Review of the proposed ratio-based method to adjust the scale of annual catch estimates from MRIP-FCAL to GFSC for gag grouper.

by
Robert Ahrens
Pacific Islands Fisheries Science Center
NOAA Fisheries
May 23, 2022

Thank you for the opportunity to provide a review of the ratio-based method for calibrating GRFS and MRIP-FCAL estimates. The following documentation was used to review the proposed approach:

- Tiffanie A. Cross, Colin P. Shea, and Beverly Sauls. 2020. A ratio-based method for calibrating GRFS and MRIP-FCAL estimates of total landings (numbers and pounds of fish), and releases (numbers of fish). In preparation for Red Snapper V Calibration Workshop August 5, 2020
- R code for GRFS-MRIP Calibration. Colin Shea. 4/6/2022
- Datafile. MRIP_TimeSeries_noKeys_lbs_2019fin_new.csv. MRIP-FCAL estimates and variance by month for gag grouper from 1981 to 2019
- Datafile. CalibrationData_YearsOverlap_noKeys_ag_2019fin_new.csv. GRFS and MRIP-FCAL estimates and variance for gag grouper from 2015-2019.
- Cross, Tiffanie A., Colin P. Shea, and Beverly Sauls. 2020. A ratio-based method for calibrating GRFS and MRIP-FCAL estimates of total landings (numbers and pounds of fish), and releases (numbers of fish). SEDAR72-WP-04. SEDAR, North Charleston, SC. 10 pp.
- Calibration Methods: Florida Gulf Reef Fish Survey (GRFS) Beverly Sauls. Florida Fish and Wildlife Conservation Commission Fish and Wildlife Research Institute Presentation for: NOAA Science and Technology Calibration Workshop for Red Snapper August 5, 2020
- Lynne Stokes, Ginny Lesser, and Jean Opsomer. Gulf States Marine Fisheries Commission and Science and Technology Calibration Workshop for Red Snapper. August 6, 2020. Comments on August 5, 2020 Presentations.
- Marine Recreational Information Program (MRIP). Gulf State Recreational Catch and Effort Surveys Transition: A Workshop of the Gulf of Mexico Subgroup of the MRIP Transition Team. Summary Report. February 23-25, 2022. Prepared by Paul Rago
- Mid-Atlantic Fishery Management Council Scientific and Statistical Committee. May 9, 2022

The terms of reference (TOR) for the use of calibrated estimates for stock assessment and management was established April 21, 2022 and this review addresses each point in the TOR. My responses to TOR are in relation to the overall objective of the ratio-based method applied – *"to develop species-specific conversion factors that may be applied to annual, fully calibrated MRIP estimates, and produce a historic time series in the same currency as the GRFS"* -

Provision of "fit for purpose" documentation

The supplied R statistical computing code and the associated data files were sufficient to reproduce the results presented in SEDAR72-WP-04 under an assumed correlation between the data sets of 50%. Results could also be produced assuming 0% correlation using the code and data provided. SEDAR72-WP-04 provides documentation on the calibration procedures are they relate to the use of summary data from the MRIP and GRFS sampling programs. Detail of how summary catch statistic and associated variances are calculated for each program were not provided though not required for the development of the ratio and associated variance. The documentation does not provide a presentation of the calculation used to estimate the variance of the ratios derived for landing in numbers and pounds as well as releases. The provided R script uses the msm[1] package which calculates variances based on the delta method. The delta method is well documented as a first-order Taylor series approximation. Note that one of the recommendations of the statistical consultants from the Calibration Workshop for Red Snapper August 6, 2020 was to use such a statistical package to approximate the variance and this was done with the incorporation of the msm package. The consultants also recommended that a correlation of 0 be assumed between the data sets when calculation the variance of the ratio. The R script provided has incorporated these suggestions.

No description of how the method is intended to be used in future years when new data become available, or how it is expected to be modified was provided in the documentation. The ratio was developed for the sole purpose of calibrating historic MRIP estimates to GRFS and since currently and in to the near future, the GFRS and MRIP surveys will have overlap, the option to update the ratio used is available. In addition, there is ongoing research exploring the differences between the MRIP and GRFS focused on: under-coverage, non-response, sample size and stratification, and cognitive recall. As these studies unfold the results may be incorporated in to the ratio estimate. It is not clear if or how the results from these studies would impact the estimation procedure.

Based on recommendation from the statistical consultants following the 2020 workshop the delta method - a first order Taylor's series approximation – was adopted assuming no correlation between the data sets even though there is some overlap in the data collection. The variance estimated for the ratio is based on data from 2015-2019 and applied back in time to data to 1981. The GFRS started in 2015 making it impossible to understand the temporal dynamics that may have occurred prior to that time point. Certainly, it is possible to quantify the temporal changes in precision from 2015 onward. From 2015 onward a ratio can be estimated at various spatial and temporal scales however the use of the average over the time series available was thought to be the best compromise for the intended hindcasting procedure.

Underlying assumptions for developing, and applying calibrations to the recreational catch time series of landings and discards.

The GRFS was implemented because of a region-wide need for more precise and timely estimates of recreational catch. The GRFS runs concurrent with the MRIP survey in Florida but the estimates that are produced are consistently lower. Although there is work that is ongoing

[1] Christopher H. Jackson (2011). Multi-State Models for Panel Data: The msm Package for R. Journal of Statistical Software, 38(8), 1-29. URL https://www.jstatsoft.org/v38/i08/.

to understand these differences, the timeliness of the GRFS data for in season management and its greater precision the GRFS data will be used in future fisheries decisions. As a result, a method is needed to convert between the two catch streams as well as for the harmonization of disparate catch streams though the Gulf. The data available to do this conversion was 2015-2019, requiring the development of a ratio to convert data prior to 2015. GFRS and MRIP overlapped spatially and temporally during this time period and continue to do so. Ratios were developed for landings in numbers caught and pounds caught, as well as releases were While there is spatial and temporal overlap between the two data sources there is the *fundamental assumption that the ratio and associated variance applied backward in time are representative of the central tendency expected should the GFRS have existed prior to 2015*. It is not possible to know this.

Identify underlying assumptions for development of variance approximations

The delta method was used to approximate the variance for the ratio. This is a first order Taylor series expansion and it is assumed that a linear approximation is sufficient and that higher order terms are not necessary. Because this is a simple ratio this assumption is reasonable and well founded in the statistical literature. While the two surveys overlap spatially and temporally that are drawing form fundamentally different sampling frames and the assumption of no correlation between the data though likely incorrect is warranted until a greater understanding of the relationship between the two data sets has been established. Both of the aspects of the methods were recommended by statistical consultant coming out of the Fifth Red Snapper Workshop (2020) where this approach was presented.

Overall, the approach presented is repeatable, makes reasonable assumption with regards to the approach's objectives, and is similar to what was presented at the Fifth Red Snapper Workshop in 2020 except with the addition of modification recommended by the statistical consultants evaluating the approaches at that time.

Comments on the SRFS/MRIP Validation documentation for GAG

Lynne Stokes (5/26/22)

To prepare this summary, I reviewed the documents provided by the Florida team (Cross, Shea, & Sauls) to respond to the terms of reference to document the calibration method developed for Gag. First, the team responded to the major recommendation of the previous consultant review by amending the variance estimation method as suggested. Specifically, they the previous suggestion involved their method to approximate the variance of a ratio of two random variables. They used a delta method approximation, which is standard and well understood. However, the approximation requires knowledge of the correlation between the two random variables. This, however, is unknown, since there is only one realization of each random variable, and therefore no way to estimate a correlation. (Note, the replicated values from month to month are not an iid sample, so it is unclear how to use them to estimate a correlation.) Previously, they assumed an arbitrary value of 0.5 for the correlation. The previous review suggested a better approach was to assume an arbitrary value of 0 correlation, since such an assumption would at least provide a conservative (biased upward) value for variance of the ratio. So they did make that correction in the revision.

However, there are still a few concerns.

1. The method they propose appears only in the form of documentation in R code. The R code calls on functions that I am unfamiliar with, but on reading appears to make a numerical approximation to a delta method. When the calculation needed is as simple as delta method approximation for the ratio of two independent random variables ($V(\frac{Y}{X}) \approx \frac{\mu_y^2}{\mu_x^2}\left(\frac{\sigma_y^2}{\mu_y^2}+\frac{\sigma_x^2}{\mu_x^2}\right)$), one should provide the formula that is used and confirm that the function is computing it properly. However, I know nothing about how well that function works and none is offered. I suggest that using an R function when a simple formula is available should not be used as documentation.

In fact, I suggest that an actual documentation report outlining the steps of the calculation should be provided, along with illustration of the computations.

