Simulation specifications for first round of jack mackerel assessment evaluations

The first step for establishing a simulation-estimation process for Chilean jack mackerel was to tune a statistical age-structured assessment model to example data that closely reflects actual data that may become available for this stock. Appendix A contains tables describing of the model used for tuning. The estimation model was written in ADMB (Anon . 2009) as was the simulator (included in entirety in and Appendix B).

The estimation model was fit (784 parameters total estimated) to original data (preliminary) Chilean jack mackerel data. Given these, the next step was to generate new data sets given the expected variability from observations (including autocorrelation in residuals). The results from the tuned model (that were subsequently carried forward as the "truth") are shown in Fig. 1. An excel-spreadsheet containing the values can be found on the site established for the workshop (http://tiny.cc/3fwq7).

References

Anon. 2009. AD Model Builder (developed by David Fournier and now freely available from *www.admb-project.org*) was used to implement the model.

Figures

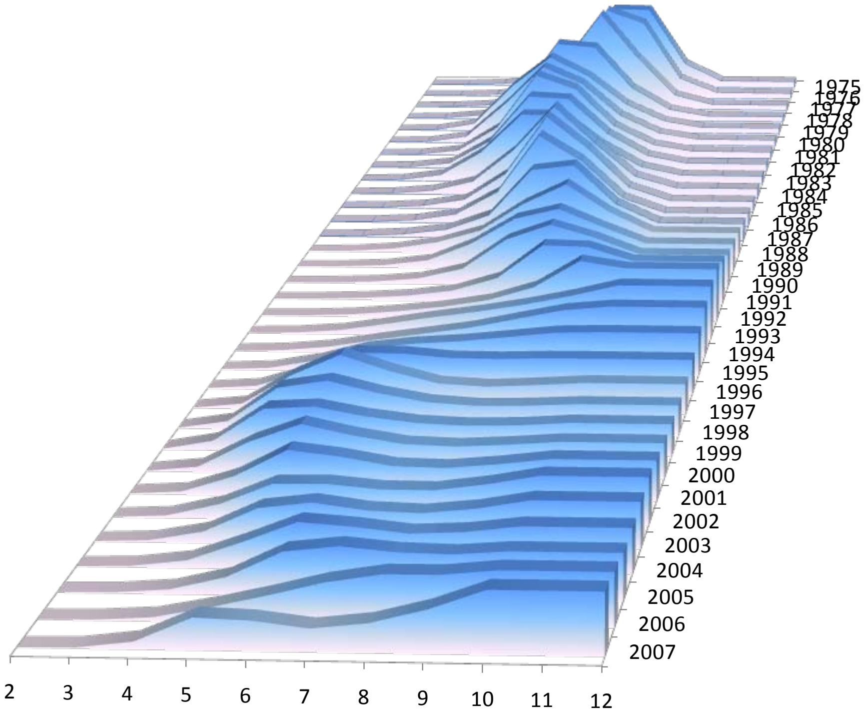

Figure 1. Actual age-selectivity pattern over time used in the simulations for Fishery 1.

