

博士論文 平成 28(2016)年度

企業間アライアンスの相互補完数理モデルの提案と応用

慶應義塾大学大学院政策・メディア研究科

富田 賢

主論文要旨

主論文題目：

企業間アライアンスの相互補完数理モデルの提案と応用

氏名：富田 賢

内容の要旨：

経営学におけるアライアンス研究では、これまで、企業間アライアンスのマッチング段階での成立メカニズムを表現する数理モデルが存在しなかった。そのため、2社間のアライアンスの関係性を数値として把握することができなかった。本研究では、資源ベース理論 (Resource Based View, RBV) を基本フレームワークとしつつ、フローの概念やギブ・アンド・テイクの考え方を組み合わせ、サイバー・インフォマティクス (CI) 分野の数学のモデル化の手法を用いて、世界初となる数理モデルの構築を行った。

提案する相互補完モデルでは、A社の弱みをB社の強みで補い、B社の弱みをA社の強みで補うと考え、企業間の強みと弱みの相互補完関係が強い時に、アライアンスは成立すると考える。これは、フロー・インテンシティ (Flow Intensity) が強く、かつ、フロー・バランス (Flow Balance) が保たれている時にアライアンスは成立しやすいとも説明できる。

本研究では、営業力や技術力、生産力などの主として経営資源を代表する特徴ごとの評点付けを、1次元行列と2極のベクトルで数学表現し、2社の組み合わせを2次元マップの上で示し、そして、最大の相互補完関係の点からの距離で相互補完強度を数学的に表現することとした。

提案するモデルは、Python言語で実装し、筆者の152社のコンサルティング先企業の実データを用いて、相互補完強度係数を算出した。アライアンスが成立した組み合わせと不成立となった組み合わせの係数の比較から、当モデルの正当性を確認した。

加えて、相互補完モデルを発展させ、営業エリアが異なる企業間のアライアンスについては加算モデルとして追加した。また、各企業の意欲の度合いについては相乗モデルとして追加した。それらについても、Python言語で実装し、係数を算出し、同じく、平均、中央値、最頻値などでの比較検討を行い、モデルの正当性を確認した。

企業間アライアンスの成立のメカニズムを説明する数理モデルを提案し、2社間のアライアンス成立の関係性を数値として演算可能 (コンピューティショナル) にしたことが本研究の成果である。他方では、CI分野の手法のアプリケーション分野を、アライアンス研究に拡大することにも寄与した。

提案するモデルは、フレキシブルなものであり、ユーザーは、パラメーター数や評点付け、重み付けなどを自由に変えられ様々な企業データで試してみることができる。さらには提案するモデルは、企業間だけでなく、チーム構築や結婚のマッチングなどの人と人との関係や国と国とのアライアンスにも応用でき、発展可能性がある。

キーワード：企業間アライアンス、数理モデル、マッチング・モデル、資源ベース理論 (RBV)、相互補完関係、Python言語

Thesis Abstract

Title of Thesis:

A New Mathematical Model of Mutually Complementary for Corporate Alliances

Name: Satoshi Tomita

Summary of Thesis:

In the research of corporate alliances in business administration studies, there have not been any mathematical models presented to numerically represent the mechanism of the matching phase of corporate alliance formation between two companies. Using the resource based view (RBV) as a base framework combined with the concepts of flow and Give-and-Take along with mathematical modeling techniques used in the cyber-informatics (CI) field we construct the world's first mathematical model to numerically calculate and express the mechanism of the establishment of corporate alliances.

We consider a corporate alliance to be successful when the strengths of two companies complement the weaknesses of each other and there is a stronger mutual complementary relationship of these strengths and weaknesses between both companies. Corporate alliances are considered more likely to be successful when the flow intensity is strong and flow balance is maintained.

The characteristics representing main management resources including sales force, technical ability, capital resources, and human resources are graded and expressed as a one-dimensional matrix and bipolar vector. The relationship between two companies is then expressed as a two-dimensional mapping with the mutually complementary strength expressed as the distance from the point of maximum mutual complementarity.

This model is implemented in Python, making use of actual empirical data of 152 Japanese companies in order to calculate the mutually complementary strength coefficients and compare them for companies in both successful and unsuccessful alliances to verify the model validity.

This Mutually Complementary Model was further developed to include an "addition model" for corporate alliances between companies in different business areas and a "synergistic model" in order to measure the degree of willingness of each company in a corporate alliance.

Making use of this model, we can even select the optimal alliance partner in the case of multiple potential alliance partners.

This proposed model is a flexible model that may be freely modified for future use to conform to other desired models. Users can experiment with the number of parameters, grading and weighting with various other corporate data not only for expressing relationships between companies, but potentially for application to human relationship matters and even alliances between countries.

Keywords : Corporate Alliance, Mathematical Model, Matching Model, Resource Based View (RBV), Mutually Complementary Relationship, Python Programming Language

目次

第1章 本研究の概要と意義

- 1.1 本研究の動機と概要
- 1.2 経営学領域に CI のモデル化の手法を導入
- 1.3 アライアンスの定義と研究対象とする段階

第2章 先行研究のサーベイ

- 2.1 アライアンス研究の概観
- 2.2 本研究の根拠となる資源ベース理論によるアライアンス研究
- 2.3 アライアンス研究における他の理論(取引コスト理論、ゲーム理論等)
- 2.4 多くの計量分析と数理モデルの欠如
- 2.5 マッチングの段階に研究対象を拡大
- 2.6 オープン・イノベーション研究との関係
- 2.7 人間関係のギブ・アンド・テイクに関する社会的交換理論

第3章 アライアンスの相互補完モデルの構築

- 3.1 相互補完モデル構築の前提としたアライアンス成立の考え方
- 3.2 相互補完の捉え方と物理モデルの応用
- 3.3 強みと弱みの8つの特徴の選定
- 3.4 企業の強み・弱みの1次元行列と2極ベクトルでの表現
- 3.5 正の整数の総和と負の整数の総和
- 3.6 最大の強さの相互補完関係の状態
- 3.7 強度をマキシマム・ポイントからの距離で数学的に表現
- 3.8 相互補完強度と相互補完強度係数の数式化

第4章 フロー・インテンシティとフロー・バランスの概念の導入

- 4.1 フロー・インテンシティとフロー・バランスの概念の導入
- 4.2 相互補完モデルをギブ・アンド・テイクの考え方で説明
- 4.3 アライアンスが成立しないパターン1：
バランスが保たれておらず、一方的な関係性の時
- 4.4 アライアンスが成立しないパターン2：
フロー・インテンシティがない時

- 4.5 最大の相互補完関係の状態：
最大のフロー・インテンシティで、バランスが均衡している時
- 4.6 相互補完モデルの図のグレーの部分の説明

第5章 152社のコンサルティング先企業データと評点付け方法

- 5.1 152社の企業データの内訳
- 5.2 当データの特殊性と有用性
- 5.3 アライアンスの組み合わせの範囲の限定
- 5.4 強み・弱みの8つの特徴の選定
- 5.5 評点付けにおける主観性排除への対応
- 5.6 アライアンス成立・不成立のカウントの定義と件数

第6章 相互補完強度係数のPython言語での算出と検証

- 6.1 Python言語による提案したモデルの実装
- 6.2 成立した組み合わせと不成立となった組み合わせの相互補完強度係数の分布
- 6.3 成立・不成立・全体の相互補完強度係数の平均
- 6.4 成立・不成立・全体の評点の実数による検証
- 6.5 相互補完強度係数の外れ値に関する検討
- 6.6 複数の潜在的なアライアンス候補先からの最適なパートナーの選択への利用
- 6.7 相互補完モデルの応用可能性
- 6.8 相互補完モデルの限界
- 6.9 相互補完モデルのまとめ

第7章 モデルの発展1：アライアンスの加算モデルの構築と検証

- 7.1 アライアンスの相互補完数理モデルの未対応部分
- 7.2 日本国内の地域、ネット販売、海外販売
- 7.3 加算モデルの評点の付け方
- 7.4 相互補完・加算モデルの数式化
- 7.5 152社の企業データについてのPython言語での演算結果
- 7.6 相互補完・加算モデルのまとめ

第 8 章 モデルの発展 2 : アライアンスの相乗モデルの構築と検証

- 8.1 相乗モデルの追加 ～意欲の重要性と影響～
- 8.2 相乗モデルでの評点の付け方
- 8.3 相互補完・加算・相乗モデルの数式化
- 8.4 152 社の企業データについての Python 言語での演算結果
- 8.5 相互補完・加算・相乗モデルのまとめ

第 9 章 本研究のまとめ ～研究の成果と今後の研究課題～

- 9.1 本研究の成果
- 9.2 経営学と CI の組み合わせにより、双方において貢献
- 9.3 当モデルの利用面の意義
- 9.4 当モデルの改良と今後の研究課題

謝辞

参考文献

- 付録 1 : 相互補完モデルの最もシンプルな特徴数 4 つでの構築
- 付録 2 : 8 つの特徴数での評点付けの 4 つの因子としてのクライテリア
- 付録 3 : 特徴数が奇数の場合の相互補完モデルについての考察
- 付録 4 : 相互補完モデルの Python 言語のプログラミング
- 付録 5 : 加算モデル・相乗モデルの Python 言語のプログラミング

第1章 本研究の概要と意義

1.1 本研究の動機と概要

本研究は、企業間のアライアンス成立のメカニズムを表現する世界初となる数理モデルを提案し、提案するモデルによって、企業間アライアンス成立の関係性を数値として演算可能にした研究である。

筆者は、コンサルティング会社を経営しており、2008年5月から2015年3月までの約7年間で、152社の日本企業に対して、アライアンスによる新規事業立ち上げや売上向上についてのコンサルティングを行った。本研究は、その実務経験に基づいた研究内容となっており、実際のコンサルティング先企業データを用いて、提案したモデルを検証している。企業間アライアンス成立の関係性を数値として把握できるようにできないかという問題意識が、本研究の動機となっている。

これまで、経営学におけるアライアンス研究においては、2社間のアライアンスが成立するメカニズムを表現する数理モデルが存在しなかった。マッチングの数理モデルとしては、マーケット・デザイン分野では存在するが、経営学の企業間アライアンスの成立のマッチング段階を表現する数理モデルは存在していなかった。それゆえ、企業間のアライアンスにおける2社の関係性を数値として算出することができない状況にあった。

本研究では、アライアンス研究で用いられてきている経営学の理論の中から、資源ベース理論(Resource Based View、RBV)を基本的な理論フレームワークとして採用し、アライアンスは経営資源の交換であるとの先行研究を発展させ、2社間の強みと弱みの相互補完がより強い時に、アライアンスは成立しやすいと考えた。さらに、物理学のフローの概念や人間関係におけるギブ・アンド・テイクの考え方も取り込み、フロー・インテンシティ(Flow Intensity)とフロー・バランス(Flow Balance)の概念を導入することで、数理モデルを構築した。

企業間アライアンスにおける2社間の強みと弱みの相互補完関係を数学的に捉えるにあたり、企業の強みと弱みを示す特徴ごとの各企業の評点を、1次元行列と2極のベクトルで表現し、最大の相互補完関係を示す点からの距離で、相互補完の強さを数学表現することとした。このよう

に、最大の相互補完関係を示す点からの距離で、強度を数学表現することにしたことが、本研究において、数理モデルの構築に成功した最大の要因となっている。

本研究では、構築したモデルをオープン・ソースのプログラミング言語 (Python 言語) で実装し、コンサルティング先企業 152 社について、営業力や技術力、生産力などの主として経営資源を代表する 8 つの特徴数で評点付けを 5 段階で評点付けをしたデータを用いて、実際に「相互補完強度」及び「相互補完強度係数」を算出した。それにより、提案するモデルの正当性を確認した。

相互補完モデルでは取り扱えなかった営業エリアの異なる企業同士のアライアンスについては加算モデルとして発展させ、また、アライアンスによる事業展開に対する意欲(やる気)については相乗モデルとして発展させた。それらについても、Python 言語で提案するモデルを実装し、その正当性を確認した。これにより、アライアンスの相互補完・加算・相乗モデルを完成させることができ、2 社間のアライアンス成立の関係性を数値として演算可能となった。

1.2 経営学領域に CI のモデル化の手法を導入

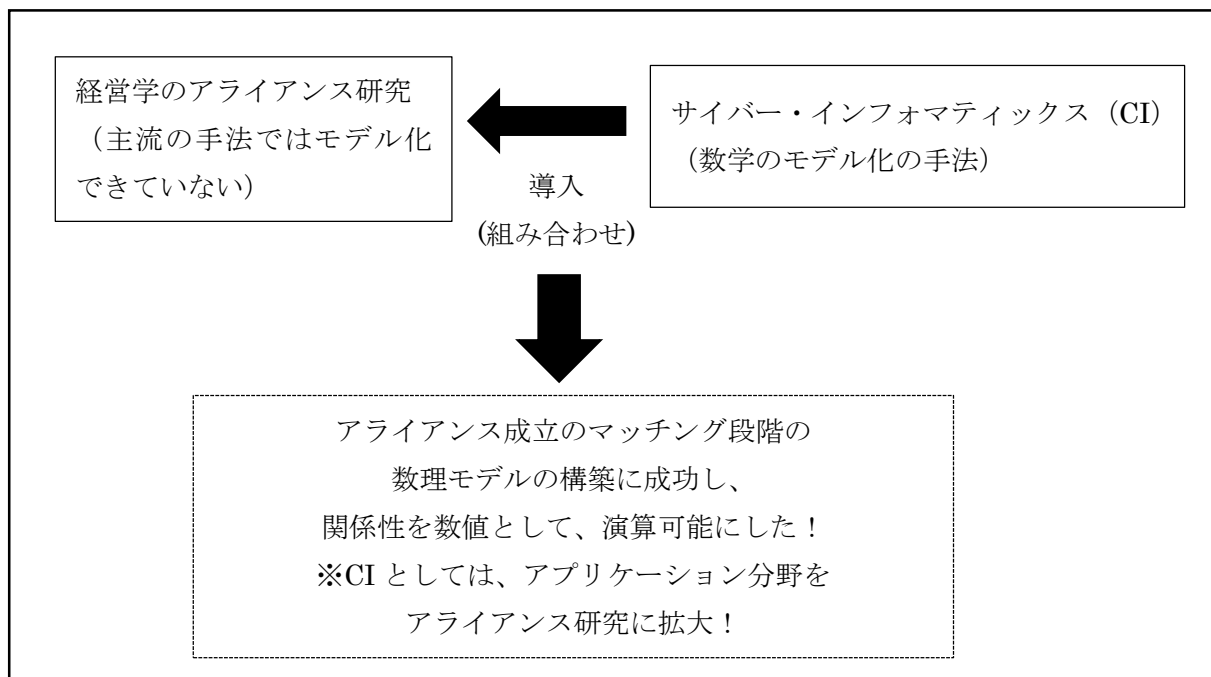
本研究では、経営学の観点からのアライアンス研究だけでなく、数学のモデル化をして、プログラミングによってコンピューティショナルにするコンピュータ・サイエンスの手法を組み合わせ、学際的な研究の形で取り組んだことによって、世界初となる企業間アライアンスの数理モデルの構築に成功した。まさにサイバー・インフォマティクス分野に所属していたからこそ、得られた研究成果であると言える。

経営学では、企業の内部に焦点を当てる資源ベース理論や外部の企業との競争環境などのポジショニングに焦点を当てる競争戦略論などの理論フレームワークで研究が進展してきているが、そういった経営学で主流の手法では、企業間アライアンスのマッチング段階のメカニズムはモデル化しにくい領域であった。そのため、経営学分野の研究者では、数理モデル化が実現できていなかった。

また、経営学においては、多変量解析による計量分析の研究が非常に多くなされており、現在の主流となっており、経営学の世界では統計処理の研究に終始している傾向が見られる。そのため、数理モデル化への取り組みが手つかずの状態となっていた。

ただし、経営学においても、数学的な分析手法であるゲーム理論やリアル・オプションなどの研究手法によって、経営判断を数学表現し、数値として把握して、意思決定をしていこうという流れにあり、本研究も、そういった数理的なアプローチの流れを発展させるものであると位置づけられる。

図1 本研究の意義と学際研究を示す概略図



このことは、翻って考えてみれば、サイバー・インフォマティクス(CI)としては、そのアプリケーション分野を、企業間アライアンスの領域に拡大することにつながっている。本研究は、経営学研究及びCI研究分野の両方において意義ある学際的な研究となっている。

1.3 アライアンスの定義と研究対象とする段階

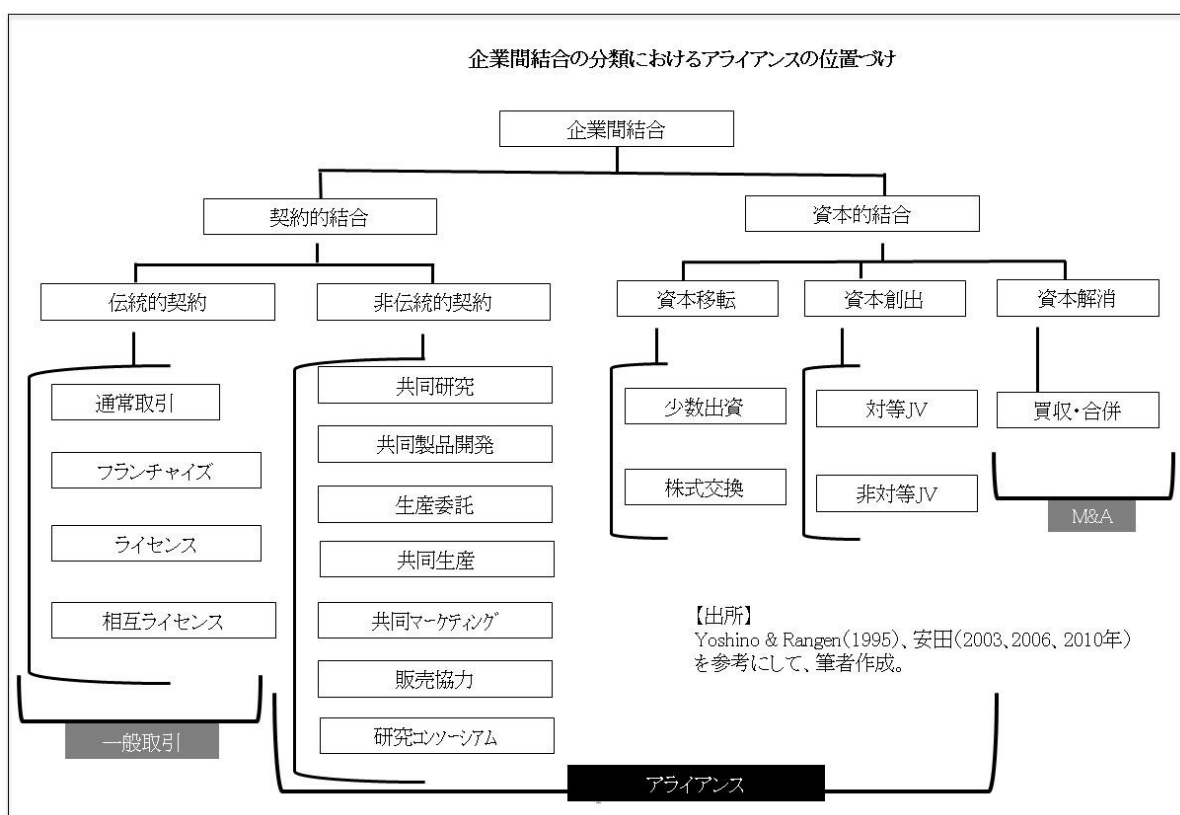
本研究では、企業間アライアンスを取り扱っているが、アライアンスとは何か？について、ここで、定義しておきたい。

アライアンス研究の起点となった文献の一つである Yoshino and Rangan(1995)におけるアライアンスの定義、そして、その研究を発展・整理させた安田(2003、2006、2010)の定義を踏まえ、筆

者は、アライアンスとは「複数の企業が独立したままの状態、新規事業構築や既存事業の拡大のために各企業が持つ経営資源を提供し合って相互補完し、契約の締結や資本関係の有無に関わらず、継続的な協力を行って、その成果を分け合うこと」と定義づける¹。

Yoshino and Rangan(1995)及び安田(2003, 2006, 2010)を参考に、企業間結合の分類、すなわち、M&A、一般取引、アライアンスの分類については、次の図2の通りである。

図2 企業間結合の分類におけるアライアンスの位置づけ



アライアンス研究²では、アライアンスを行う企業の当事者側からのスタンスでの研究として、アライアンスの提携先の選定、条件や目標の設定、協力関係の構築・維持・解消、利益の享受などについての研究がなされている。

本研究では、アライアンスの各過程における最初のマッチングの段階、つまり、不特定多数の

1 本研究においては、第5章で説明するように、152社のデータにおけるアライアンス成立・不成立のカウンタについては、別の異なる定義を用いていることに注意されたい。

2 主として、安田(2003, 2006, 2010)、元橋(2014)、Doz and Gary Hamel(1998)、Tjemkes et al.(2012)などで、アライアンスの各段階の研究の整理がなされている。

企業が存在する中で、企業と企業がアライアンスをすることに意義を見出し、2社の企業がアライアンスの実行に向けて結び付き始める段階の成立メカニズムを明らかにするものである。

先行研究についての詳細は、第2章に譲るが、安田(2003, 2006, 2010)やDoz and Hamel(1998)などのアライアンス研究の主流は、アライアンス先候補企業が見つかった後の段階、あるいは、アライアンスをすることを決めた後の段階に関するものとなっており、多くの潜在的なマッチング候補先が存在する中で、アライアンス先企業として適した企業を選択する段階(すなわち、マッチング段階)の研究が十分なされておらず、アライアンス研究が狭い範囲だけにとどまっている。本研究は、これまでのアライアンス研究では十分取り上げられてこなかった領域であるアライアンス成立の最初のマッチング段階にまでアライアンス研究を拡大し、まだ構築がなされていないマッチング段階の数理モデルを提案するものである。

加えて、これまでのアライアンス研究は、アライアンスをしようとする片方の企業の視点からだけの分析がメインとなっており、多くの潜在的なアライアンス候補先企業の中から、最適なアライアンス先企業を選択し、紹介によって橋渡しをする「アライアンスの仲介者」³の視点が欠けている。本研究は、アライアンスの仲介者がどのような活動を行っているかを解明する研究でもある。

本研究で提案した数理モデルは、フレキシブルなものであり、ユーザーがウエイト付けの変更やパラメーターの数の変更などを自由に行うことができる発展可能なモデルとなっている。今後、他の企業データや異なる評点付け、パラメーターの変更などにより、モデルをより実態に近いように進化させていくことができる。本研究の目的は、そういった研究を進展させていくにあたっての発展可能なモデルをまず提案することにある。

なお、本研究では、筆者の152社のコンサルティング先企業データを用いており、そのデータの内容・詳細は、本稿の第5章に掲載しているが、それらのデータはモデル構築にあたって利用し、その妥当性を確認するために用いたものであり、データ自体は、あくまで一つの例にすぎず、

³ 「アライアンスをしようとする企業に適したアライアンス候補先企業を探索・発掘し、それらのアライアンス候補各社の強み・弱みを評価して、紹介・引き合わせを行い、2社間のアライアンスが成立するようにアレンジメントをする人や企業」と筆者は定義する。

このたびの企業データや評点付けが絶対的なものであることを主張しているわけわけではない。

第2章 先行研究のサーベイ

2.1 アライアンス研究の概観

経済学の研究においては、研究に関連する先行研究の正確かつ網羅的なサーベイが重要であるため、本研究の推進・取りまとめにあたっては、本研究に関連する先行研究のサーベイを行った。

アライアンスについての研究としては、ライバル企業と競合しすぎることを避ける観点からの戦略的提携の理論として、Hamel and Doz(1989)を起点として、Hamel and Prahalad(1994)にて、「アライアンスが必要になる最も明白な理由は、新製品や新サービスを生むのに必要な経営資源をすべて取り揃えている企業など1社たりとて存在しないという事実である」とされており、また、その有用性として、①経営資源を補う、②成功を早める、③パートナー企業同士でリスクを共有し合うことにあると、アライアンスの有用性が主張されている。

その後、Yoshino and Rangan(1995)において、初めて包括的にアライアンスの形態の類型・分類・定義付けがなされている。それについては、図2で示した通りである。

アライアンス研究の概観については安田(2003, 2006, 2010)や、米倉(2012)、中村(2013)、湯川(2013)、石井(2003)、Mitsubishi and Greve(2009)、Martin and Stiefelmeyer(2001)にてまとめられており、アライアンス研究は、経営学の理論の発展の延長戦上とともに、経済学の発展の延長戦上にも位置付けられる。

他の経営学分野と同様に、経済学のミクロ経済学の応用として、企業は必要な資源を獲得するにあたって、外部企業と通常の市場取引をするためにかかる取引コストと、内部化して独自に内製で行う場合に必要となる内部化コストの比較・合算の中で、アライアンスをしたほうが最適である点において、企業間のアライアンスが行われると捉える取引コスト理論や、囚人のジレンマやナッシュ均衡をはじめとした当事者間の関係性を数学的に説明しようとするゲーム理論が、理論フレームワークとして、アライアンス研究においても用いられている。

アライアンス研究における経営学の系譜としては、Wernerfelt(1984)及びBarney(1991)を起点として提示された資源ベース理論(Resource-Based View, RBV)が主要な基礎理論となっている。

本研究においては、資源ベース理論を発展させ、“アライアンスは経営資源の交換である”と結論付けている安田(2003, 2006, 2010)が、最大の先行研究との位置づけとなる。

筆者は、企業の競争優位性の源泉は、企業内部の経営資源に依存しており、また、経営資源の獲得のために、企業はアライアンスを行うと捉えることが適切であると考え、すでにアライアンス研究に用いられている資源ベース理論を主たる理論フレームワークとして選択した。

取引コスト理論は、アライアンスの形成の静観的に現象を捉えているだけであり、ダイナミックに企業と企業がどのように結びつくのかを説明しきれていないため、採用しない。また、ゲーム理論は、アライアンス先企業がすでに見えており、戦略的提携の相手先がいくつかに絞られて特定している段階での企業間の関係性を分析したものとなっていたため、アライアンス候補先を探索する段階での理論背景としては適さないため、採用しない。

2.2 本研究の根拠となる資源ベース理論によるアライアンス研究

前述のように、アライアンスの成立を説明する理論としては、Wernerfelt (1984)及び Barney (1991)で主張されてスタートした資源ベース理論がある。これは、企業が有する経営資源に企業の持続的な競争優位性が依存するという考え方である。企業の持続可能な競争優位を生み出すための4つの実証的な指標として、価値、希少さ、模倣困難性、代替困難性を用いて、分析している。

Das and Teng(1998a)では、外部の競争環境に対応する戦略として、アライアンスを捉え、資金、技術、物質、管理の4つの資源を協調的に調整する観点で、アライアンスの形態やリスク・リターンとの関係について分析されている。

Das and Teng(2000)では、資源の移動困難性(imperfect mobility)、模倣困難性(imitability)、代替可能性(substitutability)といった資源の特徴の観点から、アライアンスのパートナー間の資源連携の調整について、資源の類似性(similarity)や効用(utilization)に着目して、アライアンスの4つのタイプ、すなわち、補填(supplementary)、余剰(surplus)、補完(complementary)、無駄(wasteful)について分析しており、エクイティ・ジョイント・ベンチャー、契約に基づくアライアンスなどについて研究している。

その発展として、安田(2003, 2006, 2010)は、戦略的提携の研究に新しい分析アプローチとして、“アライアンスは経営資源の交換である”という考え方を提示している。すなわち、安田(2003, 2006, 2010)は、企業の有する資源を、(1)技術資源、(2)人材資源、(3)生産資源、(4)販売資源、(5)資金資源という5つの経営資源に、簡略化して分類・整理し、アライアンスとは、それら5つの経営資源の交換であると結論づけている。本研究は、この考え方を発展・応用させている。

その他、資源ベース理論を用いたアライアンス研究としては、Lavie(2006)があり、インターコネクティッド企業のネットワーク資源の研究に応用されており、異質性、移動困難性、模倣困難性、持続可能性について、再度の検証が行われ、ネットワーク化された環境においては、資源の本質よりも、関係性の本質のほうがより重要であると結論づけている。本研究も、資源ベース理論の資源の特性についての検討よりも、2社間の資源をめぐる関係性に注目しており、当該研究を参考にしている。

しかしながら、これらの先行研究においては、想定されるアライアンス候補先企業間の関係性を表現する数理モデルが存在していない。例えば、アライアンスのマッチング段階を取り扱っている研究である Mitsuhashi and Greve(2009)においても、資源ベース理論や、あるいは、社会的交換理論、組織論を手法として用いて、アライアンスの関係性を研究しているが、観念的な議論にとどまっており、マッチングのメカニズムを表現する数理モデルは提示されていない。安田(2003, 2006, 2010)においても、アライアンス・パートナー間の関係性を、経営資源の交換の観点から説明しているが、数理モデルの構築には至っていない。

なお、Das and Teng(1998a)では、企業のアライアンス能力について、物的資源(プロパティ)と知識ベース(アイデアやスキル)に分けて、説明しており、企業組織にアライアンスを受け入れる素地があるかが重要であることが主張されており、これは、本稿で取り上げる相乗モデルでのアライアンスに対する意欲(やる気)に関連している。

2.3 アライアンス研究の他の理論(取引コスト理論、ゲーム理論等)

次に、資源ベース理論以外の理論によるアライアンス研究について、概観する。

先に述べたように、一つ目は、取引コスト理論である。この理論では、一般の市場取引にかか

わる取引コストよりも、アライアンスをしたほうが、相対的にコストが安い時で、かつ、内部化(内製化)をするコストよりも、外部企業とアライアンスをしたほうが、相対的にコストが安い時に、アライアンスが行われると考える。つまり、一般取引における相手先の与信審査や契約書締結、それらの交渉などといった取引コストと、事業運営において必要とする機能を企業内部に保有するための内部化コストの2つのコストが最低となる中間のゾーンにおいて、アライアンスが形成されると説明する理論である。しかし、この理論は、企業行動としてのアライアンスの現象を表層的かつ鳥瞰的に説明することはできているものの、どのようにアライアンスにおける最適な提携先企業(ビジネス・パートナー)を選択するのかという説明付けはできず、アライアンスにおけるマッチングを説明するものになっていない。そのため、本研究の理論的フレームワークとしては採用しない。

もう一つ、アライアンス研究で用いられている理論としては、ゲーム理論がある。ゲーム理論は、すでにアライアンスが形成された後のアライアンスをしている企業間の関係性や、あるいは、アライアンスを行う候補先企業の数固定されている時の関係性を検証することができる。アライアンスにおける相手先パートナー企業がわかっている時の関係性を説明する際には適しているが、本研究で取り扱っているような、企業が不特定多数の企業群の中からアライアンス先企業候補を探索し、最適なアライアンス先企業を選択するといったアライアンスのマッチングの段階における理論的な基礎としては適しない。本稿では、アライアンスのマッチング段階において、どの企業とどの企業が最適な組み合わせかを決定する研究であるため、ゲーム理論は理論的フレームワークとして採用しない。

その他、社会的交換理論を用いたアライアンス研究として Das and Teng(2002)があり、企業間のアライアンスにおける非経済的な資源の交換について研究している。アライアンス・コンステレーションと呼ばれる、企業のグループ化について検討されており、独立している企業が、個々の企業のゴールに適合するためだけでなく、アライアンスをしているすべての企業に共通したメリットのためにも、ユニファイド・ネットワークを構築するとしている。企業間が協力するによって得られるメリットが、アライアンス・コンステレーションの研究としてまとめられている。しかしながら、本研究では、アライアンスにおける経済的な恩恵を研究しているため、経済的な

資源の交換に主に関心を置いているため、本研究の理論的な基礎としては、Das and Teng (2002) で用いられているような文脈での社会的交換理論はメインの理論とはならない。ただし、本研究の8つの強み・弱みの特徴の一つとして、信用・ブランドを評価項目として入れており、これについては、非経済的なものとして、社会的交換理論に該当する。また、Das and Teng(1998b)においては、アライアンスにおける信頼関係の醸成における信用のコントロールについて、研究されており、信用の大切さが述べられており、これが、後述する各企業の評点付けの8つの特徴として信用・ブランドを盛り込んだ理由となっている。

表1 本研究に関連する主たる先行研究

本研究に関連する主たる先行研究の一覧表	
1	アライアンス研究全体を包括している文献
	Yoshino and Rangan(1995) 企業間結合の分類
	安田(2003, 2006, 2010) アライアンスは経営資源の交換
	石井(2003) 組織間調整、取引コスト理論
	牛丸(2007) 競争、不確実性、ゲーム理論
2	アライアンスのマネジメントに関する文献
	Doz and Hamel(1998) 価値創造のアライアンス戦略の実施
	Barnford, Gomes-Casseres, and Robinson(2003) アライアンス戦略の各段階の実行
	安田(2016) パートナー間多様性の影響
	元橋(2014) アライアンス・マネジメント全般
3	資源ベース理論自体のメイン文献
	Wernerfelt(1984) 最初にRBVという用語を使用
	Barney(1991) RBVのスタートとなった文献
	Collis and Montgomery (1998) RBVの経営戦略全般
	Verona(1999) RBVによる製品開発の研究
4	資源ベース理論のアライアンス研究
	Das and Teng(1998a) 4つの資源の調整、リスク・リターン
	Das and Teng(2000) 移動困難性、模倣困難性、4つのタイプ
	安田(2003, 2006, 2010) アライアンスは経営資源の交換
	Lavie(2006) インターコネクティッド企業の競争優位
5	小企業についてのアライアンス研究
	Gomes-Casseres(1997) 小規模な企業が研究対象
6	マッチング段階についての研究
	Mitsuhashi and Greve(2009) マッチング段階の研究、数理モデルなし
7	相互補完性についての研究
	Adegbesan(2009) 不足するものを補完、数理モデルなし
8	社会的交換理論のアライアンス研究
	Das and Teng(1998b) トラストとコントロール、信頼関係構築
	Das and Teng(2002) コンステレーション、束になることのメリット
9	ギブ・アンド・テイクの社会的交換理論
	Scbaufeli(2006) 人間関係のギブ・アンド・テイク、バーン・アウト
	Kaysar, Converse, Wang, and Epley(2008) 人間関係のギブ・アンド・テイク、非互惠性
	Gassmann and Enkel(2004) オープン・イノベーション、ギブ・アンド・テイク
10	オープン・イノベーションに関する研究
	Chesbrough(2003, 2006a, 2006b) オープン・イノベーションの提唱
	米倉・清水(2015) オープン・イノベーションのプロセスと事例
	米倉(2012) オープン・イノベーションと戦略的提携
11	コーポレート・ベンチャリングに関する研究
	湯川(2013) コーポレート・ベンチャリング、ICT企業
	中村(2013) 大企業とベンチャー企業の提携
12	マーケット・デザイン(マッチメイク)の研究
	Akbarpour, Li, and Gharan(2014) マーケット・デザインの論文の一例
	Roth(2015) 2012年ノーベル経済学賞・受賞者の著書
	坂井(2010) 臓器、結婚などのマッチングの数理モデル
13	データセット作成の研究
	DeVellis (2012) スケール・デベロップメントの理論と応用

2.4 多くの計量分析と数理モデルの欠如

これまでのアライアンス研究では、アライアンスによる業績向上や研究開発促進、アライアンス先からのスキル習得などについての多変量解析を用いた研究は多数存在する。しかし、それらは、アライアンスの成果のパフォーマンスを売上高や特許出願件数、市場シェア、企業価値、マネジメント能力等をパラメーターとして、統計手法によって実証分析をしているものの、アライアンスが成立するメカニズムを数学的に表現するモデルは提示していない。

アライアンス研究における定量的な研究は多数存在しているが、それらは、統計手法によるものであるため、アライアンスが成立するか不成立となるかのメカニズムを表現する数理モデルではない。数理モデルが存在しないため、アライアンスにおける 2 社間の関係性を数値として算出することができない状況にある。

アライアンスの統計手法による計量分析としては、たとえば、アライアンスによる企業価値向上については、Anand and Khanna(2000)、Chang, Chen and Lai(2008)、アライアンスによるイノベーション創出力や製品開発力の強化については、Kelley and Rice(2002)、Lee and Cavusgil(2006)、アライアンスによる売上拡大については、Leiblein and Reuer(2004)、アライアンスによるマネジメント能力の向上については、Rothaermel and Deeds(2006)、アライアンスのパートナーの多様性に関しては、安田(2015)がある。このように、統計分析の研究は、多数存在する。