2. The documentation also states that a correlation of 0 as "assumed" for the next step, which was to approximate the variance of the product of the calibration factor and each historical FCAL estimate of total. Actually, this is not an assumption at all, but rather the actual correlation, since there should be no correlation due to the fact that the samples from the two terms of the product are temporally unrelated (don't overlap).

Further, there is no reason to use a delta method approximation for the product of two random variables, since the exact variance of the product can be written in closed form. (See Goodman 1962, *Journal of the American Statistical Association*, 57, 54-60.) This variance, when the two estimators are independent, can be estimated as

$$v(\hat{t}_y\hat{R}) = \hat{R}^2 v(\hat{t}_y) + v(\hat{R})[\hat{t}_y^2 - v(\hat{t}_y)].$$

I don't know if the R function deltamethod they use provides the same value as this calculation or not, but I suggest that be confirmed.

3. Finally, there is a notation that NA is replaced with 0. I am not sure what NA represented. Does it mean that the estimate itself was 0 and the variance estimate was 0? There were several months when the FCAL estimate = 0 with a variance of 0 and the GRFS estimate is not 0. This clearly indicates a problem with the GRFS estimate, which cannot be improved when backcasting with a multiplicative calibration factor (since it will still be a zero). So I believe it would be better to exclude those months from the calibration exercise. In other words, the calibration factor will only be used (or make any difference) for those months when the FCAL estimate is non-zero. So I suggest including only such months in the calibration calculation.

```
# CleanX <-
read.table("C:/Users/rachel.germeroth/Documents/GRFS/GRFS_License_2020_20200401_Cle
an.txt",
#                       header=T, sep="\t", quote=NULL, comment="", na.strings=T,
fill=T)
# ExtraVars <-
read.table("D:/FDM_new/NFWF/LocalSurveyProcessing/GRFS_License_20211201_ExtraVars.t
xt",
#                           header=T, sep="\t", quote=NULL, comment="",
na.strings=T, fill=T, check.names = FALSE)
#llowEscapes = TRUE, fill=T, )
# ExtraVars <-
read.table("G:/DATA/ANGLER/Reef_fish/NFWF/GRFS/EffortSurvey/2021_11/GRFS_License_20
211201_ExtraVars.txt",
#                            header=T, sep="\t", quote=NULL, comment="",
na.strings=T, fill=T)
# ExtraVars <-
read.table("C:/Users/rachel.germeroth/Documents/GRFS/GRFS_License_20200401_ExtraVar
s.txt",
#                          header=T, sep="\t", quote=NULL, comment="", na.strings=T,
fill=T)
# ExtraVars <- subset(ExtraVars, select=-c(customerCounty))
ExtraVars$AgentBusinessName <- gsub("[[:punct:]]", " ",
ExtraVars$AgentBusinessName)
Errors <-
read.table("D:/FDM_new/NFWF/LocalSurveyProcessing/GRFS_License_20220501_Errors.txt"
,
header=T, sep="\t", quote=NULL, comment="", na.strings=T, fill=T)
# Errors <-
read.table("C:/Users/rachel.germeroth/Documents/GRFS/GRFS_License_20200401_Errors.t
xt",
#                        header=T, sep="\t", quote=NULL, comment="", na.strings=T,
fill=T)
#
# Errors <-
read.table("G:/DATA/ANGLER/Reef_fish/NFWF/GRFS/EffortSurvey/2021_11/GRFS_License_20
211201_Errors.txt",
#            header=T, sep="\t", quote=NULL, comment="", na.strings=T, fill=T)
#ZipCounties <-
read.table("C:/Users/tiffanie.cross/Documents/FDM_new/NFWF/LocalSurveyProcessing/FL
zipcounties.txt",
#                         header=T, sep="\t", quote=NULL, comment="", na.strings=T,
fill=T)
#ZipCounties <-
read.table("C:/Users/rachel.germeroth/Documents/GRFS/FLzipcounties.txt",
#                    header=T, sep="\t", quote=NULL, comment="", na.strings=T,
fill=T)
ZipCounties <-
read.table("G:/DATA/ANGLER/Reef_fish/NFWF/GRFS/EffortSurvey/_R_&_SAS_Scripts/FLzipc
ounties.txt",
```

```
header=T, sep="\t", quote=NULL, comment="", na.strings=T, fill=T)
ZipCounties <- subset(ZipCounties, select=c("mailZip5", "mailCounty"))
ZipCounties$mailCounty <- toupper(ZipCounties$mailCounty)
##Merge Clean data with extra variables
Clean <- merge(CleanX, ExtraVars, by="customerId", all.x=T)
Clean$customerCounty <- toupper(Clean$customerCounty)
#Parse out zipcodes in Clean file
Clean <- separate(Clean, mailZip, into = c("mailZip5","mailZip4"), sep = "-", extra
= "merge")
#bring in counties by zipcode
Clean1 <- merge(Clean, ZipCounties, by="mailZip5", all.x=T)
Clean1 <- Clean1[order(Clean1$mailState),]
Clean1$customerCounty <- ifelse(as.character(Clean1$mailState) != "FL",
"Out-of-State",
ifelse(is.na(Clean1$customerCounty), "Out-of-State",
as.character(Clean1$customerCounty)))
Clean1$customerCounty <- ifelse((as.character(Clean1$customerCounty)=="" &
as.character(Clean1$mailState)=="FL"), as.character(Clean1$mailCounty),
as.character(Clean1$customerCounty))
#Create flag for Coastal
Clean1$Coastal <- with(Clean1, ifelse(customerCounty %in% c("ESCAMBIA","SANTA
ROSA","OKALOOSA","WALTON","HOLMES",
"WASHINGTON","JACKSON","CALHOUN","BAY","GULF",
"GADSDEN", "LIBERTY","FRANKLIN","LEON","WAKULLA",
"JEFFERSON","TAYLOR","DIXIE","LEVY","CITRUS",
"HERNANDO","PASCO","PINELLAS","HILLSBOROUGH","MANATEE",
"SARASOTA","CHARLOTTE","LEE","COLLIER","MONROE"), 1,
ifelse(customerCounty %in% c("MIAMI-DADE","BROWARD","PALM BEACH","MARTIN","ST
LUCIE", "SAINT LUCIE",
"INDIAN RIVER","BREVARD","VOLUSIA","FLAGLER","ST JOHNS", "SAINT JOHNS",
"DUVAL","NASSAU"),2,0)))
#Create flag for Region (PAN, NOR, SOU, KEY)
Clean1$Region <- with(Clean1, ifelse(mailState=="FL" &
customerCounty %in% c("ESCAMBIA","SANTA ROSA","OKALOOSA","WALTON","HOLMES",
"WASHINGTON","JACKSON","CALHOUN","BAY","GULF"), "PAN",
ifelse(mailState=="FL" &
customerCounty %in% c("GADSDEN","LIBERTY","FRANKLIN","WAKULLA","LEON",
"JEFFERSON","MADISON","TAYLOR","HAMILTON","SUWANNEE",
"LAFAYETTE","DIXIE","COLUMBIA","GILCHRIST","LEVY",
"BAKER","UNION","BRADFORD","ALACHUA","MARION",
"NASSAU","DUVAL","CLAY","ST JOHNS","SAINT JOHNS", "PUTNAM",
"FLAGLER","VOLUSIA","CITRUS","HERNANDO","PASCO",
"SUMTER","LAKE","SEMINOLE","ORANGE","OSCEOLA",
"BREVARD","INDIAN RIVER"), "NOR",
ifelse(mailState=="FL" &
customerCounty %in% c("PINELLAS","HILLSBOROUGH","POLK","MANATEE","SARASOTA",
"HARDEE","DESOTO","HIGHLANDS","OKEECHOBEE","ST LUCIE","SAINT LUCIE",
"MARTIN","CHARLOTTE","LEE","GLADES","HENDRY",
"PALM BEACH","COLLIER","BROWARD"), "SOU",
ifelse(mailState=="FL" & customerCounty %in% c("MONROE", "MIAMI-DADE"), "KEY",
```

```
NA)))))
#Perform a check on the states listed in the mailState vector.
Clean1$stateValid <- with(Clean1, ifelse(mailState %in%
c("AL","AK","AZ","AR","CA","CO","CT","DE","FL","GA",
"HI","ID","IL","IN","IA","KS","KY","LA","ME","MD","MA","MI","MN","MS",
"MO","MT","NE","NV","NH","NJ","NM","NY","NC","ND","OH","OK","OR","PA",
"RI","SC","SD","TN","TX","UT","VT","VA","WA","WV","WI","WY","DC"), 1, 0))
####Assign strata based on region, coastal and boat vectors
Clean1$Strata <- ifelse(Clean1$Region=="PAN" & Clean1$Coastal=="1" &
Clean1$Boat=="1", 111,
ifelse(Clean1$Region=="PAN" & Clean1$Coastal=="1" & Clean1$Boat=="0", 110,
ifelse(Clean1$Region=="NOR" & Clean1$Coastal=="1" & Clean1$Boat=="1", 211,
ifelse(Clean1$Region=="NOR" & Clean1$Coastal=="1" & Clean1$Boat=="0", 210,
ifelse(Clean1$Region=="NOR" & Clean1$Coastal=="0" & Clean1$Boat=="1", 201,
ifelse(Clean1$Region=="NOR" & Clean1$Coastal=="0" & Clean1$Boat=="0", 200,
ifelse(Clean1$Region=="NOR" & Clean1$Coastal=="2" & Clean1$Boat=="1", 221,
ifelse(Clean1$Region=="NOR" & Clean1$Coastal=="2" & Clean1$Boat=="0", 220,
ifelse(Clean1$Region=="SOU" & Clean1$Coastal=="1" & Clean1$Boat=="1", 311,
ifelse(Clean1$Region=="SOU" & Clean1$Coastal=="1" & Clean1$Boat=="0", 310,
ifelse(Clean1$Region=="SOU" & Clean1$Coastal=="0" & Clean1$Boat=="1", 301,
ifelse(Clean1$Region=="SOU" & Clean1$Coastal=="0" & Clean1$Boat=="0", 300,
ifelse(Clean1$Region=="SOU" & Clean1$Coastal=="2" & Clean1$Boat=="1", 321,
ifelse(Clean1$Region=="SOU" & Clean1$Coastal=="2" & Clean1$Boat=="0", 320,
