



アダプティブデザインを計画するための シミュレーションの検討

日本製薬工業協会

データサイエンス部会

2023 年度 継続タスクフォース 1

Ver 1.0

2024 年 4 月

目次	
1はじめに	3
2シミュレーション報告書の概要.....	5
2.1 アダプティブデザインの検討手順	5
2.2 シミュレーション報告書の内容	9
2.3 シミュレーションの評価指標	11
2.3.1 アダプティブデザイン共通の評価指標	11
2.3.2 症例数再推定または早期中止を伴うアダプティブデザイン.....	14
2.3.3 治療群の選択を伴うアダプティブデザイン	15
3シミュレーション報告書の例：症例数再推定および早期中止を伴うアダプティブデザイン	
16	
4シミュレーション報告書の例：治療群の選択を伴うアダプティブデザイン	33
5おわりに	64
参考文献.....	66
6 APPENDIX A: 症例数再推定および早期中止を伴うアダプティブデザインのプログラムコード.	68
7 APPENDIX B: 治療群の選択を伴うアダプティブデザインのプログラムコード.	80
7.1 治療群の選択を伴うアダプティブデザインのプログラムコード.....	80
7.2 ADAPTIVE DUNNETT のプログラムコード	107

1 はじめに

本成果物は、医薬品の承認申請を目的とする検証的試験としてアダプティブデザインを計画する際に、その動作特性を評価するためのシミュレーションの実施方法とその評価結果の記載内容についてまとめたものである。

アダプティブデザインに関するガイダンスが European Medicine Agency (EMA)^[1]、Food and Drug Administration (FDA)^{[2][3]}により公表されており、さらに現在 ICH-E20 の検討が進められている。FDA ガイダンスは、アダプティブデザインを「臨床試験に参加した被験者の蓄積されたデータに基づいて、試験デザインの1つ以上の側面について、予め計画された変更を行うことができる臨床試験デザイン」と定義している。アダプティブデザインにより与えられる柔軟性により、試験の参加者がより良い治療を受けられる機会が増え、より効率的な医薬品開発、さらには利用可能なリソースの活用といった利点が期待される^[4]。その一方、統計手法を適切に用いなければ、第一種の過誤確率の増大、点推定値へのバイアスの発生、信頼区間の被覆確率が名義上の信頼係数より低下するなど、統計的妥当性の観点から望ましくない現象が起こる可能性がある。また、FDA は近年、

「Interacting with the FDA on Complex Innovative Trial Designs for Drugs and Biological Products」というガイダンス^[5]を公表し、革新的な試験デザイン (CID) の取り組みを進めている。FDA は、CID に固定の定義は存在しないが、この試験デザインの事例の一つとしてアダプティブデザインを挙げている。アダプティブデザインは従来の試験デザインと比べて複雑であり、伴って試験の動作特性も複雑である。CID に関する FDA ガイダンスにて「多くの CID に共通する一つの特徴は、試験の動作特性を推測するためには数式よりもシミュレーションが必要となることである」と述べられており、規制当局とアダプティブデザインの合意を得る上でシミュレーションにより動作特性を明示することは重要であると認識できる。

実際にアダプティブデザインを立案する際には、計画段階でデザインの動作特性を評価することが重要であり、ある仮想的な状況を想定したシミュレーションを通じて動作特性を評価することになる。そのシミュレーション結果をもとに、アダプティブデザインの適切性を十分に説明する場面が予想される。アダプティブデザインを計画する場合は、その比較対象（例えば、試験開始時の症例数で固定する固定デザイン）を含むいくつかのデザインが候補として考えられるだろう。アダプティブデザインのシミュレーション結果は、臨床試験の計画に携わる臨床メンバー、規制当局とのディスカッション、外部へ情報発信するための論文化や、さらにはその論文に対する医師等からの問い合わせなど、様々な場面で用いられることが想定される。そうした場면을想定すると、実施したシミュレーションの結果を適切に要約して報告書としてまとめておき、そのデザインの特性を広く周知させることも重要となる。

本成果物は、アダプティブデザインを立案する場合に試験計画の動作特性を評価するためのシミュレーションについて、その方法と手順、さらにはシミュレーションで評価すべき指標とその結果のまとめ方について、主に Mayer et al. (2019)^[6] の論文及び FDA ガイダンスを参考にし、本タスクフォースにて議論した内容をまとめた。具体的には、シミュレーション実施後に、その結果をシミュレーション報告書としてまとめる際の記載内容や記載すべき項目などを、症例数の変更を伴うアダプティブデザイン（以下、症例数再推定と呼ぶ）と、複数の試験治療を中間解析の結果から治療群を絞り込んで検証する治療群の選択（いわゆる、シームレス第 2/3 相試験）のアダプティブデザインに注目している。また、これら 2 つのアダプティブデザインがそれぞれ実際に利用された臨床試験を事例として取りあげ、その試験実施計画書や審査報告書などから読み取れる情報をもとに、本タスクフォースでシミュレーション報告書を作成した。

本成果物は、以下のように構成されている。2 章は、主に Mayer et al. (2019) の論文及び FDA ガイダンスを参考にし、医薬品開発の過程におけるアダプティブデザインの検討手順やその具体的な流れを例示し、その結果としてアダプティブデザインの動作特性をシミュレーションにより評価することになった際に検討すべき指標やその報告書に記載すべき内容について、症例数再推定と治療群の選択のアダプティブデザインに注目して記載している。3 章は、症例数再推定および早期中止を伴うアダプティブデザインの事例として COV-BARRIER 試験^[7]を取り上げ、実際にアダプティブデザインの動作特性を評価したシミュレーション報告書を作成した。4 章は、治療群の選択を伴うアダプティブデザインとして INHANCE 試験^[8]を事例に挙げ、3 章と同様に実際にアダプティブデザインの動作特性を評価したシミュレーション報告書を作成した。3 章と 4 章それぞれの末尾に、シミュレーション報告書とは別に検討したデザインの設定及び検定手法の比較について、統計的観点の興味から実施しシミュレーションで評価した内容についても記載した。また、3 章と 4 章の参考文献のいくつかについては、本成果物の参考文献と共通して利用している。

本成果物の読者として臨床試験の計画や実施に携わる生物統計担当者、臨床試験を計画する臨床メンバー、規制関連業務を担う薬事メンバーを想定している。アダプティブ臨床試験を計画し、規制当局との議論を円滑に進めるためには生物統計担当者以外のメンバーの理解が必要不可欠であるため、特に 2 章については、臨床試験を計画する臨床メンバーと、規制関連業務を担う薬事メンバーも参照されたい。日本製薬工業協会医薬品評価委員会データサイエンス部会は、FDA から公表されているアダプティブデザインに関するガイダンスの邦訳^[6]と、アダプティブデザインの基本的な統計的推測法に関してまとめた「アダプティブデザインの統計的推測に関する検討」^[9]を作成しており、本成果物と併せて確認いただきたい。本成果物が、アダプティブデザインの理解と適切な試験実施の一助となれば幸いである。

2 シミュレーション報告書の概要

本章では、アダプティブデザインを計画するにあたり、試験実施の企画段階から試験デザインの確定に至るまで、アダプティブデザインを検討する作業手順及びシミュレーション報告書に記載すべき内容について紹介する。

本章は、Mayer et al. (2019)^[6]の論文及びFDA ガイダンス^{[2][3]}を基本として、本タスクフォースにて議論された内容を一部に含めている。そのため、二つの文献には記載されていない内容が一部含まれていることに注意されたい。

2.1 アダプティブデザインの検討手順

Mayer et al. (2019)^[6]は、アダプティブデザインに関するシミュレーションを実行するための作業手順を表（論文中の Table 1）にまとめている。本タスクフォースは、Mayer et al. (2019)がまとめている内容を包含する形で、アダプティブデザインを含む試験デザインを検討する際の一連の作業手順を表 2-1 にまとめた。

表 2-1 アダプティブデザインを計画する手順

ステップ	作業内容	関連担当者
1. 実施可能性を見積もる	<ul style="list-style-type: none">● 試験目的、対象患者、治療法（用法・用量）、評価項目、対照群など、試験計画の概要を決める。● 潜在的患者数、試験期間、地域、合理的な期間で登録できる被験者数、被験者の登録スピードなど、試験規模の概算を見積もる。● 過去に実施された試験の事例を参考に加えて、試験が現実的に実施可能な程度を検討する。● 必要に応じて規制当局と相談する。	Clinical、Stat、責任医師、RA.
2. 動作特性を比較する試験デザインの候補を決める	<ul style="list-style-type: none">● アダプティブデザインを検討する背景と目的を明らかにする。● 中間解析の時期と回数、アダプテーションの決定規則など、アダプテーションの詳細を検討する。● アダプティブデザインを実行するためのロジスティクス、例えば、中間解析に必要な	Clinical、Stat、責任医師、RA. アダプテーションの運用手順について

ステップ	作業内容	関連担当者
	<p>データのクリーニング手順、独立データモニタリング委員会の運用手順を検討する。</p> <ul style="list-style-type: none"> ● 試験に適用するアダプティブデザインの利点と欠点を整理する。 ● 必要に応じて、回数を少なくするか簡易な方法によるシミュレーションを実行し、動作特性を比較する試験デザインの候補を絞り込む。候補には、アダプティブデザインだけでなく固定デザインも含める。 ● アダプティブデザインを試験に適用しない場合は検討を終了し、固定デザインに基づく試験計画を立案する。 	<p>は、DM も検討に加わる。</p>
<p>3. 動作特性を測るために、シミュレーションを実行する</p>	<ul style="list-style-type: none"> ● シミュレーションにて仮定する治療効果や分散などのパラメータを設定する。 ● 試験デザインの動作特性を測るためのシミュレーションの評価指標を設定する。 ● 主要解析と異なる仮定にて、試験デザインの動作特性がどの程度頑健であるかを確認するために、必要に応じて感度シナリオを設定する。 ● 候補となる試験デザインそれぞれのアダプテーションの決定規則を反映させたシミュレーションを実行する計算コードを作成する。 ● 設定したシナリオに基づいて、動作特性を十分な精度の繰り返し回数でシミュレーションを実行する。 ● シミュレーションが再現可能なように計算環境を特定し、計算コード、シミュレーション結果などを保管する。 	<p>Stat. シナリオの設定には、 Clinical も検討に加わる。</p>
<p>4. 適用する試験デザインを決定する</p>	<ul style="list-style-type: none"> ● シミュレーションの結果を表やグラフを用いて要約する。 ● 第一種の過誤確率やバイアスなど、統計的妥当性に関する動作特性を確認する。 	<p>Clinical、 Stat. 適用する試験デザインの議論には、責任</p>

ステップ	作業内容	関連担当者
	<ul style="list-style-type: none"> ● 検出力、症例数、試験期間、アダプテーションに関する評価指標など、試験実施に関わる動作特性を確認する。 ● シミュレーション結果に基づく動作特性を総合的に考察し、推奨する試験デザインを決定する。 ● 検討した試験デザイン、シミュレーションの方法、結果、結論などをまとめた報告書を作成する。 	医師、RA も 検討に加わる。

Clinical：臨床担当部門、Stat：生物統計担当部門、DM：データマネジメント担当部門、RA：薬事担当部門

第1ステップは、試験を実施することがまだ確定しておらず、まずは実施できるかどうかを検討する段階である。試験目的、対象患者、治療法（用法・用量）、評価項目、対照群など、試験計画の概要を決め、過去に実施された試験の事例を参考にしつつ、治療効果を検証するために必要な試験の規模を見積もる。この第1ステップは、アダプティブデザイン特有ではなく、固定デザインの検討でも一般的に発生する作業である。このステップでアダプティブデザインを検討することは必須ではなく、一般的な固定デザインを想定して試験規模を見積もることでも十分と考えられる。試験の実施がある程度可能であると判断される場合には、第2ステップに進む。

第2ステップでは、最初に、アダプティブデザインを検討する背景と目的を明らかにしておくことよい。規制当局にアダプティブデザインの相談をする際には、その背景と目的を十分に説明することが求められると考えられる。次に、中間解析の時期や回数、アダプテーションの決定規則など、様々な面からアダプティブデザインを検討し、アダプティブデザインを適用することの是非を議論する。例えばSSRを検討する場合、中間解析の時期と回数だけでなく、SSRの方法（中間解析で参照するパラメータ、症例数を再推定する計算方法、登録する症例数の上限・下限など）、無益中止や有効中止を設定するかどうかなどを検討する。これらの設定の組み合わせは数多くあるため、試験デザインの候補を絞り込む作業が必要となる。必要に応じて、検討に耐えられる程度の少ない繰り返し回数（例えば、1000回程度）や計算パッケージを用いる簡易なシミュレーションを実行し、より適切な試験デザインに絞りこむ。ここでのシミュレーションの評価指標は、症例数、試験期間、検出力などが考えられる。シミュレーションを実行する試験デザインの数が多くなると、動作特性を測る第3ステップにて膨大な計算時間が必要になるため、ある程度の候補数に絞り込むことで、以降の作業が効果的に進むと考えられる。また、このステップでのシミュレーションは試験デザインの見当をつけることが目的であり、試験デザインの動作

特性を確認する目的のシミュレーションは、次の第3ステップにて実行される。さらに、試験のインテグリティを維持するためのロジスティクス、例えば、中間解析に必要なデータのクリーニング手順、独立データモニタリング委員会の運用など、アダプティブデザインに関連して必要となるオペレーションを臨床やデータマネジメント担当と検討する。この時点で実施体制を確定して詳細な手順書を作成することは必須でないが、より複雑になるオペレーションの面も、ある程度検討しておくことが推奨される。第2ステップでの検討作業は一方向に進むとは限らず、アダプテーションの方法だけでなく、運用面など様々な角度から繰り返し検討される。以上の様々な観点から、アダプティブデザインを試験に適用するかどうか、適用する場合にはどのようなデザインが良いかを検討する。第3ステップに進む試験デザインの候補を絞り込み、その中にアダプティブデザインが含まれる場合には、次の第3ステップに進む。候補にアダプティブデザインが残らない場合は検討作業を中断し、固定デザインに基づく試験計画を立案する。

第3ステップでは、候補の中から適用する試験デザインを決定するために、試験デザインの動作特性をシミュレーションにて測定する。まず、シミュレーションにインプットする治療効果などのパラメータのと、シミュレーションによりアウトプットする動作特性を評価するための指標を設定する。計画する主要解析に設定している仮定が実際と異なる場合に、アダプティブデザインの動作特性が変化する可能性がある。必要に応じて、主要解析と異なる仮定に基づく感度シナリオを設定する。アダプテーションの決定規則を反映させてシミュレーションを実行する計算コードを作成する。コードの作成後、設定したシナリオに基づいてシミュレーションを実行する。このステップでのシミュレーションは、試験デザインの設定根拠を得るために動作特性を確認し、適用する試験デザインを決定することであるため、それらが十分な精度のシミュレーション回数にて実行する（詳細は次節を参照）。シミュレーションの結果は、再現可能であることが望ましい。シミュレーションを実行した計算環境を特定し、計算コードやシミュレーション結果などを保管しておくことで、後に規制当局と試験デザインを議論する際に、共通の方法と環境でシミュレーションを追加実行できる。

第4ステップでは、第3ステップにて実行したシミュレーションの結果から試験デザインの動作特性を評価し、適用する試験デザインを決定する。試験デザインの第一種の過誤確率やバイアスなど統計的側面の妥当性だけでなく、検出力、症例数、試験期間など、アダプテーションに関わる動作特性を参照し、候補である試験デザイン間で比較する。様々な観点から比較した結果を臨床担当部門のメンバーと議論し、試験デザインを決定する。もしシミュレーション結果に望ましい結果が得られず、アダプティブデザインの計画を見直す場合には、第2ステップに戻り、試験デザインを再検討する。最後に、検討した試験デザイン、シミュレーションの方法と結果、結論としてどのような試験デザインに決まったかなどをまとめた、シミュレーション報告書を作成する。

以上より、アダプティブデザインを選択肢に加えて試験デザインを検討すると、複数の作業ステップを重ねるため、固定デザインを計画する場合と比較してより多くの時間を必要とすることが分かる。シミュレーションに関わる一連の作業である、計算コードの作成、シミュレーションの実行、結果の要約には、多くの時間を要することがある。例えば、積分計算を含む特別な解析方法をシミュレーションに反映させると膨大な計算時間を要するため、ある程度の作業期間を見込んだ検討が望ましい。

2.2 シミュレーション報告書の内容

本節では、シミュレーション報告書に記載すべき具体的な内容を紹介する。シミュレーション報告書は、試験に適用するデザインの動作特性を説明するための文書であり、最終的に決定した試験デザインは治験実施計画書に記述される。治験実施計画書に試験デザインの動作特性を説明する場合に、必要に応じて、シミュレーション報告書の内容が治験実施計画書に反映される。

現在、FDA にて進められている **Complex Innovative Trial Design Paired Meeting Program** (<https://www.fda.gov/drugs/development-resources/complex-innovative-trial-design-meeting-program>) では、アダプティブデザインを含む複雑な試験デザインの動作特性を確認するために、シミュレーションの詳細を提出することが求められている。シミュレーションを適切に実施し、再現可能なシミュレーションコードを準備しておくことは、アダプティブデザインの計画を規制当局と議論するために重要な作業であると考えられる。

シミュレーション報告書の具体的な章立ておよびその内容について下記に記載する。

1 緒言

1.1 本治験の目的

治験の目的及び試験デザインの主要な要素（例：開発の相、対象集団、試験の期間、試験群、対照群、評価項目など）を記述する。

1.2 シミュレーションの目的

シミュレーションを実施する目的を記述する。アダプティブ臨床試験におけるシミュレーションの主な用途は、試験の動作特性を推定し、動作特性が目標とする水準を満たしていることを確認することである。

2 シミュレーション入力設定

2.1 試験デザイン案

候補とするアダプティブデザインと、比較するためのベンチマークとしての試験デザインの概要を記述する。アダプティブデザインについては、アダプティブデザインを検討する背景や動機を合わせて述べる。ベンチマークの試験デザインについては、例えば、中間解析を伴わない固定デザインが候補となる。他には、効果が想定よりも小さい場合に症例数を増やすアダプティブデザインを計画する場合に、症例

数再推定のアダプティブデザインと、小さい効果を仮定した群逐次デザインがベンチマークになり得る。

治療効果の群間比較をする検定方法、点推定や区間推定の方法を記述する。未調整の方法だけでなく、特別な統計手法も検討することができる。

2.2 中間解析及び判定基準

中間解析の回数及び時期、並びに中間解析時に適用する規則（例：中止規則などアダプテーションに関する決定規則）を説明する。

2.3 仮想データ生成モデル

評価項目（例：変数の種類とその分布、単一時点や時系列データの取り扱い、関連する共変量）の乱数データを生成する方法を説明する。必要に応じて、脱落パターンや欠測データが発生するメカニズムをデータ生成モデルに含める。ここでは基本的に単一のデータ生成モデルを仮定する。異なる仮定に基づく複数のデータ生成モデルを検討する場合は、シミュレーション報告書 2.5 節にて述べる。

2.4 治療効果の仮定

治療効果に仮定するデータ生成モデル（例：効果の群間差と SD）を記述する。仮定が不確実な場合には、臨床的背景を踏まえて適切なグリッドで範囲を設定する。

2.5 その他の臨床上の仮定

シミュレーションに設定した、治療効果以外の仮定、被験者の登録スピード、脱落率、欠測の発生など、パラメータ設定値が妥当と考える根拠を含め記載する。一つのパラメータに、複数の異なる仮定を設定することもある。シミュレーション報告書 2.3 節と 2.4 節にて設定した何らかの仮定（例：time-to-event 型の主要評価項目の比例ハザード性）など、設定した特定の仮定からの逸脱や極端な仮定に基づくシナリオを検討する。

2.6 評価指標

複数の試験デザイン間で臨床試験の動作特性を比較するための評価指標を記述する。例えば、第一種の過誤確率、検出力、試験の中止確率、1 群あたりの症例数の要約（平均値、最小値、最大値など）、治療効果の推定値のバイアス、平均二乗誤差（MSE）、信頼区間の被覆確率などがある。詳細は、本成果物 2.3 節にて説明する。

2.7 シミュレーションの繰り返し回数と乱数シード

各シナリオについて実行したシミュレーションの繰り返し回数と、その根拠を説明する。例えば、一般的には繰り返し回数を 10 万回とすると 95%信頼区間の幅が約±0.1%になることが保証され、第一種の過誤確率についてはほとんどの場合に十分な回数と考えられる（FDA ガイダンス参照^{[2][3]}）。シミュレーションの結果が乱数のシードに依存することを避けるために、シミュレーションのシナリオごとに異なる乱数シードを用いる。乱数の発生方法には、十分な周期を持つアルゴリズム（例えば、メルセンヌ・ツイスター）を用いる。

3 シミュレーション結果

異なるシナリオによって試験デザイン間の動作特性を比較しやすいように、シミュレーションの結果を表または図を用いて要約する。特に、アダプティブデザインとベンチマークの試験デザインとの比較を強調する。

4 要約及び推奨事項

4.1 考察

シミュレーションの結果から、シミュレーション報告書 2.6 節に設定した評価指標それぞれの目的に応じて、候補とした試験デザインの動作特性を考察する。

4.2 推奨されるデザイン

シミュレーションを通して得られる考察を総合的に判断し、結論として推奨する試験デザインを述べる。

5 参考文献

アダプティブデザインや統計手法に関連する適切な参考文献を記述する。

6 付録

6.1 バリデーション

シミュレーションを実行した計算環境（ソフトウェア、コード）のバリデーション記録を記述する。シミュレーションの実行環境が商用ソフトウェアではなく自作である場合、コード作成者と別の統計担当者が確認する（いわゆる、ダブルプログラミング）か、コード作成者自身が別のソフトウェアで確認する方法が考えられる。

6.2 計算環境またはその他の技術的詳細

シミュレーションを実行した計算環境の詳細（例えば、ソフトウェア、バージョン番号）を記述する。コードを自作した場合、コードを記述するか、その保管場所を記述する。商用ソフトウェアの場合、用いたモジュールの入力設定などを記述する。必要に応じて、証明などの数学的な技術的詳細を記述する。

2.3 シミュレーションの評価指標

本節では、試験デザインの動作特性を測るためにシミュレーションにて出力する評価指標をまとめる。本成果物 2.3.1 項ではアダプティブデザイン共通の評価指標を述べ、2.3.2 項と 2.3.3 項それぞれに、症例数再推定または早期中止のアダプティブデザイン、治療群の選択のアダプティブデザインに分けて、評価指標を述べる。検討する試験デザインの特性に応じて、シミュレーションにて検討する評価指標を決定する。

2.3.1 アダプティブデザイン共通の評価指標

試験デザインの統計的妥当性を確認するための評価指標として、第一種の過誤確率、点推定値のバイアスと MSE、信頼区間の被覆確率がある。

- ・ 第一種の過誤確率

帰無仮説（通常、効果の群間差の真値がゼロという仮定）のもとで、臨床試験を誤って肯定的に判断する（通常、効果に差がある）確率を計算する。臨床または数学的考察の面から、考慮すべき帰無仮説のシナリオを特定して、第一種の過誤確率を推定する。効果の適当な値のグリッド上でシミュレーションを実行し、シミュレーション結果から総合的に判断して、検討した範囲にて第一種の過誤確率の最大値が目標とする有意水準を超えていないことを確認する。

複数の評価項目を検定の対象としてファミリーワイズの第一種の過誤確率を制御したい場合、帰無仮説下のシナリオにおけるすべての評価項目のシミュレーションを実行する必要がある。このためには複数の評価項目の相関構造の仮定が必要となるが、複数の評価項目間は独立であると仮定することで第一種の過誤確率の上限を得ることができるとする。

- ・ 治療効果の推定値のバイアスと MSE、信頼区間の被覆確率

各群または試験群と対照群との評価項目の群間差について、点推定値のバイアスと MSE、信頼区間が真値を含む被覆確率を計算する。臨床試験の結果が誤った結論を導く可能性を制御するために、バイアスや MSE を確認しそれらが臨床的な観点からも受け入れられる程度に抑えられていること、被覆確率が信頼水準に近く制御されていることは重要である。アダプティブデザインに対応する特別な推定方法を用いることで未調整の推定方法よりもバイアスを減らすことができる場合があるが、一方で MSE が増加することがある。そのため、バイアスが最も小さい推定方法を選ぶことが最善とは限らない。推定方法の決定にはバイアスと MSE それぞれの程度を考慮することが重要であり、場合によっては未調整の方法が選択されることも考えられる。バイアスが抑えられている程度についてコンセンサスの得られている基準はないが、効果に関する何らかのマージンを設けて、そのマージン内に収まっていることを確認する方法が考えられる。シミュレーションにて得られたこれらの指標の結果に課題がある場合には、望ましい性質が得られるようにアダプテーションの決定規則や中間解析の方法を修正するなど、別の手段を検討する。

また、臨床試験の実施可能性という観点からも、様々な指標が考えられる。検出力、症例数、試験期間など、試験実施に関わる動作特性を確認する。

- ・ 検出力

対立仮説（通常、効果に差がある）のもとで、臨床試験を正しく肯定的に判断する確率を計算する。複数の仮説を設定する場合、検出力の定義（例えば、少なくとも一つの仮説が有意となる確率、全ての仮説が有意となる確率など）を記載する。効果がある仮定のもとで目標の検出力（例えば 90%）が達成されているか、アダプテーションの目的

が特定のシナリオ下での検出力の低下を防ぐことである場合に目的通りにアダプテーションが機能しているか、などを確認する。

- ・ 症例数

中間解析の結果に基づくアダプテーションによって症例数が変動することがある。例えば、症例数再推定を伴うアダプティブデザインの場合、試験に登録される被験者数は中間解析の結果に依存する。そのため、症例数は何らかの広がりを持つ分布を形成する。症例数の分布の平均値、最小値、最大値などの要約、あるいはヒストグラムを示して、症例数の規模が想定範囲内に収まっていることを確認する。

早期中止のアダプティブデザインでは、中間解析の実施中も被験者登録を継続する場合に、中間解析で試験を中止すると登録症例数が超過する可能性がある。アダプテーションによって被験者数の超過が懸念される場合には、その程度をシミュレーションにて確認しておくことができる。

- ・ 試験期間

症例数と同様、中間解析の結果に基づくアダプテーションによって試験期間が変動することがある。例えば、症例数再推定を伴うアダプティブデザインの場合、アダプテーションの結果によって被験者の登録期間が変動し、伴って試験期間が変動する。主要評価項目が **time-to-event** である場合は主要解析の時期がイベント数に依存することが多く試験期間はより複雑になる。試験期間には、被験者の登録期間、中間解析の実施期間、中間解析の完了から被験者登録を再開する期間など、様々な期間に分けられる。適用するアダプティブデザインによって、どのような期間が変動しうるのかを考慮し、シミュレーションに反映させる。試験期間の分布の平均値、最小値、最大値などの要約統計量、あるいはヒストグラムを示して、試験期間が想定範囲内に収まっていることを確認する。試験期間を確認する目的の一つに、承認申請までのタイムラインが確保できているかを確認するためという点がある。試験期間のシミュレーションは、その目的から、規制当局との議論の対象とならない可能性があるが、合理的な期間内で臨床試験が実施できることを確認しておくことが重要な場合には、シミュレーションの評価指標に含めることが推奨される。

別の観点として、アダプテーションを実行した結果が治験薬製造や試験コストに影響する程度を事前に見積もっておきたい場合が考えられる。例えば治療群の選択のアダプティブデザインの場合は、選択された群によって必要な治験薬製造の種類と量が影響するかも知れない。その他の評価指標として、上市までの時間、試験実施にかかる金銭的成本などが考えられる。試験目的を鑑みて様々な観点から評価することが重要となる。

2.3.2 症例数再推定または早期中止を伴うアダプティブデザイン

症例数再推定または早期中止を伴うアダプティブデザインを計画する場合に、シミュレーションにて評価する指標をまとめた。

表 2-2 症例数再推定または早期中止を伴うアダプティブデザインのシミュレーション
評価指標

指標	内容
1. 第一種の過誤確率 (GSD、SSR)	第一種の過誤確率の最大値が有意水準以下に制御できていることを確認する。
2. 検出力 (GSD、SSR)	目標とする検出力が一定の基準以上であることを確認する。
3. 試験の中止割合 (GSD)	いずれかのステージ、または各ステージでの試験を中止する割合を計算する。アダプテーションに応じて、無益中止または有効中止割合の程度を評価する。 無益中止割合：帰無仮説のもとで正しく試験を中止する割合、対立仮説のもとで誤って試験を中止する割合。 有効中止割合：帰無仮説のもとで誤って試験を中止する割合、対立仮説のもとで正しく試験を中止する割合。
4. 症例数の分布 (GSD、SSR)	症例数の分布の要約統計量やヒストグラムを用いて、症例数の規模が想定範囲内に収まっていることを確認する。例えば、帰無仮説のもとでの無益中止または対立仮説のもとでの有効中止や症例数再設計による症例追加によって、症例数が増減していることを確認する。
5. 最小又は最大の症例数を達成した回数の割合、症例数が特定の水準よりも上回る割合 (SSR)	症例数が、試験デザインに設定している最小または最大に達する割合を計算する。 症例数が、特定の水準（例えば、ベンチマークの試験デザインにて設定する症例数）よりも上回る割合を計算する。
6. 中止決定時及び最終解析時の治療効果の推定 (GSD、SSR)	点推定の平均値、バイアス、MSE、信頼区間の被覆確率を計算する。

GDS：早期中止を伴うアダプティブデザイン、SSR：症例数再推定を伴うアダプティブデザイン

2.3.3 治療群の選択を伴うアダプティブデザイン

治療群の選択を伴うアダプティブデザインを計画する場合に、シミュレーションにて評価する指標をまとめた。早期中止を伴うアダプティブデザインを検討に含める場合には、表 2-2 も参照されたい。

表 2-3 治療群の選択を伴うアダプティブデザインのシミュレーション評価指標

指標	目 容
1. 第一種の過誤確率	第一種の過誤確率を評価することが目的のシナリオ（つまり、帰無仮説を含めるシナリオ）の全てにおいて、第一種の過誤確率（効果のない用量のいずれかが選択されかつプラセボ群との有意差が得られる確率）が名義水準以下になることを確認する。
2. 検出力	検出力の評価が目的のシナリオ（つまり、対立仮説を含めるシナリオ）において、目標の検出力を確認する。 <ul style="list-style-type: none"> ・ 試験治療群ごとの検出力 ・ 試験治療群全体の検出力（効果のあるいずれかの試験治療群が中間解析で選択され、かつ有意となる確率）
3. 治療群別と試験全体の症例数の分布	検出力の評価が目的のシナリオにおいて、各試験治療群および試験全体における症例数の分布を確認する。症例数の分布の要約統計量やヒストグラムを用いて、症例数の規模が想定範囲内に収まっていることを確認する。
4. 治療群別の選択割合、選択用量数の分布	検出力の評価が目的のシナリオにおいて、各試験治療群が選択される割合および選択される試験治療群の数の分布を確認する。設定した群選択ルールが期待通りに機能していることを確認する。選択される試験治療群の数が 0 である割合は、基準を満たす試験治療群が存在せず、中間解析にて試験を中止する無効中止確率に相当する。
5. 治療効果の推定	検出力の評価が目的シナリオにおいて、ステージごとおよび両ステージにおける、各群の点推定値の平均値、バイアス、MSE、信頼区間の被覆確率を評価する。 中間解析にて選択された場合および選択されなかった場合における、条件付き推定の評価も行う。

3 シミュレーション報告書の例：症例数再推定および早期中止を伴

うアダプティブデザイン

本章では、症例数再推定および早期中止を伴うアダプティブデザインを想定した、仮想的なシミュレーション報告書の例を示す。

COV-BARRIER 試験は、COVID-19 に感染した患者を対象としたバリシチニブの二重盲検プラセボ対照ランダム化比較第 III 相試験であり、非盲検下の症例数再推定が計画された。本タスクフォースは、COV-BARRIER 試験を計画する状況を想定したシミュレーション報告書を作成した。一部、シミュレーションを簡便に実行するために、実際とは異なる試験デザインの設定を仮定している。COV-BARRIER 試験の実際のデザインは、Marconi et al. (2021)^[7]を参照されたい。

1 緒言

1.1 本試験の目的

試験表題	COVID-19 に感染した患者を対象としたバリシチニブの二重盲検プラセボ対照ランダム化比較第 III 相試験
目的	COVID-19 感染患者の疾患進行に対するバリシチニブ 4 mg 1 日 1 回 (QD) の効果をプラセボと比較して評価すること
試験の期間	投与期間：14 日間 評価期間：28 日間 Follow-up 含む観察期間：60 日間
投与群	プラセボ群 (プラセボ+基礎療法) バリシチニブ群 (バリシチニブ 4mg QD+基礎療法)
割付比	1 : 1
主要評価項目	Day 28 までに死亡、又は非侵襲的換気/高流量酸素療法もしくは侵襲的機械的換気 [体外式膜型人工肺 (ECMO) を含む] を必要とした被験者の割合
解析方法	主要評価項目に対して、各群で頻度及び割合を算出し、オッズ比とその信頼区間を算出する。また、治療群間の比較には層別因子 (病期、年齢、地域、デキサメタゾン及び/又は全身性コルチコステロイドの使用の有無) で調整したロジスティック回帰モデルを用いて、両側有意水準 5% で検定を行う。

1.2 シミュレーションの目的

本試験ではデキサメタゾン又は全身性コルチコステロイドの併用の有無が治療

効果に影響を与える可能性がある。しかしながら、試験計画時にデキサメタゾン又は全身性コルチコステロイドを併用する患者の治療効果を利用可能なデータから正確に想定できないため、試験途中で観察される治療効果の大きさに応じて症例数を再推定するデザインとした。

デキサメタゾン又は全身性コルチコステロイドを併用した患者の治療効果の差（主要評価項目の、プラセボ群と実薬群のリスク差）を変化させ、患者の割合で重み付けて統合した治療効果を以下に示す。なお、デキサメタゾン又は全身性コルチコステロイドを併用した患者の割合を 60%と仮定した。

治療効果のリスク差		統合した治療効果
デキサメタゾン又は全身性コルチコステロイドを併用した患者	デキサメタゾン又は全身性コルチコステロイドを併用しなかった患者	
7.5 %	7.5 %	7.5 %
4 %	7.5 %	5.4 %

本シミュレーション報告書では、7.5%を期待する差、5.4%を臨床的意義のある最低の差として、いくつかのシナリオで症例数再推定を行った場合の動作特性をシミュレーションにより確認し、各シナリオによるシミュレーション結果が目標とする水準を満たすことを示すために実施した。

2 シミュレーション入力設定

2.1 試験デザイン案

本シミュレーションでは、非盲検下の症例数再推定を伴う群逐次デザインについて検討した。競合するデザインとして、症例数再推定を伴わない群逐次デザインについても検討した。いずれのデザインにおいても、中間解析は1回とし、有効中止と無効中止を行うこととした。なお、症例数再推定を伴わない群逐次デザインは症例数再推定を行わない場合に加えて、最初から臨床的意義のある最低差を想定し、検出力を 80%まで下げた場合をベンチマークとして設定した。

パラメータ	設定
試験開始時の症例数 ($N_{initial}$)	685 例 (各群) <設定根拠> 主要評価項目をプラセボ群 25%、実薬群 17.5% (リスク差 7.5%、オッズ比 0.636) とし、第一ステージにおける症例数の比を $t = 0.5$ とし、中間解析において有効中止および無効中止を設

