があり（例えば、Bugzilla と Columba に

[15] 定性評価は筆者が単独で実施した.

対して Weka が生成した J48 モデルは、それぞれ 241 ノードと 245 ノードをもつ)、モデルレベルでは常に透明性があるとは限らない。

- NBTree は、J48 と比べてモデル全体がコンパクトであるため、モデルレベルでは透明性がある。しかしながら、入力と出力の間の処理は直感的な理解が難しいため、コンポーネントレベルの透明性は不十分と評価した。
- RLR は線形回帰モデルの一種であり、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいて透明性があると考えられる。
- SVM と MLP は、アルゴリズムの内部パラメータの調整に乱数を使用しており、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいてブラックボックス（透明性がない）とみなした。
- NBc と NBd は、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいて透明性があると考えられる。両者の違いは、仮定した確率分布の違いのみであり、NBc は正規分布を使用し、NBd はステップ関数を使用している。
- TAN はモデル全体がコンパクトであるため、モデルレベルでは透明性がある。しかしながら、入力と出力の間の処理は直感的な理解が難しいため、コンポーネントレベルの透明性は不十分と評価した。
- AODE および HNB は、モデルおよびコンポーネントレベルではほぼブラックボックスだが、アルゴリズムレベルでは透明性がある。AODE はベイジアン集団学習モデルだが、アルゴリズム自体は乱数を使用せず確定的である。
- AdaBst と RF は、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいてブラックボックス（透明性がない）とみなした。ただし、部分従属プロット（PDP）などのツールは使用しないことを前提としている。
- NBd2 は NBd の異なる実装（Java と R）であり、評価結果は NBd と同じとなる。すなわち、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいて透明性があると評価した。
- TAN2 は TAN の異なる実装（Java と R）であり、評価結果は TAN と同じとなる。すなわち、モデルおよびアルゴリズムレベルにおいて透過性がある。
- NBE と NBE2 は、AdaBst や RF と同様に、モデル、コンポーネント、アルゴリズムの 3 つのレベルにおいてブラックボックス（透明性がない）とみなした。
- SNB は NBd2 とモデル構造上は同じであるため、モデルとコンポーネントの透

明性は NBd2 から継承される。しかしながら、SNB は乱数を使用する集団学習モデルの一種であるため、アルゴリズムの透明性は低くなる。

- SNB2 は TAN2 とモデル構造上は同じであるため、モデルとコンポーネントの透過性は TAN2 から継承される。しかしながら、SNB2 は乱数を使用する集団学習モデルの一種であるため、アルゴリズムの透明性は低くなる。

表 6.6：解釈可能性の評価結果（出典：Mori and Uchihira 2019）

	Model Transparency (Simulatability)	Component Transparency (Decomposability)	Algorithmic Transparency		
	Q1. Is the entire model simple enough to be fully understood by a user?	Q2. Is each part of the model (each input, parameter, and calculation) intuitively explainable?	Q3. Is the algorithm deterministic (non-stochastic) without using any random numbers?	# of Yes	RFR
OneR	Yes	Yes	Yes	3	0.875
JRip	Yes	Yes	Yes	3	0.875
J48		Yes	Yes	2	0.6
NBTree	Yes		Yes	2	0.6
RLR	Yes	Yes	Yes	3	0.875
SVM				0	0.18
MLP				0	0.18
NBc	Yes	Yes	Yes	3	0.875
NBd	Yes	Yes	Yes	3	0.875
TAN	Yes		Yes	2	0.6
AODE			Yes	1	0.4
HNB			Yes	1	0.4
AdaBst				0	0.175
RF				0	0.175
NBd2	Yes	Yes	Yes	3	0.875
NBE				0	0.175
SNB	Yes	Yes		2	0.6
TAN2	Yes		Yes	2	0.6
NBE2				0	0.175
SNB2	Yes			1	0.4

なお、上記アセスメントに加えて、付録 A3 では、RLR、SNB、RF+PDP の解釈可能性に関する詳細な比較分析を実施している。必要に応じて、参照して頂きたい。

6.5.3. 予測精度と解釈可能性のトレードオフ分析

機械学習モデルの予測精度と解釈可能性のトレードオフ関係を分析するために、予測精度の評価結果（6.5.1 節）と解釈可能性の評価結果（6.5.2 節）を組み合わせた。図 6.11 は、予測精度の RFR（図 6.10）と解釈可能性の RFR（表 6.6）を比較した散布図である。対角線 $y = 1 - x$ は、予測精度と解釈可能性のトレードオフ関係、すなわち予測精度が高いほど解釈可能性が低くなり、逆に解釈可能性が高いほど予測精度が低くなる関係を示している。

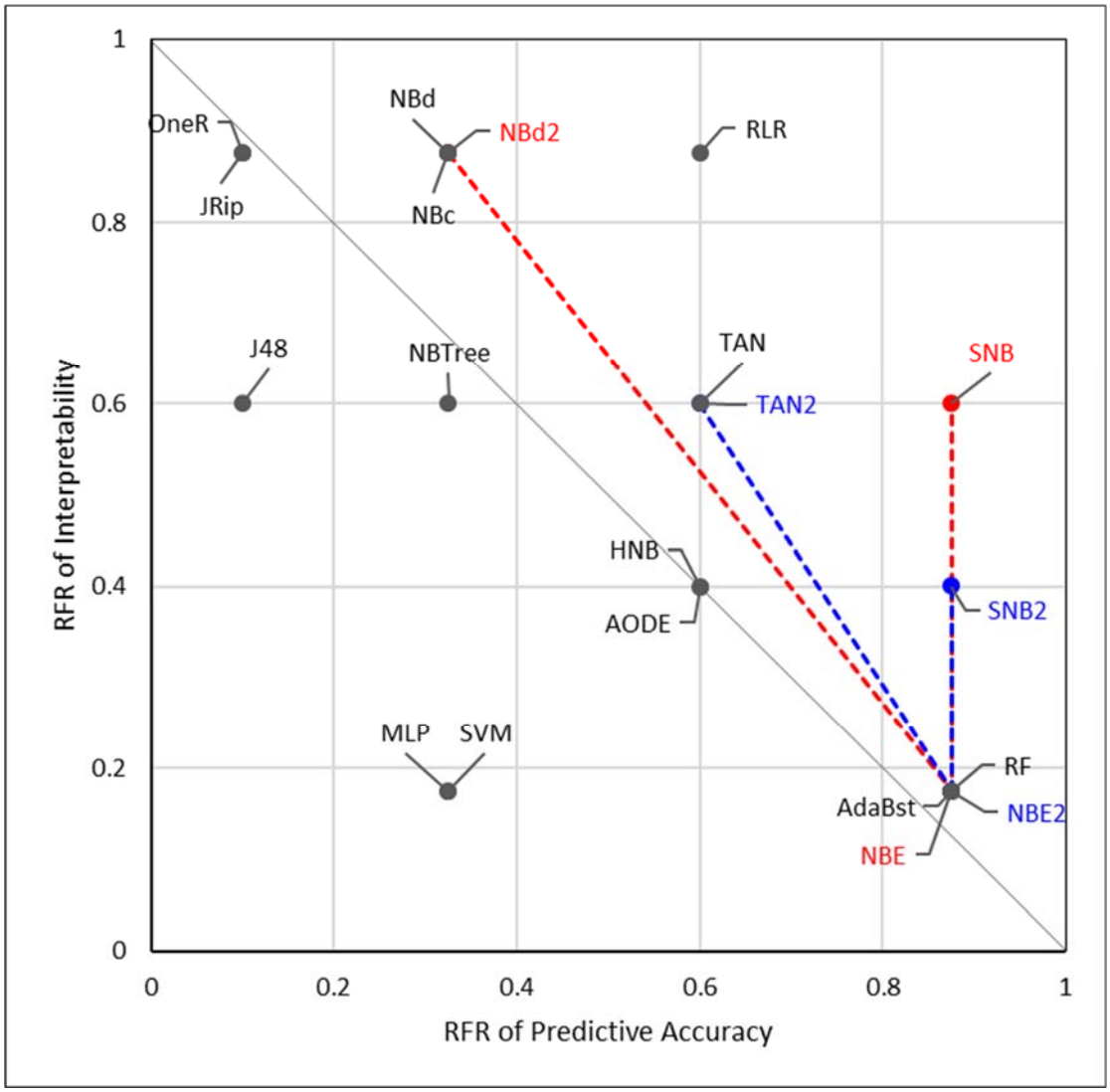


図 6.11：予測精度と解釈可能性の比較（出典：Mori and Uchihira 2019）

図 6.11 からわかるように、提案手法 superposed naive Bayes（SNB、SNB2）は右上の領域に配置されている。これは、高い予測精度と高い解釈可能性を両立していることを意味する。NBd2 から NBE を経由して SNB へ至る点線、および TAN2 から NBE2 を経由して SNB2 に至る点線は、それぞれ提案手法におけるモデル変換の流れを示す。NBd2 は、集団学習の適用（6.3.3 節）により対角線方向に沿って NBE に移動し、さらに WoE の重ね合わせ（6.3.4 節）によって SNB へと上方向に移動する。同様に、TAN2 も対角線方向に沿って NBE2 に移動し、さらに SNB2 へと上方向に移動する。本実験結果は、提案手法である SNB と SNB2 が、予測精度と解釈可能性の両面においてバランスの取れた予測結果を出力できることを示している。

6.6. まとめ

人間が機械学習モデルと協調して効果的にリスクマネジメントを実践するには、予測精度が高く、かつ、解釈可能性に優れた機械学習モデルが必要となる。本章では、“ホワイトボックス設計法”の立場から、ナイーブベイズ分類器をベースモデルとする高精度な拡張モデルの設計について論じた。

ナイーブベイズは、構造が単純なため計算効率がよく、予測結果が安定しており、ノイズに対して頑健で、モデルの解釈が容易であるという特徴があるが、その一方、予測精度は必ずしも高いとは言えない。まず第 1 ステップとして、ナイーブベイズの変数間の制約を緩めた拡張モデル、TAN（tree augmented naive Bayes)、および、新しいナイーブベイズの集団学習モデル、NBE（naive Bayes ensemble）を構築した。特に NBE は、元のナイーブベイズと比較して予測精度の大幅な向上が期待できる一方、解釈可能性は低下する。そこで、第 2 ステップとして、NBE を線形近似して単一のナイーブベイズの形式に逆変換した新しい機械学習モデル、superposed naive Bayes（SNB）を構築した。すなわち、SNB は 2 段階アプローチで生成され、第 1 のステップは主に予測精度の向上に寄与し、第 2 のステップは主に解釈可能性の向上に寄与すると考えられる。

さらに、提案手法（SNB）と主要な機械学習モデルを予測精度と解釈可能性の 2 つの観点から比較・評価した。まず、比較対象として提案手法を含む 20 個の機械学習モデルを選択し、実験用のデータセットとしてソフトウェア開発プロジェクトに関連す

る 2 つのデータソースから 13 のデータセットを準備した。予測精度の評価方法としては、Ghotra et al.（2015）と同様のアプローチを採用し、5 × 5 分割交差検証の結果に対して Scott-Knott 検定の二重適用によるランク付けを実施した。解釈可能性については、Lipton（2016）を参考に、モデル、コンポーネント、およびアルゴリズムの透明性に関する 3 つの質問を設定して、その定性的評価を実施した。実験の結果、対象モデルの予測精度と解釈可能性に関する相対的なランク付けが得られ、トレードオフ分析の結果、提案手法（SNB）が、予測精度と解釈可能性の両面においてバランスの取れた予測結果を出力できることが確認できた。

第7章 考察

本章では、本博士論文で提案した機械参加型（machine-in-the-loop）リスクマネジメントの導入・展開と、未知のリスクへの対応について考察を行う。

7.1. 機械参加型リスクマネジメントの導入と展開

機械参加型（machine-in-the-loop）リスクマネジメントは、筆者が支援した複数の製品開発部門において導入･展開に成功しており､有効に働くという実感をもっている。しかしながら､いくつかの部門において導入が難しかった事例も存在する｡本節では、両者の違いを考察することで、機械参加型リスクマネジメントの導入と展開のポイントについて整理する。

第１のポイントは、クロスファンクショナル・チームの構成である。機械参加型リスクマネジメントの導入に成功した組織においては、活動の初期段階で多様なメンバーからなるクロスファンクショナル・チームを構成し、定期的に会合を行った。導入推進チームには､（１）リーダー､（２）部門側のさまざまな業務知識をもつメンバー、（３）データ分析を支援するメンバー、などが含まれる。（１）は、部門側の関係者への説明や調整の役割が期待されており、ある程度部門内に影響力のある経験豊富な人が望ましい。また、組織の現状に問題意識をもち、組織全体のプロセス改善に強いモチベーションをもっている人が適していると考えられる。（２）は、補佐役としてリーダーの作業を分担・サポートする。必要な業務知識としては、プロジェクトの内容やプロジェクト運営の実態に関連する知識と、組織のプロセスや収集・蓄積データに関連する知識の２種類に大別でき、通常、別々の担当者が割り当てられる。また、後者については、分析のためのデータ集計などの作業をサポートする役割も期待される。（３）は、いわゆるデータサイエンティストであり、統計手法や機械学習などのデータ分析手法に精通している必要がある。先入観があるとかえってバイアスの元となる可能性もあるので、部門の業務知識はあまり必要ない。筆者は、主に（３）の立場で導入推進チームに参加した。上記のようなメンバー構成で活動を立ち上げた部門は、比較的導入がスムーズに進んだ。なお、導入推進チームは、人数が多すぎると十分に議論できなかったり議論が収束しない恐れがあるため､比較的少人数（４～６名程度）の構成が適当と思われる。

第 2 のポイントは、標準的なプロセスとデータ収集の仕組みの整備である。プロセスについては、できればステージゲート／フェーズレビュー管理を実施していることが望ましい。機械参加型リスクマネジメントの導入に成功した組織においては、程度の差はあれ、最初から組織の標準的なプロセスやデータ収集の仕組みを保有していた。仮に、たとえ形骸化しているとしても、一般的には、既存のプロセスや仕組みを改善する方がゼロベースで構築するよりもはるかに手間がかからない。機械参加型リスクマネジメントのデータ分析作業を通じて、プロセスや仕組みの非効率な部分や不適切な部分が浮き彫りになるので、改善のモチベーションにもつながる。また、ステージゲート／フェーズレビュー管理は、フェーズごとのデータ収集の節目やリスクマネジメントの施策展開の場として有効に活用できる可能性がある。ただし、上記想定はあくまでプロセスや仕組みが“改善の余地”を含んでいる場合である。プロセスや仕組みがあまりにも重厚長大で、その改善に多大な取引コストを要する場合には、第 4 章で述べたような「合理的に“失敗”する不条理」に陥る可能性もあるので、その点も考慮する必要がある。

第 3 のポイントは、部門内でのプロジェクトデータの蓄積である。筆者の所属する企業における製品開発プロジェクトは、対象製品によって異なるが、通常数か月程度から長いもので数年程度かかるため、ある程度の量のプロジェクトデータの蓄積には一般に長い年月がかかる。統計手法や機械学習などのデータ分析手法を適用するには、必要最低限のデータ数[16]をそろえる必要があるため、機械参加型リスクマネジメントの導入を効率的に進めるには、事前のプロジェクトデータの蓄積が不可欠である。ただし、提案アプローチはいわゆるビッグデータは必要とせず、小～中規模のデータであっても比較的安定した推論が可能であるため、必要最低限のプロジェクトデータから試行の検討を開始できる。過去、機械参加型リスクマネジメントの導入に成功した組織の多くは、部門固有のプロジェクト・データベースを保有しており、その中に蓄積されたプロジェクトのデータ数は 100 件程度から多いもので 10000 件程度だった。

第 4 のポイントは、組織的知識創造プロセス（SECI モデル）の反復的実行である。プロジェクト側のリスク知識更新と機械学習側のモデル化知識更新を含む 2 サイクル構造の知識創造プロセスについては、5.4 節で説明した。機械参加型リスクマネジメン

[16] 最低、30～50 件程度が必要と言われている.

トの導入に成功した組織においては、上記プロセスが複数回、反復して回っていたと考えられる。その際、各反復において部門側のメンバーとデータ分析側のメンバーの双方に気づきや学びがあることが重要であり、それが次の反復に向けた活動の動機付けにつながる。各反復で解決すべき課題が明らかになり、次の反復で課題解決に必要な新しいメンバーを巻き込んでいくことで徐々に活動の浸透・定着化が進み、最終的には組織ルーティンへの埋め込みにつながっていくと考えられる。

これらは、5.5 節で示した機械参加型リスクマネジメントの 5 つの阻害要因、A. データ収集・蓄積・前処理の負荷、B. 組織マインド醸成の難しさ、C. データの独り歩きへの懐疑、D. 失敗根本原因の定量化の困難性、E. リスクをタブー視する集団心理、に対しても有効である。すなわち、阻害要因 A、C に対しては第 2 および第 3 のポイントが、阻害要因 B、C、E に対しては第 1 および第 4 のポイントがそれぞれ効果的に作用すると考えられる。

7.2. 未知のリスクへの対応

本研究の提案アプローチ、機械参加型（machine-in-the-loop）リスクマネジメントは、新しいプロジェクトのリスクマネジメントに過去の経験がある程度活かせるような事業領域、すなわち、“既知のリスク”の割合が高く、比較的予測可能な事業領域において、もっとも効果を発揮すると考えられる。一方、まったく新規のビジネスや環境変化がきわめて激しい事業領域においては、むしろ“未知のリスク”の割合が高くなり、提案アプローチが十分に効果を発揮できない可能性がある。

佐藤・亀山（2012）は、“未知のリスク”を「対策可能範囲でしかリスクを想定していなかったケース（ケース１）」、「個々の機能の連鎖によって想定外の事象が発生したケース（ケース２）」、「現状ではリスク想定できないケース（ケース３）」の３種類に分類している。人間の知識や過去のデータが有効なのは、主にケース１とケース２であり、まったく想定外のリスク（ケース３）には対応することが難しい。

ケース 3 に対しては、Cleden（2009）が主張するような多面的なアプローチが必要となるだろう。Cleden は、不確実性のライフサイクル（図 2.2）のさまざまな局面に対応して、問題のフレーミングや根本原因分析などから構成される“問題解決戦略”、主要因子の関連図や知識マップに基づく“知識戦略”、前向き思考、後向き思考、シナ

リオ分析、複合探索などを含む“予見戦略”、プロセスの機敏性や高速学習ループを重視した“回復戦略”、マインドセットや創造性を醸成する“学習戦略”などを挙げている。また、濱口（2009）は、上位概念への抽象化と下位概念への具体化を繰り返し行うことによって“既知のリスク”の知識を新たな対象に水平展開していく創造的アプローチの重要性を強調している。これらに共通するのは、人間による気づきや創発への期待である。機械参加型リスクマネジメントにおいても、機械学習モデルの解釈・利用が人間の潜在意識や知識に新たな視点や刺激を与え、それがデータの外側の世界に対する気づきや創発を促進すれば、ケース 3 への対応につながると考えられる。

第8章 結論

8.1. リサーチクエスチョンへの回答

冒頭の 1.2 節で述べたリサーチクエスチョン RQ1、RQ2、RQ3 に対して、本研究で得られた結果を示す。

RQ1：プロジェクト・リスクマネジメントの本質的な課題は何か？

第 3 章において、プロジェクト・リスクマネジメントの課題に関して実務者のインタビュー調査を実施し、帰納的テーマティック・アナリシス法（TA: Thematic Analysis）による分析を実施した。その結果、プロジェクト・リスクマネジメントの標準的なプロセスや技法などの“理想形”と、現場におけるリスクマネジメントの“実践”との間には大きなギャップがあり、それがリスクマネジメントの形骸化など、さまざまな実践上の難しさにつながっていることがわかった。

