Appendix A7. AD Model Builder Code for the Striped Bass Statistical Catch-At-Age Model

```
//--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--
//
//      Striped bass Statistical Catch-At-Age Model
//      Gary Nelson
//      Massachusetts Division of Marine Fisheries
//      Gloucester, MA 01930
//
//  Some Code Adapted from Erik H. Williams' Menhaden Statistical Catch-At-Age Model
//
//
//
//--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--><>--
TOP_OF_MAIN_SECTION
 arrmblsize=1000000;

DATA_SECTION
 !! ad_comm::change_datafile_name("scamdata.dat");

// STARTING AND ENDING YEAR OF MODEL
 init_int styr;
 init_int endyr;

// NUMBER OF AGES
 init_int nages;

// VECTOR OF AGES FOR AGE BINS
 init_ivector agebins(1,nages);

//CALCULATE RECRUITMENT YEARS
 int styrR;
 LOCAL_CALCS
  styrR=styr-(nages-1);
 END_CALCS

//TOTAL CATCH CVs
 init_vector total_catch_CV(styr,endyr);

//TOTAL CATCH LIKELIHOOD WEIGHTS
 init_number l_wgt;

//REC & COM HARVEST AND RELEASE AND DISCARDS MORTALITY NUMBERS
 init_vector obs_total_catch(styr,endyr);

//CATCH AGE COMPOSITION
 init_vector ss_age_comp(styr,endyr);
 init_number caa_wgt;
 init_matrix obs_age_comp(styr,endyr,1,nages);

// YOY AND YEARLING SURVEYS
 init_int age_surv_num;
 init_vector age_surv_flag(1,age_surv_num);
 init_vector age_surv_ages(1,age_surv_num);
 init_vector yoy_wgt(1,age_surv_num);
 init_matrix age_surv_CV(styrR,endyr,1,age_surv_num);
 init_matrix age_obs_surv_indices(styrR,endyr,1,age_surv_num);

// AGGREGATE SURVEYS
 init_int agg_surv_num;
 init_vector agg_surv_flag(1,agg_surv_num);
 init_vector agg_surv_ages(1,agg_surv_num);
 init_vector agg_wgt(1,agg_surv_num);
 init_matrix agg_surv_CV(styrR,endyr,1,agg_surv_num);
 init_matrix agg_obs_surv_indices(styrR,endyr,1,agg_surv_num);

//SURVEYS WITH AGE COMPOSITION
 init_int ac_surv_num;
 init_vector ac_surv_flag(1,ac_surv_num);
 init_vector ac_surv_wgt(1,ac_surv_num);
 init_vector ac_age_wgt(1,ac_surv_num);
 init_matrix ac_surv_CV(styrR,endyr,1,ac_surv_num);
```

```
 init_matrix ac_obs_surv_indices(styrR,endyr,1,ac_surv_num);

//SAMPLE SIZES OF SURVEY AGE COMPOSITIONS
 init_matrix ac_ss(styrR,endyr,1,ac_surv_num);

//SURVEY AGE COMPOSITION IN PROPORTIONS
 init_3darray surv_comps(1,ac_surv_num,styrR,endyr,1,nages);

//SPAWNING STOCK WEIGHTS-AT-AGE
 init_matrix ssw(styr,endyr,1,nages);
 init_vector mat(1,nages);
 init_number pM;
 init_number pF;

//INPUT CONSTANT M
 init_number M;
 init_number R_lam;
 init_number F_lam;

 int cnt;
 int y;
 int a;
 int t;
 int realage;
 int d;
 int total;
 int n_parms;
 int df;

 LOCAL_CALCS
  n_parms=1+(endyr-styrR+1)+1+(endyr-styr+1)+2+2+2+2+2+2+2+1+age_surv_num+agg_surv_num+ac_surv_num+1;
  df=n_parms+(endyr-styr+1)+(endyr-styrR+1)+age_surv_num+agg_surv_num+ac_surv_num;
 END_CALCS
 matrix sigma(1,df,1,df+1);
 !! set_covariance_matrix(sigma);

PARAMETER_SECTION
//TEMPORARY VARIABLES
 number adds;
 number pgroup;
 number diff;
 number diff2;
 number sel;
 number aveN;
 number sump;
 number sumage;
 number maxs;
 number dodo;
 number dodo1;
 number sumdo;
 number sumdo1;
 number fpen;

//---------------------INITIATE SCAM ARRAYS--------------------------//
//AVERAGE RECRUITMENT
 init_number log_avg_R(1);

//RECRUITMENT DEVIATIONS
 init_bounded_dev_vector log_R_dev(styrR,endyr,-20.,20.,3);//Age 1 recruitment values from styr to endyr

//AVERAGE FISHING MORTALITY
 init_number log_avg_F(2);

//FISHING MORTALITY DEVIATIONS
 init_bounded_dev_vector log_F_dev(styr,endyr,-15.,15.,2);//

//NUMBERS,F,Z MATRICES
 matrix N(styrR,endyr,1,nages);//Population numbers by year and age
 matrix F(styr,endyr,1,nages);
 matrix Z(styrR,endyr,1,nages);

//CATCH SELECTIVITIES
 init_bounded_number p1_A50(0,150,4);
 init_bounded_number p1_slope(0,150,4);
 init_bounded_number p2_A50(0,150,4);
 init_bounded_number p2_slope(0,150,4);
 init_bounded_number p3_A50(0,150,4);
 init_bounded_number p3_slope(0,150,4);
 init_bounded_number p4_A50(0,150,4);
 init_bounded_number p4_slope(0,150,4);
```

```
 vector p1_sel(1,nages);
 vector p2_sel(1,nages);
 vector p3_sel(1,nages);
 vector p4_sel(1,nages);

//SURVEY SELECTIVITIES
 init_bounded_vector DE_surv(1,2,0,150,9);
 init_number MD_surv(10);
 init_vector NY_surv(1,2,7);
 init_bounded_number NY_e(1e-22,0.9999,7)
 init_vector NJ_surv(1,2,8);
 matrix surv_sel(1,nages,1,ac_surv_num);

//STARTING VALUES FOR SURVEY SELECTIVITY PARAMETERS
 LOCAL_CALCS
  NY_e=0.95;
  NY_surv(1)=-1;
  NY_surv(2)=1;
  NJ_surv(1)=3;
  NJ_surv(2)=1;
  MD_surv=0.3;
  DE_surv(1)=3;
  DE_surv(2)=1;
 END_CALCS

//SURVEY CATCHABILITY COEEFFICIENTS AND PREDICTED INDICES
 init_bounded_vector age_qs(1,age_surv_num,-50.,0.,5);
 matrix age_pred_surv_indices(styrR,endyr,1,age_surv_num);
 init_bounded_vector agg_qs(1,agg_surv_num,-50.,0.,5);
 matrix agg_pred_surv_indices(styrR,endyr,1,agg_surv_num);
 init_bounded_vector ac_qs(1,ac_surv_num,-50.,0.,6);
 matrix ac_pred_surv_indices(styrR,endyr,1,ac_surv_num);

//PREDICTED SURVEY AGE COMPOSITIONS
 3darray calc_comps(1,ac_surv_num,styrR,endyr,1,nages);
 3darray surv_pred_comps(1,ac_surv_num,styrR,endyr,1,nages);

//INDIVIDUAL LIKELIHOOD SAVE VECTORS
 vector like_age(1,age_surv_num);
 vector like_agg(1,agg_surv_num);
 vector like_ac_surv(1,ac_surv_num);
 vector like_ac_age(1,ac_surv_num);

