

学位論文 博士 (工学)

変数選択制約と楕円形判別面を考慮した
信用リスク判別モデルの効率的解法と評価

2023 年度

慶應義塾大学大学院理工学研究科

田中 克弘

目次

第 1 章	序論	1
1.1	研究の背景	1
1.2	統計型モデルの概観と問題点	2
1.3	統計型モデルの改良に関する先行研究	5
1.4	本論文の目的と貢献	6
1.5	本論文の構成	8
第 2 章	統計型信用リスクモデルの概観	10
2.1	記号の定義	10
2.2	判別モデル	11
2.2.1	ロジスティック回帰モデル	11
2.2.2	SVM	12
2.2.3	エラスティックネット	16
2.3	判別モデル評価指標	17
第 3 章	楕円形判別面と変数選択制約による問題の拡張と MISDP 計算用アルゴリズム	19
3.1	定式化	19
3.1.1	楕円形への拡張	19
3.1.2	変数選択制約の導入	20
3.2	アルゴリズム	22
3.2.1	切除平面アルゴリズム	23
3.2.2	MISDP の構造についての解釈	24
3.2.3	ヒューリスティックアルゴリズム	26

第 4 章	MISDP に拡張した CVaR 最小化モデルの評価	29
4.1	序論	29
4.2	モデルの定式化	31
4.2.1	SVM ベースの CVaR 最小化モデル	31
4.2.2	楕円形の導入	32
4.2.3	変数選択制約の導入	33
4.3	MISDP の解法	34
4.3.1	切除平面アルゴリズムへの実装	34
4.3.2	ヒューリスティックアルゴリズムの実装	35
4.4	計算機実験	36
4.4.1	セットアップ	36
4.4.2	比較用の判別面モデル	37
4.4.3	ヒューリスティックアルゴリズムの評価	38
4.4.4	汎化性能の評価	40
4.5	結論	42
第 5 章	MISDP に拡張した bAUC 最大化モデルの評価	44
5.1	序論	44
5.2	bAUC 最大化モデル	46
5.2.1	AUC	46
5.2.2	bAUC	47
5.2.3	bAUC の最大化	49
5.2.4	定式化	50
5.3	MISDP への拡張	51
5.3.1	楕円分類器	51
5.3.2	離散変数を用いた変数選択	52
5.3.3	混合整数半正定値計画モデル	53
5.4	MISDP の解法	53
5.4.1	切除平面アルゴリズムへの実装	54
5.4.2	分枝切除アルゴリズムへの実装	54
5.4.3	ヒューリスティックアルゴリズム	55
5.4.3.1	ヒューリスティックアルゴリズムのベース部分の実装	55
5.4.3.2	高速化のための離散変数の削減部分の実装	56

5.5	計算機実験	57
5.5.1	比較用の分類器モデル	58
5.5.2	セットアップ	58
5.5.2.1	データセット	58
5.5.2.2	基本設定	59
5.5.3	アルゴリズム性能評価	59
5.5.3.1	詳細設定	60
5.5.3.2	計算結果	61
5.5.4	判別性能評価	65
5.5.4.1	シミュレートデータを用いての判別性能評価	65
5.5.4.2	実データを用いての判別性能評価	66
5.6	結論	73
第 6 章	結論と今後の課題	76
	参考文献	78
	謝辞	84
	付録	86
	付録 A 4 章で使用されている説明変数の一覧	86
	付録 B HA+ 内の Step2 の詳細	87
	付録 C 5 章で使用されている説明変数の一覧	88

目次

1.1	全体構成	8
2.1	最小幾何マージン最大化の例示	13
2.2	CVaR 最小化モデルの例示	16
3.1	変数の候補数と選択上限数が大きく離れていないケースの解の違い	24
3.2	変数の候補数と選択上限数がある程度離れているケースの解の違い	25
5.1	POE ($P(H(\mathbf{w}) \geq \tilde{q})$) と bPOE ($p_{\tilde{q}}(H(\mathbf{w}))$)	48
5.2	実データの歪度と尖度	68
5.3	計算手順の概要図	69
5.4	各 bAUC 最大化モデルによるテストデータによる AUC 値	71
5.5	(E) と (L) および機械学習モデルによるテストデータによる AUC 値	73

表目次

1.1	CVaR 最小化モデルと bAUC 最大化モデルの違いおよび各検証内容一覧	7
4.1	各データセットの高格付と低格付の企業数	37
4.2	パラメータの検証範囲	38
4.3	アルゴリズム別の計算時間 (単位: 分)	39
4.4	$P = 51$ での二次係数の各絶対値の和	39
4.5	各アルゴリズムによる解の相対誤差 (単位: %)	40
4.6	$P = 51, S = 10$ における判別面毎の計算時間 (単位: 分)	40
4.7	$P = 51$ での (ES) による解の AR 値 (単位: %)	41
4.8	$S = 10$ での判別面毎の AR 値 (単位: %)	42
5.1	bAUC 最大化モデルに関する先行研究と本章の貢献箇所	46
5.2	各年度のデータセットの格付別の企業数	59
5.3	ソフトウェアと計算環境	60
5.4	パラメータの範囲	60
5.5	10800 秒以内に 10 個のインスタンスから求解できた数	61
5.6	$P = 20$ での計算時間 (単位: 秒)	62
5.7	$P = 50$ での計算時間 (単位: 秒)	63
5.8	$P = 20$ でトレーニングデータから得られた bAUC (単位: %)	64
5.9	$P = 50$ でトレーニングデータから得られた bAUC (単位: %)	65
5.10	bAUC 最大化モデルおよび LR から得られる AUC 値 (単位: %)	67
5.11	(E)、(L)、(Q) の評価に用いる説明変数の個数	69
5.12	各 bAUC 最大化モデルによる AUC 値 (単位: %)	70
5.13	モデル別のトレーニングとテストデータによる AUC 値 (単位: %)	72
5.14	モデル別のテストデータによる bAUC 値 (単位: %)	74

A.1	説明変数の一覧	86
C.2	説明変数の一覧	88

第1章

序論

1節では研究の背景について、主に個別企業の信用リスクの計測が求められるニーズやその評価に用いるモデルを紹介し、2節では本論文で焦点を当てる統計型モデルの説明とその問題点について解説する。3節ではそれらの問題点を改善するための方法について、既存研究で提案されている手法を説明し、4節で本論文の目的について説明する。5節では論文の構成を述べる。

1.1 研究の背景

リスク管理の重要な領域の一つに信用リスクがある。信用リスクは、広く言えば、特定の企業の信用力の低下、更に最悪の場合は倒産により被る損失を指す。この信用リスク管理を行うべき場面が、金融機関には多くある。例えば、債券などの発行体のリスクを有する商品を保有している際は、その発行体の信用力の上下により保有する商品の時価の変動を注視する必要がある。またスワップ取引のように取引先のリスクを有する商品を保有している際は、その取引先の倒産で、相手先と結んでいる金融商品の清算が困難になることに伴い、損失が発生する可能性がある。このようなリスクの管理を遂行するため、信用リスクを適正に評価する方法が必要になる。

信用リスクを評価するアプローチは大きく二つある。一つは、債券やクレジットデリバティブといった信用リスクを含んだ商品の時価を評価するアプローチで、数理ファイナンスの枠組みを用いる。もう一つは、様々な個別企業の信用リスクを示す倒産確率や格付のような指標を推計するアプローチ（統計型モデル）で、主に統計学の枠組みを用いる。本論文では統計型モデルに焦点を当て、その判別性能の向上を目的として議論を行う。

1.2 統計型モデルの概観と問題点

信用リスク評価に用いられる統計型モデルは様々なモデルが提案されている。統計型モデルの初期的研究の代表に、Altman (1968) が提案した、財務データを用いて、存続企業と倒産企業を判別する線形回帰式がある。なお、これは Z-score と呼ばれている。これが出発点となり、今日までに信用リスクを評価するための様々なモデルが提案されている。

その中で特に広く知られているモデルにロジスティック回帰モデル (LR) がある。LR は、各企業が存続もしくは倒産のいずれかの状態に属する確率を、ロジスティック分布を前提に定式化して、構築されている。その際、回帰分析の手法を基に、各企業の倒産の可能性の大小を、Z-score 同様に線形回帰式で表現している。そして、その線形回帰式を入力値に用いて、倒産確率を推計する。このモデルは、あるサンプルデータに占める倒産企業の倒産確率および存続企業の存続確率の累積確率を最大化する最尤推定法によりパラメータを推計することで、判別性能を高めることを目的としている。そして、このモデルが広く知られている理由は、その構築方法がわかりやすいことに加え、パラメータの推計が凹関数の最大化問題になるので大域的最適解が容易に得られることがある。このモデルの実証例として Laitinen and Laitinen (2000) などがあり、また 株式会社日本格付研究所 (2011) は、実際の格付付与の業務に、LR を使用していることを公表している。

他にも最近注目が集まっているモデルに、Zuo and Hastie (2005) が提案したエラスティックネットがある。エラスティックネットは、最小二乗法をベースに、Tibshirani (1996) が提案したラッソ回帰および Hoerl and Kennard (1970) が提案したリッジ回帰の二つを取り込んだモデルである。これら二つを簡単に説明する。

ラッソ回帰は係数の絶対値の合計値となる 1 ノルムの項と、最小二乗法を組み合わせるパラメータ推計を行う回帰モデルである。このモデルは、1 ノルムの項を小さくすることで、求める係数の多くをゼロにする効果を持つので、余分な変数が除去され、変数選択を促す。リッジ回帰は係数の二乗の合計値となる 2 ノルムの項と、最小二乗法を組み合わせるパラメータ推計を行う回帰モデルである。このモデルは、2 ノルムの項を小さくすることで、多重共線性の影響を抑制する効果があるので、相関の強い変数同士の選択を回避できる。そして、これら二つのモデルの特性を導入したエラスティックネットの利点は、ある程度の変数選択および多重共線性の回避を同時実行できることにある。更にパラメータ推計が凸型の二次計画問題の最小化モデルとなることから、大域的最適解が容易に得られるという利便性もある。一方で、ラッソ回帰およびリッジ回帰のハイパーパラメータは 1 つだが、エラスティックネットのハイパーパラメータは 2 つなので、チューニングの

負荷が増加する欠点もある。これらのモデルを用いた実証例として、Tian et al. (2015) や Saito et al. (2021) はラッソ回帰により、Saito and Yamanaka (2021) はエラスティックネットワークにより信用リスクの評価を念頭に入れた変数選択を行っている。

エラスティックネットワークは機械学習モデルに属するが、他にも同じ分野で判別性能の高さから注目されているモデルに、Vapnik (1995) が提案したサポートベクターマシーン (SVM) がある。ここでは前提として、倒産企業および存続企業の二つの判別を考える。このとき、SVM は、その判別のための最適な超平面を、双方のグループ同士の中で最も近いサンプルおよびその超平面の距離を最大化することで表現する。これはマージン最大化もしくは幾何マージン最大化と呼ばれている。ただし、このように誤判別が一切なく、綺麗に判別できるデータは稀であるため、誤判別したサンプルと超平面との距離を最小化することも併せて実行する。これはソフトマージン最小化と呼ばれている。このソフトマージン最小化により、SVM は、標準的な凸型の二次計画問題の最小化モデルとなるので、大域的最適解が容易に得られる。

また、この SVM は、正しい判別を適切に実行するための最大化と、誤判別を可能な限り抑制するための最小化と、二つの最適化を同時に実行するために、一つのハイパーパラメータを導入している。そのハイパーパラメータの水準の表現および取り扱いの容易さについても研究され、複数の再定式化したモデルが提案されている (Schölkopf et al., 2000; Perez-Cruz et al., 2003; Gotoh and Takeda, 2005)。特に Gotoh and Takeda (2005) が定式化した、Conditional-value-at-risk (CVaR) 最小化モデルと呼ばれているモデルは、ハイパーパラメータが、誤判別したサンプルと超平面との距離を表現した分布の信頼区間に相当することから、統計学の観点から分析する上でその水準を設定しやすい利点がある。ハイパーパラメータを内包したモデルにおいて、そのハイパーパラメータの水準自体に意味はなく、与えたデータに対して、網羅的な検証を行った上で、適した水準を見定めるケースが多い。しかし、予めその水準に解釈を持てるモデルであれば、使用できる機会はより広まるものと考えている。

このように多くの判別モデルが提示されているので、様々なデータや問題に対して複数の判別モデルを実行した際には、それぞれの判別モデルの性能を評価する必要がある。そのための有名な指標に Area Under the Curve (AUC) がある。そして、この指標が高い判別モデルは、判別の性能が高いモデルと評価できる。この指標はトレーニングに用いるデータから判別を実施するための係数を計測した後、テストデータに対して判別結果の性能を計測する汎化性能の評価で使用されるケースがある。そのため、この AUC を最大化したい一方、離散型の非凸型の目的関数になるため、AUC を直接最大化することは困難と考えられていた。

この課題に対処するため、Norton and Uryasev (2019) は代替指標として buffered Area Under the Curve(bAUC) を挙げると共に、その最大化を提示した。併せて bAUC の最大化は、凸型の問題に定式化できるので容易に解を得られること、および、bAUC の最大化は AUC の最大化につながることも主張している。またハイパーパラメータはわずか 1 個で、更に統計学上ある分位点を上回る範囲の期待値を指すため、水準設定がしやすい利点がある。

bAUC 自体は 2019 年と比較的最近提示された指標である。ただし、この指標は 1 から buffered probability of exceedance(bPOE) を引くことで得られ、その bPOE 自体を用いた実証例は複数ある (Rockafellar and Royset, 2010; Norton et al., 2017; Rockafellar and Uryasev, 2018; Pertaia et al., 2021; Norton et al., 2021)。そのため、bPOE も含めると bAUC の認知は少しずつ得られ始めている段階といえる。

このように様々な統計型モデルが提案されているが、ほぼ全てのモデルは企業の財務指標と倒産/非倒産や信用格付で表現される信用力との関係を線形で表現している。例えば、LR は財務指標を線形結合した式を変換したものを倒産確率の推計に用いており、SVM も財務指標の線形関数を判別面とすることが標準的である。しかしながら、企業の財務指標は必ずしも信用力と線形の関係の有していない。例えば佐野 (1990) は、現金手持日数について、小さくても大きくても企業経営にとって望ましくない財務指標であることを報告している。このような財務指標をモデルに組み込むには線形モデルでは不十分であり、非線形モデルに拡張することでより高い判別精度が実現できる可能性がある。

また統計モデルの構築の際に数多くの財務指標を取り入れると、多重共線性を引き起こす可能性が高まることやオーバーフィットを起こしやすくなることが指摘されている (Guyon and Elisseeff, 2003)。さらに実務で行う結果の解釈も複雑になることから、モデルに取り込む財務指標は必要最低限であることが望ましい。モデルに組み込む財務指標を選択する標準的な方法としてステップワイズ法を用いて財務指標を限定した上で信用リスク評価モデルを構築することが行われている。しかし、ステップワイズ法はヒューリスティックに変数選択を行う方法であり、構築するモデルに対して最適な変数である保証はない。また先述したエラスティックネットは変数選択の効果は持つものの、それはある程度に変数の個数を限定することに留まるものであり、ユーザーが指定した個数まで選択するといった、より強い要求を実現することはできない。

1.3 統計型モデルの改良に関する先行研究

これらの問題点を解決する方法が、いくつかの先行研究で提案されている。まず財務指標の線形性に関する改良として、判別面^{*1}を線形から楕円形に拡張する試みである。判別面を線形から楕円形に拡張することで、説明変数の線形の効果だけでなく、二次の効果ならびに二つの変数の相互関係も表現できることから、モデルの表現力が高まり、それが判別性能の向上につながっていく。ここで二次関数ではなく、楕円形に限定するのは、二つ理由がある。一つはKonno and Wu (2002) の報告によると、企業の信用状態を財務データで評価する場合、楕円形を用いると線形よりも判別性能が高まることに加え、楕円形以上に自由度のある二次関数を用いると線形よりも却って判別性能が下がるためである。もう一つは、判別面の形状を限定することでパラメータの自由度が減り、オーバーフィットを抑えられるためである。よって、楕円形は線形や二次関数と比較して、企業の信用リスクと財務変数の関係性を、より正確に表現できることが示唆される。一方で、二次係数行列に半正定値制約を追加することから、パラメータを推計する問題が半正定値計画問題 (SDP) へと拡張される。これにより、解を得るまでの計算時間がより長くなる。ただし、求める変数が連続変数のみである限りは、一般的な計算環境下では、相応に大きいデータサイズでも解けることが知られている。なお、Konno and Wu (2002); Konno et al. (2003a,b) は LR を、Okada and Konno (2009); Konno and Saito (2013) は SVM を、Konno and Kobayashi (2000); Konno et al. (2002) は判別分析を、Konno et al. (2003a) は最小二乗法をそれぞれ対象にしてモデルを SDP に拡張している。

変数選択に関する改良としては、解の取る値を 0 か 1 に限定した離散変数を導入し、変数選択制約を追加する方法が提案されている。離散変数を用いることで、明確にユーザーが指定した個数だけモデルで変数を選択することができる。先行研究では、Konno and Yamamoto (2008) は最小二乗法に離散変数を導入したモデルを用いることで、既存解法であるステップワイズ法の結果と比較して、より良い変数選択を実行できることを報告している。また、離散変数を導入した結果、混合整数線形計画問題 (MILP) に拡張したことで計算時間の増加は想定されるものの、実用時間内での求解は可能であることを報告している。なお、Tanaka and Nakagawa (2014); Sato et al. (2017); Bertsimas and King (2017); Naganuma et al. (2019) は LR を、Sato et al. (2016) は AIC 最小化モデルを、Saito and Konno (2009); Tanaka and Nakagawa (2014); Maldonado et al. (2014) は

*1 分類器とも呼ばれている。

SVM をそれぞれ対象にしてモデルを MILP に拡張している。

1.4 本論文の目的と貢献

このように線形判別面と変数選択の問題に対し、独立に解決方法を提案する研究は行われており、これらの改良方法を統計型モデルに同時に組み込むことで、より判別性能を向上できる可能性がある。特に楕円判別面を導入する場合、判別面が非線形となるため、候補変数が多いと数多くの財務指標をモデルに組み入れてしまう可能性がある。そのため楕円判別面に変数選択制約を組み込むことは判別精度を向上させるうえで重要性が高いと考えられる。しかしながら既存研究でこれらの改良を統計型モデルに同時に組み込んだ研究は存在しない。その理由はモデルのパラメータを推計することは容易ではないからである。

楕円形を表現するためには半正定値制約を組み込んだ問題から解を得る必要があるが、その問題となる SDP の変数が全て連続変数のみであれば、短時間で計算を終了することが可能である。しかし、SDP に変数選択で使用する離散変数を組み込んだ場合、そのパラメータを推計する問題は混合整数半正定値計画問題 (MISDP) となり、短時間の計算終了が難しくなる。また、CPLEX^{*2}、Gurobi^{*3}、Mosek^{*4}などの商業的なソルバーでも MISDP から直接解を得るための機能は、現在実装されていない。なお非商業用のソルバーであれば SCIP-SDP^{*5}があるものの使用できる場面は非常に限られる^{*6}。そのため、MISDP の解を得るためには、別途専用のアルゴリズムを準備する必要がある。

提案されている既存のアルゴリズムにはKonno and Wu (2002) が提案した切除平面アルゴリズムがある。このアルゴリズムは元々 SDP 用に作られたものであるが、理論的には MISDP にも適用でき、 ϵ -最適解が得られることが保証されている。一方で、計算負荷の観点からは、企業数が数百、財務変数が数十に及ぶデータサイズでは計算を完了できるケースは極めて限定的と予想できる。またKobayashi and Takano (2020) は、切除平面アルゴリズムをより高速化するための分枝切除アルゴリズムを提案した。併せて、切除平面アルゴリズムより分枝切除アルゴリズムはより速く解を得られていることを報告している。ただし、扱った問題の規模は限定的なものである。

そこで本論文は、混合整数半正定値計画問題 (MISDP) に対し、大規模の問題に対して

^{*2} <https://www.ibm.com/products/ilog-cplex-optimization-studio>

^{*3} <https://www.gurobi.com/documentation/9.0/refman/index.html>

^{*4} <https://www.mosek.com/products/mosek/>

^{*5} <http://www.opt.tu-darmstadt.de/scipsdp/>

^{*6} Kobayashi and Takano (2020) の報告から、サンプル数 30 から 50 程度の小さいデータサイズでも問題によっては 7200 秒以内で 5 個のインスタンスで 1 つも解を得られない。

も精度の良い近似解を得られるヒューリスティックアルゴリズムを提案する。そして統計型モデルに楕円形判別面と変数選択を組み込んだ問題を求解することで、提案アルゴリズムの有効性と信用リスク判別問題に対する改良効果を検証する。統計型モデルに関しては、最適化問題の規模が大きくない CVaR 最小化モデルと、規模が大きく近年新たに提案された bAUC 最大化モデルを取り上げて有効性を検証する。2つのモデルの規模は CVaR 最小化モデルが制約式の本数が二クラスに判別するサンプルの合計数となるのに対し、bAUC 最大化モデルは二クラスに判別するサンプルの積の数となる。そのため bAUC 最大化モデルは、制約式の本数が数十倍にもなることから*7、相対的に解を得るための計算時間が相当に長くなる。問題の規模の大きい CVaR 最小化モデルをはじめに取り上げることで、提案したヒューリスティックアルゴリズムと大域的最適解を求解できるアルゴリズムをある程度の規模まで比較可能であり、解の精度をより詳細に評価できる。次に問題の規模の大きい bAUC 最大化モデルを取り上げることで、求解規模の観点でアルゴリズムの有効性を詳細に議論できると考える。以上を含め、CVaR 最小化モデルと bAUC 最大化モデルの違いおよび検証内容の一覧をまとめた表1.1は以下のとおりである。

表1.1 CVaR 最小化モデルと bAUC 最大化モデルの違いおよび各検証内容一覧

	CVaR 最小化モデル	bAUC 最大化モデル
問題規模	中 (制約式本数は二つのグループのデータサイズの「和」)	大 (制約式本数は二つのグループのデータサイズの「積」)
特徴	正しい判別を行う可能性を高める	過度な誤判別 (テイルリスク) を抑制する
ハイパーパラメータ	信頼区間	あるパーセンタイル点を上回る範囲の期待値 (CVaR)

本論文の貢献は、大きく2点ある。1つは大規模な MISDP を求解するヒューリスティックアルゴリズムを提案することである。MISDP は信用リスク判別問題だけでなく、様々な応用分野で利用される最適化問題の1つである。その問題に対し、精度の良い解を短時間で求解できるアルゴリズムを提案することは既存研究にはない大きな貢献となる。も

*7 仮に二つのグループのサンプル数がそれぞれ 200 と 100 の場合、制約式の本数が、SVM および CVaR 最小化モデルは $300(=200+100)$ に対し bAUC 最大化モデルは $20000(=200 \times 100)$ と大きく増加する。

う1つは、信用リスク判別モデルの中の統計型モデルにおいて、判別面の楕円形への拡張と変数選択制約を追加することで判別精度が向上することを実証することである。精度の高い信用リスクモデルの構築は金融実務における重要性の高い課題である。その課題に対し、既存研究にはない改良を行い、その有効性を検証することは金融実務にとっても大きな貢献となると考える。

1.5 本論文の構成

本論文は6章で構成しており、全体の関係図は表1.1のとおりである。

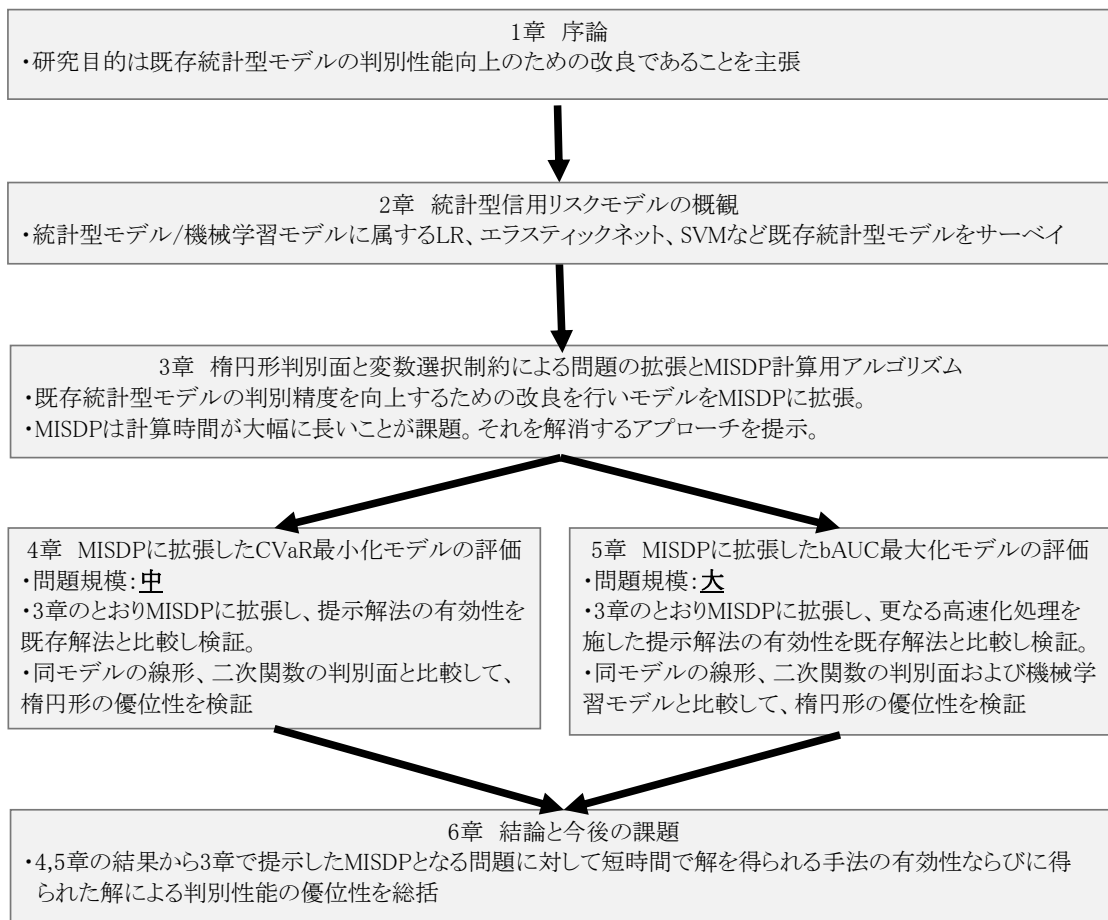


図1.1 全体構成

2章では、統計型信用リスクモデルを代表する LR、SVM、エラスティックネットを説

明するとともに、判別性能を評価する有名な指標である AUC と AR を説明する。3 章では、対象とするモデルは判別面を楕円形とすることで SDP になり、更に変数選択を用いるために離散変数を導入することで MISDP となることを説明するとともに、その MISDP から解を得るための切除平面アルゴリズムとヒューリスティックアルゴリズムを述べる。4 章では、誤判別を示す分布のシナリオをベクトル数で表現した問題規模が大きい SVM をベースに MISDP へと拡張したモデルに対して、ヒューリスティックアルゴリズムを用いることで、 ϵ -最適解は得られないものの、計算誤差は限定的であり、かつ計算速度は最大数百倍であることを示す。更に楕円形の汎化性能が、線形や二次といった既存の判別面と比較検証したところ、最も高いことを、ハイパーパラメータの変化に伴う汎化性能の変化と共に確認する。5 章では、誤判別を示す分布のシナリオを行列で表現した問題規模が相当に大きい bAUC 最大化モデルを MISDP へと拡張したモデルに対しても、ヒューリスティックアルゴリズムが有効であることを示すと共に、4 章で用いた CVaR 最小化モデルに再定式化可能な SVM を含めた既存の有名な機械学習モデルとの比較も網羅的に行った上で、当該モデルから最も高い汎化性能が得られたことを報告する。6 章では、結論と今後の課題を述べる。

第 2 章

統計型信用リスクモデルの概観

この章では 1 節で記号を定義し、2 節では先行研究で取り上げられてきた判別モデルであるロジスティック回帰、SVM、エラスティックネットを取り上げる。最後の 3 節は判別モデルの評価指標である AR および AUC を説明する。