第 4 章では、そうした実践上の難しさに対して、「限られた時間やコストやリソースの中で、不確実さを伴うさまざまな事象や状態に対して、トレードオフを含む意思決定を適切なタイミングで実施することの難しさ」を本質的な課題と仮定し、これに利害関係者の限定合理性に起因する取引コストの影響と人間の心理的特性に起因する認知バイアスの影響が作用して合理的な判断からの乖離が生じていると考察した。第 3 章の分析結果に対して、上記仮説との対応付けを行ったところ、リスクマネジメントの"難しさ"に関するコード全 41 件のうち、31 件（75.6%）は取引コストまたは認知バイアスの影響を受けていることが示され、上記仮説および理論的考察の妥当性を確認した。

RQ2：その課題に対応するには、どのような仕組みが有効か？

プロジェクト・リスクマネジメントの本質的な課題に対応するには、人間の気づきや直観的な判断（システムⅠ）と合理的な思考（システムⅡ）を効果的に連携した意思決定支援の枠組みが有効である。第 5 章では、人間と機械学習が協調してリスクに関する意思決定や知識創造を行う機械参加型（machine-in-the-loop）リスクマネジメントの枠組みを提案した。そこでは、人間と機械学習は相補的な関係にある。すなわち、

人間は、機械学習の予測結果を用いて人間がもつ先入観やバイアス（偏り）を排除し、合理的な意思決定につなげることができる。一方、機械学習は、人間からの気づきやフィードバック（現実との矛盾点や経験・感覚との違いなど）を得ることにより、データに不足している背景知識などを補完し、新たな変化や想定外の事象への対応力を高めることができる。人間と機械学習は、それぞれ現実とのギャップを埋めるために継続的な知識更新およびモデル更新が必要であり、相互に連携した 2 サイクルの知識創造プロセスが回る。

機械参加型リスクマネジメントの適用により、（１）不確実さの低減による意思決定の合理化、（２）利害関係者間の合意形成による取引コストの低減、（３）システム I とシステム II の連携による認知バイアスの影響の緩和、が期待される。提案アプローチの実適用の効果についてインタビュー調査を実施し、演繹的テーマティック・アナリシス法（演繹的 TA）を用いて上記仮説を検証した。

RQ3：その枠組みを実現する上で、最も重要な技術要素は何か？

機械参加型リスクマネジメントを実現するには、機械学習モデルが人間の意思決定プロセスに介入して、（１）新たな情報の提示による気づきの支援や（２）予測・推定結果に基づく新たな判断材料の提供、などを行う必要がある。そのためには、まず、予測精度に優れた機械学習モデルが求められる。また、特に、後者（２）においては、予測・推定結果を人間が解釈して意思決定に反映させる必要があり、機械学習モデルの解釈可能性も重要である。しかしながら、一般に、機械学習モデルの予測精度と解釈可能性はトレードオフの関係にあった。

そこで、第 6 章では、提案アプローチに適した機械学習モデルとして、ナイーブベイズ分類器を拡張した新たな機械学習モデル、SNB（superposed naive Bayes）を提案した。SNB は、ナイーブベイズの集団学習モデルを構築した後、線形近似で単一のナイーブベイズに逆変換することにより、予測精度と解釈可能性の両立を実現している。公開データセットを用いて SNB と他の主要な機械学習モデルのベンチマーク評価を実施し、提案手法（SNB）が、予測精度と解釈可能性の両面においてバランスのとれた予測結果を出力することを確認した。

8.2. 理論的含意

プロジェクト・リスクマネジメントの困難性に対する理論的考察

プロジェクト・リスクマネジメントのプロセスや手法はほぼ標準化されている（ISO 2009; PMI 2017）にも関わらず、その効果的な実践や定着化は必ずしも容易ではない（木野 2000; Bannerman 2008; Kutsch and Hall 2010）。そうした困難性への対応として、ナレッジマネジメントの適用（Massingham 2010; Alhawari et al. 2012; 内田 2016）や人工知能（AI）・機械学習の適用（Takagi et al. 2005; Lee et al. 2009; Mendes et al. 2018）などが提案されている。しかしながら、これらの研究では、リスクマネジメントの本質的な課題に対して、十分に踏み込んだ分析・考察が行われているとは言い難い。

そこで、本研究では、プロジェクト・リスクマネジメントの困難性に対して、取引コスト理論（Williamson 1981; 菊澤 2016）とプロスペクト理論（Kahneman 2011; 友野 2006）という新たな切り口から分析・考察した。その結果、不確実な状況下でのトレードオフを伴う意思決定の問題に加えて、取引コストや認知バイアスの影響が作用することにより合理的な判断からの偏りが生じて、リスクマネジメントの形骸化などのさまざまな実践上の難しさにつながっていることを示した。プロジェクト・リスクマネジメントの本質的な課題に対して、取引コスト理論およびプロスペクト理論という新たな切り口から理論的考察を行っている点に本研究の新規性がある。

machine-in-the-loop（機械参加型）リスクマネジメントの提案

本研究では、プロジェクト・リスクマネジメントの本質的な課題への対応として、機械学習と知識創造の統合アプローチである機械参加型（machine-in-the-loop）リスクマネジメントの概念を示した。それは、人間と機械学習の相補性を最大限に利用し、人間の意思決定プロセスに機械学習が積極的に介在して気づきの支援や意思決定のための判断材料の提示を行うと共に、人間側からも行動結果や予測のギャップなどのフィードバックをもらうことにより、人間の知識更新と機械学習のモデル更新を同時並行的に行っていく仕組みである。ビジネスナレッジ側とデータ分析側がそれぞれ独立したサイクルをもつという意味では Wang and Wang（2008）とも似ているが、SECI モデルの各知識変換モードにおいて両者が密接に連携している点が異なっている。

機械が人間の活動を支援する machine-in-the-loop の概念自体は既に発表されている

（Clark et al. 2018）が、その定義やアーキテクチャーについては、まだ十分に確立しているとは言い難い。プロジェクト・リスクマネジメントへの適用において本質的な課題を掘り下げた上で、machine-in-the-loop の具体的な枠組みや適用効果について踏み込んで論じている点に、本研究の理論的貢献があると考える。

予測精度と解釈可能性を両立した新しい機械学習モデルの提案

従来、さまざまな機械学習モデルが提案されてきたが、一般に、解釈可能性に優れた単純なモデル（例えば、決定木、ナイーブベイズ、ロジスティック回帰、など）は予測精度が低くなる一方、予測精度が高い複雑なモデル（例えば、SVM、ニューラルネットワーク、ランダムフォレスト、など）は解釈可能性が低下するというトレードオフ関係が存在した。しかしながら、machine-in-the-loop における人間と機械学習の協調においては、予測精度と解釈可能性をバランスよく両立することが望ましい。

そこで、本研究では、新しい機械学習モデル SNB（superposed naive Bayes）を提案、ナイーブベイズのアンサンブル（集団学習）モデルを構築して予測精度を高めた上で（ステップ１）、その線形近似により単一のナイーブベイズに逆変換する（ステップ２）という２段階アプローチの採用により、予測精度と解釈可能性の両立を実現した。ここで、第１のステップは主に予測精度の向上に寄与し、第２のステップは主に解釈可能性の向上に寄与すると考えられる。

さらに、実験によって、提案手法（SNB）と主要な機械学習モデルを予測精度と解釈可能性の２つの観点から比較・評価した。その結果、提案手法（SNB）が、予測精度と解釈可能性の両面においてバランスの取れた予測結果を出力できることが確認できた。

上記、新しい機械学習モデル（SNB）の提案、および、予測精度と解釈可能性のトレードオフ分析の方法において、本研究の新規性を主張する。

8.3. 実務的含意

本博士論文の第 3 章と第 5 章は、リスクマネジメントの課題に関する実務者のインタビュー調査の結果をまとめている。経験豊富な各インタビュイーの生の言葉は含蓄があり、ヒアリング内容そのものからも多くの実務的な知見やアイデアを学ぶことが

できる。将来、別の切り口からリスクマネジメントを分析する際においても、本調査結果は貴重な資産になるものと思われる。

第 4 章は、リスクマネジメントの実践上の難しさに対して理論的な根拠を示すものであり、リスクマネジメントの施策を検討する上でさまざまな示唆を与えてくれる。例えば、“取引コスト”を低減するためには、リスクの経験・知識の組織的共有、不確実性の客観的な評価、意思決定の柔軟性などが重要であり、さらに、“認知バイアス”の影響を逆に利用することで、ナッジ理論の応用なども導かれる。こうした示唆は、リスクマネジメント・プロセスの改善において実務的に有益である。

第 6 章では、新しい機械学習モデル SNB（superposed naive Bayes）について、全体的なコンセプトと共に詳細なアルゴリズムも提示した。それに基づいて Java と R で実装されたプログラムは、各種の機械学習モデルとの比較評価で使用され、予測精度、解釈可能性、実行速度などの面で十分に実用的であることが確認できた。提案手法は、リスクマネジメントに限らず、さまざまな領域への応用が可能であると確信する。

8.4. 本研究の限界

本研究は発展途上のテーマであり、現状いくつかの限界がある。主な限界を以下に示す。

- リスクマネジメントの本質的な課題の究明と機械参加型（machine-in-the-loop）リスクマネジメントの有効性の評価において、本研究では、実務者へのインタビュー調査を実施して、その結果をテーマティック・アナリシス法（TA: Thematic Analysis）により分析した。インタビュー対象者の選定では、いわゆる合目的的サンプリング（purposive sampling）を実施し、社会インフラ、電力、半導体などのさまざまな事業領域を対象として、なるべくバラエティを保つように努めたが、あくまで同一企業内からの選定であり、ある程度の選択バイアスは残る。他企業や他分野での調査結果について、外的妥当性を完全に保証することはできない。
- 新しい機械学習モデルの評価では、ソフトウェア工学関連の 13 のデータセットを使用して、あらかじめ選択した 14 個の代表的な機械学習モデルとのベンチマーク評価を実施した。データセットや機械学習モデルの選定において、なるべ

くバラエティを保つように努めたが、ある程度の選択バイアスは残るため、新しい機械学習モデルや他分野のデータセットを使用した場合の結果について、外的妥当性を完全に保証することはできない。

- 機械学習モデルの解釈可能性の評価において、実験などによる定量的な評価は難しく、Lipton（2016）の分類に基づく定性的な評価を実施した。複数の評価基準を用意してなるべく客観性を保つように努めたが、どうしても主観的な部分は残ってしまう。将来的には、定量的実験などに基づく解釈可能性の客観的な評価方法の確立が望まれる。

8.5. 将来研究への示唆

将来研究として、いくつかの方向性が考えられる。第 1 は、プロジェクトマネジメントおよびリスクマネジメントの領域における研究の進化である。近年の IoT（internet of things）技術および人工知能（AI）技術の飛躍的発展は、その可能性を大いに広げていると考えらえる。IoT 技術の応用によりプロジェクト活動のさまざまな情報がセンサーに取り込まれデータ化される可能性がある。今までにない新しいデータの活用によるプロジェクトマネジメントやリスクマネジメントの高度化が期待される。また、AI 技術の応用においては、過去データの蓄積が進み、予測・推定の活用はさらに広まると予想される。そして、将来的に実用レベルのプロジェクトシミュレーターが構築できれば、強化学習などの技術と組み合わせて、状況に応じて最適な計画や対応策を提案する AI プロジェクトマネジャーの実現も可能となるだろう。ただし、プロジェクトマネジメントもリスクマネジメントも基本は人間の営みであり、将来においても人間の関与はなくなることはないと考える。したがって、機械による自動化だけを一方的に進めるのではなく、ヒューマン・ファクターについても十分に考慮し、人間と機械の最適な役割分担とコンビネーションを確立することが重要となるだろう。

第 2 は、知識創造の領域における研究の進化である。機械参加型（machine-in-the-loop）プロセスのさらなる深耕が期待される。本研究では、主にリスクマネジメントへの適用にフォーカスして、machine-in-the-loop の概念を掘り下げた。しかしながら、machine-in-the-loop は登場してからまだ日が浅く、その定義やアーキテクチャーについては発展途上であり十分に確立しているとは言い難い。Machine-in-the-loop の定義、分類、具

体的なアーキテクチャー、応用、評価方法については、さらなる研究の余地があると考えられる。

第 3 は、機械学習の領域における研究の進化である。人間と機械の協調作業が増えると、機械学習モデルの解釈可能性は今後ますます重要になると考えられる。本研究においても、予測精度と解釈可能性を両立したホワイトボックスの機械学習モデルとして SNB（superposed naive Bayes）を提案し、Lipton（2016）の分類に基づく解釈可能性の定性的評価を実施した。しかしながら、解釈可能性の研究を進める上で、定性的な評価基準だけでは十分ではなく、定量的な実験など、より客観的な評価方法の確立が重要となる。もし、解釈可能性の客観的な評価方法が確立されれば、さまざまな技術的探索が可能となり、解釈可能 AI（XAI）を含む関連研究の発展を大いに促すものと想像する。

上記以外にも多くの研究課題が残っている。本研究は、発展途上のテーマであり、今後も継続的に取り組んでいく必要がある。

参考文献

Agterberg, Frederic P., Graeme F. Bonham-Carter, and Daniel Frederic Wright, 1990, "Statistical pattern integration for mineral exploration," *Computer applications in resource estimation*, Pergamon, 1-21.

Alhawari, Samer, Louay Karadsheh, Amine Nehari Talet, and Ebrahim Mansour, 2012, "Knowledge-based risk management framework for information technology project," *International Journal of Information Management*, 32.1: 50-65.

Allahyari, Hiva, and Niklas Lavesson, 2011, "User-oriented assessment of classification model understandability," *11th scandinavian conference on Artificial intelligence*, IOS Press.

Aristodemou, Leonidas, and Frank Tietze, 2018, "The state-of-the-art on Intellectual Property Analytics (IPA): A literature review on artificial intelligence, machine learning and deep learning methods for analysing intellectual property (IP) data," *World Patent Information*, 55: 37-51.

Arnott, David, and Graham Pervan, 2005, "A critical analysis of decision support systems research," *Journal of information technology*, 20.2: 67-87.

Arnott, David, and Graham Pervan, 2014, "A critical analysis of decision support systems research revisited: the rise of design science," *Journal of Information Technology*, 29: 269–293.

Baesens, Bart, Rudy Setiono, Christophe Mues, and Jan Vanthienen, 2003, "Using neural network rule extraction and decision tables for credit-risk evaluation," *Management science*, 49.3: 312-329.

Bahrammirzaee, Arash, 2010, "A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems," *Neural Computing and Applications*, 19.8: 1165-1195.

Bandler, Richard and John Grinder, 1981, *Reframing: Neuro-linguistic programming and the transformation of meaning*, Real People Press.（ 吉本武史・越川弘吉訳, 1988,『リフレーミング――心理的枠組の変換をもたらすもの』星和書店.）

Bannerman, Paul L., 2008, "Risk and risk management in software projects: A reassessment," *Journal of Systems and Software*, 81.12: 2118-2133.

Bauer, Eric, and Ron Kohavi, 1999, "An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," *Machine learning*, 36.1-2: 105-139.

Bengio, Yoshua, 2019, "From System 1 Deep Learning to System 2 Deep Learning," *Neural Information Processing Systems*, December 11th.

Bishop, Christopher M., 2006, *Pattern recognition and machine learning*, Springer.（元田浩・栗田多喜夫・樋口知之・松本裕治・村田昇監訳，2012,『パターン認識と機械学習――ベイズ理論による統計的予測』丸善出版.）

Block, Thomas R., and J. Davidson Frame, 1998, *The project office*, Crisp Publications.（仲村薫訳，2002,『プロジェクトマネジメントオフィス』生産性出版.）

Boehm, Barry W., 1991, "Software risk management: principles and practices," *IEEE software*, 8.1: 32-41.

Bohanec, Marko, Marko Robnik-Sikonja, and Mirjana Kljajic Borstnar, 2017, "Organizational learning supported by machine learning models coupled with general explanation methods: A Case of B2B sales forecasting," *Organizacija*, 50.3: 217-233.

Bonham-Carter, Graeme F., Frederic P. Agterberg, and Daniel Frederic Wright, 1988, "Integration of geological datasets for gold exploration in Nova Scotia," *Photogrammetric Engineering and Remote Sensing*, 54.11: 1585-1592.

Boyatzis, Richard E., 1998, *Transforming qualitative information: Thematic analysis and code development*, sage.

Breiman, Leo, 1996, "Bagging predictors," *Machine learning*, 24.2: 123-140.

Breiman, Leo, 2001, "Random forests," *Machine learning*, 45.1: 5-32.

Burges, Christopher J.C., 1998, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, 2.2: 121-167.

Bury, Hanna, and Dariusz Wagner, 2008, "Group judgement with ties. Distance-based methods," *New Approaches in Automation and Robotics*, IntechOpen.

Carbone, Thomas A., and Donald D. Tippett, 2004, "Project risk management using the project risk FMEA," *Engineering Management Journal*, 16.4: 28-35.

Clark, Elizabeth, Anne Spencer Ross, Chenhao Tan, Yangfeng Ji, and Noah A. Smith, 2018, "Creative writing with a machine in the loop: Case studies on slogans and stories," *23rd International Conference on Intelligent User Interfaces*, ACM, 329-340.

Cleden, David, 2009, *Managing project uncertainty*, Gower.

Cohen, William W., 1995, "Fast effective rule induction," *Proceedings of the twelfth international conference on machine learning*, 115-123.

Croskerry, Pat, 2009, "A universal model of diagnostic reasoning," *Academic medicine*, 84.8: 1022-1028.

Davenport, Thomas H., and Jeanne G. Harris, 2007, *Competing on Analytics: The New Science of Winning*, Harvard Business Press.（村井章子訳，2008,『分析力を武器とする企業』日経 BP 社.）

Davenport, Thomas H., and Laurence Prusak,1998, *Working knowledge: How organizations manage what they know*, Harvard Business Press.（梅本勝博訳，2000,『ワーキング・ナレッジ――「知」を活かす経営』生産性出版.）

De Bakker, Karel, Albert Boonstra, and Hans Wortmann, 2010, "Does risk management contribute to IT project success? A meta-analysis of empirical evidence," *International Journal of Project Management*, 28.5: 493-503.

Dixon, Nancy M, 2000, *Common knowledge: How companies thrive by sharing what they know*, Harvard Business School Press.（梅本勝博・遠藤温・末永聡訳，2003,『ナレッジ・マネジメント５つの方法』生産性出版.）

Domingos, Pedro, and Michael Pazzani, 1997, "On the optimality of the simple Bayesian classifier under zero-one loss," *Machine learning*, 29.2-3: 103-130.

Doshi-Velez, Finale, and Been Kim, 2017, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*.

Dosilovic, Filip Karlo, Mario Brcic, and Nikica Hlupic, 2018, "Explainable artificial intelligence: A survey," *41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*, IEEE.

Drucker, Peter F., 2002, *Managing in the next society*, Tokyo: Tuttle-Mori Agency Inc.（上田惇生訳，2002,『ネクスト・ソサエティ――歴史が見たことのない未来がはじまる』ダイヤモンド社.）Freitas, Alex A., 2014, "Comprehensible classification models: a position paper," *ACM SIGKDD explorations newsletter*, 15.1: 1-10.

Fawcett, Tom, 2006, "An introduction to ROC analysis," *Pattern recognition letters*, 27.8: 861-874.

Freitas, Alex A., 2014, "Comprehensible classification models: a position paper," *ACM SIGKDD explorations newsletter*, 15.1: 1-10.

Freund, Yoav, and Robert E. Schapire, 1997, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, 55.1: 119-139.