//CATCH-AT-AGE,PREDICTED TOTAL CATCH, PREDICTED CATCH AGE COMPOSITION, AND SSB
 matrix C(styr,endyr,1,nages);
 vector pred_total_catch(styr,endyr);
 matrix pred_age_comp(styr,endyr,1,nages);
 number f_total_catch;
 number f_age_comp;
 matrix SSB(styr,endyr,1,nages);
 matrix rwgts(styr,endyr,1,nages);
 matrix W2(styr,endyr,1,nages);
 matrix jan1bio(styr,endyr,1,nages);
 matrix catchbio(styr,endyr,1,nages);
 vector tSSB(styr,endyr);

//REPORT STANDARD DEVIATIONS FOR ANNUAL FS,RS, AND CATCHABILITY COEFFICIENTS
 sdreport_vector F_ann(styr,endyr);
 sdreport_vector R(styrR,endyr);
 sdreport_vector q_YOY(1,age_surv_num);
 sdreport_vector q_Agg(1,agg_surv_num);
 sdreport_vector q_AC(1,ac_surv_num)

 objective_function_value f;

INITIALIZATION_SECTION
 //STARTING VALUES FOR REMAINING PARAMETERS
 log_avg_R 10.6;
 log_avg_F -2.6;
 p1_A50 3;
 p1_slope 1;
 p2_A50 3;
 p2_slope 1;
 p3_A50  3;
 p3_slope 1;
 p4_A50 3;
 p4_slope 1;
 age_qs -20.4;
 agg_qs -19.7;
 ac_qs  -20.2;
```

```
RUNTIME_SECTION
 maximum_function_evaluations 10000, 10000, 10000;
 convergence_criteria 1e-5, 1e-7, 1e-16;

PRELIMINARY_CALCS_SECTION
 F.initialize();
 C.initialize();
 calc_comps.initialize();

PROCEDURE_SECTION
 calc_selectivity();
 calc_F_mortality();
 calc_Z_matrix();
 calc_numbers_at_age();
 calc_catch_at_age();
 calc_pred_age_comp();
 calc_indices_selectivity();
 calc_predict_indices_age();
 calc_predict_indices_agg();
 calc_predict_indices_ac();
 scam_likelihood();
 calc_biomass();
 evaluate_the_objective_function();

//CALCULATE CATCH SELECTIVITIES VALUES FOR CURRENT PARAMETER ESTIMATES
FUNCTION calc_selectivity //gompertz function
 for (a=1;a<nages;a++){
   p1_sel(a)=exp(-1.*exp(-1.*p1_slope*(double(agebins(a))-p1_A50)));
   p2_sel(a)=exp(-1.*exp(-1.*p2_slope*(double(agebins(a))-p2_A50)));
   p3_sel(a)=exp(-1.*exp(-1.*p3_slope*(double(agebins(a))-p3_A50)));
   p4_sel(a)=exp(-1.*exp(-1.*p4_slope*(double(agebins(a))-p4_A50)));
 }
 p1_sel(nages)=p1_sel(nages-1);
 p2_sel(nages)=p2_sel(nages-1);
 p3_sel(nages)=p3_sel(nages-1);
 p4_sel(nages)=p4_sel(nages-1);

 p1_sel=p1_sel/max(p1_sel);
 p2_sel=p2_sel/max(p2_sel);
 p3_sel=p3_sel/max(p3_sel);
 p4_sel=p4_sel/max(p4_sel);

//MATCH PERIOD SELECTVITIES TO YEARS AND CALCULATE ANNUAL F AND F-AT-AGE
FUNCTION calc_F_mortality
 for(y=styr;y<=endyr;y++){
   for(a=1;a<=nages;a++){
     if (y<1985) sel=p1_sel(a);
     if (y>=1985 && y<=1989) sel=p2_sel(a);
     if (y>=1990 && y<=1995) sel=p3_sel(a);
     if (y>=1996) sel=p4_sel(a);
     F(y,a)=sel*mfexp(log_avg_F+log_F_dev(y));
     F_ann(y)=mfexp(log_avg_F+log_F_dev(y));
   }
 }

//FILL Z MATRIX
FUNCTION calc_Z_matrix
 for(y=styrR;y<=endyr;y++){
   for(a=1;a<=nages;a++){
     if(y<styr)Z(y,a)=F(styr,a)+M;
     if(y>=styr)Z(y,a)=F(y,a)+M;
   }
 }

//CALCULATE AND FILL NUMBERS-AT-AGE MATRIX
FUNCTION calc_numbers_at_age
 N(styrR,1)=mfexp(log_avg_R+log_R_dev(styrR));//Fill in Recruits in first year and age

 for(a=2;a<=nages;a++){
   N(styrR,a)=N(styrR,a-1)*mfexp(-1.*Z(styrR,a-1));//Fills in top row of matrix
 }
 N(styrR,nages)=N(styrR,nages-1)*mfexp(-1.*Z(styrR,nages-1))/(1.-mfexp(-1.*Z(styrR,nages)));

 for(y=styrR+1;y<=endyr;y++){  //Rest of pre-data years
   N(y,1)=mfexp(log_avg_R+log_R_dev(y));
   N(y)(2,nages)=++elem_prod(N(y-1)(1,nages-1),(mfexp(-1.*Z(y-1)(1,nages-1))));
   N(y,nages)+=N(y-1,nages)*mfexp(-1.*Z(y-1,nages));//plus  group
 }

 for(y=styrR;y<=endyr;y++){
   R(y)=mfexp(log_avg_R+log_R_dev(y));
```

```
 }

FUNCTION calc_biomass
 //Rivard weights

 for(a=2;a<=nages-1;a++){
   for(y=styr+1;y<=endyr;y++){
      W2(y,a)=(log(ssw(y,a))+log(ssw(y-1,a-1)))/2;
     }
    }
 for(y=styr;y<=endyr-1;y++){
      W2(y,1)=2.*log(ssw(y,1))-W2(y+1,2);
     }
 for(a=1;a<=nages-2;a++){
      W2(styr,a)=2.*log(ssw(styr,a))-W2(styr+1,a+1);
    }
 W2(styr,nages-1)=(W2(styr,nages-1)+W2(styr,nages-2))/2;
 W2(endyr,1)=2.*log(ssw(endyr,1))-W2(endyr,2);
 for(y=styr;y<=endyr;y++){
      W2(y,nages)=log(ssw(y,nages));
     }
 for(y=styr;y<=endyr;y++){
      for(a=1;a<=nages;a++){
      rwgts(y,a)=exp(W2(y,a));
      jan1bio(y,a)=rwgts(y,a)*N(y,a);
      catchbio(y,a)=ssw(y,a)*obs_total_catch(y)*obs_age_comp(y,a);
     }
    }
 for(y=styr;y<=endyr;y++){
   for(a=1;a<=nages;a++){
     SSB(y,a)=(N(y,a)*rwgts(y,a)*mat(a)*mfexp(-1*(pF*F(y,a)+pM*M)))/2;
   }
 }
 tSSB=rowsum(SSB);

//CALCULATE CATCH-AT-AGE MATRIX
FUNCTION calc_catch_at_age
 for(y=styr;y<=endyr;y++){
   for(a=1;a<=nages;a++){
     C(y,a)=N(y,a)*F(y,a)*(1.-mfexp(-1.*Z(y,a)))/Z(y,a);
   }
 }

//CALCULATE PREDICTED CATCH AGE COMPOSITION
FUNCTION calc_pred_age_comp
 for(y=styr;y<=endyr;y++){
   sumage=0;
   for(a=1;a<=nages;a++){
     sumage+=C(y,a);
   }
   pred_total_catch(y)=sumage;

   for(a=1;a<=nages;a++){
     pred_age_comp(y,a)=C(y,a)/(sumage+0.001);
   }
 }

//CALCULATE SURVEY SELECTIVITY INDICES
FUNCTION calc_indices_selectivity //NYOHS NJTRL MDAdults DESSN
 for(int s=1;s<=ac_surv_num;s++){
   maxs=0;
   for(a=1;a<nages;a++){
     surv_sel(a,s)=0;
     if(s==1){
       if(a>=2) surv_sel(a,s)=(1/(1-NY_e))*pow((1-NY_e)/NY_e,NY_e)*(exp(NY_surv(1)*NY_e*(NY_surv(2)-
double(a)))/(1+exp(NY_surv(1)*(NY_surv(2)-double(a)))));
     }

     if(s==2){
         if(a>=2) surv_sel(a,s)=pow(double(a),NJ_surv(1))*exp(-1.*NJ_surv(2)*double(a));