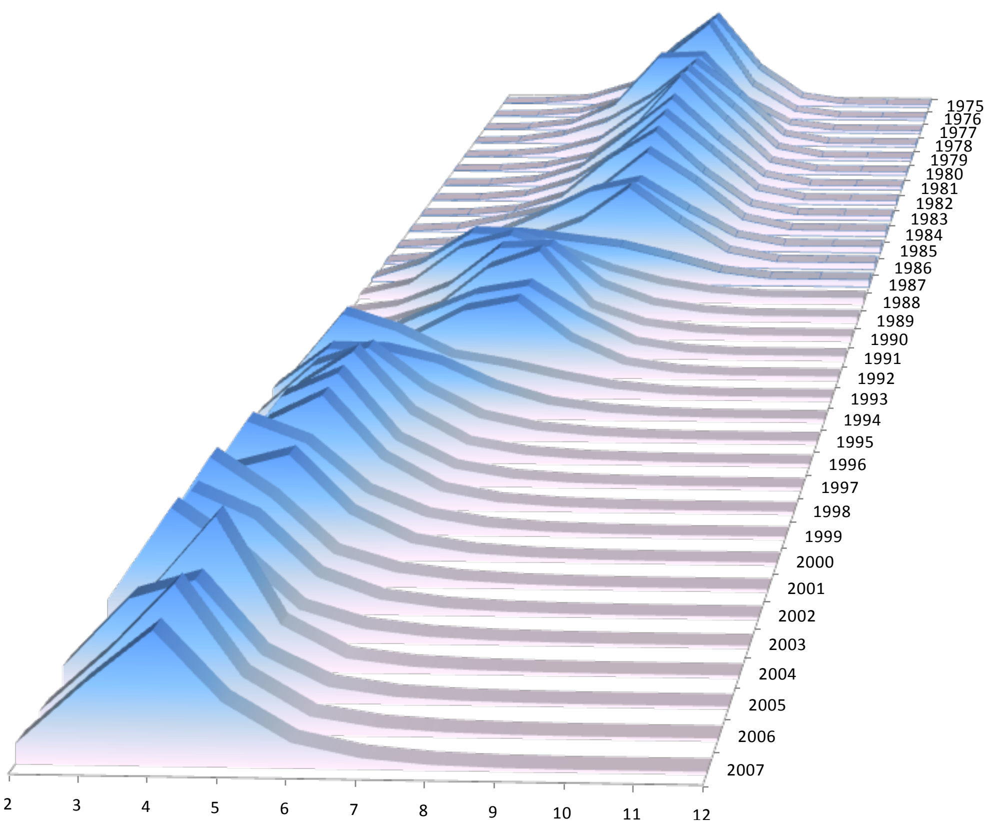

Figure 1 (cont'd). Actual age-selectivity pattern over time used in the simulations for Fishery 2.

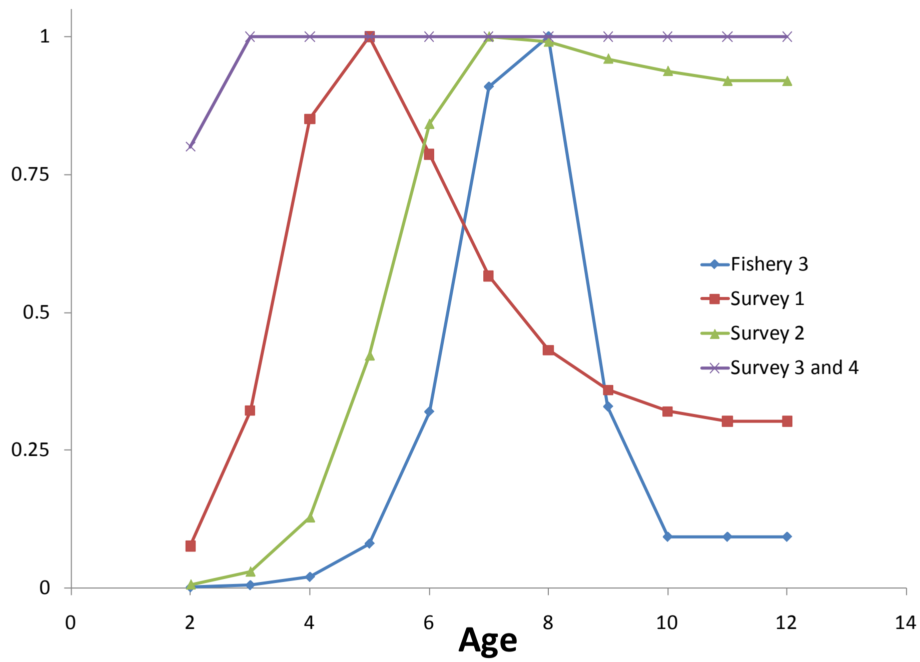

Figure 1 (cont'd). Actual age-selectivity pattern over time used in the simulations for Fishery 3 and the surveys (constant over time).