Friedman, Jerome H., 2001, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, 1189-1232.

Friedman, Jerome H., 2002, "Stochastic gradient boosting," *Computational statistics & data analysis*, 38.4: 367-378.

Friedman, Nir, Dan Geiger, and Moises Goldszmidt, 1997, "Bayesian network classifiers," *Machine learning*, 29.2-3: 131-163.

Ghotra, Baljinder, Shane McIntosh, and Ahmed E. Hassan, 2015, "Revisiting the impact of classification techniques on the performance of defect prediction models," *Proceedings of the 37th International Conference on Software Engineering*, IEEE, 789-800.

Girard, John, and JoAnn Girard, 2015, "Defining knowledge management: Toward an applied compendium," *Online Journal of Applied Knowledge Management*, 3.1: 1-20.

Glaser, Barney G. and Anselm L. Strauss, 1967, *The Discovery of Grounded Theory: Strategies for Qualitative Research*, Chicago, Aldine.（後藤隆・大出春江・水野節夫訳，1996,『データ対話型理論の発見――調査からいかに理論をうみだすか』新曜社.）

Goldstein, Alex, Adam Kapelnery, Justin Bleichz, and Emil Pitkin, 2015, "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation," *Journal of Computational and Graphical Statistics*, 24.1: 44-65.

Good, I. J., 1985, "Weight of evidence: A brief survey," *Bayesian statistics 2*, 249-270.

Grant, Kevin P., and James S. Pennypacker, 2006, "Project management maturity: An assessment of project management capabilities among and between selected industries," *IEEE Transactions on engineering management*, 53.1: 59-68.

Guidotti, Riccardo, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi, 2018, "A survey of methods for explaining black box models," *ACM computing surveys*, 51.5: 93:1-42.

Gunning, David, 2017, "Explainable artificial intelligence (XAI)," *Defense Advanced*

Research Projects Agency (DARPA), nd Web 2.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman, 2009, *The elements of statistical learning: data mining, inference, and prediction*, Springer.（杉山将・井手剛・神嶌敏弘・栗田多喜夫・前田英作監訳，2014,『統計的学習の基礎 ――データマイニング・推論・予測』共立出版.）

Heckerman, David Earl, Eric J. Horvitz, and Bharat N. Nathwani, 1992, "Toward normative expert systems: Part i the pathfinder project," *Methods of information in medicine*, 31.02: 90-105.

Herschel, Richard T., and Nory E. Jones, 2005, "Knowledge management and business intelligence: the importance of integration," *Journal of knowledge management*, 9.4: 45-55.

Holte, Robert C., 1993, "Very simple classification rules perform well on most commonly used datasets," *Machine learning*, 11.1: 63-90.

Holzinger, Andreas, 2016, "Interactive machine learning for health informatics: when do we need the human-in-the-loop?," *Brain Informatics*, 3.2: 119-131.

Holzinger, Andreas, Markus Plass, Katharina Holzinger, Gloria Cerasela Crisan, Camelia-M. Pintea, and Vasile Palade, 2017, "A glass-box interactive machine learning approach for solving NP-hard problems with the human-in-the-loop," *arXiv preprint arXiv:1708.01104*.

Huysmans, Johan, Karel Dejaeger, Christophe Mues, Jan Vanthienen, and Bart Baesens, 2011, "An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models," *Decision Support Systems*, 51.1: 141-154.

Ibbs, C. William, and Young Hoon Kwak, 2000, "Assessing project management maturity," *Project management journal*, 31.1: 32-43.

ISO21500, 2012, *Guidance on project management*, International Organization for Standardization.

ISO21503, 2017, *Project, programme and portfolio management -- Guidance on programme management*, International Organization for Standardization.

ISO21504, 2015, *Project, programme and portfolio management -- Guidance on portfolio management*, International Organization for Standardization.

Jain, Anil K., Jianchang Mao, and K. M. Mohiuddin, 1996, "Artificial neural networks: A tutorial," *Computer*, 3: 31-44.

Jelihovschi, Enio G., Jose Claudio Faria, and Ivan Bezerra Allaman, 2014, "ScottKnott: a package for performing the Scott-Knott clustering algorithm in R," *TEMA (Sao Carlos)*, 15.1: 3-17.

Jiang, Fei, Yong Jiang, Hui Zhi, Yi Dong, Hao Li, Sufeng Ma, Yilong Wang, Qiang Dong, Haipeng Shen, and Yongjun Wang, 2017, "Artificial intelligence in healthcare: past, present and future," *Stroke and vascular neurology*, 2.4: 230-243.

Jiang, Liangxiao, Harry Zhang, and Zhihua Cai, 2009, "A novel Bayes model: Hidden naive Bayes," *IEEE Transactions on knowledge and data engineering*, 21.10: 1361-1371.

Kahneman, Daniel, 2011, *Thinking, fast and slow*, New York: Farrar, Straus and Giroux.（村井章子訳，2014,『ファスト＆スロー――あなたの意思はどのように決まるか？』早川書房.）

Kamei, Yasutaka, Emad Shihab, Bram Adams, Ahmed E. Hassan, Audris Mockus, Anand Sinha, and Naoyasu Ubayashi, 2013, "A large-scale empirical study of just-in-time quality assurance," *IEEE Transactions on Software Engineering*, 39.6: 757-773.

Kamilaris, Andreas, and Francesc X. Prenafeta-Boldu, 2018, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, 147: 70-90.

Knight, Frank H., [1921] 2006, *Risk, uncertainty and profit*, Mineola, New York: Dover Publications, Inc.

Kohavi, Ron, 1996, "Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid," *Proceedings of the 2nd international conference on knowledge discovery and data mining*, 202-207.

Kononenko, Igor, 1993, "Inductive and Bayesian learning in medical diagnosis," *Applied Artificial Intelligence: an International Journal*, 7.4: 317-337.

Kotsiantis, Sotiris B., 2007, "Supervised machine learning: A review of classification techniques," *Informatica: An International Journal of Computing and Informatics*, 31.3: 249-268.

Kulesza, Todd, Margaret Burnett, Weng-Keen Wong, and Simone Stumpf, 2015, "Principles of explanatory debugging to personalize interactive machine learning," *Proceedings of the 20th international conference on intelligent user interfaces*, ACM.

Kutsch, Elmar, and Mark Hall, 2010, "Deliberate ignorance in project risk management,"

International journal of project management, 28.3: 245-255.

Le Cessie, Saskia, and Johannes C. Van Houwelingen, 1992, "Ridge estimators in logistic regression," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 41.1: 191-201.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton, 2015, "Deep learning," *nature*, 521.7553: 436-444.

Lee, Eunchang, Yongtae Park, and Jong Gye Shin, 2009, "Large engineering project risk management using a Bayesian belief network," *Expert Systems with Applications*, 36.3: 5880-5887.

Lessmann, Stefan, Bart Baesens, Christophe Mues, and Swantje Pietsch, 2008, "Benchmarking classification models for software defect prediction: A proposed framework and novel findings," *IEEE Transactions on Software Engineering*, 34.4: 485-496.

Lewis, David D., 1998, "Naive (Bayes) at forty: The independence assumption in information retrieval," *European conference on machine learning*, Springer, Berlin, Heidelberg.

Li, Bo-hu, Bao-cun Hou, Wen-tao Yu, Xiao-bing Lu, and Chun-wei Yang, 2017, "Applications of artificial intelligence in intelligent manufacturing: a review," *Frontiers of Information Technology & Electronic Engineering*, 18.1: 86-96.

Lipton, Zachary C., 2016, "The Mythos of Model Interpretability," *Proceedings of the ICML 2016 Workshop on Human Interpretability in Machine Learning (WHI 2016)*.

Madigan, David, Krzysztof Mosurski, and Russell G. Almond, 1997, "Graphical explanation in belief networks," *Journal of Computational and Graphical Statistics*, 6.2: 160-181.

Martens, David, Bart Baesens, Tony Van Gestel, and Jan Vanthienen, 2007, "Comprehensible credit scoring models using rule extraction from support vector machines," *European journal of operational research*, 183.3: 1466-1476.

Martens, David, Jan Vanthienen, Wouter Verbeke, and Bart Baesens, 2011, "Performance of classification models from a user perspective," *Decision Support Systems*, 51.4: 782-793.

Massingham, Peter, 2010, "Knowledge risk management: a framework," *Journal of knowledge management*, 14.3: 464-485.

McAfee, Andrew and Erik Brynjolfsson, 2017, *Machine, platform, crowd: Harnessing our digital future*, WW Norton & Company.（村井章子訳，2018,『プラットフォームの経

済学』日経 BP 社.）

Mendes, Emilia, Pilar Rodriguez, Vitor Freitas, Simon Baker, and Mohamed Amine Atoui, 2018, "Towards improving decision making and estimating the value of decisions in value-based software engineering: the VALUE framework," *Software Quality Journal*, 26.2: 607-656.

Menzies, Tim, Jeremy Greenwald, and Art Frank, 2007 "Data mining static code attributes to learn defect predictors," *IEEE transactions on software engineering*, 33.1: 2-13.

Menzies, Tim, Bora Caglayan, Ekrem Kocaguneli, Joe Krall, Fayola Peters, and Burak Turhan, 2016, "The promise repository of empirical software engineering data," http://openscience.us/repo. North Carolina State University, Department of Computer Science bibtex.

Merkert, Johannes, Marcus Mueller, and Marvin Hubl, 2015, "A survey of the application of machine learning in decision support systems," *Twenty-Third European Conference on Information Systems (ECIS)*, Münster, Germany.

Miller, Tim, 2019, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, 267: 1-38.

Mori, Toshiki, Shurei Tamura, and Shingo Kakui, 2013, "Incremental estimation of project failure risk with Naive Bayes classifier," *7th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, Baltimore: IEEE, 283-286.

Mori, Toshiki, and Naoshi Uchihira, 2019, "Balancing the trade-off between accuracy and interpretability in software defect prediction," *Empirical Software Engineering*, 24.2: 779-825.

Ngai, Eric WT, Li Xiu, and Dorothy CK Chau, 2009, "Application of data mining techniques in customer relationship management: A literature review and classification," *Expert systems with applications*, 36.2: 2592-2602.

Nonaka, Ikujiro, and Hirotaka Takeuchi, 1995, *The knowledge-creating company: How Japanese companies create the dynamics of innovation*, Oxford university press.（梅本勝博訳，1996,『知識創造企業』東洋経済新報社.）

Power, Daniel J., 2002, *Decision support systems: concepts and resources for managers*, Greenwood Publishing Group.

Project Management Institute, 2017, *A guide to the project management body of knowledge (PMBOK guide) 6th edition*, Project Management Institute, Inc.（プロジェクトマネジメント協会訳，2018,『プロジェクトマネジメント知識体系ガイド（PMBOK ガイド）第 6 版』プロジェクトマネジメント協会.）

Quinlan, J. Ross, 1993, *C4.5: programs for machine learning*, Morgan Kaufmann.

Ramasesh, Ranga V., and Tyson R. Browning, 2014, "A conceptual framework for tackling knowable unknown unknowns in project management," *Journal of Operations Management*, 32.4: 190-204.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin, 2016, "Why should i trust you?: Explaining the predictions of any classifier," *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, ACM.

Ridgeway, David Madigan, and Thomas Richardson, 1998, "Interpretable Boosted Naive Bayes Classification," *Proceedings of the 4th international conference on knowledge discovery and data mining*, 101-104

Rish, Irina, 2001, "An empirical study of the naive Bayes classifier," *IJCAI 2001 workshop on empirical methods in artificial intelligence*, 3.22: 41-46.

Robert, Sebastian, Sebastian Buttner, Cartsten Rocker, and Andreas Holzinger, 2016, "Reasoning under uncertainty: Towards collaborative interactive machine learning," *Machine learning for health informatics, Springer*, 357-376.

Royer, Paul S., 2001, *Project risk management: a proactive approach*, Management Concepts Inc.（峯本展夫訳，2002,『プロジェクト・リスク・マネジメント』生産性出版.）

Russell, Stuart J., and Peter Norvig, 2003, *Artificial intelligence: a modern approach*, Pearson Education.（古川康一監訳，2008,『エージェントアプローチ――人工知能 第 2 版』共立出版.）

Schwarz, Gideon, 1978, "Estimating the dimension of a model," *The annals of statistics*, 6.2: 461-464.

Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra, 2017, "Grad-cam: Visual explanations from deep networks via gradient-based localization," *Proceedings of the IEEE International Conference on Computer Vision*, 618-626.

Shaw, Michael J., Chandrasekar Subramaniam, Gek Woo Tan a, and Michael E. Welge, 2001, "Knowledge management and data mining for marketing," *Decision support systems*, 31.1: 127-137.

Shepperd, Martin, Qinbao Song, Zhongbin Sun, and Carolyn Mair, 2013, "Data quality: Some comments on the nasa software defect datasets," *IEEE Transactions on Software Engineering*, 39.9: 1208-1215.

Smith, Preston G., and Guy M. Merritt, 2002, *Proactive Risk Management: Controlling Uncertainty in Product Development*, New York: Productivity Press.（澤田美樹子・捧保浩・丹治秀明・手塚大・樋地正浩・宮林主則訳，2003,『実践・リスクマネジメント――製品開発の不確実性をコントロールする５つのステップ』生産性出版.）

Strauss, Anselm, and Juliet Corbin, 1998, *Basics of qualitative research techniques*, Thousand Oaks, CA: Sage publications.

Surden, Harry, 2014, "Machine learning and law," *Washington Law Review*, 89: 87, 87-115.

Takagi, Yasunari, Osamu Mizuno, and Tohru Kikuno, 2005, "An empirical approach to characterizing risky software projects based on logistic regression analysis," *Empirical Software Engineering*, 10.4: 495-515.

Thaler, Richard H., and Cass R. Sunstein, 2008, *Nudge: Improving decisions about health, wealth, and happiness*, Yale University Press.（遠藤真美訳，2009,『実践 行動経済学――健康、富、幸福への聡明な選択』日経 BP 社.）

Van Assche, Anneleen, and Hendrik Blockeel, 2007, "Seeing the forest through the trees: Learning a comprehensible model from an ensemble," *European Conference on Machine Learning*, Springer, 418-429.

Wang, Hai, and Shouhong Wang, 2008, "A knowledge management approach to data mining process for business intelligence," *Industrial Management & Data Systems*, 108.5: 622-634.

Webb, Geoffrey I., 2000, "Multiboosting: A technique for combining boosting and wagging," *Machine learning*, 40.2: 159-196.

Webb, Geoffrey I., Janice R. Boughton, and Zhihai Wang, 2005, "Not so naive Bayes: aggregating one-dependence estimators," *Machine learning*, 58.1: 5-24.

Williamson, Oliver E., 1981, "The economics of organization: The transaction cost approach," *American journal of sociology*, 87.3: 548-577.

Witten, Ian H., Eibe Frank, Mark A. Hall, Chris J. Pal, 2011, *Data mining: practical machine learning tools and techniques, 3rd edition*, Morgan Kaufmann

Yin, Rovert K., 2008, *Case Study Research: Design and Methods*, SAGE Publications.（近藤公彦訳，2011,『新装版 ケース・スタディの方法』千倉書房.）

Zhang, Du, and Jeffrey J.P. Tsai, 2003, "Machine learning and software engineering," *Software Quality Journal*, 11.2: 87-119.

Zhang, Harry, 2004, "The optimality of naive Bayes," *Proceedings of the 17th Florida artificial intelligence research society conference (FLAIRS 2004)*, 562-567.

内田吉宣，2016,「開発プロジェクトにおけるリスク知識の組織内知識移転マネジメント」北陸先端科学技術大学院大学 知識科学研究科 博士論文.

内平直志，2010,「研究開発プロジェクトマネジメントの知識継承」北陸先端科学技術大学院大学 知識科学研究科 博士論文.

大島丈史・内平直志，2018,「プロジェクトマネジメントへの AI 活用の知識分類モデル」『 国際 P2M 学会誌』13.1: 121-141.

菊澤研宗，2009,『組織は合理的に失敗する――日本陸軍に学ぶ不条理のメカニズム』日本経済新聞出版社.

菊澤研宗，2016,『組織の経済学入門――新制度派経済学アプローチ』有斐閣.

木野泰伸，2000,「プロジェクトにおけるリスクマネジメントシステムの構造と課題」『プロジェクトマネジメント学会誌』2.2: 33-38.

戈木クレイグヒル滋子編，2013,『質的研究方法ゼミナール――グラウンデッド・セオリー・アプローチを学ぶ 第 2 版』医学書院.

佐藤達男・亀山秀雄，2012,「P2M におけるバランス・スコアカード適用による統合リスクマネジメントの検討――高度・複雑化する IT システムのトラブル事例への対応」『 国際 P2M 学会誌』7.1: 49-59.

人工知能学会編，2017,『人工知能学大事典』共立出版.

土屋雅子，2016,『テーマティック・アナリシス法――インタビューデータ分析のためのコーディングの基礎』ナカニシヤ出版.

友野典男，2006,『行動経済学――経済は「感情」で動いている』光文社.

西原（廣瀬）文乃，2019,「組織的知識創造理論が示す AI 時代の人間の役割」『研究 技術 計画』34.1: 58-66.

西村崇・斉藤壮司・田中淳，2018,「特集 半数が 『失敗』――1700 プロジェクトを納期 コスト 満足度の 3 軸で独自調査」『日経コンピュー タ 2018.3.1 号』，日経 BP 社，27-47.

新田克己・佐藤健，2019,「人工知能の法学への応用――背景と概要」『人工知能』34.6: 870-875.

日本プロジェクトマネジメント協会，2014,『改訂 3 版 P2M プロジェクト＆プログラムマネジメント標準ガイドブック』日本能率協会マネジメントセンター.

畑村洋太郎，2000,『失敗学のすすめ』講談社.

濱口哲也，2009,『失敗学と創造学――守りから攻めの品質保証へ』日科技連.

森俊樹・覚井真吾・田村朱麗・藤巻昇，2013,「プロジェクト失敗リスク予測モデルの構築 」『プロジェクトマネジメント学会誌』15(4): 3-8.

森俊樹・覚井真吾・田村朱麗，2014,「プロジェクト失敗リスク予測モデル」『東芝レビュー』69(1): 47-50.