       }
     if(s==3){
       if (a==2) surv_sel(a,s)=MD_surv;
       if (a>=3) surv_sel(a,s)=1;
     }

     if(s==4){
       if(a>=2) surv_sel(a,s)=exp(-1.*exp(-1.*DE_surv(2)*(double(agebins(a))-DE_surv(1))));
      }
     if(surv_sel(a,s)>=maxs) maxs=surv_sel(a,s);
   }
```

```
   for(a=1;a<nages;a++){
     surv_sel(a,s)=surv_sel(a,s)/maxs;
     }
   surv_sel(nages,s)=surv_sel(nages-1,s);
 }


//CALCULATE PREDICTED YOY AND YEARLING INDICES
FUNCTION calc_predict_indices_age
 for(t=1;t<=age_surv_num;t++){
     realage=0;
     for(y=styrR;y<=endyr;y++){
       if (age_obs_surv_indices(y,t)>=0.)                              // Skip Missing Values (-1)
        {
          realage=(int)floor(age_surv_ages(t));
           age_pred_surv_indices(y,t)=mfexp(age_qs(t))*N(y,realage)*mfexp(-1.*age_surv_flag(t)*Z(y,realage));
        }
       if (age_obs_surv_indices(y,t)==-1) age_pred_surv_indices(y,t)=-1;
     }//y loop
     q_YOY(t)=mfexp(age_qs(t));
   }//t loop

//CALCULATE PREDICTED AGGREGATE INDICES
FUNCTION calc_predict_indices_agg
 for(t=1;t<=agg_surv_num;t++){
     cnt=0;
     adds=0;
     realage=0;
     diff2=0;
     for(y=styrR;y<=endyr;y++){
       if (agg_obs_surv_indices(y,t)>=0.)                              // Skip Missing Values (-1)
        {
           realage=(int)floor(agg_surv_ages(t));
           diff2=int(ceil(agg_surv_ages(t)*100.)-(floor(agg_surv_ages(t))*100.));
           pgroup=0;
           for (a=realage;a<=diff2;a++)
             {
              pgroup+=N(y,a)*mfexp(-1.*agg_surv_flag(t)*Z(y,a));
             }
              agg_pred_surv_indices(y,t)=mfexp(agg_qs(t))*pgroup;
        }//agg_surv_indices>=0
       if (agg_obs_surv_indices(y,t)==-1) agg_pred_surv_indices(y,t)=-1;
     }//y loop
     q_Agg(t)=mfexp(agg_qs(t));
  }//t loop

//CALCULATE PREDICTED SURVEY WITH AGE COMPOSITION INDICES
FUNCTION calc_predict_indices_ac
  for(int t=1;t<=ac_surv_num;t++){
       for(y=styrR;y<=endyr;y++){
          for(a=1;a<=nages;a++){
             calc_comps(t,y,a)=-1;
             if (surv_comps(t,y,a)>=0.)// Skip Missing Values (-1)
              {
                calc_comps(t,y,a)=surv_sel(a,t)*mfexp(ac_qs(t))*N(y,a)*mfexp(-1.*ac_surv_flag(t)*Z(y,a));
              }
            }//a loop
         }//y loop
      q_AC(t)=mfexp(ac_qs(t));
     }//t loop


  for(int t=1;t<=ac_surv_num;t++){
     for(y=styrR;y<=endyr;y++){
         sumage=0;
         for (a=1;a<=nages;a++){
            if(surv_comps(t,y,a)>=0.){sumage+=calc_comps(t,y,a);}
           }
            if(sumage>0.) {ac_pred_surv_indices(y,t)=sumage;}
            if(sumage<=0.){ac_pred_surv_indices(y,t)=-1;}

         for (a=1;a<=nages;a++){
            surv_pred_comps(t,y,a)=-1;
            if(sumage>0.){
              if(surv_comps(t,y,a)>=0.){surv_pred_comps(t,y,a)=calc_comps(t,y,a)/sumage;}
             }
            if(sumage<=0.){surv_pred_comps(t,y,a)=-1;}
           }
       }
    }
```

```
//CALCULATE LIKELIHOODS
FUNCTION scam_likelihood
 f_total_catch=0.;
 f_age_comp=0.;
 cnt=0;

 //CALCULATE TOTAL CATCH WEIGHTED RESIDUAL SUM OF SQUARES
    for(y=styr;y<=endyr;y++){
        f_total_catch+=square((log(obs_total_catch(y)+0.00001)-
log(pred_total_catch(y)+0.00001))/total_catch_CV(y));
      cnt+=1;
      }
     f_total_catch=f_total_catch*l_wgt;

//CALCULATE CATCH AGE COMP LIKELIHOOD
     for(y=styr;y<=endyr;y++){
       for(a=1;a<=nages;a++){
          f_age_comp-=ss_age_comp(y)*obs_age_comp(y,a)*log(pred_age_comp(y,a)+1e-7);
        }
       }
    f_age_comp=f_age_comp*caa_wgt;

//CALCULATE YOY AND YEARLING WEIGHTED RESIDUAL SUM OF SQUARES
   for(t=1;t<=age_surv_num;t++){
     like_age(t)=0;
     for(y=styrR;y<=endyr;y++){
         if(age_obs_surv_indices(y,t)>=0.){
       like_age(t)+=square((log(age_obs_surv_indices(y,t)+0.00001)-
log(age_pred_surv_indices(y,t)+0.00001))/age_surv_CV(y,t));
        cnt+=1;
          }
     }
     like_age(t)=like_age(t)*yoy_wgt(t);
   }

//CALCULATE AGGREGATE SURVEY WEIGHTED RESIDUAL SUM OF SQUARES
   for(t=1;t<=agg_surv_num;t++){
      like_agg(t)=0.;
      for(y=styrR;y<=endyr;y++){
           if(agg_obs_surv_indices(y,t)>=0.){
         like_agg(t)+=square((log(agg_obs_surv_indices(y,t)+0.00001)-
log(agg_pred_surv_indices(y,t)+0.00001))/agg_surv_CV(y,t));
          cnt+=1;
           }
      }
     like_agg(t)=like_agg(t)*agg_wgt(t);
   }

// CALCULATE SURVEY WITH AGE COMPOSITIONS WEIGHTED RESIDUAL SUM OF SQUARES
   for(t=1;t<=ac_surv_num;t++){
      like_ac_surv(t)=0;
      for(y=styrR;y<=endyr;y++){
        if(ac_obs_surv_indices(y,t)>=0.){
         like_ac_surv(t)+=square((log(ac_obs_surv_indices(y,t)+0.00001)-
log(ac_pred_surv_indices(y,t)+0.00001))/ac_surv_CV(y,t));
         cnt+=1;
        }
      }
      like_ac_surv(t)=like_ac_surv(t)*ac_surv_wgt(t);
   }

// CALCULATE SURVEY AGE COMPOSITIONS LIKELHOOD
   for(t=1;t<=ac_surv_num;t++){
      like_ac_age(t)=0.;
      for(y=styrR;y<=endyr;y++){
         for(a=1;a<=nages;a++){
           if(surv_comps(t,y,a)!=-1){
            like_ac_age(t)-=ac_ss(y,t)*surv_comps(t,y,a)*log(surv_pred_comps(t,y,a)+1e-7);
            }
         }
      }
        like_ac_age(t)=like_ac_age(t)*ac_age_wgt(t);
   }

FUNCTION evaluate_the_objective_function
  f=0;