2.1 記号の定義

まず定式の準備として各記号を定義する。 $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m)\}$ を与えられたデータセットとし、 $\mathbf{x}_i \in \mathbf{R}^P$ はサンプル i に対する説明変数を指し、 $y_i \in \{\pm 1\}$ はそのラベルである。また、 $i \in \mathcal{I} := \{1, \dots, m\}$ 、 $p \in \mathcal{P} := \{1, \dots, P\}$ とおく。

更に、 $\mathcal{I}_+ := \{i \mid y_i = 1, i \in \mathcal{I}\}$ と $\mathcal{I}_- := \{i \mid y_i = -1, i \in \mathcal{I}\}$ は、二つのラベルが属するグループの集合とする。 $m_+ := |\mathcal{I}_+|$ と $m_- := |\mathcal{I}_-|$ は、それぞれのグループに属するサンプルの個数である。なお、これらから $\mathcal{I} = \mathcal{I}_+ \cup \mathcal{I}_-$ と $m = m_+ + m_-$ となる。

ここで係数ベクトル $\mathbf{b} \in \mathbf{R}^P$ を用いて次の線形式を定式化する。

$$R(\mathbf{b}|\mathbf{x}_i) = \mathbf{b}^T \mathbf{x}_i \quad (i \in \mathcal{I}). \quad (2.1)$$

ただし、切片 $b_0 \in \mathbf{R}$ を導入することで、次のように記載されている場合もある。

$$R(\mathbf{b}|\mathbf{x}_i) + b_0 = \mathbf{b}^T \mathbf{x}_i + b_0 \quad (i \in \mathcal{I}). \quad (2.2)$$

また (2.1) あるいは (2.2) を、信用リスクの分野では、信用スコアとも呼ばれている。

そして、 \mathcal{I}_+ は信用力の高い企業 (例. 高格付が付与された企業、非倒産企業など)、 \mathcal{I}_- は信用力の低い企業 (例. 低格付が付与された企業、倒産企業など) を指すものとする。

2.2 判別モデル

2.2.1 ロジスティック回帰モデル

ここでは、McFadden (1974) が提案したロジスティック回帰モデル (LR) を説明する。LR は、サンプル毎に各グループに属する確率をロジスティック分布を前提として定式化し、そのうえでそれらの累積確率を最大化することで、パラメータを推計する。LR が広く知られている理由は、そのモデル構築方法がわかりやすいことに加え、凹関数の最大化問題に定式化できるので大域的最適解が容易に得られるためである。

まずサンプル $i^+ \in \mathcal{I}_+$ が \mathcal{I}_+ に属する確率 $Pr(R(\mathbf{b}|\mathbf{x}_{i^+}), b_0)$ を以下のとおり定式化する。なお、木島・小守林 (1999) を参照に、個別企業の信用スコアが高いほど信用リスクが低く (信用力が高く) なるよう記述している。

$$Pr(R(\mathbf{b}|\mathbf{x}_{i^+}), b_0) = \frac{\exp(R(\mathbf{b}|\mathbf{x}_{i^+}) + b_0)}{1 + \exp(R(\mathbf{b}|\mathbf{x}_{i^+}) + b_0)} \quad (i^+ \in \mathcal{I}_+) \quad (2.3)$$

次にサンプル $i^- \in \mathcal{I}_-$ が \mathcal{I}_- に属する確率 $1 - Pr(R(\mathbf{b}|\mathbf{x}_{i^-}), b_0)$ も以下のとおり定式化する。

$$1 - Pr(R(\mathbf{b}|\mathbf{x}_{i^-}), b_0) = \frac{1}{1 + \exp(R(\mathbf{b}|\mathbf{x}_{i^-}) + b_0)} \quad (i^- \in \mathcal{I}_-) \quad (2.4)$$

ここで (2.3) と (2.4) を掛け合わせて、次の尤度関数 $L(\mathbf{b}|\mathbf{x}_i, i \in \mathcal{I})$ を定式化する。

$$\begin{aligned} L(\mathbf{b}, b_0|\mathbf{x}_i, i \in \mathcal{I}) &= \prod_{i^+ \in \mathcal{I}_+} Pr(R(\mathbf{b}|\mathbf{x}_{i^+}), b_0) \prod_{i^- \in \mathcal{I}_-} (1 - Pr(R(\mathbf{b}|\mathbf{x}_{i^-}), b_0)), \\ &= \prod_{i^+ \in \mathcal{I}_+} \frac{\exp(R(\mathbf{b}|\mathbf{x}_{i^+}) + b_0)}{1 + \exp(R(\mathbf{b}|\mathbf{x}_{i^+}) + b_0)} \prod_{i^- \in \mathcal{I}_-} \frac{1}{1 + \exp(R(\mathbf{b}|\mathbf{x}_{i^-}) + b_0)}, \\ &= \frac{\prod_{i^+ \in \mathcal{I}_+} \exp(R(\mathbf{b}|\mathbf{x}_{i^+}) + b_0)}{\prod_{i \in \mathcal{I}} 1 + \exp(R(\mathbf{b}|\mathbf{x}_i) + b_0)}. \end{aligned} \quad (2.5)$$

更に計算を容易にするため (2.5) の対数を取る。

$$\ln L(\mathbf{b}, b_0|\mathbf{x}_i, (i \in \mathcal{I})) = \sum_{i^+ \in \mathcal{I}_+} (R(\mathbf{b}|\mathbf{x}_{i^+}) + b_0) - \sum_{i \in \mathcal{I}} \ln(1 + \exp(R(\mathbf{b}|\mathbf{x}_i) + b_0)). \quad (2.6)$$

最後に (2.6) を最大化する次のモデルを定式化する。

$$\max_{\mathbf{b}, b_0} \ln L(\mathbf{b}, b_0 | \mathbf{x}_i (i \in \mathcal{I})). \quad (2.7)$$

(2.7) が LR と呼ばれており、凹型の最大化モデルなので、標準的なソルバーを用いることで、容易に最適解を得ることができる。

なお、LR を判別に用いる際は一つ注意がある。それは LR が定式化する対象は確率であるため、その確率を用いて別途判別する手順が必要になる点である。そこで、例えば Konno et al. (2003b) は、LR で計測した各サンプルの確率に対して、予め定めた閾値を超えたか否かで二つのどちらかのグループに属するか判別することで、LR の汎化性能を評価している。

2.2.2 SVM

Vapnik (1995) が提示した SVM は LR と異なり分布の前提を置かず、与えられたデータに対して、判別を行うモデルである。このモデルは、二つのグループが完全に線形判別可能な場合を前提とした際、判別を行う超平面と各グループの中で最も近いサンプルとの距離を最大化するモデルである。またこの距離は幾何学的要素を持ち、最小幾何最大化マージンと呼ばれる (図2.1を参照)。一方で、実データには線形判別が不可能なケースが多数ある。そこで、線形判別不可能なサンプルと判別面との距離の最小化を同時に実施する。これは、ソフトマージン最小化と呼ばれる。これにより、如何なるデータセットに対しても二グループの判別を可能とする。

SVM は、先行研究で複数の定式化が提案されている。最初に Vapnik (1995) が提示した C-SVM と呼称されるモデルがあり、その後ハイパーパラメータの取り扱いに注目して Schölkopf et al. (2000) は ν -SVM、Perez-Cruz et al. (2003) は $E\nu$ -SVM、そして Gotoh and Takeda (2005) は統計型モデルである CVaR 最小化モデルと呼ばれるモデルへの再定式化を提案している。これらを以下に紹介する。

まず C-SVM について説明する。ここでもし入力するデータセットが線形判別可能であれば、次のとおり判別できる。

$$f(\mathbf{b}, b_0 | \mathbf{x}_i)_i = y_i(R(\mathbf{b}, b_0 | \mathbf{x}_i) - b_0) > 0 \quad (i \in \mathcal{I}). \quad (2.8)$$

その場合、図2.1のとおり閾値 b_0 とサンプル $i \in \mathcal{I}$ との距離の最小値の最大化を考える。

このとき、次の問題の解を得ることで、適切な判別のための超平面を求めることがで

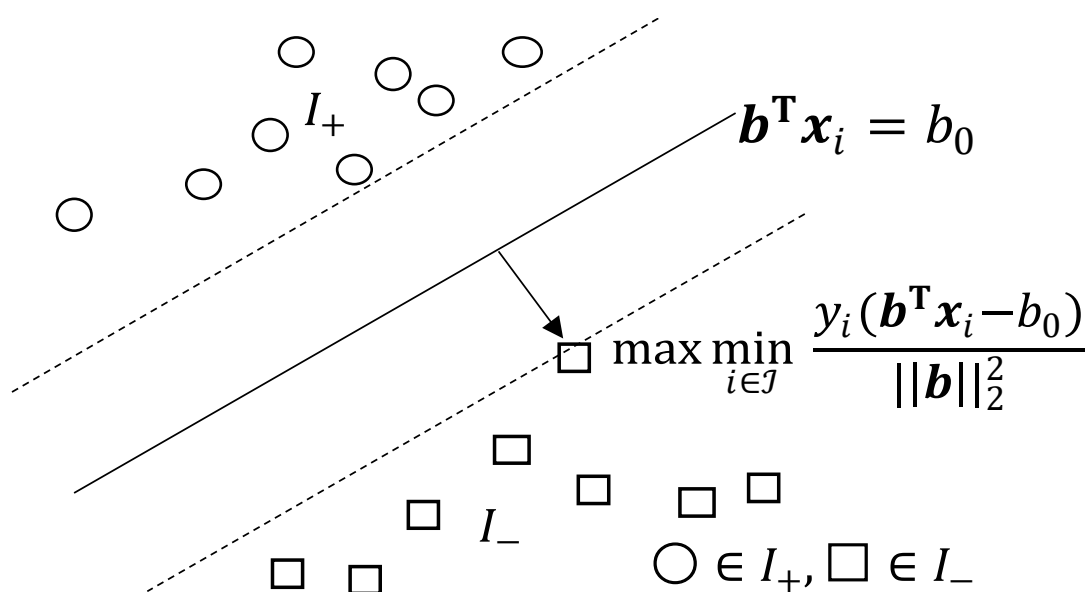


図2.1 最小幾何マージン最大化の例示

きる。

$$\left| \max_{\mathbf{b}, b_0} \min_{i \in \mathcal{I}} \frac{f(\mathbf{b}, b_0 | \mathbf{x}_i)_i}{\|\mathbf{b}\|_2^2} \right. \quad (2.9)$$

次に、(2.9) に変数 α を導入して、 $f(\mathbf{b}, b_0 | \mathbf{x}_i)_i, i \in \mathcal{I}$ に注目して、次のとおりに書き直す。

$$\left| \begin{array}{l} \max_{\mathbf{b}, b_0, \alpha} \quad \alpha \\ \text{s.t.} \quad \|\mathbf{b}\|_2^2 \\ f(\mathbf{b}, b_0 | \mathbf{x}_i)_i \geq \alpha, \quad i \in \mathcal{I}. \end{array} \right. \quad (2.10)$$

最後に、Charnes-cooper 変換を用いて、目的関数の分母と分子を変換することで、次のとおり二次計画問題に定式化できる。

$$\left| \begin{array}{l} \min_{\mathbf{b}, b_0} \quad \|\mathbf{b}\|_2^2 \\ \text{s.t.} \quad f(\mathbf{b}, b_0 | \mathbf{x}_i)_i \geq 1, \quad i \in \mathcal{I}. \end{array} \right. \quad (2.11)$$

先述したとおり、問題 (2.11) はデータセットが線形判別可能なケースは解を得られるが、線形判別不可能なケースでは解を得られない。しかしその不可能なケースは現実的に頻繁に存在するので、判別する上で正しく判別できないサンプルも考慮する必要がある。そこで、ここからはその線形判別不可能なケースの判別を考える。

この点においてVapnik (1995) が提示したアイデアは、誤判別したサンプルと閾値 b_0 との距離を、可能な限り小さくすることである。そこでスラック変数 $\xi_i, i \in \mathcal{I}$ を (2.11) を導入することで、次の C-SVM を提示している。

$$\left| \begin{array}{ll} \min_{\mathbf{b}, b_0, \xi_i (i \in \mathcal{I})} & \|\mathbf{b}\|_2^2 + C \frac{1}{m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & f(\mathbf{b}, b_0 | \mathbf{x}_i)_i + \xi_i \geq 1 (i \in \mathcal{I}), \\ & \xi_i \geq 0 (i \in \mathcal{I}). \end{array} \right. \quad (\text{C-SVM})$$

(C-SVM) は二次計画問題の最小化モデルなので容易に解を得られる。

ここで C は、目的関数の第一項と第二項の最適化におけるバランスをチューニングするためにユーザーが設定するハイパーパラメータである。そして、この C の値の決定において理論的根拠がない点が、(C-SVM) の使用や分析を困難にさせていることを、Tanaka and Nakagawa (2014) は指摘している。

次にSchölkopf et al. (2000) は ν -SVM を提示した。 ν -SVM は、(C-SVM) を基にして変数 ρ を導入した、次のモデルである。

$$\left| \begin{array}{ll} \min_{\mathbf{b}, b_0, \rho, \xi_i (i \in \mathcal{I})} & \|\mathbf{b}\|_2^2 - \nu\rho + \frac{1}{m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & f(\mathbf{b}, b_0 | \mathbf{x}_i)_i + \xi_i \geq \rho (i \in \mathcal{I}), \\ & \xi_i \geq 0 (i \in \mathcal{I}), \\ & \rho > 0. \end{array} \right. \quad (\nu\text{-SVM})$$

ここで $\nu \in (0, 1]$ はハイパーパラメータである。また、(ν -SVM) も (C-SVM) 同様に二次計画問題の最小化モデルなので容易に解を得られる。

更にPerez-Cruz et al. (2003) は $E\nu$ -SVM と呼ばれる次の式を提示している。

$$\left| \begin{array}{ll} \min_{\mathbf{b}, b_0, \rho, \xi_i (i \in \mathcal{I})} & -\nu\rho + \frac{1}{m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & f(\mathbf{b}, b_0 | \mathbf{x}_i)_i + \xi_i \geq \rho (i \in \mathcal{I}), \\ & \xi_i \geq 0 (i \in \mathcal{I}), \\ & \|\mathbf{b}\|_2^2 = 1. \end{array} \right. \quad (E\nu\text{-SVM})$$

($E\nu$ -SVM) は (ν -SVM) の目的関数に置いてあるノルムの項 ($\|\mathbf{b}\|_2^2$) を、制約条件に移したモデルである。そのため非凸なノルム制約があるので、($E\nu$ -SVM) は非凸の最適化モデルになる。これが原因で、($E\nu$ -SVM) は標準的なソルバーでは容易には解を得られないが、Takeda and Sugiyama (2008) が提案しているアルゴリズムを別途用いることで、解を得ることができる。

更にGotoh and Takeda (2005) は (E ν -SVM) を、次のとおりに再定式化している。
(図2.2参照)

$$\left| \begin{array}{ll} \min_{\mathbf{b}, b_0, \alpha, \xi_i (i \in \mathcal{I})} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i + f(\mathbf{b}, b_0 | \mathbf{x}_i)_i \geq -\alpha (i \in \mathcal{I}), \\ & \xi_i \geq 0 (i \in \mathcal{I}), \\ & \|\mathbf{b}\|_2^2 = 1. \end{array} \right. \quad (2.12)$$

ここで β はユーザーが設定するパラメータで、統計分野で分布の信頼区間を意味する。また $\beta \in [1 - 2 \min\{m_+, m_-\}, 1)$ となるので、 β の下限は各グループのデータのサンプル数で決まる。

なお、後藤・山本 (2012) は、(E ν -SVM) の変数とパラメータを、次のとおり置き換えたモデルが (2.12) であると説明している。

$$\begin{cases} 1 - \nu & \rightarrow \beta, \\ -\rho & \rightarrow \alpha. \end{cases} \quad (2.13)$$

また (2.12) の目的関数は CVaR と呼ばれる指標で、金融分野で統計学を適用したりリスク計測を実施する際に、広く使われている指標である。この CVaR を最小化したモデルはRockafellar and Uryasev (2000) が別に提示しており、それぞれのモデルの目的関数は共通している*¹。またTakeda (2009) は、CVaR 最小化モデルは、トレーニングデータとテストデータに対する判別性能の乖離を表す汎化誤差を抑止する特徴を持つことを報告している。

なお、CVaR の定義は次のとおりで、ある分布の $(1 - \beta)\%$ を超える値の期待値を指す。

$$\begin{aligned} \alpha &= VaR_\beta = \inf\{a \in \mathbf{R} | P(\xi_i > a) \leq 1 - \beta, (i \in \mathcal{I})\}, \\ CVaR_\beta &= E[\xi_i | \xi_i \geq VaR_\beta, (i \in \mathcal{I})], \end{aligned} \quad (2.14)$$

ここで α は $\xi_i (i \in \mathcal{I})$ の分布におけるパーセンタイル点である。

SVM 全般のメリットとして、LR と異なり、モデル自体で判別を実施している点にある。よって別途判別のための閾値設定も不要なので、LR より使用しやすいと考える。

*¹ Rockafellar and Uryasev (2000) はポートフォリオ最適化を目的としたモデルを組んでおり、制約式に違いがある。具体的には、非凸なノルム制約はなく、投資制約を表現した線形の制約式を記述している。

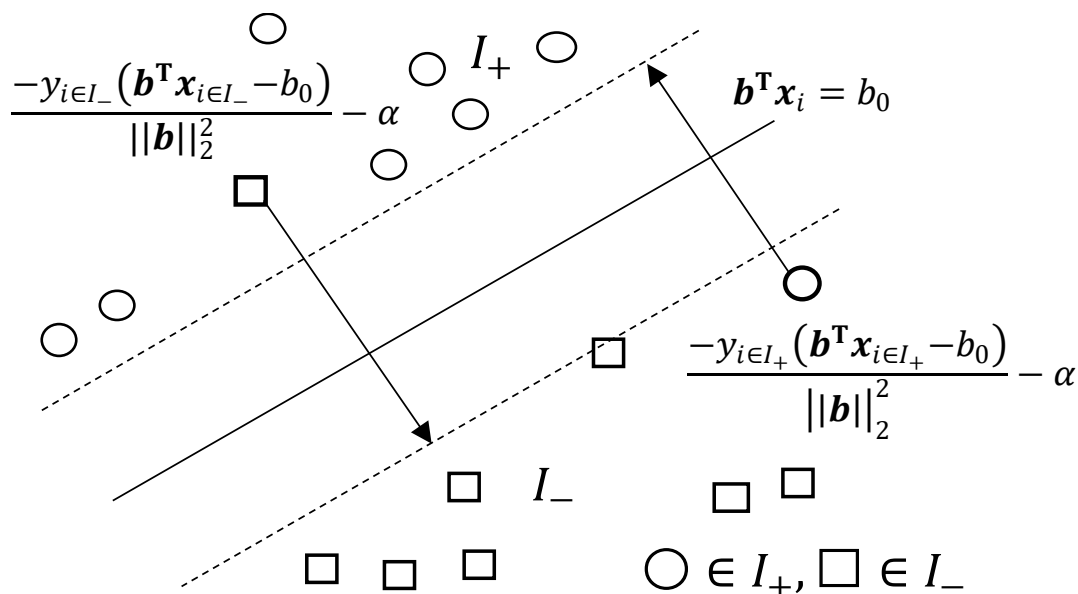


図2.2 CVaR 最小化モデルの例示

2.2.3 エラスティックネット

まず一般に知られている最小二乗法は以下の通りである。

$$\left| \min_{\mathbf{b}, b_0} \frac{1}{m} \sum_{i \in \mathcal{I}} [y_i - \{b_0 + R(\mathbf{b} | \mathbf{x}_i)\}]^2 \right. \quad (2.15)$$

ここで、 $\hat{\mathbf{x}} = (\mathbf{x}_1, \dots, \mathbf{x}_I)^\top$ 、 $\hat{\mathbf{y}} = (y_1, \dots, y_I)^\top$ を導入したとき、(2.15) の解 \mathbf{b}^{OLS} は次のとおりである。

$$\mathbf{b}^{OLS} = (\hat{\mathbf{x}} \hat{\mathbf{x}}^\top)^{-1} (\hat{\mathbf{x}}^\top \hat{\mathbf{y}}^\top). \quad (2.16)$$

次にHoerl and Kennard (1970) は、(2.15) に係数ベクトルの 2 ノルムの最小化を追加したリッジ回帰を提案した。

$$\left| \min_{\mathbf{b}, b_0} \frac{1}{m} \sum_{i \in \mathcal{I}} [y_i - \{b_0 + R(\mathbf{b} | \mathbf{x}_i)\}]^2 + \theta \|\mathbf{b}\|_2^2 \right. \quad (2.17)$$

ここで $\theta \geq 0$ はハイパーパラメータであり、単位行列 \mathbf{E} を導入することで、(2.17) の解 \mathbf{b}^{ridge} は次のとおりである。

$$\mathbf{b}^{ridge} = (\hat{\mathbf{x}} \hat{\mathbf{x}}^\top + \theta \mathbf{E})^{-1} (\hat{\mathbf{x}}^\top \hat{\mathbf{y}}^\top). \quad (2.18)$$

(2.18) からわかることとして、 θ を大きくすると、係数の分散が小さくなることから、計測した係数の信頼性が高まる (廣瀬, 2016)。これにより相関関係が強い変数同士を選択することで発生する多重共線性の影響を抑えることができる。ただし θ を大きくしすぎると、全ての解が 0 なるので、適切な値を設定する必要がある。一方で、妥当な水準の θ を決定しても、2 ノルムの最小化を行う場合、0 に非常に近い係数が複数残る課題がある。

そこで、Tibshirani (1996) は (2.15) に係数ベクトルの 2 ノルムではなく 1 ノルム最小化を実施したラッソ回帰を提案した。

$$\left| \min_{\mathbf{b}, b_0} \frac{1}{m} \sum_{i \in \mathcal{I}} [y_i - \{b_0 + R(\mathbf{b}|\mathbf{x}_i)\}]^2 + \Theta \|\mathbf{b}\|_1 \right. \quad (2.19)$$

ここで $\Theta \geq 0$ はハイパーパラメータである。

(2.19) は Θ を大きくすることで、一定個数の変数の係数を 0 にできるので、ある程度の変数選択を実行できるといえる。また 1 ノルムの最小化は、2 ノルムの減少に寄与するので、多重共線性の回避にもある程度効果がある。

最後に (2.15) にリッジ回帰とラッソ回帰の双方を組み込んだ Zuo and Hastie (2005) が提案しているエラスティックネットは、次のとおりである。なお、表記は、Friedman et al. (2010) を参照にしている。

$$\left| \min_{\mathbf{b}, b_0} \frac{1}{m} \sum_{i \in \mathcal{I}} [y_i - \{b_0 + R(\mathbf{b}|\mathbf{x}_i)\}]^2 + \theta_2 \left(\frac{1 - \theta_1}{2} \|\mathbf{b}\|_2^2 + \theta_1 \|\mathbf{b}\|_1 \right) \right. \quad (2.20)$$

ここで $\theta_1 \in \{0, 1\}$ 、 $\theta_2 \geq 0$ はハイパーパラメータである。

これにより、多重共線性を抑える 2 ノルムの最小化と、変数選択を実行する 1 ノルムの最小化を同時に実施する。更にパラメータ推計が二次計画問題の最小化モデルとなるので、大域的最適解が容易に得られるという利点もある。ただし、ハイパーパラメータが増えるので、(2.18)、(2.19) よりもチューニングの負荷が増加する欠点もある。

2.3 判別モデル評価指標

個々のモデルによる判別性能を評価する良く知られた指標に Accuracy Ratio(AR) と AUC がある。AR は信用リスク管理業務において、個別企業の存続と倒産の判別性能の評価に、また、AUC は金融分野に限らず、統計学分野で二値判別モデルの評価に広く使われている*2。ただし、双方の指標の特徴は同じで、違いは定義域が違う程度である。

*2 山下・三浦 (2011) は、AR は日本で、AUC はグローバルに使われている場面が多いと記述している。

これらの指標は、個別企業の存続と倒産、もしくは高格付と低格付というような二値判別に関し、各グループの判別性能ではなく全体としての判別性能、言い換えれば順位性を評価するものである。そして、指標の値が高いほど、モデルの判別性能が高いことを示す。

それぞれの指標は次のとおりに、定義されている。まず AR は次のとおりで、その範囲は $AR \in (-1, 1)$ である。

$$AR := \frac{2}{m_+ m_-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \mathbb{1}_{(R(\mathbf{b}|\mathbf{x}_{i_+}) - R(\mathbf{b}|\mathbf{x}_{i_-}))} - 1. \quad (2.21)$$

ここで $\mathbb{1}_{R(\mathbf{b}|\mathbf{x}_{i_+}) - R(\mathbf{b}|\mathbf{x}_{i_-})}$ は次に示している指示関数である。

$$\mathbb{1}_{R(\mathbf{b}|\mathbf{x}_{i_+}) - R(\mathbf{b}|\mathbf{x}_{i_-})} := \begin{cases} 1, & \text{if } R(\mathbf{b}|\mathbf{x}_{i_+}) - R(\mathbf{b}|\mathbf{x}_{i_-}) \geq 0, \\ 0, & \text{else if } R(\mathbf{b}|\mathbf{x}_{i_+}) - R(\mathbf{b}|\mathbf{x}_{i_-}) < 0, \end{cases} \quad (i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-). \quad (2.22)$$

次に AUC は次のとおりで、その範囲は $AUC \in (0, 1)$ である。

$$AUC := \frac{1}{m_+ m_-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \mathbb{1}_{(R(\mathbf{b}|\mathbf{x}_{i_+}) - R(\mathbf{b}|\mathbf{x}_{i_-}))}. \quad (2.23)$$

また、(2.21) および (2.23) より、AR と AUC は次の関係にある。

$$AR = 2AUC - 1. \quad (2.24)$$

山下・三浦 (2011) によると、複数の統計型モデルによる企業の信用力の調査結果から AR は大体 0.7 前後、併せて AUC で 0.85 前後の値であると報告している。なお AR が 0、AUC が 0.5 であるとき無作為な判別を実施していることを意味する。

第 3 章

楕円形判別面と変数選択制約による 問題の拡張と MISDP 計算用アルゴ リズム

この章では楕円形判別面と変数選択制約の導入による MISDP への拡張とその計算アルゴリズムを説明する。

3.1 定式化

この節では判別モデルを MISDP へ拡張することについて説明する。具体的には、半正定値制約を加えることで判別面を線形から楕円形に拡張できること、および離散変数の導入で変数選択を実行できること、そしてそれらを組み込むことで MISDP となることを説明する。

3.1.1 楕円形への拡張

前章に説明したモデルの信用スコアは、(2.1) にあるとおり、線形を前提としている。ここでは、半正定値制約を加えることで、線形から楕円形へと拡張する。まず二次係数行列 $B \in \mathbf{R}^{P \times P}$ を導入して、次のとおり定式化する。

$$r(\mathbf{b}, \mathbf{B} | \mathbf{x}_i) = \mathbf{b}^T \mathbf{x}_i + \mathbf{x}_i^T \mathbf{B} \mathbf{x}_i. \quad (3.1)$$

(3.1) は、楕円形以外にもなり得るので、次のとおり半正定値制約を追加して、楕円形に固定する。

$$\begin{aligned} r(\mathbf{b}, \mathbf{B} | \mathbf{x}_i) &= \mathbf{b}^T \mathbf{x}_i + \mathbf{x}_i^T \mathbf{B} \mathbf{x}_i, \\ \mathbf{B} &\succeq 0. \end{aligned} \quad (3.2)$$

線形では自由度が P なのに対し、半正定値制約がない (3.1) の自由度は $0.5P^2 + 1.5P$ と非常に大きい。しかし Okada and Konno (2009) の Section 2.1 の記述のとおり半正定値制約がある (3.2) の自由度は $3P - 1$ まで小さくなる。例として、仮に Shirakawa (2001) が提示した 18 個の説明変数を用いた場合、自由度は線形 (2.1) は 18、二次関数 (3.1) は 189、楕円形 (3.2) は 53 になる。このため、楕円形のパラメータ数は、二次関数と同じであるが、関数形の指定で自由度を減らすことができる。そして、これによりオーバーフィットの回避が期待できる。

3.1.2 変数選択制約の導入

2.3 節で述べたラッソ回帰およびその要素を取り込んだエラスティックネットにも変数選択の効果はある。ただし、これらのモデルによる効果は、判別への影響が小さい説明変数の係数を 0 にしやすくすることで、変数がある程度削減する程度である。そのため、ステップワイズ法のように明確に指定した個数だけ、変数を選択するほどの強い効果はない。

