

SR-03(2022)



frasyrを用いた 再生産関係の推定: 診断紹介編①

再生産関係モデルにおける推定パラメータの診断

水産研究



ANNIVERSARY



動画製作者 漁業情報解析部 資源解析グループ 福井 眞

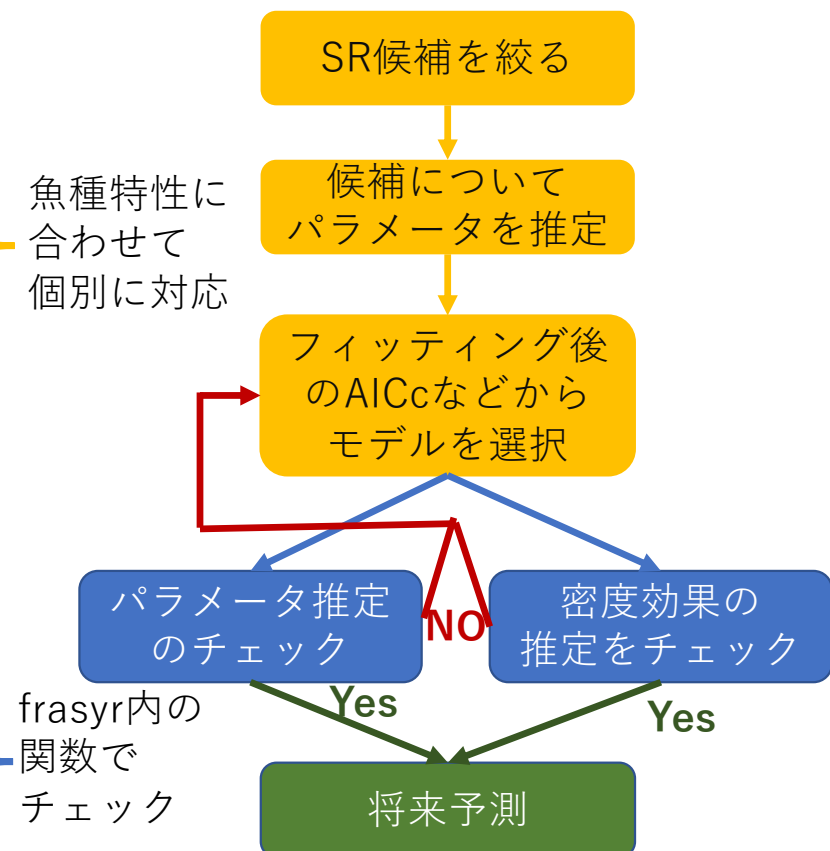
shinfukui@affrc.go.jp

fukui_shin87@fra.go.jp

SR-03(2022) frasyrを用いた再生産関係の推定；診断紹介編①

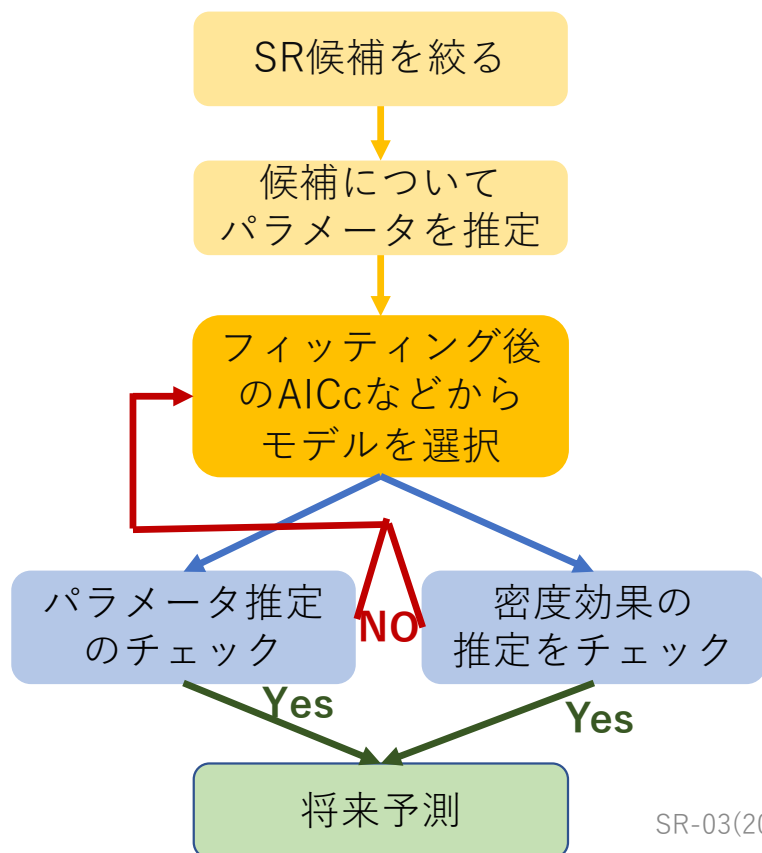
SR関係推定手順と念頭におくこと

- 先行研究などから再生産関係についての生物学的特性が既知か？
→ BH/RI/HSのうち、積極的に採択・不採択となるものは？
- 過去の加入量のトレンドで著しくパターンが変わっていないか？
→ レジームシフトの仮定が必要？
- 再生産曲線や自己相関の有無などの組み合わせで複数の候補となるSRについてパラメータの推定ののちAIC(c)で比較、モデルを選択
 - 選択したモデルの診断、推定結果は安定？
 - 推定した再生産曲線は密度効果が表れている？