Mitsuhashi and Greve(2009)では、アライアンスのマッチング段階の研究が行われ、特に、海運業界におけるパートナー選択について検討されているが、観念的な議論となっており、数理モデルが全く提示されていない。アライアンスにおける相互補完関係が数学的に示されていないため、観念的にはアライアンスの相互補完性を捉えることはできるものの、アライアンスの相互補完メカニズムを厳密に把握して操作性を与えたり、数値として捉えたりすることができない。

このように、アライアンス研究においては、計量分析の研究は存在するものの、アライアンスが成立する際の企業と企業の結びつきがどのようなメカニズムで成立しているのかを表す数理モデルが存在していないため、アライアンスを行う 2 社の組み合わせの関係性を数値で算出することができていない。

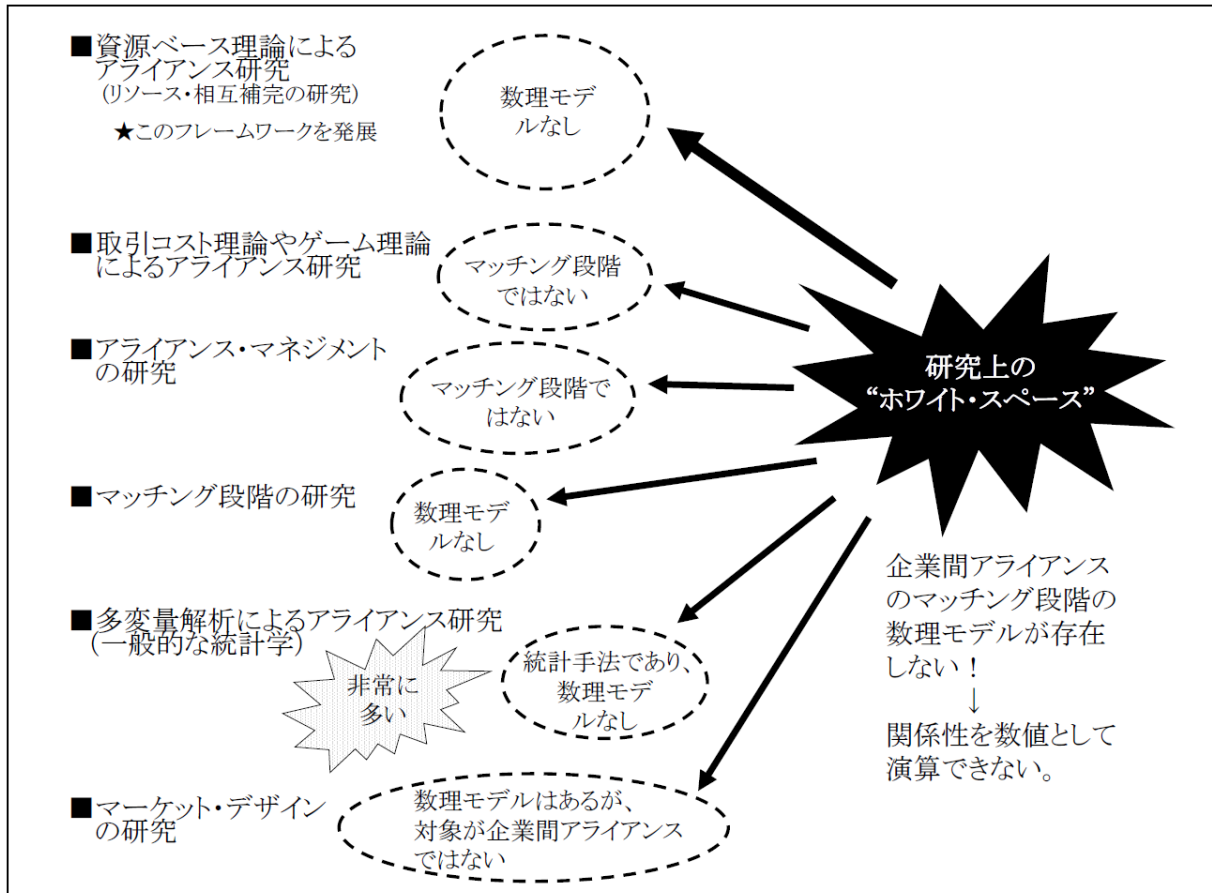
相互補完性については、一部、先行研究が存在している。たとえば、Adegbesan(2009)において

は、企業の強みをより強くするための企業間で補い合うことや、企業がグループになることの価格交渉力についてなど、相互補完性についての議論がなされているが、ゲーム理論による双方の企業の関係性の検討のための数理モデルは提示されているが、相手先がまだ定まっていない段階のマッチングの場面での数理モデルが提示されていない。

Roth(2015)で取り上げられているような臓器や結婚などの一般市場では取り扱いにくいものに関するマーケット・デザインの分野ではマッチングの数理モデルは存在している。マーケット・デザインの分野では、Akbarpour, Li, and Ghara(2014)や坂井(2010)のように、数理モデルが研究されており、マッチ・メイクの数学モデルとしては、マーケット・デザインという一つの研究分野として確立している。しかし、それらは、企業間アライアンスを対象としていないため、やはり、企業間アライアンスのマッチング段階の数理モデルは存在していない状況にある。

次に示す図3は、ここまで述べてきた先行研究のサーベイの状況を図としてまとめたものであり、企業間アライアンスのマッチング段階のメカニズムに関連する先行研究について、関連研究の位置づけを示し、その中で、研究されていない領域、すなわち、“ホワイト・スペース”を示している。

図3 先行研究における“ホワイト・スペース”



2.5 アライアンス・マネジメントの研究

アライアンス研究においては、アライアンスの相手先企業が見えている段階での契約条項の設定やアライアンスの形態、アライアンス後の段階からの関係調整、アライアンスの終了、アライアンスの成果評価といった部分が研究の中心となっており、不特定多数の企業群の中から適切なアライアンス先企業を選定するという最初のマッチングの段階の研究が十分行われていない。

これまでのアライアンスの研究は、アライアンス成立後の関係調整、契約内容の設定や契約の解消、成果の評価などが中心となっている。代表的なものとしては、安田(2003, 2006, 2010, 2015)、元橋(2014)、Doz and Hamel(1998)、Bamford et al. (2003)、Tjemkes et al. (2012)である。これらの研究では、アライアンスの相手先企業が見えた後のアライアンスの形態、契約条項の決め方、協業の仕方、アライアンスの管理などが研究対象となっている。

ハーバード・ビジネス・レビューに掲載されるような実務に近い研究においても、アライアンスやジョイントベンチャーの成否を分ける要因については Bamford, et al. (2004)、相手先企業が見えた段階のアライアンスか買収かの分析については Dyer, et al. (2004)、アライアンスのリスクについては Ernst and James Bamford (2005)、スキルと知識の獲得による競争力向上については Hamel and Doz (1989) アライアンスの管理については Kaplan, Robert S., David P. Norton, and Bjarne Rugelsjoen (2009) などがあり、いずれも、アライアンスをする相手先企業が見えている上での関係性の調整についての研究となっている。

本研究では、それらのアライアンスの相手先が見えている段階やアライアンスの実施後ではなく、最初のマッチングの部分、つまり、不特定多数の企業が存在する中で、企業と企業がアライアンスをすることに意義を見つけ出し、2社の企業がアライアンスの実行に向けて結び付き始める段階の成立メカニズムを明らかにすることを研究のメイン対象としている。

アライアンスの最初のマッチングの段階を取り上げたものとしては、前述の通り、Mitsubishi and Greve (2009) が先行研究として存在するが、観念的な議論だけにとどまっており、マッチングの成立の数理モデルは提示されていない。

Gompers and Lerner (2000) において、プリンシパル・エージェント理論を用いて、ベンチャーキャピタルと投資先企業の関係、ベンチャーキャピタルとファンド出資者の関係が研究されており、ベンチャーキャピタルが投資先企業の研究がなされているが、アライアンスを行う企業の当事者の判断について研究がなされているが、本稿では、片方側の企業だけでなく、2社の企業のアライアンスのアレンジメントをする仲介者がどのようなメカニズムで、組み合わせを行っているかを明らかにすることも、本研究の目的の一つである。

本研究での検証データの内訳としては、小規模な企業が多く含まれている。研究対象とする企業の規模の面では、Gomes-Casseres (1997) が小規模な企業のアライアンスについて研究しており、先行研究となるが、その論文でも、本稿で提案するような数理モデルは提案されていない。

2.6 オープン・イノベーション研究との関係

本研究のアライアンス研究の中での位置づけは、上述の通りであるが、本研究は、加えて、

Chesbrough(2003, 2006a, 2006b)で提唱されたオープン・イノベーション(自前主義で、自社内での研究開発で生み出されたものだけを経営資源として利用するのではなく、外部企業が生み出した技術などを取り込んで利用していく考え方)の研究の系譜・文脈⁴の中でも捉えることができる。清水(2012)及び米倉・清水(2015)で述べられているような、不特定多数の企業の中から、自社のアライアンス先として適合する企業を探索する際、とりわけ、企業と企業が出会って、アライアンスに向けて話し合いをし始めるマッチングの段階に焦点を当てている。この文脈における探索コスト⁵を低減させるための理論的なモデルとなる。

知の探索(エクスプロレーション、Exploration)と、知の進化(エクスプリテーション、Exploitation)に大きく分けられ、March(1991)で研究されている。

米倉・清水(2015)のオープン・イノベーションの研究においても、探索コストや「エージェント活用型探索⁶」について述べられているが、マッチングが成立するメカニズムについての数理モデルは示されていない。

米倉・清水(2015)では、中小企業が新しい事業展開をするにあたって、経営資源の制約が大きいことが述べられているが、本研究においては、中小企業も企業データの中に含まれており、中小企業のオープン・イノベーションに関する研究とも呼べる。

このように、本研究は、アライアンス研究における資源ベース理論の研究である Das and Teng(1998a, 2000)及び安田(2006, 2010)を発展させた研究であるとともに、上述のように、オープン・イノベーション研究の発展上にあるという捉え方もできる。

アライアンスが形成される最初のマッチングの段階、すなわち、相手先企業を探索している段階に研究対象を広げるだけでなく、そのマッチングの成立確率が高まるように、翻って言えば、アライアンス先の探索コストを低減させることの寄与するものと考える。

⁴ オープン・イノベーションの研究における清水(2012)や米倉・清水(2015)、星野達也(2015a, 2015b)でも、同様のことが主張されている。

⁵ アライアンスをする相手先企業を探し出すためにかかる費用と筆者は定義する。

⁶ コンサルティング会社などの代理人に、アライアンスの提携先の探索を依頼する形のことと、米倉・清水(2015)において紹介・解説されている。米国オハイオ州に本社を置くナインシグマが代表例とされ、自社が有する既存ネットワーク以外に素早くアクセスできることやエージェントが相手先をスクリーニングするため、信頼性が担保されることなどがメリットとして挙げられている。

2.7 人間関係のギブ・アンド・テイクに関する社会的交換理論

もう一つの理論は、社会的交換理論として知られるものである。これは、経済的な側面よりも、非経済的な側面に注目している。本研究では、主として、経済的な経営資源の交換にフォーカスを当てているので、その面では、前述の通り、本研究のメインの理論フレームワークとはならない。

しかし、社会的交換理論には、人間同士、あるいは、人と企業の間の方の研究におけるギブ・アンド・テイクについて分析し、説明付けをしている研究がある。

Schaufeli (2006)による心理学分野の学術的な論文などの広範囲にわたる研究の結果、8000人以上の専門家(教師、医者、看護師、警察官、囚人、警護官、ソーシャルワーカー、精神障害労働者)について調査しており、人間関係や人と企業の関係において、ギブ・アンド・テイクのバランスが崩れると、バーンアウトを引き起こすことが示されている。関係において、バランスが大きく崩れていたり、あるいは、ミスマッチが起こったりしている場合には、ギブ・アンド・テイクのバランスが崩れており、さもなくば、ある集団と他の集団にとって、偏ったいびつな不均衡な状態となっており、関係性が崩壊し、それは、バーンアウトと呼ばれている。その論文では、報酬とコストのギャップが大きくなりすぎて、インバランスとなっている時にバーンアウトが引き起こされるとしている。

ギブ・アンド・テイクに関する近い領域の先行研究としては、Chesbrough(2003)によってスタートしたオープン・イノベーションの研究において、企業間のギブ・アンド・テイクが非常に重要であると述べられている。例えば、Grassmann and Enkel(2004)は、124社の企業データを用いた実証研究を行っており、成功のために、ギブ・アンド・テイクが極めて重要であることを示す3つのコア・プロセス・アーキタイプを用いて、オープン・イノベーションの研究となっている。

これにより、個人の間関係において研究されてきたギブ・アンド・テイクの社会的交換理論の考え方を、アライアンスの数理モデルの構築の本研究に取り込むこととした。

第3章 アライアンスの相互補完モデルの構築

3.1 相互補完モデル構築の前提としたアライアンス成立の考え方

企業間のアライアンスの成立においては、資源ベース理論を応用した研究はあるものの、企業間のアライアンスのマッチングを表現する数理モデルが存在していない。それゆえ、アライアンスにおける2社の関係性を数値として算出することができていない。本研究では、資源ベース理論によるアライアンス研究のフレームワークを発展させ、「アライアンスは経営資源の交換である」と考える。

各企業が自社の強みを提供して、自社の弱みを補うのが、アライアンスであり、2社間の強みと弱みの相互補完がより強い時に、アライアンスは成立しやすいと考える。

たとえ過当競争を避けスケール・メリットを求めるための競合企業同士による戦略的提携(Strategic Alliance)⁷であっても、Tjemkes et al. (2012)で示されているように、それぞれ競争優位性のある提携先パートナーを見つけ出し、各社の弱みを他社の強みで補完することを意図している。

本研究では、アライアンスの成立における強みと弱みの相互補完関係が強ければ強いほど、アライアンスは成立しやすく、相互補完関係が弱ければ弱いほど、アライアンスは成立しにくいと考えた。2社間において、どちらか片方からだけしか強みの提供が行われない場合には、アライアンスは成立しないと仮定しており、アライアンスにおいては、両方の企業から強みが提供され、相手企業の弱みを補完していることが成立の前提となると考える。

なお、富田(2014)で示されているように、大手企業と中小ベンチャー企業の企業規模・企業属性が異なる企業間でのアライアンス⁸の組み合わせやその有用性、新製品開発における技術面での補完といった観点からもアライアンスを見ることもできる⁹。しかし、本稿においては、アライア

⁷ 新日鉄住金(新日本製鐵と住友金属の提携及び合併)や三井住友信託銀行(中央三井信託銀行と住友信託銀行の提携及び合併)、ルネサスエレクトロニクス(三菱電機、日立、NECの半導体事業部が統合)が、この事例となる。

⁸ 大企業とベンチャー企業のアライアンスについては、コーポレート・ベンチャリングの研究である中村(2013)及び湯川(2013)にて、ICT産業での事例を中心に検討・整理がなされている。

⁹ 詳しくは、富田賢(2014)を参照。

ンスの成否に与える企業規模や業歴の影響について考察していない。

3.2 相互補完の捉え方と物理モデルの応用

アライアンス成立の相互補完のメカニズムを示す数理モデルを構築するにあたり、物理学の分野での磁力のスピングラス・モデル(N極とS極が引き合う理論のこと)を応用させて、考案した。

アライアンスの相互補完モデルの考え方としては、ある企業A社の弱みを別の企業B社の強みで補完し、逆に、B社の弱みをA社の強みで補完することが基本となり、片方の企業だけの補完や、両方の補完が小さい場合は、相互補完の強度は小さいと考え、相互に強みや弱みを、強く補完し合っている場合には、強い相互補完だと捉える。つまり、アライアンスの相互補完は、2社間の互いに引き合う力に基づく2極モデルであると考えられる。

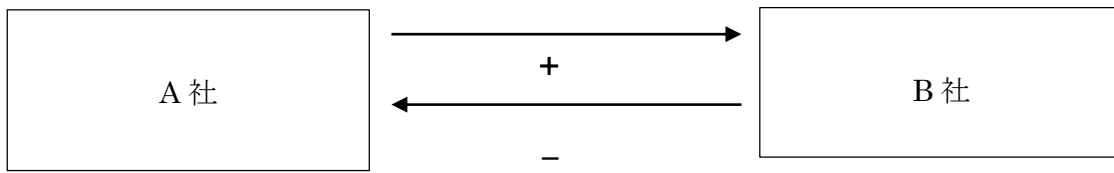
アライアンス成立の相互補完のメカニズムを示す数理モデルを構築するにあたり、まず、物理学の磁石のN極とS極が互いに引き合う力を示すスピングラス・モデルの理論を参考にした。

アライアンスの相互補完モデルの考え方としては、ある企業A社の弱みを別の企業B社の強みで補完し、逆に、B社の弱みをA社の強みで補完することが基本となり、片方の企業だけの補完や、両方の補完が小さい場合は、相互補完の強度は小さいと考え、相互に強みや弱みを、強く補完し合っている場合には、強い相互補完だと捉える。つまり、アライアンスの相互補完は、2社間の互いに引き合う力に基づく2極モデルであると考えられる。

相互補完モデルにおいては、A社の強みがB社の弱みを補完し、そのまた逆も同じであり、B社の強みがA社の弱みを補完するということが基礎となる。片方の企業から、あるいは、両方の企業からの補完が小さい時、相互補完強度は小さい。反対に、片方の企業から、あるいは、両方の企業からの補完が大きい時、相互補完強度は大きい。つまり、アライアンスにおける相互補完関係は、2社のお互いに引き合う力に基づく2極のモデルである。

そのイメージ図は、図4の通りである。

図4 アライアンスにおける2社間の双方向のアピール



3.3 強みと弱みの8つの特徴の選定

相互補完数理モデルの構築においては、富田・武藤(2015)では当初、より単純化して、強みと弱みの特徴付けを4つの特徴数にて、モデルを構築しており、本稿でも、付録1として掲載している。富田・武藤(2015)及びTomita and Takefuji(2016a, 2016b)では、強み・弱みの8つの特徴での相互補完関係の評点付けを用いている。本稿では、最初から、8つの特徴数での説明を行う。

それらの8つの特徴数評点付けの項目の選定については、本研究のメインの先行研究として位置付けている安田(2003、2006、2010)で取りまとめられている5つの経営資源、すなわち、販売資源(=営業力)、技術資源(=技術力)、資金資源(=資金力)、人材資源(=人材力)、生産資源(=生産力)を評点付けする特徴として取り上げた。

その5つに加え、Das and Teng(1998b)にて、信用の重要性が述べられているため、信用・ブランドを追加しており、また、アライアンスは新規事業立ち上げを目的として行われることが多いため、事業シーズとなるアイデアを生み出す力、すなわち、アイデア力も追加した。さらには、Hamel and Prahalad(1994)で述べられているように、アライアンスは事業展開のスピードアップを目的の一つとしているため、組織としての機動性を追加した。これら3つを加えて、合計8つの特徴による強み・弱みの評価を行っている。このことは、152社のアライアンスのアレンジメントを行った実務的な経験からも妥当であると考えている。

なお、ここで提案するモデルはフレキシブルなモデルであるため、特徴の数や項目も、ユーザーが自由に決めることができる。前述の通り、富田・武藤(2015)での最初のモデル構築を4つの特徴数で行っていることからわかるように、特徴数は自由に設定できるため、8つという数自

体は本質的な意味は持たない。なお、特徴数を偶数ではなく、奇数となる場合についての考察は、付録3にて、掲載している。

5段階での評点付けとした理由は、日本に限らず、米国をはじめとする諸外国においても、学校の評価方法として、5段階が用いられており、なじみ深いものであるため、採用した。

3.4 企業の強み・弱みの1次元行列と2極ベクトルでの表現

企業の強みと弱みを8つの特徴数で、5段階の評点付け(5が一番良い評価)すると、それらは、1次元行列で数学的に表現できる。

それらの2社の相互補完関係は、1次元行列で表現できる。A社とB社の強み・弱みを8つの項目で、1から5の評点を付けると、次のように表現できる。

$$\text{A社} \quad a = (1, 3, 4, 2, 5, 1, 3, 1)$$

$$\text{B社} \quad b = (4, 1, 1, 3, 1, 5, 3, 1)$$

上記において、A社からB社を引いた結果のcは、各特徴において0~4(正か負)の間の数値で、プラスかマイナスの方向性のある2極のベクトルとなる。すなわち、2社間の相互補完関係は、2極のベクトルで数学的に表現できる。

$$\text{A社} - \text{B社}$$

$$c = a - b = (-3, 2, 3, -1, 4, -4, 0, 0)$$

企業の強みと弱みを1次元行列と、2極のベクトルで数学表現するところが、提案する相互補完数理モデルの基本となる。

3.5 正の整数の総和と負の整数の総和

先に述べたA社とB社の例において、A社からB社を引いた際のプラスの2極のベクトルを示

す正の整数の総和は、A社からB社への強みの提供を示しており、次のようになる。

$$2 + 3 + 4 = 9$$

この9という数字は、A社の強みがB社の弱みを補完している度合いを表す数値となる。

反対に、マイナスの2極のベクトルを示す負の整数の総和は、B社からA社への強みの提供をしめしており、次のようになる。

$$-3 + (-1) + (-4) = -8$$

この-8という数字は、B社の強みがA社の弱みを補完している度合いを表す数値となる。

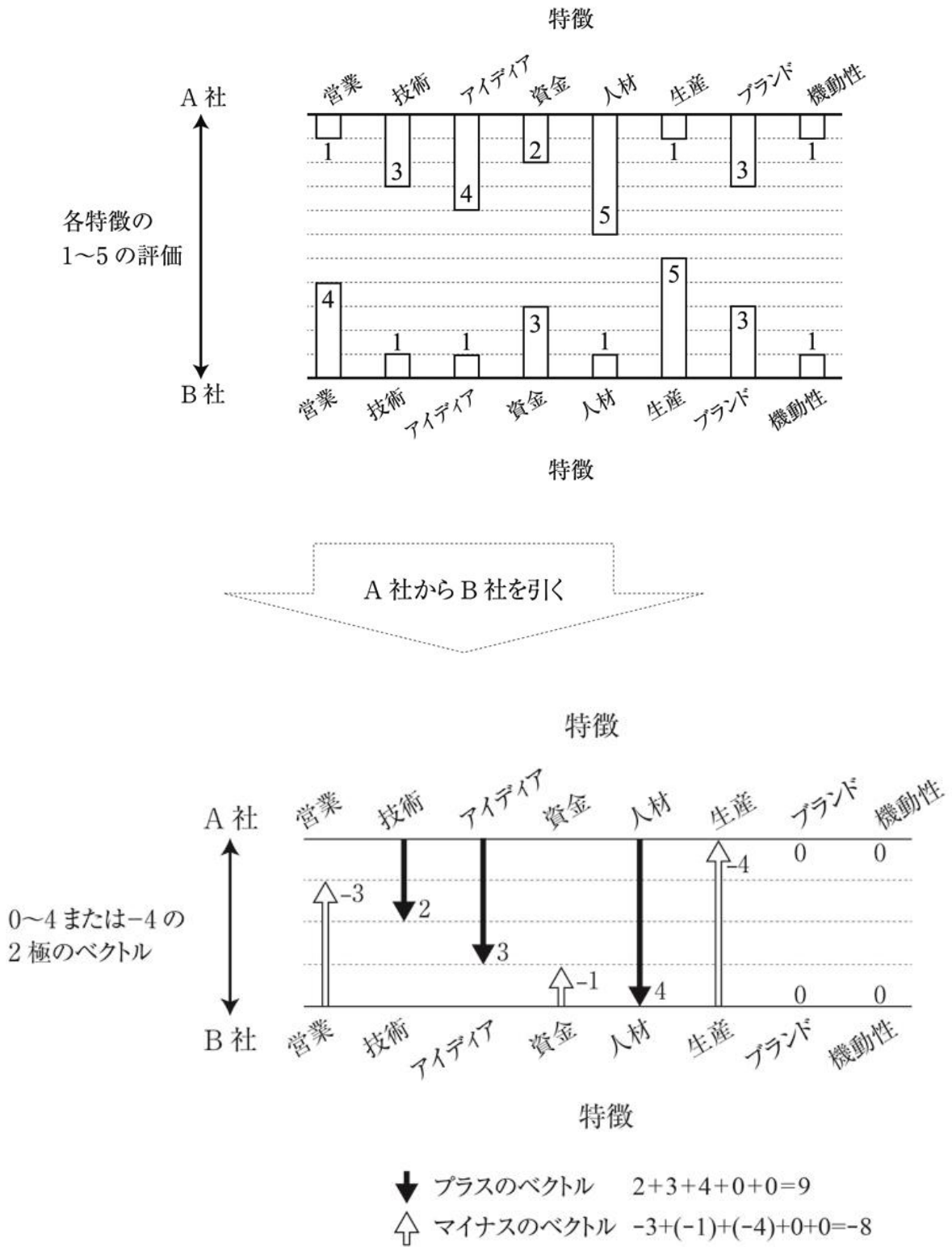
この2つの数字のペア(9, -8)は、A社とB社の相互補完関係を表す数値となる。

このように、1次元行列で各社の強みと弱みを表現した後、演算(引き算)をした結果の正の整数(プラスのベクトル)の総和と、負の整数(マイナスのベクトル)の総和の2つの数字が2社間の相互補完関係を示す数字のペアとなる。

8つの特徴数で、5段階評価の場合のプラスとマイナスのベクトルの出現と強みと弱みの相互補完を示すプラスとマイナスの数字の導出は、図で示すと、次の図5のようになる。

図5 特徴数8つでの5段階評価の相互補完関係の例

(プラスのベクトルの総和とマイナスのベクトルの総和)



3.6 最大の強さの相互補完関係の状態

ここで、2社間の最大の相互補完関係となる状態とはどのような状態かを考えてみよう。

まず、相互補完関係が強いということは、強みと弱みの各特徴において、A社とB社の評点の差、つまり、プラスまたはマイナスの2極のベクトルの値が大きい時となる。5段階評価での評点付けの場合、最大の長さのベクトルは4または-4となるため、最大の相互補完関係となる状態は、ベクトルの長さが4または-4の時となる。

次に、8つの特徴数(強み・弱みの評点を付ける項目の数)の場合、その半分の4つずつで、双方向に、最大の長さのベクトルが存在する時が、最も強い相互補完関係となっている状態である。その状態が、先に述べた正の整数(プラスのベクトル)の総和、並びに、負の整数(マイナスのベクトル)の総和が最も大きな数字となる。

そのように考えると、8つの特徴数において、5段階評価の場合、最大の相互補完関係の状態となる2社の組み合わせは、8つの特徴数の半分の4つの特徴において、最も大きい数値である4または-4の長さのベクトルが双方向で存在する時となる。

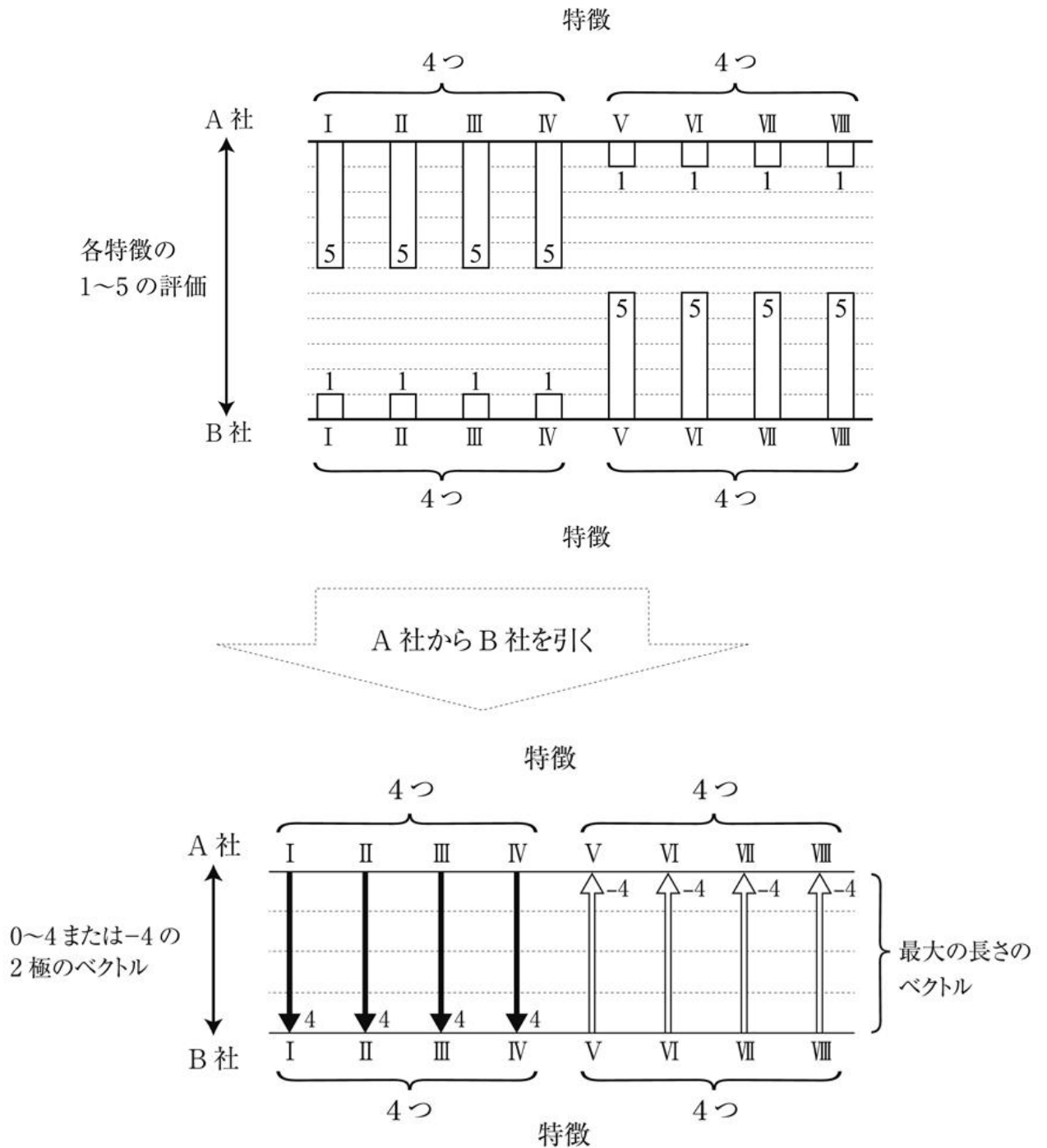
その相互補完関係のプラスとマイナスの整数(2極のベクトル)の組み合わせを求める式は、次の式となる。

$$(\text{特徴数 } 8/2) \times \text{最大の長さ } 4 \text{ または } -4 = (16, -16)$$

この2つの数値(16, -16)が、最大の強さの相互補完関係、すなわち、最大の相互補完強度を示す点となる。

図で表すと次の図6の通りである。

図6 最大の強さの相互補完関係の状態



3.7 強度をマキシマム・ポイントからの距離で数学的に表現

2次元のマップにおいて、最大の相互補完強度を示す点(「マキシマム・ポイント」と名付ける)をプロットすると、図7のようになる。

前述のA社とB社の相互補完関係を示す2つの数値の組み合わせ(9, -8)も、この2次元のマッ

プ上にプロットしてある。各社の特徴付けを1次元行列で表し、差し引きして2極のベクトルで表現し、正の整数(プラスのベクトル)の総和と負の整数(マイナスのベクトル)の総和で表すことにより、あらゆる2社間の相互補完関係は、このように、2次元のマップ上の点で示すことができる。

ここで、本研究で提案するモデルでは、2社間の相互補完関係の強さを、最も強い相互補完強度の点からの距離で、数学的に表現することを考案した。相互補完関係の強さを数学的にどのように表すかを検討した結果、最大の相互補完強度の点からの距離で、2社の企業間の相互補完関係の強さを表現できることに気がついた。このことが、本研究にて、相互補完数理モデルを構築することに成功した最大の要因である。

図7で示すように、A社とB社の強みと弱みの相互補完関係の強さは、その相互補完関係を示す(9, -8)の点と、最大の相互補完関係を示す(16, -16)の点の距離で、数学的に表現される。

図7 アライアンスの2極の相互補完モデル

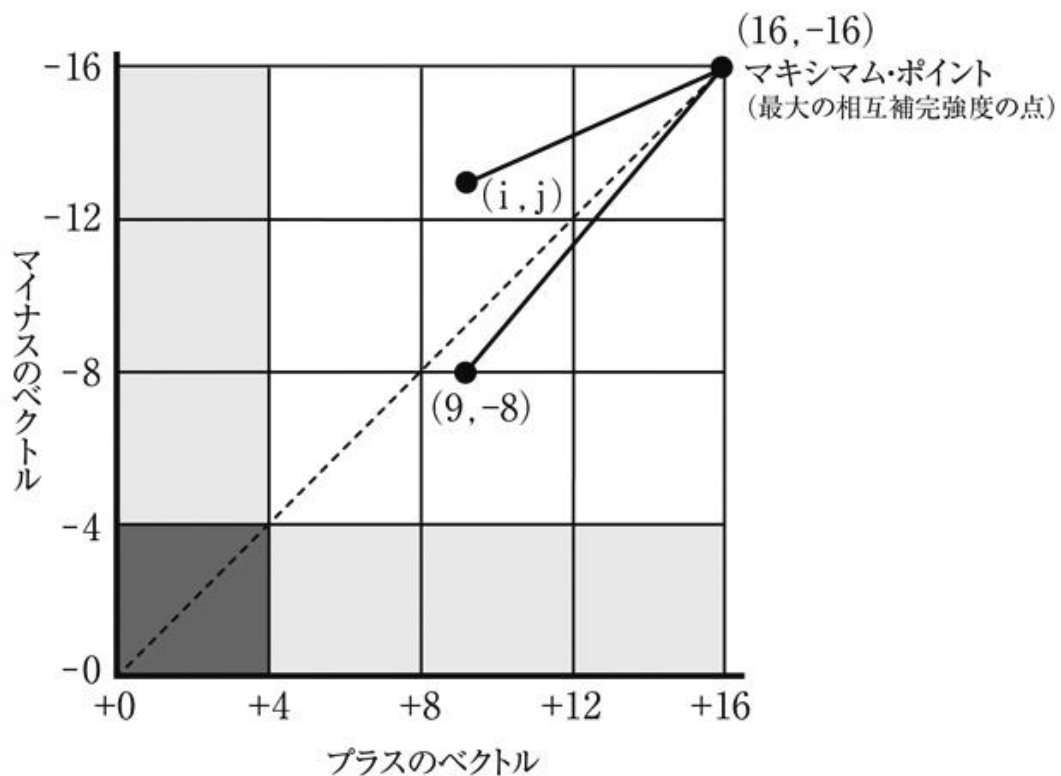


図 7 において、(16, -16) の点が、最大の相互補完強度を示す点であるため、その点に距離的に近ければ近いほど、相互補完強度が強いということとなる。

そこで、この 2 点間の距離を求めることとなるが、数学の一般式として、2 つの点の間の距離 d は、次の式で計算できる。

$$d = \sqrt{(a_1 - a_2)^2 + (b_1 - b_2)^2}$$

最大の点(16, -16)の相互補完強度は、(0, -0)から(16, -16)の距離となるため、次の式で求められる。

$$\sqrt{(16 - 0)^2 + (-16 + 0)^2} = 22.63$$

最大の相互補完強度は、22.63 となるため、特徴数 8 つで、5 段階評点の場合の 2 社間の相互補完強度は、0~22.63 の間の数値となり、数字が大きくなると相互補完強度が強いことを示す。この数値が最大の時、相互補完強度が最大となる。

先に示した A 社と B 社の例の(9, -8)の相互補完強度は、どのようになるであろうか。上記の式に当てはめてみると、A 社と B 社の相互補完関係を示す(9, -8)の点と、最大の相互補完関係を示す(16, -16)の点との距離は、次の式で算出できる。

$$\sqrt{(16 - 9)^2 + (-16 - (-8))^2} = 10.63$$

これにより、第一段階として、A 社と B 社の相互補完強度を数値として表すことができた。ただし、この計算式では、最大の点からの距離が小さければ小さいほど、相互補完強度が強いということになり、数値が小さければ小さいほど、相互補完強度が強いという表現となってしまう。それでは、指標としてわかりにくいいため、第一段階として求められた 2 点間の距離の数値を最大の相互補完強度の数値から差し引くことで、数値が大きければ大きいほど、相互補完関係が強いと