付録

A1：リスクマネジメントの課題に関するインタビュー調査

リスクマネジメントの課題に関するインタビュー調査の結果に対して、テーマティック・アナリシス法（TA）により切片化とコーディングを行った結果を示す。各切片番号は、インタビュー対象者（A から G）と質問項目（1 から 11）の組み合わせを表している。１人のインタビュー対象者の１つの質問項目に対して複数の切片が存在する場合には、ハイフンの後に識別番号を追加した（例えば、B2-1、B2-2、など）。なお、各表の中の記述では、RM（リスクマネジメント）、PJ（プロジェクト）、PJL（プロジェクトリーダー）、DR（デザインレビュー）、SW（ソフトウェア）などの略語を使用している。

付表 A1.1：Q1「リスクマネジメントは重要か？」への回答

切片番号	インタビュー・データ	要約	コード
A1	リスクマネジメントは重要。特に、リスクの特定が一番大事であり、リスクが特定できていれば、それに対して対策も考えられる。リスクを特定して最初に手を打つということと、リスクの予兆が現れたのをとらえてすぐ手を打つということができないと、大規模プロジェクトではうまく行かない。	大規模PJのRMでは、リスクを特定してその予兆を早期検知し、すぐに対策することが重要	リスクの特定と予兆の早期検知が重要
B1	リスクマネジメントは重要。プロジェクトを進める中で、想像できないものが起きてしまうことはあり得るが、それでも、何も考えずに進めるよりも、考え得る範囲の中でどういったリスクがあって、それが起きたらどうするの？とか、やっぱり止めた方がいいんじゃない？とか、そういったことをプロジェクトの関係者と共有しておけば、計画の見直しなどにも結び付くと思う。	事前にリスクを想定してPJ関係者と共有することにより、計画の見直しなどに結び付く	事前のリスク想定による柔軟な計画見直し
C1	リスクマネジメントは重要。ここ数年の製品開発というのは業界動向とか技術動向とかに加えて顧客要求も常に変化する。なおかつ、資源制約、すなわち、人とかモノとかお金とか含めてさまざまな資源の制約の中でQCDを達成する必要がある。そう考えたときに、やはりプロジェクトマネジメントとかリスクマネジメントというのは開発の安定化という意味では必須であると考えている。	常に変化する製品開発において、資源制約の中、QCD達成に向けて開発を安定化するためにRMは必須	RMで開発を安定化し環境変化に対応
D1	リスクマネジメントは重要。なぜなら、プロジェクトは経験の豊富なリーダーやメンバーが担当するとは限らなくて、新人だったり色々な人が携わっている。経験が豊富な人であれば、基本的に頭の中でリスク管理が出来ていてあまり必要ないのかもしれないが、そうでない人にとっては体系立った管理の仕組みがないと、かなりの危機に直面してしまうと思う。	担当者が経験不足のPJは、体系立った管理の仕組みがないと危機に直面してしまう	体系的な仕組み構築によるRMの底上げ
E1	リスクマネジメントは重要。リスクというのは、潜在化しているものもあるし、顕在化するものもある中で、受容するのか、軽減するのかなど、どうするのかをコントロールする必要がある。本来低減しなければいけなかったリスクを放置しておくと、そのまま顕在化して、実際の課題や問題になってしまう。その結果、プロジェクトの損益を悪化させてしまうということがある。	本来低減すべきリスクを放置しておくと、後に顕在化してPJの損益を悪化させる	本来低減すべきリスクが放置されている現状
F1	リスクマネジメントは重要だと思う。それがないと、プロジェクト任せになってしまって放置状態になり、管理していない状況になる。できるところはうまく行くけど、できないところはできないままというような形になるので、会社や事業部として管理できていないことになる。	PJ任せのRMでは改善しないため、組織的な取り組みが必要	RMへの組織的な取り組みの必要性
G1	リスクマネジメントは大変重要。プロジェクトを進める中で、色々なリスクが予測されるので、それらが顕在化する前に防ぐということは非常に大切。もし防げなかったら、当然ながら、後戻りが起こる、重大事故が見付かる、QCDが守れないというようなことになる。	PJには色々なリスクがあり、顕在化する前に防止しないと後戻りが発生する	さまざまなリスクへの早期対策の重要性

付表 A1.2：Q2「リスクマネジメントは難しいか？」への回答

切片番号	インタビュー・データ	要約	コード
A2	リスクマネジメントの実践は難しい。プロジェクトマネジメントの中で、一番難易度が高い。リスクの特定に関して言えば、過去の経験からきて特定できるものもあるが、新しいことをやったときには未知のものになる。未知のものを予測することは大変難しい。	未知のリスクを特定し予測することは、非常に難しい	未知のリスクを予測することの難しさ
B2-1	リスクマネジメントでは、特に、リスクの抽出の部分が難しい。見えている範囲のリスクを洗い出すというのは、時間と経験もしくは適切な人がいれば可能だと思うが、そこから漏れているものをどうやって見つけ出すかというのはきわめて難しい。その漏れたリスクが、後々大きな影響になって出て来たり、本来それを計画に加えておくべきだったというような議論にもなる。	見えている範囲のリスクの抽出は可能だが、そこから漏れているリスクの抽出はきわめて難しい	見えている範囲外のリスク抽出の難しさ
B2-2	リスクを抽出した後はした後で、自分ではコントロールできないものなど、どうにもできないものもある。いわゆる天変地異的なリスクは、極端な話いつでも起こり得るけど自分ではコントロールできない。でも、シンプルな答えでも、対応できない場合もある。例えば、単純に、スキルがある人を投入すれば解決できるという問題も、スキルがある人をすぐアサインできない場合は解決できなくなってしまう。	自分では制御できないリスクや、簡単な施策でも対応不可能なものが存在する	自分では制御できないリスクの存在
C2	リスクマネジメントの効果的な実践は、現時点では難しいと思っている。理由としては、顧客要求が変化する上に、かつ、開発進捗に与える影響要因が多い。これは、例えば、開発規模が大きくて非常に多くの開発チームが存在するとか、あるいは、色々なスキルをもったメンバーがいるとか、国内・海外の関係会社を含めて色々な人がいるとか、そういった色々な因子、要因があるので、そういったことを踏まえて計画遂行、予測、遅延対策といったことを柔軟に行う必要がある。これらの因子をパズルのように組み合わせてマネジメントして行かなければいけないので、そういった意味でも、技術的にも実際マネジメントを実行する上でも難しいと考えている。	環境変化に対応しつつ多数の影響因子を組み合わせて柔軟に施策を実施する必要がある	複雑な因子の柔軟な制御が求められる
D2	リスクマネジメントは難しい。なぜなら、現状、文化がないから。まずプロジェクトが始まると、リスクの洗い出しを行うが、洗い出して終わりっていう現状がある。日々変化するリスクをウォッチして、それに追従して改善して、というところは習慣付いてないと回らないところがある。その習慣が付けば回って行って難しくなくなるのかもしれないが、現状は文化がないので回すのが難しい。	RMの文化がなく習慣付いていない場合、継続的な改善サイクルを回すことは難しい	RMの文化がないと継続的な改善が回らない
E2	リスクマネジメントは、本来は難しくないと思う。ただし、人に依存してしまうところが難しい。人によってリスク感度が高いとか低いとか言うのは、要するに、会話の中でリスクにつながりそうな言葉を拾えるかどうか、雰囲気を感じ取れるかどうかという意味。それが出来ていれば、会話の中のリスクの芽を、その場でメモしておくのだが、出来ていないとそれを受け流して右から左に流してしまう。そうしたことを気づく人と、ふうっと流す人の違いというのがあって、マネジメント側でこの人は上手いなって思う人はやっぱり長けている。そういう人がプロマネにいると、問題があったとしても、ちゃんとコントロールされているという気がする。	会話の中からリスクの芽を感じ取るリスク感度が備わっている人にとって、RMは本来難しくないはず	リスク感度が高ければRMは難しくない
F2	リスクマネジメントは難しい。リスクの評価指標を定義したり、それを部門内で一般化して、周知することが出来ていないし、それを行うことはとても労力がかかると思っている。結果、手順がマニュアル化されないままになっていて人依存なので、やらなくてはいけないことを徹底させるのが非常に難しいというのが実感。	RMの手順がマニュアル化されないまま人依存で、組織内に周知徹底することが難しい	RMを周知徹底することの難しさ
G2	初めて何かをするのであれば、なかなか想定できないものがあるので難しいと思うが、今の自分の仕事であれば、もう大分長い間やってきて、色々な知見もたまってきているので、決して難しいことだとは思っていない。リスクを挙げること自体は難しくはないが、それを組織的に是正することについては、色々と難しい面はあるかもしれない。また、新しい技術について言えば、なかなか想定できないものがある場合は、難しいだろうと思う。	知見がたまった領域でのリスクの特定は難しくないが、それを組織的に是正することは難しい	リスクを組織的に是正することの難しさ

付表 A1.3：Q3「有識者の知見は必要か？」への回答

切片番号	インタビュー・データ	要約	コード
A3	有識者の知見は大変重要。過去の知見があるというのは、そういう失敗をしているということもあるし、そこを工夫したから上手く行ったということの両方があると思う。そういう知見をリスト化しておいて、次にプロジェクトが始まるときにそれを見ながらリスクを検討するというのはすごく有効だと思っている。未然防止リストという名前で、そういうリストを作っている。	有識者による過去の失敗経験や工夫点をリスト化して有効活用している	過去の失敗経験や工夫点の再利用
B3	有識者の知見の必要性として、有識者がいない場面では、既知の問題というのは漏れてしまう可能性がある。例えば、初めてプロジェクトリーダーをやる人が、新規というよりも何か既存の製品の改造とか改良とかといったプロジェクトをやろうとしたときには、有識者の知見がないと一からリスクを全部洗い出して行かなくてはならないので、既に過去のプロジェクトで起きているリスクや知見が入ってこないと、同じような問題を起こしてしまう可能性がある。	有識者の知見がないと一から洗い出しが必要であり、既知の問題が漏れて同じような問題を起こしてしまう可能性がある	有識者の知見活用による既知の問題の再発防止
C3	理論と実際のバランスを持った有識者の知見は必要。ただし、あくまで必要条件の一つであり、有識者の知見があれば十分という訳でもないと考えている。有識者の知見というのは、ある意味、過去の話なので、場合によっては過去の知見が判断の柔軟性を欠落させるようなリスクもある。知見は知見として尊重しつつも、参考程度に留めておくべきかと思う。	有識者の知見は判断の柔軟性を欠落させてしまう危険もあるため、参考程度に留めておくべき	過去の知見がかえって判断の柔軟性を欠落
D3	有識者の知見は必要。まだ体系立って出来てないが、プロジェクトのリスクは、やはり経験で見付けられるところはあると思うので。知識がない人だけでリスクを抽出できるかというとそうではないので、有識者の考えや知識や経験は必要だと思う。ただし、有識者の知識や経験がある程度体系立ってプロジェクト・リスクマネジメントの仕組みの中に取り込まれてくれば、要らなくなるのかも知れない。	リスクの抽出において有識者の知見は必ず必要であり、体系立ててRMの仕組みの中に取り込みたい	有識者の知見を体系化して取り込む
E3-1	有識者の知見は、節目管理でのリスクの確認などでは確かに必要だと思う。しかし、リスクマネジメントは普段からやっていないといけないものなので、節目では遅いし、有識者がたまに入ってきてもあまり効果がない。	RMは普段からやっておくべきであり、有識者の知見がたまに入ってもあまり効果がない	有識者がたまに入っても効果なし
E3-2	過去、すごい火の車のプロジェクトに投入されたことがあり、人間関係とか、会社の仕組み的なところとか、生産管理部と品証の絡みとか、プロジェクトの実態みたいなものを経験できた。こんなことは他のプロジェクトでは起こらないだろうなと思っていたが、今でも、同じことがたまに起こる。その経験によって、こういうパターンのときはこうなるということが、なんとなく分かるようになった。	過去の炎上PJで経験したさまざまな出来事が、今でも繰り返し起こっている	同じパターンの失敗が繰り返し発生
F3-1	有識者の知見は、ある面では必要。有識者と言われているからにはきっと経験が豊富で、社内からも頼られているということなので、そういう人のコメントは、やはり、ある程度適切なものが多いと思う。ただし、それが全てではない。有識者からの提言にも足りない部分は絶対にあると思っている。例えば、大分経営寄りの方に行ってしまっていて、今の現場や現場の本当の声を知らない可能性がある。また、有識者には出来る人、優秀な人が多いが、現場のレベルはまちまちで、必ずしも皆が出来るわけではない。どうすればうまくやれるかという部分で、欠けている部分が生じる可能性がある。有識者の声を100%信じるのは、危険だと思う。	有識者のコメントは適切なものも多いが、今の現場や本当の声を反映できていない可能性もあるため全面的に信じるのは危険	有識者の提言が今の現場と乖離している可能性
F3-2	有識者の知見の重要性を認識した経験として、あるシステム開発の最終試験の段階で、いくつかの準備不足で不安を感じたことがあった。プロジェクト・リーダーに話したが、あまり真剣に取り合ってもらえなかった。しかし、DR（デザイン・レビュー）の場で有識者から自分が気になっていたことを全て指摘してもらい、そこからしっかりやって行くことができた。そういう意味で、DRのような場で有識者の知見を反映することは重要であると実感した。	公式の場で有識者からリスクを指摘してもらうことで、担当者が真剣にリスク対応するという効果がある	リスク対応が強制力をもつことの効果
G3	知見は必ず必要。なぜ、必要かというと、大体同じようなことが起こるものだから。同じような問題を再発しないという意味で、予め防止することが重要。特に、自分が属する組織の場合、大体似たような種類のお客様なので、同じようなことが起きていた。しかし、最近は、新たな分野のお客様に対して、リスクマネジメントがなかなか出来なくて、異常値を出しているところもある。それは新たなお客様に対する知見が少なかったというところが問題だと思っている。	既存顧客に対して過去の知見に基づいて同じ問題の再発を予め防止することは重要だが、新規顧客では当てはまらない	既存顧客では知見は必要だが、新規は当てはまらない

付表 A1.4：Q4「有識者の知見の活用は難しいか？」への回答

切片番号	インタビュー・データ	要約	コード
A4	有識者の知見の限界として、それはは過去の経験に基づいているので、新しいものをやるときにはそこにはない話になる。リスク分類とか観点を並べて、そこに気付きがあるかを人間が見ている。そういうところに、例えば、プロジェクトデータなどの別の視点が入ってきてここが危ないとか教えてくれたりすると大変よいと思ってる。	新規開発では過去の経験に基づく有識者の知見は役に立たず、複数の視点からの新たな気付きが必要	新規開発では複数の視点からの気付きが必要
B4	有識者の知見は必要だが、あんまりそちらに引っ張られるのも問題という気がする。有識者が全知全能の人であればよいが、有識者にも偏りがある。有識者の人のすごく得意とする領域、経験した領域があって、あるプロジェクトのリスクがあったときに、そこと重なる部分はよいけど、全然重なっておらず、見えない部分というのもあると思う。あまり有識者に依存すると、全体の俯瞰という意味では見えなくなってしまう部分もあるので、一長一短な気がする。逆に、何も知らない人の方が、よい気づきが得られる場合もある。	未経験の領域では有識者にも偏りが存在し、有識者の知見に依存し過ぎると見えなくなってしまう部分もある	有識者に依存し過ぎると見えなくなる部分もある
C4	リスクマネジメントにおける有識者の知見の活用は、ファースト・ステップとしてはトライするべきだと思う。ただし、有識者の知見を形式知化することは技術的な困難度が高いと思う。すなわち、有識者が経験したことというのは、具体的にどういう条件下で何をやってどのような効果があったのかということであり、そういったものを精度良く形式知化することは、すごく大変だと思う。有識者自身も、言葉にできる部分だけでなく、潜在意識というか自分でも気が付いてない、もしくは言語化できてない条件とかあった場合に、それを精度良く拾うというの難しい。	細かい条件も含めて有識者が過去に経験したリスクを正確に形式知化することはきわめて困難	暗黙知を正確に形式知化するのは困難
D4	有識者の知見をリスクマネジメントの中に取り込んで使って行くことは、うまく有識者の経験を、例えば、チェックリストに反映するなど、表に出すことができれば、そこまで難しくはないんじゃないかと思っている。	有識者の経験をチェックリストなどで抽出できれば、RMでの活用はあまり難しくない	有識者の経験をチェックリストで抽出
E4	リスクには気付くのだけど、言いっ放しというのがリスクマネジメントの次の段階の難しさ。リスクに気付いて、チームで合意して、リスク管理票に入れて、それを管理するという流れや仕組みがない。普段からできるように躾られれば、実践力や行動力になっていくのだが。	リスクに気付いていても、管理する仕組みがないと実践力や行動力につながらない	RMにおける実践力や行動力の確立
F4-1	有識者の知見を活用することの難しさとして、受け取る側の問題もある。有識者の提言を、その通りだと思って素直に受け取って実行できる人もいれば、そんなこと言っても現場じゃうまく行かないという姿勢を初めから持っている人もいる。そこをうまく引き上げていくことが難しい。	初めから否定的な態度など、有識者の提言を受け取る側にも問題がある	有識者の提言を受け取る側の問題
F4-2	相手側が提供する情報を必要としているかどうかが重要。単なるアドバイスだと、必要ないと思ってる人には受け入れられない。その点DR（デザイン・レビュー）だと強制力が働くので、そういうやり方の方が浸透はさせやすい。ただし、有識者のコメントが的外れだった場合には、かえって、困った状況になってしまう。	DRでの指摘は強制力が働くが、コメントが的外れだと逆効果	強制力をもつ指摘が的外れだと逆効果
G4	有識者の知見を引き出すことは難しい。一生懸命伝えようという努力はしているが、なかなか伝えられないことがあるかもしれない。簡単なプロジェクト管理であれば大体分かるのだが、例えば、複雑なコア技術の注意すべき点など、複雑さが増したとき、それから分かりやすく知見を引き出すのは難しいかもしれない。だから、そこを一つ一つ分解していく手間は必要だと思う。	複雑な事象に関して有識者から分かりやすく知見を引き出すのは難しい	複雑な事象での有識者の知見抽出の難しさ

付表 A1.5：Q5「有識者の知見にバイアスはあるか？」への回答

切片番号	インタビュー・データ	要約	コード
B5	有識者の知見の思い込みや偏りの例として、例えば、新しい技術を導入する際に、古い技術のリスクを入れ込んでしまうと、必ずしもそうではない可能性がある。過去の時点では新しい技術だったかもしれないが、今の技術と乖離があったときに、そのまま過去の知見は活かせない。どう咀嚼するか、一般化するかが重要で、一般化されないまま使ってしまうと、あまり合致しない可能性がある。	過去の知見を一般化しないで使うと、現在の状況と乖離していて合致しない可能性がある	過去の知見は一般化しないと合致しない
C5	有識者の知見の中の偏見やバイアスは、故意というよりも、本人が意識しないところで、どうしても入ってしまうと思う。例えば、開発規模であるとか、開発人員数とか、どういうスキルバランスでやったのかとか、その当時の案件の技術的な難易度とか、相当色々な条件が複雑に絡み合っていて、プロジェクトの遂行中に何か問題が起きても上手く対処できたとか、こうやれば上手く行くとか、なかなかその因果関係を全て解き明かして、こういう場合にはこうすれば良いだとか、こういう風にすれば上手く行くとか、なかなか言えないと思う。そういう意味で、有識者の知見は参考程度に留めておくべきと考える。	PJの複雑な因果関係を全て解き明かすことは不可能であり、有識者の知見には偏見が無意識的に混入している	すべての知見には偏りが無意識的に混入
D5	有識者の知見の中の思い込み、バイアス、偏見は、今のところ、あまり感じることはない。大体的確であり、あまり偏りがないと思っている。	有識者の知見は大体的確であり、あまり偏りを感じない	有識者の知見は概ね的確
E5	有識者の知見の中に、思い込みやバイアスはあると思う。情報には、人から聞いた情報、自分の意見、自分が調べた情報の3種類があるが、人から聞いた情報をどうやら人間は先天的に脳で信頼するらしい。特に、その人がすごくプライオリティの高い人だったり、有識者だったり、あるいは事象の方が事件に関わるようなことだったりすると、人から聞いた情報はすごく信じるらしい。色々な情報が入ってきたときに、最初つい信じてしまうが、本当は自分で調べたら違うかもしれない。本当はそうしなくてはいけないのだが、バイアスというものがあるので、どうしても知ったものをそのまま受け入れてしまう。有識者も、人から聞いた情報をそのまま言っている可能性があるし、自分の意見であっても、今の段階ではこう思うというだけで不確定要素があって本当かどうかはまだ分からない可能性がある。実際にその事実を確認して、自分で見たときに初めて「こうすべきだ」と言うなら本来あるべき適切なアドバイス、すなわち、事実に基づいたアドバイスだと思う。事実に基づいたアドバイスまで行かないケースも結構あって、そこにバイアスを感じる。	人間は他人から聞いた情報をそのまま信じる傾向があり、その結果として偏りが混入する可能性があるため、自分の目で確認した事実に基づくアドバイスを行うべき	他人から聞いた情報をそのまま信じる傾向
F5-1	DRの指摘などで、思い込みやバイアスを感じることがある。例えば、この人なら大丈夫と思われてるような人だとあまり指摘されないけれども、ちょっと周りから心配されてる人だと初めから厳しく指摘されるようなところはある。結構当たってはいるのだが、説明する前から決め付けられている場合もある。	DR指摘での偏りとして、人に対する印象だけで決め付けていると感じることがある	DR指摘における有識者の思い込みや偏り
F5-2	指摘の中身については、当該技術の経験を積んだ人であれば、それほど変なことはない。ただし、違う部門、違う事業、違う技術を経験して、その製品の経験がない人が上に来た場合（往々にしてあること）には、一般的なこと言えるけれども、その製品に特有の技術的なところなどは、あまり深く突っ込めない。ある製品で、有識者が大勢異動してしまって、新しいレビューアーに対するその製品そのものの説明に時間を取られて、技術的な中身の本質的な議論ができなくなってしまったという話を聞いたことがある。	有識者でも経験のない分野では一般的な指摘しか行えず、本質的な議論ができない	未経験の分野では有識者の知見は当てにならない
G5	有識者の知見の中には、どうしても、思い込みやバイアスは含まれる。一つのプロジェクトについての評価をするのであれば、複数の人から聞かないと駄目だと思う。自分を正当化する人や、私的感覚が入る人が結構いるので、客観的な知見を淡々と引き出すというのは非常に難しい。	有識者でも自己の正当化や私的感覚の混入により偏った評価に陥りがちであり、客観的に評価するには複数の人から話を聞くべき	複数視点による客観的評価の必要性

