Table of Contents

# Measuring Output for U.S. Commercial Fisheries From Theory to Practice

Generated: 2020-09-01

**Emily Markowitz**1 ([Emily.Markowitz@noaa.gov](mailto:Emily.Markowitz@noaa.gov))

**Sun Ling Wang**2 ([Sun-Ling.Wang@noaa.gov](mailto:Sun-Ling.Wang@noaa.gov))

1Contractor, ECS Federal in support of NOAA Fisheries Office of Science and Technology Economics & Social Analysis Division; as of Sept. 28, 2020: Alaska Fisheries Science Center, National Marine Fisheries Service, National Oceanic and Atmospheric Administration, Seattle, WA 98195

2On detail with the NOAA Fisheries Office of Science and Technology Economics & Social Analysis Division

* *The views expressed are those of the author and should not be attributed to the NOAA, ECS or ERS*

GitHub: <https://github.com/emilyhmarkowitz/FisheriesEconomicProductivityIndex>

R Package is forthcomming.

## A quick message from our sponsors: NOAA README

*This repository is a scientific product and is not official communication of the National Oceanic and Atmospheric Administration, or the United States Department of Commerce. All NOAA Git Hub project code is provided on an ‘as is’ basis and the user assumes responsibility for its use. Any claims against the Department of Commerce or Department of Commerce bureaus stemming from the use of this Git Hub project will be governed by all applicable Federal law. Any reference to specific commercial products, processes, or services by service mark, trademark, manufacturer, or otherwise, does not constitute or imply their endorsement, recommendation or favoring by the Department of Commerce. The Department of Commerce seal and logo, or the seal and logo of a DOC bureau, shall not be used in any manner to imply endorsement of any commercial product or activity by DOC or the United States Government.*

## Study Purpose

* Develop alternative approaches to measure national and regional fishery outputs for productivity measurements.
* Evaluate the impacts of missing data and other issues on output estimates.

## Theoretical Framework: Törnqvist index

### A Flexible Function and Superlative Quantity Index (Diewert 1976)

Of course, we could calculate something as simple as the simple sum of fisheries quantity from species’ quantities. There, you would simply sum all of the species from the entire commercial fishing sector.

When you have a dataset with missing data, different groups that require their own subsetted analysis (so we can recogize the difference between the economic stuff of finfish, shellfish, etc.), and other unqiue caveats as this one does, you will find that this method will likely provide a grossely incomplete image of what is actually happening.

Instead, we have adapted the **General Total Factor Productivity () Equation**

The equation for the 2 main models described here can be described by this theoretical Törnqvist index framework. It is a flexible function and superlative quantity index.

The general form of the can be measured as aggregate output () divided by real total inputs (). Rates of TFP growth are constructed using the Törnqvist index approach. The TFP growth over two time periods is defined as:

Such that:

* Output represents
* Input represents

The first part is the function is the output, which is composed of a 2 year average revenue shares and quantity change. Output, for our purposes, will represent National or Regional commercial fisheries landings.

The second part of this function represents (in a similar fashion to the first part of the equation) all the input (e.g., capital, labor, energy, materials, and services costs) that went into obtaining the output and follows a similar equation setup. Finding the data for this input side of the equation has proven to be a bit more difficult than anticipated, …so for this exercise we are simply going to attempt to solve the output side of the equation.

where:

* = individual outputs. This will later be referred to as in the following equations.
* = individual inputs
* = output revenue shares
* = input cost shares
* and = time, where 1 is the minimum year in the data set
* = fishery category, e.g., Finfish (=1), Shellfish (=2)
* = species, e.g., Salmon, Alewife, Surf Clams

## Download Package for Output Portion of Function

library(devtools)  
devtools::install\_github("emilyhmarkowitz/FishEconProdOutput", dependencies = TRUE)  
library(FishEconProdOutput)

## Output Method: From Quantity to Quantity Measures

### Variable Summary

Variables

* = individual quantity outputs in pounds (lbs).
* = individual value outputs in dollars ($)
* and = simple sum of Quantity (Q) and Value (V)
* = output revenue shares
* is the year to base all indices from

Subscript Indices

* and are time subscripts, where 1 is the minimum year in the data set
* is category, e.g., Finfish (=1), Shellfish (=2)
* is species, e.g., Salmon, Alewife, Surf Clams

### Data requirements and source

The Tornqvist quantity index requires data on quantity and revenue shares. We employ landings quantity (pounds) and landings value ($USD) data by year, state, and species.

* Data source: [Fisheries One Stop Shop downloaded August 13 2020](https://foss.nmfs.noaa.gov/apexfoss/f?p=215:200::::::)
* More information about the data: [Commercial Fisheries Landings Data](https://www.fisheries.noaa.gov/national/sustainable-fisheries/commercial-fisheries-landings)

Here is the original data:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | year | V1\_1Salmon | Q1\_1Salmon | V1\_2Cod | Q1\_2Cod | V2\_1Shrimp | Q2\_1Shrimp | V2\_2Clam | Q2\_2Clam | V1\_3Flounder | Q1\_3Flounder | V1\_4SeaBass | Q1\_4SeaBass |
| 1 | 2007 | NA | NA | 2800 | 2000 | 800 | 100 | 1000 | 150 | 1000 | NA | NA | 1000 |
| 2 | 2008 | NA | NA | 2700 | 1900 | 1000 | 120 | 1200 | 160 | 1200 | NA | 120 | 1200 |
| 3 | 2009 | NA | NA | 2900 | 2000 | 900 | 110 | 900 | 140 | 900 | NA | 110 | 900 |
| 4 | 2010 | 100 | 20 | 3000 | 2500 | 700 | 90 | NA | NA | NA | NA | 90 | NA |
| 5 | 2011 | 100 | 10 | 3100 | 2400 | 900 | 80 | NA | NA | NA | NA | 80 | NA |
| 6 | 2012 | 150 | 12 | 2900 | 2300 | 1000 | 100 | NA | NA | NA | NA | 100 | NA |
| 7 | 2013 | 180 | 11 | 2800 | 2000 | 1200 | 100 | 1000 | 140 | 1000 | NA | 100 | 1000 |
| 8 | 2014 | 170 | 11 | 3200 | 2300 | 1100 | 110 | 900 | 110 | 900 | NA | NA | 900 |
| 9 | 2015 | 200 | 10 | 3500 | 2400 | 1000 | 90 | 1000 | 130 | 1000 | NA | NA | 1000 |
| 10 | 2016 | 180 | 15 | 3200 | 2200 | 1200 | 100 | 1100 | 160 | 1100 | NA | NA | 1100 |

#### In this data, we use these naming conventions for the column names.

For example, in “V1\_0Finfish”:

* “V”… refers to the variable represented in the column (here V = “Value”)
* …“1”… refers to the category iteration (here, = Finfish)
* …"\_"… is simply a separator in the title
* …“0”.. refers to the total of the specific category.
* …“Finfish” is purely descriptive (here the name of the category), so you can follow along with what is happening!

Similarly for “Q2\_2Clam”:

* “Q”… refers to the variable represented in the column (here Q = “Quantity”)
* …“2”… refers to the category iteration (here, = Shellfish)
* …"\_"… is simply a separator in the title
* …“2”.. refers to the iteration of the species, such that this organism happens to be the second species of this category.
* …“Clams” is purely descriptive (here the name of the species), so you can follow along with what is happening!

### Lets get started

### Calculate Category and Entire Fishery Sums of and

## Commercial fisheries data availability, issues, and mitigation

* How should we deal with missing data?
* How much of the time series should be assessed?
* How should species data be categorized?

### Missing data

NA in the commercial fisheries data set does not mean 0, but rather that the data may be confidential (following the rule of 3) or simply be missing. This can be a serious issue here, as missing data could lead to artificially large price () and quantity () changes for years in the time series.

There are a lot of NAs in this data set. Some data columns are completely filled with NA and even those that are not – So first thing we did was to take care of columns that were mostly made of NAs. We instituted a % missing data threshold. Here, these columns have too few data according to a 40% threshold we’ve instituted, so we are simply going to remove that data. Honestly, what could data with that much missing really tell us and at what point are we just making the data up to make up for what is missing?

Now with those offending columns of missing data gone, we can go after the loose, infrequent, NAs. Here we impute the values from the closest value and hearkening back to our previous example, the fictitious code value data looks a lot more realistic!

When we apply these practices for missing data to real data examples, we see that the removal of nearly 400 species data results in a plot for quantity index (one of our targeted end products) almost the same to one where no data was removed. This provides evidence that those removed data weren’t really contributing much to the results. This is also a large data set such that the impact of the data removed (35% of the original data) is cushioned by how much data is remaining.

On the other hand, the removal of 58 species (approximately 25% of the original data) radically changes this regional plot. The y-axis is displaying beyond-reasonable values and the spike in the “Other” category can’t possibly be correct. With the percent missing threshold implemented, QI values appear to be in a much more sensible range.

### Time Series Reporting Consistency

Consistent reporting throughout the time series. Looking simply at the summed quantity for each category and the entire fishery, there have been several periods of improved reporting, such that the increasing trend is so steep and is not indicative of real increases in the quantity of fish caught from 1950 (when data was first started to be collected) to today (2016). If we just take the last part of that timeline, the trend seems more level and reasonable.

If we look at the quantity index result, we see that much of our missing data is pre-1990 and our analysis inherently removes less data when we subset, giving us more species data to work with.

### Defining Species Categories

The next question is something we are still thinking about: How to define our species categories. These can be specific or broad?

Theoretically, categories should group species with similar economic impact (e.g., fishing costs) which can be difficult to define.

It is possible that we might be able to use taxonomic group as a proxy for this since species in the same taxonomic group are more likely to be caught in similar ways (an idea that is very pleasing to the biologist in me!).

More specifically, we applied two methods:

1. We used the same species groupings as were used in Fisheries Economics of the US report. This could work because there is a precedent for using this species split up, but it is fairly over-generalized. “Shellfish” is not really the same as saying “all invertebrates”, for example.
2. Alternatively, thanks to renewed data managing efforts done by ST1, we now have ITSN numbers associated with each species, and with some fancy footwork, can resort these species into a variety of taxonomically-relevant groups.

However, with the more categories we have, the less data we have for each category.

These plots were created using the same data, just by splitting the categories up differently. We can see that the QI is increasing in the first plot using the FEUS categories and that the second plot using the taxonomically defined species has species increasing and decreasing.

This may be a key to better seeing what is actually going on in the data.

## QUANTITY METHOD

In most of the following examples, we will just focus on the finfish (*i*=1) side of the equation. Here *baseyr* is set to 2010 and the *pctmiss* (The percent of data in a column that we will allow to be missing for analysis; more on that later) is set to 0.5%.

This method works directly from the quantity data so it is good for when is often available.

I won’t get to deep in the math here – we can review these later if needed in the discussion – but the main takeaway is that this method simply uses the available quantity data at the species level to develop revenue-share weighed quantity changes.

### At the species level:

#### Remove any V and Q data where V column has less data than the specifed

*No warning.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Q1\_1Salmon | Q1\_2Cod | Q2\_1Shrimp | Q2\_2Clam | REMOVED\_Q1\_3Flounder | Q1\_4SeaBass |
| 2007 | NA | 2000 | 100 | 150 | NA | 1000 |
| 2008 | NA | 1900 | 120 | 160 | NA | 1200 |
| 2009 | NA | 2000 | 110 | 140 | NA | 900 |
| 2010 | 20 | 2500 | 90 | NA | NA | NA |
| 2011 | 10 | 2400 | 80 | NA | NA | NA |
| 2012 | 12 | 2300 | 100 | NA | NA | NA |
| 2013 | 11 | 2000 | 100 | 140 | NA | 1000 |
| 2014 | 11 | 2300 | 110 | 110 | NA | 900 |
| 2015 | 10 | 2400 | 90 | 130 | NA | 1000 |
| 2016 | 15 | 2200 | 100 | 160 | NA | 1100 |

#### Total Value of species with available Q and V data

For where and is not available to a certain threshold (say 60% of the data is missing we call it “unavailable”), the data is simply removed from the analysis.

Because we removed some columns for not meeting a perecent missing threshold of 0.5% and those columns will not be used at all in any part of the further analysis, we need to re-calculate the totals of and for the catagories and the fishery as a whole.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | QE1\_0Finfish | VE1\_0Finfish | Q1\_0Finfish | V1\_0Finfish |
| 2007 | 3000 | 3800 | 3000 | 2800 |
| 2008 | 3100 | 4020 | 3100 | 2820 |
| 2009 | 2900 | 3910 | 2900 | 3010 |
| 2010 | 2520 | 3190 | 2520 | 3190 |
| 2011 | 2410 | 3280 | 2410 | 3280 |
| 2012 | 2312 | 3150 | 2312 | 3150 |
| 2013 | 3011 | 4080 | 3011 | 3080 |
| 2014 | 3211 | 4270 | 3211 | 3370 |
| 2015 | 3410 | 4700 | 3410 | 3700 |
| 2016 | 3315 | 4480 | 3315 | 3380 |

#### Address Infrequent Missing Data

There may be instances where there are no or too few Q data for that species in a year or ever. The next goal will be to calculate the quantity change, so we need to have a value in there that won’t show change. If we left an NA (which would be treated as a 0) in the cell, then the change from year to year would be very large and misrepresent the index trend. To avoid this, we do the following:

##### 1. If there are instances for a species where there are too are completely missing from the timeseries or where a percent of data that is missing from the timeseries, we will remove the offending columns entierly, so they don’t influence the downstream price change or price index calculations.

