2019年10月9日
資源管理研修会（上級）

３．個体群動態モデルを用いた資源評価

西嶋　翔太
（中央水産研究所 資源研究センター）

本日の内容

1. 一般化線形モデルを使ったランダム効果の説明・RとTMBによる最尤推定
2. TMBを使った余剰生産（surplus production）モデルのパラメータ推定
3. Delay-difference modelを用いた、生物特性とMSYの関係
4. 年齢別個体群動態モデル（ridge VPA & SAM）
5. SAMの発展

本日の内容

1. 一般化線形モデルを使ったランダム効果の説明・RとTMBによる最尤推定
2. TMBを使った余剰生産（surplus production）モデルのパラメータ推定
3. Delay-difference modelを用いた、生物特性とMSYの関係
4. 年齢別個体群動態モデル（ridge VPA & SAM）
5. SAMの発展

ランダム効果

予測できない（確率的な）変動要因

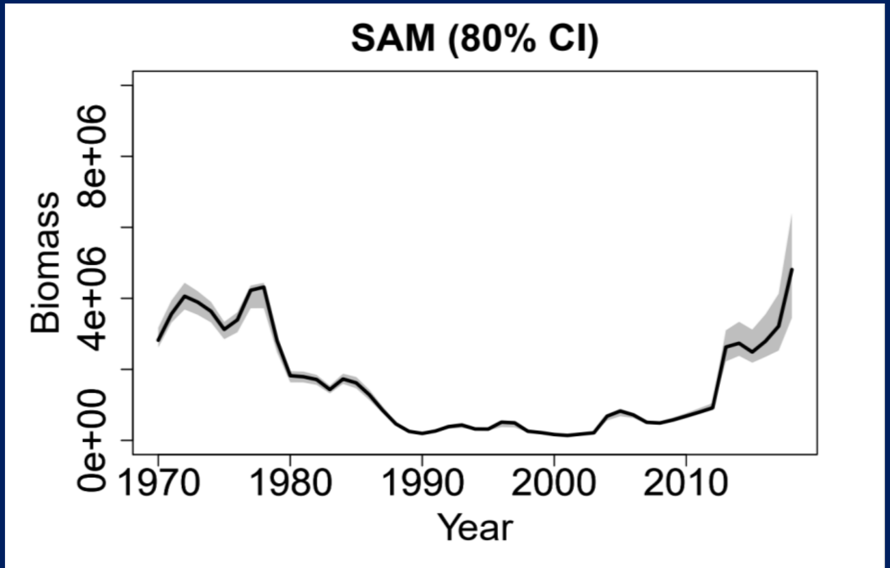
例：

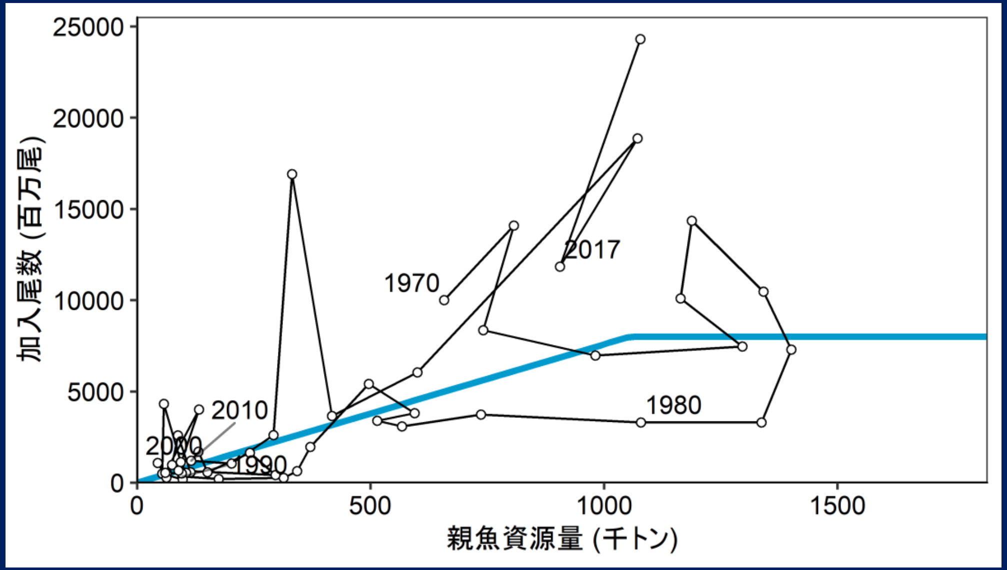
- あるサイトにおける各種の個体数の平均は10匹だった
- 平均10のポワソン分布を発生したところ、実際のデータの分散は10より明らかに大きい（ポワソン分布は平均＝分散）
- 種の違いによって、期待よりも大きな変動（過分散）が生じたのだろう

⇒ よく分からない（確率的な）変動を<u>ランダム効果</u>で扱おう！

なぜ資源評価モデルでランダム効果？

- 水産資源の変動は予測が難しく、確率的な変動を示す（過程誤差）





- これまでベイズ推定＆MCMC（Beyesian Surplus Production Modelなど）が使用されてきたが、TMB（ADMBの進展版）が開発され、複雑で大量のランダム効果の高速最尤推定が可能に

一般化線形混合モデルのシミュレーション

「ポワソン分布＋正規分布」のデータ生成

```
# data simulation
N = 100
beta = c(1,0.5,1) # parameter setting
X = seq(-1,1,length=N) # explanatory variable
set.seed(12345)
epsilon = rnorm(n=N,mean=0,sd=beta[3]) # random variable
r = beta[1] + beta[2]*X + epsilon # linear predictor
Y = rpois(n=N, lambda = exp(r)) # response variable
```

固定効果（切片・傾き・SD）

説明変数（観測可能）

ランダム変数（観測不能）

線形予測子（固定効果・説明変数・ランダム効果）

目的変数（観測可能）

一般化線形モデルの適用

ポワソン分布（過分散を無視）

```
> poisson_model = glm(Y~X, family=poisson(link="log"))
> summary(poisson_model)

Call:
glm(formula = Y ~ X, family = poisson(link = "log"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.2752  -2.2331  -1.2467   0.4113   8.7905

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.80810    0.04174  43.321   <2e-16 ***
X            0.61597    0.06985   8.818   <2e-16 ***
```

推定値が結構ちがう
真値: beta = (1,0.5)

一般化線形混合モデルの適用

ポワソン分布＋正規分布（過分散を考慮）

```
> library(lme4)
> poisson_model_RE = glmer(Y~X + (1|id), family=poisson(link="log"))
> summary(poisson_model_RE)
[途中省略]
Random effects:
 Groups Name        Variance Std.Dev.
 id     (Intercept) 1.206    1.098
Number of obs: 100, groups:  id, 100

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.2323     0.1309   9.413  < 2e-16 ***
X             0.5995     0.2161   2.774  0.00554 **
```

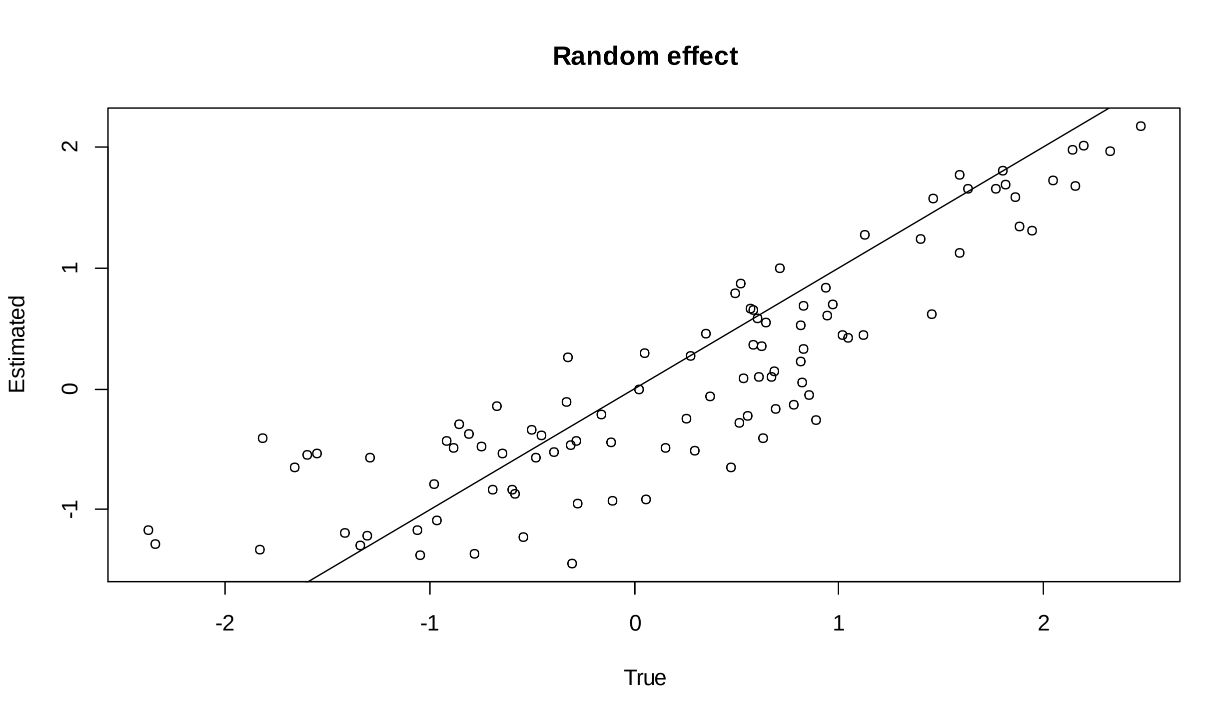
パッケージの読み込み

切片にランダム効果

より真値に近い

AICも小さくなる ⇒ 予測性能の向上

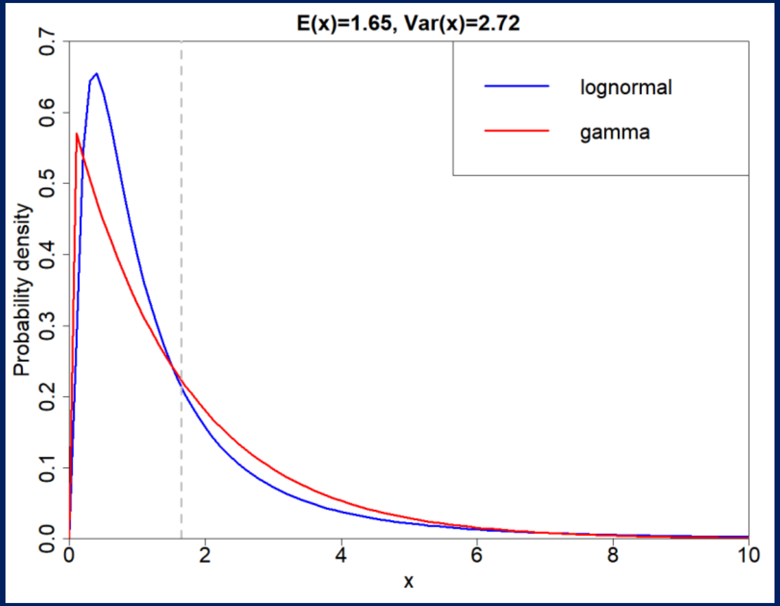
```
> AIC(poisson_model,poisson_model_RE)
                 df       AIC
poisson_model     2 1067.7451
poisson_model_RE  3  581.9344
```