診断するSRモデルを絞る

- 再生産曲線や自己相関の有無などの組み合わせで複数の候補となるSRについてパラメータの推定ののちAIC(c)・BICで比較



SR-03(2022) fra

```
SR-02.R x SR-03.R x
Source on Save Run
5 # AIC,AICcやBICを比較してフィットの良いモデルは何かを探る ----
6 SRarglist <- expand.grid(SR=c("HS","BH","RI"),AR=c(0,1),out.AR=c(FALSE,TRUE))
7 SRarglist <- SRarglist[-which(SRarglist$AR==0 & SRarglist$out.AR==TRUE)]
8 resfitSRlist<-list()
9 AIC <- AICc <- BIC <- c()
10 for(i in 1:nrow(SRarglist)){
11   resfitSRlist[[i]]<-fit.SR(SRdata_ex,SR=SRarglist$SR[i],AR=SRarglist$AR[i],out.AR=SRarglist$out.AR[i])
12
13   AIC <- c(AIC,resfitSRlist[[i]]$AIC)
14   AICc <- c(AICc,resfitSRlist[[i]]$AICc)
15   BIC <- c(BIC,resfitSRlist[[i]]$BIC)
16 }
17 SRlist <- cbind(SRarglist,AIC,AICc,BIC)

17:7 # AIC,AICcやBICを比較してフィットの良いモデルは何かを探る

Console Terminal x Background Jobs x
R 4.1.2 ~ /FishStockManageTraining/
> SRlist <- cbind(SRarglist,AIC,AICc,BIC)
> SRlist
  SR AR out.AR      AIC      AICc      BIC
1 HS  0  FALSE  9.870773 10.79385 14.07436
2 BH  0  FALSE 10.267899 11.19098 14.47149
```

SRモデルの診断内容

モデル診断とは？→VPA-04(2020)を参照

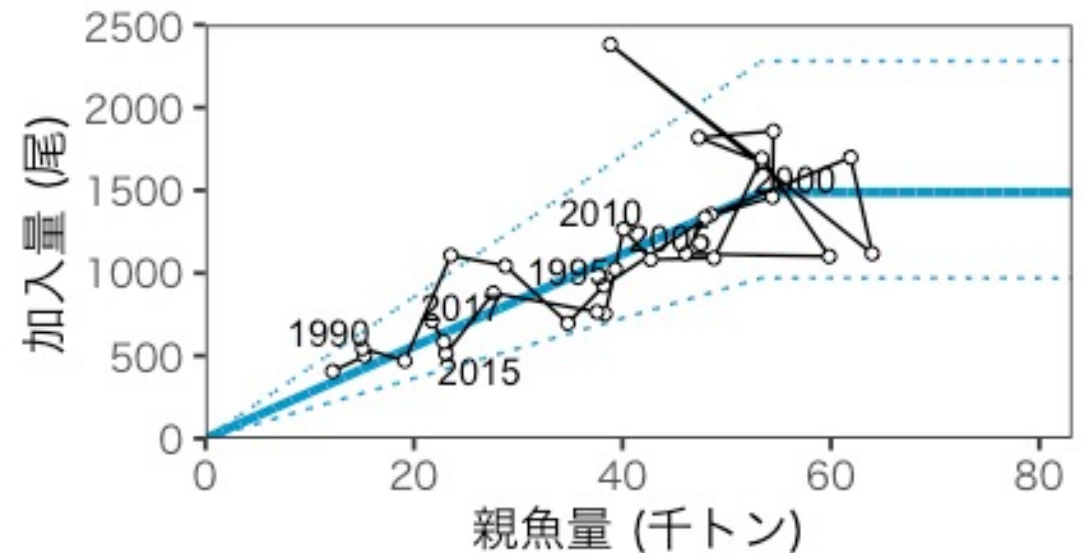
- 果たしてうまく推定できているのか？
 - 推定結果は安定？
 - 残差の分布に想定外の偏りはない？

- 密度効果は表現できているか？
 - パラメータ間相関は？
 - steepnessは？

- plot_SR関数のオプション
plot_CIで予測区間
(デフォルトで90%) を表示

- ここで紹介する手順はfrasyrのwikiを参照

<https://github.com/ichimomo/frasyr/wiki/Diagnostics-for-Stock-Recruitment-Relationships>
(wikiはパッケージ最新状態を反映しているわけではないことに注意)



関数形: HS, 自己相関: 0, 最適化法L1, AICc: 14.08

SR関係推定結果を吟味する関数

- 再生産関係推定のモデル診断の関数
 - check.SRfit：推定値の収束の有無や最適解に達しているか ※③以外
 - check.SRdist：残差の分布
 - prof.likSR：プロファイル尤度
 - [calc.residAR：残差の自己相関]
 - [autocor.plot：残差の自己相関のプロット]
 - boot.SR：ブートストラップ解析
 - jackknife.SR：ジャックナイフ解析
- 再生産関係の密度効果を調べる関数
 - check.SRfit③：推定値が密度効果が現れる範囲にあるか
 - corSR：パラメータ間相関

check.SRfit：推定値の収束の有無や大域最適か？

パラメータの最尤推定がうまくできているか？

• check.SRfitでチェックされる内容：

- ① 収束しているか (convergence)
- ② ヘッセ行列 (Hessian) の対角成分が正定値 (positive definite) になっているか
- ③ 境界条件 (HSの場合SSBの両端、それ以外の場合直線関係や極端に強い密度依存関係) に達しているか (←密度効果のチェックで後述)
- ④ 対数尤度が大域解 (global optimum) に達しているか
- ⑤ 大域的最適解をもつパラメータがひとつに決まるかどうか (④で大域解が出られている場合のみ)