Let’s say here that if 50% of the data is missing in a given , don’t use that to calculate that species

|  |  |  |  |
| --- | --- | --- | --- |
|  | Q1\_1Salmon | Q1\_2Cod | Q1\_4SeaBass |
| 2007 | NA | 2000 | 1000 |
| 2008 | NA | 1900 | 1200 |
| 2009 | NA | 2000 | 900 |
| 2010 | 20 | 2500 | NA |
| 2011 | 10 | 2400 | NA |
| 2012 | 12 | 2300 | NA |
| 2013 | 11 | 2000 | 1000 |
| 2014 | 11 | 2300 | 900 |
| 2015 | 10 | 2400 | 1000 |
| 2016 | 15 | 2200 | 1100 |

*No warning.*

##### 2. If the first value of is 0/NA in a timeseries, we (impute) let the next available non-zero/non-NA value of in the timeseries inform the past.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Q1\_1Salmon | Q1\_2Cod | Q1\_4SeaBass |
| 2007 | 20 | 2000 | 1000 |
| 2008 | NA | 1900 | 1200 |
| 2009 | NA | 2000 | 900 |
| 2010 | 20 | 2500 | NA |
| 2011 | 10 | 2400 | NA |
| 2012 | 12 | 2300 | NA |
| 2013 | 11 | 2000 | 1000 |
| 2014 | 11 | 2300 | 900 |
| 2015 | 10 | 2400 | 1000 |
| 2016 | 15 | 2200 | 1100 |

##### 3. If there is a value in the middle of ’s timeseries that is 0/NA, we (impute) let the most recent past available non-zero/non-NA of in the timeseries inform the future.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Q1\_1Salmon | Q1\_2Cod | Q1\_4SeaBass |
| 2007 | 20 | 2000 | 1000 |
| 2008 | 20 | 1900 | 1200 |
| 2009 | 20 | 2000 | 900 |
| 2010 | 20 | 2500 | 900 |
| 2011 | 10 | 2400 | 900 |
| 2012 | 12 | 2300 | 900 |
| 2013 | 11 | 2000 | 1000 |
| 2014 | 11 | 2300 | 900 |
| 2015 | 10 | 2400 | 1000 |
| 2016 | 15 | 2200 | 1100 |

#### Impute values of where P was able to be calculated

To ensure that the price index does not rise or fall to quickly with changes (that are really because of NA values) we fill in the missing instances of .

##### 1. If the first value of is 0/NA in a timeseries, we let the next available non-zero value of in the timeseries inform the past.

|  |  |  |  |
| --- | --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod | V1\_4SeaBass |
| 2007 | 100 | 2800 | 120 |
| 2008 | NA | 2700 | 120 |
| 2009 | NA | 2900 | 110 |
| 2010 | 100 | 3000 | 90 |
| 2011 | 100 | 3100 | 80 |
| 2012 | 150 | 2900 | 100 |
| 2013 | 180 | 2800 | 100 |
| 2014 | 170 | 3200 | NA |
| 2015 | 200 | 3500 | NA |
| 2016 | 180 | 3200 | NA |

##### 2. If there is a value in the middle of ’s timeseries that is 0/NA, we let the most recent past available non-zero of in the timeseries inform the future.

|  |  |  |  |
| --- | --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod | V1\_4SeaBass |
| 2007 | 100 | 2800 | 120 |
| 2008 | 100 | 2700 | 120 |
| 2009 | 100 | 2900 | 110 |
| 2010 | 100 | 3000 | 90 |
| 2011 | 100 | 3100 | 80 |
| 2012 | 150 | 2900 | 100 |
| 2013 | 180 | 2800 | 100 |
| 2014 | 170 | 3200 | 100 |
| 2015 | 200 | 3500 | 100 |
| 2016 | 180 | 3200 | 100 |

##### Analysis Warnings Checks

Just so we can get a sense of the data, we want to see how many species are significantly increasing or decreasing over time for V and Q.

We’ll use the below function to collect our info:

## function (Columns, temp)   
## {  
## lm\_check <- data.frame(col = rep\_len(x = NA, length.out = length(Columns)),   
## slope = rep\_len(x = NA, length.out = length(Columns)),   
## intercept = rep\_len(x = NA, length.out = length(Columns)),   
## R2 = rep\_len(x = NA, length.out = length(Columns)), R2adj = rep\_len(x = NA,   
## length.out = length(Columns)), Pr = rep\_len(x = NA,   
## length.out = length(Columns)), Fstat = rep\_len(x = NA,   
## length.out = length(Columns)))  
## for (c0 in 1:length(Columns)) {  
## if (sum(is.na(temp[, Columns[c0]])) == length(temp[,   
## Columns[c0]]) | length(temp[, Columns[c0]]) %in%   
## sum(temp[, Columns[c0]] %in% c(NA, 0))) {  
## lm\_check$col[c0] <- NA  
## lm\_check$slope[c0] <- NA  
## lm\_check$intercept[c0] <- NA  
## lm\_check$R2[c0] <- NA  
## lm\_check$R2adj[c0] <- NA  
## lm\_check$Pr[c0] <- NA  
## lm\_check$Fstat[c0] <- NA  
## }  
## else {  
## temp0 <- summary(lm(rownames(temp) ~ temp[, Columns[c0]]))  
## lm\_check$col[c0] <- Columns[c0]  
## lm\_check$slope[c0] <- temp0$coefficients[2]  
## lm\_check$intercept[c0] <- temp0$coefficients[1]  
## lm\_check$R2[c0] <- temp0$r.squared  
## lm\_check$R2adj[c0] <- temp0$adj.r.squared  
## lm\_check$Pr[c0] <- temp0$coefficients[8]  
## lm\_check$Fstat[c0] <- ifelse(is.null(temp0$fstatistic[1]),   
## NA, as.numeric(temp0$fstatistic[1]))  
## }  
## }  
## lm\_check$var <- substr(x = Columns, 1, 1)  
## lm\_check$slopecheck <- "Insig"  
## lm\_check$slopecheck <- ifelse(lm\_check$slope >= 0 & lm\_check$Pr <=   
## 0.05, "Sig Pos", "Insig")  
## lm\_check$slopecheck <- ifelse(lm\_check$slope < 0 & lm\_check$Pr <=   
## 0.05, "Sig Neg", lm\_check$slopecheck)  
## return(lm\_check)  
## }  
## <bytecode: 0x000000002326c3a0>  
## <environment: namespace:FishEconProdOutput>

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | NameBasecategory | col | slope | intercept | R2 | R2adj | Pr | Fstat | var | slopecheck |
| 1 | 1\_0Finfish | V1\_1Salmon | 0.0653553 | 2002.481 | 0.8159514 | 0.7929453 | 0.0003399 | 35.4667781 | V | Sig Pos |
| 2 | 1\_0Finfish | V1\_2Cod | 0.0093573 | 1983.335 | 0.5614367 | 0.5066163 | 0.0126083 | 10.2413793 | V | Sig Pos |
| 3 | 1\_0Finfish | V1\_4SeaBass | -0.1176471 | 2023.500 | 0.2281640 | 0.1316845 | 0.1626498 | 2.3648961 | V | Insig |
| 4 | 1\_0Finfish | Q1\_1Salmon | -0.4845469 | 2018.720 | 0.5432798 | 0.4861898 | 0.0150066 | 9.5161963 | Q | Sig Neg |
| 5 | 1\_0Finfish | Q1\_2Cod | 0.0065000 | 1997.200 | 0.2048485 | 0.1054545 | 0.1890352 | 2.0609756 | Q | Insig |
| 6 | 1\_0Finfish | Q1\_4SeaBass | -0.0010417 | 2012.521 | 0.0012626 | -0.1235795 | 0.9223686 | 0.0101138 | Q | Insig |

How many slopes are significantly increaseing or decreaseing

|  |  |  |  |
| --- | --- | --- | --- |
|  | var | slopecheck | Freq |
| 1 | Q | Insig | 2 |
| 3 | Q | Sig Neg | 1 |
| 5 | Q | Sig Pos | 0 |
| 2 | V | Insig | 1 |
| 4 | V | Sig Neg | 0 |
| 6 | V | Sig Pos | 2 |

#### Value of species where Q available

, as defined and discussed in the subsequent step, will need to sum to 1 across all species in a category. Therefore, you will need to sum a new total of available (called ) for the category using only values for species that were used to calculate (called ).

where:

* is the new total of (called ) for the category using only values for species that were used to calculate
* are the where were able to be calculated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod | V1\_4SeaBass | VV1\_0Finfish |
| 2007 | 100 | 2800 | 120 | 3020 |
| 2008 | 100 | 2700 | 120 | 2920 |
| 2009 | 100 | 2900 | 110 | 3110 |
| 2010 | 100 | 3000 | 90 | 3190 |
| 2011 | 100 | 3100 | 80 | 3280 |
| 2012 | 150 | 2900 | 100 | 3150 |
| 2013 | 180 | 2800 | 100 | 3080 |
| 2014 | 170 | 3200 | 100 | 3470 |
| 2015 | 200 | 3500 | 100 | 3800 |
| 2016 | 180 | 3200 | 100 | 3480 |

#### Revenue-share

Revenue Share for each species (; e.g., Salmon and Flounder)

where:

* is the revenue share per individual species (s), category (i), for each year (t)
* is the value ($) per individual species (s), category (i), for each year (t)

Here we divide by because only includes species used to calculate as per the above price calculations.

|  |  |  |  |
| --- | --- | --- | --- |
|  | R1\_1Salmon | R1\_2Cod | R1\_4SeaBass |
| 1 | 0.0331126 | 0.9271523 | 0.0397351 |
| 2 | 0.0342466 | 0.9246575 | 0.0410959 |
| 3 | 0.0321543 | 0.9324759 | 0.0353698 |
| 4 | 0.0313480 | 0.9404389 | 0.0282132 |
| 5 | 0.0304878 | 0.9451220 | 0.0243902 |
| 6 | 0.0476190 | 0.9206349 | 0.0317460 |
| 7 | 0.0584416 | 0.9090909 | 0.0324675 |
| 8 | 0.0489914 | 0.9221902 | 0.0288184 |
| 9 | 0.0526316 | 0.9210526 | 0.0263158 |
| 10 | 0.0517241 | 0.9195402 | 0.0287356 |

####Analysis Warnings Checks

As an additional check, let’s make sure that each row sums to 1.

|  |  |
| --- | --- |
|  | x |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 1 |
| 7 | 1 |
| 8 | 1 |
| 9 | 1 |
| 10 | 1 |

Is there a warning?

*No warning.*

#### Revenue-share weighted quantity changes

Revenue Share-Weighted Qunatity Changes for each species (; e.g., Salmon and Flounder)

Where:

* = Revenue share-weighted quantity change for a species (s)

Such that:

* category’s (i) Quantity Change for each species (s) =
* category’s (i) Revenue Share for each species (s) =

### At the fishery level:

Then we calculate the revenue share, QI, and revenue-share weighted quantity changes at the category level, which are used at the commercial fishery level to develop the annual quantity change and index.

#### Quantity change

Quantity Changes for the category (; e.g., Finfish). These, specifically the QC, are what go into the output equation.

Where:

* = Quantity change for a category (i)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | QCW1\_1Salmon | QCW1\_2Cod | QCW1\_4SeaBass | QC1\_0Finfish |
| 2007 | 0.0000000 | 0.0000000 | 0.0000000 | 0.0000000 |
| 2008 | 0.0000000 | -0.0474927 | 0.0073686 | -0.0401241 |
| 2009 | 0.0000000 | 0.0476292 | -0.0109989 | 0.0366303 |
| 2010 | 0.0000000 | 0.2089644 | 0.0000000 | 0.2089644 |
| 2011 | -0.0214306 | -0.0384862 | 0.0000000 | -0.0599168 |
| 2012 | 0.0071203 | -0.0397029 | 0.0000000 | -0.0325827 |
| 2013 | -0.0046142 | -0.1278630 | 0.0033828 | -0.1290945 |
| 2014 | 0.0000000 | 0.1279717 | -0.0032286 | 0.1247431 |
| 2015 | -0.0048429 | 0.0392239 | 0.0029045 | 0.0372855 |
| 2016 | 0.0211563 | -0.0800763 | 0.0026235 | -0.0562965 |

#### Implicit quantity index

Quantity Index for the each category ()

We calculate the quantity index first by comparing by multiplying the previous years by that year’s quantity change , where the of the first year

Where

Note that the first row of this column is = 1

Then, to change the price index into base year dollars, we use the following equation:

|  |  |
| --- | --- |
|  | QI1\_0Finfish |
| 2007 | 0.8142640 |
| 2008 | 0.7822391 |
| 2009 | 0.8114241 |
| 2010 | 1.0000000 |
| 2011 | 0.9418429 |
| 2012 | 0.9116497 |
| 2013 | 0.8012406 |
| 2014 | 0.9076914 |
| 2015 | 0.9421740 |
| 2016 | 0.8905983 |

#### Redo Analysis for Other Categories

Now lets redo that whole analysis up to this point (via function) for the two species of the shellfish group, as we will need them for the next steps of this analysis.

We use the *QuantityMethodOutput\_Category* function to calculate everything we did above at category level.

What does the Shellfish data look like?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | R2\_1Shrimp | R2\_2Clam | QCW2\_1Shrimp | QCW2\_2Clam | QC2\_0Shellfish | QI2\_0Shellfish |
| 2007 | 0.4444444 | 0.5555556 | 0.0000000 | 0.0000000 | 0.0000000 | 1.0918916 |
| 2008 | 0.4545455 | 0.5454545 | 0.0819526 | 0.0355288 | 0.1174814 | 1.2280076 |
| 2009 | 0.5000000 | 0.5000000 | -0.0415282 | -0.0698005 | -0.1113287 | 1.0986305 |
| 2010 | 0.4375000 | 0.5625000 | -0.0940644 | 0.0000000 | -0.0940644 | 1.0000000 |
| 2011 | 0.5000000 | 0.5000000 | -0.0552108 | 0.0000000 | -0.0552108 | 0.9462857 |
| 2012 | 0.5263158 | 0.4736842 | 0.1145079 | 0.0000000 | 0.1145079 | 1.0610904 |
| 2013 | 0.5454545 | 0.4545455 | 0.0000000 | 0.0000000 | 0.0000000 | 1.0610904 |
| 2014 | 0.5500000 | 0.4500000 | 0.0522040 | -0.1090710 | -0.0568670 | 1.0024330 |
| 2015 | 0.5000000 | 0.5000000 | -0.1053521 | 0.0793507 | -0.0260014 | 0.9767043 |
| 2016 | 0.5217391 | 0.4782609 | 0.0538255 | 0.1015627 | 0.1553882 | 1.1408993 |

#### Value of categories available

Value for all fisheries for species where Q was able to be calculated

, defined and discussed in the subsequent step, will need to sum to 1 across all species in a category. Therefore, you will need to sum a new total of (called ) for the category using only values for species that were used to calculate .

where:

* is the new total of for the entire fishery using only values for species that were used to calculate