パラメータ	設定
	定した場合に全体の検出力が 90%以上となるように被験者数を設定した。
最大症例数 (N_{max})	症例数再推定を伴う場合 1500 例 (各群) <設定根拠> 固定デザインにおいて臨床的意義のある最低の差を用いて算出した症例数を最大症例数とすることとした。つまり、主要評価項目をプラセボ群 25%、実薬群 19.6% (リスク差 5.4%、オッズ比 0.731) とし、第一ステージにおける症例数の比を $t = 0.5$ として、中間解析において有効中止および無効中止を設定した場合に全体の検出力 90%とした場合の必要症例数に基づいて設定する。 症例数再推定を伴わない場合は、以下の二つのシナリオを設定する。 ● 期待通りの治療効果で症例数再推定を行わないシナリオ：685 例 (各群) ● 臨床的意義の最低差で検出力を 80%まで下げた場合のシナリオ：1020 例 (各群)
第一種の過誤確率の調整方法	Lan-DeMets の α 消費関数 症例数再推定；Cui-Hung-Wang (CHW) の方法

症例数再推定では、Cui et al. (1999)^[10]と Mehta and Pocock (2011)^[11]で与えられた以下の式に基づいて最終解析での検定統計量を計算する。

$$z_{chw} = \frac{\sqrt{n_1}}{\sqrt{n_1 + n_2}} z_1 + \frac{\sqrt{n_2}}{\sqrt{n_1 + n_2}} z_2$$

本シミュレーションでは主要評価項目の群間差を検定する事とし、 z_1 は中間解析までに評価期間を終えた患者データのみを用いた検定統計量、 z_2 は中間解析後に評価期間を終えた患者データのみを用いた検定統計量とした。 n_1 は中間解析までに評価期間を終えた症例数とし、 n_2 は試験計画時に設定した中間解析後に評価期間を終えた症例数である。

2.2 中間解析及び判定基準

検討するデザイン共通として、有効中止及び無効中止を目的とする中間解析を予定する。中止の判定基準を以下の通りとした。

パラメータ	設定
中間解析の回数及び目的	1回（有効中止及び無効中止）
中間解析の時期（ $n_1/N_{initial}$ ）	0.5
有効中止の基準	O'Brien-Fleming 法 ^[67] に基づく棄却限界値（ $z = 2.963$ ）
無効中止の基準	条件付き検出力（CP）が10%未満の場合に中止（ $z_1 = 0.751$ ）
症例数再推定の方法（症例数再推定を伴う場合のみ）	症例数を変更する条件； <ul style="list-style-type: none"> ● 無効中止基準以上（$CP \geq 10\%$）～$CP < 20\%$（Unfavorable zone）：症例数は変更しない ● $20\% \leq CP < 90\%$（Promising zone）：CPが90%となるように第2ステージの症例数を設定 ● $CP \geq 90\%$～有効中止基準以下（Favorable zone）：症例数は変更しない

CP は Mehta and Pocock (2011)で与えられた以下の式を用いて算出した。

$$CP(z_1, n_2) = 1 - \Phi\left(\frac{z_\alpha\sqrt{n_1 + n_2} - z_1\sqrt{n_1}}{\sqrt{n_2}} - \frac{z_1\sqrt{n_2}}{\sqrt{n_1}}\right)$$

$\Phi(\cdot)$ は標準正規分布における累積分布関数である。また、CP の計算では中間解析後のデータが中間解析時に得られた治療効果を持つ分布に従う事を想定している。

症例数を変更する条件において Promising zone に該当する場合、Mehta and Pocock (2011) で与えられた以下の式を用いて最終解析までの症例数 N_{new} を再計算した。

$$N_{new} = n_1 + \frac{n_1}{z_1^2} \left[\frac{z_\alpha\sqrt{n_1 + n_2} - z_1\sqrt{n_1}}{\sqrt{n_2}} + \Phi^{-1}(0.90) \right]^2$$

ここでは 90%の条件付き検出力を維持するよう症例数を再計算する事とした。なお、 N_{new} と N_{max} のいずれか小さい方を最終の症例数とする。

2.3 仮想データ生成モデル

主要評価項目に二項分布を適用した。全ての患者がデータを有する事を想定し、欠測値や脱落を考慮せずに生成した。ただし、実際の解析では脱落・欠測データメカニズムに missing at random を仮定し、利用可能なデータは全て使用する予定である。欠測があった場合には多重代入により欠測値の補完を行う予定である。

2.4 治療効果の仮定

主要評価項目に想定するプラセボ群の発現割合及び発現割合のオッズ比を以下の通りとした。

パラメータ	設定
プラセボ群の発現割合	25% <設定根拠> 外部試験データ（ACTT-2 試験 ^[13] ）に基づき、プラセボ群の発現割合は 25%と想定された。
発現割合のオッズ比	<ul style="list-style-type: none"> ● 帰無仮説下：1 ● 期待する効果：0.636（リスク差 7.5%） ● 臨床的意義のある最低の差：0.731（リスク差 5.4%） <設定根拠> プラセボ群の発現割合が 25%の場合に期待する発現割合の差（プラセボ群ーバリシニブ群）を 7.5%、臨床的意義のある最低差を 5.4%とし、それに対応するオッズ比（バリシニブ群のオッズ／プラセボ群のオッズ）を算出した。臨床的意義のある最低差とした 5.4%は、全身性コルチコステロイドの併用が治療効果の大きさを減少させる可能性に基づき設定している。

2.5 その他の臨床上の仮定

本シミュレーションでは被験者の登録スピードの影響や脱落は考慮しなかった。

2.6 評価指標

動作特性の評価のため、下記の指標について検討を行った。

指標	目容
1. 第一種の過誤確率（各解析時点及び全体）	帰無仮説下で有意となる確率が名義水準以下になる事を確認する。
2. 検出力（各解析時点及び全体）	期待する効果が得られたときに 90%以上の検出力を有する事を確認し、臨床的意義のある最低差となる治療効果で選択したデザインが固定デザインより検出力を増加させられる事を確認する。
3. 必要症例数の分布	症例数が増加した割合や最大症例数に到達した割合を確認し、症例数再推定が適度に実施されて

	<p>いる事を確認する。</p> <p>また、極端な治療効果の場合に必要な以上の症例数再推定が行われていない事を確認する。</p>
4. 治療効果の推定値、バイアス、MSE、信頼区間の被覆確率	<p>有効中止時点、無効中止時点、最終解析時点及び全体について、対数オッズ比のバイアスとMSEの程度を確認する。</p>

2.7 シミュレーションの繰り返し回数と乱数シード

各シナリオについて、繰り返し回数は100,000回とした。これにより、第一種の過誤確率の95%信頼区間の幅が約±0.1%になることが保証される。シミュレーションのシナリオごとに異なる乱数シードを用いた。

3 シミュレーション結果

3.1 シミュレーションの結果の概要

以下の候補デザインの動作特性を検討した。なお、全て候補デザインに有効中止及び無効中止を目的とする中間解析を設定している。

- ① 症例数再推定あり、期待する効果を仮定するデザイン
期待する効果（オッズ比 0.636）で検出力が90%になるように症例数を設定（各群 685）し、症例数再推定を実施する
- ② 症例数再推定なし、期待する効果を仮定するデザイン
期待する効果（オッズ比 0.636）で検出力が90%になるように症例数を設定（各群 685）し、症例数再推定を実施しない
- ③ 症例数再推定なし、臨床的意義のある最低の差（オッズ比 0.731）で検出力が80%になるように症例数を設定（各群 1020）し、症例数再推定を実施しない

帰無仮説下、すなわち、真のオッズ比 1（対数オッズ比 0）としたときのシミュレーション結果の概要を表 3-1 に示した。候補デザイン全てにおいて帰無仮説下では、第一種の過誤確率が 0.025 未満であり、第一種の過誤確率は制御されていた。

対立仮説下で、期待する効果である真のオッズ比 0.636（対数オッズ比 -0.452）としたとき及び臨床的意義のある最低差である真のオッズ比 0.731（対数オッズ比 -0.313）のシミュレーション結果の概要を表 3-2 に示した。期待する効果を想定した場合は、候補デザイン全てにおいて検出力が90%を超えた。最低差の効果を想定した場合のデザイン①の検出力は73.5%であり、デザイン②の63.1%とデザイン③の80.2%の間であった。

表 3-1 シミュレーション結果の概要 (帰無仮説下)

	真のオッズ比 1 (対数オッズ比 0)		
	デザイン①	デザイン②	デザイン③
有効中止 (全体: 検出力/第1種の過誤確率)	0.023	0.022	0.023
第1ステージ	0.002	0.002	0.002
第2ステージ	0.021	0.021	0.022
無効中止	0.977	0.978	0.977
第1ステージ	0.773	0.774	0.774
第2ステージ	0.204	0.203	0.203
症例数の平均値 (中央値)	515(343)	419(343)	624(510)
最小値-最大値	343-1500	343-685	510-1020
第1ステージで有効又は無効で中止した割合	0.775	0.776	0.776
計画された症例数で第2ステージ完了の到達割合	0.079	0.224	0.224
最大症例数の到達割合	0.091	-	-
当初より症例数が増加した回数の割合	0.146	-	-
無効中止時 (第1ステージ)			
平均	0.069	0.07	0.056
バイアス	0.069	0.07	0.056
MSE	0.022	0.023	0.015
被覆確率	0.968	0.967	0.969
有効中止時 (第1ステージ)			
平均	-0.585	-0.581	-0.477
バイアス	-0.585	-0.581	-0.477
MSE	0.345	0.339	0.230
被覆確率	0.000	0.000	0.000
第1+第2ステージ (第2ステージ完了時)			
平均	-0.081	-0.116	-0.096
バイアス	-0.081	-0.116	-0.096
MSE	0.016	0.023	0.016
被覆確率	0.909	0.906	0.903
Unconditional*1			
平均	0.034	0.027	0.021
バイアス	0.034	0.027	0.021
MSE	0.021	0.023	0.015
被覆確率	0.954	0.952	0.952

*1 第1ステージで中止した場合は第1ステージの結果を、第2ステージに到達した場合は、第1+第2ステージの結果を評価した。

表 3-2 シミュレーション結果の概要 (対立仮説下)

	真のオッズ比 0.636 (対数オッズ比 -0.452)			真のオッズ比 0.731 (対数オッズ比 -0.313)		
	デザイン ①	デザイン ②	デザイン ③	デザイン ①	デザイン ②	デザイン ③
有効中止 (全体: 検出力/第1種の過誤確率)	0.933	0.903	0.976	0.735	0.631	0.802
第1ステージ	0.286	0.287	0.487	0.102	0.104	0.187
第2ステージ	0.647	0.616	0.489	0.633	0.527	0.615
無効中止	0.067	0.097	0.024	0.265	0.369	0.198
第1ステージ	0.047	0.048	0.014	0.171	0.170	0.092
第2ステージ	0.021	0.049	0.01	0.094	0.199	0.106
症例数の平均値 (中央値)	699(685)	570(685)	764(510)	800(685)	591(685)	877(1020)
最小値-最大値	343-1500	343-685	510-1020	343-1500	343-685	510-1020
第1ステージで有効又は無効で中止した割合	0.333	0.335	0.501	0.273	0.274	0.279
計画された症例数で第2ステージ完了の到達割合	0.390	0.665	0.499	0.328	0.726	0.721
最大症例数の到達割合	0.089	-	-	0.163	-	-
当初より症例数が増加した回数の割合	0.278	-	-	0.399	-	-
無効中止時 (第1ステージ)						
平均	-0.061	-0.063	-0.061	-0.039	-0.040	-0.042
バイアス	0.391	0.389	0.391	0.274	0.273	0.270
MSE	0.158	0.156	0.155	0.082	0.081	0.077
被覆確率	0.470	0.466	0.000	0.848	0.855	0.729
有効中止時 (第1ステージ)						
平均	-0.681	-0.682	-0.581	-0.640	-0.642	-0.533
バイアス	-0.229	-0.230	-0.129	-0.327	-0.329	-0.220
MSE	0.062	0.063	0.025	0.114	0.115	0.054
被覆確率	0.915	0.914	0.950	0.759	0.751	0.867
第1+第2ステージ (第2ステージ完了時)						
平均	-0.432	-0.420	-0.396	-0.330	-0.322	-0.302
バイアス	0.020	0.032	0.056	-0.017	-0.009	0.010
MSE	0.011	0.013	0.011	0.010	0.012	0.008
被覆確率	0.975	0.979	0.961	0.975	0.984	0.983
Unconditional*1						
平均	-0.486	-0.478	-0.481	-0.312	-0.308	-0.322
バイアス	-0.034	-0.026	-0.029	0.001	0.005	-0.009
MSE	0.032	0.034	0.02	0.033	0.034	0.023
被覆確率	0.935	0.936	0.943	0.931	0.938	0.938

*1 第1ステージで中止した場合は第1ステージの結果を、第2ステージに到達した場合は、第1+第2ステージの結果を評価した。

3.2 検出力及び平均症例数

真のオッズ比が 0.5 から 1.0 までの範囲における、各デザインの検出力及び平均症例数を図 3-1 に示した。図中の点線は、それぞれ、真のオッズ比 0.636 (期待す

る効果)と真のオッズ比 0.731 (臨床的意義のある最低差)を示している。

真のオッズ比 0.636 の場合、候補デザイン全てにおいて検出力 90%以上が確保されていた。真のオッズ比 0.731 の場合、検出力はデザイン①が 73.5%、デザイン②が 63.1%、デザイン③が 80.2%であった。本試験において期待される真のオッズ比 0.636~0.731 の範囲において、デザイン①は多くの場合に検出力 80%以上が確保され、真のオッズ比が 0.703 未満の場合に検出力 80%以上が期待され、真のオッズ比 0.731 の場合でも検出力 73.5%であった。

真のオッズ比 0.636 の場合、平均症例数はデザイン①が 699 例、デザイン②が 570 例、デザイン③が 764 例であった。デザイン①はデザイン③と比較して、対立仮説下では約 70 例、帰無仮説下では約 100 例少なかった。

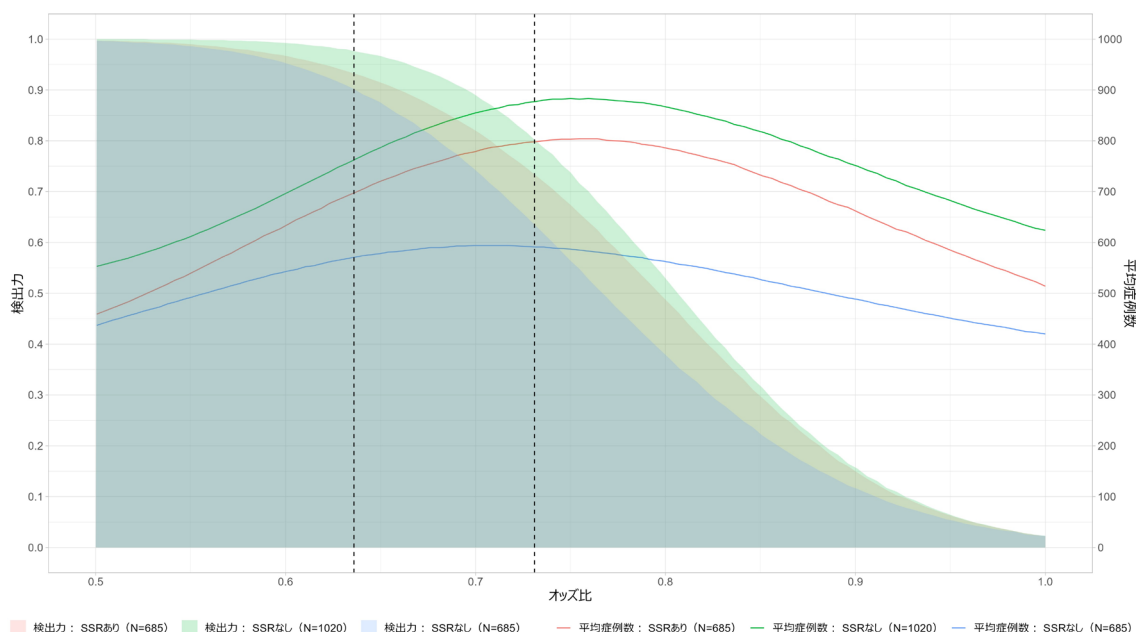


図 3-1 真の治療効果の検出力と平均症例数への影響

3.3 症例数の分布

各シナリオにおける各デザインの症例数の分布を図 3-2 症例数の分布に、真のオッズ比が 0.5 から 1.0 までの範囲における、症例数が増加した割合や最大症例数に到達した割合を図 3-3 症例数が増加した割合及び最大症例数に到達した割合に示した。帰無仮説下にて、デザイン①は 14.6%の割合で症例数が増加し、9.1%の割合で最大症例数に到達した。真のオッズ比 0.731 の場合、症例数が増加する割合はピークに近づき 39.9%の割合で症例数が増加し、16.3%の割合で最大症例数に到達した。症例数が増加する割合はオッズが小さくなるに従い低下していた。

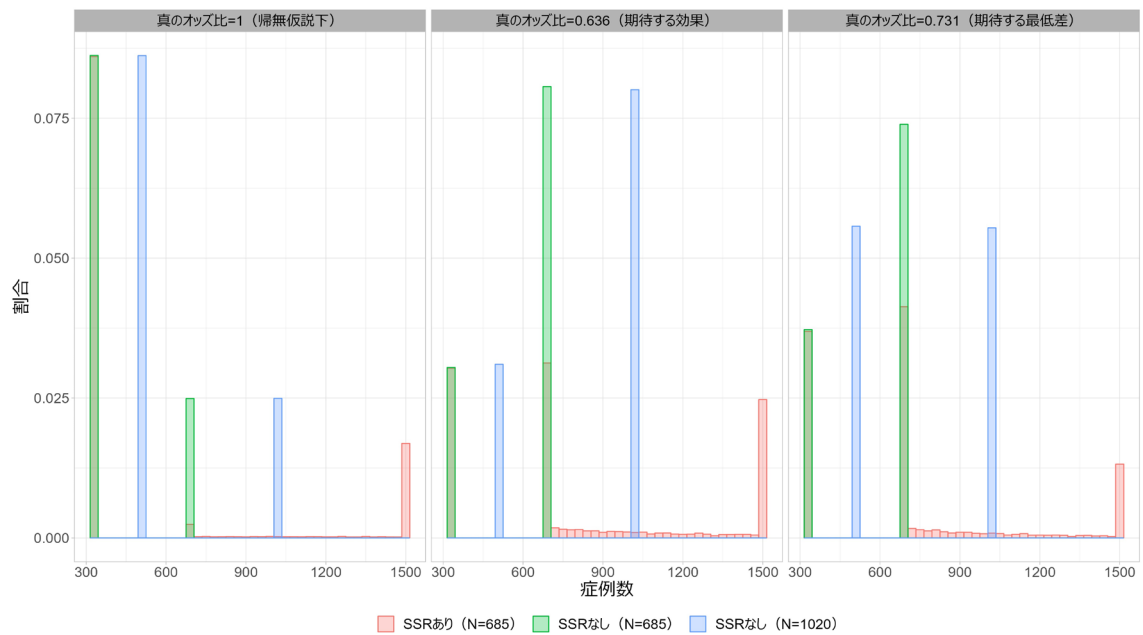


図 3-2 症例数の分布

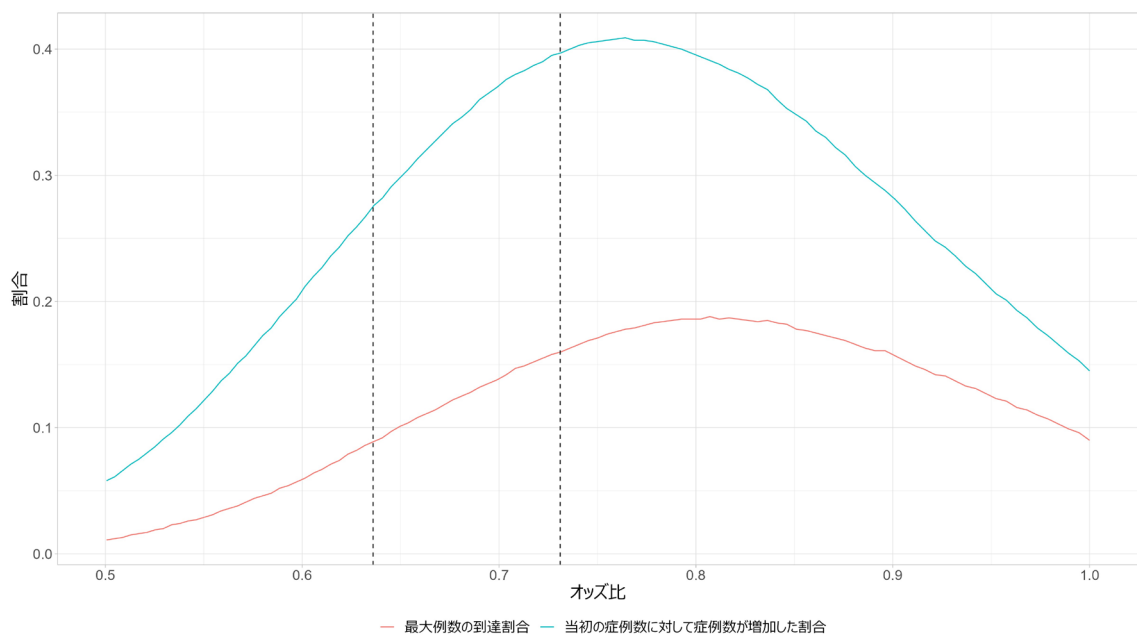


図 3-3 症例数が増加した割合及び最大症例数に到達した割合

3.4 バイアス、MSE 及び信頼区間の被覆確率

真のオッズ比が 0.5 から 1.0 までの範囲における、各デザインのバイアス、MSE 及び 95%信頼区間の被覆確率をそれぞれ図 3-4、図 3-5、図 3-6 に示した。

バイアス及び MSE について、デザイン③は①と②と比較して有効中止時の場合のみ小さい傾向であった。その他時点では各デザイン間で大きな差は確認されなかった。

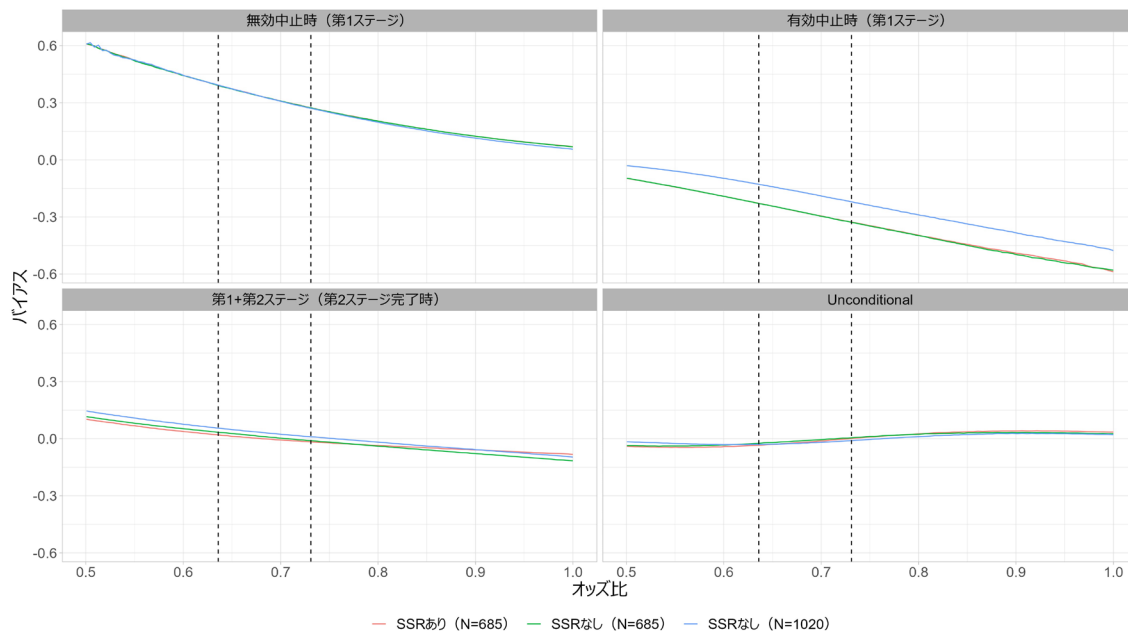


図 3-4 Bias

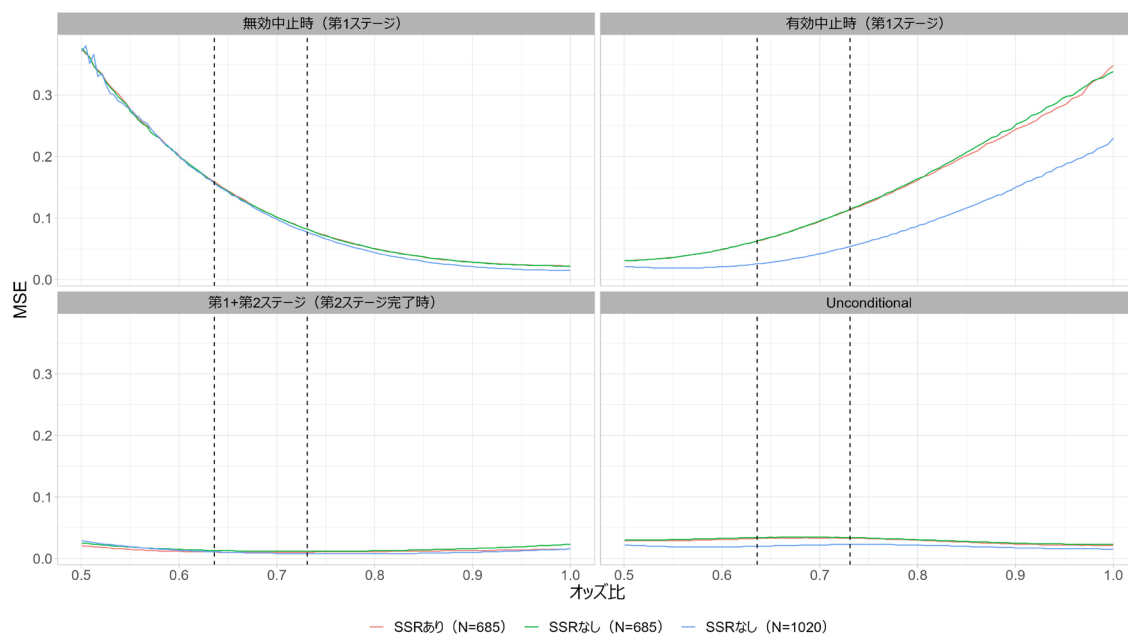


図 3-5 MSE

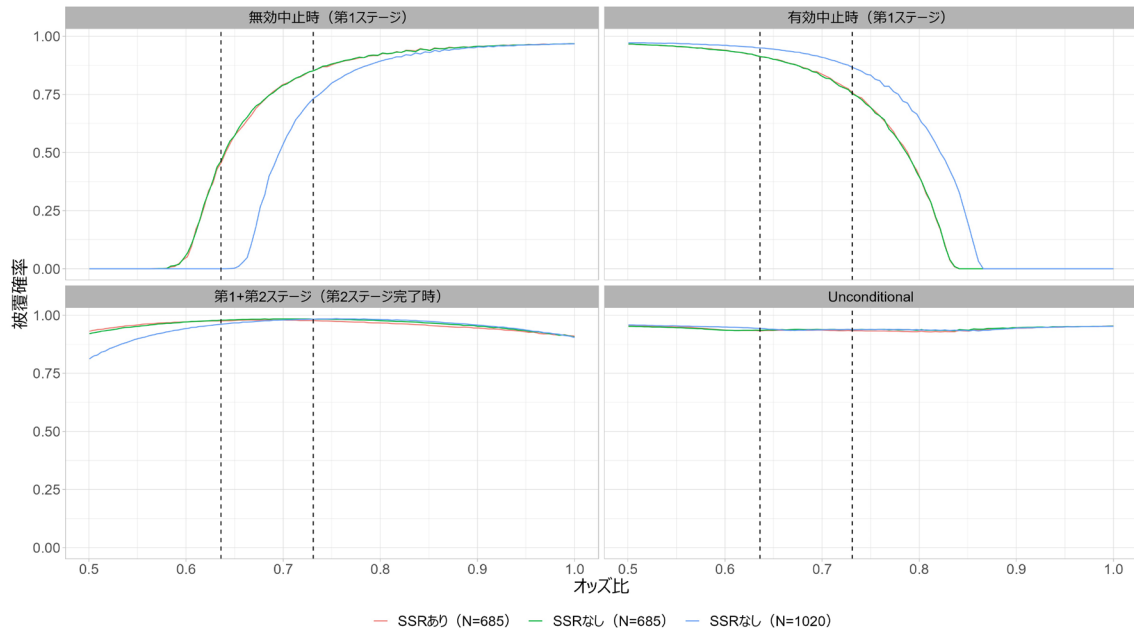


図 3-6 被覆確率

4 要約及び推奨事項

4.1 考察

第一種の過誤確率はいずれのデザインにおいても制御できていた。また、検出力においては、デザイン②は治療効果が下振れした場合に十分な検出力が維持できず、デザイン③は検出力が高いものの、治療効果が極端に小さい場合や、治療効果が極端に高い場合に必要以上に症例を登録してしまっていた。一方、デザイン①は症例数を増やす程度を抑えつつ、治療効果がやや下振れしたとしても検出力をある程度維持できていた。最後にバイアス、MSE、及び信頼区間の被覆確率に関しては、極端に悪いシナリオはなかった。

4.2 推奨されるデザイン

期待される治療効果と平均症例数のバランスから、デザイン①を推奨する。

- 試験開始時の症例数：685 例
- 中間解析：50%の症例が観察を完了した時点で有効及び無効中止の判断を行う。有効中止の基準は O'Brien-Fleming 法に基づく棄却限界値 ($z=2.963$)、無効中止の基準は条件付き検出力が 10%未満($z_1=0.751$) を用いる。
- 症例数再推定: 中間解析時点の条件付き検出力が 20%以上 90%未満の場合に、最大症例数 1500 例を上限に症例数再推定を実施する。

5 参考文献

※参考文献リストについては本成果物の末尾にまとめて記載した。

6 付録

6.1 バリデーション

シミュレーションコードを別添に添付する。シミュレーションの結果は、独立した統計担当者が SAS 9.4 を用いて独立したプログラムを作成し検証を行った。検証プログラムは社内フォルダに保管した。

6.2 計算環境またはその他の技術的詳細

使用したソフトウェア(version) : R (4.2.0)

仕様した R パッケージ : rpact(3.3.2), tidyverse(2.2.2), gt(0.8.0), webshot2(0.1.0)

以降は、本章のシミュレーション報告書のスコープ外ではあるが、本タスクフォースにて検討した内容を参考のために記載した。シミュレーション報告書には含まれない内容であることを留意されたい。

Promising zone 及び無効中止基準の検討

本シミュレーション報告書の作成中に、**promising zone** 及び無効中止基準の検討も行った。これらの基準は社内的に検討するものであり、シミュレーション報告書には記載しないが、参考までに本タスクフォースで検討した内容を以下に記載した。

Promising zone の妥当性を評価するため、**promising zone** を広げた場合のシミュレーションを実行した。

- $10\% \leq CP < 90\%$ (**Promising zone**) : CP が 90% となるように第 2 ステージの症例数を設定
- $CP \geq 90\%$ ~ 有効中止基準以下 (**Favorable zone**) : 症例数は変更しない

また、無効中止基準の妥当性を評価するため、無効中止の基準を条件付き検出力が 20% 未満とした場合のシミュレーションを実行した。

- $20\% \leq CP < 90\%$ (**Promising zone**) : CP が 90% となるように第 2 ステージの症例数を設定
- $CP \geq 90\%$ ~ 有効中止基準以下 (**Favorable zone**) : 症例数は変更しない

帰無仮説下、対立仮説下それぞれのシミュレーションの結果を、それぞれ表 3-3 と表 3-4 に示した。帰無仮説下では、**promising zone** を広げることで、平均症例数が各デザインで 515 から 564 まで増加した。無効中止の基準を大きくすると、検出力 80% を得るための

症例数は 731 例となったため、無効中止の割合が 77.3%へ 0.835 増加したが、平均症例数に大きな変化はなかった。

対立仮説下では、**Promising Zone** を広げることで、真のオッズ比=0.731 のとき、検出力は 0.735 から 0.763 まで増加したが、平均症例数も 800 から 847 まで増加した。また、無効中止の基準を大きくすると、検出力は若干低下し、平均症例数は若干増加した。

表 3-3 Promising zone 及び無効中止基準を変更した場合の
シミュレーション結果の概要 (帰無仮説下)

	真のオッズ比 1 (対数オッズ比 0)		
	SSR あり*1 (N=685)	SSR あり*2 (N=685)	SSR あり*3 (N=731)
有効中止 (全体: 検出力/第 1 種の過誤確率)	0.023	0.022	0.021
第 1 ステージ	0.002	0.002	0.002
第 2 ステージ	0.021	0.021	0.020
無効中止	0.977	0.978	0.979
第 1 ステージ	0.773	0.773	0.835
第 2 ステージ	0.204	0.205	0.144
平均症例数 (中央値)	515(343)	564(343)	516(366)
最大値-最小値	343-1500	343-1500	366-1500
第 1 ステージで有効又は無効で中止した割合	0.775	0.774	0.836
計画された症例数で第 2 ステージ完了の到達割合	0.079	0.020	0.020
最大症例数の到達割合	0.091	0.151	0.094
当初の症例数に対して症例数が増加した回数の割合	0.146	0.206	0.144
無効中止時 (第 1 ステージ)			
平均	0.069	0.069	0.051
バイアス	0.069	0.069	0.051
MSE	0.022	0.022	0.021
被覆確率	0.968	0.968	0.971
有効中止時 (第 1 ステージ)			
平均	-0.585	-0.585	-0.568
バイアス	-0.585	-0.585	-0.568
MSE	0.345	0.346	0.326
被覆確率	0.000	0.000	0.000
第 1+第 2 ステージ (第 2 ステージ完了時)			
平均	-0.081	-0.071	-0.084
バイアス	-0.081	-0.071	-0.084
MSE	0.016	0.014	0.016
被覆確率	0.909	0.907	0.885
Unconditional*4			
平均	0.034	0.037	0.028
バイアス	0.034	0.037	0.028
MSE	0.021	0.021	0.021
被覆確率	0.954	0.953	0.956

*1 Promising zone を $20\% \leq CP < 90\%$ 、無効中止の基準を 10%とした場合

*2 Promising zone を $10\% \leq CP < 90\%$ 、無効中止の基準を 10%とした場合

*3 Promising zone を $20\% \leq CP < 90\%$ 、無効中止の基準を 20%とした場合

*4 第 1 ステージで中止した場合は第 1 ステージの結果を、第 2 ステージに到達した場合は、第 1+第 2 ステージの結果を評価した。

表 3-4 Promising zone 及び無効中止基準を変更した場合の
シミュレーション結果の概要 (対立仮説下)