check.SRfit：推定値の収束の有無や大域最適か？

- check.SRfitでチェックされる内容：

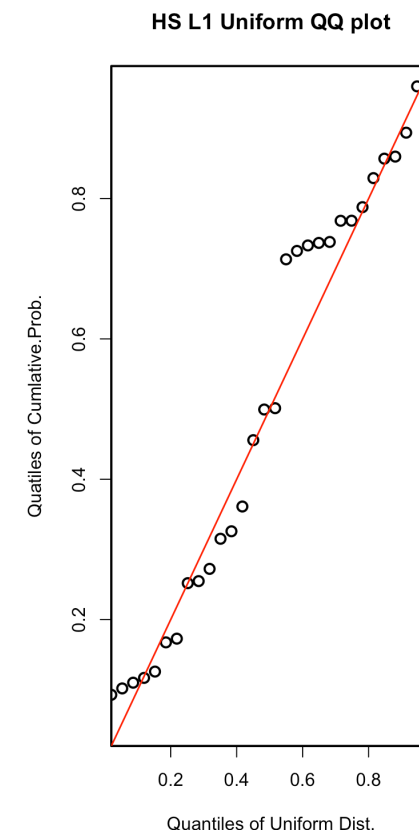
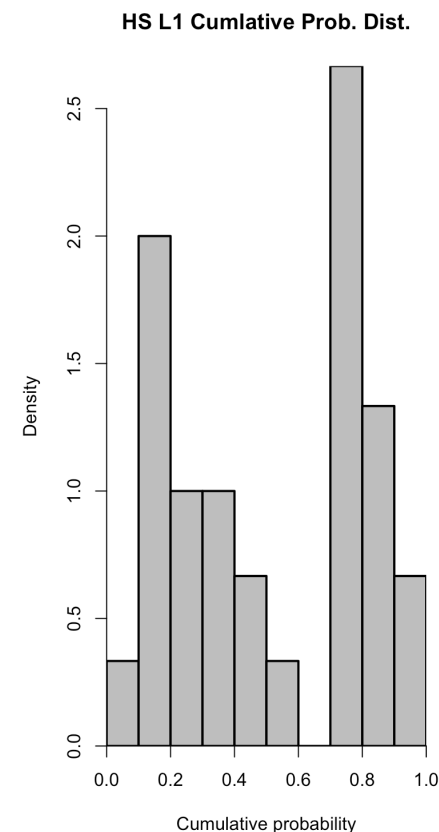
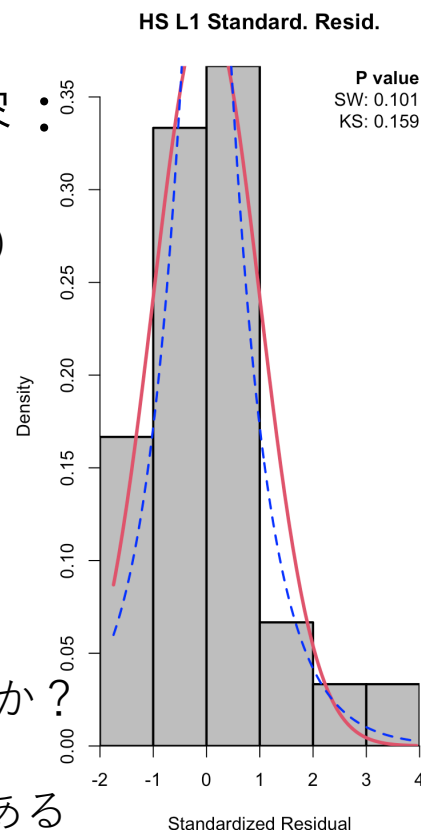
- ① 収束しているか (convergence)
- ② ヘッセ行列 (Hessian) の対角成分が正定値 (positive definite) になっているか
- ③ 境界条件 (HSの場合SSBの両端、それ以外の場合直線関係や極端に強い密度依存関係) に達しているか (←密度効果のチェックで後述)
- ④ 対数尤度が大域解 (global optimum) に達しているか
- ⑤ 大域的最適解をもつパラメータがひとつに決まるかどうか (④で大域解が出られている場合のみ)

- SR="HS", method="L1"では⑤で引っかかることが多い。
 - 複数の推定値の中央値が戻り値オブジェクトoptimumに格納されるのでそれを使う

```
SR-03.R x SR-02.R x
Source on Save Run
22 # 収束判定 ----
23 checkL1HS<-check.SRfit(resL1HS)
24 # L1HSは解が複数でる
25
24:1 # 収束判定
R Script
Console Terminal Background Jobs
R 4.1.2 ~/FishStockManageTraining/
+ regime.name = c("A","B"))
Joining, by = c("Regime", "SSB", "R", "Category")
> # 収束判定 ----
> checkL1HS<-check.SRfit(resL1HS)
1. 収束しています (OK)
2. Hessian行列の対角成分が正定値になっています (OK)
3. どの推定パラメータも壁(boundaries)にあたっていないのでOKです (OK)
4. パラメータが大域解に達しているのでOKです (OK)
5. 同じ最大尤度(1e-06よりも小さい違い)を持つ複数のパラメータが見つかりました (L1かつHSでよく見られます)
   ほとんど同じ尤度を持つパラメータの範囲 ( 100 回試行のうち 94 回分),
           a          b          sd
Min.    0.02785966 53370.82 0.2598904
1st Qu. 0.02786579 53379.12 0.2599183
Median  0.02787042 53385.41 0.2599298
Mean    0.02786997 53386.91 0.2599315
3rd Qu. 0.02787367 53395.98 0.2599452
Max.    0.02788033 53403.32 0.2599733
0.0279 53380.54 0.2599 0 をoptimumに出力します(そのときの初期値は -4.30234673762356-2.39519806
6362 です)
```