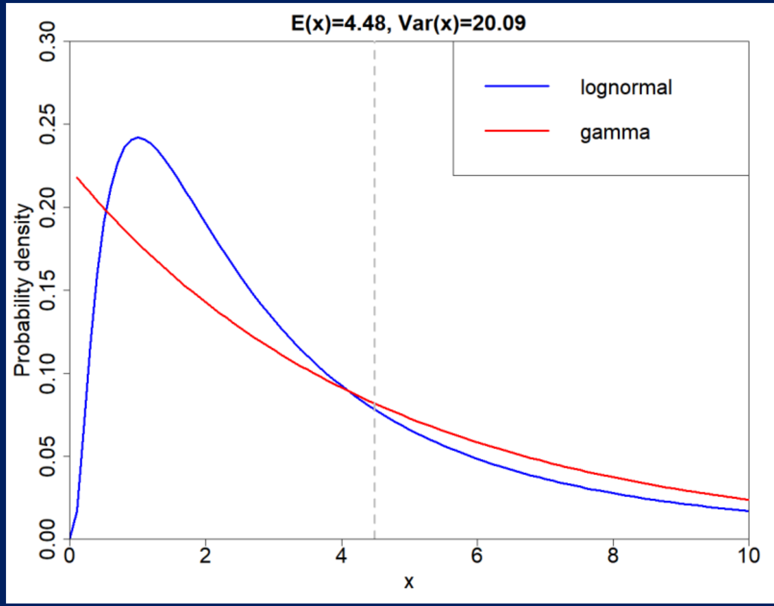


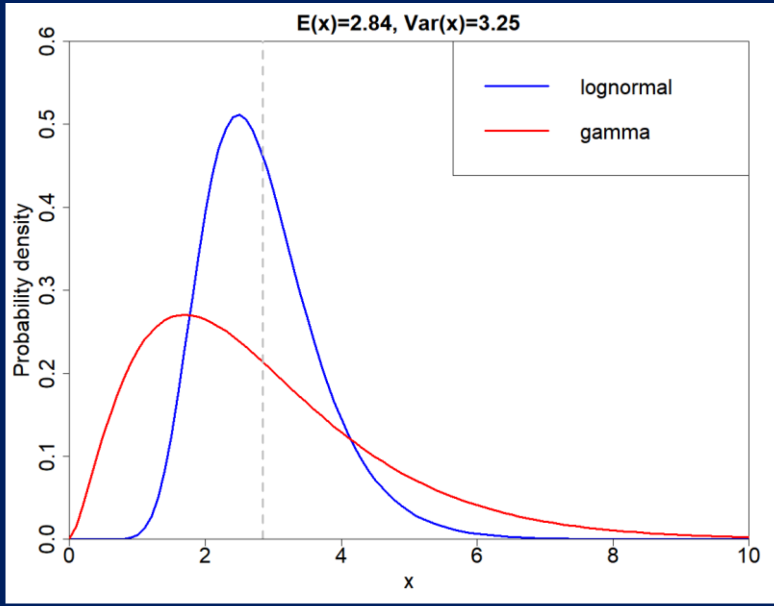
ちなみに…負の二項分布

- ポワソン分布の平均がガンマ分布である確率分布
- つまり、一種の混合分布（モデル）

（ガンマ分布の方が対数正規分布よりも極端な値を取りやすい）







自分で最尤推定のプログラミングを書く

ポワソン分布

```
> poisson_obj = function(beta) {  固定効果のパラメータの関数
+   r = beta[1] + beta[2]*X
+   negative_loglik = -sum(dpois(Y, lambda = exp(r), log = TRUE))
+   return(negative_loglik)  ポワソン分布の尤度を計算する関数
+ }  負の対数尤度が目的関数（最小化すべき関数）
>
>
> poisson_opt = nlminb(beta[1:2],poisson_obj)  最適化関数（optimなども同様）
> poisson_opt$par #parameter
[1] 1.8081027 0.6159796
```

自分で最尤推定のプログラミングを書く

ポワソン分布＋正規分布

```
> poissonRE_obj = function(beta) {
+   negative_loglik = 0
+   for (i in 1:length(X)) {
+     likelihood = function(epsilon) {
+       r = beta[1] + beta[2]*X[i] + epsilon
+       dpois(Y[i],exp(r))*dnorm(epsilon,0,beta[3])
+     }
+     res = integrate(likelihood,-Inf,Inf)
+     negative_loglik = negative_loglik-log(res$value)
+     }
+   return(negative_loglik)
+ }
> poissonRE_opt = nlminb(beta, poissonRE_obj)
> poissonRE_opt$par # parameters
[1] 1.2306966 0.5978468 1.1106555
```

固定効果のパラメータの関数

ランダム効果の関数を定義

ポワソン分布×正規分布

積分によりランダム効果を消去

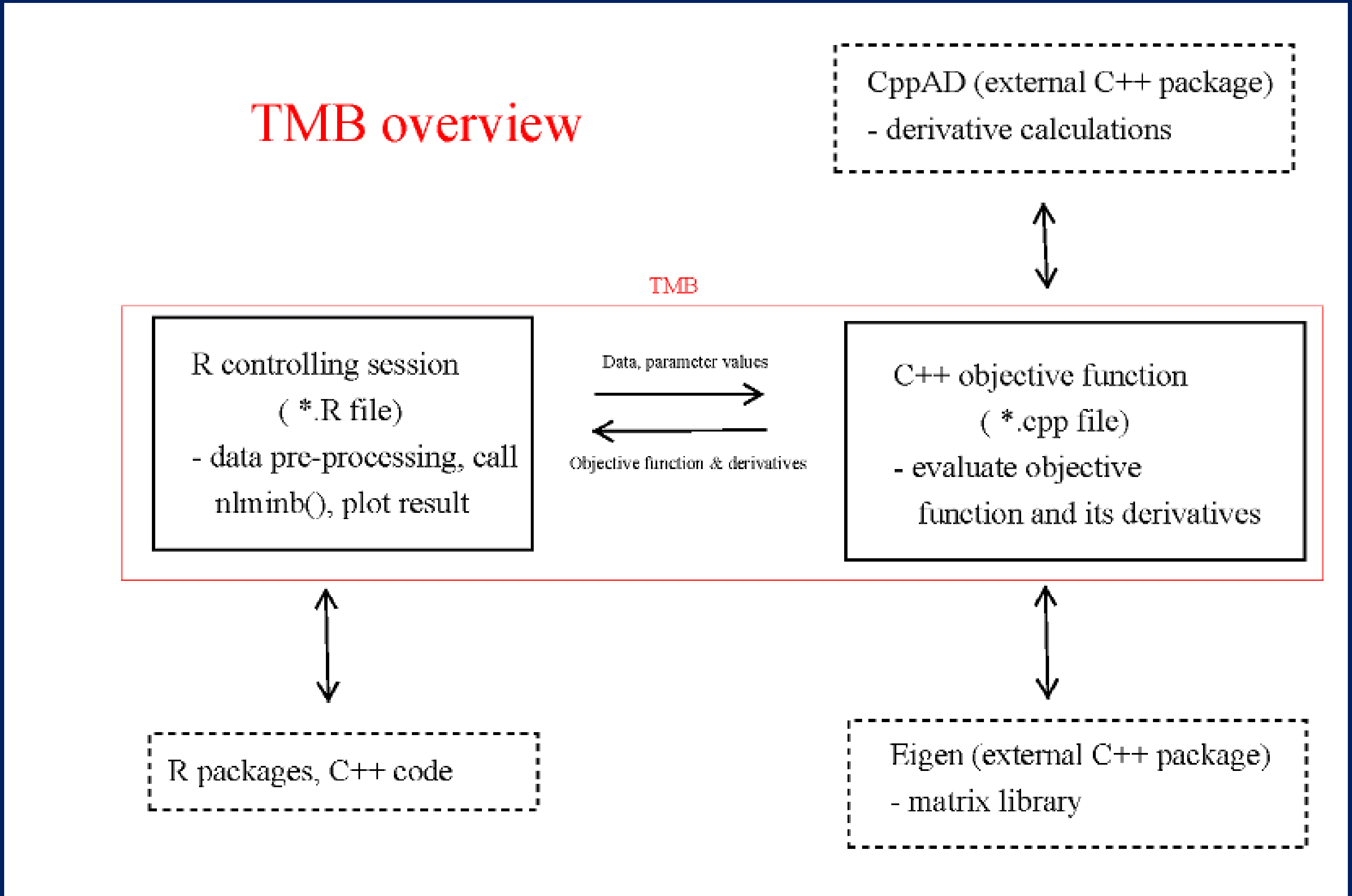
各データの尤度

$$L_i(\boldsymbol{\beta}, \sigma | x_i, y_i) = \int_{-\infty}^{\infty} f(y_i | \boldsymbol{\beta}, \varepsilon_i) \times g(\varepsilon_i | \sigma) d\varepsilon_i$$

ポワソン分布（条件付き確率）

正規分布

Template Model Builder (Kristensen et al. 2016)



- 複雑なランダム効果を含むモデルを高速計算
- 目的関数の<u>自動微分</u>
- <u>ラプラス近似</u>により周辺尤度を計算
- RでC++ファイルを読み込んで使用
- 多くのRの関数が使用可能
- 他の外部パッケージを使える（CppAD, Eigen）
- デルタ法でパラメータのSDを計算

https://kaskr.github.io/adcomp/_book/Structure-TMB.html

glmm_poisson.cppファイル①

```
// DATA //
DATA_VECTOR(Y);
DATA_VECTOR(X);

// PARAMETER //
PARAMETER_VECTOR(beta); // fixed effect
PARAMETER_VECTOR(epsilon); // random effect
```

- インプットするデータ（DATA_...）とパラメータ（PARAMETER_）を定義
- 構造を明記する必要がある
  - SCALAR, VECTOR, MATRIX, ARRAY, INTEGER, IVECTOR, IARRAY, ...
- 文末にセミコロン；が必要

glmm_poisson.cppファイル②

```
vector<Type> r(X.size());

  Type nll=0;
  for(int i=0;i<X.size();i++){
    r(i) =
beta(CppAD::Integer(Type(0)))+beta(CppAD::Integer(Type(1)))*X(i)+epsilon(i);
    nll -= dpois(Y(i),exp(r(i)),true); // poisson distribution
    nll -= dnorm(epsilon(i),Type(0),beta(CppAD::Integer(Type(2))),true); //
random effect
  }
  return nll;
```

ベクトルを定義

負の対数尤度 negative log-likelihood

0から始める！（Rと違う）

整数であることを明記する必要

尤度関数（dpois, dnorm）はRと同じ

この時点で固定効果・ランダム効果を分ける必要はない

- .size()でベクトルのサイズ（.cols(), .rows(), .dim()などもある）
- 数字そのものよりもType(0)とかとした方が良い
- -=で前の値から右辺を引く（+=, *=, /=などもある）

RでのTMBの利用：一般化線形混合モデル

```
library(TMB)
compile("glmm_poisson.cpp")
dyn.load(dynlib("glmm_poisson"))

data = list(Y=Y,X=X)
params = list(beta=beta,epsilon=rep(0,N))
obj = MakeADFun(data, params, random="epsilon",DLL="glmm_poisson")

opt = nlminb(obj$par, obj$fn, obj$gr)

sdrep = sdreport(obj) #random effect
```

TMBパッケージの読み込み

cppファイルのコンパイル

dllファイルの読み込み

データをリストで与える

パラメータの初期値

自動微分（Automatic Differentiation）の関数

ここでランダム効果を定義

初期値　目的関数　グラディエント（重要！）

標準偏差（分散）の計算

本日の内容

1. 一般化線形モデルを使ったランダム効果の説明・RとTMBによる最尤推定
2. TMBを使った余剰生産（surplus production）モデルのパラメータ推定
3. Delay-difference modelを用いた、生物特性とMSYの関係
4. 年齢別個体群動態モデル（ridge VPA & SAM）
5. SAMの発展

余剰生産モデル（Surplus Production Model）

資源量の変化

$$B_{t+1} = B_t + f(B_t) - C_t$$

余剰生産

=成長+加入+自然死亡

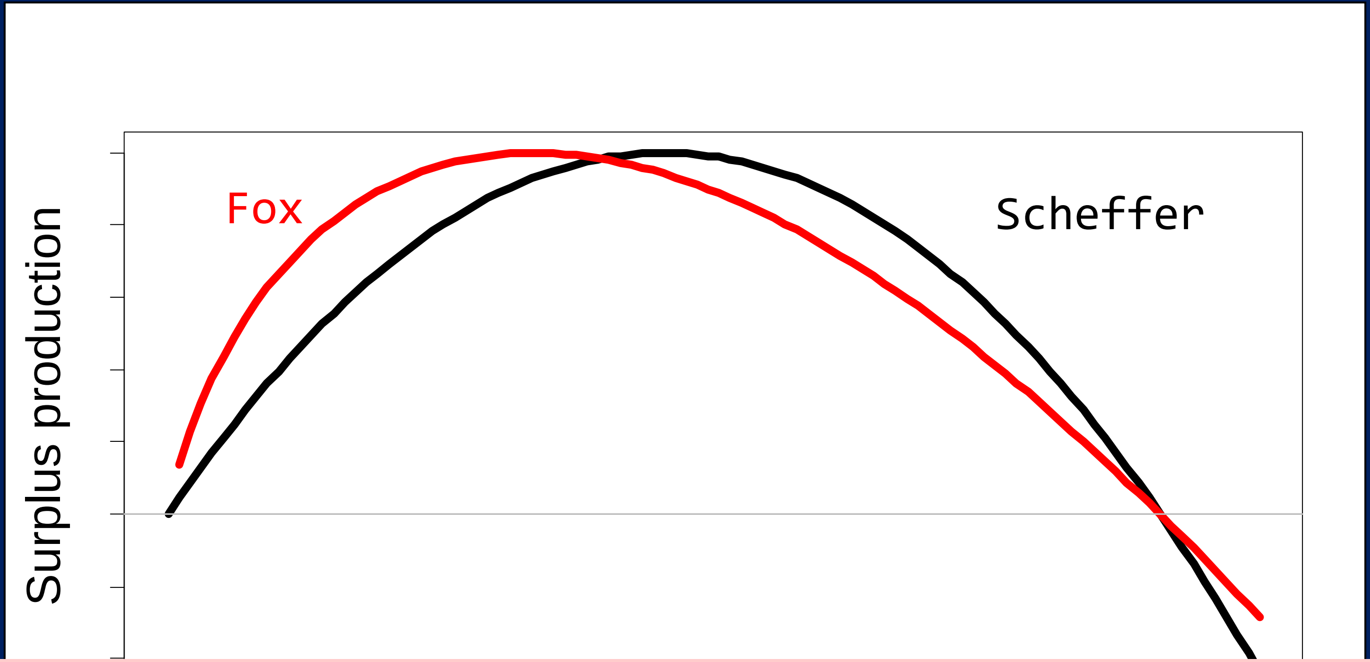
- シンプルな個体群モデル
- 個体群内の構造（年齢・成長・成熟）を無視する
- MSY（最大持続可能漁獲量）の基礎となるモデル
- 2系・3系ルールのMSEに使用されている