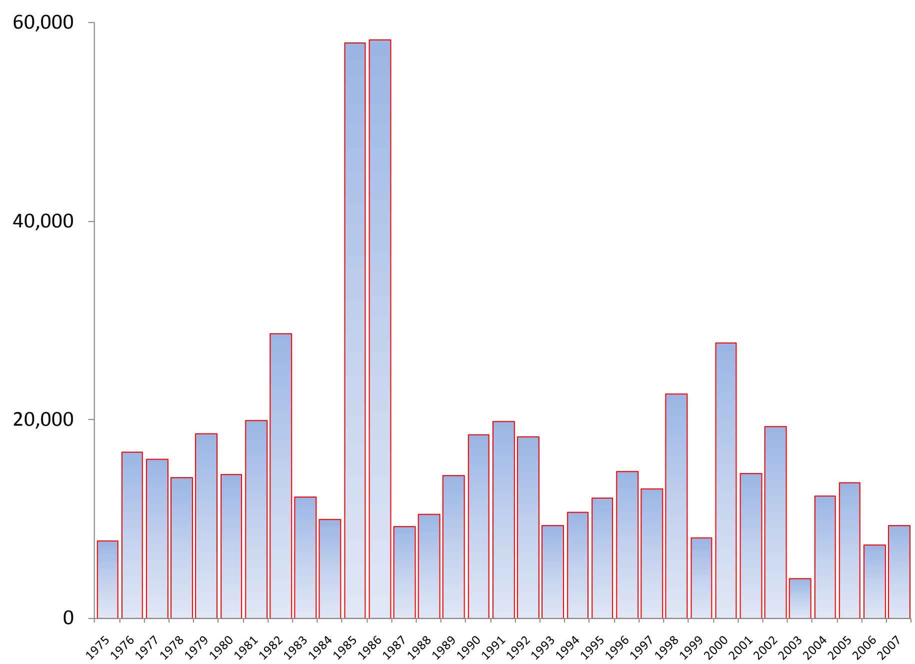

Figure 1 (cont'd). Actual age-2 recruitment assumed as "true" and used to generate simulated data.

Appendix A

Table A-1. Variable descriptions and model specification.

General Definitions	Symbol/Value	Use in Catch at Age Model
Year index: i = {1975, …., 2007}	i	
Age index: j = { 2, 3, …, 12^+}	j	
Mean weight by age j	W_j	
Maximum age beyond which selectivity is constant	*Maxage*	Selectivity parameterization
Instantaneous Natural Mortality	M	Fixed M=0.23, constant over all ages
Proportion females mature at age j	p_j	Definition of spawning biomass
Sample size for proportion at age j in year i	T_i	Scales multinomial assumption about estimates of proportion at age
Survey catchability coefficient	q^s	Prior distribution = lognormal(1.0 , σ_q^2)
Stock-recruitment parameters	R_0	Unfished equilibrium recruitment
	h	Stock-recruitment steepness
	σ_R^2	Recruitment variance
Age-error	**E**	Applied to age composition data to make "observed" values
Estimated parameters		
$\phi_i(26),\ R_0,\ h,\ \varepsilon_i(41), \sigma_R^2,\ \mu^f,\ \mu^s,\ M,\ \eta_j^s(39),\ \eta_j^f c(13),\ q^s(3)$		

Note that the number of selectivity parameters estimated depends on the model configuration.

Table A-2. Variables and equations describing implementation of the simulation model.

