

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.

Install/Update R: <https://cran.r-project.org/>

Install/Update R Studio: <https://www.rstudio.com/>

Install JAGS from: <https://sourceforge.net/projects/mcmc-jags/files/latest/download>.

Project Folder Setup

Set up your Rproject to include code, data, document, data/prior_weeks_data, data/yyyy_inseason, and output folders. Make sure the .gitignore file contains:

Taku .csv *.pdf so that output and data 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 Setup

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:

1. release data,
2. recapture data, and
3. commercial catch data.

Release data should include the variable names: Year, TagID, ReleaseDate, and ReleaseStatWeek (which starts on Sunday). Recapture data from DFO should include the variable names: Year, TagID, RecoveryDate, and RecoveryStatWeek (starts on Sunday). Recovery type should only include those records with RecoveryType="Commerical" and TagPrefix="s" for sockeye salmon. 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 usually in the first half of the week). This is important to allow the half week analysis to work properly. The commercial catch data from DFO should include commercial catch including recoveries of tagged fish with the following names: Year, Date, StatWeek, and CdnCommCt. The recovery type should only include commercial sockeye salmon catch. If the fishery was not open or there were zero catches, a '0' should be placed in the cell. All dates should in yyyy-mm-dd format.

These three csv files should be the master files, updated weekly, and then placed in the data folder. Old versions of weekly data should be kept in the data/prior_weeks_data folder. The data files should be 'clean' data and free of NAs or missing information (i.e. missing recovery dates or tag id numbers).

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 because the vignettes take a long time to compile.

Analysis

All packages need to be loaded prior to running models. These packages 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, recoveries, and catch data. This is stratum index, `n1`, `m2`, and `u2`. The variable `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., and `u2` is the number of recoveries.
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, year, and `data.directory` call. Input these at the top of the `inseason_analysis.R` code.

- (1) `fw.stat.weeks <- 23:28` # stat weeks with releases and recoveries to be included
- (2) `Year <- 2019` #input year
- (3) `data.directory <- file.path('data','2019_inseason')` #directory for current year

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 of data will be used in the estimate. There should be a folder in the data folder for `yyyy_inseason`.

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 and Half week stratified with and without dropout. The output files will appear in the output folder as:

“Taku-FullWeek-Inseason-Wxx-Wxx-YYYY,”
“Taku-HalfWeek-Inseason-Wxx-Wxx-YYYY,”
“Taku-FullWeek-Inseason-Wxx-Wxx-fallback-YYYY,” and
“Taku-HalfWeek-Inseason-Wxx-Wxx-fallback-YYYY,” where x is the stat week numbers and YYYY is the year.

In the functions of `fit.BTSPAS` and `fit.BTSPAS.dropout`, if the Canadian commercial sockeye fishery does not commence until 2 or more weeks after sockeye are tagged in the fishwheels, then the “`add.ones.at.start=TRUE`” hack should be implemented in `inseason`. If the ‘hack’ is implemented in the first weeks of estimates, it should be used the entire season (along with postseason). There are four functions in the code that need to be changed.

```
fit.BTSPAS(fw.data,prefix=fw.prefix, add.ones.at.start=FALSE) fit.BTSPAS.dropout(fw.data,prefix=fw.prefix.dropout,  
n=50, dropout=11, add.ones.at.start=FALSE) fit.BTSPAS(hw.data,prefix=hw.prefix, add.ones.at.start=FALSE)  
fit.BTSPAS.dropout(hw.data,prefix=hw.prefix.dropout, n=50, dropout=11, add.ones.at.start=FALSE)
```

The code to implement the inseason estimates are not expanded by fishwheel runtiming.

The weekly inseason estimate should be the median (50%) estimate from the file/folder ‘Taku-FullWeek-Inseason-Wxx-Wxx-fallback-2019’ output to the file ‘inseason-run.siz.csv’.

Dropout rate

Model the year-to-year variation in drop out (including sampling uncertainty) as binomial with $n = 50$ and $x = 11$ which gives $p(\text{dropout}) = 0.22$ and standard deviation of 0.05938262.