#### Chapter 1

#### Establishing a progressive data management workflow for biological data to inform adaptive management decisions

**Abstract**

Advances in technology such as expanded remote sensing and animal tracking platforms have triggered rapid expansion of data available for ecologists and natural resource scientists to understand how plants, animals, and their environments interact and respond to anthropogenic change. This increase in new data creates both opportunities for learning and challenges for managing these data and creating data workflows that lead to reproducible results. We customized a modern data workflow for continuous and discrete long-term ecological data to assist in adaptive decision making related to a large habitat restoration project. This workflow focuses on the data management concerns commonly encountered with large restoration efforts such as simultaneously managing data from autonomous sensors and field observations to inform ongoing restoration efforts. To promote reproducibility in our workflows and reduce data collection errors, we incorporated specific standards into our program including (1) standardizing field datasheets linked to an electronic data entry platform; (2) performing quality assurance and control (QA/QC); (3) creating scripts to analyze data and inform decision making; and (4) use a version control workflow to track changes to data, scripts and documents. The workflow uses open source software and tools to create a modern-day data management structure and is an example which could be implemented in many research efforts.

#### Introduction and Background

Traditional field biology programs, many of which are designed to monitor animal populations and their environments, have experienced a substantial evolution in data collection, management, and storage technology in recent years. Changes include new sensor technology, data collection methods, and data observing platforms that are being used in large-scale monitoring programs including SECOORA (Southeast Coastal Ocean Observing Regional Association) and NEON (National Ecological Observing Network). As an example, advancements in sensor technology have allowed for significant changes in water quality monitoring such as transitioning from discrete single location and single point in time sample collections to real-time continuous observations at multiple locations ((Martinelli et al., 2016)). While the scale and technological capacity of many monitoring programs has increased these monitoring programs are still most often conceived, planned, and used by personnel trained as biologists and not data scientists. The lack of training in basic data management, curation, and workflow of data generated from these types of data collection platforms was demonstrated in a recent NSF (National Science Foundation) survey (Lowndes et al., 2017)which highlighted that of the 704 scientists who participated in the survey, “data skills” (e.g. multi-step workflows, ability to store, share and publish data) was identified as the largest unmet need (Barone et al., 2017).

The US Gulf of Mexico region is undergoing a large restoration effort to reverse observed declines in key ecosystem components including seagrass, fish communities, and oyster reefs using funding from the consolidated Deepwater Horizon settlements (see https://www.nfwf.org/gulf/Pages/home.aspx as an example). These restoration projects vary in spatial scale, but, like other restoration efforts, these projects have data collection and evaluation efforts that occur frequently throughout the project. Several of the restoration programs in this funding program require basic adaptive management concepts be used to guide restoration actions (Zedler, 2017). Under this framework, decisions related to restoration actions are made iteratively based on stating, testing, and updating hypotheses based on observed outcomes (Figure 1). In a restoration context, this information can be used to inform the restoration actions such type of substrate to use in an oyster restoration project or monitoring program design as the project is ongoing, increasing efficiency by maximizing return on investment from restoration dollars. Doing so requires a data management plan designed to improve restoration actions by maximizing learning from previous and ongoing restoration efforts (Tompkins & Adger, 2004).

One example restoration effort funded by NFWF (National Fish and Wildlife Foundation) as part of the consolidated Deepwater Horizon settlements is the Lone Cabbage Reef (LCR) oyster reef project in the northeastern Gulf of Mexico. The primary goal of this project is to restore specific oyster reefs to historic levels so that they may be resilient to changing sea level and river discharge. This project generates data from multiple sources including continuous autonomous water quality data via sensors and observations of oyster populations by field biologists. These data are generated at different time steps with sensor data obtained at hourly time intervals from multiple spatial locations and biological data collected at discrete time intervals from multiple spatial locations. For both cases, there is a need to capture and process data to meet standards and then complete routine analyses of these data to ensure they are useful for informing project objectives and questions. This is critical because this project by design uses adaptive management principles to inform the restoration through an interactive process of collecting data, analyzing these data, and informing restoration actions from these analyses. For this project to efficiently operate in an adaptive management framework, we developed a system which captures data as it is collected, guides the data to analyses, documents data and analyses decisions via version control, and archives and makes these data available for long-term reproducible exploration. Here we describe this data management system and the structure and decisions made in implementing the system to improve data quality and reduce the likelihood of data collection and errors in analyses.

**#### Box 1. Terminology**

**“Living data”**

Living data” are defined as data which are continuously collected and updated (Yenni et al., 2018). These types of data are critical to adaptive learning to inform restoration and management actions. Examples of learning as part of a restoration project includes small changes like shifting the location of an autonomous sensor, to larger changes such as revamping of sampling programs because of low statistical power. Living data can inform these decisions, but living data are challenging to work with from a data management perspective because the data (by design) change as new data are collected. In a restoration context as these data are collected, they must be processed, and analyses of these data to be completed to help draw inferences on how the system of interest is responding to the restoration action. This idea of iteratively integrating new data, analyses, and comparing these outcomes with previously stated objectives is not new and is a central aspect of the adaptive management process for natural resources as originally described (Holling 1978; Walters 1986).

**Adaptive management**

By design, an adaptive management system requires rapid feedback loops between data collection, analyses, and interpretation to drive the process of updating knowledge, examining management and restoration options, making decisions and implementing actions (Nie & Schultz, 2012). This process is repeated (Figure 1) to improve management actions such as identifying the best restoration approach. Data used must meet quality assurance/quality control (QA/QC) protocols to identify and correct inconsistencies and errors in field or sensor observations before these data are used in an analysis. Errors in these data, or delays in producing the data in a framework useable for analysis, can quickly lead to a breakdown in the adaptive learning process either in terms of slowing the analyses and limiting their utility for timely decision making, or worse, erroneously informing the decision-making process because of errors in data management or analyses.

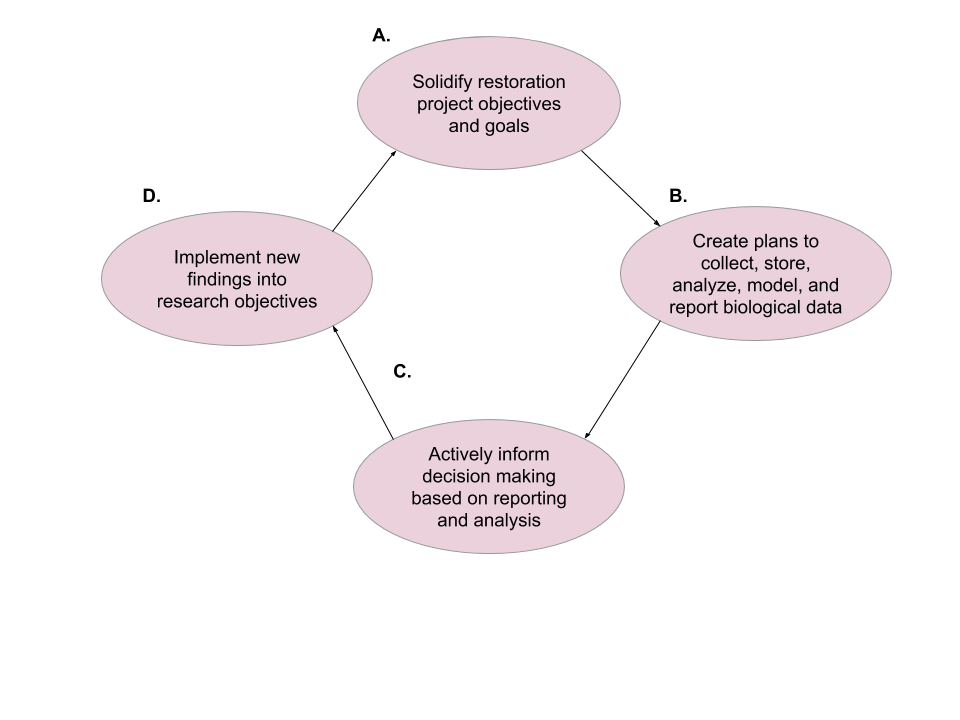


Figure 1- The adaptive management process for ecological restoration projects.

**Version control**

Version control software is a type of software that helps to manage documents, scripts of computer code, and other developmental information documents that are shared and iteratively updated over time. The key purpose of using version control software is to document and confirm that changes in content are intended and planned. The advantages of using version control (1) a version control system saves all versions of a file, (2) version control records who made what changes to specific files and makes the user write detailed notes about what they changed (3) allows these changes to be undone if needed, (4) can facilitate reproducibility and transparency of project code and decision making (Ram, 2013). Version control can be incorporated into a data workflow using software such as Github (https://github.com).

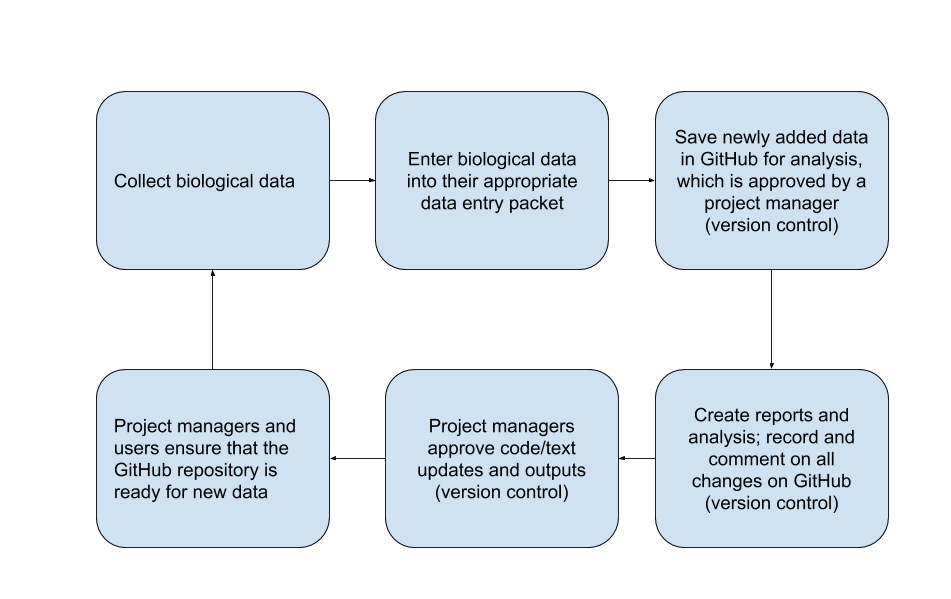


Figure 2 – Generalized version control workflow for the LCR project, detailed workflow information can be found here (zenodo link for Github workflow).

**#### End of Box 1**

The LCR project collects data on multiple parameters to measure ecosystem response to oyster reef restoration. One response metric are observations of water conductivity and temperature collected hourly from autonomous sensors. These types of data are measured and recorded by the sensor and are output in a standard format than can be interpreted for analyses directly by a computer. A second metric are counts of oysters at locations where restoration has been done (restored sites) and sites where restoration has not been done (wild oyster bars). Oyster counts are made by people conducting the fieldwork during winter low tide events. These data are collected by people and then must be entered into a computer as a standard data form before these data can be analyzed. We created a data management workflow to efficiently process and analyze data from both of these data streams. These data streams are then consolidated, and used to actively inform decision-making for the project such as the amount of sampling trips needed to optimize oyster density estimates. We use software and tools that are open source, widely available and familiar to many field biologists such as program R (Lefcheck, 2016) and Microsoft Excel. This paper documents this workflow and provides an example for use in other restoration and conservation projects.

