

令和7(2025)年度 資源評価におけるモデル診断手順と診断結果の情報提供指針

2025年6月12日¹

資源評価高度化作業部会

(文責) 八木達紀・濱邊昂平・市野川桃子

モデル診断の概要

我が国の1系資源の評価には主にVPAが用いられているが、近年では多くの資源で複数の資源量指数を導入したチューニングVPAが資源量推定に利用されている。チューニングVPAでは、最終年の漁獲係数(F)の推定方法・資源量と資源量指数との間の非線形関係・複数資源量指数間の分散(σ^2)の推定の有無・リッジVPA(最終年のFの推定値をペナルティとして尤度に加えたVPA)導入の有無など多岐にわたるオプションが利用できる。毎年の資源評価で提案されるモデル(ベースケースモデル)は、これらのオプションを組み合わせたものであるが、ベースケースモデルが統計的に見て妥当なものであるか、また、資源評価結果がモデルの仮定に対してどの程度頑健なものであるかをきちんと説明する(モデル診断)ことが必要である。

本指針では、特にチューニングVPAで実施可能なモデル診断手法を列挙し、これらを実施するためのRの関数(パッケージfrasyr, <https://github.com/ichimomo/frasyr>)を紹介する。それをふまえ、資源評価で用いられるベースケースモデルについて、資源評価会議時に報告書やドキュメントの形で提供することが求められるモデル診断の内容を示す。VPAの手法によって適用できるモデル診断手法は異なるが、それぞれの手法で実施可能なものについてはできるだけ実施し、情報提供することが望ましい。また、VPA以外の資源評価手法を導入する場合でもここで示された考え方に基づき同様のモデル診断結果を示す必要がある。

モデル診断の内容と資源評価報告書への掲載の有無

モデル診断結果は定量化できない部分があり、「問題ない」と「問題あり」の閾値を明確に決定することができない。したがって、担当者が「問題ない」と考えても、他の人(査読者や有識者)は「問題あり」と捉える可能性があり、その逆もあり得る。そこで、「問題あり」の結果のみを示すのではなく、基本的には、計算できる全てのモデル診断の結果はドキュメントとして資源評価前に共有することが望ましい。また、資源評価ピアレビュー委員会によるピアレビューを受ける際にもそのドキュメントを事前にレビューアーに共有しておくとよい。それにより、資源評価結果に対する質問(たとえば非線形性パラメータ(b)を推定しているが、推定しない場合はどうなるのか?など)に、すぐに回答することができる。あるいは、ドキュメントを配布しておくことでそのような質疑応答を減らし、より本質的な議論をおこなう時間を確保することができる。その上で、資源評価報告書には「一般的に重要なモデル診断結果」と「問題があって今後の課題にしたいなど、評価報告書内で言及される結果」を「補足資料2 計算方法」における補足図2-1に載せる。あらかじめ、さまざまなパターンを網羅したドキュメントを作っておく

¹ 令和6年度版からの変更点：(6) ブートストラップ法でplot_resboot_vpa関数とdo_caa_boot関数を廃止し、boo.vpa関数とplot_boot関数へ変更したことを反映。編集上の語句の修正

ことにより、報告書に新たなモデル診断結果を急遽載せることになったとしても、その場で報告書の修正について議論することができるため、修正した報告書を再度回覧して合意を改めてとるような労力を省くことができる。

表 1 に、ドキュメントとして示すべき内容の例を網羅的に列挙した。その中で、資源評価報告書の補足資料として掲載すべきものは、基本的には、下線をひいたもの、すなわち、残差分析とレトロスペクティブ解析の結果になる。また、上述しているように資源評価報告書内で言及される重要な結果が他にもあれば、それも適宜補足資料に入れ込む。これらのモデル診断の説明と R のコマンドは Appendix のコードで実行ができる。なお、最新版のコードについては問い合わせに応じて共有を行う。

モデル診断手法の詳細

(1) パラメータ推定が適切になされているかの確認 : `do_estcheck_vpa()`

モデル診断においては、推定されたパラメータが適切なものになっているかをまず確認する必要がある。適切なパラメータ推定がなされていると判断するための基準として、ここでは、解が収束している、大域解が得られている、ヘッセ行列の逆行列が正定値であることと考える (図 1)。

`frasyr` の `do_estcheck_vpa` 関数はこれらの項目を確認するための関数である。`do_estcheck_vpa` は初めにヘッセ行列の逆行列が正定値となっているかどうかと、パラメータ推定が収束しているかどうか (局所最適解が得られている) を確認し、これらが満たされていない場合にはその旨を伝えるメッセージとともに関数の実行が止まる。この場合は、初期値を変えてみる、あるいは推定するパラメータ数を減らすなどの工夫をする必要がある。この条件がクリアされると、`do_estcheck_vpa` はジッター解析を行う。ジッター解析はパラメータ推定をするときの初期値を様々なものに変えて、それぞれの初期値からパラメータ推定を複数回実施する。それにより、推定値が大域解であるかを確認できる (Carvalho et al., 2021, 図 2)。ジッター解析の結果は図 (図 2) とともに出力され、関数に与えた VPA の推定結果の最大対数尤度とジッター解析で得られた最大対数尤度が等しいことを確認する必要がある、ジッター解析完了後に R のコンソール画面に両者の最大対数尤度が表示される。

(2) 残差分析 : `plot_residual_vpa()`

次に、与えられたデータが資源評価モデルによってどの程度説明されているか、また、パラメータ推定の際に与えた仮定にどの程度整合しているかを確認するため、残差分析を行う。チューニング VPA においては、年齢別漁獲尾数に誤差は仮定されていないが、資源量指数には対数正規分布の観測誤差が仮定されている。そのため、チューニング VPA では、資源量指数の対数から資源量指数の予測値を引いた残差の平方和を最小にする最小乗法によってパラメータ推定がなされている。これは、残差の分布が正規分布に従うと仮定した場合の最尤推定値と同等である。そのため、チューニング VPA における残差は、平均が 0、分散が σ^2 の正規分布に従い、 σ^2 は一定 (等分散)、かつ、個々の残差が独立であることが仮定されている。

このような仮定が満たされているかどうかを検討するため、残差を様々な形でプロットしたものを残差プロットと呼び、モデル診断に用いる。例えば、残差を時系列でプロットした残差の時系列プロットにおいて、ある一定の年代で残差が大きく、別の年代で残差が小さくなるような傾向にある場合、等分散の

仮定が満たされていないかもしれない。また、過去は残差が負の値を、最近は残差が正の値を取るような傾向にある場合には、独立の仮定が満たされていないかもしれない。独立の仮定が満たされているかどうかは、例えば、残差の自己相関係数の大きさや有意性を見ることによって数値化できる。

関数 `plot_residual_vpa()` では、各資源量指数の観測誤差の大きさ (σ) と 1 次の自己相関係数 (ρ) が残差プロット中に併記され、自己相関が有意である場合には*がつく (図 3 左上)。ただし、残差の自己相関は、用いられている資源量指数だけにバイアスがあることを示すわけではなく、資源量指数が正しいとしても、モデル内の他の部分での仮定の破れ等によって残差に自己相関傾向が現れていることもある。そのため、残差に自己相関がみられたとしても当該資源量指数をすぐに取り除くのではなく、モデル内の他の仮定の妥当性など、モデル全体の設定を総合的に確認し、見直す必要がある。しかし、VPA というモデルの構造上の制約や知見の不足等により、すぐにこのような問題を解決できないことが多い。

現実的な対処法としては、外れ値の影響を考慮したり、資源量指数が十分に資源動向を反映しているかどうかを確認し、複数の資源量指数がある場合には重みづけの方法の変更や、資源量指数と資源量の非線形的な関係を考慮、漁獲能率 (q) が年代によって変化している可能性などを検討する。その上で、まだ残差に自己相関が残っているようなモデルを用いる場合には、残差の自己相関が有意であることを注釈に加えつつ、当該資源量指数を除いた時の感度分析やレトロスペクティブ解析などの結果とともに、その推定結果を用いることもある。その場合は、残差に自己相関が見られる理由 (例えば、資源量指数を参照する資源年齢が異なるから等) について考察の上、本文中に記載する必要がある。残差に自己相関が見られる場合には、レトロスペクティブパターンが出やすいため、レトロスペクティブ解析の結果とも併せてモデルの使用の判断を行うことが望ましい。

`plot_residual_vpa()` の残差の時系列プロットでは観測値の 1.28σ (80%) 区間と 1.96σ (95%) 区間も同時に出力される。 1.96σ 区間は観測値の 95% がこの区間内に入ると考えられる範囲であるため、この範囲から逸脱しているデータはパラメータ推定への影響力が大きすぎる可能性があり、そのデータの影響力の大きさや外れ値である可能性について慎重に検討する必要がある。

また、モデルが資源量指数をどの程度説明できているのかを判断するために、資源量指数とモデルの予測値の両方を時系列にプロットした図 (図 3 右上) も `plot_residual_vpa()` から出力される。資源量指数がモデルによって十分に説明されていない場合、海外の資源評価ではリジェクトされる事例もある (Punt et al., 2020)。

資源量と資源量指数には $CPUE=qB$ という比例関係が仮定されている。これは資源量が大きい場合、資源量指数も大きくなることを意味する。しかし、資源量指数によっては線形関係が認められない場合があり、非線形性のパラメータ (b , $CPUE=qB^b$) を推定することでモデルと資源量指数のフィットが向上する可能性がある (Hashimoto et al., 2018)。 b が 1 より小さく、資源量の減少に対して資源量指数の減少が鈍感で、ある枯渇レベルに達すると資源量指数が急減する現象を *hyperstability* と呼ぶ。一方、 b が 1 より大きく、資源量の減少に対して、資源量指数がそれ以上に大きく減少する現象を *hyperdepletion* と呼ぶ。一般に、本パラメータは推定が難しいことが知られているため、本パラメータを推定した場合は、資源量と資源量指数の関係のプロット (図 3 左下) を通して推定結果が妥当であるか確認する必要がある。

(3) レトロスペクティブ解析：do_retrospective_vpa()

レトロスペクティブ解析は、推定の最終年から1年ずつデータを取り除いて資源評価モデルによる推定を繰り返し行う解析手法である。1年ずつデータを取り除いたモデル（トロモデル）の推定値と、得られている全ての時系列データを用いたモデル（フルモデル）の結果の推定値を比較することで、新しいデータが追加されたときの資源量推定値の修正度合いが確認できる（図4）。比較のために用いる資源量推定値としては資源量・漁獲係数・加入量・産卵親魚量などが目的に応じて用いられる。フルモデルからの推定値に対して、各トロモデルからの推定値が大きく異なっている場合、毎年の資源評価の頑健性が低いことが示唆される。また、フルモデルからの推定値に対して、トロモデルからの推定値が毎年同じ傾向を持ってずれているような現象を、レトロスペクティブパターンと呼ぶ。このレトロスペクティブパターンが大きい場合は、フルモデルからの推定結果がバイアスしている可能性が高い。

レトロスペクティブパターンの強さを表す指標として Mohn's ρ という尺度がある (Mohn, 1999) :

$$\rho = \frac{1}{N} \sum_{n=1}^N \frac{\theta_{R,T-n} - \theta_{F,T-n}}{\theta_{F,T-n}}$$

ここで n はレトロスペクティブ解析で遡った年数（最大 N 年まで遡る）、 T はフルモデルの推定最終年、 $\theta_{R,T-n}$ はトロモデルの推定最終年 ($T-n$ 年) の推定値、 $\theta_{F,T-n}$ はフルモデルの $T-n$ 年の推定値をそれぞれ表す。トロスペクティブパターンの強さを判断するうえで Mohn's ρ の大きさの絶対的な指標は存在しないが、経験則的な事例はまとめられている。Hurtado-Ferro et al. (2015) ではレトロスペクティブパターンが問題視される Mohn's ρ の経験則について、寿命の長い種では -0.15 以下または 0.2 以上、短い種では -0.22 以下または 0.30 以上としている。ただし、この値は長寿命種は yellowtail flounder、短寿命種は Pacific sardine の生物学的パラメータを基に、シミュレーションによって算出されたものであり、魚種が異なる場合はこの限りではない点に注意が必要である。トロスペクティブパターンが認められた場合、これらの値を参考に許容範囲内であるかどうかを議論する必要がある。

また、レトロスペクティブパターンが大きく表れる一つの要因は、モデルに与えられた複数のデータがそれぞれ異なる情報を持つ場合であることが知られている。チューニング VPA の場合には、年齢別漁獲尾数と資源量指数の2種類の質の異なるデータが与えられているが、それぞれが資源状態に対して異なる情報を持つ場合、資源量指数を用いないチューニングなしの VPA ではレトロスペクティブパターンが出ないのに、チューニング VPA でレトロスペクティブパターンが大きく表れることがある。特にチューニング VPA はモデルの構造上、近年の資源量推定値は資源量指数に大きく影響を受ける一方で、過去年にさかのぼるほど資源量推定値は年齢別漁獲尾数によって決定論的に決まるため、資源量指数と年齢別漁獲尾数がそれぞれ矛盾した情報を持つ場合にトロスペクティブパターンが出やすくなると考えられる。このようなケースでは、資源量指数がバイアスしている可能性だけでなく、年齢別漁獲尾数が何らかの理由でバイアスしている可能性も排除できない。そのため、一概にトロスペクティブパターンだけを理由に用いる資源量指数を選択したり、資源量指数を用いなかったりすることは、資源量指数が正しくて、年齢別漁獲尾数がバイアスしている、ひいては、資源量推定が潜在的にバイアスしている可能性を無視していることになる。したがって、Mohn's ρ だけでなく、データの特性や管理に与える悪影響などを総合的に加味した慎重な判断が必要である。レトロスペクティブパターンの大きいチューニング VPA の対応の一例として、リッジ VPA (Okamura et al., 2017) を導入することで、トロスペクティブ

パターンを減らすことが可能と考えられる。

(4) 感度分析 : `do_sensitive_vpa()`

感度分析は、資源評価モデルにおける様々な仮定の変化が、推定資源量などにどのような影響を与えるかを確認するための診断である。VPAにおいては、生物パラメータ（年齢別成熟率や年齢別体重、自然死亡係数など）や最高齢と最高齢-1歳の漁獲係数の比（ α ）、最終年のFの仮定、チューニング指標値間の重み付けなど、様々な仮定が置かれている。これらの仮定を変えて資源量推定を実施した場合、推定結果が変わることが予想されるが、それが意図した程度の変化量なのか、資源評価におけるその仮定の影響について確認する（図5）必要がある。特に、資源特有の仮定や資源評価モデルのパラメータを大きく変更する場合には、変更前の設定を用いた感度分析の結果を示すことが望ましい。

`frasyr` では `do_sensitive_vpa` 関数で実行可能だが、感度分析をしたいパラメータとその値の設定については、パラメータに応じて引数の指定方法が異なる。詳細な引数の設定方法については、Appendix（1. 感度分析（チューニングあり・なし））を確認されたい。

(5) ジャックナイフ法 : `do_jackknife_vpa()`

一般的なジャックナイフ法は、推定に用いたデータを1つずつ取り除いてパラメータ推定を繰り返すことである。それにより、各データがパラメータ推定に与える影響の強さを調べることができる。`frasyr` の `do_jackknife_vpa` では、複数の資源量指数を用いている場合、それぞれの資源量指数の影響を評価するために、資源量指数の系列を1つずつ取り除いて資源量推定を行うことを繰り返す（図6）。複数の資源量指数がありそれぞれの傾向が異なる場合に、どのデータが推定にどのような影響を与えているか確認したい際に有用である。特に、ジャックナイフ法で推定値に対する影響力が大きいと判断された資源量指数の傾向が、その他の資源量指数の傾向と全く異なる場合には、特に影響力が強い資源量指数の信頼性を確認し、必要があれば、データ間の重みづけの方法を変更するなどの手法を通して、代表性の高い資源量指数の傾向を推定結果に反映させるような検討もおこなう。

また、単一の系列の資源量指数を用いていた場合にも、極端な資源量指数を示す年を除いて資源量推定を行うことで、そのような値に対して資源量推定の感度が高いかを確認することができる。

(6) ブートストラップ法 : `boo.vpa()`

ブートストラップ法はパラメータの信頼区間を推定するために用いる方法である。データ、ここでは資源量指数の疑似データを生成し資源量推定を繰り返すことで、パーセンタイル点としてブートストラップ信頼区間を計算することができる（図7左）。`frasyr` で提供されている `boo.vpa` では2通りの疑似データの生成方法が実装されている。ノンパラメトリックブートストラップ法では、資源量指数とモデルから計算された予測資源量指数の残差をリサンプリングする手法である。一方、パラメトリックブートストラップでは対数を取った予測資源量指数を平均に、推定された観測誤差を標準偏差に持つ正規分布から、疑似データを乱数生成する。R関数 `boo.vpa()` ではパラメトリックブートストラップ法をデフォルトとしている。

この他に、ブートストラップサンプル分の観測誤差や年齢別漁獲係数、加入量や産卵親魚量などの推定

値が得られる。これらをプロットすることで、パラメータ間の相関の強さを把握することができる（図7右）。パラメータ間の相関が強い場合、パラメータ間で影響しあうことで推定が難しい場合がある。

引用文献

- Carvalho, F., Winker, H., Courtney, D., Kapur, M., Kell, L., Cardinale, M., Schirrip, M., Kitakado, T., Yamane, D., Piner, K. R., Maunder, M. N., Taylor, I., Wetzel, C. R., Doering, K., Johnson, K. F., Methot, R. D. (2021). A cookbook for using model diagnostics in integrated stock assessment. *Fisheries Research* 240, 105959
- Hashimoto, M., Okamura, H., Ichinokawa, M., Hiramatsu, K., Yamakawa, T. (2018). Impacts of the nonlinear relationship between abundance and its index in a tuned virtual population analysis. *Fish Sci* 84 (2), 335-347
- Hurtado-Ferro, F., Szuwalski, C.S., Valero, J.L., Anderson, S.C., Cunningham, C.J., Johnson, K.F., Licandeo, R., McGilliard, C.R., Monnahan, C.C., Muradian, M.L., Ono, K., Vert-Pre, K.A., Whitten, A.R., Punt, A.E. (2014). Looking in the rear-view mirror: bias and retrospective patterns in integrated, age-structured stock assessment models. *ICES Journal of Marine Science* 72, 99–110.
- Mohn, R., (1999). The retrospective problem in sequential population analysis: An investigation using cod fishery and simulated data. *ICES Journal of Marine Science* 56, 473–488.
- Okamura, H., Yamashita, Y., Ichinokawa, M. (2017). Ridge virtual population analysis to reduce the instability of fishing mortalities in the terminal year. *ICES Journal of Marine Science* 74 (9), 2427-2436
- Punt, A. E., Tuck, G. N., Day, J., Canales, C. M., Cope, J. M., de Moor, C. L., De Oliveira, J. A.A., Dickey-Collas, M., Elvarsson, B., Haltuch, M. A., Hamel, O. S., Hicks, A. C., Legault, C. M., Lynch, P. D., Wilberg M. J. (2020). When are model-based stock assessment rejected for use in management and what happens then? *Fisheries Research* 224, 105465.