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | R2\_1Shrimp | R2\_2Clam | QCW2\_1Shrimp | QCW2\_2Clam | QC2\_0Shellfish | QI2\_0Shellfish |
| 2007 | 0.4444444 | 0.5555556 | 0.0000000 | 0.0000000 | 0.0000000 | 1.0918916 |
| 2008 | 0.4545455 | 0.5454545 | 0.0819526 | 0.0355288 | 0.1174814 | 1.2280076 |
| 2009 | 0.5000000 | 0.5000000 | -0.0415282 | -0.0698005 | -0.1113287 | 1.0986305 |
| 2010 | 0.4375000 | 0.5625000 | -0.0940644 | 0.0000000 | -0.0940644 | 1.0000000 |
| 2011 | 0.5000000 | 0.5000000 | -0.0552108 | 0.0000000 | -0.0552108 | 0.9462857 |
| 2012 | 0.5263158 | 0.4736842 | 0.1145079 | 0.0000000 | 0.1145079 | 1.0610904 |
| 2013 | 0.5454545 | 0.4545455 | 0.0000000 | 0.0000000 | 0.0000000 | 1.0610904 |
| 2014 | 0.5500000 | 0.4500000 | 0.0522040 | -0.1090710 | -0.0568670 | 1.0024330 |
| 2015 | 0.5000000 | 0.5000000 | -0.1053521 | 0.0793507 | -0.0260014 | 0.9767043 |
| 2016 | 0.5217391 | 0.4782609 | 0.0538255 | 0.1015627 | 0.1553882 | 1.1408993 |

|  |  |  |
| --- | --- | --- |
|  | V1\_0Finfish | V2\_0Shellfish |
| 2007 | 2800 | 1800 |
| 2008 | 2820 | 2200 |
| 2009 | 3010 | 1800 |
| 2010 | 3190 | 700 |
| 2011 | 3280 | 900 |
| 2012 | 3150 | 1000 |
| 2013 | 3080 | 2200 |
| 2014 | 3370 | 2000 |
| 2015 | 3700 | 2000 |
| 2016 | 3380 | 2300 |

#### Revenue share

Revenue Share for the each category ()

TOLEDO - Which this wrong?

where:

* is the revenue share per individual species (s), category (i), for each year (t)
* is the value ($) per individual species (s), category (i), for each year (t)

Here, we don’t use beacause we want to expand the proportion to include all of the species caught, regardless if they were used in the quantity calculations.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | R1\_0Finfish | R2\_0Shellfish | V1\_0Finfish | V2\_0Shellfish | V0\_0Total |
| 2007 | 0.6086957 | 0.3913043 | 2800 | 1800 | 4600 |
| 2008 | 0.5617530 | 0.4382470 | 2820 | 2200 | 5020 |
| 2009 | 0.6257796 | 0.3742204 | 3010 | 1800 | 4810 |
| 2010 | 0.8200514 | 0.1799486 | 3190 | 700 | 3890 |
| 2011 | 0.7846890 | 0.2153110 | 3280 | 900 | 4180 |
| 2012 | 0.7590361 | 0.2409639 | 3150 | 1000 | 4150 |
| 2013 | 0.5833333 | 0.4166667 | 3080 | 2200 | 5280 |
| 2014 | 0.6275605 | 0.3724395 | 3370 | 2000 | 5370 |
| 2015 | 0.6491228 | 0.3508772 | 3700 | 2000 | 5700 |
| 2016 | 0.5950704 | 0.4049296 | 3380 | 2300 | 5680 |

##### Analysis Warnings Checks

As an additional check, let’s make sure that each row sums to 1.

|  |  |
| --- | --- |
|  | x |
| 2007 | 1 |
| 2008 | 1 |
| 2009 | 1 |
| 2010 | 1 |
| 2011 | 1 |
| 2012 | 1 |
| 2013 | 1 |
| 2014 | 1 |
| 2015 | 1 |
| 2016 | 1 |

Is there a warning?

*No warning.*

#### Revenue share weighted quantity changes

Revenue Share-Weighted Qunatity Changes for each category (; e.g., Finfish and Shellfish)

Where:

* = Revenue share-weighted quantity change for each category (i)

Such that:

* category’s (i) Quantity Change for each category (i) =
* category’s (i) Revenue Share for each category (i) =

### At the entire commercial fisheries sector level:

#### Quantity change

Quantity Changes for the entire fishery ()

Where:

* = Quantity change for the entire fishery

|  |  |  |  |
| --- | --- | --- | --- |
|  | QCW1\_0Finfish | QCW2\_0Shellfish | QC0\_0Total |
| 2007 | 0.0000000 | 0.0000000 | 0.0000000 |
| 2008 | -0.0234816 | 0.0487284 | 0.0252468 |
| 2009 | 0.0217499 | -0.0452255 | -0.0234756 |
| 2010 | 0.1510636 | -0.0260638 | 0.1249998 |
| 2011 | -0.0480755 | -0.0109113 | -0.0589868 |
| 2012 | -0.0251493 | 0.0261235 | 0.0009742 |
| 2013 | -0.0866462 | 0.0000000 | -0.0866462 |
| 2014 | 0.0755254 | -0.0224371 | 0.0530883 |
| 2015 | 0.0238009 | -0.0094036 | 0.0143972 |
| 2016 | -0.0350218 | 0.0587217 | 0.0236999 |

#### Quantity index

Quantity Index for the entier fishery ()

where and then

We calculate the quantity index first by comparing by multiplying the previous years by that year’s quantity change , where the of the first year

Where

Note that the first row of this column is = 1

Then, to change the price index into base year dollars, we use the following equation:

|  |  |
| --- | --- |
|  | QI0\_0Total |
| 2007 | 0.8809353 |
| 2008 | 0.9034593 |
| 2009 | 0.8824970 |
| 2010 | 1.0000000 |
| 2011 | 0.9427192 |
| 2012 | 0.9436381 |
| 2013 | 0.8653175 |
| 2014 | 0.9124970 |
| 2015 | 0.9257294 |
| 2016 | 0.9479312 |

##### Other Analysis Warnings Checks

To make sure our analyses worked as inteded, let’s see if we can back calculate our numbers.

We want the calcuated V to equal this check:

When back calculated, growth rate?

|  |  |  |
| --- | --- | --- |
|  | part1 | part2 |
| 2007 | NA | NA |
| 2008 | -0.0252468 | 0.0252468 |
| 2009 | 0.0234756 | -0.0234756 |
| 2010 | -0.1249998 | 0.1249998 |
| 2011 | 0.0589868 | -0.0589868 |
| 2012 | -0.0009742 | 0.0009742 |
| 2013 | 0.0866462 | -0.0866462 |
| 2014 | -0.0530883 | 0.0530883 |
| 2015 | -0.0143972 | 0.0143972 |
| 2016 | -0.0236999 | 0.0236999 |

Is there a warning?

*Warning: When back calculated, ln(Q\_t/Q\_{t-1}) = did not equal sum( ((R\_{i, t} - R\_{i, t-1})(2))*  ln((Q\_{t,i})(Q\_{t-1,i}))\*

#### Sum Total Simple Sum Quantity Output Index

Where:

* is the sum of Q before these calculations; the simple sum
* is the index of the sum of Q before these equations

|  |  |  |
| --- | --- | --- |
|  | QE0\_0Total | QEI0\_0Total |
| 2007 | 3250 | 1.2452107 |
| 2008 | 3380 | 1.2950192 |
| 2009 | 3150 | 1.2068966 |
| 2010 | 2610 | 1.0000000 |
| 2011 | 2490 | 0.9540230 |
| 2012 | 2412 | 0.9241379 |
| 2013 | 3251 | 1.2455939 |
| 2014 | 3431 | 1.3145594 |
| 2015 | 3630 | 1.3908046 |
| 2016 | 3575 | 1.3697318 |

#### View Total Outputs

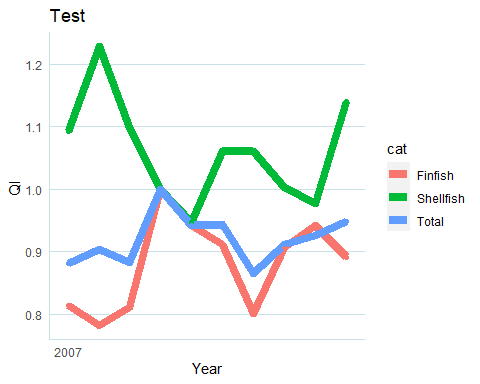
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | QE0\_0Total | VE0\_0Total | VV0\_0Total | V0\_0Total | QC0\_0Total | QI0\_0Total | QEI0\_0Total |
| 2007 | 3250 | 5600 | 4820 | 4600 | 0.0000000 | 0.8809353 | 1.2452107 |
| 2008 | 3380 | 6220 | 5120 | 5020 | 0.0252468 | 0.9034593 | 1.2950192 |
| 2009 | 3150 | 5710 | 4910 | 4810 | -0.0234756 | 0.8824970 | 1.2068966 |
| 2010 | 2610 | 3890 | 4790 | 3890 | 0.1249998 | 1.0000000 | 1.0000000 |
| 2011 | 2490 | 4180 | 5080 | 4180 | -0.0589868 | 0.9427192 | 0.9540230 |
| 2012 | 2412 | 4150 | 5050 | 4150 | 0.0009742 | 0.9436381 | 0.9241379 |
| 2013 | 3251 | 6280 | 5280 | 5280 | -0.0866462 | 0.8653175 | 1.2455939 |
| 2014 | 3431 | 6270 | 5470 | 5370 | 0.0530883 | 0.9124970 | 1.3145594 |
| 2015 | 3630 | 6700 | 5800 | 5700 | 0.0143972 | 0.9257294 | 1.3908046 |
| 2016 | 3575 | 6780 | 5780 | 5680 | 0.0236999 | 0.9479312 | 1.3697318 |

#### How many data were missing at the end of the analysis?

*FYI: 0 of species V columns are completely empty, 1 of species Q columns are completely empty.*

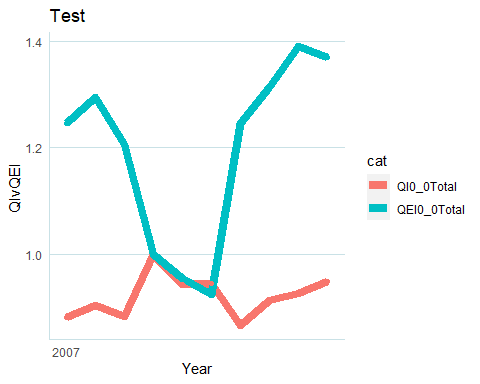
##### Graph 1: Quantity Index Categories

For comparison, let’s recreate those graphs to make sure we are getting the same output:



##### Graph 2: Quantity Index Compare

For comparison, let’s recreate those graphs to make sure we are getting the same output:



### Do same analysis via a function!

Now that we know the method, we can simplify most of it into a function and do this whole analysis in 4 easy steps:

* A. Import and Edit data
* B. Enter base year
* C. Run the function
* D. Obtain the implicit quantity estimates

#### Function

We use the *QuantityMethodOutput* function to calculate the Quanity Output at Fishery Level

#### A. Import and Edit data

#### B. Enter base year

#### C. Run the function

#### D. Obtain the implicit quantity estimates

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | QE0\_0Total | VE0\_0Total | VV0\_0Total | V0\_0Total | QC0\_0Total | QI0\_0Total | QEI0\_0Total |
| 2007 | 3250 | 5600 | 4820 | 4600 | 0.0000000 | 0.8809353 | 1.2452107 |
| 2008 | 3380 | 6220 | 5120 | 5020 | 0.0252468 | 0.9034593 | 1.2950192 |
| 2009 | 3150 | 5710 | 4910 | 4810 | -0.0234756 | 0.8824970 | 1.2068966 |
| 2010 | 2610 | 3890 | 4790 | 3890 | 0.1249998 | 1.0000000 | 1.0000000 |
| 2011 | 2490 | 4180 | 5080 | 4180 | -0.0589868 | 0.9427192 | 0.9540230 |
| 2012 | 2412 | 4150 | 5050 | 4150 | 0.0009742 | 0.9436381 | 0.9241379 |
| 2013 | 3251 | 6280 | 5280 | 5280 | -0.0866462 | 0.8653175 | 1.2455939 |
| 2014 | 3431 | 6270 | 5470 | 5370 | 0.0530883 | 0.9124970 | 1.3145594 |
| 2015 | 3630 | 6700 | 5800 | 5700 | 0.0143972 | 0.9257294 | 1.3908046 |
| 2016 | 3575 | 6780 | 5780 | 5680 | 0.0236999 | 0.9479312 | 1.3697318 |

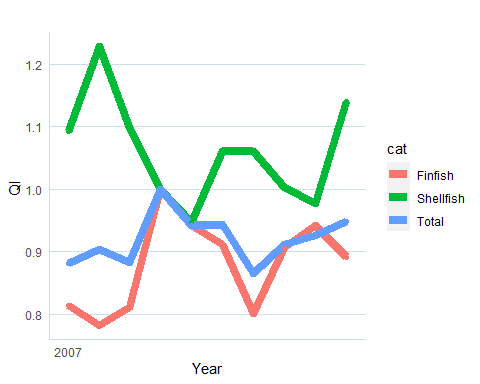
Did all of the analyses work as intended?