check.SRdist : 残差の分布

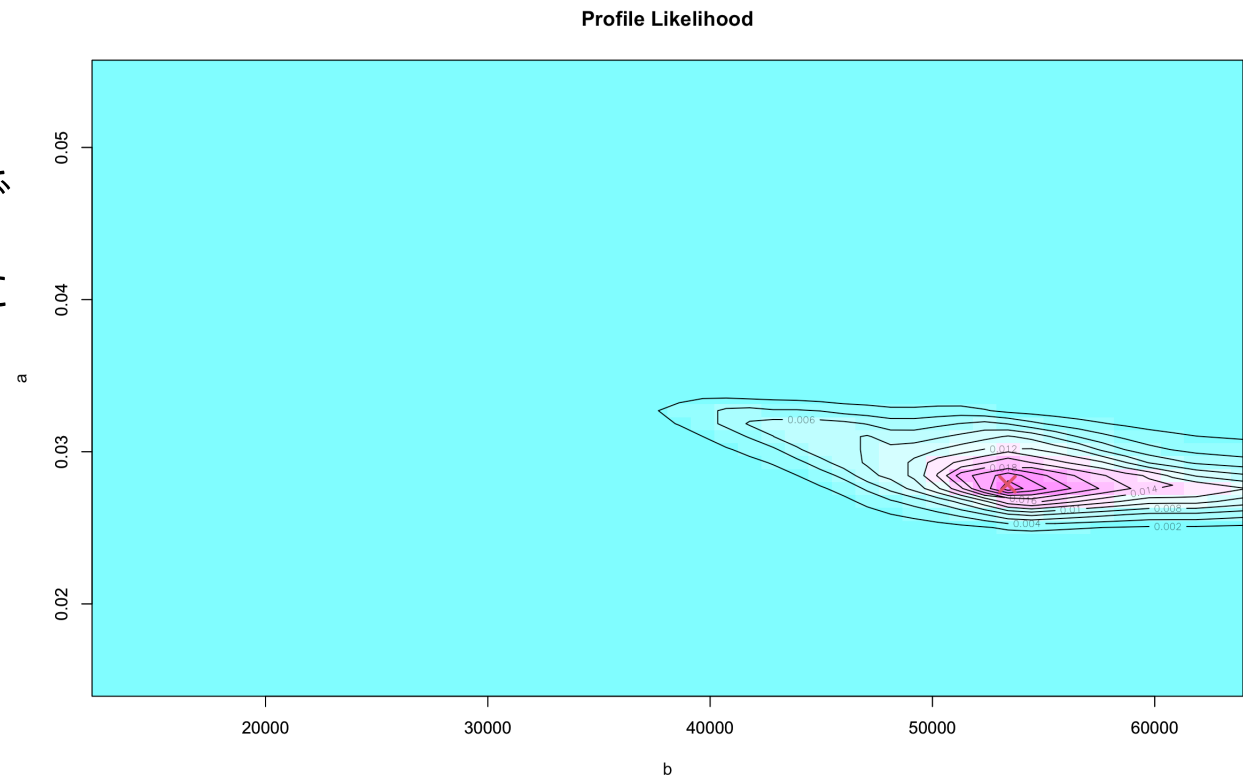
- check.SRdistで返る内容 :
 - 標準化残差の分布
 - L1:ラプラス分布 (青点線)
 - L2:正規分布 (赤線)
 - になっているか?
 - Shapiro-Wilk検定
 - Kolmogorov-Smirnov検定
 - 残差の累積確率密度分布
 - 一様になっているか?
 - QQプロット (一様分布)
 - 1:1の直線上にのっているか?
 - 端が直線に乗りやすく
 - 中心が外れやすい傾向がある



prof.likSR：プロフィール尤度

- prof.likSRで示されるもの：
推定されたパラメータ近辺で
少しパラメータを変更させた
時の尤度の差

局所解に落ちているか、
 a と b の相関が強すぎないか、
などの様子を可視化



calc.residAR : 残差の自己相関

- 残差の自己相関
 - AR=1, outAR=Tとしたときの自己相関係数を計算
- 出力される\$parsはfit.SRでAR=1,outAR=Tとしたものと同値
- AICなども出力