表 1. ドキュメントとして資源評価報告書とともに提出することが推奨されるモデル診断の候補の例。状況に応じて示すべきもので、すべてを示さないといけないわけではないが、下線を引いた項目は資源評価報告書の補足資料 2 の図 2-1 として基本的には示すようにする（マアジ太平洋系群の例：[https:// abchan.fra.go.jp wpt wp-content uploads/2024 03 details_2023_03.pdf](https://abchan.fra.go.jp/wpt/wp-content/uploads/2024_03/details_2023_03.pdf)）。

1. 収束の確認とジッター解析 (jitter analysis)
 - ・失敗している場合にはパラメータ推定がうまくいっていないことを示すため、これについては必ず成功している結果を載せることになる（特に問題がなければ載せなくて良いが、確認して問題なかったことは記述しておく）
2. 残差プロット
3. トロスぺクティブ解析
 - ・基本的に本文に載せることとする
 - ・特にレトロスぺクティブパターンが問題になる場合や、それによってリッジ VPA などを導入した場合
4. 生物パラメータやチューニング方法についての感度分析
 - ・系群特有のパラメータや仮定に対して感度分析したい場合
 - ・チューニング方法などで特に昨年度から手法を変えた場合、昨年度の手法を用いたときの結果と比較する
 - ・自然死亡係数 (M) や最高齢と最高齢-1 歳の漁獲係数の比 (α) など、明確な根拠はないものの毎年利用している値について他の候補の値が指摘された場合、管理基準値を含めた資源量推定結果に大きな影響を及ぼしそうな場合
 - ・リッジ VPA を用いている場合にはリッジ VPA を用いない場合、資源量指数 の非線形性パラメータ (b) を推定している場合には b を推定しない場合など、より単純な仮定をおいたときの結果と比較（リッジ VPA や b 推定の効果をすぐに確認することができるため）
5. ジャックナイフ
 - ・複数の資源量指数の中で傾向の異なる資源量指数がある場合
 - ・残差分析などから外れ値の多い資源量指数がある場合
6. ブートストラップ信頼区間
 - ・資源量推定の不確実性を評価したい（信頼区間を確認したい）場合

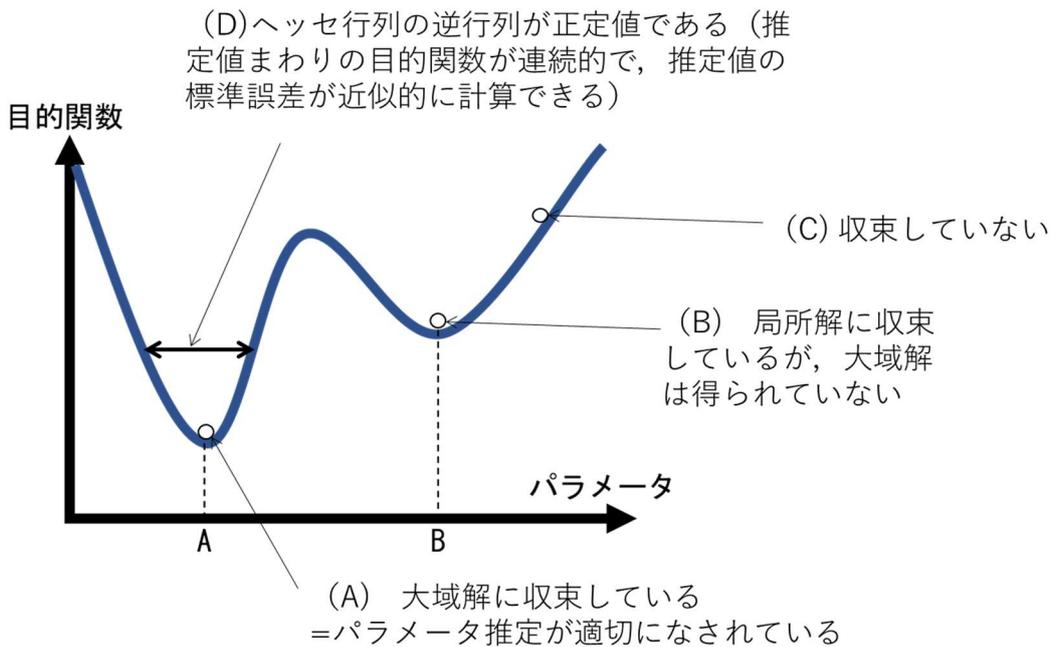


図1. 最適化を用いたパラメータ推定の概念. ここで縦軸の目的関数は、最小乗法の場合には残差平方和、最尤法の場合には負の尤度となり、目的関数が最も小さくなる時のパラメータの値（図のA）がいわゆる「推定値」（最尤推定値）となる. 適切な推定値が得られている場合というのは、解が収束していること（目的関数の微分がゼロになるような値に解が推定されているか）だけでなく、現実的なパラメータの範囲内で最も目的関数の値が小さくなるようなところの解が得られている（大域解が得られている）必要がある. 例えば、図のBは、解が収束はしているが局所解に陥ってしまっており、大域解が得られていない. また、図のCの状態は、解が収束もしていない. さらに、実用上は、大域解を得た上で、ヘッセ行列の逆行列が正定値であること（図のD）も適切なパラメータ推定の条件として求められる.

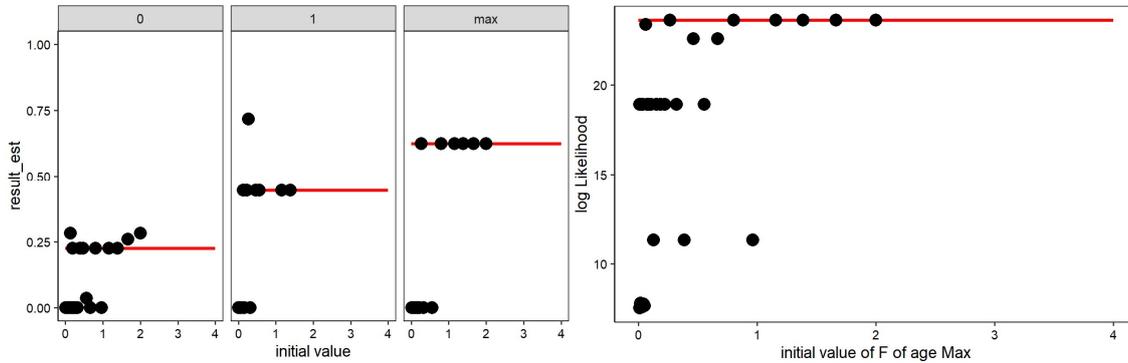


図 2. frasyr でジッター解析 (do_estcheck_vpa) を実行した時の作図. 左図は初期値 (横軸) を変えてパラメータ推定した場合のパラメータ推定値 (縦軸) を表す. 赤線はデフォルトの初期値を用いた場合のパラメータ推定値. ここでは 0 に近い初期値を用いると推定値は極端に小さな値をとる傾向がみられる. 右図は最終年最高齢の F の初期値 (横軸) と対数尤度 (縦軸) を示し, 左図右側の “max” の横軸と対応する. この結果から, 0.5~2 までの間の初期値を取れば, 多くの場合で最尤推定値を得られていることが確認でき, 考えうる初期値の範囲でデフォルトの初期値から推定された対数尤度 (右図赤線) よりも大きい結果がないことから, デフォルトの初期値の結果が大域解であることが確認できる.

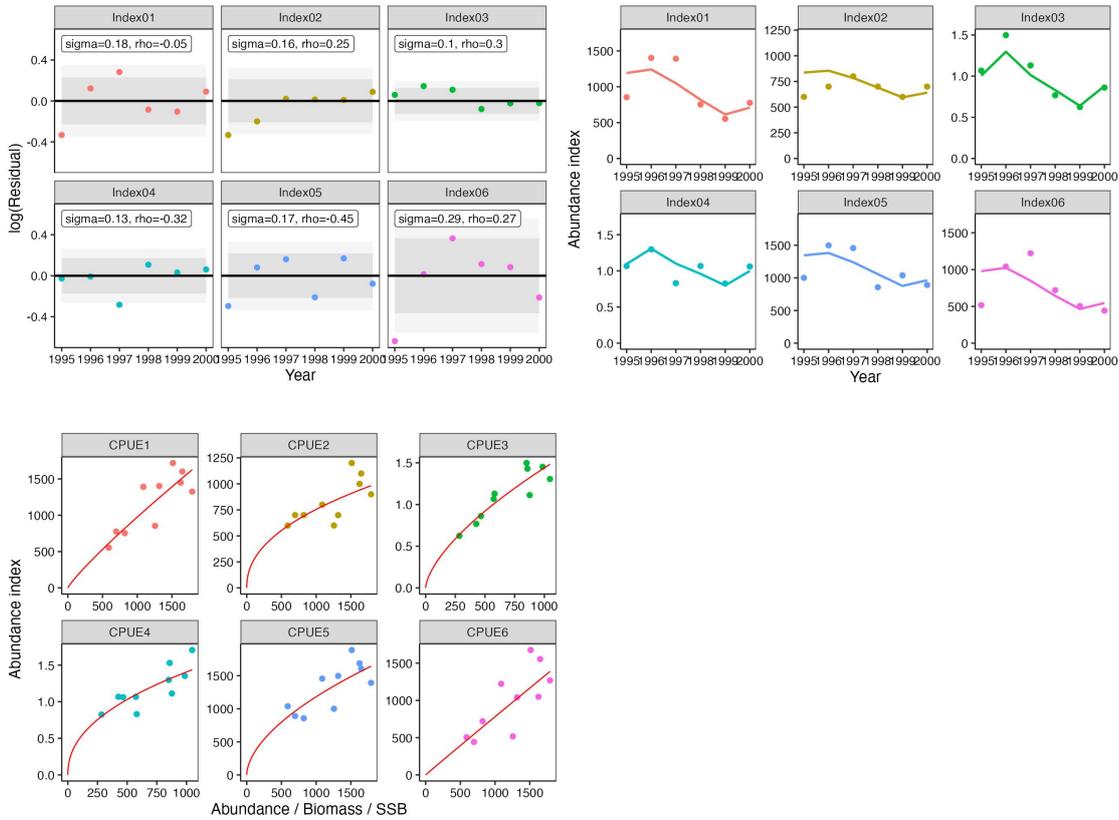


図 3. frasyr で残差解析 (plot_residual_vpa) を実行した時の作図結果. 残差の時系列プロット (左上図), 資源量指数の予測値と観測値の時系列プロット (右上図) と資源量指数に対する推定資源量指数のプロットの図 (左下図). 左上図において, σ は観測誤差, ρ は残差の自己相関係数, 薄い灰色は 1.96σ 区間, 濃い灰色は 1.28σ 区間を表す.

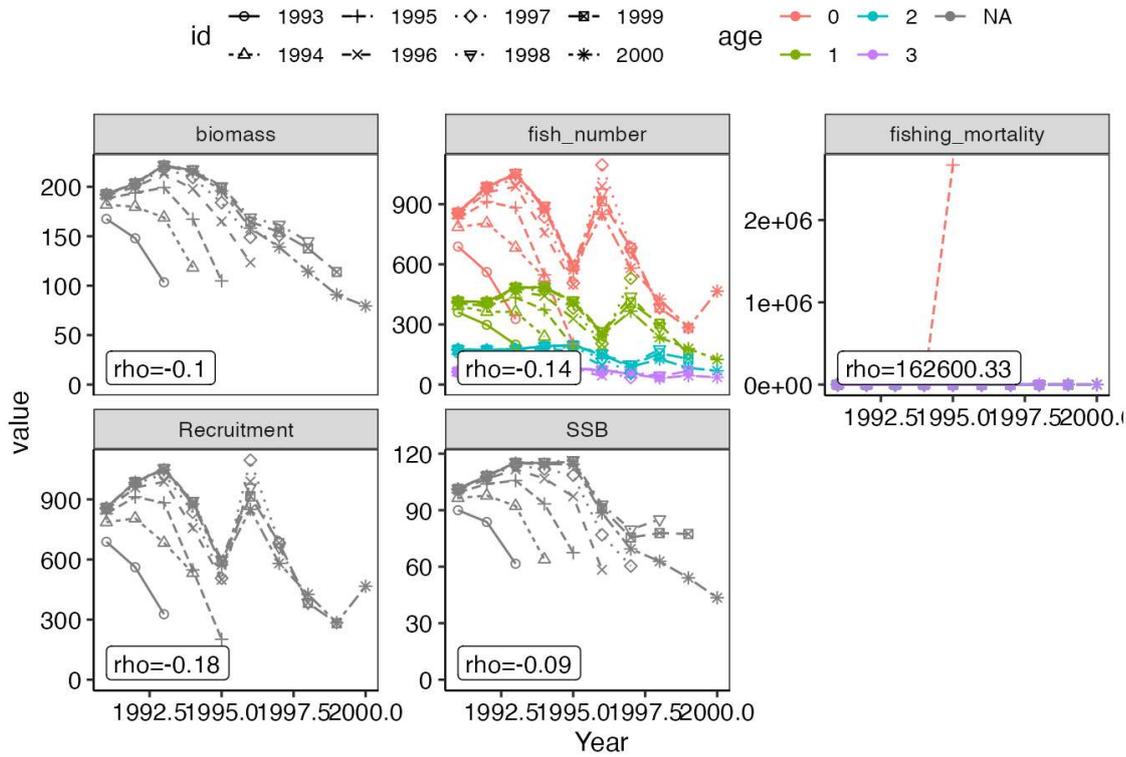


図 4. frasyr でレトロスペクティブ解析 (do_retrospective_vpa) を実行した時の作図. 各パラメータ (biomass:資源量, fish_number:年齢別資源尾数, fishing_mortality:年齢別漁獲係数, Recruitment:加入尾数, SSB:産卵親魚重量) のフルモデルとレトロモデルの動態, および Mohn' s ρ (rho) を表す.