**#### Box 2. The LCR project data types**

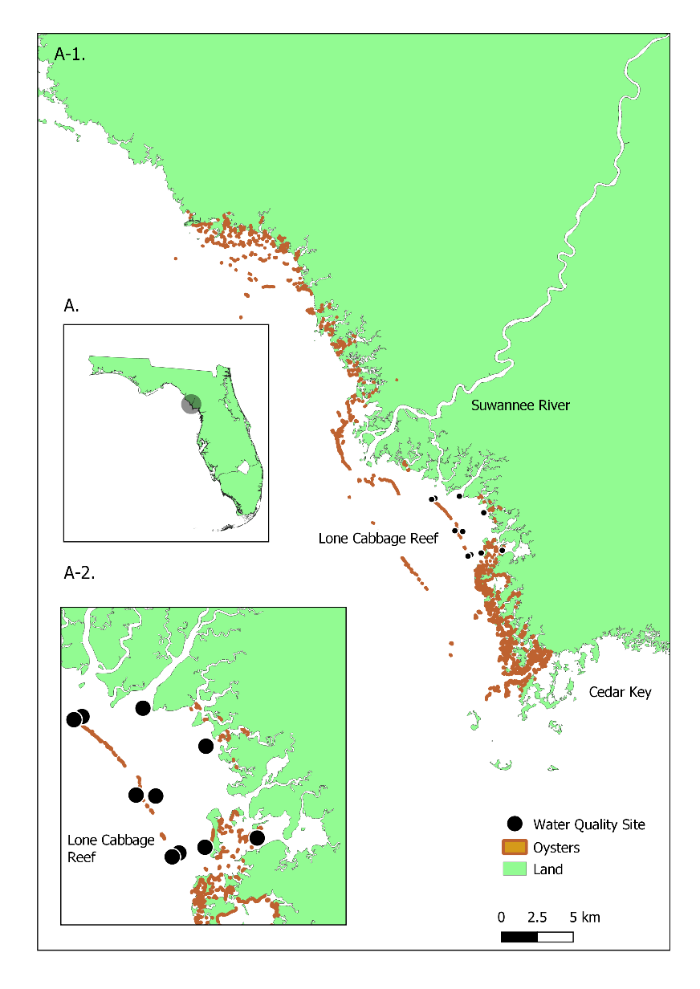
The LCR project generates data from autonomous sensors, and human observed counts/measurements. Several types of data are collected at various frequencies (seasonally, bi-monthly) and each data type requires a specific plan to monitor data workflow from collection to analyses.

Figure 3- Water quality location map. A) Map of Florida identifying general Lone Cabbage Reef area; A-1) Map scale 1:23,100 of Florida coastline between the mouth of the Suwannee River and Cedar Key, Fl; A-2) Map scale 1:9,000 of Lone Cabbage Reef with water quality sites identifies (black circles). The oyster shapefile used in this map is a from a University of Florida sampling effort in 2001.

*LCR project naming conventions*

A critical component of our data management plan was to create a naming convention standard for every sampling location. Every oyster transect sampling location is identified by its location, which we abbreviate (e.g, Lone Cabbage= LC, Horseshoe Beach = HB), and then a number added to the end to identify the location as separate from other sampling locations. Each sampling location is then recorded in a master sampling list to ensure that we can go back to the same location and to also reference the same location in the exact way throughout multiple sampling events. Even though the oyster transect sampling locations are randomized when selected, it is an integral part of our sampling workflow to establish the naming convention of the sampling location prior to the sampling event.

Another aspect of our naming convention standards, which directly relates to data management, are the way we name project files. We use a standard of referencing the date the file was created and what the file is so that every project member will be able to decipher the subject matter of the file without having to view it’s content. For example, our water quality sensor download files are named in a specific format YYYYMMDD\_sitelocation\_sensortype.file (e.g., 20200902\_wq7\_diver.MON). The file name is important to identify the date and site in a clear fashion especially when we are investigating sensor readings which may be corrupt or uncalibrated. This file naming format has saved time for project team members because all files are uniform and consistent in their naming, making it easier for each team member to follow the naming convention guidelines.

*Water quality data from autonomous sensors*

We collect hourly water quality observations from 11 different sites around Lone Cabbage reef (Figure 3). These observations are downloaded from autonomous sensors approximately every two weeks. maintenance of these sensors and their protective housing are completed to ensure continuous stream of data by reducing data errors due to biofouling or equipment loss. These “living data” have the highest frequency of occurrence (most number of observations) and require strict data management protocols (Box 3) both in terms of launching and maintaining the sensors and in importing the data files to maintain database integrity.

*Oyster counts and measurements from field sampling by people*

The lowest lunar tides of the year in the area around Lone Cabbage Reef occur during winter, and these low-tide events de-water oyster reefs which allow teams of people to count and measure oysters to document status and trends of oyster populations (Moore et al. 2020). These count data are recorded in the field on datasheets and then entered into a computer by people through a dual data entry system where each data record is entered independently into the computer. The dual entry system also includes data validation drop downs that ensure that the user is only entering appropriate data. For example, if a user is entering the height of an oyster to be 1000 cm the data validation will reject this entry because there is a range of acceptable oyster heights that it will allow. Any mismatch or errors in the dual data entry worksheets will then be reconciled by the research coordinator. This double data entry system was created to reduce the chance of data entry errors and human introduced errors (Box 4 3A).

*Water quality measured by field-crews*

During water quality service trips, we also collect water quality measurements using a hand-held YSI (Yellow Springs Instrument) device to provide a supplemental check on our autonomous sensor observations. These measurements are recorded once during the water quality service trip for each site location. These observations are the least intensive data type as their frequency is low, and they are manually entered in the MySQL database (Box 3).

While conceptually each of these data types appear to be similar, because of differences in the frequency the data are collected and the collection method (with a machine or by hand), each data stream must be managed differently. Addressing the variety of concerns which have been discovered through regularly updating these data types may also address many data management challenges which researchers may confront.

**#### End of Box 2**

### Establishing a modern data workflow

Data collected in the field are stored in a relational database. Database development efforts started prior to data collection through development of database “blueprints” via white board exercises to clarify (1) database goals, (2) data types and data sources, and (3) relationships among data types within the database. Blueprinting development efforts were led by University of Florida Academic Research Consulting & Services (ARCS, <http://arcs.uflib.ufl.edu/>). A key database need identified in blueprinting was the ability in the database to track observations at a particular site in space, and not focus on tracking observations recorded by an individual sensor, which could change locations over time. The workflow we have developed for water quality management (Box 3) addresses goals and special concerns identified through whiteboarding but requires open source computational tools, some level of knowledge of computational tools (e.g., MySQL and R) and version control (e.g., GitHub), which are tools essential for basic data management. While this example is specific to the LCR project, we feel that the workflow developed could\ could be implemented in similar restoration efforts.

**#### Box 3. Water quality workflow**

Extensive details on the MySQL import process are provided in the project management library (zenodo link for MySQL). An overview is provided here:

1. Datasheets are standardized and include pre-populated fields including the location and date to minimize error.

2. Water quality hourly sensor observations are downloaded in the field to a field laptop while simultaneously notes are made on paper datasheets related to field weather conditions and equipment status. Water quality observations using the YSI device are also taken and recorded on the same data sheet.

3.A. Water quality sensor files are then uploaded into a secure University of Florida internal server and a trigger starts the Python import process into the MySQL relational database, which permanently stores raw files as an archive. The YSI measurements are manually entered into our MySQL relational database in its appropriate table.

3.B. QA/QC R scripts pull and process the water quality observations to check for flatlined or out of bound measurements (i.e., outside of expected range).

3.C. Processed data, edited scripts, and documents are then stored and updated unto GitHub. Standardized GitHub workflows are used during collaborative projects to ensure proper version control utility (zenodo link for GitHub workflow).

Figure 4- Data workflow for water quality observations.

**### End of Box 3**

**Adding water quality measurements to our permanent MySQL relational database and version control**

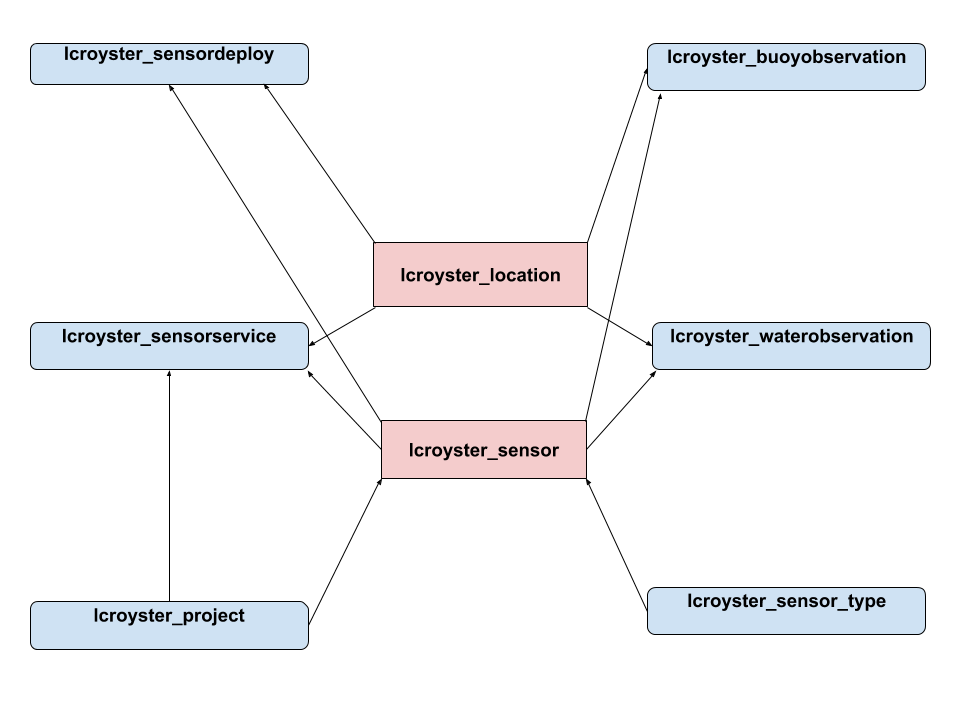
We use dedicated username and password controls to maintain access to the MySQL database to track access and database changes. We store water quality data in specific tables where the sensor serial number and location must be pre-defined prior to importing the sensor observations (Box 3). These pre-definitions allow us to track which sensors are in which location at a specific time. The MySQL database relates to multiple data tables through foreign keys (e.g, specified MySQL columns) and in our project the tables are related through sensor serial number and site location (Figure 5). We use R scripts to pull these unedited observations and process additional QA/QC procedures. These processed observations and their accompanying scripts are then updated using version control in the project GitHub master data repository (https://github.com/LCRoysterproject). This repository includes an up to date master branch which is protected from any unintended or incorrect updates using GitHub repository restrictions. To submit any changes to the master branch (referred to as a pull request), it is mandatory to have the changes/edits reviewed by another member of our team to ensure data integrity (zenodo link for GitHub workflow). Every pull request requires a thorough message describing each change, in the event an update to the master branch has to be investigated. Version control allows for team members to view a previous iteration of the master data branch and go back to that iteration if needed (Perez-Riverol et al., 2016). This workflow protects the master branch from possibly merging accidental or incorrect changes, giving a layer of needed security to the data.

Figure 5 - Diagram of how the tables in our MySQL relational database are connected by a sensor’s location (lcroyster\_location) and serial number (lcroyster\_sensor). More information on the description of the tables and the data they archive can be found here (zenodo link for MySQL).

**Automated data checks through Python and R scripts**

Water quality observations are imported into our MySQL relational database through custom Python scripting. The Python import process provides QA/QC procedures such as flagging duplicate water quality observations. If observations are flagged through the Python import process a review takes place to find out why the observations are labeled as a duplicate. All unique observations are imported into our MySQL relational database, where they will be additionally reviewed via R programming scripts. The R scripts check for out of range measurements and additional scripts remove flatlined water quality measurements (usually due to ocean fouling). Additionally, water quality visualizations help check for data integrity. The R scripts are not automated, but they do provide a way to provide quick and efficient checks on the data.

**#### Box 4. Oyster observation workflow**

Detailed information on the dual data entry system using a structured data packet is available in the data entry documentation for the Lone Cabbage project (zenodo link for data packet). Several of these entry processes are similar to those in the water quality workflow (Box 3) and will only be briefly reviewed here where:

1. Datasheets are standardized prior to going in the field include pre-populated fields including the location and date to minimize error.

2. In the field, counts of oysters are recorded by team members on datasheets by hand.

3.A. In the lab data are entered using a dual entry system and data validation tools are used to ensure that the data entered are within range and standardized (e.g., site location, capitalization, appropriate oyster height range, etc.).

3.B. Standard R scripts are used to estimate oyster densities (e.g., population abundances) and power analyses are done using these data as they are entered to inform field sampling efforts within the field season.

3.C. Processed data, scripts, and documents are then stored unto GitHub. Standardized GitHub workflows are used during collaborative projects to ensure proper version control utility (zenodo link for GitHub workflow).

Figure 6- Data workflow for oyster measurements.