 //CALCULATE CONCENTRATED LIKELIHOOD FOR ALL DATA WITH LOGNORMAL ERRORS
  f+=0.5*cnt*log((sum(like_age)+sum(like_agg)+sum(like_ac_surv)+f_total_catch)/cnt);
 //SUM REMAINING LIKELIHOODS
  f+=sum(like_ac_age);
```

```
  f+=f_age_comp;
  f+=R_lam*norm2(log_R_dev);

//CALCULATE PENALTY CONSTRAINT FOR F
    if(current_phase()<3){
     fpen=10.*norm2(mfexp(log_avg_F+log_F_dev)-0.15);
    }
    else{
     fpen=0.001*norm2(mfexp(log_avg_F+log_F_dev)-0.15);
    }
    if(active(log_F_dev)){
      fpen+=norm2(log_F_dev);
     }
  f+=F_lam*fpen;


REPORT_SECTION
 report <<"Likelihood Components" << endl;
 report <<" "<<endl;
 report <<"                      "<<"\t"<<"Weight"<<"   "<<"RSS"<<endl;
 report <<" Total Catch        : "<<"\t"<<l_wgt<<"\t"<<setw(10)<<f_total_catch<<endl;
 report <<" YOY/Yearl Surveys    " << endl;
 for(t=1;t<=age_surv_num;t++){
 report <<"   Survey "<<t<<"         : "<<"\t"<<yoy_wgt(t)<<"\t"<<setw(10)<<like_age(t)<<endl;
 }
 report <<" Aggregate Surveys    " << endl;
   for(t=1;t<=agg_surv_num;t++)
      {
 report <<"   Survey "<<t<<"         : "<<"\t"<<agg_wgt(t)<<"\t"<<setw(10)<<like_agg(t)<<endl;
      }
 report <<" Age Survey Indices   " << endl;

  for(t=1;t<=ac_surv_num;t++)
      {
 report <<"   Survey "<<t<<"         : "<<"\t"<<ac_surv_wgt(t)<<"\t"<<setw(10)<<like_ac_surv(t)<<endl;
      }
 report<<" "<<endl;
 report <<" Total RSS            "<<"\t"<<"
"<<"\t"<<setw(10)<<(sum(like_age)+sum(like_agg)+sum(like_ac_surv)+f_total_catch)<<endl;
 report <<" No. of Obs           "<<"\t"<<" "<<"\t"<<setw(10)<<cnt<<endl;
 report <<" Conc. Likelihood     "<<"\t"<<"  "<<"\t"<<setw(10)<<
           0.5*cnt*log((sum(like_age)+sum(like_agg)+sum(like_ac_surv)+f_total_catch)/cnt) <<endl;

 report<<" "<<endl;
 report <<" Catch Age Comps    : "<<"\t"<<caa_wgt <<"\t"<<setw(10)<<f_age_comp<<endl;
 report <<" Survey Age Comps       " <<endl;
  for(t=1;t<=ac_surv_num;t++)
      {
 report <<"   Survey "<<t<<"         : "<<"\t"<<ac_age_wgt(t)<<"\t"<<setw(10)<<like_ac_age(t)<<endl;
      }
 report <<" "<<endl;

 report <<"Recr Devs "<<"          : "<<"\t"<<R_lam<<"\t"<<setw(10)<<R_lam*norm2(log_R_dev)<<endl;
 report <<"F Devs    "<<"          : "<<"\t"<<F_lam<<"\t"<<setw(10)<<F_lam*norm2(log_F_dev)<<endl;

 report <<" "<<endl;
 report <<"Total Likelihood    : "<<"\t"<<"  "<<"\t"<<setw(10)<<f<<endl;
 report << " " << endl;
 report<<"*********************************************************************************************"<<endl;
 report<<"Mortality Rates "<<endl;
 report << "Natural" << endl;
 report << M << endl;
 report<<" "<<endl;
 report << "Fishing" << endl;
 report << mfexp(log_avg_F+log_F_dev)<< endl;
 report<<" "<<endl;
 report <<"*******************************************SCAM Output**********************"<<endl;
 report << "Total Catch" << endl;
 report << "Observed" << obs_total_catch << endl;
 report << "Predicted" << pred_total_catch <<endl;
 report <<" "<<endl;
 report << "Obs Catch Age Comp "<< endl;
 report<<obs_age_comp<<endl;
 report <<" "<<endl;
 report <<"Pred Catch Age comp"<<endl;
 report<<pred_age_comp<<endl;
 report <<" "<<endl;
 report << "Number-At-Age "<< endl;
 report << N<<endl;
 report << "Selectivity Period 1" << endl;
 report <<"Age " << agebins << endl;
```

```
report << "p1_sel" << p1_sel << endl;
report <<" "<<endl;
report << "Selectivity Period 2" << endl;
report <<"Age " << agebins << endl;
report << "p2_sel" << p2_sel << endl;
report <<" "<<endl;
report << "Selectivity Period 3" << endl;
report <<"Age " << agebins << endl;
report << "p3_sel" << p3_sel << endl;
report << "Selectivity  Period 4" << endl;
report <<"Age " << agebins << endl;
report << "p4_sel" << p4_sel << endl;
report <<" "<<endl;

report <<"Period Selectivity Parameters"<<endl;
report <<"P1: "<<p1_A50<<" "<<p1_slope<<endl;
report <<"P2: "<<p2_A50<<" "<<p2_slope<<endl;
report <<"P3: "<<p3_A50<<" "<<p3_slope<<endl;
report <<"P4: "<<p4_A50<<" "<<p4_slope<<endl;

report<<"Observed Age Indices"<<endl;
report<<age_obs_surv_indices<<endl;
report <<" "<<endl;
report<<"Predicted Age Indices"<<endl;
report<<age_pred_surv_indices<<endl;
report <<" "<<endl;
report<<"Age Survey qs"<<endl;
report<<mfexp(age_qs)<<endl;
report <<" "<<endl;
report<<"YOY/Yearling CVs"<<endl;
report<<age_surv_CV<<endl;
report <<" "<<endl;
report<<"Observed Aggregate Indices"<<endl;
report<<agg_obs_surv_indices<<endl;
report <<" "<<endl;
report<<"Predicted Aggregate Indices"<<endl;
report<<agg_pred_surv_indices<<endl;
report <<" "<<endl;
report<<"Aggregate Survey qs"<<endl;
report<<mfexp(agg_qs)<<endl;
report <<" "<<endl;
report<<"Aggregate Indices CVs"<<endl;
report<<agg_surv_CV<<endl;
report <<" "<<endl;
report<<"Observed Age Comp Indices"<<endl;
report<<ac_obs_surv_indices<<endl;
report <<" "<<endl;
report<<"Predicted Age Comps Indices"<<endl;
report<<ac_pred_surv_indices<<endl;
report <<" "<<endl;
report<<"Age Comps Survey qs"<<endl;
report<<mfexp(ac_qs)<<endl;
report <<" "<<endl;
report<<"Age Comps Indices CVs"<<endl;
report<<ac_surv_CV<<endl;
report <<" "<<endl;
report<<"Observed Survey Age Comps "<<endl;
report<<surv_comps<<endl;
report <<" "<<endl;
report<<"Predicted Survey Age Comps "<<endl;
report<<surv_pred_comps<<endl;
report <<" "<<endl;
