5122 Final Project Report:

Fish Stocking in the Great Lakes with an R Shiny Application

Group 1:

Joshua Allard, Siqi Li, Mariam Olupitan, Caroline Valade

Dr. Jinwen Qiu & TA Alvis Chung

August 03 2021

Introduction (What is the problem to solve?)

The R Shiny app that we created was based off the #tidytuesday challenge week 24[1] on Great Lakes commercial fishing. This particular challenge aimed to explore commercial fish data provided by the Great Lakes Fishery Commission for the years of 1867-2018. Since the Great Lakes Fishery Commission compiled all of this fishing data from various sources over a time span of about 150 years[2], we have enough information to analyze and determine different trends in fish species production and stocking over quite a long time period. This analysis is helpful in predicting future trends and can be used for things like determining where the largest populations of certain fish species can be found in the Great Lakes, knowing where to target more or less stocking of artificially grown fish to increase population for certain species, identifying changes in aquatic ecosystems that may be driven by invasive species, and more.

Our objective from this data was to build an application that would focus on the stocking of different species in the Great Lakes during the time period of 1950-2018. Specifically, we sought to solve the problem of identifying which species need to be stocked in which areas, based on data of stocked fish from previous years. With the target user being a member of the US Fish and Wildlife services, or an aquaculture farmer that grows the fish that are used to restock the Great Lakes, we also thought that providing data visualizations for information on the weight, age, and condition of fish previously stocked would be useful as well. This could help the farmers know which type of fish have been needed in the past and which may be needed to be stocked in the future, in addition to the number of fish needed for each species in each location.

In order to help solve this problem we created an R Shiny application. This app has a drop down list to choose which of the Great Lakes the user wishes to look at, and a slider to

choose which year of data to isolate. The left hand side also includes a table that serves as a legend for the fish species abbreviations used in the visualizations in the application. There are also four tabs of data visualizations to choose from to view. The first tab shows the number of fish stocked, the second displays the weight of the fish stocked, the third tab has information on the condition of the fish, and the last is about the age of the fish stocked. Each tab has a set of visualizations that are divided by state province and categorize each different species.

Data & Operation Abstraction Design (How did we prepare the data?)

Since our midterm project focused on the production side of the data that was given in the fishing.csv file on the #tidytuesday github repository for week 24, we decided to focus our final project and R Shiny application on using the stocked.csv provided. While this stocked file was provided through the Great Lakes Fishery Commission website, the data was actually collected and provided to them from the US Fish and Wildlife Services[3]. The stocked file had more data and variables to work with than the fishing production data, so we decided that using this file would provide the most helpful information to the user. However, with the additional variables and data, we had to do more data cleansing and also had to go to the Great Lakes Fishery Commission website to search for data variable definitions that were not provided in the #tidytuesday github repository for this challenge. Some of the data definition tables we found on the Great Lakes Fishery Commission website can be found in the appendix for easy reference[4].

Once we had downloaded the stocked.csv from the #tidytuesday github repository, we uploaded the file into an Rstudio project and loaded the data into a dataframe in our project. Our next step was to view the data and explore the distribution of the variables to gain a better understanding of what cleansing needed to be done and what variables would be most beneficial

to use in our R Shiny application. From this analysis, we decided to keep only 'YEAR', 'MONTH', 'DAY', 'LAKE', 'STATE_PROV', 'SPECIES', 'NO_STOCKED', 'AGEMONTH', 'WEIGHT', 'CONDITION', and 'STAGE' from the stocked.csv, and left out all other columns from the file when adding to our dataframe. We found that there were a lot of missing values for the columns month, day, and condition, so we replaced those values by inputting the median value for each of the columns and substituting any missing values with those medians. There were also some values for 'DAY' that were recorded as 0, which isn't logical as days of the month can be values of 1-31, so we replaced any instances where 'DAY' = 0 to be 'DAY'= 1, instead. Finally we changed the data variable type for condition from numeric to a dummy variable, and changed species and lake names to abbreviations[4].

Encoding & Interaction Design(Did you use the most effective data visualization design?)