ifelse(Clean1$Region=="KEY" & Clean1$Coastal=="1" & Clean1$Boat=="1", 411,
ifelse(Clean1$Region=="KEY" & Clean1$Coastal=="1" & Clean1$Boat=="0", 410,
ifelse(Clean1$Region=="KEY" & Clean1$Coastal=="2" & Clean1$Boat=="1", 421,
ifelse(Clean1$Region=="KEY" & Clean1$Coastal=="2" & Clean1$Boat=="0", 420,
NA))))))))))))))))))
Clean1$Strata <-with(Clean1, ifelse(is.na(Region) & mailState %in% c("GA", "AL"),
500,
ifelse(is.na(Region) & !mailState %in% c("GA", "AL"), 600,
ifelse(stateValid=="0", 999, Strata))))
#Create incomplete address flag to flag address with incomplete parts of the
address
Clean1$incAddress <- ifelse(Clean1$mailStreet1=="" | Clean1$mailStreet1=="0" |
Clean1$mailStreet1=="1" | Clean1$mailCity=="" | Clean1$mailState=="" |
(is.na(Clean1$mailZip5)),1,0)
test <- subset(Clean1, incAddress==1)
remove(test)
#####Flag errors####
Errors$errors <- 1
#Subset error dataset by only those columns needed
Errors <- subset(Errors, select=c("customerId", "errors"))
#Merge Clean address data with error flags
Clean3 <- merge(Clean1, Errors, by="customerId", all.x=T)
#Set the badaddress flag if the error column has a flag
Clean3$badAddress <- ifelse(Clean3$errors==1,1,Clean3$badAddress)
Clean3 <- Clean3[order(Clean3$mailStreet1, Clean3$mailCity, Clean3$mailState),]
####Create unique household identifier####
Clean3$householdID <- ifelse((Clean3$incAddress!=1 | Clean3$badAddress!=1),
```

```
(vec_group_id(Clean3[c("mailStreet1", "mailCity", "mailState")])),NA)
Clean3 <- Clean3[order(Clean3$householdID, -Clean3$Boat),]
# Clean3 <- Clean3 %>% group_by(householdID) %>% mutate(Boat = max(Boat))
#Subset just the necessities for household boat ownership assignment processing -
makes it faster
BoatHH <- subset(Clean3, select = c("customerId", "Boat", "Strata", "householdID"))
# test <- test[1:1000,]
#Assign household boat owner flag
BoatHH <- ddply(BoatHH, .(householdID), mutate,
HHBoat=max(Boat))
BoatHHx <- subset(BoatHH, select=c("customerId", "HHBoat"))
#Merge household boat owner data back to original data by customerId
Clean4 <- merge(Clean3, BoatHHx, by="customerId", all.x=T)
# #How many folks were mislabeled as non-boat people?
# test <- subset(Clean4, Boat!=BoatHH)
# remove(test)
#Assign Household strata
####Assign strata based on region, coastal and boat vectors
####library(dplyr)
# dat <- data.frame(HouseholdID = rep(1:5,each = 3), BoatOwner =
c(rep(0,6),1,0,0,1,rep(0,5)))
# dat <- dat %>% group_by(HouseholdID) %>% mutate(BoatOwner = max(BoatOwner))
# Clean4$HHBoat <- as.character(Clean4$HHBoat)
# Clean4$stateValid <- as.character(Clean4$stateValid)
# Clean4 <- Clean4[order(Clean4$householdID),]
Clean4$HHStrata <- ifelse(Clean4$Region=="PAN" & Clean4$Coastal=="1" &
Clean4$HHBoat=="1", 111,
ifelse(Clean4$Region=="PAN" & Clean4$Coastal=="1" & Clean4$HHBoat=="0", 110,
ifelse(Clean4$Region=="NOR" & Clean4$Coastal=="1" & Clean4$HHBoat=="1", 211,
ifelse(Clean4$Region=="NOR" & Clean4$Coastal=="1" & Clean4$HHBoat=="0", 210,
ifelse(Clean4$Region=="NOR" & Clean4$Coastal=="0" & Clean4$HHBoat=="1", 201,
ifelse(Clean4$Region=="NOR" & Clean4$Coastal=="0" & Clean4$HHBoat=="0", 200,
ifelse(Clean4$Region=="NOR" & Clean4$Coastal=="2" & Clean4$HHBoat=="1", 221,
ifelse(Clean4$Region=="NOR" & Clean4$Coastal=="2" & Clean4$HHBoat=="0", 220,
ifelse(Clean4$Region=="SOU" & Clean4$Coastal=="1" & Clean4$HHBoat=="1", 311,
ifelse(Clean4$Region=="SOU" & Clean4$Coastal=="1" & Clean4$HHBoat=="0", 310,
ifelse(Clean4$Region=="SOU" & Clean4$Coastal=="0" & Clean4$HHBoat=="1", 301,
ifelse(Clean4$Region=="KEY" & Clean4$Coastal=="2" & Clean4$HHBoat=="1", 421,
ifelse(Clean4$Region=="SOU" & Clean4$Coastal=="0" & Clean4$HHBoat=="0", 300,
ifelse(Clean4$Region=="SOU" & Clean4$Coastal=="2" & Clean4$HHBoat=="1", 321,
ifelse(Clean4$Region=="SOU" & Clean4$Coastal=="2" & Clean4$HHBoat=="0", 320,
ifelse(Clean4$Region=="KEY" & Clean4$Coastal=="1" & Clean4$HHBoat=="1", 411,
ifelse(Clean4$Region=="KEY" & Clean4$Coastal=="1" & Clean4$HHBoat=="0", 410,
ifelse(Clean4$Region=="KEY" & Clean4$Coastal=="2" & Clean4$HHBoat=="0", 420,
NA))))))))))))))))))
test <- subset(Clean4, HHStrata=="111" | HHStrata=="110")
Clean4$HHStrata <-with(Clean4, ifelse(is.na(Region) & mailState %in% c("GA", "AL"),
500,
ifelse(is.na(Region) & !mailState %in% c("GA", "AL"), 600,
ifelse(stateValid=="0", 999, HHStrata))))
```

```
test1<- subset(Clean4, prior==1 | MAIL_RMVE==1)
# Clean4$prior <- NULL
# # class(Clean4$prior)
# # Clean4$prior <- as.numeric(Clean4$prior)
#
# ####Get Prior Records####
# #Connect to the FDM-GRFS database
# GRFS <- odbcConnect("SPSD1_FDM_GRFS", uid="FDMsamplers", pwd="Monsterfish2013")
# #Query data to a dataframe
# PriorsBads <- sqlQuery(GRFS, paste("SELECT [CUSTOMERID] AS customerId,
[SURVEYMONTH], [SURVEYYEAR], [BADADDRESS] AS badAddress FROM
FDM_GRFS.dbo.LU_SURVEY_CUSTOMER
#                                    WHERE ([SURVEYYEAR] = 2021 AND [SURVEYMONTH] =
09) OR ([SURVEYYEAR] = 2021 AND [SURVEYMONTH] = 10) OR BADADDRESS = 1"))
#
# MailRMV <- sqlQuery(GRFS, paste("SELECT [CUSTOMERID] AS customerId,
[SURVEYMONTH], [SURVEYYEAR], [MAIL_RMVE] FROM FDM_GRFS.dbo.SURVEYRESPONSE
#                                 WHERE [MAIL_RMVE] = 1"))
#
# CRprior <-
read.table("G:/DATA/ANGLER/Reef_fish/SRFS/EffortSurvey/Tests/CognitiveBias_Test/Dra
w Files/CognitiveRecall_DRAW_07080910_2021_priors.txt",
#                       header=T, sep="\t", quote=NULL, comment="", na.strings=T,
fill=T)
#
#
#
# PriorsBads <- merge(PriorsBads, MailRMV, by=c("customerId", "SURVEYYEAR",
"SURVEYMONTH"), all=T)
# PriorsBads <- merge(PriorsBads, CRprior, by=c("customerId", "SURVEYYEAR",
"SURVEYMONTH"), all=T)
# # [SURVEYMONTH] = 03 OR [SURVEYMONTH] = 04 OR
#
# #Create flag to mark prior customers selected for survey
# PriorsBads$prior <- ifelse((PriorsBads$SURVEYYEAR==2021 &
PriorsBads$SURVEYMONTH==09), 1,
#                             ifelse((PriorsBads$SURVEYYEAR==2021 &
PriorsBads$SURVEYMONTH==10), 1, 0))
# PriorsBads <- PriorsBads[order(-PriorsBads$prior),]
#
# #Order the data by customerId ascending, and SURVEYMONTH descending.
# PriorsBads <- PriorsBads[order(PriorsBads$customerId, -PriorsBads$SURVEYMONTH),]
#
# #Remove duplicate customers leaving the most recent survey month in the data
# PriorsBads_uniq <- subset(PriorsBads, !duplicated(customerId))
# PriorsBads_dups <- subset(PriorsBads, duplicated(customerId))
#
#
# #Subset the data for the customerId, and prior and badaddress flags in
preparation for data merge
```

```
# Prior <- subset(PriorsBads_uniq, select=c("customerId", "prior", "badAddress",
"MAIL_RMVE"))
#
#
# #Merge the Prior and Badaddress data with the survey pool data to identify folks
who either have bad
# #addresses listed or who have received a survey in the prior two months
# Clean5 <- merge(Clean4, Prior, all.x=TRUE)
write.table(Clean4,
"D:/FDM_new/NFWF/LocalSurveyProcessing/GRFS_License_20220501_FINAL.txt",
sep="\t", row.names=F)
View(Survey1)
View(Permit)
Permit <- read.table("D:/FDM_new/NFWF/Permit Data/Raw
Data/GulfReefWithSaltwaterLicenseByDate_202203.txt",
header=T, sep="\t", quote="", na.strings=T, fill=T)
View(Permit)
View(Permit)
####Permit Data Processing#############################################################
#Read in permit data
Permit <- read.table("D:/FDM_new/NFWF/Permit Data/Raw
Data/GulfReefWithSaltwaterLicenseByDate_202203.txt",
header=T, sep=",", quote="", na.strings=T, fill=T)
View(Permit)
View(Permit)
Permit <- read.table("D:/FDM_new/NFWF/Permit Data/Raw
Data/GulfReefWithSaltwaterLicenseByDate_202204.txt",
header=T, sep=",", quote="", na.strings=T, fill=T)
View(Permit)
Permit <- read.table("D:/FDM_new/NFWF/Permit Data/Raw
Data/GulfReefWithSaltwaterLicenseByDate_202204.txt",
header=T, sep="/t", quote="", na.strings=T, fill=T)
####Permit Data Processing#############################################################
#Read in permit data
