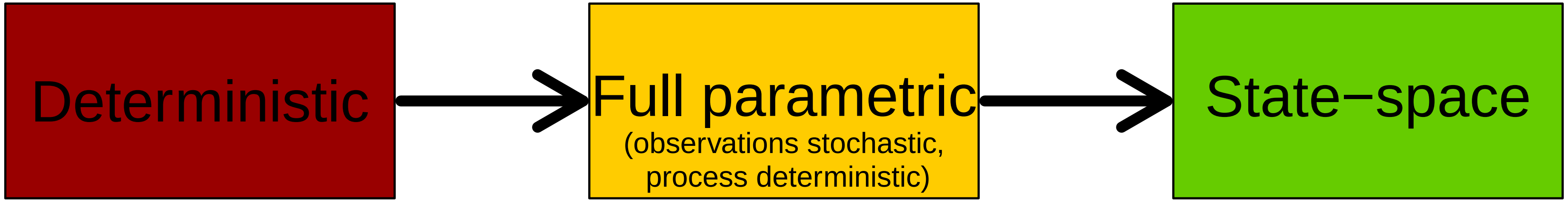
Model validation

Anders Nielsen, Ethan Lawler, & Sean Anderson

an@aqua.dtu.dk

Fish stock assessment models have evolved



- The obvious tool for time series data
- Quantification of observation errors
- Quantification of process errors
- Process formulation of time-varying quantities
- Reasonable (low) number of model parameters
- Prediction as part of model formulation

Some validations options are:

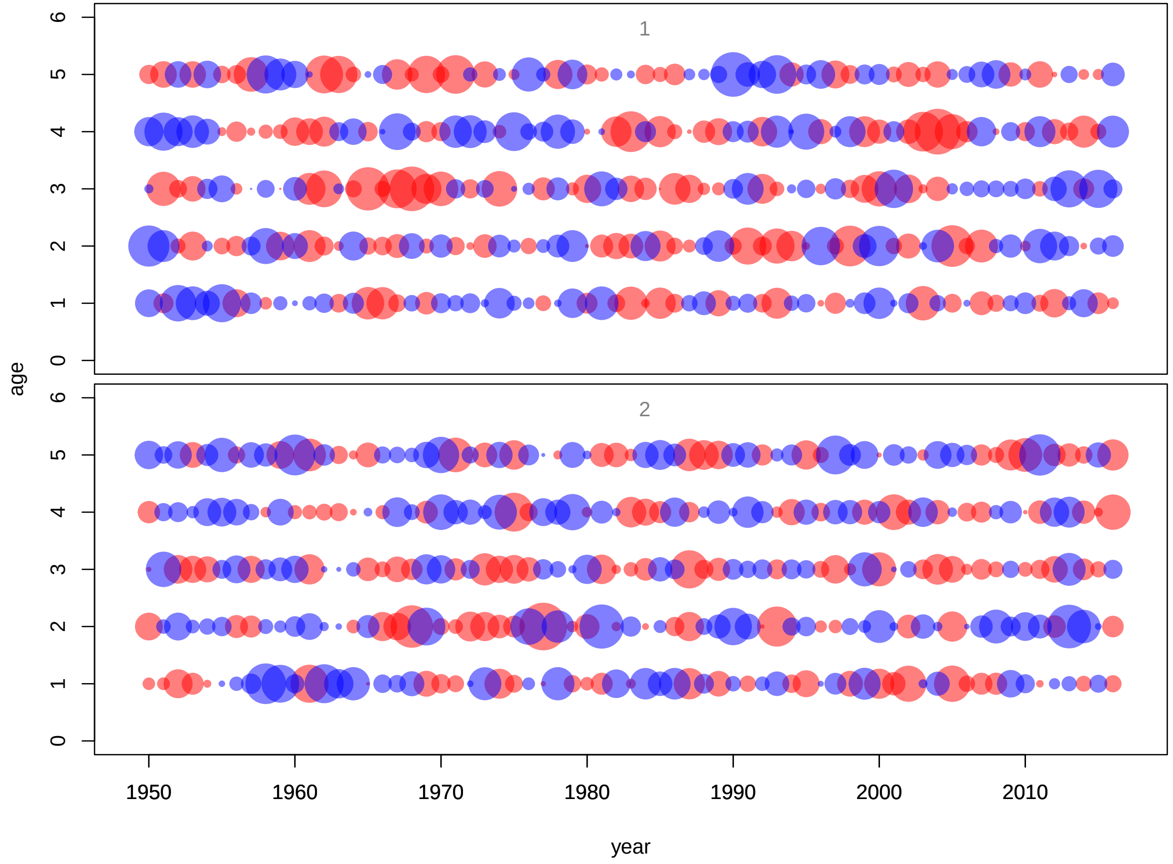
- Residuals (one-observation-ahead, process)
- Retrospective patterns of key outputs
- Leave-fleet-out runs (to check consistency between data sources)
- Jittered starting point analysis
- Simulation testing
- Likelihood ratio test
- Laplace checker
- Prediction based

Residuals (Pearson)

- Residuals are classically defined as:

$$r_i = \frac{\text{obs}_i - \text{pred}_i}{\text{sd}(\text{obs}_i)}$$

- What are we looking for in residuals?
- What does the model say about observation noise?
- When are those residuals good enough?