*list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(2, 1, 0, 1, 0, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(2, 1, 0, 1, 0, 2)), list(var = c(1, 1, 2, 2), slopecheck = c(1, 2, 1, 2), Freq = c(2, 0, 1, 1)), Warning: When back calculated, ln(Q\_t/Q\_{t-1}) = did not equal sum( ( R\_{i, t} - R\_{i, t-1} ) / 2 ) x ln( (Q\_{i,t}) / (Q\_{i,t-1} ) ), FYI: 0 of species V columns are completely empty, 2 of species Q columns are completely empty.*

#### E. Graph

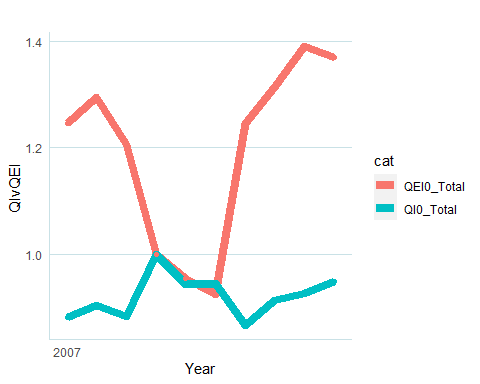
##### Graph 1: Quantity Index Categories

For comparison, let’s recreate those graphs to make sure we are getting the same output:



##### Graph 2: Quantity Index Compare

For comparison, let’s recreate those graphs to make sure we are getting the same output:



### Practice with real data (For National Data)

#### A. Import and Edit data

Load and subset Data

Edit/Restructure Data

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Q01\_0002ALEWIFE | Q01\_0003ALEWIFE. | Q01\_0004ALFONSINO. | Q01\_0006AMBERJACK.GREATER | Q01\_0007AMBERJACK.GREATER. |
| 1950 | NA | 735961 | NA | NA | NA |
| 1951 | NA | 758873 | NA | NA | NA |
| 1952 | NA | 722115 | NA | NA | NA |
| 1953 | NA | 750022 | NA | NA | NA |
| 1954 | NA | 650472 | NA | NA | NA |

#### B. Enter base year

#### C. Run the function

#### D. Obtain the implicit quantity estimates

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | VE\_Total | VV\_Total | V\_Total | QC\_Total | QI\_Total |
| 1950 | 2596863100 | 2794604681 | 2593845000 | 0.0000000 | 0.5833753 |
| 1951 | 2711142600 | 2900126381 | 2707189300 | 0.1808450 | 0.6990175 |
| 1952 | 2904551900 | 3094353181 | 2900834400 | -0.0581459 | 0.6595316 |
| 1953 | 2945724500 | 3131700481 | 2943735500 | 0.0916487 | 0.7228333 |
| 1954 | 3131528000 | 3322992081 | 3129278700 | 0.0715284 | 0.7764304 |
| 1955 | 3208381300 | 3399979081 | 3206680100 | 0.0483422 | 0.8148868 |
| 1956 | 3413031000 | 3617472081 | 3411352200 | 0.0898360 | 0.8914820 |
| 1957 | 3065775700 | 3269932281 | 3064051500 | -0.0817660 | 0.8214896 |
| 1958 | 3003831240 | 3192160321 | 3000058540 | 0.0066717 | 0.8269887 |
| 1959 | 3552786100 | 3742561181 | 3550436800 | 0.0862896 | 0.9015185 |
| 1960 | 3431488969 | 3621283050 | 3429038869 | -0.1508356 | 0.7752960 |
| 1961 | 3545366800 | 3729969881 | 3537472400 | 0.0942802 | 0.8519477 |
| 1962 | 3740472904 | 3941834181 | 3734589304 | 0.0735513 | 0.9169715 |
| 1963 | 3436281347 | 3636121924 | 3428888647 | -0.0241395 | 0.8951012 |
| 1964 | 3082842508 | 3284893585 | 3077660908 | -0.0277494 | 0.8706042 |
| 1965 | 3312327541 | 3509281118 | 3306663741 | 0.1488608 | 1.0103461 |
| 1966 | 2883071824 | 3082537701 | 2877970424 | -0.0601130 | 0.9514006 |
| 1967 | 2744821593 | 2941375070 | 2738756093 | -0.1708434 | 0.8019865 |
| 1968 | 2859020482 | 3021220759 | 2838969682 | 0.0476535 | 0.8411292 |
| 1969 | 2995595955 | 3180276732 | 2976017755 | 0.1936074 | 1.0208110 |
| 1970 | 3289726569 | 3490531746 | 3284116169 | 0.2190917 | 1.2708540 |
| 1971 | 3657625234 | 3856485856 | 3654919813 | 0.0732409 | 1.3674258 |
| 1972 | 3420056087 | 3821528621 | 3417179485 | 0.0337448 | 1.4143567 |
| 1973 | 3323261015 | 3693691834 | 3320568292 | 0.5501115 | 2.4517114 |
| 1974 | 3501940970 | 3869173264 | 3496834422 | -0.0500974 | 2.3319128 |
| 1975 | 3268937259 | 3637613906 | 3265136356 | -0.1298601 | 2.0479285 |
| 1976 | 3550765040 | 3917851484 | 3545559881 | 0.2572091 | 2.6486176 |
| 1977 | 3456110039 | 3824752205 | 3452906402 | 0.0717087 | 2.8455220 |
| 1978 | 4293546946 | 4446059629 | 4289847393 | 0.2304973 | 3.5831554 |
| 1979 | 4415372581 | 4567973390 | 4411863460 | 0.1452215 | 4.1431871 |
| 1980 | 4413228333 | 4565856246 | 4410212249 | 0.0175105 | 4.2163753 |
| 1981 | 4208881700 | 4199769740 | 4195935855 | -0.1134433 | 3.7641892 |
| 1982 | 4814459232 | 4786892219 | 4801533282 | 0.1436368 | 4.3456236 |
| 1983 | 4992086437 | 4957376601 | 4978087659 | 0.0760852 | 4.6891649 |
| 1984 | 4906589699 | 4890270598 | 4866852348 | -0.0021455 | 4.6791151 |
| 1985 | 4678743893 | 4674097298 | 4649387420 | -0.0831762 | 4.3056703 |
| 1986 | 4349847101 | 4330715720 | 4327978634 | -0.0039701 | 4.2886103 |
| 1987 | 4788480254 | 4696479795 | 4763388776 | 0.1316480 | 4.8920464 |
| 1988 | 4041389595 | 3899842610 | 4004196923 | -0.0837429 | 4.4990572 |
| 1989 | 4023369359 | 3842546597 | 3974910973 | -0.0804014 | 4.1514865 |
| 1990 | 4021450928 | 3878674910 | 3979163122 | 0.0285336 | 4.2716495 |
| 1991 | 4032118379 | 3865661788 | 3995315782 | 0.0761421 | 4.6096048 |
| 1992 | 3725995400 | 3582777275 | 3687951116 | -0.0380712 | 4.4374100 |
| 1993 | 4105010022 | 3942994707 | 4062059127 | 0.1099665 | 4.9532174 |
| 1994 | 4372163988 | 4217669145 | 4336688968 | 0.1186832 | 5.5773881 |
| 1995 | 3888584141 | 3711645486 | 3849488988 | -0.0478938 | 5.3165617 |
| 1996 | 3876679172 | 3715532966 | 3836741499 | -0.0489903 | 5.0623787 |
| 1997 | 4094095068 | 3936173876 | 4057326451 | 0.1098330 | 5.6500784 |
| 1998 | 3731867387 | 3582969082 | 3697377040 | -0.0868527 | 5.1800604 |
| 1999 | 3990116584 | 3830900260 | 3962104908 | 0.0620856 | 5.5118611 |
| 2000 | 3793869254 | 3576918541 | 3733876374 | 0.0167412 | 5.6049132 |
| 2001 | 3765148707 | 3547935875 | 3688174498 | -0.0834742 | 5.1560427 |
| 2002 | 3722911790 | 3474372249 | 3612665910 | -0.1562316 | 4.4102783 |
| 2003 | 3704927643 | 3448902742 | 3606132813 | -0.0195427 | 4.3249264 |
| 2004 | 1889813769 | 2016983232 | 1860167720 | -0.4037095 | 2.8883506 |
| 2005 | 1741044062 | 1870255471 | 1705470703 | 0.0884218 | 3.1553751 |
| 2006 | 1753094099 | 1890964774 | 1726077750 | 0.2093299 | 3.8901093 |
| 2007 | 517159611 | 667070680 | 475161635 | -1.3996532 | 0.9596219 |
| 2008 | 521159877 | 670125837 | 486480769 | 0.0043617 | 0.9638166 |
| 2009 | 534736167 | 677145565 | 494251216 | -0.0363905 | 0.9293733 |
| 2010 | 605385696 | 714949228 | 551050632 | 0.0732448 | 1.0000000 |
| 2011 | 661229735 | 755626132 | 601024253 | 0.0834047 | 1.0869816 |
| 2012 | 655629951 | 772527014 | 602002953 | 0.0405896 | 1.1320094 |
| 2013 | 658197330 | 734787347 | 590869779 | -0.0059572 | 1.1252858 |
| 2014 | 587575682 | 720097666 | 519339299 | -0.0424983 | 1.0784650 |
| 2015 | 577149068 | 699010404 | 533666838 | -0.0443819 | 1.0316473 |
| 2016 | 556835463 | 711262899 | 519781163 | 0.0820077 | 1.1198162 |
| 2017 | 525822797 | 668362521 | 488629852 | -0.0481920 | 1.0671298 |
| 2018 | 490972000 | 671036323 | 439699415 | 0.0076003 | 1.0752712 |
| 2019 | 430592128 | 622763099 | 397062280 | -0.0190156 | 1.0550175 |

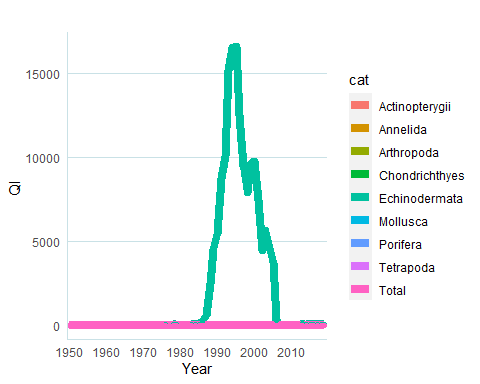
Did all of the analyses work as intended?

*list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = 1:2, slopecheck = c(1, 1), Freq = c(2, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = 1:2, slopecheck = c(1, 1), Freq = c(2, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(8, 1, 18, 9, 8, 10)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = 1:2, slopecheck = c(1, 1), Freq = c(2, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(8, 1, 18, 9, 8, 10)), FYI: 10\_0000Other is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = 1:2, slopecheck = c(1, 1), Freq = c(2, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(8, 1, 18, 9, 8, 10)), FYI: 10\_0000Other is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, FYI: 11\_0000Plantae is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = 1:2, slopecheck = c(1, 1), Freq = c(2, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(8, 1, 18, 9, 8, 10)), FYI: 10\_0000Other is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, FYI: 11\_0000Plantae is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 2, 2), slopecheck = c(1, 2, 1, 2), Freq = c(1, 2, 0, 3)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(64, 21, 119, 57, 84, 63)), FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(6, 1, 18, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(5, 4, 5, 5, 4, 5)), FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, list(var = 1:2, slopecheck = c(1, 1), Freq = c(2, 2)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(8, 1, 18, 9, 8, 10)), FYI: 10\_0000Other is no longer being calculated because there were no more available columns o P after data was removed for not meeting the pctmiss, FYI: 11\_0000Plantae is no longer being calculated because there were less than 2 columns of P available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 2, 2), slopecheck = c(1, 2, 1, 2), Freq = c(1, 2, 0, 3)), list(var = c(1, 1, 1, 2, 2, 2), slopecheck = c(1, 2, 3, 1, 2, 3), Freq = c(1, 1, 1, 0, 3, 0)), Warning: When back calculated, ln(Q\_t/Q\_{t-1}) = did not equal sum( ( R\_{i, t} - R\_{i, t-1} ) / 2 ) x ln( (Q\_{i,t}) / (Q\_{i,t-1} ) ), FYI: 2 of species V columns are completely empty, 2 of species Q columns are completely empty.*

#### E. Graph

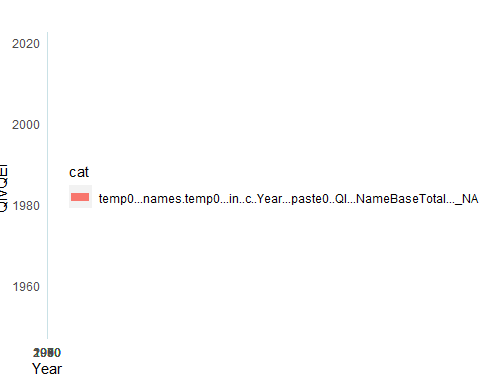
##### Graph 1: Quantity Index Categories

For comparison, let’s recreate those graphs to make sure we are getting the same output:

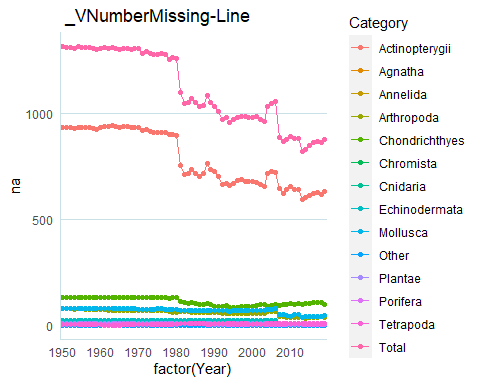


##### Graph 2: Quantity Index Compare

For comparison, let’s recreate those graphs to make sure we are getting the same output:



##### Graph 3: Missing values of V



## PRICE METHOD

In most of the following examples, we will just focus on the finfish (*i*=1) side of the equation. Here *baseyr* is set to 2010 and the *pctmiss* (The percent of data in a column that we will allow to be missing for analysis; more on that later) is set to 0.5%.

Alternatively, we have a price model method to calculate implicit quantity. Here, on top of all the work that is done for the Quantity-derived output, we also calculate price and use price to weigh the revenue share.

Essential by calculating price we are developing a deflator for the total landings values: We use the total value were was available () to calculate and extrapolate by dividing the total value ()

### At the species level:

#### Remove any V and Q data where V column has less data than the specifed

*No warning.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Q1\_1Salmon | Q1\_2Cod | Q2\_1Shrimp | Q2\_2Clam | REMOVED\_Q1\_3Flounder | Q1\_4SeaBass |
| 2007 | NA | 2000 | 100 | 150 | NA | 1000 |
| 2008 | NA | 1900 | 120 | 160 | NA | 1200 |
| 2009 | NA | 2000 | 110 | 140 | NA | 900 |
| 2010 | 20 | 2500 | 90 | NA | NA | NA |
| 2011 | 10 | 2400 | 80 | NA | NA | NA |
| 2012 | 12 | 2300 | 100 | NA | NA | NA |
| 2013 | 11 | 2000 | 100 | 140 | NA | 1000 |
| 2014 | 11 | 2300 | 110 | 110 | NA | 900 |
| 2015 | 10 | 2400 | 90 | 130 | NA | 1000 |
| 2016 | 15 | 2200 | 100 | 160 | NA | 1100 |

#### Caluclate New Category Sums of and

Because we removed some columns for not meeting a perecent missing threshold of 0.5% and those columns will not be used at all in any part of the further analysis, we need to re-calculate the totals of and for the catagories and the fishery as a whole.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | QE1\_0Finfish | VE1\_0Finfish | Q1\_0Finfish | V1\_0Finfish |
| 2007 | 3000 | 3800 | 3000 | 2800 |
| 2008 | 3100 | 4020 | 3100 | 2820 |
| 2009 | 2900 | 3910 | 2900 | 3010 |
| 2010 | 2520 | 3190 | 2520 | 3190 |
| 2011 | 2410 | 3280 | 2410 | 3280 |
| 2012 | 2312 | 3150 | 2312 | 3150 |
| 2013 | 3011 | 4080 | 3011 | 3080 |
| 2014 | 3211 | 4270 | 3211 | 3370 |
| 2015 | 3410 | 4700 | 3410 | 3700 |
| 2016 | 3315 | 4480 | 3315 | 3380 |

#### Price

We first measure output price for each species (; e.g., Salmon and Flounder) in each of the categories (e.g., Finfish & Others and Shellfish) using detailed landings time series data on value ($) and pounds (lbs).