Permit <- read.table("D:/FDM_new/NFWF/Permit Data/Raw
Data/GulfReefWithSaltwaterLicenseByDate_202204.txt",
header=T, sep="\t", quote="", na.strings=T, fill=T)
View(Permit)
# PriorsBads_dups <- subset(PriorsBads, duplicated(customerId))
#
#
# #Subset the data for the customerId, and prior and badaddress flags in
preparation for data merge
# Prior <- subset(PriorsBads_uniq, select=c("customerId", "prior", "badAddress",
"MAIL_RMVE"))
#
#
# #Merge the Prior and Badaddress data with the survey pool data to identify folks
who either have bad
# #addresses listed or who have received a survey in the prior two months
```

```
# Clean5 <- merge(Clean4, Prior, all.x=TRUE)
Clean5 <- Clean4[order(-length(Clean4$lastName),]
# PriorsBads_dups <- subset(PriorsBads, duplicated(customerId))
#
#
# #Subset the data for the customerId, and prior and badaddress flags in
preparation for data merge
# Prior <- subset(PriorsBads_uniq, select=c("customerId", "prior", "badAddress",
"MAIL_RMVE"))
#
#
# #Merge the Prior and Badaddress data with the survey pool data to identify folks
who either have bad
# #addresses listed or who have received a survey in the prior two months
# Clean5 <- merge(Clean4, Prior, all.x=TRUE)
Clean5 <- Clean4[order(-length(Clean4$lastName)),]
View(Clean5)
Clean4$lastNamelength <- length(Clean4$lastName)
View(Clean4)
# PriorsBads_dups <- subset(PriorsBads, duplicated(customerId))
#
#
# #Subset the data for the customerId, and prior and badaddress flags in
preparation for data merge
# Prior <- subset(PriorsBads_uniq, select=c("customerId", "prior", "badAddress",
"MAIL_RMVE"))
#
#
# #Merge the Prior and Badaddress data with the survey pool data to identify folks
who either have bad
# #addresses listed or who have received a survey in the prior two months
# Clean5 <- merge(Clean4, Prior, all.x=TRUE)
Clean5 <- Clean4[order(-str_length(Clean4$lastName)),]
View(Clean5)
write.table(Clean4,
"D:/FDM_new/NFWF/LocalSurveyProcessing/GRFS_License_20220501_FINAL.txt",
sep="\t", row.names=F)
View(Clean5)
Clean4$lastNamelength <- str_length(Clean4$lastName)
View(Clean4)
# PriorsBads_dups <- subset(PriorsBads, duplicated(customerId))
#
#
# #Subset the data for the customerId, and prior and badaddress flags in
preparation for data merge
# Prior <- subset(PriorsBads_uniq, select=c("customerId", "prior", "badAddress",
"MAIL_RMVE"))
#
#
# #Merge the Prior and Badaddress data with the survey pool data to identify folks
```

```
who either have bad
# #addresses listed or who have received a survey in the prior two months
# Clean5 <- merge(Clean4, Prior, all.x=TRUE)
Clean5 <- Clean4[order(-str_length(Clean4$lastName)),]
View(Clean5)
setwd("G:/DATA/ANGLER/Reef_fish/NFWF/GRFS/Calibration/Calibration_noKEYS/Calibratio
n_noKeys_GAG_FINAL_20220503")
# ---
#   title: "GRFS-MRIP Calibration"
# author: "Colin Shea (FWC/FWRI ~ colin.shea@myfwc.com)"
# date: "4/6/2022"
# output: html_document
# ---
#   * Process for calculating calibrated GRFS estimates and PSEs for focal species
(common) and variables (varname: Landings (no. fish), Releases (no. fish), Landings
(pounds))
#
# (1) For each species and variable, sum monthly GRFS and FCAL totals and variances
across overlapping years (2015-2019)
# (2) Calculate ratios as GRFS/FCAL and use delta method to calculate ratio SEs
# (3) Apply ratios to historical FCAL totals and variances that have been summed
across waves (i.e., annual totals)
# (4) Calculate calibrated GRFS estimates as GRFS_H = FCAL*ratio and use delta
method and associated variances to calculate SEs
# (5) Calculate GRFS_H PSEs as SE/GRFS_H*100
#
# #### Consultant (L Stokes, G Lessner, J Opsomer) comments from 8/6/20 report are
as follows:
#
# Floridaâs presentation team provided a summary of their proposed calibration
method
# that is the ratio of averages method. We believe the approach is reasonable.
#
# + One difference between Floridaâs method and others is the separate treatment of
catch
# and released fish. That is, separate calibration ratios for these two components
are
# calculated, and the pounds calibration factor is for catch only. This differs
from some
# other states, who produce a combined calibration factor. We do not have input on
# which method is most reasonable, but do urge consistency among the states.
#
# + We do not believe the variance estimation method for the calibration ratio is
# adequately justified. This is not a problem at all for setting ACLâs in the new
âcurrencyâ,
# but if it is to be used for any purpose, we are not willing to endorse the method
as
# correct. We have two suggestions that we believe would be an improvement.
#
# - A simple way to estimate the variance is to use ratio estimation software, such
```

```
as survey in R or surveymeans in SAS. Both will calculate a standard error estimate
for a ratio, even when numerator and denominator are correlated. As part of the
variance estimation process, the software estimates correlation between the two
series, so no assumptions need to be made for it. This is not quite the correct
method either, since the ratio estimation function assumes that the individual
âobservationsâ are from the same population, and therefore have the same variance,
which these do not.
#
# - A conservative approach is to assume the correlation is 0. This will produce a
standard error that is larger than it should be, but at least you know its bias. In
contrast, when a correlation value such as 0.5 is selected, the bias can be either
positive or negative, i.e. the estimator can be either too small or too large.
Still, we believe this is a more defensible approach than simply picking a
correlation to use with no justification.
#
# + *_FL Team NOTE 1_*: Our interpretation of these two suggestions was that our
existing approach for calculating ratio SEs (delta method) was appropriate provided
we used the more conservation assumption of 0 correlation between the GRFS and FCAL
surveys. As such, our decision was to keep our existing method and use a
correlation of 0.
#
# + *_FL Team NOTE 2_*: Please see end of document for comparison of delta method
to an alternative, and in the end equivalent, approach for estimating ratio
variances.
#### Load required packages
library(tidyverse) # for data manipulation
library(msm) # for delta method calculation of ratio and hindcasted standard errors
##### Assumed correlation strength between GRFS and FCAL (for ratio estimation).
It's set up to write a CSV results file that uses the specified rho in the name. We
have assumed a correlation of 0.
rho <- 0.0
##### (1) _GRFS and FCAL OVERLAP DATA: 2015-2019_ Read in the GRFS data (2015 -
2019; years that overlap with MRIP), convert NAs to 0, and sum monthly GRFS and
FCAL/MRIP means and variances across YEARS for each varname and species
grfs <- read.csv("CalibrationData_YearsOverlap_noKeys_ag_2019fin_new.csv")
grfs[is.na(grfs)] <- 0
grfs$varname <- factor(grfs$varname, levels = c("landing", "RELEASE", "wgt_ab1"),
labels = c("Landings (no. fish)", "Releases (no. fish)", "Landings (pounds)"))
```

```
# ---
#   title: "GRFS-MRIP Calibration"
# author: "Colin Shea (FWC/FWRI ~ colin.shea@myfwc.com)"
# date: "4/6/2022"
# output: html_document
# ---
#   * Process for calculating calibrated GRFS estimates and PSEs for focal species
(common) and variables (varname: Landings (no. fish), Releases (no. fish), Landings
(pounds))
#
# (1) For each species and variable, sum monthly GRFS and FCAL totals and variances
across overlapping years (2015-2019)
# (2) Calculate ratios as GRFS/FCAL and use delta method to calculate ratio SEs
# (3) Apply ratios to historical FCAL totals and variances that have been summed
across waves (i.e., annual totals)
# (4) Calculate calibrated GRFS estimates as GRFS_H = FCAL*ratio and use delta
method and associated variances to calculate SEs
# (5) Calculate GRFS_H PSEs as SE/GRFS_H*100
#
# #### Consultant (L Stokes, G Lessner, J Opsomer) comments from 8/6/20 report are
as follows:
#
# Florida's presentation team provided a summary of their proposed calibration
method
# that is the ratio of averages method. We believe the approach is reasonable.
#
# + One difference between Florida's method and others is the separate treatment of
catch
# and released fish. That is, separate calibration ratios for these two components
are
# calculated, and the pounds calibration factor is for catch only. This differs
from some
# other states, who produce a combined calibration factor. We do not have input on
# which method is most reasonable, but do urge consistency among the states.
#
# + We do not believe the variance estimation method for the calibration ratio is
# adequately justified. This is not a problem at all for setting ACL's in the new
"currency",
# but if it is to be used for any purpose, we are not willing to endorse the method
as
# correct. We have two suggestions that we believe would be an improvement.
#
# - A simple way to estimate the variance is to use ratio estimation software, such
as survey in R or surveymeans in SAS. Both will calculate a standard error estimate
for a ratio, even when numerator and denominator are correlated. As part of the
variance estimation process, the software estimates correlation between the two
series, so no assumptions need to be made for it. This is not quite the correct
method either, since the ratio estimation function assumes that the individual
"observations" are from the same population, and therefore have the same variance,
which these do not.
```

```
#
# - A conservative approach is to assume the correlation is 0. This will produce a
standard error that is larger than it should be, but at least you know its bias. In
contrast, when a correlation value such as 0.5 is selected, the bias can be either
positive or negative, i.e. the estimator can be either too small or too large.
Still, we believe this is a more defensible approach than simply picking a
correlation to use with no justification.
#
# + *_FL Team NOTE 1_*: Our interpretation of these two suggestions was that our
existing approach for calculating ratio SEs (delta method) was appropriate provided
we used the more conservation assumption of 0 correlation between the GRFS and FCAL
surveys. As such, our decision was to keep our existing method and use a
correlation of 0.
#
# + *_FL Team NOTE 2_*: Please see end of document for comparison of delta method
to an alternative, and in the end equivalent, approach for estimating ratio
variances.

#### Load required packages
library(tidyverse) # for data manipulation
library(msm) # for delta method calculation of ratio and hindcasted standard errors

##### Assumed correlation strength between GRFS and FCAL (for ratio estimation).
It's set up to write a CSV results file that uses the specified rho in the name. We
have assumed a correlation of 0.
rho <- 0.0

##### (1) _GRFS and FCAL OVERLAP DATA: 2015-2019_ Read in the GRFS data (2015 -
2019; years that overlap with MRIP), convert NAs to 0, and sum monthly GRFS and
FCAL/MRIP means and variances across YEARS for each varname and species
grfs <- read.csv("CalibrationData_YearsOverlap_noKeys_ag_2019fin_new.csv")
grfs[is.na(grfs)] <- 0
grfs$varname <- factor(grfs$varname, levels = c("landing", "RELEASE", "wgt_ab1"),
labels = c("Landings (no. fish)", "Releases (no. fish)", "Landings (pounds)"))
grfs_fcal <- grfs %>% group_by(varname, common) %>% summarise(grfs_sm =
sum(grfs_sum), grfs_vr = sum(grfs_var), fcal_sm = sum(fcal_sum), fcal_vr =
sum(fcal_var))

##### (2) Use `deltamethod` function in the R package msm to calculate the SE of
the ratio of 2015-2019 GRFS/FCAL assuming correlation of rho between the two
surveys.
grfs_fcal$obsRatio <- grfs_fcal$grfs_sm/grfs_fcal$fcal_sm # calculate observed
ratio: grfs/fcal
grfs_fcal$varRatio <- NA # empty column
grfs_fcal$covRatio <- NA # empty column
R <- matrix(c(1,rho,rho,1), ncol = 2) # correlation matrix: no correlation
for (i in 1:nrow(grfs_fcal)){
  grfs_s <- grfs_fcal$grfs_sm[i] # summed grfs mean for
species/month/release/landing i
  grfs_v <- grfs_fcal$grfs_vr[i] # summed grfs variance for
```

```
species/year/release/landing i
  fcal_s <- grfs_fcal$fcal_sm[i] # summed fcal mean across years for each month for
row i
  fcal_v <- grfs_fcal$fcal_vr[i] # summed fcal variance across years for each month
for row i
  sigma <- c(sqrt(grfs_v),sqrt(fcal_v)) # grfs and acal SDs
  C <- diag(sigma)%*%R%*%diag(sigma) # VC matrix (matrix multiplication returns
variance on diagonal and covariances on off diagonal)
  grfs_fcal$varRatio[i] <- deltamethod(~x1/x2, mean = c(grfs_s,fcal_s), cov = C,
ses = F)
  grfs_fcal$covRatio[i] <- C[1,2] # covariances (o for 0 correlation)
}

##### Look at results
grfs_fcal

##### (3) _MRIP-FCAL: 1981 - 2019_ Read in MRIP-FCAL data (1982 - 2019), convert
NAs to 0, re-order and rename variables of interest, and then calculate summed
totals and variances across variables, species, and years: this is used for
hindcasting after we calcuate ratios and ratio variances.
cal <- read.csv("MRIP_TimeSeries_noKeys_lbs_2019fin_new.csv", stringsAsFactors = F,
na.strings = c("."))
cal[is.na(cal)] <- 0
cal$varname <- factor(cal$varname, levels = c("landing", "RELEASE", "wgt_ab1"),
labels = c("Landings (no. fish)", "Releases (no. fish)", "Landings (pounds)"))

##### (3, cont'd) Now sum up FCAL totals and variances, join with ratios and ratio
variances, then calculate GRFS_H (obs ratio*FCAL total)
calSumVar <-  cal %>% select(year, common, varname, fcal_sum, fcal_var) %>%
group_by(year, varname, common) %>% summarise(total_sum = sum(fcal_sum), total_var
= sum(fcal_var)) %>% ungroup() %>% mutate(varname = factor(varname), common =
factor(common)) %>% group_by(year, varname, common) %>% full_join(grfs_fcal, by =
c("varname", "common")) %>% mutate(grfsH = total_sum*obsRatio) %>% select(year,
varname, common, total_sum, grfs_sm, grfs_vr, fcal_sm, fcal_vr, obsRatio, varRatio,
covRatio, grfsH, total_var) %>% mutate(deltaSD = NA)

##### (4) Use `deltamethod` function to calculate SE of the product of the
grfs/fcal ratio and the historical FCAL annual totals: We have again assume a
correlation of 0.
Rd <- matrix(c(1,0,0,1), ncol = 2) # correlation matrix: 0 correlation
for (j in 1:nrow(calSumVar)){
  sigmad <- c(sqrt(calSumVar$varRatio[j]),sqrt(calSumVar$total_var[j])) # ratio and
fcal SDs
  Cd <- diag(sigmad)%*%Rd%*%diag(sigmad) # VC matrix (matrix multiplication returns
variance on diagonal and covariances on off diagonal)
  obsRatio <- calSumVar$obsRatio[j]
  total_sum <- calSumVar$total_sum[j]
  calSumVar$deltaSD[j] <- deltamethod(~x1*x2, mean = c(obsRatio,total_sum), cov =
Cd, ses = T)
}
```

```
##### (5) Calculate annual PSEs for hindcasted GRFS estimates
calSumVarFIN <- calSumVar %>% mutate(method = paste("GRFS-FCAL-DELTA", rho*100),
pse = deltaSD/grfsH*100) %>% arrange(varname, common, year)
calSumVarFIN

##### Write results to csv
write.csv(calSumVarFIN, paste0("calPlotFINAL_FCAL_DELTA_NO_KEYS_",rho*100,".csv"),
row.names = F)

##### For estimating the variance of a ratio, formula (18) from
http://www.stat.cmu.edu/~hseltman/files/ratio.pdf, an approximation method that is
apparently (trying to get a copy of the ebook) also covered in
https://link.springer.com/book/10.1007%2F978-0-387-35099-8, simplifies to the
formula below when independence is assumed (no correlation so the covariance term =
0), and the result is identical to what we get using the delta method.

# + var(X/Y) = [var(X) / (mean(Y)^2)] + [var(Y)*(mean(X)^2) / mean(Y)^4)]
#
# + var(GRFS/FCAL) = [var(GRFS) / (mean(FCAL)^2)] + [var(FCAL)*(mean(GRFS)^2) /
mean(FCAL)^4)]

grfs_fcal$varRatio2 <- (grfs_fcal$grfs_vr/(grfs_fcal$fcal_sm^2)) +
(grfs_fcal$fcal_vr*(grfs_fcal$grfs_sm^2))/grfs_fcal$fcal_sm^4

(data.frame(delta_var = grfs_fcal$varRatio, ratio_var = grfs_fcal$varRatio2))
```