Exercise: Pearson residuals for the basic assessment model

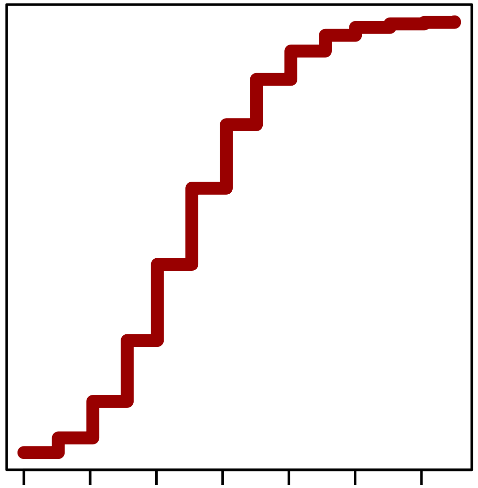
- An example where the Pearson residuals match the model formulation is the basic assessment model with independent observations
- Use the observations in `fsa.RData` and the code in `fsa.R`
- Add code to compute the Pearson residuals
- Plot the residuals as a function of e.g. predicted, age, year, and fleet

Residuals

- In state-space models, and other models with correlated observations, residuals calculated as $r_i = (y_i - \hat{y}_i)/\hat{\sigma}_i$ are not supposed to be independent $N(0,1)$ even in perfectly correct models.
- A safer alternative is the **one-observation-ahead** residuals $(y_i - \hat{y}_{i|i-1})/\hat{\sigma}_{i|i-1}$.
- More generally the **one-observation-ahead-quantile-residuals**

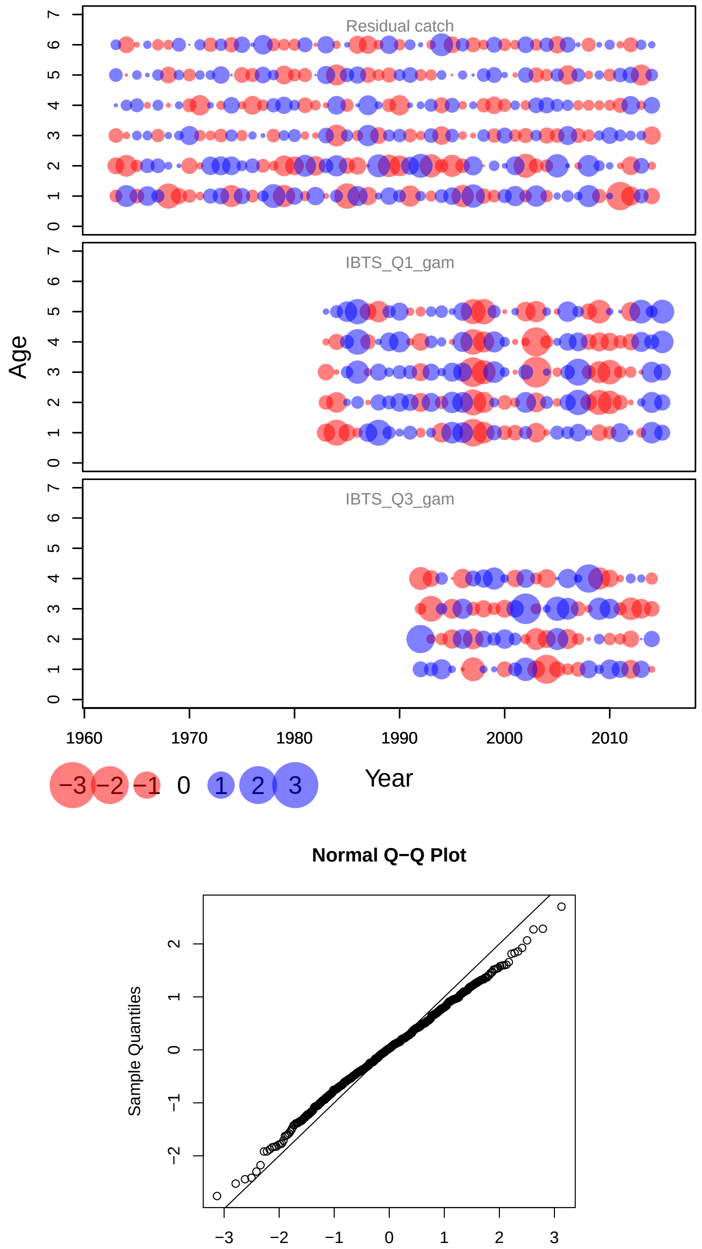
$$\Phi^{-1}(P(Y_i <= y_i | Y_{i-1} = y_{i-1} \ldots Y_1 = y_1))$$