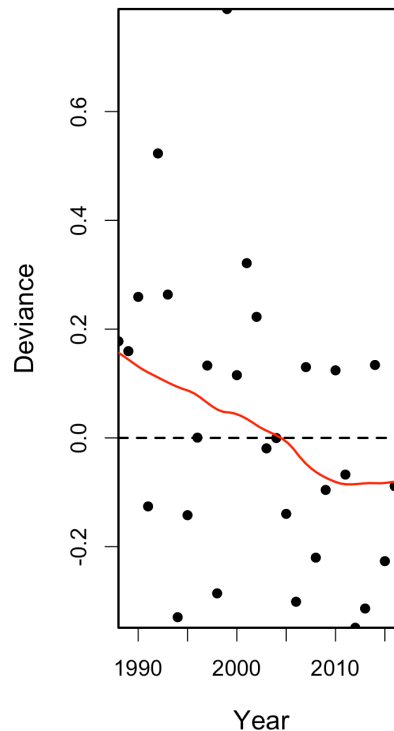
```
R 4.1.2 ~ /FishStockManageTraining/ ➔  
$loglik  
      AR(0)    AR(1)  
-2.149188 -2.142661  
  
$k  
AR(0) AR(1)  
    1    2  
  
$AIC  
      AR(0)    AR(1)  
6.298376 8.285321  
  
$AICc
```

```
SR-02.R x SR-03.R x  
Source on Save 🔍 🚀 📄  
Run ⏏ ⬆ ⬇  
36 prof.likSR(resL1HS)  
37 prof.likSR(resL1HSchecked)  
38 # 図を保存する場合  
39 prof.likSR(resL1HS, output = TRUE, filename = "ResidDistCheck_L1HS")  
40  
41 prof.likSR(resL2RIARout1)  
42 prof.likSR(resL2BHARout0)  
43  
44 # 残差の自己相関のチェック ----  
45 outer1HS = calc.residAR(resL1HS, output = TRUE, filename = "L1HSresidARouter")  
46 outer1HS  
47  
48 # 残差の自己相関をプロット  
49 autocor.plot(resL1HS) #デフォルトはuse.resid = 1  
50 autocor.plot(resL1HS, output = TRUE, filename = "L1HSdevianceAR")  
48:1 # 残差の自己相関のチェック  
Console Terminal Background Jobs  
R 4.1.2 ~ /FishStockManageTraining/ ➔  
> outer1HS  
$pars  
      a      b      sd      rho  
1 0.02786598 53372.54 0.259883 0.02080394  
  
$arima0  
  
Call:  
arima(x = deviance, order = c(0, 0, 0), include.mean = FALSE, method = "ML")  
  
sigma^2 estimated as 0.06757: log likelihood = -2.15, aic = 6.3  
  
$arima1
```

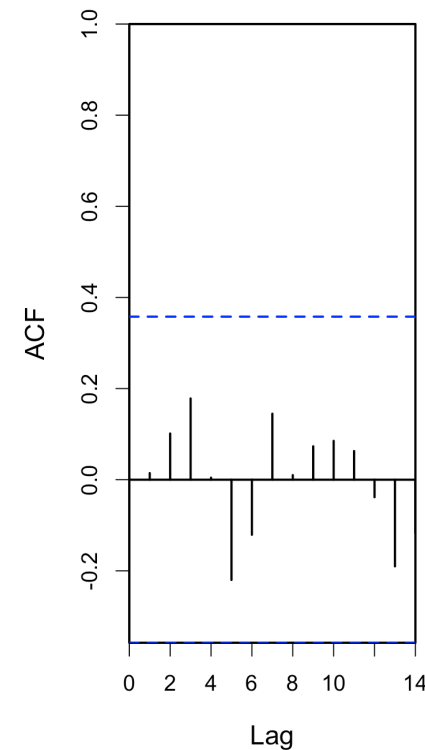
autocor.plot : 残差の自己相関のプロット

- 再生産関係との残差の時系列
 - 赤い線は平滑化された曲線
- ラグを1,2,3...年と増やした場合の自己相関係数
 - 青い線は95%信頼区間
- Ljung-Box検定におけるp値
 - 帰無仮説；ラグが1~mまでの全ての自己相関が0である
 - 対立仮説；ラグ1からmまでの自己相関のうち、少なくとも一つが0でない
 - 赤点で表示される箇所が有意
- output=Tで図を保存
 - autocor.plot(resL1HS,
output = TRUE,
filename = "devianceAR")

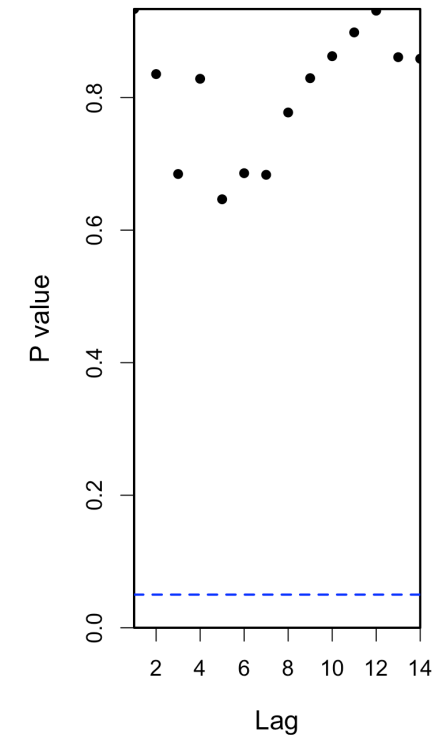
Time series of deviance to SR



Autocorrelation (rho vs. lag)