Price for a species (s) of category (i) in year (t) =

where:

* is the price per individual species (s), category (i), for each year (t)
* is the quantity (lb) per individual species (s), category (i), for each year (t)
* is the value ($) per category (i), for each year (t)

|  |  |  |  |
| --- | --- | --- | --- |
|  | P1\_1Salmon | P1\_2Cod | P1\_4SeaBass |
| 2007 | NA | 1.400000 | NA |
| 2008 | NA | 1.421053 | 0.1000000 |
| 2009 | NA | 1.450000 | 0.1222222 |
| 2010 | 5.00000 | 1.200000 | NA |
| 2011 | 10.00000 | 1.291667 | NA |
| 2012 | 12.50000 | 1.260870 | NA |
| 2013 | 16.36364 | 1.400000 | 0.1000000 |
| 2014 | 15.45455 | 1.391304 | NA |
| 2015 | 20.00000 | 1.458333 | NA |
| 2016 | 12.00000 | 1.454546 | NA |

##### Address Infrequent Missing Data

There may be instances where there are no or too few Q data for that species in a year or ever. The next goal will be to calculate the quantity change, so we need to have a value in there that won’t show change. If we left an NA (which would be treated as a 0) in the cell, then the change from year to year would be very large and misrepresent the index trend. To avoid this, we do the following:

###### 1. If there are instances for a species where there are too few pairs of and/or are completely missing from the timeseries or where a percent of is missing from the timeseries, we will remove the offending price columns entierly, so they don’t influence the downstream price change or price index calculations.

Let’s say here that if 50% of the data is missing in a given , don’t calculate that species

|  |  |  |
| --- | --- | --- |
|  | P1\_1Salmon | P1\_2Cod |
| 2007 | NA | 1.400000 |
| 2008 | NA | 1.421053 |
| 2009 | NA | 1.450000 |
| 2010 | 5.00000 | 1.200000 |
| 2011 | 10.00000 | 1.291667 |
| 2012 | 12.50000 | 1.260870 |
| 2013 | 16.36364 | 1.400000 |
| 2014 | 15.45455 | 1.391304 |
| 2015 | 20.00000 | 1.458333 |
| 2016 | 12.00000 | 1.454546 |

###### 2. If the first value of is 0/NA in a timeseries, we (impute) let the next available non-zero/non-NA value of P in the timeseries inform the past.

|  |  |  |
| --- | --- | --- |
|  | P1\_1Salmon | P1\_2Cod |
| 2007 | 5.00000 | 1.400000 |
| 2008 | NA | 1.421053 |
| 2009 | NA | 1.450000 |
| 2010 | 5.00000 | 1.200000 |
| 2011 | 10.00000 | 1.291667 |
| 2012 | 12.50000 | 1.260870 |
| 2013 | 16.36364 | 1.400000 |
| 2014 | 15.45455 | 1.391304 |
| 2015 | 20.00000 | 1.458333 |
| 2016 | 12.00000 | 1.454546 |

###### 3. If there is a value in the middle of ’s timeseries that is 0/NA, we (impute) let the most recent past available non-zero/non-NA of in the timeseries inform the future.

|  |  |  |
| --- | --- | --- |
|  | P1\_1Salmon | P1\_2Cod |
| 2007 | 5.00000 | 1.400000 |
| 2008 | 5.00000 | 1.421053 |
| 2009 | 5.00000 | 1.450000 |
| 2010 | 5.00000 | 1.200000 |
| 2011 | 10.00000 | 1.291667 |
| 2012 | 12.50000 | 1.260870 |
| 2013 | 16.36364 | 1.400000 |
| 2014 | 15.45455 | 1.391304 |
| 2015 | 20.00000 | 1.458333 |
| 2016 | 12.00000 | 1.454546 |

##### Impute values of where P was able to be calculated

To ensure that the price index does not rise or fall to quickly with changes (that are really because of NA values) we fill in the missing instances of .

###### 1. If the first value of is 0/NA in a timeseries, we let the next available non-zero value of in the timeseries inform the past.

|  |  |  |
| --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod |
| 2007 | 100 | 2800 |
| 2008 | NA | 2700 |
| 2009 | NA | 2900 |
| 2010 | 100 | 3000 |
| 2011 | 100 | 3100 |
| 2012 | 150 | 2900 |
| 2013 | 180 | 2800 |
| 2014 | 170 | 3200 |
| 2015 | 200 | 3500 |
| 2016 | 180 | 3200 |

###### 2. If there is a value in the middle of ’s timeseries that is 0/NA, we let the most recent past available non-zero of in the timeseries inform the future.

|  |  |  |
| --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod |
| 2007 | 100 | 2800 |
| 2008 | 100 | 2700 |
| 2009 | 100 | 2900 |
| 2010 | 100 | 3000 |
| 2011 | 100 | 3100 |
| 2012 | 150 | 2900 |
| 2013 | 180 | 2800 |
| 2014 | 170 | 3200 |
| 2015 | 200 | 3500 |
| 2016 | 180 | 3200 |

#### Total value of species ()

And then I follow similar steps or the category and national level.

#### Total value of species where P is available ()

A.K.A.: Value of species where P was able to be calculated

, as defined and discussed in the subsequent step, will need to sum to 1 across all species in a category. Therefore, you will need to sum a new total of available (called ) for the category using only values for species that were used to calculate (called ).

When I say “available” here, I am asking how many values of P were we able to calculate. As you can see here, even though there were plenty of Q and V, they didn’t amount to many P. Even if a value of P for a species doesn’t make the cut, that gets applied to the total value of the category.

where:

* is the new total of (called ) for the category using only values for species that were used to calculate
* are the where P were able to be calculated

|  |  |  |  |
| --- | --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod | VV1\_0Finfish |
| 2007 | 100 | 2800 | 2900 |
| 2008 | 100 | 2700 | 2800 |
| 2009 | 100 | 2900 | 3000 |
| 2010 | 100 | 3000 | 3100 |
| 2011 | 100 | 3100 | 3200 |
| 2012 | 150 | 2900 | 3050 |
| 2013 | 180 | 2800 | 2980 |
| 2014 | 170 | 3200 | 3370 |
| 2015 | 200 | 3500 | 3700 |
| 2016 | 180 | 3200 | 3380 |

##### Analysis Warnings Checks

Just so we can get a sense of the data, we want to see how many species are significantly increasing or decreasing over time for V and Q.

We’ll use the below function to collect our info:

## function (Columns, temp)   
## {  
## lm\_check <- data.frame(col = rep\_len(x = NA, length.out = length(Columns)),   
## slope = rep\_len(x = NA, length.out = length(Columns)),   
## intercept = rep\_len(x = NA, length.out = length(Columns)),   
## R2 = rep\_len(x = NA, length.out = length(Columns)), R2adj = rep\_len(x = NA,   
## length.out = length(Columns)), Pr = rep\_len(x = NA,   
## length.out = length(Columns)), Fstat = rep\_len(x = NA,   
## length.out = length(Columns)))  
## for (c0 in 1:length(Columns)) {  
## if (sum(is.na(temp[, Columns[c0]])) == length(temp[,   
## Columns[c0]]) | length(temp[, Columns[c0]]) %in%   
## sum(temp[, Columns[c0]] %in% c(NA, 0))) {  
## lm\_check$col[c0] <- NA  
## lm\_check$slope[c0] <- NA  
## lm\_check$intercept[c0] <- NA  
## lm\_check$R2[c0] <- NA  
## lm\_check$R2adj[c0] <- NA  
## lm\_check$Pr[c0] <- NA  
## lm\_check$Fstat[c0] <- NA  
## }  
## else {  
## temp0 <- summary(lm(rownames(temp) ~ temp[, Columns[c0]]))  
## lm\_check$col[c0] <- Columns[c0]  
## lm\_check$slope[c0] <- temp0$coefficients[2]  
## lm\_check$intercept[c0] <- temp0$coefficients[1]  
## lm\_check$R2[c0] <- temp0$r.squared  
## lm\_check$R2adj[c0] <- temp0$adj.r.squared  
## lm\_check$Pr[c0] <- temp0$coefficients[8]  
## lm\_check$Fstat[c0] <- ifelse(is.null(temp0$fstatistic[1]),   
## NA, as.numeric(temp0$fstatistic[1]))  
## }  
## }  
## lm\_check$var <- substr(x = Columns, 1, 1)  
## lm\_check$slopecheck <- "Insig"  
## lm\_check$slopecheck <- ifelse(lm\_check$slope >= 0 & lm\_check$Pr <=   
## 0.05, "Sig Pos", "Insig")  
## lm\_check$slopecheck <- ifelse(lm\_check$slope < 0 & lm\_check$Pr <=   
## 0.05, "Sig Neg", lm\_check$slopecheck)  
## return(lm\_check)  
## }  
## <bytecode: 0x000000002326c3a0>  
## <environment: namespace:FishEconProdOutput>

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | NameBasecategory | col | slope | intercept | R2 | R2adj | Pr | Fstat | var | slopecheck |
| 1 | 1\_0Finfish | P1\_1Salmon | 0.4644811 | 2006.562 | 0.7230807 | 0.6884658 | 0.0018246 | 20.8892823 | P | Sig Pos |
| 2 | 1\_0Finfish | P1\_2Cod | 7.0307005 | 2001.848 | 0.0437863 | -0.0757404 | 0.5617834 | 0.3663306 | P | Insig |
| 3 | 1\_0Finfish | V1\_1Salmon | 0.0653553 | 2002.481 | 0.8159514 | 0.7929453 | 0.0003399 | 35.4667781 | V | Sig Pos |
| 4 | 1\_0Finfish | V1\_2Cod | 0.0093573 | 1983.335 | 0.5614367 | 0.5066163 | 0.0126083 | 10.2413793 | V | Sig Pos |
| 5 | 1\_0Finfish | Q1\_1Salmon | -0.2014388 | 2015.561 | 0.1151079 | -0.0618705 | 0.4565969 | 0.6504065 | Q | Insig |
| 6 | 1\_0Finfish | Q1\_2Cod | 0.0065000 | 1997.200 | 0.2048485 | 0.1054545 | 0.1890352 | 2.0609756 | Q | Insig |

How many slopes are significantly increaseing or decreaseing

|  |  |  |  |
| --- | --- | --- | --- |
|  | var | slopecheck | Freq |
| 1 | P | Insig | 1 |
| 4 | P | Sig Pos | 1 |
| 2 | Q | Insig | 2 |
| 5 | Q | Sig Pos | 0 |
| 3 | V | Insig | 0 |
| 6 | V | Sig Pos | 2 |

#### 

Value of species where Q available

, as defined and discussed in the subsequent step, will need to sum to 1 across all species in a category. Therefore, you will need to sum a new total of available (called ) for the category using only values for species that were used to calculate (called ).

where:

* is the new total of (called ) for the category using only values for species that were used to calculate
* are the where were able to be calculated

|  |  |  |  |
| --- | --- | --- | --- |
|  | V1\_1Salmon | V1\_2Cod | VV1\_0Finfish |
| 2007 | 100 | 2800 | 2900 |
| 2008 | 100 | 2700 | 2800 |
| 2009 | 100 | 2900 | 3000 |
| 2010 | 100 | 3000 | 3100 |
| 2011 | 100 | 3100 | 3200 |
| 2012 | 150 | 2900 | 3050 |
| 2013 | 180 | 2800 | 2980 |
| 2014 | 170 | 3200 | 3370 |
| 2015 | 200 | 3500 | 3700 |
| 2016 | 180 | 3200 | 3380 |

#### Revenue-share

A.K.A.: Revenue Share for each species (; e.g., Salmon and Flounder). Here we divide by because only includes species used to calculate as per the above price calculations.

where:

* is the revenue share per individual species (s), category (i), for each year (t)
* is the value ($) per individual species (s), category (i), for each year (t)

|  |  |  |  |
| --- | --- | --- | --- |
|  | R1\_1Salmon | R1\_2Cod | R1\_4SeaBass |
| 2007 | 0.0344828 | 0.9655172 | NA |
| 2008 | 0.0357143 | 0.9642857 | 0.0428571 |
| 2009 | 0.0333333 | 0.9666667 | 0.0366667 |
| 2010 | 0.0322581 | 0.9677419 | 0.0290323 |
| 2011 | 0.0312500 | 0.9687500 | 0.0250000 |
| 2012 | 0.0491803 | 0.9508197 | 0.0327869 |
| 2013 | 0.0604027 | 0.9395973 | 0.0335570 |
| 2014 | 0.0504451 | 0.9495549 | NA |
| 2015 | 0.0540541 | 0.9459459 | NA |
| 2016 | 0.0532544 | 0.9467456 | NA |

##### Analysis Warnings Checks

As an additional check, let’s make sure that each row sums to 1.

|  |  |
| --- | --- |
|  | x |
| 2007 | 1.000000 |
| 2008 | 1.042857 |
| 2009 | 1.036667 |
| 2010 | 1.029032 |
| 2011 | 1.025000 |
| 2012 | 1.032787 |
| 2013 | 1.033557 |
| 2014 | 1.000000 |
| 2015 | 1.000000 |
| 2016 | 1.000000 |

Is there a warning?