Description	Symbol/Constraints	Key Equation(s)
Survey abundance index (s) by year (survey mid-point taken to be on June 1st)	Y_i^s	$\hat{Y}_i^s = q_i^s \sum_{j=2}^{12^+} s_j^s W_{ij} e^{Z_{i,j}\frac{5}{12}} N_{ij}$
Catch biomass by year	C_i	$\hat{C}_i = \sum_j W_{ij} N_{ij} \frac{F_{ij}}{Z_{ij}}\left(1-e^{-Z_{ij}}\right)$
Proportion at age j, in year i	P_{ij}, $\sum_{j=2}^{12} P_{ij} = 1.0$	$P_{ij} = \frac{N_{ij}s_{ij}^f}{\sum_{k=2}^{12} N_{ik}s_{ik}^f}$
Initial numbers at age	$j = 1$	$N_{1975,2} = e^{\mu_R+\varepsilon_{1975}}$
	$2 < j < 11$	$N_{1975,j} = e^{\mu_R+\varepsilon_{1976-j}} \prod_{j=1}^{j} e^{-M}$
	$j = 12^+$	$N_{1975,12} = N_{1975,11}\left(1-e^{-M}\right)^{-1}$
Subsequent years ($i > 1963$)	$j = 2$	$N_{i,2} = e^{\mu_R+\varepsilon_i}$
	$2 < j \leq 11$	$N_{i,j} = N_{i-1,j-1} e^{-Z_{i-1,j-1}}$
	$j = 12^+$	$N_{i,12^+} = N_{i-1,11}\, e^{-Z_{i-1,10}} + N_{i-1,12}\, e^{-Z_{i-1,11}}$
Year effect, i = 1963, …, 2007	ε_i, $\sum_{i=1963}^{2007} \varepsilon_i = 0$	$N_{i,1} = e^{\mu_R+\varepsilon_i}$
Index catchability Mean effect	μ^s, μ^f	$q_i^s = e^{\mu^s}$
Age effect	η_j^s, $\sum_{j=2}^{12^+} \eta_j^s = 0$	$s_j^s = e^{\eta_j^s}$ $j \leq$ maxage $s_j^s = e^{\eta^s_{\text{maxage}}}$ $j >$ maxage
Instantaneous fishing mortality		$F_{ij} = e^{\mu_f+\eta_j^f+\phi_i}$
Mean fishing effect	μ_f	
annual effect of fishing in year i	ϕ_i, $\sum_{i=1975}^{2007} \phi_i = 0$	
age effect of fishing (regularized) In year time variation allowed	η_{ij}^f, $\sum_{j=2}^{12^+} \eta_{ij} = 0$	$s_{ij}^f = e^{\eta_j^f}$, $j \leq$ maxage $s_{ij}^f = e^{\eta^f_{\text{maxage}}}$ $j >$ maxage
In years where selectivity is constant over time	$\eta_{i,j}^f = \eta_{i-1,j}^f$	$i \neq$ change year
Natural Mortality	M	*Set fixed at 0.23*
Total mortality		$Z_{ij} = F_{ij} + M$
Spawning biomass	B_i	$B_i = \sum_{j=2}^{12} e^{-\frac{9.5}{12}Z_{ij}} W_j p_j$ *(note spawning taken to occur in October)*
Recruitment Beverton-Holt form	$\tilde{R}_i$	$\tilde{R}_i = \frac{\alpha B_i}{\beta + B_i}$, $\alpha = \frac{4hR_0}{5h-1}$ and $\beta = \frac{B_0(1-h)}{5h-1}$ where *h=0.8* $B_0 = \tilde{R}_0 \varphi$ $\varphi = \frac{e^{-12M}W_{12}p_{12}}{1-e^{-M}} + \sum_{j=2}^{12} e^{-M(j-1)} W_j p_j$

Table A-3. Specification of objective function that is minimized (i.e., the penalized negative of the log-likelihood).

