Salvage Genetic Loss Data QA/QC

Brian Mahardja

2023-01-17

## Salvage Data

A copy of salvage database query data was sent by Vanessa Gusman (CDFW) in December 2022. The file is titled **“ChinookLoss\_20221215.csv”**. This file is compared to the genetic data and the loss numbers contained within. When there are discrepancies in loss numbers between the two datasheets (salvage and genetic), the loss numbers from the salvage database should be considered the “correct” one. The salvage dataset is also used to ensure that every fish listed in the genetic datasheet does exist in the salvage database and therefore can be calculated for loss properly.

# Load salvage database sent by Vanessa Gusman  
  
salvage\_data <- read.csv(file = file.path("data\_input/ChinookLoss\_20221215.csv"))  
salvage\_data$SampleDateTime<-as.POSIXct(paste(salvage\_data$Date, salvage\_data$Time), format="%m/%d/%Y %H:%M:%S", tz="America/Los\_Angeles")  
  
#Keep data with non-existent date and time  
salvage\_data\_wrong\_time <- salvage\_data %>% filter(is.na(SampleDateTime))  
  
#Change date to the proper format  
salvage\_data$SampleDate<-as.Date(salvage\_data$Date, format = "%m/%d/%Y")  
#Change time to the proper format  
salvage\_data<-salvage\_data %>% mutate(SampleTime=as.POSIXct(paste("2022-10-01", Time), format="%Y-%m-%d %H:%M:%S", tz="America/Los\_Angeles")) %>%  
 mutate(SampleTime=format(SampleTime, "%H:%M:%S"))

### Assign unique ID to every fish and minor issues with dataset

The CDFW salvage database would combine data from multiple fish that were of the same length and captured within the same site, date, and time. This would make comparison with the genetic data problematic because the genetic dataset contains a single row/line for every individual fish. The code below assigns a “duplicate ID” to ensure that a single fish in the genetic dataset isn’t counted twice and vice versa. Consequently, loss and expanded salvage numbers were divided by the number of actual fish observed at salvage to get individual fish loss and salvage numbers.

# Prep salvage data  
  
#Rename columns to make it easier to work in R and divide Loss + Expanded Salvage by nfish  
salvage\_data\_adjusted<- salvage\_data %>%   
 dplyr::mutate(Facility =case\_when(Facility==1 ~ "SWP",  
 Facility==2 ~ "CVP")) %>%  
   
 #Remove erroneous sample date and time, ignore for now  
 #filter(!is.na(SampleDateTime)) %>%  
 #Remove single sample with NA count   
 filter(!is.na(Count)) %>%  
 #Remove adclip fish  
 filter(AdClip == FALSE) %>%  
 #Ensure that data for salvage and loss are for a single fish  
 mutate(Salvage=Salvage/Count, Encounter=Encounter/LengthFrequency, Entrain=Entrain/LengthFrequency, Release=Release/LengthFrequency, Loss=Loss/LengthFrequency)  
  
#Multiply rows by nfish  
salvage\_data\_adjusted<- setDT(expandRows(salvage\_data\_adjusted, "LengthFrequency"))   
  
salvage\_data\_adjusted<- salvage\_data\_adjusted%>%  
 # build grouping by combination of variables  
 dplyr::group\_by(SampleDateTime, Race, FL, Facility) %>%  
 # add row number which works per group due to prior grouping  
 dplyr::mutate(duplicateID = dplyr::row\_number()) %>%  
 # ungroup to prevent unexpected behaviour down stream  
 dplyr::ungroup()

One notable finding through this QA/QC process is that the CDFW salvage database contains date and time that technically do not exist. The list below are date and time in which daylight savings time starts in spring and 2:00 AM for these dates should not exist because time should have been moved forward to 3:00 AM. For the purpose of this effort, we’re keeping these dates and time because the errors were propagated into the genetic datasheet.