Scheffer modelとFox model

Scheffer $\hat{f}(B_t) = rB_t\left(1 - \frac{B_t}{K}\right)$

Fox $\hat{f}(B_t) = rB_t(\log K - \log B_t)$

$Bmsy = K/2$
$Fmsy = r/2$
$MSY = rK/4$



$Bmsy = K/e$
$Fmsy = r$
$MSY = rK/e$

SchefferではBmsyがKの半分だが、Foxでは37%
Foxの方が最初の立ち上がりが急

surplus_production.cppファイル

```
Type nll=0;

  for(int i=1;i<Catch.size();i++){ //Process likelihood
    Type pred_SP = 0;
    if(SP_type==0){ //Scheffer
      pred_SP += r*B(i-1)*(1-B(i-1)/K);
    }else{ // Fox
      pred_SP += r*B(i-1)*(log_K-log_B(i-1));
    }
    nll -= dnorm(log(B(i)-B(i-1)+Catch(i-1)),log(pred_SP),sigma_pro,true);
  }

  // observation likelihood
  nll -= sum(dnorm(log(cpue),log_q+log_B,sigma_obs,true));

  ADREPORT(B);
  return nll;
```

前年からの予測値

過程誤差の尤度

観察誤差の尤度（CPUEの当てはまり）

推定値やSEを知りたい

Derived parameter

Rで実行

```
data = list(Catch=Catch,cpue=cpue,SP_type=0)
params =
list(log_r=log(r),log_K=log(k),log_sigma_pro=log(sigma_pro),
log_q=log(q),log_sigma_obs=log(sigma_obs),log_B=rep(log(0.5*k),nyea
r))

obj = MakeADFun(data, params,random="log_B",
DLL="surplus_production")
opt = nlminb(obj$par, obj$fn, obj$gr)
sdrep = sdreport(obj,bias.correct=TRUE) #random effect
```

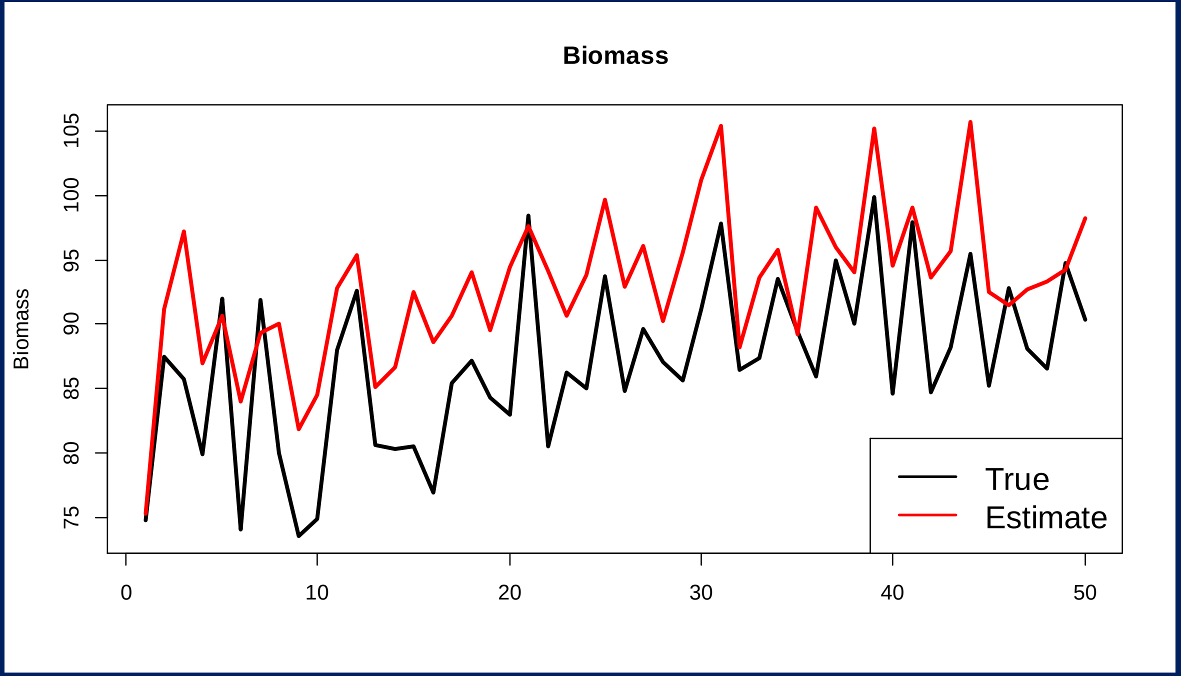
漁獲データとCPUEデータ

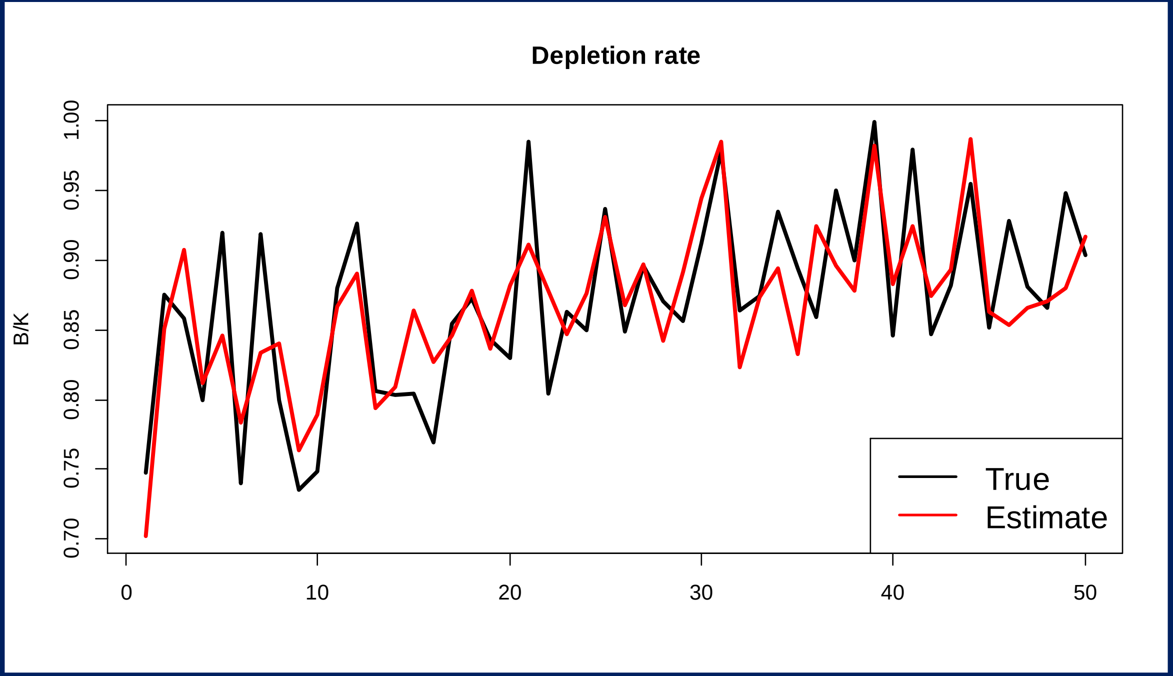
資源量の対数をランダム効果に指定

ランダム効果の平均補正

Thorson, J and Kristensen, K. 2016. Implementing a generic method for bias correction in statistical models using random effects, with spatial and population dynamics examples. Fisheries Research, 175: 66–74.

パラメータ推定結果①





絶対的な資源量推定値ははずれるが、depletion rate (B/K) は大体一致する

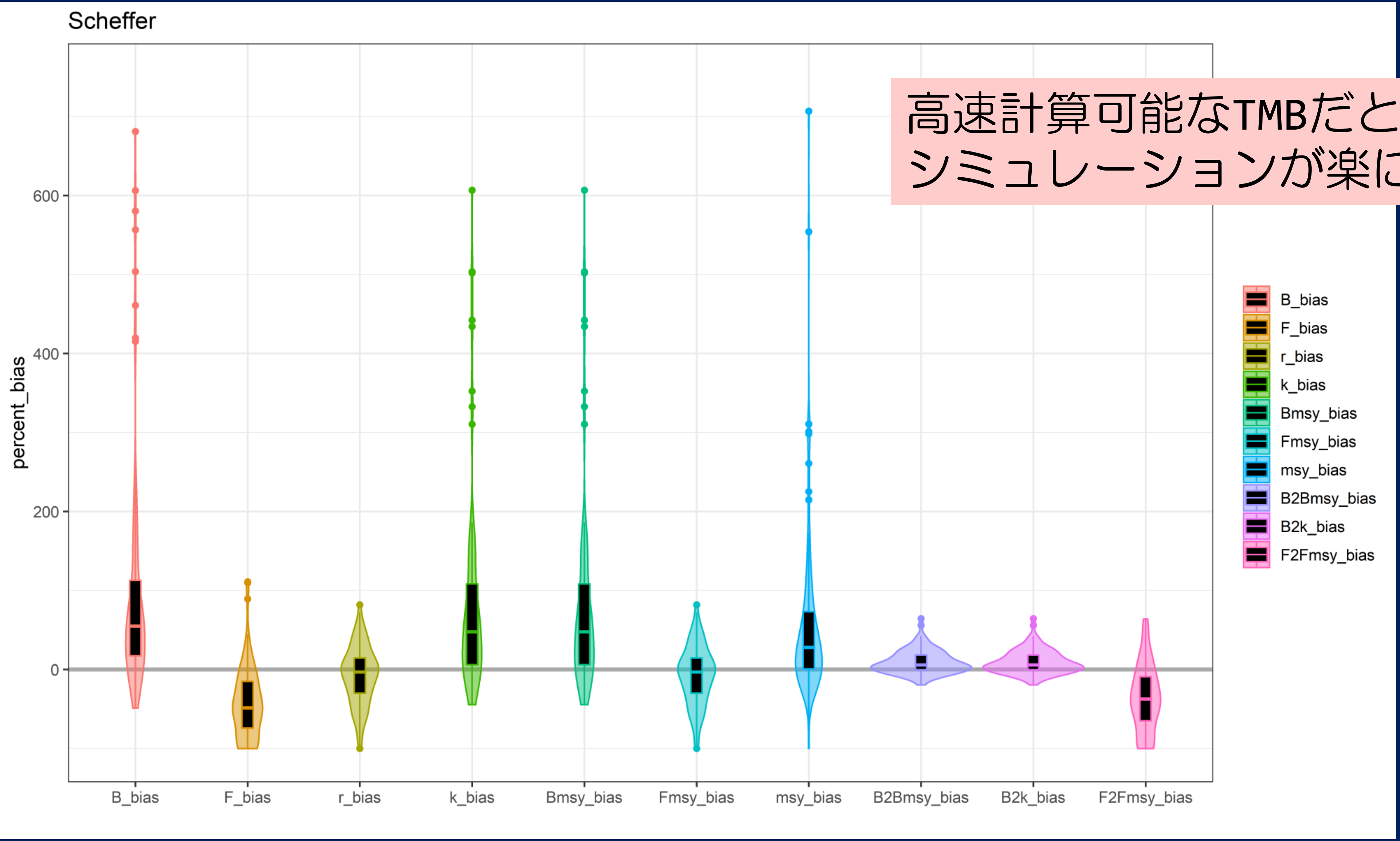
パラメータ推定結果②

```
> (r_est = exp(obj$env$parList()$log_r))
[1] 1.422481                  真値: 1.5
>
> (K_est =exp(obj$env$parList()$log_K))
[1] 107.1503                  真値: 100
>
> r_est*K_est/4  # MSY estimate
[1] 38.1048                   真値: 37.5
```

内的自然増加率、環境収容力、MSYはおおむね一致する

ただ、乱数・パラメータ設定に依存し、結構不安定…

シミュレーションによるバイアス評価



高速計算可能なTMBだと
シミュレーションが楽にできる

本日の内容

1. 一般化線形モデルを使ったランダム効果の説明・RとTMBによる最尤推定
2. TMBを使った余剰生産（surplus production）モデルのパラメータ推定
3. Delay-difference modelを用いた、生物特性とMSYの関係
4. 年齢別個体群動態モデル（ridge VPA & SAM）
5. SAMの発展