---
title: "SRFS-MRIP Calibration"
author: "Colin Shea, Tiffanie Cross, and Beverly Sauls (FWC/FWRI;
colin.shea@myfwc.com, tiffanie.cross@myfwc.com, beverly.sauls@myfwc.com)"
date: "4/29/2022"
output: html_document
---
* Process for calculating calibrated GRFS (now SRFS) estimates and PSEs for focal species (common) and variables (varname: Landings (no. fish), Releases (no. fish), Landings (pounds))

(1) For each species and variable, sum monthly GRFS and FCAL totals and variances across overlapping years (2015-2019)
(2) Calculate ratios as GRFS/FCAL and use delta method to calculate ratio SEs
(3) Apply ratios to historical FCAL totals and variances that have been summed across waves (i.e., annual totals)
(4) Calculate calibrated GRFS estimates as GRFS_H = FCAL*ratio and use delta method and associated variances to calculate SEs
(5) Calculate GRFS_H PSEs as SE/GRFS_H*100

Consultant (L Stokes, G Lessner, J Opsomer) comments from 8/6/20 report are as follows:

Florida’s presentation team provided a summary of their proposed calibration method that is the ratio of averages method. We believe the approach is reasonable.

+ One difference between Florida’s method and others is the separate treatment of catch
and released fish. That is, separate calibration ratios for these two components are
calculated, and the pounds calibration factor is for catch only. This differs from some
other states, who produce a combined calibration factor. We do not have input on which method is most reasonable, but do urge consistency among the states.

+ We do not believe the variance estimation method for the calibration ratio is adequately justified. This is not a problem at all for setting ACL’s in the new “currency”,
but if it is to be used for any purpose, we are not willing to endorse the method as
correct. We have two suggestions that we believe would be an improvement.

  - A simple way to estimate the variance is to use ratio estimation software, such as survey in R or surveymeans in SAS. Both will calculate a standard error estimate for a ratio, even when numerator and denominator are correlated. As part of the variance estimation process, the software estimates correlation between the two series, so no assumptions need to be made for it. This is not quite the correct method either, since the ratio estimation function assumes that the individual “observations” are from the same population, and therefore have the same variance, which these do not.

- A conservative approach is to assume the correlation is 0. This will produce a standard error that is larger than it should be, but at least you know its bias. In contrast, when a correlation value such as 0.5 is selected, the bias can be either positive or negative, i.e. the estimator can be either too small or too large. Still, we believe this is a more defensible approach than simply picking a correlation to use with no justification.

+ *_FL Team NOTE 1_*: Our interpretation of these two suggestions was that our existing approach for calculating ratio SEs (delta method) was appropriate provided we used the more conservation assumption of 0 correlation between the GRFS and FCAL surveys. As such, our decision was to keep our existing method and use a correlation of 0.

+ *_FL Team NOTE 2_*: Please see end of document for comparison of delta method to an alternative, and in the end equivalent, approach for estimating ratio variances.

Load required packages
```{r warning=F, message=FALSE}
library(tidyverse) # for data manipulation
library(msm) # for delta method calculation of ratio and hindcasted standard errors
```
Assumed correlation strength between GRFS and FCAL (for ratio estimation). It's set up to write a CSV results file that uses the specified rho in the name. We have assumed a correlation of 0.
```{r warning=F, message=FALSE}
rho <- 0.0
```
(1) _GRFS and FCAL OVERLAP DATA: 2015-2019_ Read in the GRFS data (2015 - 2019; years that overlap with MRIP), convert NAs to 0, and sum monthly GRFS and FCAL/MRIP means and variances across YEARS for each varname and species.
```{r warning=F, message=FALSE}
grfs <- read.csv("CalibrationData_YearsOverlap_noKeys_ag_2019fin_new.csv")
grfs[is.na(grfs)] <- 0
grfs$varname <- factor(grfs$varname, levels = c("landing", "RELEASE", "wgt_ab1"),
labels = c("Landings (no. fish)", "Releases (no. fish)", "Landings (pounds)"))
grfs_fcal <- grfs %>% group_by(varname, common) %>% summarise(grfs_sm =
sum(grfs_sum), grfs_vr = sum(grfs_var), fcal_sm = sum(fcal_sum), fcal_vr =
sum(fcal_var))
```
(2) Use `deltamethod` function in the R package msm to calculate the SE of the ratio of 2015-2019 GRFS/FCAL assuming correlation of rho between the two surveys.
```{r warning=F, message=FALSE}
grfs_fcal$obsRatio <- grfs_fcal$grfs_sm/grfs_fcal$fcal_sm # calculate observed
ratio: grfs/fcal
grfs_fcal$varRatio <- NA # empty column
grfs_fcal$covRatio <- NA # empty column
R <- matrix(c(1,rho,rho,1), ncol = 2) # correlation matrix: no correlation
```

````
for (i in 1:nrow(grfs_fcal)){
  grfs_s <- grfs_fcal$grfs_sm[i] # summed grfs mean for
species/month/release/landing i
  grfs_v <- grfs_fcal$grfs_vr[i] # summed grfs variance for
species/year/release/landing i
  fcal_s <- grfs_fcal$fcal_sm[i] # summed fcal mean across years for each month for
row i
  fcal_v <- grfs_fcal$fcal_vr[i] # summed fcal variance across years for each month
for row i
  sigma <- c(sqrt(grfs_v),sqrt(fcal_v)) # grfs and acal SDs
  C <- diag(sigma)%*%R%*%diag(sigma) # VC matrix (matrix multiplication returns
variance on diagonal and covariances on off diagonal)
  grfs_fcal$varRatio[i] <- deltamethod(~x1/x2, mean = c(grfs_s,fcal_s), cov = C,
ses = F)
  grfs_fcal$covRatio[i] <- C[1,2] # covariances (o for 0 correlation)
}
```
##### Look at results
```{r warning=F, message=FALSE}
grfs_fcal
```
##### (3) _MRIP-FCAL: 1981 - 2019_ Read in MRIP-FCAL data (1982 - 2019), convert
NAs to 0, re-order and rename variables of interest, and then calculate summed
totals and variances across variables, species, and years: this is used for
hindcasting after we calcuate ratios and ratio variances.
```{r warning=F, message=FALSE}
cal <- read.csv("MRIP_TimeSeries_noKeys_lbs_2019fin_new.csv", stringsAsFactors = F,
na.strings = c("."))
cal[is.na(cal)] <- 0
cal$varname <- factor(cal$varname, levels = c("landing", "RELEASE", "wgt_ab1"),
labels = c("Landings (no. fish)", "Releases (no. fish)", "Landings (pounds)"))
```
##### (3, cont'd) Now sum up FCAL totals and variances, join with ratios and ratio
variances, then calculate GRFS_H (obs ratio*FCAL total)
```{r warning=F, message=FALSE}
calSumVar <-  cal %>% select(year, common, varname, fcal_sum, fcal_var) %>%
group_by(year, varname, common) %>% summarise(total_sum = sum(fcal_sum), total_var
= sum(fcal_var)) %>% ungroup() %>% mutate(varname = factor(varname), common =
factor(common)) %>% group_by(year, varname, common) %>% full_join(grfs_fcal, by =
c("varname", "common")) %>% mutate(grfsH = total_sum*obsRatio) %>% select(year,
varname, common, total_sum, grfs_sm, grfs_vr, fcal_sm, fcal_vr, obsRatio, varRatio,
covRatio, grfsH, total_var) %>% mutate(deltaSD = NA)
```
##### (4) Use `deltamethod` function to calculate SE of the product of the
grfs/fcal ratio and the historical FCAL annual totals: We have again assume a
correlation of 0.
```{r warning=F, message=FALSE}
Rd <- matrix(c(1,0,0,1), ncol = 2) # correlation matrix: 0 correlation
for (j in 1:nrow(calSumVar)){
  sigmad <- c(sqrt(calSumVar$varRatio[j]),sqrt(calSumVar$total_var[j])) # ratio and
````