- Randomized if originating from a discrete distribution

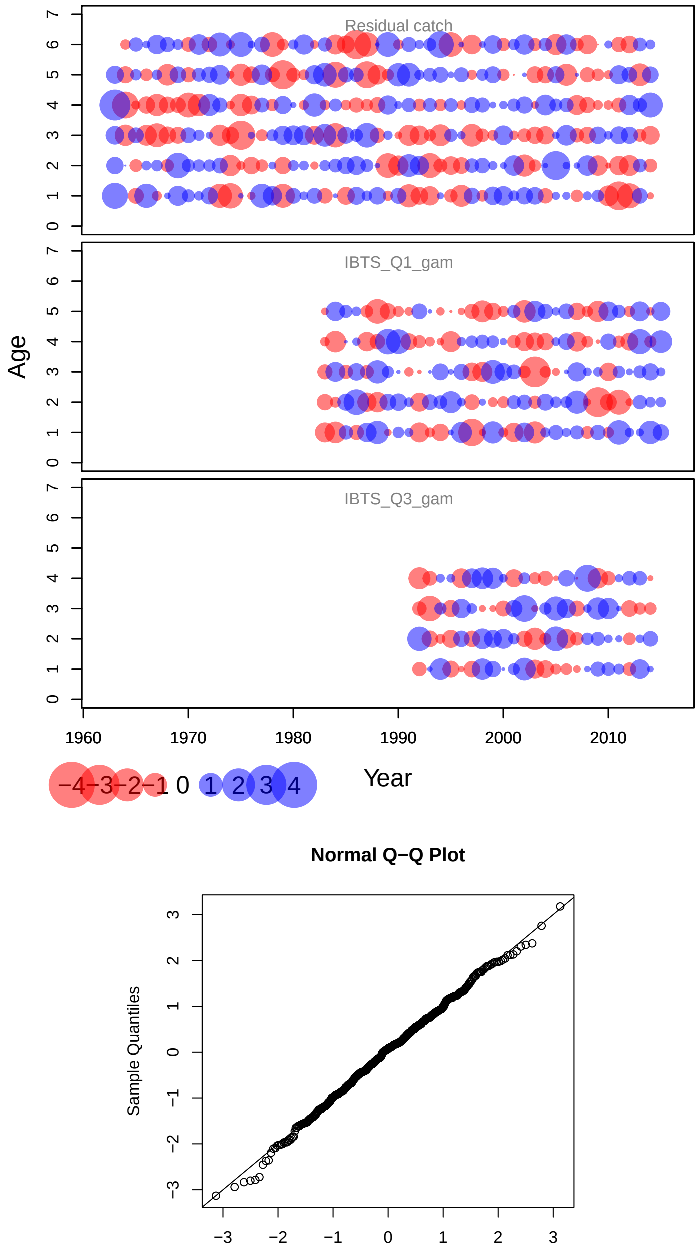


- Requires extra work when the model is solved via Laplace approximation
- But it does matter — the residuals are different.

Wrong $(y_i - \hat{y}_i)/\hat{\sigma}_i$



Right $\Phi^{-1}(P(Y_i <= y_i | Y_{i-1} = y_{i-1} \ldots Y_1 = y_1))$



Exercise: Same in independent Gaussian model?

- Compare the Pearson with the one-step-ahead residuals in the basic assessment model (`fsa.R`)
- First optimize the model, and remember to 'flag' the observation vector with

```
logObs <- OBS(logObs)
```

- Then the one-step-ahead residuals can be computed with:

```
res <- oneStepPredict(obj)
```

For discrete observations (randomization)

- Let $x_i \sim \text{pois}(\lambda)$ (with a c.d.f. P)
- Define $u_i \sim \text{unif}(P(x_i - 1), P(x_i))$
- Define $z_i = \Phi^{-1}(u_i)$
- Now $z_i \sim \mathcal{N}(0, 1)$

```
# observations
x <- rpois(1000,3)
ppois.u <- function(x, lambda){
  runif(length(x), ppois(x-1,lambda), ppois(x,lambda)) #uses the fact that ppois(-1,lambda)=0
}
U <- ppois.u(x,3)
Z <- qnorm(U)
```

rand.R

Mini exercise: Repeat this example with a different distribution (e.g. Negative binomial) to see that you can in fact get perfect $N(0, 1)$ residuals using this approach.