の表現となるように、インバートする。

A 社と B 社の場合の相互補完強度の計算は、(9, -8)の点と(16, -16)の点との距離を求め、それを最大の相互補完の数値から引く形となり、次のようになる。

$$\sqrt{(16-0)^2 + (-16+0)^2} - \sqrt{(16-9)^2 + (-16-(-8))^2} = 11.99$$

これにより、A 社と B 社の相互補完強度が求めることができた。

2 社間の相互補完関係を示す 2 つの数値を一般化して (i, j)とすると、相互補完強度 AS(i, j)は、次の式となる。この(i, j)も、図 7 でプロットしてある。

$$AS(i, j) = \sqrt{(16-0)^2 + (-16+0)^2} - \sqrt{(16-i)^2 + (-16-j)^2}$$

このように、2 社間の相互補完関係を示す(i, j)は、2 次元のマップ上にプロットされ、その点と最大の相互補完強度を示す点（「マキシマム・ポイント」と名付ける）との距離で、その強度を数学的に表現され、それをインバートしたものを、相互補完強度を算出する式とする。

3.8 相互補完強度と相互補完強度係数の数式化

ここまでのように、2 社間の相互補完関係は、1 次元行列と 2 極のベクトルで表現され、その正の整数(プラスのベクトル)の総和と負の整数(マイナスのベクトル)の総和による 2 つの数字によって相互補完関係が示され、その点が 2 次元マップ上に示される。その点と最大の相互補完関係を示す点(マキシマム・ポイント)の距離で、2 社間の相互補完の強度を数学的に表現することができる。

このモデルにおける相互補完関係の強さを示す数値を「相互補完強度」（英語では Mutually Complementary Strength)と名付け、一般化した数式で表わすと、次のようになる。

アライアンスの相互補完強度 AS の式

$$AS = \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} - \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ plus}\right)^2 + \left(\frac{-(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ minus}\right)^2}$$

なお、s(max)は評点付けの最大値、s(min)は評点付けの最小値、len(c)は特徴数(項目数)、Σ plusは正の整数の総和、Σ minusは負の整数の総和である。

この数値を取り扱いやすいように、全体で割って、0~1 の間の数字に正規化すると、計算式は、下記となり、この式で求められる数値を「相互補完強度係数」(英語では Mutually Complementary Strength Coefficient)と名付ける。

特徴数や評点付けが異なる設定での 2 社間のアライアンスの相互補完関係を数値で比較する際、正規化する前の「相互補完強度」では、特徴数の違いにより、最大の相互補完関係の数値が異なってしまうため、比較がしにくい。そのため、0~1 の間で表現されるように、正規化された「相互補完強度係数」のほうが、相互補完関係を示す数値として利用しやすいと考える。

アライアンスの相互補完強度係数 ASC の式

$$ASC = 1 - \frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ plus}\right)^2 + \left(\frac{-(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ minus}\right)^2}}{\sqrt{\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} \times 2}}$$

なお、s(max)は評点付けの最大値、s(min)は評点付けの最小値、len(c)は特徴数(項目数)、Σ plusは正の整数の総和、Σ minusは負の整数の総和である。

以上のように、本研究で提案する相互補完数理モデルでは、2 社の強みと弱みの特徴を、1 次元の行列と 2 極のベクトルで数学的に表現し、そして、最大の相互補完関係を示す点からの距離で、相互補完関係の強さを数学的に表すこととし、上記のように、一般式で規定した。

第4章 フロー・インテンシティとフロー・バランスの概念の導入

4.1 フロー・インテンシティとフロー・バランスの概念の導入

本研究では、経営資源の交換がアライアンスであるという、アライアンス研究における資源ベース理論(RBV)のフレームワークを発展させて、2社間の強みと弱みの相互補完が強ければ強いほど、アライアンスは成立しやすいと考えて、アライアンス成立のメカニズムを表現する数理モデルを構築してきた。この相互補完数理モデルの説明に、本章では、物理学におけるフローの概念や、人間関係におけるギブ・アンド・テイクの考え方を盛り込んで、解説したい。

ここまで、アライアンスの数理モデルの構築においては、物理学における磁石のN極とS極が引き合う力を示すスピングラス・モデルをイメージして構築してきた。

しかしながら、スピングラス・モデルだけでは、2社間の引き合う力は説明できるものの、このモデルで説明付けようとしている、2社間が魅力を感じて引き合う関係性を説明するには、不十分である。

そこで、物理学におけるフローの概念と、人間関係の研究におけるギブ・アンド・テイクの考え方を取り入れることとした。これまでの経営学における研究では、理論(ディシプリン)をまたぐことをタブー視して、資源ベース理論に他の理論を組み合わせることがなされていなかったがために、アライアンスの数理モデル構築に至らなかったように思われるが、本研究では、既存概念に捉われず、資源ベース理論に、物理学での広くベースとして踏まえられているフローの概念や、心理学分野の人間関係におけるギブ・アンド・テイクの社会的交換理論を組み合わせた。

これにより、本章では、A社とB社間の強みと弱みのやり取り(相互補完)の度合いを意味するフロー・インテンシティ(Flow Intensity)と、A社とB社間の強みの提供と弱みの補完のバランスを示すフロー・バランス(Flow Balance)の概念を導入し、それら2つの用語を用いて、前章で説明した相互補完モデルの説明付けをさらに補強する。

フロー・インテンシティとフロー・バランスを用いて、アライアンスの相互補完モデルを捉えると、企業間アライアンスにおいては、フロー・インテンシティが強く、かつ、フロー・バランスが保たれている時に、アライアンスは成立しやすいと説明できる。

アライアンスをしようとする 2 社の関係性が、プラスもマイナスも 2 極のベクトルの数値が大きく、つまり、フロー・インテンシティが強く、双方のプラスとマイナスのフロー・バランスが保たれている時が、最もアライアンスに適している状態である。もしフロー・バランスが損なわれた場合は、2 社間のアライアンスの関係は、安定性が損なわれることとなる。

4.2 相互補完モデルをギブ・アンド・テイクの考え方で説明

前章で説明したように、企業の強みと弱みを、8 つの特徴数で、1~5 の 5 段階評価をした場合、A 社と B 社の関係性は、1 次元のベクトルと 2 極のベクトルで表現できる。ここでは、8 つの特徴を、営業力、技術力、アイデア力、人材力、生産力、資金力、信用力、組織の機動性としている。前章で提示した例をここで、ギブ・アンド・テイクの考え方、そして、フロー・インテンシティ、フロー・バランスの概念を導入して、図で示しながら、解説する。前章でも用いた、A 社と B 社の例を用いる。

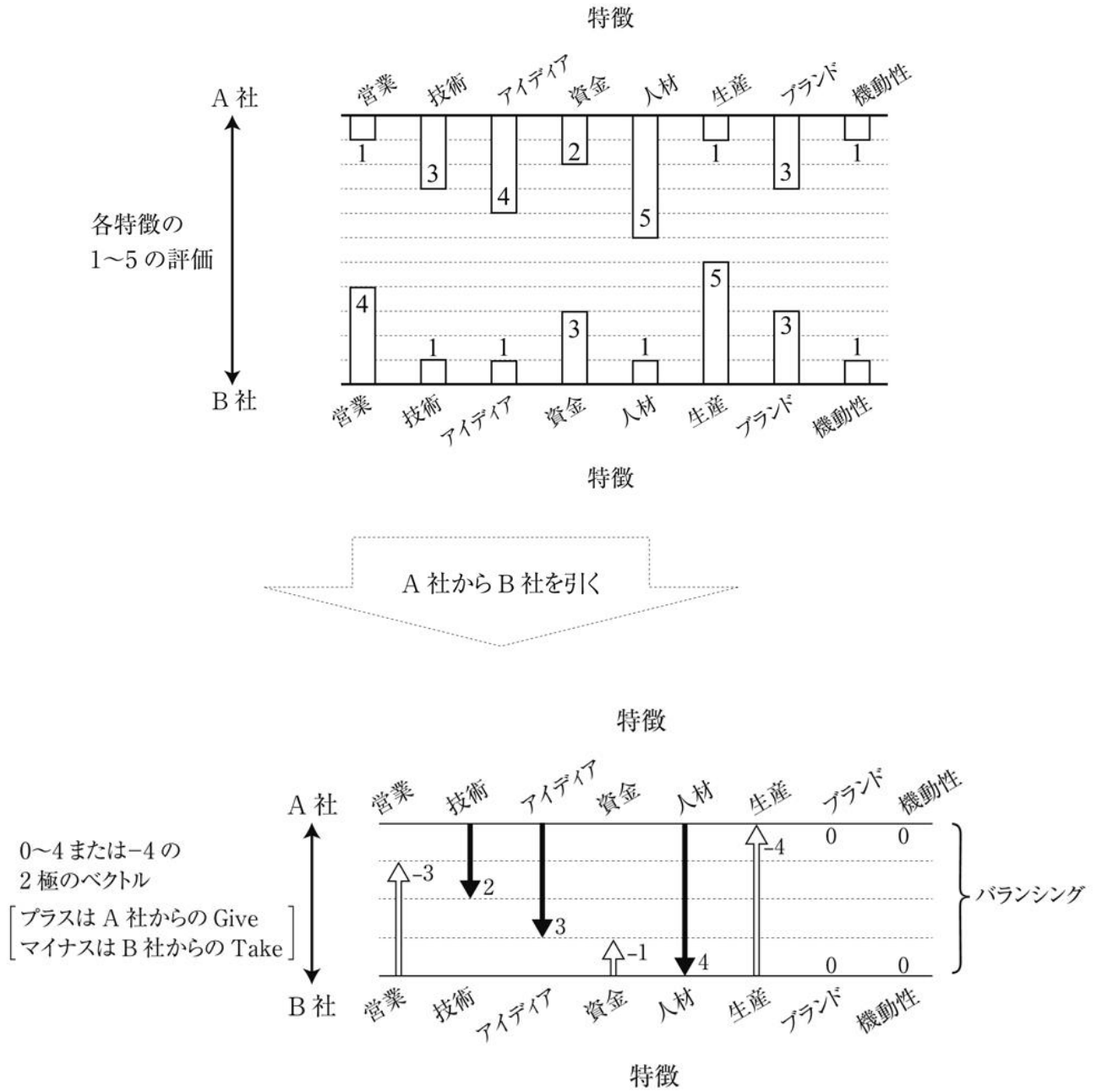
前述の通り、A 社と B 社の強みと弱みの各特徴の評点付けは、1 次元行列で表現され、その差し引きが 2 極のベクトルとして示されるわけだが、プラスのベクトルは、A 社から B 社への強みの提供、つまり、ギブ (Give) を示し、マイナスのベクトルは、A 社が B 社に提供してもらっている強み (弱みの補完)、つまり、テイク (Take) を示す。0 は、A 社と B 社の強みと弱みが同じで、A 社から B 社も、B 社から A 社も、強みの提供 (Give) も獲得 (Take) もなく、フロー・インテンシティがないことを表している。

相互補完モデルでは、B 社の強みが A 社の弱みを補完し、A 社の強みが B 社の弱みを補完することが考え方のベースとなる。もし片方か両方の企業からの強みの提供 (Give) が大きく、相手にとっての弱みの補完 (Take) が大きければ、相互補完関係が強くなる。このことを、フロー・インテンシティの概念で説明すれば、相互補完関係が強い時、つまり、ギブ (Give) とテイク (Take) のやり取りが大きい時、フロー・インテンシティが強くなっていると言い、反対に、相互補完強度が弱い時、フロー・インテンシティが弱いということとなる。

プラスとマイナスの 2 極のベクトルが存在する時、すなわち、ギブ (Give) とテイク (Take) が双方向にある時、フロー・バランスが保たれている状態であり、アライアンスが成立しやすい。

改めて、図で示すと、次の図8のようになる。

図8 特徴数8つでの5段階評価の相互補完のA社とB社の例



4.3 アライアンスが成立しないパターン1 :

フロー・バランスが保たれておらず、一方的な関係性の時

次に、2社間で、強み(アライアンスの相手先企業にとってのメリット)を提供し(Give)、自社の弱みを補う(アライアンスを行う企業のメリット)にあたって、バランスが重要であるということについて、考えてみる。逆に言えば、どのような時に、アライアンスが成立しにくい状況となるのかということである。

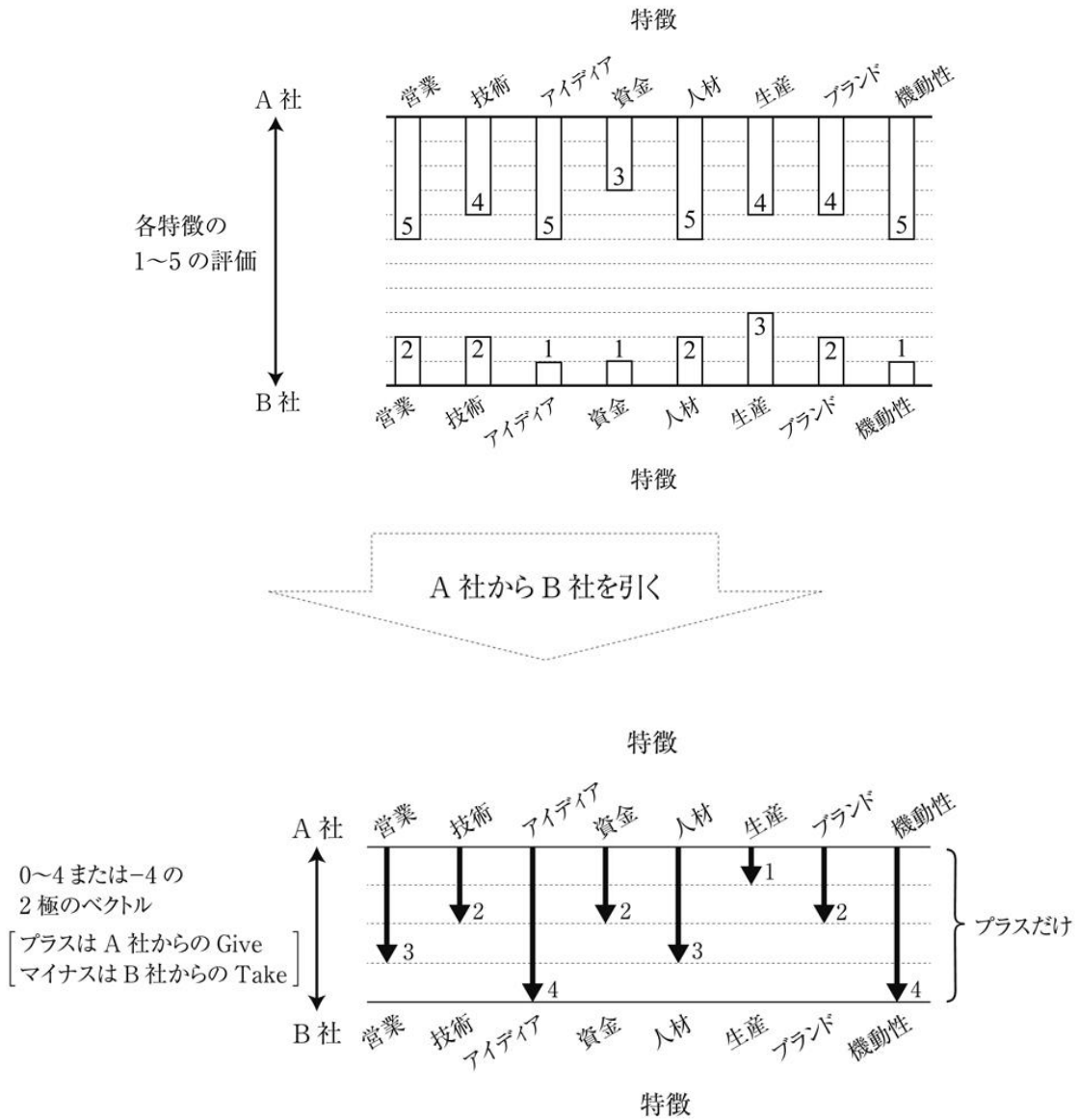
図9のように、A社のそれぞれの特徴におけるスコアがすべて、B社のそれぞれの特徴におけるスコアよりも大きい状態を考えてみよう。このケースでは、A社はB社に対して、資源・メリットを一方的に提供するだけとなり、バランスが保たれていない。言い換えれば、A社からB社へのプラスの2極のベクトルが、一方的(ワンサイド)である。B社からA社へのアピール、強みの提供(Give)がなにもない状態であり、こういうケースでは、アライアンスは成功しない。

このような状態が、アライアンスが成立しないパターン1となる。

片方の企業から、ギブ、ギブ、ギブばかりとなり、片方の企業がテイク、テイク、テイクばかりとなると、相互補完の関係性は成り立たず、その関係性はバーンアウトしてしまうのである。

図9 アライアンス成立しないパターン 1-1 :

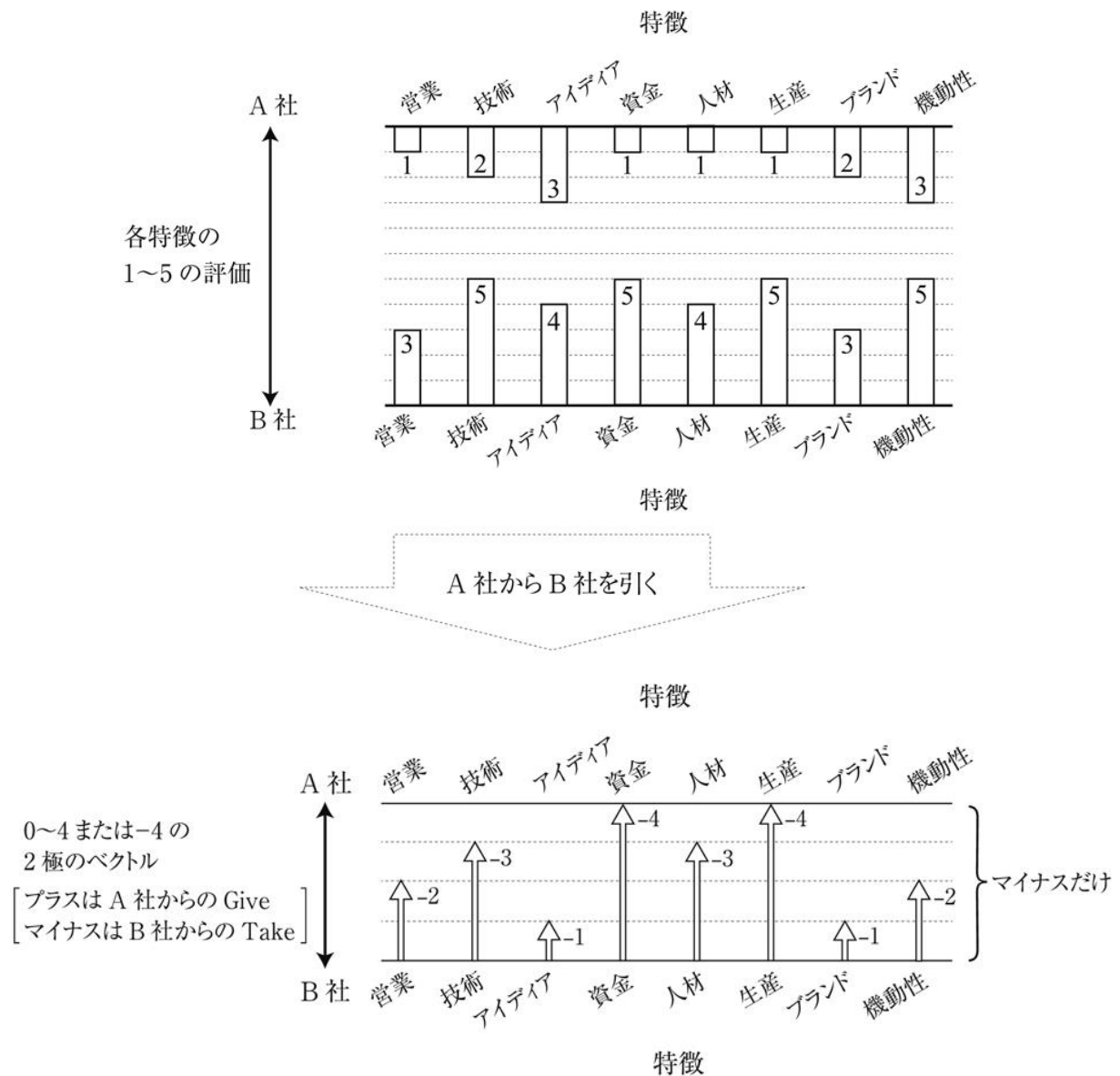
A社からB社への一方向的な強みの提供 (B社からA社への魅力の提供がない状況)



逆もまた真なりで、反対に、次の図10のように、B社のスコアが、すべての特徴において、A社よりも大きい場合には、B社だけが、一方向的に、A社に対して、強みを提供している状態となる。マイナスの2極のベクトルだけとなり、ワンサイドな関係性となる。A社からB社への魅力の提供がなにもないため、このケースでは、アライアンスは成立しない。

図 10 アライアンスが成功しないパターン 1-2 :

B社からA社への一方向的な強みの提供 (A社からB社への魅力がない状況)

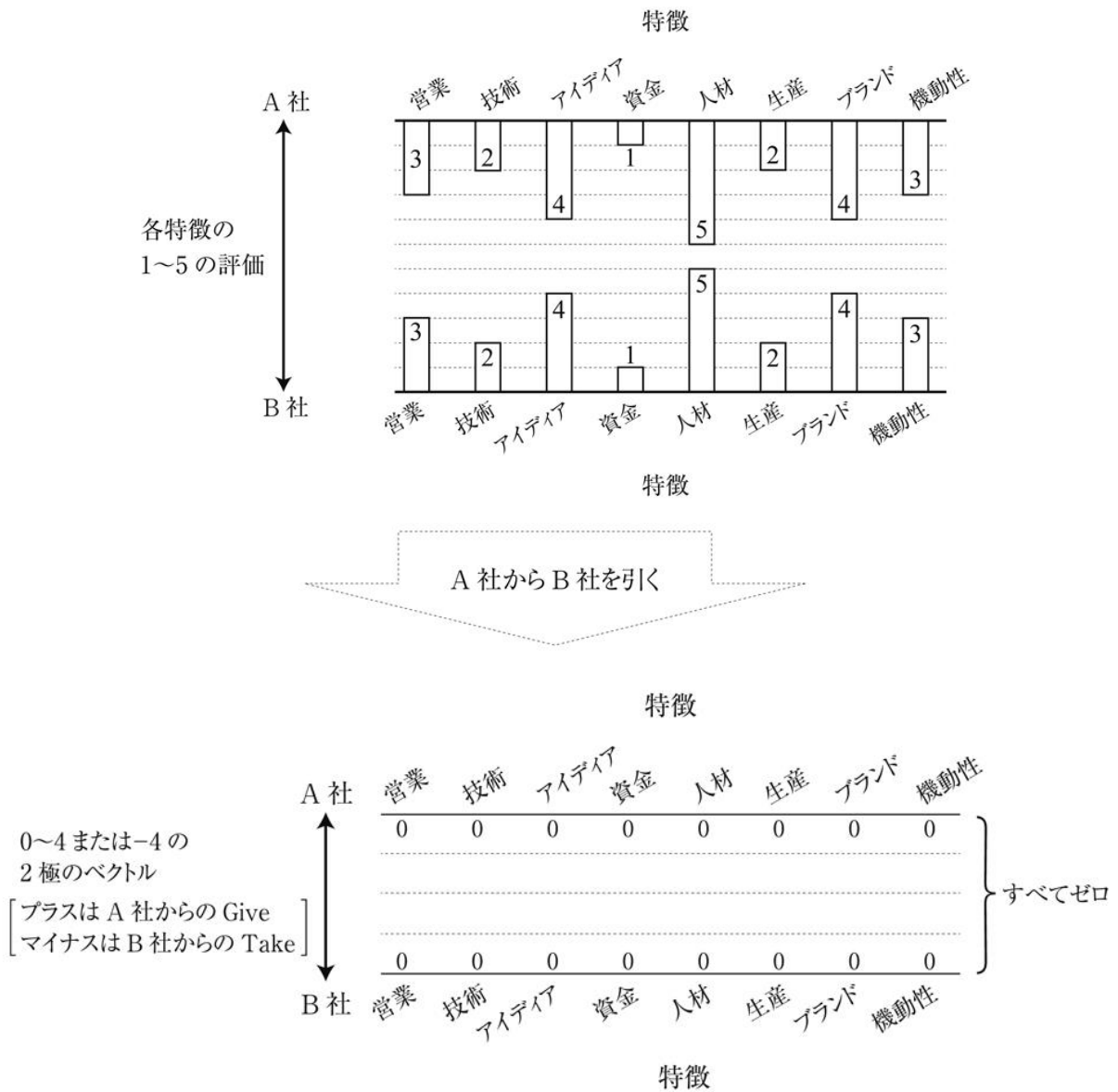


4.4 アライアンスが成立しないパターン 2 : フロー・インテンシティがない時

図 11 のように、A社とB社のスコアが、すべて同じ状況の場合は、それぞれの企業にとって、相互補完の恩恵が何もない。そのため、アライアンスは成立しない。A社とB社の強みと弱みが、すべての特徴において全く同じ場合には、フロー・インテンシティがゼロとなる。

このような状態が、アライアンスが成立しないパターン 2 となる。

図 11 アライアンスが成立しないパターン 2 : 2 社の強み・弱みの評点がすべて同じ時
(フロー・インテンシティがない状況)



このように、ここで提案するモデルでは、プラスかマイナスかどちらかだけのベクトルしかなく、強みの提供が片方からの一方的な関係性の場合には、アライアンスは成立しないと考える。なぜなら、2社間の強みの提供 (Give) と弱みの補完 (Take) のバランス、すなわち、フロー・バランスが保たれていないからである。また、2社の評点付けがすべて同じで、プラスやマイナ

スのベクトルがない状態も、フロー・インテンシティがゼロであるため、アライアンスは成立しないと考える。この2つのパターンが、アライアンスが成立しない状態となる。

先に掲載した図7に示したように、プラスとマイナスの両方の2極のベクトルがあり、双方向の強み (Give) と弱みの補完 (Take) が存在し、フロー・インテンシティがあり、フロー・バランスが保たれている時にアライアンスは成立しやすいと説明付けることができる。

4.5 最大の相互補完関係の状態：

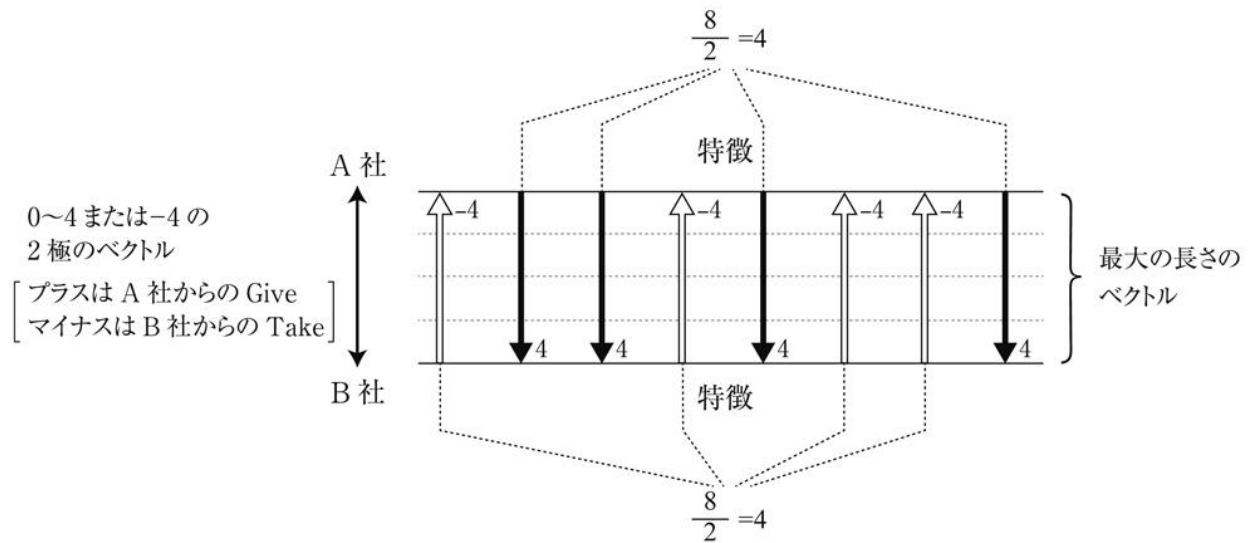
最大のフロー・インテンシティで、バランスが均衡している時

次に、アライアンスにおける2社間の最も強い関係性について考えてみよう。これは、前章でも解説したように、1~5の評点付けで、8つの特徴数の場合、8つの半分の4つずつの特徴数で、双方向で、取りうる最大の2極のベクトル4か-4の値となっている状態であり、2社の相互補完関係が(16, -16)の時となる。

この最大の相互補完強度の関係となる状態は、フロー・インテンシティとフロー・バランスの概念を用いて説明すれば、最大のフロー・インテンシティの状態で、かつ、フロー・バランスが均等に保たれている時となる。

図6では、最大の大きさのプラスとマイナスのベクトルが、右側から4つ、左側から4つの続いた順番で、シンプルに図解したが、実際は、次の図12のように、最大の大きなプラスとマイナスのベクトルは、ランダムに8つの特徴の中で、4つずつ存在する形となる。

図 12 最大の相互補完強度となる状況：最大の大きさの 2 極のベクトルが、
 特徴数の半分の数で、双方向で、存在する時（ランダムに配置）



このように、相互補完数理モデルでは、フロー・インテンシティが最大で、フロー・バランスが均等な時に、相互補完強度が最大になると捉える。

4.6 相互補完モデルの図のグレーの部分の説明

図 7 で示されているように、原点に近い 2 つの軸に近い薄いグレーの部分は、2 社間のバランスが崩れており、相互補完関係が弱く、アライアンスが成立しにくいゾーンとなる。さらには、原点から最大の強度の点への線(オプティマル・ライン)に対して、オプティマル・ラインから離れば離れるほど、フロー・バランスが崩れており、すなわち、2 社間のギブ・アンド・テイクのバランスが崩れている。本研究で提案するモデルでは、このインバランス(不均衡さ)は、アライアンスを成立しにくくしていると考ええる。

加えて、図 7 の原点により近い濃いグレーの部分は、フロー・インテンシティが弱く、原点に近づけば近づくほど、さらに弱くなっており、最大の点から原点に近いほど、よりインテンシティが弱い。原点では、フロー・インテンシティは、ゼロとなる。それゆえ、濃いグレーのゾーンは、相互補完関係が弱く、アライアンスがより成立しにくい。

第5章 152社のコンサルティング先企業データと評点付け方法

筆者は、Python 言語によって、この数式のモデルを実装した。2008年5月～2015年3月までのコンサルティング先企業152社の企業データを用いて、すべての組み合わせとアライアンスが成立した組み合わせ、アライアンスが不成立となった組み合わせ、すべての組み合わせについて、相互補完強度係数を実際に算出した。この章では、本研究で用いた企業データの内容、そして、Python 言語での算出結果について掲載する。

経営学をはじめとする社会科学研究においても、医療などの自然科学研究においても、データセット作成のプロセスやその大きさの設定、測定の方法などについては、それらのデータを用いた多変量解析などの実証分析の結果の確からしさを担保するためにも、重要である。データセット作成のプロセス自体についても、DeVellis (2012)、Hinkin(1998)、Hensley (1999)、Pierre et al. (2009)、Thanos et al. (2016)等において、適切かつ妥当な手法について、研究がなされてきている。しかしながら、本研究は、データを用いて実証する研究ではなく、企業間アライアンスのマッチング段階の成立メカニズムを示す数理モデルを提案することにあるため、データセットの構築のプロセスについては、簡素化して行っている。

5.1 152社の企業データの内訳

本研究において、分析対象として用いた企業データは、2008年5月～2015年3月でコンサルティングを行った152社同士の間のみとしている。

実際の筆者によるコンサルティングのアライアンス先の企業紹介では、コンサルティング先企業以外の企業でも、筆者が接点のある約5000社の中からアライアンスが成立する可能性のある企業の紹介も行っており、その結果、コンサルティング先企業以外の企業とのアライアンスも多数、成立している。

しかし、コンサルティング先企業152社以外の企業については、公開されている情報及びミーティング等において入手した情報があるものの、強み・弱みの8つの評点項目について、コンサルティング先企業と同程度に、評価するだけの正確な情報を入手・整備できない状況にある。

それらの情報入手が不完全な企業を対象とする企業データとして盛り込んでしまうと、本研究の検証結果にゆがみが生じてしまう。そのため、本稿においては、データを正確に入手・把握できている 152 社のコンサルティング先企業同士のアライアンス成立・不成立に限定して検証した。

本研究の筆者は、2008 年 5 月から 2015 年 3 月までに、152 社の日本企業に対して、新規事業構築や売上向上のコンサルティングを行った。本研究で用いられている企業データは、7 年間にわたって得られたものであり、守秘義務にかかわる通常は非公開となるインフォーマルな内容を含んでいる。

本稿で用いられたデータは、コンサルティング先企業から提出された資料や、それら企業の経営者や従業員との助言のためのミーティングでのやり取りから収集されたものである。紹介したアライアンス先企業から得られた情報もまた、このデータには含まれている。ヒアリング活動やアンケート調査といった表層的なサーベイとは異なり、筆者自身が、ビジネス活動に参加して得た情報となっている。

アライアンスに関する多くの先行研究のほとんどが、検証対象としては、株式市場に上場しているような大手主要企業となっている。例えば、Doz and Hamel (1998) や安田 (2003, 2006, 2010) も大企業を対象としている。しかし、本稿は、中小企業を対象とした研究であることも興味深いことである。

本稿で対象としたコンサルティング先企業 152 社の概要は次のようなものとなっている¹⁰。

所在地としては、東京 (23 区) 71%、首都圏 13%、非首都圏 (地方) 16%、企業規模及び業歴としては、上場企業 (その子会社を含む) 13%、業歴の長い中小企業 40%、業歴の浅いベンチャー企業 47%となっている。

業態としては、自社製品のメーカー 17%、受託製造・加工・施工 35%、販売・営業会社 30%、士業・コンサルティング 11%、また、IT 系か否かについては、IT 系 40%、非 IT 系 60%となっている。

売上高としては、10 億円未満 78%、10 億円以上 100 億円未満 15%、100 億円以上 7%、社員数

¹⁰ %は小数点以下第 1 位で四捨五入をしている。

としては、20名未満56%、20名以上100名未満32%、100名以上13%となっている¹¹。

それらを円グラフで示したものが図13～18である。

図13 152社のデータ・内訳 企業属性

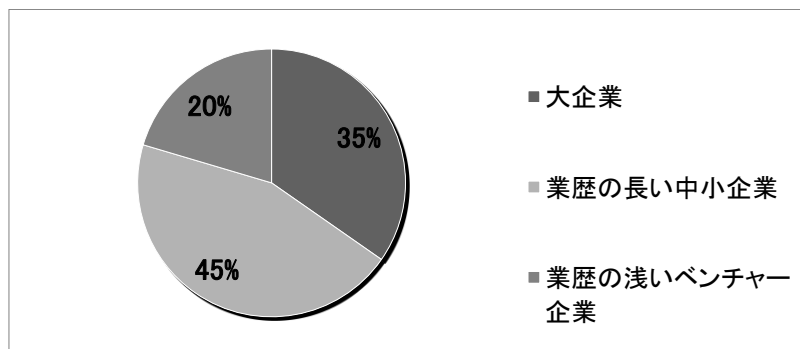


図14 152社のデータ・内訳 所在地

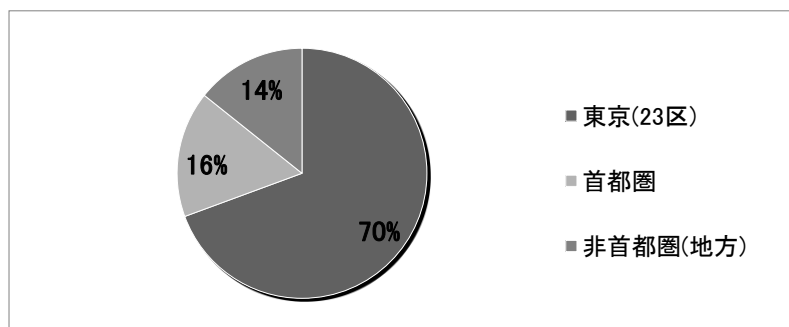
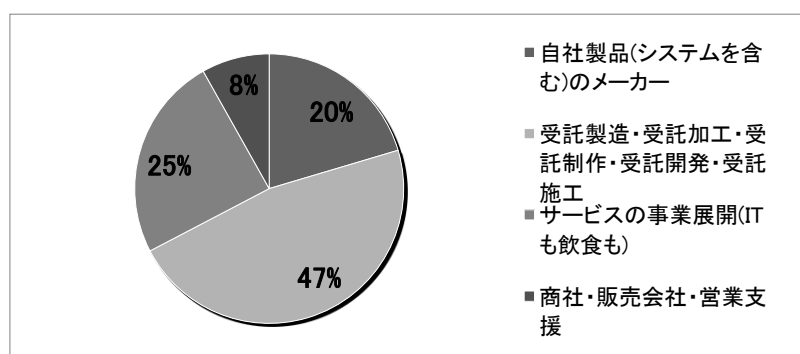