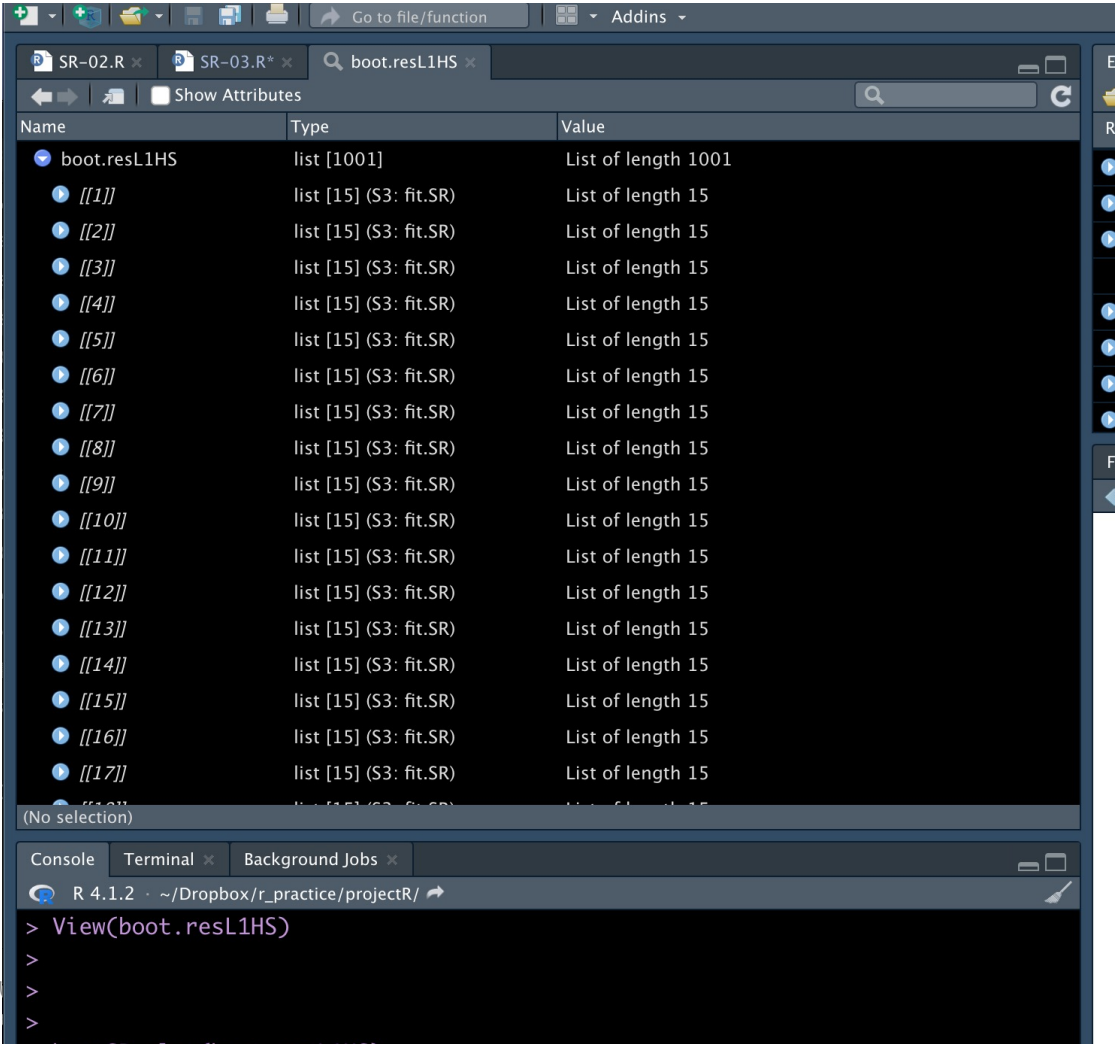
Ljung-Box test



boot.SR : ブートストラップ解析

- SR関係パラメータの推定誤差から生じる推定区間を可視化
- 3種類のブートストラップを実装
 - 残差のパラメトリックブートストラップ(オプションmethod="p")
 - 残差のノンパラメトリックブートストラップ(method="n")
 - データブートストラップ(method="d")
- 回数nは500回や1000回
- 自己相関を推定した場合は ρ も
- fit.SRの引数にbio_parを入れてステイプネスを計算した場合、SB0、R0、B0、hの図も出力
- いずれも図示するにはbootSR.plotを使用