一方で、Konno and Yamamoto (2008) は、事前にユーザーが指定した個数以下の変数選択を可能にするため、0 か 1 となる離散変数を導入した MILP への定式化を提案した。

ここからはその制約式を説明する。まず $\mathbf{z} \in \{0, 1\}^P$ を導入する。ここで信用スコアが線形 (2.1) の場合、ユーザーが選択したい個数を $S (< P)$ で指定するための、次の制約式を定式化する。

$$\sum_{p_1 \in \mathcal{P}} z_{p_1} \leq S. \quad (3.3)$$

また不要な変数に相当する係数を 0 にするために次の制約式を定式化する。

$$-z_{p_1} M \leq b_{p_1} \leq z_{p_1} M, \quad (p_1, \in \mathcal{P}), \quad (3.4)$$

この式により、(3.4) で得られた解 $\mathbf{z}^*, \mathbf{b}^*$ には次の関係性がある。

$$\begin{cases} -M \leq b_{p_1}^* \leq M, & \text{if } z_{p_1}^* = 1, \\ b_{p_1}^* = 0, & \text{otherwise.} \end{cases} \quad (p_1, \in \mathcal{P}), \quad (3.5)$$

ここで M は十分大きな値を入力するユーザーが定義する入力パラメータである。この方法論は big-M 法と呼ばれる。

これらの制約式 (3.3)、(3.4) を追加することで変数選択が可能になる。また、Konno and Yamamoto (2008) は最小二乗法にこの制約を取り込んだモデルを提案し、ベンチマークとするステップワイズ法と比較検証を行った。その結果、提案モデルのほうが最適な変数の組み合わせを選択できたことを報告している。

更に、このアプローチは線形ではなく二次関数にも適用可能である。Saito and Konno (2009) は、二次関数用に次の制約式を定式化している。

$$\begin{aligned} -z_{p_1} M &\leq B_{p_1, p_2} \leq z_{p_1} M \\ -z_{p_2} M &\leq B_{p_1, p_2} \leq z_{p_2} M \end{aligned} \quad (p_1, p_2 \in \mathcal{P}). \quad (3.6)$$

(3.6) により得られた解 \mathbf{z}^* , \mathbf{B}^* には次の関係性がある。

$$\begin{cases} -M \leq B_{p_1, p_2}^* \leq M, & \text{if } (z_{p_1}^*, z_{p_2}^*) = (1, 1), \\ B_{p_1, p_2}^* = 0, & \text{otherwise.} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (3.7)$$

(3.7) は二つの変数が選ばれた場合のみ、それに相当する要素の係数を計算することがわかる。

なお、(3.4) と (3.6) はユーザーが定義するパラメータである M をある程度大きな値に設定する必要がある。ノルム制約など他の制約式から係数の上限または下限が判断できる場合は、容易に設定できるが、そのようなモデルは多くはない。また最適解を求めるためには十分大きな値を設定したい一方で、計算時間を抑制するためにはなるべく小さい値が望ましい。このため、big-M 法は、問題によっては、 M に与える値を考慮する必要がある。

一方で、これを不要とする記載方法である CPLEX^{*1}などの商業用ソルバーで標準的に実装している Special order set1(SOS1) と呼ばれている関数を紹介する。SOS1 とは指定した変数の内、最大でも一つは非ゼロにするというものである。これを用いて、次の制約式を定式化する^{*2}。

$$z_{p_1} = 0 \Rightarrow b_{p_1} = 0 \quad (p_1 \in \mathcal{P}). \quad (3.8)$$

また (3.8) により得られた解を \mathbf{z}^* , \mathbf{b}^* と置いたとき、次の関係性がある。

$$\begin{cases} -\infty \leq b_{p_1}^* \leq \infty, & \text{if } z_{p_1}^* = 1, \\ b_{p_1}^* = 0, & \text{otherwise,} \end{cases} \quad (p_1 \in \mathcal{P}). \quad (3.9)$$

^{*1} <https://www.ibm.com/products/ilog-cplex-optimization-studio>

^{*2} プログラム上は $\text{SOS1}(1 - z_{p_1}, b_{p_1}) \quad (p_1 \in \mathcal{P})$ と実装する

同様に次の制約式を定式化する*3。

$$z_{p_1} = 0 \text{ もしくは } z_{p_2} = 0 \Rightarrow B_{p_1, p_2} = 0 \quad (p_1, p_2 \in \mathcal{P}). \quad (3.10)$$

(3.10) により得られた解を \mathbf{z}^* , \mathbf{B}^* と置いたとき、次の関係性がある。

$$\begin{cases} -\infty \leq B_{p_1, p_2}^* \leq \infty, & \text{if } (z_{p_1}^*, z_{p_2}^*) = (1, 1), \\ B_{p_1, p_2}^* = 0, & \text{otherwise.} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (3.11)$$

この SOS1 は、big-M 法と違って、ユーザーが定義するパラメータがないことから、扱いやすくなる場面もある。

なお、4章で取り扱うモデルはノルム制約から M を 1 に設定できるので big-M 法を、5章で取り扱うモデルは M が設定できないので SOS1 を用いて計算している。

3.2 アルゴリズム

楕円形判別面と変数選択を統計型信用リスク判別モデルに追加することで、パラメータ推計を行う最適化問題が、MISDP として定式化できる。MISDP は求解に要する計算時間が非常に長い問題であるため、実用的な規模の問題から解を得ることは既存のアルゴリズムでは難しい。そこでこの節では、MISDP を解くための既存のアルゴリズムを説明したあと、大規模な問題を求解するためのヒューリスティックアルゴリズムを提案する。

単純化のため、変数となる行列 $\mathbf{B} \in \mathbf{R}^{P \times P}$ 、 $\mathbf{z} \in \{0, 1\}^P$ を導入し、次の MISDP を考える。

$$\left| \begin{array}{ll} \min_{\mathbf{B}, \mathbf{z}} & g(\mathbf{B}) \\ \text{s.t.} & \mathbf{z}\mathbf{z}^T \bullet \mathbf{B} \succeq 0, \quad \dots (\text{半正定値制約}) \\ & \sum_{p \in \mathcal{P}} z_p \leq S, \quad \dots (\text{変数選択制約}) \\ & (3.6) \text{ もしくは } (3.10). \end{array} \right. \quad (3.12)$$

ここで $g(\mathbf{B})$ は任意の凸関数とし、 $\mathbf{z}\mathbf{z}^T \bullet \mathbf{B} \succeq 0$ は半正定値制約を意味する。また $S > 0$ はユーザーが定義する入力パラメータである。

また、問題 (3.12) から離散変数および変数選択制約を除去した次の SDP となる問題も準備する。

$$\left| \begin{array}{ll} \min_{\mathbf{B}} & g(\mathbf{B}) \\ \text{s.t.} & \mathbf{B} \succeq 0. \quad \dots (\text{半正定値制約}) \end{array} \right. \quad (3.13)$$

*3 プログラム上は $\text{SOS1}(2 - (z_{p_1} + z_{p_2}), B_{p_1, p_2})$ ($p_1, p_2 \in \mathcal{P}$) と実装する

3.2.1 切除平面アルゴリズム

一つ目のアルゴリズムは切除平面アルゴリズムである。このアルゴリズムはKonno et al. (2003a) が Theorem 1 で示しているとおり、 ε -最適解を求めることができる。このアルゴリズムは、半正定値制約を緩和した問題を繰り返し計算しながら、そこで得た解を基に適宜制約条件を追加することで、最終的に近似的に半正定値制約を満たした解を得る手順である。変数が全て連続変数であれば、繰り返し凸型の最適化問題を計算するだけなので、計算時間は限定的である。しかし、離散変数が含まれる場合は、繰り返し MILP を計算しなくてはならないので、計算時間が飛躍的に長くなる。その結果、当該アルゴリズムは MISDP に対し、極めてデータサイズが小さい場合を除くと計算が終了しないケースが多発する。

切除平面アルゴリズムの手順は次のとおりである。

切除平面アルゴリズム (CPA)

Step1 $k \leftarrow 0$. $\varepsilon > 0$. $\mathcal{F}^k := \{zz^T \bullet B \mid B^T = B, B_{p,p} \geq 0, p \in \mathcal{P}\}$.

Step2 次の問題を計算する。

$$\left| \begin{array}{ll} \min_{B,z} & g(B) \\ \text{s.t.} & zz^T \bullet B \in \mathcal{F}^k, \\ & \sum_{p \in \mathcal{P}} z_p \leq S, \\ & (3.6) \text{もしくは}(3.10), \end{array} \right. \quad (3.14)$$

ここでの解を B^* とする。

Step3 B^* の最小固有値が $-\varepsilon$ を超えると終了。そうでなければ、 $-\varepsilon$ 以下の固有値を n_k とし、それらの固有値に相当する基準化された固有ベクトル d_l を計算したうえで、次のとおり \mathcal{F}^{k+1} を定義する:

$$\mathcal{F}^{k+1} := \mathcal{F}^k \cap \{zz^T \bullet B \mid d_l^T (zz^T \bullet B) d_l \geq 0, l = 1, 2, \dots, n_k\} \quad (3.15)$$

Step4 $k \leftarrow k + 1$. Step2 に戻る。

ここで $\varepsilon > 0$ はユーザーが定義する入力パラメータである。 ε を 0 に近づけるほど、 B^* はより半正定値行列の形に近づいていくことから、計算精度がより高まる。一方で、繰り返し計算する量が増加することから、計算時間も徐々に長くなる。そのためKonno et al.

(2003a,b) は、計算精度と計算時間の双方を考慮したところ、 ε の値は少なくとも 10^{-3} であれば、線形よりも高い判別性能を得られることを報告している。

3.2.2 MISDP の構造についての解釈

問題 (3.12) は半正定値制約と離散変数による変数選択制約の双方を組み込んでいることから、解を得るまでの計算時間が非常に長くなる。特に変数選択制約が整数変数を含むことが求解の難易度を高めている。そこで変数選択制約を含む問題である MISDP と含まない問題である SDP を比較することで、問題同士の特性を整理する。

まずは、 $P = 25$ から $S = 20$ を選択するような、変数の候補数と選択上限数が大きく離れていないケースを考える。以下は、横軸を MISDP である (3.12) で得た行列 (\mathbf{B}) の要素、また縦軸を SDP である問題 (3.13) で得た行列の要素をプロットした一例の図である。

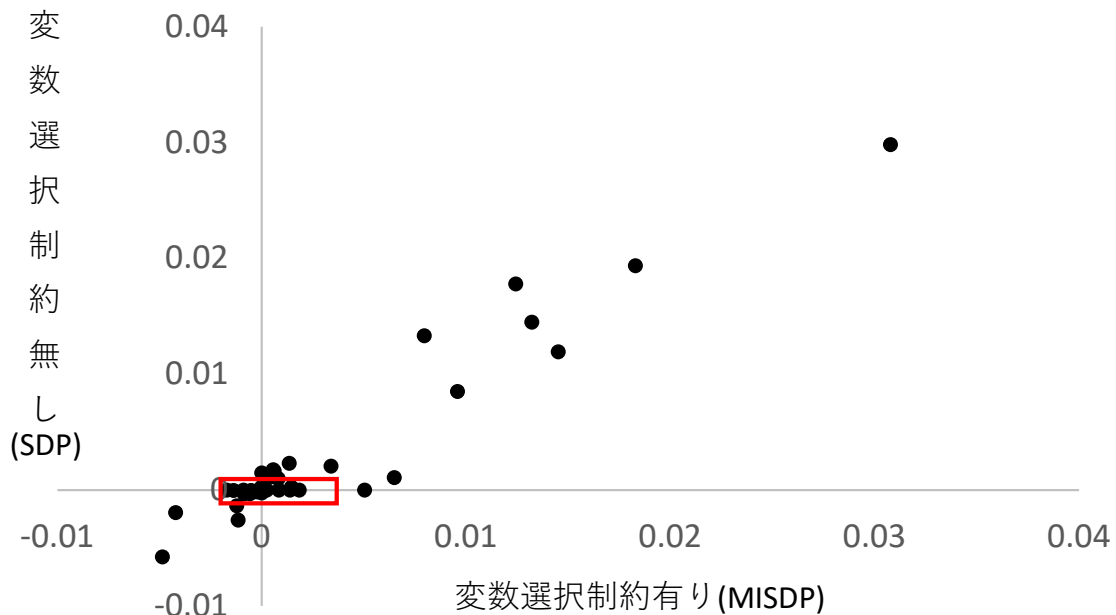


図3.1 変数の候補数と選択上限数が大きく離れていないケースの解の違い

図3.1から、二つのことが読み取れる。一つは0から十分に離れた水準である要素ほど、双方とも非ゼロとなりやすいことである。つまり要素の水準の絶対値が高いほど、変数選択制約の有無に関わらず同一の要素が選択されやすいということである。もう一つは、枠線箇所のとおり、SDPでは0となる選択されない要素に対して、MISDPでは非ゼロとな

り選択されるケースが存在することである。しかし、それに該当する要素の個数は僅少であること、および相当する要素の水準はいずれも相対的に小さいことから、これらの要素の選択の違いが判別に与える影響は限定的と想定できる。よって、SDP あるいは MISDP のそれぞれから得た行列同士は、判別に与える影響が高い要素ほど共通して選択される傾向が強いこと、および、共通して選択されない要素はあるもののそれらの判別への影響は小さいことが確認できた。

次に $P = 25$ から $S = 10$ の変数選択のような、変数の候補数と選択上限数がある程度離れているケースを考える。以下は、図3.1と同様に横軸を MISDP から得た行列の要素、また縦軸を SDP から得た行列の要素をプロットした一例の図である。

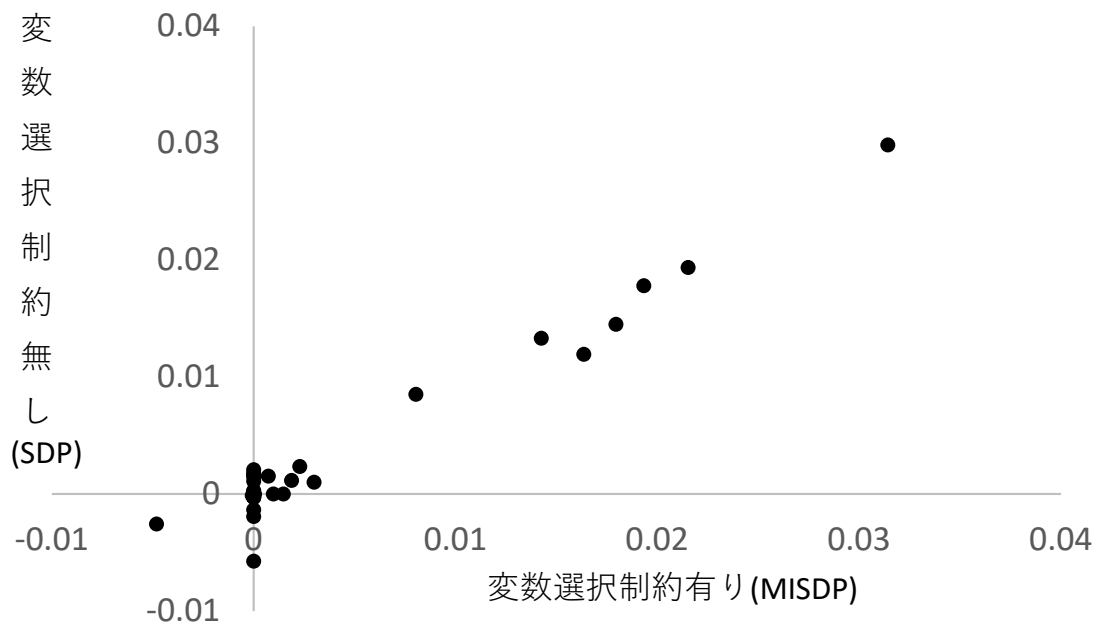


図3.2 変数の候補数と選択上限数がある程度離れているケースの解の違い

図3.2が図3.1と異なる点は、SDP から得た行列で 0 となる要素は、MISDP から得た行列でも同一の要素の水準はほぼ 0 となることである。つまり、図3.1であった枠線のケースがほとんど見当たらなくなっている。これは選択する個数を制限するほど、SDP で選択されない要素は、MISDP でも選択されなくなる傾向が強まることを意味する。言い換えると、SDP で選択される要素は、MISDP で選択される要素を包含する傾向があるといえる。

以上より SDP から得た行列は、変数選択を実行するための情報として活用できることが示唆された。具体的には、その行列の要素を見た際に、判別に与える寄与度が高い要素ほど、MISDP でも選択されやすいということである。加えて、変数の候補数と選択上限数に一定の差異がある場合、言い換えると変数選択の実行が特に求められる場面であるほど、SDP から得られる行列の非ゼロの要素は、MISDP から得られる行列の非ゼロの要素を包含する関係性がより顕在化するので、その有効性が期待できる。

この点を踏まえ次の小節では、半正定値制約と離散変数による変数選択制約の両方を組み込んだ MISDP を直接計算するのではなく、半正定値制約のみを組み込んだ SDP から解を得て、その次にその解を用いて判別において重要性の高い変数を優先して選択するような、逐次に問題を計算する方法を説明する。

3.2.3 ヒューリスティックアルゴリズム

二つ目のアルゴリズムは、3.2.1 節にある解法と比較して、より大きいデータサイズで問題 (3.12) の計算を終了するためのヒューリスティックアルゴリズムである。このアルゴリズムは ϵ -最適解を理論的には保証できないものの、近似精度の高い解を非常に速く得られることができる。特に切除平面アルゴリズムでは、解を得られないことができないサイズのデータやハイパーパラメータの水準でも短い時間で解を得られることができる点に大きなメリットがある。

ステップは 3 段階あり、まず最初に (3.12) から変数選択制約を除去したモデルを求解し、ランク P の半正定値行列を得る。次にそれを用いて (3.12) から変数選択制約および半正定値制約に注目して制約式を書き直したモデルを計算することで、行列の半正定性を維持しながら S 個の説明変数を選択する。最後に再び (3.12) から変数選択制約を除去したモデルを計算することで、ランク S の半正定値行列を得る。

1 段階目は $P \times P$ の半正定値行列を得る。そのため、問題 (3.12) から変数選択制約を除去した次のモデルを計算する。

$$\left| \begin{array}{ll} \min_{\mathbf{B}} & g(\mathbf{B}) \\ \text{s.t.} & \mathbf{B} \succeq 0. \end{array} \right. \quad (3.16)$$

ここで解 $\hat{\mathbf{B}} \in \mathbf{R}^{P \times P}$ を保存する。(3.16) は (3.12) の緩和問題であるため、 $\hat{\mathbf{B}}$ は (3.12) の解の情報を有していることが期待される。また変数選択制約がないため、このモデルは短い時間で解くことができる。

2 段階目は、 $\hat{\mathbf{B}}$ を用いながら、指定した個数以下の説明変数を選択する。ここでは得られる解は $\hat{\mathbf{B}}$ に比例するようにする。それにより行列の半正定性を大きく崩さないま

ま、0となる要素を増やすことができる。これを実行するために、入力値 $\hat{\mathbf{B}}, \delta > 0$, 変数 $\lambda \in [1, \delta)$, $\mathbf{u}, \mathbf{t} \in \mathbf{R}^{P \times P}$ を導入し Al-Khayyal (1990); McCormick (1976) を参照することで、次の制約式を定式化する。

$$\left\{ \begin{array}{l} u_{p_1, p_2} \geq 0, u_{p_1, p_2} \geq z_{p_1} + z_{p_2} - 1, \\ z_{p_1} \geq u_{p_1, p_2}, z_{p_2} \geq u_{p_1, p_2}, \\ t_{p_1, p_2} \geq 0, t_{p_1, p_2} \geq \delta u_{p_1, p_2} + \lambda - \delta, \\ \delta u_{p_1, p_2} \geq t_{p_1, p_2}, \lambda \geq t_{p_1, p_2}, \\ B_{p_1, p_2} = \hat{B}_{p_1, p_2} t_{p_1, p_2}, u_{p_1, p_2} \in \{0, 1\}, \\ \delta \geq \lambda \geq 0, \end{array} \right. \quad (p_1, p_2 \in \mathcal{P}). \quad (3.17)$$

(3.17) 式の効果の説明する。まず得られた解を $\mathbf{u}^*, \mathbf{z}^*$ と置くと、制約条件の1段目および2段目から次を得られる。

$$u_{p_1, p_2}^* = \begin{cases} 1, & \text{if } (z_{p_1}^*, z_{p_2}^*) = (1, 1), \\ 0, & \text{otherwise,} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (3.18)$$

このため、 $u_{p_1, p_2}, (p_1, p_2 \in \mathcal{P})$ は $u_{p_1, p_2} \in \{0, 1\}^{P \times P}$ と離散変数になる。

次に得られた解を \mathbf{t}^*, λ^* と置くと、制約条件の3段目および4段目から次を得られる。

$$t_{p_1, p_2}^* = \begin{cases} \lambda^*, & \text{if } u_{p_1, p_2}^* = 1, \\ 0, & \text{otherwise,} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (3.19)$$

よって (3.18)、(3.19) より得られた解 $\mathbf{B}^*, \mathbf{z}^*, \lambda^*$ を用いて、次を得られる。

$$B_{p_1, p_2}^* = \begin{cases} \hat{B}_{p_1, p_2} \lambda^*, & \text{if } (z_{p_1}^*, z_{p_2}^*) = (1, 1), \\ 0, & \text{otherwise,} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (3.20)$$

これにより、選択した変数同士に相当する要素に限り、半正定値行列 $\hat{\mathbf{B}}$ から一律で同じ比率 (λ^*) を掛けた値を取れる。そのためには以下の MILP を計算すればよい。

$$\left| \begin{array}{l} \min_{\mathbf{B}, \mathbf{z}, \mathbf{u}, \mathbf{t}, \lambda} g(\mathbf{B}), \\ \text{s.t.} \quad (3.17), \\ \sum_{p \in \mathcal{P}} z_p \leq S. \end{array} \right. \quad (3.21)$$

ここでランク P となる入力値 $\hat{\mathbf{B}}$ が半正定値行列であれば、 $P > S$ とランクが小さくなるので、ランク S となる解 \mathbf{B}^* は、半正定値行列を維持できていることが期待できる。またここで求めた解 \mathbf{z}^* を用いて選択した説明変数の情報を保存する集合 \mathcal{J} を次のとおり定義する。

$$\mathcal{J} := \{p | z_p^* = 1 \ (p \in \mathcal{P})\}. \quad (3.22)$$

最後は選択した説明変数を用いて、改めて半正定値行列を求める。 \mathcal{I} を用いて、説明変数の個数を P から S に減らした入力データ \mathbf{x}_i を次のとおり再定義する。

$$x_{i,s \in \{1, \dots, S\}} \leftarrow x_{i,p \in \mathcal{I}} \quad (i \in \mathcal{I}). \quad (3.23)$$

そのうえで変数 $\mathbf{B} \in \mathbf{R}^{S \times S}$ もランクを減らして問題 (3.16) と同様の次の SDP から解を得る。なお一段階目とモデル自体に違いはなく、違いは行列のランクおよびベクトルの長さのみである。

$$\left| \begin{array}{l} \min \quad g(\mathbf{B}) \\ \mathbf{B} \\ \text{s.t.} \quad \mathbf{B} \succeq 0. \end{array} \right. \quad (3.24)$$

以上よりヒューリスティックアルゴリズムを次のとおりまとめる。

ヒューリスティックアルゴリズム (HA)

Step1 問題 (3.16) を計算し、解 $\hat{\mathbf{B}} \in \mathbf{R}^{P \times P}$ を得る。

Step2 問題 (3.21) を計算し、得られた解 \mathbf{z}^* より (3.23) を用いて、入力データ $\mathbf{x}_i \in \mathbf{R}^S, (i \in \mathcal{I})$ を再定義する。

Step3 問題 (3.24) を計算し、解 $\hat{\mathbf{B}} \in \mathbf{R}^{S \times S}$ を得る。終了。

なお、当該アルゴリズムのように事前に入力データとなる行列を計算し、その後その情報と離散変数を導入することで変数選択を実施している先行研究は他にも Tamura et al. (2017, 2019)^{*4}がある。

^{*4} 双方とも最小二乗法をベースとして、多重共線性を抑制することを目的に、最適化モデルを構築している。Tamura et al. (2017) は、事前に係数ベクトルを求めるための、説明変数間の相関係数行列を作成し、その後、その情報を用いて、その行列の最大固有値と最小固有値の比率を一定値以下に抑えるよう、変数を選択する MISDP を構築している。また Tamura et al. (2019) は、事前に多重共線性を測る指標の一つの Variance inflation factor を作成し、その後、その情報を用いて、その値を一定値以下に抑えるよう変数を選択するモデルを構築している。特に Tamura et al. (2017) は、Tamura et al. (2019) と違って行列の要素を一定比率で抽出する必要があることから、問題 (3.21) と構造が近い。

第 4 章

MISDP に拡張した CVaR 最小化モデルの評価

4.1 序論

Vapnik (1995) が提案した SVM は、機械学習の分野で最も知られている判別モデルの一つである。SVM が多くの分野で広く使われている理由は二つある。一つは凸二次計画問題に定式化できることから容易に解を得ることができること、もう一つは、説明変数間の多重共線性の影響を強く受ける LR と違って、多重共線性の影響をあまり受けないことである。また、Okada and Konno (2009); Tanaka and Nakagawa (2014) は、LR と比較実証したところ、SVM のほうがより高い汎化性能を得られること、および、より少ない説明変数を選択したことを報告している。更に、Cristianini and Shawe-Taylor (2000); Takeda and Sugiyama (2008); Takeda (2009); Gotoh and Takeda (2011) は、汎化誤差を抑制する特徴を持つことも報告している。

SVM のような判別モデルの汎化性能を、より高めるためのアイデアも提案されている。そこで、2つのアイデアを SVM に導入する。一つ目は、判別面を楕円形に拡張して、説明変数の線形性だけでなく、非線形性や変数間の相関構造も表現可能とすることである。Konno and Wu (2002); Konno et al. (2003b); Okada and Konno (2009); Konno and Saito (2013) は、楕円形の適用により、判別モデルの汎化性能がより高くなることを報告している。また、二次関数に半正定値制約条件を追加することで対応可能である。二つ目は、判別モデルに最適な変数選択を実行するために、0 か 1 となる離散変数を導入することである。Konno and Yamamoto (2008); Saito and Konno (2009) は、判別モデルで変数選択を実行するため、離散変数を用いて、MILP と再定式化したモデルを提案し

た。変数は多いほどより高い汎化性能が得られるとは限らず、例えば、Okada and Konno (2009) は楕円形を適用した判別モデルを実証したところ、Shirakawa (2001) が提案した 18 個の説明変数よりも、独自に選択したより少ない 10 個の変数のほうが、汎化性能が向上したことを報告している。そのため、説明変数はより必要な変数に限定して使用したほうが、汎化性能を向上させる可能性が高いと考えられる。しかし、これら二つを SVM に取り込んだ汎化性能の実証報告は無い。そこで、SVM に双方の制約を組み込んだモデルを提案することで、更なる汎化性能の向上を狙う。

一方で、その提案モデルの大きな問題は計算時間が非常に長いことである。それは、双方の制約を組み込んだ SVM は MISDP となるためである。この MISDP は理論的には Konno et al. (2003b) が提案した切除平面アルゴリズムを用いることで ϵ -最適解を得られるが、このアルゴリズムの構造上、計算時間を要する MILP の計算を相当な回数で反復しなければならない。そのため、実証上、解を得られるケースはデータケースが非常に小さいケースに限定される。この点に関して、Kobayashi and Takano (2020) は、小さいサイズの MISDP は解を得られることを報告しているが、本章が対象とするデータサイズはそこでの報告対象より大きい。そこで、これを解消するため、短い時間で適切な解を得るためのヒューリスティックアルゴリズムを提案し、その有効性を報告するとともに、得られた解による汎化性能を検証する。

ただし、Vapnik (1995) が提案した C-SVM ではなく、それを Gotoh and Takeda (2005) が再定式化した CVaR 最小化モデルを検証対象とする。C-SVM のハイパーパラメータの水準に妥当な意味はないが、CVaR 最小化モデルのそれは統計学での信頼区間に相当するので、分析や使用の際に扱いやすいのが理由である。