**#### End of Box 4**

**Datasheets, data entry and validation of oyster data from the field**

We developed standardized datasheets for recording information by hand from field observations. These datasheets were designed to (1) clearly detail format of information to be recorded, (2) minimize errors, (3) allow for easy transcription from field observation, to paper, to entry into the computer. As the oyster counts and measurements from the field transects are entered into the data entry form built in Microsoft Excel. Within Excel, data validation checks are automatically applied. These data validation checks provide initial assessments that every new manually entered observation is restricted and limited to what is applicable for that column. As an example, restrictions include oyster height measurement ranges, site location names, and acceptable date ranges for surveys. For some types of data entry, such as site names, we use drop down menus such that the person entering the data must choose the name of the site from a predefined list, instead of typing the name. This reduces the changes of entering a name incorrectly, but would still allow for an incorrect name to be selected. For other types of data entered, such as oyster heights, we pre-define a minimum and maximum range that is expected of any given oyster height such that if a value entered for a height falls outside the accepted range, then, a warning is issued and must be addressed. We also require that two people separately enter oyster observations, in two separate Microsoft Excel tabs (Box 4 3.A). An additional Microsoft Excel tab compares the two entry sheets to determine whether the separate entered versions are identical. If the dual-entry versions are not the same a “check” notification will appear on the Excel cells (e.g., the cell column and row number) that do not match. The flagged cells will then be reconciled by a third team member, who will investigate the discrepancy using the original data sheets. The process of a dual-entry workflow is known to significantly reduce data entry errors (Barchard & Pace, 2011).

**Adding oyster observations to a central storage and version control**

Reconciled oyster observations are ultimately stored in our master data repository on GitHub and team members are required to follow the same workflow as defined for water quality data to upload this info to GitHub (Box 4 3.C). The workflow ensures that every new type of oyster data updated are reviewed prior to merging with the protected `master` branch. It is also important to note that oyster measurements are not stored in MySQL since our MySQL relational database was created specifically for water quality observations. Oyster data are also stored in a University of Florida protected server with remote storage as a backup to the GitHub servers.

**Regularly updated data and adaptive management**

We have developed scripts of computer code that use these data to create summaries and routine analyses to provide information to (1) adaptively update sampling efforts and (2) meet reporting requirements of funding agency. Because of data workflow is standardized, each time new data are collected, then these standard computer scripts can be run to inform ongoing research efforts. As an example, we routinely use a type of power analyses to guide field sampling efforts during winter oyster sampling. Prior to the field sampling season, data from previous years are used to develop preliminary sampling guides in terms of location and number of transect samples to take. As field collections begin and new data are collected and entered into the computer, we re-run the power analyses to then update the sampling effort for that field season based on observed oyster density and variability from within season sampling. This allows us to allocate effort to locations where they provide the most information to meet project objectives.

#### Discussion

Establishing a data management workflow is receiving more attention in ecological efforts. Thus, creating a data management workflow from the beginning of the research initiative makes data management an easier endeavor to maintain than trying to reconcile and document the aspects of the study after a manuscript has been prepared (Archmiller et al., 2020) .Data and scripts without proper initial data management workflows can lead to an increased effort and time to properly archive and clean, and though it is possible for post-reconciliation in theory it is rarely followed in practice (Nelson & Grubesic, 2018) . Our data management addresses many of the challenges with “living data” such as reducing human introduced error, permanent data storage, and version control for text and scripts. Our workflow uses a mixture of familiar software (e.g., Microsoft Excel) and versatile software (e.g., R programming, GitHub, MySQL), which encompasses an array of skills needed from a biologist to employ a modern workflow.

Our data management workflow may not work for every ecological project, however many elements discussed in this paper should still be applicable. The concept of creating a data management workflow prior to an conservation venture is one of our main talking points. Much of project planning time is allocated to the ecological question asked and how to set up the sampling design. However, the planning should continue and expand to how the collected data will be managed and to train team members on the workflow. Knowing the frequency of data collected, where it will be stored, how it will be entered, is necessary to ensure data integrity. The data collected and analyzed will ultimately guide ecological efforts and inform funding agencies of the progress. The principles of securing and validating data should also be considered of high importance for monitoring efforts as well.

There are many advantages to using open-sourced tools (e.g., GitHub, R programming, and MySQL) in a data management workflow. Firstly, these software are free and there is continuous support for these applications online. Secondly, this workflow can be achieved by few biologists using online training programs such as The Carpentries (<https://carpentries.org/>). Many universities also offer R programming courses which teach the basics of statistical analysis with R (e.g., WIS 4601, Quantitative Ecology, <https://wec.ifas.ufl.edu/undergraduate-students/undergraduate-course-listing/>) and similar data management techniques described in this paper (e.g., WIS 6934, <https://datacarpentry.org/semester-biology/> ). Using GitHub offers much desired flexibility in code development through “pull requests” (Rahman & Roy, 2014) and version control (Blischak et al., 2016). GitHub consistently updates their software features making it a reliable resource for many projects.

Some initial difficulties to our workflow may arise in teaching team members how to use the workflow and to ensure that they are following workflow processes. It is important to communicate effectively with team members to guarantee they are collecting and maintaining data within the workflow procedures. Another disadvantage to our current workflow is that it can only handle only certain types and a limited amount of storage space. Our MySQL database can only store numerical or character information, it cannot store images or completed maps (<https://www.mysqltutorial.org/mysql-data-types.aspx> ). The MySQL database can also be difficult to make fundamental changes to, which we do not want to do at this time, and would require the expertise of ARCS to make any real changes to the functionality of the relational database. GitHub has a repository limit of 1 GB and up to 100 MB for an individual file (<https://help.github.com/en/github/managing-large-files/what-is-my-disk-quota#file-and-repository-size-limitations>), which can make it difficult to store large files without compressing them. However, despite these limitations with MySQL and GitHub, their functionality greatly outweighs their restrictions.

Adaptive management is described as a process which continually improves policies and practices based on data outcomes (Pahl-Wostl, 2007). Due to the recent advancements of technology, one would assume that adaptive management should be widely employed among ecological programs, however adaptive management is infrequently implemented (Weimer et al., 2007.). We have described some of our challenges and our approaches to address these concerns through our data management workflow, hoping that it can provide guidance to future research efforts. Our data management workflow currently does not address some common concerns such as (1) citation and authorship credit, (2) managing for maps (e.g., geodatabases), and (3) data licensing (for collaborative data efforts). Collecting and managing “living data” is becoming the norm in many research programs. Making the effort to train teams and cultivate this new data type will ensure that scientists will be able to effectively manage these data, and these data will ultimately provide a feed-back loop for adaptive management.

**Chapter 2**

**Tailoring GitHub for Ecology**

**Abstract**

Ecology is transitioning from an anecdotal science into a data driven science. Biologists are finding that they are needing skills to manage, analyze, and store data effectively. Data, such as code and text, require thoughtful management practices to keep these files organized and available to other researchers. We developed a modern structure using GitHub so that files may be available to all members of the team, other collaborators, and publishers. We do this by 1) evaluating our previous repository structure and workflow; 2) creating a new and consistent structure and workflow among all project repositories; 3) and establishing and maintaining a file naming convention which encompasses any file that could be in a repository. This repository structure takes into account the need to manage “living data” and the necessity for ecological efforts to be transparent.

**Introduction**

Traditional field ecology is currently experiencing a data revolution. Advancing technology drives the need to collect and analyze regularly updated data for natural resource efforts, (i.e., “living data”, (Yenni et al., 2018). These developments require scientists to work effectively with these “living data” (Box 1), despite they are rarely trained to do so (Lowndes et al., 2017). A recent NSF (National Science Foundation) survey (Lowndes et al., 2017) demonstrated that of the 704 scientists who participated in the survey, “data skills” (e.g. multi-step workflows, ability to store, share and publish data) was identified as the largest unmet need (Barone et al., 2017). Ecologists are also increasingly needing to write code as part of their field, lab, and modeling research (Mislan et al., 2016). The necessity to write efficient code, store the code, and increase analysis reproducibility is frequently mandated by publishers (Alarid-Escudero et al., 2019).

The LCR (Lone Cabbage Reef) project is a large ecological effort to restore an oyster reef to historic elevation levels so that it may endure sea level rise and river discharge. Our project generates data from multiple sources including observations of oyster populations and measurements by field biologists and continuous autonomous water quality data via sensors. These data are updated at different frequencies and require specific attention to be processed (Moreno et al., 2020 hopefully). However, once they are processed there is a need to store these data so that they made be used among project team members and collaborators. We use GitHub version control software to keep track of regularly updated data, and to keep the multiple working projects using these data, organized. This paper describes our approach to address concerns regarding standardizing naming conventions, GitHub repository structures, and data availability to differing LCR project repositories to increase reproducibility and transparency.

**#### Box. 1 Terminology**

GitHub- an online version control software, free, and accessible to anyone with Internet ([www.github.com](http://www.github.com))

living data- data that are being analyzed while also still being collected (Yenni et al., 2018)

version control- a system which allows the users to track iterative changes to code and text (Blischak et al., 2016)

project repository- term used to identify one type of analysis that is conducted on an LCR project dataset

README.md- a markdown file which includes information about folder and files contained in the repository

user- any person using GitHub

team member- specifically referring to an LCR project collaborator

admin- specific members of the LCR project tasked to monitor project repository’s pull requests in GitHub

pull request- a process in GitHub to submit changes and a message containing a description of the change, in order to track the version of the change, an additionally an admin can review and accept the changes

`master` branch- the first branch created with a GitHub repository, in the contexts of the LCR project, it is the most up to date branch with the most user limitations for pull requests, this branch requires admin reviews

branches- essentially copies of the GitHub repository `master` branch, in the context of the LCR project each collaborator specifically make edits to their own branch prior to a pull request to the `master` branch

merge conflicts- when branches have competing commits during a pull request, needing to be resolved by an admin (<https://help.github.com/en/github/collaborating-with-issues-and-pull-requests/about-merge-conflicts>)

dark data- data which is not easily found by potential users (Heidorn, 2008)

commit message- a written text of why a pull request is being submitted

**#### End of Box 1**

**GitHub and version and control**

Scripts and text can be difficult to track changes in collaborative efforts (Blischak et al., 2016) . To keep track of these changes version control systems are generally implemented. Version control helps the user track changes by requiring the user to write comments for each change. The version control software will then create a unique version identifier for each iteration and allow the user to revert back to those changes if needed (Noble, 2009). This is especially useful when a user decides to share their code or text with other collaborators. When the user receives new comments from the code/text they have shared, they are able to see what has been changed in relation to the original document. The user will then be able to implement those changes through a version control system.

In the LCR project, code collaboration is a common practice. Code writing collaboration in our project ultimately ended in several files of the same script but each file was a slight iteration of each other (e.g, rscript\_1.R, rscript\_2.R) . It became confusing which script was the most up to date, and which script should be used for which analysis. Using version control stores each iteration of every file, keeping repositories free of duplicate code files. We utilize GitHub to keep a record of which files have been changed, who has changed them, and why they were changed. Using a version control software has saved us timed in trying to determine when and why certain script changes were implemented.