付表 A1.6：Q6「異なる立場間の対立は生じるか？」への回答

切片番号	インタビュー・データ	要約	コード
B6	リスクマネジメントにおける個人間や組織間の対立はあると思う。例えば、人的なリソースに関して、など。現場レベルで、プロジェクトリーダーやメンバーが「これぐらいの人数が必要なんじゃないか」と思っていても、組織の上の方は「その半分ぐらいで大丈夫だろう」とか、その危機感の認識の違いというのは現場側と上位組織側では乖離があると思う。結局、最終的にはお金に結びついていくのだろうが、人員の不足とか、スキルを持ってる人がいないとか、開発の期間が足りないとか、そういったことでの組織側と個人（プロジェクトリーダー）との対立は一般的にあると思う。	リソース配分において管理側と現場側での危機意識の違いから対立することは一般にあり得る	管理側と現場側に危機感の温度差
C6	リスクマネジメントにおける個人間や組織間の対立として、例えば、お客さんに一番近いフロントに出ているエンジニアと開発部門との間だったり、あるいは開発部門と品質保証部門との間だったり、実務上、人間の心理的な要素というのは無視できないだろうと思う。開発部門の中にいくつかの開発チームがある場合、あるチームの進捗遅れが他のチームに影響を与えると、プロジェクトの進行上、後で影響を受けるチームの方がどうしても時間的な制約がよりきつくなって不満が生じるなど、そういう心理はどうしても働くと思う。	進捗遅れ等の責任所在に関して、チーム間の対立を生む人間の心理的要素は無視できない	RMでの対立には人間の心理的要素が影響
D6	リスクマネジメントを実施する過程で、対立というよりも、プロジェクトリスクに対する温度差は感じる。ここまでやらなくてはいけないのかと感じているグループや個人がいれば、こんなことしかやってないのかと思っている人もいる。例えば、デザインレビューの際、リスクを必ず確認しましょうというルールにしてはいるが、やはり各フェーズでのデザインレビュー本来の目的がある中でリスクも一緒に確認するという形になっているので、リスクなんて後回しだという考えの人がいたりする。そういう管理ポイントでの力の掛け具合などで、結構反発を感じることがある。	リスク対応への力の掛け具合に温度差があり、それが反発を生むことがある	リスク対応への力の掛け具合に温度差
E6	リスクマネジメントを実施する過程で、組織間や個人間の対立はあり得る。結局、リスクがあるということは、それを低減するための施策が必要になってくるので、それがどこに落ちるかによって、誰かの仕事が増える。「それはソフトでやってくれ」とか、「機械的に抑えてもらわないと無理だ」とかいうような技術者同士の衝突は起こる。それと、お金が絡む場面では、マネジメント層と実装部隊あるいは外注業者との衝突もある。マネジメント層は「お金がかかるから無理なので、そのリスクはもう受容しよう」という話をするが、実装部隊あるいは外注業者は「そのリスクは受容できない、今お金かけてでも低減しないと後で大問題になる」と思っているかも知れない。	リスク対応策がどこに落ちるかによって誰かの仕事が増えるため、費用対効果の認識ずれから組織間や個人間の対立が生じる	費用対効果と責任所在の認識ずれ
F6-1	あまり対立があるという話は聞かない。局所的な個人間の対立はあるかもしれないが、組織間の対立まではないと思う。	局所的な個人間の対立はあるが、組織間の対立はあまりない	局所的な対立はあるが組織間の対立はない
F6-2	リスクマネジメントに対する不満が出るというのはあると思う。何らかの問題が発生すると、結局、何かをチェックして、そのエビデンスを残して、ということになるが、その結果、設計部門の書類的な負荷が大きくなるような施策が増えているように感じる。何かの施策を打とうとすると、設計者が大変になるだけということが多いような気がしており、そういうことをせずに助けられないかと前々から考えている。	問題が発生すると何らかの対策が必要になり開発側の負荷が増えるため、RMへの不満が噴出する可能性あり	リスク対応責任の押し付けに対する不満
G6	個人間や組織間の対立はある。例えば、体制であれば、メンバーのスキルが３段階があったとして、全ての人が良いわけではないので、それをどう割り付けるかというような問題がある。リスクの評価は、その人の感覚によって違ったり、若干、個人の思惑が入るところもあるので、特に自分たちの利益を考えたときに、こっちの部じゃなくて、こっちの部でしょとか、難しいところがある。	リソース配分においてリスク評価に個人の感覚や思惑が入る可能性があり、部門間やPJ間の対立につながる	リソース配分での主観的判断や個人の思惑

付表 A1.7：Q7「データの活用は必要か？」への回答

切片番号	インタビュー・データ	要約	コード
A7-1	リスクマネジメントにおけるデータ活用について、今、データをうまく利用できているのは、予測モデルを使ってリスクの予兆をとらえるところ。予測モデルは結構当たるので、予測結果は客観的なデータとして扱うこともできると思っている。	予測モデルによるリスクの予兆検知は結構当たるので、客観的なデータとして利用可能	予測モデルによるリスク予兆検知の有効性
A7-2	データ活用の良い点は、客観性があり、見えるということ。リスクが見えて、皆で、やっぱり高いねとか、何が起きているんだろうとか議論が起きたりするので、見える形でリスクが表現されるのはすごくよいと思っている。ただ単に人が危ないとか言うんじゃなくて、データが危ないと示せば、それはなんでなの？とか会話になるし、それに対して、大丈夫だと思っていたけど、危ないって言われているのはなぜだろう？とか議論にもなる。	データを使うとリスクが客観的に見えるようになり、皆で議論ができる	客観的データによる議論の活性化
B7	組織側と個人（プロジェクトリーダー）の対立を埋める手段として、データでお互いに客観的に状況を見るというのが有効。定性的に大変だ大変だと言っても組織側はその大変さの度合いがよくわからないので、それを定量的に見せてあげると、差が埋まりやすくなると思う。	管理側と現場側の対立を埋める手段として、大変さの度合いを定量的に見せるとよい	対立を埋める手段としてのデータ活用
C7	リスクマネジメントにおける判断の際、データは必要。人間の個人的な感覚だけで判断するのはまずい。ただし、データだけで判断するのではなくて、やはりプロジェクトリーダーが、データによる客観的な情報もちゃんと頭に入れながら、人間系でヒアリングした状況とかも踏まえて最終判断するべき。現状では、どうしても収集データの精度には限界があると思っている。ある程度データの精度には限界がある前提で、あくまで客観的な情報の一つとして関与するというスタンスでいる方が健全だと思っている。	RMの判断にデータは必要だが精度には限界もあるので、客観的な情報の一つとして利用した上で人間が最終判断すべき	定性・定量両方の情報を使い人間が最終判断
D7	リスクマネジメントにおいて、データによる客観的な判断は必要。有識者の経験が重要なのは分かっているが、それだけだとちょっと弱い。やはり蓄積されたデータなどの裏付けがないと説得力が薄かったりする。こういうリスクには注意すべきという、例えば、チェックリストを作ったとしても、説得力を高めるためのバックデータが必要だと感じている。	有識者の経験だけでは説得力が弱いので、蓄積データによる裏付けが必要	有識者の経験を裏付けるバックデータの必要性
E7	データによる客観的な判断は必要だと思う。データ分析の結果を見ると、人間のバイアスがかかって有識者が経験から「ここのところは怪しい」と決めつけていた部分に対して、それが外れたところも見えている。有識者に頼ったリスクアセスメントは、その人たちの経験の中でやってきたことがベースになって、リスクの有り無しや、その範囲や、対策を決めたりするのだが、その経験は10年以上も前のその人たちが現役バリバリだった頃の話というケースも多い。「こうすべきと言ってもらうのは有難いが、その頃とは納期も、お金も、人も、技術も何もかもが違う」ということを、よく若手の技術者から言われる。過去の蓄積された主観的な知識だけから施策を持って来られても困るので、ちゃんと現役レベルの状態を確認して、データを使って客観的に物事を決めて欲しい。	有識者の過去の経験からの決め付けを是正する上で、データによる客観的な判断は必要	データ分析による有識者知見の偏り補正
F7	いわゆる有識者の勘と経験だけでは漏れるところがあるので、説得力を増すという意味では、データはあった方がいいとは思う。ただし、それが受け入れられるかどうかは、また少し違う話。	有識者の勘と経験の漏れを補完する意味でデータはあった方がよい	有識者の勘と経験の漏れをデータで補完
G7	リスクの重要度などを含めて、データの活用は必要だと思う。起こりやすいリスクだったら、当然、そこを優先的に処理しないといけない。でも、低コストで発生しても大した金額がかからないのであれば、ほっとけばいい、など。そういった優先度を付けるところで、データは必要となる。	リスクの重要度などの優先度付けにおいてデータは必要	リスク対応の優先度付けでデータは有用

付表 A1.8：Q8「データの活用は難しいか？」への回答

切片番号	インタビュー・データ	要約	コード
A8-1	リスクを特定するところは、なかなかデータを使うのが難しい。過去のプロジェクトでうまく行った、行かないは判定できるけれども、それがどの要因によるものかを数値化して表現するのが困難。現状、リスクの特定にはデータをうまく使えていない。	過去PJの成功・失敗の要因をデータで数値的に表現するのが困難	PJ失敗要因の完全なデータ化は困難
A8-2	プロジェクト体制とか、うまく稼働出来ているとか、チームワークがいいとか、ステークホルダーの人がちゃんとしているとか、要求仕様の決まり方が正しいとか、プロジェクトがうまく行くためには色々な要素が入っている。こういう要素がプロジェクトには多く含まれており、リスクの特定のところは、結局、人間が色々考えないと厳しいという気がする。機械的に取れるものは取って、プラス人間が考えるというのが一番よいような気がする。	PJの成功には色々な要素が影響しており、リスクの特定は機械的に取れるデータを集めた上で人間がきちんと考えるべき	機械的に収集したデータから人間主体で考える
A8-3	例えば、体制において、プロジェクトリーダーとサブリーダーが仲がよいとか、同じことをやっても仲がよいところはうまく行くけど、仲が悪い人を近くにするとうまく行かなかったりするとか、コミュニケーションのうまい人が中に入っていると全体がしっくりいくけど、いないと駄目だとかいうことがデータから分かれば、例えば、体制表を見たときにその関係を点数化できれば、それは厳しい体制だねとかいうことが分かるかもしれない。	PJ成功要因として人間関係は重要であり、体制の点数化するなどしてデータとして取り込めるとよい	人的要素の定量化の難しさ
B8	データ活用の難しさとして、出てきた数字の根拠のところに論点が行ってしまうと、結構、谷を埋めるのは難しい気がする。統計的な話をしても、統計に疎い人たちにはあまり響かない。この数字を超えると危ないとか、下回ると大変だみたいなところの合意をとるのはなかなか難しい。その中にも個人的な定性的な感覚が何となく入ってきてしまうし、全く数字や統計で判断したことのない人からすると、よく分からないと言われてしまう可能性もある。	判断基準の合意において出てきた数字の根拠に論点が行くと、統計が疎い人との感覚のずれを埋めるのは難しい	現場感覚と異なる基準を合意するのは難しい
C8	リスクマネジメントにおけるデータ活用の難しさとして、例えば、データを見て、何か進捗が遅れてるということになったときに、データ上は当然進捗が遅れてることを表しているが、なぜそうなっているかってところはなかなかデータでは表現できないと思う。対策を打つ前に、そこを人間系によるヒアリングで補完しないといけない。	データで状況が分かっても理由が分からないと対策を打てないので、人間系で補完する必要あり	データがあっても理由が不明だと対策が打てない
E8	自分たちの感覚とずれてしまったときの説明しやすさや説得力に関して、リスクマネジメントのデータ活用の難しさを感じる。例えば、「なぜ、あなたのプロジェクトは今見張られているのか？」「なぜ、リスクがあると見られているのか？」を説明しないといけない。「規模が大きいからリスクが高い」など、分かりやすい事例ならば説得しやすいが、直感と異なる場合、データだけをもって来られても説得力がないかも知れない。自分たちの感覚とずれてしまったところで、どうリスクがあると言えるのかという部分に、難しさを感じる。	データが個人の感覚とずれたときの説得力で、データ活用の難しさを感じる	データが個人の感覚とずれたときの説得力
F8	危ないということがデータの傾向として分かったとしても、そこを具体的にどうリカバーするかというところまで紐付けられないと、既に知っていることという受け取り方にしかされないような気がする。結局、そこで本当は何が問題になってるのかというところまで踏み込む必要がある。そうなると、データだけでは言い切れない。	データで状況が分かっても具体的な対策まで紐付けられないと受け入れてもらえないため、根本原因への踏み込みが必要となる	根本原因まで踏み込まないと受け入れられない
G8-1	定量的な評価は非常に難しい。例えば、リスク評価に関して、コスト換算でやるにしろ、大中小で評価するにしろ、どうしても人の感覚が入ってしまう。	定量的な評価はどうしても人の感覚が入ってしまうので難しい	定量化にはどうしても主観性が混入
G8-2	データの難しさは、プロジェクトの状況は必ずしも同じではないということ。昔データを取ったときの状況と今の状況とは、技術的にも、組織の体制的にも変わっているため、そのデータが示すことが必ずしも正しいとは言えない。そこは想定していくしかないのだが、新しい事象があったら変化するし、100%同じような状況は起こり得ないので、そういうところが難しい。しかし、そういうところが、リスクの本質でもある。	PJ状況は必ずしも同じではなく過去と現在で状況も変わっているため、データが示すことが必ずしも正しいとは限らない	PJでは完全に同じ状況は起こり得ない

付表 A1.9：Q9「知見とデータはどう補完し合えるか？」への回答

切片番号	インタビュー・データ	要約	コード
A9-1	現状、過去にあった問題に対しては有識者の知見、新しい未知のものにはリスク全体の観点リストを利用し、両方とも人間がみてここが怪しいねと言っている。それらに対して、有識者の知見とデータをうまく組み合わせて、例えば、データから今この辺が怪しいよとか言ってくれればうれしい。	既知・未知の両方のリスクに対して、現状、人間が怪しそうな箇所を判断している	リスク特定で人間の気づきに依存している現状
A9-2	観点リストは幅広くマネジメントとかエンジニアリングとかさまざまな観点を含んでいて、未知のリスクを防止しようとしているが、やはり未知だから難しい。そこに対して、プロジェクトデータを入れると、この辺のリスクが高いとか言ってくれたり、あと過去のリストに照らしてここが一致しているから危険では？とか言ってくれたり、そういう風にデータを活用できるとより客観性があってよいと思う。	未知のリスクの特定は観点リストだけでは難しく、データから新たな観点を提示できるとよい	未知のリスクに対するデータからの気づき
C9	有識者の存在が、データによるプロジェクト管理に対する補完に成り得るかということに関して、プロジェクトに関わっている有識者がデータを見れば、プロジェクトに関わっている中で感じていることとデータとを照合して、自分の感触とデータが一致しているとか、一致していなかった場合には何でそうなのかを調べたりできるので、その有識者にとってデータを活用できるだろうし、逆に、データの取り方に一部課題があるとしたら、有識者が自分の感触とデータを見て違和感を感じたときに、この辺のデータの取り方が悪くて実態を表してないとか、そういうことをデータ分析者にアドバイスすることが可能と思う。少なくとも今後の開発への知見としてコメントすることはできると思う。一方で、プロジェクトに関わってない人がデータを見たときには、多分表面的なことしか言えないんじゃないかという気がする。それなりのデータ分析の知識がないと、なかなか有益なコメントは難しいんじゃないか。そういう意味では、プロジェクトに関わっている有識者の方が、データ分析側との相互の補完関係は強いのではないかと思う。	PJに深く関わっている有識者ならば、データと自分の感触を突き合わせて新たな気付きも得られるし、データ収集の改善に向けたアドバイスも可能	自分の感覚と突合せて新たな気づきを得る
G9	コーディングや製造でリスクが顕在化したことをデータで検知できたとしても、真因は要求定義かも知れない。プロセスで淡々と集めているデータの場合、必ずしもそのデータが真因とは限らない。真因をデータ化することは非常に難しくて、有識者とか開発に関わった人たちが必要。なぜなぜ分析の深いところまでやらないといけない。リスクに対するデータを作るには、収集したデータと有識者の知見を突き合わせていく必要がある。	リスクの真因のデータ化は非常に難しく、データと有識者の知見を突き合わせていく必要がある	有識者知見と突き合わせた真因の深掘りが必要