図15 152社のデータ・内訳 業態



¹¹ 初期の2008年～2013年くらいの約5年間は、中小ベンチャー企業のコンサルティングが中心であったため、調査期間の全期間の累積で見た場合には、この比率となる。最近の2013年～2015年の約2年間で見た場合には、東証一部上場企業を中心とした大手企業がコンサルティング先企業として急増しており、企業属性や売上高、社員数の内訳は時期によって変化している。

図 16 152 社のデータ・内訳 IT系/非IT系

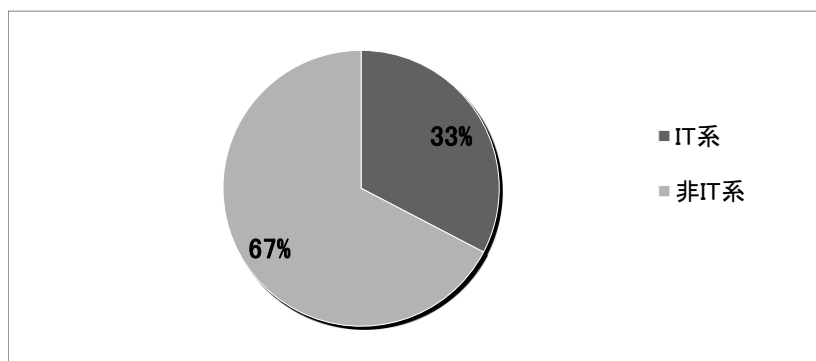


図 17 152 社のデータ・内訳 売上規模

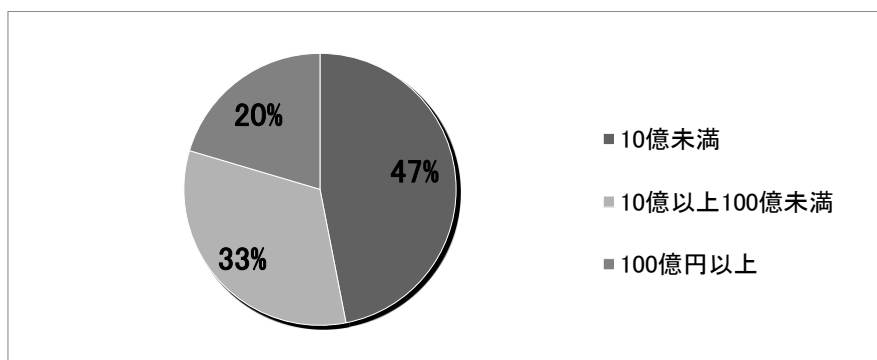
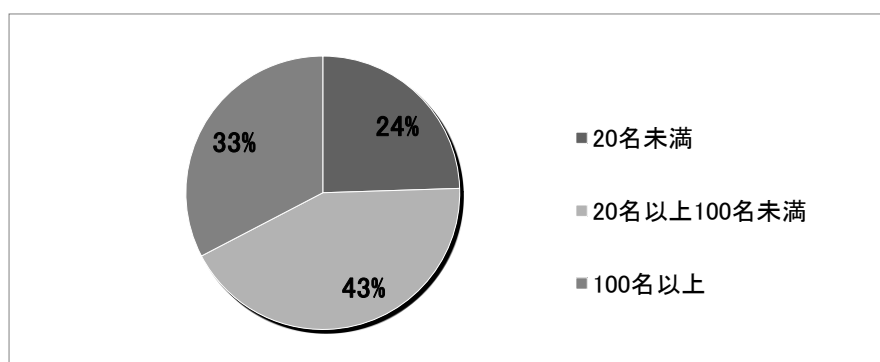


図 18 152 社のデータ・内訳 売上規模・社員数



5.2 当データの特殊性と有用性

当研究で用いた企業データは、一般には、公開されていない非公開のインフォーマル情報である。

コンサルティング先企業から提出された資料や、ミーティング(経営者や社員が対象)でのアド

バイスで得られた情報、提携先の紹介・アレンジメントで得られた情報などとして把握・収集された情報となる。すなわち、単なるアンケート実施の表層的な調査形態ではなく、筆者自らが事業活動に関与した参与分析の形態となっている。

また、アライアンスのアレンジメントをする仲介者は、Gompers and Lerner(2000)では、ベンチャーキャピタリストと同じく、情報収集をして、一般的に公表されていない企業の内部情報(財務面、技術開発面での今後の見通しも含め)を収集し、それを分析・評価し、情報生産をしている。この観点からも、既存の一般公表されているデータではなく、アライアンスの仲介者が情報生産した独自性のあるデータとなっている。

牛丸(2006)では、経営学における定量的研究は、統計データが豊富な経済学研究とは異なり、データの収集に多大なる時間を要することを指摘している。本研究も、筆者が約7年間で実際にコンサルティングをした企業を検証データの対象としており、より内実を把握しているデータともなっている。そのため、データ分析の算出結果の検証において、具体性を持って、検討ができるという意味においても、有用なデータとなっている。

当研究で用いた企業データは、筆頭著者が約7年間で実際にコンサルティングをした企業についての一般には公開されていない非公開のインフォーマル情報を含んでいる点に特色がある。

本稿で用いたデータは、コンサルティング先企業から提出された資料や、ミーティング(経営者や社員が対象)でのアドバイスで得られた情報、提携先の紹介・アレンジメントで得られた情報などとして把握・収集された情報となっている。単なるアンケート実施の表層的な調査形態ではなく、論文執筆者自らが事業活動に参加した参与分析の形態となっている点に特徴がある。

また、多くのアライアンスの先行研究が、株式市場に上場をしているような大手企業を対象とした研究となっている。たとえば、Doz and Hamel(1998)や安田(2006、2010、2015)も、対象を大手企業¹²としているが、本稿では中小企業も対象としていることが特色の一つとなっている。

5.3 アライアンスの組み合わせの範囲の限定

¹²株式市場に上場しているような相対的に売上規模が大きい企業(おおよそ売上300億円以上)と筆者は定義する。

筆者によるコンサルティングの実務における企業紹介では、コンサルティング先企業以外の企業の中からマッチする企業の紹介も行い、アライアンスが成立している。

しかし、コンサルティング先企業 152 社以外の企業については、公開されている情報及びミーティング等において入手した情報があるものの、コンサルティング先企業と同じように評価するだけの正確な情報を入手・整備できない状況である。

それら、情報入手が不完全な企業をデータとして盛り込むと、本研究の検証結果にゆがみが生じてしまうため、本稿においては、データを正確に取れる 152 社のコンサルティング先企業同士のアライアンス成立・不成立に限定して検証した。

本稿での 152 社の企業データにおけるアライアンス成立の組み合わせは 121 件、アライアンス不成立の組み合わせ 30 件となった。152 社と 152 社の間のすべての組み合わせ 11476 件(152×152-152)についても算出し、比較対象とした。

筆者は、おおよそ、4000 社以上の潜在的なアライアンス候補先企業とのリレーションを有しており、それらのコンサルティング先企業以外の企業も、152 社のコンサルティング先企業に対して、紹介を行っている。それらのコンサルティング先企業以外の企業は、非常に数も多く、8 つの特徴について、表層的な評価しかできない。筆者によるコンサルティング先企業への紹介業務においては、コンサルティング先企業以外の企業とのアライアンスが成立しているケースも当然、存在する。しかし、それらの紹介した企業については、入手可能な情報が限られており、正確でない場合もあり、データセットをゆがめてしまうため、本研究の対象とするデータから除外した。この研究は、コンサルティング先企業 152 社の間でのアライアンスが成立したか、不成立になったかだけに限定している。

コンサルティング先企業の実証データは、2008 年 5 月から 2005 年 3 月までの間のデータであり、それらのコンサルティング先企業に対して、コンサルティングで関与された期間は、最も短い場合で 2 ヶ月間、最も長い場合で 3 年間である。コンサルティングの期間、すなわち、アライアンスのアレンジメントを行っていた間に、各企業の評点付けが変わることもあるが、トータルの期間を総合的に判断して、評点付けを行っている。

5.4 強み・弱みの8つの特徴の選定

強みと弱みについて、この研究では、8つの特徴を用いている。このことは、第3章においても、説明したが、ここで、再度、補足しておきたい。(1)営業力が強いかな弱いかな、(2)技術力が高いか低いか、(3)アイデア創出力が強いかな弱いかな、(4)資金力があるかないかな、(5)人材が豊富か否か、(6)生産力が豊富か否か、(7)ブランドや信用があるかないかな、(8)組織の機動性があるかないかな、である。152社のコンサルティング先企業について、筆者は、8つの特徴について、1~5段階(5が高い点数)で、評点付けをした。後述もするように、評点付けにおける主観性の問題を解決するために、評点付けのプロセスにおいて、それぞれの特徴について4つずつのクライテリアを設けて、それらを総合するステップを踏んで、評点付けを行った。加えて、他のもう一人のコンサルタントが同じ方法で、評点を付けて、その比較検討によって、評点を決めた。

152社の企業についての強み・弱みの評点付けについては、前述の通り、富田(2014)で紹介されたアライアンスの機能別のパターンで関係している項目とするともに、本研究の理論的背景としている資源ベース理論(RBV)での経営資源の分類を意識して、設定した。

本研究での強み・弱みの8つの項目(特徴)は、①営業力がある・なし、②技術力がある・なし、③アイデア力がある・なし、④資金力がある・なし、⑤人材がいる・いない、⑥生産力がある・なし、⑦ブランド・信用がある・なし、⑧機動的な組織風土である・ないとした。

これらの8つの項目(特徴)について、1~5の5段階評価を行った。強み・弱みが最も強い時が5、最も弱い時が1という評点付け方法とした。5段階評価の方法を採用した理由は、日本においても、米国をはじめとする諸外国においても、学校の成績評価は5段階が用いられており、最も馴染み深い評点付け方法であるためである。

本研究で提案するモデルの特徴数は、自由に決められるため、特徴数は、8つである必要はない。事実、付録1に掲載しているように、最初のモデル構築においては、特徴数4つで行っている。

ここで8つの特徴とした理由としては、安田(2003, 2006, 2010)において、アライアンスは経営資源の交換であり、5つの経営資源として、①技術資源、②販売資源、③生産資源、④人材資源、⑤資金資源であるとされているため、まず、この5つを採用した。

加えて、Das and Teng(1998b)において、信用の大切さが述べられているため、信用・ブランドを追加し、また、Chesbrough(2003)や米倉・清水(2015)で、オープン・イノベーションにおけるアイデア獲得の重要性が述べられているため、アイデア力を追加した。さらには、Hamel and Prahalad(1994)にて、アライアンスをする理由として、スピードアップと組織としての学習効果が述べられているため、組織の機動性を追加した。これら3つを追加し、合計8つを企業の強みと弱みの評価付けの特徴数とした。

5.5 評点付けにおける主観性排除への対応

経営学の研究においては、その研究結果の妥当性を担保するために、そもそもの研究の素材となるデータの正確さが大切となり、そこにおいては、客観性が求められる。

本研究で用いているデータは、コンサルティング先企業152社について、8つの項目について、筆者が中心となって、強み・弱みの評点付けを行った形となっている。すなわち、評価を付けている者の主観性という問題が残る。そのため、できる限り、評価者の主観性を排除するために、次の2つの方策を講じた。

1つ目の方策としては、強み・弱みについての8つの各項目の評点を付けるにあたり、各項目を構成する4つの項目を因子としてさらに作成し、それらの4つの項目で評価付けをした上で、それらを取りまとめる形で、最終的な各特徴における評点付けを決めるというステップを取った。

たとえば、営業力があるということは、どういうことかという項目として、営業マンの数が多、営業マンが生き生きしているなどの基準、また、資金力があるというのは、財務数値としてどういう状態かを判断する基準を因子として作った。これらについては、本稿の付録2として、その内容を掲載している。

強みと弱みの8つの特徴ごとの評価にあたって設けた4つずつのクライテリア(因子)の概要は、次の表2の通りである。より詳細なクライテリアの内容については、本稿の付録2として掲載している。

表2 強みと弱みの8つの特徴の評点付けにおける4つの因子(クライテリア)要点のみ

1. 営業力	2. 技術力
1. 営業マンがアクティブかどうか 2. 営業マネジメントができているか 3. 営業マンの数が多いか少ないか 4. 経営者自身が営業が得意か否か	1. ユニークな独自の技術を有しているか 2. 技術力のレベルか高いか低い 3. 特定領域で長く培っている技術があるか 4. 研究開発部門が充実しているかどうか
3. アイディア創出力	4. 資金力
1. 自由に新しいアイディアを出せる環境にあるか 2. アイディアを出せる人が多いか少ないか 3. 継続してアイディア出しを行っているかどうか 4. アイディア出しができる仕組みを有しているか	1. 使用できる資金量が豊富かどうか 2. 自己資本比率が高いか低い 3. 資金調達力があるかどうか 4. 営業利益率が高いか低い
5. 人材力	6. 生産力
1. 従業員数が多いか少ないか 2. 人員が余剰しているか不足しているか 3. 人材採用力が高いか低い 4. 人材紹介や派遣の免許を持っているかどうか	1. 工場設備が大きい小さいか 2. 工場での作業員を多いかどうか 3. 生産工程でのノウハウを有しているか 4. ファブレス企業の方針であるかないか
7. ブランドや信用	8. 組織の機動性
1. 上場企業や上場企業の子会社であるかどうか 2. 業歴が長い短い 3. 売上規模が大きい小さいか 4. ブランディングに力を入れているかどうか	1. 新しいことに挑戦する体質であるか 2. 議論ばかりでなく、実行する社風か 3. 年功序列であるかどうか 4. 外部と組もうという雰囲気があるか

次に、2つ目の方策としては、一人だけでなく、もう一人別のコンサルタントも、同じように評点を付け、その比較から総合して、会社組織として評点を決定した。

なお、コンサルティング期間は、数か月程度から4年程度の幅はあるが、コンサルティング開始時点で付けた評価を、その後のコンサルティングの中で分かった情報を踏まえ、最終的に、2015年4月時点での評点とした。

このように、強み・弱みの8つの項目での5段階評価の評点については、筆者の属する会社のもう一人の担当者の評価と照らし合わせて、二人の協議の上、評点を決定する方式を採用し、また、8つの各項目について4つの観点での評価基準を作ることで、できる限り主観性を排するための方策を行った。しかし、当然ながら、完全に主観性が排除されているわけではない。

ここで、留意すべきは、本研究は、アライアンスのための数理モデルを提案することが目的

としていることである。あくまで、この企業データは、本研究のモデル構築のために用いたデータに過ぎず、日本企業全体や世の中の企業を代表するものではない。

相互補完強度や相互補完強度係数は、それぞれの評点付けに基づいて算出されているため、当然、評点の付け方や評点付けのスコアが変われば、算出された数値も変わってくる。評点付けによって、算出された相互補完強度及び係数も影響を受ける。むしろあえてバイアスをかけた評点付けとして、試してみることもできよう。将来的には、評点付けの仕組みを変えることも、本研究の発展のためには、必要となろう。

本研究の目的は、アライアンスの数理モデルを提案して、それをプログラミングで実装し、実際のデータで機能するか否かを確認することにある。あくまで、ここで用いたデータは、データの一例に過ぎないことに留意されたい。

5.6 アライアンス成立・不成立のカウン트의定義と件数

本研究における企業データでのアライアンスの成立・不成立のカウンについては、「アライアンスの成立」は、「企業紹介をして、それら2社間にて、事業・サービスの構築や新製品開発、既存事業の拡大、新規取引(受注・販売)、営業協力・販売代理などのメリットを感じて、なんらかの話し合いが前に進んだ場合」としてカウントし、「アライアンスの不成立」は、「まったく前に進まなかった場合」をとしてカウントした。

本研究の目的・意義は、アライアンスの初期であるマッチングの段階まで、アライアンス研究を拡大し、その段階でのアライアンス成立のメカニズムを数学的に表現し、発展可能なモデルを提案することにあるため、本稿におけるアライアンス成立の定義は、「企業紹介をして、それら2社間にて、事業・サービスの構築や新製品開発、既存事業の拡大、新規取引(受注・販売)、営業協力・販売代理などのメリットを感じて、なんらかの話し合いが前に進んだ場合」と定義し、また、「まったく前に進まなかった場合」を不成立と定義した。また、同様に、マッチングとは、「企業と企業がアライアンスの契約締結や共同での事業構築などのために話し合いをし始めること」と定義する。

あくまで、アライアンスの最初の段階であるマッチングのフェーズを研究しているため、まず

は、話し合いが進み始めたか否かで、アライアンスの成立をカウントすることとした。

日本においては、アライアンスの成立において、必ずしも、アライアンスの契約締結に限定することは適当ではない。契約主義が強い米国においてや、大企業同士のアライアンスにおいては、アライアンスとして成立するには、通常は、双方が目的を遂行するために契約を取り交わすことがある。それゆえ、アライアンスが成立の要件としては、厳密には契約の締結がなされた時点、あるいは、アライアンスによって成果として、新規事業が構築されたり、既存事業が拡大されたりして、売上や利益を双方が享受できた時点と捉えることもできる。

しかしながら、本稿の検証する企業データは、業歴の長い中小企業や業歴の浅いベンチャーを多く含んでおり、また、日本の商慣習の中においては、必ずしも、アライアンスにおいては、契約締結を行わないことが散見される。口頭での合意だけで、アライアンスが進むこともあり、文書でのアライアンス契約が締結されなくとも、アライアンスが成立したと把握することが適切なケースも存在する。アライアンス契約ではなく、単に、紹介料やマージンの支払いの契約だけでアライアンスが行われていくこともある。日本における商習慣では、そのような形式でのアライアンスが手法の一つとして行われることおあり、アライアンスの成立において、フォーマルな契約締結が必ずしも要求されているわけではないことが、その背景としてある。

さらには、本研究は、第1章でも述べたように、提携先の探索、契約締結、協働での事業構築、提携先企業との関係の調整、売上・利益の享受といったアライアンスのプロセスの中で、最初の段階の提携先の探索やマッチングの段階を研究対象としているため、アライアンス成立のカウントでは、仲介者が紹介をした後、話し合いが、なんらか進捗したことを成立の要件とした。

また、日本においては、アライアンスの成立・推進において、必ずしも、契約締結が行われるとは限らず、口頭での合意やアライアンス契約ではなく、紹介料やマージン支払などが実際に発生した時点で契約締結や請求書発行をする形で進めることもある。日本のビジネス社会の風土が、米国のような契約主義でないことにも考慮して、アライアンスの契約締結をアライアンス成立の要件とはしていない。

152社のコンサルティング先企業データにおいて、121件の成立したアライアンスの組み合わせ、そして、30件の不成立となった組み合わせが存在する。加えて、152社同士において、すべ

での組み合わせとして、 $152 \times 152 - 152$ で、11,476 の組み合わせが存在し(「全体」と呼ぶこととする)、それら、全体についても、検証した。なお、152 社間での企業紹介の成立率は、80.1%、不成立率は 19.9%である。

表3 成立したアライアンスと不成立となったアライアンスの数

分類	数
成立したアライアンスの数	121
不成立となったアライアンスの数	30
すべての組み合わせの数(全体)	11,476 ($152 \times 152 - 152$)

アライアンスにおいては、水平統合と垂直統合という分類の仕方がある¹³。水平統合は、同じ分野で企業同士が提携することによって、その分野での価値をさらに向上させたり、営業協力などを行って事業拡大をしたりするタイプのアライアンスである。他方、垂直統合は、研究開発をする企業と生産を担う企業が提携して、新製品開発を完成させるといったタイプのアライアンスでさる。当データには、それら両方が含まれている。

本研究で用いた企業データにおけるアライアンスの成立は、当該データにおけるアライアンス成立の中には、本稿の冒頭でのアライアンスの定義には合致しない一般的な取引(契約獲得や受注、売買取引等)が含まれていることを注記しておきたい。これらの一般的な取引は、2社を引き合わせている仲介者の存在がなければ、それら企業は取引をしていないことになる。これは、市場での取引の補完をアライアンスのアレンジメントを行っているコンサルタントが、仲介者としての役割を担っていると言えよう¹⁴。

¹³ 下山弘一「アライアンスの2つの形。垂直統合・水平統合」、2002年、NPO法人・ITコーディネーター京都のWebサイトに掲載を参考にした。

¹⁴ アライアンスの仲介者の果たしている機能については、ベンチャーキャピタル研究における Gompers and Lerner(2000)でのプリンシパル・エージェント理論による分析(ファンドの出資者とベンチャーキャピタリストの関係、投資先企業の経営者とベンチャーキャピタリストの関係を研究)やベンチャーキャピタルの情報生産の機能分析を応用させることができよう。エージェントである仲介者(アライアンスのコンサルタント)とプリンシパルであるコンサルティング先企業との関係の説明付け、エイジェンシー・コストやモラル・ハザード、利益相反の問題の解消につ

第6章 相互補完強度係数の Python 言語での算出と検証

6.1 Python 言語による提案したモデルの実装

第4章で提案した相互補完モデルをオープン・ソースのプログラミング言語である Python 言語で実装し、第5章で紹介した152社のコンサルティング先企業データを用いて、実際にアライアンスの相互補完強度係数を算出した。

本モデルは、Python 言語で実装しなければならないわけではないが、Python 言語は、高い自由度を持っているため、利便性が高く、表計算ソフトの Excel で算出を行うよりも、モデル構築や計算が容易であった。また、近年、急速に普及しているため、スクリプトをオープン・ソースとして公開して、幅広く、多くのユーザーに利用してもらうにあたって、適していると考え、Python 言語を本研究の実装するツールとして採用することを決めた。

事実、オープン・ソースのプログラミング言語の自由さのおかげで、試行錯誤をしながら、フレキシブルなモデルとして、提案するモデルを構築することが可能となった。

なお、経営学の学界においては、これまで、Stata のような統計分析のソフトウェアが利用されることが一般的であったが、Python 言語への関心が高まりつつある。本研究は、Python 言語を経営学分野の研究に用いた先駆的な研究であると言える。

6.2 成立した組み合わせと不成立となった組み合わせの相互補完強度係数の分布

提案するモデルを Python 言語で実装し、そのスクリプトによって、相互補完強度係数を実際に算出した。

アライアンスが成立した組み合わせ(121件)の相互補完強度係数の0.05刻み(区切りの数値は未満。以下、同じ)での分布は、図19の通りである。また、アライアンスが不成立となった組み合わせ(30件)の相互補完強度係数の0.05刻みでの分布は、図20の通りである。

さらに、アライアンス成立と不成立の相互補完強度係数の0.05刻みでの分布を重ねたグラフが

いても、今後の研究対象となろう。

図 20 である。

図 19 アライアンス成立の相互補完強度係数の分布のグラフ

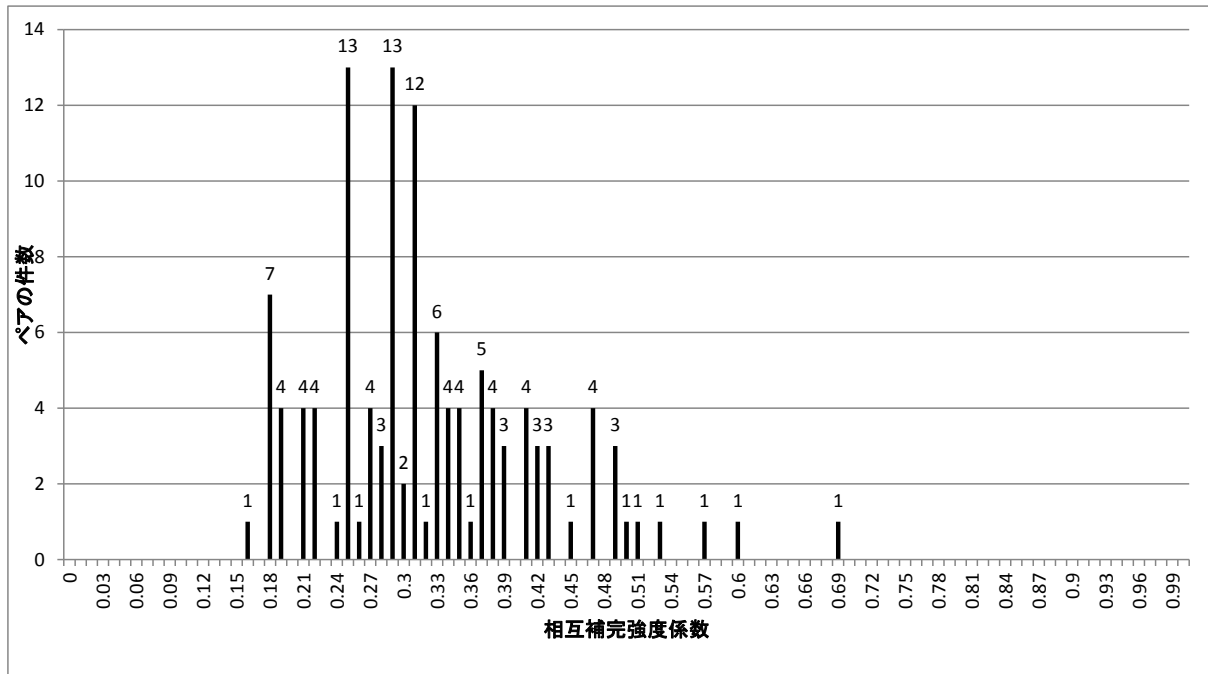
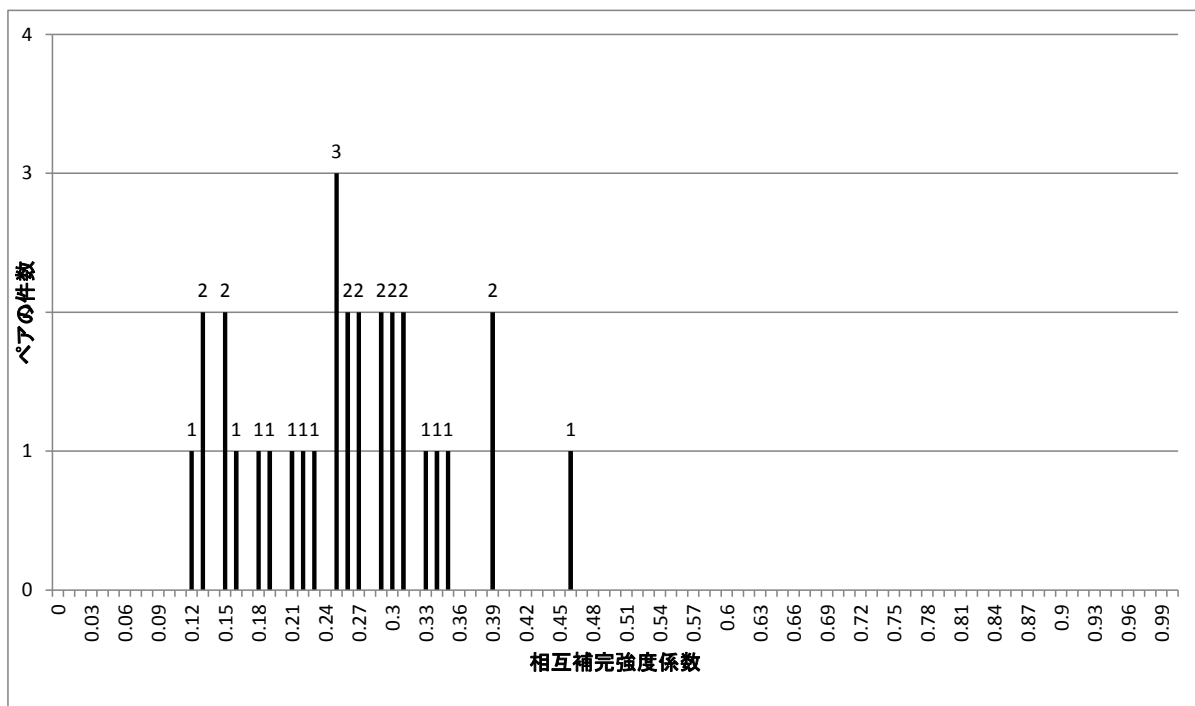
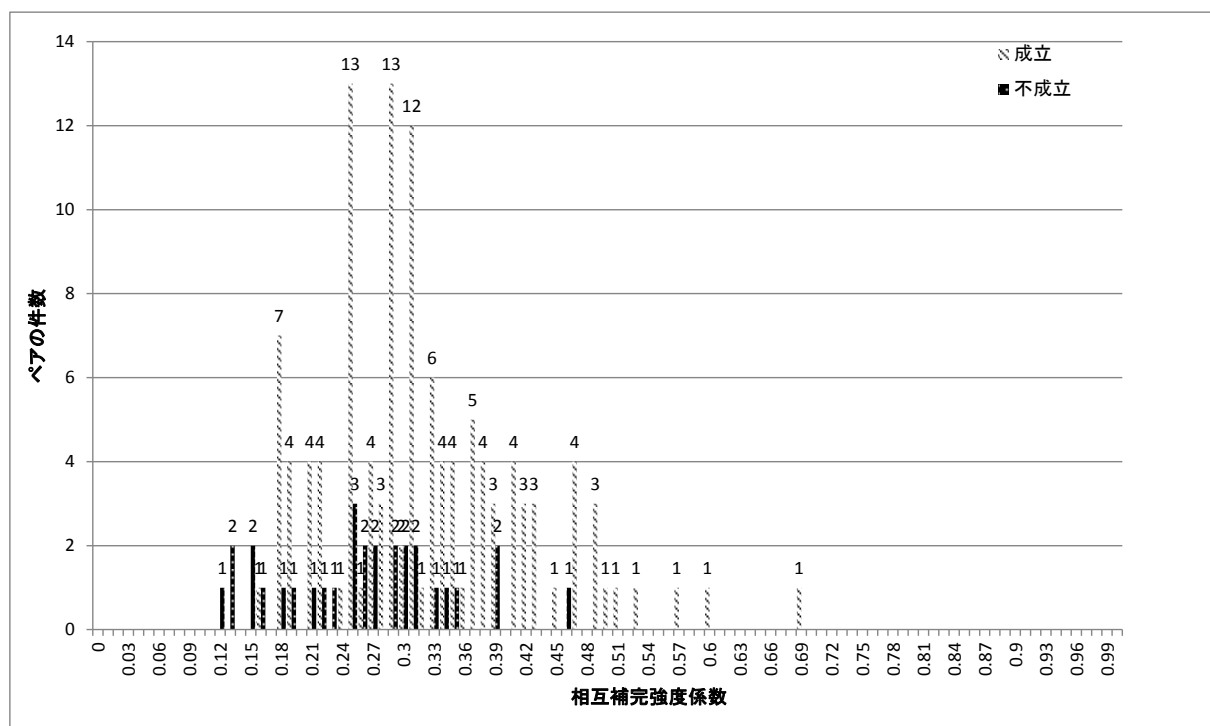


図 20 アライアンス不成立の相互補完強度係数の分布のグラフ



アライアンス成立・不成立の相互補完強度係数のグラフの分布(図 19、20、21)では、全体的に、アライアンスが成立した組み合わせは、アライアンスが不成立となった組み合わせよりも、高い相互補完強度係数として分布している。相互補完関係が強い時、すなわち、相互補完強度係数が大きい時に、アライアンスが成立しやすいというモデルの正当性を確認できた。

図 21 アライアンス成立と不成立の相互補完強度係数の分布を重ねたグラフ



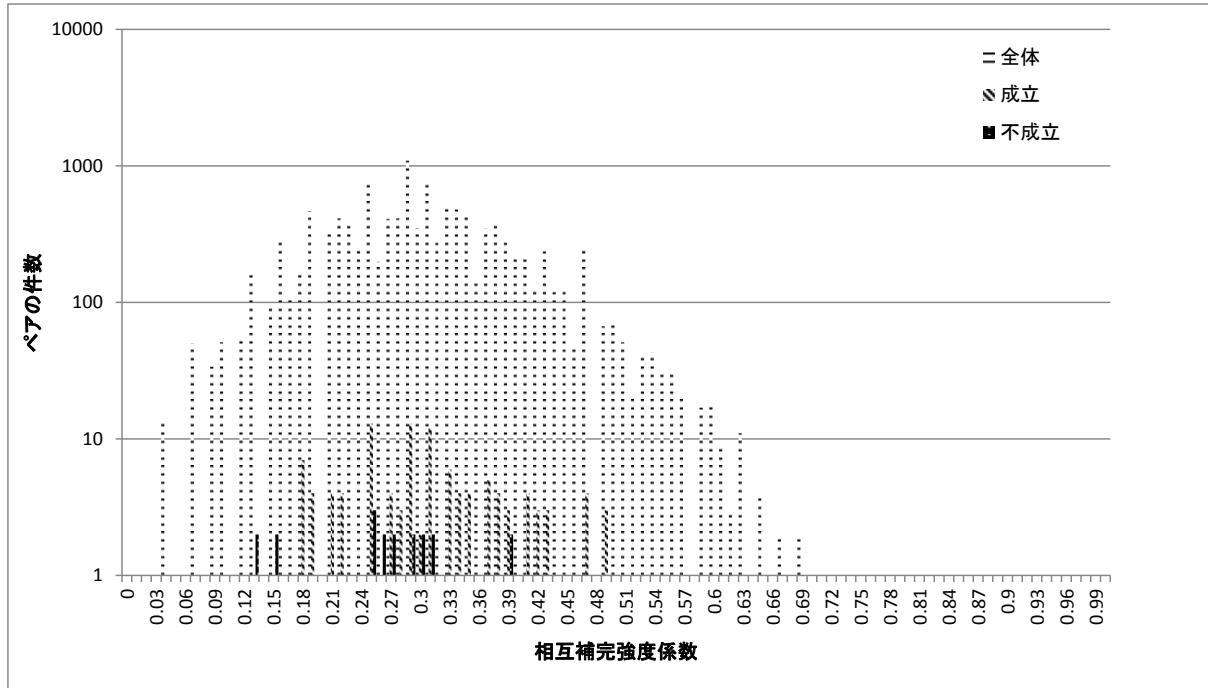
これを、さらに 152 社と 152 社すべての組み合わせの相互補完強度係数の分布と重ね合わせたグラフは、図 22 である。組み合わせすべての全体の分布は、正規分布に近い形状となっているように見える。ただし、評点付けが変われば、当然、この係数の分布は変わってくる。

すべての組み合わせ(以下、「全体」と呼ぶ)の相互補完強度係数よりも、アライアンス成立の組み合わせの係数は大きい方向に偏りが見られ、他方、アライアンス不成立の組み合わせについては、より低い係数の方向に偏っていると読み取れる。

ここにおいても、相互補完関係が強い時、すなわち、相互補完強度係数が大きい時、アライア

ンスが成立しやすいという提案するモデルの正当性が確認できた。

図 22 成立・不成立・全体の相互補完強度係数の分布のグラフ



6.3 成立・不成立・全体の相互補完強度係数の平均

次に、アライアンスが成立した組み合わせと、アライアンスが不成立となった組み合わせの相互補完強度係数の平均による比較を行ってみたい。

全体の相互補完強度係数の分布は、図 21 に掲載した通りであるが、全体の分布は、正規分布に近い形状となっている。そのため、ここでは、成立・不成立・全体の比較において、平均を用いることとした。このグラフでの分布を見てみると、成立の組み合わせは、全体よりも、数値が大きい方向に分布している傾向があり、他方、不成立の組み合わせは、全体の分布よりも、数値が小さい方向に分布する傾向が読み取れる。