本章の目的は、SVM を再定式化した CVaR 最小化モデルを、楕円形と変数選択を同時実行できるモデルへ拡張することで汎化性能を向上することである。問題点は、半正定値制約と離散変数を追加したことで MISDP となることから計算時間が急激に増加することである。そこで、この点を解消するためのヒューリスティックアルゴリズムを提案し、一定時間内でも解を得ること、および得られた解による汎化性能の優位性を検証する。

この章の構成は次の通りである。4.2 節では、SVM から CVaR 最小化モデルへの定式化および半正定値制約と離散変数を追加した MISDP への定式化を説明する。4.3 節では、本章で用いる MISDP となる SVM から解を得るためのアルゴリズムの具体的な実装について説明する。4.4 節では、企業の格付と財務データを用いた判別問題を対象にした場合の計算時間と汎化性能の結果を展開する。最後の 4.5 節では、結論と今後の課題をまとめる。

4.2 モデルの定式化

4.2.1 SVM ベースの CVaR 最小化モデル

本章では 2.2 節で紹介した、SVM を再定式化した CVaR 最小化モデル (2.12) を用いる。

$$\left| \begin{array}{ll} \min_{\mathbf{b}, b_0, \alpha, \xi_i \ (i \in \mathcal{I})} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i \geq -f(\mathbf{b}, b_0 | \mathbf{x}_i)_i - \alpha \ (i \in \mathcal{I}), \\ & \xi_i \geq 0 \ (i \in \mathcal{I}), \\ & \|\mathbf{b}\|_2^2 = 1. \end{array} \right. \quad (4.1)$$

このモデルから容易に解を得るため、Gotoh and Takeda (2005); Gotoh et al. (2014) を参照にして、二点改良を行う。

一つ目は、ノルム制約を見直す。問題 (4.1) 内の 2 ノルムだと、最適解を得るために専用のアルゴリズム (Gotoh and Takeda, 2005) が必要で、計算の手間が嵩む。そこで 2 ノルムの $\|\mathbf{b}\|_2^2 = 1$ ではなく、1 ノルムの $\|\mathbf{b}\|_1 = 1$ に書き直す。Gotoh et al. (2014) が報告したとおり、仮に係数が正となる制約条件下ならば、1 ノルムとすることで、線形計画問題となり、2 ノルムよりも容易に解を得られる。また Mangasarian (1999) は、ノルム制約に関して、必ずしも 2 ノルムとする必要はないとしている。

二つ目は、 β の範囲を見直す。問題 (4.1) では $\beta \in [1 - 2 \min\{m_+, m_-\}, 1)$ となり、入力データのサンプル数に依存して β の下限が変化する。そのため、 $\beta \in [0, 1)$ と書き直して、下限を 0 に固定して、 β の変化に伴う計算時間や汎化性能への影響分析を容易にする。なお、 β は $\xi_i \ (i \in \mathcal{I})$ の分布における信頼区間に該当するパラメータなので、統計学の観点からも水準の設定が行いやすくなる。これら二つの改良を施すことで問題 (4.1) は次のとおり書き直せる。

具体的には、まず、制約式で $\|\mathbf{b}\|_2^2 = 1$ から、 $\|\mathbf{b}\|_1 = 1$ とノルム制約を書き換える。

$$\left| \begin{array}{ll} \min_{\mathbf{b}, b_0, \alpha, \xi_i \ (i \in \mathcal{I})} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i \geq F(\mathbf{b}, b_0 | \mathbf{x}_i)_i - \alpha \ (i \in \mathcal{I}), \\ & \xi_i \geq 0 \ (i \in \mathcal{I}), \\ & \|\mathbf{b}\|_1 = 1, \end{array} \right. \quad (4.2)$$

次に、 β の下限を 0 にするため、 $F(\mathbf{b}, b_0 | \mathbf{x}_i)_i$ において \hat{m}_i で調整している。

$$F(\mathbf{b}, b_0 | \mathbf{x}_i)_i := -y_i(\mathbf{b}^T \mathbf{x}_i - b_0)\hat{m}_i, \hat{m}_i = \begin{cases} \frac{m}{2m_+} & \text{if } i \in \mathcal{I}^+, \\ \frac{m}{2m_-} & \text{else if } i \in \mathcal{I}^-, \end{cases} (i \in \mathcal{I}). \quad (4.3)$$

また、Gotoh et al. (2014) は (4.2) を用いて企業の格付判別に適用している。

4.2.2 楕円形の導入

この小節では、分類器を線形から楕円形に拡張する。その準備として Okada and Konno (2009) に沿って、信用スコアを線形から二次関数に拡張する。ここで二次係数行列 $\mathbf{B} \in \mathbf{R}^{P \times P}$ とベクトル $\mathbf{X}_i, \bar{\mathbf{B}} \in \mathbf{R}^{P+P^2}$ を次の通りに定義する。

$$\begin{aligned} \mathbf{X}_i^T &:= (x_{i,1}, x_{i,2}, \dots, x_{i,P}, x_{i,1}^2, x_{i,1}x_{i,2}, \dots, x_{i,P}x_{i,P-1}, x_{i,P}^2), \\ \bar{\mathbf{B}}^T &:= (b_1, b_2, \dots, b_P, B_{1,1}, B_{1,2}, \dots, B_{P,P-1}, B_{P,P}). \end{aligned} \quad (4.4)$$

ここで (4.3) にある信用スコアを、次のとおりに再定式化する。

$$\begin{aligned} F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i &= -y_i(\bar{\mathbf{B}}^T \mathbf{X}_i - b_0)\hat{m}_i, \\ &= -y_i(\mathbf{x}_i^T \mathbf{B} \mathbf{x}_i + \mathbf{b}^T \mathbf{x}_i - b_0)\hat{m}_i \quad (i \in \mathcal{I}). \end{aligned} \quad (4.5)$$

(4.5) の形状を楕円形に限定するため半正定値制約を追加する。なお、半正定値制約は $\mathbf{B} \succeq \mathbf{O}$ と表記する。

ここで問題 (4.2) を基に、次の SDP を定式化する。

$$\left| \begin{array}{ll} \min_{\mathbf{B}, \mathbf{b}, b_0, \alpha, \xi_i \ (i \in \mathcal{I})} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha \quad (i \in \mathcal{I}), \\ & \xi_i \geq 0 \quad (i \in \mathcal{I}), \\ & \|\bar{\mathbf{B}}\|_1 = 1, \\ & \mathbf{B}^T = \mathbf{B}, \\ & \mathbf{B} \succeq \mathbf{O}. \end{array} \right. \quad (4.6)$$

ここからはノルム制約の取り扱いについて記載する。問題 (4.6) は非凸な制約であるノルム制約 $\|\bar{\mathbf{B}}\|_1 = 1$ がある。そこで、連続変数 $\phi, \psi \in \mathbf{R}^P$ 、 $\Phi, \Psi \in \mathbf{R}^{P \times P}$ と離散変数 $\zeta \in \{0, 1\}^P$ 、 $\mathbf{Z} \in \{0, 1\}^{P \times P}$ を導入することで離散変数を含めた線形式へと定式化する。

まず \mathbf{b} に関しては、次のとおりに定式化する。

$$\left\{ \begin{array}{l} \phi_{p_1} - \psi_{p_1} = b_{p_1}, \\ 0 \leq \phi_{p_1} \leq \zeta_{p_1}, \\ 0 \leq \psi_{p_1} \leq (1 - \zeta_{p_1}), \\ \zeta_{p_1} \in \{0, 1\}, \end{array} \right. \quad (p_1 \in \mathcal{P}). \quad (4.7)$$

ここで得られた解を ϕ^* 、 ψ^* 、 ζ^* とすると、次の関係性がある。

$$\begin{cases} 1 \geq \phi_{p_1}^* \geq 0, \psi_{p_1}^* = 0 & \text{if } \zeta_{p_1}^* = 1, \\ 1 \geq \psi_{p_1}^* \geq 0, \phi_{p_1}^* = 0 & \text{otherwise } \zeta_{p_1}^* = 0, \end{cases} \quad (p_1 \in \mathcal{P}). \quad (4.8)$$

得られた解を $b_{p_1}^*$ とすると、 $b_{p_1}^*$ が正であれば、 $\phi_{p_1}^*$ が 0 以上に、 $b_{p_1}^*$ が負であれば、 $\psi_{p_1}^*$ が 0 以上の値となり、相補性条件を満たすことがわかる。

同様に \mathbf{B} に関しては、次のとおりに定式化する。

$$\begin{cases} \Phi_{p_1, p_2} - \Psi_{p_1, p_2} = B_{p_1, p_2}, \\ 0 \leq \Phi_{p_1, p_2} \leq Z_{p_1, p_2}, \\ 0 \leq \Psi_{p_1, p_2} \leq (1 - Z_{p_1, p_2}), \\ Z_{p_1, p_2} \in \{0, 1\}, \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (4.9)$$

ここで得られた解を Φ^* 、 Ψ^* 、 \mathbf{Z}^* とすると、次の関係性がある。

$$\begin{cases} 1 \geq \Phi_{p_1, p_2}^* \geq 0, \Psi_{p_1, p_2}^* = 0 & \text{if } Z_{p_1, p_2}^* = 1, \\ 1 \geq \Psi_{p_1, p_2}^* \geq 0, \Phi_{p_1, p_2}^* = 0 & \text{otherwise } Z_{p_1, p_2}^* = 0, \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (4.10)$$

得られた解を B_{p_1, p_2}^* とすると、 B_{p_1, p_2}^* が正であれば、 Φ_{p_1, p_2}^* が 0 以上に、 B_{p_1, p_2}^* が負であれば、 Ψ_{p_1, p_2}^* が 0 以上の値となり、相補性条件を満たすことがわかる。

最後に各係数の絶対値の和を 1 とする制約式を、次のとおり定式化する。

$$\sum_{p \in \mathcal{P}} (\phi_p + \psi_p) + \sum_{p_1 \in \mathcal{P}} \sum_{p_2 \in \mathcal{P}} (\Phi_{p_1, p_2} + \Psi_{p_1, p_2}) = 1. \quad (4.11)$$

以上からノルム制約 ($\|\bar{\mathbf{B}}\|_1 = 1$) は (4.7)、(4.9)、(4.11) により次のとおり再定式化できる。

$$\left| \begin{array}{ll} \min_{\mathbf{B}, \mathbf{b}, b_0, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \Phi, \Psi, \mathbf{Z}} & \alpha + \frac{1}{(1 - \beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha \quad (i \in \mathcal{I}), \\ & \xi_i \geq 0 \quad (i \in \mathcal{I}), \\ & \mathbf{B}^\top = \mathbf{B}, \\ & \mathbf{B} \succeq \mathbf{O}, \\ & (4.7), (4.9), (4.11), \dots (\text{ノルム制約}) \end{array} \right. \quad (4.12)$$

4.2.3 変数選択制約の導入

この小節では 3.3 節で述べた変数選択制約を、問題 (4.12) に導入する。またノルム制約により、 \mathbf{b} および \mathbf{B} の係数の上限および下限は 1 もしくは -1 と設定できる。なお、以下

に 3.1.2 小節の制約式 (3.4)、(3.6)、(3.3) を再掲している。

$$-z_{p_1} \leq b_{p_1} \leq z_{p_1} \quad (p_1 \in \mathcal{P}) \quad (4.13a)$$

$$-z_{p_1} \leq B_{p_1,p_2} \leq z_{p_1}, -z_{p_2} \leq B_{p_1,p_2} \leq z_{p_2} \quad (p_1, p_2 \in \mathcal{P}). \quad (4.13b)$$

$$\sum_{p_1 \in \mathcal{P}} z_{p_1} \leq S, \quad (4.13c)$$

ここで S ($S < P$) はユーザーが定義する入力パラメータである。

以上より本章で対象とする MISDP は次のとおりである。

$$\left| \begin{array}{ll} \min_{\mathbf{B}, \mathbf{b}, b_0, \mathbf{z}, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \Phi, \Psi, \mathbf{Z}} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha \quad (i \in \mathcal{I}), \\ & \xi_i \geq 0 \quad (i \in \mathcal{I}), \\ & \mathbf{B}^T = \mathbf{B}, \\ & \mathbf{B} \succeq \mathbf{O}, \\ & (4.7), (4.9), (4.11), \dots (\text{ノルム制約}) \\ & (4.13a), (4.13b), (4.13c) \dots (\text{変数選択制約}) \end{array} \right. \quad (\text{ES})$$

4.3 MISDP の解法

この節では MISDP となる問題 (ES) から解を得るため、3 章で述べた既存アルゴリズムとヒューリスティックアルゴリズムの実装を説明する。

4.3.1 切除平面アルゴリズムへの実装

切除平面アルゴリズムの手順を以下のとおりにまとめる。

Algorithm 1: 切除平面アルゴリズム (CPA)

Step1 $k \leftarrow 0$. $\epsilon > 0$. $\mathcal{F}^k := \{\mathbf{B} | \mathbf{B}^T = \mathbf{B}, B_{p,p} \geq 0, p \in \mathcal{P}\}$.

Step2 次の問題を計算する。

$$\begin{array}{ll}
 \min_{\mathbf{B}, \mathbf{b}, b_0, \mathbf{z}, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \Phi, \Psi, \mathbf{Z}} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\
 \text{s.t.} & \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha (i \in \mathcal{I}), \\
 & \xi_i \geq 0 (i \in \mathcal{I}), \\
 & \mathbf{B}^T = \mathbf{B}, \\
 & \mathbf{B} \in \mathcal{F}^k, \\
 & (4.7), (4.9), (4.11), \dots (\text{ノルム制約}) \\
 & (4.13a), (4.13b), (4.13c), \dots (\text{変数選択制約})
 \end{array} \tag{4.14}$$

ここで得た解を \mathbf{B}^* とする。

Step3 \mathbf{B}^* の最小固有値が $-\epsilon$ を超えると終了。そうでなければ、 $-\epsilon$ 以下の固有値を n_k とし、それらの固有値に相当する固有ベクトル \mathbf{d}_l を計算したうえで、 $\mathcal{F}^{k+1} := \mathcal{F}^k \cap \{\mathbf{B} | \mathbf{d}_l^T \mathbf{B} \mathbf{d}_l \geq 0, l = 1, 2, \dots, n_k\}$ として \mathcal{F}^{k+1} を再定義する。

Step4 $k \leftarrow k + 1$ 。Step2 に戻る。

4.3.2 ヒューリスティックアルゴリズムの実装

1 段階目ではランク P となる半正定値行列を得るために問題 (ES) から変数選択制約 (4.13a), (4.13b), (4.13c) を除去した次の問題を CPA を用いて解を得る。

$$\begin{array}{ll}
 \min_{\mathbf{B}, \mathbf{b}, b_0, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \Phi, \Psi, \mathbf{Z}} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\
 \text{s.t.} & \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha (i \in \mathcal{I}), \\
 & \xi_i \geq 0 (i \in \mathcal{I}), \\
 & \mathbf{B}^T = \mathbf{B}, \\
 & \mathbf{B} \succeq \mathbf{O}, \dots (\text{半正定値制約}) \\
 & (4.7), (4.9), (4.11), \dots (\text{ノルム制約})
 \end{array} \tag{4.15}$$

ここで得られた解を $\hat{\mathbf{B}} \in \mathbf{R}^{P \times P}$ とする。

2 段階目では、半正定値制約を維持しながら変数選択を行うため (3.17) の制約式を追加する。また (3.17) は \mathbf{B} に関する変数選択も実行しているので、(4.13b) と重複する。そのため (3.17) の導入に伴い、(4.13b) は不要になる。

またノルム制約に関しても、 \hat{B}_{p_1, p_2} は事前に各要素の符号が判明できているので、(4.9) にある離散変数を含んだ制約条件は不要になる。よって (4.11) は次のとおりに再定式化

できる。

$$\sum_{p_1 \in \mathcal{P}} (\phi_{p_1} + \psi_{p_1}) + \sum_{p_1 \in \mathcal{P}} \sum_{p_2 \in \mathcal{P}} (|B_{p_1, p_2}|) = 1. \quad (4.16)$$

以上から次の問題が定式化できる。

$$\begin{array}{ll} \min_{\mathbf{B}, \mathbf{b}, b_0, \mathbf{z}, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \mathbf{u}, \mathbf{t}, \lambda} & \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\ \text{s.t.} & \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha \quad (i \in \mathcal{I}), \\ & \xi_i \geq 0 \quad (i \in \mathcal{I}), \\ & \mathbf{B}^T = \mathbf{B}, \\ & (3.17), \\ & (4.7), (4.16), (4.11), \dots (\text{ノルム制約}) \\ & (4.13a), (4.13c). \dots (\text{変数選択制約}) \end{array} \quad (4.17)$$

この問題を計算することで、 S 個の変数を選択できる。

3 段階目は、変数選択後に入力データ \mathbf{x}_i のサイズを P から S に減らした上で、再び (4.15) を CPA を用いて解を得る。その際は、事前に変数のサイズおよび行列のランクを P から S に落とす ($\mathbf{b}, \phi, \psi \in \mathbf{R}^S$, $\mathbf{B}, \Phi, \Psi \in \mathbf{R}^{S \times S}$, $\zeta \in \{0, 1\}^S$, $\mathbf{Z} \in \{0, 1\}^{S \times S}$, $\bar{\mathbf{B}} \in \mathbf{R}^{S+S^2}$)。更に $p \in \{1, 2, \dots, P\}$ から $s \in \{1, 2, \dots, S\}$ と変更した上で解く。これにより、再び問題 (4.15) を計算することで、ランク S の半正定値行列を得る。

ヒューリスティックアルゴリズムの手順を以下のとおりにまとめる。

Algorithm 2: ヒューリスティックアルゴリズム (HA)

Step1 問題 (4.15) を計算し、解 $\hat{\mathbf{B}}$ を得る。

Step2 問題 (4.17) を計算し、得られた解 \mathbf{z}^* より (3.23) を用いて、入力データ $\mathbf{x}_i \in \mathbf{R}^S$, ($i \in \mathcal{I}$) を再定義する。

Step3 問題 (4.15) を計算し、解を得る。終了。

4.4 計算機実験

4.4.1 セットアップ

使用するのは日本の東証一部に上場している企業のデータである。被説明変数 (y_i ($i \in \mathcal{I}$)) は格付投資情報センター (R&I) が 2011 および 2012 年度に取得した格付データを使用する。また説明変数 (\mathbf{x}_i ($i \in \mathcal{I}$)) の候補数 (P) は 25 と 51 の財務データを使用する (詳細は付録 A の表 A.1 を参照。)。

まず表4.1は被説明変数に用いる高格付と低格付のグループに属する企業数をトレーニングデータとテストデータ別に載せている。なお、“高格付”はAAAからA-を、“低格付”はBBB+からCCCを指す。

表4.1 各データセットの高格付と低格付の企業数

Year	データセット	高格付 (m_+)	低格付 (m_-)
2011	トレーニング	285	120
2012	テスト	269	124

また、説明変数に関しては Yeo Johnson 変換 (Yeo and Johnson, 2000) を用いてスケールリングしている*1。なお、ここでこの変換処理自体は R パッケージの “bestNormalize” *2で実施した。

CPA および HA の計算用に作成したプログラムは Microsoft Windows Visual Studio 2015 上で C++ でコーディングし、使用した計算環境は Core i9-9980, 32.0 GB, 2.40 GHz である。また全ての最適化問題は CPLEX12.90 で計算した。

CPA を解く際は $\varepsilon = 10^{-3}$ とし、ヒューリスティックアルゴリズム内の問題 (4.17) を解く際は $\delta = 3$ とした。計算時間は 720 分 (12 時間) 以内である。

4.4.2 比較用の判別面モデル

信用スコアが線形判別面である問題 (LS) は次のとおり。なおこのモデルは (ES) において二次係数を 0 とし、半正定値制約を外している。

$$\begin{array}{l}
 \min_{\mathbf{B}, \mathbf{b}, b_0, \mathbf{z}, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \Phi, \Psi, \mathbf{Z}} \quad \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\
 \text{s.t.} \quad \xi_i \geq F(\bar{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha \quad (i \in \mathcal{I}), \\
 \xi_i \geq 0 \quad (i \in \mathcal{I}), \\
 \mathbf{B} = \mathbf{O}, \\
 (4.7), (4.9), (4.11), \dots (\text{ノルム制約}) \\
 (4.13a), (4.13c), \dots (\text{変数選択制約})
 \end{array} \tag{LS}$$

また二次判別面の問題 (QS) は次のとおり。なお問題 (ES) から、半正定値制約のみを

*1 Yeo Johnson 変換を用いることで、元の財務データの符号の正負に関わらず、各変数を平均 0 および標準偏差 1 とする正規分布に従うよう均一化できる。

*2 <https://cran.r-project.org/web/packages/bestNormalize/bestNormalize.pdf>

外している。

$$\begin{array}{l}
 \min_{\mathbf{B}, \mathbf{b}, b_0, \mathbf{z}, \alpha, \xi_i (i \in \mathcal{I}), \phi, \psi, \zeta, \Phi, \Psi, \mathbf{Z}} \quad \alpha + \frac{1}{(1-\beta)m} \sum_{i=1}^m \xi_i \\
 \text{s.t.} \quad \xi_i \geq F(\overline{\mathbf{B}}, b_0 | \mathbf{X}_i)_i - \alpha \quad (i \in \mathcal{I}), \\
 \xi_i \geq 0 \quad (i \in \mathcal{I}), \\
 \mathbf{B}^T = \mathbf{B}, \\
 (4.7), (4.9), (4.11), \dots (\text{ノルム制約}) \\
 (4.13a), (4.13b), (4.13c), \dots (\text{変数選択制約})
 \end{array} \tag{QS}$$

4.4.3 ヒューリスティックアルゴリズムの評価

表4.2は計算時間および汎化性能に大きな影響を与えると考えられるパラメータの検証範囲をまとめている。

表4.2 パラメータの検証範囲

パラメータ	値
β	0.10, 0.20, ..., 0.70, 0.80
S	10, 15, 20
P	25, 51

最も影響のあるパラメータは β である。その変化は判別面を表現する係数 (\mathbf{b}, \mathbf{B}) の値に影響を与え、それに伴い判別性能が決まる。またその他に影響のあるパラメータは、指定する説明変数の個数の上限である S と、説明変数の候補数である P である。なぜならば、これらの組み合わせにより離散変数 (\mathbf{z}) の組み合わせ数が決まるからである。

これらのパラメータを変化させてトレーニングデータを用いて問題 (ES) を CPA と HA で比較検証を行った。

表4.3は CPA と HA を表4.2にあるパラメータの値での計算時間を示している。なおこの表の“—”は720分以内で計算終了できなかったことを意味する。

この表から、HA は CPA より非常に短い時間で解を得ていることがわかる。また β が大きいほど、計算時間が大きくなることがわかる。特に、 $\beta \geq 0.5$ はその傾向が顕著となった。 $P = 25$ に限って見ると、 $\beta = 0.6$ の双方とも720分以内で解けているケースを見ると、最大で数百倍程度も計算時間が高速化できていることがわかる。加えて $P = 51$ は、HA なら720分以内で解を得ることができるケースは CPA のそれよりはるかに多いことがわかった。

表4.3 アルゴリズム別の計算時間 (単位: 分)

P	アルゴリズム	$S \setminus \beta$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
25	CPA	10	0.0	0.0	0.1	0.1	8.2	473.7	—	—
		15	0.0	0.0	0.1	0.2	2.0	191.7	—	—
		20	0.0	0.0	0.1	0.1	0.8	20.7	283.9	372.6
	HA	10	0.0	0.0	0.0	0.0	0.2	0.9	2.1	—
		15	0.0	0.0	0.0	0.0	0.2	0.9	1.9	3.1
		20	0.0	0.0	0.0	0.0	0.3	1.1	1.9	3.3
51	CPA	10	0.1	0.3	0.6	15.3	—	—	—	—
		15	0.1	0.2	0.4	3.1	—	—	—	—
		20	0.1	0.2	0.4	1.9	—	—	—	—
	HA	10	0.1	0.2	0.3	0.8	1.5	8.1	46.3	—
		15	0.1	0.2	0.5	0.8	3.5	9.5	63.2	—
		20	0.1	0.2	0.5	0.9	8.2	13.0	67.6	379.1

β の変化に伴い計算時間が変化する点について、更に分析した。表4.4が示しているのは、HA で得られた解を用いて計算した、二次の係数行列の要素の絶対値の和 ($\sum_{p_1 \in \mathcal{P}} \sum_{p_2 \in \mathcal{P}} |B_{p_1, p_2}|$) である。この表より β が大きい場合、特に $\beta \geq 0.5$ の際は、線形係数

表4.4 $P = 51$ での二次係数の各絶対値の和

$S \setminus \beta$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
10	0.00	0.02	0.05	0.08	0.17	0.26	0.57	—
15	0.00	0.02	0.05	0.08	0.19	0.31	0.60	—
20	0.00	0.02	0.05	0.08	0.18	0.35	0.60	0.73

\mathbf{b} より二次係数 \mathbf{B} の影響が大きいことがわかる。また表4.3の結果も踏まえると、 β が大きいほど、解の疎性が弱くなり、計算時間が増加していると考えられる。

表4.5が示しているのは、各アルゴリズムで得られた解による相対誤差である。この表から HA と CPA の相対誤差は、最大で数 % 以内と小さいので、近似精度は充分と判断できる。

よって、HA は既存アルゴリズムの CPA より、高速かつ、良質な解を得られる手法であ

表4.5 各アルゴリズムによる解の相対誤差 (単位: %)

P	$S \setminus \beta$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
25	10	0.0	0.0	0.0	0.0	0.5	4.8	—	—
	15	0.0	0.0	0.0	0.0	0.0	1.3	—	—
	20	0.0	0.0	0.0	0.0	0.0	0.4	0.1	0.9
51	10	0.0	0.0	0.0	0.0	—	—	—	—
	15	0.0	0.0	0.0	0.0	—	—	—	—
	20	0.0	0.0	0.0	0.0	—	—	—	—

ることがわかる。

表4.6は判別に用いた判別面毎の計算時間を示している。なお (ES) に関しては HA を用いた結果を記載している。この表から (ES) と (QS) はパラメータの数は同一にも関わらず、(ES) は (QS) よりも相当に速く解を得られることがわかる。加えて更に線形 (LS) と比較しても計算時間に大きな乖離がないことが確認できる。そのため、時間内に質の良い解を得るという点でも、楕円形は有効的といえる。

表4.6 $P = 51, S = 10$ における判別面毎の計算時間 (単位: 分)

β	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
(ES)	0.1	0.2	0.3	0.8	1.5	8.1	46.3	—
(LS)	0.0	0.0	0.0	0.0	0.0	0.0	8.8	—
(QS)	0.1	0.1	0.2	0.4	1.7	276.5	—	—

4.4.4 汎化性能の評価

この小節では、 $P = 51$ に絞って、問題 (ES) で得られた解を用いると、汎化性能は優位になることを確認する。

表4.7は、(ES) から得られたトレーニングデータとテストデータおよびその差異の AR 値を S および β 別に、それぞれ“トレーニング”、“テスト”、“差異”で示している。なお、問題 (ES) は HA を用いて得た解で AR 値を計算している。

この表から、 $\beta = 0.5$ か $\beta = 0.6$ の AR 値が最も良い。また S の違いによる AR 値の変化は顕著にはなかった。この場合、変数の個数が少ないほうが望ましいことから、最も少

ない $S = 10$ が最適な設定と判断する。このため、以後の汎化性能の分析は $S = 10$ を軸

表4.7 $P = 51$ での (ES) による解の AR 値 (単位: %)

$S \setminus \beta$		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
10	“トレーニング”	72.6	81.0	85.4	86.4	88.8	89.8	92.9	—
	“テスト”	70.2	76.1	79.4	80.1	83.4	83.7	62.8	—
	“差異”	2.4	4.9	6.0	6.3	5.4	6.1	30.1	—
15	“トレーニング”	72.6	81.0	85.4	86.5	89.0	90.6	94.2	—
	“テスト”	70.2	76.1	79.4	80.1	84.0	83.2	76.0	—
	“差異”	2.4	4.9	5.9	6.3	5.0	7.3	18.2	—
20	“トレーニング”	72.6	81.0	85.4	86.4	88.8	91.1	94.6	96.9
	“テスト”	70.2	76.1	79.4	80.2	83.8	83.3	75.9	56.3
	“差異”	2.4	4.9	5.9	6.2	5.0	7.7	18.6	40.6