付表 A1.10：Q10「RM の本質的な課題は何か？」への回答

切片番号	インタビュー・データ	要約	コード
B10-1	リスクマネジメントの教育は重要。一般論のリスクマネジメントというところと、組織でやっていることの周知徹底。そこがないと共通の言語でしゃべれない気がするので、教育と組織のプロセス周知はとても重要だと思う。それから、定性情報と定量情報をバランスよく使って、「これ危ないんじゃないの？」とか議論の材料にする活動を続けていくことが重要。教育をやって、プロセスを作って、それを周知徹底して、やった結果はどうだったかを振り返るという、組織的なリスクマネジメントのPDCAを、効果がすぐに出なくても粛々とやっていくことが重要だと思う。	RMの教育とプロセス周知、組織的なPDCAの実践により、RMに関して共通言語でしゃべれるようにしたい	組織的なRMで皆が共通言語でしゃべれるように
B10-2	リスクマネジメントはしなきゃいけないという点に関しては、多分、ノーという人はいないと思う。でも、できないところとできているところがあるので、基本的なリスクマネジメントの仕方について、概念的なものと現場で使えるもののセットになったものがあると、実際にプロジェクトを運営している人はやりやすいだろうと思う。	基本的なRMの方法について、現場で使える実践的なガイドを提供できるとよい	RMの基本を学べる現場向け実践ガイドの提供
C10-1	プロジェクト・リスクマネジメントの本質的な課題は、色々な事象があり因果関係が複雑な中で、何か対策を打ったときの費用と効果がどの位の大きさなのかを見極め、判断や対策を迅速に実施することが求められることだと思う。因果関係の解きほぐしと、費用対効果がリーズナブルかどうかという評価は、非常にパラメータも多いし、単純な式では表せない複雑な要素があるので、それをどうマネジメントするか、コントロールして行くかという部分が本質的な課題だと思う。	複雑な因果関係の中で対策の費用対効果を迅速に見極めて、因果関係の解きほぐしと費用対効果の妥当性評価を行う必要がある	因果関係を紐解いて施策効果を迅速に見極め
C10-2	どうしても人間とか組織の心理的な要素が働くので、実際のプロジェクト管理の枠と考えると、そこをどう取り扱うかというのが、別の切り口での本質的な課題という気がする。	人間とか組織の心理的な要素をどう取り扱うかが課題	人間や組織の心理的要素をどう取り扱うか
D10	リスクマネジメントの一番難しいところは、文化を作るところだと思う。皆、リスクマネジメントが必要だというのは頭では認識していると思うが、それを習慣づけてできるようにならないと、多分回らない、うまく行かない。そこがリスクマネジメントの本質、すなわち、一番必要なことだと思う。	RMの必要性を頭では理解していても、習慣づけてできる文化をつくらないと回らない	RMが習慣付いて自然に回るような文化へ
E10	リスクマネジメントの肝は、普段の会話の中でプロジェクトのメンバーの人たちの声をちゃんと聞いてあげて、それに対してリスクがあるかどうかの感度を常に高めておいて、それを一緒にちゃんと拾ってあげて、管理してコントロールしてあげること。そういうことをちゃんとやってあげると、課題も出てくるし、リスクも出てくるし、プロジェクトは回りだす。逆に、皆がそれぞれ課題を抱えていて、誰も何も言わないというプロジェクトもある。	リスク感度を常に高めて一緒に拾って管理してコントロールしてあげると、PJは回り出す	リスク感度を高めて一緒に拾うとPJは回り出す
F10	一つは、人間系で回ってるところが多いということ。もう一つは、リスクの見える化が現状出来ていないし、結構難しいのかなと思う。特にソフト系だと外からは本当に何も見えない。建築系で作った建物が倒れたなんてことは、まず日本じゃありえないけど、システム開発では頻繁にあったりする。ここの根本的な違いっていうのは何かな？っていうのが、その本質的な難しいところにつながるのかなと思う。見えづらい、ブラックボックス、でも中で色々動いている、というところが難しいのだと思う。	SW開発は人間系で回っている部分が多いが、ブラックボックスでリスクの見える化ができていない	人間系で回っている部分がブラックボックス
G10-1	リスク対応の優先度に関しての全一致が難しい。人の体験によっても、感覚によっても違う。例えば、プロジェクト・リーダーが率先して進める場合は、プロジェクト・リーダーの感覚になる。色々なリスクがあって、一回でも経験した人はその重みが分かるが、経験しないと分からなかったりする。	人の経験や感覚による違いがあり、リスク対応の優先度に関する全一致が難しい	リスク優先度への組織内全一致の難しさ
G10-2	プロジェクト・リーダーの力量でリスクは変わってくると思う。一番重要なのはプロジェクト・リーダーの力量だと思っている。力量が劣る、スキルが低いプロジェクト・リーダーのときは、それをカバーするものが必要。プロジェクト開始前の体制を作るところをどうするかなど、どこからリスクマネジメントをするか、誰がリスクマネジメントをするかを考えないといけない。標準品でないソフトウェア開発だと、特に重要だと思う。ソフトウェア開発は、残念ながら、人の力量に依存するので。	PJLの力量が一番重要なので、PJLが力量不足のときそれをカバーするものを考える必要がある	PJリーダーの力量不足を補う施策が必要

付表 A1.11：Q11「RM の有効性を高めるには何が必要か？」への回答（その 1）

切片番号	インタビュー・データ	要約	コード
A11-1	リスクマネジメントにおける予測モデルは、リスクが高いか低いかを示すだけで、プロジェクトリーダーに対して具体的な対策は示せない。また、対策とその影響がモデルの中に反映されていない場合、対策を打ってもリスクは下がらないので、対策が効いているかどうか分からないという課題がある。	予測モデルがリスク対策とその影響を含んでいないと対策を打ってもリスクは下がらない	リスク対策とその効果を予測モデルに反映
A11-2	現状の予測モデルの構築においては、組織データベースに入っているデータだけを使っているが、本当はもう少し色々な種類のデータがあった方がよいと思っている。要件の決まり具合など、定性データとして取れるものは一応集めているが、プロジェクトの状況を表現するには、他にももっと色々なパラメータがあるはず。	組織DBに入っているデータだけでなく、PJの状況を表す色々なデータを収集すべき	PJ状況を表す多様なデータの収集
A11-3	データの精度も課題。データの精度が良くないと、結局、そのリスクが高いか低いかの判断を誤ってしまう。プロジェクト状況が正しくて、かつ、データがちゃんと出てきているということをセットで見た方がよい。プロジェクトが混乱していると、そもそも出てくるデータがうそかも知れない。ちゃんと進捗管理がされているとか、プロジェクトが正しく運営されているということと、データの精度が正しいということがペアで表現できないと、判断を間違えてしまう。	リスク判断を誤らないために、PJ運営の正しさと収集データの精度をセットで考えるべき	データ精度の確保は正しいPJ運営が前提
B11-1	客観的にリスクを分析するスキルをもった人がいる方がよい。リスクマネジメントという意味では、定性的・定量的の両面で第三者的に見て、「これ大丈夫？」とか言える人が必要。いわゆるPMA（Project Management Assistant）と言われる人なのかもしれない。	客観的にリスクを分析するスキルをもった人が第三者的にPJを見ることが重要	分析の専門家による第三者的リスク監査
B11-2	リスク分析のやり方の標準セットのようなスキルを育成する手段があるとよい。PMBOKには、リスク分析のやり方とか、Howの部分はあまり書いていないから、リスクをどう分析して、解釈して、自分たちのマネジメントに使っていくかという教科書的なものがあるよい。リスクをマネジメントする側の人たちにそれを勉強してもらって、組織の中で共通言語で語れるとよい。	リスク分析のやり方の標準セットなど、組織の中でリスク分析のスキルを育成する手段が必要	組織内でのリスク分析の基本スキルの育成
C11-1	プロジェクト・リスクマネジメントの高度化に向けて、狭義の意味だと、まず、開発環境とかソースコード解析とか構成管理とか色々なエンジニアリング・ツール群で自動化して、データの精度をできるだけ上げることが必要。その上で、プロジェクト管理手法やその手法を実現したツールによる予測やシミュレーションの高度化は必要。現在の開発状況がどうなっているかを理解するのと同時に、今後の対策を判断する上で色々実験できるような仕組みがあると、プロジェクトリーダーの助けになるという気がする。	ツールによる自動化でデータ精度を向上した上で、予測やシミュレーションの高度化によるPJ状況把握や判断支援ができるとよい	予測の高度化による意思決定支援への期待
C11-2	プロジェクト・リスクマネジメントの高度化に向けて、広い意味だと、他の関連する技術領域に広げて、人間が関わる要素の部分とかも含めて考えていかないといけないと思う。データによる分析とか判断だけでは限界があるような気がする。というのは、分析できたとしても、もう手が打てないということもあるかも知れないので。例えば、そもそも要求うまくコントロールしないと、どんなに優秀な人がやってもこの納期には間に合わないという結果が出てお客さんと折衝する話になったり、必要なスキルを持った人をちゃんと用意しないと終わらないようなプロジェクトだったら、ある意味チームビルディングみたいな話も関係する。さらに、プロジェクトリーダーを誰にするかとか、リーダーとサブリーダーとの関係はどうなのかとか、どういう風にチーム編成すればいいのかとか、それこそ相性も含めて色々考えていかないと、本当にマネジメントの高度化というところにはたどり着けないという感触がある。	データによる分析・判断だけでは限界があり、人間が関わる要素の部分と統合したマネジメントが必要	人間的要素も含む統合マネジメントへ
C11-3	組織の視点で考えると、人材育成的な側面もマネジメント上は重要。例えば、極端な話、全員優秀なメンバーを集めてやればプロジェクトはうまく行くかもしれないが、やはり今後のその組織のことを考えて、ちょっと荷が重いかもしれないけど、見込みがある人に、経験を積んでもらうためにプロジェクトリーダーやチームリーダーをやってもらうとか。当然、組織の中ではそういうことも考えてマネジメントをしていく。そういった要素もリスクマネジメントの高度化という範囲の中で、もしかしたら考えて行かなければいけないことなのかも知れない。	組織の視点で考えると、将来を見据えた人材育成的な側面も重要	組織の視点では人材育成的側面も重要

付表 A1.12：Q11「RM の有効性を高めるには何が必要か？」への回答（その２）

切片番号	インタビュー・データ	要約	コード
D11-1	今まであったことならば、経験や過去のデータの分析などで未然に抽出して防げると思うが、今まで経験したことのない事象やリスクは、なかなか過去のものから生み出しにくいと思う。環境や時代の変化などの情報から、未来に何がリスクとして出てくるのかが分かるというのが、リスクマネジメントの高度化なのだと思う。	今まで経験したことのないリスクは過去のものから生み出しにくいので、環境や時代の変化の考慮した未来のリスクの洞察が必要	環境の変化の考慮して未知のリスクを洞察
D11-2	リスクの予測に関して、過去の経験と同じであれば、過去トラなどをチェックしていれば何とか防げると思うが、それだけでは未来を十分に予測できない。過去の経験からでは分かり得ないようなところのリスクが大事であり、そこまで踏み込まないといけないと思う。	未来を予測するには、過去の経験だけからでは分かり得ないリスクが重要であり、そこまで踏み込むべき	過去に経験のないリスクにも踏み込むべき
E11	リスクがあって、それに気付いていても、自分たちの課題として見ていないことが問題。他の人の課題だから別に関係ないと思ってしまうと、結局、回り回って、自分たちの工程にしわ寄せが来たりすることもある。弱いチームは、他人事になっている。逆に、良いチームは、マネージャーやPMA（Project Management Assistant）がそこを全部引き取って、自分で動いて調整して、全体を同じ方向に向かせようとする。最終的には人に依存するが、そういう経験なり躾ができているかどうかが、すごく大きいと思う。	リスクに気付いていても、自分たちの課題として見ないことが問題であり、良いチームではマネージャーが全部引き取って自分で動いて調整する	良いチームはリスクに気付いたら自律的に動く
F11-1	相反しているが、ある程度の強制力は必要な一方、なるべく楽にして負荷をあまりかけないという、その両方のバランスが重要。優しくしていると絶対にやらないが、無理に強制して何か別の本質的な作業が滞ってしまうのも避けたい。リスクマネジメントの施策が自分たちの役に立っていると思ってもらい、リスクを正直に申告できる雰囲気にしたい。そういう姿勢がうまく伝わると、皆協力してくれると思うが、トップダウンで施策を展開すると、なかなかそうはならないところが難しい。	優しくしていると絶対にやらないが、無理に強制して本質的な作業が滞るのも避けたいので、強制力と負荷軽減のバランスが重要	強制力と負荷軽減のバランスが重要
F11-2	昔聞いた話として、ある部門で新しいシステムを導入しようとした際、「こんなの大変だ」と課長級から強い反対があった。そのとき、部門長は、「まず一ヶ月使ってほしい。それで駄目なら仕方ないけど、まずは一か月、文句を言わずに使ってほしい。」と説得した。そうしたら一か月後、皆文句言わずに使うようになったとのこと。最初の取っ掛かりは大変なのだが、使ってみて、使い方が分かって、使いこなすと楽だということが分かって、文句言わずに使うようになる。そこの最初のハードルを強制的に飛ばさせたというところがポイント。一種の化学反応。	新しいシステムの導入に対して、メンバーから強い抵抗があったとき、最初のハードルを強制的に飛ばさせることでうまく行った	最初のハードルを強制的に飛ばさせる
G11	組織のリソースが決まっている中で、どのリスクから手を付けるのかを何らかの方法で順序付けできるとよい。組織として優先度付けの部分で失敗する場合が多いように思える。限られたリソースの中でどう優先度付けするかというのが、プロジェクト・リーダーの力量でもある。今は人間系でやっているが、データを集めて、このパターンだとこのリスクが重要というのが自動的にでるとよい。	限られたリソースの中でどのようにリスク対応の優先度付けをするかが、データから自動的に出るとよい	リソース制約を考慮したリスク優先度付け

A2：提案アプローチの効果に関するインタビュー調査

提案アプローチの適用効果に関するインタビュー調査の結果に対して、演繹的テーマティック・アナリシス法により切片化とコーディングを行った結果を示す。各切片番号は、インタビュー対象者（H と I）と質問項目（1 から 4; 附表 A2.1）の組み合わせを表している。1 人のインタビュー対象者の 1 つの質問項目に対して複数の切片が存在する場合には、ハイフンの後に識別番号を追加した（例えば、H1-1、H1-2、など）。

付表 A2.1：インタビューの質問項目（再掲）

Q1	before/afterで、プロジェクト失敗の要因がより明確になったか？
Q2	before/afterで、プロジェクトリスクについて第三者への説明が容易になったか？
Q3	before/afterで、プロジェクトリスクに関する自身の想定や思い込みが変化したか？
Q4	before/afterで、その他、何か気づいたことはあるか？

初期コードとして、（1）不確実さやあいまいさの低減、（2）利害関係者間の合意形成、（3）データによる認知バイアスの補正、を設定した。すなわち、各切片データが、「わからなかったことが、わかるようになる」ことを示唆する場合は（1）、「わかっていたつもりだったことが、正しくわかるようになる」ことを示唆する場合は（3）、「わかっていても変えられなかったことが、変えられるようになる」ことを示唆する場合は（2）、上記以外は“その他”に分類した。また、切片データが、主に、after での具体的な効果や期待を語っている場合は“肯定的”（P）、before での課題や阻害要因を語っている場合は“否定的”（N）とした。

コーディング作業は筆者を含めて計 2 名で実施した。切片データが初期コードの内容を直接的には語っていない場合においても、語りの背景や意図を推察して可能な限りもっとも関連性の高いコードを割り当てる方針とした。付表 A2.2、および、附表 A2.3 に、切片データとそのコーディング結果を示す。切片番号に※印がついているものは、最初のコーディングにおいて作業者間の不一致があったが、その後の議論で収束した切片データを示している。最終的に、各切片には 3 つの初期コードのいずれかが割り当てられ、“その他”に分類される切片データは存在しなかった。なお、表中の記述では、DR（デザインレビュー）などの略語を使用している。

付表 A2.2：Q1 から Q4 への回答（その 1）

切片番号	インタビュー・データ	コード
H1-1	before/afterでプロジェクト失敗の要因は、どちらかと言えば、明確になったと思う。今までも感覚的にはすごく感じていたものが、データに基づいて確かめられるという面と、予想外に違うものが入っているような気がするという面がある。例えば、設計変更などはそこまでプロジェクト失敗に影響があるとは思っていなかった。	データによる認知バイアスの補正 (P)
H1-2	自部門は本当にモノしかなかったんで、感覚がある人とない人の差が大きい。感覚がある人にとってはなるほどとなるし、感覚がない人にとっては知らないことが明確になるということだと思う。	不確実さやあいまいさの低減 (P)
H2-1	プロジェクトリスクについて第三者への説明は容易になったと思う。まだ、残念ながら説明変数が足りていないが、もっと色々なものが測れるようになったら、プロジェクト側に、例えば、品質保証について説明する際、こちら側が感じた感覚だけで説得するよりもデータに基づいて説明した方が、聞く側もやはりそういう風にデータに出ているのかと思ってもらえるし、説明する側もデータを使うことで客観的な話ができると思う。そういう意味では、やはりデータがあった方が絶対に説得しやすいと思っている。	利害関係者間の合意形成 (P)
H2-2	今までは施策を打ち出しても、それはデータに基づいているのか？とプロジェクト側から言われることがたくさんあった。そう言いたくなる気持ちは理解できる気がする。それがデータを使って説明できるようになったら、すごく良いと個人的には思っている。	利害関係者間の合意形成 (P)
H3	自身の想定や思い込みが変化したというよりも、裏付けられたという方が正しいような気がする。もしかしたら、これからもっと説明変数が増えてくると、意外な影響が見えてくるかも知れない。そういう意味では面白いなと思う。現状、まだ、プロジェクト失敗に影響しそうだと思っているところのデータを取り切れていない。データの蓄積がなかったり、データが関連づけられていなかったりする。その辺が取れるようになったら、想定や思い込みを変えるようなものが出てくるのかも知れない。やっぱりそうかと感じる部分の方が、今は強い。	データによる認知バイアスの補正 (N)
H4-1 ※	これから、おそらく人が減っていって、リスクの匂いを感じられなくなったり、感覚がない人が増えてくる。例えば、今は、技術や設計をやってからスタッフに来てる人が結構多いが、最初からスタッフとして入社している人が色々な部署で増えてきている。今後、経験がないと多分感じられないところを感じなくなっていく人も増えていくと思う。そうなったときに、データでリスクを感づけるようになれれば、そこが補えると期待している。	不確実さやあいまいさの低減 (P)
H4-2	自分は、データをリスク予測だけでなくもっと色々な場面で分析したり活用したりできたらすごく良いと思っている。ただし、統計にしろ、機械学習にしろ、何にしても前処理があるので、そこをもう少しうまくできるようにする必要がある。そうなれば、もっと色々な場面で活用できると思う。人手不足のところや、人がいても経験がないところで、データが物語ってくれたりするようになると良いと思う。	不確実さやあいまいさの低減 (N)
H4-3 ※	データに基づいた話ができるというところは、すごく有効だと思う。また、人が不足したり、リスクの匂いを感じられない人を補えるというところが、afterとしてはすごく良いと思う。ただし、前処理や、データが取り切れないということになると、やはりスタートしようとした時点でデータがないと始められないので、そこを如何に普段からできるようにするかということをやらないと、逆に作業が増えてしまう可能性もあるような気がする。	不確実さやあいまいさの低減 (N)
H4-4 ※	beforeは何もやっていなかった。確かに、やらなかったら余計な作業も発生しないけど、結局何もやらない。本活動をすることで、データを取らなきゃ、データを貯めなきゃというきっかけになったような気がする。そして、こうした活動が組織の中でもっと受け入れられるようになれば、afterはもっと良いものになっていくと思う。ただし、今はまだ途中段階なので、一部の人たちだけの活動だと思われている。これがもう少しまわりの皆にわかってもらえるようになると、きっとデータの入力ももっと正確になっていくだろうし、そうなれば、データ分析がすごくやりやすくなるようなプラスの循環になっていくと思う。	利害関係者間の合意形成 (N)
H4-5 ※	リスク予測を組織内に展開していくための一番大きな壁は、やはりマインドなのかも知れない。何をやるにしても、結局、そこに行き着く気がする。その技術が必要だって思ってもらえたり、それがあると良いねという認識を皆にもってもらえるようになると、前向きにとらえてもらえる。今のままだと、一部の人たちだけが何かやってるねで終わってしまうかも知れない。そこの境目を乗り越えないといけないと思っている。	利害関係者間の合意形成 (N)
H4-6 ※	マインドというのは、思い込みだったりするのかなと思う。結局、今、皆がその場の作業に追われちゃっているので、例えば、不具合だったらその場でクローズできればよいという考えになる。それだとデータ入力もあんまりよいものになっていかない。そのあたりが変わって、今だけでなくこの先にも使えると思えるようになれば、データ入力なども少し変わってくるのかなと思う。	利害関係者間の合意形成 (N)
H4-7 ※	よいサイクルで回るようになると、機械学習に限らず、すべてよい方向に行くと思う。そういう意味では、自分にとってのbefore/afterとして、よい気付きになった活動だと思っている。自分のように、技術にいてその場その場で設計して目の前のモノをつくったり納期に追われるということをやってきた人間からすると、こういう世界はあまりなかった。トラブっているプロジェクト、修羅場のプロジェクトばかりやってきたので、こういうプロジェクトは怪しいね、こうなってきたら怪しいねが何となく感覚で感じられるようになった。それをやってきてない人はそうじゃないと思うので、そうじゃなくて育ってきている人に対して、データで伝えられると良いと思う。	不確実さやあいまいさの低減 (P)