report<<"Predicted Survey Age Comps Selectivities"<<endl;
report<<surv_sel<<endl;
report <<" "<<endl;
report<<"Predicted Survey Age Comps Selectivities Parameters"<<endl;
report<<NY_e<<NY_surv<<endl;
report<<NJ_surv<<endl;
report<<MD_surv<<endl;
report<<DE_surv<<endl;
report <<" "<<endl;
report<<"Fishing Mortality at age"<<endl;
report<<F<<endl;
report <<" "<<endl;
report<<"SSB at age"<<endl;
report<<SSB<<endl;
report <<" "<<endl;
report<<"Rivards Weights"<<endl;
report<<rwgts<<endl; report <<" "<<endl;
report<<"Catch Weights"<<endl;
report<<ssw<<endl; report <<" "<<endl;
```

```
 report<<"January-1 stock biomass"<<endl;
 report<<jan1bio<<endl; report <<" "<<endl;
 report<<"Catch biomass"<<endl;
 report<<catchbio<<endl; report <<" "<<endl;


FINAL_SECTION
 // Output data to files for import into R
  ofstream ofs28("effss.out");
      sumdo1=0;
      dodo1=0;
      for(y=styr;y<=endyr;y++)
        {
          sumdo=0;
          dodo=0;
          for(a=1;a<=nages;a++)
            {
               if(obs_age_comp(y,a)!=-1)
                {
                 sumdo+=pred_age_comp(y,a)*(1-pred_age_comp(y,a));
                 dodo+=square(obs_age_comp(y,a)-pred_age_comp(y,a));

               }
               if(obs_age_comp(y,a)==-1)
                {
                 sumdo=0;
                 dodo=0;
               }
            }
           if(sumdo>0 && dodo>0) sumdo1+=sumdo/dodo;
        }

     for(y=styr;y<=endyr;y++)
       {
        if (obs_total_catch(y)!=-1) dodo1+=1;
       }
     ofs28<<sumdo1/dodo1<<endl;
 //Survey age comps
   for(t=1;t<=ac_surv_num;t++)
     {
      sumdo1=0;
      dodo1=0;
      for(y=styrR;y<=endyr;y++)
        {

          sumdo=0;
          dodo=0;
          for(a=1;a<=nages;a++)
            {
               if(surv_comps(t,y,a)!=-1)
                {
                 sumdo+=surv_pred_comps(t,y,a)*(1-surv_pred_comps(t,y,a));
                 dodo+=square(surv_comps(t,y,a)-surv_pred_comps(t,y,a));

               }
               if(surv_comps(t,y,a)==-1)
                {
                 sumdo=0;
                 dodo=0;
               }
            }
           if(sumdo>0 && dodo>0) sumdo1+=sumdo/dodo;
       }

     for(y=styrR;y<=endyr;y++)
       {
        if (ac_obs_surv_indices(y,t)!=-1) dodo1+=1;
       }
     ofs28<<sumdo1/dodo1<<endl;

    }
 // Calculate F and sd
   ofstream ofs1("F.out");
   d=n_parms+1;
   for(t=styr;t<=endyr;t++)
     {
       ofs1<<F_ann(t)<<"\t"<<sigma(d,1)<<endl;
       d+=1;
     }
 //Calculate R and sd
  ofstream ofs2("R.out");
```

```
   for(t=styrR;t<=endyr;t++)
   {
       ofs2<<R(t)<<"\t"<<sigma(d,1)<<endl;
       d+=1;
   }
//Output Indices qs
  ofstream ofs13("YOYqs.out");
    for(t=1;t<=age_surv_num;t++)
   {
     ofs13<<mfexp(age_qs(t))<<"\t"<<sigma(d,1)<<endl;
     d+=1;
   }
   ofstream ofs14("Aggqs.out");
     for(t=1;t<=agg_surv_num;t++)
   {
    ofs14<<mfexp(agg_qs(t))<<"\t"<<sigma(d,1)<<endl;
    d+=1;
   }
  ofstream ofs15("ACqs.out");
    for(t=1;t<=ac_surv_num;t++)
   {
    ofs15<<mfexp(ac_qs(t))<<"\t"<<sigma(d,1)<<endl;
    d+=1;
   }
//Output N-at-age
  ofstream ofs4("N.out");
  ofs4<<N<<endl;
//Output Catch Age Comp
  ofstream ofs5("CACpred.out");
  ofs5<<pred_age_comp<<endl;
//Output Catch Age Comp
  ofstream ofs51("CACobs.out");
  ofs51<<obs_age_comp<<endl;
//Output Total Catch
  ofstream ofs6("CatPred.out");
  for(y=styr;y<=endyr;y++)
    {
     ofs6<<pred_total_catch(y)<<endl;
    }
 //Output Total Catch
  ofstream ofs61("CatObs.out");
  for(y=styr;y<=endyr;y++)
    {
     ofs61<<obs_total_catch(y)<<endl;
    }
//Output Total Catch
  ofstream ofs7("Fatage.out");
     ofs7<<F<<endl;
//Output Selectivity
  ofstream ofs8("Select.out");
  for(a=1;a<=nages;a++)
    {
    ofs8<<p1_sel(a)<<"\t"<<p2_sel(a)<<"\t"<<p3_sel(a)<<"\t"<<p4_sel(a)<<endl;
    }
//Output Selectivity Parameters
  ofstream ofs9("Selparms.out");
    ofs9<<p1_A50<<"\t"<<p1_slope<<endl;
    ofs9<<p2_A50<<"\t"<<p2_slope<<endl;
    ofs9<<p3_A50<<"\t"<<p3_slope<<endl;
    ofs9<<p4_A50<<"\t"<<p4_slope<<endl;
//Output Indices
  ofstream ofs10("YOYPred.out");
     ofs10<<age_pred_surv_indices<<endl;

   ofstream ofs101("YOYObs.out");
     ofs101<<age_obs_surv_indices<<endl;

   ofstream ofs11("AggPred.out");
     ofs11<<agg_pred_surv_indices<<endl;
   ofstream ofs111("AggObs.out");
     ofs111<<agg_obs_surv_indices<<endl;

  ofstream ofs12("ACPred.out");
     ofs12<<ac_pred_surv_indices<<endl;
  ofstream ofs121("ACObs.out");
     ofs121<<ac_obs_surv_indices<<endl;


//Output Sruvey age comps
  ofstream ofs16("survacpred.out");
     ofs16<<surv_pred_comps<<endl;
```

```
 ofstream ofs161("survacobs.out");
     ofs161<<surv_comps<<endl;
 ofstream ofs169("calccomps.out");
     ofs169<<calc_comps<<endl;


//Output Sruvey select
  ofstream ofs17("survsel.out");
     ofs17<<surv_sel<<endl;
//Output Sruvey select parms

 // ofstream ofs18("survparms.out");
 //    ofs18<<surv_A50<<endl;
  //   ofs18<<surv_slope<<endl;


//Output Total SSB
  ofstream ofs27("SSB.out");
     ofs27<<SSB<<endl;

//Output jani1biomass
  ofstream ofs29("jan1bio.out");
     ofs29<<jan1bio<<endl;
//Output catch biomass
  ofstream ofs30("catchbio.out");
     ofs30<<catchbio<<endl;
```

Data used in the striped bass statistical catch-at-age model.