Likelihood /penalty component		Description / notes
Abundance indices	$L_1 = \lambda_1 \sum_{s,i} \ln\left(Y_i^s \big/ \hat{Y}_i^s \right)^2 \frac{1}{2\sigma_i^{s\,2}}$	Survey abundances
Prior on smoothness for selectivities	$L_2 = \sum_l \lambda_2^l \sum_{j=2}^{12^+} \left(\eta_{j+2}^l + \eta_j^l - 2\eta_{j+1}^l \right)^2$	Smoothness (second differencing), Note: l={s, or f} for survey and fishery selectivity
Prior on recruitment regularity	$L_3 = \lambda_3 \sum_{i=1963}^{2007} \varepsilon_i^{\,2}$	Influences estimates where data are lacking (e.g., if no signal of recruitment strength is available, then the recruitment estimate will converge to median value).
Catch biomass likelihood	$L_4 = \lambda_4 \sum_{i=1975}^{2007} \ln\left(C_i / \hat{C}_i \right)^2$	Fit to catch biomass in each year (
Proportion at age likelihood	$L_5 = -\sum_{l,i,j} T_{ij}^l P_{ij}^l \ln\left(\hat{P}_{ij}^l \cdot P_{ij}^l \right)$	L={s, f} for survey and fishery age composition observations
Fishing mortality regularity	$L_6 = \lambda_6 \sum_{i=1975}^{2007} \phi_i^{\,2}$	(relaxed in final phases of estimation)
Priors	$L_7 = \left[\lambda_7 \frac{\ln\left(M / \hat{M} \right)^2}{2\sigma_M^2} + \lambda_8 \frac{\ln\left(q / \hat{q} \right)^2}{2\sigma_q^2} \right]$	Prior on natural mortality, and survey catchability (reference case assumption that these are precisely known at 0.3 and 1.0, respectively).
Overall objective function to be minimized	$\dot{L} = \sum_{i=1}^{7} L_i$	

Autocorrelation of residuals

For each abundance index used in the conditioning model, an empirical lag-1 correlation was computed from residuals in order to approximate similar patterns of residual errors for the simulation. This was considered a proxy for underlying processes (e.g., changes in availability to surveys or fishery CPUEs due to fish movement and progression of age classes). The random variability of surveys was thus generated to have similar pathologies to the actual data that were used to set up the simulated "true" model. The form of these residuals was (in the exponent):

$$e_{k,i} = r_k\, e_{k,i} + \sqrt{1 - r_k^{\,2}}\; n_{k,i}$$

where $n_{k,i}$ are unit normal random variables and k indexes the survey and r_k is the estimated lag-1 correlation coefficient for series k.