	真のオッズ比 0.636 (対数オッズ比 -0.452)			真のオッズ比 0.731 (対数オッズ比 -0.313)		
	SSR あり*1 (N=685)	SSR あり*2 (N=685)	SSR あり*3 (N=731)	SSR あり*1 (N=685)	SSR あり*2 (N=685)	SSR あり*3 (N=731)
有効中止 (全体: 検出力/第1種の過誤確率)	0.933	0.942	0.928	0.735	0.763	0.731
第1ステージ	0.286	0.286	0.316	0.102	0.102	0.113
第2ステージ	0.647	0.656	0.612	0.633	0.661	0.617
無効中止	0.067	0.058	0.072	0.265	0.237	0.269
第1ステージ	0.047	0.046	0.063	0.171	0.171	0.213
第2ステージ	0.021	0.012	0.010	0.094	0.066	0.057
平均症例数 (中央値)	699(685)	722(685)	708(731)	800(685)	847(685)	811(731)
最大値-最小値	343-1500	343-1500	366-1500	343-1500	343-1500	366-1500
第1ステージで有効又は無効で中止した割合	0.333	0.332	0.378	0.273	0.273	0.326
計画された症例数で第2ステージ完了の到達割合	0.390	0.362	0.363	0.328	0.271	0.281
最大症例数の到達割合	0.089	0.117	0.089	0.163	0.22	0.172
当初の症例数に対して症例数が増加した回数の割合	0.278	0.306	0.259	0.399	0.456	0.393
無効中止時 (第1ステージ)						
平均	-0.061	-0.062	-0.098	-0.039	-0.039	-0.07
バイアス	0.391	0.390	0.354	0.274	0.274	0.243
MSE	0.158	0.157	0.13	0.082	0.082	0.066
被覆確率	0.470	0.471	0.607	0.848	0.850	0.878
有効中止時 (第1ステージ)						
平均	-0.681	-0.681	-0.663	-0.640	-0.640	-0.620
バイアス	-0.229	-0.229	-0.211	-0.327	-0.327	-0.308
MSE	0.062	0.062	0.054	0.114	0.113	0.101
被覆確率	0.915	0.914	0.922	0.759	0.761	0.783
第1+第2ステージ (第2ステージ完了時)						
平均	-0.432	-0.435	-0.432	-0.330	-0.333	-0.334
バイアス	0.020	0.017	0.02	-0.017	-0.020	-0.021
MSE	0.011	0.010	0.009	0.010	0.01	0.009
被覆確率	0.975	0.976	0.979	0.975	0.974	0.978
Unconditional*4						
平均	-0.486	-0.488	-0.484	-0.312	-0.314	-0.31
バイアス	-0.034	-0.036	-0.032	0.001	-0.001	0.002
MSE	0.032	0.032	0.031	0.033	0.033	0.032
被覆確率	0.935	0.935	0.937	0.931	0.931	0.934

*1 Promising zone を $20\% \leq CP < 90\%$ 、無効中止の基準を 10%とした場合

*2 Promising zone を $10\% \leq CP < 90\%$ 、無効中止の基準を 10%とした場合

*3 Promising zone を $20\% \leq CP < 90\%$ 、無効中止の基準を 20%とした場合

*4 第1ステージで中止した場合は第1ステージの結果を、第2ステージに到達した場合は、第1+第2ステージの結果を評価した。

治療効果の仮定 (2.4 節)

本シミュレーション報告書では登録スピードの検討を省いたが、実際には状況に応じて様々な登録シナリオを設定することが考えられる。

その他の臨床上の仮定 (2.5 節)

本シミュレーション報告書では想定外の被験者募集や脱落が起こらないと仮定した。これらが実際に起こり得ることが想定される場合には、いくつかのシナリオを設定することが考えられる。

評価指標 (2.6 節)

本シミュレーション報告書では、治療効果の推定に対して補正などは行わない未調整の方法を用いた。補正が可能な場合には、補正後の推定値でバイアス、MSE、信頼区間の被覆確率の評価を行うことが考えられる。

バイアス、MSE 及び信頼区間の被覆確率 (3.4 節)

本シミュレーション報告書は仮想的に作成されているため、バイアスや MSE の程度に関する議論を含めなかった。バイアスや MSE の程度の臨床的な意義を考察することが推奨される。

今回のシミュレーションでは、症例数再推定のデザインは、症例数再推定のない群逐次デザインと比較して、性能上の確実な利点はみられなかった。このように、統計学のコミュニティでは、症例数再推定の利点について常に統一された意見があるわけではない。Jennison et al (2015)^[14] は、一様に小さい平均症例数と一致する検出力を持つ「改善された」群逐次デザインの設計を見つけれられることを報告した。一方、Liu et al (2021)^[15] は、中央値、分散、裾の確率のようなサンプルサイズ分布の他の側面も検討されるべきで、平均症例数と検出力に注目しても、特定のスポンサーにとってどのデザインがより適しているかについての洞察はあまり得られないことを指摘した。Liu et al (2021)^[15] はさらに、各指標の関連性と重要性は、パイプライン、開発段階とタイムライン、潜在的な財務的影響など、スポンサーの状況によっても異なることを述べている。例えば、群逐次デザインは、治療効果が極端に高い場合や低い場合にサンプルサイズが過剰になる可能性がある。中小規模のスポンサーにとって、限られた資源を最初に投資し、有望な中間結果を見た後にのみさらに投資する症例数再推定のデザインは、群逐次デザインと同じ利点だけでなく、中間結果が有望な兆候を示した場合にのみサンプルサイズを増加させることができる点で利点を享受できると述べられている。

4 シミュレーション報告書の例：治療群の選択を伴うアダプティブ

デザイン

本章では、INHANCE 試験⁸⁾を参考に、治療群の選択を伴うアダプティブデザインを想定した、仮想的なシミュレーション報告書を作成した。INHANCE 試験の概略を以下に示す。

INHANCE 試験は、慢性閉塞性肺疾患の患者を対象に、Indacaterol 4 用量（75, 150, 300, 600 μ g o.d.）を試験治療、プラセボと Formoterol および Tiotropium を対照治療とした試験であり、2 ステージのアダプティブデザインを採用している。第 1 ステージの目的は、第 2 ステージにて検証する Indacaterol の用量を選択することであり、試験全体の主目的は第 1 ステージにて選択された少なくとも 1 つの Indacaterol の用量のプラセボに対する優越性の検証、重要な副次目的は少なくとも 1 つの Indacaterol の用量の Tiotropium に対する非劣性の検証であった。第 1 ステージの症例数は、2 つの用量を選択するのに十分と考えられる例数として 1 群あたり 110 例、試験全体の症例数は、最終解析において、重要な副次項目まで検証できる検出力に基づき設定され、1 群あたり 340 例（第 2 ステージの症例数として 1 群あたり 230 例）としていた。なお、選択群の症例者数を 1 群あたり 340 例とすると、主要評価項目の検出力はほぼ 100%となっていた。

中間解析における用量選択のルールとして以下の 2 つを設定していた。

- 基準 1：2 週時点のトラフ FEV₁ を評価項目として、プラセボとの群間差の点推定値が 0.12 L よりも大きく、かつ点推定値が Formoterol 群および Tiotropium 群のいずれよりも大きい
- 基準 2：2 週時点の FEV₁AUC_{1-4h} を評価項目として、点推定値が Formoterol 群および Tiotropium 群のいずれよりも大きい

2 つの評価項目トラフ FEV₁ と FEV₁AUC_{1-4h} はいずれも数値が大きいほど効果が高いことを意味する。2 つの基準を満たす用量が複数ある場合、両基準を満たす最低の用量とそれより 1 つ高い用量が第 2 ステージに進むなど、2 つの基準を満たす状況に応じた用量選択ルールを事前に規定していた。なお、中間解析の目的は用量選択のみであり、症例数再推定等は計画していなかった。

約 770 例（1 群あたり約 110 例）が 2 週間の治療を終えたタイミングで中間解析が実施され、用量選択ルールに基づき選択された Indacaterol 2 用量（150 μ g, 300 μ g）、プラセボおよび Tiotropium の計 4 群が有効性、安全性および忍容性を検討する 26 週間の第 2 ステージに進むとともに、追加の被験者組み入れが行われた。有効性の最終解析には 12 週時点のトラフ FEV₁ が評価項目に用いられ、各群 376~393 例が解析対象となった。試験開始時の Indacaterol が 4 用量であったため、Bonferroni 法に基づき有意水準を $\alpha/4$ （片側 $\alpha=0.025$ ）として Indacaterol 2 用量のプラセボに対する優越性の検定が行われた。より検出力の高い方法を適用することは可能であったが、試験計画が複雑になるため採用しなかったとしている。

最終解析の結果、Indacaterol 2 用量とともにプラセボに対する優越性が検証された。

以降、INHANCE 試験の計画を簡略化のため一部改変し、実際に作成したシミュレーション報告書について記載する。想定する臨床シナリオは、オリジナルの INHANCE 試験から主に 3 つの点を変更している。第一に、最終解析の目的は Indacaterol のプラセボ群に対する優越性検証のみとし、Tiotropium 群に対する非劣性検証は検討対象外とした。第二に、陽性対照群は両ステージを通して 1 群のみとし、Indacaterol の用量選択の判定にのみ用いた。第三に、用量選択と最終解析に用いる評価項目はいずれも 12 週時点のトラフ FEV₁ とし、2 週時点のトラフ FEV₁ および FEV₁AUC_{1-4h} は検討対象外とした。そのため、オリジナルと異なり用量選択と最終解析に同一のデータを使用している。

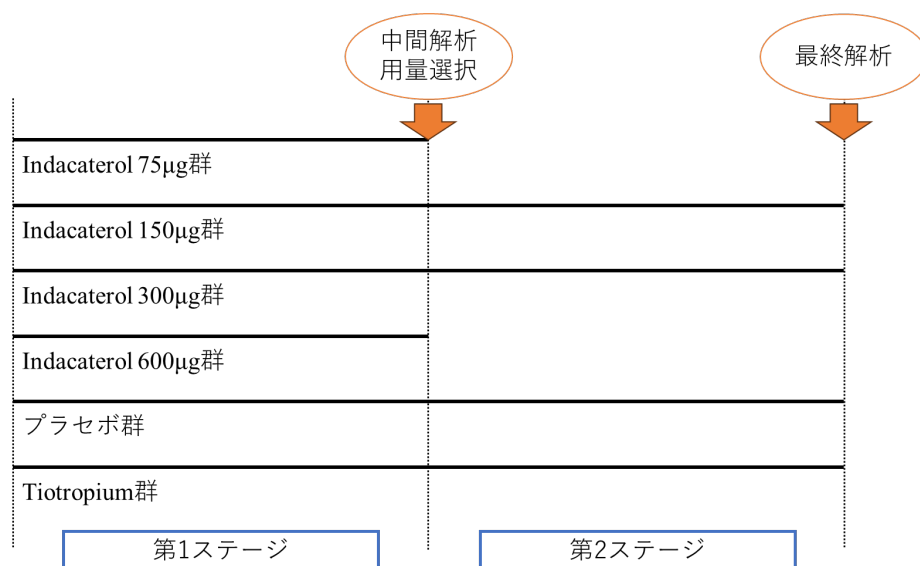
用量選択ルールとして、オリジナルの INHANCE 試験に基づき、有効性の基準を満たした最低用量とそれより 1 つ高い用量を選択するルールを設定した。なお、1 つ高い用量については有効性の基準を満たしていなくとも選択可能である。加えて、INHANCE 試験終了後の FDA との審査時のやり取りにおいて、選択されなかった 75µg と選択された 2 用量 (150µg、300µg) とで有効性の差異が小さかったため、75µg について追加の検討がなされるべきであった^[17]と審査報告書に記載があった点を考慮し、前述のルールに加えて、有効性の基準を満たした最低用量とそれより 1 つ低い用量を選択するルールも設定した。

1 緒言

1.1 本試験の目的

試験表題	A 26-week Treatment, Multicenter, Randomized, Double Blind, Double Dummy, Placebo-controlled, Adaptive, Seamless, Parallel-group Study to Assess the Efficacy, Safety and Tolerability of Two Doses of Indacaterol (Selected From 75, 150, 300 & 600 µg o.d.) in Patients with Chronic Obstructive Pulmonary Disease
目的	試験の第 1 ステージにて、Indacaterol の 4 つの投与量 (75, 150, 300, 600 µg o.d.) に関するリスクベネフィットを評価し、第 2 ステージに持ち越す用量を選択し、COPD 患者における選択された Indacaterol 用量の有効性、安全性、および忍容性をプラセボと比較して評価すること
投与群	第 1 ステージ : Indacaterol 群 (75, 150, 300, 600 µg) プラセボ群 Tiotropium 群 (陽性対照群) 第 2 ステージ : 選択された Indacaterol 群 (1 または 2 用量) プラセボ群

	Tiotropium 群 (陽性対照群)
主要評価項目	12 週間の治療後 24 時間後に肺活量測定によって評価された 1 秒のトラフ強制呼気量 (FEV1)



1.2 シミュレーションの目的

本試験は、複数の Indacaterol 群の用量から、リスクベネフィットを評価し適切な用量を選択する第 1 ステージおよび選択された用量の有効性を検証するための症例を追加する第 2 ステージからなる。適切な用量の選択には、中間解析によりプラセボ及び Formoterol 群との有効性を比較してより良い用量を選択するアダプテーションを可能としている。

これに伴い、本シミュレーション報告書では、用量選択を行った場合の動作特性をシミュレーションにより確認し、各シナリオによるシミュレーション結果が目標とする水準を満たすことを示すために実施した。

2 シミュレーション入力設定

2.1 試験デザイン案

2.2 中間解析及び判定基準

検討した試験デザインは、第 1 ステージの中間解析により治療群選択を行い最終解析時点で選択された治療群を検証する下記の 2 つである。

- Inferentially Seamless デザイン：最終解析は第 1 及び第 2 ステージのすべてのデータで実施する治療群選択アダプティブデザイン
- Operationally Seamless デザイン：最終解析は第 2 ステージのデータのみで実施する治療群選択ベンチマーク試験デザイン

シミュレーションの入力設定

パラメータ	設定
有意水準 (α)	0.025 (片側)
群	第1ステージ：6群（試験治療4用量、プラセボ群、陽性対照群） 第2ステージ：3~4群（選択された試験治療1~2用量、プラセボ群、陽性対照群）
第1ステージの症例数/群	110 <設定根拠> 本シミュレーション報告書はあくまで仮想的な事例のため、INHANCE試験の症例数にあわせて設定した。 ※実際にシミュレーション報告書を作成する際は適切な設定根拠を記載することを推奨する
第2ステージの症例数/群	230 <設定根拠> 第1ステージと同様。
割付比	全ての群で同じ
中間解析のタイミング	第1ステージ終了時
中間解析の目的	最終解析において検証する試験治療の選択 ※有効中止を意図した中間解析の実施はなし
用量選択ルール	選択条件： 「FEV1のプラセボ群との差が0.4よりも大きい」かつ「陽性対照群との差が0より大きい」 選択ルール1： 選択条件を満たす最低用量①と、①より1つ高い用量②の2群 (以降、「高い用量を選択」と記載) 選択ルール2： 選択条件を満たす最低用量①と、①より1つ低い用量②の2群 (以降、「低い用量を選択」と記載) ※②については選択条件を満たしていなくても選択可能とし、②がなければ1用量のみ選択する
最終解析の目的	中間解析において選択された試験治療のプラセボ群に対する優越性の検証
検討する試験デザイン案	1 Inferentially Seamless：最終解析に第1ステージおよび第2ステージのすべてのデータを使用する、治療群選択のアダプティブデザ

パラメータ	設定
および統計手法	イン
	1.1 逆正規法 ※重みには群ごとの各ステージの症例数を用いる。ステージごとの多重調整には Bonferroni 法を用いる
	2 Operationally Seamless：最終解析には第 2 ステージのデータのみ使用する、治療群選択の固定デザイン(ベンチマークとして設定)
	2.1 Step-down Dunnett 検定 いずれのデザインも、推定手法はナイーブな方法(未調整の平均値)を用いる。

2.3 仮想データ生成モデル

本試験のアウトカム変数は、連続変数である。脱落・欠測データメカニズムは検討していない。また、アウトカムに影響を与える共変量についても検討していない。

アウトカムである 12 週時点のトラフ FEV1 (L)について、正規分布に従う乱数を 6 群の両ステージの分だけ生成する。治療効果は本シミュレーション報告書「2.4 治療効果の仮定」の設定に従う。 $\mu_1, \mu_2, \mu_3, \mu_4$ は試験治療の用量 1,2,3,4 の治療効果(低用量から高用量の順)、 μ_9 は陽性対照群の治療効果を表している(標準化されたプラセボ群との群間差)。標準偏差は全群共通で 1 とする。第 1 ステージの被験者と第 2 ステージの被験者は別々のため、ステージ間の相関は考慮せず独立に乱数生成する。第 1 ステージのデータに基づき用量選択の判定を行い、選択されなかった用量の第 2 ステージのデータは評価に使用しない。1 用量も選択されなかった場合は試験自体が無効中止となるため、第 2 ステージの全データは評価に使用しない。

2.4 治療効果の仮定

治療効果のシナリオとして、第一種の過誤確率評価用のシナリオと検出力評価用のシナリオを検討した。

① 第一種の過誤確率評価用のシナリオ

シナリオ No.	μ_1	μ_2	μ_3	μ_4	μ_9	効果の想定
100	0	0	0	0	0	全群効果なし
101	0.5	0	0	0	0	1 用量効果あり
102	0	0.5	0	0	0	
103	0	0	0.5	0	0	
104	0	0	0	0.5	0	

シナリオ No.	μ_1	μ_2	μ_3	μ_4	μ_9	効果の想定
105	0.5	0.5	0	0	0	2 用量効果あり
106	0.5	0	0.5	0	0	
107	0.5	0	0	0.5	0	
108	0	0.5	0.5	0	0	
109	0	0.5	0	0.5	0	
110	0	0	0.5	0.5	0	
111	0.5	0.5	0.5	0	0	3 用量効果あり
112	0.5	0.5	0	0.5	0	
113	0.5	0	0.5	0.5	0	
114	0	0.5	0.5	0.5	0	

② 検出力評価用のシナリオ

シナリオ No.	μ_1	μ_2	μ_3	μ_4	μ_9	効果の想定
201	0.4	0.5	0.5	0.5	0.4	用量2で頭打ち
202	0.4	0.45	0.5	0.5	0.4	用量3で頭打ち
203	0.4	0.433	0.467	0.5	0.4	用量4まで用量反応
204	0.4	0.4	0.5	0.5	0.4	用量1 = 用量2
205	0.4	0.5	0.5	0.5	0.3	201~204より 陽性対照が0.1低い
206	0.4	0.45	0.5	0.5	0.3	
207	0.4	0.433	0.467	0.5	0.3	
208	0.4	0.4	0.5	0.5	0.3	

2.5 その他の臨床上の仮定

本シミュレーションでは被験者の登録スピードの影響や脱落は考慮しなかった。

2.6 評価指標

検討した動作特性の評価指標を下表に示す。

指標	目的
1. 第一種の過誤確率	2.4 治療効果の仮定 ①第一種の過誤確率評価用のシナリオの全てにおいて、検討する試験デザイン案/統計手法案における第一種の過誤確率（効果なしのいずれかの用量が選択されかつ

	プラセボ群との有意差が得られる確率) が名義水準以下になることを確認する。
2. 検出力	2.4 治療効果の仮定 ②検出力評価用のシナリオにおいて、検討する試験デザイン案における検出力を確認する。 ・用量ごとの検出力 ・用量群全体の検出力 (効果ありのいずれかの用量が選択されかつプラセボ群との有意差が得られる確率)
3. 各群と試験全体の症例数の分布	2.4 治療効果の仮定 ②検出力評価用のシナリオにおいて、治療群の選択割合及び中止割合を考慮した期待症例数を確認する。
4. 各群の選択割合、選択された治療の数の分布	各群が選択される割合および選択用量数の分布を確認する。設定した用量選択ルールが期待通りに機能していることを確認する。選択用量数が 0 である割合は、基準を満たす用量が存在せずに試験を中止する無効中止確率に相当する。
5. 治療効果の推定	検出力評価用のシナリオにおいて、ステージごとおよび両ステージにおける、各群の点推定値 (平均値) のバイアス、MSE および信頼区間の被覆確率を評価する。 中間解析にて選択された場合および選択されなかった場合における、条件付き推定の結果の評価も行う。

2.7 シミュレーションの繰り返し回数と乱数シード

各シナリオのシミュレーション回数は 100,000 回とした。シミュレーションのシナリオごとに異なる乱数シードを用いた。

3 シミュレーション結果

3.1 第一種の過誤確率

第一種の過誤確率を、用量選択ルール別にそれぞれ図 4-1 および図 4-2 に示した。また、全体の第一種の過誤確率を表 4-1 に示した。第一種の過誤確率（全体）は、シナリオ No.100~114 の全シナリオにおける第一種の過誤確率（効果なしのいずれかの用量が選択されかつプラセボ群との有意差が得られる確率）のうちの最大値を意味している。いずれの用量選択ルールおよびデザインにおいても、第一種の過誤確率が強い意味で制御されていた。Operationally Seamless デザインでは名義水準の 2.5%に近い値となっていた一方で、Inferentially Seamless デザインではかなり保守的な結果となっていた。

表 4-1 第一種の過誤確率（全体）

試験デザイン	用量選択ルール	
	高い用量を選択	低い用量を選択
Inferentially Seamless	0.35%	0.38%
Operationally Seamless	2.21%	2.26%

3.2 検出力

検出力を、用量選択ルール別にそれぞれ図 4-3 および図 4-4 に示した。

高い用量を選択するルールにおける用量群全体の検出力（効果ありのいずれかの用量が選択されかつプラセボ群との有意差が得られる確率）について、シナリオ 201~208 の全てで検出力が 80%を超えていた。手法間の差異は微々たるものであったが、いずれのシナリオでも Operationally Seamless デザインより Inferentially Seamless デザインにおいてわずかに検出力が高い様子が見られた。

低い用量を選択するルールにおいても、用量群全体の検出力では同様の傾向が見られた。群ごとの検出力では、用量選択ルールを反映して低い用量における検出力が高くなる傾向が見られた。

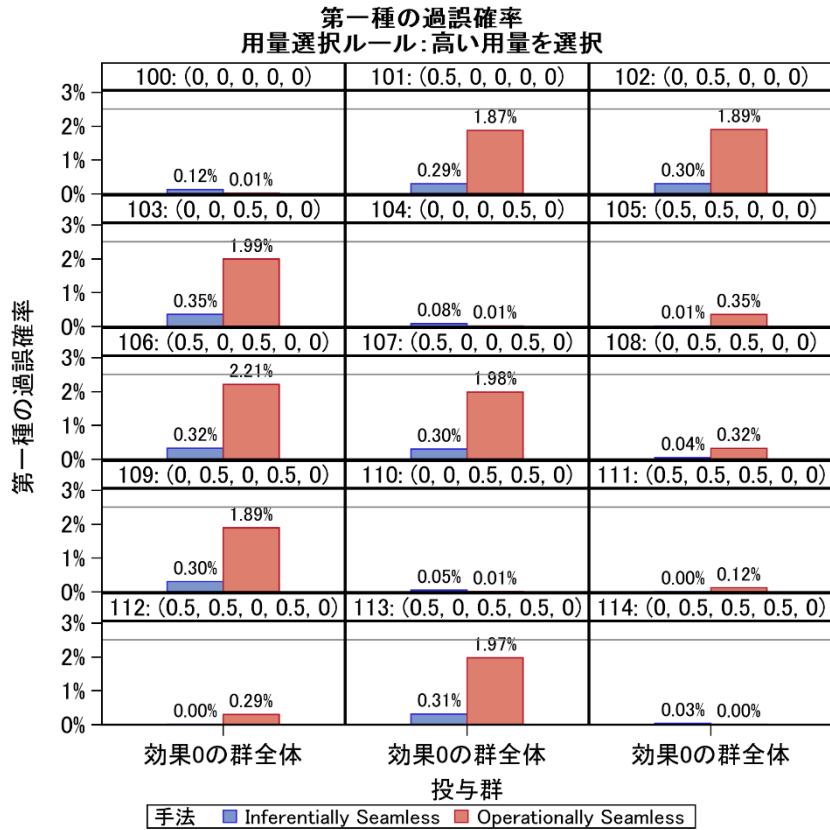


図 4-1 第一種の過誤確率 (用量選択ルール: 高い用量を選択)

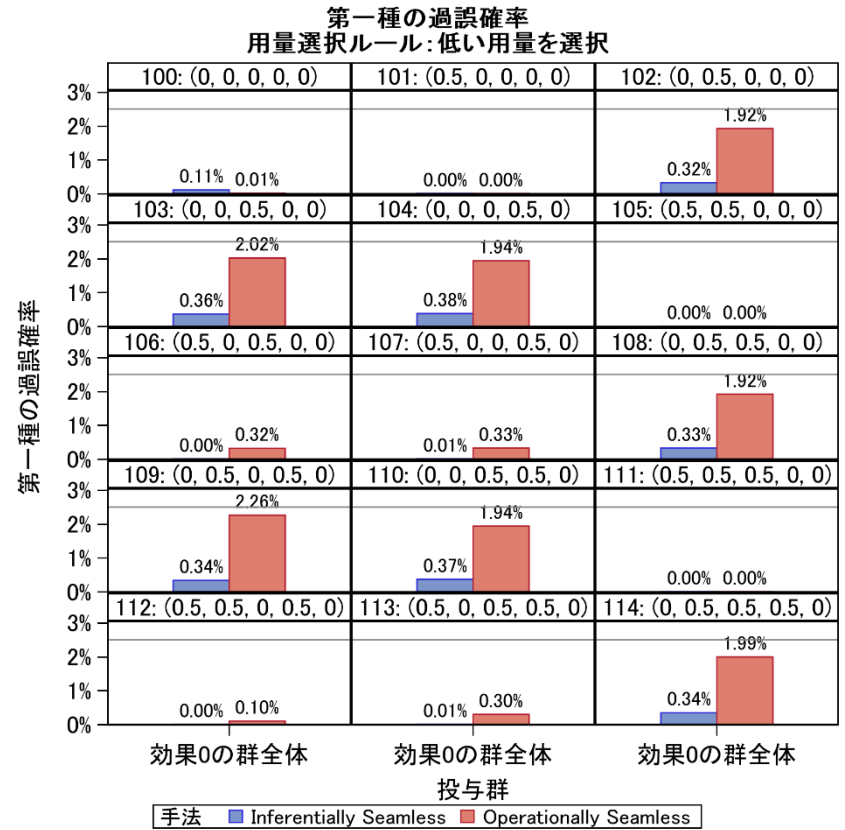


図 4-2 第一種の過誤確率 (用量選択ルール: 低い用量を選択)

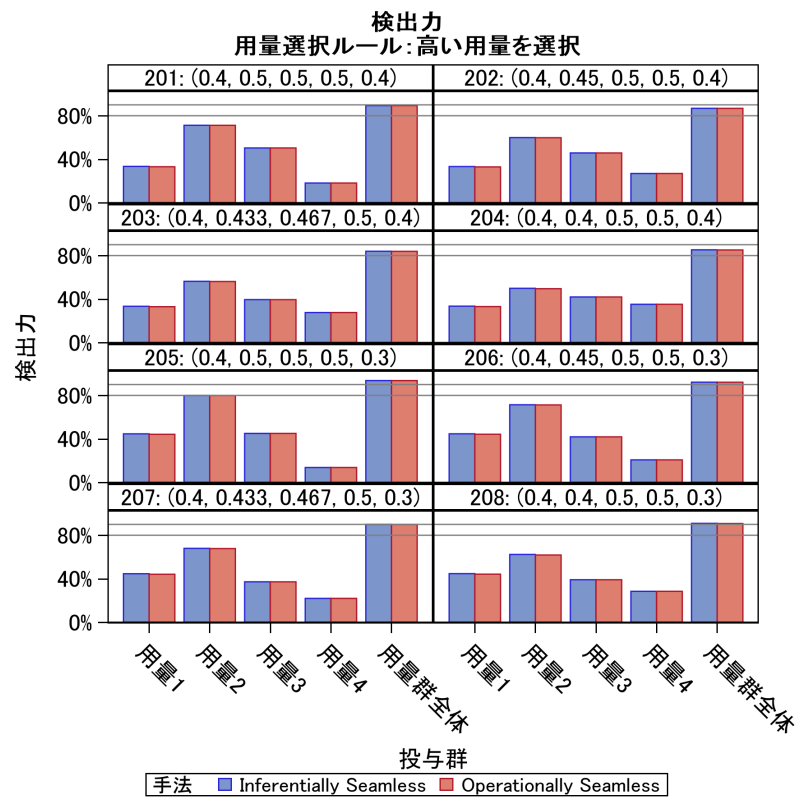


図 4-3 検出力 (用量選択ルール: 高い用量を選択)

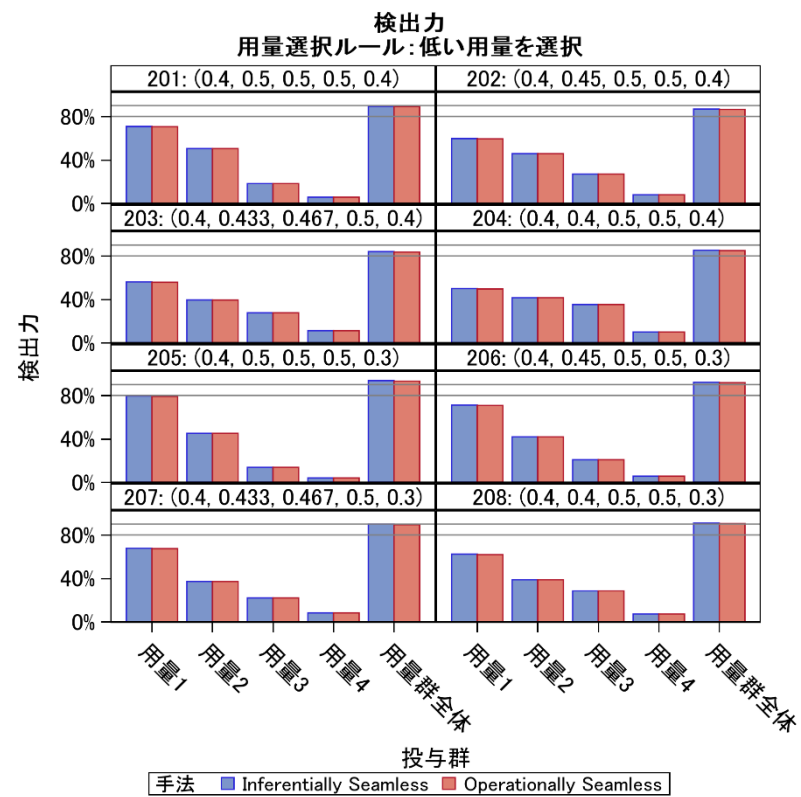


図 4-4 検出力 (用量選択ルール: 低い用量を選択)

3.3 各群の選択割合、選択された治療の数の分布

3.3.1 選択割合

各群の選択割合の分布をそれぞれ図 4-5 から図 4-8 までに示した。

高い用量を選択するルール（図 4-5 および図 4-7）においては、最低用量の用量 1 はすべて「選択条件を満たす最低用量」としてのみ選択されている一方、用量 2 から 4 は「選択条件を満たす最低用量」として選択される場合と、「その 1 つ高い用量」として選択される場合の 2 通りがある。第一種の過誤確率のシナリオ（図 4-5）では、シナリオ 100 を除き、真値が非ゼロのうち一番下の用量が「選択条件を満たす最低用量」として選択されている。逆に、真値がゼロの用量が「選択条件を満たす最低用量」として選択されているケースは 0.3%未満であり、ほとんど発生していない。検出力のシナリオ（図 4-7）では、いずれのシナリオにおいても、用量 2 の選択割合が一番高いことが分かる。

低い用量を選択するルール（図 4-6 および図 4-8）においても同様に、最高用量の用量 4 はすべて「選択条件を満たす最低用量」としてのみ選択されている一方、用量 1 から 3 は「選択条件を満たす最低用量」として選択される場合と、「その 1 つ低い用量」として選択される場合の 2 通りがある。第一種の過誤確率のシナリオ（図 4-6）では、高い用量を選択するルールと同様に、シナリオ 100 を除き、真値が非ゼロのうち一番下の用量が「選択条件を満たす最低用量」として選択されていることが分かる。検出力のシナリオ（図 4-8）では、いずれのシナリオにおいても、用量 1 の選択割合が一番高いことが分かる。

なお、用量選択ルールの定義から、「選択条件を満たす最低用量」として選択される割合は両ルールで同一となっており、高い用量を選択するルールにおける用量 2~4 の選択割合は、低い用量を選択するルールにおける用量 1~3 の選択割合とそれぞれ同一となっている。

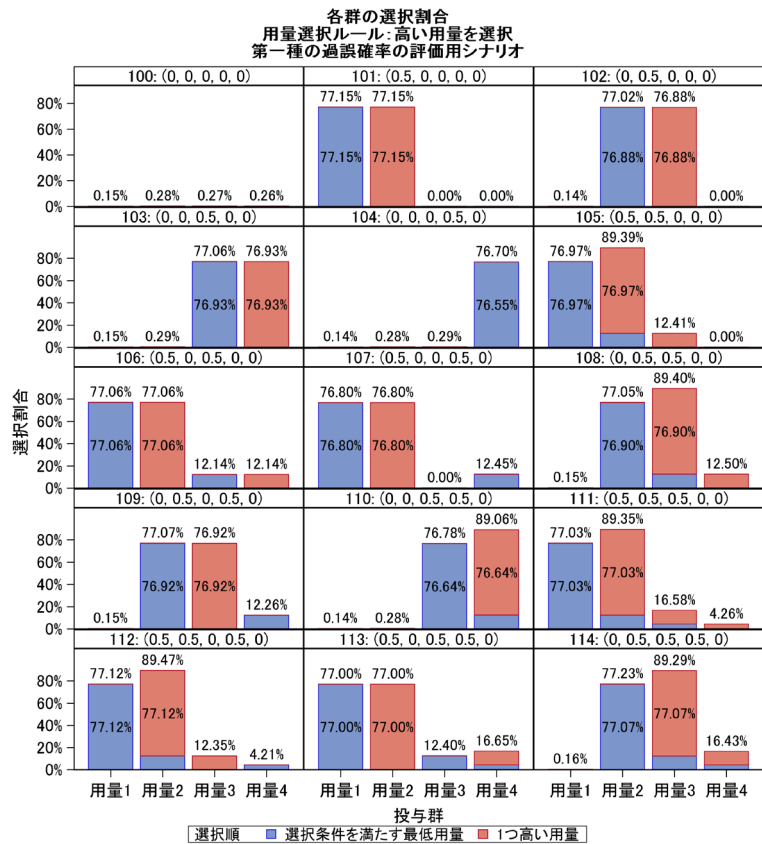


図 4-5 各群の選択割合

(用量選択ルール：高い用量を選択；シナリオ 100～114)

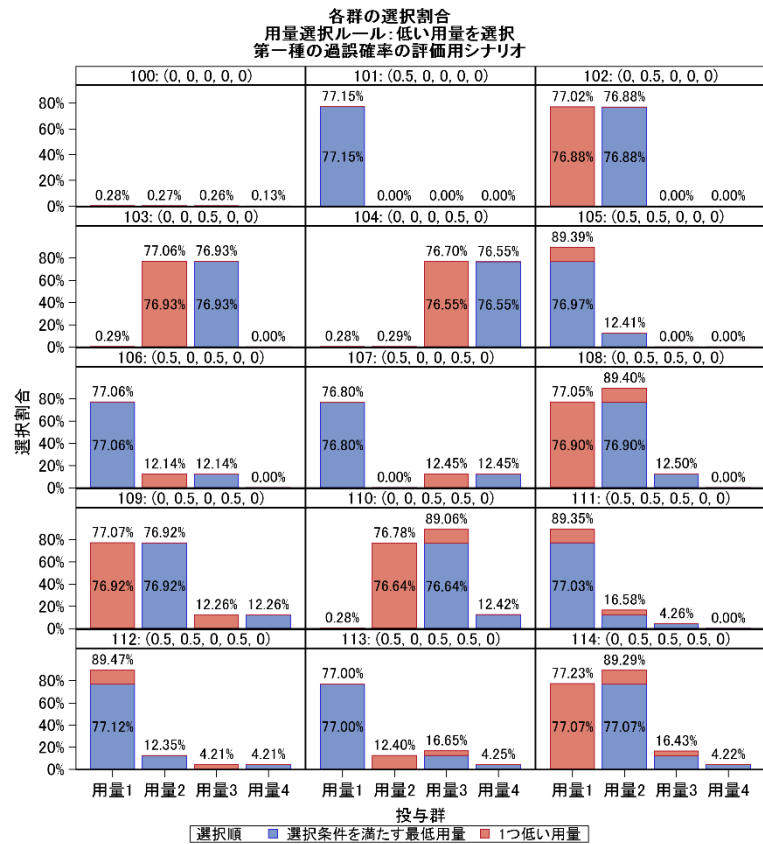


図 4-6 各群の選択割合

(用量選択ルール：低い用量を選択；シナリオ 100～114)