*Rows of R\_{t,i,s} for 1\_0Finfish did not sum to 1*

#### Revenue-share weighted price changes

Where:

* = Revenue share-weighted price change for a species (s)

Such that:

* category’s (i) Price Change for each species (s) =
* category’s (i) Revenue Share for each species (s) =

### At the fishery level:

#### Value of categories available

#### Price change

A.K.A., Price Changes for the category (; e.g., Finfish)

Where:

* = Price change for a category (i)

|  |  |  |  |
| --- | --- | --- | --- |
|  | PCW1\_1Salmon | PCW1\_2Cod | PC1\_0Finfish |
| 2007 | 0.0000000 | 0.0000000 | 0.0000000 |
| 2008 | 0.0000000 | 0.0144018 | 0.0144018 |
| 2009 | 0.0000000 | 0.0194695 | 0.0194695 |
| 2010 | 0.0000000 | -0.1830357 | -0.1830357 |
| 2011 | 0.0220102 | 0.0712743 | 0.0932846 |
| 2012 | 0.0089738 | -0.0231613 | -0.0141875 |
| 2013 | 0.0147572 | 0.0989356 | 0.1136927 |
| 2014 | -0.0031679 | -0.0058852 | -0.0090532 |
| 2015 | 0.0134715 | 0.0445941 | 0.0580655 |
| 2016 | -0.0274080 | -0.0024612 | -0.0298692 |

#### Price index

A.K.A.: Price Index for the each category ()

We calculate the price index first by comparing by multiplying the previous years by that year’s price change , where the PI of the first year

Where

Then, to change the price index into base year dollars, we use the following equation:

In this example, our base year is 2010. Notice that the

And we add the to the data

|  |  |  |  |
| --- | --- | --- | --- |
|  | tempPI\_yr1 | tempPI\_yrb | PI1\_0Finfish |
| 2007 | 1.0000000 | 1.160864 | 1.160864 |
| 2008 | 1.0145060 | 1.177703 | 1.177703 |
| 2009 | 1.0344514 | 1.200857 | 1.200857 |
| 2010 | 0.8614275 | 1.000000 | 1.000000 |
| 2011 | 0.9456527 | 1.097774 | 1.097774 |
| 2012 | 0.9323310 | 1.082309 | 1.082309 |
| 2013 | 1.0445909 | 1.212628 | 1.212628 |
| 2014 | 1.0351767 | 1.201699 | 1.201699 |
| 2015 | 1.0970642 | 1.273542 | 1.273542 |
| 2016 | 1.0647803 | 1.236065 | 1.236065 |

where and then

#### Implicit quantity

Note here that all columns of are being used, despite having been removed earlier in the analysis when could not be calculated and columns have functionally been removed from the analysis.

|  |  |
| --- | --- |
|  | temp…ncol.temp.. |
| 2007 | 3273.424 |
| 2008 | 3413.423 |
| 2009 | 3256.007 |
| 2010 | 3190.000 |
| 2011 | 2987.864 |
| 2012 | 2910.443 |
| 2013 | 3364.594 |
| 2014 | 3553.302 |
| 2015 | 3690.494 |
| 2016 | 3624.405 |

#### Implicit quantity index

|  |  |
| --- | --- |
|  | QI1\_0Finfish |
| 2007 | 1.0261518 |
| 2008 | 1.0700387 |
| 2009 | 1.0206920 |
| 2010 | 1.0000000 |
| 2011 | 0.9366346 |
| 2012 | 0.9123647 |
| 2013 | 1.0547316 |
| 2014 | 1.1138877 |
| 2015 | 1.1568948 |
| 2016 | 1.1361771 |

##### Analysis Warnings Checks

1. When back calculated, should equal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | V1\_0Finfish | PI1\_0Finfish | Q1\_0Finfish | V1\_0Finfish\_Check |
| 2007 | 2800 | 1.160864 | 3273.424 | 3800 |
| 2008 | 2820 | 1.177703 | 3413.423 | 4020 |
| 2009 | 3010 | 1.200857 | 3256.007 | 3910 |
| 2010 | 3190 | 1.000000 | 3190.000 | 3190 |
| 2011 | 3280 | 1.097774 | 2987.864 | 3280 |
| 2012 | 3150 | 1.082309 | 2910.443 | 3150 |
| 2013 | 3080 | 1.212628 | 3364.594 | 4080 |
| 2014 | 3370 | 1.201699 | 3553.302 | 4270 |
| 2015 | 3700 | 1.273542 | 3690.494 | 4700 |
| 2016 | 3380 | 1.236065 | 3624.405 | 4480 |

Is there a warning?

*Warning: When back calculated, V\_{t,i} did not equal PI\_{t,i}*  Q\_{t,i}\*

1. When back calculated, should equal

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | V1\_0Finfish | PI1\_0Finfish | Q1\_0Finfish | V1\_0Finfish\_Check | Q1\_0Finfish\_Check |
| 2007 | 2800 | 1.160864 | 3273.424 | 3800 | 2411.997 |
| 2008 | 2820 | 1.177703 | 3413.423 | 4020 | 2394.491 |
| 2009 | 3010 | 1.200857 | 3256.007 | 3910 | 2506.543 |
| 2010 | 3190 | 1.000000 | 3190.000 | 3190 | 3190.000 |
| 2011 | 3280 | 1.097774 | 2987.864 | 3280 | 2987.864 |
| 2012 | 3150 | 1.082309 | 2910.443 | 3150 | 2910.443 |
| 2013 | 3080 | 1.212628 | 3364.594 | 4080 | 2539.938 |
| 2014 | 3370 | 1.201699 | 3553.302 | 4270 | 2804.362 |
| 2015 | 3700 | 1.273542 | 3690.494 | 4700 | 2905.283 |
| 2016 | 3380 | 1.236065 | 3624.405 | 4480 | 2734.484 |

Is there a warning?

*Warning: When back calculated, Q\_{t,i} did not equal V\_{t,i}/PI\_{t,i}*

#### Redo Analysis for Other Categories

Now lets redo that whole analysis up to this point (via function) for the two species of the shellfish group, as we will need them for the next steps of this analysis.

We use the *PriceMethodOutput\_Category* function to calculate everything we did above at category level.

What does the Shellfish data look like?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | PCW2\_1Shrimp | PCW2\_2Clam | PC2\_0Shellfish | PI2\_0Shellfish | Q2\_0Shellfish | QI2\_0Shellfish |
| 2007 | 0.0000000 | 0.0000000 | 0.0000000 | 1.030337 | 1747.0009 | 2.495716 |
| 2008 | 0.0183493 | 0.0648402 | 0.0831894 | 1.119717 | 1964.7830 | 2.806833 |
| 2009 | -0.0087575 | -0.0805788 | -0.0893363 | 1.024023 | 1757.7726 | 2.511104 |
| 2010 | -0.0237392 | 0.0000000 | -0.0237392 | 1.000000 | 700.0000 | 1.000000 |
| 2011 | 0.1730144 | 0.0000000 | 0.1730144 | 1.188883 | 757.0129 | 1.081447 |
| 2012 | -0.0604413 | 0.0000000 | -0.0604413 | 1.119154 | 893.5320 | 1.276474 |
| 2013 | 0.0977034 | 0.0488994 | 0.1466028 | 1.295862 | 1697.7119 | 2.425303 |
| 2014 | -0.0998625 | 0.0614193 | -0.0384432 | 1.246990 | 1603.8619 | 2.291231 |
| 2015 | 0.0553143 | -0.0293044 | 0.0260098 | 1.279850 | 1562.6836 | 2.232405 |
| 2016 | 0.0393171 | -0.0549436 | -0.0156266 | 1.260005 | 1825.3889 | 2.607699 |

#### Value for all fisheries for species where P was able to be calculated

, defined and discussed in the subsequent step, will need to sum to 1 across all species in a category. Therefore, you will need to sum a new total of (called ) for the category using only values for species that were used to calculate .

where:

* is the new total of for the entire fishery using only values for species that were used to calculate

#### Revenue share

Revenue Share for the each category ()

where:

* is the revenue share per individual species (s), category (i), for each year (t)
* is the value ($) per individual species (s), category (i), for each year (t)

Here, we don’t use beacause we want to expand the proportion to include all of the species caught, regardless if they were used in the price calculations.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | R1\_0Finfish | R2\_0Shellfish | V1\_0Finfish | V2\_0Shellfish | V0\_0Total |
| 2007 | 0.6086957 | 0.3913043 | 2800 | 1800 | 4600 |
| 2008 | 0.5617530 | 0.4382470 | 2820 | 2200 | 5020 |
| 2009 | 0.6257796 | 0.3742204 | 3010 | 1800 | 4810 |
| 2010 | 0.8200514 | 0.1799486 | 3190 | 700 | 3890 |
| 2011 | 0.7846890 | 0.2153110 | 3280 | 900 | 4180 |
| 2012 | 0.7590361 | 0.2409639 | 3150 | 1000 | 4150 |
| 2013 | 0.5833333 | 0.4166667 | 3080 | 2200 | 5280 |
| 2014 | 0.6275605 | 0.3724395 | 3370 | 2000 | 5370 |
| 2015 | 0.6491228 | 0.3508772 | 3700 | 2000 | 5700 |
| 2016 | 0.5950704 | 0.4049296 | 3380 | 2300 | 5680 |

##### Analysis Warnings Checks

As an additional check, let’s make sure that each row sums to 1.

|  |  |
| --- | --- |
|  | x |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 1 |
| 7 | 1 |
| 8 | 1 |
| 9 | 1 |
| 10 | 1 |

Is there a warning?

*No warning.*

#### Revenue share weighted price changes

Revenue Share-Weighted Price Changes for each category (; e.g., Salmon and Flounder)

Where:

* = Revenue share-weighted price change for a category (i)

Such that:

* Price Change for each category (i) =
* Revenue Share for each category (i) =

### At the entire commercial fisheries sector level:

#### Price change

Price Changes for the entire fishery (; e.g., Finfish)

Where:

* = Price change for the entire fishery

|  |  |  |  |
| --- | --- | --- | --- |
|  | PCW1\_0Finfish | PCW2\_0Shellfish | PC0\_0Total |
| 2007 | 0.0000000 | 0.0000000 | 0.0000000 |
| 2008 | 0.0084283 | 0.0345050 | 0.0429332 |
| 2009 | 0.0115603 | -0.0362914 | -0.0247311 |
| 2010 | -0.1323193 | -0.0065778 | -0.1388971 |
| 2011 | 0.0748488 | 0.0341928 | 0.1090416 |
| 2012 | -0.0109508 | -0.0137889 | -0.0247398 |
| 2013 | 0.0763088 | 0.0482052 | 0.1245141 |
| 2014 | -0.0054812 | -0.0151679 | -0.0206491 |
| 2015 | 0.0370656 | 0.0094067 | 0.0464723 |
| 2016 | -0.0185815 | -0.0059053 | -0.0244869 |

#### Price index

We calculate the price index first by comparing by multiplying the previous years by that year’s price change , where the PI of the first year

Where

|  |  |
| --- | --- |
|  | PI0\_0Total |
| 2007 | 1.128281 |
| 2008 | 1.177776 |
| 2009 | 1.149006 |
| 2010 | 1.000000 |
| 2011 | 1.115209 |
| 2012 | 1.087957 |
| 2013 | 1.232218 |
| 2014 | 1.207035 |
| 2015 | 1.264452 |
| 2016 | 1.233866 |

#### Implicit quantity

|  |  |
| --- | --- |
|  | Q0\_0Total |
| 2007 | 4963.304 |
| 2008 | 5281.138 |
| 2009 | 4969.513 |
| 2010 | 3890.000 |
| 2011 | 3748.177 |
| 2012 | 3814.488 |
| 2013 | 5096.500 |
| 2014 | 5194.548 |
| 2015 | 5298.737 |
| 2016 | 5494.925 |

#### Implicit quantity index

where and then

#### Simple Sum Quantity Output Index

Where:

* is the sum of Q before these calculations; the simple sum
* is the index of the sum of Q before these equations

|  |  |
| --- | --- |
|  | temp…paste0..QEI…NameBaseTotal.. |
| 2007 | 1.2452107 |
| 2008 | 1.2950192 |
| 2009 | 1.2068966 |
| 2010 | 1.0000000 |
| 2011 | 0.9540230 |
| 2012 | 0.9241379 |
| 2013 | 1.2455939 |
| 2014 | 1.3145594 |
| 2015 | 1.3908046 |
| 2016 | 1.3697318 |

#### Quantity change

Same as before, these are the values that would go into the output portion of the output equation. This method is good for data that are missing many of the quantity values.

Solve Output portion of the equation for the Output Changes:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Q0\_0Total | QI0\_0Total | QC0\_0Total |
| 2007 | 4963.304 | 1.2759136 | 0.0000000 |
| 2008 | 5281.138 | 1.3576191 | 0.0732370 |
| 2009 | 4969.513 | 1.2775099 | -0.0732619 |
| 2010 | 3890.000 | 1.0000000 | -0.2699238 |
| 2011 | 3748.177 | 0.9635417 | -0.0370504 |
| 2012 | 3814.488 | 0.9805883 | 0.0175616 |
| 2013 | 5096.500 | 1.3101544 | 0.3083746 |
| 2014 | 5194.548 | 1.3353593 | 0.0106022 |
| 2015 | 5298.737 | 1.3621431 | 0.0147757 |
| 2016 | 5494.925 | 1.4125770 | 0.0474804 |

##### Analysis Warnings Checks

To make sure our analyses worked as inteded, let’s see if we can back calculate our numbers.

We want the calcuated V to equal this check:

1. When back calculated, should equal ?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | V0\_0Total | PI0\_0Total | Q0\_0Total | V0\_0Total\_Check |
| 2007 | 4600 | 1.128281 | 4963.304 | 5600 |
| 2008 | 5020 | 1.177776 | 5281.138 | 6220 |
| 2009 | 4810 | 1.149006 | 4969.513 | 5710 |
| 2010 | 3890 | 1.000000 | 3890.000 | 3890 |
| 2011 | 4180 | 1.115209 | 3748.177 | 4180 |
| 2012 | 4150 | 1.087957 | 3814.488 | 4150 |
| 2013 | 5280 | 1.232218 | 5096.500 | 6280 |
| 2014 | 5370 | 1.207035 | 5194.548 | 6270 |
| 2015 | 5700 | 1.264452 | 5298.737 | 6700 |
| 2016 | 5680 | 1.233866 | 5494.925 | 6780 |

Is there a warning?

*Warning: When back calculated, V\_t did not equal PI\_t x Q\_t*

1. When back calculated, should ?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | V0\_0Total | PI0\_0Total | Q0\_0Total | Q0\_0Total\_Check |
| 2007 | 4600 | 1.128281 | 4963.304 | 4077.000 |
| 2008 | 5020 | 1.177776 | 5281.138 | 4262.269 |
| 2009 | 4810 | 1.149006 | 4969.513 | 4186.228 |
| 2010 | 3890 | 1.000000 | 3890.000 | 3890.000 |
| 2011 | 4180 | 1.115209 | 3748.177 | 3748.177 |
| 2012 | 4150 | 1.087957 | 3814.488 | 3814.488 |
| 2013 | 5280 | 1.232218 | 5096.500 | 4284.956 |
| 2014 | 5370 | 1.207035 | 5194.548 | 4448.919 |
| 2015 | 5700 | 1.264452 | 5298.737 | 4507.880 |
| 2016 | 5680 | 1.233866 | 5494.925 | 4603.418 |

Is there a warning?