に進める。

次に、楕円判別面 (ES) による汎化性能は、線形判別面 (LS) および二次判別面 (QS) による汎化性能より高いことを確認する。表4.8が示しているのは、 $S = 10$ のときの各判別面のトレーニングデータとテストデータの AR 値、およびそれらの差異である。また、(LS) と (QS) の列にある “[]” は、720 分時点での暫定解を、(ES) の列の “—” は 720 分時点では半正定値制約 ($\mathbf{B} \succeq \mathbf{O}$) を満たした解を得られなかったことを意味する。そして下線部はテストデータにおいて、最も AR 値が高いケースを示す。それ以外の、ラベルの記載は表4.7と同様である。

この表から多くのケースで (QS) のトレーニングデータによる AR 値は、(LS) と (ES) のそれより高いことがわかる。これは (QS) の係数の自由度が最も高いことから、自然な結果と考える。

一方で、テストデータの AR 値を見ると、 $\beta = 0.5$ および 0.6 で、(ES) は、(LS) と (QS) よりも高かった。また各判別面の最も汎化性能が高いケースを比較すると、楕円形が最も良い結果となった ($83.7((ES)) > 81.8(LS) > 73.1(QS)$)。これが意味することは楕円形は汎化性能の改善に大きく貢献しているということである。更に (QS) の結果と比較すると、トレーニングとテストデータのそれぞれの AR 値の差が、(ES) より (QS) のほうが非常に小さくなっていることから、オーバーフィッティングを抑制できていることがわかる。このため楕円判別面が最も適合しているといえる。

加えて、各判別面とハイパーパラメータである β の変化を注目した。 $S = 10$ の場合、

係数の自由度は、(ES) が 29、(LS) が 10、(QS) が 65 となる。そして汎化性能が最も高かったケースを見ると、自由度が最も大きい (QS) では $\beta = 0.3$ であった。一方でその半分以下の自由度である (ES) は $\beta = 0.6$ 、(LS) は $\beta = 0.5$ だった。このため、判別面自体の自由度の高さと β の水準は逆比例の関係にあることがわかる。そのため、ハイパーパラメータの設定は判別面毎にチューニングが必要であることが確認できた。なお、(ES) のほうが (LS) よりも自由度が高い一方で β も高いが、この点は自由度やハイパーパラメータの水準の影響だけでなく、楕円形が線形よりも判別面として適している影響も含んでいると考えられる。

また、 β を十分に大きくした場合、いずれの判別面もオーバーフィットを起こす傾向があることも確認した。これは β を大きくすると解の疎性が弱まるためと考えられる。なお、 $\beta = 0.8$ では、いずれの判別面も計算負荷が高すぎて時間内に計算が終了しなかったものの、そのケースの解はオーバーフィットを起こす可能性が高いので、解を得る必要性は無い。

表4.8 $S = 10$ での判別面毎の AR 値 (単位: %)

β		0.1	0.2	0.3	0.4	0.5	<u>0.6</u>	0.7	0.8
(ES)	“トレーニング”	72.6	81.0	85.4	86.4	88.8	89.8	92.9	—
	“テスト”	70.2	76.1	79.4	80.1	83.4	<u>83.7</u>	62.8	—
	“差異”	2.4	4.9	6.0	6.3	5.4	6.1	30.1	—
(LS)	“トレーニング”	72.6	79.8	83.1	83.6	85.4	88.3	89.5	[86.7]
	“テスト”	70.2	76.5	80.2	80.8	<u>81.8</u>	79.7	81.5	[21.0]
	“差異”	2.4	3.3	2.8	2.7	3.6	8.6	7.9	[65.7]
(QS)	“トレーニング”	72.5	81.3	86.1	88.7	90.1	91.5	[95.0]	[97.8]
	“テスト”	69.6	67.6	<u>73.1</u>	48.9	66.0	62.2	[65.3]	[51.0]
	“差異”	3.0	13.7	13.0	39.8	24.1	29.4	[29.7]	[46.8]

4.5 結論

本章では判別問題に対して楕円形と変数選択を追加して拡張した SVM を提案した。このモデルは期待通り高い汎化性能を得られたが、その一方で計算時間が非常に長い。そこで、既存アルゴリズムより短時間で適切な解を得るためのヒューリスティックアルゴリズムを用いた。

本章の計算結果により、提案したヒューリスティックアルゴリズムは、既存アルゴリズムである切除平面アルゴリズムよりも最大で数百倍速く解を得られることを実証した。これまでの先行研究では SDP に定式化する事例 (Okada and Konno, 2009) はあるものの、更に変数選択制約を含めた MISDP へと拡張した例は見受けられなかった。これは計算時間が長すぎることから、解を得ることが実証的に非常に困難になるためだが、このヒューリスティックアルゴリズムは、この MISDP の解を短時間で得ることに成功した。更に、楕円判別面の汎化性能は、線形や二次判別面よりも高いことを実証した。その理由としては、オーバーフィッティングを回避できたことで、トレーニングデータによる AR 値とテストデータの AR 値の差異を抑えられたことが挙げられる。

最後に今後の課題として、本章では Gotoh and Takeda (2005); Gotoh et al. (2014) が構築した SVM を再定式化した CVaR 最小化モデルを用いたが、他のモデルに楕円形と変数選択の同時実行を行うことが挙げられる。判別問題に関しては、多くのモデルが研究されており、最近でも Norton and Uryasev (2019) が提案した AUC を直接最大化することに寄与するモデルが提案されている。このモデルも含めて、様々な判別モデルを MISDP に拡張することで、高い汎化性能が得られることを検証したい。

第 5 章

MISDP に拡張した bAUC 最大化モデルの評価

5.1 序論

判別モデルの性能を測る有名な指標に AUC がある。この指標が高いほど、モデルの判別性能が高いと評価できる。しかし、AUC は非凸で離散型の関数であるため、容易に最大化ができない。この課題に対応するため、Norton and Uryasev (2019) は buffered AUC および Generalized buffered AUC (bAUC) を提示した。そして bAUC の最大化は、AUC の最大化の代用になると主張すると共に、bAUC 最大化モデルは凸型の最小化問題に定式化できることから容易に解を得られることを報告している。更に、Norton and Uryasev (2019) は、誤判別する確率を表現した分布のテイルリスクを軽減することで、過度な誤判別を起こす可能性を抑制する特徴があることも報告している。なお、bAUC は、1 から bPOE (buffered Probability of Exceedance) を差し引いた指標であり、その bPOE を格付付与やポートフォリオ最適化に適用した先行研究もある (Norton et al., 2021; Pertaia et al., 2021)。

bAUC 最大化モデルは、2 つの利点がある。1 つ目は、bAUC 最大化モデルのハイパーパラメータが、誤判別を表す分布上のある一定値以上の期待値^{*1}と統計学で解釈できるため、ハイパーパラメータの水準の設定が扱いやすくなる点である。2 つ目は、Norton et al. (2017) が説明したとおり、bAUC 最大化モデルの仕組みが Vapnik (1995) が提示した SVM と近いことから、多重共線性を引き起こしにくい点である。その結果、bAUC 最

^{*1} Super-quantile や超分位点とも呼ばれている。金融分野では CVaR、Expected shortfall と呼ばれることもある。

大化モデルにより得られた係数の安定性は高いので、他の判別モデルよりもより良い分類が期待できる*²。よって、この bAUC 最大化モデルに前章でも取り組んだ分類器*³を線形から楕円形へ拡張すること、および、変数選択の導入を実施して、汎化性能を向上を目的としたモデルを提案する。一方で、それらを組み合わせた提案モデルは、前章で実施した SVM よりも、より多くの線形の制約式を含んだ MISDP として定式化されることから、時間内に解を得ることがより難しくなる課題がある。

前章で取り組んだ SVM を基にした中規模の MISDP でさえ、既存アルゴリズムの切除平面アルゴリズムを用いた際、時間内に解を求めることは非常に困難であった。それに対処するために提案したヒューリスティックアルゴリズムを適用することで、大幅な計算時間削減を可能にしたものの、説明変数が 50 個程度あるケースではハイパーパラメータの水準次第では、計算時間内に解を得られないケースも存在した。このため、提案されたヒューリスティックアルゴリズムをそのまま bAUC 最大化モデルに適用しても、妥当な時間内に解が得られる範囲は限られることが予想できる。そこで、この課題を解決するために、不要な離散変数を予め削減して、より高速化できる手法を提示する。

その上で、Tanaka and Nakagawa (2014) が行った実際の企業の格付データを用いた信用リスクによる判別（倒産か否かの判別など）への応用と同様の手順で、提案モデルが判別問題において高い汎化性能を得られることを実証する。Wei et al. (2018) は、信用リスクを例とした判別問題において、線形分類器の bAUC 最大化モデルが、LR と同じ水準の汎化性能を算出したことを報告している。一方で、LR 以外に、多くのハイパーパラメータを有することで高い柔軟性を持つ複数の機械学習 (ML) モデルと比較した例は、見かけない。そこで、bAUC 最大化モデルの汎化性能が、ニューラルネットワーク (NN) およびランダムフォレスト (RF) を含めた標準的な既存の ML モデルよりも優れていることを実証する。また、そこで bAUC 最大化モデルの汎化性能が上回る要因として、誤判別を行った際に確率的には小さいものの影響が大きい誤り (テイルリスク) を抑制できる特徴を持つためであることも検証する。

なお、本章における bAUC 最大化モデルに対して実施した新規事項と、先行研究の報告結果は表5.1のとおりである。

本章は以下のとおりに構成する。5.2 節では AUC と bAUC を概説し、bAUC 最大化モデルを定式化する。5.3 節では bAUC 最大化モデルを楕円分類器および変数選択を同時に評価できる MISDP に拡張する。5.4 節では MISDP を解くためのアルゴリズムの実装お

*² 多重共線性は標準的な統計モデルである LR などに発生しうる課題である

*³ 前章では「判別面」と呼称していた。Norton and Uryasev (2019) は classifier/分類器と表記していたことから先行研究に沿った表現で記載している。

表5.1 bAUC 最大化モデルに関する先行研究と本章の貢献箇所

	bAUC 最大化モデル		判別性能実験	
	分類器	変数選択	用いたデータ	比較対象モデル
本章	楕円形	採用	トレーニング 及びテストデータ	LR, SVM, NN, RF
Norton and Uryasev (2019)	線形	未採用	トレーニングデータ のみ	LR および SVM.
Wei et al. (2018)			トレーニング 及びテストデータ	LR

よびヒューリスティックアルゴリズムの高速化について述べる。5.5 節ではヒューリスティックアルゴリズムの評価および楕円分類期の汎化性能の結果を展開する。なお楕円分類期の汎化性能に関しては、変数選択を考慮したケースは他の標準的な分類器と比較し、変数選択を考慮していないケースではそれに加えて標準的な機械学習と比較した。5.6 節では結論と今後の課題を述べる。

5.2 bAUC 最大化モデル

5.2.1 AUC

この章では先行研究 (Mafusalov and Uryasev, 2018; Norton and Uryasev, 2019; Rockafellar and Royset, 2010) を参照に、AUC および bAUC を説明する。

まず定式の準備として各記号を定義する。 $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m)\}$ を所与とするデータセットとし、 $\mathbf{x}_i \in \mathbf{R}^P$ はサンプル i に対する説明変数であり、 $y_i \in \{\pm 1\}$ はそのラベルとする。更に $i \in \mathcal{I} := \{1, \dots, m\}$ かつ $p \in \mathcal{P} := \{1, \dots, P\}$ とする。各ラベルが属するグループの集合を $\mathcal{I}_+ = \{i \mid y_i = 1, i \in \mathcal{I}\}$, $\mathcal{I}_- = \{i \mid y_i = -1, i \in \mathcal{I}\}$ とし、それぞれの個数を $m_+ = |\mathcal{I}_+|$, $m_- = |\mathcal{I}_-|$ とする。なおここでは $\mathcal{I} = \mathcal{I}_+ \cup \mathcal{I}_-$, $m = m_+ + m_-$ となる。

係数 $\mathbf{w} \in \mathbf{R}^P$ を導入して、次の線形分類器を定式化する。

$$R(\mathbf{w}|\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i \quad (i \in \mathcal{I}). \quad (5.1)$$

この時 AUC を $R(\mathbf{w}|\mathbf{x}_{i_+})$ ($i_+ \in \mathcal{I}_+$) が $R(\mathbf{w}|\mathbf{x}_{i_-})$ ($i_- \in \mathcal{I}_-$) を上回る確率としてランキングエラー関数 $\xi(\mathbf{w}|\mathbf{x}_{i_+}, \mathbf{x}_{i_-}) = -\{R(\mathbf{w}|\mathbf{x}_{i_+}) - R(\mathbf{w}|\mathbf{x}_{i_-})\}$ ($i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-$) を用

いて以下のように定義する。なお、(2.23) と式は同様で、パラメータの記号のみ異なっている。

$$\text{AUC} = 1 - \frac{1}{m_+ m_-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \mathbb{1}_{\xi(\mathbf{w}|\mathbf{x}_{i_+}, \mathbf{x}_{i_-})}, \quad (5.2)$$

ここで関数 $\mathbb{1}_{\bullet}$ は (2.22) にある指示関数と同様である。

AUC は $R(\mathbf{w}|\mathbf{x}_{i_-}) - R(\mathbf{w}|\mathbf{x}_{i_+})$ の計算において、 $m_+ \times m_-$ の回数の計算を行う。表記を簡単にするためランキングエラー関数の全組み合わせを次のとおりに表記する。

$$H(\mathbf{w}) := \xi(\mathbf{w}|\mathbf{x}_{i_+}, \mathbf{x}_{i_-}) \quad (i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-). \quad (5.3)$$

これより (5.2) 式の AUC は以下のようにも記述できる。

$$\text{AUC} = 1 - P(H(\mathbf{w}) \geq 0). \quad (5.4)$$

5.2.2 bAUC

次に AUC を拡張した bAUC を説明する。

ある分位点 $Q_\gamma(H(\mathbf{w}))$ を^{*4}確率水準 $\gamma \in \{0, 1\}$ を導入して次のとおり定義する。

$$Q_\gamma(H(\mathbf{w})) := \inf\{a \in \mathbf{R} | P(H(\mathbf{w}) > a) \leq 1 - \gamma\}. \quad (5.5)$$

更にある超分位点 $q_\gamma(H(\mathbf{w}))$ を次のとおり定義する。

$$\begin{aligned} q_\gamma(H(\mathbf{w})) &:= E[H(\mathbf{w}) | H(\mathbf{w}) > Q_\gamma(H(\mathbf{w}))], \\ &= \min_{\alpha} \left\{ \alpha + \frac{E[H(\mathbf{w}) - \alpha]^+}{1 - \gamma} \right\}, \end{aligned} \quad (5.6)$$

ここで $[\cdot]^+ := \max\{\cdot, 0\}$ を指す。

(5.6) 内の最適解 α^* を導入することで、 $\alpha^* = Q_\gamma(H(\mathbf{w}))$ とする。更に $q_\gamma(H(\mathbf{w}))$ は Rockafellar and Uryasev (2000) が提示している CVaR に相当する。

この $q_\gamma(H(\mathbf{w}))$ の逆関数に相当する確率 $p_{\tilde{q}}(H(\mathbf{w}))$ を次のとおりに定義する。

$$p_{\tilde{q}}(H(\mathbf{w})) := \max\{1 - \gamma \mid q_\gamma(H(\mathbf{w})) \geq \tilde{q}\} = \min\{1 - \gamma \mid q_\gamma(H(\mathbf{w})) \leq \tilde{q}\}, \quad (5.7)$$

ここで $\tilde{q} \leq 0$ はユーザーパラメータを表す。Norton and Uryasev (2019) は、この $p_{\tilde{q}}(H(\mathbf{w}))$ を用いることで bAUC を以下の通りに定義している。

$$\text{bAUC} := 1 - p_{\tilde{q}}(H(\mathbf{w})). \quad (5.8)$$

^{*4} 金融分野では VaR と呼ばれている。

Rockafellar and Royset (2010) はハイパーパラメータ (\tilde{q}) を超える確率 $P(H(\mathbf{w}) \geq \tilde{q})$ を Probability of Exceedance (POE) と、確率 $p_{\tilde{q}}(H(\mathbf{w}))$ を Buffered Probability of Exceed (bPOE) と定義している。ここで、Mafusalov and Uryasev (2018) は、(5.5)、(5.6)、(5.7) および $\tilde{q} \leq 0$ より、次の式を示している。

$$P(H(\mathbf{w}) \geq 0) \leq P(H(\mathbf{w}) \geq \tilde{q}) \leq p_{\tilde{q}}(H(\mathbf{w})). \quad (5.9)$$

表5.1が示しているのは、式 (5.9) にある POE と bPOE の関係性である。更にNorton

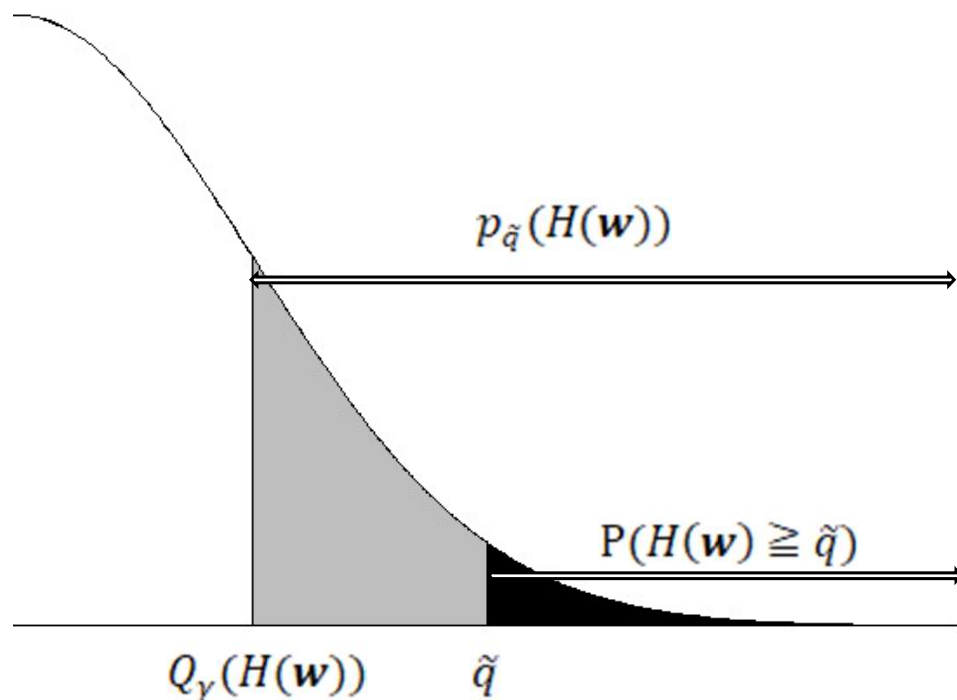


図5.1 POE ($P(H(\mathbf{w}) \geq \tilde{q})$) と bPOE ($p_{\tilde{q}}(H(\mathbf{w}))$)

and Uryasev (2019) は (5.9) から次式を示している。

$$1 - P(H(\mathbf{w}) \geq 0) \geq 1 - p_{\tilde{q}}(H(\mathbf{w})) \Leftrightarrow \text{AUC} \geq \text{bAUC} \quad (5.10)$$

(5.10) を考慮すると、AUC の最大化の代用に bAUC の最大化を用いることができる。更に、 \tilde{q} により誤判別に関する分布 ($H(\mathbf{w})$) のテイルリスクを管理することができるので、bAUC 最大化モデルは、誤判別の大きいケースを抑制することが期待できる^{*5}。

^{*5} 誤判別した場合、 $R(\mathbf{w}|\mathbf{x}_{i-})$ と $R(\mathbf{w}|\mathbf{x}_{i+})$ の乖離が非常に大きいシナリオが起こりうる。これが分布のテイル部分に位置されるが、このような稀に起こるシナリオの発生確率を抑制する効果を持つ。

5.2.3 bAUC の最大化

本節では bAUC 最大化を考える。これは (5.8) 式より $p_{\tilde{q}}(H(\mathbf{w}))$ の最小化問題と等価なので、以下のように定式化できる。

$$\min_{\mathbf{w}} p_{\tilde{q}}(H(\mathbf{w})) = \left| \begin{array}{l} \min_{\gamma, \alpha, \mathbf{w}} 1 - \gamma \\ \text{s.t.} \quad \alpha + \frac{E[H(\mathbf{w}) - \alpha]^+}{1 - \gamma} \leq \tilde{q}. \end{array} \right. \quad (5.11)$$

ここで (5.11) の制約式を考慮することで次式が得られる。

$$1 - \gamma \geq \frac{E[H(\mathbf{w}) - \alpha]^+}{\tilde{q} - \alpha}. \quad (5.12)$$

さらに表記の単純化のため、(5.11) および (5.12) から次の式が導ける。

$$\min_{\alpha, \mathbf{w}} \frac{E[H(\mathbf{w}) - \alpha]^+}{\tilde{q} - \alpha}, \quad (5.13)$$

ここで図5.1より、 $\tilde{q} > \alpha$ である。

しかし、Norton et al. (2017) が報告しているとおり、(5.13) では一意な解を求められない。この課題を解決するために Norton and Uryasev (2019) が提示したとおり、ベクトル \mathbf{w} を任意のノルムで固定する。具体的には次のとおり書き直す。

$$\left| \begin{array}{l} \min_{\alpha, \mathbf{w}} \frac{E[H(\mathbf{w}) - \alpha]^+}{\tilde{q} - \alpha} \\ \text{s.t.} \quad \|\mathbf{w}\| = 1, \end{array} \right. \quad (5.14)$$

ここで $\|\cdot\|$ は任意のノルムである。

$\|\cdot\|$ に関してはノルムの選択は任意に記述可能であるため、計算上取り扱いやすい 1 ノルムを選択する (Mangasarian, 1999; Gotoh et al., 2014; Norton et al., 2017)。それにより、問題 (5.14) は次のように定式化できる。

$$\left| \begin{array}{l} \min_{\alpha, \mathbf{w}} \frac{E[H(\mathbf{w}) - \alpha]^+}{\tilde{q} - \alpha} \\ \text{s.t.} \quad \|\mathbf{w}\|_1 = 1, \end{array} \right. \quad (5.15)$$

ここで $\|\mathbf{w}\|_1 := \sum_{p \in \mathcal{P}} |w_p|$ を意味する。

5.2.4 定式化

Norton et al. (2017) に沿って分数計画問題である問題 (5.15) から容易に解を得るための手順を説明する。まずベクトル $\mathbf{b} \in \mathbf{R}^P$ を次のとおりとする。

$$\mathbf{b}^T = \frac{\mathbf{w}^T}{\tilde{q} - \alpha}. \quad (5.16)$$

ここで (5.16) から次の式を得る。

$$\|\mathbf{b}\|_1^1 = \frac{\|\mathbf{w}\|_1^1}{|\tilde{q} - \alpha|}. \quad (5.17)$$

(5.17) および (5.15) の $\|\mathbf{w}\|_1^1 = 1$ を用いて次の式を得る。

$$\frac{\|\mathbf{w}\|_1^1}{|\tilde{q} - \alpha|} = \frac{1}{\tilde{q} - \alpha} \quad (5.18)$$

更に式 (5.17) と (5.18) を用いて、次の式を得る。

$$\frac{\tilde{q}}{\tilde{q} - \alpha} = (\|\mathbf{b}\|_1^1)\tilde{q}. \quad (5.19)$$

そして (5.16) と (5.19) を用いて、(5.15) を次のとおりに再定式化する。

$$\begin{aligned} \min_{\alpha, \mathbf{w}} \left\{ \frac{E[H(\mathbf{w}) - \tilde{q} + \tilde{q} - \alpha]^+}{\tilde{q} - \alpha} \right\} &= \min_{\alpha, \mathbf{w}} \left\{ E \left[\frac{H(\mathbf{w})}{\tilde{q} - \alpha} - \frac{\tilde{q}}{\tilde{q} - \alpha} + \frac{\tilde{q} - \alpha}{\tilde{q} - \alpha} \right]^+ \right\}, \\ &= \min_{\mathbf{b}} \left\{ E[H(\mathbf{b}) - (\|\mathbf{b}\|_1^1)\tilde{q} + 1]^+ \right\}. \end{aligned} \quad (5.20)$$

(5.20) の内、 $(\|\mathbf{b}\|_1^1)$ および $E[\cdot]^+$ は、変数 $\beta \in \mathbf{R}$ および非負変数 $\phi, \psi \in \mathbf{R}^P$ および ξ_{i_+, i_-} , ($i_+ \in \mathcal{I}_+$, $i_- \in \mathcal{I}_-$) を導入することで以下の線形計画問題へ定式化できる (Konno and Yamamoto, 2005; Rockafellar and Uryasev, 2000)。

$$\left| \begin{array}{l} \min_{\xi, \mathbf{b}, \beta, \phi, \psi} \quad \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\ \text{s.t.} \quad \xi_{i_+, i_-} \geq H(\mathbf{b}) - \beta \tilde{q} + 1 \quad (i_+ \in \mathcal{I}^+, i_- \in \mathcal{I}^-), \\ \xi_{i_+, i_-} \geq 0 \quad (i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-), \\ \sum_{p_1 \in \mathcal{P}} (\phi_{p_1} + \psi_{p_1}) = \beta, \\ \phi_{p_1} - \psi_{p_1} = b_{p_1}, \phi_{p_1} \geq 0, \psi_{p_1} \geq 0 \quad (p_1 \in \mathcal{P}). \end{array} \right. \quad (\text{L})$$

加えて, (L) の最適解 ϕ^*, ψ^* に対し, 以下の関係性がある。

$$\begin{cases} \phi_{p_1}^* \geq 0, & \text{if } \psi_{p_1}^* = 0, \\ \psi_{p_1}^* \geq 0, & \text{otherwise } \phi_{p_1}^* = 0, \end{cases} \quad (p_1 \in \mathcal{P}). \quad (5.21)$$

更に (L) の最適解 $\xi_{i_+, i_-}^*, (i_+ \in \mathcal{I}^+, i_- \in \mathcal{I}^-), \mathbf{b}^*, \beta^*$ に対し, 以下の関係性がある。

$$\xi_{i_+, i_-}^* = \begin{cases} H(\mathbf{b}^*) - \beta^* \bar{q} + 1, & \text{if } H(\mathbf{b}^*) - \beta^* \bar{q} + 1 \geq 0, \\ 0, & \text{otherwise,} \end{cases} \quad (i_+ \in \mathcal{I}^+, i_- \in \mathcal{I}^-). \quad (5.22)$$

5.3 MISDP への拡張

判別問題の中で, 倒産と非倒産のような判別分析は信用リスクの分野では, Altman (1968) を筆頭に 50 年以上扱われている。その中で, 様々な先行研究で, 判別モデルとその拡張方法が提案されている。本節ではその中でも bAUC 最大化モデルの汎化性能を向上するために 2 点の改良を実施する。

5.3.1 楕円分類器

LR や SVM のような判別モデルにおいて, その分類器を楕円形に拡張すると汎化性能が向上することが報告されている (Konno et al., 2003b; Okada and Konno, 2009)。そのため, bAUC 最大化モデルに楕円分類器を適用する。

まずは, 楕円形にするため, 二次係数行列 $\mathbf{B} \in \mathbf{R}^{P \times P}$ を置く。Okada and Konno (2009) に沿い, まずは線形分類器 ($R(\mathbf{b}|\mathbf{x}_i), i \in \mathcal{I}$) から二次分類器 ($r(\mathbf{b}, \mathbf{B}|\mathbf{x}_i), i \in \mathcal{I}$) に拡張する。

そこでベクトル $\boldsymbol{\omega}, \mathbf{X}_i, \in \mathbf{R}^{P+P^2}$ を, 次のとおり定義する。

$$\boldsymbol{\omega}^T := (b_1, b_2, \dots, b_P, B_{1,1}, B_{1,2}, \dots, B_{P,P-1}, B_{P,P}). \quad (5.23)$$

$$\mathbf{X}_i^T := (x_{i,1}, x_{i,2}, \dots, x_{i,P}, x_{i,1}^2, x_{i,1}x_{i,2}, \dots, x_{i,P}x_{i,P-1}, x_{i,P}^2). \quad (5.24)$$