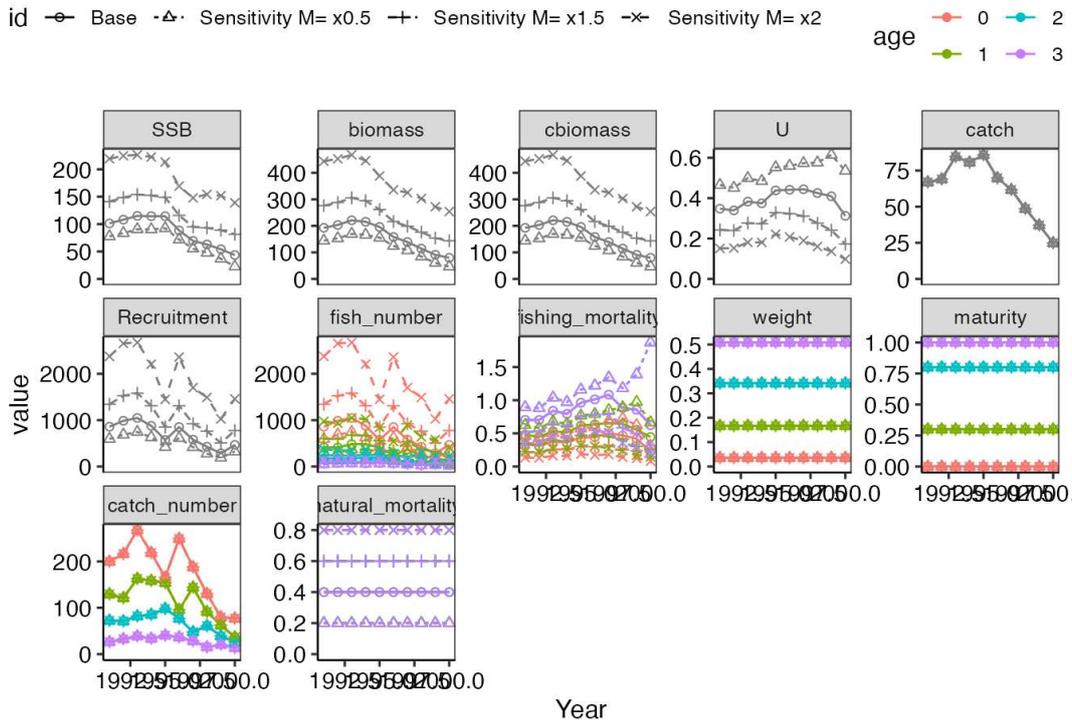


図5. frasyr で自然死亡係数 (M) についての感度分析 (do_sensitive_vpa) を実行した時の作図. 各パラメータ (cbiomass: 漁獲資源量, U: 漁獲割合, catch: 漁獲量, weight: 年齢別体重, maturity: 年齢別成熟率, catch_number: 年齢別漁獲量, natural_mortality: 年齢別自然死亡係数) の動態結果を表す.

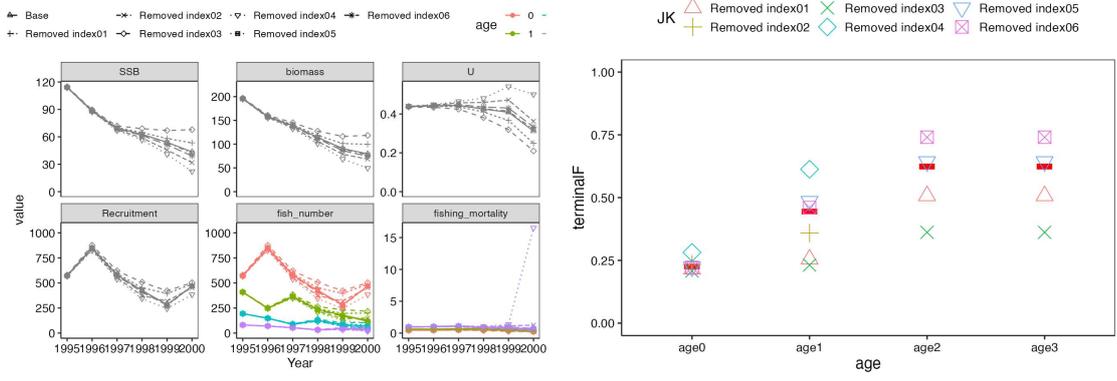


図 6. frasyr でジャックナイフ法 (do_jackknife_vpa) を実行した時の作図. 左図は資源量指数の種類を 1 つずつ取り除いた場合の各パラメータの動態結果を表す. 右図は各年齢の最終年の漁獲係数の推定値を表す. 左図の漁獲係数を見ると Index4 を取り除くと極端な推定値を取っていることから, Index4 の推定への貢献度が高いことが示唆される.

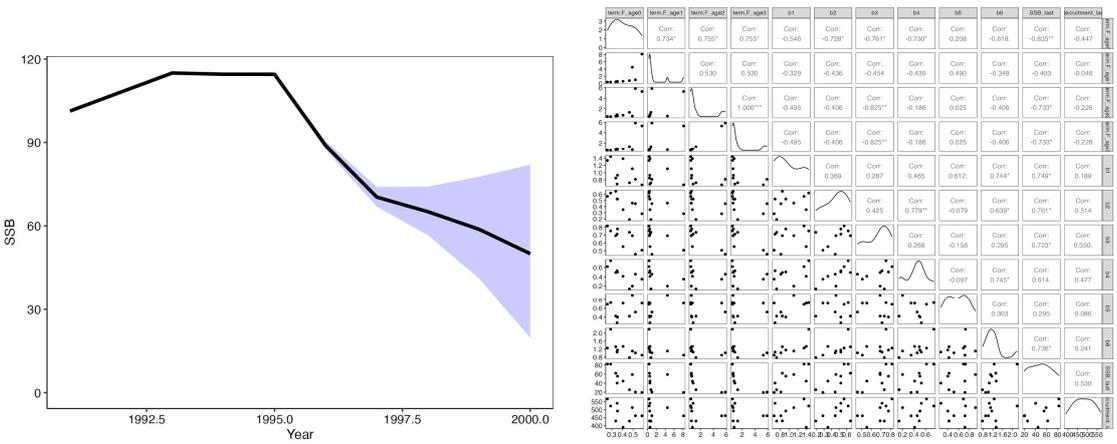


図 7. frasyr でブートストラップ法 (plot_vpa) を実行した時の作図. 左図は SSB の推定値 (黒実線) と 95%信頼区間 (青色) を表す. 右図は各推定パラメータの関係をプロットしたものである.

VPA モデル診断用スクリプト

八木達紀・濱邊昂平・市野川桃子

2025 年 6 月 12 日

2025 年度変更点

- ブートストラップ法で関数の統廃合に伴い R の例コードを変更
 - plot_resboot_vpa 関数と do_caaboot_vpa 関数が廃止され、boo.vpa 関数と plot_boot 関数へ変更
- 注意事項
 - 以下のような計算法の資源には対応していません
 - * 最高齢や + グループが時系列で変化する資源
 - * 直近年の加入尾数を、過去の平均で計算している資源
 - * 直近年の加入尾数に RPS 等を乗じている資源
 - * etc...
 - 上述のような資源担当の方は別途お問い合わせください
 - その場合、関数の修正 or VPA の再実行等でご対応頂きます

VPA モデル診断用スクリプト

- VPA のモデル診断を網羅的に実施できる関数と、使用例のスクリプトを配布します
- 基本的には「チューニングあり」VPA が対象ですが、「感度分析」と「レトロスペクティブ解析」はチューニングなし VPA でも実施できますので、実施してください

0. 事前準備

- frasyr や ggplot2 などインストールして呼び出しておいてください

※ **tidyr**, **dplyr**, **purrr**, **ggplot2**, **stringer** の 5 つは”tidyverse” というパッケージの中にまとめているので `library(tidyverse)` だけでも大丈夫です

```
# frasyr の最新版 (dev) をインストールする
# devtools::install_github("ichimomo/frasyr@dev")
library(frasyr)
library(tidyr)
library(dplyr)
library(purrr)
library(ggplot2)
library(stringr)
```

1. 感度分析 (チューニングあり・なし)

- 資源評価モデルにおけるさまざまな仮定の変化が、推定資源量などにどのような影響を与えるか確認するために実施するものです
- 理想的には再生産関係・管理基準値・将来予測などへの影響も見たほうが良いですが、どこまでやるかは難しいところです
- 全ての項目を実施頂く必要はありません
- あくまで効きそうな仮定、その資源で問題となっているパラメータについて実施してください

- 感度分析の対象：
 - 生物パラメータ各種（自然死亡係数、年齢別体重、成熟率など）
 - α （プラスグループとプラスグループ-1歳のFの比）
 - 最終年のFの仮定
 - * チューニングなしの場合：最終年のF = 過去数年のFと同じと仮定しているか
 - * チューニングありの場合：全F推定か sel.update か
- 近年の資源量推定に関わる重要な部分については、別途資源評価票で別途議論してください（たとえば近年の体重の変化が問題になっており、その値を変えたときのABCへの影響を調べたいなど）

感度分析は `do_sensitivity_vpa` 関数を使えばできます。引数で重要なものは、VPA計算の結果（`vpa`関数の実行結果のオブジェクト）、感度分析を行いたい指標 `what_replace`、感度分析の値 `value` の3つです。また、結果のグラフについての引数は `what_plot` と `ncol` があります。出力数を減らしたい場合は `what_plot` に出力したい結果だけ入力してください。`ncol` はグラフの列数です。適宜調整してください。

結果では、`$result` にVPAの結果がリスト形式で、`$graph` に複数のVPAの結果を重ね書きしたグラフが入っています。

関数の引数について（特に `value` に与えるデータの型について）、不明点等ありましたら、以下のように `help` を使って確認してみてください。

```
help(do_sensitivity_vpa)
```

生物パラメータ各種

自然死亡係数

`what_replace="M"` にすると自然死亡係数について感度分析を行えます。`value` には、感度分析を行いたい割合を `numeric` 型に入れてください。（以下の例では、もとの自然死亡係数を0.5倍、1.5倍、2倍にした感度分析結果を返す）

```
res_vpa_sensitivity <- do_sensitivity_vpa(res_vpa_estb,
                                         what_replace = "M",
                                         value = c(0.5, 1.5, 2))
```

- `$result` の中には `vpa` 計算の結果
- `$graph` の中には結果の図がそれぞれ、入っています。

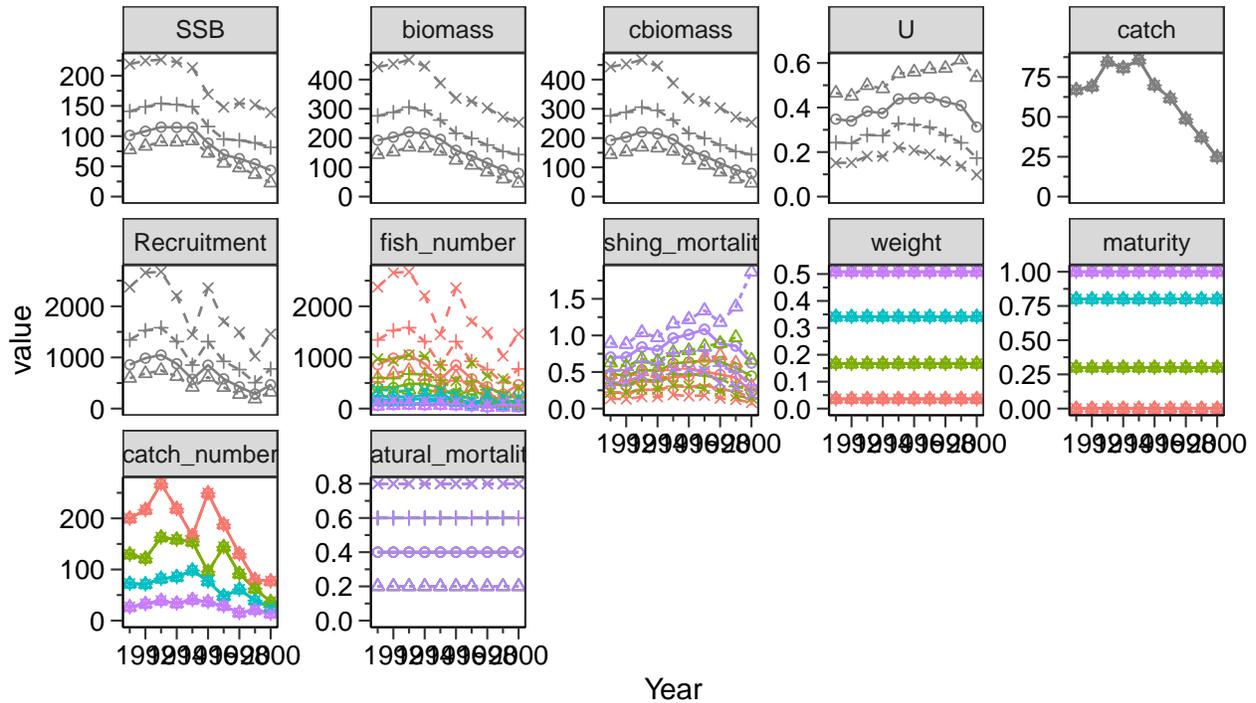
今回の例では、`value` を3通り仮定したので、それぞれの結果がリスト型式で以下のように入っております。

```
names(res_vpa_sensitivity$result)
```

```
[1] "Sensitivity M= x0.5" "Sensitivity M= x1.5" "Sensitivity M= x2"
```

```
res_vpa_sensitivity$graph
```

age — 0 — 2 — NA id — Base —△— Sensitivity M= x0.5 —+— Sensitivity M= x1.5 —x— Sensiti
 —●— 1 —●— 3



自然死亡係数を大きくすると、資源量のスケールも併せて大きくなっています。

直接 M の行列を入力しても実行できます。最近年の M の仮定について見たい、掛け算ではなく足し算で実行したい、等の場合は行列を与えてください

年齢別体重

年齢別体重を置き換える場合は、`what_replace = "waa"` とします。自然死亡係数の場合と同様に、`value` に入れた値がもとの年齢別体重にかけられます。

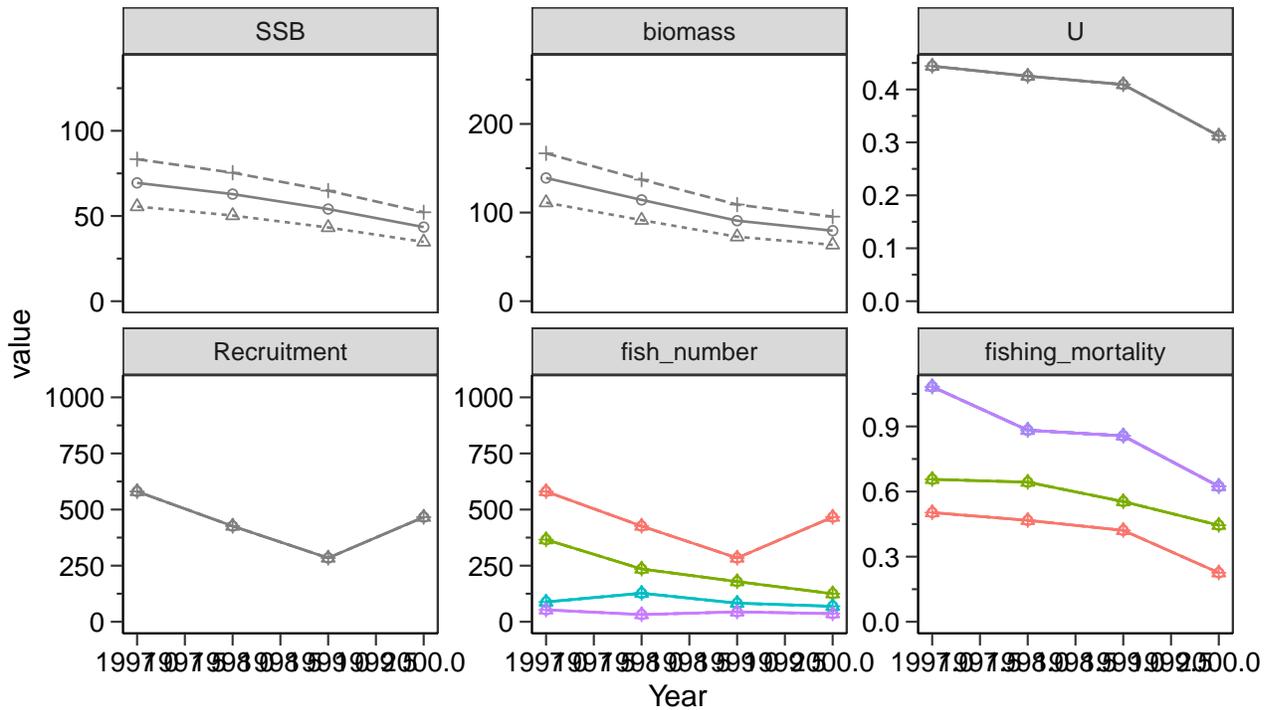
また、`what_plot` に作図で出力したい結果を指定することができます。

この他、`plot_year` に作図したい年 (x 軸の範囲) を指定することもできます

```
res_vpa_sensitivity <- do_sensitivity_vpa(res_vpa_estb,
  what_replace = "waa",
  value = c(0.8, 1.2),
  what_plot=c("SSB", "biomass", "U", "Recruitment",
    "fish_number", "fishing_mortality"),
  plot_year = c(1997, 2000),
  ncol=3
)
res_vpa_sensitivity$graph
```