*Warning: When back calculated, Q\_t did not equal V\_t/PI\_t*

1. When back calculated, growth rate?

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Q0\_0Total | Q1\_0Finfish | R1\_0Finfish | Q2\_0Shellfish | R2\_0Shellfish | part1 | part2 |
| 2007 | 4963.304 | 3273.424 | 0.6086957 | 1747.0009 | 0.3913043 | NA | NA |
| 2008 | 5281.138 | 3413.423 | 0.5617530 | 1964.7830 | 0.4382470 | -0.0620701 | 0.0732370 |
| 2009 | 4969.513 | 3256.007 | 0.6257796 | 1757.7726 | 0.3742204 | 0.0608198 | -0.0732619 |
| 2010 | 3890.000 | 3190.000 | 0.8200514 | 700.0000 | 0.1799486 | 0.2449128 | -0.2699238 |
| 2011 | 3748.177 | 2987.864 | 0.7846890 | 757.0129 | 0.2153110 | 0.0371395 | -0.0370504 |
| 2012 | 3814.488 | 2910.443 | 0.7590361 | 893.5320 | 0.2409639 | -0.0175368 | 0.0175616 |
| 2013 | 5096.500 | 3364.594 | 0.5833333 | 1697.7119 | 0.4166667 | -0.2897476 | 0.3083746 |
| 2014 | 5194.548 | 3553.302 | 0.6275605 | 1603.8619 | 0.3724395 | -0.0190555 | 0.0106022 |
| 2015 | 5298.737 | 3690.494 | 0.6491228 | 1562.6836 | 0.3508772 | -0.0198589 | 0.0147757 |
| 2016 | 5494.925 | 3624.405 | 0.5950704 | 1825.3889 | 0.4049296 | -0.0363564 | 0.0474804 |

Is there a warning?

*Warning: When back calculated, ln(Q\_t/Q\_{t-1}) = did not equal sum( ((R\_{i, t} - R\_{i, t-1})(2))*  ln((Q\_{t,i})(Q\_{t-1,i}))\*

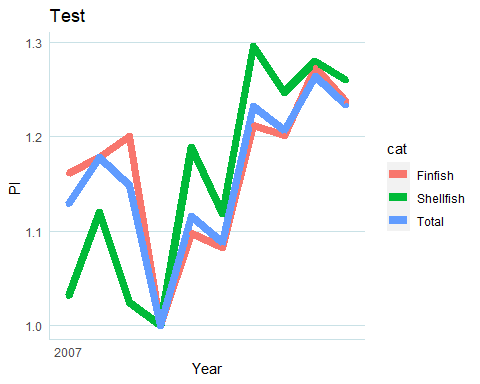
#### View Total Outputs

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | QE0\_0Total | VE0\_0Total | VV0\_0Total | V0\_0Total | PC0\_0Total | PI0\_0Total | Q0\_0Total | QI0\_0Total | QEI0\_0Total | QC0\_0Total |
| 2007 | 3250 | 5600 | 4700 | 4600 | 0.0000000 | 1.128281 | 4963.304 | 1.2759136 | 1.2452107 | 0.0000000 |
| 2008 | 3380 | 6220 | 5000 | 5020 | 0.0429332 | 1.177776 | 5281.138 | 1.3576191 | 1.2950192 | 0.0732370 |
| 2009 | 3150 | 5710 | 4800 | 4810 | -0.0247311 | 1.149006 | 4969.513 | 1.2775099 | 1.2068966 | -0.0732619 |
| 2010 | 2610 | 3890 | 4700 | 3890 | -0.1388971 | 1.000000 | 3890.000 | 1.0000000 | 1.0000000 | -0.2699238 |
| 2011 | 2490 | 4180 | 5000 | 4180 | 0.1090416 | 1.115209 | 3748.177 | 0.9635417 | 0.9540230 | -0.0370504 |
| 2012 | 2412 | 4150 | 4950 | 4150 | -0.0247398 | 1.087957 | 3814.488 | 0.9805883 | 0.9241379 | 0.0175616 |
| 2013 | 3251 | 6280 | 5180 | 5280 | 0.1245141 | 1.232218 | 5096.500 | 1.3101544 | 1.2455939 | 0.3083746 |
| 2014 | 3431 | 6270 | 5370 | 5370 | -0.0206491 | 1.207035 | 5194.548 | 1.3353593 | 1.3145594 | 0.0106022 |
| 2015 | 3630 | 6700 | 5700 | 5700 | 0.0464723 | 1.264452 | 5298.737 | 1.3621431 | 1.3908046 | 0.0147757 |
| 2016 | 3575 | 6780 | 5680 | 5680 | -0.0244869 | 1.233866 | 5494.925 | 1.4125770 | 1.3697318 | 0.0474804 |

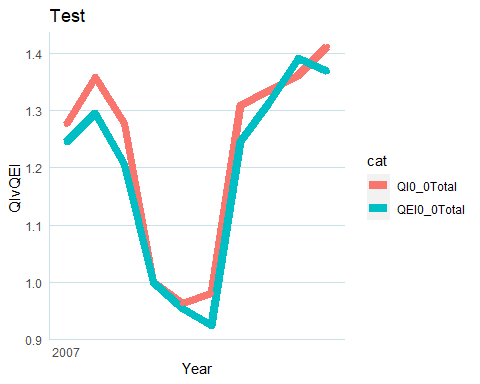
#### How many data were missing at the end of the analysis?

*FYI: 0 of species V columns are completely empty, 1 of species Q columns are completely empty.* #### Plots

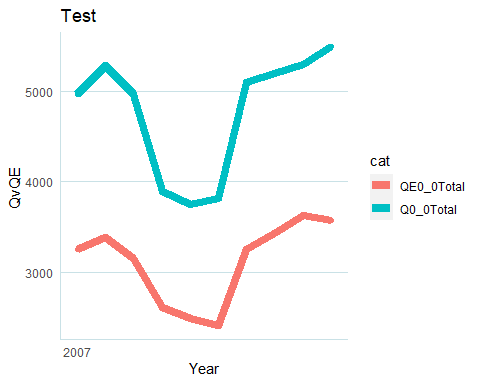
##### Graph 1: Price Index



##### Graph 2: Quantity Index Compare



##### Graph 3: Quantity Compare



### Do same analysis via a function!

Now that we know the method, we can simplify most of it into a function and do this whole analysis in 4 easy steps:

* A. Import and Edit data
* B. Enter base year
* C. Run the function
* D. Obtain the implicit quantity estimates

#### Function

We use the *PriceMethodOutput* function to calculate the Implicit Quanity Output at Fishery Level

#### A. Import and Edit data

#### B. Enter base year

#### C. Run the function

#### D. Obtain the implicit quantity estimates

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | QE0\_0Total | VE0\_0Total | VV0\_0Total | V0\_0Total | PC0\_0Total | PI0\_0Total | Q0\_0Total | QI0\_0Total | QEI0\_0Total | QC0\_0Total |
| 2007 | 3250 | 5600 | 4700 | 4600 | 0.0000000 | 1.128281 | 4963.304 | 1.2759136 | 1.2452107 | 0.0000000 |
| 2008 | 3380 | 6220 | 5000 | 5020 | 0.0429332 | 1.177776 | 5281.138 | 1.3576191 | 1.2950192 | 0.0732370 |
| 2009 | 3150 | 5710 | 4800 | 4810 | -0.0247311 | 1.149006 | 4969.513 | 1.2775099 | 1.2068966 | -0.0732619 |
| 2010 | 2610 | 3890 | 4700 | 3890 | -0.1388971 | 1.000000 | 3890.000 | 1.0000000 | 1.0000000 | -0.2699238 |
| 2011 | 2490 | 4180 | 5000 | 4180 | 0.1090416 | 1.115209 | 3748.177 | 0.9635417 | 0.9540230 | -0.0370504 |
| 2012 | 2412 | 4150 | 4950 | 4150 | -0.0247398 | 1.087957 | 3814.488 | 0.9805883 | 0.9241379 | 0.0175616 |
| 2013 | 3251 | 6280 | 5180 | 5280 | 0.1245141 | 1.232218 | 5096.500 | 1.3101544 | 1.2455939 | 0.3083746 |
| 2014 | 3431 | 6270 | 5370 | 5370 | -0.0206491 | 1.207035 | 5194.548 | 1.3353593 | 1.3145594 | 0.0106022 |
| 2015 | 3630 | 6700 | 5700 | 5700 | 0.0464723 | 1.264452 | 5298.737 | 1.3621431 | 1.3908046 | 0.0147757 |
| 2016 | 3575 | 6780 | 5680 | 5680 | -0.0244869 | 1.233866 | 5494.925 | 1.4125770 | 1.3697318 | 0.0474804 |

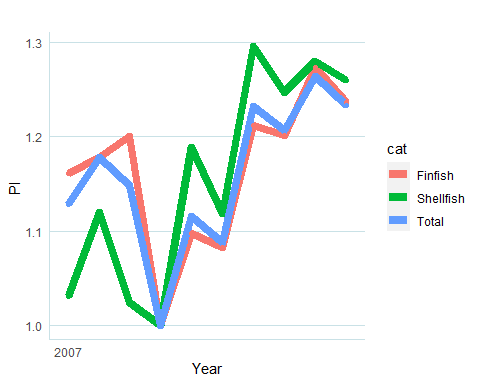
Did all of the analyses work as intended?

*, list(var = c(1, 1, 2, 2, 3, 3), slopecheck = c(1, 2, 1, 2, 1, 2), Freq = c(1, 1, 2, 0, 0, 2)), FYI: Rows of R\_{s,i,t} for 1\_0Finfish did not sum to 1, Warning: When back calculated, V\_{i,t} did not equal PI\_{i,t}*  Q\_{i,t}, Warning: When back calculated, Q\_{i,t} did not equal V\_{i,t}/PI\_{i,t}, list(var = c(1, 1, 2, 2, 3, 3), slopecheck = c(1, 2, 1, 2, 1, 2), Freq = c(1, 1, 2, 0, 1, 1)), Warning: When back calculated, V\_t did not equal PI\_t \* Q\_t, Warning: When back calculated, Q\_t did not equal V\_t/PI\_t, Warning: When back calculated, ln(Q\_t/Q\_{t-1}) = did not equal sum( ( R\_{i, t} - R\_{i, t-1} ) / 2 ) x ln( (Q\_{i,t}) / (Q\_{i,t-1} ) ), FYI: 0 of species V columns are completely empty, 2 of species Q columns are completely empty, and 0 of 5 species P columns are completely empty. \*

#### E. Graph

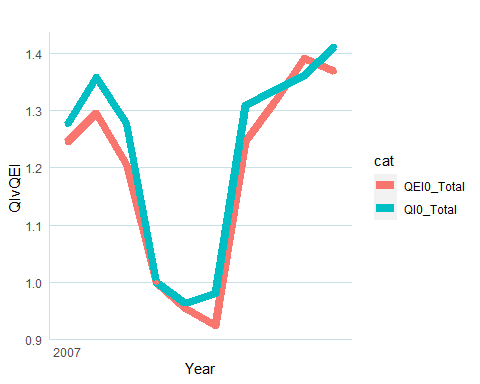
##### Graph 1: Price Index

For comparison, let’s recreate those graphs to make sure we are getting the same output:

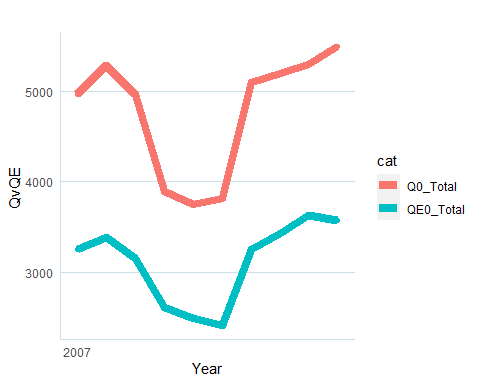


##### Graph 2: Quantity Index Compare

For comparison, let’s recreate those graphs to make sure we are getting the same output:



##### Graph 3: Quantity Compare



### Practice with real data (For National Data)

#### A. Import and Edit data

Load and subset Data

Edit/Restructure Data

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Q01\_0002ALEWIFE | Q01\_0003ALEWIFE. | Q01\_0004ALFONSINO. | Q01\_0006AMBERJACK.GREATER | Q01\_0007AMBERJACK.GREATER. |
| 1950 | NA | 735961 | NA | NA | NA |
| 1951 | NA | 758873 | NA | NA | NA |
| 1952 | NA | 722115 | NA | NA | NA |
| 1953 | NA | 750022 | NA | NA | NA |
| 1954 | NA | 650472 | NA | NA | NA |