(5.23)、(5.24) を用いて線形分類器 (5.1) およびランキングエラー関数 (5.3) を次のと

おりに再定式化する。

$$\begin{aligned}
r(\mathbf{b}, \mathbf{B} | \mathbf{x}_i) &= \mathbf{b}^T \mathbf{x}_i + \mathbf{x}_i^T \mathbf{B} \mathbf{x}_i, \\
&= \boldsymbol{\varpi}^T \mathbf{X}_i \\
&= R(\boldsymbol{\varpi} | \mathbf{X}_i) (i \in \mathcal{I}).
\end{aligned} \tag{5.25}$$

$$\begin{aligned}
h(\mathbf{b}, \mathbf{B}) &= -\{r(\mathbf{b}, \mathbf{B} | \mathbf{x}_{i_+}) - r(\mathbf{b}, \mathbf{B} | \mathbf{x}_{i_-})\}, \\
&= -\{R(\boldsymbol{\varpi} | \mathbf{X}_{i_+}) - R(\boldsymbol{\varpi} | \mathbf{X}_{i_-})\}, \\
&= H(\boldsymbol{\varpi}), (i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-).
\end{aligned} \tag{5.26}$$

線形分類器 (L) 内の制約式に関しては、非負変数行列 $\Phi, \Psi, \in \mathbf{R}^{P \times P}$ の導入および $H(\mathbf{b})$ から $h(\mathbf{b}, \mathbf{B})$ への置き換えにより、下線部分を次のとおりに再定式化する。

$$\left\{ \begin{array}{l} \xi_{i_+, i_-} \geq \underline{h(\mathbf{b}, \mathbf{B})} - \beta \tilde{q} + 1 \quad (i_+ \in \mathcal{I}^+, i_- \in \mathcal{I}^-), \\ \xi_{i_+, i_-} \geq 0 \quad (i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-), \\ \sum_{p_1 \in \mathcal{P}} (\phi_{p_1} + \psi_{p_1}) + \sum_{p_1 \in \mathcal{P}} \sum_{p_2 \in \mathcal{P}} (\Phi_{p_1, p_2} + \Psi_{p_1, p_2}) = \beta, \\ \phi_{p_1} - \psi_{p_1} = b_{p_1}, \phi_{p_1} \geq 0, \psi_{p_1} \geq 0 \quad (p_1 \in \mathcal{P}), \\ \underline{\Phi_{p_1, p_2} - \Psi_{p_1, p_2} = B_{p_1, p_2}}, \underline{\Phi_{p_1, p_2} \geq 0}, \underline{\Psi_{p_1, p_2} \geq 0} \quad (p_1, p_2 \in \mathcal{P}). \end{array} \right. \tag{5.27}$$

(5.27) を適用することで、問題 (L) から二次分類器の問題 (Q) と拡張できる。

$$\left| \begin{array}{l} \min_{\xi, \mathbf{B}, \mathbf{b}, \beta, \Phi, \Psi, \phi, \psi} \quad \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\ \text{s.t.} \quad \mathbf{B}^T = \mathbf{B}, \\ \quad \quad \quad (5.27). \end{array} \right. \tag{Q}$$

更に、半正定値制約 $\mathbf{B} \succeq \mathbf{O}$ を (Q) に適用することで楕円分類器となる bAUC 最大化モデルを次の通り定式化する。

$$\left| \begin{array}{l} \min_{\xi, \mathbf{B}, \mathbf{b}, \beta, \Phi, \Psi, \phi, \psi} \quad \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\ \text{s.t.} \quad \mathbf{B} \succeq \mathbf{O}, \mathbf{B}^T = \mathbf{B}, \\ \quad \quad \quad (5.27). \end{array} \right. \tag{E}$$

5.3.2 離散変数を用いた変数選択

変数選択は汎化性能に影響を与える重要な問題であることが知られており、Konno and Yamamoto (2008) は整数変数を用いて変数の最適な組み合わせを選択する最適化モデルを提案している。それゆえ、その手順に沿って、bAUC 最大化モデルに変数選択制約を導入する。

まず離散変数 $z \in \{0, 1\}^P$ を導入することで、ユーザーパラメータ $S (< P)$ で選択する個数を指定する制約式を以下のとおり定式化する。なお 3.1.2 小節の制約式 (3.3) を再掲している。

$$\sum_{p_1 \in \mathcal{P}} z_{p_1} \leq S. \quad (5.28)$$

次に離散変数 z と係数 \mathbf{b}, \mathbf{B} に関して以下の制約を追加する。なお 3.1.2 小節の制約式 (3.8) および (3.10) を再掲している。

$$z_{p_1} = 0 \Rightarrow b_{p_1} = 0 \quad (p_1, \in \mathcal{P}), \quad (5.29)$$

$$z_{p_1} = 0 \text{ もしくは } z_{p_2} = 0 \Rightarrow B_{p_1, p_2} = 0 \quad (p_1, p_2 \in \mathcal{P}). \quad (5.30)$$

これらの制約は CPLEX のような商用ソルバーで記載できる。

5.3.3 混合整数半正定値計画モデル

最後に、SDP となる問題 (E) に変数選択制約 (5.28)、(5.29)、(5.30) を追加することで、変数選択と判別を同時に実行できる次の MISDP が定式化できる。

$$\left| \begin{array}{l} \min_{\xi, \mathbf{B}, \mathbf{b}, \beta, \Phi, \Psi, \phi, \psi, z} \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\ \text{s.t.} \quad \mathbf{B} \succeq \mathbf{O}, \mathbf{B}^T = \mathbf{B}, \\ (5.27), \\ (5.28), (5.29), (5.30) \dots (\text{変数選択制約}) \end{array} \right. \quad (\mathbb{E})$$

先行研究では、計算コストが相当量ある MISDP を解くための様々なアルゴリズムが提案されている。そこで、次の節で二つのアルゴリズムの実装について説明する。

5.4 MISDP の解法

(E) は MISDP として定式化されるが、既存のアルゴリズムでは $P = 50$ かつ $S > 5$ のような大きなデータサイズで適切な解を得ることは容易ではない (Kobayashi and Takano, 2020)。加えて、前章の CVaR 最小化モデル (4.1) の制約式の本数が $m_+ + m_-$ と二つのグループの和に相当するのに対し、その bAUC 最大化モデルの制約式の本数は $m_+ \times m_-$ と二つのグループの積に相当することから、より問題の規模が大きくなるので、時間内の求解がより難しくなることが予想できる。この節では、求解するための MISDP (E) 既存およびヒューリスティックアルゴリズムの実装を説明する。

5.4.1 切除平面アルゴリズムへの実装

切除平面アルゴリズム (CPA) は半正定値制約 ($\mathbf{B} \succeq \mathbf{O}$) を満たす解を得ることができ
る。前章との同アルゴリズムと比較して注目すべき点は Step2 で、問題の制約式の本数が
相当に多くなったことから、時間内の計算終了がより難しいことが予想できる。

切除平面アルゴリズムの手順を以下のとおりにまとめる。

Algorithm 1: 切除平面アルゴリズム (CPA)

Step1 $k \leftarrow 0$. $\epsilon > 0$. $\mathcal{F}^k := \{\mathbf{B} | \mathbf{B}^T = \mathbf{B}, B_{p,p} \geq 0, p \in \mathcal{P}\}$.

Step2 次の問題を解く。

$$\left| \begin{array}{ll} \min_{\xi, \mathbf{B}, \mathbf{b}, \beta, \Phi, \Psi, \phi, \psi, \mathbf{z}} & \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\ \text{s.t.} & (5.27), \\ & (5.28), (5.29), (5.30), \dots (\text{変数選択制約}) \\ & \mathbf{B} \in \mathcal{F}^k. \end{array} \right. \quad (5.31)$$

ここでの解を \mathbf{B}^* とする。

Step3 \mathbf{B}^* の最小固有値が $-\epsilon$ を超えると終了。そうでなければ、最小固有値に相当する
基準化された固有ベクトル \mathbf{d} を計算したうえで、 $\mathcal{F}^{k+1} := \mathcal{F}^k \cap \{\mathbf{B} | \mathbf{d}^T \mathbf{B} \mathbf{d} \geq 0\}$
として \mathcal{F}^{k+1} を再定義する。 $k \leftarrow k + 1$. Step2 に戻る。

5.4.2 分枝切除アルゴリズムへの実装

Kobayashi and Takano (2020) が提示した二つ目の既存アルゴリズムである分枝切除
アルゴリズム (B&C) は、計算終了時に得られる解が ϵ -最適解となることが保証されて
いる点は CPA と同じである。違いは、計算の高速化を図るための工夫が施されている点
にある。このため、この章ではこの分枝切除アルゴリズムを比較対象の既存アルゴリズム
として用いる。

B&C は、CPA で実施した、繰り返し MILP を解き、かつ、 \mathcal{F}^k を更新する膨大な時間
の消費を緩和できるアルゴリズムである。そして `lazy constraint callback` の機能を
使い、反復計算ではなく、1 回計算開始すると、動的に \mathcal{F} を更新し続ける。

分枝切除アルゴリズムの手順を以下のとおりにまとめる。

Algorithm 2: 分枝切除アルゴリズム (B&C)

Step1 $\varepsilon > 0$. $\mathcal{F} \leftarrow \{\mathbf{B} \mid \mathbf{B}^T = \mathbf{B}, B_{p_1, p_1} \geq 0 (p_1 \in \mathcal{P})\}$.

Step2 次の問題の計算を開始する、もしくは継続する。

$$\left| \begin{array}{l} \min \\ \xi, \mathbf{B}, \mathbf{b}, \beta, \\ \Phi, \Psi, \phi, \psi, z \\ \text{s.t.} \end{array} \right. \quad \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \quad (5.32)$$

$$\begin{array}{l} (5.27), (5.28), (5.29), (5.30), \\ \mathbf{B} \in \mathcal{F}. \end{array}$$

Step3 いったん $\mathbf{B}^* \in \mathcal{F}$ を探索できたら、次のコールバックの手順を実行する:

Step3.1 \mathbf{B}^* の最小固有値が $-\varepsilon$ 以下ならば、次のとおり \mathcal{F} を更新する。

$$\mathcal{F} \leftarrow \mathcal{F} \cap \{\mathbf{B} \mid \mathbf{d}^T \mathbf{B} \mathbf{d} \geq 0\}, \quad (5.33)$$

ここで $\mathbf{d} \in \mathbf{R}^P$ は \mathbf{B}^* の最小固有値に相当する基準化された固有ベクトルである。

Step3.2 Step2 に戻る。

なおこの lazy constraint callback の機能は商業ソルバーである CPLEX や Gurobi で実装されている。

5.4.3 ヒューリスティックアルゴリズム

5.4.3.1 ヒューリスティックアルゴリズムのベース部分の実装

1 段階目は、変数選択制約の無い SDP(E) を解き、ランク P の半正定値行列となる解 $\hat{\mathbf{B}}$ を得る。

2 段階目は、 $\hat{\mathbf{B}}$ の情報を用いて、説明変数を選択する。 δ はユーザーが指定する正の数であり、 $\lambda \in [0, \delta)$, $\mathbf{u}, \mathbf{t} \in \mathbf{R}^{P \times P}$ は変数とする。ここで半正定値条件を維持するため、 \mathbf{B} の 0 となる成分を増やししながら、一定の比率 (λ) で非ゼロとなる成分の大きさを変化させるために、次の MILP となるモデルを定式化する。

$$\left| \begin{array}{l} \min \\ \xi, \mathbf{B}, \mathbf{b}, \beta, \\ \Phi, \Psi, \phi, \psi, z, \mathbf{u}, \mathbf{t}, \lambda \\ \text{s.t.} \end{array} \right. \quad \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \quad (5.34)$$

$$\begin{array}{l} \mathbf{B}^T = \mathbf{B}, \\ (5.27), (5.28), (5.29), (3.17). \end{array}$$

3 段階目は、変数選択後に入力データ \mathbf{x}_i のサイズを P から S に減らした上で、再び SDP(E) を解く。具体的には変数のサイズおよび行列のランクを P から S に落とし ($\mathbf{b}, \phi, \psi \in \mathbf{R}^S$, $\mathbf{B}, \Phi, \Psi \in \mathbf{R}^{S \times S}$, $\zeta \in \{0, 1\}^S$, $\mathbf{Z} \in \{0, 1\}^{S \times S}$, $\overline{\mathbf{B}} \in \mathbf{R}^{S+S^2}$)、更に $p \in \{1, 2, \dots, P\}$ から $s \in \{1, 2, \dots, S\}$ と変更した上で解く。これにより、ランク S の半正定値行列を得る。

ヒューリスティックアルゴリズムの手順を以下のとおりにまとめる:

アルゴリズム 3-1: ヒューリスティックアルゴリズム (HA)

Step1 問題 (E) を計算し、解 $\hat{\mathbf{B}}$ を得る。

Step2 問題 (5.34) を計算し、得られた解 \mathbf{z}^* より (3.23) を用いて、入力データ $\mathbf{x}_i \in \mathbf{R}^S$, ($i \in \mathcal{I}$) を再定義する。

Step3 問題 (E) を計算し、解を得る。終了。

5.4.3.2 高速化のための離散変数の削減部分の実装

HA を用いる際、Step 2 内の問題 (5.34) の計算時間が相当に長くなる。このため、制約式の本数が相応にある bAUC 最大化モデルに適用すると、時間内に良質な解を得ることが難しい可能性がある。そこで離散変数 \mathbf{u} の数を削減することで、高速化する手法を提示する。

ここでは、Step1 で得られる半正定値行列 $\hat{\mathbf{B}}$ の 0 の要素をできる限り増加して Step 2 の離散変数を減らすことで、問題の削減を行う。その際、以下のとおり、行列の半正定値性を維持しながら、影響の小さい要素の値を 0 に変換する。

$$\dot{B}_{p_1, p_2} := \begin{cases} 0, & \text{if } |\hat{B}_{p_1, p_2}| \leq \vartheta, \\ \hat{B}_{p_1, p_2}, & \text{otherwise,} \end{cases} \quad (p_1, p_2 \in \mathcal{P}), \quad (5.35)$$

ここで ϑ は $\hat{\mathbf{B}}$ の半正定値性を維持しつつ、要素の値をできるだけ 0 に変更できる値を指す。その ϑ は次の最適化問題の解として定義できる。

$$\left| \begin{array}{l} \max_{\vartheta} \quad \vartheta \\ \text{s.t.} \quad \dot{\mathbf{B}} \succeq \mathbf{O}, \\ \quad \quad (5.35). \end{array} \right. \quad (5.36)$$

この問題は一つの変数の最適化問題なので、最適解を得ること自体は徐々に ϑ を引き上げるだけで容易に探索可能である (詳細は付録 B を参照)。

その上で、次の制約式を定式化する。

$$\begin{cases} B_{p_1, p_2} = 0, & \text{if } |\hat{B}_{p_1, p_2}| \leq \vartheta, \\ (3.17), & \text{otherwise,} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (5.37)$$

ここで、HA の Step 2 における制約式 (3.17) を (5.37) に書き換えた次の問題を計算する。

$$\begin{array}{l} \min_{\xi, \mathbf{B}, \mathbf{b}, \beta, \Phi, \Psi, \phi, \psi, \mathbf{z}, \mathbf{u}, \mathbf{t}, \lambda} \quad \frac{1}{m^+ m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\ \text{s.t.} \quad \mathbf{B}^T = \mathbf{B}, \\ (5.27), (5.28), (5.29), (5.37). \end{array} \quad (5.38)$$

以上から、高速化を狙ったヒューリスティックアルゴリズムの手順を以下のとおりまとめる：

Algorithm 3-2: ヒューリスティックアルゴリズム + (HA+)

- Step 1 問題 (E) を計算し、解 $\hat{\mathbf{B}}$ を得る。
 Step 2 問題 (5.36) を計算することで解 ϑ を得る。
 Step 3 問題 (5.38) を計算し、得られた解 \mathbf{z}^* より (3.23) を用いて、入力データ $\mathbf{x}_i \in \mathbf{R}^S, (i \in \mathcal{I})$ を再定義する。
 Step 4 (E) から解を得る。終了。

このアプローチは、HA を用いても時間内に解を得ることが困難なデータサイズが大きい問題に対して、より効果的である。

なお、問題削減を用いた適用例には、Tamura et al. (2022) による問題の構造から予め理論的に不要な変数を外す手法や、田中・山本 (2022) による緩和問題から得た解から経験的に不要な変数を判定して外す手法などが提案され、いずれも計算の高速化に貢献していることを報告している。

5.5 計算機実験

本節では信用リスクに関する判別問題に対して実際の企業データを使い、提案した bAUC 最大化モデルの汎化性能は、従来の bAUC 最大化モデル、および、標準的な機械学習モデルに比べて高いことを実証する。

5.5.1 比較用の分類器モデル

線形分類器となる bAUC 最大化モデル (L) に変数選択制約 ((5.28),(5.29)) を追加したモデルは、次のとおり定式化できる。

$$\begin{array}{l}
 \min_{\boldsymbol{\xi}, \mathbf{b}, \mathbf{z}, \beta, \boldsymbol{\phi}, \boldsymbol{\psi}} \quad \frac{1}{m^+m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\
 \text{s.t.} \quad \xi_{i_+, i_-} \geq H(\mathbf{b}) - \beta \tilde{q} + 1 \quad (i_+ \in \mathcal{I}^+, i_- \in \mathcal{I}^-), \\
 \xi_{i_+, i_-} \geq 0 \quad (i_+ \in \mathcal{I}_+, i_- \in \mathcal{I}_-), \\
 \sum_{p_1 \in \mathcal{P}} (\phi_{p_1} + \psi_{p_1}) = \beta, \\
 \phi_{p_1} - \psi_{p_1} = b_{p_1}, \phi_{p_1} \geq 0, \psi_{p_1} \geq 0 \quad (p_1 \in \mathcal{P}), \\
 (5.28), (5.29) \dots (\text{変数選択制約})
 \end{array} \tag{L}$$

二次分類器となる bAUC 最大化モデル (Q) に変数選択制約 ((5.28), (5.29), (5.30)) を追加したモデルは、次のとおり定式化できる。

$$\begin{array}{l}
 \min_{\boldsymbol{\xi}, \mathbf{B}, \mathbf{b}, \mathbf{z}, \beta, \boldsymbol{\Phi}, \boldsymbol{\Psi}, \boldsymbol{\phi}, \boldsymbol{\psi}} \quad \frac{1}{m^+m^-} \sum_{i_+ \in \mathcal{I}_+} \sum_{i_- \in \mathcal{I}_-} \xi_{i_+, i_-} \\
 \text{s.t.} \quad \mathbf{B}^T = \mathbf{B}, \\
 (5.27), \\
 (5.28), (5.29), (5.30) \dots (\text{変数選択制約})
 \end{array} \tag{Q}$$

5.5.2 セットアップ

5.5.2.1 データセット

日本の東京証券取引所第一部に上場している現実の企業データを対象に、説明変数 (\mathbf{x}_i ($i \in \mathcal{I}$)) は、企業の財務項目の 50 種類 (詳細は付録 C の表 C.2 を参照) を、Laitinen and Laitinen (2000); Shirakawa (2001); Gotoh et al. (2014); Shaonan et al. (2015) を参照にして採用した。また目的変数 (y_i ($i \in \mathcal{I}$)) は日本格付研究所 (R&I) から 2011-2021 年度分を取得した。

目的変数について各格付グループに属する企業の内、取得できた数は表 5.2 の通り。各年でデータを得られた企業数は大体 300 個であった。なお、“高格付” は AAA から BBB+ を、“低格付” は BBB から CCC を指す*6。

*6 二つに判別する理想的な境界線は BBB- と BB+ のような投資適格と投資不適格の間であった。しかし投資不適格のデータが年次に 5 個程度しかなく、それではモデルの判別性能の検証には不足していると判断した。そのため、ある程度のサンプル数が確保できるよう、表 5.2 にある二つのグループで分けた。

表5.2 各年度のデータセットの格付別の企業数

Year	m	高格付 (m_+)	低格付 (m_-)
2011	345	277	68
2012	346	280	66
2013	326	273	53
2014	322	271	51
2015	323	277	46
2016	312	270	42
2017	307	267	40
2018	302	267	35
2019	307	273	34
2020	303	272	31
2021	265	237	28

また、前章同様に説明変数は Yeo Johnson 変換 (Yeo and Johnson, 2000) を用いて、R パッケージの “bestNormalize”^{*7} を用いて、スケーリングしている。

5.5.2.2 基本設定

表5.3は使用したソフトウェアとその環境である。

用いたパーソナルコンピューターは Core i9-9980、32.0 GB、2.40 GHz であり、最大の計算時間は 10800 秒 (3 時間) である。加えて (E) を HA の (5.34) および HA+ の (5.38) で計算する際に δ は 2 であり、B&C で解く際の (5.32) の中にある ε は 10^{-6} とした。

5.5.3 アルゴリズム性能評価

この小節では、HA と HA+ が既存解法の B&C よりも適切な解を得られることを示す。

^{*7} <https://cran.r-project.org/web/packages/bestNormalize/bestNormalize.pdf>

^{*8} SDP を求解可能なソルバーであることから使用した。また Gurobi では求解できない。

^{*9} <https://www.mosek.com/products/mosek/>

^{*10} 離散変数の有無を問わず LP を素早く求解可能なソルバーとして知られていることから使用した。

^{*11} <https://www.gurobi.com/documentation/9.0/refman/index.html>

表5.3 ソフトウェアと計算環境

	Models	Software	Environments
bAUC	(E)	Mosek 9.3.6 *8 *9	Julia 1.6.3
	(5.34),(5.38) (5.32) (L),(Q) (L),(Q)	Gurobi 9.0.3 *10 *11	
ML	NN SVM RF LR	Scikit-learn 0.23.2 (Pedregosa et al., 2011)	Spyder 4.1.5 and Python 3.8

5.5.3.1 詳細設定

(E) のハイパーパラメータで最も計算時間に影響するのは \tilde{q} であり、これで分類を決定する係数 (\mathbf{b}, \mathbf{B}) がより密に、言い換えると正負問わず 0 から離れた値になる度合いを決定している。具体的には、 \tilde{q} を 0 により近づけるほとに、連続変数 (\mathbf{b}, \mathbf{B}) が密になることが期待される。

さらに、説明変数の候補数 P と説明変数の選択数 S の大きさが、離散変数 (\mathbf{z}) の組み合わせ数を決定するため、二番目に影響があるパラメータである。

これらのパラメータの影響を検証するために、2011 から 2020 年度の 10 個の学習データを用いて下表のパラメータの範囲で、問題 (E) を求解した。

表5.4 パラメータの範囲

パラメータの種類	範囲
\tilde{q}	-0.15 , -0.10, -0.05
S	5, 10, 15, 20
P	20, 50

5.5.3.2 計算結果

10 個のデータセットに対し、各アルゴリズムで 10800 秒以内に解を得られた回数は表5.5のとおり。なお太字は各ケースで最も大きな値を示している。 $P = 20$ において、B&C で 10800 秒以内で解を得られたケースは、 $\tilde{q} = -0.05$ のみであった。一方で、HA と HA+ は全てのパラメータの範囲で 10 個全てのインスタンスから時間内に解を得た。 $P = 50$ において、HA と HA+ は B&C よりも多くのインスタンスで解を得ている。更に HA+ は全てのパラメータの範囲で、求解したインスタンスの数が一番多かった。

それゆえ、HA+ は B&C および HA と比較すると、最も時間内に解を得られる高速性を持つことがわかる。

表5.5 10800 秒以内に 10 個のインスタンスから求解できた数

P	20								
Method	B&C	HA	HA+	B&C	HA	HA+	B&C	HA	HA+
$S \setminus \tilde{q}$	-0.15	-0.15	-0.15	-0.1	-0.1	-0.1	-0.05	-0.05	-0.05
5	10	10	10	10	10	10	5	10	10
10	10	10	10	10	10	10	4	10	10
15	10	10	10	10	10	10	9	10	10
P	50								
Method	B&C	HA	HA+	B&C	HA	HA+	B&C	HA	HA+
$S \setminus \tilde{q}$	-0.15	-0.15	-0.15	-0.1	-0.1	-0.1	-0.05	-0.05	-0.05
5	0	10	10	0	9	10	0	3	7
10	0	10	10	0	8	9	0	1	1
15	0	10	10	0	7	8	0	1	4
20	0	10	10	0	5	10	0 ^a	0	8

^a 10 個のインスタンスの内、3 個は初期解でさえ得られなかった。

HA および HA+ で求解した際の計算時間の 4 つの統計量 (平均値 (mean)、標準偏差 (s.d.)、最大値 (max)、最小値 (min)) を、 $P \in \{20, 50\}$ のそれぞれのケースに対し、表5.6、5.7で示している。なお 4 つの統計量は HA で 10800 秒で計算を終了したケースのみを使用して評価している。

その結果に注目したところ、HA+ は HA よりも計算時間が数倍速い。その際、計算時

間のバラつきは S が大きいほど、もしくは \tilde{q} が 0 に近いほど大きくなっている。この結果から、HA+ は HA よりも高速な計算性能を持っていると判断した。

表5.6 $P = 20$ での計算時間 (単位: 秒)

		HA	HA+	HA	HA+	HA	HA+
$S \setminus \tilde{q}$		-0.15	-0.15	-0.1	-0.1	-0.05	-0.05
5	mean	59	32	95	51	229	127
	s.d.	29	12	47	23	117	67
	max	115	49	186	98	452	259
	min	21	17	28	21	73	51
10	mean	69	30	150	51	256	97
	s.d.	45	9	80	22	108	38
	max	157	43	274	90	459	175
	min	23	17	52	23	120	42
15	mean	68	32	129	43	188	68
	s.d.	44	9	65	15	71	24
	max	155	44	252	67	302	107
	min	26	17	36	19	83	37

表5.7 $P = 50$ での計算時間 (単位: 秒)

		HA	HA+	HA	HA+	HA	HA+
$S \setminus \tilde{q}$		-0.15	-0.15	-0.1	-0.1	-0.05	-0.05
5	mean	693	375	2396	2558	6620	3992
	s.d.	817	165	1753	1849	2194	1777
	max	3070	771	5453	6747	9126	5278
	min	206	161	581	337	3783	1479
10	mean	740	346	3303	2111	5979	2592
	s.d.	803	109	1355	1619	*a	*a
	max	3112	538	5795	4765	5979	2592
	min	210	159	1555	581	5979	2592
15	mean	1133	416	3871	2043	10607	4583
	s.d.	1082	253	2827	2654	*a	*a
	max	4130	1130	9390	7835	10607	4583
	min	340	166	955	265	10607	4583
20	mean	1384	372	3697	1028	**b	**b
	s.d.	1243	294	2276	881	**b	**b
	max	3727	1228	6681	2749	**b	**b
	min	204	142	1157	322	**b	**b

^a HA を用いて計算終了できた回数は、わずか一回のみ。

^b HA を用いて計算終了できた回数は、0。

問題 (E) の目的関数に関連した bAUC 値の 4 つの統計量 (平均値 (mean)、標準偏差 (s.d.)、最大値 (max)、最小値 (min)) を、 $P = \{20, 50\}$ のそれぞれのケースに対し、表 5.8、5.9 で示している。また 10800 秒以内で計算が終わらない場合は、その時点の解を用いて評価した。なお太字は各パラメータの範囲における bAUC 値の最大値を示している。

HA と HA+ は最適解を確実に得られることが保証されていないヒューリスティックアルゴリズムだが、それらの解と、 ε -最適解を保証されている B&C から得られた解によるそれぞれの目的関数値を比較したところ、多くのケースで、ほぼ同じ値を得られたことを表 5.8 で示している。HA と HA+ により得られた解は、わずかなケースでは B&C で得ら

れた解より劣化している箇所はあるものの、その相対誤差は最大でも 2.14%^{*12}と限定的な水準であった。それゆえ、HA と HA+ は最適解に十分近い解を得られるといえる。

表5.8 $P = 20$ でトレーニングデータから得られた bAUC (単位: %)

Method		B&C	HA	HA+	B&C	HA	HA+	B&C	HA	HA+
$S \setminus \tilde{q}$		-0.15	-0.15	-0.15	-0.1	-0.1	-0.1	-0.05	-0.05	-0.05
5	mean	32.8	32.7	32.7	41.1	41.1	41.1	52.3	52.1	52.1
	s.d.	4.5	4.5	4.5	4.4	4.4	4.4	4.0	3.9	3.9
	max	43.4	43.4	43.4	51.5	51.5	51.5	60.6	59.3	59.3
	min	27.9	27.9	27.9	35.8	35.8	35.8	46.4	45.6	45.6
10	mean	33.9	33.9	33.9	43.2	43.2	43.2	56.8	56.9	56.9
	s.d.	4.7	4.7	4.7	4.5	4.5	4.5	4.2	4.1	4.1
	max	44.7	44.7	44.7	53.4	53.4	53.4	64.6	64.5	64.5
	min	29	29	29	37.6	37.6	37.6	51	51.6	51.6
15	mean	34.0	33.9	33.9	43.4	43.4	43.4	57.6	57.6	57.6
	s.d.	4.7	4.7	4.7	4.5	4.5	4.5	4.1	4.1	4.1
	max	44.7	44.7	44.7	53.5	53.5	53.5	65.2	65.2	65.2
	min	29.0	29.0	29.0	37.7	37.7	37.7	52.4	52.4	52.4