fcal SDs
  Cd <- diag(sigmad)%*%Rd%*%diag(sigmad) # VC matrix (matrix multiplication returns variance on diagonal and covariances on off diagonal)
  obsRatio <- calSumVar$obsRatio[j]
  total_sum <- calSumVar$total_sum[j]
  calSumVar$deltaSD[j] <- deltamethod(~x1*x2, mean = c(obsRatio,total_sum), cov = Cd, ses = T)
}
```
##### (5) Calculate annual PSEs for hindcasted GRFS estimates
```{r warning=F, message=FALSE}
calSumVarFIN <- calSumVar %>% mutate(method = paste("GRFS-FCAL-DELTA", rho*100), pse = deltaSD/grfsH*100) %>% arrange(varname, common, year)
calSumVarFIN

```
Write results to csv
```{r warning=F, message=FALSE}
write.csv(calSumVarFIN, paste0("calPlotFINAL_FCAL_DELTA_NO_KEYS_",rho*100,".csv"), row.names = F)
```

For estimating the variance of a ratio, formula (18) from http://www.stat.cmu.edu/~hseltman/files/ratio.pdf, an approximation method that is apparently (trying to get a copy of the ebook) also covered in https://link.springer.com/book/10.1007%2F978-0-387-35099-8, simplifies to the formula below when independence is assumed (no correlation so the covariance term = 0), and the result is identical to what we get using the delta method.

+ var(X/Y) = [var(X) / (mean(Y)^2)] + [var(Y)*(mean(X)^2) / mean(Y)^4)]

+ var(GRFS/FCAL) = [var(GRFS) / (mean(FCAL)^2)] + [var(FCAL)*(mean(GRFS)^2) / mean(FCAL)^4)]

```{r}
grfs_fcal$varRatio2 <- (grfs_fcal$grfs_vr/(grfs_fcal$fcal_sm^2)) + (grfs_fcal$fcal_vr*(grfs_fcal$grfs_sm^2))/grfs_fcal$fcal_sm^4

(data.frame(delta_var = grfs_fcal$varRatio, ratio_var = grfs_fcal$varRatio2))
```
```

varname	common	year	month	grfs_sum	grfs_var	fcal_sum	fcal_var
landing	GAG	2015	5	825.35	679162	0	0
landing	GAG	2015	6	446.71	167656.2	439	192761
landing	GAG	2015	7	17360.62	37831681	60856	2.74E+08
landing	GAG	2015	8	6650.53	5151081	10762	45851183
landing	GAG	2015	9	4376.61	3237299	12102	71829121
landing	GAG	2015	10	17611.02	92569114	24063	2.24E+08
landing	GAG	2015	11	100794.6	1.36E+09	153782	3.52E+09
landing	GAG	2015	12	788.91	172665.1	0	0
landing	GAG	2016	1	0	0	0	0
landing	GAG	2016	2	0	0	0	0
landing	GAG	2016	3	0	0	0	0
landing	GAG	2016	4	5270.21	11965115	2465	3078653
landing	GAG	2016	5	767.39	186999.8	5114	16110639
landing	GAG	2016	6	18510.98	46082419	37729	50552567
landing	GAG	2016	7	5482.53	8720892	54390	2.26E+09
landing	GAG	2016	8	2464.99	2968889	0	0
landing	GAG	2016	9	2685.21	1502590	11876	49061215
landing	GAG	2016	10	8340.49	16373047	1589	1272859
landing	GAG	2016	11	11267.55	52387023	21641	77199810
landing	GAG	2016	12	25646.1	94269262	59298	1.96E+08
landing	GAG	2017	1	0	0	0	0
landing	GAG	2017	2	0	0	0	0
landing	GAG	2017	3	0	0	0	0
landing	GAG	2017	4	7740.14	19691704	7089	37477140
landing	GAG	2017	5	9337.11	51203334	39620	1.42E+09
landing	GAG	2017	6	25129.01	55696411	65119	8.31E+08
landing	GAG	2017	7	4409.02	5942416	16108	1.12E+08
landing	GAG	2017	8	3248.91	2274429	3834	5570669
landing	GAG	2017	9	3032.3	6302798	8425	48259180
landing	GAG	2017	10	20474.57	68760685	7533	6032769
landing	GAG	2017	11	14747.08	22789685	41603	3.65E+08
landing	GAG	2017	12	10176.86	13502985	64590	5.48E+08
landing	GAG	2018	1	0	0	0	0
landing	GAG	2018	2	0	0	0	0
landing	GAG	2018	3	0	0	0	0
landing	GAG	2018	4	1691	337158	4261	0
landing	GAG	2018	5	799	237050	1396	1002324
landing	GAG	2018	6	21612	23692974	68482	6.12E+08
landing	GAG	2018	7	35626	43697843	61415	46175359
landing	GAG	2018	8	3754	5723725	57502	1.52E+09
landing	GAG	2018	9	7363	10209481	22676	1.62E+08
landing	GAG	2018	10	4863	7362135	2263	5120610
landing	GAG	2018	11	41	557	32048	9.47E+08
landing	GAG	2018	12	15355	18692042	30007	2.92E+08
landing	GAG	2019	1	0	0	0	0
landing	GAG	2019	2	271	12039	0	0