Exercise: AR1-Poisson example

Consider a model for catch-per-tow along a path. It is assumed that the counted catch follows:

$$y_i \sim \mathcal{P}(e^{\gamma_i})$$

where

$$(\gamma_i - \mu) \sim \mathcal{N}(\phi(\gamma_{i-1} - \mu), \sigma^2)$$

The data is available in the file `cpue.RData`.

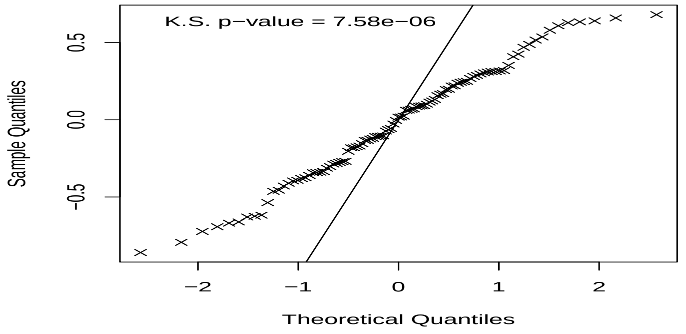
Implement the model and compute the one-obs-ahead-residuals

Try to change the observations to violate the model assumptions

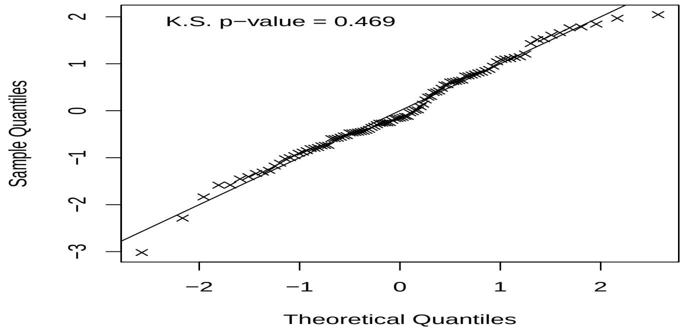
Process residuals

- Cannot just use the predicted random effects U, but
- If (Y, U) is distributed according to joint pdf. $L(y, u)$
- Observed y is then a sample from marginal distribution with pdf. $\int L(y, u)du$
- Generate one sample $u^\star$ from conditional distribution of $U|Y = y$
- Then the set $(y, u^\star)$ is a sample from joint distribution of (Y, U)
- Assumed distribution of $u^\star$ can be validated by standard tests

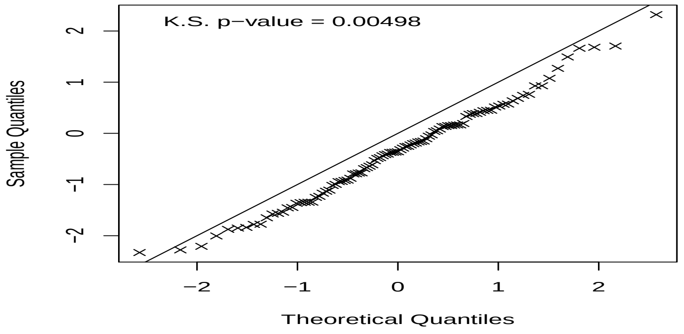
Wrong (using est. RE)