平均を計算してみると、表 4 に示すように、121 件の成立したアライアンスの組み合わせの相互補完強度係数の平均が 0.3180、30 件の不成立となったアライアンスの組み合わせの相互補完強度係数の平均が 0.2538 である。152 社間のすべての組み合わせ(の相互補完強度係数の平均が 0.2997 となっている。その他のパターンの平均の数値も、表 4 に掲載した通りである。グラフ化

すると図 23 となる。

表 4 成立・不成立・全体等の相互補完強度係数の平均

	相互補完強度係数の平均
アライアンス成立	0.3180
アライアンス不成立	0.2538
全体	0.2997
全体－成立	0.2995
全体－成立－不成立	0.2996
全体－不成立	0.2998

図 23 成立・不成立・全体等の相互補完強度係数の平均

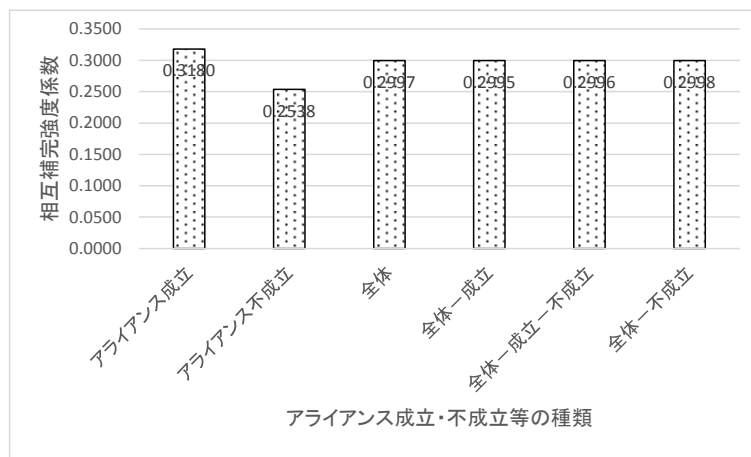


表 4 及び図 22 ように、成立したアライアンスの係数の平均は、アライアンスの不成立の係数の平均よりも、大きい数字となっており、さらに、全体の係数の平均よりも、大きな数字となっている。

このように、アライアンスが成立した組み合わせのほうが、不成立となった組み合わせよりも、2 社間の相互補完強度係数がより大きい。2 社間の相互補完関係が強い時に、アライアンスが成立しやすいという相互補完モデルの正当性が確認できた。

ただし、アライアンスが成立した組み合わせの係数と、不成立となった組み合わせの係数に大きな差がないのは、アライアンスの仲介者たるコンサルタントが、企業紹介にあたって、相互補完関係があるだろうという想定のもと、企業紹介を行っているためと考える。相互補完強度係数の成立と不成立の分布では、不成立となったアライアンスの組み合わせでも、大きな数値となっ

ているものもあるが、その理由も同様である。

6.4 成立・不成立・全体の評点の実数による検証

以上のように、相互補完強度係数による検証を試みたが、さらに相互補完強度及び係数にすることの有用性の検証を試みるために、評点の実数での分析結果についても見てみることにした。

相互補完強度は、双方のペアの評点の差し引きであるため、実際、絶対値としての評点の総数(総合点)が高い企業、つまり、強みをトータルで多く持つ企業のほうが、アライアンスがより成立しやすくなっている否かについて、検討した。逆に、アライアンスが不成立となって組み合わせの企業は、トータルとして評点の数が小さかった否かも確認してみた。

図 23 は、アライアンス成立・不成立・全体の組み合わせの片方の企業の総合点(8項目で最大5点のため、最大値は40点)の分布を示している。アライアンスが成立した組み合わせの企業も、アライアンスが不成立となった組み合わせの企業も、すべての組み合わせの中に納まった分布となっている。アライアンスが不成立となった組み合わせの企業の総合点は、成立した組み合わせよりも、多少、低い数字となっていることが読み取れるものの、特段際立った特徴は見られない。

図 24 は、アライアンス成立・不成立・全体の組み合わせの両方の企業の総合点(8項目について最大5点で、2社のため、最大値80点)の分布を示している。このグラフの分布からは、片方の企業や両方の企業の評点の総合点からは、アライアンス成立・不成立についての意義ある結論は導き出せなかった。

このように、各企業の強みの総合点の実数からは、明確な見解は導き出せず、やはり、相互補完関係性を重視してみるべきだと考える。すなわち、相互補完関係は、片方からの強みの提供と片方からの強みの提供の均衡性・対称性、片方からだけでなく、両方の企業がそれぞれ強みを提供して、相手の弱みを相互補完しているかどうかという互惠性を勘案することが重要であるため、実数ではなく、本研究で提案している数理モデルのように2社間の評点付けを演算によって2極化した形で把握することが望ましいと考える。

図 24 成立・不成立・全体の各企業評点の総合点(満点 40 点)

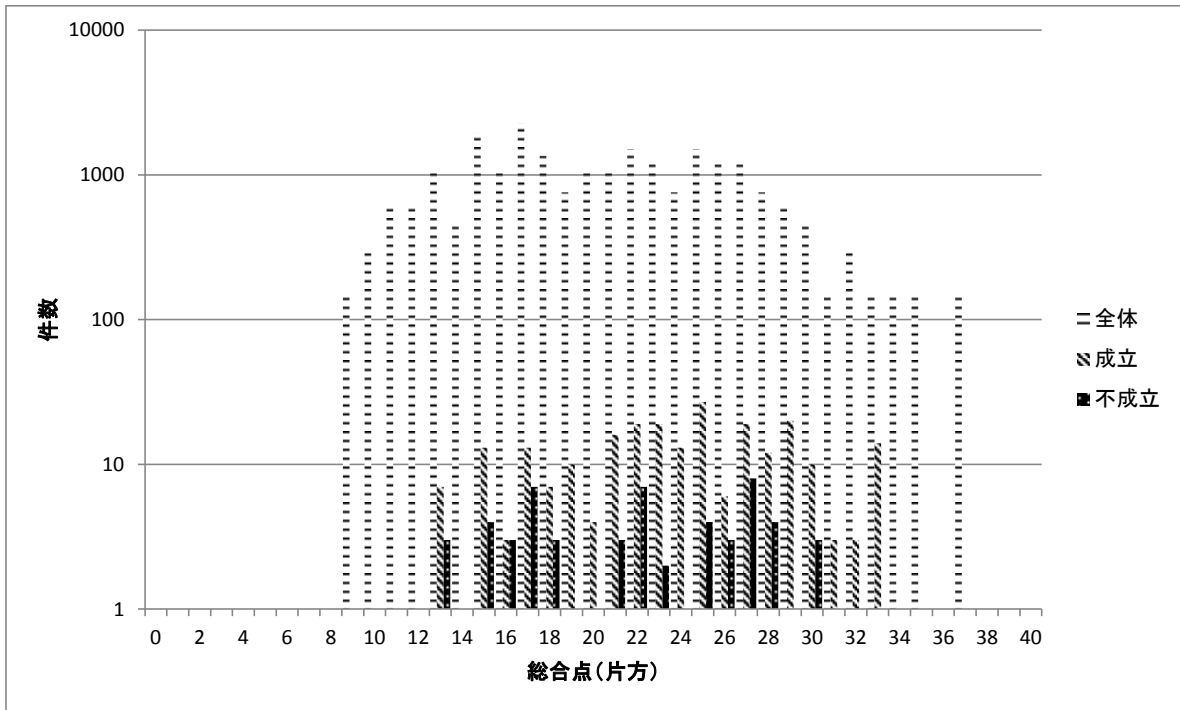
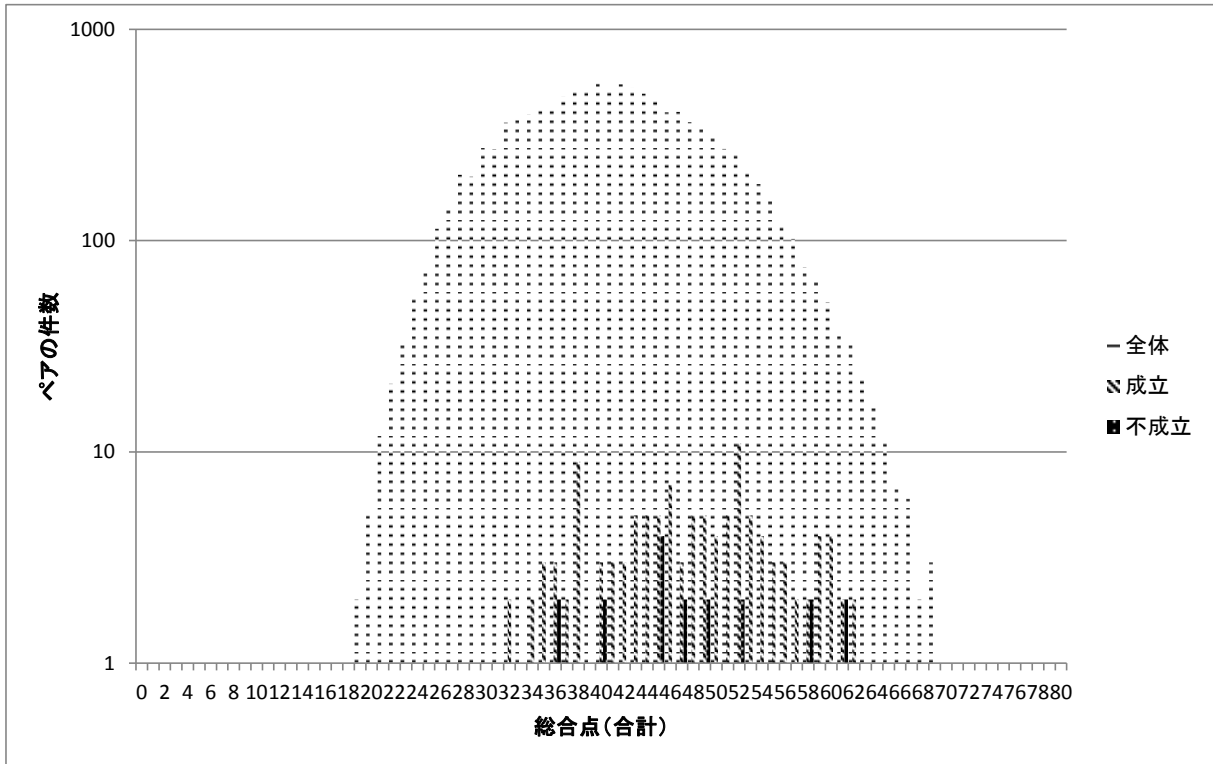


図 25 成立・不成立・全体の 2 社のペアの評点の総合点(満点 80 点)



6.5 相互補完強度係数の外れ値に関する検討

アライアンスの成立・不成立の相互補完強度係数の分布の中で、特徴的な数値となっている組み合わせの内容(どの企業とどの企業か)を検討してみた。

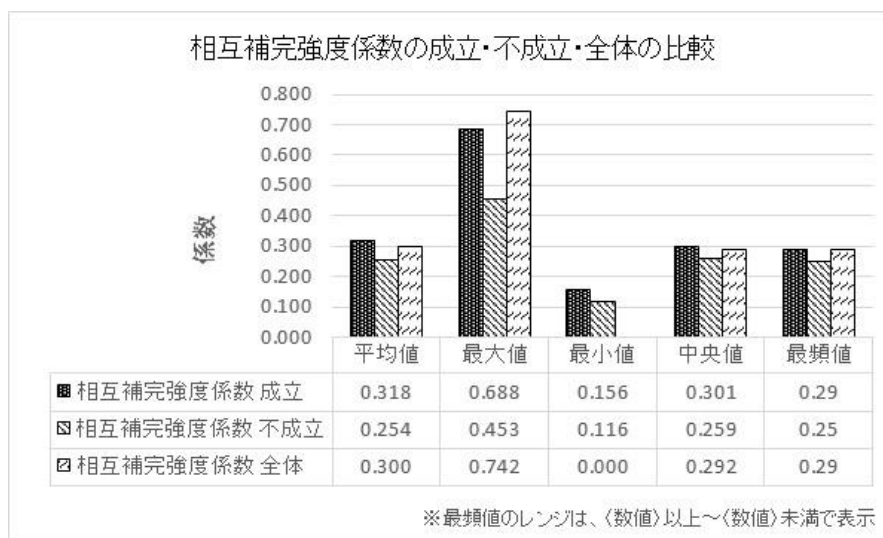
その結果、アライアンスが成立したものの中にも、全体の中でも低い係数のものがあるが、これは、既存事業とは、まったく異なる分野や必要となる経営資源が異なる事業内容での新規事業立ち上げのためのアライアンスの場合であることがわかった。つまり、既存事業をメインとした8つの観点での評点付けでの相互補完での分析をすると、相互補完関係が低いという結果となっていることがわかった。

新規事業の事業構築の場合には、既存事業に基づく企業の評価付けではなく、新規事業として取り組もうとしている事業内容に基づいた評点付けを別途行って、その評点付けに基づいて、検証する必要があることがわかった。

また、相互補完強度係数の成立・不成立・全体の各組み合わせの平均値・最大値・最小値・中央値・最頻値は、次の図26の通りである。

いずれの数値においても、成立が不成立よりも大きい数字となっており、不成立は全体よりも小さい数字となっている。

図26 成立・不成立・全体の相互補完強度係数の平均値・最大値・最小値・中央値・最頻値



6.6 複数の潜在的なアライアンス候補先からの最適なパートナーの選択への利用

ここまで、提案してきたアライアンスの相互補完モデルによって算出できる相互補完強度係数の比較によって、複数の選択肢から相互補完の観点で適切なアライアンス先を選択したり、3社から2社の最適な相互補完の組み合わせを選択したりする意思決定を行うことができる。

ここでは、このモデルを用いることによって、複数の潜在的なアライアンス候補先の中から、最適なアライアンス先を選択してみたい。これは、本モデルを記述した Python 言語のスクリプトを用いた提案したモデルを応用となる。

提案したモデルを利用すれば、アライアンスの形成において、複数の選択可能な企業の組み合わせがある時に、相互補完関係がより強い候補先企業を選択することによって、最適なアライアンスを形成することができる。相互補完強度及び係数をすべての組み合わせについて算出し、それを用いて、アライアンスにおける最適な組み合わせがどれであるかが判断できるようになる。

たとえば、3つの企業、I社、J社、K社があり、それぞれについて、8つの特徴について、1～5段階の評点付けがされているとする。それらの企業は、次のように1次元行列で、表現できる。

$$i = (1, 3, 4, 2, 5, 1, 3, 1)$$

$$j = (4, 1, 1, 3, 1, 5, 3, 1)$$

$$k = (3, 5, 2, 4, 2, 3, 5, 4)$$

これらの3社の企業について、選択可能な組み合わせであるI社とJ社、I社とK社、J社とK社について、1次元行列を相互に差し引きして、相互補完強度関係の強さを算出する。これらの関係については、次のような式によって、求められることとなる。

$$x=i-j$$

$$y=i-k$$

$$z=j-k$$

Python 言語による算出結果として、 x 、 y 、 z の相互補完強度及びその係数の数値は次のようになる。

相互補完強度 x ($=i-j$) $=11.99$

相互補完強度係数 x ($=i-j$) $=0.530$

相互補完強度 y ($=i-k$) $=11.22$

相互補完強度係数 y ($=i-k$) $=0.496$

相互補完強度 z ($=j-k$) $=9.025$

相互補完強度係数 z ($=j-k$) $=0.398$

この数値に従って考えてみると、I 社は、J 社か K 社かの比較では、 x が y より大きいため、それゆえ、I 社は、J 社を、最適なアライアンス先企業として選択すべきと判断できる。

$$x > y > z$$

同じように、J 社にとっては、I 社と K 社を比べた場合、 x が z よりも大きいため、J 社としては、I 社を最適なアライアンス先企業として選択すべきであると判断できる。

同様に、K 社にとっては、I 社と J 社を比べた場合、 y が z よりも大きいため、K 社にとっては、I 社が最適なアライアンス先企業として選択すべきであると判断できる。

さらには、3 社の中での最適なアライアンスの組み合わせの決定としては、 x が、 y や z よりも大きいため、I 社、J 社、K 社の中では、I 社と J 社が 3 つの組み合わせのパターンの中で最適な組み合わせであると判断できる。

このように、複数の潜在的にアライアンス候補先企業の中から、本稿で提案した数理モデルを用いて、相互補完強度係数を算出し、その数値を比較することで、最適なアライアンス先企業を

選択することができる。すなわち、3社のアライアンス候補先企業の中から、数理モデルによって係数を算出することで、他にも選択できるオプションがある場合に、2社の間で、どの組み合わせが双方にとって適しているかを決定づけることができるわけである。

提案するモデルの利用方法としては、さらに、 m の数の潜在的な候補先企業から、アライアンス先企業 n の数の最適なアライアンス先企業を選択するということにも、応用できるであろう。

6.7 相互補完モデルの応用可能性

本稿で提案した相互補完数理モデルは、強み・弱みの評点付けができる企業データには応用可能であり、このモデルの応用可能性は広いと考えられる。

今後、当研究とは違う企業データ、たとえば、TOPIX や日経 225、SandP500 などの株式インデックスに含まれる上場企業データなどで、本稿で提案した数学モデルに当てはめ、各企業の特徴付けをする項目を決めて、評価付けを行い、アライアンスの相互補完強度係数を算出することによる実証研究を行うことができる。

また、本稿では企業間のアライアンスの問題を取り扱っているが、このたび構築した相互補完モデルは、新規事業のチームの構築や建設等の現場での作業チームの構築などの人事の分野、つまり、人と人の組み合わせにも適用可能である。もし、結婚についても、夫婦間の相互補完が強いほうがよい結婚になるとの前提であれば、結婚紹介事業などにおいても、当モデルを活用可能である。

さらには、国と国の同盟関係についても、各国の様々な側面について、強み・弱みの評点付けをすれば、応用可能である。たとえば、日本の戦国時代の同盟関係や第二次世界大戦の同盟関係が本当に相互補完の観点から見た場合、適切なものであったかを検証することが可能となる。今これからの国際的な同盟において、経済面や軍事面などにおいて、どの国とどの国がアライアンスをするのがよいのかという選択において、当該数理モデルは科学的な示唆を与えてくれるであろう。

そういった企業間提携以外の分野においても、本研究で提案した数理モデルが活用・応用されていくことが期待できる。

6.8 相互補完モデルの限界

ここで提案するアライアンスの相互補完モデルは、いくつかの限界を有している。次のような限界がある。

- (1) 相互補完モデルでは、2 社間の相互補完関係が最も強い状況は、双方からの特徴数(企業の評価付けの強み・弱みの項目の数)の半分ずつの数で、最大の長さのベクトルが双方に提供されている時に、最大の相互補完関係となると設定する。すなわち、特徴数の半分の特徴からの最大の大きさの双方向の強みの提供から、最大の相互補完強度を求めるとしている。そのため、特徴数を偶数であれば、2 で割れるため、すぐに利用できるが、特徴数が奇数になる場合には、ダミー項目としての特徴を一つ追加し、特徴数が偶数にする対処が必要となる。あるいは、特徴数が奇数で、このモデルを利用する場合には、2 つのマキシмум・ポイントが存在することを理解しなければならない。そして、その2 点間が線形(リニア)であると想定して、平均を取る形で、相互補完強度を求めることが可能である。特徴数が7つのケースの相互補完強度係数の算出の仕方について、付録3に掲載しているので、参照されたい。
- (2) 相互補完強度及びその係数は、あくまで、各特徴の評点に基づいて算出されるため、評点の付け方次第で、相互補完関係についての評価が変わり、算出される相互補完強度及び係数が左右される。評点付けを今後より一層工夫していくことにより、より精度を高めることができるであろう。
- (3) 実際のビジネスの推進においては、2 社の経営者同士や社員同士が意気投合して事業に取り組めるかといった相性もアライアンスの成立・不成立に影響を与えることが考えられる。そういった経営資源以外の目に見えない要素については、ブランド・信用力や機動的な組織風土については、企業評価の8つの特徴に入っているが、2 社間の考え方やカルチャーなどの相性については、加味されていない。
- (4) 富田(2004)で示された7つのアライアンスのタイプのうち、強いエリアが異なる企業同士の地域的補完や営業エリア(日本国内の地域別や海外)が異なる企業のアライアンス、インターネット販売やインターネットでの販売を行っている会社と行っていない会社のアライアンスは、相

互補完数理モデルでは、取り扱えていない。このタイプのアライアンスは、引き算で演算する相互補完ではなく、足し算になるためである。これについては、第7章にて、加算モデルとして追加する。

(5)外部企業とのアライアンスによって新規事業を立ち上げようとする意欲や、事業を拡大させようという意欲(やる気)の強弱によって、アライアンスによる事業推進がうまくいくかどうかが決まってくる面があり、重要な要因であるが、そういった意欲の面が加味されていない。そういった意欲については、相乗モデルとして、第8章にて、相互補完モデルとして追加する。本研究は、(4)のように加算モデルを追加して発展させるとともに、(5)のように相乗モデルを追加して発展させ、それにより、「アライアンスの相互補完・加算・相乗モデル」として完成させる。

6.9 相互補完モデルのまとめ

第3章にて、企業の強みと弱みを1次元行列と2極のベクトル、そして、最大の相互補完の地点からの距離で表わす相互補完数理モデルを構築し、第4章にて、フロー・インテンシティ及びフロー・バランスの概念を用いて説明した。そして、第5章で紹介した152社のデータを用いて、第6章にて、提案する数理モデルを、Pythonで実装し、実際の企業データを使って、相互補完強度及び係数を算出した。それにより、提案した相互補完数理モデルの正当性が確認できた。

すなわち、提案するモデルによって、アライアンスを行う2社間のアライアンス成立のメカニズムを数学的に表現することが可能となった。また、アライアンスを行う2社間の相互補完関係を2次元のマップ上の点で表現し、数値として演算可能となった。さらには、アライアンスは相互補完が強い時に成立するとのモデルの考え方の正当性を確認することができた。

このように、企業間アライアンスの相互補完関係を表す数理モデルを構築し、2社間の関係性を数値として、演算可能(コンピューティショナル)にしたことが、本モデルを構築・提案した成果である。

第7章 モデルの発展1：アライアンスの加算モデルの構築と検証

7.1 相互補完数理モデルの未対応部分

アライアンスの機能別にみると、富田(2014)に示されているように、次の7つに分けられる。

- (1) 技術開発力(アイデア力)のある企業と営業力のある企業
- (2) 新規事業のシーズ¹⁵のある企業とシーズはないが資金が余っている企業
- (3) 技術系企業¹⁶同士の製品開発の技術補完
- (4) 開発力のある企業と生産力を持っている企業
- (5) 人が余っている企業と人の足りない企業
- (6) 営業先が共通の企業同士の営業先の共同利用
- (7) 強いエリアが違う企業同士の地域的補完

富田・武藤(2015)及び Tomita and Takefuji(2016a, 2016b)では、これらのパターンのうちの(1)～(5)に該当するものをアライアンスの相互補完数理モデルにて取り扱ったが、本稿では、それらの論文で取り扱えなかった(6)及び(7)についてアライアンスの相互補完・加算モデルとして盛り込むこととし、モデルを発展させる。

営業力がある・ないについては、相互補完関係となるが、営業力の特に営業エリアについては、足し算(加算)と考えられる。そのため、相互補完モデルには盛り込むことができていなかった。

営業エリアが異なる企業同士のアライアンスは、相互補完のようにも見えるが、地域的な補完性は、相互補完モデルでの差し引き(引き算)ではなく、加わっていく足し算となる。

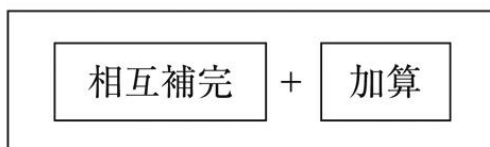
アライアンスの相互補完・加算モデルは、図26のように、相互補完モデルに加算モデルを加える形となる。

なお、本稿の第7章と第8章の内容は、富田・武藤(2016)で、主として掲載された内容となっている。

¹⁵新規事業の事業構築のもとになるアイデア、企画、新技術などを指す。

¹⁶技術を用いて製品を開発・製造するメーカーや、技術を競争優位のメインとしている企業のことを指す。

図 27 アライアンスの相互補完・加算モデル



相互補完モデルまでの段階については、富田・武藤(2015)及び Tomita and Takefuji(2016a, 2016b)、本稿の第3、4、5、6章では、152社の企業についての強み・弱みの評点付けについては、先に述べたように、①営業力がある・なし、②技術力がある・なし、③アイデア力がある・なし、④資金力がある・なし、⑤人材がいる・いない、⑥生産力がある・なし、⑦ブランド・信用がある・なし、⑧機動的な組織風土である・ないという8つの項目(特徴)について、1~5の5段階評価を行っており、加算モデルの構築においても、後述の相互補完・加算・相乗モデルにおいても、そのまま踏襲した。

繰り返しとなるが、ここで用いている企業データや評点付けの仕方については、マッチング数理モデルを構築するにあたって、データの一例として用いており、この評点付けが実態を表すにあたって、絶対的なものではない。

7.2 日本国内の地域、ネット販売、海外販売

本研究で取り上げている152社の企業について、日本国内において、都道府県を次の5つに分け、各企業が営業展開をしている地域か否かの評価付けを行った。地域分類は以下となる。①首都圏(東京、神奈川、埼玉、千葉)、②関西エリア(京都、大阪、兵庫、奈良、滋賀)、③東海エリア(愛知、岐阜、静岡、山梨、長野)、④北陸エリア(石川、福井、富山、新潟)、⑤その他地域(①~④に含まれていない地域すべて)。

加えて、インターネット販売をしているかどうかの有無を、152社について判別をした。インターネット販売については、Webでのプロモーションだけでなく、Web上での受注をしているかどうか

かで判断をした。

さらに、海外での営業展開をしているか否かでの判別も行った。海外での営業展開については、商社等を通じてではなく、直接的に自社で海外営業をしているか否かで判断をした。

152社の企業について、該当するエリアについては、表5の配分でスコアをカウントした。すなわち、営業展開をしているエリアは1、していないエリアは0を入れ、1については、配分されたスコアが加算されることとした。重なっているエリアについては、営業力としては割り引くとの考え方もできるが、やはり、営業力については加算になるとの考えから、加算されていく形とした。

表5 営業エリアとインターネット販売の加算スコア

	首都圏	関西エリア	東海エリア	北陸エリア	その他地域	ネット販売	海外
県内総生産(名目、平成24年度)	161,673,053	74,231,964	67,805,216	20,587,764	175,860,233	-	-
全国の中での比率	32.3%	14.8%	13.6%	4.1%	35.2%	-	-
モデル構築のために整数に単純化したスコア	8	4	3	1	9	12	5
日本国内のスコアの合計:25						※首都圏とのバランスでの今回の規定	

7.3 加算モデルの評点の付け方

前述のように、日本国内5区分、インターネット販売の有無、海外販売の有無の7つの観点で、そのエリア・手法にて行っているか否かで区分をただけでなく、各エリア・手法の重み付けを行って、相互補完・加算モデルとしてのスコアを付けた。相互補完・加算モデルにおける各エリア及び手法のスコアは、前述の図表6の通りである。

日本国内については、各エリアの営業エリアとしての重要さをなんらかの形で把握することが必要であり、人口や面積などでウエイトを付けることもできるが、ここでは、内閣府が公表している県別の県内総生産(名目、平成24年度)を用いて、①～⑤の各地域の重みを算出して、各地域のスコアを規定した。エリア分けにおいては、このたび用いた企業データの中で、営業展開している企業が少ない北海道や九州は、その他エリアに含めた。逆に、当該データには金沢など、北陸企業が相対的に多く含まれていたため、北陸エリアは、エリア分けにおいて、取り上げた。また、インターネット販売のスコアと海外販売のスコアについては、首都圏のスコアとの balan

スを勘案して、各手法のスコアを規定した。ここで、ウエイト付けを行いやすくするため、32.3%、14.8%、13.6%、4.1%、35.2 を割りやすい整数が4となるため、4で割って、ウエイト付けを1桁の整数とする丸め計算を便宜上、行ったことを付記しておく。

加算モデルにおいて、加算スコアの算出のためのベースとするデータを何にするか、各エリアの分け方をどのようにするか、ネット販売のウエイトを地域エリアとのバランスをどのようにするかは、今後、このモデルを利用するユーザーが自由に変更・調整可能である。

7.4 相互補完・加算モデルの数式化

このような営業展開しているエリア及び手法によってスコア付けをしたものを、本稿の第3章～第6章で提案された相互補完数理モデルに、加算モデルとして追加した。

相互補完モデルへの追加にあたり、相互補完モデルでの一つの項目に割り当てられている数値を、最大の相互補完強度を項目数（特徴数）で割ることで算出し、その一つの項目に割り当てられた数値の何倍の重みで、加算モデルを相互補完モデルに追加するかを、 $ad(w)$ として規定することとした。

その上で、A社とB社のアライアンスにおいて、A社の加算スコア（営業展開エリアのスコアのこと）の合計（ $\Sigma \text{area}(a)$ ）とB社の営業展開エリアのスコアの合計（ $\Sigma \text{area}(b)$ ）を加算した。それを加算スコアの最大値で割り、当該組み合わせの加算モデルに配分された点数の何%を、相互補完モデルに足し合わせるのかを計算した。

アライアンスの相互補完・加算モデルの式は下記となる。この式によって求められる数値を、アライアンスの加算値と呼ぶこととする。

アライアンスの相互補完・加算モデルの加算値 APの式

$$AP = \frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} - \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ plus}\right)^2 + \left(\frac{-(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ minus}\right)^2}}{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2}} \times \text{ad}(w) \times \frac{\Sigma \text{ area}(a) + \Sigma \text{ area}(b)}{\Sigma \text{ area}(t)}$$

なお、s(max)は評点付けの最大値、s(min)は評点付けの最小値、len(c)は評点付けの項目数(特徴数)、Σ plusは正の整数の総和、Σ minusは負の整数の総和、Σ area(a)はA社の営業エリアの加算スコアの合計、Σ area(b)はB社の営業エリアの加算スコアの合計、ad(w)は相互補完強度の算出の一つの項目の何倍の重みで加算モデルを追加するかを規定する数値、Σ area(t)は営業エリアの加算スコアの最大値である。

上記の式で求められるアライアンスの加算値を、相互補完モデルの最大値に加算モデルの最大値を足し合わせた相互補完・加算モデルの最大値で割り、0～1 の間の数値となるように正規化した数値を、アライアンスの加算係数と呼ぶこととする。

アライアンスの加算係数を算出する式は、次の通りである。

アライアンスの相互補完・加算モデルの加算係数APC

$$APC = \frac{S + A}{T(AP)}$$

$$S = \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} - \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ plus}\right)^2 + \left(\frac{-(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ minus}\right)^2}$$

$$A = \left(\frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2}}{\text{len}(c)} \times \text{ad}(w) \right) \times \frac{\Sigma \text{ area}(a) + \Sigma \text{ area}(b)}{\Sigma \text{ area}(t)}$$

$$T(AP) = \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} + \frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2}}{\text{len}(c)} \times \text{ad}(w) \times \frac{\Sigma \text{ area}(t)}{\Sigma \text{ area}(t)}$$

なお、s(max)は評点付けの最大値、s(min)は評点付けの最小値、len(c)は評点付けの項目数(特徴数)、Σ plusは正の整数の総和、Σ minusは負の整数の総和、Σ area(a)はA社の営業エリアの加算スコアの合計、Σ area(b)はB社の営業エリアの加算スコアの合計、ad(w)は相互補完強度の算出の一つの項目の何倍の重みで加算モデルを追加するかを規定する数値、Σ area(t)は営業エリアの加算スコアの最大値である。

たとえば、冒頭の例のX社とY社について、地域及び手法の評点が下記だったとする。（首都圏、関西、東海、北陸、その他、ネット、海外）について、該当すれば1、該当しなければ0と、1次元行列にて表示する。

A社 (1, 1, 0, 0, 1, 0, 1)

B社 (1, 0, 0, 1, 0, 1, 0)

各エリアの配分は、(首都圏、関西、東海、北陸、その他、ネット、海外) = (8, 4, 3, 1, 9, 12, 5) であるため、A社の加算スコアは26、B社の加算スコアは21で、この組み合わせの加算スコアは47となる。

ネット販売を、この加算モデルの構築・算出において、含めないということもできるが、今日、

ネット販売の能力は、非常に重要になっており、また、このたびの企業データの中には、ネット販売を得意とする企業がいくつも含まれていたため、加算スコアの項目として追加し、相対的に大きめのウェイトとした。この配点も自由に変更可能である。

同じく、海外での営業展開も、加算モデルの構築・算出において含めないということもできるが、昨今、日本国内市場の成熟化・縮小に伴い、海外での販売を伸ばそうとする企業も増えており、その面で、海外での営業展開をすでに行っているか否かは、企業の強み・弱みの判定において、意味があると考え、加算モデルの項目として追加した。削除して、日本国内だけで検討することも可能である。

ここでの例としている A 社と B 社における加算モデルにおける $ad(w)$ を 3 として、加算値を実際に計算すると、18.43 となる¹⁷。また、それを正規化した加算係数は 0.5924 となる¹⁸。

7.5 152 社の企業データについての Python 言語での演算結果

上記の式における $ad(w)$ については、本研究では 3 として、相互補完・加算モデルの数値（ここでは、加算値と呼ぶ）を Python 言語によって算出を行った。

152 社の企業データにおけるアライアンス成立の組み合わせ（152 社と 152 社の間にて、121 件存在）とアライアンス不成立の組み合わせ（同じく 30 件存在）のアライアンスの加算係数のグラフは、図 28 である。また、アライアンス成立・不成立・全体の加算係数の平均値、最大値、最小値、中央値、最頻値については、表 6 の通りであり、そのグラフは、図 29 である。

¹⁷ 加算値 AP の計算は次のようになる。まず、IV の相互補完モデルの解説に掲載しているように、相互補完強度の最大値は、 $\sqrt{((16-0)^2 + (-16+0)^2)}$ で計算でき、正確には 22.627416998 となる。四捨五入で 22.63、切り捨てで 22.62 となり、この A 社・B 社の相互補完強度 13.68 となる。

この相互補完強度 13.68 をもとに加算値を計算することになり、例としている A 社、B 社の地域の評点は、

A 社 (1, 1, 0, 0, 1, 0, 1)

B 社 (1, 0, 0, 1, 0, 1, 0)

としており、地域ごとの配点 (8, 4, 3, 1, 9, 12, 5) と設定しているため、A 社=26、B 社=21 で、加算スコアは 47 になる。この加算スコアを、加算モデルの式に当てはめると、

$AP = 13.68 + 22.62/8 * 3 * 47/84 = 13.68 + 4.75 = 18.43$ となる。これによって、加算値が求められる。

¹⁸ 加算値の最大値は、 $T(AP) = 22.62 + 22.62/8 * 3 * 84/84 = 31.11$ となるため、例の A 社・B 社の加算係数は、 $18.43/31.11 = 0.5924$ となる。