Delay-difference model

- 余剰生産モデル（非現実的）と年齢別モデル（複雑）の中間
- 現実的かつシンプルで考えやすい

翌年の親の個体数 = 加入 + 生残

$$N_{t+1} = R_{t+1} + l_t N_t$$

親が漁獲と自然死亡で減少

$$l_t = l_0 \times l_{Ft} = \exp(-M - F_t)$$

産卵親魚量と加入の関係
「再生産関係」

$$R_t = f(S_{t-a_0}) \times \exp(\varepsilon_t)$$

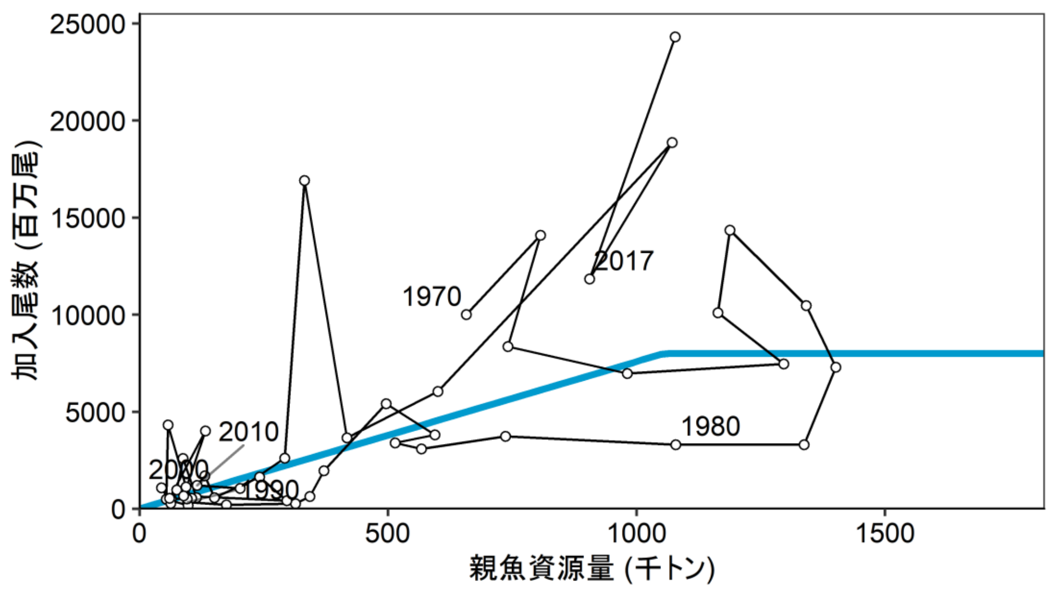
漁獲と自然死亡後に
繁殖すると仮定」

$$S_t = l_t N_t$$

Delay-difference model

- 成長・体重を組み込んだモデルもあるが、今回は説明しない
- １系の新ルール（漁獲管理規則）のMSEに使用
- <u>加入変動を考慮したMSYと生物特性の関係について考察</u>
- 再生産関係はHockey-stick型を仮定

$$f(S_t) = \begin{cases} a \times S_t, & S_t < b \\ a \times b, & S_t \geq b \end{cases}$$



マサバ太平洋系群の再生産関係

無次元化（非次元化？）

F=0の時の平衡点を考える

$$l_0 = \exp(-M)$$

$$N_0^* = R_0^* + l_0 N_0^*$$

$S_0^* = l_0 N_0^* \geq b$として

$$R_0^* = a \times b$$

$$N_0^* = \frac{a \times b}{1 - l_0}$$

$$N_{t+1} = R_{t+1} + l_t N_t$$

N_0^*で割って$X_t = N_t / N_0^*$，$r_t = R_t / N_0^*$と置くと

$$X_{t+1} = r_{t+1} + l_t X_t$$

無次元化（非次元化？）

$$r_t = \underbrace{g(l_{t-a_0} \times X_{t-a_0})}_{\text{産卵親魚量S}} \times \exp(\varepsilon_t)$$

$$\varepsilon_t \sim Normal(0, \sigma^2)$$

再生産関係

$$g(X_t) = \begin{cases} \dfrac{l_{Ft}X_t}{\beta} \times (1 - l_0), & l_{Ft}X_t < \beta \\ 1 - l_0, & l_{Ft}X_t \geq \beta \end{cases}$$