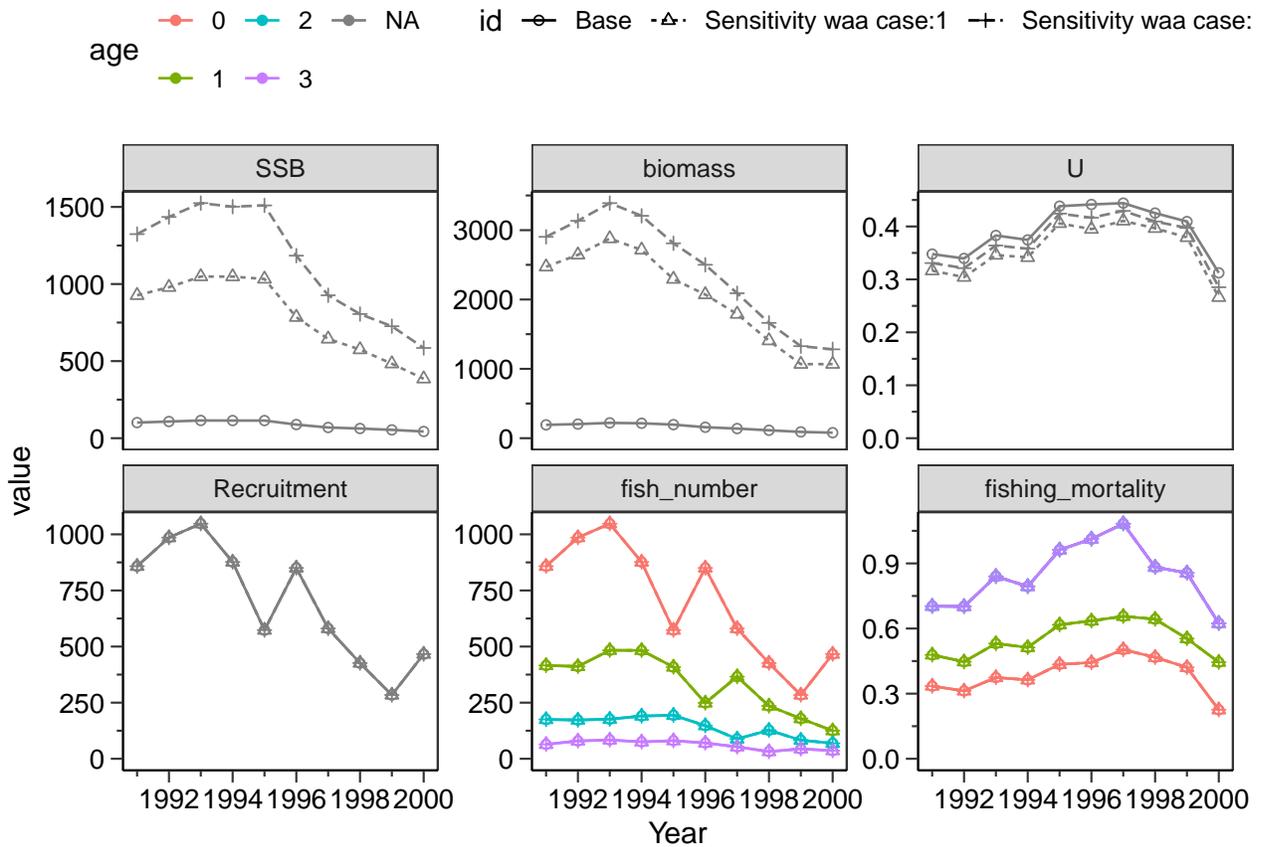
id ○ Base △ Sensitivity waa= x0.8 + Sensitivity waa= x1.2

age
 ● 0 ● 1 ● 2 ● 3 ● NA



また、解析に用いた waa と同じ長さの matrix であれば、list にまとめて value に任意の値を与えることができます。

```
res_vpa_sensitivity <- do_sensitivity_vpa(res_vpa_estb,
  what_replace = "waa",
  value = list(matrix(rep(1:4,10),nrow = 4),
    matrix(rep(c(1,2,4,8),10),nrow = 4)
  ),
  what_plot=c("SSB","biomass","U","Recruitment",
    "fish_number","fishing_mortality"),
  ncol=3
)
res_vpa_sensitivity$graph
```



漁獲物中の年齢別体重

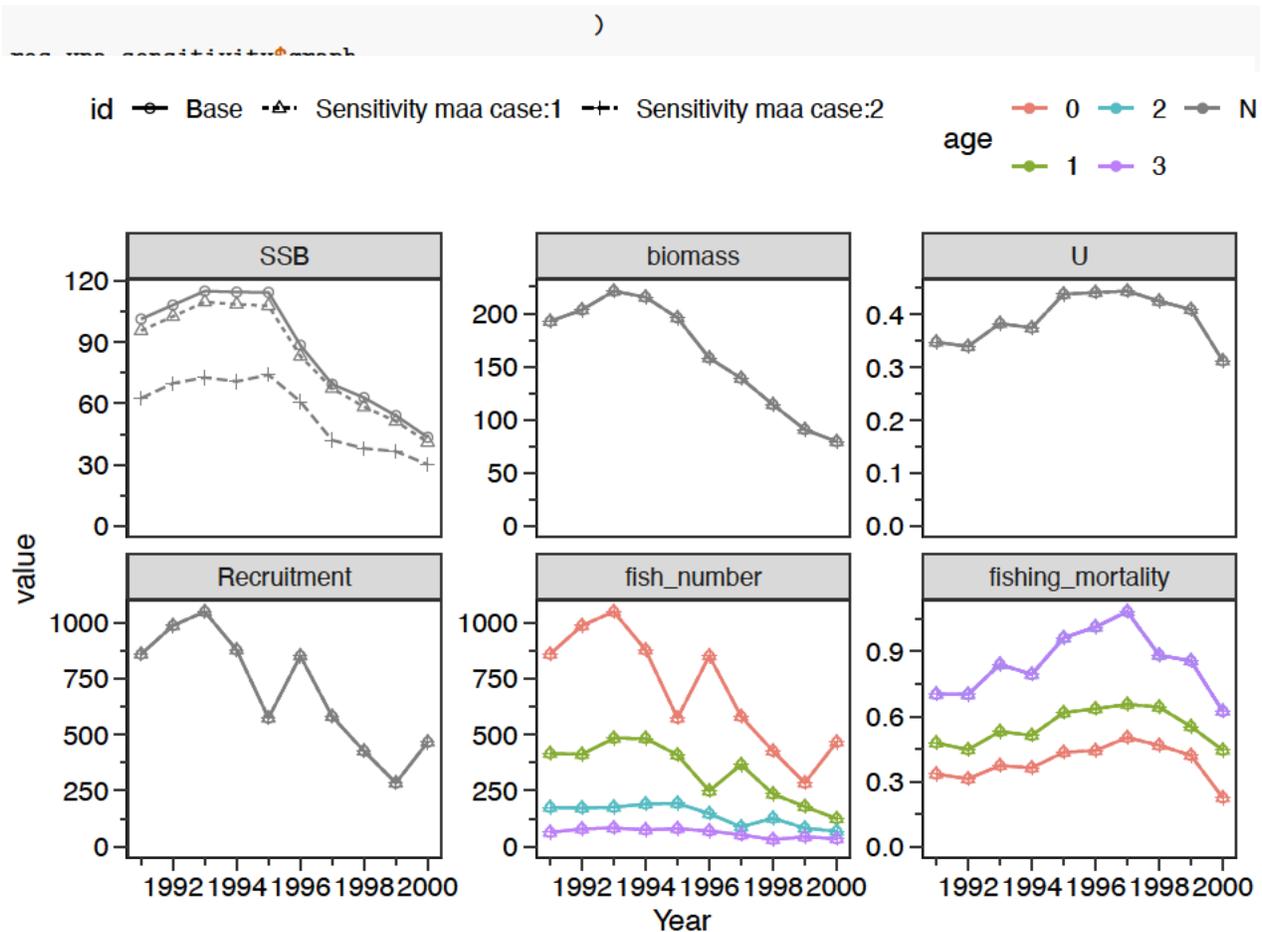
漁獲量計算と資源量計算で異なる年齢別体重を用いている場合には、漁獲物中の年齢別体重の設定についても感度分析ができます。この場合は、`what_replace = "waa.catch"` とします。value に入れる値の型は `index = waa` の場合と同じく、numeric または list 型です。(この例の場合には、漁獲量用の年齢別体重が設定されていないため、エラーとなります。)

```
res_vpa_sensitivity <- do_sensitivity_vpa(res_vpa_estb,
  what_replace = "waa.catch",
  value = c(0.8, 1.2),
  what_plot=c("SSB","biomass","U","Recruitment",
             "fish_number","fishing_mortality"),
  ncol=3
)
```

成熟率

成熟率を置き換える場合は、`what_replace = "maa"` とします。年齢別体重の場合と同様 value には、list に格納した行列を与えてください。

```
res_vpa_sensitivity <- do_sensitivity_vpa(res_vpa_estb,
  what_replace = "maa",
  value = list(matrix(rep(seq(0,1,length=4),10),nrow = 4),
             matrix(rep(c(0,0,.5,1),10),nrow = 4)
             ),
  what_plot=c("SSB","biomass","U","Recruitment",
             "fish_number","fishing_mortality"),
  ncol=3
)
```



α

α を置き換えたい場合は、`what_replace = "alpha"` とします。value には直接仮定したい値を入れてください。

```
res_vpa_sensitivity <- do_sensitivity_vpa(res_vpa_estb,
  what_replace = "alpha",
  value = c(0.9,0.8),
  what_plot=c("SSB","biomass","U","Recruitment",
    "fish_number","fishing_mortality"),
  ncol=3
)
```

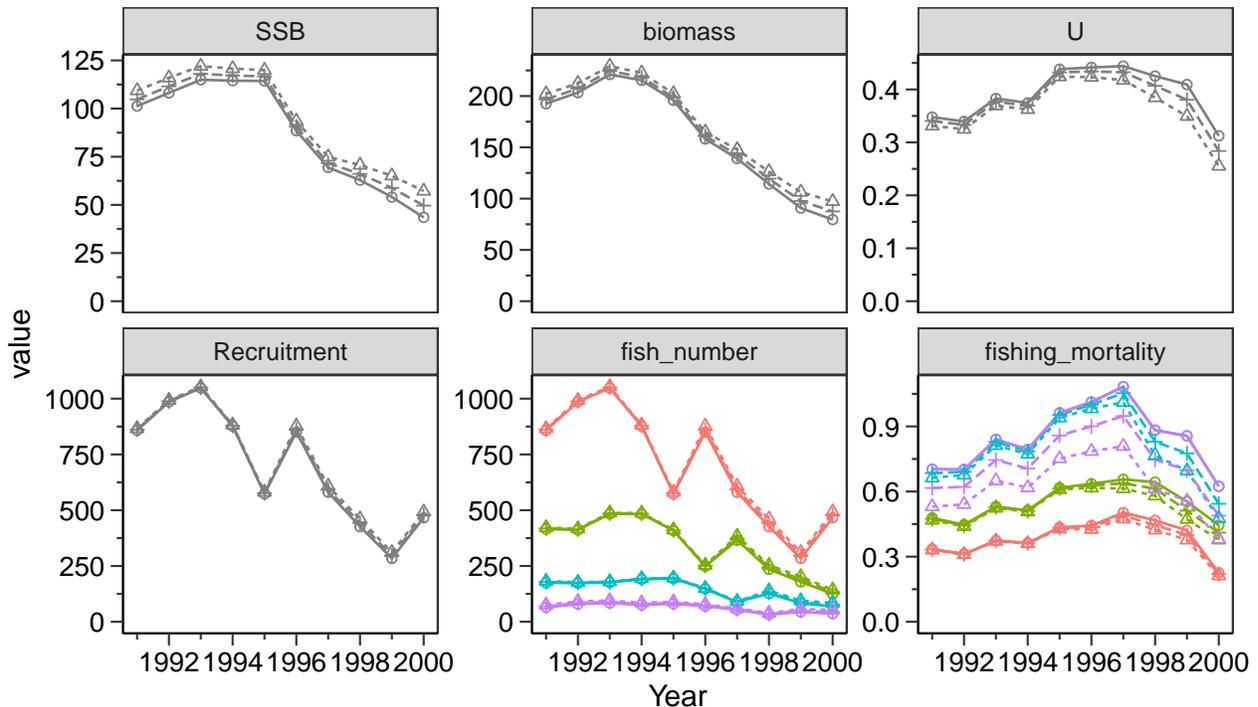
[1] "Warning! The estimated F for the older ages may not be accurate if $C \ll N$ is not satisfied for the o

[1] "Warning! The estimated F for the older ages may not be accurate if $C \ll N$ is not satisfied for the o

```
res_vpa_sensitivity$graph
```

id ○ Base △ Sensitivity alpha=0.8 + Sensitivity alpha=0.9

age
 ● 0 ● 1 ● 2 ● 3 ● NA



チューニング方法について

チューニングの有無で行える感度分析が大きく変わります。

- チューニングなし VPA(`tune=FALSE` で解析している) : `tf.year` について感度分析を行います
- チューニング VPA(`tune=TRUE` で解析している)
 - 選択率更新法 (`sel.update=TRUE`) : 選択率更新法の代わりに全 F 推定法を行います
 - 全 F 推定法 (`term.f="all"`) : 全 F 推定の代わりに選択率更新法を行います

チューニングなし VPA の場合

チューニングなし VPA の場合、最終年の F が過去何年分の平均になるかという仮定の部分の値を変えて感度分析を行います。

```
# チューニングなし VPA の場合
res_tmp <- vpa(vpadat_estb, tf.year=1996:1999, last.catch.zero = FALSE,
              Pope = TRUE, p.init = 0.8, tune=FALSE)
tmp <- do_sensitivity_vpa(res_tmp, what_replace = "tuning", value = list(1995:1999, 1998:1999))
# 2つの数列を list 型にして value を与えています
```

選択率更新法の場合

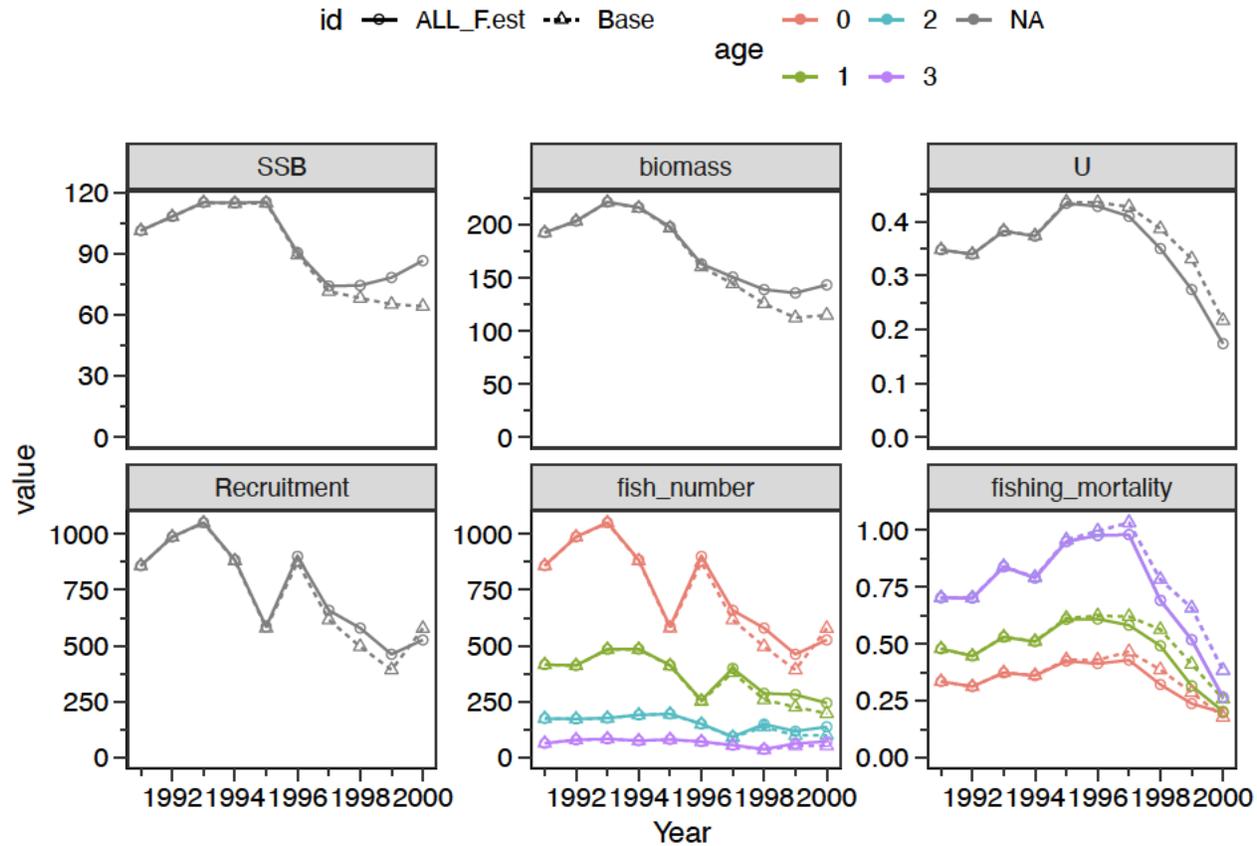
今回の例では選択率更新法で推定されているため、全 F 推定で感度分析が行われます。これまでと同様に、`what_replace = "tuning"` といれば大丈夫です。選択率更新法の代わりに感度分析で全 F 推定を行う場合、`value` に値を入れる必要はありません

```
res_tmp <- vpa(vpadat_estb, tf.year=1998:2000, last.catch.zero = FALSE,
              Pope = TRUE, p.init = 0.5, tune=TRUE, sel.update=TRUE)
res_vpa_sensitivity <- do_sensitivity_vpa(res_tmp,
```

```

what_replace = "tuning",
what_plot=c("SSB","biomass","U","Recruitment",
            "fish_number","fishing_mortality"),
ncol=3
)
res_vpa_sensitivity$graph