図 28 アライアンス成立・不成立の加算係数の分布

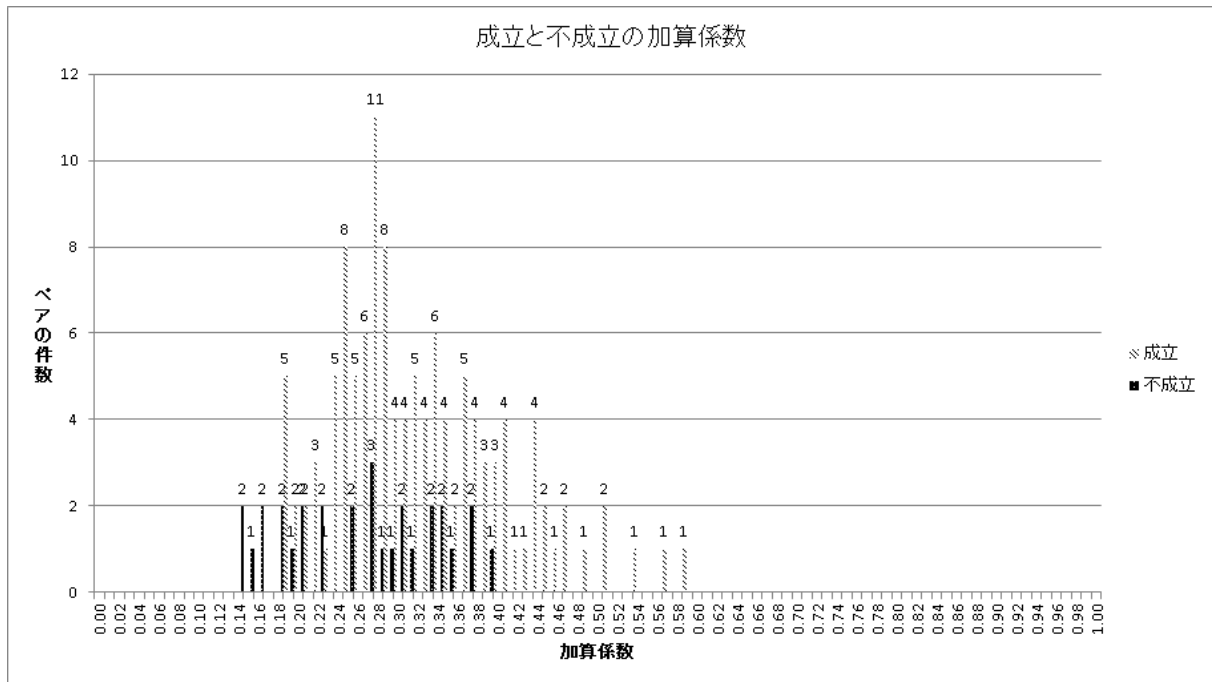
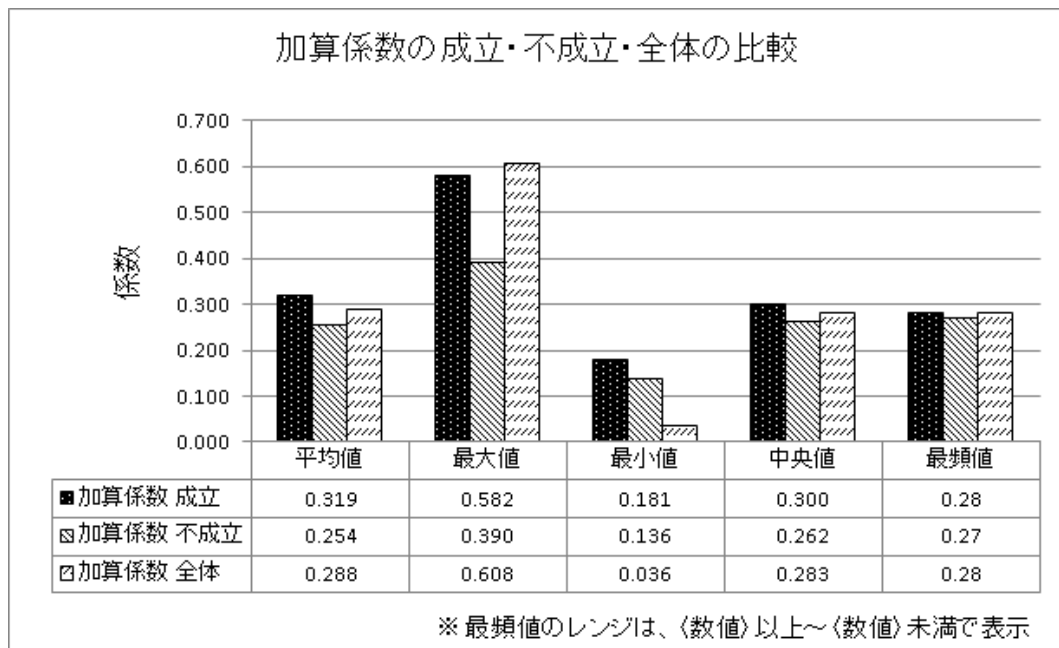


表 6 加算係数の成立・不成立・全体の平均値、中央値、最大値、最小値、最頻値

	加算係数		
	成立	不成立	全体
平均値	0.319	0.254	0.288
最大値	0.582	0.390	0.608
最小値	0.181	0.136	0.036
中央値	0.300	0.262	0.283
最頻値	0.28	0.27	0.28
最頻値件数	11	3	693
最頻値個数	1	1	1
データ数	121	30	11476

図 29 加算係数の成立・不成立・全体の比較



以上のように、アライアンスの加算係数を算出することができた。これにより、アライアンスの相互補完・加算の数理モデルが機能することが確認できた。

相互補完で説明付けられるアライアンスの機能別パターンだけでなく、営業エリア（インターネット販売の有無も含む）の足し合わせとなるアライアンス成立のメカニズムを、数理モデルに組み込むことができた。

7.6 相互補完・加算モデルのまとめ

第3～6章で提案されたアライアンスの相互補完数理モデルで盛り込むことができていなかった営業エリアが足し算となる観点の加算モデルを第7章で追加することができた。Python 言語で実装し、その数理モデルの正当性を確認できた。これにより、営業エリアの加算を数理モデルとして表現し、その数値（加算値及び加算係数）を算出することが可能となった。

第8章 モデルの発展2：アライアンスの相乗モデルの構築と検証

8.1 相乗モデルの追加 ～意欲の重要性と影響～

次に、アライアンスによる事業展開に対する意欲をモデルに取り入れることとする。

筆者は、外部企業とのアライアンスによって新規事業を立ち上げようとする意欲や、事業を拡大させようという意欲の強弱によって、アライアンスによる事業推進がうまくいくかどうかが決まってくる面があると考える。

本稿で提案されたアライアンスの相互補完数理モデルでは、意欲の面が加味されていないが、2社間において行われるアライアンスにおいては、その成立に向けて、片方の企業に意欲があっても、もう一方の企業に意欲がなければ、アライアンスの成立が難しくなると推測される。両方の企業に意欲があれば、さらに成立がしやすくなると推測され、逆に、両方の企業に意欲がなければ、アライアンスは成立がさらに困難になる。

相互補完モデルによる数値（アライアンスの相互補完強度）に加算モデルの数値（アライアンスの加算値）を足し合わせたものに、意欲の重み付けをかけることにより、意欲に関する要素を数理モデルに取り入れ、アライアンスの相互補完・加算・相乗モデルを構築する。図に表わすと図30のようになる。

図30 相互補完・加算・相乗モデル

$$\left(\boxed{\text{相互補完}} + \boxed{\text{加算}} \right) \times \boxed{\text{相乗}}$$

8.2 相乗モデルでの評点の付け方

152社の企業について、アライアンスによる新規事業展開への意欲を0点～100点で、評点を付けた¹⁹。このたびの152社の意欲の評点付けの平均は、62.8点となっている。

¹⁹ これについても、前述の通り、筆者の所属する会社の別の者も評価付けを行い、その比較に

意欲の中には、売上拡大への意欲、新規事業構築への意欲、外部とのアライアンスへの意欲といった多少異なる角度からの意欲が存在するが、事業展開を前向きに推し進めようとする意欲、特に、アライアンスによる新事業展開への意欲という観点をメインに評点付けを行った。

意欲の強弱の重み付けとしては、3パターンを試してみた。すなわち、意欲によつての増減が2割の0.8~1.2のレンジ、4割の0.6~1.4、6割の増減する0.4~1.6のレンジである。これらのレンジにおいては、0~100点での評点付けの50点が1となる。

これらの重み付けでの相乗モデルの追加を行った。

8.3. 相互補完・加算・相乗モデルの数式化

相乗モデルについては、アライアンスの相互補完モデルに加算モデルを足し合わせたものに、重み付けをかけることになるため、算出する一般的な数式は次のように表現できる。なお、この式によって求められる数値をアライアンスの相乗値と呼ぶこととする。

アライアンスの相互補完・加算・相乗モデルによる相乗値AMの式

$$AM = \left(\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} - \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ plus}\right)^2} + \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ minus}\right)^2} \right. \\ \left. + \frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2}}{\text{len}(c)} \times \text{ad}(w) \times \frac{\Sigma \text{ area}(a) + \Sigma \text{ area}(b)}{\Sigma \text{ area}(t)} \right) \times \frac{\text{mot}(a)}{100} \times \text{mot}(w) \times \frac{\text{mot}(b)}{100} \times \text{mot}(w)$$

なお、s(max)は評点付けの最大値、s(min)は評点付けの最小値、len(c)は評点付けの項目数(特徴数)、Σ plusは正の整数の総和、Σ minusは負の整数の総和、Σ area(a)はA社の営業エリアの加算スコアの合計、Σ area(b)はB社の営業エリアの加算スコアの合計、ad(w)は相互補完強度の算出の一つの項目の何倍の重みで加算モデルを追加するかを規定する数値、Σ area(t)は営業エリアの加算スコアの最大値、mot(a)はA社の意欲のスコア、mot(b)はB社の意欲のスコア、mot(w)は意欲の重み付けの度合いである。

より、総合的に会社組織として決定した。意欲の評点の付け方において、バイアスがかかるとの懸念もあるだろうが、むしろ、バイアスをかけて評点付けをしてみることも、今後の研究課題として試してみることも必要になろう。

上記の式で求められるアライアンスの相乗値を、0～1の間の数字となるように、相互補完強度の最大値と加算値の最大値を足し合わせたものに、最大の意欲の重み付けである $\text{mot}(w)$ をかけた数値で割ったものを、アライアンス係数と呼ぶこととする。算出式は、次のようになる。

アライアンスの相互補完・加算・相乗モデルによるアライアンス係数 AC の式

$$AC = \frac{(S + A) \times M}{T(AC)}$$

$$S = \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} - \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ plus}\right)^2} + \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2} - \Sigma \text{ minus}\right)^2}$$

$$A = \frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2}}{\text{len}(c)} \times \text{ad}(w) \times \frac{\Sigma \text{ area}(a) + \Sigma \text{ area}(b)}{\text{area}(t)}$$

$$M = \frac{\text{mot}(a)}{100} \times \text{mot}(w) \times \frac{\text{mot}(b)}{100} \times \text{mot}(w)$$

$$T(AC) = \sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2} + \frac{\sqrt{\left(\frac{(s(\max) - s(\min)) \times \text{len}(c)}{2}\right)^2 \times 2}}{\text{len}(c)} \times \text{ad}(w) \times \frac{\text{area}(t)}{\text{area}(t)} \times \text{mot}(w)^2$$

なお、 $s(\max)$ は評点付けの最大値、 $s(\min)$ は評点付けの最小値、 $\text{len}(c)$ は評点付けの項目数(特徴数)、 $\Sigma \text{ plus}$ は正の整数の総和、 $\Sigma \text{ minus}$ は負の整数の総和、 $\Sigma \text{ area}(a)$ はA社の営業エリアの加算スコアの合計、 $\Sigma \text{ area}(b)$ はB社の営業エリアの加算スコアの合計、 $\text{ad}(w)$ は相互補完強度の算出の一つの項目の何倍の重みで加算モデルを追加するかを規定する数値、 $\Sigma \text{ area}(t)$ は営業エリアの加算スコアの最大値、 $\text{mot}(a)$ はA社の意欲のスコア、 $\text{mot}(b)$ はB社の意欲のスコア、 $\text{mot}(w)$ は意欲の重み付けの度合いである。

たとえば、前述のA社とB社の例において、各社の意欲がA社=90、B社=40とすると、各重み付けの相乗値を算出すると下記となる。

0.8～1.2の相乗の場合 67.57

0.6～1.4の相乗の場合 73.69

0.4～1.6の相乗の場合 79.03

さらに、その相乗値を正規化したアライアンス係数は、下記となる。

0.8~1.2 の相乗の場合 0.440

0.6~1.4 の相乗の場合 0.352

0.4~1.6 の相乗の場合 0.289

8.4 152 社の企業データについての Python 言語での演算結果

上記の式を、Python 言語で実装して、アライアンス係数を算出した。

0.8~1.2 の相乗のアライアンス成立・不成立の組み合わせのアライアンス係数のグラフが図表 31 である。アライアンス成立の組み合わせのアライアンス係数は、相対的に高い傾向があり、アライアンス不成立の組み合わせのアライアンス係数は、相対的に低い傾向があると言える。また、図 32 として掲載するように、重み付けをより強くした 0.6~1.4 の相乗のアライアンス成立・不成立の組み合わせのアライアンス係数、そして、図 33 として掲載するように、0.4~1.6 のレンジでのアライアンス成立・不成立の組み合わせのアライアンス係数も同様に算出した。

さらに、0.8~1.2、0.6~1.4、0.4~1.6 の相乗でのアライアンス成立・不成立のアライアンス係数の平均・最大値・最小値のまとめたものが、表 7 である。アライアンス成立のアライアンス係数は、どの相乗のパターンにおいても、また、平均値、最大値、最小値のいずれにおいても、アライアンス不成立のアライアンス係数よりも、高い数値となっている。相乗の度合いを大きくしたほうが、アライアンス成立とアライアンス不成立の割合が大きくなっている。アライアンス成立の組み合わせの係数の平均値とアライアンスの不成立の組み合わせの係数の平均値の割合（成立/不成立）を、成立/不成立比率と呼ぶこととし、この数値がモデルの正当性を評価する一つの指標として、参考までに用いてみる。成立/不成立比率の数値が高いほど、モデルの正当性が高いということとなる。

アライアンスの相互補完数理モデルでのアライアンス成立の相互補完強度係数の平均値 0.31 とアライアンス不成立の相互補完強度係数の平均値 0.253 の割合（成立/不成立）は 1.225 であった。本章で提案しているアライアンスの相互補完・加算・相乗モデルでは、アライアンス成立の

アライアンス係数の平均値 0.319 と、アライアンス不成立のアライアンス係数 0.269 の割合（成立/不成立）は 1.186 となった。

このたび、アライアンス相互補完・加算・相乗モデルに発展させたことで、より一層、アライアンス成立とアライアンス不成立の差が付いている。

なお、アライアンス係数が低くても、成立している組み合わせがある理由は、本研究の評点付けは既存事業についての評価であるが、既存事業とは異なる事業領域・方向性での新規事業立ち上げでのアライアンスの場合、既存事業の面での評点付けでは、必ずしもフィットしないためである。他方、アライアンス係数が高くても、不成立となっている組み合わせがいくつか存在している理由は、相手の事業や技術の魅力を理解することができなかつたり、相性が合わなかつたりしたケースがあるためである。これらのことは、今後の研究課題となる。

図 31 0.8~1.2 の相乗でのアライアンス成立・不成立のアライアンス係数の分布

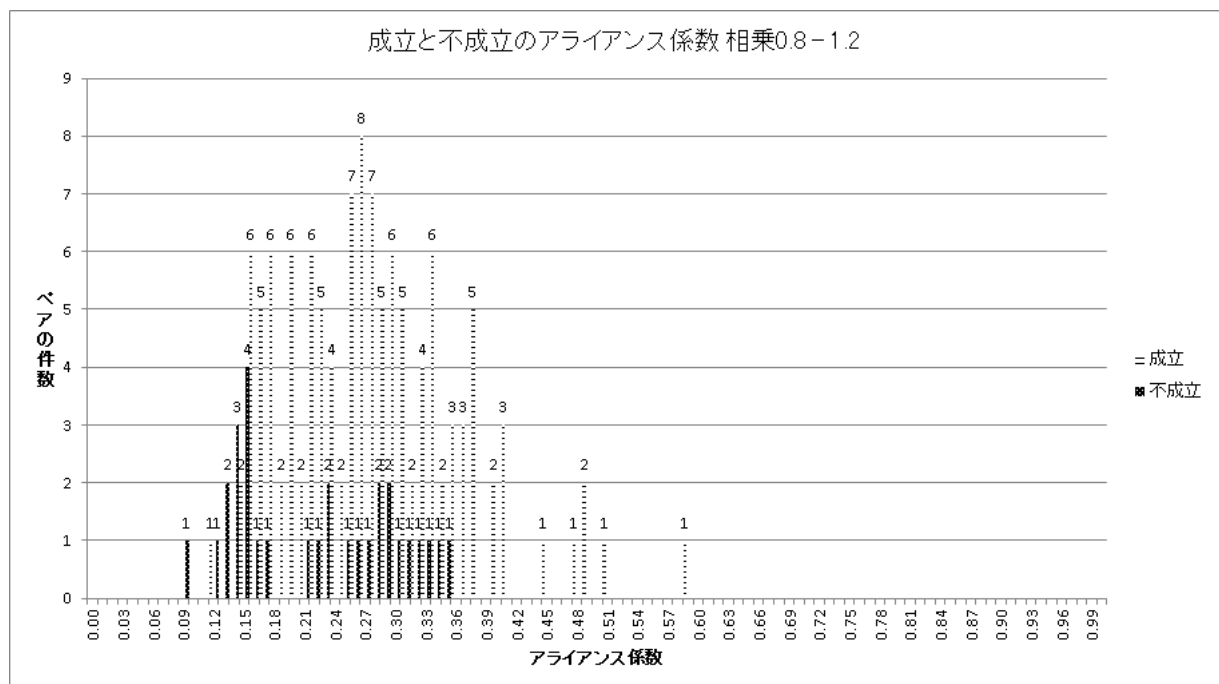


図 32 0.6~1.4 の相乗でのアライアンス成立・不成立のアライアンス係数の分布

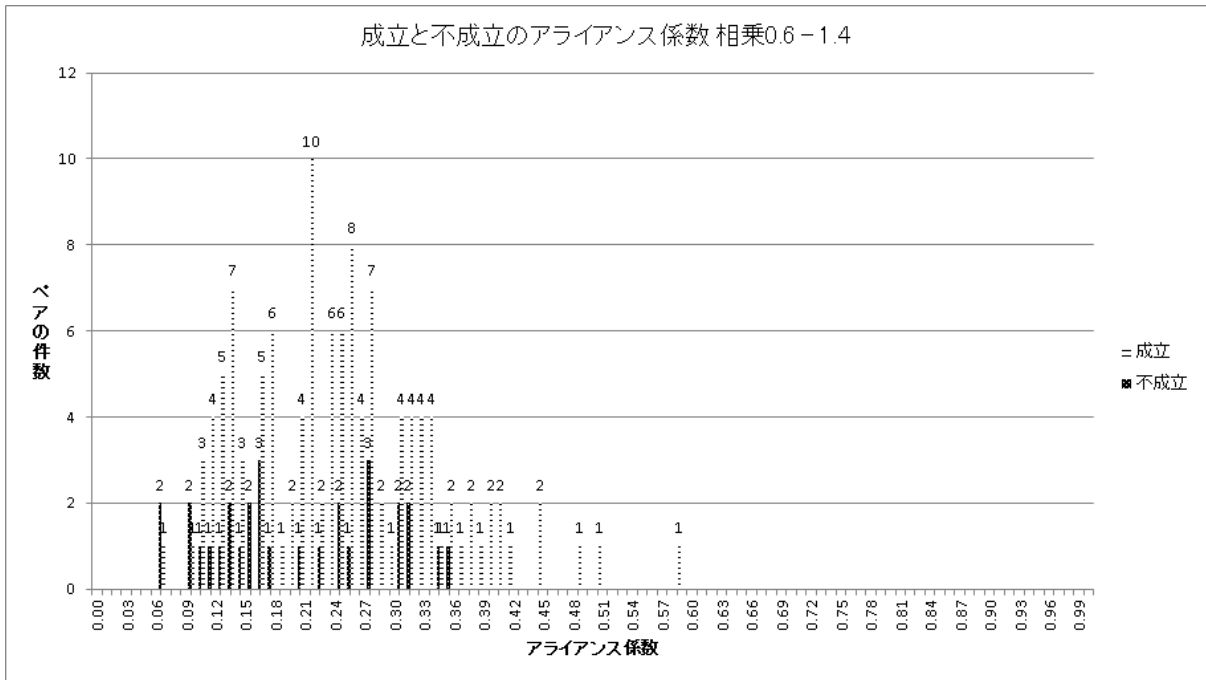
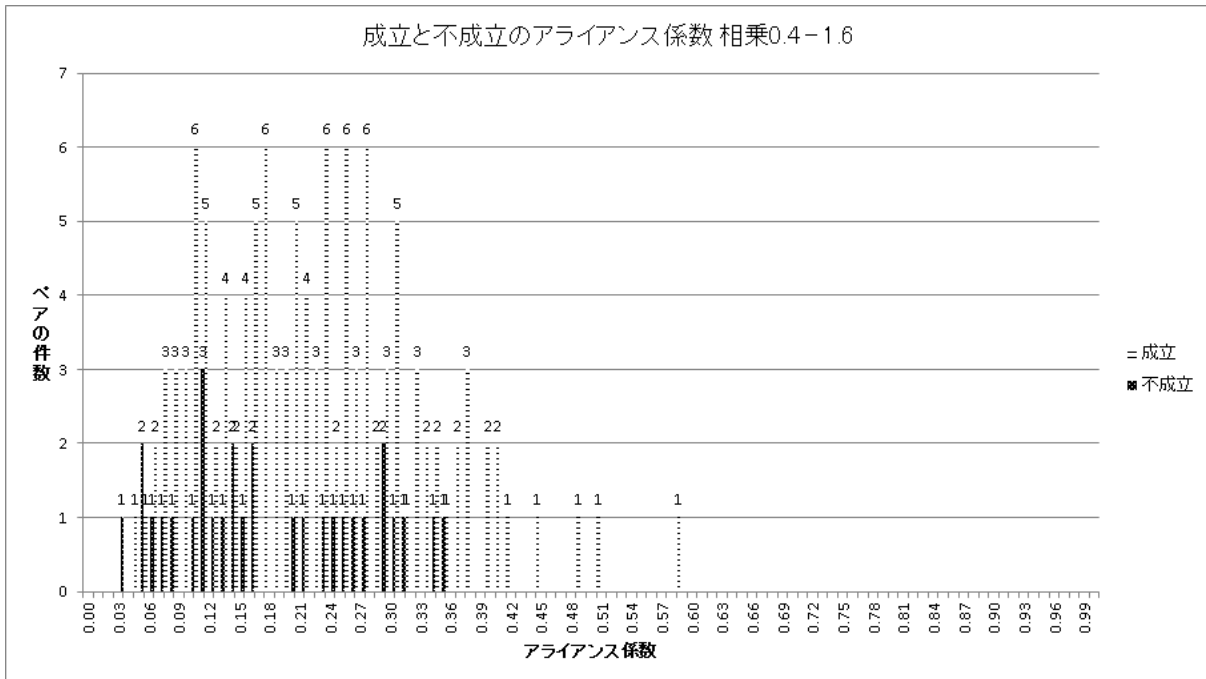


図 33 0.4~1.6 の相乗でのアライアンス成立・不成立のアライアンス係数の分布



次に、0.8~1.2、0.6~1.4、0.4~1.6 の相乗でのアライアンス成立・不成立のアライアンス係数の平均値、最大値、最小値、最頻値の数値の表にまとめたものが、表 7 である。0.8~1.2、0.6

～1.4、0.4～1.6の相乗モデルでのアライアンス係数において、いずれも、最頻値が複数出てきている。これは、アライアンス係数の分布が正規分布でないことを示しており、これは、当該データにおいて、正規分布を前提とした有意検定などは有効ではないことを示している。

アライアンス係数の平均値の重み付けごとにグラフにしたものが図34である。相乗モデルでの重み付けにおいては、重みをより大きくかけると、アライアンス係数が下がっていく傾向が見られる。これは、意欲（やる気）の度合いをパラメーターとして追加したことによって、マッチングの状況が変化し、重みが強いと、加算モデルまでよりも、マッチングの強さが低下することを示している。これは、相乗の要素が加わったことの影響である。また、アライアンス係数の中央値の重み付けごとにグラフにしたものが図35である。平均値と同様の変化が見られる。

表7 0.8～1.2、0.6～1.4、0.4～1.6の相乗でのアライアンス成立・不成立のアライアンス係数の平均値、最大値、最小値、最頻値の数値の表

	アライアンス係数(0.8-1.2)			アライアンス係数(0.6-1.4)			アライアンス係数(0.4-1.6)			
	成立	不成立	全体	成立	不成立	全体	成立	不成立	全体	
平均値	0.277	0.216	0.207	0.249	0.192	0.168	0.228	0.175	0.142	
最大値	0.743	0.522	0.743	0.743	0.522	0.743	0.743	0.522	0.743	
最小値	0.108	0.085	0.036	0.064	0.048	0.021	0.037	0.021	0.009	
中央値	0.224	0.199	0.183	0.208	0.180	0.148	0.186	0.166	0.118	
最頻値	0.200	0.110	0.160	0.170	0.180	0.110	0.110	0.190	0.060	
最頻値件数	8	4	805	10	3	675	7	3	713	
最頻値個数	3	1	1	1	2	1	4	1	1	
データ数	121	30	11476	121	30	11476	121	30	11476	
	複数の最頻値				複数の最頻値			複数の最頻値		
	0.19-0.20				0.17-0.18			0.10-0.11		
	0.20-0.21				0.18-0.19			0.11-0.12		
	0.21-0.22							0.14-0.15		
								0.18-0.19		
	※数値のレンジは、いずれも、数値以上～数値未満で表示									

図 34 アライアンス係数の相乗の重み付けによる平均値の変化のグラフ

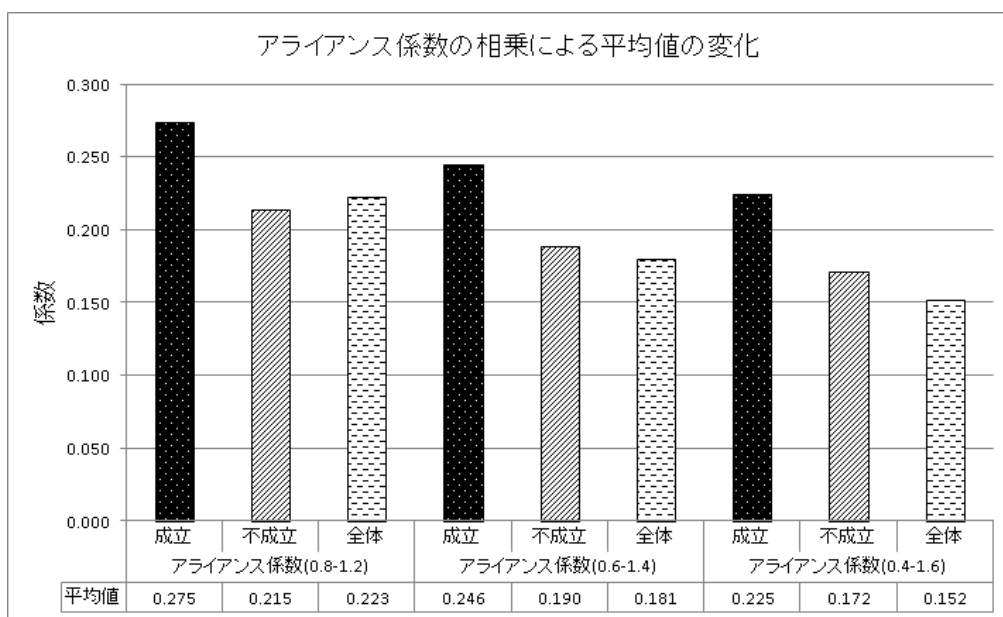
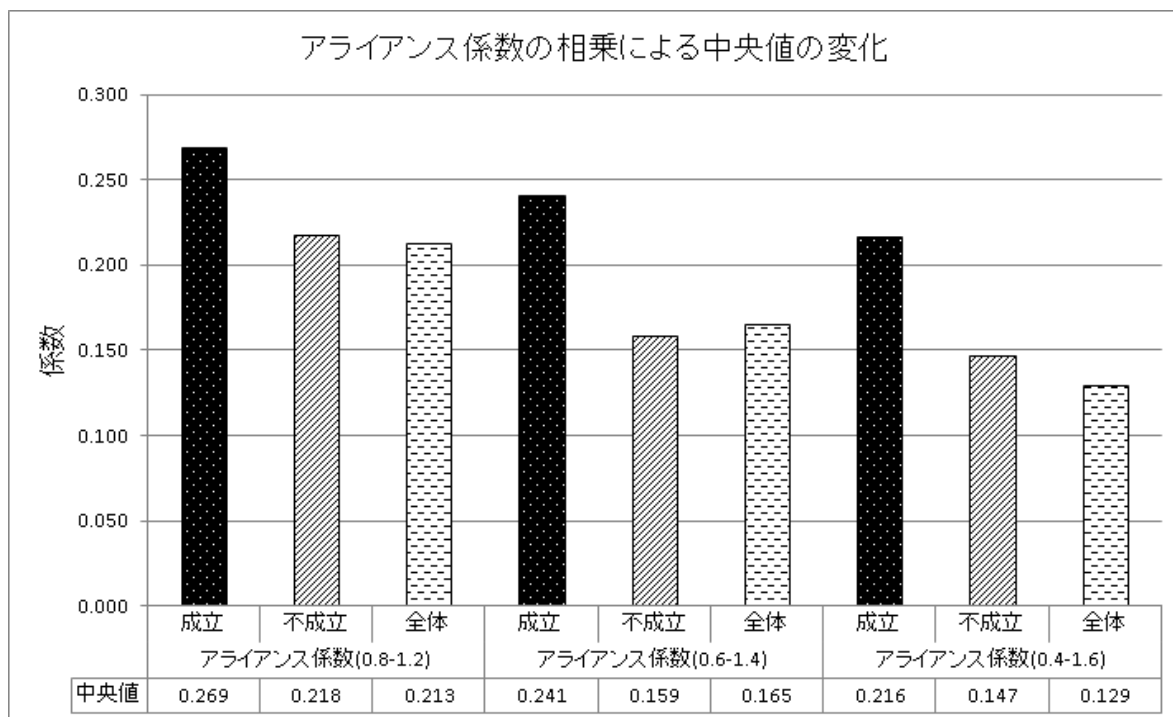


図 35 アライアンス係数の相乗の重み付けによる中央値の変化のグラフ



さらに、0.8～1.2の相乗での152社の企業データのすべての組み合わせ（全体）のアライアンス係数のグラフが図36、0.6～1.4の相乗での152社の企業データのすべての組み合わせ（全体）のアライアンス係数のグラフが図37、0.4～1.6の相乗での152社の企業データのすべての組み合

わせ（全体）のアライアンス係数のグラフが図 38 である。

意欲の度合いについて、弱い重み付けの 0.8~1.2 の相乗から、0.6~1.4 の相乗、そして、0.4~1.6 の相乗へと強い重み付けにすると、分布の山が左に偏ってきている。これは、意欲が低い企業の影響がより大きく出てくるものと考えられる。意欲の平均が 62.8 点（満点 100 点）であったが、平均に満たない企業で、大きく 1 を下回っていた企業があったためではないかと推測される。これは、たとえば、意欲の評価が 0.8 の企業と意欲の評価が 1.2 の企業の組み合わせの場合、重みが 0.96 となり、アライアンス係数が下がる方向になるためである。

なお、ここで、統計学の有意検定を行うことも検討できるが、これらの係数の分布は、正規分布ではないため、一般的な有意検定を行うことは適さないと考えるため、グラフでの視覚的な判断及び平均値・中央値・最頻値の相乗の重み付けの度合いでの比較での視覚的な検討に留めることとする。

図 36 0.8~1.2 の相乗でのすべての組み合わせのアライアンス係数

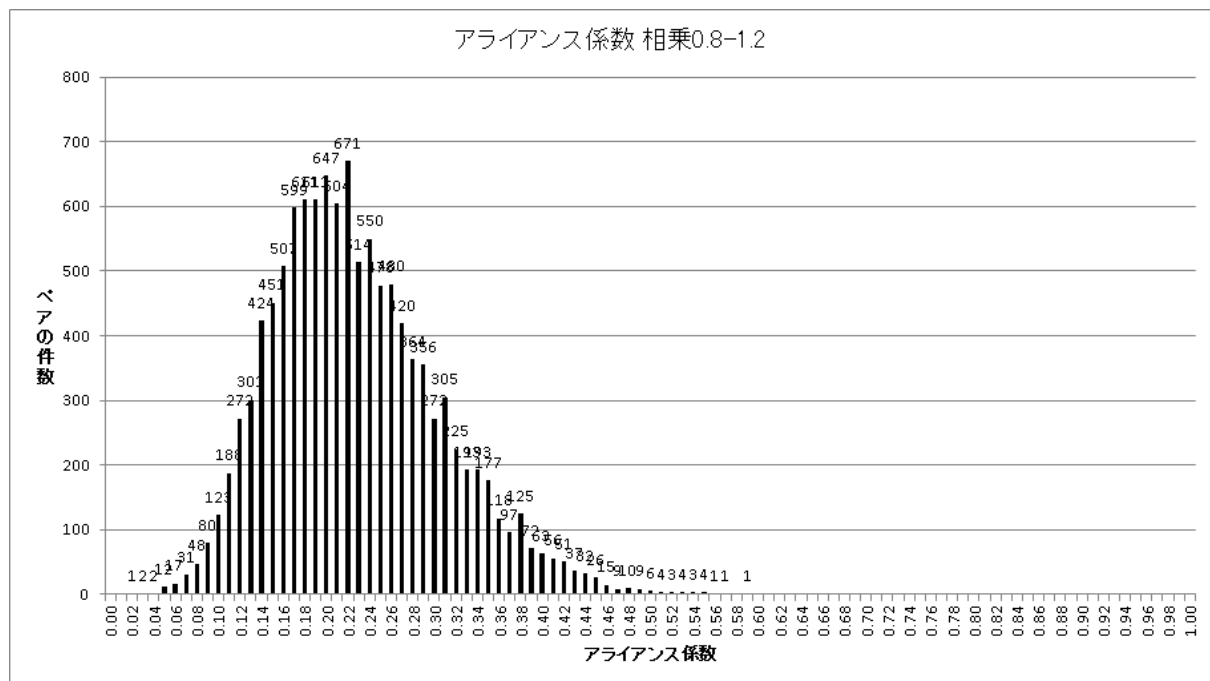


図 37 0.6~1.4 の相乗でのすべての組み合わせのアライアンス係数

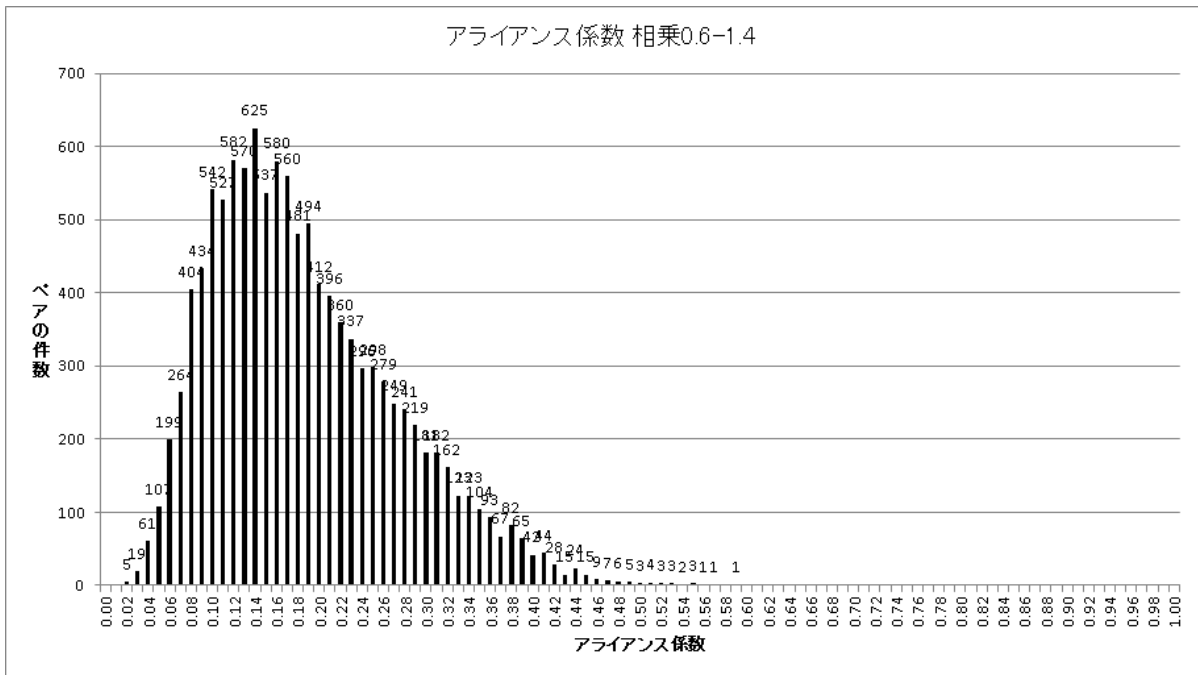
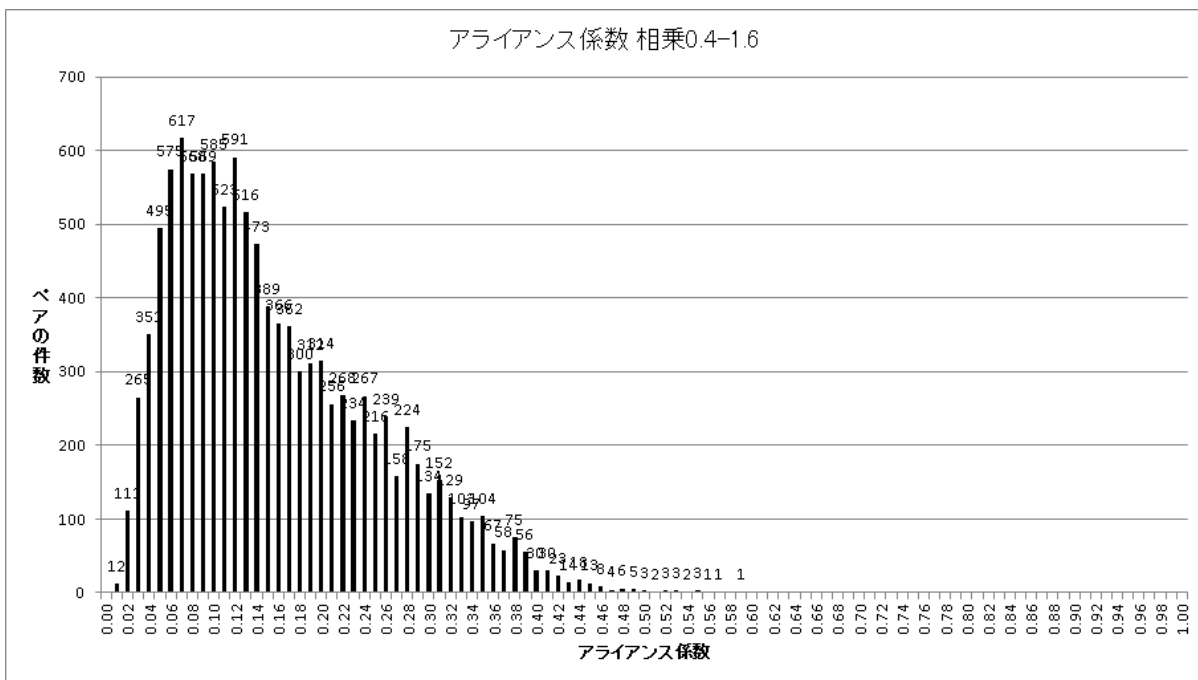