付表 A2.3：Q1 から Q4 への回答（その 2）

切片番号	インタビュー・データ	コード
I1-1	before/afterで、プロジェクト失敗の要因がより根拠のあるわかりやすさになったと思う。以前は、経験則だった。規模が大きいとか、初号機の後のやつはヤバいよねとか、漠然とした経験値でヤバそうな匂いがするプロジェクトがうわさになっている感じだった。それが今は、定量的にやはりそうかっていう根拠みたいなものに変わってきている。	不確実さやあいまいさの低減 (P)
I1-2	それに加えて、開発の途中でわかるようになった。以前は、規模が大きくて怪しいなど、最初だけしかわからなかった。そして、わからないねって言っている中で、やっぱりだめだったかとか、そうでもなかったかと言っていたのが、開発の途中でも設計変更の数とか、DRの遅れの状態とかでわかるようになってきたというのが、すごい進化だと思う。	不確実さやあいまいさの低減 (P)
I2 ※	before/afterで、事業の戦略会議に情報を出せるようになったのは非常に大きな成果だと思っている。今まではその会議に出ている人たちの経験でものごとを決定していたのが、客観的なデータとして情報を出せるようになったというところは非常に大きな変化だと思う。これはお前の考え方なのかって聞かれても、そうじゃないです、データが言ってるんですと答えられるようになった。それは今までにはなかった。	データによる認知バイアスの補正 (P)
I3-1	ある程度、プロセス改善を長くやっていると、定量分析で上限下限に入っていないとか、このデータが怪しいとか、色々気づくことはあるが、それをプロジェクト側に提示したとしても、行動が良い方向に変化するかというと、必ずしも、そうではないことが多かった気がする。なぜ、そんな指摘をするのか？と反応して、データを隠したりごまかしたりするなど、逆側に振れてしまうことも多かった。	利害関係者間の合意形成 (N)
I3-2	先日も上位マネージャ・クラスの人たちと話をしたときに、リスクが見えるようになるのはすごく良いが、リスクの低いプロジェクトからリスクの高いプロジェクトに人を出すというような使い方をされると困ると話していた。プロジェクトを良い方向に変化させるということは、いろいろ難しいと実感している。	利害関係者間の合意形成 (N)
I4-1 ※	before/afterでの良い面の気づきとして、以前は組織にデータがないと思われていた。それで何もやれなかったし、やらなかった。ソフトウェア開発の定量的な評価なんてできないと言われていた。本活動を始めたら、実はシステム側にたくさんデータがあるということがわかり、それを使えばプロジェクト全体の良し悪しが判断・予測できるということがわかった。ないと言われていたのが、実はあったというのが、すごく大きな気づき。	不確実さやあいまいさの低減 (P)
I4-2	事業の戦略会議などの場で、データを活用して、蓄積して、経過を追って、その経過をフィードバックして、過去のデータからみて大丈夫か？など、フィードバックがかかるようになった。これはすごい変化だと思う。今までだと、事業の戦略会議などで、データをちらっと見ることはあるが、結局は、がんばれ！がんばります！で終わるという感じだった。	利害関係者間の合意形成 (P)
I4-3	経営層も、マネージャ側と、もともと一緒にプロジェクトを苦労した仲間内だったりするので、一応、腹割って話そうぜということにはなるが、その先に踏み込めない。データを活用すれば、個人を責めるのではなく、あくまで過去の失敗の事実に照らして、今後、リスクに対してどう考えるべきかを客観的に話し合うことできるようになると思う。これは非常に大きい。	利害関係者間の合意形成 (P)
I4-4 ※	今後、シニアの人たちが経営的な判断をする立場から去り、今の現役の人たちに移っていくはず。今の現役の人たちというと、ようやく40代。そうすると、50代が経験したこととはまったく違ってくるので、経験値が下がる。DRなどに出てくるエキスパートは、50代で経験豊かだが、どちらかというと技術的な視点でしか見ない。プロジェクトリスク的な視点では、やはり上位マネージャ・クラスが見ることになるが、そこの経験値がどうなのかというと、あまり大きな人数を率いてきた経験がないなど、新たな時代のかじ取りをしたり、新たな時代の人たちの考え方や仕事のやり方を想像することが難しいかもしれない。	データによる認知バイアスの補正 (P)
I4-5 ※	ちょうど、今、旧世代と新世代の入れ替わりのギャップにあると思う。昔の人たちは同じ考え方の人たちがそのまま上がってきたので、こうだと言ったら大体こうだったのが、今は少しずれていて、こうだと言ってもいや違うっていう、そんなイメージを持っている。このギャップを埋めるためには、データはやはり必要なのだろうと感じる。	データによる認知バイアスの補正 (P)
I4-6 ※	before/afterのマイナス面は、数字が独り歩きしたときの怖さ。まだ、自分たちがいる間は数字が独り歩きしないように、うまい使い方をしようとするが、他部門を見ていると、先頭で引っ張っているリーダーたちがいなくなった途端に活動が形骸化してしまうなど、本当に正しい運用を継続できるのか心配になることがある。仕組みだけ残って、魂が抜けるというイメージ。そこは、マイナス面というよりも、気をつけておかなきゃいけないことかと思う。自分たちは、まだ本格的に展開しているわけではないので、正直、マイナス面はよくわからない。ただし、怖いのは、そうなる可能性はリスクとして常にあるということ。	利害関係者間の合意形成 (N)
I4-7 ※	プロジェクトで火を噴いているところに何回も入った経験があるが、いろいろ悪いことが起きている中で、皆が一生懸命、何とかしようとしているときに、何か新しく悪いことが起きても、誰も見ようとしない。見なかったことにする。それは今言うな、今言ったらこっちが進まなくなる、今この歩みを止めたら終わりだから言うな、あまりうるさかったら切るぞ、という雰囲気になる。こうした一致団結モードをどうするかというのは非常に難しい問題。例え、定量的にリスクを測って提案しても、最後は、リスクがあっても仕方ない、目を閉じるみたいなデータの使われ方をすると、まったく意味がない。最後に判断するのは人間なので。	データによる認知バイアスの補正 (N)

A3： RLR、SNB、RF+PDP の解釈可能性の比較分析

リッジロジスティック回帰（RLR）、superposed naive Bayes（SNB）、ランダムフォレスト（RF）について解釈可能性の比較分析を実施した。ただし、ランダムフォレスト（RF）の出力は、事後解釈性の支援ツールである部分従属プロット（PDP）を用いて評価している。付表 A3.1 は、Bugzilla と Columba の 2 つのデータセットに対する RLR、SNB、RF+PDP の予測モデルの比較であり、それぞれ説明変数の重要度ランクと方向を示している。ここで、変数の方向は、各説明変数の値の増加に対して、リスク増加（+）、リスク減少（-）、非単調変化（+/-）、一定（0）の 4 種類に分類される。また、期待される方向（expected direction）は、各変数の定義や特性から論理的に導かれたものである。

付図 A3.1 は、SNB の重要度ランク上位 3 変数の WoE（weight of evidence）と RF の重要度ランク上位 3 変数の PDP を示している。SNB の上位 3 変数、AGE、LAn、NF の方向は、それぞれリスク減少（-）、リスク増加（+）、リスク増加（+）である。一方、RF の上位 3 変数、LAn、AGE、LTn の方向は、それぞれリスク増加（+）、リスク減少（-）、非単調変化（+/-）となる。非単調変化（+/-）はこのように複雑な形状をしているため、一般に、結果の解釈が難しい。

付表 A3.2 から付表 A3.7 は、MDP データセット、すなわち、MC2、KC3、MW1、CM1、PC1、PC2、PC3、PC4、PC5、MC1、JM1 における RLR、SNB、RF+PDP の予測モデルの比較を示す。期待される方向は、各変数の定義や特性から論理的に導かれている。重要度ランクおよび方向は、モデルやデータセットによって大きくばらついているが、いくつかの共通的な特徴も観察できる。

一つ目は、重要度ランクの相関性である。SNB と RF+PDP の重要度ランクは比較的高い相関性をもっているが、RLR に関しては、特に MDP データセットにおいて、あまり相関性がない。例えば、*Halstead_content* と *Halstead_effort* の重要度ランクは、SNB と RF+PDP では比較的類似した傾向をもっているが、RLR では大きく異なっている。また、*condition_count*、*modified_condition_count*、*multiple_condition_count* は、RLR では高い重要度ランクをもつが、SNB と RF+PDP ではそうではない。こうした違いは、RLR が他の 2 つの機械学習モデルよりも変数間の多重共線性の影響を受けやすいことを示唆している。例えば、*condition_count*、*modified_condition_count*、

multiple_condition_count は、相関係数が 0.99 以上という強い相関関係をもっており、それが RLR の結果に影響を与えた可能性がある。

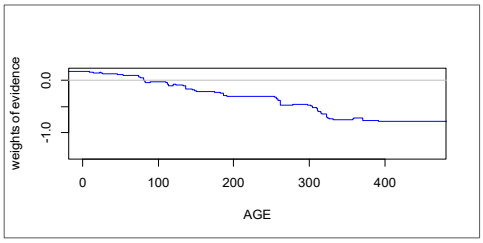
二つ目は、非単調変化（+/-）の割合である。非単調変化（+/-）は、RF+PDP においてもっとも出現しやすく、RF+PDP の出力全体の 40%超を占めていた。SNB では、全体の約 20%が非単調変化（+/-）だった。RLR は、モデルの性質上、必ずリスク増加（+）またはリスク減少（-）のいずれかとなり、非単調変化（+/-）は出現しない。非単調変化（+/-）は、付図 A3.1 からもわかるように結果の解釈が難しく、解釈可能性の観点からは好ましくない。RF は PDP を用いることで解釈可能性をある程度補うことができるが、非単調変化（+/-）の出現割合から判断すると、その結果は必ずしも解釈容易とは言い難い。

三つ目は、期待される方向と実際に観察された方向の整合性である。RLR、SNB、RF+PDP における期待される方向と実際に観察された方向の一致率は、それぞれ 45%、59%、43%であった。すなわち、SNB の出力における変数の方向がもっとも期待された方向に近かった。これは、SNB が、RLR と比較して多重共線性の影響を受けにくく、また、RF+PDP と比較して非単調変化（+/-）の割合も低いため、総じてバランスの取れた結果を出力していると考えられる。

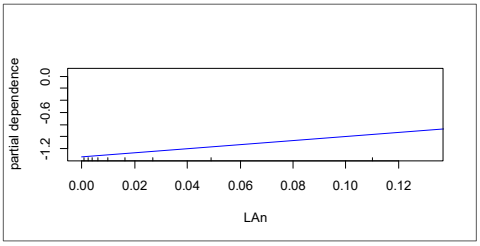
付表 A3.1：Bugzilla と Columba に対する RLR、SNB、RF+PDP の予測モデルの比較

Dimension	Metrics (expected Dir.)		Bugzilla						Columba					
			RLR		SNB		RF + PDP		RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.
Diffusion	NS	(+)	12	+	7	+	14	+	13	-	13.5	0	14	+
	ND	(+)	6	+	5	+	13	+	1	+	4	+	11	+
	NF	(+)	3	+	3	+	9	+	7	-	2	+	7	+
	Entropy	(+)	7	+	4	+/-	10	+/-	14	-	6	+/-	10	+/-
Size	LAn	(+)	11	+	2	+	1	+	11	+	7	+	5	+
	LDn	(+)	2	-	9	+	4	0	8	+	8	+	6	+
	LTn	(+)	8	-	6	+/-	3	+/-	10	+	1	+	1	+
Purpose	FIX	(+)	4	+	14	0	11	+	12	+	13.5	0	13	+
History	NDEV	(+)	14	-	12	+	8	+/-	9	+	11	+	12	+
	AGE	(-)	1	-	1	-	2	-	4	-	10	+/-	8	+
	NUCn	(+)	13	+	8	-	12	-	2	-	3	-	9	-
Experience	EXP	(-)	10	-	10	-	5	+/-	3	-	5	+/-	2	+/-
	REXP	(-)	5	-	11	-	7	+/-	6	+	12	+/-	3	+/-
	SEXP	(-)	9	+	13	+/-	6	+/-	5	+	9	+/-	4	+/-

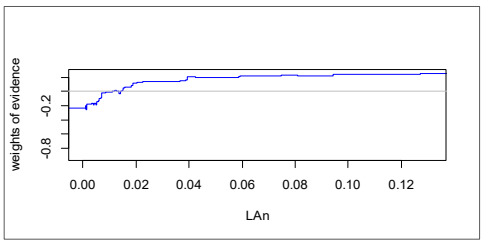
Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)



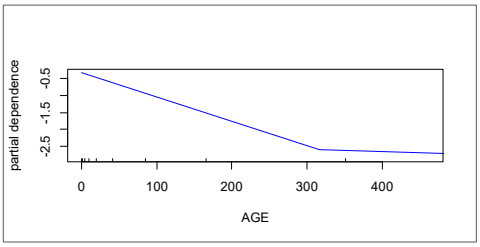
(a) SNB: AGE (Rank=1, Dir.=-)



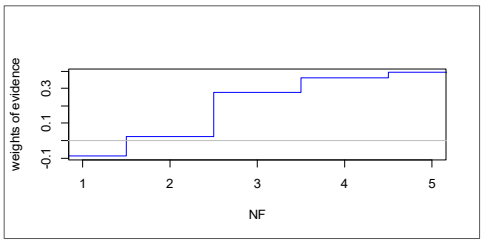
(d) RF+PDP: LAn (Rank=1, Dir.=+)



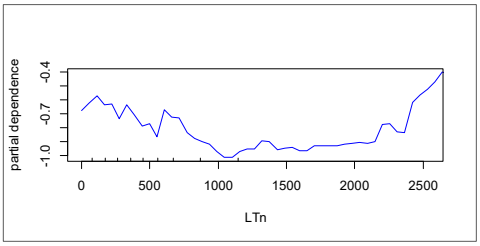
(b) SNB: LAn (Rank=2, Dir.=+)



(e) RF+PDP: AGE (Rank=2, Dir.=-)



(c) SNB: NF (Rank=3, Dir.=+)



(f) RF+PDP: LTn (Rank=3, Dir.=+/-)

付図 A3.1：SNB の重要度上位 3 変数の WoE と RF の重要度上位 3 変数の PDP

付表 A3.2：MC2 と KC3 に対する RLR、SNB、RF+PDP の予測モデルの比較

	Metrics (expected Dir.)		MC2						KC3					
			RLR		SNB		RF + PDP		RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.
LOC counts	LOC_total	(+)	33	-	28	+	27	+/-	11	-	16	+	12	+/-
	LOC_blank	(+)	14	+	24	+	9	+	18	-	3	+	10	+
	LOC_code_and_comment	(+)	28	-	34	+	33	+/-	24	+	1	+	1	+
	LOC_comments	(+)	35	+	9	+	4	+	15	-	34	+	20	+
	LOC_executable	(+)	37	-	29	+	29	+/-	10	-	15	+	22	+/-
	Number_of_lines	(+)	13	-	19	+	5	+/-	6	+	10	+	3	+/-
Halstead attributes	content	(+)	20	+	35	+	18	+/-	16	+	7	+	6	+/-
	difficulty	(+)	23	-	1	+	1	+	26	-	27	+	18	+/-
	effort	(+)	10	+	2.5	+	2	+	31	-	5.5	+/-	4	+/-
	error_est	(+)	1	+	30	+	30	+	3	-	17	+	28	+
	length	(+)	22	-	25	+	25	+/-	21	-	11	+	23	+
	level	(+)	32	-	8	-	20	-	28	+	36	-	25	+
	prog_time	(+)	11	+	2.5	+	6	+	30	-	5.5	+/-	5	+
	volume	(+)	2	-	26	+	24	+/-	1	+	9	+	14	+/-
	num_operands	(+)	17	+	23	+	19	+	13	-	19	+	9	+/-
	num_operators	(+)	12	-	22	+	22	+	32	-	13	+	19	+/-
	num_unique_operands	(+)	15	-	32	+	38	+/-	14	-	18	+	7	+
	num_unique_operators	(+)	16	+	18	+	10	+	25	+	23	+	26	+/-
McCabe attributes	cyclomatic_complexity	(+)	8	-	10	+	15	+	8	+	14	+	17	+
	cyclomatic_density	(+)	36	-	38	+	17	+/-	29	+	22	-	16	+/-
	design_complexity	(+)	18	+	21	+	26	+	39	+	25	+	30	+
	essential_complexity	(+)	19	-	17	+	32	+	37	-	32	+	34	+
Miscellaneous	branch_count	(+)	5	+	16	+	23	+	7	-	12	+	8	+
	call_pairs	(+)	24	-	12	+	12	+	19	-	8	+	11	+
	condition_count	(+)	6	-	13.5	+	35	+	5	+	28.5	+	35	+/-
	decision_count	(+)	4	-	20	+	31	+	9	+	33	+	27	+
	decision_density	(+)	34	-	33	+	39	+/-	38	-	35	0	39	0
	design_density	(+)	29	-	37	+/-	21	+/-	35	+	37	-	33	-
	edge_count	(+)	21	+	5	+	3	+	17	+	20	+	24	+/-
	essential_density	(+)	39	-	6	+	11	+	23	+	38	+	38	+
	global_data_complexity	(+)	25	-	27	+	37	+	12	-	24	+	36	+
	global_data_density	(+)	38	+	11	-	16	-	27	+	26	+	32	+/-
	maintenance_severity	(+)	26	+	7	+	13	+/-	22	-	30	-	15	+/-
	modified_condition_count	(+)	7	-	15	+	36	+	4	+	31	+	31	+/-
	multiple_condition_count	(+)	3	+	13.5	+	34	+	2	-	28.5	+	29	+
	node_count	(+)	9	+	4	+	14	+	34	+	21	+	21	+/-
	normalized_cyclomatic_compl.	(+)	31	+	36	+/-	28	+/-	20	-	4	-	13	+/-
	parameter_count	(+)	27	+	39	+	8	+	36	+	39	0	37	+/-
	percent_comments	(-)	30	+	31	+	7	+	33	+	2	+	2	+

Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)