SR-03(2022) frasyrを用

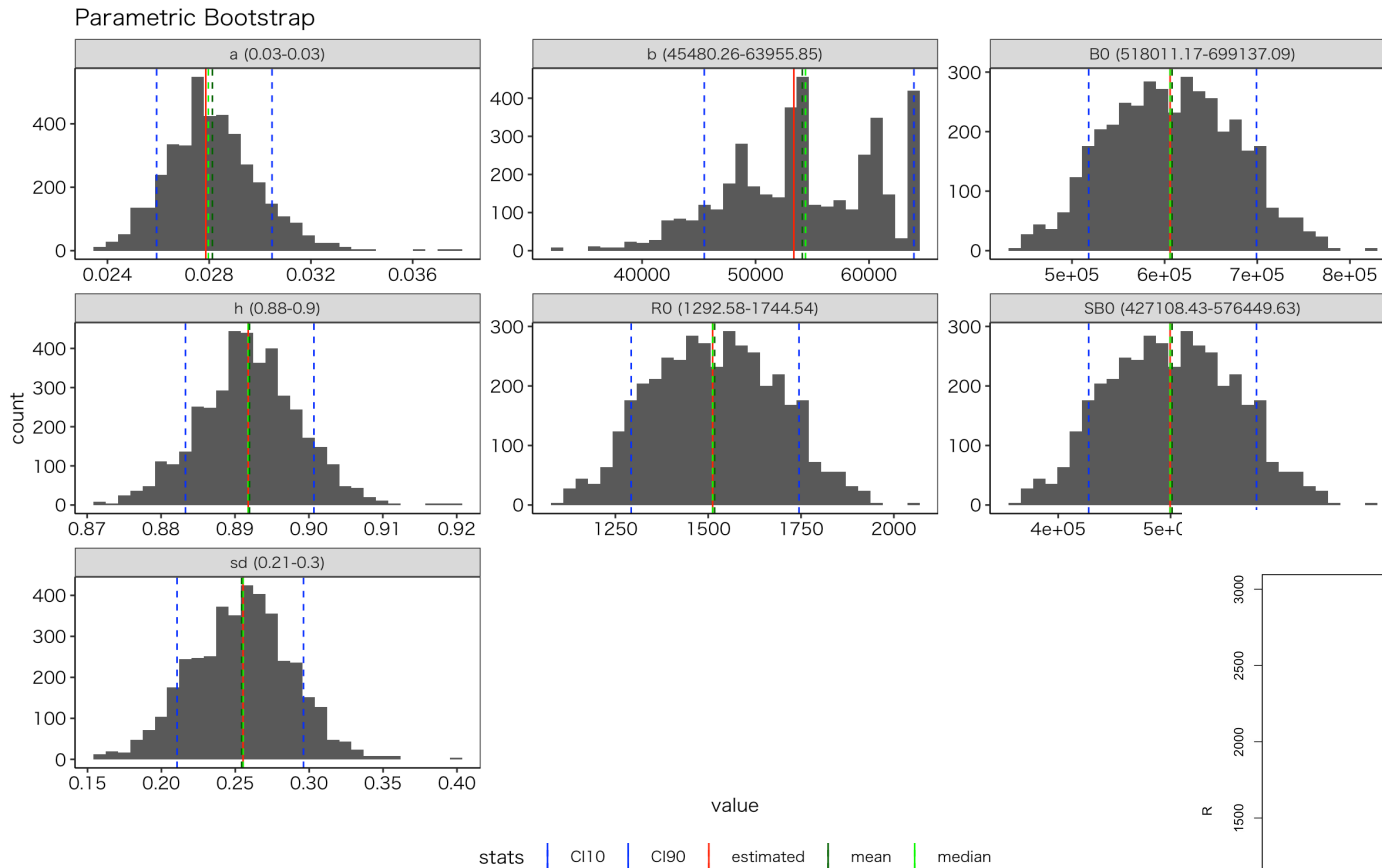


The screenshot displays the RStudio environment pane and console. The environment pane shows a list of objects under the name 'boot.resL1HS', including a main list of length 1001 and 17 sub-lists of length 15 each, all of type 'list [15] (S3: fit.SR)'. The console shows the command `View(boot.resL1HS)` being executed in an R 4.1.2 session.

| Name | Type | Value |
|--------------|------------------------|---------------------|
| boot.resL1HS | list [1001] | List of length 1001 |
| [[1]] | list [15] (S3: fit.SR) | List of length 15 |
| [[2]] | list [15] (S3: fit.SR) | List of length 15 |
| [[3]] | list [15] (S3: fit.SR) | List of length 15 |
| [[4]] | list [15] (S3: fit.SR) | List of length 15 |
| [[5]] | list [15] (S3: fit.SR) | List of length 15 |
| [[6]] | list [15] (S3: fit.SR) | List of length 15 |
| [[7]] | list [15] (S3: fit.SR) | List of length 15 |
| [[8]] | list [15] (S3: fit.SR) | List of length 15 |
| [[9]] | list [15] (S3: fit.SR) | List of length 15 |
| [[10]] | list [15] (S3: fit.SR) | List of length 15 |
| [[11]] | list [15] (S3: fit.SR) | List of length 15 |
| [[12]] | list [15] (S3: fit.SR) | List of length 15 |
| [[13]] | list [15] (S3: fit.SR) | List of length 15 |
| [[14]] | list [15] (S3: fit.SR) | List of length 15 |
| [[15]] | list [15] (S3: fit.SR) | List of length 15 |
| [[16]] | list [15] (S3: fit.SR) | List of length 15 |
| [[17]] | list [15] (S3: fit.SR) | List of length 15 |

```
R 4.1.2 ~ /Dropbox/r_practice/projectR/
> View(boot.resL1HS)
>
>
>
```