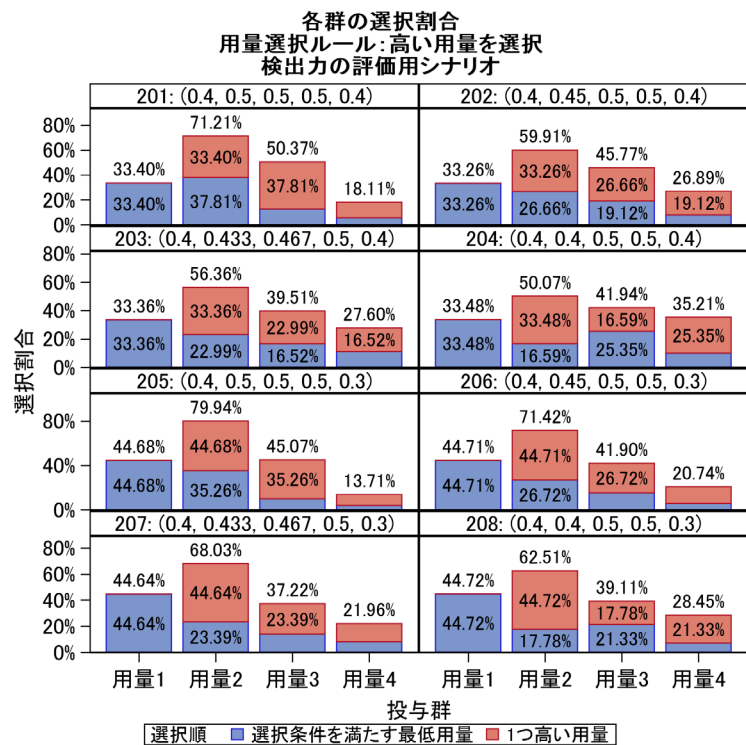


図 4-7 各群の選択割合

(用量選択ルール: 高い用量を選択; シナリオ 201~208)

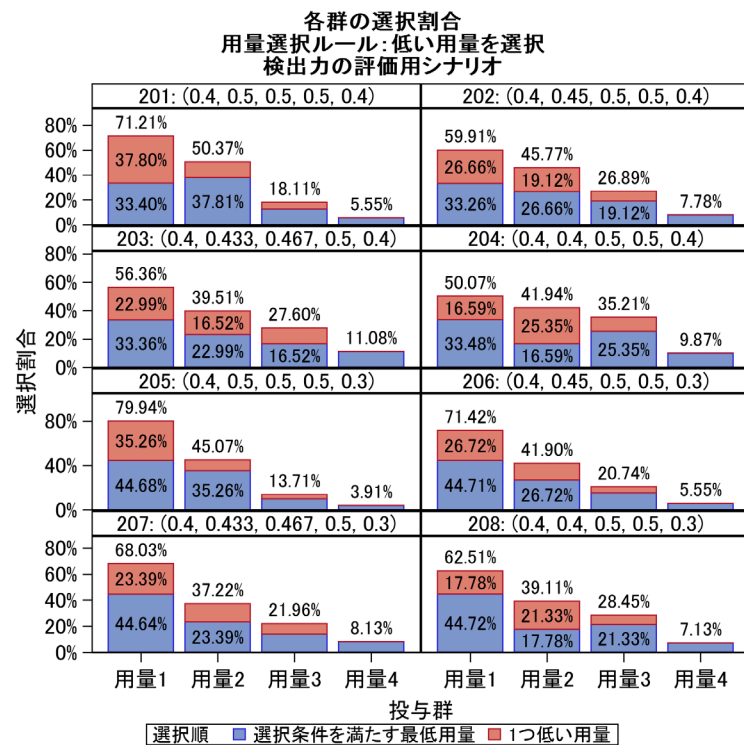


図 4-8 各群の選択割合

(用量選択ルール: 低い用量を選択; シナリオ 201~208)

3.3.2 選択用量数

選択用量数の分布をそれぞれ図 4-9～図 4-12 に示した。

高い用量を選択するルールにおいて、第一種の過誤確率のシナリオ（図 4-9）のうちシナリオ No.100 では、すべての群の効果がゼロであるにも関わらず試験が無効中止せずに第 2 ステージに進んでしまう確率が 0.54%と十分に小さいことが確認できる。真値が非ゼロの用量数が増えるにしたがって、少なくとも 1 用量選択される確率が増えていることが分かる。検出力のシナリオ（図 4-11）では、7 割以上の確率で 2 用量が選択されている。

低い用量を選択するルールにおいて、第一種の過誤確率のシナリオ（図 4-10）でも同様に、真値が非ゼロの用量数が増えるにしたがって、少なくとも 1 用量選択される確率が増えている。検出力のシナリオ（図 4-12）では、高い用量を選択するルールと比較して、2 用量が選択される割合が低く、5 割前後にとどまっている。

なお、前述のとおり、「選択条件を満たす最低用量」として選択される割合が両ルールで同一のため、無効中止（選択用量数 0）が発生する割合も両ルールで同一となっている。

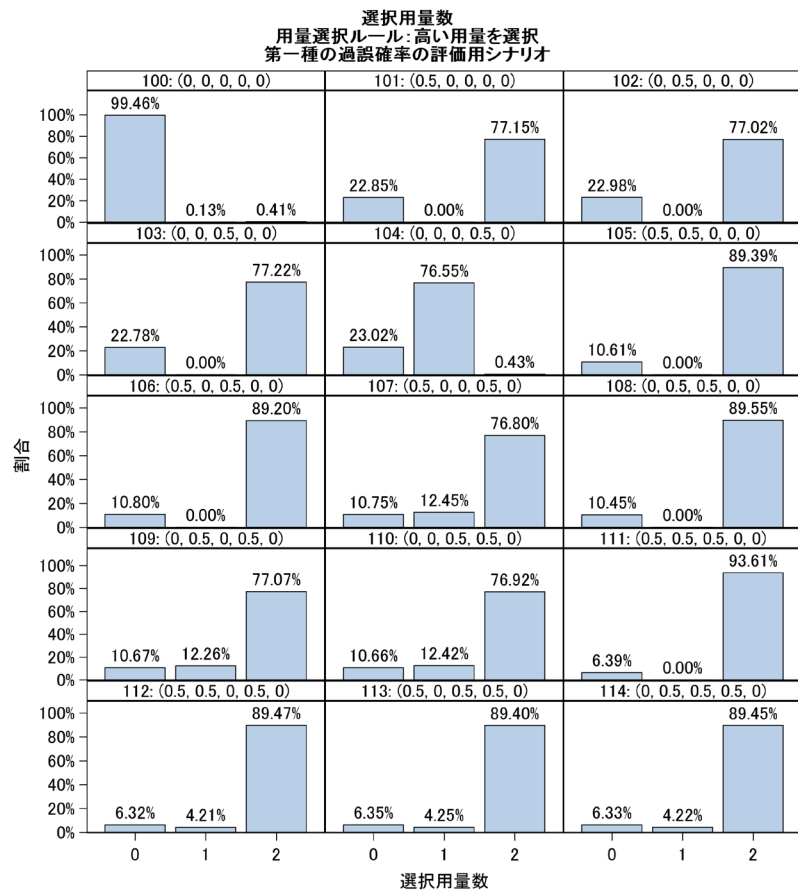


図 4-9 選択用量数

(用量選択ルール：高い用量を選択；シナリオ 100～114)

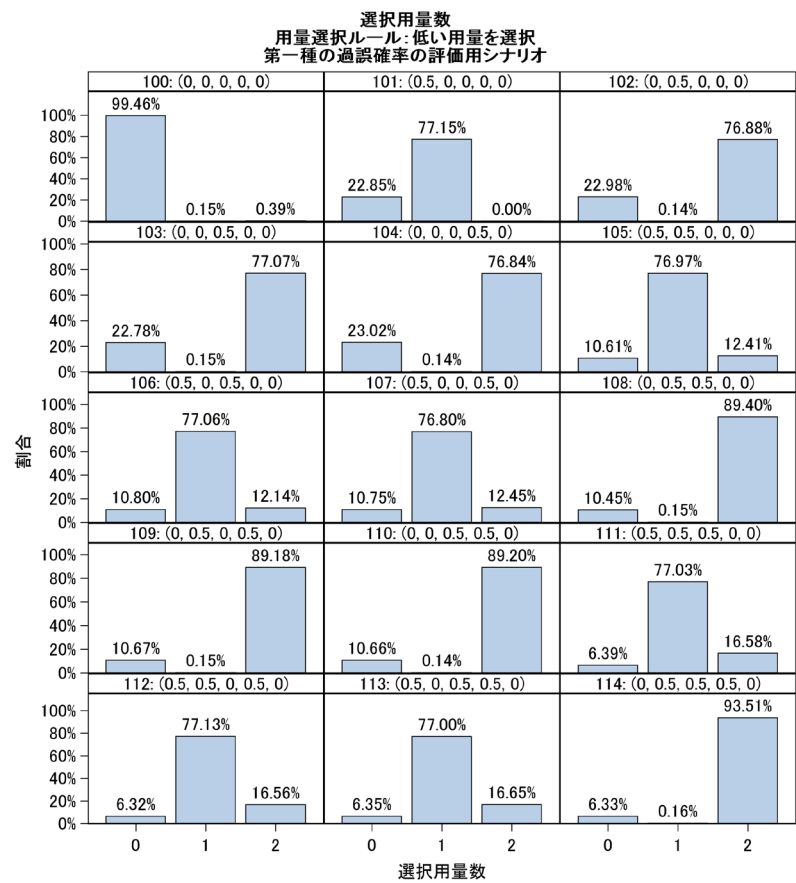


図 4-10 選択用量数

(用量選択ルール：低い用量を選択；シナリオ 100～114)

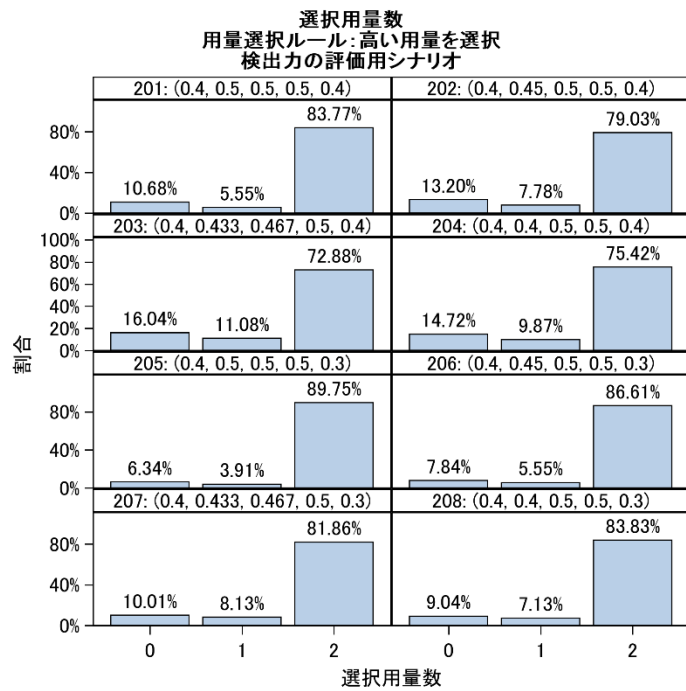


図 4-11 選択用量数

(用量選択ルール：高い用量を選択；シナリオ 201～208)

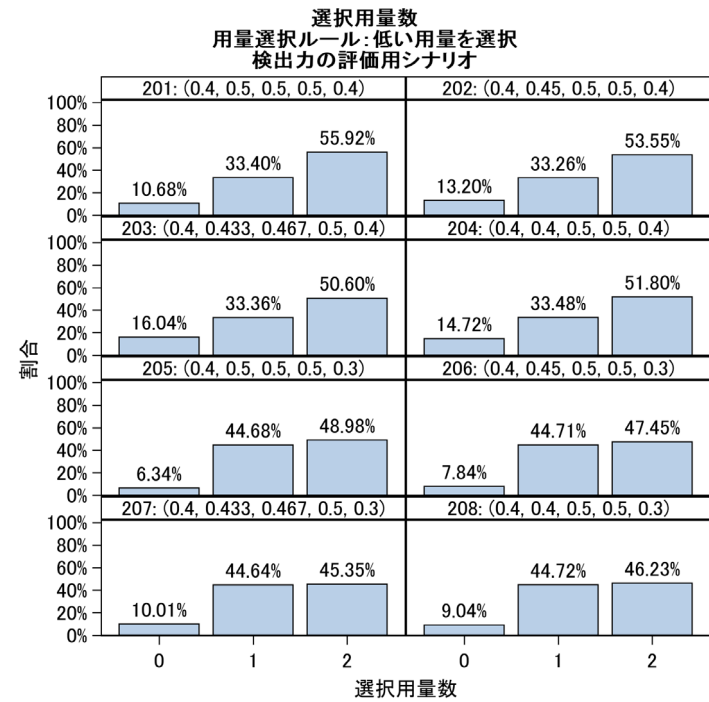


図 4-12 選択用量数

(用量選択ルール：低い用量を選択；シナリオ 201～208)

3.3.3 期待症例数

症例数(両ステージ)の要約統計量を表 4-2 に示した。なお、本セクション以降では検出力のシナリオにおける結果のみを示す。また、いずれの用量選択ルールでも、症例数の最小値は第 1 ステージにおいて 1 用量も選択されなかった場合 (110 例×6 群) であり、最大値は 2 用量ともに選択された場合 (110 例×6 群+230 例×4 群) となる。

高い用量を選択するルールでは、シナリオ 205 にて最も期待症例数が多く (1512.6 例)、シナリオ 203 にて最も期待症例数が少ない (1407.0 例) 結果となった。

低い用量を選択するルールでも同様に、シナリオ 205 にて最も期待症例数が多く (1418.9 例)、シナリオ 203 にて最も期待症例数が少ない (1355.7 例) 結果となった。

いずれの選択ルールにおいても、各群の選択割合および選択用量数を反映している様子が確認された。また、いずれのシナリオにおいても高い用量を選択するルールと比較して低い用量を選択するルールにおいて期待症例数が少なくなっていた。

表 4-2 症例数(両ステージ)の要約統計量

シナリオ No.	回数	症例数 (高い用量を選択)		症例数 (低い用量を選択)	
		平均値	標準偏差	平均値	標準偏差
201: (0.4, 0.5, 0.5, 0.5, 0.4)	10000	1469.0	284.6	1404.9	278.3
202: (0.4, 0.45, 0.5, 0.5, 0.4)	10000	1440.7	310.5	1382.1	300.2
203: (0.4, 0.433, 0.467, 0.5, 0.4)	10000	1407.0	334.2	1355.7	321.1
204: (0.4, 0.4, 0.5, 0.5, 0.4)	10000	1421.9	323.7	1367.6	311.7
205: (0.4, 0.4, 0.5, 0.5, 0.3)	10000	1512.6	226.3	1418.9	226.6
206: (0.4, 0.45, 0.5, 0.5, 0.3)	10000	1495.1	249.2	1405.1	243.7
207: (0.4, 0.433, 0.467, 0.5, 0.3)	10000	1469.2	277.1	1385.2	265.4
208: (0.4, 0.4, 0.5, 0.5, 0.3)	10000	1480.4	265.3	1393.9	256.1

3.4 治療効果の推定

各ステージおよび両ステージにおいて得られる各用量群および陽性対照群の治療効果（プラセボ群との差）の推定値について、バイアス、MSE および信頼区間の被覆確率を評価した結果を示す。

それぞれの条件/ステージ/投与群において、シミュレーション 1 回ごとの治療効果の推定に用いる症例数(n)と、100,000 回のシミュレーションで得られる推定値の個数(N)は以下のとおり。なお、最終解析においては、Inferentially Seamless デザインでは選択された条件付きの両ステージの推定値、Operationally Seamless デザインでは選択された条件付きの第 2 ステージの推定値を使用することを想定している。

	第 1 ステージ	第 2 ステージ	両ステージ
条件なし	n=110 (N=シミュレーション回数)	n=230 (N=選択回数)	n=340 (N=選択回数)、 n=110 (N=非選択回数)
選択された条件付き	n=110 (N=選択回数)	n=230 (N=選択回数)	n=340 (N=選択回数)
選択されなかった条件付き	n=110 (N=非選択回数)	-	-

※陽性対照群の場合は、以下に読み替え

選択回数⇒非無効中止回数（1 用量でも選択された回数）

非選択回数⇒無効中止回数（1 用量も選択されなかった回数）

3.4.1 推定値のバイアスの評価

各ステージおよび両ステージにおける治療効果の推定値のバイアス（条件なし）をそれぞれ図 4-13 および図 4-14 に、第 2 ステージに選択された条件付き、もしくは選択されなかった条件付き（第 1 ステージのみ）のバイアスをそれぞれ図 4-15 および図 4-16 に示した。

高い用量を選択するルールにおいて、条件なしの場合（図 4-13）、第 1 ステージおよび第 2 ステージそれぞれの推定値のバイアスは 0 に近い値となっていた。一方、両ステージの推定値では用量 1,2 の低用量において負のバイアスがわずかに含まれていた。ここで、条件なしの両ステージにおける推定値には、当該群が選択された場合は両ステージの、選択されなかった場合は第 1 ステージの推定値を使用しているため、それぞれの条件付きに分けた結果を確認することにする。

選択された条件付きの場合（図 4-15）、第 1 ステージの用量 1,2 の低用量の推定値に正のバイアスが含まれており、その影響で両ステージの推定値にもバイアス

が含まれている様子が見られた。これは、選択条件（プラセボ群との差が 0.4 よりも大きいかつ陽性対照群との差が 0 より大きい）を満たすのは当該用量群の効果が真値よりも高く、プラセボ群の効果が真値よりも低いためである。最低用量の用量 1 におけるバイアスが他の用量群と比較して大きい、これは用量 1 が選択されるケースは必ず選択条件を満たす最低用量として選択されるため、低い効果が得られた場合は選択されない一方、用量 2~4 については選択条件を満たす最低用量として選択されるケースと、それより 1 つ高い用量として選択されるケースとが混在しているためである。

選択されなかった条件付きの場合（図 4-15）は、選択された条件付きと逆の理由から、第 1 ステージの各用量群の推定値に負のバイアスが含まれていた。シナリオ 201~203 では用量 1 よりも用量 2 の真値が大きいため、選択されなかった条件付きの負のバイアスも用量 2 の方が大きかった。陽性対照群に関しては、シナリオ 205~208 のみ負のバイアスが含まれていたが、これらのシナリオは、用量 1 の効果=陽性対照群の効果であったシナリオ 201~204 と比較して陽性対照群の真値が 0.1 低いため無効中止が起きづらく、選択されなかった（=試験が無効中止した）という状況が起こるのはプラセボ群の効果が特に高い状況のためである。

条件なしの場合（図 4-13）における両ステージの推定値のバイアスにはこれらが反映されたものと考えられた。すなわち、用量 1,2 の低用量においては、選択された条件付きにおいて得られた両ステージの正のバイアスよりも、選択されなかった条件付きにおいて得られた第 1 ステージの負のバイアスの方が絶対値が大きいため負のバイアスとなり、陽性対照群はその逆となったと解釈できる。例えば、シナリオ 201 における用量 1 は 100,000 回のシミュレーションのうち 33,401 回選択され、選択された条件付きの両ステージのバイアスは 0.039、選択されなかった条件付きの第 1 ステージのバイアスは -0.061 から、条件なしの両ステージのバイアスはこれらの重み付き平均として、 $(33,401 \times 0.039 + 66,599 \times (-0.061)) / 100,000 = -0.027$ となる。

低い用量を選択するルールにおける選択用量数について、条件なしの場合（図 4-14）および条件付きの場合（図 4-16）のバイアスは、高い用量を選択するルールの場合と同様の傾向だった。選択されたという条件付きの用量ごとのバイアスに注目すると、最高用量の用量 4 におけるバイアスが大きく、高い用量を選択するルールと同様に、用量 4 が選択されるケースは必ず選択条件を満たす最低用量として選択され、低い効果が得られた場合は選択されないためと考えられた。

また、用量選択ルール間での違いに着目すると、高い用量を選択するルールよりも低い用量を選択するルールにおいてプラセボ群と各用量群の群間差のバイアスが比較的小さいことが見て取れる。これは、本シミュレーション報告書では用量依存的に治療効果が単調増加（非減少）するシナリオのみ設定していたため、低い用

量を選択するルールの方が入りうるバイアスの程度が小さいことに由来していると考えられた。

3.4.2 MSE の評価

各ステージおよび両ステージにおける治療効果の推定値の MSE をそれぞれ図 4-17 および図 4-18 に、第 2 ステージに選択された条件付き、もしくは選択されなかった条件付き（第 1 ステージのみ）の MSE をそれぞれ図 4-19 および図 4-20 に示した。

高い用量を選択するルールにおいて、条件なしの場合と比較して、選択されたという条件付きの場合において、両ステージの MSE が小さくなっている様子が見られた。

低い用量を選択するルールにおいても同様の傾向が見られた。

3.4.3 信頼区間の被覆確率の評価

各ステージおよび両ステージにおける治療効果の推定値の信頼区間の被覆確率をそれぞれ図 4-21 および図 4-22 に、第 2 ステージに選択された条件付き、もしくは選択されなかった条件付き（第 1 ステージのみ）の信頼区間の被覆確率をそれぞれ図 4-23 および図 4-24 に示した。

高い用量を選択するルールにおける信頼区間の被覆確率について、条件なしの場合ではすべてのシナリオにおいて名義水準の 95% 付近の値を示していた一方、第 2 ステージに選択されたという条件付きの場合、一部のシナリオ・群において名義水準から離れているケースが見受けられた。また、第 2 ステージに選択されなかった条件付きの場合、陽性対照群では名義水準を大きく下回っている様子が見られた。

低い用量を選択するルールにおいても同様の傾向が見られた。

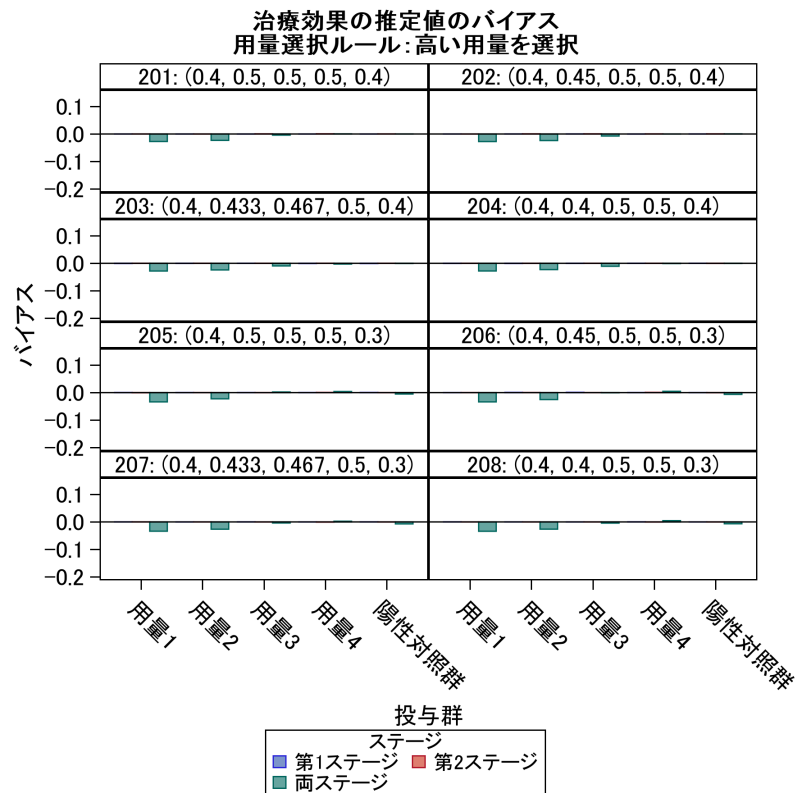


図 4-13 バイアス (用量選択ルール: 高い用量を選択)

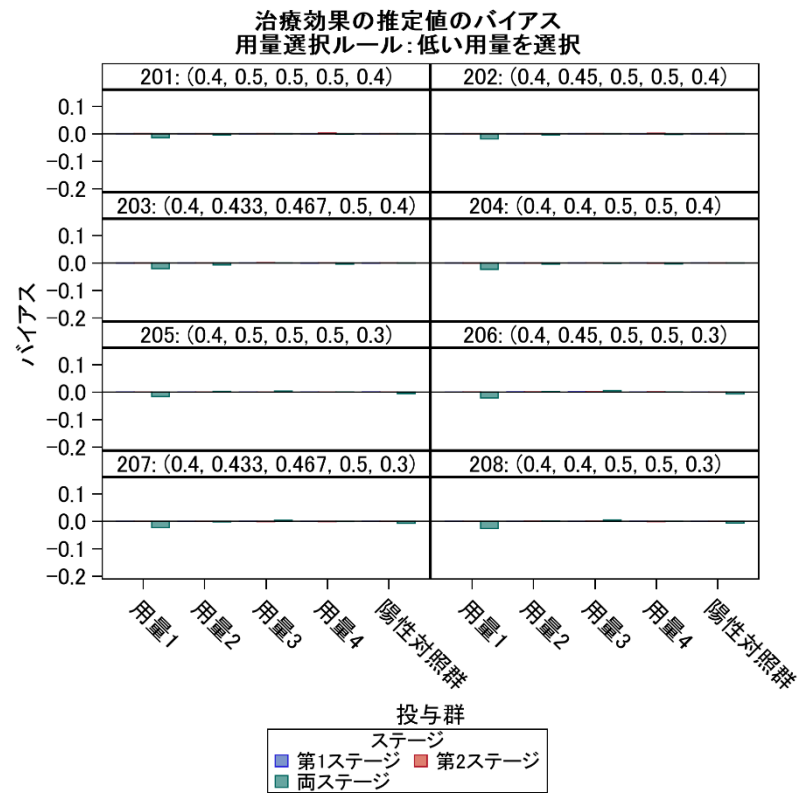


図 4-14 バイアス (用量選択ルール: 低い用量を選択)

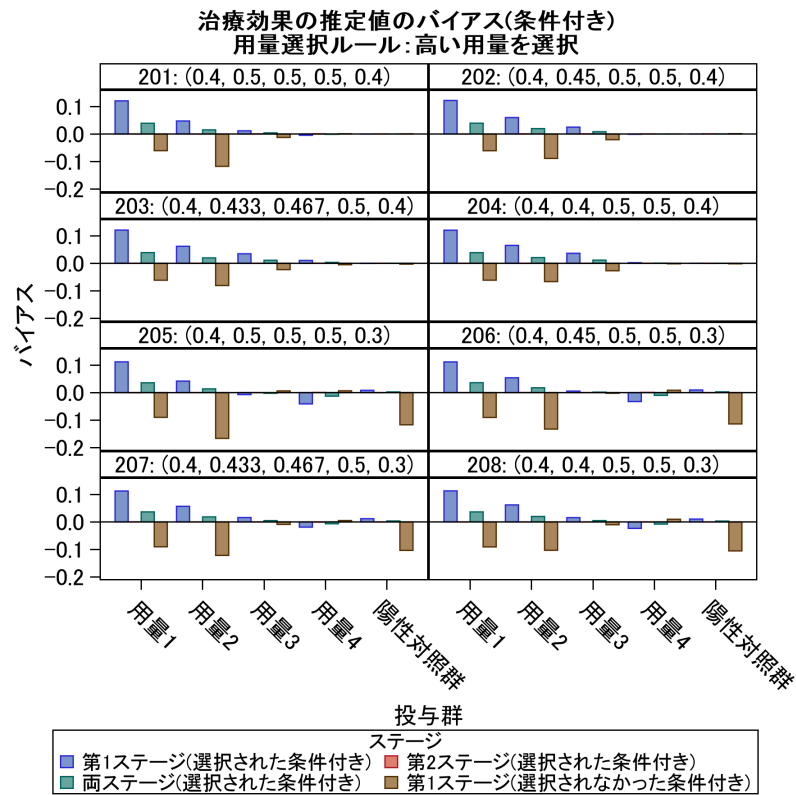


図 4-15 バイアス (条件付き ; 用量選択ルール : 高い用量を選択)

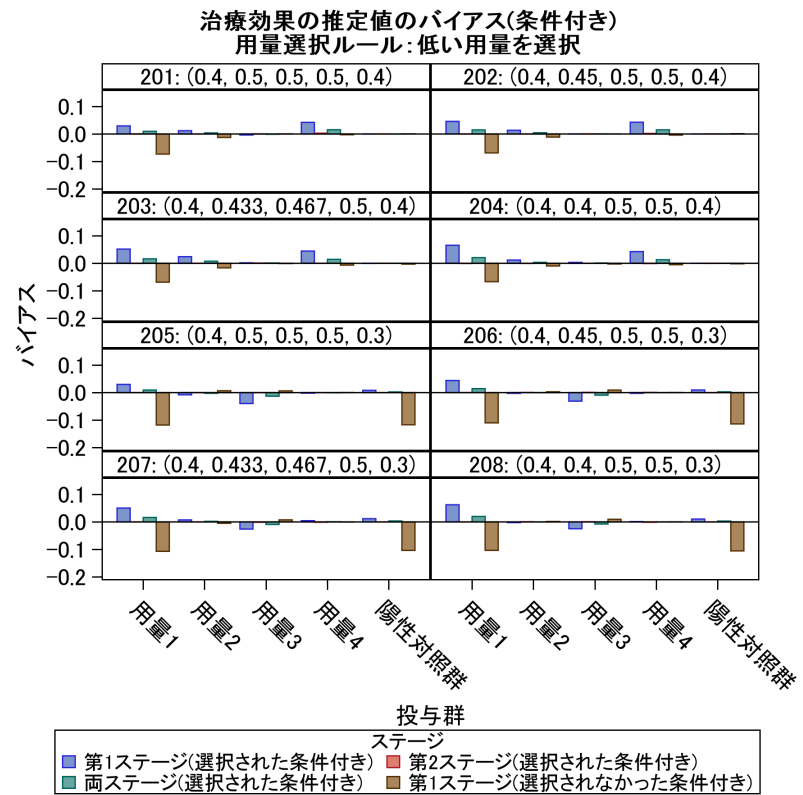


図 4-16 バイアス (条件付き ; 用量選択ルール : 低い用量を選択)

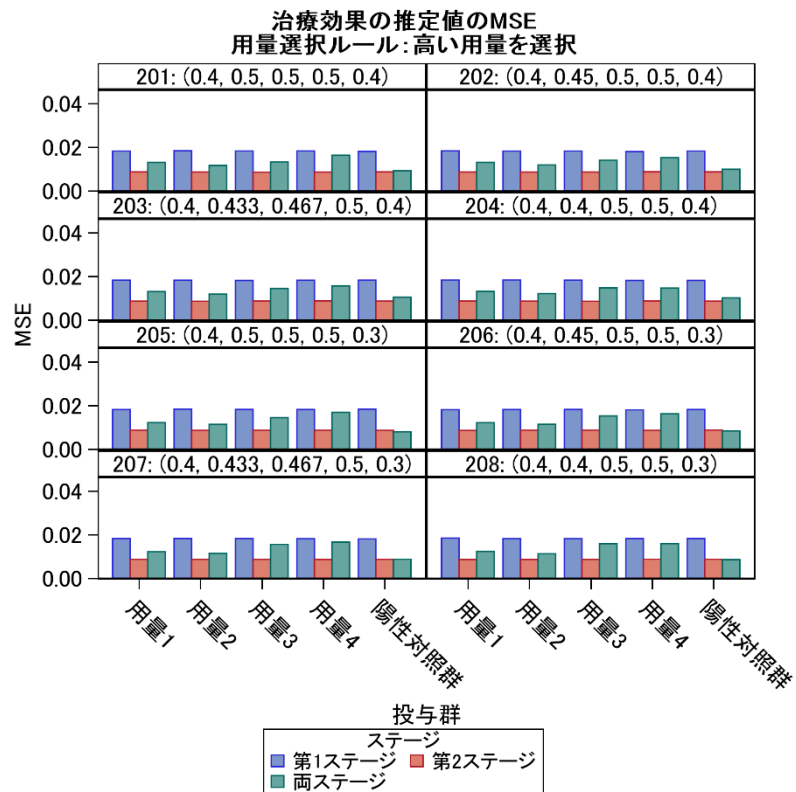


図 4-17 MSE (用量選択ルール: 高い用量を選択)

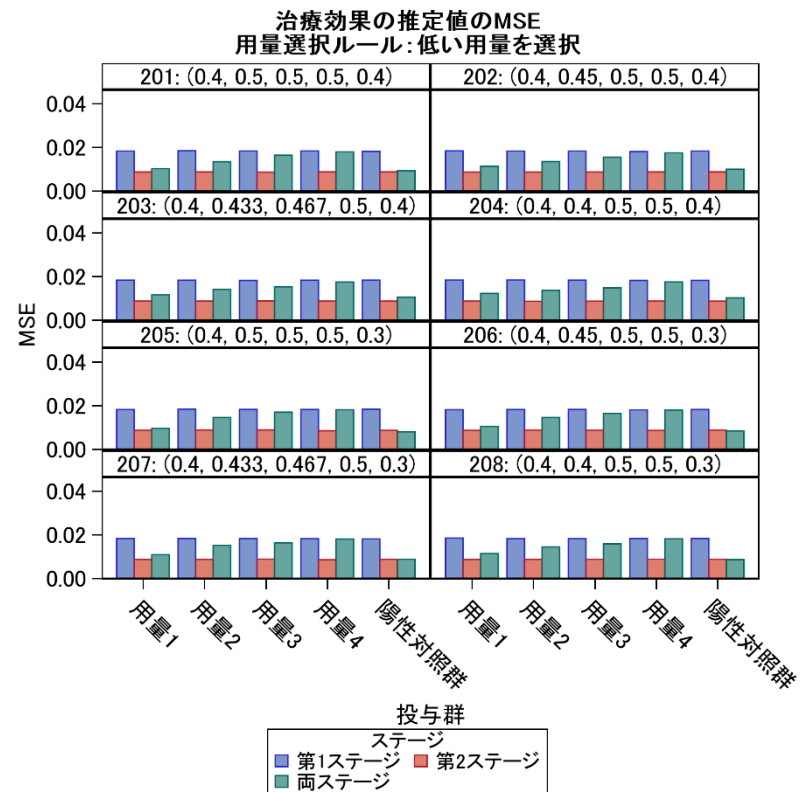


図 4-18 MSE (用量選択ルール: 低い用量を選択)