付表 A3.3：MW1 と CM1 に対する RLR、SNB、RF+PDP の予測モデルの比較

	Metrics (expected Dir.)		MW1						CM1					
			RLR		SNB		RF + PDP		RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.
LOC counts	LOC_total	(+)	27	-	13	+	10	+/-	17	+	11	+	3	+
	LOC_blank	(+)	37	+	17	+	16	+/-	21	+	18	+/-	5	+/-
	LOC_code_and_comment	(+)	5	-	36.5	0	37	-	33	-	21	+/-	7	+/-
	LOC_comments	(+)	13	+	9	+	9	+	12	+	1	+	1	+
	LOC_executable	(+)	29	+	12	+	18	+/-	16	+	10	+	9	+
	Number_of_lines	(+)	36	+	10	+	5	+/-	9	-	5	+	10	+/-
Halstead attributes	content	(+)	35	+	1	+	1	+/-	18	+	2	+	6	+
	difficulty	(+)	23	-	20	+/-	20	+/-	20	-	24	+/-	18	+/-
	effort	(+)	33	+	15.5	+	8	+/-	22	-	15.5	+	19	+
	error_est	(+)	4	+	11	+	12	+	2	+	12	+	27	+/-
	length	(+)	15	-	8	+	13	+/-	24	-	13	+	23	+/-
	level	(+)	26	-	29	+/-	24	+/-	32	+	34	-	26	+
	prog_time	(+)	32	+	15.5	+	19	+/-	23	-	15.5	+	21	+
	volume	(+)	6	-	2	+	11	+/-	3	-	6	+	12	+
	num_operands	(+)	7	-	14	+	21	+	14	+	20	+	17	+/-
	num_operators	(+)	16	+	5	+	7	+	13	-	8	+	20	+
	num_unique_operands	(+)	14	+	4	+	6	+	10	-	4	+	11	+
	num_unique_operators	(+)	28	-	32	+	22	+	11	+	7	+	13	+
McCabe attributes	cyclomatic_complexity	(+)	22	-	27	+	33	+/-	8	+	29	+	30	+
	cyclomatic_density	(+)	10	-	21	-	15	+/-	19	-	19	-	24	-
	design_complexity	(+)	19	+	18	+	26	+/-	31	+	26	+	28	+
	essential_complexity	(+)	17	-	33	+	35	+/-	36	-	37	-	37	+
Miscellaneous	branch_count	(+)	9	-	19	+	31	+/-	7	-	28	+	33	+/-
	call_pairs	(+)	34	+	3	+	4	+	37	-	14	+/-	16	+
	condition_count	(+)	3	+	22	+	28	+/-	5	+	31	+	32	+/-
	decision_count	(+)	8	+	28	+	27	+	6	+	30	+	29	+
	decision_density	(+)	12	-	34	+	30	0	34	+	35	+/-	34	+/-
	design_density	(+)	21	-	31	+/-	29	+/-	30	+	15	+	2	+/-
	edge_count	(+)	25	+	7	+/-	3	+/-	29	+	25	+/-	25	+/-
	essential_density	(+)	24	+	36.5	0	36	+	26	+	33	-	31	+/-
	global_data_complexity	(+)												
	global_data_density	(+)												
	maintenance_severity	(+)	30	-	30	-	23	+/-	27	-	27	-	22	+/-
	modified_condition_count	(+)	2	+	23	+	25	+/-	4	+	36	+	35	+
	multiple_condition_count	(+)	1	-	24	+	32	+/-	1	-	32	+	36	+/-
	node_count	(+)	18	+	6	+	2	+/-	15	-	22	+/-	14	+/-
	normalized_cyclomatic_compl.	(+)	11	+	25	-	17	+/-	28	+	9	-	15	+/-
	parameter_count	(+)	31	-	35	-	34	+/-	35	-	23	-	8	-
	percent_comments	(-)	20	-	26	+/-	14	+/-	25	+	3	+	4	+/-

Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)

付表 A3.4：PC1 と PC2 に対する RLR、SNB、RF+PDP の予測モデルの比較

	Metrics (expected Dir.)		PC1						PC2					
			RLR		SNB		RF + PDP		RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.
LOC counts	LOC_total	(+)	22	-	8	+	8	+	13	-	13	+	29	+
	LOC_blank	(+)	23	-	3	+	2	+						
	LOC_code_and_comment	(+)	28	+	24	+	6	+	11	-	15	+	10	+
	LOC_comments	(+)	33	-	19	+/-	1	+	10	-	7	+	8	+
	LOC_executable	(+)	21	-	9	+	11	+	21	+	33	-	14	-
	Number_of_lines	(+)	10	+	2	+	5	+	6	+	12	+	17	+/-
Halstead attributes	content	(+)	25	-	1	+	3	+	18	-	2	+	2	+
	difficulty	(+)	8	-	17	+/-	12	-	35	+	11	+	6	+
	effort	(+)	18	-	10.5	+	15	-	34	+	3.5	+	18	+
	error_est	(+)	5	-	15	+	22	+	3	+	14	+	16	+
	length	(+)	32	+	12	+	19	+/-	20	+	6	+	9	+
	level	(+)	37	-	28	-	29	+	31	-	18	-	24	+/-
	prog_time	(+)	19	-	10.5	+	23	-	36	+	3.5	+	12	+/-
	volume	(+)	3	+	7	+	17	+	2	-	5	+	3	+
	num_operands	(+)	13	+	14	+	14	+/-	22	-	9	+	13	+
	num_operators	(+)	20	-	13	+	21	+/-	15	+	8	+	4	+/-
	num_unique_operands	(+)	11	-	5	+	4	+	12	+	10	+	7	+
	num_unique_operators	(+)	16	+	22	+	20	+/-	25	-	22	+	15	-
McCabe attributes	cyclomatic_complexity	(+)	4	+	30	+	28	+/-	5	-	30	+	32	+
	cyclomatic_density	(+)	24	-	6	-	10	-	29	-	16	-	20	-
	design_complexity	(+)	35	-	27	+/-	26	+	16	+	35	+	26	0
	essential_complexity	(+)	26	+	37	-	37	+/-	24	-	34	+	35	+
Miscellaneous	branch_count	(+)	34	-	29	+/-	31	+/-	8	+	28	+	31	0
	call_pairs	(+)	31	-	20	+/-	24	+	27	-	24	+	27	-
	condition_count	(+)	9	-	33	+/-	27	+	7	+	25	+	30	+
	decision_count	(+)	17	-	35	+/-	36	+/-	9	+	29	+	33	+
	decision_density	(+)	30	-	32	+/-	34	+	28	-	32	+	22	+/-
	design_density	(+)	29	+	25	+/-	25	+/-	26	-	19	-	21	-
	edge_count	(+)	1	-	18	+/-	13	-	17	-	21	+	23	+/-
	essential_density	(+)	12	+	31	+/-	32	+	32	+	36	0	36	+
	global_data_complexity	(+)												
	global_data_density	(+)												
	maintenance_severity	(+)	14	-	23	+/-	18	+/-	33	-	23	-	25	+/-
	modified_condition_count	(+)	7	-	36	+	33	+/-	4	+	27	+	34	+
	multiple_condition_count	(+)	6	+	34	+/-	30	-	1	-	26	+	28	0
	node_count	(+)	2	+	21	+/-	16	+/-	19	+	20	+	11	-
	normalized_cyclomatic_compl.	(+)	15	-	4	-	9	-	23	+	17	-	19	-
	parameter_count	(+)	27	-	26	-	35	0	30	-	31	+	5	+/-
	percent_comments	(-)	36	-	16	+	7	+/-	14	+	1	+	1	+

Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)

付表 A3.5：PC3 と PC4 に対する RLR、SNB、RF+PDP の予測モデルの比較

	Metrics (expected Dir.)		PC3						PC4					
			RLR		SNB		RF + PDP		RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.
LOC counts	LOC_total	(+)	26	+	9	+	20	+	13	+	19	+	7	+
	LOC_blank	(+)	18	+	5	+	1	+	32	+	3	+/-	5	+
	LOC_code_and_comment	(+)	36	-	18	+	9	+	17	+	1	+	1	+
	LOC_comments	(+)	21	+	19	+	5	+	23	-	13	-	4	+/-
	LOC_executable	(+)	25	+	14	+/-	11	+/-	14	+	26	+/-	18	+/-
	Number_of_lines	(+)	12	-	3	+	6	+	12	-	7	+	8	+/-
Halstead attributes	content	(+)	13	-	1	+	2	+	19	-	6	+	6	+
	difficulty	(+)	30	-	16	+/-	7	+/-	35	-	12	+/-	15	+/-
	effort	(+)	24	+	12.5	+	10	+	30	-	9.5	+/-	13	+/-
	error_est	(+)	1	-	10	+	26	+	10	-	27	+/-	26	-
	length	(+)	7	+	7	+	17	+	20	-	17	+/-	20	+/-
	level	(+)	34	-	24	+/-	29	+/-	24	+	24	+	30	+
	prog_time	(+)	27	+	12.5	+	13	+	29	-	9.5	+/-	14	-
	volume	(+)	2	+	4	+	16	+	8	+	8	+/-	9	+
	num_operands	(+)	20	+	6	+	8	+	28	-	20	+/-	12	+/-
	num_operators	(+)	4	+	8	+	15	+	16	-	21	+/-	11	+/-
	num_unique_operands	(+)	5	+	2	+	3	+	15	-	25	+/-	19	+/-
	num_unique_operators	(+)	19	-	23	+	23	+/-	36	+	28	+/-	21	+/-
McCabe attributes	cyclomatic_complexity	(+)	10	+	31	+	35	+/-	5	+	36	-	35	+/-
	cyclomatic_density	(+)	32	+	21	-	12	+/-	11	-	4	-	3	-
	design_complexity	(+)	22	-	29	+	28	+/-	25	-	30	-	33	+/-
	essential_complexity	(+)	23	-	36	-	37	+/-	21	-	32	-	36	+
Miscellaneous	branch_count	(+)	9	-	30	+	34	+/-	9	-	37	-	34	+/-
	call_pairs	(+)	35	-	26	+	21	+/-	34	+	22	+/-	24	+/-
	condition_count	(+)	8	-	32	+	31	+/-	6	+	14	+	16	+
	decision_count	(+)	6	-	37	+	33	+	7	+	18	+	25	+
	decision_density	(+)	31	+	22	+	18	+	31	+	11	+	17	+
	design_density	(+)	29	+	27	+/-	24	+/-	37	+	33	+/-	31	+/-
	edge_count	(+)	33	+	17	+	22	+/-	1	-	29	-	28	+/-
	essential_density	(+)	16	+	35	+/-	36	+/-	26	+	34	-	37	+/-
	global_data_complexity	(+)												
	global_data_density	(+)												
	maintenance_severity	(+)	15	-	25	-	25	+/-	18	-	35	-	27	+/-
	modified_condition_count	(+)	11	-	34	+	32	+	4	+	15	+	29	+
	multiple_condition_count	(+)	3	+	33	+	27	+/-	3	-	16	+	22	+
	node_count	(+)	17	-	20	+	19	+/-	2	+	31	-	23	+/-
	normalized_cyclomatic_compl.	(+)	14	-	15	-	14	+/-	27	+	5	-	10	-
	parameter_count	(+)	37	-	28	-	30	+	33	-	23	-	32	+
	percent_comments	(-)	28	+	11	+	4	+/-	22	+	2	+	2	+/-

Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)

付表 A3.6：PC5 と MC1 に対する RLR、SNB、RF+PDP の予測モデルの比較

	Metrics (expected Dir.)		PC5						MC1					
			RLR		SNB		RF + PDP		RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.	Rank	Dir.
LOC counts	LOC_total	(+)	9	-	5	+	8	+	16	+	16	+	12	+/-
	LOC_blank	(+)	14	-	21	+	23	+	29	-	20	+	22	+/-
	LOC_code_and_comment	(+)	17	+	27	+	10	+	21	+	10	+	2	+
	LOC_comments	(+)	12	-	31	+	19	+/-	34	-	24	+	11	+/-
	LOC_executable	(+)	8	-	11	+	21	+	17	+	11	+/-	15	+/-
	Number_of_lines	(+)	6	+	12	+	11	+	18	+	7	+	3	+/-
Halstead attributes	content	(+)	27	-	19	+	4	-	22	-	5	+	4	+/-
	difficulty	(+)	35	+	7	+	3	+/-	19	+	9	+/-	13	+/-
	effort	(+)	29	+	1.5	+	1	-	7	-	3.5	+/-	5	+
	error_est	(+)	2	-	8	+	12	+	2	-	21	+/-	26	+/-
	length	(+)	16	-	6	+	16	+	13	-	6	+/-	7	+/-
	level	(+)	20	-	15	-	28	+/-	30	+	19	+/-	25	+/-
	prog_time	(+)	32	+	1.5	+	9	+	6	-	3.5	+/-	9	+
	volume	(+)	1	+	3	+	7	+	1	+	2	+/-	6	+
	num_operands	(+)	13	+	4	+	6	+	15	-	13	+/-	21	+
	num_operators	(+)	11	-	9	+	15	+	11	-	8	+/-	8	+/-
	num_unique_operands	(+)	34	+	14	+	20	+	32	-	15	+	14	+/-
	num_unique_operators	(+)	18	+	17	+	2	+/-	37	+	29	+/-	20	+/-
McCabe attributes	cyclomatic_complexity	(+)	37	+	28	+	34	+	25	+	30	+/-	29	+
	cyclomatic_density	(+)	21	-	30	-	17	+/-	28	-	14	+/-	19	+/-
	design_complexity	(+)	38	-	18	+	26	+	14	-	33	+	32	+
	essential_complexity	(+)	28	-	32	+	35	+	26	-	32	+/-	34	+/-
Miscellaneous	branch_count	(+)	5	+	26	+	29	+	9	-	26	+	31	+/-
	call_pairs	(+)	36	-	16	+	22	+/-	31	+	31	+	10	+
	condition_count	(+)	10	-	22	+	30	+	12	-	23	+/-	27	+/-
	decision_count	(+)	15	-	25	+	33	+	3	-	28	+/-	30	+
	decision_density	(+)												
	design_density	(+)	19	+	37	+/-	27	+/-	24	-	36.5	0	37	-
	edge_count	(+)	3	-	13	+	18	+	8	-	12	+/-	18	+
	essential_density	(+)	31	+	36	+	37	+/-	33	+	36.5	0	36	+
	global_data_complexity	(+)	26	+	20	+	25	+	36	+	27	+/-	33	+
	global_data_density	(+)	23	+	34	+	24	+/-	23	-	36.5	0	35	+
	maintenance_severity	(+)	24	-	33	-	32	+/-	27	+	36.5	0	38	+
	modified_condition_count	(+)	7	-	23	+	36	+	10	+	25	+/-	28	+
	multiple_condition_count	(+)	22	-	24	+	31	+	5	+	22	+/-	23	+
	node_count	(+)	4	+	10	+	14	+	4	+	17	+/-	16	+
	normalized_cyclomatic_compl.	(+)	33	-	35	-	13	+/-	35	+	18	+/-	17	+/-
	parameter_count	(+)	25	-	38	-	38	+	38	+	34	+/-	24	+/-
	percent_comments	(-)	30	-	29	+/-	5	+/-	20	+	1	+	1	+/-

Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)

付表 A3.7：JM1 に対する RLR、SNB、RF+PDP の予測モデルの比較

	Metrics (expected Dir.)		JM1					
			RLR		SNB		RF + PDP	
			Rank	Dir.	Rank	Dir.	Rank	Dir.
LOC counts	LOC_total	(+)	1	+	2	+	2	+
	LOC_blank	(+)	15	-	1	+	8	+
	LOC_code_and_comment	(+)	20	+	20	+	21	+
	LOC_comments	(+)	13	-	19	+	14	+
	LOC_executable	(+)	2	-	3	+	4	+
Halstead attributes	content	(+)	17	+	9	+	3	+/-
	difficulty	(+)	10	-	15	+	7	+/-
	effort	(+)	19	+	10	+	6	-
	error_est	(+)	8	-	8	+	17	+
	length	(+)	9	+	7	+	12	+/-
	level	(+)	16	+	18	-	19	+/-
	prog_time	(+)	18	+	11	+	5	-
	volume	(+)	3	-	4	+	1	+
	num_operands	(+)	12	+	12	+	11	+
	num_operators	(+)	6	+	5	+	9	+/-
	num_unique_operands	(+)	11	+	6	+	10	+/-
	num_unique_operators	(+)	7	+	13	+	13	+/-
McCabe attributes	cyclomatic_complexity	(+)	5	-	16	+	18	+
	design_complexity	(+)	21	+	14	+	16	+
	essential_complexity	(+)	14	-	21	+	20	+
Miscellaneous	branch_count	(+)	4	+	17	+	15	+

Direction: increasing (+), decreasing (-), non-monotonic (+/-), constant (0)

謝辞

本研究を進めるにあたり、多くの方々のご支援、ご協力を頂き深く感謝いたします。

まず、北陸先端科学技術大学院大学 知識マネジメント領域 教授 内平直志先生には、指導教官として大変なご心配とご苦労をおかけしつつ、いかなるときも常に親身になって励ましのお言葉をかけていただき、今日まで暖かく導いて下さったことを深く感謝いたします。博士論文を無事完成できたのは先生のご指導のおかげです。

また、学位論文審査にあたり丁寧にご指導いただき、貴重なアドバイスをいただきました、北陸先端科学技術大学院大学 神田陽治先生、白肌邦生先生、Dam Hieu-Chi 先生ならびに国立情報学研究所（NII）吉岡信和先生にお礼を申し上げます。

講義やゼミでは、井川康夫先生、伊藤泰信先生、安永裕幸先生、遠山亮子先生（現、中央大学）をはじめとする先生方および JAIST 東京社会人コースの社会人学生諸氏に多くのアドバイスと励ましをいただきました。

本研究は、株式会社 東芝での長年の経験に基づくものであり、その間ご指導ご支援いただきました数多くの上長、同僚の皆さまに感謝いたします。特に、本論文のインタビュー調査にご協力頂いた宮崎早苗氏，平井桂子氏、早瀬健夫氏、緒方勝氏、清野貴子氏、山本恭子氏、梅田泰隆氏、飯村拓志氏には大変お世話になりました。

東京大学 精密機械工学科 木村文彦先生（現、東京大学名誉教授）、乾正知先生（現、茨城大学）には、学部および修士時代にご指導いただき、深く探求することの重要さなど、研究者としての大切な心構えを教えていただきました。

米国・スタンフォード大学 石井浩介先生（2009 年 3 月逝去）には、私が東芝から派遣されて Visiting Industrial Associate（VIA）として滞在していたときにご指導いただきました。帰国の際、博士号を取ることを熱心に勧めていただき、色々な道筋をつけて下さったのですが､私の怠慢により先生のご存命中にご期待に沿うことができず、大変心残りになっていました。その後、先生と生前にご縁のあった方々が集まったパーティの席で、諸先輩方からあらためて背中を押していただき、決心して今日に至っています。ご縁とお導きに感謝します。

最後に、私の成長を温かく見守ってくれた父（2017 年 7 月逝去）と母に深く感謝します。そして、妻・珠紀の理解と支えによって、研究のモチベーションを維持し博士論文に注力することができました。ここに、深く感謝します。

研究業績リスト

A. 学術誌掲載論文

A-1.（査読あり）※ 2018 Journal Impact Factor: 4.457

Mori, Toshiki, and Naoshi Uchihira, 2019, "Balancing the trade-off between accuracy and interpretability in software defect prediction," *Empirical Software Engineering*, 24.2: 779-825.

A-2.（査読あり）

森俊樹・内平直志，2019,「プロジェクトとプログラムのリスクマネジメントにおける機械学習と知識創造の統合アプローチ」『国際 P2M 学会誌』，14.1: 415-435.

B. 国際学会口頭発表論文

B-1.（査読あり）

Mori, Toshiki, Shurei Tamura, and Shingo Kakui, 2013, "Incremental estimation of project failure risk with Naive Bayes classifier," *7th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, Baltimore: IEEE, 283-286.

B-2.（査読あり）

Mori, Toshiki, 2015, “Superposed naive Bayes for accurate and interpretable prediction,” *14th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Miami: IEEE, 6 pages.

B-3.（査読あり）※ Journal First Paper

Mori, Toshiki, and Naoshi Uchihira, 2019, "Balancing the trade-off between accuracy and interpretability in software defect prediction," *34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, San Diego: IEEE/ACM.

C. 国内学会口頭発表論文

C-1.（簡易査読あり）

森俊樹・内平直志，2018,「リスクマネジメントにおける機械学習と知識創造の統合アプローチ」，プロジェクトマネジメント学会 2018 年度春季研究発表大会，東洋大学，6 pages.

D. その他

D-1.（簡易査読あり）

森俊樹・覚井真吾・田村朱麗・藤巻昇，2013,「プロジェクト失敗リスク予測モデルの構築 」『プロジェクトマネジメント学会誌』，15.4: 3-8.

D-2.（簡易査読あり）

森俊樹・覚井真吾・田村朱麗，2014,「プロジェクト失敗リスク予測モデル」『東芝レビュー』，69.1: 47-50.