図 38 0.4~1.6 の相乗でのすべての組み合わせのアライアンス係数



8.5 相互補完・加算・相乗モデルのまとめ

第 3、4、6 章で提案された相互補完モデルに、第 7 章で営業エリアに関する加算モデルを追

加したモデルに、さらに、本章にて、アライアンス成立における双方の企業の意欲（やる気）の度合いを盛り込む相乗モデルを追加して、モデルを発展させた。

ここにて、筆者が本研究で達成したいと願っていたアライアンスのマッチング段階の成立メカニズムを数学表現する「アライアンスの相互補完・加算・相乗モデル」を完成させることができた。

当モデルについても、Python 言語で実装し、アライアンスの相乗値、そして、アライアンス係数を算出し、提案するモデルの正当性を確認した。

第9章 本研究のまとめ ～研究の成果と今後の研究課題～

最後に、本研究の内容をまとめ、その成果と今後の研究課題について述べておきたい。

9.1 本研究の成果

経営学におけるアライアンス研究においては、アライアンスが成立する最初のマッチングの段階の研究が十分なされておらず、かつ、数理モデルが存在していないというアライアンス研究の未整備の部分に取り組み、企業と企業のアライアンスに向けたマッチングがどのような関係性の時に成立しやすいのかを数学的に表現し、数値として演算可能（コンピューティショナル）にしたことが当研究の成果である。

本研究のモデル構築では、資源ベース理論のフレームワークを発展させただけでなく、物理学におけるフローの概念や人間関係におけるギブ・アンド・テイクの考え方を組み合わせた。フロー・インテンシティとフロー・バランスの概念を新しく用いることで、2社間のアライアンス成立における相互補完関係を説明づけることができた。

提案したモデルでは、2社間の相互補完関係を、1次元行列と2極のベクトルで数学表現し、そして、最大の相互補完強度の点（マキシマム・ポイント）からの距離で、数学的に表すこととした。この距離で強度を表現することに気が付いたことが、世界初のアライアンス成立の関係性を表現する相互補完数理モデル構築の最大の成功要因となった。

相互補完数理モデルを発展させるオプションとして、営業エリアに関する加算モデル及び企業の意欲に関する相乗モデルも追加開発し、アライアンスの相互補完・加算・相乗モデルへとモデルを発展させた。これらにより、筆者が実務的に行ってきたアライアンスの仲介者としての役割・機能を、数学的に表現することができた。

提案するモデルは、いずれも、Python 言語で実装し、実際の152社のコンサルティング先企業データによって、モデルの正当性を確認した。

本研究は、アライアンスの成立のメカニズムを、厳密に関係性を表現できる数学で表わし、モデル化したことに意義がある。企業間アライアンスのマッチング段階の成立メカニズムを表現す

る数理モデルを提案し、2 社間の関係性を数値として演算可能にしたことにより、複数の潜在的なアライアンス候補先企業の中から、最適なアライアンス先企業を選定する合理的な意思決定が行えるようになった。

9.2 経営学と CI の組み合わせにより、双方において貢献

本研究では、経営学で主流の説明しにくい領域であるアライアンスのマッチング段階に着目し、経営学で一般的ではないモデル化に取り組んだ。すなわち、経営学で主要な手法、とりわけ、多変量解析の手法では説明しにくいアライアンスのマッチング段階の数理モデルを構築した。アライアンスのマッチング段階は、経営学の主流のモデルでは説明しにくい領域だったため、経営学での研究だけでは、アライアンスのマッチング段階のモデル化ができていなかった。

本研究で提案した相互補完数理モデルにより、アライアンスのマッチング段階の 2 社間の関係性を 2 次元のマップ上の点で捉えることが可能となった。

さらに、Python 言語でモデルを実装し、実データとの整合性も確認したことも、経営学だけではなしえなかったことである。

経営学では容易にモデル化ができなかった領域において、サイバーインフォマティクス (CI) の手法やロジックを導入することで、課題を解決し、世界で初めて数理モデルを構築・提案した。このことが、本研究の学術面の貢献である。

これは、経営学におけるアライアンス研究を進展させただけでなく、CI にとっても、そのアプリケーション分野を、アライアンス研究に拡大することができたと言える。このように、本研究は、経営学としても、CI の分野としても、意義ある研究成果を生み出した。

9.3 当モデルの利用面の意義

このモデルの利用面の意義としては、主として、次の3つがある。

- (1) アライアンスが実際に成立していることのメカニズムを数学的に説明できる。
- (2) 各企業が最適なアライアンス先を選択する意思決定に利用できる。
- (3) アライアンスのマッチング段階の効率性を高め、探索コストを低減できる。

まず、(1)については、数学という厳密に関係性を表現できるツールを用いることで、モデルをプログラミングで実装することが可能となり、数値として演算できるようになった。また、アライアンスのアレンジメントをする仲介者がどのようなメカニズムで、企業紹介による組み合わせを行っているかを明らかにできた。

次に、(2)については、提案する数理モデルを用いて、2社間の関係性を数値化することにより、その数値の比較によって、複数の潜在的なアライアンスの候補先の中から、どの企業が最適なアライアンス先であるかを判断・選択し、合理的に意思決定を行うことができる。翻って見れば、意思決定の妥当性の検証にも用いることができる。

最後に、(3)については、多くのアライアンス候補先の中から、最適なアライアンス先を探索するにあたって、やみくもに探索するのではなく、提案するモデルによる数値を用いて、探索することにより、マッチングを効率化し、アライアンス先を探索するコストを低減することができる。

本研究で提案する数理モデルは、このような利用面での意義がある。

近年、日本において、新規事業立ち上げや売上向上のために、アライアンスの重要性やニーズが、非常に高まっている。アライアンスの傾向やタイプについても、異業種間にまたがるようになってきている。大企業とスタートアップ・ベンチャーの間のアライアンスや、中小企業同士のアライアンスも増加傾向にある。また、ビジネス・マッチングのための交流会が増加しており、アライアンスを専門とするコンサルティング会社も、日本企業同士のアライアンスを促進させるのと同様、日本企業と日本国外企業とのアライアンスも推進している。本研究で提案したモデルは、そういったアライアンスの活動の推進に貢献するものである。

9.4 当モデルの改良と今後の研究課題

本研究では、まず新しい数理モデルを提案することに注力しており、それが実現したことは大きな成果であり、意義であると考ええる。

しかしながら、提案したモデルの特徴の数や項目、各特徴の評点付けの方法などは、より実態を正確に反映するように、今後、ブラッシュアップして、改良していく余地がある。

本稿で提案したモデルは、フレキシブルなものであり、ユーザーは、自由にパラメーターや評点付けなどを変更することができるようになっている。具体的には、相互補完モデルにおける 8 つの項目の設定の仕方やそれぞれの評点付け、加算モデルにおける地域区分の仕方や各区分へのスコアの配分、相互補完モデルへの加算モデルや相乗モデルを追加する際の重み付けなどの設定をどのようなにするのがよいかをさらに検討していく必要がある。今後、ユーザーが様々なデータでモデルを利用可能とするため、本稿で提案し、Python 言語によって実装した数理モデルのプログラミングを、オープン・ソースとして、Web 上にて、公開することとする。

提案したモデルを使って、様々なデータ、たとえば、S&P500 などの株式インデックスの構成する銘柄企業のデータや、他のコンサルティング会社のクライアント企業データ、インターネット上で財務データが公開されている企業データなどで、提案したモデルがテストされることによって、さらに研究が発展していくものとする。

謝辞

本研究を遂行し完成させるにあたって、指導教授・主査の武藤佳恭先生（慶應義塾大学環境情報学部教授）には、非常に熱心にご指導をいただいた。心から、深く、最大限の感謝の気持ちをお伝えしたい。副査の清木康先生（慶應義塾大学環境情報学部教授）、小澤太郎先生（慶應義塾大学総合政策学部教授）にも的確な助言をいただき、とりわけ、琴坂将広先生（慶應義塾大学総合政策学部・准教授）には、経営学の観点から、有益な助言をいただいた。御礼申し上げたい。

その他、Python 言語でのプログラム作成においては、山本浩之氏（慶應義塾大学大学院政策・メディア研究科・後期博士課程）に協力いただいた。コンサルティング先企業データを用いた研究の進め方については、岡部光明先生（慶應義塾大学・名誉教授）に助言をいただいた。経営学分野の研究として、本研究を推進するにあたっては、亀川雅人先生（立教大学大学院ビジネスデザイン研究科・教授）に、有益なご助言をいただいた。そういった方々の支えによって、この博士論文が完成したと言える。ここに、その旨を記載し、感謝申し上げます。

参考文献

<日本語文献>

- 石井真一（2003）『企業間提携の戦略と組織』中央経済社.
- 牛丸元（2007）『企業間アライアンスの理論と実証』同文館出版.
- 坂井 豊貴（2010）『マーケットデザイン入門—オークションとマッチングの経済学』ミネルヴァ
書房
- 清水洋・星野雄介（2012）「オープン・イノベーションのマネジメント ～探索と知識マネジメント」、『一橋ビジネスレビュー』、60 巻 2 号、2012 年 8 月号
- 富田賢（2014）『新規事業立ち上げの教科書 ～ビジネスリーダーが身につけるべき最強スキル』
総合法令出版.
- 富田賢・武藤佳恭（2015） 「アライアンスの相互補完数理モデルの構築と実証分析 ～152 社の
コンサルティング先企業データを用いて」、経営会計研究 第 20 巻第 1 号
- 富田賢・武藤佳恭（2016） 「アライアンスの相互補完・加算・相乗に関する数理モデルの提案
と Python 言語による実証～152 社の企業データをもとに」、ビジネスクリエーター研究第 7
号
- 中村裕一郎（2013）『アライアンス・イノベーション：大企業とベンチャー企業の提携：理論と実
際』白桃書房.
- 星野達也（2015a）「オープン・イノベーションという新たな武器 ～製造業復活をかけて自前主
義を脱却せよ～」ダイヤモンド・ハーバード・ビジネス・レビュー2015 年 6 月号
- 星野達也（2015b）『オープン・イノベーションの教科書 ～社外の技術でビジネスをつくる実践
ステップ』ダイヤモンド社.
- 元橋一之編著（2014）『アライアンスマネジメント ～米国の実践論と日本企業への適用』白桃書
房.
- 安田洋史（2006）『競争環境における戦略的提携 その理論と実践』NTT 出版.
- 安田洋史（2010）『アライアンス戦略論』NTT 出版.
- 安田洋史（2015）「アライアンス成果に対するパートナー間多様性の影響」、『日本経営学会誌』第

35号 pp. 16~27

湯川 抗 (2013) 『コーポレートベンチャリング新時代：本格化するベンチャーの時代と大手 ICT 企業の成長戦略』白桃書房.

米倉誠一郎・清水洋 (2015) 『オープン・イノベーションのマネジメント ～高い経営成果を生む 仕組みづくり』有斐閣.

米倉穰 (2012) 『オープン・イノベーションと企業の戦略的提携 ～再生医療のネットワーク型総合産業化の創造に関する研究』税務経営協会.

<英語文献>

Adegbesan, J. Adetunji(2009) “On the Origins of Competitive Advantage : Strategic Factor Markets and Heterogeneous Resource Complementarity,” *Academy of Management Review*, 2009, Vol.34, No.3, pp.463-475.

Akbarpour, Mohammad, Shengwu Li, and Shayan Oveis Gharan(2014) “Dynamic Matching Market Design,” *Proceedings of the fifteenth ACM conference on Economics and computation*

Anand, B. and Khanna, T. (2000) “Do Firms Learn to Create Value? The Case of Alliances,” *Strategic Management Journal*, vol.21, pp.295-315.

Bamford, James, David Ernst and David G. Fubini(2004) “Launching a World-Class Joint Venture,” *Harvard Business Review*, February 2004(邦訳：バンフォード、ジェームズ、デイビッド・アーンスト、デイビッド・フビニ(2005)「JVの成否は100日で決まる ～戦略的提携に大半が失敗する～」ダイヤモンド・ハーバード・ビジネス・レビュー2005年2月号)

Bamford, James, Benjamin Gomes-Casseres, and Michael Robinson(2003) *Mastering Alliance Strategy -A Comprehensive Guide to Design, Management , and Organization*,” Jossey-Bass

Barney, Jay(1991) “Firm Resources and Sustained Competitive Advantage,” *Journal of Management*, Vol.17, No.1, pp.99-120.

Chang, Shao-Chi, Sheng-Syan Chen, Jung-Ho Lai(2008) “The effect of alliance experience

- and intellectual capital on the value creation of international strategic alliances,”
Omega, vol. 36, no. 2, pp. 298–316.
- Chesbrough, Henry (2003) *Open Innovation*, Harvard Business School Press (大前恵一朗訳
(2004) 『OPEN INNOVATION』 産業能率大学出版部)
- Chesbrough, Henry (2006a) *Open Business Models : How to Thrive in the New Innovation
Landscape*, Harvard Business School Press. (邦訳 ヘンリ・チェスブロウ (2007) (栗原潔
訳) 『オープンビジネスモデル ～知財競争時代のイノベーション～』 翔泳社)
- Chesbrough, Henry, Wim Vanhaverbeke, and Joel West (2006b) *Open Innovation -Research in
a New Paradigm*, Oxford University Press.
- Collis, David J. and Cynthia A. Montgomery (1998) *Corporate Strategy : A Resource-
Based Approach*, McGraw-Hill Companies , Inc. (デビッド・J・コリス and シンシア・A・モ
ンゴメリー (2004) 『資源ベースの経営戦略論』, 根来龍之他訳, 東洋経済新報社.)
- Das, T. K., and Bing-Sheng Teng (1998a) “Resource and Risk Management in the Strategic
Alliance Making Process,” *Journal of Management*, Vol. 24, No. 1, pp. 21–42
- Das, T. K., and Bing-Sheng Teng (1998b) “Between Trust and Control : Developing
Confidence in Partner Cooperation in Alliances,” *Academy of
Management Review*, Vol. 23, No. 3, pp. 491–512
- Das, T. K., and Bing-Sheng Teng (2000) “A Resource-Based Theory of Strategic
Alliances,” *Journal of Management*, Vol. 26, No. 1, pp. 31–61
- Das, T. K., and Bing-Sheng Teng (2002) “Alliance Constellations : A Social Exchange
Perspective,” *Academy of Management Review*, 2002 Vol. 27, No. 3, 445–456
- DeVellis, Robert F. (2012) *Scale Development -Theory and Applications*, Third Edition,
Applied Social Research Methods Series, SAGE
- Doz, Yves and Gary Hamel (1998) *Alliance Advantage -The Art of Creating Value through
Partnering*, Harvard Business School Press. (邦訳 志太勤一・柳孝一監訳、和田正春訳

- (2001) 『競争優位のアライアンス戦略 ～スピードと価値創造のパートナーシップ』ダイヤモンド社)
- Dyer, Jeffrey H. ,Prashant Kale,Harbir Singh,Harpreet Singh(2004) “When to Ally and When to Acquire,” Harvard Business Review,2004 Jun-Aug. (邦訳：ダイアー,ジェフリー,プラシャント・ケール,ハーバー・シン(2005)「提携すべき時,買収すべき時 ～3つの視点から分析する～」ダイヤモンド・ハーバード・ビジネス・レビュー2005年2月号)
- Ernst, David and James Bamford(2005) “Your Alliances Are Too Stable,” Harvard Business Review ,2005 June pp.133-141(邦訳：アーンスト,デイビッド,ジェームズ・バンフォード(2006)「いかに提携事業をリストラするか ～大半の低収益のままに放置されている～」ダイヤモンド・ハーバード・ビジネス・レビュー2006年6月号)
- Gassmann, Oliver and Ellen Enkel(2004) “Towards a Theory of Open Innovation: Three Core Process Archetypes,” conference paper, R and D Management Conference (RADMA) (Lisabon, Portugal), 6-7 2004, double-blind review
- Gomes-Casseres,Benjamin(1997) “Alliance Strategies of Small Firms,” Small Business Economics,Vol. 9, pp. 33-44
- Gompers, Paul A. and Josh Lerner(2000) *The Venture Capital Cycle*, The MIT Press(富田賢訳者代表・吉田和男監訳(2002)『ベンチャーキャピタル・サイクル ～ファンド設立から投資回収までの本質的理解』シュプリンガーフェアラーク東京)
- Hamel, Gary, C.K. Prahalad and Yves Doz(1989) “Collaborate with Your Competitors - and Win,” *Harvard Business Review*, 1989 Jan-Feb. (邦訳：ハメル,ゲイリー,C.K. プラハラッド,イブ L. ドーズ(2005)「ライバルとのコラボレーション戦略 ～新たなスキルと知識を獲得し,競争力を高める～」ダイヤモンド・ハーバード・ビジネス・レビュー2005年2月号)
- Hamel, Gary and C.K. Prahalad(1994) *Competing for the Future*, Harvard Business School Press. (邦訳：ハメル,ゲイリー,C.K. プラハラッド『コア・コンピタンス経営』,日本経済新聞社,1995年,日経文庫2001年)
- Hensley, Rhonda L (1999) “A review of operations management studies using scale

- development techniques,” *Journal of Operations Management*, Volume 17, Issue 3, March 1999, Pages 343-358
- Hinkin, T. R. (1998) “A brief tutorial on the development of measures for use in survey questionnaires,” *Organizational Research Methods*, 2(1), 104-121.
- Kaplan, Robert S., David P. Norton, and Bjarne Rugelsjoen(2009) “Managing Alliances with the Balanced Scorecard,” *Harvard Business Review*, 2010 Jan-Feb. (邦訳：キャプラン, ロバート S, デイビッド P. ノートン, ビヤーン・ルジェルスヨエン(2010)「戦略的提携を実現するバランス・スコアカード」*ダイヤモンド・ハーバード・ビジネス・レビュー*2010年11月号)
- Kelley, Donna J. and Mark P. Rice(2002) “Advantage beyond founding : The strategic use of technologies,” *Journal of Business Venturing*, vol. 17, no.1, pp.41-57.
- Kaysar, Boaz, Benjamin A. Converse, Jiunwen Wang, and Nicholas Epley(2008) “Reciprocity is Not Give and Take : Asymmetric Reciprocity to Positive and Negative Acts,” *Psychological Science*, Volume19, No.12
- Lavie, Dovev(2006)” The Competitive Advantage of Interconnected Firms : An Extension of the Resource-Based View,” *Academy of Management Review*, Vol. 31, No. 3, pp. 638-658
- Lee, Yikuan and S. Tamer Cavusgil (2006) “Enhancing alliance performance: The effects of contractual-based versus relational-based governance,” *Journal of business research*, vol. 59, no. 8, p. 896-905.
- Leiblein, Michael J and Jeffrey J Reuer (2004) “Building a foreign sales base: the roles of capabilities and alliances for entrepreneurial firms,” *Journal of Business Venturing*, vol. 19, no. 2, p. 285-307.
- March, James G. (1991) “Exploration and Exploitation in Organizational Learning,” *Organization Science* , Vol. 2, No. 1, pp. 71-87
- Mitsubishi, Hitoshi and Henrich R. Greve(2009)” A Matching Theory of Alliance Formation and Organizational Success: Complementarity and Compatibility,” *Academy of*

- Management Journal, 52 (5): 975-995.
- Pierre J., Richard, Timothy M. Devinney, George S. Yip and Gerry Johnson(2009) "Measuring Organizational Performance: Towards Methodological Best Practice," *Journal of Management*, June 2009 35: 718-804
- Roth, Alvin E. (2015) "Who Gets What - and Why, The New Economics of Matchmaking and Market Design," William Collins (邦訳: アルビン・E・ロス著、櫻井裕子訳「フー・ゲッツ・ホワット ～マッチメイキングとマーケットデザインの新しい経済学」日本経済新聞出版社、2016年)
- Rothaermel, Frank T. and David L Deeds(2006) "Alliance type, alliance experience and alliance management capability in high-technology ventures," *Journal of business venturing*, vol. 21, no. 4, p.429-460.
- Schaufeli, Wilmar B. (2006) "The Balance of Give and Take : Toward Social Exchange Model of Burnout," *Revue Internationale de Psychologie Sociale*, Vol 19(1), Mar 2006, 87-131
- Thanos, Ioannis C, Pavlos Dimitratos, Panagiota Sapouna(2016) "The implications of international entrepreneurial orientation, politicization, and hostility upon SME international performance," *International Small Business Journal*, April 25, 2016 0266242616641749
- Tjemkes, Brian and Pepijn Vos, Koen Burgers(2012), *Strategic Alliance Management*, Routledge.
- Tomita, Satoshi and Yoshiyasu Takefuji(2016a) "A New Mathematical Model of Mutually Complementary for Corporate Alliances : Selection of Optimal Partners using Eight Characteristics," Proceedings (HI092415299, double-blind review) of 2016 Hawaii Global Conference on Business and Finance (GCBF)
- Tomita, Satoshi and Yoshiyasu Takefuji(2016b), "A Mathematical Model for Optimal Corporate Alliances : Evidence from Japan," *International Journal of Management and Marketing Research(IJMMR)*, Volume 9, Number 1, 2016.

Wernerfelt, Birger (1984) “A Resource-Based View of the Firm,” Strategic Management Journal, 5: 171-180

Yasuda, Hiroshi (2003) “New Analytical Approach for Strategic Alliances from the Perspective of Exchange of Management Resources,” Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy, Graduate School of Decision Science and Technology, Tokyo Institute of Technology

Yoshino, Michael and U. Srinivasa Rangan (1995), Strategic Alliances : An Entrepreneurial Approach to Globalization, Harvard Business School Press.

<参考資料>

内閣府・県民経済計算・名目(平成 24 年度)

http://www.esri.cao.go.jp/jp/sna/data/data_list/kenmin/files/contents/main_h24.htm
1

(閲覧日 : 2015 年 7 月 30 日)

下山弘一「アライアンスの 2 つの形。垂直統合・水平統合」、2002 年、NPO 法人・IT コーディネーター京都 Web サイト

<http://www.itc-kyoto.jp/2002/09/23/%E3%82%A2%E3%83%A9%E3%82%A4%E3%82%A2%E3%83%B3%E3%82%B9%E3%81%AE%EF%BC%92%E3%81%A4%E3%81%AE%E5%BD%A2-%E5%9E%82%E7%9B%B4%E7%B5%B1%E5%90%88-%E6%B0%B4%E5%B9%B3%E7%B5%B1%E5%90%88-%E4%B8%8B%E5%B1%B1-%E5%BC%98%E4%B8%80/>

(閲覧日 : 2016 年 3 月 15 日)

付録1：相互補完モデルの最もシンプルな特徴数4つでの構築

本稿の第3章に掲載した相互補完モデルの構築にあたっては、シンプルな特徴数として、特徴数4つで、最初にモデル構築を行った。これについては、富田・武藤(2015)に掲載したものであるが、ここでも、最初にモデル構築を行った特徴数4つのケースにおいて、付録として再録しておく。

1次元行列と2極のベクトルでの数学表現

企業2社の相互補完関係は、1次元行列で表現できる。X社とY社の強み・弱みを4つの項目で、1から5の評点を付けると、下記のように表現できる。

一例としては、

$$X \text{ 社 } x=(5, 1, 4, 2)$$

$$Y \text{ 社 } y=(1, 4, 2, 5)$$

となる。

その上で、A社とB社の項目ごとに演算(引き算)した結果 c は、特徴数4の長さ0~4の+か-の方向を持つ2極のベクトルとして表現できる。

$$z=x-y=(4, -3, 2, -3)$$

このように、相互補完関係は、各社の強み・弱みは、特徴数の1次元行列、2社間の相互補完関係は、2極のベクトルで表現することとする。

最大の相互補完の点からの距離での強度の数学表現

相互補完の強さを表すにあたっては、最大の相互補完強度の地点からの距離で、その企業ペアの相互補完強度を測定・表現することとする。

すなわち、特徴数(強み・弱みの評点を付ける項目の数)4つで、長さ0~4の2極ベクトルでは、最大の相互補完は、特徴数4のうち、特徴数2つずつ、最大の長さ4となるケースとなり、(特徴数 $4 \div 2$) \times 最大の長さ4で、(8, -8)となる。

2点間の距離は、下記で算出される。

$$d = \sqrt{(a_1 - a_2)^2 + (b_1 - b_2)^2}$$

最大の点(8, -8)の相互補完強度は、(0, -0)から(8, -8)の距離となり、

$$\sqrt{(8 - 0)^2 + (-8 - 0)^2} = 11.31$$

となる。

相互補完強度は0から11.3の間の値で、大きいほど相互補完強度は大きいと解釈できる。

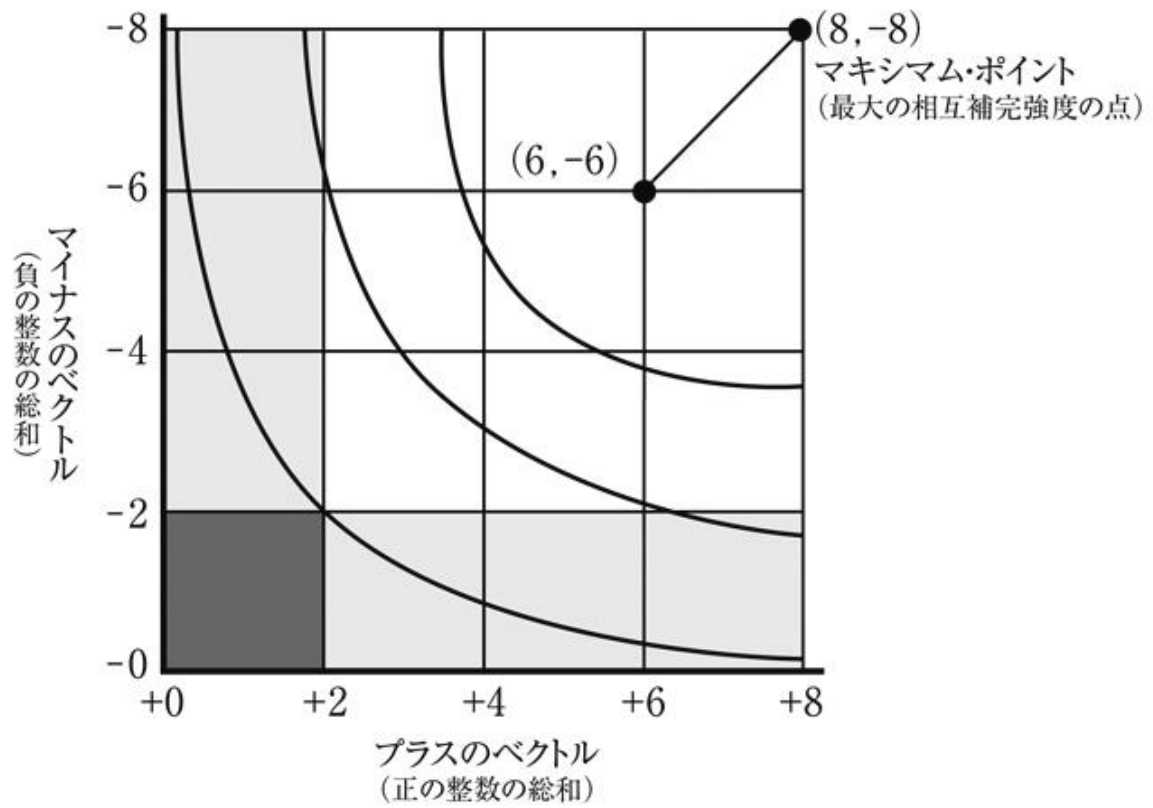
最大の相互補完の地点である(8, -8)からの距離が小さければ小さいほど、相互補完強度が強いということとなり、強度を表す指標は数字が大きければ大きいほうが取り扱いやすいため、最大の数値から引くことで、数字の大小の向きを反転させることとした。

たとえば、前述で導出した(6, -6)の相互補完強度は、下記のように、(8, -8)からの距離として把握することとなり、その数字を最大値から差し引くことで、算出することができる。

$$\sqrt{(8 - 0)^2 + (-8 - 0)^2} - \sqrt{(8 - 6)^2 + (-8 - (-6))^2} = 8.485$$

特徴数4つの最もシンプルなアライアンスの2極の相互補完モデルを表す図は、次の図39である。

図 39 アライアンスの 2 極の相互補完モデル



付録2：8つの特徴数での評点付けの4つの因子としてのクライテリア

強み・弱みの評点付けの8つの項目については、各項目について、次のような4つの判断基準を因子として設定した。そして、それらの因子での判断を総合して評点付けを行った。一部、数値基準を設けられるものについては数値基準を設けた。

1. 営業力がある・ない <販売資源>

- ①営業マンが生き生きしている(元気があり、能動的か)。
- ②営業マネジメントが出来ている(営業会議の整備、営業マンの営業数値の管理、有能な営業部長の有無)。
- ③営業マンの数が多い・少ない(人数で区切りを設けて評価)。
- ④社長の営業力がある(社長自身が営業活動が得意か不得手か、好きか嫌いか)。

2. 技術力がある・ない <技術資源>

- ①固有の技術を保有している。(ニッチトップなど際だったものなどを有しているか?)
- ②技術者の質が高い(熟練度・勤続年数、学士・修士・博士の比率、論文や特許の輩出数など)。
- ③特定技術分野に取り組んでいる期間が長い(取り組んでいる領域での業歴など)。
- ④研究所や開発部門が充実している(独自技術を開発する部署の有無、所属する技術者・研究者の人数)。

3. アイディア力がある・ない

- ①アイディアを自由に発想し、新しいことを創造する社風である(それらを尊重する社風か)。
- ②アイディア出しが得意な人が多い(誰もいないか、社長だけか、複数名のアイディア出しができる人がいるか)。
- ③次々に新しい事業・アイディア・技術を生み出している(過去5年間で新しいものを輩出したか、その数はどうか)。

④アイデアを形にする仕組みがある(社内ベンチャー公募制度などのアイデア吸い上げの仕組みがあるか、機能しているか)。

4. 資金力がある・ない <資金資源>

①現預金を多く有している(預貯金の額で区切りを設けて評価)。

②自己資本比率が高い(その数値によって区切を設けて評価)。

③資金調達力がある(銀行借入や第三者割当増資ができるか、親会社の有無・状況)。

④資金効率・利益率が高い(営業利益率で区切りを設けて評価)。

5. 人材がいる・いない <人材資源>

①社員数が多い(人数で区切りを設けて評価)。

②社員の余剰感がある(不足しているか外にも提供できる状況かで評価)

③人材採用力が高い(採用がうまくいっているか、人を集めやすいかで評価)。

④人材派遣の免許を有している(特定派遣や一般派遣の免許の有無)。

6. 生産力がある・ない <生産資源>

①工場の余力(機械・スペース)を多く持っている(工場や設備の余剰感も加味して評価)。

②工場の工員(ブルーワーカー)を多く有している(人数で区切りを設けて評価)。

③生産管理や品質管理のノウハウがある(どのくらい行えているか得意か苦手かで評価)。

④ファブレスの方針ではない(工場を持つ方針かOEMかで評価)。

7. ブランド・信用がある・ない

①上場企業や上場企業の関連子会社である。

②業歴が長い(年数で区切りを設けて評価)。

③売上高が大きい(金額で区切りを設けて評価)。

④ブランディングに力を入れている(Web・SNS、パンフ・チラシ、ロゴなどの整備で評価)。

8. 機動的な組織風土である・でない

- ①新しいことに意欲的に取り組む社風である(後ろ向きでネガティブな体質か前向きで新しいことに意欲的かで評価)。
- ②議論ばかりせず、実行に移す姿勢である(前に進める雰囲気、実行力の面で判断して評価)。
- ③年功序列(若い人の意見の尊重)やセクショナリズムの度合いが低い(それらの度合いで評価)。
- ④ 自前主義ではなく、外部や異質な組織との対応力がある(それらの度合いで評価)。

付録3：特徴数が奇数の場合の相互補完モデルについての考察

アライアンスの相互補完モデルは、本稿の第3章、第4章で説明したように、最大の相互補完強度を、特徴数の半分ずつの数の最大の長さのベクトルから求めるため、特徴数が偶数の場合はよいが、奇数の場合、どうするのかという問題が残っている。

それについては、本編においては、特徴数を奇数にする場合は、一つダミー変数を追加する解決策について記述している。

この点についても、もう少し、考察を深めてみたい。

本編でも用いた特徴数8つのケースから、一つ特徴数を削って、7つにした場合で、考えてみる。評点付けは、同じく、5段階とする。すなわち、最大の長さのベクトルは4ということとなる。

相互補完モデルの考え方では、特徴数が7つのケースでは、2極のベクトルが、片方からの強みの提供が4つで、相手先からの弱みを補完する強みの提供が3つの場合で、4つと3つの組み合わせとなり、

4, 4, 4, 4, -4, -4, -4

と、

-4, -4, -4, 4, 4, 4, 4

である。

このように、特徴数が奇数の場合は、最大の点が、2つ、出現する。

つまり、

(16, -12)

と、

(12, -16)

の2つの点が出てくる。

その際の相互補完強度の求め方は、(16, -12)と(12, -16)を結ぶ線は、リニア（線形）であると想定した場合、平均で算出することができる。ただし、その点は、実際には存在しない点とな