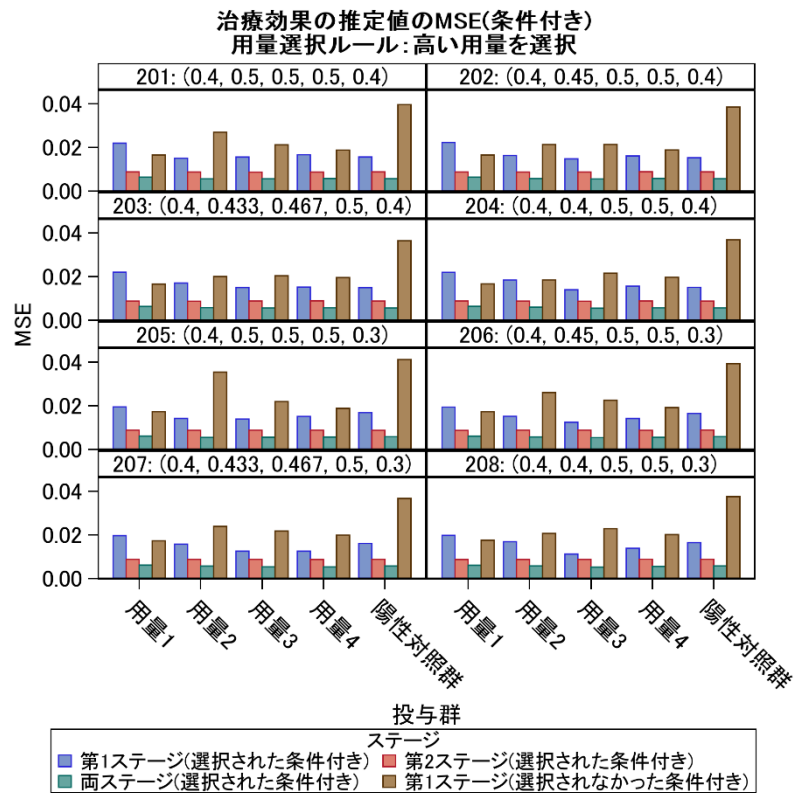


図 4-19 MSE (条件付き ; 用量選択ルール : 高い用量を選択)

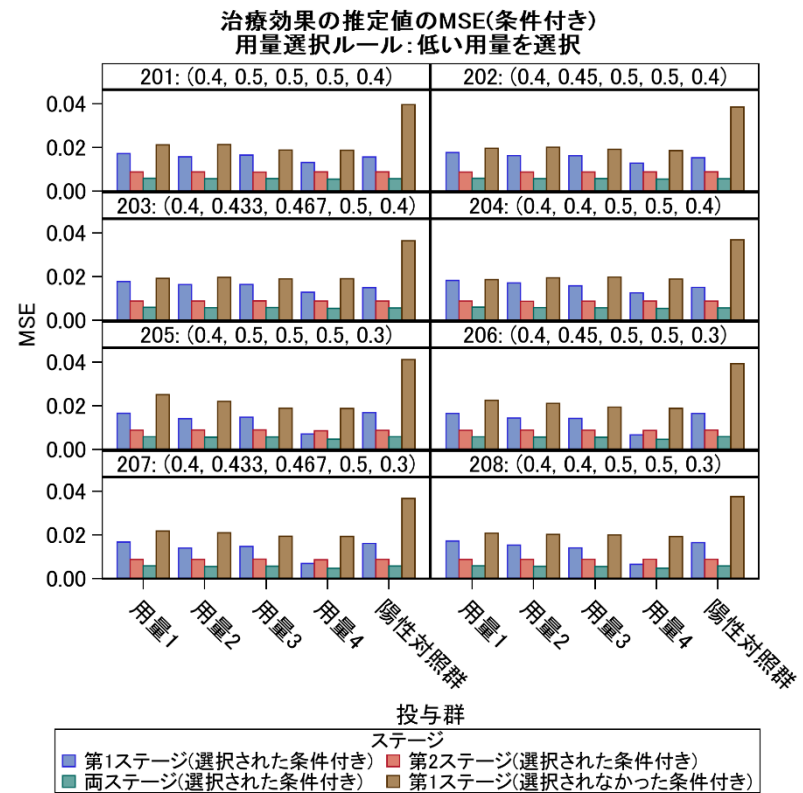


図 4-20 MSE (条件付き ; 用量選択ルール : 低い用量を選択)

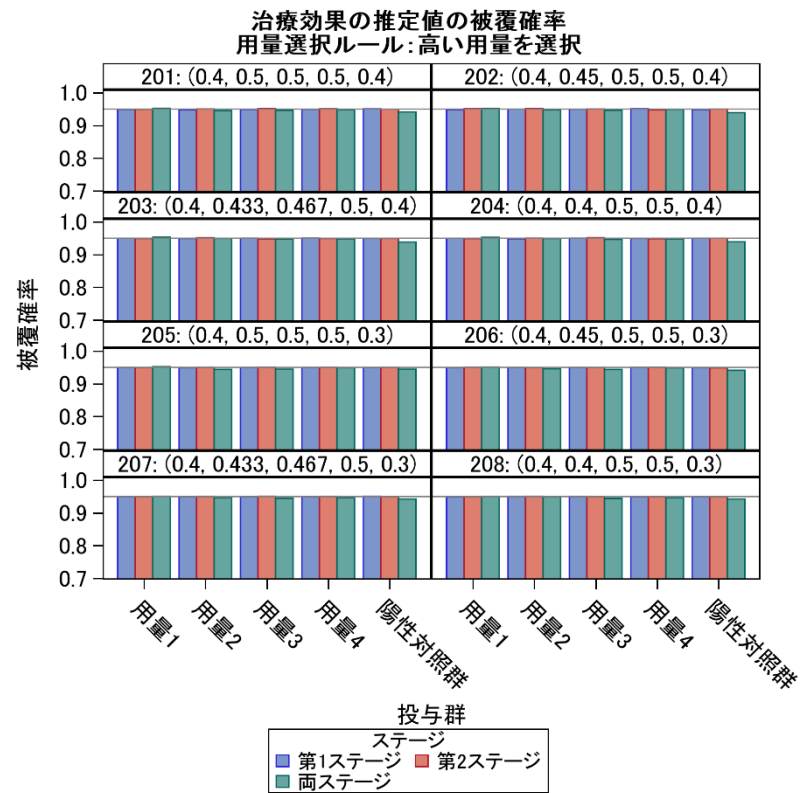


図 4-21 被覆確率 (用量選択ルール: 高い用量を選択)

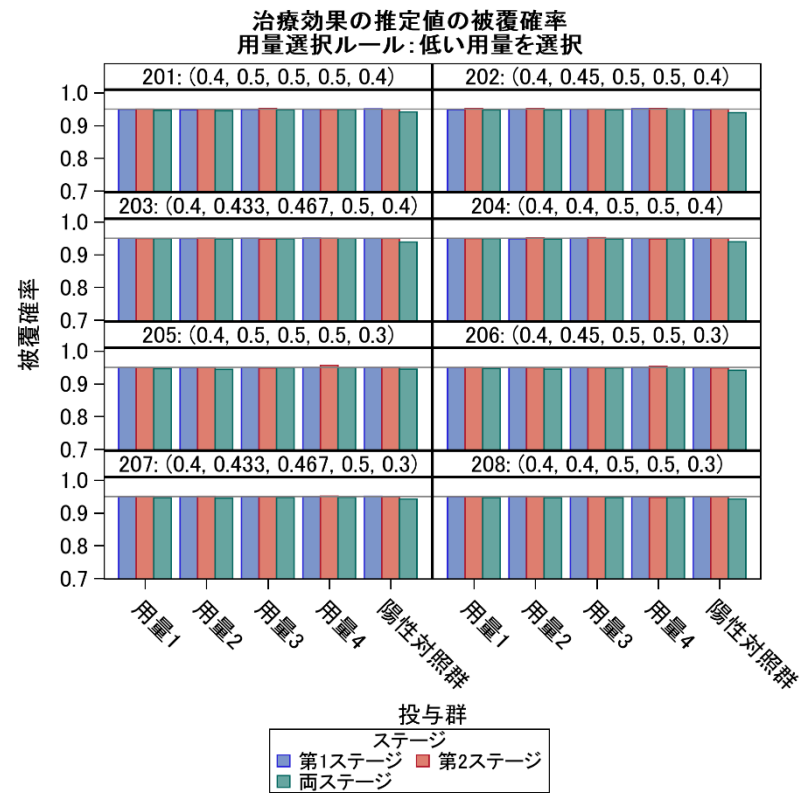


図 4-22 被覆確率 (用量選択ルール: 低い用量を選択)

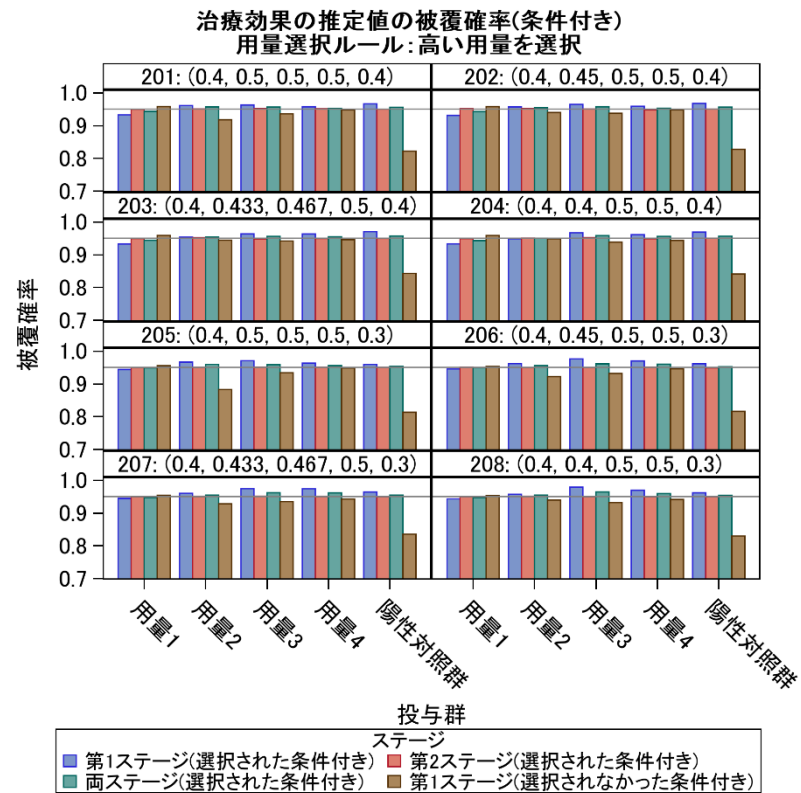


図 4-23 被覆確率 (条件付き ; 用量選択ルール : 高い用量を選択)

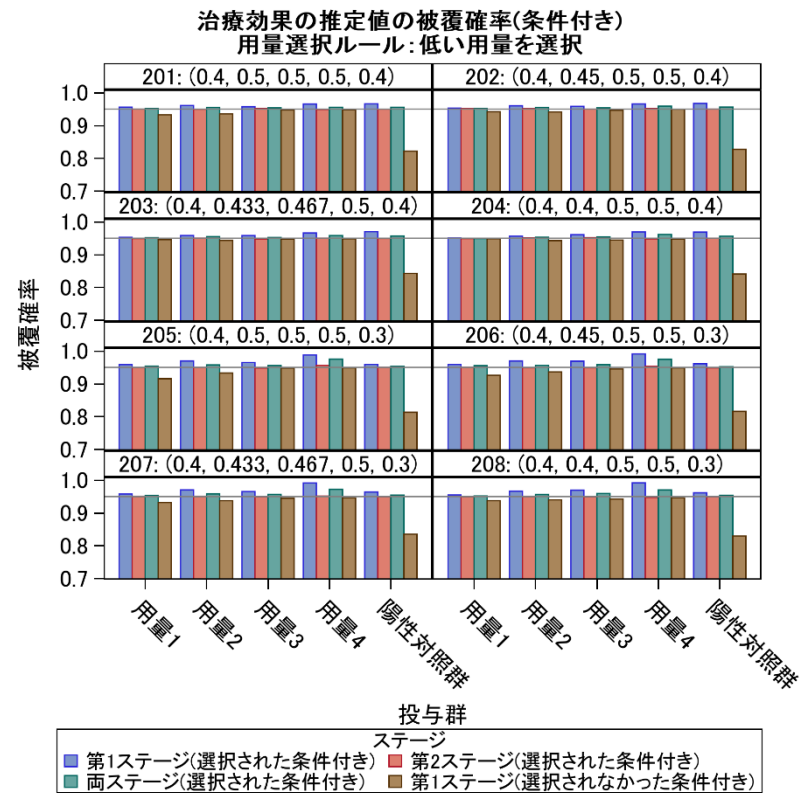


図 4-24 被覆確率 (条件付き ; 用量選択ルール : 低い用量を選択)

4 要約及び推奨事項

4.1 考察

第一種の過誤確率はいずれの統計手法も名義水準以内に制御されていた。検出力の観点からは **Inferentially Seamless** と **Operationally Seamless** とでほぼ同一であったことから、運営上のコスト等も考慮し、**Operationally Seamless** デザインを選択する。

治療群の選択ルールについては、**Operationally Seamless** の最終解析において使用する、選択されたという条件付きの第 2 ステージの推定値に対するバイアスは両ルールともにほぼ 0 であり、ルール間で差異がなかった。その一方で、高い用量を選択するルールの方が 2 用量を第 2 ステージに進める割合が高かったことから、高い用量を選択するルールを用いることとする。

4.2 推奨されるデザイン

- デザイン： **Operationally Seamless** デザイン
- 用量選択ルール：選択条件を満たす最低用量と、それより 1 つ高い用量の 2 群を選択するルール

5 参考文献

※参考文献リストについては本成果物の末尾にまとめて記載した。

6 付録

6.1 バリデーション

シミュレーションコードを別添に添付する。シミュレーションの結果は、独立した統計担当者が SAS9.4 を用いて独立したプログラムを作成し検証を行った。検証プログラムの本シミュレーション報告書への添付は割愛する。検証プログラムは社内フォルダに保管した。

6.2 計算環境またはその他の技術的詳細

シミュレーション実行ソフトウェアとして、SAS 9.4TS1M7 を使用した。

以上が、シミュレーション報告書の例である。以降は、本章のシミュレーション報告書のスコープ外ではあるが、本タスクフォースにて複数の検定方法の動作特性を検討した内容を紹介する。

検定方法の検討

Inferentially Seamless デザインの最終解析における検定方法として、①逆正規法、②Step-

down Dunnett 検定の拡張および③adaptive Dunnett 検定の 3 つの方法を用いてシミュレーションにて動作特性を検討した。なお、シミュレーションの実行には、SAS に加えて R を利用した。

各検定手法の概要およびソフトウェアでの実装方法について簡単に紹介する。

① 逆正規法

逆正規法は、各ステージから得られる独立した p 値を結合して検定を行う手法である^{[16][17]}。2 ステージデザインの場合において、逆正規法の検定統計量 Z_{INV} は以下の式で表され、標準正規分布に従う。

$$Z_{INV} = w_1 \Phi^{-1}(1 - p_1) + w_2 \Phi^{-1}(1 - p_2)$$

ここで、 $\Phi^{-1}(\cdot)$ は標準正規分布の分布関数の逆関数、 p_1 および p_2 は各ステージのデータから得られる p 値である。 w_1 および w_2 は 2 つの各ステージの重みであり、 $w_1^2 + w_2^2 = 1$ を満たす範囲で自由に設定できる。各ステージで等しい重みとする $w_1 = w_2 = \frac{1}{\sqrt{2}}$ という設定や、群ご

との各ステージの症例数 n_1 および n_2 を重みに用いる $w_1 = \sqrt{\frac{n_1}{n}}$, $w_2 = \sqrt{\frac{n_2}{n}}$ ($n = n_1 + n_2$) という設定が考えられる。逆正規法に基づく最終解析の p 値は $p_{INV} = 1 - \Phi(Z_{INV})$ で得られ、この p 値を有意水準と比較して検定を行う。正規分布に従うエンドポイントだけでなく、二値や生存時間などあらゆるエンドポイントに対して適用可能な手法である。治療群の選択を伴うデザインでは、複数の治療群とプラセボ群の比較のように多対一の比較を行う状況が想定され、適切に多重性を調整したステージごとの調整 p 値を逆正規法の導出に用いる必要がある。

SAS で逆正規法を実装する場合、MULTTEST プロシジャなどを用いて得られた多重調整されたステージごとの p 値を、データステップで統合することで逆正規法の p 値を算出できる。R では、rpact パッケージの getDesignInverseNormal() 関数および getAnalysisResults() 関数を用いて逆正規法の解析を実施できる。getAnalysisResults() 関数の引数 intersectionTest=において、ステージごとの p 値に対する多重調整の方法（積仮説の検定方法）を選択でき、Bonferroni 法、Sidak 法、Simes 法、Dunnett 法、階層的検定手順の 5 つの手法が利用可能である。

② Step-down Dunnett 検定の拡張

Step-down Dunnett 検定は、多対一の比較において用いられるシングルステップの Dunnett 検定^[19]を閉検定手順によって改良した手法である^{[20][21]}。Dunnett 検定と同様に、連続量のエンドポイントに対してのみ適用可能な手法である。

例として、試験治療群 3 用量（低、中、高用量群）とプラセボ群の 4 群の固定デザイン（治療群選択なし）を考える。各群の評価項目の平均値をそれぞれ $\mu_L, \mu_M, \mu_H, \mu_P$ とする。3 つの基本帰無仮説（elementary null hypotheses）をそれぞれ H_L, H_M, H_H とし、 $H_L : \mu_L = \mu_P, H_M :$

$\mu_M = \mu_P, H_H : \mu_H = \mu_P$ である。さらに、複数の帰無仮説の共通部分である積帰無仮説 (intersection null hypothesis) として $H_{LM}, H_{LH}, H_{MH}, H_{LMH}$ を考える ($H_{LM} : \mu_L = \mu_M = \mu_P, H_{LH} : \mu_L = \mu_H = \mu_P, H_{MH} : \mu_M = \mu_H = \mu_P, H_{LMH} : \mu_L = \mu_M = \mu_H = \mu_P$)。ここで、例えば低用量群のプラセボ群に対する優越性に興味があるとして、基本帰無仮説 H_L を棄却するためには、 H_L が属する全ての積帰無仮説 (H_{LM}, H_{LH}, H_{LMH}) を棄却する必要がある。最初に、 H_{LMH} について 3:1 の Dunnett 検定を有意水準 α にて行う。3 用量のいずれも有意でなかった場合、 H_{LMH} を棄却できずに手順を終了する。3 用量のいずれかが有意であった場合、ひとつ下の積仮説 H_{LM} および H_{LH} について、2:1 の Dunnett 検定を有意水準 α にて行う。 H_{LM} と H_{LH} のいずれも棄却された場合、基本帰無仮説 H_L について、t 検定を有意水準 α にて行う。

Step-down Dunnett 検定は固定デザインの統計手法であるが、治療群の選択を伴うアダプティブデザイン (Inferentially Seamless) においては、第 1 ステージで選択されなかった群と対照群の差の検定統計量を $-\infty$ と置き換えることで適用可能である^{[22][23]}。これにより、実際に第 2 ステージに進んだ群の数よりも多い数での対比を行うこととなり、検定の棄却限界値が大きくなるため、ファミリーワイズエラーを名義水準 α 以下に保つことが出来る。そのため、中間解析にて脱落した治療群が多いほど、保守的な結果を与える統計手法である。

SAS では、orthoreg プロシジャを利用して Step-down Dunnett 検定による解析が可能である。治療群の選択を伴うアダプティブデザイン (Inferentially Seamless) では、あらかじめデータステップにて選択されなかった治療群の第 1 ステージにおけるエンドポイント変数に充分大きな数を減算 (小さいほど改善を意味するエンドポイントの場合は加算) しておくことで適用可能である。R では、multcomp パッケージの glht() 関数で Step-down Dunnett 検定が実施できる。治療群の選択を伴うアダプティブデザイン (Inferentially Seamless) へ拡張する方法は SAS と同様である。

③ 両ステージのデータを用いた Adaptive Dunnett 検定

Adaptive Dunnett 検定^[22]は、step-down Dunnett 検定に、第 1 ステージのデータを用いて条件付き過誤確率を考慮する検定方法で、R のパッケージ rpact で実装可能である。Adaptive Dunnett 検定は積仮説の検定に脱落した第 1 ステージのデータも用いるため、中間解析で治療群の脱落が起きた場合、Step-down Dunnett 検定よりも性能が改善されている。Adaptive Dunnett 検定の詳細は、2022 年に本タスクフォースから発出された「アダプティブデザインの統計的推測に関する検討」の 3.5.1 項を参照されたい。

上記の手法について、シミュレーションに基づき検出力の比較を行った。①の逆正規法については SAS および rpact の両方で実施した。なお、いずれもステージごとの多重調整の手法として Bonferroni 法を用いているが、SAS 側はシングルステップの Bonferroni 法を用いている一方、rpact は閉検定手順における積仮説の検定において Bonferroni 型の調整を行っており、これは Holm 法に相当することから、両方で多重調整の手法が異なる点に注意され

たい。②の Step-down Dunnett 検定の拡張は SAS で、③の Adaptive Dunnett 検定は rpact にて評価を行った。

4 種類の検定方法の検出力を、2 つの用量選択ルール別にそれぞれ図 4-26 および図 4-25 に示した。高い用量を選択するルールにおける用量群全体の検出力（効果のあるいずれかの用量が選択され、かつ、プラセボ群との有意差が得られる確率）は、4 種類いずれの検定においてもシナリオ 201～208 の全てで検出力が 80%を超えていた。手法間の差異はほとんどなかった。これは、本シミュレーションで設定した効果が大きく、第 2 ステージに進んだ場合の検出力（選択されたという条件付きの検出力）が 100%近くであることが理由であると考えられた。低い用量を選択するルールにおいても、用量群全体の検出力では同様の傾向が見られた。群ごとの検出力では、用量選択ルールを反映して低い用量における検出力が高くなる傾向が見られた。

治療効果の仮定 (2.4 節)

本シミュレーション報告書では登録スピードの検討を省いたが、実際には状況に応じて様々な登録シナリオを設定することが考えられる。

その他の臨床上の仮定 (2.5 節)

本シミュレーション報告書では被験者の登録スピードの影響や脱落は考慮しなかった。これらを考慮する場合には、対応するパラメータを設定することが考えられる。

評価指標 (2.6 節)

本シミュレーション報告書では、治療効果の推定に対して補正などを行わない未調整の方法を用いた。補正が可能な場合には、補正後の推定値でバイアス、MSE、信頼区間の被覆確率の評価を行うことが考えられる。

第一種の過誤確率 (3.1 節)

本シミュレーション報告書において検討した手法では、Inferentially Seamless デザインにおける第一種の過誤確率が名義水準を大きく下回っており、過度に保守的となっていた。より名義水準に近い第一種の過誤確率とする方法のひとつとして、Adaptive Dunnett 検定の利用が考えられる。

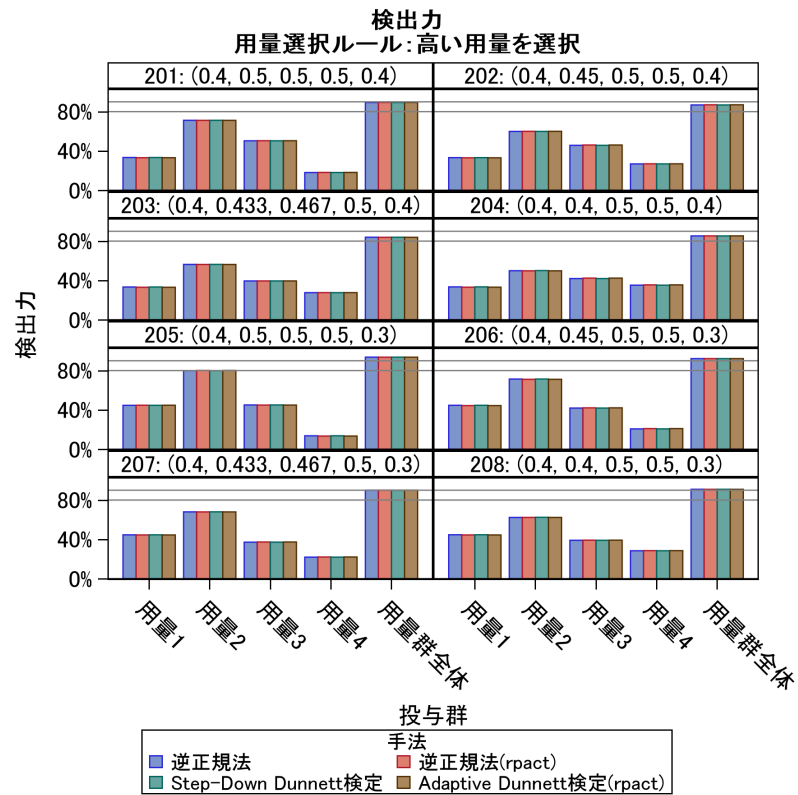


図 4-26 検出力

(4 手法の比較 ; 用量選択ルール : 高い用量を選択)

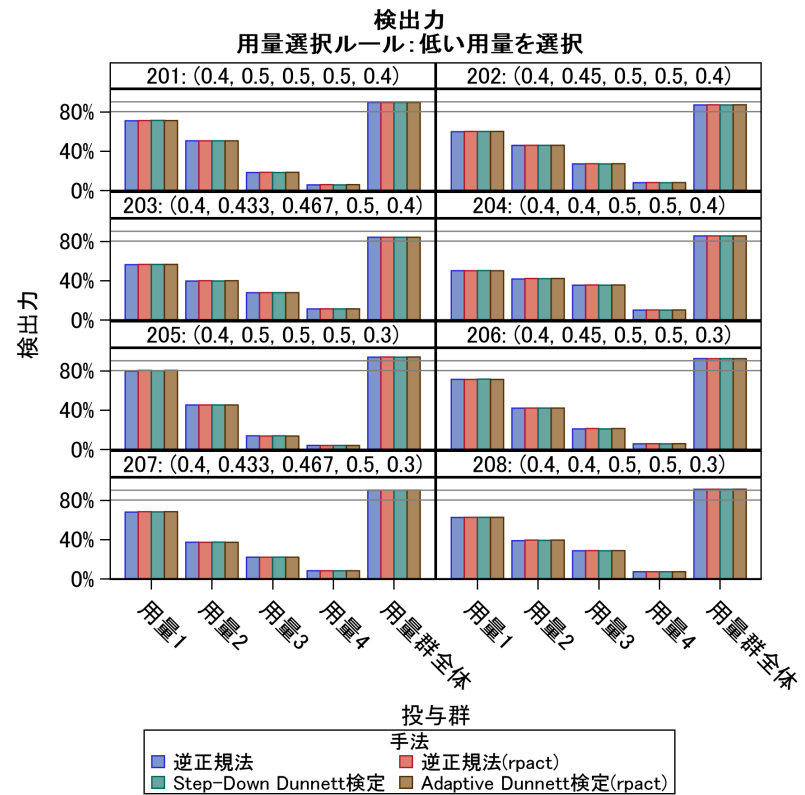


図 4-27 検出力

(4 手法の比較 ; 用量選択ルール : 低い用量を選択)

5 おわりに

本成果物は、アダプティブデザインの動作特性を確認するためのシミュレーションに焦点を当て、シミュレーションの計画、実施、方法をまとめた。さらに、過去に実施された2つの臨床試験を想定した仮想的なシミュレーション報告書を統計家の側面から作成した。しかし、実際には統計的な側面だけでなく、安全性などの臨床的背景、試験の運営や試験期間、コストやリソース等を踏まえて、適用する試験デザインやサンプルサイズを決定する。シミュレーション報告書は、適用する状況に応じて統計以外の側面も考慮して作成されることが現実には考えられるため、もし3章と4章に似たケースのアダプティブデザインを検討する場合であっても、シミュレーション報告書に示した結論が常に正しいとは限らないことに留意いただきたい。例えば症例数再推定を伴うアダプティブデザインでは、疾患の罹患者数、試験実施期間、製剤製造する部門及び開発・薬事戦略等を配慮し、実現可能な被験者の上限が必然と決まってくるものが考えられる。治療群選択をとるアダプティブデザインでは、群の選択ルールを決める際に、有効性以外の臨床的側面（例えば、安全性）を考慮に入れて選択ルールを決めることが考えられる。仮に4章のシミュレーション報告書の例にて、安全性が用量依存的に大きく損なわれてしまう可能性があれば、高い用量ではなく低い用量を選択するルールを適用することが考えられる。

アダプティブデザインに関する規制当局との議論を円滑に進めるためには、本成果物に示すシミュレーション報告書を事前に作成しておくことが役立つと考えられる。しかし、アダプティブデザインは固定デザインと比較して試験デザインの検討に時間がかかる。手順3にあたる計算コードの作成から完成、計算の実施はもちろん、手順1-2での検討段階でも多くの時間を要する。また、規制当局との議論を進める中にシミュレーションの再実行が必要となる状況が考えられる。シミュレーションを速やかに再実行可能な計算環境を準備しておくだけでなく、計算からデザインを検討するまでの作業時間も見積もり、規制当局とのコミュニケーションをしていくことが望ましい。なお、本シミュレーション報告書の数値の部分を作成する完成している計算コードを実行するだけで、筆者たちの環境

(OS: Windows 10; CPU: Intel Core i5-1235U 1.30 GHz; RAM: 16 GB) で約20時間(約25分/シナリオ×シナリオ23通り×用量選択ルール2通り)かかる。これは完成したコードを実行する時間であり、コード作成とそれ以前のデザインを検討するディスカッションにより多くの期間を費やした。また、計算結果のバリデーションのために、アダプティブデザインを検討した担当者と別の統計家が、ダブルプログラミング等の確認作業を実施するリソースも必要となってくるかもしれない。

シミュレーション報告書を通じて試験デザインの動作特性を開発担当者間で共有できる事のほかに、規制当局への説明資料へ引用することや、試験結果の公表に伴う学会発表・論文作成などにも参考資料として利用することが想定される。また、試験実施計画書や統計解析計画書にシミュレーション報告書を添付することで、検出力や症例数の設計の根拠

として利用することも考えられる。試験デザインの検討に留まらず、新しい統計手法を用いた既存手法との検出力の比較や第一種の過誤のシミュレーションなどを行った場合は、統計的な観点においても興味深いシミュレーション結果であることが多く、それらの統計関連の学会などへの公表・発信も期待したい。

最後に、アダプティブデザインを適用するには、デザインや統計手法だけでなく試験運用上の工夫も必要である。アダプティブデザインの検討は統計担当者だけでできることではなく、臨床、データマネジメント、薬事、治験薬製造など、様々な部門の協力が重要である。アダプティブデザインの検討を担う統計担当者には、統計手法の検討だけでなく、シミュレーションの計算環境の整備、関連部署や規制当局とコミュニケーションなど、これまで以上に多面的なスキルが求められるため、統計担当者の育成も重要である。

参考文献

- [1] European Medicines Agency. Reflection Paper on Methodological Issues in Confirmatory Clinical Trials Planned with an Adaptive Design (CHMP/EWP/2459/02) 2007. https://www.ema.europa.eu/en/documents/scientific-guideline/reflection-paper-methodological-issues-confirmatory-clinical-trials-planned-adaptive-design_en.pdf [2023年7月31日アクセス]
- [2] Food and Drug Administration. Adaptive Design Clinical Trials for Drugs and Biologics Guidance for Industry. <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/adaptive-design-clinical-trials-drugs-and-biologics-guidance-industry> [2023年7月31日アクセス]
- [3] 日本製薬工業協会, アダプティブデザインに関するFDAガイダンスの邦訳, 2021年8月. https://www.jpma.or.jp/information/evaluation/results/allotment/lofurc000000n5a8-att/adaptive_design.pdf [2024年1月25日アクセス]
- [4] 小宮山 靖, 越水 孝, 菅波 秀規, 酒井 弘憲, 渡橋 靖, 東宮 秀夫. (2009) 医薬品の臨床開発におけるアダプティブ・デザイン —米国研究製薬工業協会ワーキング・グループのエグゼクティブ・サマリー邦訳—. 臨床薬理. 40巻6号 p. 303-310.
- [5] Food and Drug Administration. Interacting with the FDA on Complex Innovative Trial Designs for Drugs and Biological Products. <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/interacting-fda-complex-innovative-trial-designs-drugs-and-biological-products> [2024年1月24日アクセス]
- [6] Mayer C, Perevozskaya I, Leonov S, Dragalin V, Pritchett Y, Bedding A, Hartford A, Fardipour P, Cicconetti G. Simulation Practices for Adaptive Trial Designs in Drug and Device Development. *Statistics in Biopharmaceutical Research* 2019;11(4):325-335.
- [7] Marconi VC, Ramanan AV, de Bono S, et al. Efficacy and safety of baricitinib for the treatment of hospitalised adults with COVID-19 (COV-BARRIER): a randomised, double-blind, parallel-group, placebo-controlled phase 3 trial [published correction appears in *Lancet Respir Med*. 2021 Oct;9(10):e102]. *Lancet Respir Med*. 2021;9(12):1407-1418.
- [8] Lawrence D, Bretz F, Pocock S. INHANCE: An adaptive confirmatory study with dose selection at interim. In *Indacaterol - The First Once-Daily Long-Acting Beta2 Agonist for COPD*, Trifilieff A (ed.) Springer: Basel, 2014; 77-93.
- [9] 日本製薬工業協会, アダプティブデザインの統計的推測に関する検討, 2022年2月. https://www.jpma.or.jp/information/evaluation/results/allotment/jtrngf0000001cuh-att/DS_202302_adaptive.pdf [2024年1月24日アクセス]
- [10] Cui L, Hung, HMJ, Wang SJ. Modification of sample-size in group sequential trials. *Biometrics* 1999;55:853-857.
- [11] Mehta CR, Pocock SJ. Adaptive increase in sample size when interim results are promising: a

- practical guide with examples. *Statistics in Medicine* 2011;30(28):3267-3284
- [12] O'Brien PC, Fleming TR. A multiple testing procedure for clinical trials. *Biometrics* 1979;35:549-612
- [13] Kalil AC, Patterson TF, Mehta AK, et al. Baricitinib plus Remdesivir for Hospitalized Adults with Covid-19. *N Engl J Med*. 2021;384(9):795-807. doi:10.1056/NEJMoa2031994
- [14] Jennison C, Turnbull BW. Adaptive sample size modification in clinical trials: start small then ask for more?. *Stat Med*. 2015;34(29):3793-3810
- [15] Liu Y, Xu H. Sample size re-estimation for pivotal clinical trials. *Contemp Clin Trials*. 2021;102:106215
- [16] Chow S-C and Chang M. *Adaptive design methods in clinical trials*. Chapman and Hall/CRC, 2006.
平川 晃弘, 五所 正彦訳. 臨床試験のためのアダプティブデザイン. 朝倉書店. 東京. 2018.
- [17] Food and Drug Administration. Medical Review, APPLICATION NUMBER:022383Orig1s000. https://www.accessdata.fda.gov/drugsatfda_docs/nda/2011/022383Orig1s000MedR.pdf [2024年4月5日アクセス]
- [18] Lehman W, Wassmer G. Adaptive sample-size calculations in group sequential trials. *Biometrics* 1999;55:1286-1290.
- [19] Dunnett CW. A multiple comparison procedure for comparing several treatments with a control. *Journal of the American Statistical Association* 1955;50:1096-1121.
- [20] Dunnett CW, Tamhane AC. Step-down multiple tests for comparing treatments with a control in unbalanced one-way layouts. *Stat Med*. 1991 Jun;10(6):939-47.
- [21] 土居 正明. 多重性制御の基礎理論 (閉検定手順). *計量生物学* Vol. 36, Special Issue, S 99-S 121 (2015).
- [22] Koenig F, Brannath W, Bretz F, Posch M. Adaptive Dunnett tests for treatment selection. *Statistics in Medicine* 2008;10;27(10):1612-25.
- [23] Friede T, Stallard N. A comparison of methods for adaptive treatment selection. *Biometrical Journal* 2008;50(5):767-81.