```



全 F 推定の場合

全 F 推定の場合、代わりに選択率更新法で感度分析が行われます。また先のチューニングなし VPA のように、最終年 F の仮定についても値を変えて、感度分析することができます。

```

# 全 F 推定法の場合
res_tmp <- vpa(vpadat_estb, tf.year=1997:1999, last.catch.zero = FALSE,
              Pope = TRUE, p.init = 0.5, tune=TRUE, sel.update=FALSE, term.F = "all")
tmp <- do_sensitivity_vpa(res_tmp, what_replace = "tuning", value = list(1995:2000))
# 数列を list 型にして value を与えています
# チューニングなし VPA と同じです

```

リッジペナルティ λ について

全 F 推定でリッジ VPA を行っている場合 ($\lambda > 0$) には λ の値についての感度分析も実施してみてください。

```

# リッジ VPA の場合
res_tmp <- vpa(vpadat_estb, tf.year=1997:1999, last.catch.zero = FALSE,
              Pope = TRUE, p.init = 0.5, tune=TRUE, term.F = "all", lambda = 0.2)
tmp <- do_sensitivity_vpa(res_tmp, what_replace = "lambda", value = c(0,0.1,0.5,0.7),
                          what_plot=c("SSB","biomass","U","Recruitment",

```

```

"fish_number", "fishing_mortality"),
ncol=3)
# lambda の値を "numeric 型" で与えています
# list 型ではないので気を付けてください

```

b (hyperstability/depletion)

b を推定した場合、b を推定しない (b=1) 場合について、感度分析を実施してください。what_replace = "b" とすると b について感度分析できます。また、b を推定していない場合も適宜 b を推定 (あるいは固定) して、感度分析を行ってみてください。

- b の値を与えたい場合、numeric 型、または list 型 (複数与えたい場合) で値を入れる (ベースケースで b を推定しているが、b を推定しない感度分析を実施する場合は value=1 を与える)
- b を推定したい場合、"b.est" と入れる

```

# b 推定しているが、b を固定する場合
tmp <- do_sensitivity_vpa(res_vpa_estb, what_replace = "b", value = list(c(1,1),c(1.2,1.2)))
# b を固定しているが、b を推定する場合
tmp <- do_sensitivity_vpa(res_vpa_estb, what_replace = "b", value = c("b.est"))

```

2. レトロスペクティブ解析

- 過去~7年分くらいのデータを削除して資源計算をしない場合に、資源量や F の推定値が傾向を持って変化するようなレトロスペクティブパターンが見られるかどうかを調べます。その際には
 - 非線形パラメータ b を推定しないか
 - リッジの場合、λ を推定しないかという問題がありますが、デフォルトは、これらのパラメータは固定 (最終年で推定されたものを用いる) して計算してみます。
- vpa(rec) で rec 引数を指定しても利用可能です
- ただし、指定している rec.year より遡ると、効果が消えます

レトロスペクティブ解析は do_retrospective_vpa 関数を用いることで、

- 結果の出力
- Mohn's rho の出力
- b の再推定結果の値
- 作図

までを一貫して行えます。

引数として、

- res : vpa 計算の結果のオブジェクトの他に
- n_retro : レトロスペクティブ解析を何年さかのぼって実行するか (デフォルトで 5 年)
- b_reest : b(hyperstability/depletion) をレトロスペクティブ解析内で再推定するか
- what_plot : 作図で何を出力するか
- plot_year : 作図したい年 (作図の x 軸の範囲)
- ncol : 作図の列数

があります。

今回の解析結果 res_vpa_estb を使って、7 年分をさかのぼってレトロスペクティブ解析する場合、以下のようコードします。

```

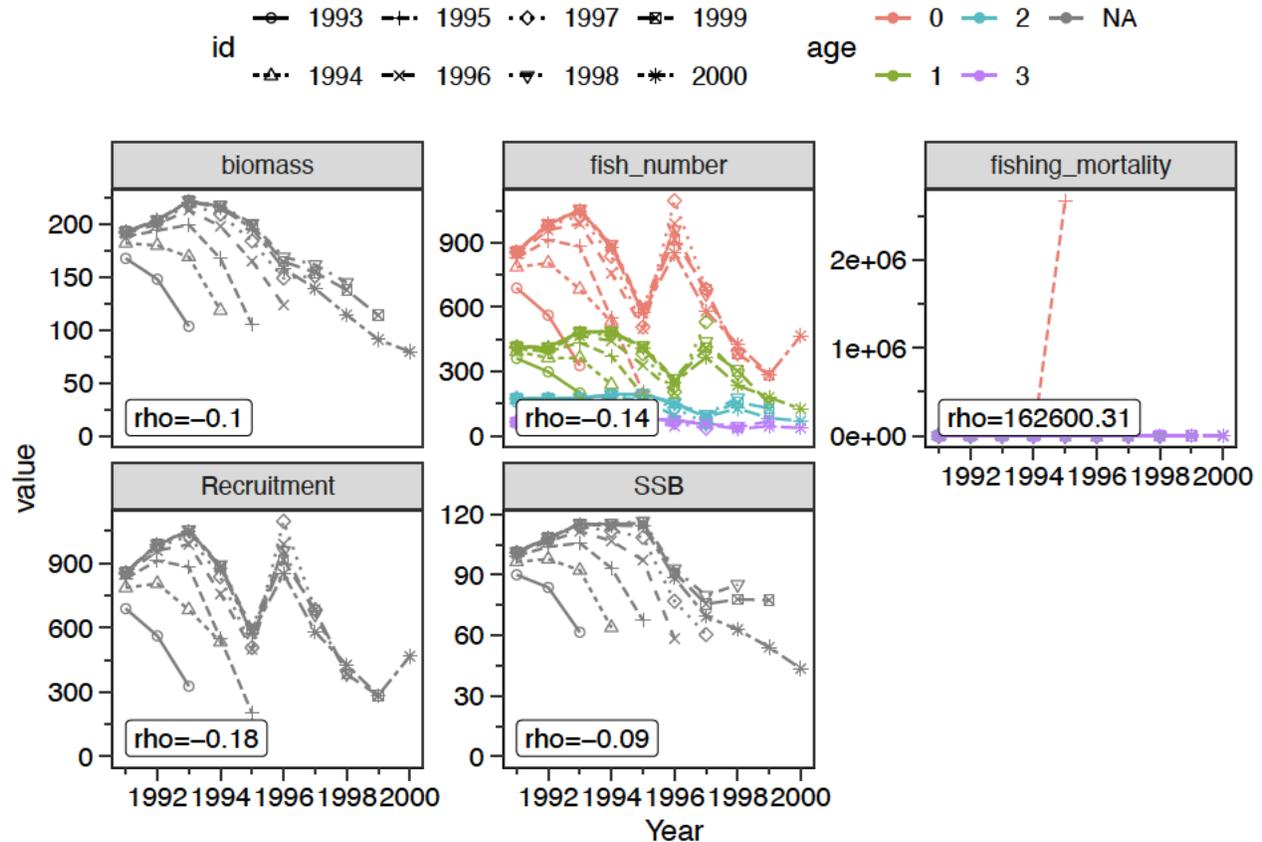
res_vpa_retrospective <- do_retrospective_vpa(res_vpa_estb,
                                             n_retro = 7,
                                             b_reest = FALSE)
res_vpa_retrospective$mohn_rho

```

N B SSB R F
 -1.353966e-01 -1.019165e-01 -9.331420e-02 -1.829777e-01 1.626003e+05

結果は list 型式でまとめてあり、\$ を使って呼び出すことが可能です。result は VPA の推定結果、mohn_rho は mohn の rho の値、graph はグラフ、b_res は b の再推定結果です。また Mohn's rho は、グラフ中にも併記してあります。ここで、漁獲係数の Mohn's rho は年齢別漁獲係数を全て足し合わせた値となっています。

res_vpa_retrospective\$graph



レトロスペクティブ解析中で b の再推定を行う場合

また、b をレトロスペクティブ解析内で再推定したい場合は以下のようにコードしてください。

```
res_tmp <- vpa(vpadat_estb, tf.year=1997:1999, last.catch.zero = FALSE, Pope = TRUE,
              p.init = 0.5, tune=TRUE, sel.update=FALSE, term.F = "all", b.est = TRUE)
# まず b を推定する vpa を実行しています
```

```
res_vpa_retrospective <- do_retrospective_vpa(res_tmp,
                                              n_retro = 4,
                                              b_reest = TRUE # ここを TRUE にしてください!!
                                              )
```

res_vpa_retrospective\$b_res

```
# A tibble: 6 x 4
  `2000` `1999` `1998` `1997`
  <dbl> <dbl> <dbl> <dbl>
1  1.11  1.04  0.792  0.603
2  0.564 0.736  0.826  0.612
3  0.854 0.679  0.356  0.257
```

```

4 0.578 0.523 0.603 0.383
5 0.657 0.892 0.628 0.480
6 1.14 1.33 1.47 1.11

```

このように各レトロスペクティブ解析ごとに、 b を推定していることが確認できました。(ダミーデータなので変な値となっています)

2 段階法の場合

VPA を 2 段階法で行っている場合は、`res_step1` 引数に 1 段階目の VPA 結果のオブジェクトを与えてください。

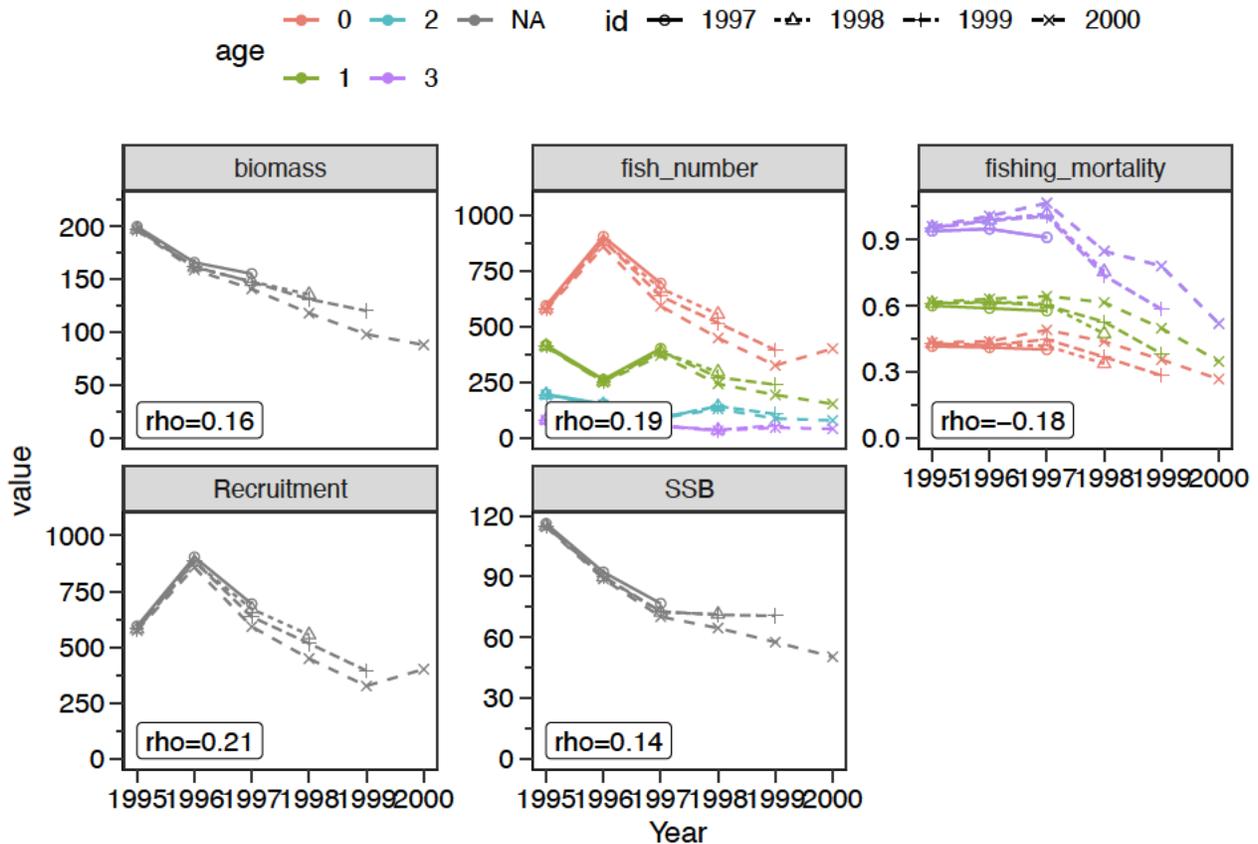
```

res_tmp1 <- vpa(vpadat_estb, tf.year=1997:1999, last.catch.zero = FALSE, Pope = TRUE,
               p.init = 0.5, tune=FALSE, term.F = "max", b.est = FALSE, fc.year=1998:2000)
res_tmp2 <- vpa(vpadat_estb, tf.year=1997:1999, last.catch.zero = FALSE, Pope = TRUE,
               p.init = 0.5, tune=TRUE, term.F = "max", b.est = TRUE, fc.year=1998:2000,
               sel.f = res_tmp1$saa$`2000`) # sel.f で 2 段階目の選択率の初期値を与えます

res_vpa_retrospective <- do_retrospective_vpa(res_tmp2,
                                              n_retro = 3,
                                              plot_year = c(1995, 2000),
                                              b_reest = TRUE,
                                              res_step1 = res_tmp1
                                              # ここに 1 段階目のチューニングなし VPA を与えてください
                                              )

res_vpa_retrospective$graph

```



3. パラメータの収束の確認 (jitter analysis)

F の推定結果が初期値に依存せずに、安定した結果となっているかを診断するために、jitter analysis (初期値を乱数で生成し、vpa 計算を繰り返し実施する) をおこないます。vpa 関数では推定パラメータの初期値は引数 `p.init` で与えられますが、全 F 推定法の場合、最終年の各年齢の F の初期値になります。

F の結果が初期値に依存する場合、初期値に応じて推定結果も変わることが考えられます。このような場合、パラメータの初期値への依存が高く、局所的な最適解に引っ張られている可能性などが考えられます。すなわち、データの持つ情報量が低いことが考えられるため注意が必要です。jitter analysis はこのようなことがおこっていないか確認するための解析です。

初期値は対数を取った 0.001 から 2 の間を 10 刻みにした後、指数を取っています。刻みの数は `n_ite` で指定できます。

`do_estcheck_vpa` 関数で、自動的に jitter analysis とその結果の作図が行われます。

基本的に、vpa 計算の結果のオブジェクトのみを与えれば計算が行われます。その他に引数として、

- `what_plot`: 作図したい年齢
- `TMB`: TMB で解析する (始めて実行する場合は `use_rvpa_tmb()` を先に実行)

を指定します。コマンドは以下の通りです。

```
res_vpa_jitter <- do_estcheck_vpa(res_vpa_estb, n_ite=10)
```

```
Maximum likelihood in jitter analysis is: 23.60018
```

```
Likelihood with estimated parameters is: 23.60018
```