## Date Time  
## 1 4/7/1996 2:00:00  
## 4 4/6/1997 2:00:00  
## 21 4/4/1999 2:00:00  
## 24 4/1/2001 2:00:00  
## 40 4/7/2002 2:00:00  
## 43 4/6/2003 2:00:00  
## 46 4/4/2004 2:00:00  
## 47 4/3/2005 2:00:00  
## 52 3/8/2009 2:00:00  
## 54 3/13/2011 2:00:00  
## 56 3/11/2018 2:00:00

## Genetic Data

The latest clean version of the genetic datasheet for all genetic winter-run was sent by Kevin Reece on January 13, 2023. This version is sent after one final QA/QC process was done on January 12 and a new version was created on January 13. The file is titled **“WY1996-2022 Genetically Confirmed WR\_CVP and SWP salvage and loss for OMR analysis\_20230113Final.xlsx”**. Unique “duplicate ID” was assigned to fish in order to ensure that multiple fish of the same length collected at the same location, date, and time are treated separately.

### Read Kevin Reece's Jan 13, 2023 genetic data  
  
genetic\_data <-read\_excel(file.path("data\_input/WY1996-2022 Genetically Confirmed WR\_CVP and SWP salvage and loss for OMR analysis\_20230113Final.xlsx"), sheet = "WY1996-2022 JUVWR Genetic\_ID", range = cell\_cols("A:K")) %>%   
 rename(SampleDate='Sample Date',SampleTime='Sample Time',FL=Forklength,Facility=Loc,Salvage\_KevinReece=Salvage,Loss\_KevinReece=Loss,LAD\_KevinReece='Model Race',GeneticAssignment=Assignment) %>%  
 mutate(SampleTime=format(SampleTime, "%H:%M:%S")) %>% mutate(SampleDate=as.Date(SampleDate)) %>%  
 mutate(SampleDateTime=as.POSIXct(paste(SampleDate, SampleTime), format="%Y-%m-%d %H:%M:%S", tz="America/Los\_Angeles")) %>% select(-Catch)  
  
  
genetic\_data\_expanded<- genetic\_data %>%  
 #adjust time zone  
 mutate(SampleDateTime=force\_tz(SampleDateTime, tz="America/Los\_Angeles")) %>%  
 mutate(WaterYear=as.numeric(WaterYear)) %>%  
 # build grouping by combination of variables  
 dplyr::group\_by(SampleDateTime, Facility, FL) %>%  
 # add row number which works per group due to prior grouping  
 dplyr::mutate(duplicateID = dplyr::row\_number()) %>%  
 # ungroup to prevent unexpected behaviour down stream  
 dplyr::ungroup()

## QA/QC process

Genetic and salvage datasets were combined and I checked to see if there were any individuals in the genetic datasheet that remain unpaired with the salvage database. There were NO individuals that remain unpaired with the salvage database (i.e., everything in the genetic dataset matched).

##################### Combine the salvage and genetic data sets  
combined\_data<-left\_join(salvage\_data\_adjusted,genetic\_data\_expanded)

## Joining, by = c("WaterYear", "Facility", "FL", "SampleDateTime", "SampleDate",  
## "SampleTime", "duplicateID")

# All data are matched up to the salvage database  
unpaired\_genetic\_data<-full\_join(salvage\_data\_adjusted,genetic\_data\_expanded) %>% filter(is.na(Salvage))

## Joining, by = c("WaterYear", "Facility", "FL", "SampleDateTime", "SampleDate",  
## "SampleTime", "duplicateID")

count(unpaired\_genetic\_data)/count(genetic\_data)

## n  
## 1 0

### Loss calculation errors

Loss numbers were calculated in the genetic datasheet; however, I wanted to ensure that the loss numbers between my “paired” dataset matched with what is in the genetic datasheet.