```
#Starting and ending year of catch data
1982
2006

#Number of ages
13

#Agebin vector
1 2 3 4 5 6 7 8 9 10 11 12 13

# CV of catch (harvest+dead discards)
0.40 0.23 0.19 0.17 0.21 0.14 0.27 0.18 0.08 0.09 0.06 0.07 0.06 0.05 0.05 0.04 0.04 0.04
0.04 0.03 0.03 0.04 0.04 0.04 0.04

# Total catch likelihood weight
10

#Total catch (numbers)
766200 727700 1084900 400800 384900 239100 444900 479900 921300 988400 986900 1437000 1866600 2999700 3376200
4580100
4118300 3704300 5044400 4344000 3889500 4836200 5272677 5596311 6113218
#Catch age comp samples sizes
380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380 380

#Catch age comp likelihood weight
1

#Fishery age comp data
0.002 0.138 0.335 0.288 0.076 0.025 0.032 0.022 0.015 0.014 0.014 0.018 0.020
0.005 0.152 0.245 0.265 0.206 0.054 0.026 0.006 0.004 0.005 0.006 0.008 0.019
0.005 0.500 0.279 0.076 0.056 0.048 0.017 0.004 0.002 0.002 0.001 0.001 0.010
0.003 0.181 0.254 0.101 0.146 0.108 0.109 0.043 0.016 0.008 0.002 0.002 0.026
0.029 0.055 0.166 0.345 0.130 0.083 0.053 0.062 0.024 0.014 0.009 0.004 0.026
0.006 0.046 0.157 0.215 0.281 0.105 0.055 0.027 0.027 0.013 0.006 0.008 0.054
0.006 0.069 0.094 0.142 0.241 0.220 0.091 0.055 0.031 0.013 0.008 0.007 0.022
0.001 0.075 0.166 0.142 0.219 0.199 0.095 0.044 0.022 0.008 0.007 0.004 0.019
0.002 0.050 0.135 0.204 0.188 0.179 0.113 0.074 0.022 0.008 0.006 0.004 0.015
0.002 0.074 0.147 0.211 0.164 0.103 0.092 0.084 0.059 0.024 0.014 0.003 0.023
0.003 0.046 0.202 0.192 0.179 0.111 0.063 0.069 0.059 0.045 0.009 0.004 0.016
0.001 0.048 0.129 0.228 0.201 0.129 0.060 0.047 0.057 0.053 0.029 0.006 0.012
0.003 0.078 0.187 0.156 0.197 0.125 0.073 0.046 0.054 0.043 0.019 0.012 0.008
0.001 0.145 0.157 0.152 0.135 0.163 0.072 0.065 0.051 0.030 0.018 0.006 0.005
0.001 0.029 0.192 0.193 0.161 0.139 0.131 0.062 0.041 0.020 0.013 0.014 0.006
0.001 0.064 0.131 0.212 0.150 0.125 0.100 0.091 0.049 0.031 0.015 0.007 0.006
0.006 0.045 0.118 0.172 0.273 0.124 0.068 0.064 0.052 0.028 0.023 0.011 0.016
0.002 0.029 0.113 0.175 0.173 0.197 0.095 0.064 0.055 0.040 0.028 0.013 0.013
0.008 0.064 0.083 0.195 0.202 0.155 0.147 0.062 0.032 0.028 0.012 0.006 0.006
0.007 0.036 0.099 0.138 0.192 0.161 0.133 0.111 0.048 0.028 0.024 0.011 0.011
0.007 0.055 0.081 0.122 0.151 0.197 0.137 0.095 0.074 0.028 0.023 0.013 0.017
0.006 0.053 0.101 0.125 0.147 0.124 0.145 0.101 0.077 0.058 0.026 0.017 0.019
0.013 0.034 0.151 0.141 0.097 0.103 0.098 0.119 0.084 0.063 0.050 0.023 0.025
0.003 0.082 0.075 0.196 0.171 0.093 0.076 0.067 0.084 0.058 0.048 0.022 0.025
0.006 0.037 0.191 0.108 0.179 0.115 0.059 0.059 0.060 0.076 0.050 0.029 0.031

#########################################################################
# YOY and Yearling Surveys - NY YOY, NJ YOY, MD YOY, VA YOY, NY 1, MD 1
#########################################################################
#Number of No age comp surveys
6

#Survey time of year fractions
0 0 0 0 0 0

# Survey ages
1 1 1 1 2 2

# YOY/Yearling Likelihood Weight
1 1 1 1 1 1

#Survey CVs
-1 -1 0.19 -1 -1 0.33
-1 -1 0.13 -1 -1 0.26
-1 -1 0.16 -1 -1 0.31
-1 -1 0.21 -1 -1 0.39
-1 -1 0.20 -1 -1 0.18
-1 -1 0.22 -1 -1 0.19
-1 -1 0.13 -1 -1 0.31
-1 -1 0.24 -1 -1 0.45
```

```
-1 -1 0.20 -1 -1 0.41
-1 -1 0.12 -1 -1 0.24
0.15 -1 0.16 -1 -1 0.36
0.20 0.714 0.15 -1 -1 0.26
0.172 1.000 0.193 -1 -1 0.52
0.132 0.353 0.136 0.215 -1 0.57
0.189 0.600 0.229 0.202 -1 0.21
0.200 0.255 0.194 0.299 -1 1.00
0.123 1.000 0.216 0.158 0.556 0.43
0.121 0.271 0.232 0.146 0.360 0.51
0.212 0.216 0.224 0.119 0.351 0.53
0.072 0.140 0.268 0.119 0.449 0.29
0.113 0.210 0.264 0.159 0.302 0.29
0.096 0.136 0.149 0.118 0.307 0.26
0.098 0.164 0.174 0.162 0.322 0.24
0.074 0.190 0.228 0.170 0.288 0.39
0.106 0.211 0.238 0.089 0.387 0.33
0.092 0.164 0.128 0.100 0.375 0.26
0.136 0.210 0.131 0.127 0.297 0.35
0.108 0.144 0.315 0.101 0.394 0.64
0.107 0.138 0.116 0.105 0.500 0.34
0.093 0.225 0.141 0.105 0.364 0.34
0.143 0.169 0.178 0.128 0.243 0.34
0.182 0.182 0.103 0.099 0.257 0.25
0.208 0.174 0.264 0.130 0.226 0.20
0.106 0.209 0.121 0.149 0.246 0.15
0.092 0.224 0.117 0.086 0.197 0.34
0.095 0.144 0.152 0.122 0.408 0.20
0.212 0.126 0.202 0.104 0.486 0.28

#Survey Indices, -1 for missing data
-1 -1 10.52 -1 -1 0.71
-1 -1 30.52 -1 -1 0.22
-1 -1 11.77 -1 -1 7.31
-1 -1 11.01 -1 -1 1.73
-1 -1 8.92 -1 -1 0.86
-1 -1 10.13 -1 -1 0.44
-1 -1 6.69 -1 -1 0.46
-1 -1 4.91 -1 -1 0.42
-1 -1 4.85 -1 -1 0.10
-1 -1 8.45 -1 -1 0.31
5.00 -1 4.24 -1 -1 0.80
23.91 0.07 1.98 -1 -1 0.30
21.44 0 1.22 -1 -1 0.04
30.50 0.17 8.45 3.05 -1 0.02
48.03 0.05 1.37 2.90 -1 0.63
37.11 0.47 4.21 5.63 -1 0.00
3.85 0.04 2.93 2.27 2.81 0.36
6.14 0.48 4.14 4.65 0.78 0.05
60.67 1.11 4.80 15.22 0.62 0.15
52.30 0.57 2.65 7.49 7.07 0.11
41.94 2.71 25.20 10.99 9.25 0.40
37.97 2.06 2.14 6.94 0.96 0.75
6.85 1.16 4.44 3.71 7.59 0.34
17.29 3.99 9.03 9.83 5.66 0.32
26.49 5.97 39.76 12.91 3.46 0.44
28.49 2.32 16.12 8.39 13.21 2.51
27.39 7.61 9.27 5.14 4.85 0.23
14.66 4.3 59.39 20.88 11.09 0.23
50.35 2.25 7.98 8.24 4.34 0.62
22.91 3.51 12.67 11.58 10.09 0.35
52.54 4.85 18.12 2.46 7.51 0.79
7.82 6.05 13.77 15.23 11.39 0.52
91.24 2.47 50.75 14.58 7.55 0.56
21.53 1.29 4.73 4.52 8.88 1.61
34.97 8.67 25.75 18.92 3.10 0.13
14.33 2.98 11.44 10.71 11.24 1.91
35.01 2.47 17.79 7.51 2.99 0.64

#################################################################################
# Aggregate Surveys MRFSS CTCPUE NEFSC CTTRL
#################################################################################

#Number of No age comp surveys
4