最初に引数で与えた VPA 結果について、

- ヘッセ行列の対角成分が正であるか
- パラメータの推定結果が収束しているか

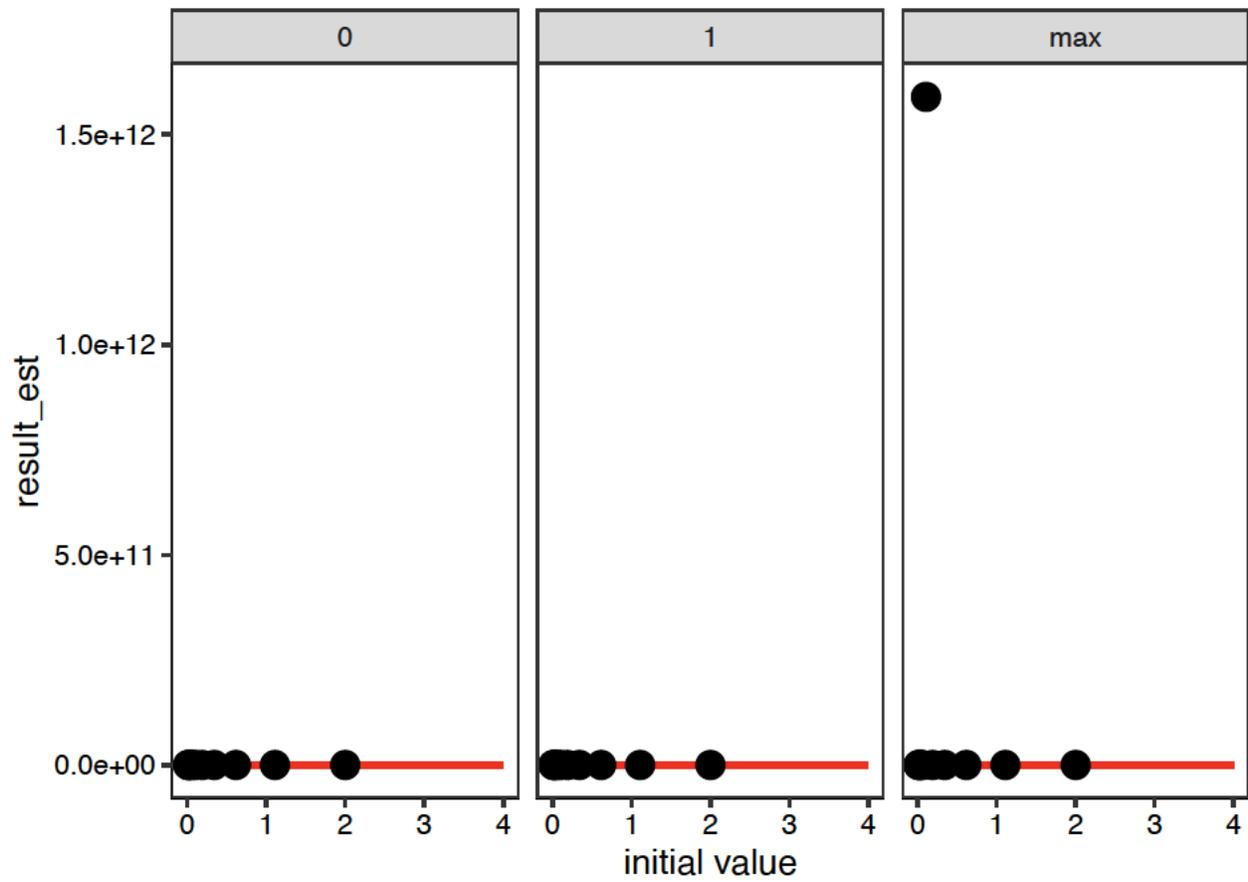
を知らせてくれます。ヘッセ行列の対角成分に負がある (本来正になる値であり、パラメータ推定が失敗して)、あるいは収束していない場合、ここで関数が止まります。再度、VPA 関数を使って解析してください。(初期値を変える、全 F 推定の場合パラメータを減らす、資源量指数の非線形性を考慮している場合は `b` を与えるあるいは 1 にする、などの工夫が必要となります)

次に、関数内で jitter analysis が行われます。`n_ite` 数分の VPA 計算を行います。生成する初期値の数は **10** 個 (`n_ite = 10`) で十分です。推定値と初期値の大体の関係が見れば問題ありませんが、初期値に応じて、F が異なる値で推定される (大きな値に発散せずに) 場合には注意が必要です。その結果が `res_vpa_jitter` 内にあり、それぞれ、

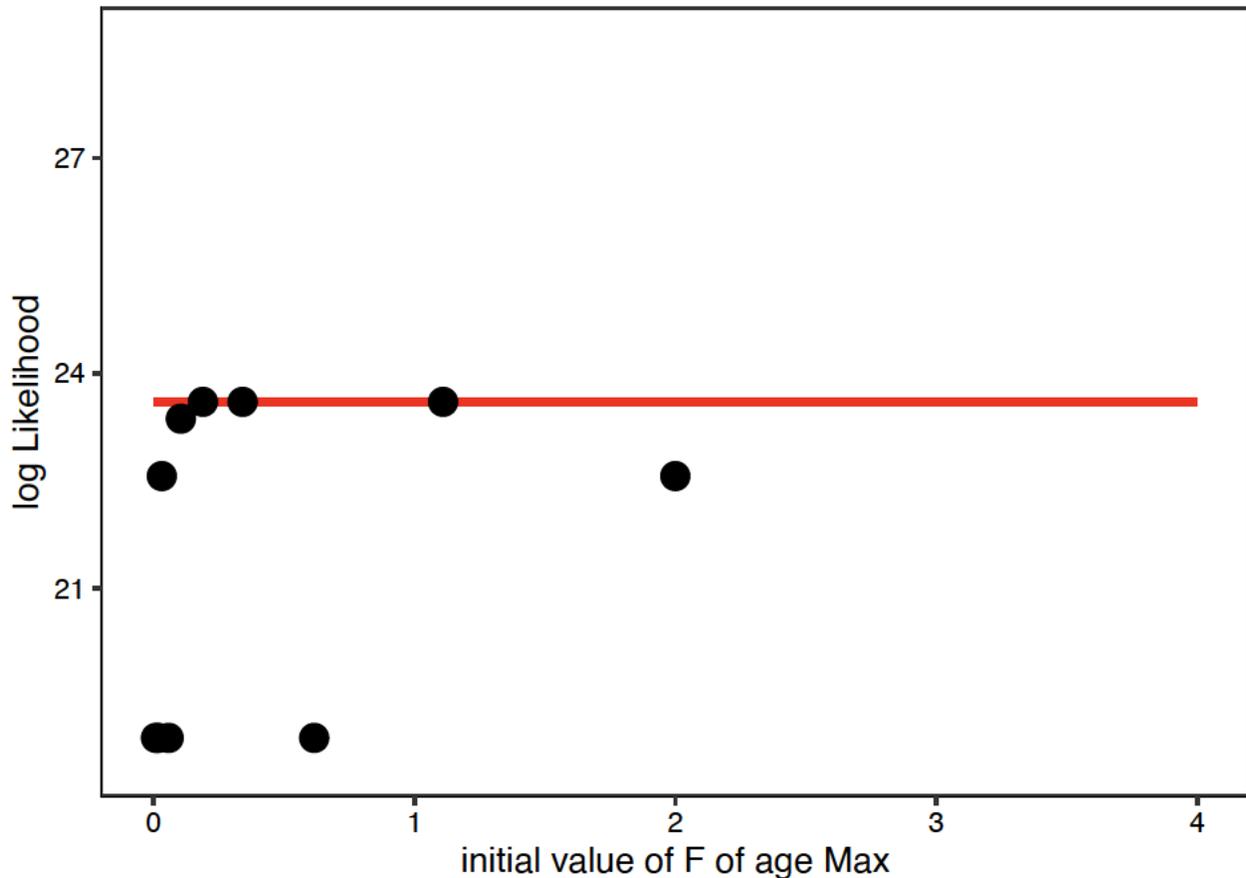
- `$initial_value` に初期値
- `$p_name` 推定したパラメータ (漁獲係数) の年齢
- `$value` 推定値や尤度、収束の値を返す
- `$graph` 初期値と推定値 or 尤度の関係の図

が入っています。

```
res_vpa_jitter$graph$estimated # 推定された F
```



`res_vpa_jitter$graph$likelihood` # 対数尤度 (大きいほど良い)



ここで赤い横線は与えた結果の推定値と尤度を表しています。

結果の図を見ると、いくつかの初期値のセットでは F が 0 におちたり、 F が高い値になってしまっていることがわかります。これらのときの尤度は、推定値が得られたときの対数尤度（赤線）よりも小さい値になっていることから、推定パラメータから計算される尤度はほかの初期値を与えたときの推定値から得られる尤度と比較しても最大になっている（＝最尤推定値が得られている）ことが確かめられました。

また VPA は、最終年の F でしか指標値とのフィットをコントロールできないことから、 F が Inf などの大きな値になることもあります。このような場合に、推定結果の最大対数尤度より、 F が発散した場合の最大対数尤度の方が大きくなることがあります。これは F が収束していないだけですので、こういった場合は推定と初期値の関係に問題ありません。（逆にこういった F が大きく発散しない F を探すという方が実際の資源に適用する場合には多いかと思います）

関数を実施したとき、“Maximum likelihood in jitter analysis is: , Likelihood with estimated parameters is :” という表示が出ますので、上下の値が一致している、または上の値よりも下の値のほうが大きければ OK です。逆に、上の値（Maximum likelihood in jitter analysis）のほうが大きい場合には、上述のように F が発散しているため問題ないか、最尤推定値は他にある可能性がある場合の 2 通り考えられます。後者の場合は初期値を変えて計算しなおしてください。

4. 残差プロット

残差プロットはモデル診断の中でも、最も重要なパートの 1 つです。残差とは、CPUE とその予測値 (qN^b) の差のことを指し、これが小さいほうがよりフィッティングの良いモデルとなります。残差の対数（対数残差）プロットは 0 を平均とした分布になります。また、残差プロットが時系列的に一貫して増える（または減る）傾向（自己相関）がみられる場合、モデルとデータの間には何らかの齟齬がある可能性があり、注意が必要です。この他、他の残差に比べて著しく大きい残差は、そのデータ（CPUE）が外れ値である場合があります。外れ値はパラメータ推定に大きな影響を及ぼすので、外れ値と思われるデータを解析に含めるか再考する必要があります。

ります。ただし、いずれも時系列データを扱っていると往々に起こりうることで、外れ値がある、あるいは自己相関係数が有意であっても、結果が妥当そうであれば問題ありません。残差プロットを行うことでこれらの傾向を一目にして把握することができます。

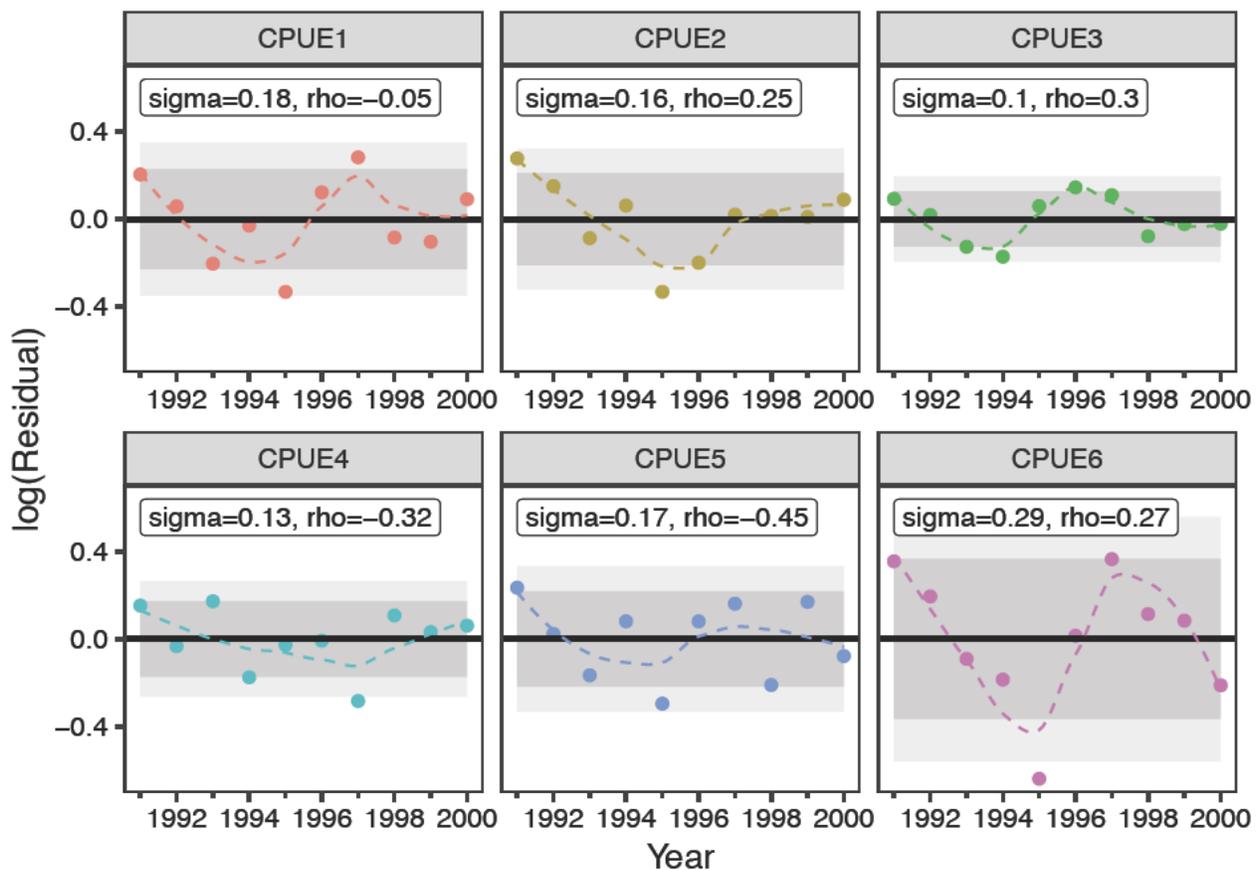
残差プロットは `plot_residual_vpa` 関数で実行できます。大きく 3 種類のプロットを書くことができます。

- `$year_resid`: 対数残差を時系列にプロットしたもの。先述の残差の傾向と外れ値の把握に利用。標準化残差のプロットについては `$year_sd_resid` 中にある。
- `$fitting_Index`: CPUE と予測 CPUE をそれぞれ重ねてプロットしたもの。データと推定結果のフィッティングを見るもの。
- `$abund_Index`: 資源量/資源重量/親魚重量とその予測 CPUE との関係。赤い線が推定した資源量とその指数の線形/非線形性を表す。

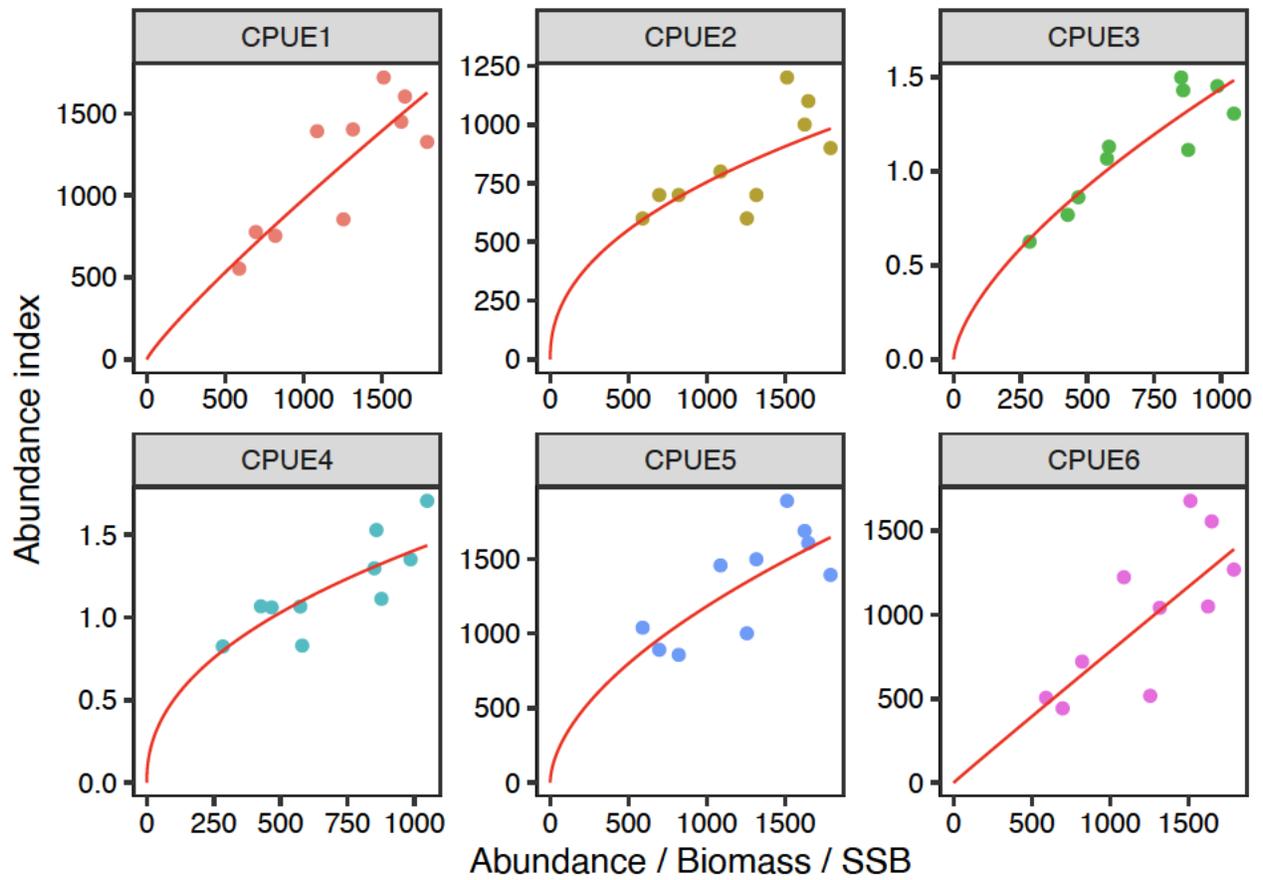
また `index_name` 引数中に、各指標の名前を与えるとプロットに反映されます (ない場合は `Index01`、`Index02`... となります)。与える場合、用いた指標の数分引数に入れてください。

```
plot_example <- plot_residual_vpa(res_vpa_estb, index_name = c("CPUE1", "CPUE2", "CPUE3",
                                                             "CPUE4", "CPUE5", "CPUE6"),
                                plot_smooth=TRUE)
```

```
plot_example$year_resid
```