6 Appendix A: 症例数再推定および早期中止を伴うアダプティブ デザインのプログラムコード.

```
library(rpact)

library(tidyverse)

library(gt)

library(webshot2)

library(cowplot)

library(ggplot2)

#関数 simulations : SSR のシミュレーションを実行する関数

#Nmin:計画された症例数、Nmax : SSR 後の最大症例数、Simnum : シミュレーション回数

#tau : 情報時点 (0~1)、CPcri : Promising Zone の下限となる条件付き検出力 (0~1)

#pi1 : 対照群のイベント発生割合、pi2 : 試験薬群のイベント発生割合

#stpcri : 中間解析時の無効中止の基準となる条件付き検出力 (0~1)

#seed : 乱数のシード

#csvon : 各シミュレーション結果の csvfile での出力の可否、(csvon=1 で seed 名.csv が出力される)

simulations=function(Nmin,Nmax,simnum,tau,CPcri,pi1,pi2,stpcri,seed,csvon){

#関数 myCPZSampleSizeCalculationFunction : Rpact の関数 getSimulationRates で SSR の方法を規定する関数

#各変数(stage, plannedSubjects 等)は getSimulationRates で出力される変数を持ち越している

myCPZSampleSizeCalculationFunction <- function(..., stage,plannedSubjects,conditionalPower,minNumberOfSubjectsPerStage,

maxNumberOfSubjectsPerStage,conditionalCriticalValue,sampleSizesPerStage,

overallRate) {

#イベント数 a,b、群を併合した発現割合 p を計算し、割合の差の検定統計量を testStatistic を計算

a=overallRate[1]*sampleSizesPerStage[1]

b=overallRate[2]*sampleSizesPerStage[2]

p=(a+b)/(sampleSizesPerStage[1]+sampleSizesPerStage[2])

testStatistic=(overallRate[1]-overallRate[2])/sqrt(p*(1-p)*(1/sampleSizesPerStage[1]+1/sampleSizesPerStage[2]))

#検定統計量から条件付き検出力を算出

cpp=round(1-pnorm((qnorm(0.975)*sqrt(minNumberOfSubjectsPerStage[2]+minNumberOfSubjectsPerStage[1])-

testStatistic*sqrt(minNumberOfSubjectsPerStage[1]))/sqrt(minNumberOfSubjectsPerStage[2])-

testStatistic*sqrt(minNumberOfSubjectsPerStage[2])/sqrt(minNumberOfSubjectsPerStage[1])),5)

#関数 calculateStageSubjects : 条件付き検出力 cp を満たすのに必要な症例数を計算
```

```

calculateStageSubjects <- function(cp) {

  subn=(plannedSubjects[1]/testStatistic^2)*((stats::qnorm(0.975)*sqrt(plannedSubjects[2])-testStatistic*sqrt(plannedSubjects[1]))/sqrt(plannedSubjects[2]-
plannedSubjects[1])+stats::qnorm(cp))^2

  return(subn)

}

#条件付き検出力の目標値を 0.9 として必要症例数を算出

stageSubjectsCPmax <- calculateStageSubjects(cp = 0.9)

#必要症例と最大症例数を比較して症例数を決定

stageSubjects <- ceiling(min(max(

  minNumberOfSubjectsPerStage[stage],

  stageSubjectsCPmax

), maxNumberOfSubjectsPerStage[stage]))

#条件付き検出力が Promising Zone の範囲にある場合のみ SSR を実行

if (cpp < CPcri) {

  stageSubjects <- minNumberOfSubjectsPerStage[stage]

}

return(stageSubjects)

}

#n1 : 第一ステージの症例数

n1=ceiling(tau*Nmin)

#中間解析時の無効中止の基準を検定統計量スケールに変換

ZF=sqrt(tau)*qnorm(0.975)+sqrt((1-tau)*tau)*qnorm(stpcr)

#getDesignInverseNormal:逆正規法で群逐次デザインを実行する関数

#https://rdrr.io/cran/rpact/man/getDesignInverseNormal.html

designIN <- getDesignInverseNormal(informationRates = c(tau, 1),

                                typeOfDesign = "asOF",futilityBounds=c(ZF), bindingFutility=FALSE,

                                kMax = 2,alpha = 0.025)

#getSimulationRates : 割合の差に関する、シミュレーションを実行する関数

#https://rdrr.io/cran/rpact/man/getSimulationRates.html

simCpower <- getSimulationRates(designIN,pi1 = c(pi1), pi2 = c(pi2),

                               plannedSubjects = 2 * c(n1, Nmin),

                               conditionalPower = 0.9,

                               minNumberOfSubjectsPerStage = 2 * c(n1, (Nmin - n1)),

                               maxNumberOfSubjectsPerStage = 2 * c(n1, (Nmax - n1)),

                               calcSubjectsFunction = myCPZSampleSizeCalculationFunction,

                               maxNumberOfIterations = simnum, showStatistics = TRUE,seed = seed)

```

```

#シミュレーション結果を代入

data <- getData(simCpower)

#

dataScp=round(1-pnorm((designIN$criticalValues[2]*sqrt(Nmin)-data$testStatisticsPerStage*sqrt(n1))/sqrt(Nmin-n1)-data$testStatisticsPerStage*sqrt(Nmin-
n1)/sqrt(n1)),5)

dataScpm=round(1-pnorm((designIN$criticalValues[2]*sqrt(Nmin)-data$testStatisticsPerStage*sqrt(n1))/sqrt(Nmin-n1)-data$testStatisticsPerStage*sqrt(Nmax-
n1)/sqrt(n1)),5)

#第1ステージでの無効中止の割合

Fst1=format(round(sum(data$futilityPerStage)/simnum,3),nsmall = 3)

#第2ステージでの有意にならなかった割合

Fst2=format(round(sum(1-data[data$stageNumber==2,]$rejectPerStage)/simnum,3),nsmall = 3)

#第1ステージでの有効中止の割合

Ast=format(round(sum(data[data$stageNumber==1,]$rejectPerStage)/simnum,3),nsmall = 3)

#第2ステージでの有意になった割合

st=format(round(sum(data[data$stageNumber==2,]$rejectPerStage)/simnum,3),nsmall = 3)

#試験全体での有意になった割合

Tst=format(round(sum(data$rejectPerStage)/simnum,3),nsmall = 3)

ncol=""

#試験全体での有意にならなかった割合

TFst=format(round((sum(data$futilityPerStage)+sum(1-data[data$stageNumber==2,]$rejectPerStage))/simnum,3),nsmall = 3)

STT=rbind(Tst,Ast,st,TFst,Fst1,Fst2)

#第2ステージまで到達した場合、第1ステージのレコードを削除し、1シミュレーション1レコードとする

data1=(data[data$stageNumber==2 | data$futilityPerStage==1 | (data$stageNumber==1 & data$rejectPerStage==1),])

#シミュレーション結果を csv として出力

if(csvon==1){write.csv(data1, paste(seed,".csv",sep=""))}

#症例数の要約統計量を算出

mean=as.integer(mean(data1$numberOfCumulatedSubjects)/2)

median=as.integer(median(data1$numberOfCumulatedSubjects)/2)

min=min(data1$numberOfCumulatedSubjects)/2

max=max(data1$numberOfCumulatedSubjects)/2

NNN=rbind(paste0(mean,"(",median,")"),paste0(min,"-",max))

#計画された症例数で第2ステージ完了の到達割合

Nonnoninc=round(length(data1[data1$numberOfCumulatedSubjects==n1*2,]$numberOfCumulatedSubjects)/simnum,3)

```

```

#当初の症例数に対して症例数が増加した割合
Noninc=round(length(data1[data1$NumberOfCumulatedSubjects==Nmin*2,]$NumberOfCumulatedSubjects)/simnum,3)

#最大症例数の到達割合
maxinc=round(length(data1[data1$NumberOfCumulatedSubjects==Nmax*2,]$NumberOfCumulatedSubjects)/simnum,3)

inc=format(round(length(data1[data1$NumberOfCumulatedSubjects>Nmin*2,]$NumberOfCumulatedSubjects)/simnum,3),nsmall = 3)

incc=format(rbind(Nonnoninc,Noninc,maxinc),nsmall = 3)

#真のオッズを計算
sinodds=log(1/((pi*(1-pi^2))/((1-pi)*pi^2)))

#対数オッズ比のバイアス、MSE、被覆確率の計算
data1$odds=(data1$overallRate2*(1-data1$overallRate1))/(data1$overallRate1*(1-data1$overallRate2))

data1$sse=sqrt(1/(data1$overallRate2*data1$NumberOfCumulatedSubjects/2)+1/((1-
data1$overallRate1)*data1$NumberOfCumulatedSubjects/2)+1/(data1$overallRate1*data1$NumberOfCumulatedSubjects/2)+1/((1-
data1$overallRate2)*data1$NumberOfCumulatedSubjects/2))

data1$logodds=log(data1$odds)

data1$logsucl=log(data1$odds)+qnorm(0.975)*data1$sse

data1$logslcl=log(data1$odds)-qnorm(0.975)*data1$sse

data1$shifuku=ifelse(data1$logslcl<sinodds & sinodds<data1$logsucl,1,0)

data1$bias=(data1$logodds-sinodds)

data1$mse=(data1$logodds-sinodds)^2

#第1ステージで無効中止の場合のバイアス、MSE、被覆確率
temp=(data1[data1$futilityPerStage==1,])

odds1=format(rbind(round(mean(temp$logodds),3),round(mean(temp$bias),3),round(mean(temp$mse),3),round(mean(temp$shifuku),3)),nsmall = 3)

#第1ステージで有効中止の場合のバイアス、MSE、被覆確率
temp=(data1[data1$stageNumber==1 & data1$rejectPerStage==1,])

odds2=format(rbind(round(mean(temp$logodds),3),round(mean(temp$bias),3),round(mean(temp$mse),3),round(mean(temp$shifuku),3)),nsmall = 3)

#第2ステージまで到達した場合のバイアス、MSE、被覆確率
temp=(data1[data1$stageNumber==2,])

odds3=format(rbind(round(mean(temp$logodds),3),round(mean(temp$bias),3),round(mean(temp$mse),3),round(mean(temp$shifuku),3)),nsmall = 3)

#試験全体のバイアス、MSE、被覆確率
temp=data1

odds4=format(rbind(round(mean(temp$logodds),3),round(mean(temp$bias),3),round(mean(temp$mse),3),round(mean(temp$shifuku),3)),nsmall = 3)

sinodds=format(round(sinodds,3),nsmall = 3)

final=rbind(STT,NNN,incc,inc,sinodds,ncol,odds1,ncol,odds2,ncol,odds3,ncol,odds4)

return(final)
}

```

```

#関数 Main_simulation : simulations を条件を（対立仮説下、対立仮説下（最低差）、帰無仮説下）で実行する関数

#pi1 : 対照群のイベント発生割合、pi2 : 試験薬群のイベント発生割合、pi3 : 試験薬群のイベント発生割合（最低差）

#simnum : シミュレーション回数、tau : 情報時点（0~1）、Nmax : SSR 後の最大症例数、Nmin:計画された症例数

#stpcr: 中間解析時の無効中止の基準となる条件付き検出力（0~1）、CPeri : Promising Zone の下限となる条件付き検出力（0~1）

#tablename:出力ファイル名

Main_simulation=function(pi1,pi2,pi3,simnum,tau,Nmax,stpcr,CPeri,tablename){

#getDesignGroupSequential : SSR なし群逐次デザインを実行する関数

#https://rdrr.io/cran/rpact/man/getDesignGroupSequential.html

designpre <- getDesignGroupSequential(sided = 1, alpha = 0.025, beta = 0.1,informationRates = c(tau, 1), typeOfDesign = "asOF")

#中間解析時の無効中止の基準を検定統計量スケールに変換

ZF=sqrt(tau)*designpre$criticalValues[2]+sqrt((1-tau)*tau)*qnorm(stpcr)

#計画された症例数の計算

design <- getDesignGroupSequential(sided = 1, alpha = 0.025, beta = 0.1,informationRates = c(tau, 1), typeOfDesign = "asOF",futilityBounds=c(ZF),
bindingFutility=FALSE)

sampleSizeResultGS <- getSampleSizeRates(design, pi2 = 0.25, pi1 = 0.175)

Nmin=ceiling(sampleSizeResultGS$maxNumberOfSubjects/2)

#対立仮説下 SSR あり

temp1=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=tau,CPeri=CPeri,

pi1=pi1,pi2=pi2,stpcr=stpcr,seed=1192,csvon=1)

#対立仮説下 SSR なし

temp2=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=tau,CPeri=1.3,

pi1=pi1,pi2=pi2,stpcr=stpcr,seed=794,csvon=1)

#対立仮説下（最低差）SSR あり

temp3=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=tau,CPeri=CPeri,

pi1=pi1,pi2=pi3,stpcr=stpcr,seed=710,csvon=1)

#対立仮説下（最低差）SSR なし

temp4=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=tau,CPeri=1.3,

pi1=pi1,pi2=pi3,stpcr=stpcr,seed=1945,csvon=1)

#帰無仮説下 SSR あり

temp5=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=tau,CPeri=CPeri,

pi1=pi1,pi2=pi1,stpcr=stpcr,seed=1603,csvon=1)

#帰無仮説下 SSR なし

temp6=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=tau,CPeri=1.3,

```



```

pi1=pi1,pi2=pi1,stpcr=stpcr,seed=1914,csvon=1)

#対立仮説下 SSR なし (N=1020)
temp7=simulations(Nmin=1020,Nmax=2000,simnum=simnum,tau=0.5,CPeri=1.3,

pi1=pi1,pi2=pi2,stpcr=stpcr,seed=1939,csvon=1)

#対立仮説下 (最低差) SSR なし (N=1020)
temp8=simulations(Nmin=1020,Nmax=2000,simnum=simnum,tau=0.5,CPeri=1.3,

pi1=pi1,pi2=pi3,stpcr=stpcr,seed=1868,csvon=1)

#帰無仮説下 SSR なし (N=1020)
temp9=simulations(Nmin=1020,Nmax=2000,simnum=simnum,tau=0.5,CPeri=1.3,

pi1=pi1,pi2=pi1,stpcr=stpcr,seed=1338,csvon=1)

final=cbind(temp1,temp2,temp7,temp3,temp4,temp8,temp5,temp6,temp9)

#Table の作成
c1=c(rep("1. 検出力及び第 1 種の過誤率",3),rep("2. 無効中止 ",3),

rep("3. 症例数 (全シミュレーションの平均症例数、中央値、範囲) ",2),

rep("4. 治験が可能な最小又は最大の症例数を達成した回数の割合",3),

rep("5. 当初の症例数に対して症例数が増加した回数の割合及び/又は特定の水準を超えて症例数が増加した回数の割合 (SSR) ",1),

rep("6. 中止決定時及び最終解析時の推定治療効果 (対数オッズ比スケール) ",21)

)

c2=c("有効中止 (全体: 検出力/第 1 種の過誤率) ", " 第 1 ステージ", " 第 2 ステージ",

"無効中止", " 第 1 ステージ", " 第 2 ステージ", "平均症例数 (中央値)", "最小-最大",

"第 1 ステージで有効又は無効で中止した割合", "計画された症例数で第 2 ステージ完了の到達割合", "最大例数の到達割合",

"当初の症例数に対して症例数が増加した割合", "真値", "無効中止時 (第 1 ステージ) ",

" 平均", " バイアス", " MSE", " 被覆確率", "有効中止時 (第 1 ステージ) ", " 平均", " バイアス", " MSE", " 被覆確率",

"第 1+第 2 ステージ (第 2 ステージ完了時) ", " 平均", " バイアス", " MSE", " 被覆確率", "Unconditional", " 平均", " バイアス", " MSE", " 被覆確率")

c3=paste("OR=0.636(症例数設計の仮定)の仮定で検出力が 90%になるように症例数を設定: 1 群 N=",Nmin,sep="")

c4=paste("OR=0.731(症例数設計の仮定より下振れ)の仮定で検出力が 90%になるように症例数を設定: 1 群 N=",1020,sep="")

c5=paste()

c6=tablename

final1=data.frame(final)

final1$g=c1

final1$r=c2

tableput=final1 %>% gt(groupname_col="g",rowname_col="r") %>%

```

```

cols_label(X1=md("SSR あり"),X2=md("SSR なし"),X3=md("SSR なし"),X4=md("SSR あり"),
           X5=md("SSR なし"),X6=md("SSR なし"),X7=md("SSR あり"),X8=md("SSR なし"),X9=md("SSR なし")) %>%

tab_spanner(label = "OR=0.636(期待する効果)", columns = c("X1","X2","X3")) %>%

tab_spanner(label = "OR=0.731(期待する最低差)", columns = c("X4", "X5","X6")) %>%

tab_spanner(label = "OR=1(帰無仮説)", columns = c("X7", "X8","X9")) %>%

tab_footnote(footnote =c3,
             locations = cells_column_labels(columns = c(X1,X2,X4,X5,X7,X8))) %>%

tab_footnote(footnote =c4,
             locations = cells_column_labels(columns = c(X3,X6,X9))) %>%

tab_options(table.border.bottom.width = 0,table.font.size=px(8)) %>%

tab_source_note(source_note = md(c5)) %>%

tab_header(title =tablename)

gtsave(tableput,paste(tablename,".docx",sep=""))

gtsave(tableput,paste(tablename,".html",sep=""))

#Histogram

d1=read.csv("1192.csv")

d2=read.csv("794.csv")

d3=read.csv("1939.csv")

d4=read.csv("710.csv")

d5=read.csv("1945.csv")

d6=read.csv("1868.csv")

d7=read.csv("1603.csv")

d8=read.csv("1914.csv")

d9=read.csv("1338.csv")

d1$Design="SSR あり (N=685) "

d2$Design="SSR なし (N=685) "

d3$Design="SSR なし (N=1020) "

d4$Design="SSR あり (N=685) "

d5$Design="SSR なし (N=685) "

d6$Design="SSR なし (N=1020) "

d7$Design="SSR あり (N=685) "

d8$Design="SSR なし (N=685) "

d9$Design="SSR なし (N=1020) "

dd1=rbind(d1,d2,d3)

```

```

dd2=rbind(d4,d5,d6)

dd3=rbind(d7,d8,d9)

dd1$Setting="真のオッズ比=0.731 (期待する最低差) "
dd2$Setting="真のオッズ比=0.636 (期待する効果) "
dd3$Setting="真のオッズ比=1 (帰無仮説下) "

dd1$Setting="真のオッズ比=0.731 (期待する最低差) "
dd2$Setting="真のオッズ比=0.636 (期待する効果) "
dd3$Setting="真のオッズ比=1 (帰無仮説下) "

dd=rbind(dd1,dd2,dd3)

dd1$Design<- factor(dd1$Design, levels = c("SSR あり (N=685) ","SSR なし (N=685) ","SSR なし (N=1020) "))
dd2$Design<- factor(dd2$Design, levels = c("SSR あり (N=685) ","SSR なし (N=685) ","SSR なし (N=1020) "))
dd3$Design<- factor(dd3$Design, levels = c("SSR あり (N=685) ","SSR なし (N=685) ","SSR なし (N=1020) "))
dd$Design<- factor(dd$Design, levels = c("SSR あり (N=685) ","SSR なし (N=685) ","SSR なし (N=1020) "))
dd$Setting<- factor(dd$Setting, levels = c("真のオッズ比=1 (帰無仮説下) ","真のオッズ比=0.636 (期待する効果) ","真のオッズ比=0.731 (期待する最低差)
"))

g=ggplot(dd, aes(x =numberOfCumulatedSubjects/2,y = after_stat(count / sum(count)),fill = Design, color = Design))+

  geom_histogram(position = "identity", alpha = .3,binwidth =30)+facet_wrap(~Setting)+theme_light()+xlab(" 症 例 数 ")+ylab(" 割 合 ")
)+scale_x_continuous(breaks=seq(300,1500,300))+

  theme(legend.position="bottom",legend.title=element_blank(),strip.text=element_text(colour="black"),text = element_text(size = 18))

ggsave(filename=paste(tablename,"Histogram.png"),plot=g,dpi=300,width = 16, height = 9)

}

#関数 Main_simulation : simulations を真のオッズ比を変化させ実行する関数

#stpcr : 中間解析時の無効中止の基準となる条件付き検出力 (0-1)、CPcri : Promising Zone の下限となる条件付き検出力 (0-1)

#Nmax : SSR 後の最大症例数、simnum : シミュレーション回数

sim=function(stpcr,CPcri,Nmax,simnum){

  tau=0.5

  designpre <- getDesignGroupSequential(sided = 1, alpha = 0.025, beta = 0.1,informationRates = c(tau, 1), typeOfDesign = "asOF")

  ZF=sqrt(tau)*designpre$criticalValues[2]+sqrt((1-tau)*tau)*qnorm(stpcr)

  design <- getDesignGroupSequential(

    sided = 1, alpha = 0.025, beta = 0.1,

    informationRates = c(tau, 1), typeOfDesign = "asOF",

    futilityBounds=c(ZF), bindingFutility=FALSE)

  sampleSizeResultGS <- getSampleSizeRates(design, pi2 = 0.25, pi1 = 0.175)

```

```

Nmin=ceiling(sampleSizeResultGSS$maxNumberOfSubjects/2)

temp0=c()

#真のオッズ比を変動させシミュレーションを実行

for (i in 0:107){

  p2=0.25-i/1000

  temp1=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=0.5,CPeri=CPeri,

                    pi1=0.25,pi2=p2,stperci=stperci,seed=4231,csvon=0)

  temp2=simulations(Nmin=1020,Nmax=Nmax,simnum=simnum,tau=0.5,CPeri=1.3,

                    pi1=0.25,pi2=p2,stperci=stperci,seed=352,csvon=0)

  temp3=simulations(Nmin=Nmin,Nmax=Nmax,simnum=simnum,tau=0.5,CPeri=1.3,

                    pi1=0.25,pi2=p2,stperci=stperci,seed=433,csvon=0)

  pi1=0.25

  trueodds=(1/((pi1*(1-p2))/((1-pi1)*p2)))

  temp=data.frame(as.data.frame(t(temp1)),as.data.frame(t(temp2)),as.data.frame(t(temp3)))

  temp$trueodds=trueodds

  temp0=rbind(temp0,temp)

}

return(temp0)

}

#各種グラフを作成する関数

makeg=function(data,figname){

#平均症例数と検出力

dd1=data %>%

pivot_longer(cols = !trueodds, # 対象列

              names_to = "cat",

              values_to = "VAL")

dd2=subset(dd1,dd1$cat=="V7" | dd1$cat=="V7.1" | dd1$cat=="V7.2")

dd2$Design<-ifelse(dd2$cat=="V7","SSR あり (N=685) ",

                  ifelse(dd2$cat=="V7.1","SSR なし (N=1020) ","SSR なし (N=685) "))

dd2$Design<- factor(dd2$Design, levels = c("SSR あり (N=685) ","SSR なし (N=685) ","SSR なし (N=1020) "))

dd2$samplesize=as.numeric(gsub("¥¥(.*)¥¥", "", dd2$VAL))

dd3=subset(dd1,dd1$cat=="Tst" | dd1$cat=="Tst.1" | dd1$cat=="Tst.2")

dd3$Design<-ifelse(dd3$cat=="Tst","SSR あり (N=685) ",

                  ifelse(dd3$cat=="Tst.1","SSR なし (N=1020) ","SSR なし (N=685) "))

dd3$Design<- factor(dd3$Design, levels = c("SSR あり (N=685) ","SSR なし (N=685) ","SSR なし (N=1020) "))

```

```

dd3$Power=as.numeric(dd3$VAL)

dd4=merge(dd2,dd3,by=c("trueodds","Design"))

dd4$Design1=paste("平均症例数 :",dd4$Design)

dd4$Design2=paste("検出力 :",dd4$Design)

p1 = ggplot(dd4)+ geom_line(aes(x = trueodds, y = samplesize/1000,color =Design1))+
  geom_area(aes(x = trueodds, y =Power,fill=Design2),alpha=0.2,position = 'identity')+
  scale_y_continuous(limits = c(0,1),breaks=seq(0,1,0.1),name = "検出力", sec.axis =sec_axis(~ . *1000, name = "平均症例数",breaks=seq(0,1000,100)))+xlab("オッズ
比")+
  scale_x_continuous(limits = c(0.5,1),breaks=seq(0.5,1,0.1))+
  theme_light()+xlab("オッズ比")+geom_vline(xintercept =0.636,linetype=2)+geom_vline(xintercept
=0.731,linetype=2)+theme(legend.position="bottom",legend.title=element_blank(),text = element_text(size = 14))

ggsave(filename=paste0(figname,"_1.png"),plot=p1,dpi=300,width = 16, height = 9)

#増加割合

dd2=subset(dd1,dd1$Scat=="inc" |dd1$Scat=="maxinc")

dd2$Design<-ifelse(dd2$Scat=="inc","当初の症例数に対して症例数が増加した割合","最大例数の到達割合")

dd2$Proportion=as.numeric(dd2$VAL)

p2=ggplot(dd2, aes(x = trueodds, y = Proportion,color =Design))+geom_line()+
  xlab("オッズ比")+ylab("割合")+geom_vline(xintercept =0.636,linetype=2)+theme_light()+
  geom_vline(xintercept =0.731,linetype=2)+theme(legend.position="bottom",legend.title=element_blank(),text = element_text(size = 18))

ggsave(filename=paste0(figname,"_2.png"),plot=p2,dpi=300,width = 16, height = 9)

#バイアス

dd2=subset(dd1,dd1$Scat=="V16" | dd1$Scat=="V21" | dd1$Scat=="V26" | dd1$Scat=="V31" |
  dd1$Scat=="V16.1" | dd1$Scat=="V21.1" | dd1$Scat=="V26.1" | dd1$Scat=="V31.1" |
  dd1$Scat=="V16.2" | dd1$Scat=="V21.2" | dd1$Scat=="V26.2" | dd1$Scat=="V31.2")

dd2$Design<-ifelse(dd2$Scat=="V16" |dd2$Scat=="V21" |dd2$Scat=="V26" | dd2$Scat=="V31", "SSR あり (N=685) ",
  ifelse(dd2$Scat=="V16.1" |dd2$Scat=="V21.1" |dd2$Scat=="V26.1" | dd2$Scat=="V31.1", "SSR なし (N=1020) ", "SSR なし (N=685) "))

dd2$Setting<-ifelse(dd2$Scat=="V16" |dd2$Scat=="V16.1" |dd2$Scat=="V16.2","無効中止時 (第 1 ステージ) ",
  ifelse(dd2$Scat=="V21" |dd2$Scat=="V21.1" |dd2$Scat=="V21.2","有効中止時 (第 1 ステージ) ",
  ifelse(dd2$Scat=="V26" |dd2$Scat=="V26.1" |dd2$Scat=="V26.2","第 1+第 2 ステージ (第 2 ステージ完了時) ", "Unconditional")))

dd2$Design<- factor(dd2$Design, levels = c("SSR あり (N=685) ", "SSR なし (N=685) ", "SSR なし (N=1020) "))

dd2$Setting<- factor(dd2$Setting, levels = c("無効中止時 (第 1 ステージ) ", "有効中止時 (第 1 ステージ) ", "第 1+第 2 ステージ (第 2 ステージ完了時)
","Unconditional"))

dd2$Bias=as.numeric(dd2$VAL)

```

```

p4=ggplot(dd2,aes(x = trueodds, y = Bias,color =Design))+theme_light()+geom_line()+facet_wrap(~ Setting)+
xlab("オッズ比")+geom_vline(xintercept =0.636,linetype=2)+geom_vline(xintercept =0.731,linetype=2)+
ylab("バイアス")+theme(legend.position="bottom",legend.title=element_blank(),strip.text=element_text(colour="black"),text = element_text(size = 18))
ggsave(filename=paste0(figname,"_3.png"),plot=p4,dpi=300,width = 16, height = 9)

##MSE
dd2=subset(dd1,dd1$cat=="V17" | dd1$cat=="V22" | dd1$cat=="V27" | dd1$cat=="V32" |
          dd1$cat=="V17.1" | dd1$cat=="V22.1" | dd1$cat=="V27.1" | dd1$cat=="V32.1" |
          dd1$cat=="V17.2" | dd1$cat=="V22.2" | dd1$cat=="V27.2" | dd1$cat=="V32.2")
dd2$Design<-ifelse(dd2$cat=="V17" |dd2$cat=="V22" |dd2$cat=="V27" | dd2$cat=="V32", "SSR あり (N=685) ",
                  ifelse(dd2$cat=="V17.1" |dd2$cat=="V22.1" |dd2$cat=="V27.1" | dd2$cat=="V32.1", "SSR なし (N=1020) ", "SSR なし (N=685) "))
dd2$Setting<-ifelse(dd2$cat=="V17" |dd2$cat=="V17.1" |dd2$cat=="V17.2", "無効中止時 (第 1 ステージ) ",
                   ifelse(dd2$cat=="V22" |dd2$cat=="V22.1" |dd2$cat=="V22.2", "有効中止時 (第 1 ステージ) ",
                           ifelse(dd2$cat=="V27" |dd2$cat=="V27.1" |dd2$cat=="V27.2", "第 1+第 2 ステージ (第 2 ステージ完了時) ", "Unconditional")))
dd2$Design<- factor(dd2$Design, levels = c("SSR あり (N=685) ", "SSR なし (N=685) ", "SSR なし (N=1020) "))
dd2$Setting<- factor(dd2$Setting, levels = c("無効中止時 (第 1 ステージ) ", "有効中止時 (第 1 ステージ) ", "第 1+第 2 ステージ (第 2 ステージ完了時) ", "Unconditional"))
dd2$Bias=as.numeric(dd2$VAL)

p4=ggplot(dd2,aes(x = trueodds, y = Bias,color =Design))+theme_light()+geom_line()+facet_wrap(~ Setting)+
xlab("オッズ比")+geom_vline(xintercept =0.636,linetype=2)+geom_vline(xintercept =0.731,linetype=2)+
ylab("MSE")+theme(legend.position="bottom",legend.title=element_blank(),strip.text=element_text(colour="black"),text = element_text(size = 18))
ggsave(filename=paste0(figname,"_4.png"),plot=p4,dpi=300,width = 16, height = 9)

##被覆確率
dd2=subset(dd1,dd1$cat=="V18" | dd1$cat=="V23" | dd1$cat=="V28" | dd1$cat=="V33" |
          dd1$cat=="V18.1" | dd1$cat=="V23.1" | dd1$cat=="V28.1" | dd1$cat=="V33.1" |
          dd1$cat=="V18.2" | dd1$cat=="V23.2" | dd1$cat=="V28.2" | dd1$cat=="V33.2")
dd2$Design<-ifelse(dd2$cat=="V18" |dd2$cat=="V23" |dd2$cat=="V28" | dd2$cat=="V33", "SSR あり (N=685) ",
                  ifelse(dd2$cat=="V18.1" |dd2$cat=="V23.1" |dd2$cat=="V28.1" | dd2$cat=="V33.1", "SSR なし (N=1020) ", "SSR なし (N=685) "))
dd2$Setting<-ifelse(dd2$cat=="V18" |dd2$cat=="V18.1" |dd2$cat=="V18.2", "無効中止時 (第 1 ステージ) ",
                   ifelse(dd2$cat=="V23" |dd2$cat=="V23.1" |dd2$cat=="V23.2", "有効中止時 (第 1 ステージ) ",
                           ifelse(dd2$cat=="V28" |dd2$cat=="V28.1" |dd2$cat=="V28.2", "第 1+第 2 ステージ (第 2 ステージ完了時) ", "Unconditional")))
dd2$Design<- factor(dd2$Design, levels = c("SSR あり (N=685) ", "SSR なし (N=685) ", "SSR なし (N=1020) "))
dd2$Setting<- factor(dd2$Setting, levels = c("無効中止時 (第 1 ステージ) ", "有効中止時 (第 1 ステージ) ", "第 1+第 2 ステージ (第 2 ステージ完了時) ", "Unconditional"))
dd2$Bias=as.numeric(dd2$VAL)

```

```

p5=ggplot(dd2,aes(x = trueodds, y = Bias,color =Design))+theme_light()+geom_line()+facet_wrap(~ Setting)+
  xlab("オッズ比")+geom_vline(xintercept =0.636,linetype=2)+geom_vline(xintercept =0.731,linetype=2)+
  ylab("被覆確率") +theme(legend.position="bottom",legend.title=element_blank(),strip.text=element_text(colour="black"),text = element_text(size = 18))
ggsave(filename=paste0(figname,"_5.png"),plot=p5,dpi=300,width = 16, height = 9)

return()

#シミュレーションの実行
Main_simulation(pi1=0.25,pi2=0.175,pi3=0.196,simnum=100000,tau=0.5,Nmax=1500,stpcri=0.1,CPeri=0.2,tablename="000_pi_1=0.25,Nmax=1500,tau=0.5,promU=0.2")
Main_simulation(pi1=0.25,pi2=0.175,pi3=0.196,simnum=100000,tau=0.5,Nmax=1500,stpcri=0.1,CPeri=0.1,tablename="000_pi_1=0.25,Nmax=1500,tau=0.5,promU=0.1")
Main_simulation(pi1=0.25,pi2=0.175,pi3=0.196,simnum=100000,tau=0.5,Nmax=1500,stpcri=0.2,CPeri=0.2,tablename="000_pi_1=0.25,Nmax=1500,tau=0.5,promU=0.2,S
P=0.2")

makeg(data=sim(stpcri=0.1,CPeri=0.2,Nmax=1500,simnum=100000),figname="fig1")
makeg(data=sim(stpcri=0.1,CPeri=0.1,Nmax=1500,simnum=100000),figname="fig2")
makeg(data=sim(stpcri=0.2,CPeri=0.2,Nmax=1500,simnum=100000),figname="fig3")

```

7 Appendix B: 治療群の選択を伴うアダプティブデザインのプログラムコード.

7.1 治療群の選択を伴うアダプティブデザインのプログラムコード

```
/******  
治療群選択のシミュレーションマクロ  
① %TSsimマクロの読み込みおよびシミュレーションのシナリオ設定  
Name:      TSsimulation_macro.sas  
Author:    Masahiro Takatsu  
Version:   1.1  
Last update: 2024/04/07  
*****/  
  
*==用量数を変更したい場合は、マクロの引数(m1-m4)を追加/削除する;  
%macro TSsim(no, m1, m2, m3, m4, m9);  
  
*==開始時処理=====;  
proc datasets lib=work kill nolist; run ; quit; ;  
  
data _NULL_;  
  call symput("start_time", datetime());  
  call symput("start_timec", put(datetime(), NLDATMS19.));  
  call symput("n", %eval(&n1.+&n2.));  
  call symputx("rep", ceil(&nsim./10000)-1);  
run;  
  
%put simulation no=&no. start_time=&start_timec.;  
  
data _NULL_; *==メモリ不足によるERR回避のため、シミュレーション回数が10000回を超える場合は  
10000回ごとに実行する;  
  %do i=0 %to %eval(&rep.);  
    call symputx("st&i.", %eval(1+&i.*10000)); *=1, 10001, 20001, ...;  
    call symputx("en&i.", min(&nsim., %eval((&i.+1)*10000))); *=10000, 20000, 30000, ...;  
シミュレーション回数;
```



```

%end;
run;
=====;

*===シミュレーションデータ生成=====;
data data01; *非選択群の第2ステージのデータも発生させている点に注意;
do sim=1 to &nsim.;
  call streaminit(%eval(&no.*1234));
  do i=1 to &n.;
    if i<=%eval(&n1.) then stage=1;
    else stage=2;

    arm=0; *プラセボ群;
    res=rand("Normal", &m0., &sd.); output;

    %do i=1 %to %eval(&narm.); *用量群;
      arm=&i.;
      res=rand("Normal", &&m&i., &sd.); output;
    %end;

    arm=9; *陽性対照群;
    res=rand("Normal", &m9., &sd.); output;
  end;
end;
drop i;
run;
=====;

proc sort data=data01;
  by sim arm stage ;
run;

*===用量選択ここから=====;
/*=====
data01に基づき、以下の2つのデータセットを生成する
①sel_arms : 選択された用量の記録を持つデータ