**Challenges working in one repository**

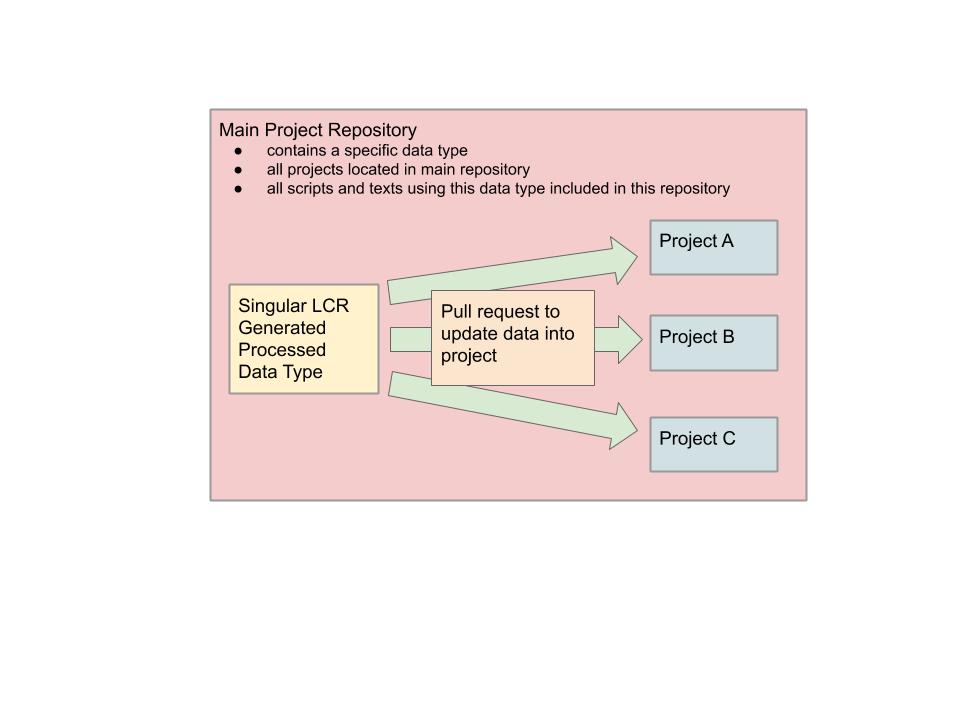
 As the LCR project started to generate a consistent stream of data, it became apparent that its GitHub structure was becoming increasingly more difficult to maintain and manage. One of the main complaints was that it was difficult to find scripts and their data source. Collaborators working in the main GitHub repository were not always following repository guidelines, however the guidelines at the time did not address many of our newfound needs such as how to account for multiple working projects using the same data. Without having proper guidelines, the main GitHub repository quickly began to grow and expand (Figure 1). Our main GitHub repository started to store multiple projects inside of itself, leading to a confusion of what was in the repository was which data and scripts were used for each of the different projects. We soon came to realize that the GitHub repository structure we had employed was not effective in keeping our files or projects organized. We want to note that this described repository structure could work for ecological efforts, but for our project it became too cluttered and unclear to continue using this particular repository structure.

Figure 1- Visualization of our main project repository structure and various projects in the same repository

**Creating a new GitHub repository structure and workflow**

The main goal of our new GitHub repository structure is to keep different projects separate but to have one common data source in which they may be able to update their data from. We extracted the different projects inside our main GitHub repository and created individual project repositories. Each project repository follows the same guidelines for folder structure (https://github.com/LCRoysterproject/repo\_structure). These new project repositories also include descriptions in their README.md about the folders and files inside of them. These README.md’s are essential in maintaining transparency of what each script does and what their outputs are. Our project team members are required to make updates to README.md’s as they create new files for scripts and text, ensuring clarity and transparency in the repository.

These individualized project repositories are self-sustaining and only team members actively working on that project have access to these types of repositories. These repositories are self-sustaining in the sense that they are independent from one another, and their scripts are not influenced by other project repositories’ scripts. Because some of these project repositories need to access the most up to date LCR project generated data (e.g., water quality, oyster measurements, oyster counts) it was imperative that our workflow included a way that the project team member could access these data. All LCR generated data are processed and then stored in a master data repository. This data repository may contain relevant data which could be used among different project repositories (Figure 2). This master data repository also contains commonly used scripts and text. Every LCR team member has access to this master data repository. In the event in which a team member needs to update the data for their analysis, they are able to do so without limitations or approval. These data are also protected from being wrongfully edited or deleted by GitHub branch permissions. Only LCR project admins are allowed to make updates or changes to the master data repository, which adds an extra layer of security to this repository.

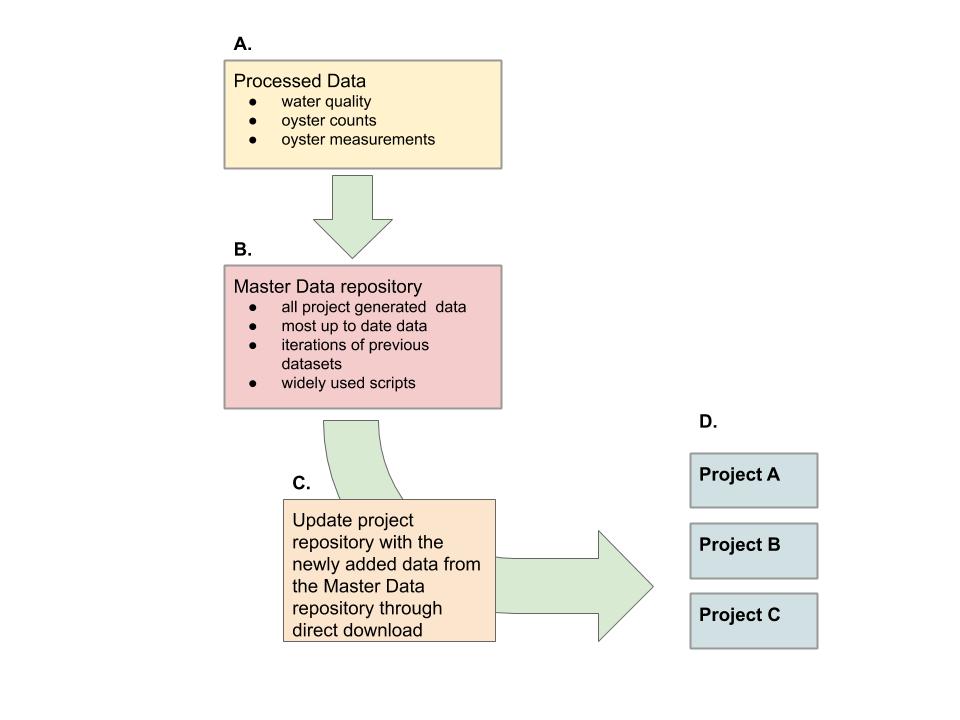


Figure 2- New LCR project workflow which describes how project repositories update to work with newly added data. A. Data are processed and cleaned via MySQL or scripts, B. Data are updated to the master data repository, reviewed, and approved by LCR project admins, C. Project repositories update their repository data by downloading the data directly from the master data repository, D) Project repositories conducting individual analysis on LCR datasets with newly downloaded data, ready for reporting and analysis

**GitHub permissions and branch workflow**

GitHub has settings which can limit who can edit or modify a repository’s branches (Perez-Riverol et al., 2016). For each LCR project repository, which has collaborators, we establish a protected `master` branch, and open collaborator branches. Collaborators are able to edit and modify their own branches however they please, but they are not able to update or modify the `master` branch unless approved by a project admin via a pull request (Box 1). Project admins are expected to review a pull request rigorously and work with the collaborator if there are any discrepancies in the pull request. Using a system that checks the work of collaborators has helped us reduce errors in code, text, and data and can be implemented across many ecological efforts.

Furthermore, GitHub has the functionality to make repositories public or private. Whether a repository is public or private is ultimately up to the adminstrators of that project repository. Public repositories are open and searchable to the public. Private repositories are only initially viewable to the creator of the repository. Additionally, in the LCR project we also limit the users who have access to any given repository. All users have access to the master data repository, however they do not have access to other collaborator repositories, unless an LCR admin grants them access. We allow some repositories to be public and protected, and actively worked on repositories to be private. GitHub allows project managers to change the status of any repository, to private or public, at any time. These types of repository functionalities can allow many ecologists to actively work on their research while protecting their data, analyses, and findings.

**Naming conventions for repository, files, and folders**

Proper file naming conventions help users understand the contents of the file without having to click on it. For scripts, naming conventions exists in which if a script file creates a function or a certain output the output file should also be named the same (<https://style.tidyverse.org/package-files.html#names-1>). In the LCR project we created a set of consistent set of guidelines for filenames (Table 1). The overall guidance to naming files is to keep the cases consistent, in our case lowercase, and to keep the structure of naming the file the same. We use an overall naming guidance of study, location, and project summary, in that order, to name files. These guidelines help each collaborator really think about (A why they are creating a new file; and (B what does this file ultimately intend to do. These guidelines have allowed the LCR project to also increase its transparency among other ecologists. We hope that these file naming guidelines it will help other ecologists in pursuit of standardizing their filenames and increase their file transparency.

|  |  |  |  |
| --- | --- | --- | --- |
| File Type | Naming Convention | Example | Definition |
| Project Repository | study\_location\_projectsummary | bird\_bb\_monitoring | Big Bend camera and survey bird monitoring project |
| Scripts | lowercase, no uppercase (camelcase) nor all caps, all names with separate words need to include a underscore ( \_ ) and no spaces, no dates in the names unless it helps with the descriptions of the conten script file names should be descriptive and concise. Scripts that have a single output should be named in a similar fashion to its filetype output. | discharge\_1941\_2018\_quantile.R | R script which reports quantiles from river discharge from 1941 to 2018 |
| Figures | study\_location\_type\_summary.filetype | oys\_lco8a\_map\_transect.tiff | oyster transect on reef element LCO8A map in a tiff image |
| Tables | study\_location\_summary.filetype | wq\_lcr\_inshore\_vs\_offshore.csv | LCR water quality inshore and offshore comparison |
| Data | every dataset file is required to be in lowercase, no uppercase (camelcase) nor all caps, all names with separate words need to include a underscore ( \_ ) with **no spaces**, no dates in the names unless it helps with the descriptions of the content | discharge\_1941\_2018.csv | River discharge data from 1941 to 2018 in a text file |

Table 1- Table of naming conventions for file types, example, and description of the example

**Transparency**

GitHub has options to increase transparency for an ecological project. Hosting a public project repository on Github can improve the probability of researchers and the public to find the repository, and possibly collaborate. Commit messages, through a pull request, are easily seen and located in GitHub, and allow for collaborators to understand any change submitted to the repository. Transparent repositories are unlikely to be “scooped” by another researcher that can claim the data and the analysis is theirs, through the continuous stream of commit messages leading to the final product. An additional benefit to a transparent repository is that many eyes will be available to evaluate code and text, which can increase the time is takes to debug a script issue. Generally, most scientists with interested in related research are more willing to collaborate with the original scientist than compete with them (Prlić & Procter, 2012).

**Conclusion**

Ecology is an ever-growing science requiring a constant need to keep up to date with the latest technological advancements in field data collection. With that, it is necessary to create a cautious and manageable plan to store and maintain ecological “living data”. These data provide unique challenges in management and reproducibility. Using a well-established GitHub repository for “living data” leads to effective and easy data sharing (White et al., 2013), which increases the transparency of the effort leading to a greater impact (Piwowar et al., 2007).

We employed a repository structure workflow on the recommendations of the University of Florida Academic Research Consulting & Services (ARCS). The repository structure we developed for the LCR project solves many of the challenges we were facing while working with our own generated “living data” and multiple working projects. Separating the multiple working projects into their own repositories keeps the projects simple and organized. There is also little confusion on what is in the repository specifically since every repository is required to include a README.md describing every folder and file. A README.md file also shapes the first impression of a repository and increases the searchability of repositories in GitHub (Prana et al., 2019), which may lead to greater transparency and collaborative efforts (Jones, 2013).

Advantages of our approach include that GitHub and free and accessible to anyone with internet. There are also many training programs which can teach a user how to utilize GitHub efficiently for their project (e.g., <https://guides.github.com/activities/hello-world/>, lab.github.com). Another advantage in using GitHub is that if a repository is accidently deleted, a user has 90 days to retrieve the repository. A benefit in using our described approach is that it can be applied to any ecological effort which has a consistent stream of data by allowing a master data repository to be accessible to all team members while still protecting the repository from adverse or unintentional changes. Additionally, this type of workflow can easily be maintained by a small group of ecologists with basic GitHub workflow training (Yenni et al., 2018) .

Some disadvantages to our approach include that ecological projects, conducted in smaller teams, normally do not have the funding to hire full-time data manager (Hampton et al., 2013), making it a timely effort to curate an efficient data management workflow, initially. Data management planning is also typically and generally underutilized and underappreciated in ecological project designs (Michener & Jones, 2012). It is up to the project team member to manage the data, take the initiative, and adhere their GitHub data management workflow. It is also important to note that even though GitHub has been discussed in detailed as an effective way of project organization and transparency, it might not be a one stop solution for reproducibility in science (Ram, 2013).

The approach we have described in this paper is meant to be a guideline for ecological efforts who desire to make their project organized through concise workflows, standardized naming conventions, and well documented README.md’s. Our hope is that this paper can serve as a mechanism for designing version control software, such as GitHub, to meet the needs of an ecological project with a continuous stream of data and multiple working analyses and projects. Increasing transparency through managing a well-documented repository, through README.md’s, may lead to useful future collaborations. The investment in creating a thorough data management workflow in GitHub will help decrease the time is takes to effectively reproduce analysis by reducing the time it takes to locate files and their outputs, which will allow ecologists more time to analyze their data and less time trying to manage it.