$P = 50$ において B&C から得られた解は、表5.5のとおり、10800 秒以内で計算は終了できなかった暫定解である。その際、表5.9が示しているとおおり、HA と HA+ は B&C より優位な解を得ている。更に HA+ は HA よりも、より良質な解を得ている。そのため、HA+ は $P = 50$ のような求解が容易ではないデータサイズの大きいケースでも、高速に充分質の良い解を得られる手法といえる。

*12 $q = -0.05, S = 5$ の最大値を比較して、 $(60.6 - 59.3) \div 60.6$ と計算した。

表5.9 $P = 50$ でトレーニングデータから得られた bAUC (単位: %)

		B&C	HA	HA+	B&C	HA	HA+	B&C	HA	HA+
$S \setminus \tilde{q}$		-0.15	-0.15	-0.15	-0.1	-0.1	-0.1	-0.05	-0.05	-0.05
5	mean	36.0	38.1	38.1	37.4	45.2	45.2	41.7	53.6	54.2
	s.d.	5.4	4.9	4.9	7.2	4.3	4.3	5.4	3.4	3.1
	max	43.7	47.0	47.0	49.5	53.0	53.2	51.6	60.4	60.6
	min	27.7	30.4	30.4	28.9	39.8	39.8	35.3	49.7	49.7
10	mean	39.1	41.6	41.6	42.2	50.8	50.8	49.1	61.5	61.8
	s.d.	4.3	4.6	4.6	7.0	4.1	4.0	5.6	4.5	4.6
	max	44.7	48.9	48.9	52.9	56.6	56.6	59.1	69.2	69.2
	min	31.2	33.9	33.9	29.6	43.7	44.0	42.4	55.7	56.7
15	mean	40.3	42.4	42.4	47.0	52.6	52.7	52.5	65.7	66.7
	s.d.	4.8	4.6	4.6	5.0	4.3	4.2	6.0	4.4	4.5
	max	46.5	49.6	49.6	55.3	58.4	58.9	61.5	73.0	73.0
	min	31.8	34.9	34.9	36.5	45.4	45.4	43.1	59.6	59.8
20	mean	41.0	42.7	42.7	49.2	53.5	53.7	***a	68.6	70.1
	s.d.	4.4	4.6	4.6	5.3	4.6	4.5	***a	4.3	4.6
	max	46.8	50.1	50.1	58.5	59.9	60.4	***a	75.8	76.8
	min	33.2	35.1	35.1	40.4	45.8	46.2	***a	61.9	62.5

^a 10 個のインスタンスの内、3 個は初期解でさえ得られなかった。

5.5.4 判別性能評価

この小節では、bAUC 最大化モデルは、信用リスク分野の格付判別問題のように説明変数の分布の裾が厚いケースでは高い判別性能を発揮することを示す。

5.5.4.1 シミュレートデータを用いての判別性能評価

5.5.4.1.1 詳細設定 bAUC 最大化モデルの有効性を調査するため、シミュレートした二つの分布のデータを活用する。まず初めに、 $m_+ = 300$ 、 $m_- = 100$ 、 $m = 400$ とした目的変数 ($y_i, (i \in \mathcal{I})$) を用意する。次に、 $P = 10$ とした説明変数を用意する。その際、説

明変数は次のそれぞれの分布に従うようシミュレートする。

裾の薄いケース (DS1) : 平均 0、標準偏差 1 となる対称な正規分布。

裾の厚いケース (DS2) : 平均 0、標準偏差 1、歪度-5、尖度 50 に従う非対称な正規分布。

DS2 は、R パッケージの “SimMultiCorrData” *¹³を用いてシミュレートした。更に目的変数 ($y_i, (i \in \mathcal{I})$) と説明変数 ($x_i, (i \in \mathcal{I})$) の相関係数は 0.2、また説明変数間 ($x_{p_1,i}, (p_1 \in \mathcal{P}, i \in \mathcal{I},)$) と $x_{p_2,i}, (p_2 \in \mathcal{P}, p_1 \neq p_2, i \in \mathcal{I})$) の相関係数は 0.4 でシミュレートしている。

AUC 値は線形分類器の bAUC 最大化モデル (L) と LR を用いて、シミュレートした 10 個のトレーニングデータおよびテストデータのそれぞれで計算している。またその際は $\tilde{q} = -0.05$ で評価した。

5.5.4.1.2 計算結果 DS1 および DS2 を用いた際のそれぞれの AUC 値に対しての 4 つの統計量 (平均値 (mean)、標準偏差 (s.d.)、最大値 (max)、最小値 (min)) を表5.10で示している。

ここで“トレーニング”は、シミュレートされたトレーニングデータを用いての AUC 値で、“テスト”は、そのトレーニングデータで得られた係数とシミュレートされたテストデータを用いての AUC 値である。加えて (L)–LR の列は、(L) と LR の AUC 値の差を示している。

DS1 の結果を見ると、(L) と LR について、大きな違いは見られない。ところが、DS2 の結果を見ると、(L) は LR よりも高い AUC 値が得られていることが観察できた。この結果には、分布の裾を考慮することで誤判別を起こしにくいよう保守的に評価する bAUC 最大化モデルの特徴が現れている。それゆえ、bAUC 最大化モデルは誤判別を表す分布の裾が厚い際は、より優位な判別性能を発揮できることがわかる。

5.5.4.2 実データを用いての判別性能評価

この小節では、実データを適用して、楕円分類器の bAUC 最大化モデルの優位性を示す。

*¹³ <https://cran.r-project.org/web/packages/SimMultiCorrData/SimMultiCorrData.pdf>

表5.10 bAUC 最大化モデルおよび LR から得られる AUC 値 (単位: %)

		(DS1)			(DS2)		
		(L)	LR	(L)-LR	(L)	LR	(L)-LR
“トレーニング”	mean	70.16	70.22	-0.06	78.12	74.18	3.94
	s.d.	2.43	2.24	0.19	3.66	2.55	1.11
	max	72.95	72.6	0.35	85.9	79.59	6.31
	min	65.14	65.7	-0.56	73.5	71.43	2.07
“テスト”	mean	65.44	64.9	0.54	75.93	70.8	5.13
	s.d.	3.06	3.13	-0.07	3.14	3.55	-0.41
	max	71.98	71.66	0.32	81.02	74.99	6.03
	min	61.25	60.85	0.4	70.51	63.45	7.06

5.5.4.2.1 信用リスクデータの分布の分析 この段落では、5.5.2.2 小節で述べた実データの分析結果を記載する。図5.2は 550 個^{*14} の説明変数の歪度および尖度をプロットしたものである。この図から多くの説明変数は強い歪度と尖度を持っている非対称な分布であることがわかる。bAUC 最大化モデルは、5.5.4.1.1 段落で実験した非常に非対称な分布 (DS2) を用いると、よりその判別性能がより活かされるので、信用リスク分野での判別問題であれば、十分に高い汎化性能を得られることが期待できる。

5.5.4.2.2 楕円分類器および変数選択による効果 この段落では、楕円分類器および変数選択を用いることで判別性能が向上することを示す。

5.5.4.2.2.1 詳細設定 汎化性能を評価するため、テストデータの AUC 値を計算する。2011 から 2020 までの各トレーニングデータで係数を計算し、その後、2012 から 2021 までの各テストデータで AUC 値を計測する。そのため、まず最初に (\tilde{q}, S) のパラメータ値を決定する必要がある。

このために、5 回の交差検証 (CV) を用いたグリッドサーチを用いる。具体的には、トレーニングデータを 5 個に分割して、そのうち一つをサブテストデータとし、残りを結合したものをサブトレーニングデータとする。そして、所与のパラメータ (\tilde{q}, S) において、

*14 表5.2にある 11 年 × 50 個の説明変数で分析した。

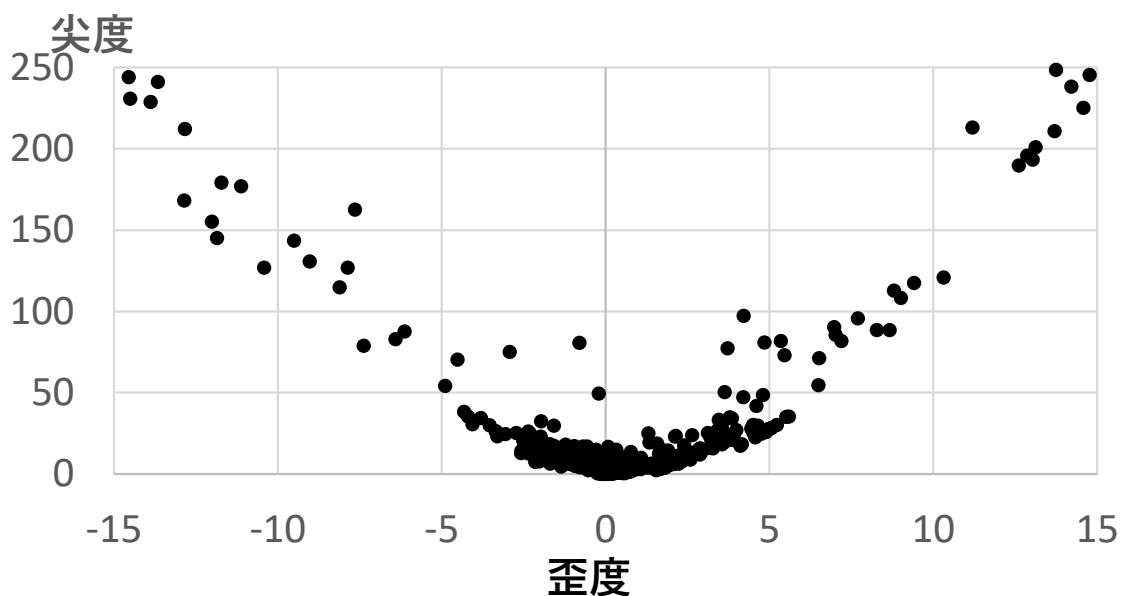


図5.2 実データの歪度と尖度

サブトレーニングデータを用いて最適化問題を解くことで係数 (\mathbf{b}, \mathbf{B}) を計算し、その後サブテストデータを用いて AUC 値を計測する。この手順を 5 回繰り返す。その上で、下のパラメータの候補となる値の組み合わせから、最もサブテストデータの AUC 値が高かった際の (\tilde{q}, S) の水準を用いることとする。なお、この手順の概要図は図5.3のとおりである。

$$\begin{aligned} \tilde{q} &\in \{-0.4, -0.3, -0.25, -0.2, -0.175, -0.15, \\ &\quad -0.125, -0.1, -0.075, -0.05, -0.025, -0.01\}, \\ S &\in \{5, 10, 15, 20\}. \end{aligned}$$

このとき、(E)、(L)、(Q) で調整の対象となるハイパーパラメータは \tilde{q} と S だが (E)、(L)、(Q) では \tilde{q} のみである。

加えて、(E)、(L)、(Q) の AUC 値の計測には、事前を選択した説明変数のセットを用いる。この選択のために Saito et al. (2021) が用いたエラスティックネット (Zuo and Hastie, 2005) を活用した手順を採用した。まず最初にエラスティックネットの係数を計算し、次に絶対値換算で最も高い係数の値に対して、10% 以上の係数を算出できた変数を採用する。表5.11はそれぞれのトレーニングデータにおいて採用した説明変数の個数を示

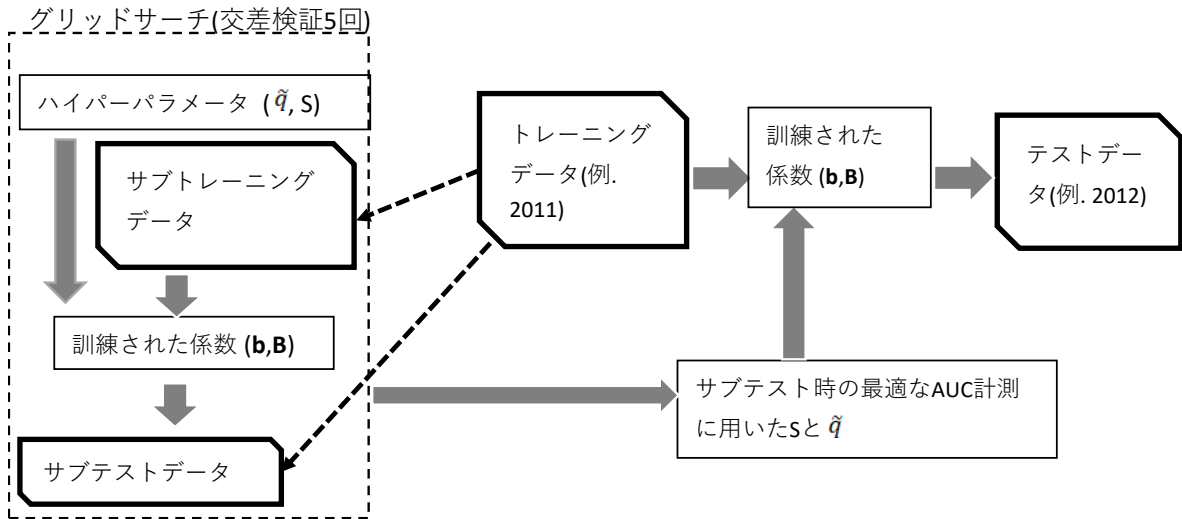


図5.3 計算手順の概要図

している。

表5.11 (E)、(L)、(Q) の評価に用いる説明変数の個数

年度	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
個数	7	6	16	12	15	11	13	16	12	15

5.5.4.2.2.2 計算結果 各 bAUC 最大化モデルから得られたそれぞれの AUC 値に対しての 4 つの統計量 (平均値 (mean)、標準偏差 (s.d.)、最大値 (max)、最小値 (min)) を表5.12で示している。さらに“トレーニング”は、2011 年度から 2020 年度のトレーニングデータを用いての AUC 値で、“テスト”は、2012 年度から 2021 年度のテストデータを用いての AUC 値である。そして“差異”は、トレーニングされた AUC 値からテストされた AUC 値を差し引いた値である。また、(E) に関しては、HA+ を用いて 10800 秒までに得られた解を採用している。

“トレーニング”の AUC 値は、(Q) のほうが (E) と (L) よりも平均的に高く、また (Q) のほうが (E) と (L) よりも平均的に高い。しかしながら、“テスト”の AUC 値は、(E) と (L) のほうが (Q) よりも平均的に高く、また (E) と (L) のほうが (Q) よりも平均的に高い。更に、(E) は (L) を上回り、(E) は (L) を上回った。“差異”の AUC 値は、楕円形の

汎化性能は線形分類器のそれよりも高いにも関わらず、(E) は (L) とほぼ同じ水準であり、(E) もまた (L) とほぼ同じ水準だった。よって、この結果から楕円分類器はNorton and Uryasev (2019) が提示している線形分類器よりも、判別性能が優れていることがわかる。

表5.12 各 bAUC 最大化モデルによる AUC 値 (単位: %)

		(E)	(Q)	(L)	(E)	(Q)	(L)
“トレーニング”	mean	88.4	89.1	86.7	84.9	85.7	82.9
	s.d.	2.2	6	3.8	2.4	4.6	3.1
	max	91.9	100	92.6	89.2	96.1	88.5
	min	85.4	81.3	81.3	80.8	79.8	76.7
“テスト”	mean	83.0	77.6	79	80.4	78.5	78.9
	s.d.	3.3	4	3.1	3.2	3.9	3.4
	max	86.7	83.4	83.8	86.3	85.6	84.8
	min	75.3	72.1	74.2	73.4	72.3	71.8
“差異”	mean	5.4	11.5	7.7	4.6	7.2	4
	s.d.	4.5	6.9	6	3.4	5.1	5.1
	max	15.6	23.3	18.4	12.9	19.1	16.7
	min	0.8	-1.3	-0.9	1.9	2.2	-1.1

図5.4で示しているのは10年間のテストデータから計測されたAUC値である。(E)は、全てのケースで、(L)と(Q)よりも高いAUC値を得ている。(E)は、2017年度と2018年度を除くと、(L)と(Q)よりも高いAUC値を得ている。更に(E)は、(E)よりもAUC値が常時上回っている。従って、変数選択を用いることで楕円分類器の汎化性能が改善することがわかる。

5.5.4.2.3 機械学習モデルとの性能比較 この段落では、各 bAUC 最大化モデルと既存の代表的な機械学習モデルとの汎化性能を比較する。

5.5.4.2.3.1 詳細設定 標準的な機械学習モデルは、通常、変数選択の制約を含めていない。それゆえ、小段落 5.5.4.2.2 内の変数選択制約 (5.28)、(5.29)、(5.30) を付けていない (E) と (L) の計算結果を用いる。更に、機械学習モデルは表5.11で述べた (E) と (L) に適用している説明変数を同様に用いる。

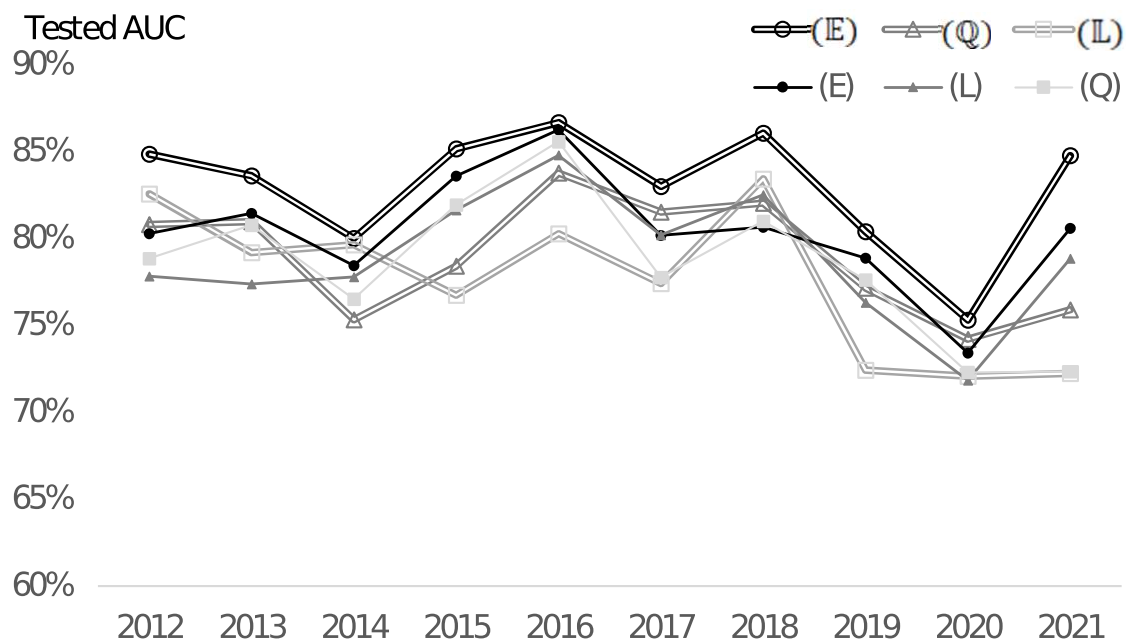


図5.4 各 bAUC 最大化モデルによるテストデータによる AUC 値

用いる機械学習モデルは NN、SVM、RF、LR である。それらのモデルのハイパーパラメータは 5.5.4.2.2.1 小段落で述べている方法と同様のグリッドサーチで決定する。NN は中間層の数を 2,3,4 の中から決定し、SVM は分類器 (判別面) を線形、ガウシアン、シグモイド、多項式の中から決定する。RF は無作為に作成した複数の決定木の中から一つを決定し、LR は L_2 -正則化項の水準を決定する。

その後、小段落 5.5.4.2.2.2 と同様に、(E) と (L) および機械学習の汎化性能を評価する。

5.5.4.2.3.2 計算結果 表5.13は、(E) と (L) および機械学習モデルの AUC 値に関する 4 つの統計量 (平均値 (mean)、標準偏差 (s.d.)、最大値 (max)、最小値 (min)) を示している。更に “トレーニング”、“テスト”、“差異” は表5.12と同様の指標を示している。“トレーニング” においては、RF は平均的に最も高い AUC 値を得た。一方で、“テスト” に関しては、bAUC 最大化モデルの (E) と (L) が機械学習モデルのいずれよりも高い AUC 値を得た。“差異” を見ると、RF はトレーニングデータとテストデータから得た AUC 値に最も大きい乖離があることがわかる。ただし、他のモデルから生じた乖離はそれより低いか、もしくはほぼ同等であった。これらの結果から bAUC 最大化モデルの汎化性能は、今回取り上げた機械学習モデルよりも上回っていることがわかる。

図5.5は 10 年分のテストデータによる AUC 値を示している。小段落 5.5.4.2.2.2 で展開

表5.13 モデル別のトレーニングとテストデータによる AUC 値 (単位: %)

		(E)	(L)	RF	NN	LR	SVM
“トレーニング”	mean	84.9	82.9	89.6	81.1	81.4	78.5
	s.d.	2.4	3.1	5.7	3.9	3.8	4.2
	max	89.2	88.5	98	88	88	85.4
	min	80.8	76.7	78.3	74	73.3	71
“テスト”	mean	80.4	78.9	74.7	76.6	76.8	74
	s.d.	3.2	3.4	2.9	2.9	3.3	4.3
	max	86.3	84.8	80.3	80.7	81.5	79.9
	min	73.4	71.8	70.9	71.6	71.4	68.2
“差異”	mean	4.6	4.0	14.9	4.4	4.5	4.5
	s.d.	3.4	5.1	5.5	5.2	5.2	5.6
	max	12.9	16.7	21.5	16.4	16.6	17.2
	min	1.9	-1.1	4	-1.3	-2	-0.7

した bAUC 最大化モデルである (E) と (L) は、機械学習モデルよりも高い AUC 値を、2017 年度の結果を除いて、得ている。また、その 2017 年度での bAUC 最大化モデルによる AUC 値の最大値と機械学習の最大値との差異はわずか 0.1% であった。それゆえ、bAUC 最大化モデルによる汎化性能は、充分優れていると判断する。

本論文で取り上げた判別問題において、格付を得られる企業数はそもそも多くないことから、トレーニングデータのサイズは一年あたりに数百程度しか取得できない。そのサイズの大きさは、財務指標を用いた信用リスクの分類という通常の問題では先行研究も同程度の水準であることから妥当な水準といえる (Gotoh et al., 2014; Saito et al., 2021; Saito and Yamanaka, 2021; Tanaka and Nakagawa, 2014)。しかし、高い自由度を持つ機械学習モデルを使用する際にはそのサイズは少ない。その結果、機械学習モデルはトレーニングデータにうまく適合できなかったことから、テストデータでも十分な分類ができなかったものと考えている。

なお、4 章で取り上げた CVaR 最小化モデルは SVM を再定式化したモデルである。そのため、(L) と SVM は線形を前提にして、bAUC 最大化モデルと CVaR 最小化モデルの判別性能を比較した結果といえる。そのテストデータを見た結果、bAUC 最大化モデルのほうが判別性能は高かった。これは bAUC 最大化モデルの持つテイルリスク抑制の特徴が大きく効いたためと考える。言い換えると bAUC 最大化モデルは計算規模が大きい

ので、計算時間は長いものの、それに見合った判別性能を得ることができるモデルと考えられる。

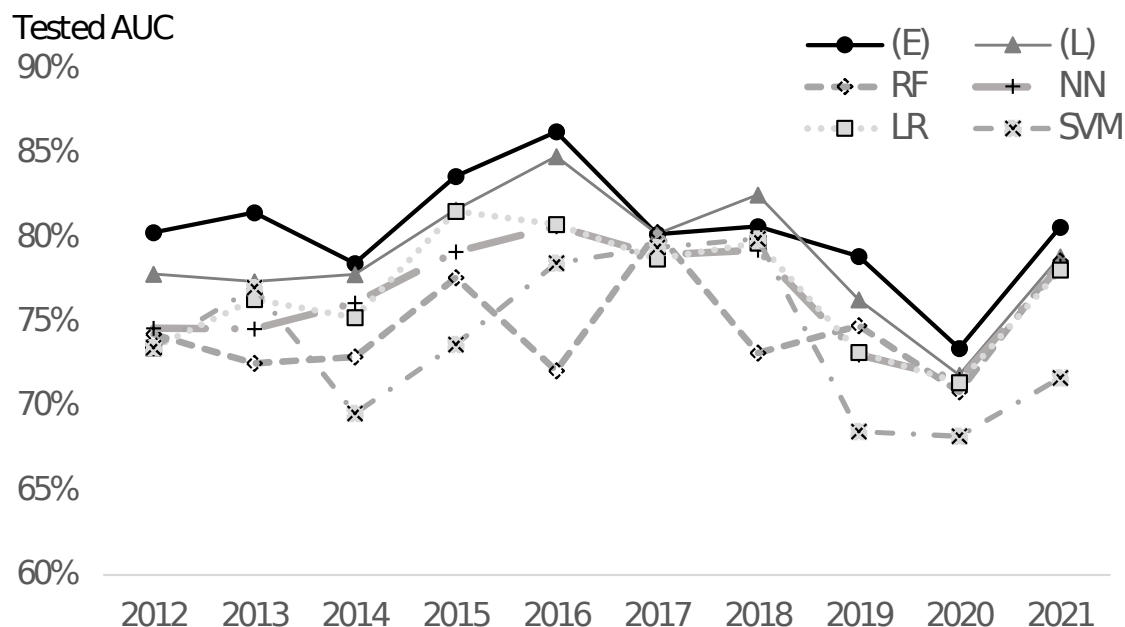


図5.5 (E) と (L) および機械学習モデルによるテストデータによる AUC 値

5.5.4.2.3.3 テイルリスクの管理 5.2.2 小節で記述したとおり、bAUC 最大化モデルは、大きく誤った分類を生じうるテイルリスクを管理することができる。これを確認するため、表5.14で、bAUC 最大化モデル ((E), (L)) および機械学習モデル*¹⁵の“テスト”での bAUC 値に関する 4 つの統計量を $\tilde{q} \in \{-0.05, -0.01\}$ のケースで比較した。

この表から、bAUC 値は、bAUC 最大化モデルのほうが機械学習モデルより高い値を得られることがわかった。それゆえ bAUC 最大化モデルはテイルリスクを最小化したことにより、優位な判別性能を得られることがわかる。

5.6 結論

本章では、Norton and Uryasev (2019) が提案した bAUC 最大化モデルを既存の線形分類器 (L) から楕円分類器 (E) に拡張し、かつ変数選択制約を導入したモデルを提案し

*¹⁵ Python では NN と RF はスコア関数を出力できないので、(5.1) 式にある $R(\mathbf{w} | \mathbf{x}_i)$, ($i \in \mathcal{I}$) に対して出力できる確率を用いて評価した。

表5.14 モデル別のテストデータによる bAUC 値 (単位: %)

\tilde{q}		(E)	(L)	RF	NN	LR	SVM
-0.05	mean	42.29	40.29	30.65	36.54	37.08	25.54
	s.d.	6.82	7.6	6.53	5.13	6.28	14.39
	max	55.62	54.03	42.92	44.3	46.85	44
	min	28.72	25.59	21.35	27.87	28.18	0.44
-0.01	mean	49.64	47.55	37.04	43.02	43.74	33.96
	s.d.	6.9	7.73	5.94	5.08	6.12	12.65
	max	62.72	61.22	48.7	50.43	53.23	51.13
	min	36.06	32.51	30.63	33.68	34.18	10.85

た。提案モデルは、相当量の制約式を有する MISDP に定式化される。そのため、既存のアルゴリズムで求解可能なケースは限定的なので、問題削減を実行するようヒューリスティックアルゴリズムを改良することで、計算の高速化を可能にした。

加えて高速化の改善を実施したヒューリスティックアルゴリズムは、 ε -最適解を得られることが保証されている既存アルゴリズムと比較しても、解の精度はほぼ劣化していないことを確認した。更に、検証において説明変数の候補数が最も多い 50 個の場合に限って見ると、10800 秒以内で得られる解は最も精度が良かった。