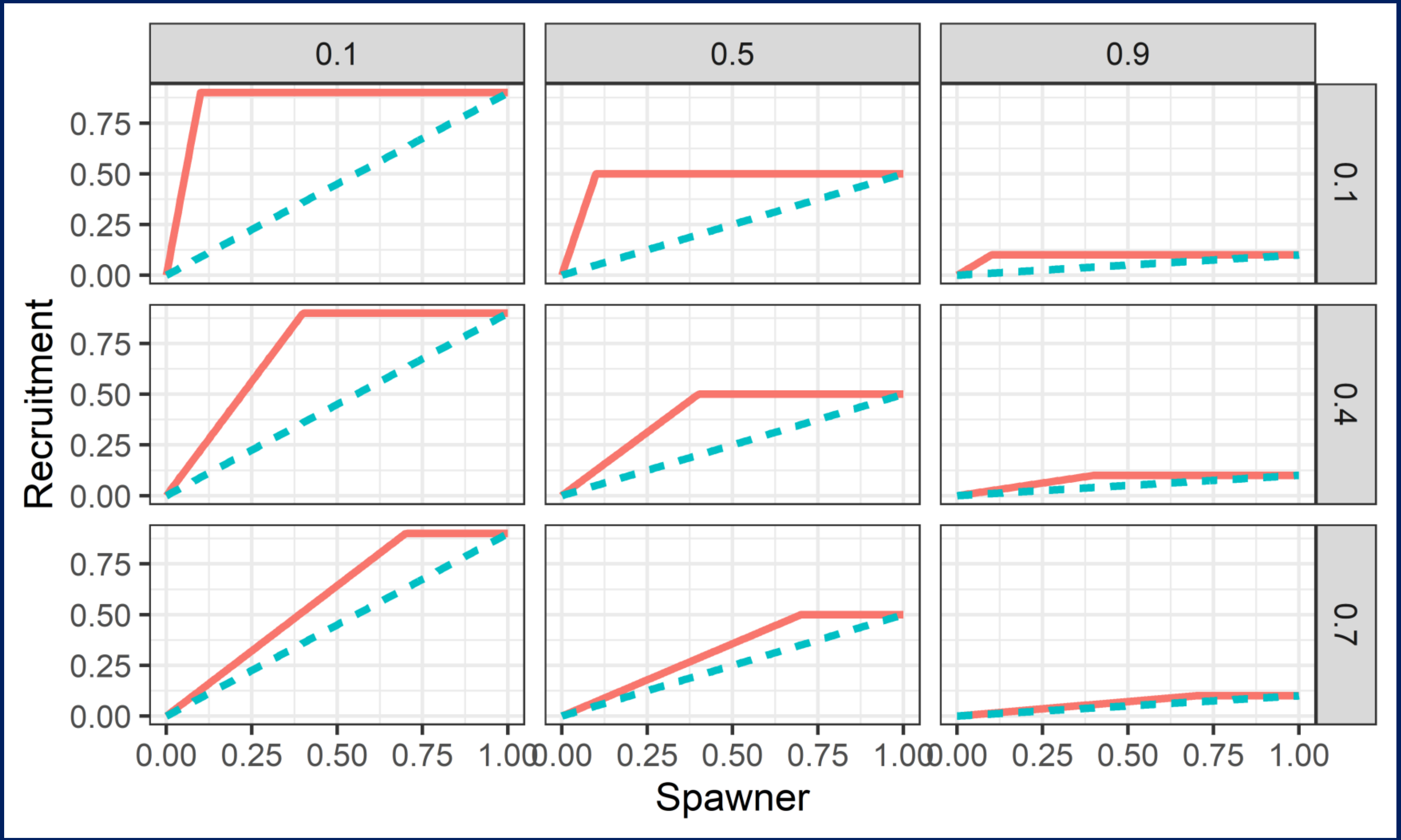
ただし $\beta = b/(l_0 N_0^*)$

パラメータは3つ

1. 自然生存率（寿命）　l_0
2. 相対的な折れ点　β
   （密度効果）
3. 加入変動の大きさ　σ

再生産関係の変化

自然死亡率高　l_0　長寿命



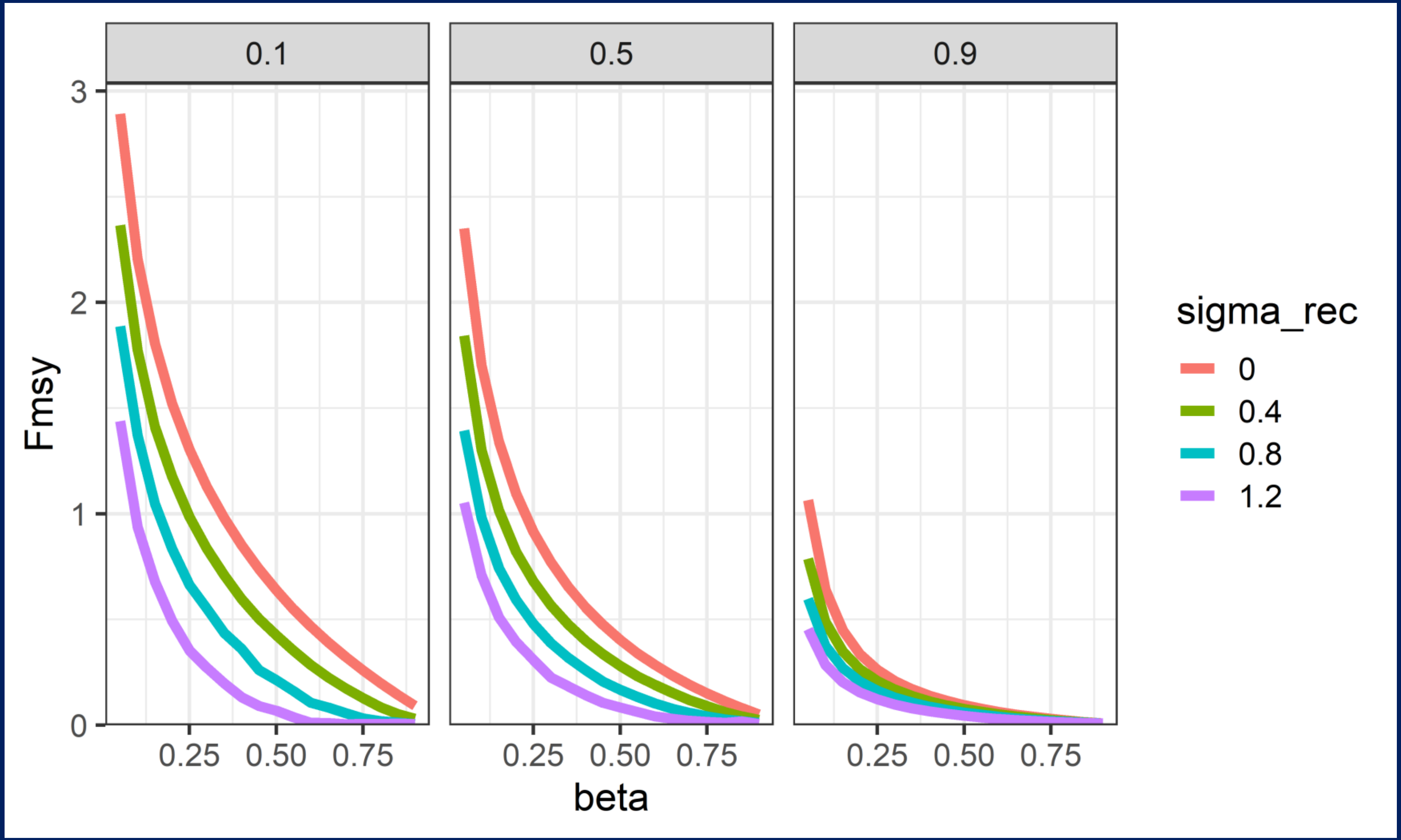
β

密度効果強い
（加入一定に近い）

密度効果弱い
（比例関係に近い）

漁獲係数Fmsyの変化

自然死亡率高　l_0　長寿命

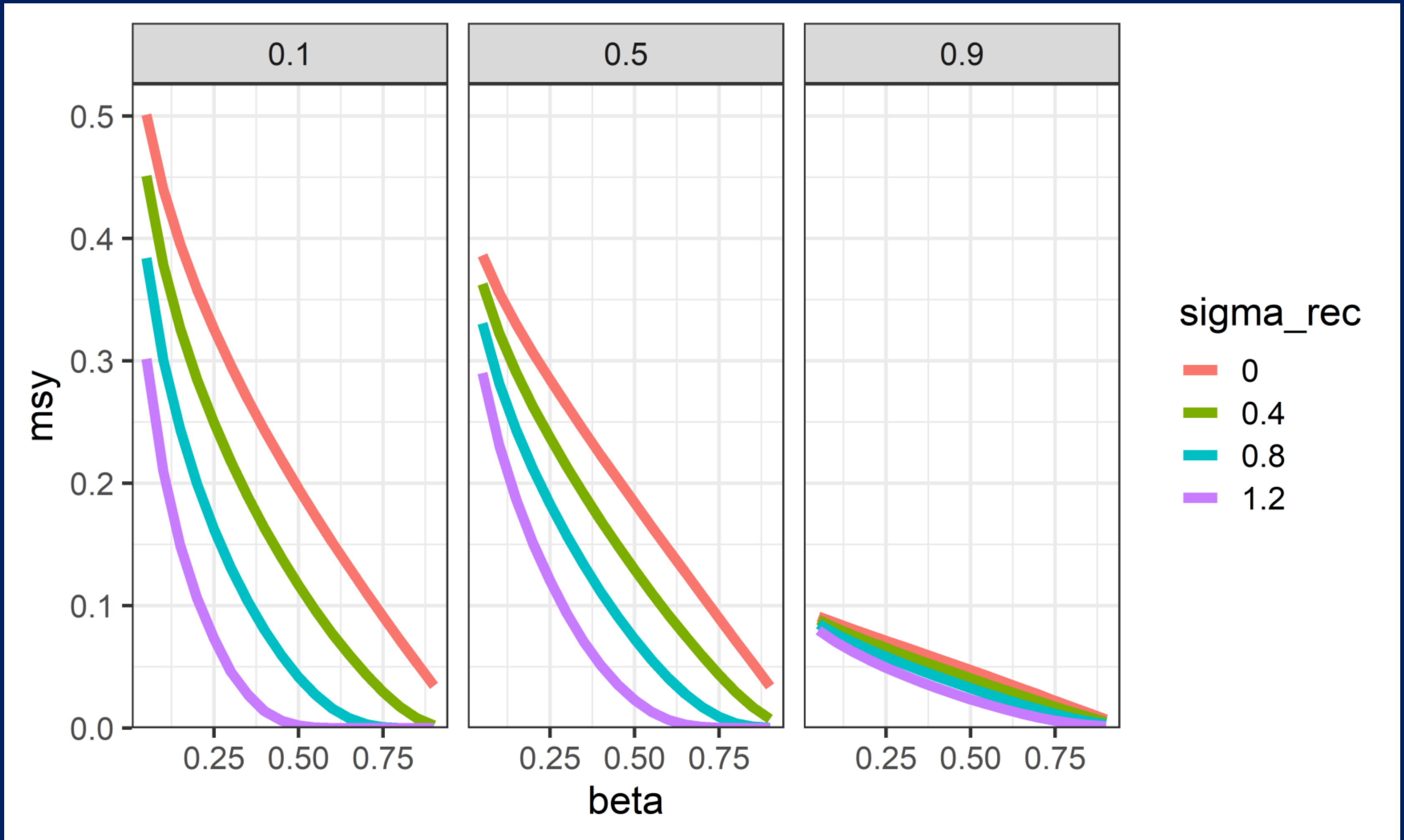


- 密度効果強い
- 自然死亡率高い
- 加入変動が小さい

方がFmsy高い

最大持続可能漁獲量MSYの変化

自然死亡率高　l_0　長寿命



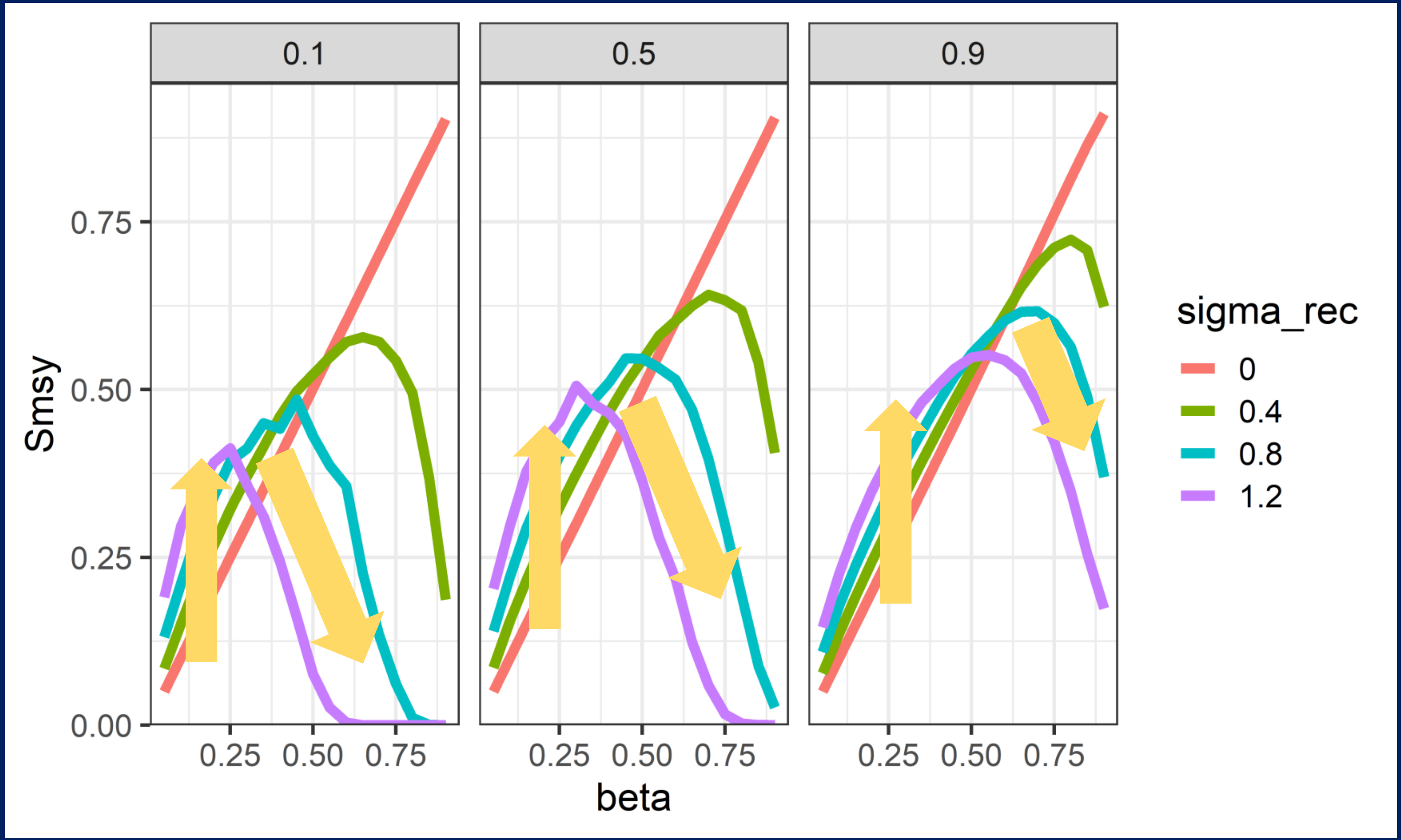
- 密度効果強い
- 自然死亡率高い
- 加入変動が小さい

方がMSY高い

- 自然死亡率が高いと、折れ点の位置に対してMSYが大きく変化する

親魚量Smsyの変化

自然死亡率高　l_0　長寿命



- 加入変動がないときはSmsy＝beta
- beta小さいとき、加入変動によりSmsyは折れ点以上
- beta大きいときや自然死亡率高いとき、加入変動によりSmsyは折れ点以下に

親魚を残せない

本日の内容

1. 一般化線形モデルを使ったランダム効果の説明・RとTMBによる最尤推定
2. TMBを使った余剰生産（surplus production）モデルのパラメータ推定
3. Delay-difference modelを用いた、生物特性とMSYの関係
4. 年齢別個体群動態モデル（ridge VPA & SAM）
5. SAMの発展

VPAの問題点

- 年齢別漁獲尾数の誤差を考慮できない

- 過去の不確実性は評価できない
- 近年の不確実性が大きい

- 再生産関係を仮定しない
- 直接的に将来予測ができない

- 選択率の変動が大きい

通常のSCAA (statistical catch at age model)

- 再生産関係を推定できる
- 年齢別漁獲尾数の誤差も推定できる

- <u>基本的には、選択率一定を仮定</u>

- 選択率の変動を考慮するためには、地域別・漁具別のデータが必要
- 収集・解析が大変

State-space assessment model (SAM)

- SCAAの１種 (Nielsen and Berg 2014 Fish.Res.)
- 状態空間モデルなので、<u>観測誤差と過程誤差を分離できる</u>
- 再生産関係も推定できる
- <u>選択率の時間変動を考慮できる</u>
- <u>VPAと同じデータから推定可能</u>
- TMBを用いて高速で最尤推定
- ICESでは資源評価に使用されている

F at ageのモデリング

$$\log \boldsymbol{F}_{\boldsymbol{y}} = \log \boldsymbol{F}_{\boldsymbol{y-1}} + \xi_{\boldsymbol{y}}$$

$$\boldsymbol{F}_{\boldsymbol{y}} = \left(F_{0,y}, F_{1,y}, \ldots F_{A,y}\right)'$$

$$\xi_y \sim \mathbf{MVN}(\mathbf{0}, \Sigma)$$

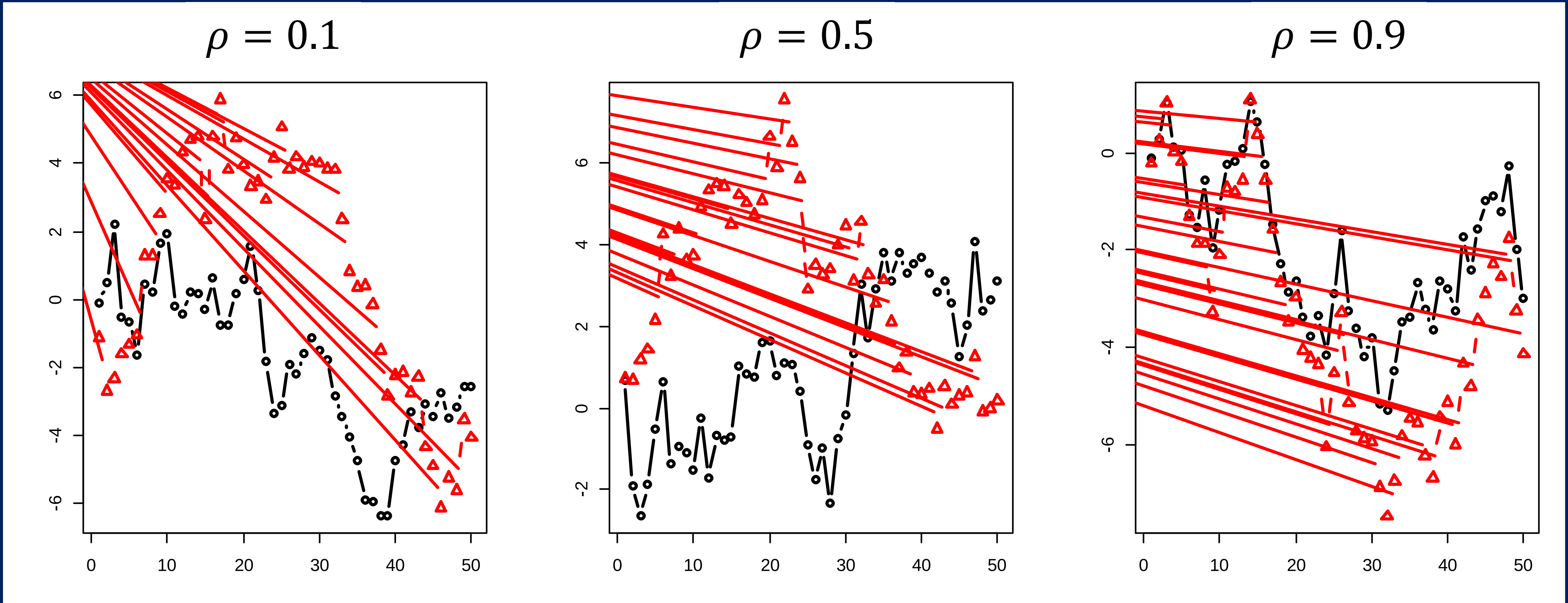
多変量正規分布

分散共分散行列

$$\Sigma = \begin{pmatrix} \sigma_0^2 & \cdots & \rho_{0,A}\sigma_0\sigma_A \\ \vdots & \ddots & \vdots \\ \rho_{0,A}\sigma_0\sigma_A & \cdots & \sigma_A^2 \end{pmatrix}$$

- 多変量正規分布によるランダムウォーク
- 対角成分が各年齢の分散（σ_a^2）
- それ以外が異なる年齢間の共分散($\rho_{a,a'}\sigma_a\sigma_{a'}$)
- ρ：相関係数（0：独立、1：選択率一定に近い）

