Directions for Inseason Estimates for Taku River Sockeye Salmon Using BTSPAS

Sara Miller

April 2019

Reference for BTSPAS function:
Bonner, S. J. and Schwarz, C. J. (2019). BTSPAS: Bayesian Time Stratified Petersen Analysis System.R package version 2019.01.07.

Project Folder Setup

Set up your Rproject to include code, data, document, figs, and output folders. Make sure the .gitignore file contains output/
figs/
data/
Taku*/
data/prior_weeks_data
so that ouput from inseason estimates is not committed to GitHub. The code files that are necessary to run the analysis are the inseason_analysis.R and the functions.R files.

Data Set Up

Data files will be provided on a weekly basis from the tagging crews and DFO and should be placed in the data folder. These include release data (when fish are released with the following variable names: Year, TagID, ReleaseDate, ReleaseStatWeek (starts on Sunday)), recapture data from DFO (which tags are recovered in commercial catch only with the following names: Year, TagID, RecoveryDate, RecoveryStatWeek (starts on Sunday), RecoveryType only those records with RecoveryType="Commerical" will be used), and commercial catch from DFO (commercial catch excluding recoveries of tagged fish with the following names: Year, Date, StatWeek, CdnCommCt). The recovery type should only include commercial catch. The commercial catch should include the count of tagged fish recovered. It is assumed that the recovery date matches a commerical opening. For example, if a tag is returned after the opening is closed, it is assumed to have occurred during the opening (which is usally in the first half of the week). This is important to allow the half week analysis to work properly.

These three csv files should be the master files, be updated weekly, and placed in the data folder. Old versions of weekly data should be kept in the data/prior_weeks_data folder.

Download the latest version of BTSPAS from: the GitHub site at https://github.com/cschwarz-stat-sfu-ca/ BTSPAS using devtools::install_github("cschwarz-stat-sfu-ca/BTSPAS", dependencies = TRUE, build_vignettes = TRUE) This could take up to 20 minutes. The vignettes take a long time to compile.

Analysis

All packages need to be loaded prior to running models. These pacakges are located at the top of the inseason_analysis.R file. The functions.R file is sourced within the inseason_analysis.R file.

There are a set of functions that need to be run in order. These functions are sourced from the functions.R file.

1. BTSPAS_input Creates the data structures required for BTSPAS for releases This is stratum index, n1, m2, and u2. n1 is the number of tagged releases m2 is a matrix with columns representing recoveries in the same stratum as release, the next stratum of release etc u2 is the number of recoveries commercial = data frame of commercial recoveries. One line per stratum with the total count given missing strata imply a commercial catch of 0 rel.stratum = variable names defining the release strata rec.stratum = variable names defining the recovery strata We assume that character version of stratum numbers wil properly sort, i.e. store as 01 etc stratum.index has index number and stratum label. only those strata that belong to stratum index will be used (to enable you truncate at front/end of study) Check that data frame as the required variables

2. fit.BTSPAS = fits input data to BTSPAS

3. fit.BTSPAS.dropout = fits input data to BTSPAS allowing for dropout/fallback

Inseason Estimates for Taku River

The only two inputs needed for the code to run are the stat weeks and the year. Input these at the top if the inseason_analysis.R code

fw.stat.weeks <- 23:28 # stat weeks with releases and recoveries to be included Year<-2017 #input year fw.stat.weeks <- 23:28 # stat weeks with releases and recoveries to be included

Select the stat weeks for which you want the BTSPAS to provide estimates on a FW and HW basis.Change the above code to account for which weeks the estimate is for.

After reading in the data and doing various merges, the inseason_analysis.R code will create a series of directories in the current workspace that will accumulate as you run the code each week. This code will compute the Full Week (FW), and Half week (HW) stratified with and without dropout. The output files will appear in the output folder as "Taku-FW-Inseason-Wxx-Wxx–YYYY," "Taku-HW-Inseason-Wxx-Wxx–YYYY," "Taku-FW-Inseason-Wxx-Wxx-fallback–YYYY," and "Taku-HW-Inseason-Wxx-Wxx-fallback-YYYY."