aspic: Biomass Dynamic Stock Assessment Model

Laurence Kell
ICCAT

Abstract

The **aspic** package is an implenentation of the ASPIC biomass dynamic stock assessment model in R using the original **FORTRAN** executable. The package provides tools for checking of diagnostics, projections, running Monte Carlo simulation and conducting Management Strategy Evaluation.

Keywords: R, aspic, stock assessment.

1. Introduction

ASPIC is a biomass dynamic model originially implemented as a Fortan executable (Prager et al. 1996). In order to allow it to be simulation tested as part. We do this for ASPIC, a biomass production model Prager et al. [1996] and ?, and discuss how the diagnostics can be applied to a range of models. ASPIC is implemented as a package in R, this allows it to be used with a variety of other packages for plotting, summarising results and to be simulation tested, e.g. as part of the FLR tools for management strategy evaluation Kell et al. [2007].

ASPIC is a biomass dynamic model originially implemented as a Fortan executable (Prager et al. [1996). In order to allow it to be simulation tested as part We do this for ASPIC, a biomass production model Prager et al. [1996] and ?, and discuss how the diagnostics can be applied to a range of models. ASPIC is implemented as a package in R, this allows it to be used with a variety of other packages for plotting, summarising results and to be simulation tested, e.g. as part of the FLR tools for management strategy evaluation Kell et al. [2007].

ASPIC is an biomass dynamic model, which uses age aggregated data, it can also perform projections for different TACs [and Fs?].

2. Inputs

2.1. Files

There are six types of files, i.e.

.bio bootstrap estimates of historic biomass and harvest rate

.prj bootstrapped projections with predicted biomass and harvest rates

.det parameter estimates by bootstrap trial

.inp the input file with data, starting guesses, and run settings and for output

.prb as .bio but with projection results

```
> library(FLAdvice)
> ### Assessments
> ## 1 file
> aspic=readASPIC(paste(dirAspic,"/",scen=scen[1],".bio",sep=""))
> class(aspic)
> names(aspic)
> aspic=readASPIC(paste(dirAspic,"/",scen=scen[1],".bio",sep=""),data.frame=T)
> class(aspic)
> names(aspic)
> ## many files
> aspics=readASPIC(dirAspic,scen=scen,type="b",data.frame=T)
>
```

```
> #### Projections
> ## 1 file
> prj=readASPIC(paste(dirAspic,"/","bumcont1bproj500",".prj",sep=""))
> class(prj)
> names(prj)
> prj=readASPIC(paste(dirAspic,"/","bumcont1bproj500",".prj",sep="",data.frame))
> class(prj)
> names(prj)
> ## many
> prjs=readASPIC(dirAspic,scen=expand.grid(scen=c("bumcont1bproj","bumhighpproj"),TAC=seq(0,6000,500)))
> class(prjs)
> names(prjs)
```

2.2. R

There is an example text data set

```
> dirInp=paste(system.file(package="aspic"),"extdata",sep="/")
> asp=aspic(paste(dirInp,"albn.inp",sep="/"))
> asp=fit(asp)
> key=data.frame(
+   name   =c("Troll Composite CPUE","JLL Old","JLL Modern","CT Old","CT Modern"),
+   series=c("I","I","II","I","II"),
+   flag  =c("OT","JA","JA","CT","CT"),
+   gear  =c("TR","LL","LL","LL","LL"))
> dimnames(key)[[1]]=c("Troll Composite CPUE","JLL Old","JLL Modern","CT Old","CT Modern")
> wts=t(array(c(1,1,1,1,1,
+               1,0,0,0,0,
+               0,1,1,0,0,
+               0,0,0,1,1),c(5,4),list(name=key$name,Scenario=1:4)))
```

```
> cpue=subset(diags(asp),!is.na(obs))[,c("year","name","obs")]
> ggplot(aes(year,obs,group=name,col=name),data=cpue)+
+   geom_point()+
+   stat_smooth()+
+   theme_ms(legend.position="bottom")
```

```
> library(gam)
> gm  =gam(log(obs)~lo(year)+name,data=cpue)
> cpue=data.frame(cpue,gam=predict(gm),gamRsdl=residuals(gm))
> scl =coefficients(gm)[3:9]
> names(scl)=substr(names(scl),5,nchar(names(scl)))
> cpue=transform(cpue,scl=scl[as.character(name)])
> cpue[is.na(cpue$scl),"scl"]=0
> cpue=cbind(cpue,key[cpue$name,])[,-2]
> cpue$name=factor(cpue$name, levels=c("Troll Composite CPUE","JLL Old","JLL Modern","CT Old","CT Modern"))
> ggplot(cpue)+ geom_line(aes(year,exp(gam)),col="red")  +
+               geom_smooth(aes(year,obs),se=FALSE)      +
+               geom_point( aes(year,obs,col=name))       +
+               facet_wrap(~name,ncol=1,scale="free_y")  +
+               theme_ms(legend.position="none")         +
+               xlab("Year") + ylab("Index")
```

```
> uMat=ddply(cpue,.(name),transform, obs=stdz(obs))
> uMat=cast(uMat,year~name,value="obs")
> uMat=uMat[apply(uMat,1,function(x) !all(is.na(x))),]
> pM=plotmatrix(uMat[,-1])
> pM$layers[[2]]=NULL
> mns=ddply(subset(pM$data,!(is.na(x) & !is.na(y))),.(xvar,yvar), function(x) mean(x$y,na.rm=T))
> pM+geom_hline(aes(yintercept=V1),data=mns,col="red") +
+    geom_smooth(method="lm",se=F)  +
+    theme(legend.position="bottom")                   +
+    xlab("Index")+ylab("Index")
```

```
> cr=cor(uMat[,-1],use="pairwise.complete.obs")
> dimnames(cr)=list(gsub("_"," ",names(uMat)[-1]),gsub("_"," ",names(uMat)[-1]))
> cr[is.na(cr)]=0
> corrplot(cr,diag=F,order="hclust",addrect=2)  +
+              theme(legend.position="bottom")
```

3. Assessment

```
> asp=fit(asp)
```

```
> plot(asp)
```

4. Diagnostics

5. Reference Points

6. Fitting

7. Plotting

There are various standard plots, i.e. for fitted time series, reference points and diagnostics. Also using ggplot2 a variety of ad-hoc plots can be produced as required and the packages **diags** and **kobe** can be used for diagnostics and providing plots in Kobe II advice framework.

7.1. CPUE

7.2. Diagnostics

Residuals

Likelihood Profiling

8. Uncertainty

8.1. Bootstrapping

9. Management Procedure

9.1. Reference points

9.2. Projections

9.3. Harvest Control Rules

10. Advice

10.1. Kobe Framework

11. MSE

Affiliation:

Laurence Kell
ICCAT Secretariat
C/Corazón de María, 8.
28002 Madrid
Spain

E-mail: Laurie.Kell@iccat.int