#### B. Enter base year

#### C. Run the function

#### D. Obtain the implicit quantity estimates

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | VE\_Total | VV\_Total | V\_Total | PC\_Total | PI\_Total | Q\_Total | QI\_Total | QC\_Total |
| 1950 | 2596863100 | 2794604681 | 2593845000 | 0.0000000 | 9.9675236 | 260532424 | 0.4303577 | 0.0000000 |
| 1951 | 2711142600 | 2900126381 | 2707189300 | -0.1212391 | 8.8294531 | 307056684 | 0.5072084 | 0.1640520 |
| 1952 | 2904551900 | 3094353181 | 2900834400 | 0.0907005 | 9.6677301 | 300437835 | 0.4962751 | -0.0216501 |
| 1953 | 2945724500 | 3131700481 | 2943735500 | -0.0487177 | 9.2080293 | 319908247 | 0.5284371 | 0.0633851 |
| 1954 | 3131528000 | 3322992081 | 3129278700 | -0.0559950 | 8.7065957 | 359673067 | 0.5941222 | 0.1170278 |
| 1955 | 3208381300 | 3399979081 | 3206680100 | 0.0003453 | 8.7096024 | 368372877 | 0.6084929 | 0.0240459 |
| 1956 | 3413031000 | 3617472081 | 3411352200 | -0.0337031 | 8.4209534 | 405302205 | 0.6694942 | 0.0955397 |
| 1957 | 3065775700 | 3269932281 | 3064051500 | -0.0060364 | 8.3702742 | 366269448 | 0.6050183 | -0.1013776 |
| 1958 | 3003831240 | 3192160321 | 3000058540 | -0.0558103 | 7.9159237 | 379466926 | 0.6268185 | 0.0349491 |
| 1959 | 3552786100 | 3742561181 | 3550436800 | 0.0828581 | 8.5997620 | 413126095 | 0.6824180 | 0.0852974 |
| 1960 | 3431488969 | 3621283050 | 3429038869 | 0.1023245 | 9.5263249 | 360211203 | 0.5950111 | -0.1370718 |
| 1961 | 3545366800 | 3729969881 | 3537472400 | -0.0576920 | 8.9922854 | 394267602 | 0.6512668 | 0.0888076 |
| 1962 | 3740472904 | 3941834181 | 3734589304 | -0.0119179 | 8.8857525 | 420951733 | 0.6953447 | 0.0660684 |
| 1963 | 3436281347 | 3636121924 | 3428888647 | -0.0379607 | 8.5547651 | 401680387 | 0.6635115 | -0.0473576 |
| 1964 | 3082842508 | 3284893585 | 3077660908 | -0.0960807 | 7.7710688 | 396707661 | 0.6552974 | -0.0118619 |
| 1965 | 3312327541 | 3509281118 | 3306663741 | -0.0792721 | 7.1788242 | 461402516 | 0.7621629 | 0.1511255 |
| 1966 | 2883071824 | 3082537701 | 2877970424 | -0.0751205 | 6.6593048 | 432938857 | 0.7151455 | -0.0636703 |
| 1967 | 2744821593 | 2941375070 | 2738756093 | 0.1348783 | 7.6208918 | 360170656 | 0.5949441 | -0.1844403 |
| 1968 | 2859020482 | 3021220759 | 2838969682 | -0.0471637 | 7.2698068 | 393273242 | 0.6496243 | 0.0871465 |
| 1969 | 2995595955 | 3180276732 | 2976017755 | -0.1312197 | 6.3758026 | 469838251 | 0.7760974 | 0.1787788 |
| 1970 | 3289726569 | 3490531746 | 3284116169 | -0.1262266 | 5.6197287 | 585388862 | 0.9669684 | 0.2203496 |
| 1971 | 3657625234 | 3856485856 | 3654919813 | 0.0512353 | 5.9151609 | 618347548 | 1.0214109 | 0.0554043 |
| 1972 | 3420056087 | 3821528621 | 3417179485 | -0.0667839 | 5.5330258 | 618116777 | 1.0210297 | -0.0004601 |
| 1973 | 3323261015 | 3693691834 | 3320568292 | -0.5744829 | 3.1150711 | 1066833123 | 1.7622371 | 0.5458324 |
| 1974 | 3501940970 | 3869173264 | 3496834422 | 0.0893255 | 3.4061325 | 1028128240 | 1.6983028 | -0.0377426 |
| 1975 | 3268937259 | 3637613906 | 3265136356 | 0.0761651 | 3.6756964 | 889338203 | 1.4690440 | -0.1447468 |
| 1976 | 3550765040 | 3917851484 | 3545559881 | -0.1824311 | 3.0627449 | 1159340783 | 1.9150449 | 0.2653499 |
| 1977 | 3456110039 | 3824752205 | 3452906402 | -0.0995799 | 2.7724507 | 1246590272 | 2.0591670 | 0.0729531 |
| 1978 | 4293546946 | 4446059629 | 4289847393 | -0.0962753 | 2.5179784 | 1705156357 | 2.8166446 | 0.3137631 |
| 1979 | 4415372581 | 4567973390 | 4411863460 | -0.1027958 | 2.2720001 | 1943385720 | 3.2101613 | -Inf |
| 1980 | 4413228333 | 4565856246 | 4410212249 | -0.0330457 | 2.1981473 | 2007703641 | 3.3164042 | 0.0325190 |
| 1981 | 4208881700 | 4199769740 | 4195935855 | 0.0541069 | 2.3203588 | 1813892644 | 2.9962595 | -0.1012606 |
| 1982 | 4814459232 | 4786892219 | 4801533282 | -0.0267555 | 2.2590996 | 2131140736 | 3.5203024 | 0.1610999 |
| 1983 | 4992086437 | 4957376601 | 4978087659 | -0.0342712 | 2.1829893 | 2286812133 | 3.7774466 | 0.0705808 |
| 1984 | 4906589699 | 4890270598 | 4866852348 | -0.0199458 | 2.1398792 | 2292928318 | 3.7875495 | -0.0007077 |
| 1985 | 4678743893 | 4674097298 | 4649387420 | 0.0567295 | 2.2647829 | 2065868564 | 3.4124833 | -0.1025628 |
| 1986 | 4349847101 | 4330715720 | 4327978634 | -0.0929672 | 2.0637231 | 2107766829 | 3.4816925 | 0.0221690 |
| 1987 | 4788480254 | 4696479795 | 4763388776 | -0.0376986 | 1.9873719 | 2409453514 | 3.9800305 | 0.1337413 |
| 1988 | 4041389595 | 3899842610 | 4004196923 | -0.1810645 | 1.6582265 | 2437175838 | 4.0258233 | Inf |
| 1989 | 4023369359 | 3842546597 | 3974910973 | 0.0826819 | 1.8011595 | 2233766303 | 3.6898234 | -0.0886921 |
| 1990 | 4021450928 | 3878674910 | 3979163122 | -0.0168059 | 1.7711423 | 2270540820 | 3.7505690 | 0.0167143 |
| 1991 | 4032118379 | 3865661788 | 3995315782 | -0.0661084 | 1.6578412 | 2432149884 | 4.0175212 | 0.0679031 |
| 1992 | 3725995400 | 3582777275 | 3687951116 | -0.0473732 | 1.5811352 | 2356531753 | 3.8926122 | -0.0297158 |
| 1993 | 4105010022 | 3942994707 | 4062059127 | -0.0068959 | 1.5702694 | 2614207525 | 4.3182512 | 0.1036201 |
| 1994 | 4372163988 | 4217669145 | 4336688968 | -0.0599662 | 1.4788740 | 2956414140 | 4.8835216 | 0.1225581 |
| 1995 | 3888584141 | 3711645486 | 3849488988 | -0.0696921 | 1.3793176 | 2819208737 | 4.6568803 | -0.0474868 |
| 1996 | 3876679172 | 3715532966 | 3836741499 | 0.0296516 | 1.4208289 | 2728463068 | 4.5069830 | -0.0330153 |
| 1997 | 4094095068 | 3936173876 | 4057326451 | -0.0450450 | 1.3582477 | 3014247800 | 4.9790535 | 0.0999581 |
| 1998 | 3731867387 | 3582969082 | 3697377040 | -0.0115640 | 1.3426313 | 2779517616 | 4.5913170 | -0.0818072 |
| 1999 | 3990116584 | 3830900260 | 3962104908 | 0.0022699 | 1.3456824 | 2965125023 | 4.8979106 | 0.0645914 |
| 2000 | 3793869254 | 3576918541 | 3733876374 | -0.0898368 | 1.2300619 | 3084291359 | 5.0947543 | 0.0392347 |
| 2001 | 3765148707 | 3547935875 | 3688174498 | 0.0751249 | 1.3260298 | 2839414822 | 4.6902575 | -0.0829086 |
| 2002 | 3722911790 | 3474372249 | 3612665910 | 0.1341718 | 1.5164335 | 2455044518 | 4.0553395 | -0.1459950 |
| 2003 | 3704927643 | 3448902742 | 3606132813 | 0.0169653 | 1.5423797 | 2402085389 | 3.9678595 | -0.0203527 |
| 2004 | 1889813769 | 2016983232 | 1860167720 | -0.1544429 | 1.3216534 | 1429886040 | 2.3619422 | -0.5222809 |
| 2005 | 1741044062 | 1870255471 | 1705470703 | -0.1648763 | 1.1207602 | 1553449201 | 2.5660487 | 0.0833327 |
| 2006 | 1753094099 | 1890964774 | 1726077750 | -0.2086514 | 0.9096966 | 1927119579 | 3.1832922 | 0.2199644 |
| 2007 | 517159611 | 667070680 | 475161635 | 0.0902594 | 0.9956248 | 519432220 | 0.8580187 | -1.3289472 |
| 2008 | 521159877 | 670125837 | 486480769 | 0.0017992 | 0.9974178 | 522509115 | 0.8631012 | 0.0211240 |
| 2009 | 534736167 | 677145565 | 494251216 | 0.0368727 | 1.0348817 | 516712348 | 0.8535259 | -0.0265683 |
| 2010 | 605385696 | 714949228 | 551050632 | -0.0342871 | 1.0000000 | 605385696 | 1.0000000 | 0.1494147 |
| 2011 | 661229735 | 755626132 | 601024253 | -0.0125710 | 0.9875077 | 669594504 | 1.1060626 | 0.0939191 |
| 2012 | 655629951 | 772527014 | 602002953 | -0.0204319 | 0.9675357 | 677628658 | 1.1193338 | 0.0143946 |
| 2013 | 658197330 | 734787347 | 590869779 | -0.0386691 | 0.9308362 | 707103280 | 1.1680211 | 0.0225857 |
| 2014 | 587575682 | 720097666 | 519339299 | -0.0602186 | 0.8764369 | 670414131 | 1.1074165 | -0.0552013 |
| 2015 | 577149068 | 699010404 | 533666838 | 0.0029147 | 0.8789951 | 656600975 | 1.0845994 | 0.0112312 |
| 2016 | 556835463 | 711262899 | 519781163 | -0.0637685 | 0.8246928 | 675203523 | 1.1153278 | 0.0335641 |
| 2017 | 525822797 | 668362521 | 488629852 | -0.0055386 | 0.8201378 | 641139586 | 1.0590597 | -0.0504760 |
| 2018 | 490972000 | 671036323 | 439699415 | -0.0069958 | 0.8144203 | 602848422 | 0.9958088 | -0.0970116 |
| 2019 | 430592128 | 622763099 | 397062280 | -0.0627348 | 0.7648975 | 562940974 | 0.9298881 | -Inf |

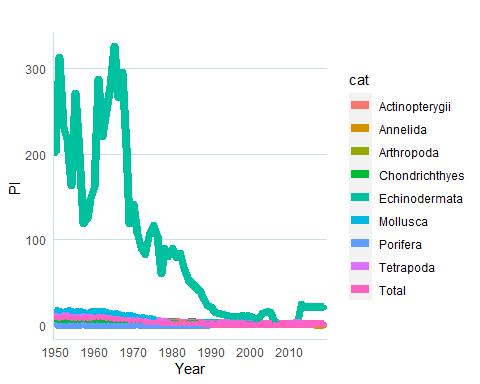
Did all of the analyses work as intended?

*, list(var = c(1, 1, 1, 2, 2, 2, 3, 3, 3), slopecheck = c(1, 2, 3, 1, 2, 3, 1, 2, 3), Freq = c(8, 192, 4, 76, 31, 97, 57, 84, 63)), FYI: Rows of R\_{s,i,t} for 01\_0000Actinopterygii did not sum to 1, Warning: When back calculated, V\_{i,t} did not equal PI\_{i,t}*  Q\_{i,t}, Warning: When back calculated, Q\_{i,t} did not equal V\_{i,t}/PI\_{i,t}, FYI: 02\_0000Agnatha is no longer being calculated because there were less than 2 columns of Q available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2, 3, 3, 3), slopecheck = c(1, 2, 3, 1, 2, 3, 1, 2, 3), Freq = c(0, 2, 0, 0, 0, 2, 1, 1, 0)), list(var = c(1, 1, 1, 2, 2, 2, 3, 3, 3), slopecheck = c(1, 2, 3, 1, 2, 3, 1, 2, 3), Freq = c(0, 25, 0, 10, 2, 13, 8, 4, 13)), list(var = c(1, 1, 1, 2, 2, 2, 3, 3, 3), slopecheck = c(1, 2, 3, 1, 2, 3, 1, 2, 3), Freq = c(3, 8, 3, 5, 6, 3, 5, 4, 5)), FYI: Rows of R\_{s,i,t} for 05\_0000Chondrichthyes did not sum to 1, FYI: 06\_0000Chromista is no longer being calculated because there were less than 2 columns of Q available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, FYI: 07\_0000Cnidaria is no longer being calculated because there were no more available columns of V after data was removed for not meeting the pctmiss, list(var = c(1, 1, 1, 2, 2, 2, 3, 3, 3), slopecheck = c(1, 2, 3, 1, 2, 3, 1, 2, 3), Freq = c(0, 2, 0, 1, 0, 1, 0, 0, 2)), list(var = c(1, 1, 1, 2, 2, 2, 3, 3, 3), slopecheck = c(1, 2, 3, 1, 2, 3, 1, 2, 3), Freq = c(1, 26, 0, 11, 1, 15, 9, 8, 10)), FYI: 10\_0000Other is no longer being calculated because there were no more available columns of V after data was removed for not meeting the pctmiss, FYI: 11\_0000Plantae is no longer being calculated because there were less than 2 columns of Q available (according to ‘MinimumNumberOfSpecies’) after data was removed for not meeting the pctmiss, list(var = c(1, 1, 2, 2, 3, 3), slopecheck = c(1, 2, 1, 2, 1, 2), Freq = c(0, 3, 1, 2, 0, 3)), list(var = c(1, 1, 2, 2, 3, 3), slopecheck = c(1, 2, 1, 2, 1, 2), Freq = c(3, 0, 1, 2, 3, 0)), Warning: When back calculated, V\_t did not equal PI\_t \* Q\_t, Warning: When back calculated, Q\_t did not equal V\_t/PI\_t, Warning: When back calculated, ln(Q\_t/Q\_{t-1}) = did not equal sum( ( R\_{i, t} - R\_{i, t-1} ) / 2 ) x ln( (Q\_{i,t}) / (Q\_{i,t-1} ) ), FYI: 2 of species V columns are completely empty, 2 of species Q columns are completely empty, and 0 of 6 species P columns are completely empty. \*

#### E. Graph

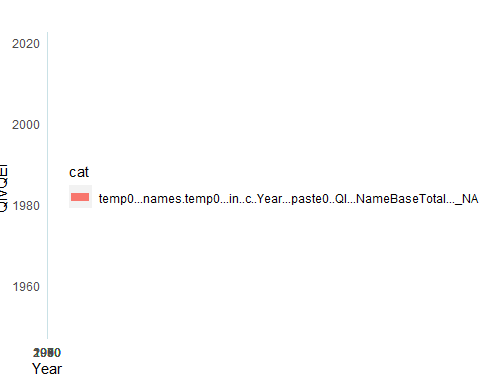
##### Graph 1: Price Index

For comparison, let’s recreate those graphs to make sure we are getting the same output:



##### Graph 2: Quantity Index Compare

For comparison, let’s recreate those graphs to make sure we are getting the same output:



##### Graph 3: Quantity Compare