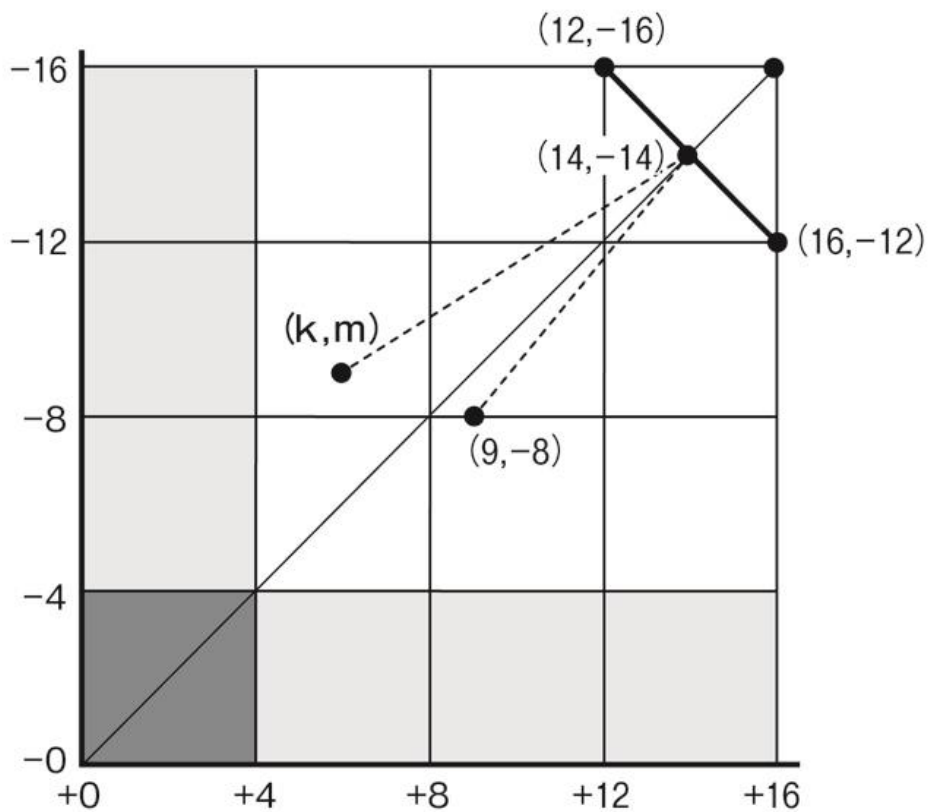
る²⁰。計算は、下記となる。

$$\left(\frac{16+12}{2}, \frac{-12+(-16)}{2} \right) = (14, -14)$$

が、特徴数7つで、最大のベクトルの長さが4の場合の最大の相互補完強度の点となる。

図で示すと、次の図40のようになる。

図40 特徴数が奇数の場合の相互補完強度の求め方



²⁰ 1~5の評点付けで、7つの特徴数の場合で、(14, -14)の組み合わせは、実在しないためである。

たとえば、本編での下記の A 社と B 社の例で、最後の 8 つ目の特徴を削って、7 つにした場合、

$$\text{A 社} \quad a=(1, 3, 4, 2, 5, 1, 3)$$

$$\text{B 社} \quad b=(4, 1, 1, 3, 1, 5, 3)$$

A 社から B 社のそれぞれの特徴における点数を引き算した結果の c は、2 極のベクトルの配列として表現され、0~4 の間のプラスかマイナスの数値となる。

$$\text{A 社} - \text{B 社}$$

$$c = a-b = (-3, 2, 3, -1, 4, -4, 0)$$

で変わらず、(9, -8) が、この 2 社の組み合わせの相互補完関係を示す点となる。

特徴数 7 つでの相互補完強度は、次の式で求められる。

$$d(a,b) = \sqrt{(14-0)^2 + (-14-0)^2} - \sqrt{(14-9)^2 + (-14-(-8))^2}$$

A 社と B 社の特徴数 7 つでの相互補完強度は、計算すると、11.989

となる。正規化した相互補完強度係数は、0.606 である。なお、7 つの場合の相互補完強度の最大値は、19.799 である。なお、これらは、Python 言語によって、計算した。

一般化して、特徴数 7 つの場合の (k, m) となる組み合わせの相互補完強度は、

$$d(k,m) = \sqrt{(14-0)^2 + (-14-0)^2} - \sqrt{(14-k)^2 + (-14-m)^2}$$

として、算出できる。それを、正規化すれば、相互補完強度係数となる。正規化の方法は、最大の相互補完の数字で、上記の右辺を割る形である。

このように、(14, -14)の点はバーチャルな点で、実際は存在しない点となるが、2つの最大の相互補完の点の平均から、マキシマム・ポイントを設定して、特徴数が奇数の場合の相互補完強度を求めることができる。

これにより、本稿で提案するアライアンスの相互補完数理モデルは、特徴数が奇数の場合にも、機能すると言える。

以上が、アライアンスの相互補完数理モデルにおける特徴数を奇数にする場合に関する追加的な考察である。

付録4 : 相互補完モデルの Python 言語のプログラミング

- cdata.xls のアライアンス対応表入力チェック VBA プログラム

ここから-----

```
Sub ボタン1_Click()
```

```
    For i = 3 To 154
```

```
        For j = 3 To 154
```

```
            If Cells(i, j).Value = "1" Then
```

```
                If Not (Cells(j, i).Value = "1") Then
```

```
                    MsgBox "Please verify the relations between No." and j - 2 and "
and No." and i - 2 and " (maybe 1)."
```

```
                    Exit Sub
```

```
                End If
```

```
            ElseIf Cells(i, j).Value = "0" Then
```

```
                If Not (Cells(j, i).Value = "0") Then
```

```
                    MsgBox "Please verify the relations between No." and j - 2 and "
and No." and i - 2 and " (maybe 0)."
```

```
                    Exit Sub
```

```
                End If
```

```
            End If
```

```
        Next j
```

```
    Next i
```

```
End Sub
```

ここまで-----

- 相互補完強度、強度係数算出用 Python プログラム

ここから-----

```
# -*- coding: utf-8 -*-

import xlrd, xlwt

import numpy as np

from math import sqrt

parameters = 8

parameter_start = 29

success_cols = (1, 2, 3, 4, 5, 6, 7)

fail_cols = (8, 9, 10, 11, 12, 13, 14)

all_cols = (15, 16, 17, 18, 19, 20, 21)

all_success_cols = (22, 23, 24, 25, 26, 27, 28)

all_success_fail_cols = (29, 30, 31, 32, 33, 34, 35)

all_fail_cols = (36, 37, 38, 39, 40, 41, 42)

def title():

    for i in success_cols:

        outputsheet.write(0, i, u'成立')

    for i in fail_cols:

        outputsheet.write(0, i, u'不成立')

    for i in all_cols:

        outputsheet.write(0, i, u'全体')

    for i in all_success_cols:
```

```

        outputsheet.write(0, i, u'全体-成立')
for i in all_success_fail_cols:
        outputsheet.write(0, i, u'全-成-不')
for i in all_fail_cols:
        outputsheet.write(0, i, u'全-不')
outputsheet.write(1, 0, u'件数')
outputsheet.write(2, 0, u'平均值')

def success(): # 成立
    relbook=xlrd.open_workbook('cdata.xls')
    relsheet=relbook.sheet_by_index(1)
    number, name1, name2, sum1, sum2, intensity, coefficient = (), (), (), (), (),
(), ()
    for i in range(0, len(code)):
        for j in range(0, len(code)):
            vector = parameter_array(i, :) - parameter_array(j, :)
            plus=0
            minus=0
            for k in range(len(vector)):
                if vector(k)>0 : plus+=vector(k)
                else : minus+=vector(k)
            if relsheet.cell(code(i)+1, code(j)+1).value == 1 :
                number.append(str(code(i)) + " - " + str(code(j)))
                name1.append(name(i))
                name2.append(name(j))

```

```

        sum1.append(sum(i))

        sum2.append(sum(j))

        intensity.append( sqrt( (2*(4*len(vector)/2))**2 ) -
sqrt( (4*len(vector)/2-plus)**2 + (-4*len(vector)/2-minus)**2 ) )

        coefficient.append( 1 - sqrt( (4*len(vector)/2-plus)**2
+ (-4*len(vector)/2-minus)**2 ) / sqrt( (2*(4*len(vector)/2))**2 ) )

    for i in success_cols:

        outputsheet.write(1, i, len(number))

    for i in range(0, len(number)):

        outputsheet.write(i+3, success_cols(0), number(i))

        outputsheet.write(i+3, success_cols(1), name1(i))

        outputsheet.write(i+3, success_cols(2), sum1(i))

        outputsheet.write(i+3, success_cols(3), name2(i))

        outputsheet.write(i+3, success_cols(4), sum2(i))

        outputsheet.write(i+3, success_cols(5), intensity(i))

        outputsheet.write(i+3, success_cols(6), coefficient(i))

    outputsheet.write(2, success_cols(2), np.average(sum1))

    outputsheet.write(2, success_cols(4), np.average(sum2))

    outputsheet.write(2, success_cols(5), np.average(intensity))

    outputsheet.write(2, success_cols(6), np.average(coefficient))

```

```
def fail(): # 不成立
```

```
    relbook=xlrd.open_workbook('cdata.xls')
```

```
    relsheet=relbook.sheet_by_index(1)
```

```
    number, name1, name2, sum1, sum2, intensity, coefficient = (), (), (), (), (),
```

() , ()

```
for i in range(0, len(code)):  
    for j in range(0, len(code)):  
        vector = parameter_array(i, :) - parameter_array(j, :)  
        plus=0  
        minus=0  
        for k in range(len(vector)):  
            if vector(k)>0 : plus+=vector(k)  
            else : minus+=vector(k)  
        if relsheet.cell(code(i)+1, code(j)+1).value == 0 :  
            number.append(str(code(i)) + " - " + str(code(j)))  
            name1.append(name(i))  
            name2.append(name(j))  
            sum1.append(sum(i))  
            sum2.append(sum(j))  
            intensity.append( sqrt( (2*(4*len(vector)/2))**2 ) -  
sqrt( (4*len(vector)/2-plus)**2 + (-4*len(vector)/2-minus)**2 ) )  
            coefficient.append( 1 - sqrt( (4*len(vector)/2-plus)**2  
+ (-4*len(vector)/2-minus)**2 ) / sqrt( (2*(4*len(vector)/2))**2 ) )  
        for i in fail_cols:  
            outputsheet.write(1, i, len(number))  
    for i in range(0, len(number)):  
        outputsheet.write(i+3, fail_cols(0), number(i))  
        outputsheet.write(i+3, fail_cols(1), name1(i))  
        outputsheet.write(i+3, fail_cols(2), sum1(i))  
        outputsheet.write(i+3, fail_cols(3), name2(i))
```

```

        outputsheet.write(i+3, fail_cols(4), sum2(i))

        outputsheet.write(i+3, fail_cols(5), intensity(i))

        outputsheet.write(i+3, fail_cols(6), coefficient(i))

    outputsheet.write(2, fail_cols(2), np.average(sum1))

    outputsheet.write(2, fail_cols(4), np.average(sum2))

    outputsheet.write(2, fail_cols(5), np.average(intensity))

    outputsheet.write(2, fail_cols(6), np.average(coefficient))

```

```
def all(): # 全社
```

```
    relbook=xlrd.open_workbook('cdata.xls')
```

```
    relsheet=relbook.sheet_by_index(1)
```

```
    number, name1, name2, sum1, sum2, intensity, coefficient = (), (), (), (), (),
```

```
    (), ()
```

```
    for i in range(0, len(code)):
```

```
        for j in range(0, len(code)):
```

```
            vector = parameter_array(i,:) - parameter_array(j,:)
```

```
            plus=0
```

```
            minus=0
```

```
            for k in range(len(vector)):
```

```
                if vector(k)>0 : plus+=vector(k)
```

```
                else : minus+=vector(k)
```

```
            if code(i) != code(j) :
```

```
                number.append(str(code(i)) + " - " + str(code(j)))
```

```
                name1.append(name(i))
```

```
                name2.append(name(j))
```

```

        sum1.append(sum(i))

        sum2.append(sum(j))

        intensity.append( sqrt( (2*(4*len(vector)/2))**2 ) -
sqrt( (4*len(vector)/2-plus)**2 + (-4*len(vector)/2-minus)**2 ) )

        coefficient.append( 1 - sqrt( (4*len(vector)/2-plus)**2
+ (-4*len(vector)/2-minus)**2 ) / sqrt( (2*(4*len(vector)/2))**2 ) )

    for i in all_cols:

        outputsheet.write(1, i, len(number))

    for i in range(0, len(number)):

        outputsheet.write(i+3, all_cols(0), number(i))

        outputsheet.write(i+3, all_cols(1), name1(i))

        outputsheet.write(i+3, all_cols(2), sum1(i))

        outputsheet.write(i+3, all_cols(3), name2(i))

        outputsheet.write(i+3, all_cols(4), sum2(i))

        outputsheet.write(i+3, all_cols(5), intensity(i))

        outputsheet.write(i+3, all_cols(6), coefficient(i))

    outputsheet.write(2, all_cols(2), np.average(sum1))

    outputsheet.write(2, all_cols(4), np.average(sum2))

    outputsheet.write(2, all_cols(5), np.average(intensity))

    outputsheet.write(2, all_cols(6), np.average(coefficient))

```

```

def all_success(): # 全社-成立

```

```

    relbook=xlrd.open_workbook('cdata.xls')

```

```

    relsheet=relbook.sheet_by_index(1)

```

```

    number, name1, name2, sum1, sum2, intensity, coefficient = (), (), (), (), (),

```

() , ()

```
for i in range(0, len(code)):  
    for j in range(0, len(code)):  
        vector = parameter_array(i, :) - parameter_array(j, :)  
        plus=0  
        minus=0  
        for k in range(len(vector)):  
            if vector(k)>0 : plus+=vector(k)  
            else : minus+=vector(k)  
        if (relsheetsheet.cell(code(i)+1, code(j)+1).value != 1) and  
(code(i) != code(j)):  
            number.append(str(code(i)) + " - " + str(code(j)))  
            name1.append(name(i))  
            name2.append(name(j))  
            sum1.append(sum(i))  
            sum2.append(sum(j))  
            intensity.append( sqrt( (2*(4*len(vector)/2))**2 ) -  
sqrt( (4*len(vector)/2-plus)**2 + (-4*len(vector)/2-minus)**2 ) )  
            coefficient.append( 1 - sqrt( (4*len(vector)/2-plus)**2  
+ (-4*len(vector)/2-minus)**2 ) / sqrt( (2*(4*len(vector)/2))**2 ) )  
        for i in all_success_cols:  
            outputsheetsheet.write(1, i, len(number))  
    for i in range(0, len(number)):  
        outputsheetsheet.write(i+3, all_success_cols(0), number(i))  
        outputsheetsheet.write(i+3, all_success_cols(1), name1(i))  
        outputsheetsheet.write(i+3, all_success_cols(2), sum1(i))
```

```

        outputsheet.write(i+3, all_success_cols(3), name2(i))

        outputsheet.write(i+3, all_success_cols(4), sum2(i))

        outputsheet.write(i+3, all_success_cols(5), intensity(i))

        outputsheet.write(i+3, all_success_cols(6), coefficient(i))

    outputsheet.write(2, all_success_cols(2), np.average(sum1))

    outputsheet.write(2, all_success_cols(4), np.average(sum2))

    outputsheet.write(2, all_success_cols(5), np.average(intensity))

    outputsheet.write(2, all_success_cols(6), np.average(coefficient))

```

```
def all_success_fail(): # 全社-成立-不成立
```

```
    relbook=xlrd.open_workbook('cdata.xls')
```

```
    relsheet=relbook.sheet_by_index(1)
```

```
    number, name1, name2, sum1, sum2, intensity, coefficient = (), (), (), (), (),
```

```
    (), ()
```

```
    for i in range(0, len(code)):
```

```
        for j in range(0, len(code)):
```

```
            vector = parameter_array(i, :) - parameter_array(j, :)
```

```
            plus=0
```

```
            minus=0
```

```
            for k in range(len(vector)):
```

```
                if vector(k)>0 : plus+=vector(k)
```

```
                else : minus+=vector(k)
```

```
            if (relsheet.cell(code(i)+1, code(j)+1).value != 1) and
```

```
(relsheet.cell(code(i)+1, code(j)+1).value != 0) and (code(i) != code(j)):
```

```
                number.append(str(code(i)) + " - " + str(code(j)))
```



```

        name1.append(name(i))

        name2.append(name(j))

        sum1.append(sum(i))

        sum2.append(sum(j))

        intensity.append( sqrt( (2*(4*len(vector)/2))**2 ) -
sqrt( (4*len(vector)/2-plus)**2 + (-4*len(vector)/2-minus)**2 ) )

        coefficient.append( 1 - sqrt( (4*len(vector)/2-plus)**2
+ (-4*len(vector)/2-minus)**2 ) / sqrt( (2*(4*len(vector)/2))**2 ) )

    for i in all_success_fail_cols:

        outputsheet.write(1, i, len(number))

    for i in range(0, len(number)):

        outputsheet.write(i+3, all_success_fail_cols(0), number(i))

        outputsheet.write(i+3, all_success_fail_cols(1), name1(i))

        outputsheet.write(i+3, all_success_fail_cols(2), sum1(i))

        outputsheet.write(i+3, all_success_fail_cols(3), name2(i))

        outputsheet.write(i+3, all_success_fail_cols(4), sum2(i))

        outputsheet.write(i+3, all_success_fail_cols(5), intensity(i))

        outputsheet.write(i+3, all_success_fail_cols(6), coefficient(i))

    outputsheet.write(2, all_success_fail_cols(2), np.average(sum1))

    outputsheet.write(2, all_success_fail_cols(4), np.average(sum2))

    outputsheet.write(2, all_success_fail_cols(5), np.average(intensity))

    outputsheet.write(2, all_success_fail_cols(6), np.average(coefficient))

```

```
def all_fail(): # 全社-不成立
```

```
    relbook=xlrd.open_workbook('cdata.xls')
```

```

relsheets=relbook.sheet_by_index(1)

number, name1, name2, sum1, sum2, intensity, coefficient = (), (), (), (), (),
(), ()

for i in range(0, len(code)):
    for j in range(0, len(code)):
        vector = parameter_array(i, :) - parameter_array(j, :)

        plus=0
        minus=0

        for k in range(len(vector)):
            if vector(k)>0 : plus+=vector(k)
            else : minus+=vector(k)

        if (relsheets.cell(code(i)+1, code(j)+1).value != 0) and
(code(i) != code(j)):

            number.append(str(code(i)) + " - " + str(code(j)))

            name1.append(name(i))
            name2.append(name(j))

            sum1.append(sum(i))
            sum2.append(sum(j))

            intensity.append( sqrt( (2*(4*len(vector)/2))**2 ) -
sqrt( (4*len(vector)/2-plus)**2 + (-4*len(vector)/2-minus)**2 ) )

            coefficient.append( 1 - sqrt( (4*len(vector)/2-plus)**2
+ (-4*len(vector)/2-minus)**2 ) / sqrt( (2*(4*len(vector)/2))**2 ) )

        for i in all_fail_cols:
            outputsheet.write(1, i, len(number))

        for i in range(0, len(number)):
            outputsheet.write(i+3, all_fail_cols(0), number(i))

```

```

        outputsheet.write(i+3, all_fail_cols(1), name1(i))

        outputsheet.write(i+3, all_fail_cols(2), sum1(i))

        outputsheet.write(i+3, all_fail_cols(3), name2(i))

        outputsheet.write(i+3, all_fail_cols(4), sum2(i))

        outputsheet.write(i+3, all_fail_cols(5), intensity(i))

        outputsheet.write(i+3, all_fail_cols(6), coefficient(i))

    outputsheet.write(2, all_fail_cols(2), np.average(sum1))

    outputsheet.write(2, all_fail_cols(4), np.average(sum2))

    outputsheet.write(2, all_fail_cols(5), np.average(intensity))

    outputsheet.write(2, all_fail_cols(6), np.average(coefficient))

inputbook=xlrd.open_workbook(' cdata.xls')

inputsheet=inputbook.sheet_by_index(0)

code, name, sum = (), (), ()

parameter_array = np.zeros((0))

for row in range(11, inputsheet.nrows):

    code.append(int(inputsheet.cell(row,0).value))

    name.append(inputsheet.cell(row,1).value)

    sum.append(inputsheet.cell(row,37).value)

    parameter_record = np.zeros((0))

    for i in range(parameter_start, parameter_start+parameters):

        parameter_record = np.append(parameter_record,

int(inputsheet.cell(row,i).value))

        parameter_array = np.concatenate((parameter_array, parameter_record), axis=0)

parameter_array.resize((len(code), parameters))

```

```
outputbook=xlwt.Workbook()
```

```
outputsheet=outputbook.add_sheet('sheet 1')
```

```
title()
```

```
success()
```

```
fail()
```

```
all()
```

```
all_success()
```

```
all_success_fail()
```

```
all_fail()
```

```
outputbook.save('comp.xls')
```

```
ここまで-----
```

付録5 : 加算モデル・相乗モデルの Python 言語のプログラミング

```
# -*- coding: utf-8 -*-

import xlrd, xlwt

import numpy as np

from math import sqrt

INPUT_FILE = './cdata.xls'      # 入力ファイル

OUTPUT_FILE = './comp2.xls'    # 出力ファイル

xl_Data_Start = 11             # Excel の何行目からデータが始まるか

xl_Status_Start = 29          # 強み弱みの開始列

xl_Status_Items = 8           # 強み弱みの項目数

xl_Area_Start = 42            # 地域の開始列

xl_Area_Items = 7            # 地域の項目数

xl_Motive_Col = 49           # やる気の列

vector = 8                    # 相互補完のベクトル数

Comp_Max = sqrt(2*((4*vector/2)**2))

Area_Weight = (8, 4, 3, 1, 9, 12, 5)  # 地域の重み付け(首都圏, 関西, 東海, 北陸, その
他, ネット, 海外)

Area_Max = np.sum(Area_Weight) * 2    # 地域を加重加算したときの最大値

Motive_Min = (0.8, 0.6, 0.4)        # やる気の換算範囲
```

```

Motive_Max = (1.2, 1.4, 1.6)

# 出力項目の列番号

brank1, ¥

Code1_Col, Code2_Col, Name1_Col, Name2_Col, Comp_Col, Areas_Col, Motive1_Col,
Motive2_Col, ¥

ResultA_Col, RatioA_Col, ResultM1_Col, ResultM2_Col, ResultM3_Col, RatioM1_Col,
RatioM2_Col, RatioM3_Col, ¥

brank2, ¥

S_Code1_Col, S_Code2_Col, S_Name1_Col, S_Name2_Col, S_Comp_Col, S_Areas_Col,
S_Motive1_Col, S_Motive2_Col, ¥

S_ResultA_Col, S_RatioA_Col, S_ResultM1_Col, S_ResultM2_Col, S_ResultM3_Col,
S_RatioM1_Col, S_RatioM2_Col, S_RatioM3_Col, ¥

brank3, ¥

F_Code1_Col, F_Code2_Col, F_Name1_Col, F_Name2_Col, F_Comp_Col, F_Areas_Col,
F_Motive1_Col, F_Motive2_Col, ¥

F_ResultA_Col, F_RatioA_Col, F_ResultM1_Col, F_ResultM2_Col, F_ResultM3_Col,
F_RatioM1_Col, F_RatioM2_Col, F_RatioM3_Col ¥

=

(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, ¥
31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50)

```

```
class Analyze:
```

```
def __init__(self):
```

```

self.__code = ()          # 企業番号
self.__name = ()         # 企業名

self.__alliance = ()     # 2社間の成立,不成立

self.__comps = ()        # 強み弱み 152×152 社

self.__areas = ()        # 地域 152×152 社

self.__motive = ()       # やる気
self.__motive_comb = ()  # やる気×やる気

self.__result_add = ()   # 加算値
self.__ratio_add = ()    # 係数

self.__result_mul = ()   # 加算相乗値
self.__ratio_mul = ()    # 係数

def get_lastrow(self, sheet):
    lastrow = 0
    for row in range(xl_Data_Start, sheet.nrows):
        if sheet.cell(row,0).value == '': break
        lastrow+=1
    self.__lastrow = xl_Data_Start + lastrow
    print "the last row is " + str(self.__lastrow) + " (" + str(self.__lastrow-

```

```
xl_Data_Start) + " entries)"
```

```
def get_codename(self, sheet):  
    print "append code, name"  
  
    for row in range(xl_Data_Start, self.__lastrow):  
        self.__code.append(int(sheet.cell(row, 0).value))  
        self.__name.append(sheet.cell(row, 1).value)  
  
def calc(self, sheet):  
    print "calculating"  
    self.calc_comp(sheet)  
    self.calc_area(sheet)  
    self.calc_motive(sheet)  
  
    result_add, ratio_add = (), ()  
    result_max_add = Comp_Max + Area_Max  
    for i in range(len(self.__comps)):  
        result_add = self.__comps(i) + self.__areas(i)  
        ratio_add = result_add / result_max_add  
        self.__result_add.append(result_add)  
        self.__ratio_add.append(ratio_add)  
    #print len(self.__result_add)  
    #print len(self.__ratio_add)
```



```

for i in range(len(Motive_Max)): # 0,1,2

    result_mul, ratio_mul = (), ()

    result_max_mul = (Comp_Max + Area_Max) * Motive_Max(i) * Motive_Max(i)

    #print result_max_add

    #print result_max_mul

    for j in range(len(self.__comps)):

        r_mul = (self.__comps(j) + self.__areas(j)) *

self.__motive_comb(i+1)(j)

        result_mul.append(r_mul)

        ratio_mul.append(r_mul/result_max_mul)

        r_mul = 0

    self.__result_mul.append(result_mul)

    self.__ratio_mul.append(ratio_mul)

    #print self.__ratio_mul

    #print len(self.__ratio_mul)

def calc_comp(self, sheet):

    print "append comp"

    status_array = np.zeros((0))

    for row in range(xl_Data_Start, self.__lastrow):

        status_record = np.zeros((0))

        for col in range(xl_Status_Start, xl_Status_Start+xl_Status_Items):

            status_record = np.append(status_record,

```

```

int(sheet.cell(row,col).value))

        status_array = np.concatenate((status_array, status_record), axis=0)

status_array.resize((len(self.__code), xl_Status_Items))

comp = ()

for i in range(len(self.__code)):

    for j in range(len(self.__code)):

        vector = status_array(i,:) - status_array(j,:)

        plus = 0

        minus = 0

        for k in range(len(vector)):

            if vector(k)>0 : plus+=vector(k)

            else : minus+=vector(k)

            comp = sqrt( 2*((4*len(vector)/2)**2) ) - sqrt( (4*len(vector)/2-
plus)**2 + (-4*len(vector)/2-minus)**2 )

            self.__comps.append(comp)

#print self.__comps

#print len(self.__comps)

def calc_area(self, sheet):

    print "append area"

    areas, area_sum = (), ()

    area_array = np.zeros((0))

    for row in range(xl_Data_Start, self.__lastrow):

```

```

area_record = np.zeros((0))

for col in range(xl_Area_Start, xl_Area_Start+xl_Area_Items):

    if sheet.cell(row,col).value == '': # regard empty cell as 0

        area_record = np.append(area_record, 0)

    else: # if 1, multiply area weights

        area_record = np.append(area_record,

int(sheet.cell(row,col).value) * Area_Weight(col-xl_Area_Start))

        area_array = np.concatenate((area_array, area_record), axis=0)

        #print area_record

area_array.resize((len(self.__code), xl_Area_Items))

for i in range(len(self.__code)): # add each area scores

    for j in range(len(self.__code)):

        area = area_array(i,:) + area_array(j,:)

        areas.append(area)

area_sum = np.sum(areas, axis=1) # total of each 7 areas

for i in range(len(area_sum)):

    self.__areas.append(area_sum(i))

#print self.__areas

#print len(self.__areas)

def calc_motive(self, sheet):

```

```

print "append motive"

motive_input = ()

for row in range(xl_Data_Start, self.__lastrow):

    motive_input.append(int(sheet.cell(row, xl_Motive_Col).value))

self.__motive.append(motive_input)

for i in range(len(Motive_Max)):

    motive_weight = ()

    for j in range(len(motive_input)):

        motive_weight.append(((Motive_Max(i)-Motive_Min(i))/100.0) *
motive_input(j) + Motive_Min(i))

    self.__motive.append(motive_weight)

for i in range(len(self.__motive)):

    motive_comb = ()

    for j in range(len(motive_input)):

        for k in range(len(motive_input)):

            motive_comb.append(self.__motive(i)(j)*self.__motive(i)(k))

    self.__motive_comb.append(motive_comb)

#print len(self.__motive)

#print len(motive_input)

```

```

def check_alliance(self): # 152*(x-1) + y - 1

    print "check alliances"

    relbook=xlrd.open_workbook(INPUT_FILE)

    relsheet=relbook.sheet_by_index(1)

    for i in range(len(self.__code)):

        for j in range(len(self.__code)):

            if relsheet.cell(self.__code(i)+1, self.__code(j)+1).value == 1 :

                self.__alliance.append(1)

            elif relsheet.cell(self.__code(i)+1, self.__code(j)+1).value == 0 :

                self.__alliance.append(0)

            else:

                self.__alliance.append(2)

    #print self.__alliance

def write_title(self, sheet):

    print "write output titles"

    sheet.write(0, brank1, u'  全社 >>')

    sheet.write(0, Code1_Col, u'  企業番号 1')

    sheet.write(0, Code2_Col, u'  企業番号 2')

    sheet.write(0, Name1_Col, u'  企業名 1')

    sheet.write(0, Name2_Col, u'  企業名 2')

    sheet.write(0, Comp_Col, u'  相互補完')

```

```
sheet.write(0, Areas_Col, u'地域合計')
sheet.write(0, Motive1_Col, u'やる気1')
sheet.write(0, Motive2_Col, u'やる気2')
sheet.write(0, ResultA_Col, u'加算値')
sheet.write(0, RatioA_Col, u'加算係数')
sheet.write(0, ResultM1_Col, str(Motive_Min(0))+u'-' +str(Motive_Max(0)))
sheet.write(0, ResultM2_Col, str(Motive_Min(1))+u'-' +str(Motive_Max(1)))
sheet.write(0, ResultM3_Col, str(Motive_Min(2))+u'-' +str(Motive_Max(2)))
sheet.write(0, RatioM1_Col, str(Motive_Min(0))+u'-' +str(Motive_Max(0)))
sheet.write(0, RatioM2_Col, str(Motive_Min(1))+u'-' +str(Motive_Max(1)))
sheet.write(0, RatioM3_Col, str(Motive_Min(2))+u'-' +str(Motive_Max(2)))
sheet.write(0, brank2, u'成立 >>')
sheet.write(0, S_Code1_Col, u'企業番号1')
sheet.write(0, S_Code2_Col, u'企業番号2')
sheet.write(0, S_Name1_Col, u'企業名1')
sheet.write(0, S_Name2_Col, u'企業名2')
sheet.write(0, S_Comp_Col, u'相互補完')
sheet.write(0, S_Areas_Col, u'地域合計')
sheet.write(0, S_Motive1_Col, u'やる気1')
sheet.write(0, S_Motive2_Col, u'やる気2')
sheet.write(0, S_ResultA_Col, u'加算値')
sheet.write(0, S_RatioA_Col, u'加算係数')
sheet.write(0, S_ResultM1_Col, str(Motive_Min(0))+u'-' +str(Motive_Max(0)))
sheet.write(0, S_ResultM2_Col, str(Motive_Min(1))+u'-' +str(Motive_Max(1)))
sheet.write(0, S_ResultM3_Col, str(Motive_Min(2))+u'-' +str(Motive_Max(2)))
sheet.write(0, S_RatioM1_Col, str(Motive_Min(0))+u'-' +str(Motive_Max(0)))
```

```
sheet.write(0, S_RatioM2_Col, str(Motive_Min(1))+u'-' +str(Motive_Max(1)))
sheet.write(0, S_RatioM3_Col, str(Motive_Min(2))+u'-' +str(Motive_Max(2)))
sheet.write(0, brank3, u'不成立 >>')
sheet.write(0, F_Code1_Col, u'企業番号1')
sheet.write(0, F_Code2_Col, u'企業番号2')
sheet.write(0, F_Name1_Col, u'企業名1')
sheet.write(0, F_Name2_Col, u'企業名2')
sheet.write(0, F_Comp_Col, u'相互補完')
sheet.write(0, F_Areas_Col, u'地域合計')
sheet.write(0, F_Motive1_Col, u'やる気1')
sheet.write(0, F_Motive2_Col, u'やる気2')
sheet.write(0, F_ResultA_Col, u'加算値')
sheet.write(0, F_RatioA_Col, u'加算係数')
sheet.write(0, F_ResultM1_Col, str(Motive_Min(0))+u'-' +str(Motive_Max(0)))
sheet.write(0, F_ResultM2_Col, str(Motive_Min(1))+u'-' +str(Motive_Max(1)))
sheet.write(0, F_ResultM3_Col, str(Motive_Min(2))+u'-' +str(Motive_Max(2)))
sheet.write(0, F_RatioM1_Col, str(Motive_Min(0))+u'-' +str(Motive_Max(0)))
sheet.write(0, F_RatioM2_Col, str(Motive_Min(1))+u'-' +str(Motive_Max(1)))
sheet.write(0, F_RatioM3_Col, str(Motive_Min(2))+u'-' +str(Motive_Max(2)))
```

```
def write(self, sheet):
```

```
    print "write output value"
```

```
    row = 1
```

```
    value = 0
```

```

for i in range(0, len(self.__code)):
    for j in range(0, len(self.__code)):
        if i != j:
            sheet.write(row, Code1_Col, self.__code(i))
            sheet.write(row, Code2_Col, self.__code(j))
            sheet.write(row, Name1_Col, self.__name(i))
            sheet.write(row, Name2_Col, self.__name(j))
            sheet.write(row, Comp_Col, self.__comps(value))
            sheet.write(row, Areas_Col, self.__areas(value))
            sheet.write(row, Motive1_Col, self.__motive(0)(i))
            sheet.write(row, Motive2_Col, self.__motive(0)(j))
            sheet.write(row, ResultA_Col, self.__result_add(value))
            sheet.write(row, RatioA_Col, self.__ratio_add(value))
            for k in range(len(self.__result_mul)): # 0,1,2
                sheet.write(row,
                            ResultM1_Col+k,
self.__result_mul(k)(value))
                sheet.write(row, RatioM1_Col+k, self.__ratio_mul(k)(value))
            row+=1
            value+=1

row = 1
value = 0
for i in range(0, len(self.__code)):
    for j in range(0, len(self.__code)):
        if i != j and self.__alliance(value) == 1:
            sheet.write(row, S_Code1_Col, self.__code(i))

```



```

        sheet.write(row, S_Code2_Col, self.__code(j))

        sheet.write(row, S_Name1_Col, self.__name(i))

        sheet.write(row, S_Name2_Col, self.__name(j))

        sheet.write(row, S_Comp_Col, self.__comps(value))

        sheet.write(row, S_Areas_Col, self.__areas(value))

        sheet.write(row, S_Motive1_Col, self.__motive(0)(i))

        sheet.write(row, S_Motive2_Col, self.__motive(0)(j))

        sheet.write(row, S_ResultA_Col, self.__result_add(value))

        sheet.write(row, S_RatioA_Col, self.__ratio_add(value))

        for k in range(len(self.__result_mul)): # 0,1,2

            sheet.write(row,

                        S_ResultM1_Col+k,

self.__result_mul(k)(value))

            sheet.write(row,

                        S_RatioM1_Col+k,

self.__ratio_mul(k)(value))

            row+=1

            value+=1

row = 1

value = 0

for i in range(0, len(self.__code)):

    for j in range(0, len(self.__code)):

        if i != j and self.__alliance(value) == 0:

            sheet.write(row, F_Code1_Col, self.__code(i))

            sheet.write(row, F_Code2_Col, self.__code(j))

            sheet.write(row, F_Name1_Col, self.__name(i))

            sheet.write(row, F_Name2_Col, self.__name(j))

```

```

        sheet.write(row, F_Comp_Col, self.__comps(value))

        sheet.write(row, F_Areas_Col, self.__areas(value))

        sheet.write(row, F_Motive1_Col, self.__motive(0)(i))

        sheet.write(row, F_Motive2_Col, self.__motive(0)(j))

        sheet.write(row, F_ResultA_Col, self.__result_add(value))

        sheet.write(row, F_RatioA_Col, self.__ratio_add(value))

        for k in range(len(self.__result_mul)): # 0,1,2

            sheet.write(row,
                        F_ResultM1_Col+k,
self.__result_mul(k)(value))

            sheet.write(row,
                        F_RatioM1_Col+k,
self.__ratio_mul(k)(value))

            row+=1

            value+=1

```

```

if __name__ == '__main__':

```

```

    analyze = Analyze()

```

```

    inputbook=xlrd.open_workbook(INPUT_FILE)

```

```

    inputsheet=inputbook.sheet_by_index(0)

```

```

    analyze.get_lastrow(inputsheet)

```

```

    analyze.get_codename(inputsheet)

```

```

    analyze.calc(inputsheet)

```

```
analyze.check_alliance()

outputbook=xlwt.Workbook()

outputsheet=outputbook.add_sheet('sheet 1')

analyze.write_title(outputsheet)

analyze.write(outputsheet)

try:

    outputbook.save(OUTPUT_FILE)

except Exception, e:

    print e

    pass
```