**Chapter 3**

**Case Study: Deer Island, Florida Time Period Shoreline Analysis Using DSAS**

**Abstract**

Climate change perpetuation and sea level rise have led to Gulf of Mexico shoreline dynamics concerns. Shoreline dynamics in areas of coastal development have been intensely studied, however many under-developed shorelines have yet to be analyzed. In this study we used seven NAIP (National Agriculture Imagery Program) aerial images, from 1994 to 2019, of our study area near Cedar Key, FL. The cloud-free images were collected during relatively similar mean river discharge levels and during (mostly) the same season. We assessed the shoreline changes using ESRI’s ArcMap© spatial data analysis extension DSAS (Digital Shoreline Analysis Systems) on three different time periods in from the imagery, 1994-2007, 2010-2019 and 1994-2019. The DSAS analysis is a transect- based approach and is used to quantify shoreline changes on a linear ocean shoreline. From this analysis we determined the greatest areas of impact and have been able to speculate on possible factors that may contribute to an escalated shoreline change rate during a selected time frame.

**1. Introduction**

Shoreline changes can occur due to multiple factors including SLR (sea-level rise), anthropogenic human activity and hurricane intensity (Yu et al., 2011). The combination of these processes can influence erosion and accretion. Shoreline changes may affect a shoreline’s resilience to storm surges including flooding and species diversity implications (Desantis et al., 2007). It was observed by USGS (United States Geological Survey) that shoreline changes along the Gulf of Mexico, specifically in Florida, were relatively steady between the 1800s and 1990s (Morton et al., 2004). Since then, the Gulf of Mexico coastline, with its low relief geomorphology particularly along the west coast of Florida, has been noted to be vulnerable to coastal erosion (Geselbracht et al., 2011).

***1.1 Climate change and SLR***

More recently, climate change induced SLR and its impacts on coastal zones is of growing interest. The Earth’s climate is warming due to an accumulation of greenhouse gases in the atmosphere, largely in part due to anthropogenic fossil fuel burning and deforestation. Warming climate change causes thermal expansion of sea water, and land ice to melt into the ocean, initiating SLR (Cazenave & Cozannet, 2014).Sea-level rise is considered to be a likely candidate for widespread global erosion. Erosion occurs when SLR drifts the high-water line (line on the shore where the water usually reaches at high water) landward in relation to the slope of the coastal area. Erosion on sandy beaches involves the relocating of sand from the beach to offshore. This is normally observed during storm events. Storm events temporarily increase the local sea-level of the sandy beach, and ultimately storm waves are able to reach higher elevations on the beach. After a storm event much of the sand returns back to the beach by swell waves during normal sea water levels. This exchange implies that sea water levels have a direct relationship with sandy beach erosion (Zhang et al., 2004).

***1.2 Characteristics of sandy shorelines and sedimentation***

Sandy shorelines are characterized by active environments and unstable substrata, which consists of sand, mixed sand, quartz, and/or silica. The unstable nature of sandy shores make a harsh ecosystem for biota and may incorporate a significant range of physical environment conditions and ecosystem functioning. These shorelines accumulate sediment accretion by wave deposited particles. Particles originate from inland erosion and may be transported by rivers (Brown & McLachlan, 2002). Sediment to sandy shores may also be added by marine biogenic sources such as pieces of marine skeletons, sponge spicules, and shell fragments (McLachlan, 1990). Threats to sandy shorelines include disruption of sand transport, storms, SLR, and human activities.

***1.3 Suwannee River sedimentation and discharge***

The Suwannee River is the second largest river in Florida spanning 396 kilometers long and is considered to be a significant point source of sedimentation near our study site, approximately 11 kilometers north. The Suwannee River is a partially spring-fed system which also drains the coastal plain of Georgia and provides a restricted point source input of siliciclastic sediment, creating a small 20-kilometer delta. The surrounding coastal regions of the Suwannee River are otherwise known to be sediment starved, but a great significant sedimentology event has been shown that the Suwannee River has reworked ancestral fluvial sands and serves as a source for sandier marsh sediments (Wright et al., 2005).The Suwannee River normally has high discharge peaks between February and April and low discharge peaks between August and October (Purtlebaugh & Allen, 2010). The average annual discharge is 300 m^3/c with a minimum of 83 m^3/c and a maximum discharge of 2400 m^3/c (Wright et al., 2002).

***1.4 Human development and impacts***

Three Florida counties encompass the region of our study site, Dixie, Levy, and Taylor. These counties which are projected to increase in human population by 2045 as depicted in Figure 3. These Florida counties are recorded to have lowest population densities along the Florida coastline (Geselbracht et al., 2011) . In the future it could be likely that businesses and people will want to develop housing along this shoreline. Human development on coastlines may accelerate coastal erosion by creating a fixed position of the shoreline and stabilizing inlets (Finkl & Charlier, 2003). Increased human developments may also negatively impact coastal species diversity. Species biodiversity is threatened by the increase of urbanization and environmental coastal degradation (Finkl & Charlier, 2003) . Czech’s et al. (2000) documents urbanization as the highest cause for species endangerment. For example, the shorebird Piping Plover (*Charadrius melodus*) is known to forage and nest in areas of low human population (Thomas et al., 2002) ,implying that shoreline areas with higher human densities would not be an ideal habitat for this species. Species biodiversity, both vegetative and animal, could be at risk due to an increase of urbanization along coastlines (McKinney, 2006) and accelerated shoreline erosion.

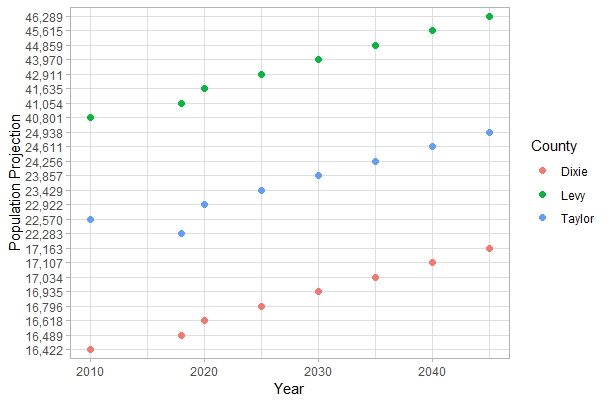


Figure 1- Generated figure based on census and projection data from Bureau of Economic and Business Research (<https://www.bebr.ufl.edu/population>).

***1.5 Big Bend habitats for species richness***

The Northeastern Gulf of Mexico region of Florida is ranked as an area of high importance for conserving and protecting habitats for at least 30 species of shorebirds. Within those thirty species, four threatened species are considered to be of “extremely high priority” for protection, and include the American Oystercatcher, Red Knot, Snowy Plover, and Piping Plover (Withers, 2002). The coastlines in the Big Bed region (Figure 2) are described as having low wave energy (described as waves falling well below the high-water line of a shore), which can be ideal for migrating shorebirds because low wave energy on shorelines can facilitate the accumulation of vegetative litter and food such as horseshoe crab eggs (Nordstrom et al., 2006) .These shorebirds use the primarily cordgrass marsh shorelines habitats of the Big Bend for foraging, mating, and shelter. Shorebirds in the Big Bend have been reported to have the least abundance and species richness, in a study comparing Gulf of Mexico regions shorebird use of coastal habitats (Withers, 2002).

***1.6 Major Hurricanes in the Gulf of Mexico***

The Storm of the Century, hit the west coast on March 1993, was a Category 5 hurricane with wind speeds up to 160.9 kmh. The Storm of the Century caused devasting damage to the Waccasassa Bay (approximately 30 kilometers south of our study site), 3-meter water storm surges (Figure 2), a storm deposit which reached 12 cm on the levees and up to 2 cm on the marsh surface (Goodbred & Hine, 1993a). Hurricane Irma, also hit the west coast of Florida on September 2017, was recorded as a Category 3 hurricane (Figure 3), with wind speeds up to 193.1 kmh. Heavy amounts of rainfall were recorded with Hurricane Irma at a peak of 550 mm in Fort Pierce, Florida. Heavy rainfall and storm surges, highest record was 2.3 m, contributed to many creeks and rivers overflooding (Pinelli et al., 2018).

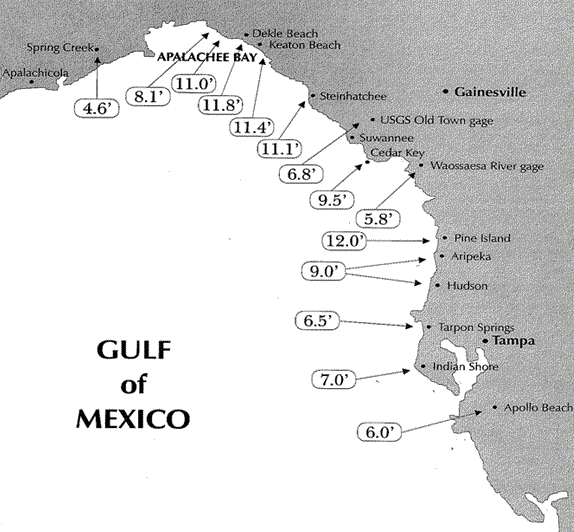
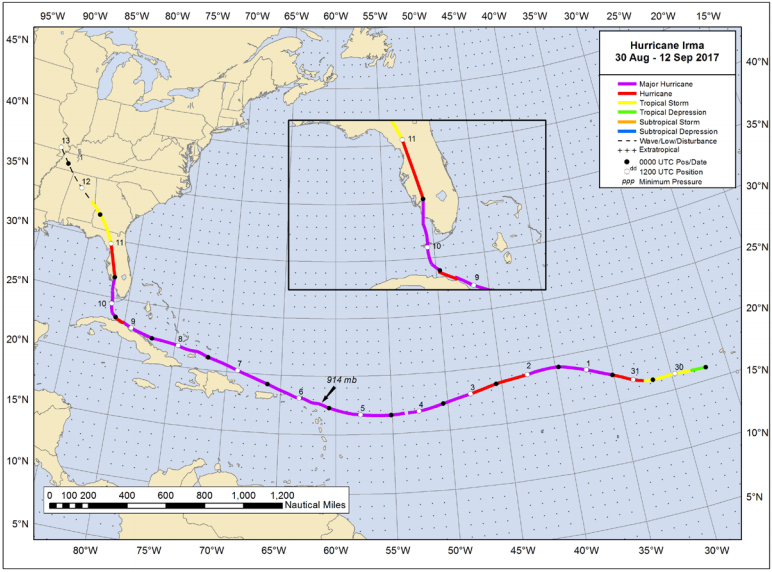


Figure 2- Storm surge, in feet, associated with the Storm of the Century, 1993. (National Hurricane Center)

Figure 3- The path and intensity of Hurricane Irma, 2017. (National Hurricane Center)

***1.7 Reason for effort***

This shoreline analysis study tried to identify possible factors that may be influencing shoreline loss. Since the study site is uninhabited, and tourism is not prevalent these were not considered as probable factors. The analysis was split into three time frames in order to locate an area of shoreline change where an identifiable factor may have triggered shoreline erosion or accretion. Two out of the three time frames spilt up the available imagery into equal years, however there are not an equal amount of imagery available covering each 12.5 year period (imagery spans a total of 25 years). The last time frame includes all imagery to calculate how much total shoreline was loss or gained from the years 1994 to 2019.

Shoreline loss as also need near our study site has also been captured recently. In the mid-1960s the US Army Corps of Engineers constructed spoil islands as part of the cross Florida barge canal project. These spoil islands consist of a straight line of islands perpendicular to the west Florida coast. Coastal changes have severely eroded or inundated these spoil islands, thus reducing habitat for animals (Vitale, 2019).Derrick Key is an example of a spoil island that was clearly visible in aerial photographs in 1982 and now the island is completely submerged (in 2016 photography). Major shoreline differences are noticeably observed in the 34 years, time between the imagery, for this specific spoil island. Large scale efforts to analyze shoreline changes in Florida have been studied in the past (Yu et al., 2011; Sassaman et al., 2017; Houston, 2015; Li & Gong, 2016) however it is interesting to note the effects of SLR on a smaller or regional scale, which might highlight processes which might be affecting ecosystems and habitats on a larger-scale.



Figure 4 - Island degradation of Derrick Key in the Cedar Keys, Florida from 1982 (left) to 2016 (right), (Vitale, 2019).