Our data visualizations in our R Shiny application can be found on each of the tabs to the right hand side of the app. The four tabs each highlight a different attribute of the fish that were stocked, with one tab for number of fish stocked, weight of fish stocked, and age of fish stocked. Each of these tabs includes bar graph visualizations. These graphs are separated by state province, with species on the x-axis and cumulative number of fish on the y-axis for the number of fish stocked, the age of fish stocked, and the condition of fish stocked or cumulative weight of fish in thousands of lbs for the weight of fish stocked. The condition of fish stocked is displayed as a stacked bar chart instead of a traditional bar graph, with different colors in the stack indicating the different conditions of the fish, which are labeled on a scale of 0-7. The age of fish stocked is shown in terms of months and is noted by a scale of colors, where darker blues are younger fish and lighter blues indicate older fish. Different species, lake, and state abbreviations

can also be found in the appendix, as well as a list of what the different codes for condition represent[4].

All of the visualizations in our app are simple to interpret by the user and well designed. We include appropriate tab names and titles for each graph. We use color in a meaningful way, especially in the age of fish stocked visualization, where the color scale is used to display the age of fish in months. Similarly, color is used on the condition graph to create different sections of the stacked bars to indicate different conditions, and a legend is provided to let the user know which color corresponds to which condition. Bar graphs are a common graph that isn't hard to understand, so the app user shouldn't have any issues drawing conclusions. On the main left hand side of the application, we include a legend in the form of a table, to conveniently let the user know what each of the species abbreviations correlate to for their full names. This also remains no matter which tab on the right hand side of the app is selected, so the user doesn't have to navigate back and forth to look up the fish species names. Splitting up the graphs by state province gives the user more information on the more specific location within the Great Lake selected to know where fish stocked were released.

Algorithmic Design(How does your system work holistically?)

Our application is very easy to use and runs smoothly. It can be found by clicking the following link: https://marolu21.shinyapps.io/GreatLakes_Stocked/ . The user needs only to choose the Great Lake they wish to view with the drop down menu and select a year on the date range slider before clicking the run button to process selections and quickly update the data parameters. By clicking each of the different tabs on the right hand side of the application, the user can select which attribute of the stocked fish to look at. The new visualizations are shown

instantly without any delay. If there is no data for a certain year and lake combination, an error message is shown letting the user know that. The application is easily scalable to incorporate more less data in different contexts.

From an application development perspective, the code used to make the R Shiny application is efficient and well organized. The first section of code loads all packages needed, as well as the data from the stocked.csv file. We then perform our data cleansing. Our next chunk is written to set them and the main parts of the application, including the lake drop down menu input and the year input slider. The fish species legend is also added during this section, as well as the creation of each of the tabs. After that, we build each of the visualizations for each of the tabs, specifically the layout, data, labels, scale, etc to be used on the different bar graphs.

User Evaluation(How would you test your system?)

One method of testing our application would be to let a certain number of aquaculture farmers or US Fish and Wildlife officers in the Great Lakes region to use it and see if the information they gather from it did lead to them targeting the correct species and type of fish to grow to be restocked in the specified regions. After giving them instructions on how to use our app, showing them the different tabs and how to choose a lake and year to look at, we could let a few dozen wildlife officer and farmers explore the information for the farmers to decide how many of fish of certain species to breed and what condition, age, and weight they need to be before they provide them to be released into the Great Lakes and for the wildlife life officers to decide where these fish need to be released out at to maximize population and production of species in the Great Lakes.

Our next step in testing our app could be to collect data on the production of the fish species that were grown and released in the regions they targeted over a period of time, say six months. Based on the results of their findings from using our app and after growing and releasing the number and type of fish into the target areas, we can collect more production data to see how many and what size and condition fish are actually being caught in the lakes. We can then compare this data to see if certain species are still underpopulated and should have been stocked more in certain areas, or if there are enough fish of the species, but they are in poor condition or underweight. Investigation should also be done to determine if these fish were initially not stocked correctly, or if there are other causes for any of the issues that came after the fish were released into the lakes.

Future Work(What to do next?)

_____There are a number of things we could do to add or improve our R Shiny application to help solve our targeted problem of identifying stocking needs of the Great Lakes. One possible thing we could do is incorporate the production data from the fishing.csv file from the #tidytuesday github repository for week 24. This could be added to the stocking data and helpful to commercial fishermen, since they could use it to see where the largest production has been over the years for certain species and cross reference it with the stocked data to see where the most desirable and populous areas for fishing a certain species. Stocked fish with higher weight and better conditions would be more favorable than ones that weighed less or were in worse condition.

Another idea for future work would be to add more visualizations/ tabs to our R Shiny application. We could add in new graphs and charts for other attributes on stocked fish that may

also be helpful to aquaculture farmers or commercial fishermen. Such might include stage, since baby or really young fish might not be allowed to be harvested by fishermen due US Fish and Wildlife restrictions, or they might just favor these fish over the smaller young fish, since they would probably be more valuable in the fish market. We could also explore the length attribute for similar reasons, as there are often regulations on the length of fish that can be kept or must be thrown back. Another attribute that could be helpful would be the time of year released, which could be determined from the month variable provided. This could let wildlife officers know when the best success in releasing certain species in specific locations may occur throughout the year.