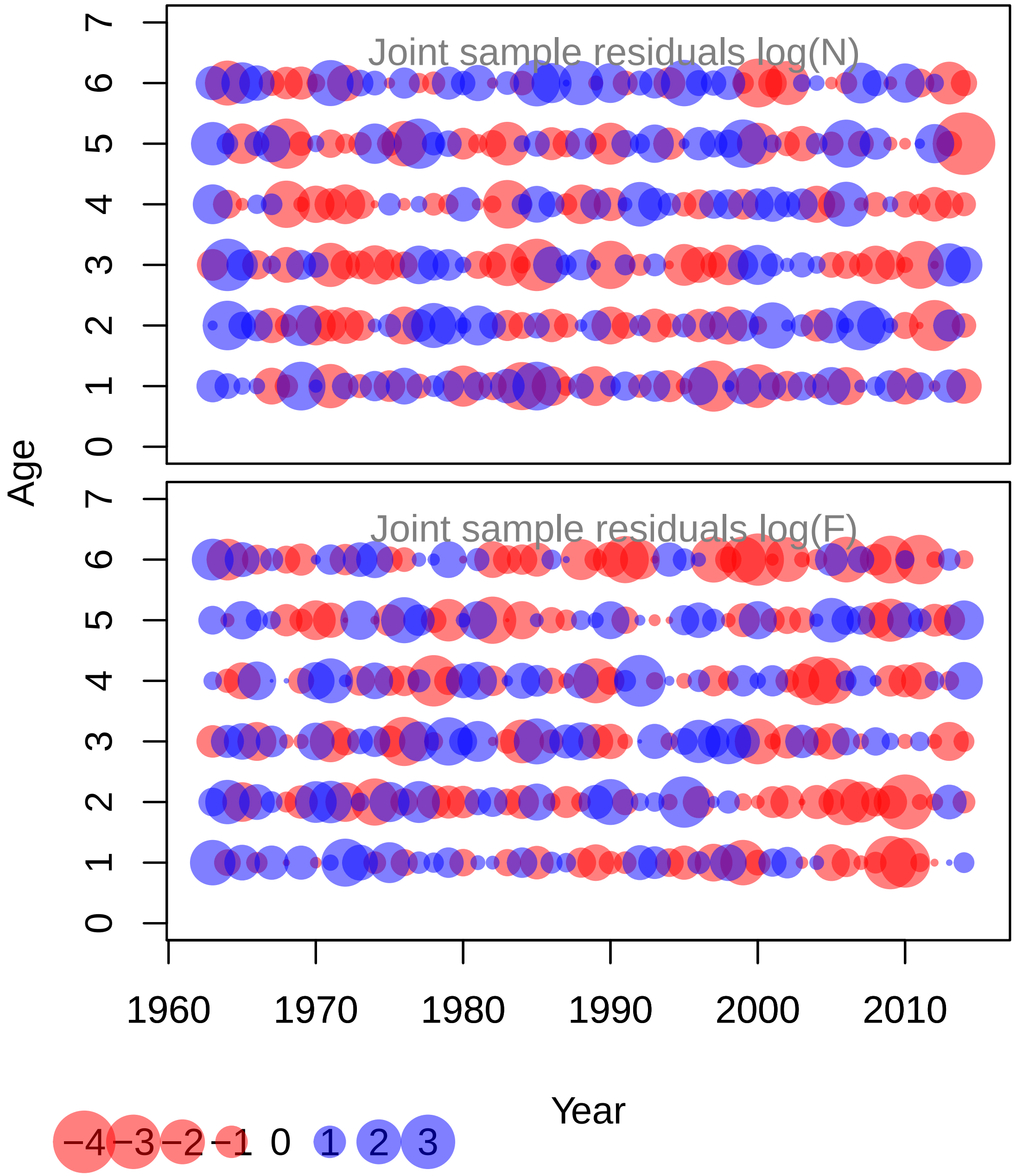


Right (joint sample)



Right. Model wrong




Joint sample residuals log(N)
Joint sample residuals log(F)
Age
Year
0
1
2
3
4
5
6
7
1960
1970
1980
1990
2000
2010
−4
−3
−2
−1
0
1
2
3

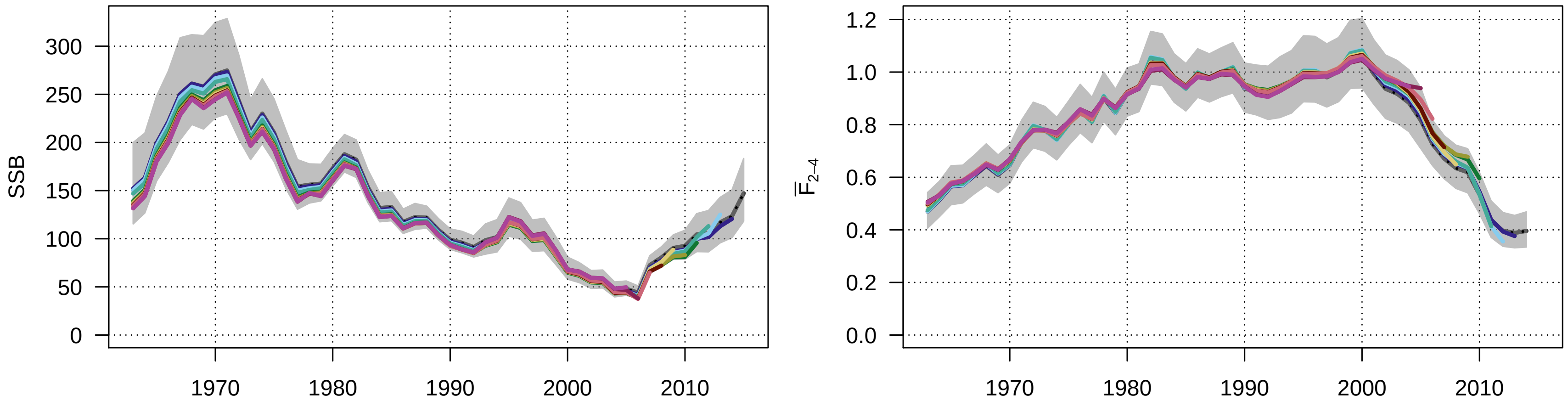
Code for the joint sample approach

```
sdr <- sdreport(obj)
estX <- summary(sdr,"random")
C <- solve(obj$env$spHess(obj$env$last.par.best, random=TRUE))
Xr <- MASS::mvrnorm(1,estX[,1],C)
```