```

変数 sim, arm, cond_min

sim: シミュレーションの繰り返し番号(1~&nsim.)

arm: 選択された用量

cond_min: 条件を満たす最低用量フラグ

②cnt : 選択用量数のデータ

変数 sim, cnt

sim: シミュレーションの繰り返し番号(1~&nsim.)

cnt: 選択用量数

=====*/

```
proc means data=data01;
```

```
  where stage=1;
```

```
  by sim arm;
```

```
  var res;
```

```
  output out=mean1(drop=_) mean=mean;
```

```
run;
```

```
data mean2;
```

```
  merge
```

```
  mean1(where=(arm not in(0,9)))
```

```
  mean1(where=(arm=0) rename=(mean=mean0 arm=_arm) )
```

```
  mean1(where=(arm=9) rename=(mean=mean9 arm=_arm) );
```

```
  by sim;
```

```
  diff0=mean-mean0; /*プラセボ群との群間差;
```

```
  diff9=mean-mean9; /*陽性対照群との群間差;
```

```
  /*用量選択条件=====;
```

```
  if diff0>0.4 & diff9>0 then cond=1; else cond=0; /*条件を満たした用量でフラグcondを立てる;
```

```
  /*=====;
```

```
run;
```

```
/*用量選択条件を満たす最低用量を特定;
```

```
proc sort data=mean2 out=mean2_cond(keep=sim arm cond);
```

```
  where cond=1;
```

```

    by sim arm;
run;

data mean2_cond_min(rename=(cond=cond_min));
    set mean2_cond;
    by sim arm;
    if first.sim; *=条件を満たす最低用量;
run;

proc sort data=mean2;
    %if &rule.=L %then by sim descending arm; ; *=条件を満たす最低用量と、それより低い用量を
    選択;
    %if &rule.=H %then by sim arm; ; *=条件を満たす最低用量と、それより高い用量を選択;
run;

*=データセットsel_armsとcntの生成;
data sel_arms(keep=sim arm cond_min selected) cnt(keep=sim cnt);
    merge mean2 mean2_cond_min;
    %if &rule.=L %then by sim descending arm; ;
    %if &rule.=H %then by sim arm; ;

    selected=1;

    retain cond_retain cnt 0;
    if first.sim then do; cond_retain=0; cnt=0; end;

    if cond_min=1 then cond_retain=1; *=条件を満たす最低用量と、それより下のobsでフラグ=1;

    if cond_retain=1 and cnt<&maxsel. then do; *=選択用量数がmaxselになるまでoutput;
        cnt+1;
        output sel_arms;
    end;

    if last.sim then output cnt;
run;

```

```

*==用量選択ここまで=====;

*=-選択された用量;
proc sort data=sel_arms;
  by sim arm;
run;

proc transpose data=sel_arms out=tran_sel_arms(drop=_) prefix=self;
  by sim;
  id arm;
  var selected;
run;

*=-各群の選択割合の導出;
data sel_arms_freq;
  set sel_arms end=eof;
  by sim arm;
  array sel{&narm.} sel1-sel&narm.;
  array sel_min{&narm.} sel_min1-sel_min&narm.; *=-条件を満たす最低用量;
  retain sel1-sel&narm. sel_min1-sel_min&narm. 0;

  do i=1 to &narm.;
    if arm=i then sel{i}+1/&nsim.;
    if arm=i and cond_min=1 then sel_min{i}+1/&nsim.;
  end;

  drop sim arm i cond_min;
  if eof;
run;

*=-選択用量数：要約統計;
proc means data=cnt noprint;
  var cnt;
  output out=cnt_mean(drop=_) mean=cnt_mean std=cnt_std
  min=cnt_min q1=cnt_q1 median=cnt_median q3=cnt_q3 max=cnt_max;
run;

```

```

*=選択用量数：頻度集計；
proc freq data=cnt noprint;
  tables cnt/out=cnt_freq;
run;
proc transpose data=cnt_freq out=tran_cnt_freq(drop=_) prefix=CNT;
  id cnt;
  var COUNT;
run;

data tran_cnt_freq;
  format CNT0-CNT&maxsel. PCT0-PCT&maxsel.;
  set tran_cnt_freq;

  %do i=0 %to &maxsel.;
    if CNT&i=. then CNT&i.=0;
    PCT&i.=CNT&i./&nsim.;
    if PCT&i=. then PCT&i.=0;
  %end;
run;

data out_sel;
  format no m1-m&narm. m9 sd rule nsim;
  merge cnt_mean tran_cnt_freq sel_arms_freq; *=すべて1obs;

  no=&no.;
  %do i=1 %to &narm.;
    m&i.=&&m&i.;
  %end;
  m9=&m9.;
  sd=&sd.;
  rule="&rule.";
  nsim=&nsim.;

  format rule $rule.;
run;

```

```

*=結果を永久ライブラリに保存=====;
proc append base=lib.out_sel data=out_sel;
run;
=====;

*selected : 選択群のフラグ. 第1ステージでは用量群のうち選択用量およびプラセボ/陽性対照群のレ
コードで1, 第2ステージは全群のレコードで1を立てる;
*cnt : 選択用量数の変数;

*=元データに選択された用量sel_armsのデータをマージ;
data data01;
  merge data01 sel_arms(in=a); *=sel_arms : 用量群のレコードのみ持つデータ;
  by sim arm;

  if a then selected=1; *=この時点では両ステージにおける用量群のみにフラグが立つ;
run;

*=元データに各用量の選択フラグtran_sel_armsのデータをマージ;
data data01;
  merge data01 tran_sel_arms;
  by sim;

  cnt=n(of selfl1-selfl&narm.);

  if cnt>=1 then selfl9=1;

*=フラグselectedの最終化=====;
  if cnt>=1 and stage in(1,2) and arm in(0,9) then selected=1; *=1用量でも選択された場合,
両ステージにおけるプラセボ群や陽性対照群でも選択フラグを立てる;
=====;

*=レコードの選定=====;
  if cnt=0 and stage=2 then delete; *=選択用量数が0の場合, 全群の第2ステージのデータを削除
する;

```

```

if selected ne 1 and stage=2 then delete; *=非選択群の第2ステージのデータを削除する;
=====;

*=Step-down Dunnett用：非選択群の検定統計量が極めて小さくなるようにデータをシフトさせる;
if stage=1 and (arm notin(0, 9) and selected ne 1) then do;
  %if &side.=U %then res_dun=res-1E10; ;
  %if &side.=L %then res_dun=res+1E10; ;
end;
else res_dun=res;
run;
*=データセットdata01確定;

*=症例数, 推定値の要約統計量, バイアス/MSE, 被覆確率の評価=====;

%if %eval(&no.>=200) %then %do; *=検出力のシナリオでのみ評価;

*=シミュレーション1回ごとの推定値(用量群のみ);

proc means data=data01 noprint;
  by sim arm stage; *=ステージごと;
  var res;
  output out=est01_1(drop=_) n=n mean=mean std=std;
run;

proc means data=data01 noprint;
  by sim arm; *=両ステージ;
  var res;
  output out=est01_2(drop=_) n=n mean=mean std=std;
run;

proc means data=data01 noprint;
  where selected=1; *=選択されたという条件付き;
  by sim arm stage; *=ステージごと;
  var res;

```

```

output out=est02_1(drop=_) n=n mean=mean std=std;
run;
proc means data=data01 noprint;
  where selected=1; *選択されたという条件付き;
  by sim arm; *両ステージ;
  var res;
  output out=est02_2(drop=_) n=n mean=mean std=std;
run;
proc means data=data01 noprint;
  where (selected ne 1 or arm=0) and stage=1; *選択されなかったという条件付き(第1ステージのみ)、プラセボ群は条件なし;
  by sim arm stage;
  var res;
  output out=est03(drop=_) n=n mean=mean std=std;
run;

*-例数のみ集計====;
proc means data=data01 noprint;
  where arm notin(0,9);
  by sim; *全用量群, 両ステージ;
  var res;
  output out=n04(drop=_) n=n ;
run;
proc means data=data01 noprint;
  by sim; *全群, 両ステージ;
  var res;
  output out=n05(drop=_) n=n ;
run;

*-各群の推定まとめ=====;
data est10;
  set est01_1(in=a) est01_2(in=b) est02_1(in=c) est02_2(in=d) est03(in=e) n04(in=f)
n05(in=g);

  if b or d then stage=9; *両ステージ;
  if c or d then stage=stage+10; *選択されたという条件付き;

```



```

if e then stage=stage+20; *=選択されなかったという条件付き(第1ステージのみ);

*各群の平均値の信頼区間;
lclm=mean-tinv(0.975, n-1)*std/sqrt(n);
uclm=mean+tinv(0.975, n-1)*std/sqrt(n);

*バイアス、MSE、被覆確率;
if arm=0 then res_true=0;
%do i=1 %to &narm.;
    if arm=&i. then res_true=&&i.;
%end;
if arm=9 then res_true=&m9.;

if .<lclm<res_true<uclm then covfl=1; *=被覆確率;
else if .<lclm then covfl=0;

_bias=mean-res_true;
_sq_bias=_bias**2;

if f then do; arm= 99; stage=9; end; *=全用量群, 両ステージ;
if g then do; arm=100; stage=9; end; *=全群, 両ステージ;
run;

*全シミュレーションにおける推定値の要約;
proc sort data=est10;
    by arm stage sim;
run;

*非選択群のstage2の例数(0例)補完処理;
data est_dummy_stage2;
    do arm=0, 1 to &narm., 9;
        do stage=2;
            do sim=1 to &nsim.;
                output; end; end; end;
run;

```

```

data est11;
  merge est_dummy_stage2 est10(in=a);
  by arm stage sim;
  if not a then n=0;
run;

proc means data=est11 noprint;
  by arm stage;
  output out=est20(drop=_)
n(n)=n_n mean(n)=n_mean std(n)=n_std min(n)=n_min q1(n)=n_q1 median(n)=n_med q3(n)=n_q3
max(n)=n_max
n(mean)=est_n mean(mean)=est_mean std(mean)=est_std min(mean)=est_min q1(mean)=est_q1
median(mean)=est_med q3(mean)=est_q3 max(mean)=est_max
mean(_bias)=bias mean(_sq_bias)=mse
mean(covfl)=cov_prob
;
run;

data est21;
  format no m1-m&narm. m9 sd rule nsim stage arm;
  set est20;

  no=&no. ;
  %do i=1 %to &narm. ;
    m&i.=&&m&i. ;
  %end;
  m9=&m9. ;
  sd=&sd. ;
  rule="&rule. ";
  nsim=&nsim. ;

  format stage stage. rule $rule. arm arm. ;
run;

*=結果を永久ライブラリに保存=====;

```

```

proc append base=lib.out_est data=est21;
run;
*-----;

*群間差の推定まとめ-----;
*est10のソート順 ; by arm stage sim;
proc sort data=est10(keep=sim arm stage n mean std) out=est_diff01;
  where arm notin(99,100); *=全用量群、全群は除外する;
  by sim stage arm;
run;

data est_diff02; *-両ステージ;
  merge
  est_diff01(where=(arm=0 and stage ne 9) rename=(n=n0 mean=mean0 std=std0))
  est_diff01(where=(arm notin(0)));
  by sim stage;
run;

*-両ステージの群間差の推定に用いるプラセボ群の効果特定（対応する群が選択されている場合は両
ステージ、選択されていない場合は第1ステージ）;
%do i=1 %to &narm.;
proc means data=data01 noprint;
  where arm=0 and ( (selfl&i.=1 and stage in(1,2)) or (selfl&i. ne 1 and stage in(1)) );
  by sim arm; *-両ステージ;
  var res;
  output out=est01_2_&i. (drop=_) n=n0 mean=mean0 std=std0;
run;
data est01_2_&i.;
  set est01_2_&i.;
  arm=&i.;
run;
%end;

*-陽性対照群;
proc means data=data01 noprint;
  where arm=0 and ( (selfl9=1 and stage in(1,2)) or (selfl9 ne 1 and stage in(1)) );

```

```

by sim arm; *-両ステージ;
var res;
output out=est01_2_9(drop=_) n=n0 mean=mean0 std=std0;
run;
data est01_2_9;
set est01_2_9;
arm=9;
run;

data est01_2_plc;
set est01_2_9;
stage=9; *-両ステージ;
run;
proc sort data=est01_2_plc;
by sim stage arm;
run;

data est_diff03;
merge est_diff02 est01_2_plc; *-プラセボ群の両ステージ(stage=9)の効果補完;
by sim stage arm;

*-平均値の差の信頼区間;
diff=mean-mean0;
pooled_se=sqrt( ((n-1)*std**2+(n0-1)*std0**2) / (n+n0-2) ) * sqrt(1/n + 1/n0);
diff_lclm=diff-tinv(0.975, n+n0-2)*pooled_se;
diff_uclm=diff+tinv(0.975, n+n0-2)*pooled_se;

*-バイアス、MSE、被覆確率;
%do i=1 %to &narm.;
if arm=&i. then diff_true=&&i.-&m0.;
%end;
if arm=9 then diff_true=&m9.-&m0.;

if .<diff_lclm<diff_true<diff_uclm then covfl=1; *-被覆確率;
else if .<diff_lclm then covfl=0;

```

```

_bias=diff-diff_true;
_sq_bias=_bias**2;

if n*n0 ne . and n ne n0 then putlog "E" "RROR";
run;

*=全シミュレーションにおける推定値の要約;
proc sort data=est_diff03;
  by arm stage;
run;

proc means data=est_diff03 noprint;
  by arm stage;
  output out=est_diff10(drop=_)
n(diff)=est_n mean(diff)=est_mean std(diff)=est_std min(diff)=est_min q1(diff)=est_q1
median(diff)=est_med q3(diff)=est_q3 max(diff)=est_max
mean(_bias)=bias mean(_sq_bias)=mse
mean(covfl)=cov_prob
;
run;

data est_diff11;
  format no m1-m&narm. m9 sd rule nsim stage arm;
  set est_diff10;

  no=&no. ;
  %do i=1 %to &narm. ;
    m&i.=&&m&i.. ;
  %end;
  m9=&m9. ;
  sd=&sd. ;
  rule="&rule. ";
  nsim=&nsim. ;

  format stage stage. rule $rule. arm arm. ;
run;

```

```

*=結果を永久ライブラリに保存=====;
proc append base=lib.out_est_diff data=est_diff11;
run;
=====;

%end; *=検出力のシナリオでのみ評価;
=====;

data dummy_simarm;
  do sim=1 to &nsim.;
  do arm=1 to &narm.;
    output;
  end; end;
run;

*=①逆正規法=====;
*=1シナリオの全シミュレーションにて一度も1用量も選択されなかった場合への対処;
data inv_stage1_0;
  sim=.;arm=.; Adj=.;
  delete;
run;
data inv_stage2_0;
  sim=.;arm=.; Adj=.;
  delete;
run;

%do i=0 %to %eval(&rep.); *=シミュレーション回数が10000回を超える場合、小分けに実行する;

*=第1ステージのp値 (Bonferroni調整);
title "Inverse Normal (stage 1)";
ods output diffs=inv_stage1_&i.;
proc mixed data=data01 order=data;
  where (&st&i.. <= sim <= &en&i..) and stage=1 and arm notin(9) and cnt>=1; *=陽性対照群
  は対比から除外する、選択群0の場合も除外する;
  by sim;

```

```

class arm;
model res=arm;
%if &side.=U %then lsmeans arm/pdiff=controlu("0") alpha=0.025 adjust=bon; ;
%if &side.=L %then lsmeans arm/pdiff=controli("0") alpha=0.025 adjust=bon; ;
run;
title;

*=第2ステージのp値(Bonferroni調整);
title "Inverse Normal (stage 2)";
ods output diffs=inv_stage2_&i.;
proc mixed data=data01 order=data;
  where (&st&i.. <= sim <= &en&i..) and stage=2 and arm notin(9) and cnt>=1; *=陽性対照群
は対比から除外する, 選択群0の場合も除外する;
  by sim;
  class arm;
  model res=arm;
  %if &side.=U %then lsmeans arm/pdiff=controlu("0") alpha=0.025 adjust=bon; ;
  %if &side.=L %then lsmeans arm/pdiff=controli("0") alpha=0.025 adjust=bon; ;
run;
title;

%end;

data inv_stage1;
  set inv_stage1_0-inv_stage1_&rep.; *=10000回ごとの結果を統合;
  %if &side.=L %then %do; *=下側の場合は変数_ARMが用量群となるため, リネームする;
    drop ARM;
    rename _ARM=ARM;
  %end;
run;

data inv_stage2;
  set inv_stage2_0-inv_stage2_&rep.; *=10000回ごとの結果を統合;
  %if &side.=L %then %do; *=下側の場合は変数_ARMが用量群となるため, リネームする;
    drop ARM;
    rename _ARM=ARM;
  %end;

```

```

run;

*片側p値の結合;
data inv01;
  merge
  inv_stage1(keep=sim arm Adjp rename=(Adjp=Adjp_stage1))
  inv_stage2(keep=sim arm Adjp rename=(Adjp=Adjp_stage2) in=a)
  sel_arms(in=a)
  ;
  by sim arm;
  if a then selected=1;
run;

data inv02;
  set inv01;
  if selected=1 then do: *選択群のみ逆正規法の計算を行う;

  *逆正規法の重みw1, w2: 各ステージの例数/群====;
  w1=sqrt(&n1./&n.);
  w2=sqrt(&n2./&n.);
  *=====;

  Adjp_stage1=round(Adjp_stage1, 1E-20); *p値が小さすぎてprobitが算出できない場合への対
  処;
  Adjp_stage2=round(Adjp_stage2, 1E-20);

  if Adjp_stage1 notin(., 0, 1) then z_stage1=-probit(Adjp_stage1);
  if Adjp_stage1=1 then z_stage1=-8.222; *SAS Helpのprobit関数の説明に沿って設定;
  if Adjp_stage1=0 then z_stage1=-probit(1E-20);

  if Adjp_stage2 notin(., 0, 1) then z_stage2=-probit(Adjp_stage2);
  if Adjp_stage2=1 then z_stage2=-8.222;
  if Adjp_stage2=0 then z_stage2=-probit(1E-20);

  z_lm=w1*z_stage1 + w2*z_stage2;

```



```

    inv_p=1-probnorm(z_lm);

    if .<inv_p<0.025 then rejected=1;
end;
run;

data inv10;
    merge dummy_simarm inv02;
    by sim arm;

    keep sim arm inv_p selected rejected;
run;
=====;

*=@step-down Dunnett=====;
*=1シナリオの全シミュレーションにて一度も1用量も選択されなかった場合への対処;
data dun_0;
    sim=. ;arm=. ; Adjp=. ;
    delete;
run;

%do i=0 %to %eval(&rep.); *=シミュレーション回数が10000回を超える場合、小分けに実行する;

title "step-down Dunnett";
ods output diffs=dun_&i.;
proc orthoreg data=data01;
    where (&st&i.. <= sim <= &en&i..) and arm notin(9) and cnt>=1; *=陽性対照群は対比から除外する、選択群0の場合も除外する;

    by sim;
    class arm;
    model res_dun=arm;
    %if &side.=U %then lsmeans arm/pdiff=controlu("0") adjust=dunnett stepdown alpha=0.025; ;
    %if &side.=L %then lsmeans arm/pdiff=controlL("0") adjust=dunnett stepdown alpha=0.025; ;
run;
title;

```

```

%end;

data dun;
  set dun_0=dun_&rep.; *=10000回ごとの結果を統合;
  %if &side.=L %then %do; *=下側の場合は変数_ARMが用量群となるため, リネームする;
    drop ARM;
    rename _ARM=ARM;
  %end;
run;

data dun01;
  merge dun sel_arms(in=a);
  by sim arm;

  if a then selected=1;

  if .<Adjp<0.025 then rejected=1;

  keep sim arm Adjp selected rejected;
run;

data dun10;
  merge dummy_simarm dun01;
  by sim arm;
run;

*=①step-down Dunnett (operationally seamless)=====;
*=1シナリオの全シミュレーションにて一度も1用量も選択されなかった場合への対処;
data osdun_0;
  sim=. ;arm=. ; Adjp=. ;
  delete;
run;

%do i=0 %to %eval(&rep.); *=シミュレーション回数が10000回を超える場合, 小分けに実行する;

title "step-down Dunnett (operationally seamless)";

```

```

ods output diffs=osdun_&i.;
proc orthoreg data=data01;
  where (&st&i.. <= sim <= &en&i..) and (selected=1 and stage=2) and arm notin(9) and
  cnt>=1; *=陽性対照群は対比から除外する, 選択群0の場合も除外する;
  by sim;
  class arm;
  model res_dun=arm;
  %if &side.=U %then lsmeans arm/pdiff=controlu("0") adjust=dunnett stepdown alpha=0.025; ;
  %if &side.=L %then lsmeans arm/pdiff=controlL("0") adjust=dunnett stepdown alpha=0.025; ;
run;
title;

%end;

data osdun;
  set osdun_0-osdun_&rep.; *=10000回ごとの結果を統合;
  %if &side.=L %then %do; *=下側の場合は変数_ARMが用量群となるため, リネームする;
    drop ARM;
    rename _ARM=ARM;
  %end;
run;

data osdun01;
  merge osdun sel_arms(in=a);
  by sim arm;

  if a then selected=1;

  if .<Adjp<0.025 then rejected=1;

  keep sim arm Adj p selected rejected;
run;

data osdun10;
  merge dummy_simarm osdun01;
  by sim arm;

```

```

run;
=====;

*=3手法のまとめ=====;
data test01;
  set inv10(in=a) dun10(in=b) osdun10(in=c);

  format method method. rule $rule.;

  no=&no.;
  %do i=1 %to &narm.;
    m&i.=&&m&i..;
  %end;
  m9=&m9.;
  sd=&sd.;
  rule="&rule.";
  nsim=&nsim.;

  if a then method=1;
  if b then method=2;
  if c then method=11;
run;

proc sort data=test01;
  by sim;
run;

data test02; *=選択用量数cntをマージ;
  merge cnt test01;
  by sim;
run;

proc sort data=test02;
  by method sim arm;
run;

```

```

data test10;
  format no m1-m&narm. m9 sd rule nsim method sel1-sel&narm. rej1-rej&narm. rej_cond1-
  rej_cond&narm. rej_true_null rej_false_null;

  set test02;
  by method sim arm;

  %do i=1 %to &narm. ;
    if arm=&i. then res_true=&&m&i.;
  %end;
  if res_true=0 then zfl=1; *=効果の真値が0の場合に1となるフラグ;
  else zfl=0;

  array sel{&narm.} sel1-sel&narm. ;
  array rej{&narm.} rej1-rej&narm. ;
  array rej_cond{&narm.} rej_cond1-rej_cond&narm. ;

  retain sel1-sel&narm. rej1-rej&narm. rej_cond1-rej_cond&narm. 0;

  if first.method then do;
    %do i=1 %to &narm. ;
      sel&i.=0;
      rej&i.=0;
      rej_cond&i.=0;
    %end;
  end;

  *=投与群ごとの選択確率/棄却確率導出;
  do i=1 to &narm. ;
    if arm=i then sel{i}+selected/&nsim. ;
    if arm=i then rej{i}+rejected/&nsim. ;
    if arm=i & sel{i} ne 0 then rej_cond{i}=rej{i}/sel{i};
  end;

  *=用量群全体での導出;
  retain rej_true_null_retain rej_true_null rej_false_null_retain rej_false_null;

```

```

*=効果が0の群がいずれか1群でもreject (Type 1 ERR): rej_true_null;
*=効果が非0の群がいずれか1群でもreject (Power): rej_false_null;
if first.method then do;
  rej_true_null=0;
  rej_false_null=0;
end;

if first.sim then do;
  rej_true_null_retain=0;
  rej_false_null_retain=0;
end;

if rejected*zfl=1 then rej_true_null_retain=1; *=シミュレーション1回で効果が0の群のいずれ
かがrejectの場合に1;
if rejected*not(zfl)=1 then rej_false_null_retain=1; *=シミュレーション1回で効果が非0のい
ずれかがrejectの場合に1;

if last.sim then do;
  rej_true_null+rej_true_null_retain/&nsim.;
  rej_false_null+rej_false_null_retain/&nsim.;
end;

keep no m1-m&narm. m9 sd rule nsim method sel1-sel&narm. rej1-rej&narm. rej_cond1-
rej_cond&narm.
  rej_true_null rej_false_null;

if last.method;
run;
=====;

*=結果を永久的ライブラリに保存=====;
proc append base=lib.out_test data=test10;
run;
=====;

```

```

*-時間計測=====;
data _NULL_;
  call symput("end_time", datetime());
run;

data time;
  no=&no.;
  %do i=1 %to &narm.;
    m&i.=&&m&i.;
  %end;
  m9=&m9.;
  sd=&sd.;
  rule="&rule.";
  nsim=&nsim.;

  format rule $rule.;

  start=put(input(&start_time., best.), NLDATMS19.);
  end=put(input(&end_time., best.), NLDATMS19.);
  time=put(input(&end_time., best.)-input(&start_time., best.), time7.);
run;
=====;

*-結果を永久ライブラリに保存=====;
proc append base=lib.out_time data=time;
run;
=====;

%mend;

%macro exe;

*-第一種の過誤確率の検討シナリオ=====;
%TSsim(100, 0, 0, 0, 0, 0);

%TSsim(101, 0.5, 0, 0, 0, 0);

```

```

% TSsim(102, 0, 0.5, 0, 0, 0);
% TSsim(103, 0, 0, 0.5, 0, 0);
% TSsim(104, 0, 0, 0, 0.5, 0);

% TSsim(105, 0.5, 0.5, 0, 0, 0);
% TSsim(106, 0.5, 0, 0.5, 0, 0);
% TSsim(107, 0.5, 0, 0, 0.5, 0);
% TSsim(108, 0, 0.5, 0.5, 0, 0);
% TSsim(109, 0, 0.5, 0, 0.5, 0);
% TSsim(110, 0, 0, 0.5, 0.5, 0);

% TSsim(111, 0.5, 0.5, 0.5, 0, 0);
% TSsim(112, 0.5, 0.5, 0, 0.5, 0);
% TSsim(113, 0.5, 0, 0.5, 0.5, 0);
% TSsim(114, 0, 0.5, 0.5, 0.5, 0);

*=====;

*==検出力の検討シナリオ==;
*==基本パターン==;
% TSsim(201, 0.4, 0.5, 0.5, 0.5, 0.4); *==群2で頭打ち;
% TSsim(202, 0.4, 0.45, 0.5, 0.5, 0.4); *==群3で頭打ち;
% TSsim(203, 0.4, 0.433, 0.467, 0.5, 0.4); *==群4まで用量反応;
% TSsim(204, 0.4, 0.4, 0.5, 0.5, 0.4); *==群1と群2が等しい;

*==陽性対照群の効果が0.1低いパターン==;
% TSsim(205, 0.4, 0.5, 0.5, 0.5, 0.3);
% TSsim(206, 0.4, 0.45, 0.5, 0.5, 0.3);
% TSsim(207, 0.4, 0.433, 0.467, 0.5, 0.3);
% TSsim(208, 0.4, 0.4, 0.5, 0.5, 0.3);

*=====;

%mend;

```



```

/*****
治療群選択のシミュレーションマクロ
② 条件設定およびシミュレーションの実行
Name:      TSimulation_macro_exe.sas
Author:    Masahiro Takatsu
Version:   1.1
Last update: 2024/04/07
*****/

*=設定箇所1：シミュレーション結果の保存先=====;
%let path=C:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX;
libname lib "&path.";
=====;

*=設定箇所2：formatの設定（任意）=====;
options fmtsearch=(work lib) nofmterr;
proc format library=lib;
  value stage
    1="第1ステージ" 2="第2ステージ" 9="両ステージ"
    11="第1ステージ(選択された条件付き)" 12="第2ステージ(選択された条件付き)" 19="両ステージ
(選択された条件付き)"
    21="第1ステージ(選択されなかった条件付き)";
  value method 1="逆正規法" 2="Step-Down Dunnett検定" 11="Operationally Seamless";
  value $ rule "H"="高い用量を選択" "L"="低い用量を選択";
  value arm 0="プラセボ群" 1="用量1" 2="用量2" 3="用量3" 4="用量4" 9="陽性対照群" 99="用量
群全体" 100="全体";
run;
=====;

*=設定箇所3：共通条件の設定=====;
%let side=U;    *=対立仮説の向き (U: プラセボ群より大きい, L: プラセボ群より小さい);
%let narm=4;    *=第1ステージの用量数(%TSsimの引数m1-m4の数を変更した場合はこちらも変更);
%let maxsel=2;  *=選択用量数(最大);
%let rule=H;    *=高い用量を選択;
%let n1=110;    *=第1ステージの例数/群;
%let n2=230;    *=第2ステージの例数/群;

```

```

%let sd=1;      *=標準偏差(全群共通);
%let m0=0;     *=プラセボ群の効果;

%let nsim=100000; *=シミュレーション回数;

*=====;

*=出力制御;
options nomacrogen;
options nonotes;
ods noresults;
ods exclude all;

*=logを外部出力;
data _NULL_;
  call symput("date", put(date(), yymmdd10.));
run;
proc printto log="&path.¥TSsimulation_macro_&date..log"; run;
*=シミュレーション実行=====;
%exe;
*=====;

*=logの出力を元に戻す;
proc printto; run;

*=出力制御を元に戻す;
options macrogen;
options notes;
ods results;
ods exclude none;

*=Excelに結果を出力=====;
proc export data=lib.out_est outfile="&path.¥out_est.xlsx" replace; run;
proc export data=lib.out_est_diff outfile="&path.¥out_est_diff.xlsx" replace; run;
proc export data=lib.out_sel outfile="&path.¥out_sel.xlsx" replace; run;
proc export data=lib.out_test outfile="&path.¥out_test.xlsx" replace; run;
proc export data=lib.out_time outfile="&path.¥out_time.xlsx" replace; run;
*=====;

```

7.2 Adaptive Dunnett のプログラムコード

```
# 計算環境 R 4.2.2 and rpact 3.3.2

# シミュレーションを実行する関数

simulation <- function(effectsize.treatment, effectsize.positivecontrol, n1, n2, selectionrule, seed, iteration) {

  library(rpact)

  library(DunnettTests)

  # 選択ルール H

  # 基準: プラセボとの群間差が 0.4 より大きい, かつ, 陽性対照との群間差が 0 より大きい

  # 基準を満たす最低用量と, 次に高い用量の群を選択する. 基準を満たす治療群がない場合はすべての群を落として試験中止.

  mySelection1 <- function(effectVector) {

    selectedArms <- effectVector > 0.4 & effectVector > effectVector[5]

    if (sum(selectedArms) > 0) {

      min.group <- min(which(selectedArms))

    } else if (sum(selectedArms) == 0) {

      min.group <- 0

    }

    if (min.group == 1) {

      return(c(T, T, F, F, F))

    } else if (min.group == 2) {

      return(c(F, T, T, F, F))

    } else if (min.group == 3) {

      return(c(F, F, T, T, F))

    } else if (min.group == 4) {

      return(c(F, F, F, T, F))

    } else if (min.group == 0) {

      return(c(F, F, F, F, F))

    }

  }

  # 選択ルール L

  # 基準: プラセボとの群間差が 0.4 より大きい, かつ, 陽性対照との群間差が 0 より大きい

  # 基準を満たす最低用量と, さらに一つ低い用量の群を選択する. 基準を満たす群がない場合はすべての群を落として試験中止.

  mySelection2 <- function(effectVector) {

    selectedArms <- effectVector > 0.4 & effectVector > effectVector[5]
```

```

if (sum(selectedArms) > 0) {
  min.group <- min(which(selectedArms))
} else if (sum(selectedArms) == 0) {
  min.group <- 0
}
if (min.group == 1) {
  return(c(T, F, F, F, F))
} else if (min.group == 2) {
  return(c(T, T, F, F, F))
} else if (min.group == 3) {
  return(c(F, T, T, F, F))
} else if (min.group == 4) {
  return(c(F, F, T, T, F))
} else if (min.group == 0) {
  return(c(F, F, F, F, F))
}
}

# 逆正規法による統合検定で解析するデザイン情報
designInvNormal <- getDesignInverseNormal(
  kMax = 2,
  alpha = 0.025,
  sided = 1,
  informationRates = c(n1 / (n1 + n2), 1),
  typeOfDesign="noEarlyEfficacy"
)

# Adaptive Dunnett 検定 (Koenig et al., 2008) で解析するデザイン情報
designAdaptiveDunnett <- getDesignConditionalDunnett(
  alpha = 0.025,
  informationAtInterim = n1 / (n1 + n2),
  secondStageConditioning = TRUE
)

arms <- length(effectsize.treatment)
effectsize.treatment <- c(effectsize.treatment, effectsize.positivecontrol)

```

```

# 計算

if(selectionrule == "mySelection1") {

  simBonferroni <- getSimulationMultiArmMeans(

    design = designInvNormal,

    activeArms = length(effectsize.treatment),

    effectMatrix = effectsize.treatment,

    stDev = 1,

    plannedSubjects = c(n1, n1 + n2),

    intersectionTest = "Bonferroni",

    typeOfShape = "userDefined",

    typeOfSelection = "userDefined",

    selectArmsFunction = mySelection1,

    successCriterion = "all",

    maxNumberOfIterations = iteration,

    seed = seed

  )

  simAdaptiveDunnett <- getSimulationMultiArmMeans(

    design = designAdaptiveDunnett,

    activeArms = length(effectsize.treatment),

    effectMatrix = effectsize.treatment,

    stDev = 1,

    plannedSubjects = c(n1, n1 + n2),

    intersectionTest = "Dunnett",

    typeOfShape = "userDefined",

    typeOfSelection = "userDefined",

    selectArmsFunction = mySelection1,

    successCriterion = "all",

    maxNumberOfIterations = iteration,

    seed = seed

  )

}

if(selectionrule == "mySelection2") {

  simBonferroni <- getSimulationMultiArmMeans(

    design = designInvNormal ,

    activeArms = length(effectsize.treatment),

```

```

    effectMatrix = effectsize.treatment,
    stDev = 1,
    plannedSubjects = c(n1, n1 + n2),
    intersectionTest = "Bonferroni",
    typeOfShape = "userDefined",
    typeOfSelection = "userDefined",
    selectArmsFunction = mySelection2,
    successCriterion = "all",
    maxNumberOfIterations = iteration,
    seed = seed
  )
simAdaptiveDunnett <- getSimulationMultiArmMeans(
  design = designAdaptiveDunnett,
  activeArms = length(effectsize.treatment),
  effectMatrix = effectsize.treatment,
  stDev = 1,
  plannedSubjects = c(n1, n1 + n2),
  intersectionTest = "Dunnett",
  typeOfShape = "userDefined",
  typeOfSelection = "userDefined",
  selectArmsFunction = mySelection2,
  successCriterion = "all",
  maxNumberOfIterations = iteration,
  seed = seed
)
}

# 結果をデータフレームに格納
resultBonferroni <- as.data.frame(simBonferroni)
resultDunnett <- as.data.frame(simAdaptiveDunnett)

# 結果を要約するコード
# kable(summary(simBonferroni))
# kable(summary(simAdaptiveDunnett))

# 結果のデータフレームを編集