The latitude and longitude data was missing for most of the records in the stocked dataset provided. If there was more accurate data populated for these fields, more work could be done with that as well. For example, geospatial maps could be created from this data. These maps could be used to help visualize where species are being restocked, more specifically in the Great Lakes, rather than just at which lake or state province.

A final idea for future work on improving or adding to our R shiny application would be to narrow the data time range down to just the most recent year or five years. This could let us be more detailed on the data visualizations for a smaller time period, rather than being less detailed or a larger time frame. The last year to five years would probably still be an appropriate time period for us to use since this data would be recent enough to still draw appropriate conclusions on trends in restocking.

Appendix

[2] Production refers to the catch, or number of fish, brought in by commercial fishermen and is measured in rounded thousand of pounds throughout this data and project.

[4] **Data Tables:**

Table: Lake

lake	description
ER	Lake Erie
HU	Lake Huron
MI	Lake Michigan
ON	Lake Ontario
SC	Lake St. Clair
SU	Lake Superior

Table: State Province

state_prov	description
IL	State of Illinois, United States
IN	State of Indiana, United States
MI	State of Michigan, United States
MN	State of Minnesota, United States
NY	State of New York, United States
OH	State of Ohio, United States
ON	Province of Ontario, Canada
PA	State of Pennsylvania, United States
WI	State of Wisconsin, United States

Table: Condition

condition	description
0	unknown condition at stocking
1	<1% mortality observed, "excellent"
2	1-2% mortality observed, "good"
3	3-5% mortality observed, "fair"
4	5-25% mortality observed, "bad," explain in Notes field
5	>25% mortality observed, "very bad," explain in Notes field
6	mortality is accounted in reported total stocked
7	distressed or sick, unknown mortality

Table: Species

species	common_name	genus_species	family	status
ATS	Atlantic Salmon	Salmo salar	SALMONIDAE	I
BKT	Brook Trout	Salvelinus fontinalis	SALMONIDAE	I
BLG	Bluegill	Lepomis macrochirus	CENTRARCHIDAE	I
BNT	Brown Trout	Salmo trutta	SALMONIDAE	X
CHS	Chinook Salmon	Oncorhynchus tshawytsha	SALMONIDAE	X
COS	Coho Salmon	Oncorhynchus kisutch	SALMONIDAE	X
HSF	Hybrid Sunfish		CENTRARCHIDAE	
LAS	Lake Sturgeon	Acipenser fulvescens	ACIPENSERIDAE	I
LAT	Lake Trout	Salvelinus namaycush	SALMONIDAE	I
LHR	Lake Herring	Coregonus artedii	SALMONIDAE	I
MUE	Muskellunge	Esox masquinongy	ESOCIDAE	I
NOP	Northern Pike	Esox lucius	ESOCIDAE	I
RBT	Rainbow Trout	Oncorhynchus mykiss	SALMONIDAE	X
SMB	Smallmouth Bass	Micropterus dolomieu	CENTRARCHIDAE	I
SPE	Splake	Salvelinus namaycush X fontinalis	SALMONIDAE	X
STN	Sturgeon (general)		ACIPENSERIDAE	I
TIM	Tiger Muskellunge	Esox masquinongy X lucius	ESOCIDAE	X
TRT	Tiger trout	Salvelinus fontinalis X Salmo trutta	SALMONIDAE	X
WAE	Walleye	Stizostedion vitreum	PERCIDAE	I
XXX	Unknown fish			
YEP	Yellow Perch	Perca flavescens	PERCIDAE	I

Packages Used:

- library(tidyverse)
- library(purrr)
- library(tidyr)
- library(ggplot2)
- library(dplyr)
- library(Hmisc)
- library(shinythemes)
- library(RColorBrewer)
- library(ggthemes)
- library(ggtext)

References

[1]https://github.com/rfordatascience/tidytuesday/blob/master/data/2021/2021-06-08/readme.md

[2]http://www.glfc.org/commercial/COMMERCIAL%20FISH%20PRODUCTION_Notes%20on%20Statistics.pdf

[3]http://www.glfc.org/great-lakes-databases.php

[4]http://www.glfc.org/fishstocking/dbstruct.htm