多変量正規分布によるランダムウォーク



Cppファイルでの多変量正規分布の設定

```
using namespace density;  // 多変量正規分布を使う宣言
  MVNORM_t<Type> neg_log_densityF(fvar);
```

分散共分散行列fvarによる負の対数尤度を定義

```
  Type ans=0;
  array<Type> logF_resid(stateDimF,timeSteps); //
  for(int i=1;i<timeSteps;i++){
    ans+=neg_log_densityF(logF.col(i)-logF.col(i-1));
```

Fのランダムウォークにおける負の対数尤度

```
    SIMULATE {
      logF.col(i) = logF.col(i-1) + neg_log_densityF.simulate();
    }
  }
```

ランダム効果のシミュレートができる

タイセイヨウダラの例 (Nielsen and Berg 201)

プロファイル尤度

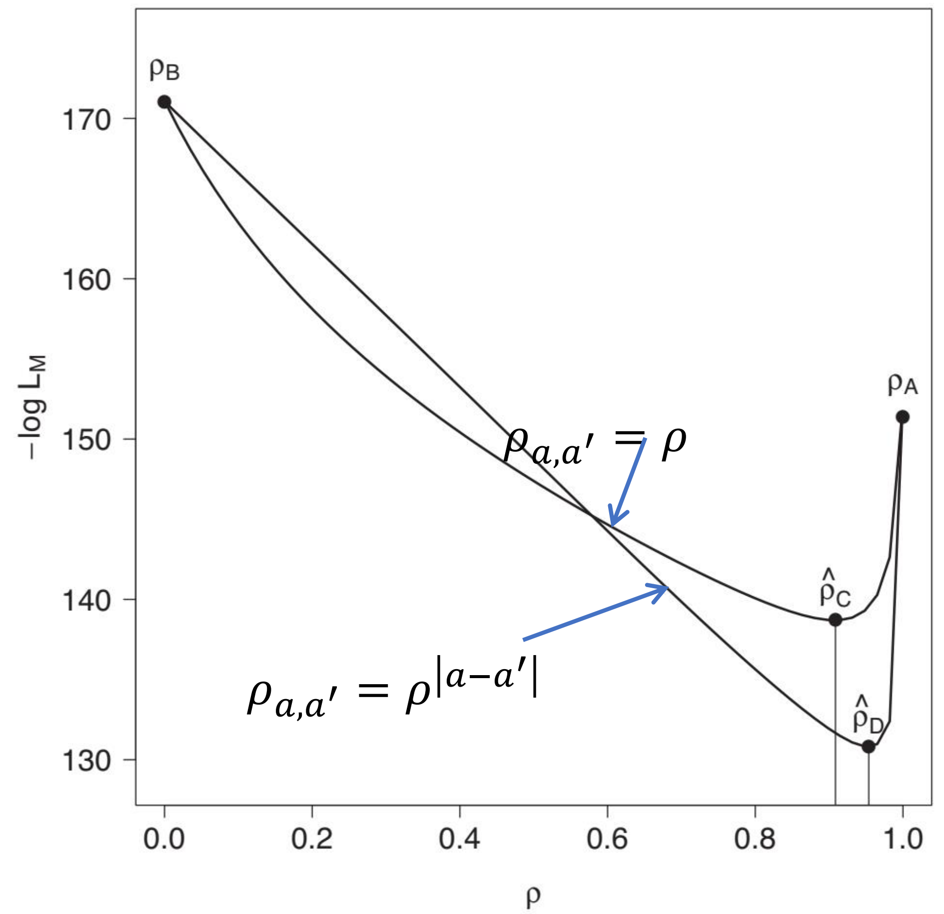
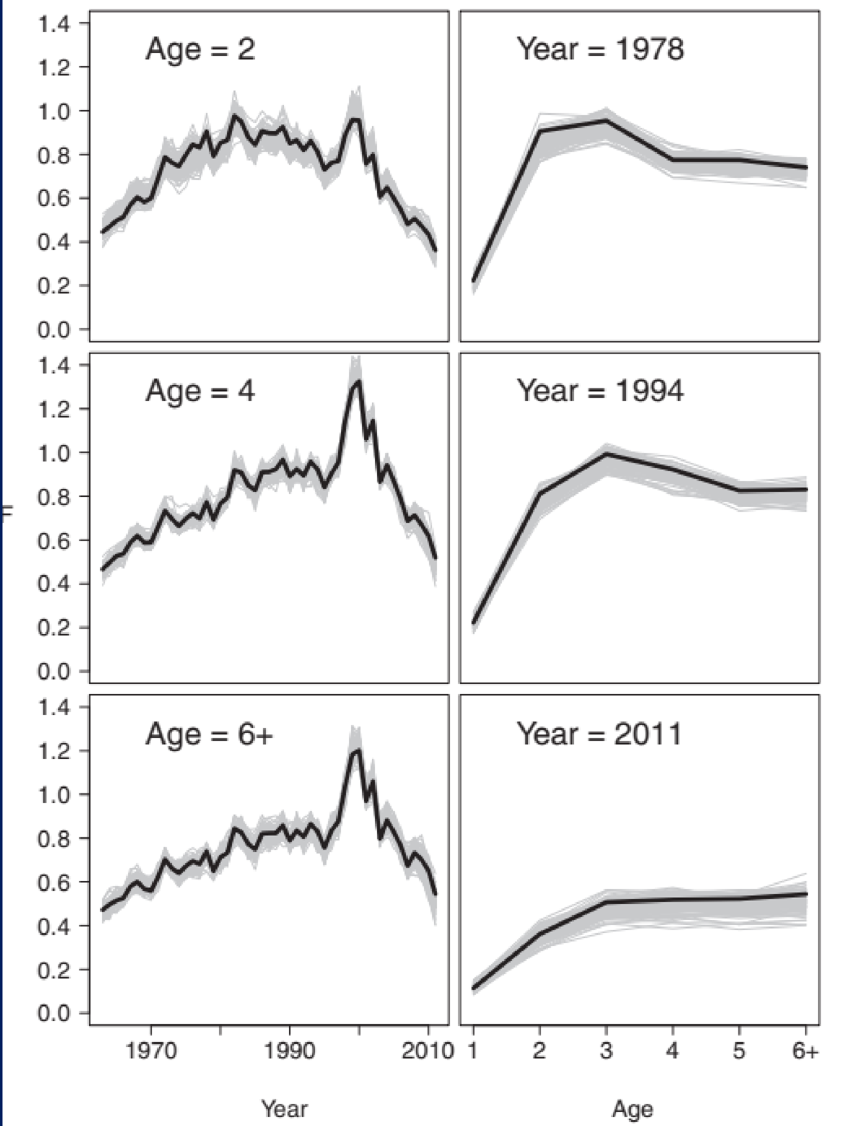


Fig. 5. North Sea cod: profile likelihood for the ρ-parameter for models C and D, $\rho = 1$ corresponds to model A, and $\rho = 0$ corresponds to model B.



そのほかの設定

- 加入プロセスは３種類　Hockey-stickは試行錯誤中...
  - Beverton-Holt
  - Ricker
  - Random walk
- チューニングの仕方はVPAと同様でb推定もできる
- CAAの分散（観測誤差）・FAAの分散（ランダムウォークの変動の大きさ）・NAAの分散（過程誤差）の設定は自由に変えられる
- 例えば、「CAAの誤差は若齢・高齢の方が大きい」とか、「個体数の誤差は加入年齢で大きい」とかを設定できる
- 複雑にしすぎたり、データの性質と合わないと（？）収束しない（ヘッセ行列の対角成分が正にならない）

rsamコード

```
sam_base <- sam(dat, #rvpaと同じデータ
                last.catch.zero = TRUE,
                abund = c("N","N","SSB","SSB"),
                cpp.file.name = "jsam9",
                index.age=c(0,0,0,0),
                b.est=TRUE,
                b.fix=c(NA,NA,1,1),
                SR = "BH",
                varC = c(0,1,2,2,3,4,4),
                varF = c(0,0,1,1,1,1,1),
                varN = c(0,1,1,1,1,1,1),
                varN.fix=c(NA,1e-4),
                rho.mode=3,
                bias.correct = TRUE,
                get.random.vcov = FALSE
)
```

rvpaとなるべく同じに

b推定

再生産関係（BH/RI/ RW)

分散の設定

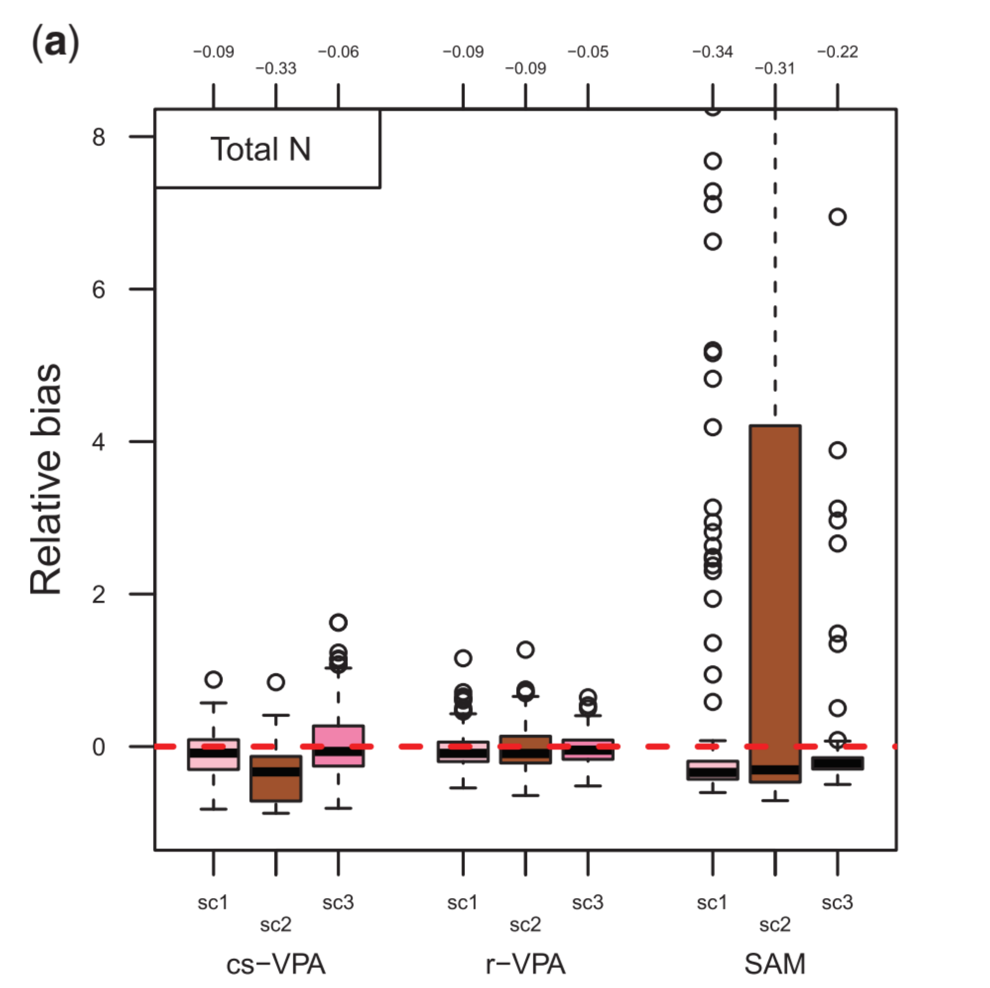
VPAとSAMの比較表

	VPA	SAM
計算方法	後ろ向き	前向き
年齢別漁獲尾数	正確と仮定	観察誤差を推定
再生産関係	推定せず	推定可
漁獲死亡係数（選択率）	仮定なし	ランダムウォーク
状態変数	固定効果	ランダム効果
個体群プロセス	決定論的	確率論的
不確実性	近年で大きい（昔は無し）	全期間で生じる

(Ridge-)VPA vs SAMの既存研究

Ridge VPA (Okamura et al. 2017, ICESJMS)