**2. Materials and methods**

***2.1 Study Area***

Our study area is located on the west-central Florida coastline in the Suwannee Sound region of the Big Bend (Figure 5). The selected shoreline is a small barrier island called Deer Island. Deer island is a privately owned uninhabited island approximately 13 kilometers north of the main villages of Cedar Key, Florida. Historically, Native Americans intermittently inhabited Deer island for thousands of years. Early Florida settlers were reported to live and camp on the island as well. The 1800 Florida census registered only 4 people to have identified this island as their home. There is a cabin near the south of the island depicted on a 1951 USGS Cedar Key Quadrangle map (USGS, 1955). This island is specifically located in the Big Bend Aquatic Seagrass Preserve and connects with the Lower Suwannee National Wildlife Refuge (http://www.beachrealtyfla.com/DeerIsland.htm). Deer Island is approximately 364217 square meters of total area and consists of 101171 upland square meters (27.7%) and 80937 square meters (22.2%) of wetland with elevations as high as 4.3 meters. The island is densely forested with large pines, cedars, palms, oaks, palmettos and many more plant species (<https://www.privateislandsonline.com/united-states/florida/deer-island>). The shoreline attributes reported on Deer island is about 1.3 +/- kilometers of Gulf of Mexico white sand beach and approximately 1.3 +/- kilometers of waterfront facing the mainland (<https://images1.loopnet.com/d2/Z4L1-alqEsAlhPT_YJ25N8OMkXU3L_mAPAZYXiq2OVg/document.pdf>).



Figure 5- Location of Deer Island, Florida. A) Map of the entire state of Florida; B) Zoomed into map scale of 2.3758 to study site; C) Zoomed into map scale of 0.03 to Deer Island with a scale bar in kilometers. Shoreline shapefile downloaded at my.fwc.com, (1 to 2,000,000 Scale, and digitized in 2017).

***2.3 Imagery selection process***

Locating relatively cloud-free imagery for a specific location in Florida can be an exhaustive effort. Since our study location is unpopulated and contains no popular historic landmarks, so historic aerial images are not frequently taken. To reduce the effort on locating usable imagery, Google Earth Pro was utilized. Google Earth Pro does not capture any of its own imagery, it does however locate and use imagery, in its finder view, that is comparatively cloud-free and with the highest resolution. Google Earth Pro was able to give minimal metadata of the imagery such as which agency captured the imagery and the date of the image, when using the time slider feature. Then USGS’s Earth Explorer (<https://earthexplorer.usgs.gov/>) was used to further locate the actual imagery and collect its metadata. After Upon inspection it was determined that NAIP (National Agriculture Imagery Program) was the agency that acquired the most frequent and most detailed aerial imagery of our study site. The specifications for NAIP aerial imagery require 1-meter ground sample distance with a horizontal accuracy that matches within six meters of photo-identifiable ground control points. These points are then used during imagery inspection. Contractually, NAIP makes attempts to comply with the specification that no more than 10% cloud cover be allowed in each aerial imagery tile. Aerial imagery are available as digital ortho quarter quad tiles (DOQQs) geotiffs, and which also correspond to the USGS topographic quadrangles (<https://www.fsa.usda.gov/programs-and-services/aerial-photography/imagery-programs/naip-imagery/>). It was also important to select imagery that were fairly at the same time of the year, similar river discharge and precipitation levels. All imagery chosen are between the months of October through January. Normally, during the Florida winter months, precipitation and river discharge levels are generally low (Bhardwaj and Misra, 2019). The table below includes all metadata associated with the imagery used in this analysis. Furthermore, observed weather and median river discharge were collected, including the observed weather for the day of imagery collection and median river discharge measured.

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Median River Discharge (cfs)  Station ID= 02323500 | Observed weather | Metadata (USGS Earth Explorer) |
| January 20, 1994 | Value= 9710 | Avg Temp (C)- 3.41 Precipitation (cm)- 0.00  Max Wind Speed (KPH)- 19.31 | Entity ID: DI00000000018672 (found in DOQ)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 3  Sensor Type RGB |
| December 30, 1998 | Value= 6370 | Avg Temp (C)- 9.30 Precipitation (cm)- 0.00  Max Wind Speed (KPH)- 25.75 | Entity ID: DI00000001164809 (found in DOQ)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 3  Sensor Type RGB |
| November 02, 2007 | Value= 2350 | Avg Temp (C)- 19.31  Precipitation (cm)- 0.00  Max Wind Speed (KPH)- 22.53 | Entity ID: N\_2908356\_NW\_17\_1\_20071102 (found in NAIP)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 3  Sensor Type CIR |
| September 19, 2010 | Value= 4240 | Avg Temp (C)- 25.32 Precipitation (cm)- 0.00  Max Wind Speed (KPH)-24.14 | Entity ID: M\_2908356\_NW\_17\_1\_20100919 (found in NAIP)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 4  Sensor Type CNIR |
| October 13, 2013 | Value= 8200 | Avg Temp (C)- 22.13 Precipitation (cm)- 0.00  Max Wind Speed (MPH)- 10 | Entity ID: M\_2908356\_NW\_17\_1\_20131013 (found in NAIP)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 4  Sensor Type CNIR |
| November 12, 2015 | Value= 6070 | Avg Temp (C)- 19.27 Precipitation (cm)- 0.00  Max Wind Speed (KPH)- 16.09 | Entity ID: M\_2908356\_NW\_17\_1\_20151112 (found in NAIP)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 4  Sensor Type CNIR |
| October 26, 2017 | Value= 7990 | Avg Temp (C)- 12.60 Precipitation (cm)- 0.00  Max Wind Speed (KPH)- 14.48 | Entity ID: M\_2908356\_NW\_17\_1\_20171026 (found in NAIP)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 1.0  Units METER  Number of Bands 4  Sensor Type CNIR |
| November 10, 2019 | Value = 5190 | Avg Temp (C)- 14.12 Precipitation (cm)- 0.00  Max Wind Speed (KPH)- 11.27 | Entity ID: M\_2908356\_NW\_17\_060\_20191110 (found in NAIP)  Map Projection UTM  Projection Zone 17N  Datum NAD83  Resolution 0.60  Units METER  Number of Bands 4  Sensor Type CNIR |

Table 1- Table of information for each aerial image used in this analysis including date, median river discharge, observed weather, and additional imagery metadata. River discharge information is calculated by data from <https://tidesandcurrents.noaa.gov/> at Cedar Key, Florida Station 8727520, and observed weather provided by [www.wunderground.com](http://www.wunderground.com). Imagery metadata is provided by USGS Earth Explorer, <https://earthexplorer.usgs.gov/> .

National Agriculture Imagery Program employed sensor types which had three band imagery categorized as RGB (red, green, blue), up until 2007. After 2007, four band color infrared imagery were collected and categorized as CIR/CNIR (red, green, blue, and infrared). Four band imagery is multispectral, which means the sensors can collect information from several parts of the electromagnetic spectrum. The metadata in Table 1 includes the sensor type associated with each image. Our November 2007 image is the first image in our series which uses color infrared (CIR/CNIR). The advantage of using CIR/CNIR imagery us that it allows the user to view the imagery in a false color for NDVI (Normalized Difference Vegetation Index) analysis. In this analysis our shoreline does need to identify vegetation on the sandy shoreline.

Most GIS software can only display three bands at one time. Vegetation can be seen on Deer island but it is not necessary for our DSAS analysis because the island vegetation is distinctly not integrated into the sandy shoreline. Additionally, the DSAS user manual does not have any recommendations for using or not using true color image composites. To specify a natural color display the settings in the GIS software should be band 1 set to red, band 2 set to green, and band 3 set to blue. True color in this instance is useful because the sandy shoreline of Deer island is in stark contrast with the dark ocean water.

|  |  |  |
| --- | --- | --- |
| Sensor Type | Color and wavelength (µm) | Band and color channel to display true color |
| RGB | Blue 400–500  Green 500–600  Red 600–700 | 1 – Red channel  2 – Green channel  3 – Blue channel |
| CIR/ CNIR | Blue 400–500  Green 500–600  Red 600–700  Near Infrared 800–900 | 1 – Red channel  2 – Green channel  3 – Blue channel  4 – Near Infrared (not show on screen display) |

Table 2 - National Agriculture Imagery Program (NAIP) aerial imagery band wavelength ranges in units (µm) (<https://www.fsa.usda.gov/Assets/USDA-FSA-Public/usdafiles/APFO/support-documents/pdfs/fourband_infosheet_2017.pdf>)

***2.3. Digital Shoreline Analysis System (DSAS)***

The DSAS is a GIS-based system created and maintained by USGS (United States Geological Survey). For this analysis the DSAS ArcMap© extension was used. The DSAS extension casts transects along the baselines (starting point for transects) and measures the gaps between the shoreline positions during defined years. Baselines are constructed by the user, and in this analysis was created using the Buffer tool in ArcMap©. These shoreline positions provide the basic data needed to calculate their shifts. One of each type of change metric (as described in the DSAS Overview, <https://pubs.usgs.gov/of/2018/1179/ofr20181179.pdf>) was used in this analysis, an LRR (Linear Regression Rate) for statistical analysis and the Net Shoreline Movement (NSM) calculation for the distance measurement. A linear regression rate-of-change can be ascertained by fitting a least-squares regression line to every shoreline point in a transect. The regression line is positioned so that the sum of the squared residuals is at its most minimal. The linear regression rate is the slope of the regression line. The NSM is the distance between the oldest shoreline portion to the youngest shoreline position for each transect, calculated in meters. The LRR statistic was used it because all the data provided is used regardless of accuracy, and the calculations is based on accepted statistical notions. In contract NSM statistics only require the baseline position and the last shoreline position to make its calculations. The justification for using NSM statistics is to know the total measurement of erosion and/or accretion, which has high biological significance in that this will be the first time this kind of measurement will be conducted on our study site and even in the surrounding areas.

The DSAS analysis generates transects which are perpendicular to the reference user created baseline (Figure 6). The analysis explains that an intersection point is a cross between the casted transect and the shoreline boundary position for each specified year. The DSAS analysis then uses the distance, in meters, to conduct various calculations, which were previously described. Using the distance between transects, the DSAS can also generate forecasted transects for10- and/or 20-year projections.

Figure 6 - Example of DSAS transect casting (<https://www.usgs.gov/centers/whcmsc/science/digital-shoreline-analysis-system-dsas?qt-science_center_objects=0#qt-science_center_objects>)

The DSAS calculations require an operational workflow to gather and create the necessary components. The components needed are shoreline baselines, additional shorelines of interest (varying in different time periods), DSAS transects (which are cast some the baseline and intersect the additional shorelines positions), measurement distances, measurement points, and shoreline uncertainty. All objects used in the DSAS are stored in an ArcMap© Personal Geodatabase, as per USGS requirements for this analysis. The DSAS operational workflow includes the following steps: (1) Set default parameters and fields to created shoreline and baseline layers, transects, shoreline calculations, metadata and file output locations; (2) Cast transects and select their maximum search distance, transect spacing, and smoothing distance; (3) Calculate change statistics such as confidence intervals, shoreline intersection threshold, rate of output display, and summary report; (4) Create data visualization for LRR and NSM; and (5) Shoreline forecasting for a 10 and/or 20 year forecast.

***2.4 DSAS parameters and selections***

Selected NAIP Geotiff aerial imagery were in the Universal Transverse Mercator (UTM) coordinate system, Zone 17 North and in the 1983 North American Datum (NAD83). (Table 1). Using ESRI’s ArcCatalog© and ArcMap©, separate shapefiles for each aerial image’s shoreline was create, traced, and digitized. Shorelines were then merged into a new single shapefile using the ArcMap© Merge tool. The ArcMap© Buffer tool was used to create a new shapefile that contained a 100-meter buffer around each shoreline in the new merged shorelines shapefile. A section of the buffer was selected to act as the baseline for transect casting for the DSAS calculations. The baseline selected can be found on the east side of Deer Island and is entirely inland. Both a baseline shapefile and merged shoreline shapefile are required for DSAS calculations (Figure 7, Inputs).