#Double check that numbers are correct for matched data  
paired\_genetic\_data <- combined\_data %>% filter(!is.na(ArchiveID)) %>%  
 #specify 2 decimals for loss  
 mutate(Loss=as.numeric(format(round(Loss, 2)))) %>%  
 mutate(Loss\_KevinReece=as.numeric(format(round(Loss\_KevinReece, 2)))) %>%  
 #Check that loss calculations match up for both genetic and salvage data sets  
 mutate(QAQC\_Loss\_Difference=abs(Loss-Loss\_KevinReece))

Even if I did not count differences in loss calculations due to rounding errors (difference less than 0.05), there were 2.1532 % of the data where loss numbers did not match. Summing all the differences in these data rows, in total, there was a loss discrepancy of 274.05 between the two datasets.

#Percent correct  
count(paired\_genetic\_data %>% filter(abs(QAQC\_Loss\_Difference)<0.05))/count(paired\_genetic\_data)\*100

## n  
## 1 97.84677

#Total difference in loss calculation  
sum(paired\_genetic\_data$QAQC\_Loss\_Difference)

## [1] 274.05

## Final, cleaned dataset

Given the discrepancy in loss calculations between the two datasets, here I create a finalized dataset which contain all the relevant information regarding these genetically confirmed winter-run Chinook Salmon collected at the salvage facilities. My recommendation would be to use the loss that was calculated in the salvage database (column: Loss\_Salvage). “QAQC\_Loss\_Difference” column was calculated as the absolute value of difference in loss between the salvage dataset and genetic dataset. The final dataset is titled **“Paired\_Genetic\_Data\_Loss\_Comparison\_2023-01-17.csv”**.

#Remove certain columns and rename some to make the dataset easier to understand and use  
paired\_genetic\_data\_final<-paired\_genetic\_data %>% select(-Date,-Month,-MonthDay,-Time,-Count) %>%  
 rename(Loss\_GeneticData=Loss\_KevinReece,Loss\_SalvageData=Loss,LAD\_GeneticData=LAD\_KevinReece)  
  
#Print out csv  
write.csv(paired\_genetic\_data\_final,file=file.path("output/Paired\_Genetic\_Data\_Loss\_Comparison\_2023-01-17.csv"),row.names = F)

For the purpose of calculating loss of genetic winter-run relative to JPE each year, I’ve also summed up the loss for each year and compared them to the JPE. This file is titled: **“JPE\_Genetic\_Loss\_Comparison\_2023-01-17.csv”**

# Read JPE file  
jpe\_data <- read.csv(file = file.path("data\_input/Winter-run\_JPE.csv")) %>% rename(WaterYear=WY)  
  
annual\_salvage<-salvage\_data\_adjusted %>% group\_by(WaterYear) %>% filter(Race=="W") %>% summarise(WinterRun\_LAD\_Loss=sum(Loss))  
  
annual\_salvage\_genetic<-paired\_genetic\_data\_final %>% group\_by(WaterYear) %>% summarise(WinterRun\_Genetic\_Loss=sum(Loss\_SalvageData),WinterRun\_Genetic\_Loss\_according\_to\_geneticdatasheet=sum(Loss\_GeneticData))  
annual\_salvage\_genetic[is.na(annual\_salvage\_genetic)] <- 0  
  
combined\_jpe\_data<-full\_join(annual\_salvage,annual\_salvage\_genetic) %>% left\_join(jpe\_data) %>% filter(WaterYear>=1996) %>%  
 mutate\_all(~replace(., is.na(.), 0)) %>% mutate(JuvenileProductionEstimate=as.numeric(JuvenileProductionEstimate)) %>%  
 mutate(PercentJPE\_genetic=WinterRun\_Genetic\_Loss/JuvenileProductionEstimate\*100,PercentJPE\_LAD=WinterRun\_LAD\_Loss/JuvenileProductionEstimate\*100)

## Joining, by = "WaterYear"  
## Joining, by = "WaterYear"

#Print out csv  
write.csv(combined\_jpe\_data,file=file.path("output/JPE\_Genetic\_Loss\_Comparison\_2023-01-17.csv"),row.names = F)