- 最終年のFに罰則を課すことでFの発散（過剰適合）を抑える
- 罰則の大きさはレトロスペクティブ解析により決定



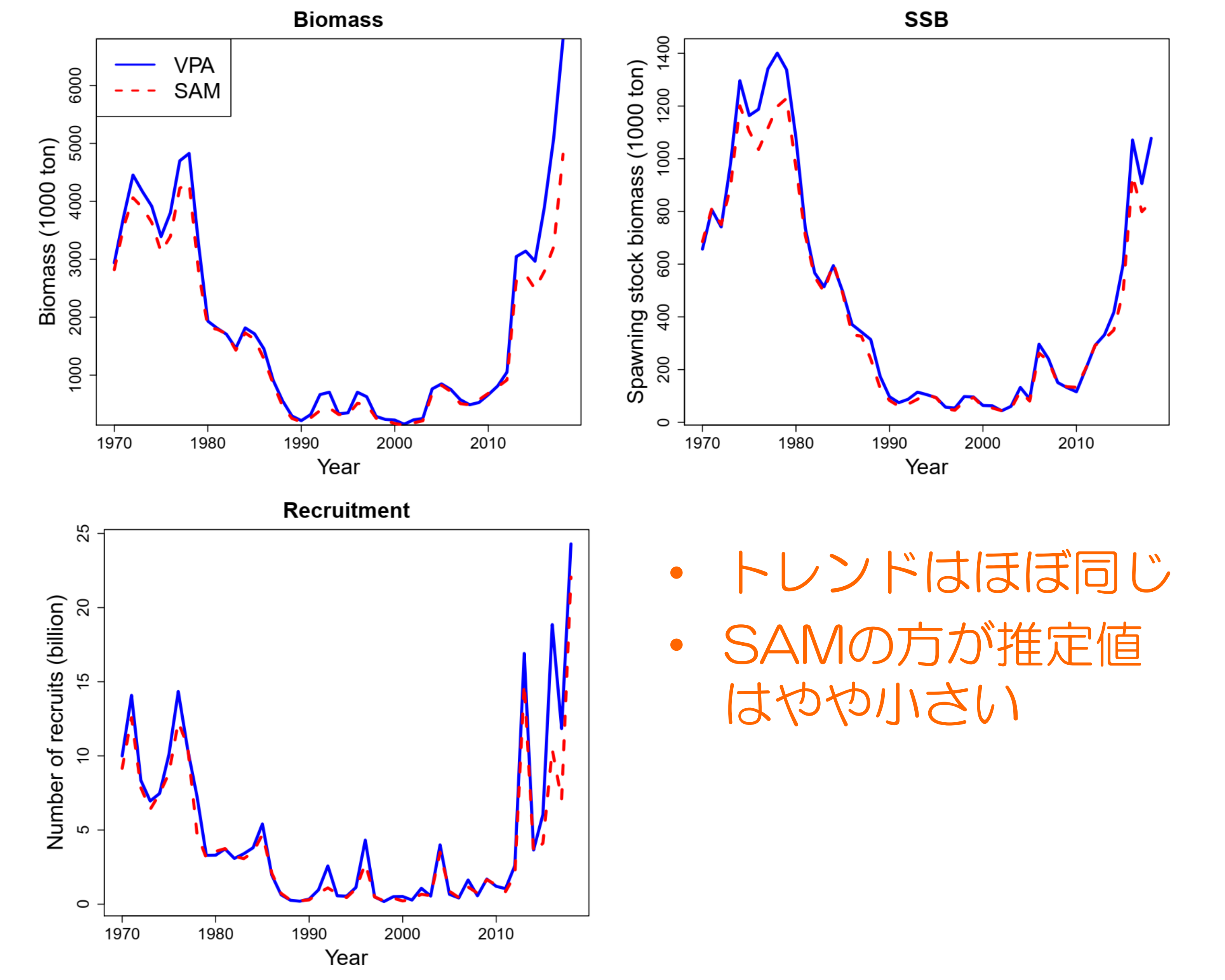
罰則付き尤度

$$\underbrace{(1-\lambda)\sum_{k=1}^{K}\sum_{y=1}^{Y}\left[\log(\sigma_k)+\frac{(I_{k,y}-\mu_{k,y})^2}{2\sigma_k^2}\right]}_{\text{対数尤度}}+\underbrace{\lambda\sum_{a=1}^{A-1}|F_{a,Y}|^{\beta}}_{\text{罰則項}}$$

スケトウダラ日本海系群を対象にしたシミュレーションでは、年齢別の資源量指数がないとき、SAMはridge VPAよりもアバンダンス推定値のバイアスが大きい

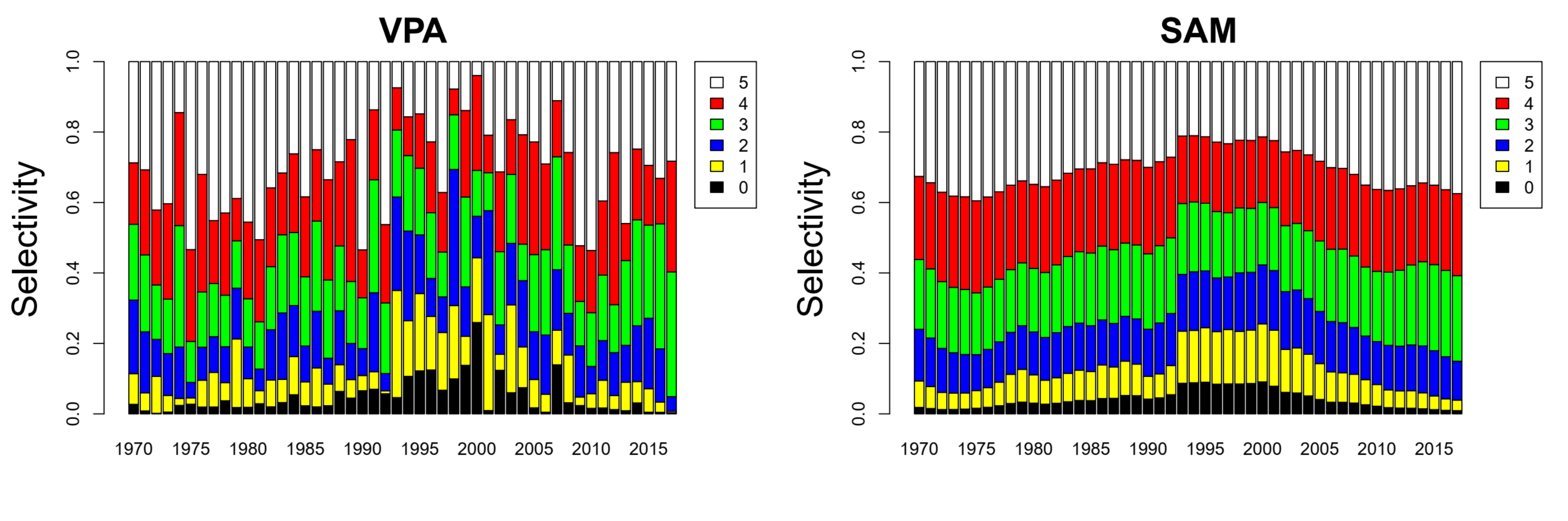
(Okamura et al. 2018, ICESJMS)

マサバ太平洋系群への適用



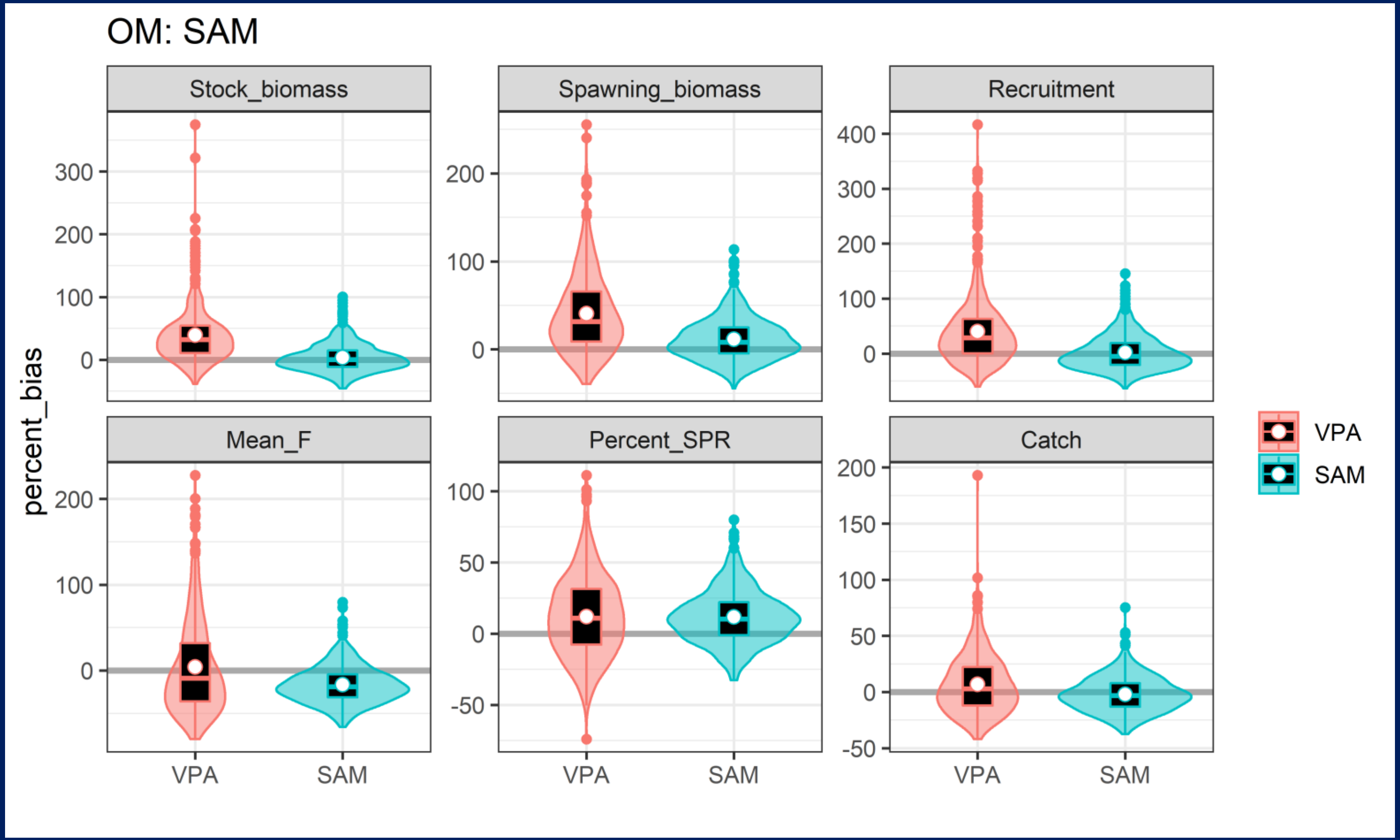
- トレンドはほぼ同じ
- SAMの方が推定値はやや小さい

マサバ太平洋系群への適用：選択率



- VPAでは変動が激しい
- SAMでは平滑化
- 近年若齢への選択率が減少傾向

SAMがシミュレーションモデルの場合 最新年のバイアス

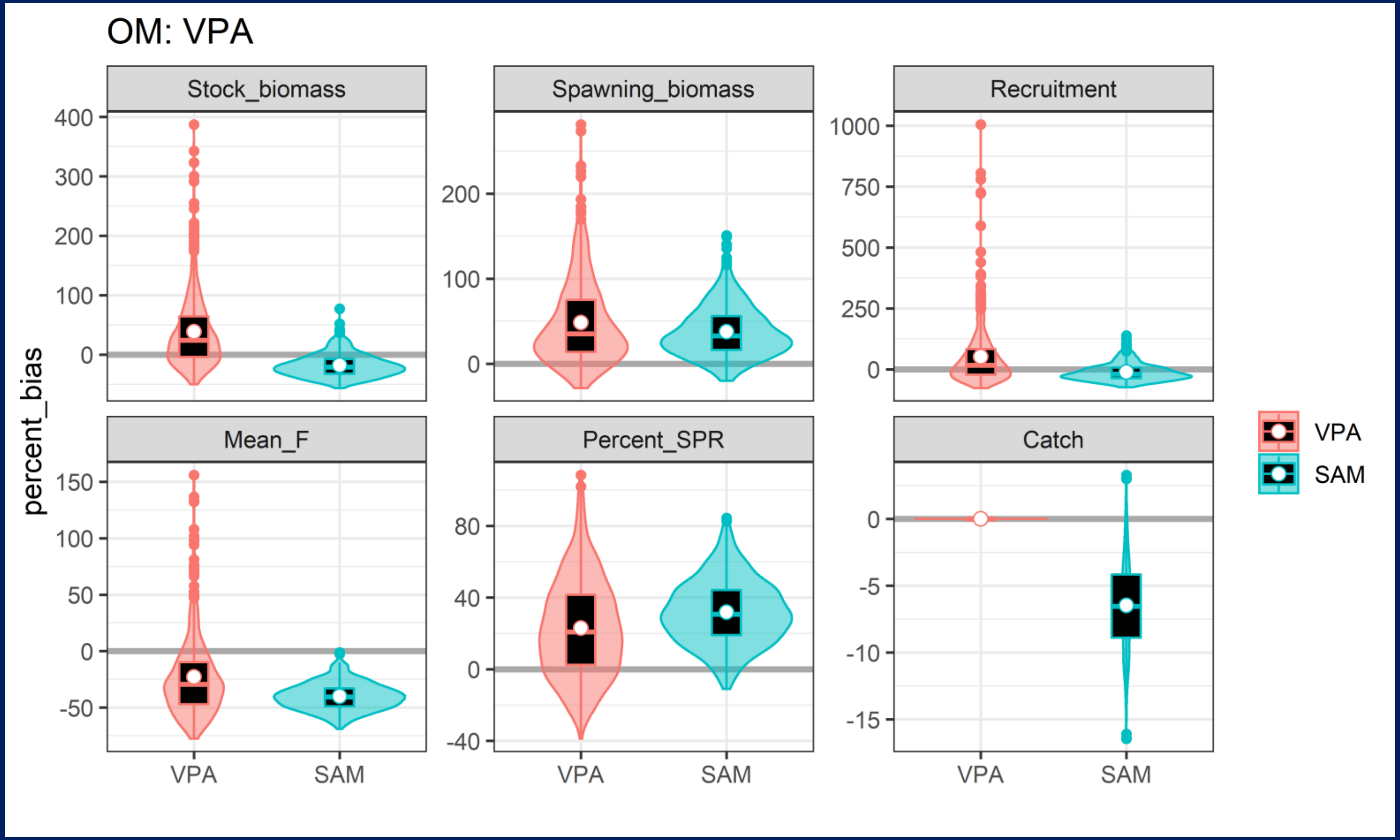


SAMだとアバンダンス推定はバイアスしない

VPAは不確実性が大きく、アバンダンスは過大評価傾向

漁獲圧はどちらも同程度に過小評価傾向

VPAがシミュレーションモデルの場合
最新年のバイアス



OMと同じ構造でもVPAはバイアスする

SAMもバイアスが生じるが、VPAと同程度

SAMの方がいい？

どちらが良いかは状況依存的？

- データの利用可能量
- 指標値の当てはまり
- 再生産関係の明瞭さ
- 漁獲圧の強さ
- etc…

- モデルの診断（レトロスペクティブ解析等）
- シミュレーションモデル

によって比較するのがいい

本日の内容

1. 一般化線形モデルを使ったランダム効果の説明・RとTMBによる最尤推定
2. TMBを使った余剰生産（surplus production）モデルのパラメータ推定
3. Delay-difference modelを用いた、生物特性とMSYの関係
4. 年齢別個体群動態モデル（ridge VPA & SAM）
5. SAMの発展