The DSAS parameters set for this analysis were a 20-meter transect spacing, a 2000-meter search distance for shorelines, and a smoothing distance of 500 meters. A 20-meter transect spacing was the minimum transect spacing allowed for the size of the study site. A 2000-meter search distance looks for shorelines 2000 meters way from the baseline. A smoothing distance is a user- specified smoothing value which can facilitate and orthogonal transect intersect by creating a baseline (which is not displayed in the final product). The intention of the smoothing distance is to prevent transects from intersecting with one another when there is a curve in the baseline. The larger the smoothing distance results in a longer reference line and produce more uniform transect orientations, which is recommended for smaller shorelines. The default setting for 90% confidence interval too calculate LRR and NSM rates remain unchanged.

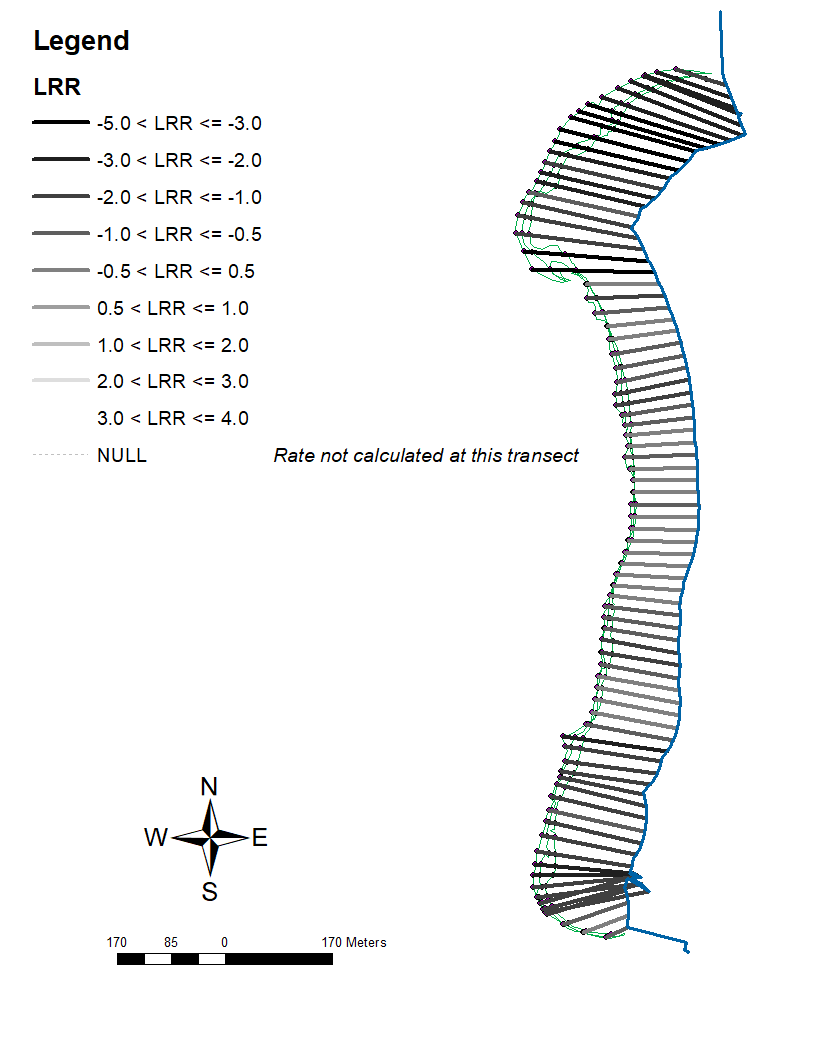
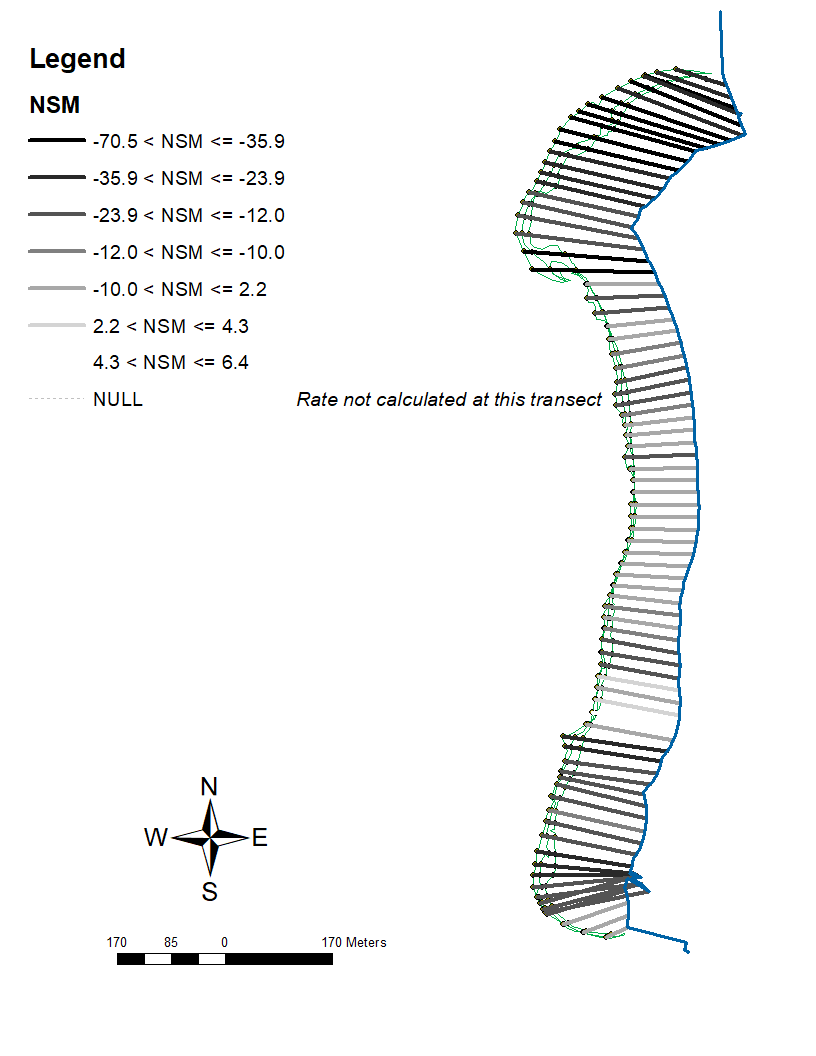


Figure 7 - DSAS components and operational workflow.

**3. Results**

The calculations for the shoreline analysis are displayed in a black and white colorramp. The LRR coloramp displays rates of change of meters per year. The NSM coloramp displays the distance of measurements in meters. The DSAS calculations follows the standard that a negative rate implies erosion and a positivie rate implies accretion. The interpretation of the results go as follows.

***3.1 Shoreline analysis from 1994-2007***

******The DSAS results, Figure 8 (left) display that there were relatively high LRR rates between the years of 1994- 2007. The high LRR erosion rates (Table 3, left) range from -5.0 to -3.0 (m/yr) and the highest LRR accretion rates range from 3.0 to 4.0 (m/yr). The most frequent LRR rate is -2.0 to -1.0 (m/yr) accounting for 30.5% of all transects calculated. The least frequent LRR rates are the accretion rates between -0.5 to 0.5 (m/yr) accounting for 25.6% of all transects calculated. For the NSM (Table 3, right), the highest erosion distance measurements range from to -35.9 meters (n= 10) and the maximum accretion distance measurements range from 4.3 to 6.4 meters (n= 1). The most frequent NSM distance is -10.0 to 2.2 meters accounting for 29.3% of all transects calculated. The least frequent NSM distance is the accretion distance between 4.3 to 6.4 meters accounting for 1.2% of all transects calculated. In the NSM analysis, there is only one transect line that falls in the maximum range of accretion, all other transects are displaying low to moderate erosion meter measurements.

|  |  |  |
| --- | --- | --- |
| Range (LRR) | Count | Percentage of total transects (n=82) |
| -5.0 < LRR <= -3.0 | 7 | 8.5% |
| -3.0 < LRR <= -2.0 | 12 | 14.6% |
| -2.0 < LRR <= -1.0 | 25 | 30.5% |
| -1.0 < LRR <= -0.5 | 17 | 20.7% |
| -0.5 < LRR <= 0.5 | 21 | 25.6% |
| 0.5 < LRR <= 1.0 | 0 | 0% |
| 1.0 < LRR <= 2.0 | 0 | 0% |
| 2.0 < LRR <= 3.0 | 0 | 0% |
| 3.0 < LRR <= 4.0 | 0 | 0% |

Figure 8 - Results of Linear Regression Rates model (left) and Net Shoreline Movement model (right). Shorelines are located on the west side of each panel. Baselines are located on the east side of each panel. The imagery used in this analysis is from 1994-2007.

|  |  |  |
| --- | --- | --- |
| Range (NSM) | Count | Percentage of total transects (n=82) |
| -70.5 <NSM <= -35.9 | 10 | 12.2% |
| -35.9 <NSM <= -23.9 | 13 | 15.9% |
| -23.9 <NSM <= -12.0 | 26 | 31.7% |
| -12.0 <NSM <=-10.0 | 6 | 7.3% |
| -10.0 <NSM <= 2.2 | 24 | 29.3% |
| 2.2 <NSM <= 4.3 | 2 | 2.4% |
| 4.3 <NSM <= 6.4 | 1 | 1.2% |

Table 3- Count statistic of the range, transect count of that range, and percentage occurring of that particular range off DSAS calculations from 1994-2007 LRR rates (left) and NSM distance (right).