Exercise: Calculate process residuals for the AR1-Poisson example. What distribution should we expect? Can we calculate quantities that we should expect are independent N(0,1)?

Retrospective pattern

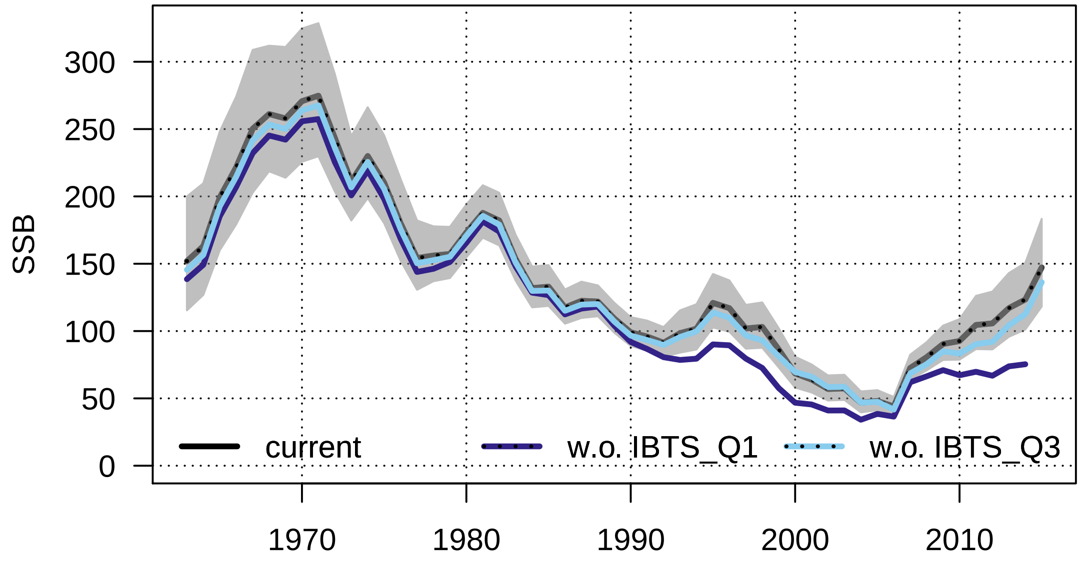
- Retrospective analysis (as done for fish stock assessments) are not predictions
- Possibly came about because prediction was not possible (within model)
- The procedure is:
  - Run model without last $1, 2, 3, \ldots, n$ years of data
  - Compare key estimates to model run with all data.

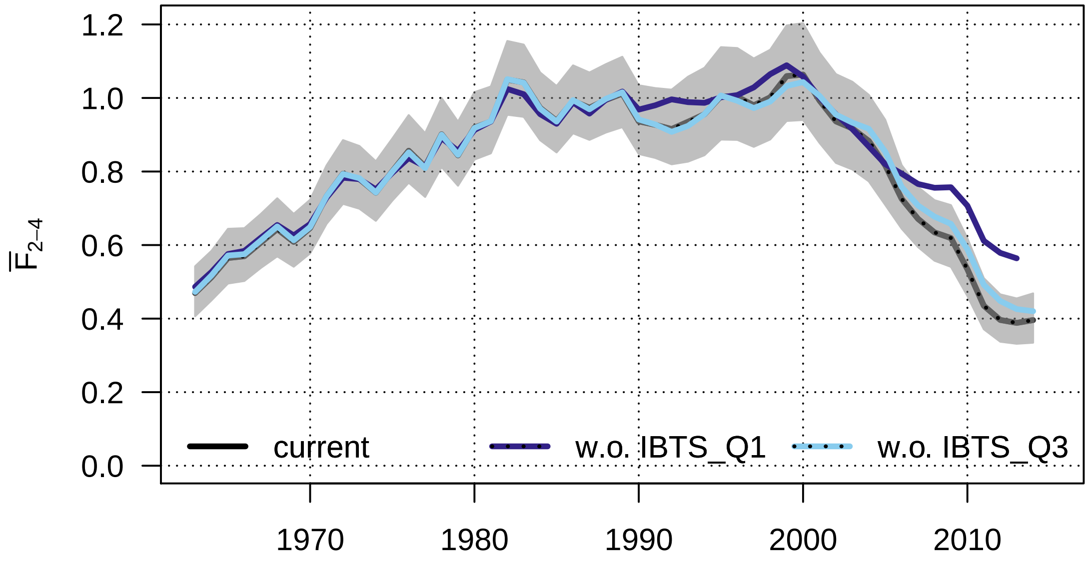


- Final year's estimate is of special interest in fish stock assessment applications
- Retrospective is still valuable (even if the models now can predict)
- Only relevant after model verified via other model diagnostics