モデルの汎化性能に関しては、楕円分類器となる bAUC 最大化モデルは、線形および二次分類器よりも汎化性能が高かった。更に変数選択を考慮した場合、その汎化性能をより押し上げる結果となった。また、bAUC 最大化モデルは既存の機械学習モデルよりも高い汎化性能を得たが、それは極端に大きな誤判別を抑制できるようテイルリスクを管理できる特徴を有しているためと考えられる。

なお、その際に機械学習モデルの性能が劣化したのは、データサイズが数百とデータが少なく複数のハイパーパラメータに対して、うまく適合しなかったことが考えられる。また、この数百というサンプル数の規模は、対象とした企業の格付の判別問題では、先行研究からも自然な水準なため (Gotoh et al., 2014; Saito et al., 2021; Saito and Yamanaka, 2021; Tanaka and Nakagawa, 2014)、ある程度的前提を基に構築されている bAUC 最大化モデルのほうが、判別性能がより発揮されやすかったと考えられる。

最後に、本章では判別は二グループを前提としている。しかし企業の格付判定 (Tanaka and Nakagawa, 2014; Konno and Saito, 2013; Gotoh et al., 2014) など、より多グループな判別が必要なケースがある。そのような判別もできるよう、MISDP となる bAUC 最

大化モデルの拡張を考えていきたい。

第6章

結論と今後の課題

本論文では、既存の統計型モデルを前提とした格付判別モデルの判別性能を向上するための二つの改良を実施した。その改良点は、一般的な判別面である線形ではなく楕円形を採用したこと、および、採用する変数の個数をユーザーが事前に設定した個数以内に限定する変数選択を採用したことである。それぞれを実行したことで判別性能が向上した例はあるが、双方を併せて実行した例は無い。楕円形の判別面は、単調性を持たず高いもしくは低いと倒産に影響するような財務変数を利用でき、判別精度を向上させることが期待されるが、モデル構造が非線形となるため選択される変数が多くなり、多重共線性やオーバーフィットなどの問題を引き起こす可能性がある。そこで、変数選択制約も考慮し、必要最低限の数で最も重要な変数を選択した上で、楕円形判別面を構成できるモデルは、判別精度を向上させる上で効果的であると考えられる。そこで、この二つを同時に実行できるよう拡張したモデルを提案した。

一方で提案したモデルを取り扱う上での課題としては、パラメータを推計する最適化問題が MISDP となり既存アルゴリズムでの求解は膨大な計算時間を要する点があった。このままでは実用的な解を一定の時間内で得られる可能性は低いいため、短時間で質の良い解を得るためのヒューリスティックアルゴリズムを提案することで、求解を行った。拡張の対象にした統計型モデルは、制約式の本数が限定的となる中規模な CVaR 最小化モデルと、その数十倍の本数を有する大規模な bAUC 最大化モデルである。いずれの問題に対しても、その複雑性を問わず提案したアルゴリズムを用いることで、既存アルゴリズムと比較して高速に良質な解を得ることができた。併せて得られた判別性能は、楕円形判別面や変数選択を片方だけ取り入れた既存モデルと比較して、より向上したことも確認した。この結果より、楕円形判別面と変数選択制約を同時に考慮することは、統計モデルの判別精度を向上させるために、重要な改良であると結論付けられる。

昨今、判別問題は金融業界では、本論文で取り上げた信用リスクの分野での企業の倒産もしくは存続といった問題だけでなく、コンプライアンス関連におけるマネーロンダリングなどのような不正取引の検知といった分野で、取り上げられている事例が散見され始めている。こういった事例には機械学習モデルを使用しているケースもあり、判別問題に使用できるモデルの研究や開発に関しては、今後更に需要が高まっていくことを想定している。それを踏まえて、実務上の判別問題を解決するためのモデルの候補を検討する際、本論文で提示した MISDP となるモデルが、有力な候補に挙げられることを期待している。

最後にモデルに関する今後の課題を三つ挙げる。

- 1つ目は、より大きな問題のデータに対して、良質な解を得る手法の検討である。本論文で用いたデータは、企業数が数百社程度および説明変数の候補数が最大で 50 程度であった。一方でより大きな問題のデータで解を得ることは、より優位な汎化性能を引き出すことにつながる。信用リスクの分野における企業を倒産もしくは存続するかの二つに判別する例であれば、Konno and Yamamoto (2008) は、中小企業数が 1000 社および説明変数の候補数が 70 個、更に Konno and Takaya (2009) は中小企業数が数千社および説明変数の候補数が最大で 114 個ものデータセットを用いて、変数選択と判別を実行している。またいずれも、候補数の 70 から 10 個と変数選択を一段階で実行するのではなく、 $70 \rightarrow 30 \rightarrow 10$ もしくは $70 \rightarrow 30 \rightarrow 20 \rightarrow 10$ と二段階もしくは三段階に分けて実施することで、大幅な計算時間が削減できること、および、得られた解の劣化は限定的であることを報告している。このため、まずは同様の多段階なアプローチを用いることで、よりデータサイズを大きくした MISDP となる 4 章で提示した CVaR 最小化モデルおよび 5 章で提示した bAUC 最大化モデルに対し、一定時間内に、充分信頼性の高い解を得られることを検証していきたい。
 - 2つ目は、判別を 2 つではなく、それ以上に判別できるように、モデルを拡張することである。これにより、複数の格付の判別を行えるようにしたい。Gotoh et al. (2014) は、CVaR 最小化モデルを基にして三つ以上に判別できる拡張したモデルを報告している。一方で、そのケースの判別面は線形のみを対象としている。更に変数選択のための離散変数を導入したモデルでもない。そのため楕円形と変数選択を同時評価できるように MISDP にまで拡張したモデルはまだ見受けられない。従って Gotoh et al. (2014) が提示したモデルに半正定値制約および離散整数を取り込み、MISDP にまで拡張することを試みたい。
- また bAUC 最大化モデルに関しては、判別が線形の場合も含めて三つ以上に判別

できるよう拡張したモデルの報告例はない。そのため、MISDPへ拡張する前に、まずは複数の判別ができるためのモデルの拡張から検討する必要がある。この点については、山下・三浦(2011)の4章で提示されている以下の Rating AUC がヒントになると想定している。

$$\text{RAUC} := \frac{\sum_{k=1}^{K-1} \text{AUC}_k}{K-1} \quad (6.1)$$

ここで、格付に $1, \dots, K$ と K 個の区分があるとすると、 AUC_k は格付 k 以下と K 以上の二つに判別している AUC の値となる。これにより仮に $K=4$ の際は $\{1\}$ と $\{2, 3, 4\}$ 、 $\{1, 2\}$ と $\{3, 4\}$ 、 $\{1, 2, 3\}$ と $\{4\}$ と複数の判別について AUC で評価している。

このように、先行研究を参照にすると CVaR 最小化モデルおよび bAUC 最大化モデルにおいて、複数の判別用にモデルを拡張できる方法は何らか有り得ると考えている。しかし複数の判別できるよう拡張したモデルは、変数および制約条件が確実に大きく増加することから、解を得るのに更なる時間を要することが予想される。この点から、1つ目の点でも述べた更なる計算時間の高速化が行えるアプローチの取り込みも平行して進めていきたい。判別問題においては、単純に二者択一を実行する判別のケースだけでなく、より多種多様な判別を求められる実務上のニーズもあると考えられる。そのため、複数の判別できる目的を満たした判別モデルを、継続して検討していきたい。

- 3つ目は、今回提案したモデルに対して採用された財務指標の解釈性を検討することが挙げられる。変数選択制約を加えたことで必要最低限の財務指標でモデルを構築できたものの、楕円形判別面は線形判別面に比べ、どの財務指標が信用リスクにどの程度影響を与えているかが解釈しにくいというデメリットがある。一方で、金融実務で信用リスクモデルを使用する際には、顧客に対してどうしてその結果になったのかという説明ができることが重要となる。そのため、楕円形判別面のモデルに関しても出力結果の要因分析を行う手法を開発できれば、実務上の説明力を向上させ、金融実務への普及が行いやすくなると考えている。

参考文献

- Al-Khayyal, F. A. (1990) “Jointly constrained bilinear programs and related problems: An overview,” *Computers and Mathematics with Applications*, Vol. 19, No. 11, pp. 53–62, DOI: [https://doi.org/10.1016/0898-1221\(90\)90148-D](https://doi.org/10.1016/0898-1221(90)90148-D).
- Altman, E. I. (1968) “Financial ratios, discriminant analysis and the prediction of corporate bankruptcy,” *The Journal of Finance*, Vol. 23, No. 4, pp. 589–609, DOI: <https://doi.org/10.1111/j.1540-6261.1968.tb00843.x>.
- Bertsimas, D. and A. King (2017) “Logistic regression: From art to science,” *Statistical Science*, Vol. 32, No. 3, pp. 367–384, DOI: <https://doi.org/10.1214/16-STS602>.
- Cristianini, N. and J. Shawe-Taylor (2000) *An introduction to support vector machines and other kernel based learning methods*, Cambridge University Press: Cambridge.
- Friedman, J., T. Hastie, and R. Tibshirani (2010) “Regularization paths for generalized linear models via coordinate descent,” *Journal of Statistical Software*, Vol. 33, No. 1, pp. 1–22, DOI: <https://doi.org/10.18637/jss.v033.i01>.
- Gotoh, J. and A. Takeda (2005) “A linear classification model based on conditional geometric score,” *Pacific Journal of Optimization*, Vol. 1, No. 2, pp. 277–296, DOI: <https://doi.org/10.1007/s10287-013-0175-5>.
- (2011) “On the role of norm constraints in portfolio selection,” *Computational Management Science*, Vol. 8, pp. 323–35. DOI: <https://doi.org/10.1007/s10287-011-0130-2>.
- Gotoh, J., A. Takeda, and R. Yamamoto (2014) “Interaction between financial risk measures and machine learning methods,” *Computational Management Science*, Vol. 11, pp. 365–402, DOI: <https://doi.org/10.1007/s10287-013-0175-5>.
- Guyon, J. and A. Elisseeff (2003) “An introduction to variable and feature selection,” *The Journal of Machine Learning Research*, Vol. 3, No. 7-8, pp. 1157–1182.

- Hoerl, Arthur E. and Robert W. Kennard (1970) “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, Vol. 12, pp. 55–67, DOI: <https://doi.org/10.2307/1271436>.
- Kobayashi, K. and Y. Takano (2020) “A branch-and-cut algorithm for solving mixed-integer semidefinite optimization problems,” *Computational Optimization and Applications*, Vol. 75, pp. 493–513, DOI: <https://doi.org/10.1007/s10287-003-0001-6>.
- Konno, H. and H. Kobayashi (2000) “Failure discrimination and rating of enterprises by semidefinite programming,” *Asia-Pacific Financial Markets*, Vol. 7, pp. 261–273, DOI: <https://doi.org/10.1023/A:1010013117888>.
- Konno, H. and M. Saito (2013) “Classification of companies using maximal margin ellipsoidal surfaces,” *Computational Optimization and Applications*, Vol. 55, pp. 469–480, DOI: <https://doi.org/10.1007/s10589-012-9508-5>.
- Konno, H. and Y. Takaya (2009) “Multi-step methods for choosing the best set of variables in regression analysis,” *Computational Optimization and Applications*, Vol. 46, pp. 417–426, DOI: <https://doi.org/10.1007/s10589-008-9193-6>.
- Konno, H. and D. Wu (2002) “Estimation of failure probability using semi-definite programming(in Japanese),” *The Japan Society for Industrial and Applied Mathematics*, Vol. 12, No. 2, pp. 121–134, DOI: <https://doi.org/10.1007/s10287-003-0001-6>.
- Konno, H. and R. Yamamoto (2005) “Integer programming approaches in mean-risk models,” *Computational Management Science*, Vol. 2, pp. 339–351, DOI: <https://doi.org/10.1007/s10287-005-0038-9>.
- (2008) “Choosing the best set of variables in regression analysis using integer programming,” *Journal of Global Optimization*, Vol. 44, pp. 273–282, DOI: <https://doi.org/10.1007/s10898-008-9323-9>.
- Konno, H., J. Gotoh, T. Uno, and A. Yuuki (2002) “A cutting plane algorithm for semi-definite programming problems with applications to failure discriminant analysis,” *Journal of Computational and Applied Mathematics*, Vol. 146, No. 1, pp. 141–154, DOI: [https://doi.org/10.1016/S0377-0427\(02\)00424-7](https://doi.org/10.1016/S0377-0427(02)00424-7).
- Konno, H., N. Kawadai, and H. Tuy (2003a) “Cutting plane algorithms for nonlinear semi-definite programming problems with applications,” *Journal of Global Optimization*, Vol. 25, pp. 141–155, DOI: <https://doi.org/10.1023/A:1021985014197>.
- Konno, H., N. Kawadai, and D. Wu (2003b) “Estimation of failure probability using semi-definite logit model,” *Computational Management Science*, Vol. 1, pp. 59–73,

- DOI: <https://doi.org/10.1007/s10287-003-0001-6>.
- Laitinen, E. K. and T. Laitinen (2000) “Bankruptcy prediction applications of the Taylor’s expansion in logistic regression,” *International Review of Financial Analysis*, Vol. 9, No. 4, pp. 327–349, DOI: [https://doi.org/10.1016/S1057-5219\(00\)00039-9](https://doi.org/10.1016/S1057-5219(00)00039-9).
- Mafusalov, A. and S. Uryasev (2018) “Buffered probability of exceedance: Mathematical properties and optimization,” *SIAM Journal on Optimization*, Vol. 28, No. 2, pp. 1077–1103, DOI: <https://doi.org/10.1137/15M1042644>.
- Maldonado, S., J. Pérez, R. Weber, and M. Labbé (2014) “Feature selection for support vector machines via mixed integer linear programming,” *Operations Research Letters*, Vol. 279, pp. 163–175, DOI: <https://doi.org/10.1016/j.ins.2014.03.110>.
- Mangasarian, O. L. (1999) “Arbitrary-norm separating plane,” *Operations Research Letters*, Vol. 24, No. 3, pp. 15–23, DOI: [https://doi.org/10.1016/S0167-6377\(98\)00049-2](https://doi.org/10.1016/S0167-6377(98)00049-2).
- McCormick, G. P. (1976) “Computability of global solutions to factorable nonconvex programs: Part I - Convex underestimating problems,” *Mathematical Programming*, Vol. 16, No. 2, pp. 147–175.
- McFadden, D. (1974) *Conditional logit analysis of qualitative choice behavior*, Chap. 4, pp. 105–142: Academic Press.
- Naganuma, M., Y. Takano, and R. Miyashiro (2019) “Feature subset selection for ordered logit model via tangent-plane-based approximation,” *IEICE Transactions on Information and Systems*, Vol. E102.D, No. 5, pp. 1046–1053, DOI: <https://doi.org/10.1587/transinf.2018EDP7188>.
- Norton, M. and S. Uryasev (2019) “Maximization of AUC and buffered AUC in binary classification,” *Mathematical Programming*, Vol. 174, pp. 575–612, DOI: <https://doi.org/10.1007/s10107-018-1312-2>.
- Norton, M., A. Mafusalov, and S. Uryasev (2017) “Soft margin support vector classification as buffered probability minimization,” *Journal of Machine Learning Research*, Vol. 18, pp. 1–43.
- Norton, M., V. Khokhlov, and S. Uryasev (2021) “Calculating CVaR and bPOE for common probability distributions with application to portfolio optimization and density estimation,” *Annals of Operations Research*, Vol. 299, pp. 1281–1315, DOI: <https://doi.org/10.1007/s10479-019-03373-1>.

- Okada, Y and H. Konno (2009) “Failure discrimination by semi-definite programming using a maximal margin ellipsoidal surface,” *Journal of Computational Finance*, Vol. 12, No. 3, pp. 63–77, DOI: <https://doi.org/10.21314/JCF.2009.206>.
- Pedregosa, F., G. Varoquaux, A. Gramfort et al. (2011) “Scikit-learn: Machine learning in python,” *Journal of Machine Learning Research*, Vol. 12, pp. 2825–2830.
- Perez-Cruz, F., J. Weston, D. J. L Hermann, and B. Schölkopf (2003) *Extension of the ν -SVM range for classification*, Vol. 190: IOS Press: Amsterdam, pp.179-196.
- Pertaia, G., A. Prokhorov, and S. Uryasev (2021) “A new approach to credit ratings,” *Journal of Banking & Finance*, Vol. 140, 106097, DOI: <https://doi.org/10.1016/j.jbankfin.2021.106097>.
- Rockafellar, T. R. and S. Uryasev (2000) “Optimization of conditional value-at-risk,” *Journal of Risk*, Vol. 2, No. 3, pp. 21–41, DOI: <https://doi.org/10.21314/JOR.2000.038>.
- (2018) “Minimizing buffered probability of exceedance by progressive hedging,” *Mathematical Programming*, Vol. 181, pp. 453–472, DOI: <https://doi.org/10.1007/s10107-019-01462-4>.
- Rockafellar, T.R. and J. Royset (2010) “On buffered failure probability in design and optimization of structures,” *Reliability Engineering & System Safety*, Vol. 95, No. 5, pp. 499–510, DOI: <https://doi.org/10.1016/j.ress.2010.01.001>.
- Saito, M. and H. Konno (2009) *Classification of companies using ellipsoidal surfaces*, ISE 09-05, Department of Industrial and Systems Engineering, Chuo University.
- Saito, M. and S. Yamanaka (2021) “Performance evaluation of least-squares probabilistic classifier for corporate credit rating classification problem,” *JSIAM Letters*, Vol. 13, pp. 9–12, DOI: <https://doi.org/10.14495/jsiaml.13.9>.
- Saito, M., T. Ohsato, and S. Yamanaka (2021) “An empirical evaluation of machine learning performance in corporate sales growth prediction,” *JSIAM Letters*, Vol. 13, pp. 25–28, DOI: <https://doi.org/10.14495/jsiaml.13.25>.
- Sato, T., Y. Takano, R Miyashiro, and A. Yoshise (2016) “Feature subset selection for logistic regression via mixed integer optimization,” *Computational Optimization and Applications*, Vol. 64, No. 3, pp. 865–880, DOI: <https://doi.org/10.1007/s10589-016-9832-2>.
- Sato, T., Y. Takano, and R. Miyashiro (2017) “Piecewise-linear approximation for feature subset selection in a sequential logit model,” *Journal of the Operations*

- Research Society of Japan*, Vol. 60, No. 1, pp. 1–14, DOI: <https://doi.org/10.15807/jorsj.60.1>.
- Schölkopf, B., A. J. Smola, R. C. Williamson, and P. L. Bartlett (2000) “New support vector algorithms,” *Neural Computation*, Vol. 12, pp. 1207–1245, DOI: <https://doi.org/10.1162/089976600300015565>.
- Shaonan, T., U. Yan, and G. Hui (2015) “Variable selection and corporate bankruptcy forecasts,” *Journal of Banking & Finance*, Vol. 52, pp. 89–100, DOI: <https://doi.org/10.1016/j.jbankfin.2014.12.003>.
- Shirakawa, H. (2001) “Credit risk management by scoring,” *Communications of Operations Research of Japan*, Vol. 46, No. 11, pp. 628–634.
- SCIP Optimization Suite (2023) “SCIP-SDP a mixed integer semidefinite programming plugin for SCIP,” <http://www.opt.tu-darmstadt.de/scipsdp/>, Accessed 21 March 2023.
- Takeda, A. (2009) “Generalization performance of ν -support vector classifier based on conditional value-at-risk minimization,” *Neurocomputing*, Vol. 72, No. 10-12, pp. 2351–2358, DOI: <https://doi.org/10.1016/j.neucom.2008.11.022>.
- Takeda, A. and M. Sugiyama (2008) *ν -support vector machine as conditional value-at-risk minimization*, Proceedings of the 25th International Conference on Machine Learning (ICML 2008), Helsinki, Finland.
- Tamura, R., K. Kobayashi, Y. Takano, R. Miyashiro, K. Nakata, and T. Matsui (2017) “Best subset selection for eliminating multicollinearity,” *Journal of the Operations Research Society of Japan*, Vol. 60, No. 3, pp. 321–336, DOI: <https://doi.org/10.15807/jorsj.60.321>.
- (2019) “Mixed integer quadratic optimization formulations for eliminating multicollinearity based on variance inflation factor,” *Journal of Global Optimization*, Vol. 73, No. 3, pp. 431–446, DOI: <https://doi.org/10.1007/s10898-018-0713-3>.
- Tamura, R., Y. Takano, and R. Miyashiro (2022) “Feature subset selection for kernel SVM classification via mixed-integer optimization,” *arXiv*, p. 2205.14325, DOI: <https://doi.org/10.48550/arXiv.2205.14325>.
- Tanaka, K. and H. Nakagawa (2014) “A method of corporate credit rating classification based on support vector machine and its validation in comparison of sequential logit model (in Japanese),” *Transactions Operations Research Society of Japan*, Vol. 57, pp. 92–111, DOI: <https://doi.org/10.15807/torsj.57.92>.

- Tian, S., Y. Yu, and H. Guo (2015) “Variable selection and corporate bankruptcy forecasts,” *Journal of Banking & Finance*, Vol. 52, pp. 89–100, DOI: <https://doi.org/10.1016/j.jbankfin.2014.12.003>.
- Tibshirani, R (1996) “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society, Series B*, Vol. 58, No. 1, pp. 267–288, DOI: <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>.
- Vapnik, V.N. (1995) *The nature of statistical learning theory*, Springer-Verlag New York, DOI: <https://doi.org/10.1007/978-1-4757-3264-1>.
- Wei, X., J. Gotoh, and S. Uryyasev (2018) “Peer-to-peer lending: Classification in the loan application process,” *Risk*, Vol. 6, No. 4, p. 129, DOI: <https://doi.org/10.3390/risks6040129>.
- Yeo, I. K. and R. A. Johnson (2000) “A new family of power transformations to improve normality or symmetry,” *Biometrika*, Vol. 87, No. 4, pp. 954–959, DOI: <https://doi.org/10.1093/biomet/87.4.954>.
- Zuo, H. and T. Hastie (2005) “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society. Series B*, Vol. 67, No. 2, pp. 301–320, DOI: <https://www.jstor.org/stable/3647580>.
- 株式会社日本格付研究所 (2011) 『JCR 大企業モデル（デフォルト率推定モデル）の概要』, <https://www.jcr.co.jp/pdf/dm25/report/20110421.pdf>, Accessed 20 September 2021.
- 田中克弘・山本零 (2022) 『銘柄数制約付き決定係数最大化ポートフォリオ構築問題の効率的解法』, ワーキングペーパー.
- 木島正明・小守林克哉 (1999) 『信用リスク評価の数理モデル』, 朝倉書店.
- 後藤順哉・山本零 (2012) 「CVaR 最小化と信用リスク判別モデル」, 『MTEC ジャーナル』, 第 24 巻, 29–48 頁.
- 佐野紳也 (1990) 『質的選択分析—理論と応用』, 三菱経済研究所.
- 廣瀬慧 (2016) 「スパースモデリングとモデル選択」, 『電子情報通信学会誌』, 第 99 巻, 第 5 号, 392–399 頁.
- 山下智志・三浦翔 (2011) 『信用リスクモデルの予測精度-AR 値と評価指標-』, 朝倉書店.

謝辞

本研究は、著者が慶應義塾大学大学院理工学研究科後期博士課程在学中に、同大学理工学部山本零教授の指導のもとに行ったものである。また副査を御担当いただいた同大学理工学部枇々木規雄教授および今井潤一教授、また一橋大学大学院経営管理研究科中川秀敏教授には、非常に多くの有益な助言をいただき、本論文を完成に向かって進めることができた。改めて、この場を借りて、御礼申し上げます。

付録 A 4章で使用されている説明変数の一覧

表A.1 説明変数の一覧

1	営業利益 / 売上高	27	売上高 / 買入債務
2	経常利益 / 売上高	28	売上高 / 運転資本
3	税引後当期利益 / 売上高	29	売上高 / 流動資産
4	売上総利益 / 売上高	30	売上高 / 固定資産
5	販管費 / 売上高	31	営業外収益 / 売上高
6	経常利益 / 純資産	32	営業外費用 / 売上高
7	当座資産 / 流動負債	33	売上高 / 社員数
8	流動資産 / 流動負債	34	経常利益 / 社員数
9	自己資本 / 総資産	35	有形固定資産 / 社員数
10	留保利益 / 総資産	36	使用総資本
11	総借入 / 総資産	37	総資産
12	固定資産 / 純資産	38	売上高水準
13	現金等価物 / 売上高	39	税引後当期利益水準
14	売上高 / 総負債	40	フリーキャッシュフロー
15	売上高 / 流動負債	41	営業活動キャッシュフロー / 売上高
16	売上高 / 固定負債	42	財務活動キャッシュフロー / 売上高
17	経常収入 / 経常支出	43	営業活動キャッシュフロー / 総資産
18	支払利息 / 有利子負債	44	財務活動キャッシュフロー / 総資産
19	支払利息 / 売上高	45	フリーキャッシュフロー / 売上高
20	運転資本 / 総資産	46	フリーキャッシュフロー / 総資産
21	利払および税引前利益 / 支払利息	47	営業活動キャッシュフロー / 流動負債
22	有利子負債 / 営業キャッシュフロー	48	フリーキャッシュフロー / 流動負債
23	売上高 / 総資産	49	営業活動キャッシュフロー / 総負債
24	売上高 / 自己資本	50	フリーキャッシュフロー / 総負債
25	売上高 / 棚卸資産	51	売上高の増加額 / 売上高
26	売上高 / 売上債権		

付録 B HA+ 内の Step2 の詳細

HA+ の Step2 の具体的な計算手順は次の通り:

Step 1 $k \leftarrow 0$. $\vartheta^k := 1.0e^{-12}$ 、 $\dot{B}^k := \hat{B}$ とする.

Step 2 次のとおり \dot{B}^{k+1} を計算する:

$$\dot{B}_{p_1, p_2}^{k+1} := \begin{cases} 0, & \text{if } |\dot{B}_{p_1, p_2}^k| \leq \vartheta^k \\ \dot{B}_{p_1, p_2}^k, & \text{otherwise} \end{cases} \quad (p_1, p_2 \in \mathcal{P}). \quad (2)$$

Step 3 \dot{B}^{k+1} の最小固有値 (Λ^{k+1}) を計算する。もし $\Lambda^{k+1} \leq -\varepsilon$ であれば、 $\vartheta := \vartheta^k$ とし、終了する。そうならなければ、 $\vartheta^{k+1} := \vartheta^k \times 10$ 、 $k \leftarrow k + 1$ とし Step2 に進む。

ここで $\varepsilon (> 0)$ は B&C でも同じく使用されているパラメーターである。

付録 C 5章で使用されている説明変数の一覧

表C.2 説明変数の一覧

1	営業利益 / 売上高	26	売上高 / 売掛金
2	経常利益 / 売上高	27	売上高 / 買入債務
3	税引後当期利益 / 売上高	28	売上高 / 運転資本
4	売上総利益 / 売上高	29	売上高 / 流動資産
5	営業費用 / 売上高	30	売上高 / 固定資産
6	経常利益 / 純資産	31	営業外収益 / 売上高
7	当座資産 / 流動負債	32	営業外費用 / 売上高
8	流動資産 / 流動負債	33	売上高 / 社員数
9	純資産 / 総資産	34	経常利益 / 社員数
10	税引後当期利益 / 総資産	35	有形固定資産 / 社員数
11	総借入 / 総資産	36	現金等価物 / 総資産
12	固定資産 / 純資産	37	買掛金の増加額 / 買掛金
13	現金等価物 / 売上高	38	売掛金の減少額 / 売掛金
14	売上高 / 総負債	39	当座資産 / 棚卸資産
15	売上高 / 流動負債	40	長期借入金 / 売上高
16	売上高 / 固定負債	41	営業活動キャッシュフロー / 売上高
17	経常収入 / 経常支出	42	財務活動キャッシュフロー / 売上高
18	支払利息 / 有利子負債	43	営業活動キャッシュフロー / 総資産
19	支払利息 / 売上高	44	財務活動キャッシュフロー / 総資産
20	運転資本 / 総資産	45	フリーキャッシュフロー / 売上高
21	利払および税引前利益 / 支払利息	46	フリーキャッシュフロー / 総資産
22	有利子負債 / 営業活動キャッシュフロー	47	営業活動キャッシュフロー / 流動負債
23	売上高 / 総資産	48	フリーキャッシュフロー / 流動負債
24	売上高 / 純資産	49	営業活動キャッシュフロー / 総負債
25	売上高 / 棚卸資産	50	フリーキャッシュフロー / 総負債