```

```
dataBonferroni <- dataAdaptiveDunnett <- data.frame(
```

```
  iteration = 1:iteration,  
  effectEstimate.IA.arm1 = NA,  
  effectEstimate.IA.arm2 = NA,  
  effectEstimate.IA.arm3 = NA,  
  effectEstimate.IA.arm4 = NA,  
  effectEstimate.FA.arm1 = NA,  
  effectEstimate.FA.arm2 = NA,  
  effectEstimate.FA.arm3 = NA,  
  effectEstimate.FA.arm4 = NA,  
  effectEstimate.stage2.arm1 = NA,  
  effectEstimate.stage2.arm2 = NA,  
  effectEstimate.stage2.arm3 = NA,  
  effectEstimate.stage2.arm4 = NA,  
  testStatistic.IA.arm1 = NA,  
  testStatistic.IA.arm2 = NA,  
  testStatistic.IA.arm3 = NA,  
  testStatistic.IA.arm4 = NA,  
  testStatistic.FA.arm1 = NA,  
  testStatistic.FA.arm2 = NA,  
  testStatistic.FA.arm3 = NA,  
  testStatistic.FA.arm4 = NA,  
  pValue.IA.arm1 = NA,  
  pValue.IA.arm2 = NA,  
  pValue.IA.arm3 = NA,  
  pValue.IA.arm4 = NA,  
  pValue.FA.arm1 = NA,  
  pValue.FA.arm2 = NA,  
  pValue.FA.arm3 = NA,  
  pValue.FA.arm4 = NA,  
  rejectPerStage.IA.arm1 = NA,  
  rejectPerStage.IA.arm2 = NA,  
  rejectPerStage.IA.arm3 = NA,  
  rejectPerStage.IA.arm4 = NA,  
  rejectPerStage.FA.arm1 = NA,  
  rejectPerStage.FA.arm2 = NA,
```

```

rejectPerStage.FA.arm3 = NA,
rejectPerStage.FA.arm4 = NA
)
for (i in 1:arms) {
  eval(parse(text=paste("vectorBonferroni.IA.arm", i, "<- simBonferroni$.data$stageNumber == 1 & simBonferroni$.data$armNumber == ", i,
sep="")))
  eval(parse(text=paste("dataBonferroni.IA.arm", i, "<- simBonferroni$.data[vectorBonferroni.IA.arm", i, "]", sep="")))
  eval(parse(text=paste("dataBonferroni$effectEstimate.IA.arm", i, "[dataBonferroni.IA.arm", i, "$iterationNumber] <- dataBonferroni.IA.arm",
i, "$effectEstimate", sep="")))
  eval(parse(text=paste("dataBonferroni$testStatistic.IA.arm", i, "[dataBonferroni.IA.arm", i, "$iterationNumber] <- dataBonferroni.IA.arm", i,
"$testStatistic", sep="")))
  eval(parse(text=paste("dataBonferroni$PValue.IA.arm", i, "[dataBonferroni.IA.arm", i, "$iterationNumber] <- dataBonferroni.IA.arm", i,
"$PValue", sep="")))
  eval(parse(text=paste("dataBonferroni$rejectPerStage.IA.arm", i, "[dataBonferroni.IA.arm", i, "$iterationNumber] <- dataBonferroni.IA.arm",
i, "$rejectPerStage", sep="")))
  eval(parse(text=paste("vectorBonferroni.FA.arm", i, "<- simBonferroni$.data$stageNumber == 2 & simBonferroni$.data$armNumber == ", i,
sep="")))
  eval(parse(text=paste("dataBonferroni.FA.arm", i, "<- simBonferroni$.data[vectorBonferroni.FA.arm", i, "]", sep="")))
  eval(parse(text=paste("dataBonferroni$effectEstimate.FA.arm", i, "[dataBonferroni.FA.arm", i, "$iterationNumber] <-
dataBonferroni.FA.arm", i, "$effectEstimate", sep="")))
  eval(parse(text=paste("dataBonferroni$testStatistic.FA.arm", i, "[dataBonferroni.FA.arm", i, "$iterationNumber] <- dataBonferroni.FA.arm", i,
"$testStatistic", sep="")))
  eval(parse(text=paste("dataBonferroni$PValue.FA.arm", i, "[dataBonferroni.FA.arm", i, "$iterationNumber] <- dataBonferroni.FA.arm", i,
"$PValue", sep="")))
  eval(parse(text=paste("dataBonferroni$rejectPerStage.FA.arm", i, "[dataBonferroni.FA.arm", i, "$iterationNumber] <-
dataBonferroni.FA.arm", i, "$rejectPerStage", sep="")))
  eval(parse(text=paste("dataBonferroni$effectEstimate.stage2.arm", i, "[!is.na(dataBonferroni$effectEstimate.FA.arm", i, ")] <- ((n1 + n2)/n2) *
dataBonferroni$effectEstimate.FA.arm", i, "[!is.na(dataBonferroni$effectEstimate.FA.arm", i, ")] - (n1/n2) *
dataBonferroni$effectEstimate.IA.arm", i, "[!is.na(dataBonferroni$effectEstimate.FA.arm", i, ")]", sep="")))
  eval(parse(text=paste("vectorAdaptiveDunnnett.IA.arm", i, "<- simAdaptiveDunnnett$.data$stageNumber == 1 &
simAdaptiveDunnnett$.data$armNumber == ", i, sep="")))
  eval(parse(text=paste("dataAdaptiveDunnnett.IA.arm", i, "<- simAdaptiveDunnnett$.data[vectorAdaptiveDunnnett.IA.arm", i, "]", sep="")))
  eval(parse(text=paste("dataAdaptiveDunnnett$effectEstimate.IA.arm", i, "[dataAdaptiveDunnnett.IA.arm", i, "$iterationNumber] <-
dataAdaptiveDunnnett.IA.arm", i, "$effectEstimate", sep="")))
  eval(parse(text=paste("dataAdaptiveDunnnett$testStatistic.IA.arm", i, "[dataAdaptiveDunnnett.IA.arm", i, "$iterationNumber] <-
dataAdaptiveDunnnett.IA.arm", i, "$testStatistic", sep="")))

```



```

eval(parse(text=paste("dataAdaptiveDunnett$PValue.IA.arm", i, "[dataAdaptiveDunnett.IA.arm", i, "$IterationNumber] <-
dataAdaptiveDunnett.IA.arm", i, "$PValue", sep="")))

eval(parse(text=paste("dataAdaptiveDunnett$rejectPerStage.IA.arm", i, "[dataAdaptiveDunnett.IA.arm", i, "$IterationNumber] <-
dataAdaptiveDunnett.IA.arm", i, "$rejectPerStage", sep="")))

eval(parse(text=paste("vectorAdaptiveDunnett.FA.arm", i, "<- simAdaptiveDunnett$.data$stageNumber == 2 &
simAdaptiveDunnett$.data$sarmNumber == ", i, sep="")))

eval(parse(text=paste("dataAdaptiveDunnett.FA.arm", i, "<- simAdaptiveDunnett$.data[vectorAdaptiveDunnett.FA.arm", i, "]", sep="")))

eval(parse(text=paste("dataAdaptiveDunnett$effectEstimate.FA.arm", i, "[dataAdaptiveDunnett.FA.arm", i, "$IterationNumber] <-
dataAdaptiveDunnett.FA.arm", i, "$effectEstimate", sep="")))

eval(parse(text=paste("dataAdaptiveDunnett$stestStatistic.FA.arm", i, "[dataAdaptiveDunnett.FA.arm", i, "$IterationNumber] <-
dataAdaptiveDunnett.FA.arm", i, "$stestStatistic", sep="")))

eval(parse(text=paste("dataAdaptiveDunnett$PValue.FA.arm", i, "[dataAdaptiveDunnett.FA.arm", i, "$IterationNumber] <-
dataAdaptiveDunnett.FA.arm", i, "$PValue", sep="")))

eval(parse(text=paste("dataAdaptiveDunnett$rejectPerStage.FA.arm", i, "[dataAdaptiveDunnett.FA.arm", i, "$IterationNumber] <-
dataAdaptiveDunnett.FA.arm", i, "$rejectPerStage", sep="")))

eval(parse(text=paste("dataAdaptiveDunnett$effectEstimate.stage2.arm", i, "[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm", i, ")] <-
((n1 + n2)/n2) * dataAdaptiveDunnett$effectEstimate.FA.arm", i, "[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm", i, ")] - (n1/n2) *
dataAdaptiveDunnett$effectEstimate.IA.arm", i, "[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm", i, ")]", sep="")))
}

# 少なくとも 1 つの仮説が有意である割合
power.Bonferroni.atleastone <- simBonferroni$rejectAtLeastOne
power.AdaptiveDunnett.atleastone <- simAdaptiveDunnett$rejectAtLeastOne

# 有意となる割合（群別）
power.Bonferroni <- matrix(simBonferroni$rejectedArmsPerStage[1:(2*arms)], ncol=arms)[2,]
power.AdaptiveDunnett <- matrix(simAdaptiveDunnett$rejectedArmsPerStage[1:(2*arms)], ncol=arms)[2,]

# 中間解析で検証する治療群に選択される条件下で有意となる割合（群別）
conditionalpower.Bonferroni <- c(
  mean(dataBonferroni$rejectPerStage.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)]),
  mean(dataBonferroni$rejectPerStage.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)]),
  mean(dataBonferroni$rejectPerStage.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)]),
  mean(dataBonferroni$rejectPerStage.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)]))
conditionalpower.AdaptiveDunnett <- c(
  mean(dataAdaptiveDunnett$rejectPerStage.FA.arm1[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm1)]),

```

```

mean(dataAdaptiveDunnett$rejectPerStage.FA.arm2[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm2)]),
mean(dataAdaptiveDunnett$rejectPerStage.FA.arm3[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm3)]),
mean(dataAdaptiveDunnett$rejectPerStage.FA.arm4[!is.na(dataAdaptiveDunnett$effectEstimate.FA.arm4)])

# 治療群が各ステージに進む割合（1行目が第1ステージ，2行目が第2ステージ）
selectarms.prob <- matrix(simBonferroni$selectedArms[1:(2*arms)], ncol=arms)[2,1:arms]

# 選択された治療群の数
numbergroups <- numbergroups.treatment <-
  (!is.na(dataBonferroni$effectEstimate.stage2.arm1)) +
  (!is.na(dataBonferroni$effectEstimate.stage2.arm2)) +
  (!is.na(dataBonferroni$effectEstimate.stage2.arm3)) +
  (!is.na(dataBonferroni$effectEstimate.stage2.arm4))
numberarms.mean <- mean(numbergroups)
numberarms.table <- table(numbergroups) / iteration
numberarms.0 <- numberarms.table[which(unlist(labels(numberarms.table)) == 0)]
numberarms.1 <- numberarms.table[which(unlist(labels(numberarms.table)) == 1)]
numberarms.2 <- numberarms.table[which(unlist(labels(numberarms.table)) == 2)]
numberarms.0 <- ifelse(length(numberarms.0) == 0, 0, numberarms.0)
numberarms.1 <- ifelse(length(numberarms.1) == 0, 0, numberarms.1)
numberarms.2 <- ifelse(length(numberarms.2) == 0, 0, numberarms.2)

# 各治療群の平均例数と試験全体の平均例数
# 試験全体には，プラセボ群と陽性対照群を含めている
samplesize <- as.data.frame(simBonferroni$sampleSizes)[,1:arms]
samplesize.mean.overall <- sum(samplesize, numberarms.0 * 2 * n1, (1 - numberarms.0) * 2 * (n1 + n2))
samplesize.mean.arm <- as.vector(apply(samplesize, 2, sum))

# 固定デザインでの検出力，条件付き検出力（Bonferroni, step-down Dunnett）
test.stage2 <- matrix(c(
  dataBonferroni$effectEstimate.stage2.arm1 / sqrt(2/n2),
  dataBonferroni$effectEstimate.stage2.arm2 / sqrt(2/n2),
  dataBonferroni$effectEstimate.stage2.arm3 / sqrt(2/n2),
  dataBonferroni$effectEstimate.stage2.arm4 / sqrt(2/n2)), ncol = arms)
Bonferroni.fixed <- 1 - pnorm(test.stage2)
for (i in which(numbergroups.treatment > 0)) {

```

```

Bonferroni.fixed[i,] <- Bonferroni.fixed[i,] < (0.025 / numbergroups.treatment[i])
}
Bonferroni.fixed <- Bonferroni.fixed[apply(is.na(Bonferroni.fixed), 1, sum) < arms,]
conditionalpower.Bonferroni.fixed.atleastone <- mean(apply(Bonferroni.fixed, 1, sum, na.rm=T) > 0)
conditionalpower.Bonferroni.fixed <- as.vector(apply(Bonferroni.fixed, 2, mean, na.rm=T))
Bonferroni.fixed[is.na(Bonferroni.fixed)] <- FALSE
power.Bonferroni.fixed.atleastone <- mean(apply(Bonferroni.fixed, 1, sum) > 0)
power.Bonferroni.fixed <- as.vector(apply(Bonferroni.fixed, 2, mean))

StepDownDunnett.fixed <- matrix(NA, ncol = 4, nrow = iteration)
for (i in 1:iteration) {
  a <- numbergroups.treatment[i]
  if (a > 1) {
    cv <- cvSDDT(k = a, alpha = 0.025, alternative = "U", corr = 0.5, df = n2 * (a + 1) - (a + 1))
    order.vector <- order(test.stage2[i,])
    for (j in 1:a) {
      StepDownDunnett.fixed[i,order.vector[j]] <- test.stage2[i,order.vector[j]] > cv[j]
      if (StepDownDunnett.fixed[i,order.vector[j]] == FALSE) break
    }
  } else if (a == 1) {
    StepDownDunnett.fixed[i,] <- test.stage2[i,] > qnorm(1-0.025)
  }
}
conditionalpower.StepDownDunnett.fixed.atleastone <- mean(apply(StepDownDunnett.fixed[apply(is.na(StepDownDunnett.fixed), 1, sum) < 4,],
1, sum, na.rm=T) > 0)
conditionalpower.StepDownDunnett.fixed <- as.vector(apply(StepDownDunnett.fixed, 2, mean, na.rm=T))
StepDownDunnett.fixed[is.na(StepDownDunnett.fixed)] <- FALSE
power.StepDownDunnett.fixed.atleastone <- mean(apply(StepDownDunnett.fixed, 1, sum) > 0)
power.StepDownDunnett.fixed <- as.vector(apply(StepDownDunnett.fixed, 2, mean))

# 推定値
mean.stage1.arm1 <- mean(dataBonferroni$effectEstimate.IA.arm1)
mean.stage1.arm2 <- mean(dataBonferroni$effectEstimate.IA.arm2)
mean.stage1.arm3 <- mean(dataBonferroni$effectEstimate.IA.arm3)
mean.stage1.arm4 <- mean(dataBonferroni$effectEstimate.IA.arm4)
mean.stage1 <- c(mean.stage1.arm1, mean.stage1.arm2, mean.stage1.arm3, mean.stage1.arm4)

```

```

mean.overall.arm1 <- mean(c(
  dataBonferroni$effectEstimate.IA.arm1[is.na(dataBonferroni$effectEstimate.FA.arm1)],
  dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)])
mean.overall.arm2 <- mean(c(
  dataBonferroni$effectEstimate.IA.arm2[is.na(dataBonferroni$effectEstimate.FA.arm2)],
  dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)])
mean.overall.arm3 <- mean(c(
  dataBonferroni$effectEstimate.IA.arm3[is.na(dataBonferroni$effectEstimate.FA.arm3)],
  dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)])
mean.overall.arm4 <- mean(c(
  dataBonferroni$effectEstimate.IA.arm4[is.na(dataBonferroni$effectEstimate.FA.arm4)],
  dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)])
mean.overall <- c(mean.overall.arm1, mean.overall.arm2, mean.overall.arm3, mean.overall.arm4)
conditionalmean.select.stage1.arm1 <- mean(dataBonferroni$effectEstimate.IA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)])
conditionalmean.select.stage1.arm2 <- mean(dataBonferroni$effectEstimate.IA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)])
conditionalmean.select.stage1.arm3 <- mean(dataBonferroni$effectEstimate.IA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)])
conditionalmean.select.stage1.arm4 <- mean(dataBonferroni$effectEstimate.IA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)])
conditionalmean.select.stage1 <- c(conditionalmean.select.stage1.arm1, conditionalmean.select.stage1.arm2, conditionalmean.select.stage1.arm3,
conditionalmean.select.stage1.arm4)
conditionalmean.nonselect.stage1.arm1 <- mean(dataBonferroni$effectEstimate.IA.arm1[is.na(dataBonferroni$effectEstimate.FA.arm1)])
conditionalmean.nonselect.stage1.arm2 <- mean(dataBonferroni$effectEstimate.IA.arm2[is.na(dataBonferroni$effectEstimate.FA.arm2)])
conditionalmean.nonselect.stage1.arm3 <- mean(dataBonferroni$effectEstimate.IA.arm3[is.na(dataBonferroni$effectEstimate.FA.arm3)])
conditionalmean.nonselect.stage1.arm4 <- mean(dataBonferroni$effectEstimate.IA.arm4[is.na(dataBonferroni$effectEstimate.FA.arm4)])
conditionalmean.nonselect.stage1 <- c(conditionalmean.nonselect.stage1.arm1, conditionalmean.nonselect.stage1.arm2,
conditionalmean.nonselect.stage1.arm3, conditionalmean.nonselect.stage1.arm4)
conditionalmean.select.stage2.arm1 <- mean(dataBonferroni$effectEstimate.stage2.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)])
conditionalmean.select.stage2.arm2 <- mean(dataBonferroni$effectEstimate.stage2.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)])
conditionalmean.select.stage2.arm3 <- mean(dataBonferroni$effectEstimate.stage2.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)])
conditionalmean.select.stage2.arm4 <- mean(dataBonferroni$effectEstimate.stage2.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)])
conditionalmean.select.stage2 <- c(conditionalmean.select.stage2.arm1, conditionalmean.select.stage2.arm2, conditionalmean.select.stage2.arm3,
conditionalmean.select.stage2.arm4)
conditionalmean.select.overall.arm1 <- mean(dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)])
conditionalmean.select.overall.arm2 <- mean(dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)])
conditionalmean.select.overall.arm3 <- mean(dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)])
conditionalmean.select.overall.arm4 <- mean(dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)])
conditionalmean.select.overall <- c(conditionalmean.select.overall.arm1, conditionalmean.select.overall.arm2,

```

```

conditionalmean.select.overall.arm3, conditionalmean.select.overall.arm4)

# バイアス
bias.stage1 <- mean.stage1 - effectsize.treatment[1:arms]
bias.overall <- mean.overall - effectsize.treatment[1:arms]
conditionalbias.select.stage1 <- conditionalmean.select.stage1 - effectsize.treatment[1:arms]
conditionalbias.nonselect.stage1 <- conditionalmean.nonselect.stage1 - effectsize.treatment[1:arms]
conditionalbias.select.stage2 <- conditionalmean.select.stage2 - effectsize.treatment[1:arms]
conditionalbias.select.overall <- conditionalmean.select.overall - effectsize.treatment[1:arms]

# MSE
mse.stage1.arm1 <- mean((dataBonferroni$effectEstimate.IA.arm1 - effectsize.treatment[1])^2)
mse.stage1.arm2 <- mean((dataBonferroni$effectEstimate.IA.arm2 - effectsize.treatment[2])^2)
mse.stage1.arm3 <- mean((dataBonferroni$effectEstimate.IA.arm3 - effectsize.treatment[3])^2)
mse.stage1.arm4 <- mean((dataBonferroni$effectEstimate.IA.arm4 - effectsize.treatment[4])^2)
mse.stage1 <- c(mse.stage1.arm1, mse.stage1.arm2, mse.stage1.arm3, mse.stage1.arm4)
mse.overall.arm1 <- mean((c(
  dataBonferroni$effectEstimate.IA.arm1[is.na(dataBonferroni$effectEstimate.FA.arm1)],
  dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)]) - effectsize.treatment[1])^2)
mse.overall.arm2 <- mean((c(
  dataBonferroni$effectEstimate.IA.arm2[is.na(dataBonferroni$effectEstimate.FA.arm2)],
  dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)]) - effectsize.treatment[2])^2)
mse.overall.arm3 <- mean((c(
  dataBonferroni$effectEstimate.IA.arm3[is.na(dataBonferroni$effectEstimate.FA.arm3)],
  dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)]) - effectsize.treatment[3])^2)
mse.overall.arm4 <- mean((c(
  dataBonferroni$effectEstimate.IA.arm4[is.na(dataBonferroni$effectEstimate.FA.arm4)],
  dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)]) - effectsize.treatment[4])^2)
mse.overall <- c(mse.overall.arm1, mse.overall.arm2, mse.overall.arm3, mse.overall.arm4)
conditionalmse.select.stage1.arm1 <- mean((dataBonferroni$effectEstimate.IA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] -
effectsize.treatment[1])^2)
conditionalmse.select.stage1.arm2 <- mean((dataBonferroni$effectEstimate.IA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] -
effectsize.treatment[2])^2)
conditionalmse.select.stage1.arm3 <- mean((dataBonferroni$effectEstimate.IA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] -
effectsize.treatment[3])^2)
conditionalmse.select.stage1.arm4 <- mean((dataBonferroni$effectEstimate.IA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] -

```

```

effectsize.treatment[4]^2)

conditionalmse.select.stage1 <- c(conditionalmse.select.stage1.arm1, conditionalmse.select.stage1.arm2, conditionalmse.select.stage1.arm3,
conditionalmse.select.stage1.arm4)

conditionalmse.nonselect.stage1.arm1 <- mean((dataBonferroni$effectEstimate.IA.arm1[is.na(dataBonferroni$effectEstimate.FA.arm1)] -
effectsize.treatment[1])^2)

conditionalmse.nonselect.stage1.arm2 <- mean((dataBonferroni$effectEstimate.IA.arm2[is.na(dataBonferroni$effectEstimate.FA.arm2)] -
effectsize.treatment[2])^2)

conditionalmse.nonselect.stage1.arm3 <- mean((dataBonferroni$effectEstimate.IA.arm3[is.na(dataBonferroni$effectEstimate.FA.arm3)] -
effectsize.treatment[3])^2)

conditionalmse.nonselect.stage1.arm4 <- mean((dataBonferroni$effectEstimate.IA.arm4[is.na(dataBonferroni$effectEstimate.FA.arm4)] -
effectsize.treatment[4])^2)

conditionalmse.nonselect.stage1 <- c(conditionalmse.nonselect.stage1.arm1, conditionalmse.nonselect.stage1.arm2,
conditionalmse.nonselect.stage1.arm3, conditionalmse.nonselect.stage1.arm4)

conditionalmse.select.stage2.arm1 <- mean((dataBonferroni$effectEstimate.stage2.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] -
effectsize.treatment[1])^2)

conditionalmse.select.stage2.arm2 <- mean((dataBonferroni$effectEstimate.stage2.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] -
effectsize.treatment[2])^2)

conditionalmse.select.stage2.arm3 <- mean((dataBonferroni$effectEstimate.stage2.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] -
effectsize.treatment[3])^2)

conditionalmse.select.stage2.arm4 <- mean((dataBonferroni$effectEstimate.stage2.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] -
effectsize.treatment[4])^2)

conditionalmse.select.stage2 <- c(conditionalmse.select.stage2.arm1, conditionalmse.select.stage2.arm2, conditionalmse.select.stage2.arm3,
conditionalmse.select.stage2.arm4)

conditionalmse.select.overall.arm1 <- mean((dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] -
effectsize.treatment[1])^2)

conditionalmse.select.overall.arm2 <- mean((dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] -
effectsize.treatment[2])^2)

conditionalmse.select.overall.arm3 <- mean((dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] -
effectsize.treatment[3])^2)

conditionalmse.select.overall.arm4 <- mean((dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] -
effectsize.treatment[4])^2)

conditionalmse.select.overall <- c(conditionalmse.select.overall.arm1, conditionalmse.select.overall.arm2, conditionalmse.select.overall.arm3,
conditionalmse.select.overall.arm4)

# 被覆確率
lowerCI.arm1 <- dataBonferroni$effectEstimate.IA.arm1 - qnorm(1-0.025) * sqrt(2/n1)

```

```

lowerCI.arm2 <- dataBonferroni$effectEstimate.IA.arm2 - qnorm(1-0.025) * sqrt(2/n1)
lowerCI.arm3 <- dataBonferroni$effectEstimate.IA.arm3 - qnorm(1-0.025) * sqrt(2/n1)
lowerCI.arm4 <- dataBonferroni$effectEstimate.IA.arm4 - qnorm(1-0.025) * sqrt(2/n1)
upperCI.arm1 <- dataBonferroni$effectEstimate.IA.arm1 + qnorm(1-0.025) * sqrt(2/n1)
upperCI.arm2 <- dataBonferroni$effectEstimate.IA.arm2 + qnorm(1-0.025) * sqrt(2/n1)
upperCI.arm3 <- dataBonferroni$effectEstimate.IA.arm3 + qnorm(1-0.025) * sqrt(2/n1)
upperCI.arm4 <- dataBonferroni$effectEstimate.IA.arm4 + qnorm(1-0.025) * sqrt(2/n1)

coverageCI.stage1.arm1 <- mean(lowerCI.arm1 < effectsize.treatment[1] & upperCI.arm1 > effectsize.treatment[1])
coverageCI.stage1.arm2 <- mean(lowerCI.arm2 < effectsize.treatment[2] & upperCI.arm2 > effectsize.treatment[2])
coverageCI.stage1.arm3 <- mean(lowerCI.arm3 < effectsize.treatment[3] & upperCI.arm3 > effectsize.treatment[3])
coverageCI.stage1.arm4 <- mean(lowerCI.arm4 < effectsize.treatment[4] & upperCI.arm4 > effectsize.treatment[4])
coverageCI.stage1 <- c(coverageCI.stage1.arm1, coverageCI.stage1.arm2, coverageCI.stage1.arm3, coverageCI.stage1.arm4)

lowerCI.arm1 <- c(dataBonferroni$effectEstimate.IA.arm1[is.na(dataBonferroni$effectEstimate.FA.arm1)] - qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] - qnorm(1-0.025) * sqrt(2/(n1 + n2)))
lowerCI.arm2 <- c(dataBonferroni$effectEstimate.IA.arm2[is.na(dataBonferroni$effectEstimate.FA.arm2)] - qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] - qnorm(1-0.025) * sqrt(2/(n1 + n2)))
lowerCI.arm3 <- c(dataBonferroni$effectEstimate.IA.arm3[is.na(dataBonferroni$effectEstimate.FA.arm3)] - qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] - qnorm(1-0.025) * sqrt(2/(n1 + n2)))
lowerCI.arm4 <- c(dataBonferroni$effectEstimate.IA.arm4[is.na(dataBonferroni$effectEstimate.FA.arm4)] - qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] - qnorm(1-0.025) * sqrt(2/(n1 + n2)))
upperCI.arm1 <- c(dataBonferroni$effectEstimate.IA.arm1[is.na(dataBonferroni$effectEstimate.FA.arm1)] + qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] + qnorm(1-0.025) * sqrt(2/(n1 + n2)))
upperCI.arm2 <- c(dataBonferroni$effectEstimate.IA.arm2[is.na(dataBonferroni$effectEstimate.FA.arm2)] + qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] + qnorm(1-0.025) * sqrt(2/(n1 + n2)))
upperCI.arm3 <- c(dataBonferroni$effectEstimate.IA.arm3[is.na(dataBonferroni$effectEstimate.FA.arm3)] + qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] + qnorm(1-0.025) * sqrt(2/(n1 + n2)))
upperCI.arm4 <- c(dataBonferroni$effectEstimate.IA.arm4[is.na(dataBonferroni$effectEstimate.FA.arm4)] + qnorm(1-0.025) * sqrt(2/n1),
  dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] + qnorm(1-0.025) * sqrt(2/(n1 + n2)))

coverageCI.overall.arm1 <- mean(lowerCI.arm1 < effectsize.treatment[1] & upperCI.arm1 > effectsize.treatment[1])
coverageCI.overall.arm2 <- mean(lowerCI.arm2 < effectsize.treatment[2] & upperCI.arm2 > effectsize.treatment[2])
coverageCI.overall.arm3 <- mean(lowerCI.arm3 < effectsize.treatment[3] & upperCI.arm3 > effectsize.treatment[3])
coverageCI.overall.arm4 <- mean(lowerCI.arm4 < effectsize.treatment[4] & upperCI.arm4 > effectsize.treatment[4])
coverageCI.overall <- c(coverageCI.overall.arm1, coverageCI.overall.arm2, coverageCI.overall.arm3, coverageCI.overall.arm4)

lowerCI.arm1 <- dataBonferroni$effectEstimate.IA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] - qnorm(1-0.025) * sqrt(2/n1)
lowerCI.arm2 <- dataBonferroni$effectEstimate.IA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] - qnorm(1-0.025) * sqrt(2/n1)
lowerCI.arm3 <- dataBonferroni$effectEstimate.IA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] - qnorm(1-0.025) * sqrt(2/n1)

```



```

conditionalcoverageCI.select.stage2.arm4 <- mean(lowerCI.arm4 < effectsize.treatment[4] & upperCI.arm4 > effectsize.treatment[4])
conditionalcoverageCI.select.stage2      <-      c(conditionalcoverageCI.select.stage2.arm1,      conditionalcoverageCI.select.stage2.arm2,
conditionalcoverageCI.select.stage2.arm3, conditionalcoverageCI.select.stage2.arm4)

lowerCI.arm1 <- dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] - qnorm(1-0.025) * sqrt(2/(n1 + n2))
lowerCI.arm2 <- dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] - qnorm(1-0.025) * sqrt(2/(n1 + n2))
lowerCI.arm3 <- dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] - qnorm(1-0.025) * sqrt(2/(n1 + n2))
lowerCI.arm4 <- dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] - qnorm(1-0.025) * sqrt(2/(n1 + n2))
upperCI.arm1 <- dataBonferroni$effectEstimate.FA.arm1[!is.na(dataBonferroni$effectEstimate.FA.arm1)] + qnorm(1-0.025) * sqrt(2/(n1 + n2))
upperCI.arm2 <- dataBonferroni$effectEstimate.FA.arm2[!is.na(dataBonferroni$effectEstimate.FA.arm2)] + qnorm(1-0.025) * sqrt(2/(n1 + n2))
upperCI.arm3 <- dataBonferroni$effectEstimate.FA.arm3[!is.na(dataBonferroni$effectEstimate.FA.arm3)] + qnorm(1-0.025) * sqrt(2/(n1 + n2))
upperCI.arm4 <- dataBonferroni$effectEstimate.FA.arm4[!is.na(dataBonferroni$effectEstimate.FA.arm4)] + qnorm(1-0.025) * sqrt(2/(n1 + n2))

conditionalcoverageCI.select.overall.arm1 <- mean(lowerCI.arm1 < effectsize.treatment[1] & upperCI.arm1 > effectsize.treatment[1])
conditionalcoverageCI.select.overall.arm2 <- mean(lowerCI.arm2 < effectsize.treatment[2] & upperCI.arm2 > effectsize.treatment[2])
conditionalcoverageCI.select.overall.arm3 <- mean(lowerCI.arm3 < effectsize.treatment[3] & upperCI.arm3 > effectsize.treatment[3])
conditionalcoverageCI.select.overall.arm4 <- mean(lowerCI.arm4 < effectsize.treatment[4] & upperCI.arm4 > effectsize.treatment[4])
conditionalcoverageCI.select.overall      <-      c(conditionalcoverageCI.select.overall.arm1,      conditionalcoverageCI.select.overall.arm2,
conditionalcoverageCI.select.overall.arm3, conditionalcoverageCI.select.overall.arm4)

# 結果を出力
result <- list(
  effectsize.treatment = effectsize.treatment[1:arms],
  effectsize.positivecontrol = effectsize.positivecontrol,
  n1 = n1,
  n2 = n2,
  seed = seed,
  iteration = iteration,
  power.Bonferroni.atleastone = power.Bonferroni.atleastone,
  power.AdaptiveDunnett.atleastone = power.AdaptiveDunnett.atleastone,
  power.Bonferroni = power.Bonferroni,
  power.AdaptiveDunnett = power.AdaptiveDunnett,
  conditionalpower.Bonferroni = conditionalpower.Bonferroni,
  conditionalpower.AdaptiveDunnett = conditionalpower.AdaptiveDunnett,
  power.Bonferroni.fixed.atleastone = power.Bonferroni.fixed.atleastone,
  power.Bonferroni.fixed = power.Bonferroni.fixed,
  power.StepDownDunnett.fixed.atleastone = power.StepDownDunnett.fixed.atleastone,
  power.StepDownDunnett.fixed = power.StepDownDunnett.fixed,

```

conditionalpower.Bonferroni.fixed.atleastone = conditionalpower.Bonferroni.fixed.atleastone,
 conditionalpower.Bonferroni.fixed = conditionalpower.Bonferroni.fixed,
 conditionalpower.StepDownDunnett.fixed.atleastone = conditionalpower.StepDownDunnett.fixed.atleastone,
 conditionalpower.StepDownDunnett.fixed = conditionalpower.StepDownDunnett.fixed,
 selectarms.prob = selectarms.prob,
 numberarms.mean = numberarms.mean,
 numberarms.0 = numberarms.0,
 numberarms.1 = numberarms.1,
 numberarms.2 = numberarms.2,
 samplesize.mean.overall = samplesize.mean.overall,
 samplesize.mean.arm = samplesize.mean.arm,
 mean.stage1 = mean.stage1,
 mean.overall = mean.overall,
 conditionalmean.select.stage1 = conditionalmean.select.stage1,
 conditionalmean.nonselect.stage1 = conditionalmean.nonselect.stage1,
 conditionalmean.select.stage2 = conditionalmean.select.stage2,
 conditionalmean.select.overall = conditionalmean.select.overall,
 bias.stage1 = bias.stage1,
 bias.overall = bias.overall,
 conditionalbias.select.stage1 = conditionalbias.select.stage1,
 conditionalbias.nonselect.stage1 = conditionalbias.nonselect.stage1,
 conditionalbias.select.stage2 = conditionalbias.select.stage2,
 conditionalbias.select.overall = conditionalbias.select.overall,
 mse.stage1 = mse.stage1,
 mse.overall = mse.overall,
 conditionalmse.select.stage1 = conditionalmse.select.stage1,
 conditionalmse.nonselect.stage1 = conditionalmse.nonselect.stage1,
 conditionalmse.select.stage2 = conditionalmse.select.stage2,
 conditionalmse.select.overall = conditionalmse.select.overall,
 coverageCI.stage1 = coverageCI.stage1,
 coverageCI.overall = coverageCI.overall,
 conditionalcoverageCI.select.stage1 = conditionalcoverageCI.select.stage1,
 conditionalcoverageCI.nonselect.stage1 = conditionalcoverageCI.nonselect.stage1,
 conditionalcoverageCI.select.stage2 = conditionalcoverageCI.select.stage2,
 conditionalcoverageCI.select.overall = conditionalcoverageCI.select.overall

)

```
result
```

```
}
```

執筆者

日本製薬工業協会 医薬品評価委員会 データサイエンス部会 2023 年度継続タスクフォース 1

タスクフォースメンバー

氏名	所属
田中 勇輔	アステラス製薬株式会社
青木 誠	ノバルティス ファーマ株式会社
高津 正寛	持田製薬株式会社
飯塚 政人	田辺三菱製薬株式会社

タスクフォースリーダー兼推進委員

棚瀬 貴紀	大鵬薬品工業株式会社
中村 将俊	ファイザーR&D 合同会社

担当副部長

角野 修司	武田薬品工業株式会社
菅波 秀規	興和株式会社