bootSR.plot

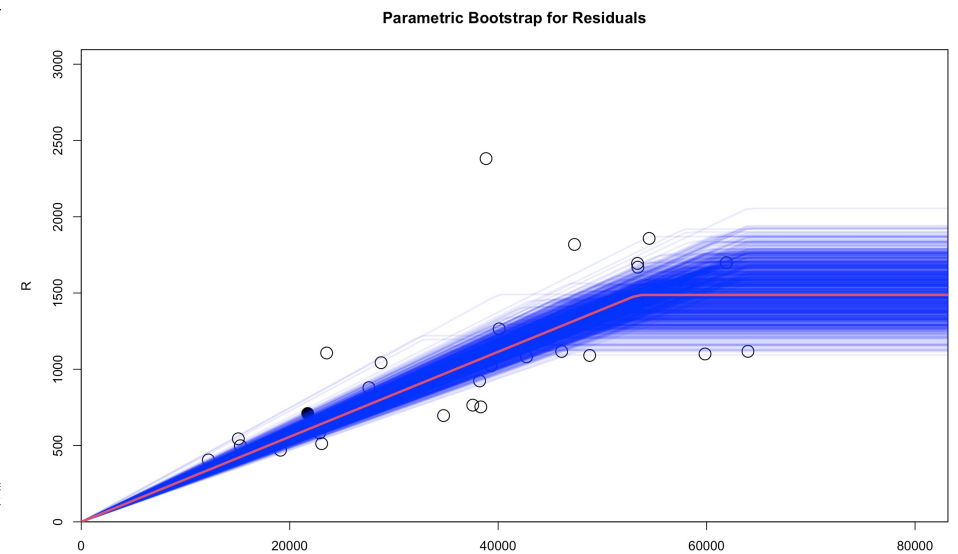


SR-03(2022) frasyrを用いた再生産関

- RstudioのPlots画面には2つ目のブートストラップ結果のSRプロットが表示されるが、←でヒストグラムに戻る

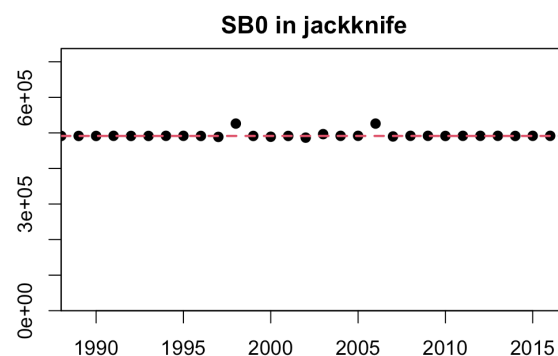
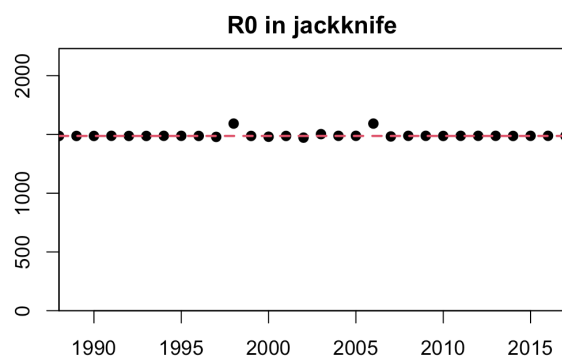
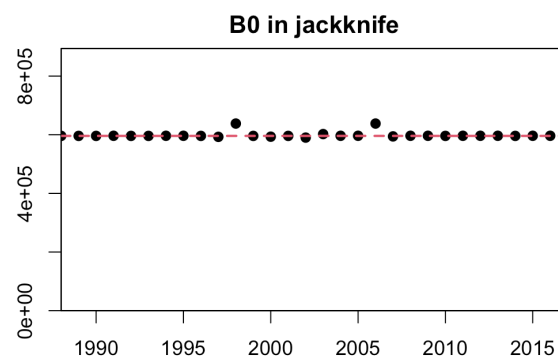
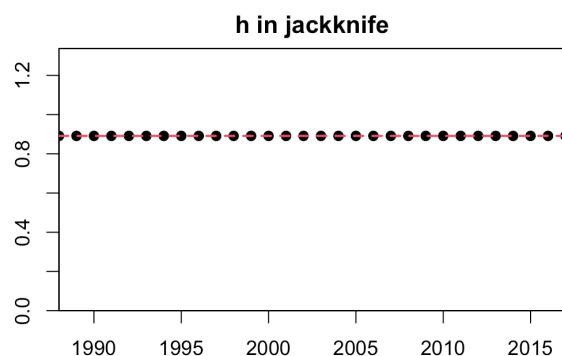
- オプションoutput=Tで図を保存

```
bootSR.plot(boot.resL1HS,  
            output=T,  
            filename="bootresL1HS")
```

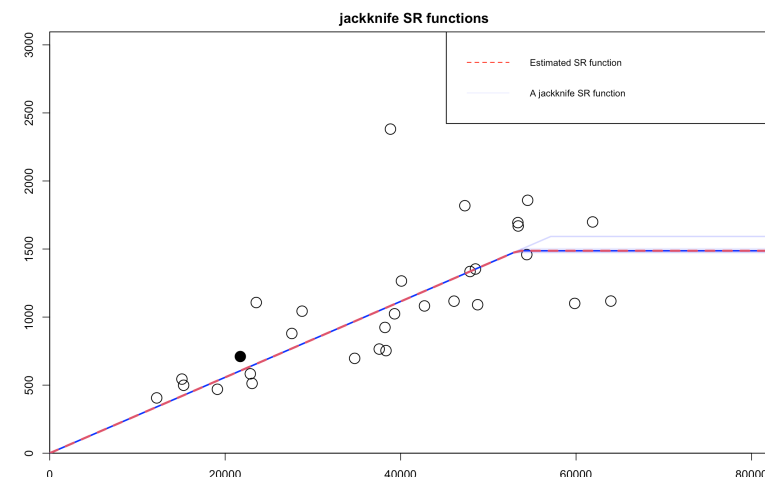


jackknife.SR : ジャックナイフ解析

- データを一点ずつ除いて再推定し、各データが与える影響を評価
- 自己相関ありのジャックナイフが有効なのは同時推定の場合のみ



- オプションoutput=Tで図を保存
jackL1HS = jackknife.SR(resL1HS,
output = TRUE,
filename="jackresL1HS")



SRモデルのパラメータ推定診断



- frasyrを使ったSR関係推定のモデル診断について、
推定パラメータに関する診断内容の紹介は以上です

お疲れ様でした！