Appendix 2. Simulation code

The following ADMB code generated the simulated datasets. Values in bold face represent variables arising from the point estimates from the tuned model.

```
int nsims;
// get the number of simulated datasets to create...
ifstream sim_in("nsims.dat"); sim_in >> nsims; sim_in.close();
char buffer [33];
// compute the autocorrelation term for residuals of fit to indices...
for (k=1;k<=nsrv;k++)
  ac(k) = get_AC(k); //function that computes the correlation coefficient for index k
for (int isim=1;isim<=nsims;isim++)
{
  // Create the name of the simulated dataset
  simname = "sim_"+ adstring(itoa(isim,buffer,10)) + ".dat";
  // Open the simulated dataset for writing (an object called "simdat")
  ofstream simdat(simname);
  simdat << "# first year" <<endl;
  simdat << styr <<endl;
  simdat << "# Last  year" <<endl;
  simdat << endyr <<endl;
  simdat << "# age recruit" <<endl;
  simdat << rec_age <<endl;
  simdat << "# oldest age" <<endl;
  simdat << oldest_age <<endl;
  simdat << "# Number of fisheries " <<endl;
  simdat << nfsh <<endl;
  simdat << fshnameread <<endl;
  simdat << "# Catch biomass by fishery " <<endl;
  for (k=1;k<=nfsh;k++)
  {
    simdat << "# " <<fshname(k) <<" " << k <<endl;
    simdat << catch_bio(k) <<endl;
  }
  simdat << "# Catch biomass uncertainty by fishery (std errors)" <<endl;
  for (k=1;k<=nfsh;k++)
  {
    simdat << "# " <<fshname(k) <<" " << k <<endl;
    simdat << catch_bio_sd(k) <<endl;
  }
  simdat << "# number of years for fishery age data " <<endl;
  for (k=1;k<=nfsh;k++)
  {
    simdat << "# " <<fshname(k)<< " " << k <<endl;
    simdat << nyrs_fsh_age(k) <<endl;
  }
  simdat << "# years for fishery age data " <<endl;
  for (k=1;k<=nfsh;k++)
  {
    simdat << "# " <<fshname(k)<< " " << k <<endl;
    simdat << yrs_fsh_age(k)  <<endl;
  }
  simdat << "# sample sizes for fishery age data " <<endl;
  for (k=1;k<=nfsh;k++)
  {
    simdat << "# " <<fshname(k)<< " " << k <<endl;
    simdat << nsmpl_fsh(k)  <<endl;
  }
  simdat << "# Observed age compositions for fishery" <<endl;
  for (k=1;k<=nfsh;k++)
  {
    dvector p(1,nages);
    double Ctmp; // total catch
    dvector freq(1,nages);
    simdat << "# " << fshname(k) <<endl;
    for (i=1;i<=nyrs_fsh_age(k);i++)
    {
      int iyr = yrs_fsh_age(k,i);
      // Add noise here
      freq.initialize();
      ivector bin(1,nsmpl_fsh(k,i));
      p = value(catage(k,iyr));
      p /= sum(p);
      bin.fill_multinomial(rng,p);
      for (int j=1;j<=nsmpl_fsh(k,i);j++)
        freq(bin(j))++;
      // Apply ageing error to samples.............
      p = age_err *freq/sum(freq);
      simdat << p  <<endl;
```

```
    // Compute total catch given this sample size
    Ctmp = catch_bio(k,iyr) / (p*wt_fsh(k,iyr));
    // Simulated catage = proportion sampled
    sim_catage(k,i) = p * Ctmp;
  }
}
simdat << "# Annual wt-at-age for fishery" <<endl;
for (k=1;k<=nfsh;k++)
{
  simdat << "# " <<fshname(k)<< " " << (k) <<endl;
  simdat << wt_fsh(k)  <<endl;
}
simdat << "# number of indices" <<endl;
simdat << nsrv <<endl;
simdat << srvnameread <<endl;
simdat << "# Number of years of index values (annual)" <<endl;
for (k=1;k<=nsrv;k++)
{
  simdat << "# " << srvname(k) <<endl;
  simdat << nyrs_srv(k)  <<endl;
}
simdat << "# Years of index values (annual)" <<endl;
for (k=1;k<=nsrv;k++)
{
  simdat << "# " << srvname(k) <<endl;
  simdat << yrs_srv(k)  <<endl;
}
simdat << "# Month that index occurs "<<endl;
for (k=1;k<=nsrv;k++)
{
  simdat << "# " << srvname(k) <<endl;
  simdat << mo_srv(k) <<endl;
}
simdat << "# values for indices (annual)"<<endl;
dmatrix new_srv(1,nsrv,1,nyrs_srv);
new_srv.initialize();
for (k=1;k<=nsrv;k++)
{
  simdat << "# " <<srvname(k)<< " " << k <<endl;
  // Add noise here
  dvector ran_srv_vect(1,nyrs_srv(k));