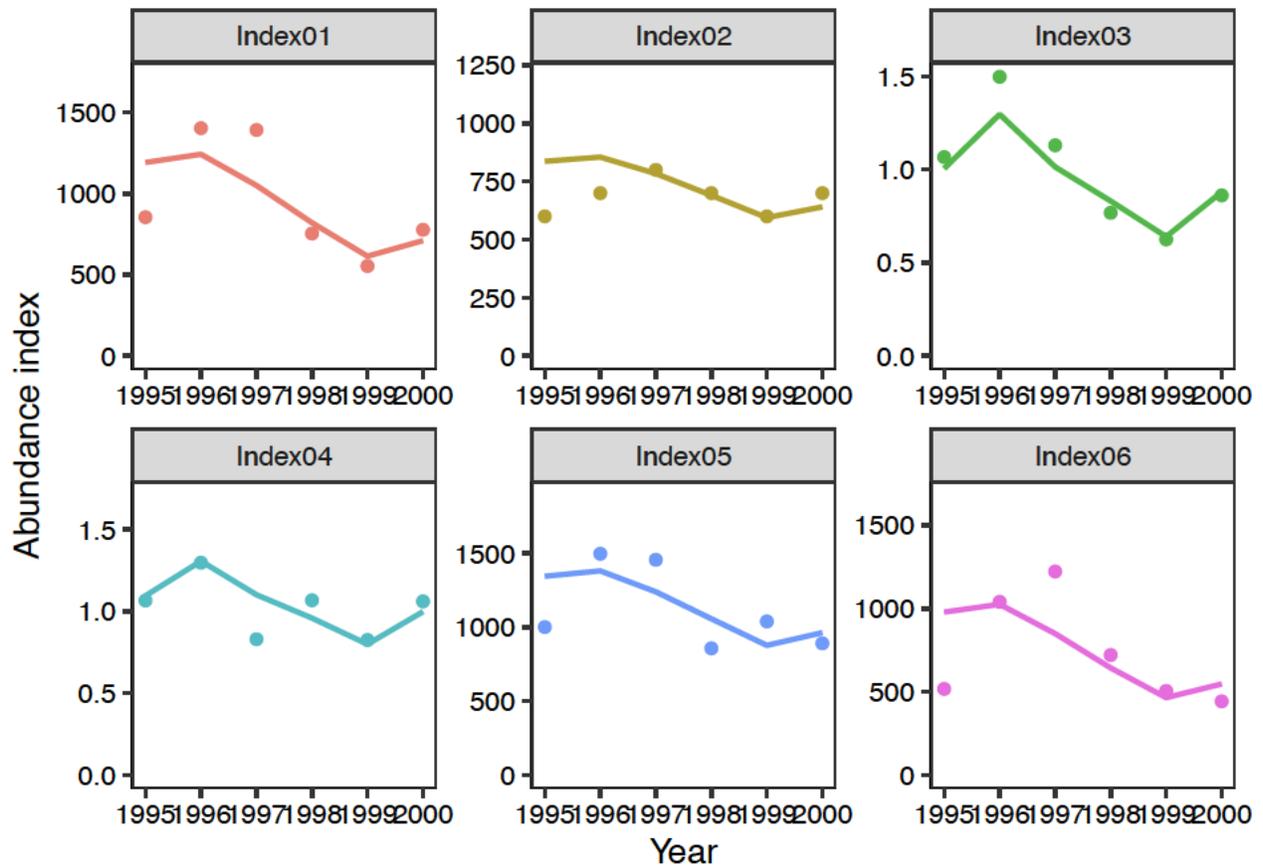


```
plot_example$abund_Index
```



また、`plot_year` 引数でプロットしたい年を指定することも可能です。

```
plot_example <- plot_residual_vpa(res_vpa_estb, plot_year = 1995:2000)
plot_example$fitting_Index
```



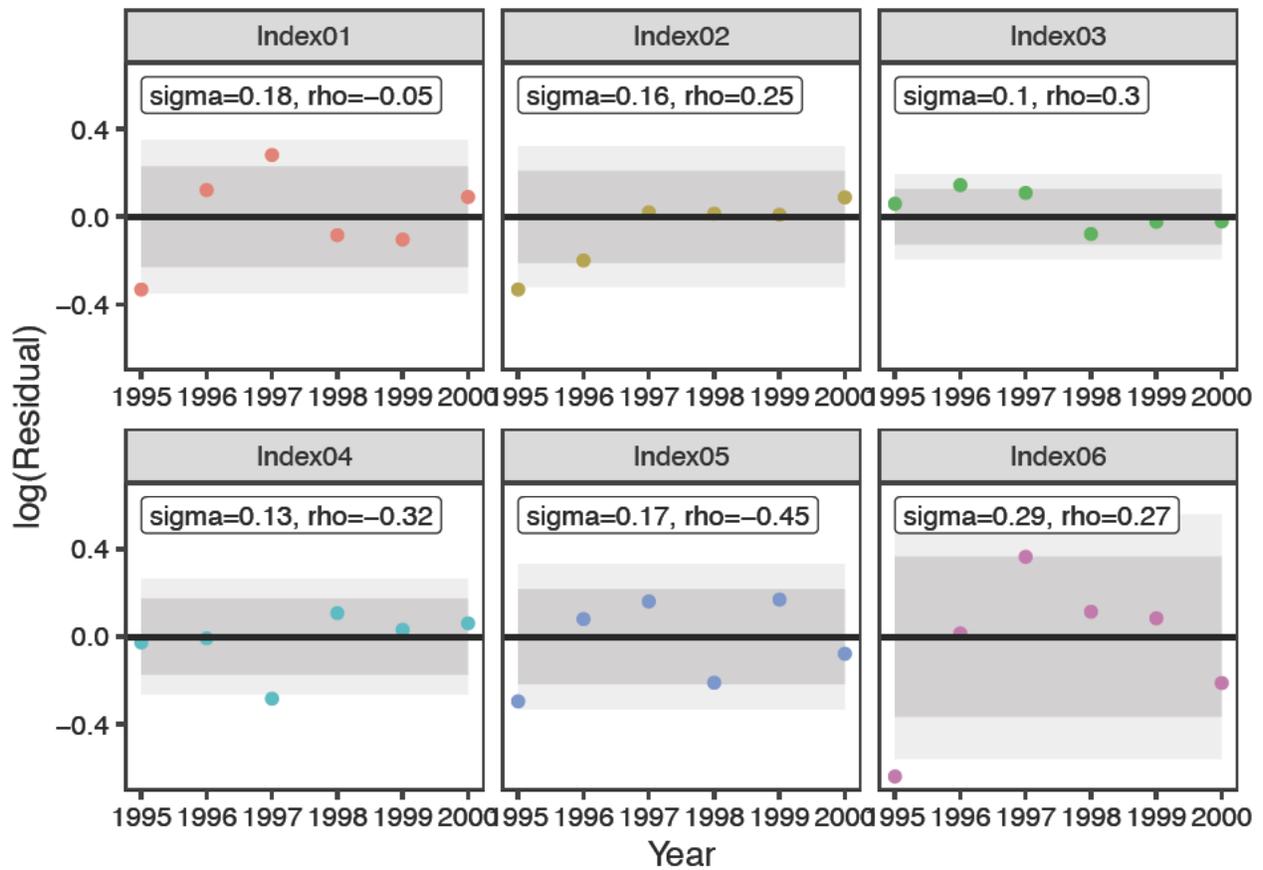
これらの残差プロットは今回のバージョンから、vpa 関数実行時に引数 `plot = T` にすると自動生成されるようになっています。モデル診断の最も基本パートとなるので、是非活用ください。

2021 年度変更点

先程の図の通り、残差プロットに信頼区間が明記されるようになりました。信頼区間は濃い色が 80%、薄い色が 95% のそれぞれ信頼区間となっております。

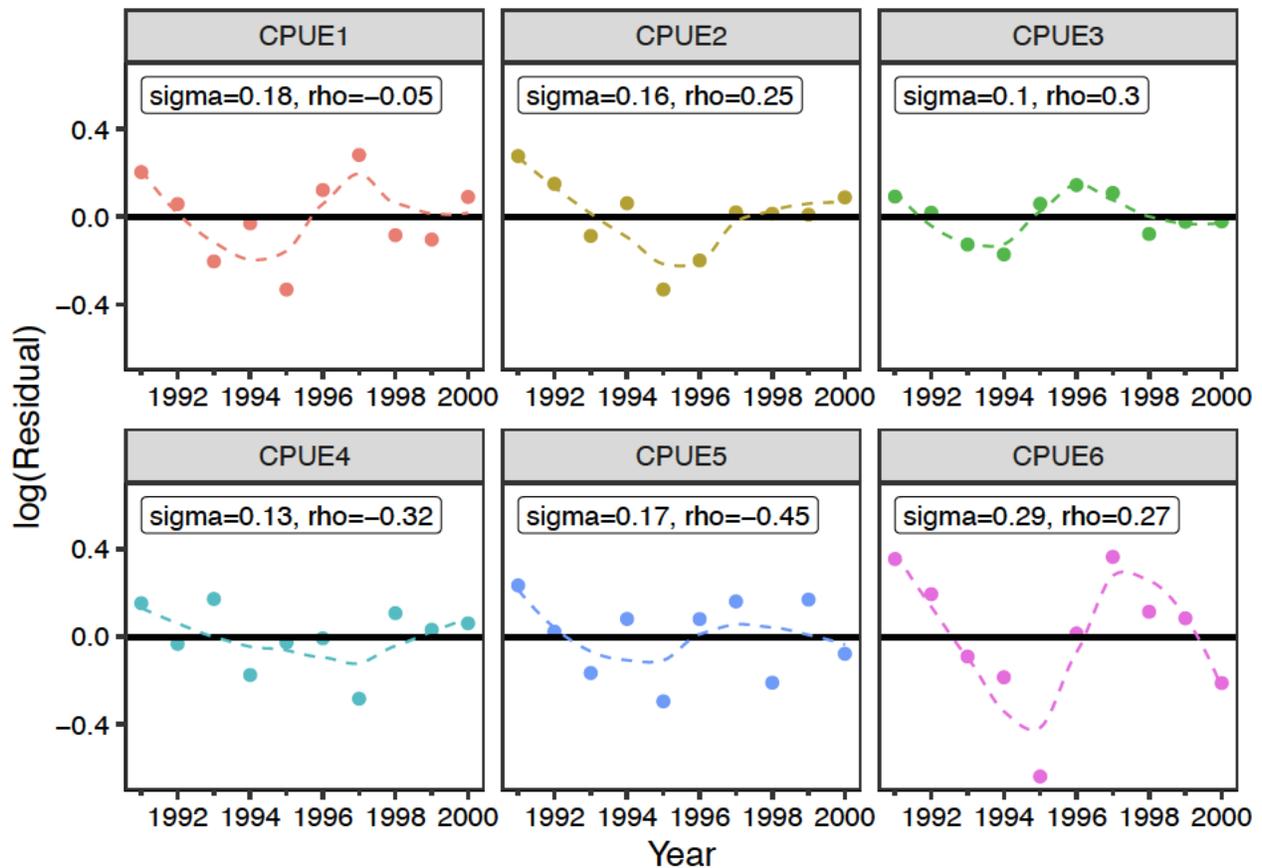
これによって、外れ値の検出ができます。特に、最近年に外れ値がある場合、その値が資源量推定結果に与える影響は大きいことが想定されるので、後述のジャックナイフ法などを通して結果の妥当性を評価してください。

```
plot_example$year_resid
```



もし信頼区間の影が邪魔で消したい場合は、`resid_CI=FALSE` としてください。

```
plot_residual_vpa(res_vpa_estb,
  index_name = c("CPUE1", "CPUE2", "CPUE3",
                 "CPUE4", "CPUE5", "CPUE6"),
  plot_smooth=TRUE,
  resid_CI = FALSE)$year_resid
```



また、自己相関係数も明記されるようになりました。もしこの自己相関係数が有意に大きいあるいは小さい場合、 $\rho=0.6^*$ のように、*印が付きます。この場合は、残差の独立の仮定が満たされていません。

ただし、満たされていないから、一概にモデルとしての妥当性が不十分で、棄却されるというわけではありません。(主観的で恐縮ですが)

ジャックナイフ法によるデータの影響力の検証等を通して、結果の妥当性が確認できれば、問題がないと考えます。

こういった妥当性の検証のプロセスを説明することが大切だと考えます。

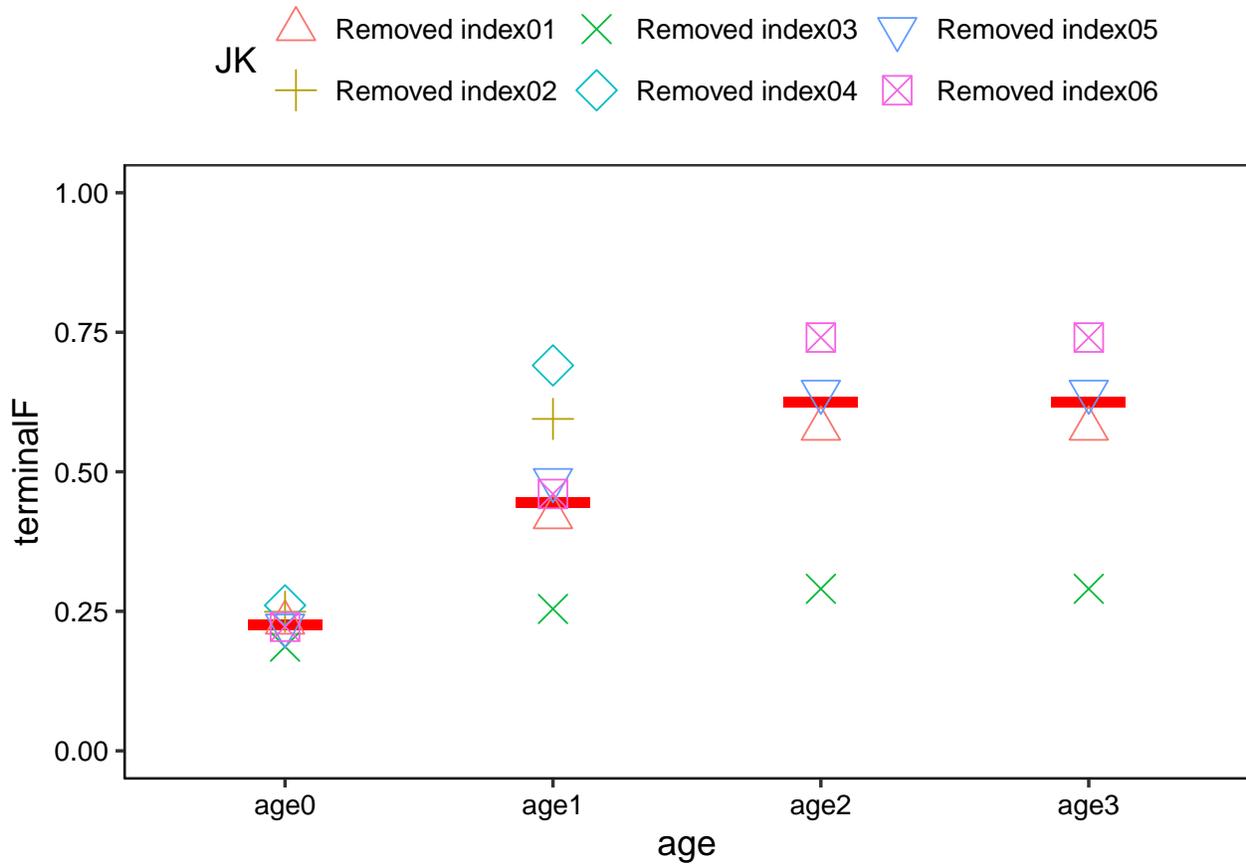
5. ジャックナイフ法

観測値(資源量指数)を1つずつ抜いて解析を行うことで、影響力の強いデータや外れ値を検出する方法です。do_jackknife_vpa 関数で自動で観測値を抜いて再解析を行います。

- plot_year に作図したい年(x軸の範囲)を指定することもできます
- scale_value に任意の値を与えることで、プロットの点の形を任意で指定できます(ベースモデル+ジャックナイフの数分与えてください)
- => ジャックナイフ法の他に、感度分析、レトロスペクティブ解析でも指定できます