Leave-out-fleet runs

- Leaving out individual data sources one at a time
- Useful to see if one fleet is having an undue influence





Jitter analysis

- Use a number of random (widely scattered) initial values
- Verify that the same solution is obtained

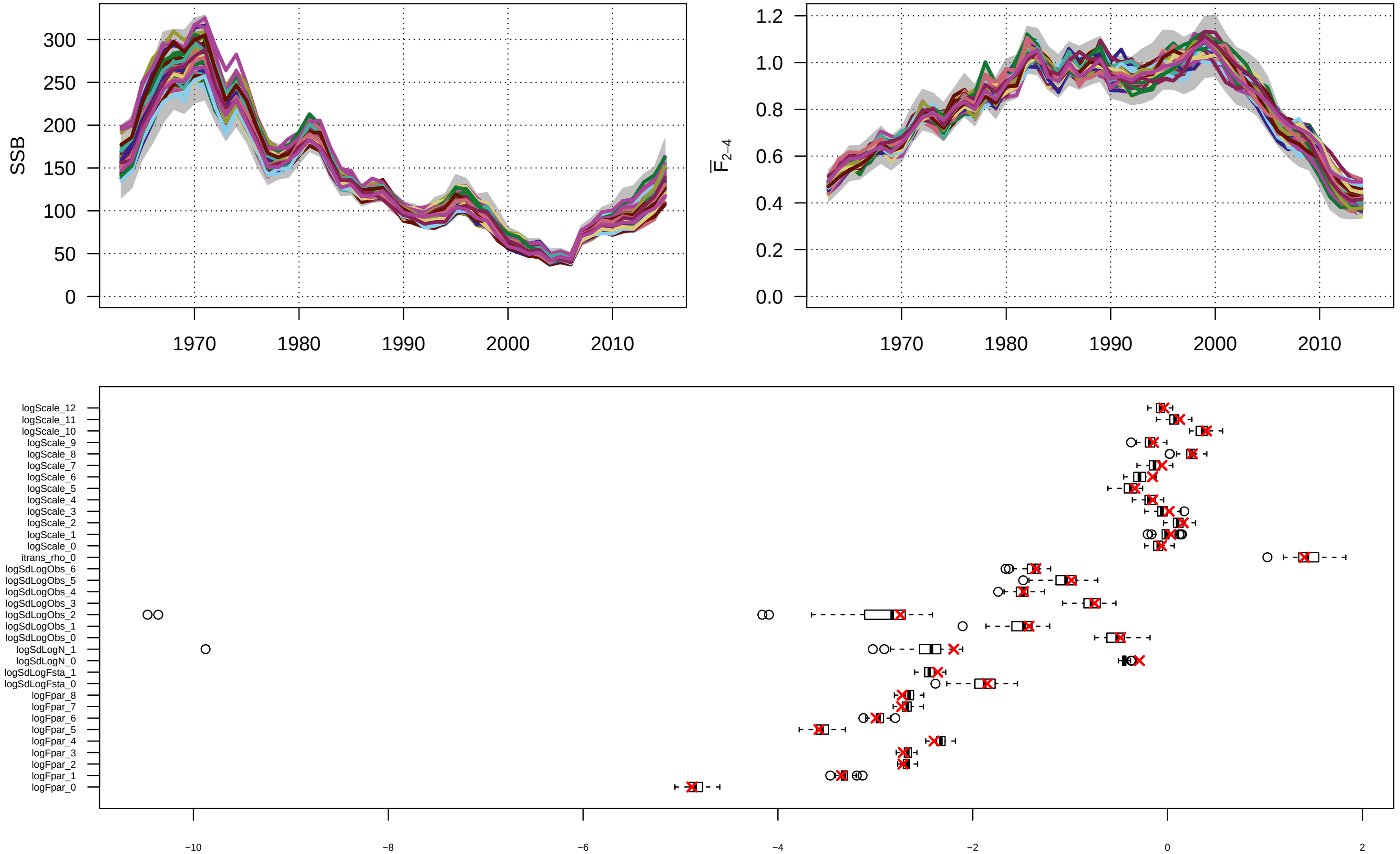
```
>  fit.jit <- jit(fit, nojit=100)
>  fit.jit
             max(|delta|)
logFpar      3.141487e-12
logSdLogFsta 4.359180e-12
logSdLogN    1.002443e-11
logSdLogObs  1.441203e-11
itrans_rho   3.332890e-11
logScale     2.292500e-12
logF         1.112670e-10
logN         8.915474e-11
ssb          5.637389e-09
fbar         5.945799e-12
rec          2.247407e-08
catch        2.957563e-09
logLik       2.924025e-10
```