#Survey time of year fractions
0.5 0.5 0.3333 0.3333

# Survey ages
3.13 2.13 2.09 2.04
```

```
#Aggregate Surveys Likelihood Weights
1 1 1 1

#Survey CVs
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 0.574 -1 -1
-1 0.453 -1 -1
-1 0.553 -1 1
-1 0.32 -1 1
-1 0.494 -1 1
-1 0.275 -1 0.40
0.79 0.268 -1 0.50
0.85 0.177 -1 0.33
0.77 0.165 -1 0.25
0.38 0.415 0.156 0.33
0.24 0.194 0.373 0.25
0.21 0.141 0.357 0.20
0.2 0.227 0.579 0.20
0.2 0.291 0.229 0.23
0.2 0.235 0.305 0.20
0.2 0.175 0.332 0.20
0.2 0.217 0.128 0.27
0.2 0.207 0.14 0.21
0.2 0.165 0.284 0.21
0.2 0.146 0.363 0.27
0.2 0.127 0.157 0.29
0.2 0.151 0.332 0.18
0.2 0.169 0.302 0.18
0.2 0.15 0.238 0.26
0.2 0.18 0.534 0.25

#Survey Indices, -1 for missing data
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 0.903 -1 -1
-1 0.751 -1 -1
-1 0.922 -1 0.022
-1 0.891 -1 0
-1 1.518 -1 0
-1 1.135 -1 0.053
0.362145961 1.361 -1 0.036
0.266005882 1.84 -1 0.063
0.24098429 2.203 -1 0.162
0.41409724 2.163 0.258 0.146
0.749170058 2.377 0.247 0.22
0.610929185 2.845 0.634 0.273
0.908054028 3.954 3.441 0.296
1.174633583 5.396 1.101 0.6
1.333341093 7.583 0.807 0.63
1.369797852 5.99 1.373 0.85
1.714551001 7.574 0.81 0.97
1.614670646 5.526 0.767 1.1
1.510928023 6.873 1.409 0.84
1.2616274 7.56 0.795 0.613
1.052792365 5.87 1.156 1.3
0.929391076 6.35 1.049 0.87
1.009113292 8.15 0.359 0.56
1.168405332 13.15470042 0.312 1.17
1.386671533 13.52818536 0.792 0.61
```

```
#########################################################################
#Surveys with Age Data - NYOHS NJTRL MDAdults DESSN
#########################################################################
#Number of Age surveys
4

#Survey time of year fractions
0.75 0.25 0.25 0.25

#Surveys Indices with Age Data Likelihood Weights
1 1 1 1

# Survey Age Comp Likelihood Weights
1 1 1 1

#Survey CVs
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 0.3 -1
-1 -1 0.3 -1
0.250804572 -1 0.25 -1
0.185277089 -1 0.25 -1
0.17721519 0.551 0.25 -1
0.285590278 0.607 0.25 -1
0.32486298 0.612 0.25 -1
0.407348243 0.449 0.25 -1
0.173025732 0.681 0.25 -1
0.73776908 0.482 0.25 -1
0.562797013 0.173 0.25 -1
0.184391737 0.752 0.25 0.190
0.260247235 0.327 0.25 0.232
0.25758645 0.229 0.25 0.250
0.4625 0.550 0.25 0.183
0.27466319 0.332 0.25 0.209
0.567319461 0.217 0.25 0.224
0.49045281 0.592 0.25 0.220
0.377706126 0.337 0.25 0.140
0.177619893 0.360 0.25 0.172
0.373695198 0.380 0.25 0.177
0.261588426 0.674 0.25 0.199

#Survey Indices, -1 for missing data
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 488.1 -1
-1 -1 1006.9 -1
90.11 -1 715.0 -1
60.45 -1 327.4 -1
21.33 0.280 395.9 -1
115.2 1.961 503.6 -1
200.7 2.461 460.8 -1
125.2 1.034 629.0 -1
45.08 1.083 625.3 -1
102.2 2.145 513.5 -1
147.3 7.165 461.8 -1
```

```
392.1 18.687 759.2 4.044
614.8 13.626 387.1 4.659
850.2 8.087 478.8 5.600
104 6.270 396.6 3.261
230.1 8.492 352.5 4.000
163.4 2.385 282.6 4.889
183.3 3.177 400.3 4.021
434.2 10.427 455.0 4.913
281.5 9.304 610.9 5.899
239.5 9.360 500.3 3.908
338.7 3.951 505.8 5.333

# Age Samples for Age Comps - (Survey (column) sample size (row))
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 77 -1
-1 -1 77 -1
56 -1 77 -1
56 -1 77 -1
56 23 77 -1
56 23 77 -1
56 23 77 -1
56 23 77 -1
56 23 77 -1
56 23 77 -1
56 23 77 -1
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
56 23 77 87
# Age Comp Data for Surveys: Ages 1-13 must be present
#NYOHS
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 0.0318 0.1949 0.3591 0.2787 0.0883 0.0349 0.0067 0.0017 0 0.0006 0 0.0028
-1 0.2255 0.2687 0.1945 0.166 0.0851 0.0218 0.0144 0.0039 0.0021 0.0007 0 0.0137
-1 0.1833 0.269 0.1478 0.1596 0.1025 0.0936 0.0217 0.003 0.002 0.003 0.002 0.0108
-1 0.0608 0.2957 0.3063 0.1139 0.0985 0.0557 0.0444 0.0158 0.0058 0.001 0 0.0023
-1 0.207 0.3666 0.2439 0.0519 0.0166 0.0253 0.0416 0.023 0.0063 0.002 0.0036 0.0115
-1 0.0792 0.4166 0.2577 0.1211 0.0329 0.0143 0.017 0.025 0.0175 0.0032 0.0058 0.0096
-1 0.1563 0.3868 0.2908 0.0701 0.0328 0.0094 0.009 0.0115 0.0131 0.007 0.0025 0.0082
-1 0.141 0.2705 0.1562 0.1346 0.0832 0.0546 0.0375 0.0222 0.0406 0.0127 0.0241 0.0203
-1 0.245 0.2695 0.2542 0.072 0.0658 0.0352 0.0123 0.0054 0.0123 0.0115 0.0031 0.0084
-1 0.0832 0.7475 0.1142 0.0328 0.0094 0.0073 0.0027 0.0013 0.0007 0 0.0005 0.0003
-1 0.2063 0.2425 0.4508 0.0669 0.0184 0.0037 0.0037 0.0039 0.0017 0.0007 0.0009 0.0006
-1 0.1876 0.2969 0.1714 0.2855 0.0366 0.0091 0.0058 0.0029 0.0002 0.001 0.0015 0.0011
```

```
-1 0.0697 0.6277 0.1722 0.0594 0.0438 0.005 0.0032 0.0046 0.0035 0.0039 0.0007 0.0046
-1 0.1273 0.193 0.4338 0.1541 0.0364 0.0368 0.0041 0.0039 0.0016 0.0018 0.001 0.0044
-1 0.0524 0.4553 0.1474 0.2129 0.0735 0.0274 0.0194 0.0032 0.0039 0.0011 0 0.0025
-1 0.3225 0.2261 0.1843 0.0805 0.0735 0.0572 0.0198 0.0198 0.0013 0.0048 0.0018 0.0057
-1 0.2022 0.3647 0.1251 0.0922 0.0406 0.0646 0.0506 0.0227 0.0177 0.0126 0.0009 0.0049
-1 0.0501 0.5698 0.2734 0.0628 0.0222 0.0076 0.0061 0.0036 0.0011 0.0014 0.0017 0.0002
-1 0.2444 0.1280 0.4126 0.1370 0.0336 0.0138 0.0035 0.0090 0.0065 0.0035 0.0037 0.0045
-1 0.0639 0.6359 0.0728 0.1610 0.0424 0.0144 0.0057 0.0025 0.0003 0.0010 0.0000 0.0000
 #NJTRL
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 0.278 0.444 0.006 0.137 0.052 0.011 0.016 0 0.056 0 0 0
-1 0.061 0.182 0.02 0.414 0.132 0.029 0.097 0.005 0.061 0 0 0
-1 0.277 0.284 0.021 0.02 0.148 0.132 0.017 0.034 0.046 0.021 0 0
-1 0.258 0.478 0.061 0.064 0.055 0.074 0.01 0 0 0 0 0
-1 0.238 0.353 0.15 0.087 0.123 0.024 0.025 0 0 0 0 0
-1 0.287 0.37 0.155 0.09 0.048 0.031 0.01 0.009 0 0 0 0