landing	GAG	2019	3	93	9327	0	0
landing	GAG	2019	4	12268	28925712	0	0
landing	GAG	2019	5	11128	14959798	11151	51723247
landing	GAG	2019	6	24068	43726529	81842	7.52E+08
landing	GAG	2019	7	13785	88656401	41490	5.85E+08
landing	GAG	2019	8	1491	703010	8297	36969589
landing	GAG	2019	9	6196	18827515	2656	4158398
landing	GAG	2019	10	4799	4376878	6059	19265212
landing	GAG	2019	11	7402	6699339	59334	2.88E+09
landing	GAG	2019	12	9325	15120837	9151	18331112
RELEASE	GAG	2015	5	59175.79	3.03E+08	85684	8.6E+08
RELEASE	GAG	2015	6	51457.42	2.66E+08	100080	1.78E+09
RELEASE	GAG	2015	7	15122.61	28488510	71483	5.2E+08
RELEASE	GAG	2015	8	38005.94	1.78E+08	69034	4.83E+08
RELEASE	GAG	2015	9	28469.08	1.09E+08	71113	1.25E+09
RELEASE	GAG	2015	10	57224.01	3.23E+08	131012	9.46E+08
RELEASE	GAG	2015	11	122233.8	2.42E+09	227754	5.39E+09
RELEASE	GAG	2015	12	82806.67	1.05E+09	31149	2.17E+08
RELEASE	GAG	2016	1	23362.67	1.69E+08	32816	3.26E+08
RELEASE	GAG	2016	2	35784.78	2.57E+08	40899	8.1E+08
RELEASE	GAG	2016	3	119306.1	1.52E+09	90341	6.46E+08
RELEASE	GAG	2016	4	94348.59	7.57E+08	112127	8.74E+08
RELEASE	GAG	2016	5	66541.96	7.1E+08	454315	1.15E+11
RELEASE	GAG	2016	6	55023.36	2.8E+08	114303	8.7E+08
RELEASE	GAG	2016	7	70026.72	6.32E+08	166898	2.03E+09
RELEASE	GAG	2016	8	40782.39	2.6E+08	43442	1.39E+08
RELEASE	GAG	2016	9	33579.21	1.64E+08	89916	2.39E+08
RELEASE	GAG	2016	10	77481.35	6.8E+08	71338	6.26E+08
RELEASE	GAG	2016	11	98014.12	7.59E+08	251236	6.59E+09
RELEASE	GAG	2016	12	73554.93	5.52E+08	167881	2.5E+09
RELEASE	GAG	2017	1	54552.63	2.24E+08	34652	1.41E+08
RELEASE	GAG	2017	2	62566.41	4.52E+08	166124	4.66E+09
RELEASE	GAG	2017	3	86354.71	8.76E+08	91883	1.49E+09
RELEASE	GAG	2017	4	121731.8	1.5E+09	145506	4.99E+09
RELEASE	GAG	2017	5	80023.02	4.89E+08	229010	1.17E+10
RELEASE	GAG	2017	6	83613.34	4.78E+08	371025	2.63E+10
RELEASE	GAG	2017	7	36487.7	1.73E+08	116204	1.24E+09
RELEASE	GAG	2017	8	59145.09	3.52E+08	100742	1.19E+09
RELEASE	GAG	2017	9	32425.96	1.07E+08	61727	2.77E+08
RELEASE	GAG	2017	10	164604.6	2E+09	204977	1.13E+09
RELEASE	GAG	2017	11	185604.8	3.32E+09	642535	6.36E+10
RELEASE	GAG	2017	12	125457.3	1.14E+09	784909	8.54E+10
RELEASE	GAG	2018	1	27809	1.1E+08	23592	36132150
RELEASE	GAG	2018	2	98568	1.35E+09	204460	5.04E+09
RELEASE	GAG	2018	3	45746	3.79E+08	188822	3.35E+09
RELEASE	GAG	2018	4	136351	3.16E+09	145589	6.71E+09
RELEASE	GAG	2018	5	60115	4.02E+08	95373	1.16E+09

RELEASE	GAG	2018	6	149353	7.52E+08	446806	3.91E+10
RELEASE	GAG	2018	7	82767	8.12E+08	196066	7.29E+09
RELEASE	GAG	2018	8	13675	26156662	92440	2.6E+09
RELEASE	GAG	2018	9	26380	95046272	92590	1.71E+09
RELEASE	GAG	2018	10	68421	5.15E+08	59817	3.98E+08
RELEASE	GAG	2018	11	614	189196	62512	6.88E+08
RELEASE	GAG	2018	12	100995	1.36E+09	326583	3.81E+10
RELEASE	GAG	2019	1	26719	59889430	19159	1656312
RELEASE	GAG	2019	2	30952	96696027	41533	17774781
RELEASE	GAG	2019	3	58586	6.58E+08	65146	2.46E+09
RELEASE	GAG	2019	4	29641	43989658	23709	1.96E+08
RELEASE	GAG	2019	5	95392	4.56E+08	105775	3.91E+09
RELEASE	GAG	2019	6	109209	4.43E+08	383763	8.54E+09
RELEASE	GAG	2019	7	127056	7.1E+09	377163	4.77E+10
RELEASE	GAG	2019	8	19251	86161765	89142	9.71E+08
RELEASE	GAG	2019	9	93565	2.07E+09	85975	1.16E+09
RELEASE	GAG	2019	10	43786	69547895	81406	1.05E+08
RELEASE	GAG	2019	11	68563	3.32E+08	272769	1.49E+10
RELEASE	GAG	2019	12	80524	6.47E+08	140452	1.51E+09
wgt_ab1	GAG	2015	5	8486.55	32570513	0	0
wgt_ab1	GAG	2015	6	2553.93	2531694	2536	2917124
wgt_ab1	GAG	2015	7	127854.2	9.37E+08	452608	6.4E+09
wgt_ab1	GAG	2015	8	41560.54	1.02E+08	71668	1.04E+09
wgt_ab1	GAG	2015	9	27579.79	52605635	63950	8.2E+08
wgt_ab1	GAG	2015	10	126407.2	2.59E+09	183626	6.46E+09
wgt_ab1	GAG	2015	11	889002.9	2.97E+10	1454252	1.50E+11
wgt_ab1	GAG	2015	12	4266.86	2512148	0	0
wgt_ab1	GAG	2016	1	0	0	0	0
wgt_ab1	GAG	2016	2	0	0	0	0
wgt_ab1	GAG	2016	3	0	0	0	0
wgt_ab1	GAG	2016	4	32483.7	1.68E+08	21523	1.06E+08
wgt_ab1	GAG	2016	5	6397.62	5632106	37589	3.49E+08
wgt_ab1	GAG	2016	6	141866.2	1.22E+09	342175	2.49E+09
wgt_ab1	GAG	2016	7	45138.36	3.32E+08	516536	9.45E+10
wgt_ab1	GAG	2016	8	15792.07	52299252	0	0
wgt_ab1	GAG	2016	9	24869.39	65473213	121192	2.41E+09
wgt_ab1	GAG	2016	10	57979.29	3.16E+08	19484	1.05E+08
wgt_ab1	GAG	2016	11	92205.67	1.52E+09	189677	2.55E+09
wgt_ab1	GAG	2016	12	236898.8	3.69E+09	546100	8.83E+09
wgt_ab1	GAG	2017	1	0	0	0	0
wgt_ab1	GAG	2017	2	0	0	0	0
wgt_ab1	GAG	2017	3	0	0	0	0
wgt_ab1	GAG	2017	4	59943.9	5.13E+08	49343	9.41E+08
wgt_ab1	GAG	2017	5	77427.3	1.43E+09	314854	3.97E+10
wgt_ab1	GAG	2017	6	200944.9	1.64E+09	476166	2.07E+10
wgt_ab1	GAG	2017	7	42462.32	2.25E+08	162457	4.47E+09
wgt_ab1	GAG	2017	8	36506.67	1.43E+08	33157	1.66E+08

wgt_ab1	GAG	2017	9	22431.51	1.52E+08	67831	1.25E+09
wgt_ab1	GAG	2017	10	163934.2	2.11E+09	56458	1.53E+08
wgt_ab1	GAG	2017	11	132630.9	7.72E+08	394773	1.54E+10
wgt_ab1	GAG	2017	12	89590.02	4.67E+08	635351	2.84E+10
wgt_ab1	GAG	2018	1	0	0	0	0
wgt_ab1	GAG	2018	2	0	0	0	0
wgt_ab1	GAG	2018	3	0	0	0	0
wgt_ab1	GAG	2018	4	13704	7888601	37325	0
wgt_ab1	GAG	2018	5	7171	7236974	12214	30134417
wgt_ab1	GAG	2018	6	205207	1.09E+09	582619	1.94E+10
wgt_ab1	GAG	2018	7	305905	1.5E+09	459945	1.02E+09
wgt_ab1	GAG	2018	8	28740	1.38E+08	455488	4.11E+10
wgt_ab1	GAG	2018	9	57006	2.42E+08	182331	3.77E+09
wgt_ab1	GAG	2018	10	36565	1.49E+08	18957	1.63E+08
wgt_ab1	GAG	2018	11	394	23366	284658	3.34E+10
wgt_ab1	GAG	2018	12	136804	6.01E+08	279326	1.01E+10
wgt_ab1	GAG	2019	1	0	0	0	0
wgt_ab1	GAG	2019	2	1542	176667	0	0
wgt_ab1	GAG	2019	3	0	0	0	0
wgt_ab1	GAG	2019	4	129654	1.6E+09	0	0
wgt_ab1	GAG	2019	5	101842	5.21E+08	80631	1.27E+09
wgt_ab1	GAG	2019	6	224787	1.61E+09	701529	2.69E+10
wgt_ab1	GAG	2019	7	100415	1.77E+09	434270	3.93E+10
wgt_ab1	GAG	2019	8	12438	23599894	76691	1.52E+09
wgt_ab1	GAG	2019	9	54576	7.72E+08	24995	1.73E+08
wgt_ab1	GAG	2019	10	35018	82583583	43126	3.39E+08
wgt_ab1	GAG	2019	11	67274	3.09E+08	588938	1.28E+11
wgt_ab1	GAG	2019	12	75619	4.43E+08	71685	5.95E+08