引数 method はデフォルトで"index"となっており、資源量指数の種類ごとに取り除いて解析を行います(つまり、資源量指数の種類分、結果が返ってきます)。一方、それぞれの種類を年ごとに取り除きたい場合は"all"を選択してください。この場合、資源量指数の種類 × 年数分、取り除いて計算するためデータ数によっては多くの計算時間を必要とする点に気を付けてください。

```
res_vpa_jackknife <- do_jackknife_vpa(res_vpa_estb,
                                     what_plot = c("SSB", "biomass", "U", "Recruitment"),
```

6. ブートストラップによる信頼区間推定

ブートストラップ法は乱数でデータを再生成し、信頼区間等を推定する方法です。ここでは

- CPUE について乱数生成する方法
- 年齢別漁獲尾数について乱数生成する方法

それぞれについて、新規関数とともに紹介します。またパッケージの更新速度の都合上、ブートストラップ標本数は **10** 個としていますが、95% 信頼区間などを正確に計算したい場合は **1000** 回（デフォルト）ほど必要です。ただし、データの特性や推定された `sigma` の大きさにも依存するので、200 回や 500 回でも、1000 回と信頼区間がそれほど変わらない場合は、少なくとも構いません。

一方で大まかな幅や各パラメータの相関関係だけを知りたい場合には、**100** 回ほどでも問題ありません。

ブートストラップ法では VPA 計算を繰り返し行うことから時間がかかるので、TMB の仕様を推奨します。最初に vpa 計算する時に `TMB=T` にしておくと、以下のブートストラップ計算も TMB で行われます。

1. CPUE のブートストラップ法について

先述の通り、ブートストラップ法はデータをリサンプリングする方法ですが、VPA では残差をリサンプリングすることで、データ (CPUE) を再生成します。frasyr の中にはブートストラップ法を自動で行う `boo.vpa` 関数があり、`type = "index"` とすると、CPUE のブートストラップを実行します。また、その結果を `plot_boot` 関数に与えることで、信頼区間のある資源動態の描画ができます。

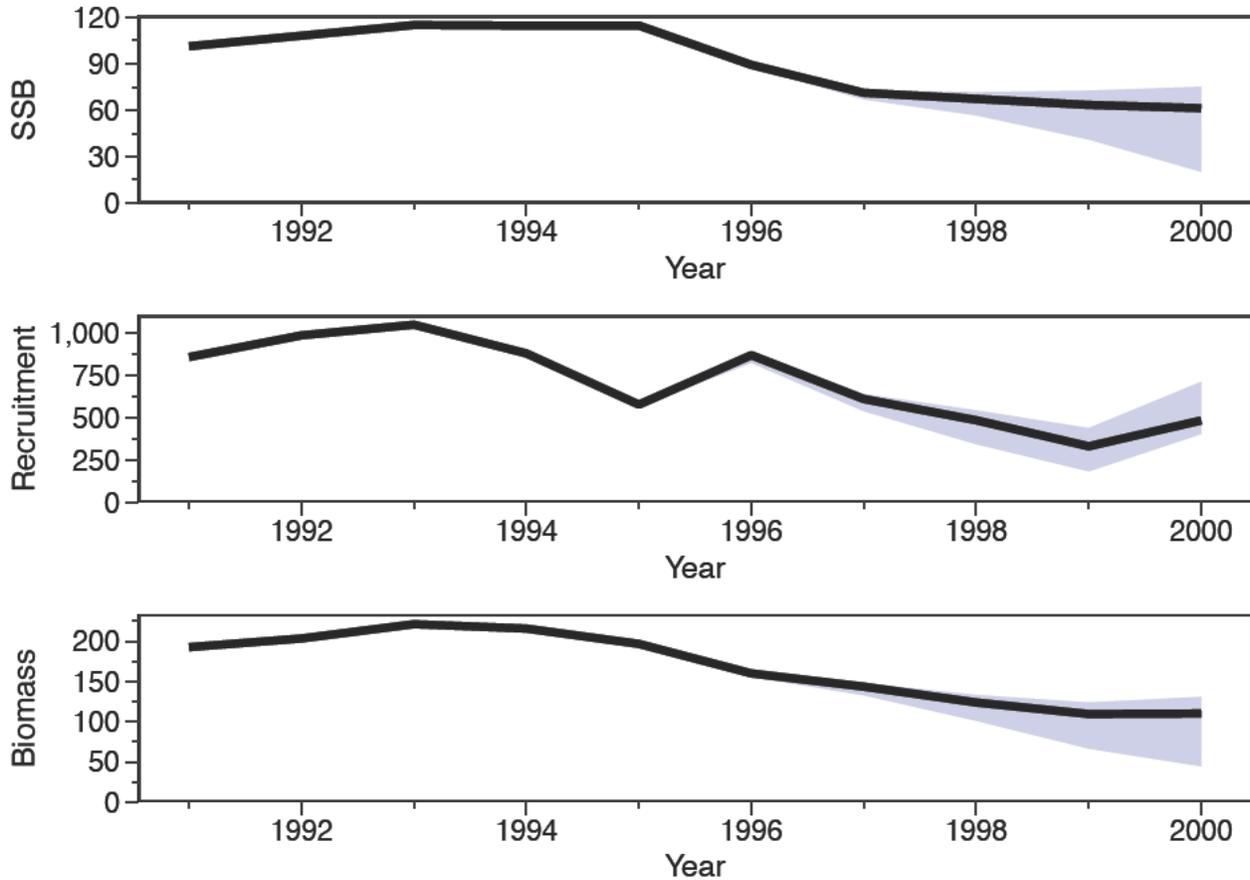
- 引数 `method` でブートストラップ法の方法を変えることができます。基本はパラメトリックブートストラップで問題ありません。ノンパラメトリックブートストラップ法では残差を重複ありでランダムにデータに足し合わせて再生成します。一方、パラメトリックブートストラップは平均 0、分散が残差の分散となるような正規分布から乱数でデータを生成して、それぞれ解析する方法です。

- "p": パラメトリックブートストラップ法 (デフォルト)
- "n": ノンパラメトリックブートストラップ法
- "r": 残差のスミージング後に Bootstrap-t 法を行う方法。

- plot_boot 関数では、引数 ci_range で信頼区間の値を指定できます。デフォルトは 0.95 です (95% 信頼区間の表記)。

```
res_boot1 = boo.vpa(res_vpa_estb, type = "index", B_ite = 10, method = "p")
plot_boot1 = plot_boot(res_boot1, ci_range = 0.95)
```

```
library(patchwork)
plot_boot1$plot_ssb/plot_boot1$plot_rec/plot_boot1$plot_biomass
```



このように SSB、加入量、バイオマスそれぞれについて 95%信頼区間とともに資源動態の作図ができます。ここで使っている patchwork パッケージは、ggplot2 の作図オブジェクトを後からくっつけることが可能なパッケージです。frasyr インストール時に一緒にインストールされます。

ブートストラップ法は資源量の信頼区間だけでなく、パラメータの信頼区間も算出出来ます。ここでは最終年最高齢の F の信頼区間を出しています。

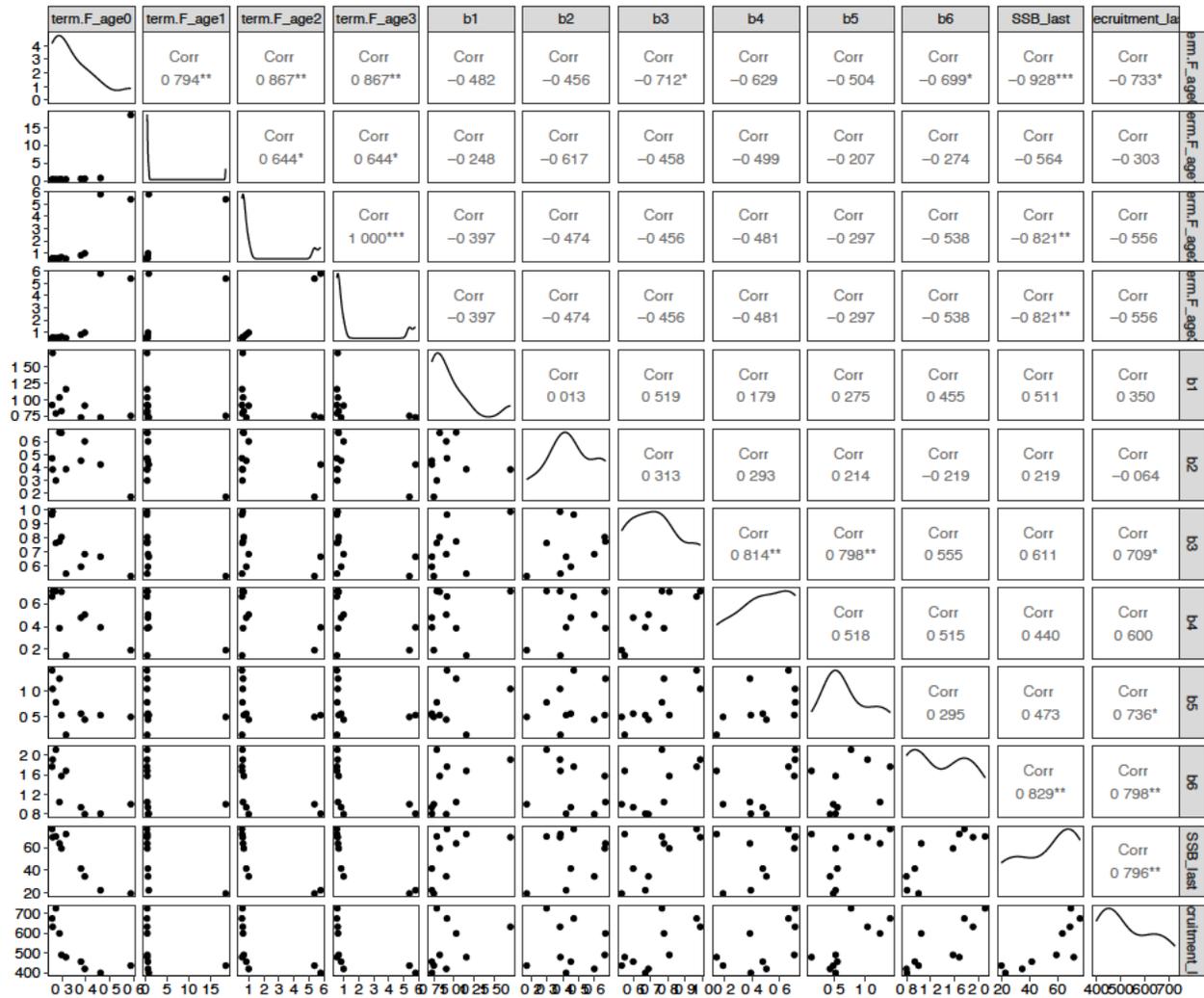
```
res_tmp <- numeric()
for(i in 1:10) res_tmp[i] <- as.numeric(res_boot1[[i]]$Fc.at.age[4])
quantile(res_tmp, probs = c(0.025,0.5,0.975))
```

```
      2.5%      50%      97.5%
0.5340341 0.6306449 5.6940095
```

この他、各パラメータの相関関係について、ブートストラップ標本を基にプロットが得られます。相関係数が非常に高い (0.9 以上など) パラメータのペアがある場合、その 2 つのパラメータを同時に推定することが難

しいことを示している場合がありますので、その場合はモデルの構造をよく検討してください。

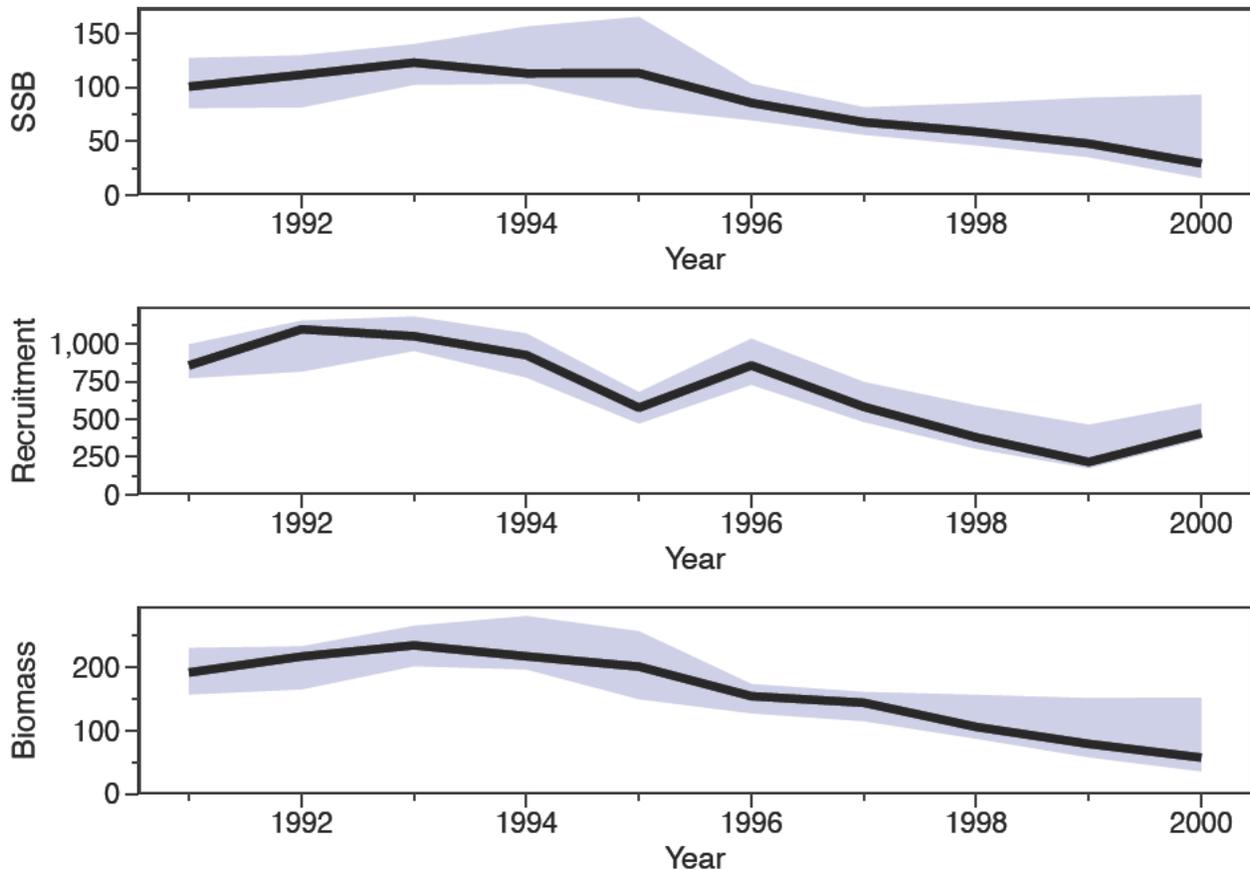
```
plot_boot1$plot_cor
```



2. 年齢別漁獲尾数について乱数生成する方法

VPA は年齢別漁獲尾数 (catch at age) が真であると仮定して推定するため、年齢別漁獲尾数の不確実性を評価することが出来ないモデルです。boo.vpa 関数では type="caa" 引数を与えると、年齢別漁獲尾数についてブートストラップ法を実行可能です。ブートストラップでは、毎年の総漁獲尾数を対数正規分布 (B_cv で変動係数を設定、デフォルトは 0.2) で振らせたあと、漁獲物の年齢組成を多項分布で振らせます (そのさいの誤差の大きさは ess 引数で設定、デフォルト値は 200、大きいほど誤差が小さくなります)。

```
res_boot2 = boo.vpa(res_vpa_estb, type = "caa", B_ite = 10, B_cv=0.2, ess=200)
plot_boot2 = plot_boot(res_boot2, ci_range = 0.95)
plot_boot2$plot_ssb/plot_boot2$plot_rec/plot_boot2$plot_biomass
```



一般に、VPA の資源量推定結果を信頼区間と重ねてプロットすると、先程の `plot_boot1$plot_ssb` のように最近年で信頼区間が広く、過去にさかのぼるほど信頼区間は狭くなっています。

これは決して過去の推定結果が正しいからではなく、確率変数として考えている資源量指数の不確実性を考慮しただけでは、最近の資源量の不確実性しか考慮できず、それより過去は正しいと仮定されている年齢別漁獲尾数データによって、決定的に求まっているだけなのです。年齢別漁獲尾数の不確実性を考慮すると、時系列に関係なく資源量推定結果に不確実性があることが明らかになるかと思います。

7. おわりに

以上が、現在開発配布されている、VPA のモデル診断関数です。

VPA のモデル診断結果が芳しくない場合、そこから推定される再生産関係、管理基準値の推定、更には将来予測などに、大きな影響を及ぼします。

VPA 関数が無事に動くと一安心してしまう所ではありますが、併せてモデル診断も行って結果の妥当性を評価してください。