-1 0.658 0.172 0.067 0.045 0.032 0.012 0.007 0.004 0.003 0 0 0
-1 0.162 0.58 0.16 0.061 0.021 0.013 0.004 0 0 0 0 0
-1 0.187 0.409 0.236 0.113 0.035 0.012 0.005 0.001 0.003 0 0 0
-1 0.442 0.193 0.043 0.13 0.086 0.054 0.025 0.014 0.011 0.002 0.001 0
-1 0.077 0.32 0.181 0.256 0.115 0.032 0.011 0.005 0.003 0 0.001 0
-1 0.152 0.14 0.157 0.274 0.167 0.073 0.027 0.006 0.002 0.001 0 0
-1 0.148 0.167 0.199 0.299 0.103 0.042 0.023 0.013 0.006 0.001 0 0
-1 0.005 0.023 0.071 0.206 0.359 0.23 0.076 0.024 0.004 0 0 0
-1 0.304 0.238 0.041 0.126 0.097 0.122 0.049 0.015 0.006 0.001 0.001 0
-1 0.182 0.519 0.09 0.04 0.058 0.043 0.036 0.021 0.008 0.004 0.001 0
-1 0.493 0.218 0.061 0.106 0.047 0.042 0.019 0.009 0.002 0.002 0.000 0.001
-1 0.061 0.100 0.055 0.248 0.256 0.100 0.069 0.046 0.045 0.013 0.007 0.001
#MD SSN
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 0.288 0.626 0.065 0.010 0.003 0.005 0.000 0.001 0.000 0.000 0.001 0.002
-1 0.229 0.259 0.494 0.004 0.005 0.002 0.003 0.003 0.000 0.000 0.000 0.001
-1 0.199 0.361 0.161 0.246 0.025 0.003 0.004 0.000 0.000 0.000 0.000 0.001
-1 0.125 0.237 0.218 0.174 0.228 0.004 0.000 0.000 0.013 0.000 0.000 0.001
-1 0.084 0.391 0.203 0.115 0.123 0.083 0.000 0.000 0.000 0.000 0.000 0.000
-1 0.155 0.314 0.239 0.096 0.068 0.064 0.059 0.002 0.000 0.000 0.001 0.002
-1 0.159 0.415 0.135 0.102 0.058 0.057 0.042 0.023 0.001 0.003 0.000 0.005
-1 0.043 0.352 0.244 0.093 0.111 0.068 0.046 0.022 0.011 0.005 0.000 0.004
-1 0.065 0.211 0.299 0.141 0.082 0.083 0.059 0.036 0.012 0.005 0.001 0.005
-1 0.052 0.202 0.191 0.230 0.116 0.066 0.084 0.034 0.017 0.006 0.002 0.001
-1 0.108 0.254 0.146 0.132 0.112 0.087 0.054 0.043 0.025 0.021 0.008 0.010
-1 0.005 0.485 0.135 0.046 0.092 0.085 0.056 0.047 0.022 0.020 0.006 0.002
-1 0.105 0.120 0.348 0.119 0.056 0.051 0.067 0.058 0.032 0.031 0.010 0.005
-1 0.075 0.298 0.068 0.312 0.067 0.028 0.039 0.036 0.031 0.019 0.021 0.005
-1 0.018 0.439 0.202 0.143 0.089 0.029 0.017 0.028 0.013 0.013 0.007 0.003
-1 0.029 0.144 0.305 0.143 0.165 0.077 0.040 0.023 0.023 0.022 0.014 0.016
-1 0.017 0.138 0.185 0.183 0.082 0.101 0.134 0.047 0.042 0.035 0.020 0.017
-1 0.241 0.104 0.096 0.208 0.085 0.075 0.079 0.057 0.019 0.010 0.013 0.014
-1 0.039 0.242 0.105 0.082 0.135 0.125 0.068 0.060 0.076 0.022 0.023 0.024
```

```
-1 0.051 0.293 0.199 0.067 0.054 0.072 0.076 0.061 0.043 0.045 0.013 0.025
-1 0.135 0.211 0.148 0.194 0.049 0.052 0.043 0.055 0.041 0.035 0.023 0.015
-1 0.017 0.526 0.082 0.097 0.060 0.030 0.025 0.037 0.043 0.026 0.021 0.037
 #DE SSN
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 0.006 0.417 0.192 0.061 0.085 0.076 0.064 0.058 0.015 0.009 0.009 0.009
-1 0.093 0.074 0.391 0.137 0.051 0.064 0.073 0.032 0.03 0.023 0.009 0.023
-1 0.04 0.087 0.098 0.347 0.09 0.061 0.105 0.095 0.034 0.025 0.008 0.011
-1 0 0.105 0.144 0.177 0.235 0.072 0.054 0.076 0.058 0.051 0.014 0.014
-1 0.036 0.036 0.21 0.171 0.138 0.223 0.066 0.03 0.039 0.032 0.01 0.01
-1 0.006 0.115 0.1 0.185 0.11 0.14 0.2 0.05 0.015 0.04 0.02 0.02
-1 0.034 0.071 0.191 0.178 0.157 0.113 0.089 0.097 0.026 0.016 0.01 0.018
-1 0.02 0.097 0.097 0.134 0.089 0.111 0.125 0.105 0.121 0.034 0.028 0.038
-1 0.007 0.166 0.231 0.098 0.068 0.054 0.112 0.078 0.081 0.044 0.014 0.047
-1 0.096 0.157 0.168 0.198 0.081 0.046 0.03 0.036 0.061 0.036 0.046 0.046
-1 0.059 0.201 0.097 0.141 0.141 0.071 0.052 0.041 0.048 0.048 0.037 0.063
########################Catch Weight-at-age############################################
0.13 0.64 1.09 1.54 2.42 3.75 4.83 5.79 6.2 8.68 10.8 11.2 14.05
0.2 0.55 0.94 1.37 2.37 3.29 3.77 5.36 6.01 8.1 9.57 10.39 11.11
0.24 0.6 1.69 1.62 2.67 3.39 5.07 5.65 6.76 7.76 8.41 12.65 12.38
0.06 0.61 1.07 1.66 2.19 3.59 4.91 5.46 6.77 7.45 9 10.69 13.91
0.14 0.57 1.27 2.4 2.44 3.12 3.95 5.05 5.44 6.09 7.75 9.16 12.78
0.2 0.77 1.41 2.11 2.5 2.91 3.61 4.74 5.52 6.49 7.77 9.78 13.15
0.31 0.91 1.1 1.98 3.12 4.02 4.38 4.7 5.24 5.62 8.58 10.4 13.27
0.16 0.83 1.22 2.23 3.06 4.53 5.37 6.23 6.04 8.68 8.94 9.74 13.36
0.08 0.89 1.14 2.05 2.35 3.83 4.91 5.96 5.7 5.97 7.44 9.08 12.6
0.21 0.92 1.29 2.17 2.62 3.17 4.81 5.64 6.46 6.24 9.46 8.3 14.22
0.1 0.69 1.31 1.93 2.81 3.67 4.9 5.79 6.96 8.15 9.77 12.44 13.97
0.07 0.76 1.31 1.99 2.77 3.58 4.8 6.11 7.03 8.01 9.53 10.76 14.55
0.24 1.05 1.69 2.21 2.85 3.5 4.94 6.2 6.8 7.53 9.73 10.69 12.73
0.28 0.7 1.35 2.18 2.77 3.65 5.38 6.16 7.27 8.86 7.57 9.73 16.66
0.14 1.05 1.47 2.32 3.23 4.52 6.39 7.11 7.81 9.2 9.31 10.1 13.7
0.13 0.62 1.18 2.46 2.81 3.64 4.51 5.07 6.73 9.17 9.94 10.24 14.78
0.39 0.77 1.2 1.62 2.25 2.95 4.69 5.66 6.82 7.03 7.76 9.87 11.87
0.62 0.9 1.11 1.44 1.91 2.51 3.36 5.03 6.56 7.85 8.69 9.76 11.98
0.37 0.55 1.1 1.45 1.96 2.79 3.89 5.09 7.11 7.37 9.7 10.7 13.55
0.16 0.38 1.12 1.75 2.21 3.25 4.12 5.02 6.36 7.79 8.65 8.29 10.87
0.12 0.31 1.06 1.51 2.18 3.17 4.19 5.48 6.03 7.56 9.09 9.75 11.52
0.1 0.6 1 1.4 2.2 3.2 4.1 5.2 6.1 7.2 8.5 9.4 11
0.23 0.33 0.84 1.4 2.43 3.11 4.14 5.17 6.07 7.12 8.18 9.03 10.71
0.1 0.6 1.0 1.6 2.2 3.2 4.0 5.6 6.2 6.7 8.0 8.9 11.7
0.2 0.5 0.8 1.3 2.0 2.8 4.1 4.9 6.2 7.0 8.1 9.0 11.1
#####Maturity-at-age###########################################
0 0 0 0.04 0.13 0.45 0.89 0.94 1 1 1 1 1
####Proportion of Natural Mortality Before Spawning############
0.33
####Proportion of Fishing Mortality Before Spawning############
0.1
####Natural Mortality##########################################
0.15
####Recruitment dev weight######################################
1
####F dev weight###############################################
```