スルメイカ資源評価の難しさ

- スルメイカは単年性資源であるため、年齢別モデルが使えない
- これまでの資源評価では資源量指数を引き延ばして資源量推定していた
- 資源量指数は観察誤差により変動が激しい傾向があるため、資源量の推定値が大きく変動する
- 寿命が長い水産資源と比べ、親魚量の推定誤差も大きく、再生産関係の推定には親魚量の誤差も考慮したほうがよさそう

スルメイカ版のSAMを開発中

- 情報不足を補うため，秋季発生系群と冬季発生系群のデータを同時に使用

個体群動態モデル

産卵尾数 $S_{i,y} = N_{i,y} \times \exp\left(-F_{i,y} - M\right)$

加入尾数 $N_{i,y} = f(a_i, b_i, S_{i,y-1}) \times \exp\left(\varepsilon_{i,y}\right)$： $f(a, b, S) = aS/(1 + bS)$

漁獲係数 $\log F_{i,y} \sim \mathrm{Normal}(\log F_{i,y-1}, \tau_i^2)$

$$\boldsymbol{\varepsilon}_y = \left(\varepsilon_{0,y}, \varepsilon_{1,y}\right)' \sim \mathrm{MVN}(\mathbf{0}, \boldsymbol{\Sigma}), \quad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_0^2 & \rho\sigma_0\sigma_1 \\ \rho\sigma_0\sigma_1 & \sigma_1^2 \end{pmatrix}$$

加入の残差の相関を考慮

スルメイカ版のSAMを開発中

- 情報不足を補うため，秋季発生系群と冬季発生系群のデータを同時に使用

観察モデル

漁獲量　$\log C_{i,y} \sim \mathrm{Normal}(\log \hat{C}_{i,y}, \omega_i^2)$

$$\hat{C}_{i,y} = \frac{F_{i,y}}{F_{i,y} + M} \times w_{i,y} \times N_{i,y} \times \left[1 - \exp\left(-F_{i,y} - M\right)\right]$$

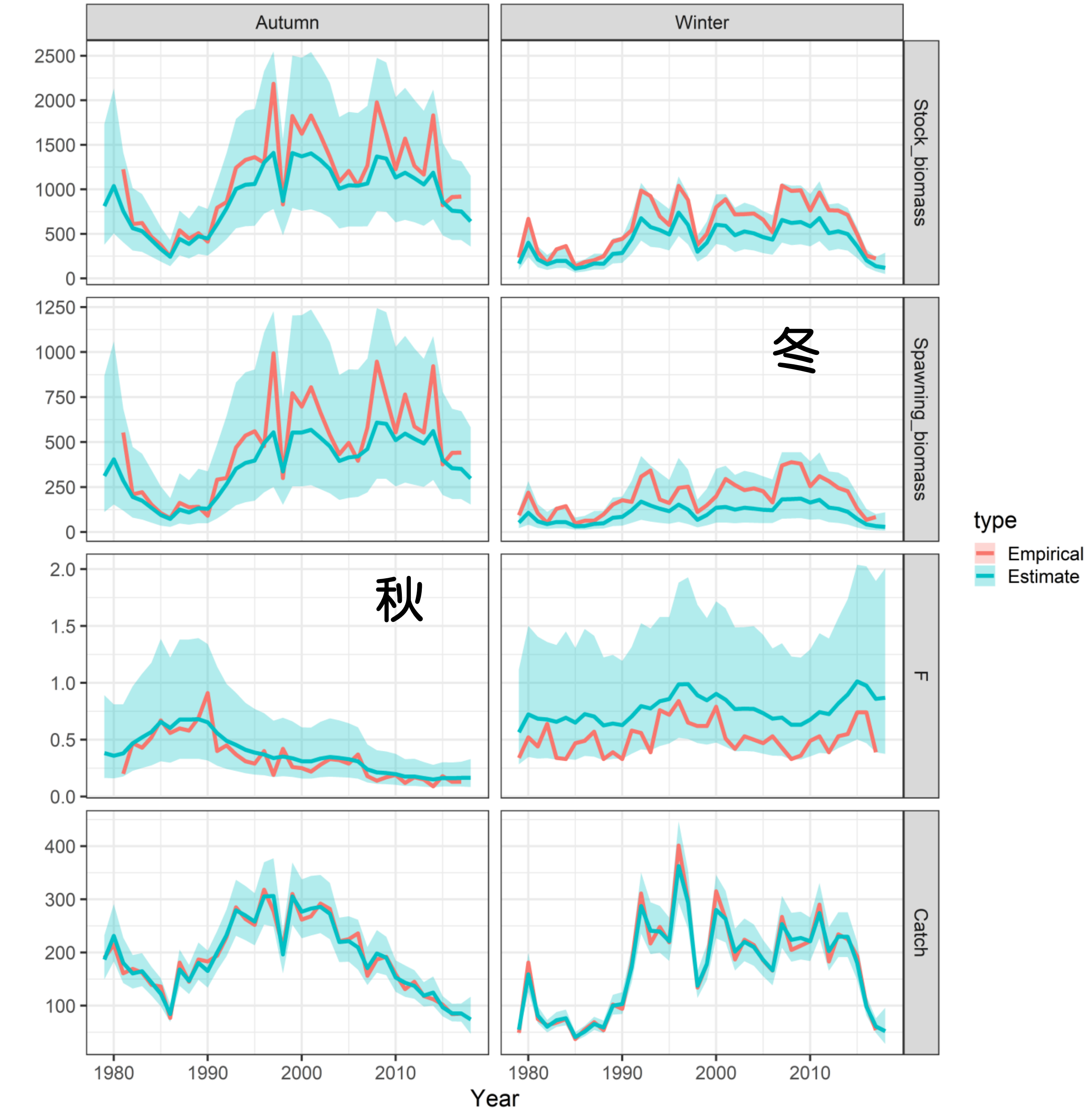
漁獲量の観察誤差も推定

資源量指数　$\log I_{i,y} \sim \mathrm{Normal}\left(\log\left(q_i \times N_{i,y}\right), \varphi_i^2\right)$

パラメータ推定結果

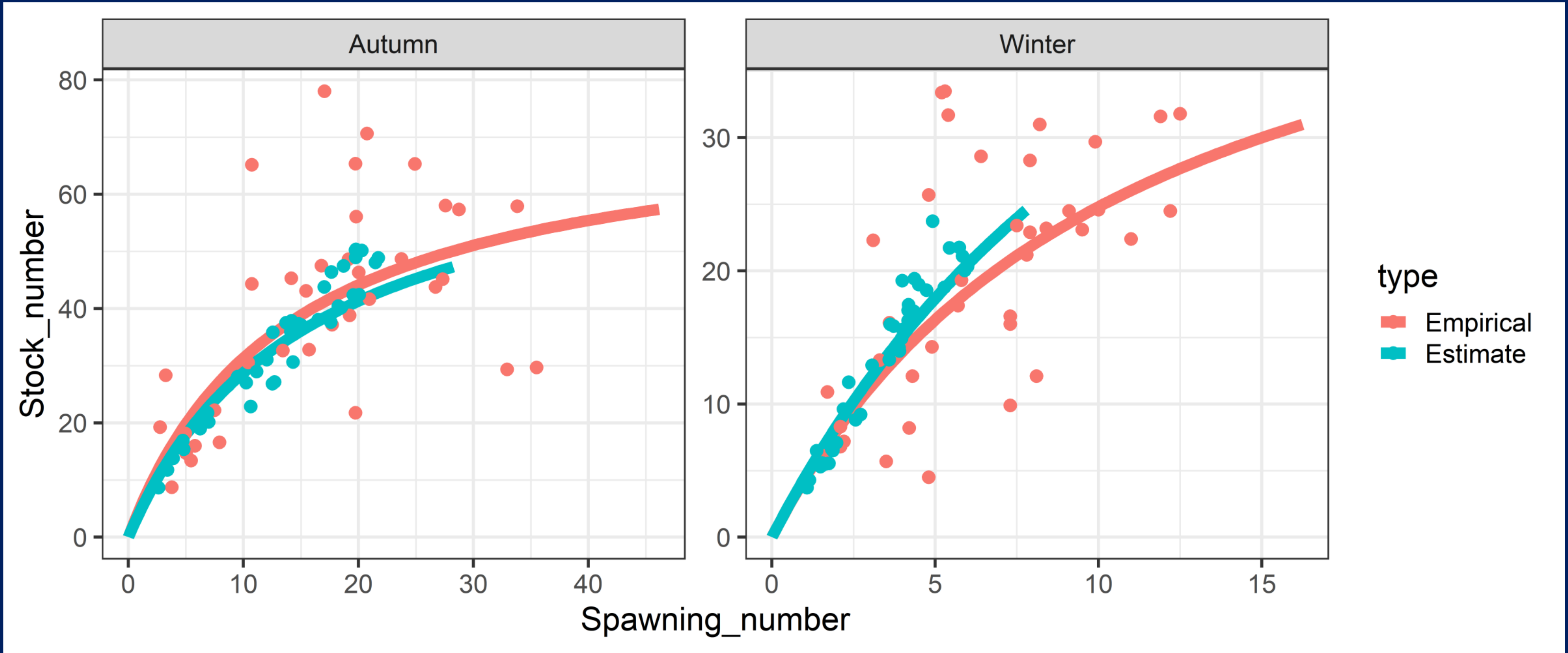
AICやレトロ
をもとに決定

記号	意味	制約	推定値*
<u>固定効果</u>			
τ_i	漁獲死亡係数のランダムウォークの標準偏差	$\tau_0 = \tau_1$	0.15
a_i	BH型再生産関係において原点での傾き	$a_0 = a_1$	4.75
b_i	BH型再生産関係における密度効果の強さ	$b_0 = b_1$	0.065
σ_i	加入変動の標準偏差	$\sigma_0 = \sigma_1$	0.29
ρ	系群間の加入変動の相関係数	-	0.64
ω_i	漁獲量の観測誤差の標準偏差	$\omega_0 = \omega_1$	0.11
q_i	漁具能率（比例定数）	-	0.261, 0.074
ϕ_i	資源量指数の観測誤差の標準偏差	$\phi_0 = \phi_1$	0.16
<u>ランダム効果</u>			
$F_{i,y}$	漁獲死亡係数	-	省略
$N_{i,y}$	資源尾数	-	省略



- 推定は可能
- 秋は今までの手法とほぼ同じ水準
- 冬の資源量やや小さく、F は大きく
- 資源量の変動パターンが平滑化
- 漁獲量の誤差は小さい

再生産関係



- これまでと大きく変わるわけではない
- 過程誤差が小さすぎ？

本日の復習

1. 個体群動態モデルには年齢構造を考えない・ちょっと考える・ちゃんと考えるモデルがある
2. ランダム効果とTMBによるプログラミングはそんなに難しくない・できそうだ
3. MSYと生物特性の関係はわずか3つの要因で決まる
4. 状態空間モデル（SAM）はVPAを上回る可能性がある
5. 情報が不足している場合、複数の系群の同時モデリングが解決策となる可能性がある