- Possibly even more important for state-space models

Mini exercise: Implement jitter analysis for the Poisson-AR(1) model

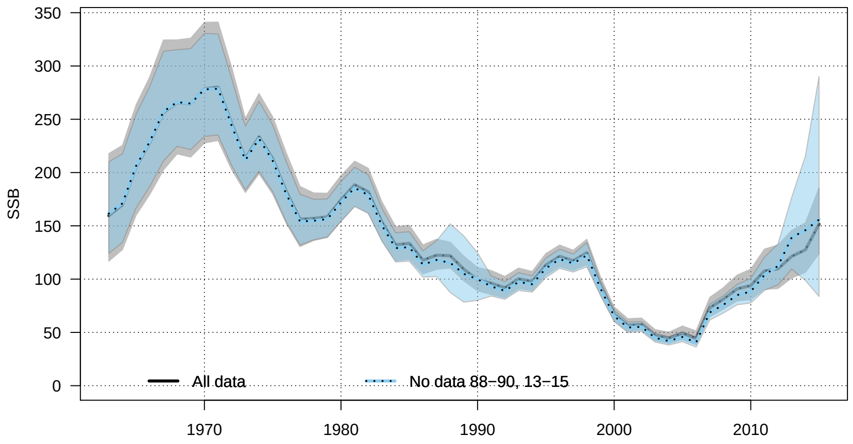
Simulation validation

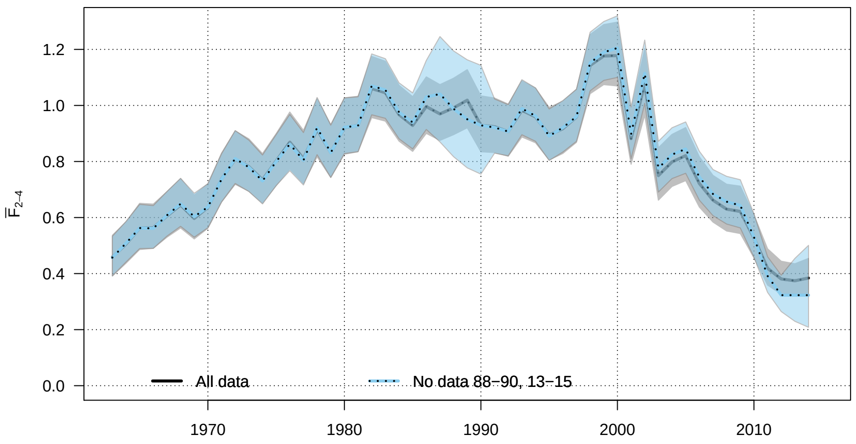
- Simulate from model and re-estimate (self test, parametric bootstrap)
- No (assessment) model should be accepted without it.
- Full versus conditional



Prediction and cross-validation

- Validate if the model is realistic w.r.t. coverage of confidence intervals
- Of special interest is the 2-3 year ahead predictions





- The only thing that is real is the observations
- when evaluating (and comparing) models we should look at their ability to predict observations.
- With state-space models we can (difficult to compare to other model types).

Check Laplace via simulation

- RTMB offers a very neat approach
- The expectation of the gradient of the negative log-likelihood is 0.

$$E_\theta \nabla \ell(\theta; X) = 0$$

- This means if we simulate from the model, then the average gradient should be zero.
- But this only holds for the real likelihood.
- So if the approximation is wrong, then the average gradient will not be zero
- We can simulate as many data sets as we wish, so we can test this.
- Notice: that even the smallest bias will be detected if we simulate enough
- Notice: Models with a modest bias can still be useful

```
$joint$p.value
[1] 0.4690289
...
$marginal$p.value
[1] 0.7745296
$marginal$bias
...
```

Appendix: The math for the Laplace checker

$$\begin{aligned}
E_\theta\left(\nabla\ell(\theta;X)\right) &= \int P_\theta(x)\nabla\ell(\theta;x)dx \\
&= -\int P_\theta(x)\frac{1}{P_\theta(x)}\nabla P_\theta(X)dx \\
&= -\nabla\int P_\theta(x)dx \\
&= 0
\end{aligned}$$