  // fill vector with unit normal RVs
  ran_srv_vect.fill_randn(rng);
  // do first year uncorrelated
  i=1;
  int iyr = yrs_srv(k,i);
  corr_dev(k)  = ran_srv_vect;
  new_srv(k,i) = mfexp(corr_dev(k,i) * obs_lse_srv(k,i) ) *
                 value(elem_prod(wt_srv(k,iyr),elem_prod(pow(S(iyr),srv_month_frac(k)), natage(iyr)))*
                 q_srv(k)*sel_srv(k,iyr));
  // do next years correlated with previous
  for (i=2;i<=nyrs_srv(k);i++)
  {
    iyr=yrs_srv(k,i);
    corr_dev(k,i) = ac(k) * corr_dev(k,i-1) + sqrt(1.-square(ac(k))) * corr_dev(k,i);
    new_srv(k,i) = mfexp(corr_dev(k,i) * obs_lse_srv(k,i) ) *
                   value(elem_prod(wt_srv(k,iyr),elem_prod(pow(S(iyr),srv_month_frac(k)),
                   natage(iyr))) * q_srv(k)*sel_srv(k,iyr));
  }
  simdat << new_srv(k)      <<endl;
}
simdat << "# standard errors for indices (by year) " <<endl;
for (k=1;k<=nsrv;k++)
{
  simdat << "# " <<srvname(k)<< " " << k <<endl;
  simdat << obs_se_srv(k)  <<endl;
}
simdat << "# Number of years of age data available for index" <<endl;
for (k=1;k<=nsrv;k++)
{
  simdat << "# " <<srvname(k)<< " " << k <<endl;
  simdat << nyrs_srv_age(k)  <<endl;
}
simdat << "# Years of index values (annual)" <<endl;
for (k=1;k<=nsrv;k++)
{
  simdat << "# " <<srvname(k)<< endl;
  simdat << yrs_srv_age(k)  <<endl;
}
simdat << "# Sample sizes for age data from indices" <<endl;
for (k=1;k<=nsrv;k++)
```

```
  {
    simdat << "# " <<srvname(k)<< endl;
    simdat << nsmpl_srv(k)  <<endl;
  }
  simdat << "# values of proportions at age in index" <<endl;
  for (k=1;k<=nsrv;k++)
  {
    simdat << "# " <<srvname(k)<< endl;
    dvector p(1,nages);
    dvector freq(1,nages);
    for (i=1;i<=nyrs_srv_age(k);i++)
    {
      int iyr = yrs_srv_age(k,i);
      // Add noise here
      freq.initialize();
      ivector bin(1,nsmpl_srv(k,i));
      p = age_err * value(elem_prod( elem_prod(pow(S(iyr),srv_month_frac(k)),
          natage(iyr))*q_srv(k) , sel_srv(k,iyr)));
      p /= sum(p);
      // fill vector with multinomial samples
      bin.fill_multinomial(rng,p);
      for (int j=1;j<=nsmpl_srv(k,i);j++)
        freq(bin(j))++;
      simdat << "# " <<srvname(k)<< " year: "<< iyr<< endl;
      simdat << freq/sum(freq) <<endl;
    }
  }
  simdat << "# Mean wts at age for indices" <<endl;
  for (k=1;k<=nsrv;k++)
  {
    simdat << "# " <<srvname(k)<< endl;
    simdat <<  wt_srv(k)  <<endl;
  }

  simdat << "# Population mean wt at age" <<endl;
  simdat << wt_pop <<endl;

  simdat << "# Population maturity at age" <<endl;
  simdat << maturity  <<endl;

  simdat << "# Peak spawning month" <<endl;
  simdat << spawnmo <<endl;

  simdat << "# ageing error " <<endl;
  simdat << age_err <<endl;

  simdat <<endl<<endl<<"Additional output"<<endl;
  simdat << "# Fishery_Effort " <<endl;
  for (k=1;k<=nfsh;k++)
  {
    dvector ran_fsh_vect(styr,endyr);
    // fill vector with unit normal RVs
    ran_fsh_vect.fill_randn(rng);
    // Sigma on effort is ~15% white noise (add red noise later)
    ran_fsh_vect *= 0.15;
    dvector avail_biom(styr,endyr);
    for (i=styr;i<=endyr;i++)
    {
      avail_biom(i) = wt_fsh(k,i)*value(elem_prod(natage(i),sel_fsh(k,i)));
    }
    act_eff(k) = elem_prod(exp(ran_fsh_vect), (elem_div(catch_bio(k), avail_biom)) );
    // Normalize effort
    act_eff(k) /= mean(act_eff(k));
    for (i=styr;i<=endyr;i++)
      simdat<<fshname(k)<<" "<<i<<" "<<act_eff(k,i) <<endl;
  }
  simdat << "# Fishery catch-at-age " <<endl;
  for (k=1;k<=nfsh;k++)
  {
    simdat << "# " <<fshname(k)<< " " << k <<endl;
    simdat << "Fishery Year "<<age_vector << endl;
    for (i=1;i<=nyrs_fsh_age(k);i++)
      simdat<<fshname(k)<<" "<<yrs_fsh_age(k,i)<<" "<<sim_catage(k,i) <<endl;
  }
}
exit(1);
// End of simulating datasets..................
```