***3.2 Shoreline analysis from*** ***2010-2019***

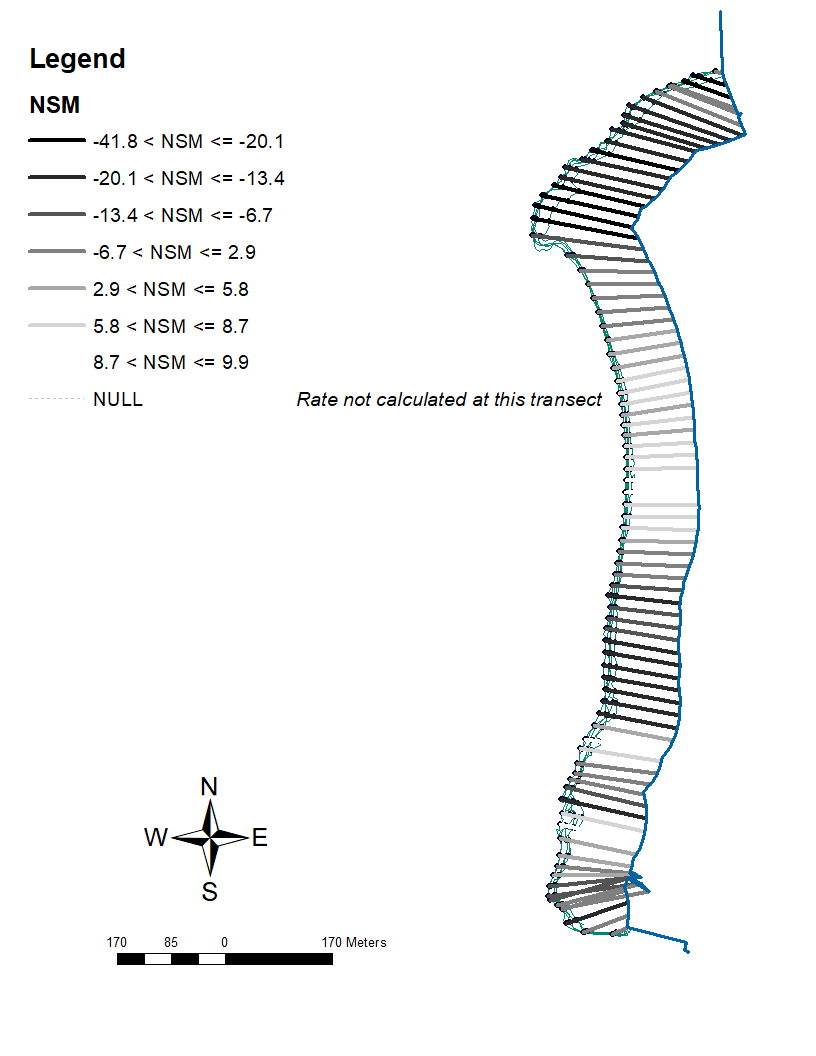
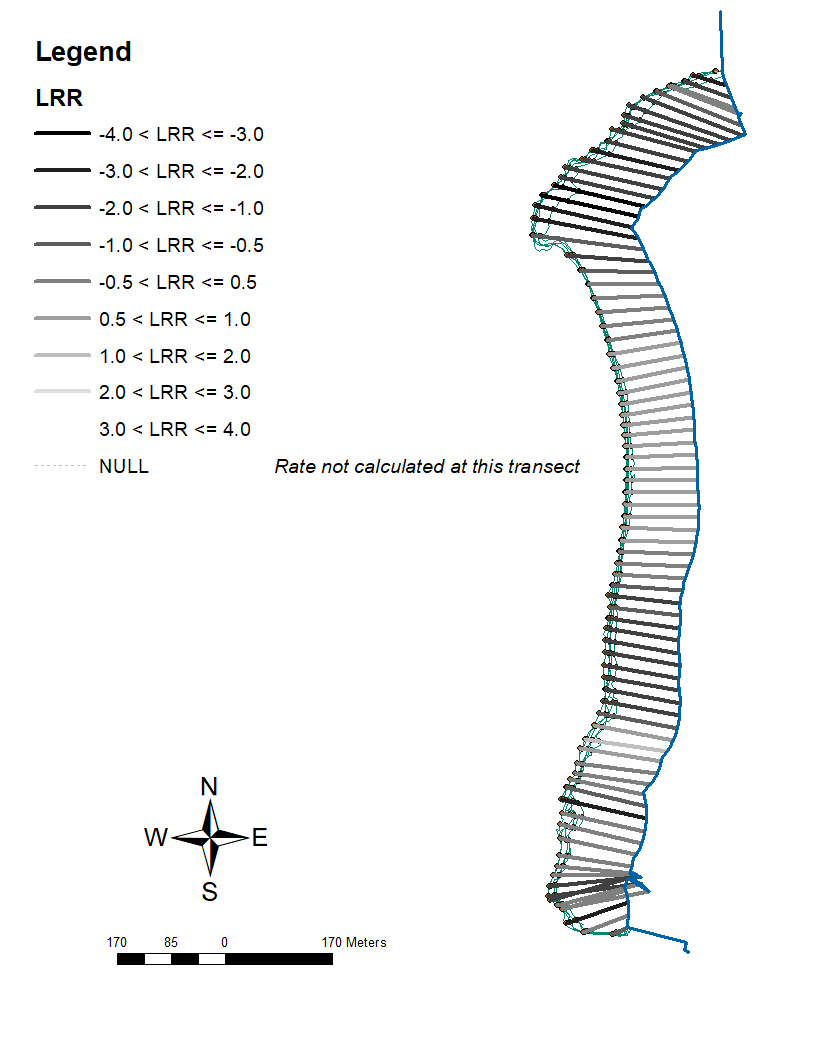
The results displayed in Figure 9 demonstrate a different outcome compared to Figure 6. The high erosion LRR rates (Table 4, left) in this analysis range from -3.0 to -4.0 (m/yr) and the highest LRR accretion rates range from 3.0 to 4.0 (m/yr). The LRR erosion rates during 2010-2019 do not go as high as in 1994-2007. The most frequent LRR rate is -1.0 to -2.0 (m/yr) accounting for 28% of all transects calculated. The least frequent LRR rates are the accretion rates between 1.0 to 2.0 (m/yr) accounting for 1.2% of all transects calculated. For the NSM (Table 4, right), the highest erosion distance measurements range from -20.41 to -41.8 meters (n= 6) and the maximum accretion distance measurements range from 8.7 to 9.9 meters (n= 4). The most frequent NSM distance is -6.7 to 2.9 meters accounting for 25.6% of all transects calculated. The least frequent NSM distance is the accretion distance between 8.7 to 9.9 meters accounting for 4.9% of all transects calculated. The figure above depicts Deer Island as having moderate to high LRR erosion rates, while the NSM shows accretion in the center of the island with some acute high erosion locations in the north and south end of the island.

Figure 9- Results of Linear Regression Rates model (left) and Net Shoreline Movement model (right). Shorelines are located on the west side of each panel. Baselines are located on the east side of each panel. The imagery used in this analysis is from 2010-2019.

|  |  |  |
| --- | --- | --- |
| Range (LRR) | Count | Percentage of total transects (n=82) |
| -4.0 < LRR <= -3.0 | 2 | 2.4% |
| -3.0 < LRR <= -2.0 | 6 | 7.3% |
| -2.0 < LRR <= -1.0 | 23 | 28.0% |
| -1.0 < LRR <= -0.5 | 8 | 9.8% |
| -0.5 < LRR <= 0.5 | 23 | 28.0% |
| 0.5 < LRR <= 1.0 | 19 | 23.2% |
| 1.0 < LRR <= 2.0 | 1 | 1.2% |
| 2.0 < LRR <= 3.0 | 0 | 0% |
| 3.0 < LRR <= 4.0 | 0 | 0% |

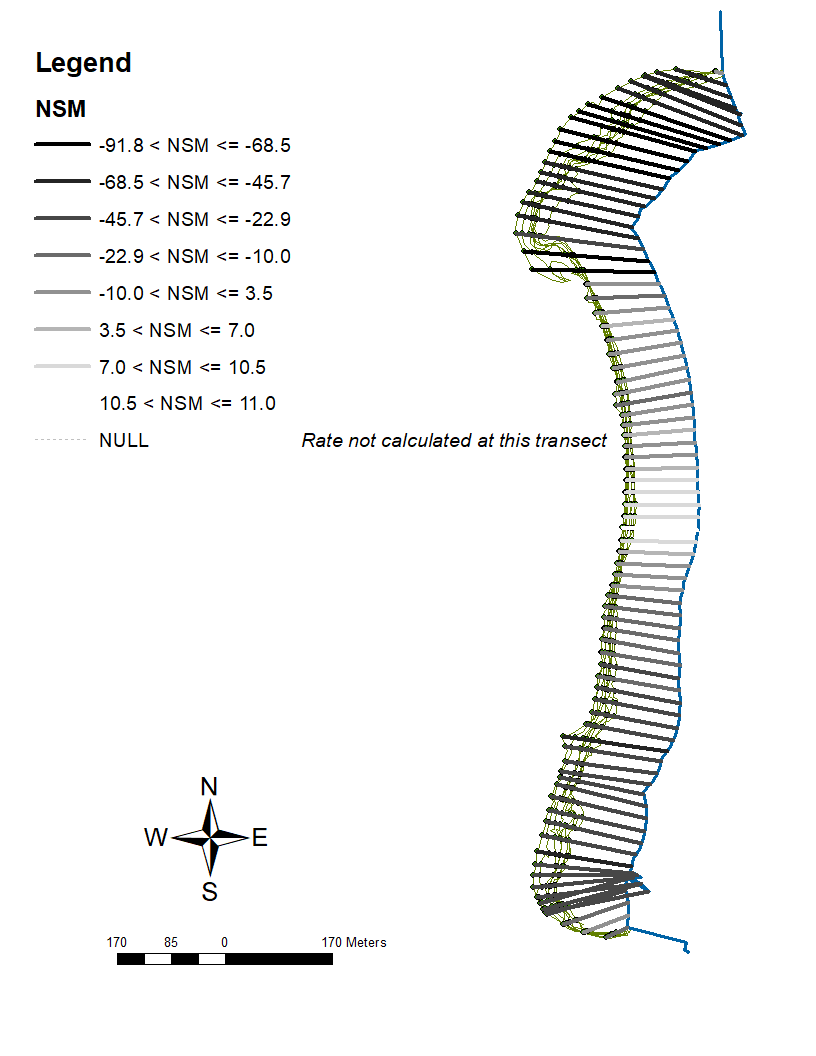
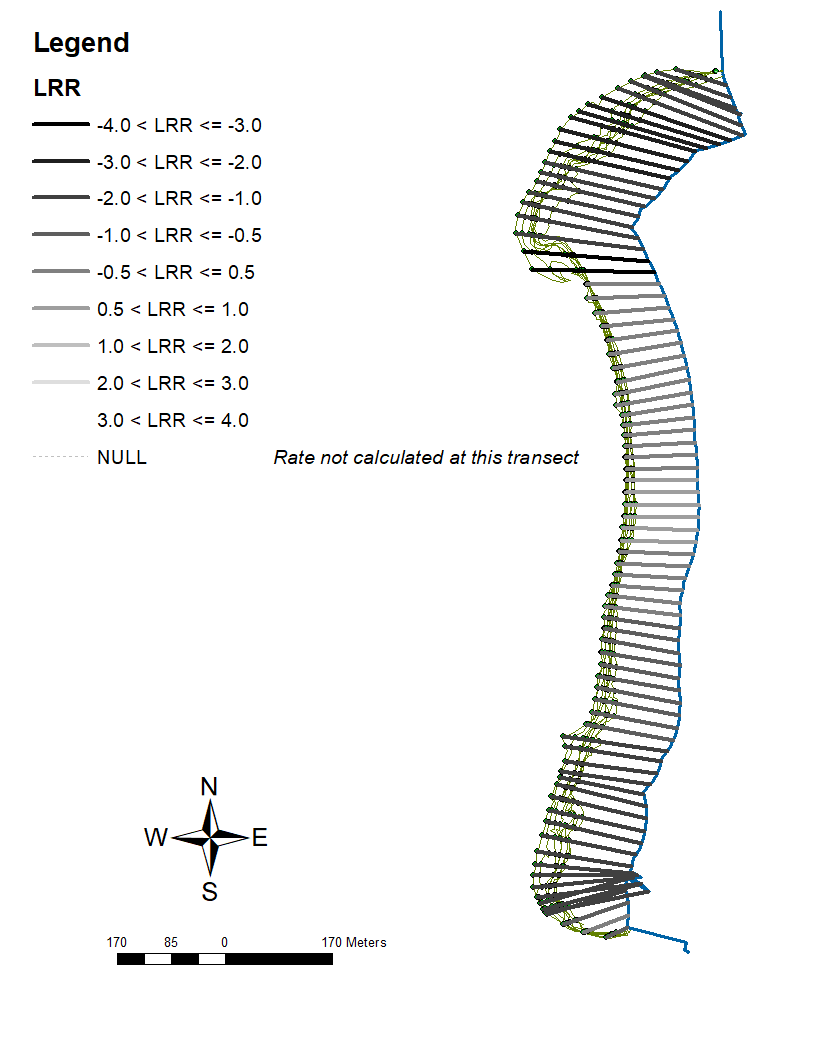
|  |  |  |
| --- | --- | --- |
| Range (NSM) | Count | Percentage of total transects (n=82) |
| -41.8 <NSM <= -20.1 | 6 | 7.3% |
| -20.1 <NSM <= -13.4 | 19 | 23.2% |
| -13.4 <NSM <= -6.7 | 10 | 12.2% |
| -6.7 <NSM <= 2.9 | 21 | 25.6% |
| 2.9 <NSM <= 5.8 | 10 | 12.2% |
| 5.8 <NSM <= 8.7 | 12 | 14.6% |
| 8.7 <NSM <= 9.9 | 4 | 4.9% |

Table 4 - Count statistic of the range, transect count of that range, and percentage occurring of that particular range off DSAS calculations from 2010-2019 LRR rates (left) and NSM distance (right).

***3.3 Shoreline analysis from 1994-2019***

The results in Figure 10 includes all the shorelines from Figures 8 and 9 for its LRR and NSM calculations. The high erosion LRR rates (Table 5, left) in this analysis range from -3.0 to -4.0 (m/yr) and the highest LRR accretion rates range from 0.5 to 1.0 (m/yr). The most frequent LRR rate is -1.0 to -2.0 (m/yr) accounting for 39% of all transects calculated. The least frequent LRR rates are the accretion rates between 0.5 to 1.0 (m/yr) accounting for 7.3% of all transects calculated. For the NSM (Table 5, right), the highest erosion distance measurements range from -91.8 to -68.5 meters (n= 9) and the maximum accretion distance measurements range from 10.5 to 11 meters (n= 1), which is also the least frequent NSM distance. The most frequent NSM distance is -45.7 to -22.9 meters accounting for 25.6% of all transects calculated. The largest erosion measurement distance is seen at the north end of Deer Island, while the middle has some areas of accretion and light erosion. The south side of Deer Island has some acute peaks of erosion, however not as high as the north end.

Tables 6 and 7 display the statistics summary generated by DSAS. In the NSM statistics summary there are a total of 67 transects with a negative distance making up 81.70% of all transect. The maximum negative distance (erosion) is 91.71 meters, while the maximum positive distance (accretion) is 10.91 meters. For the LRR analysis the average rate of yearly erosion is 0.95 meters. For erosional transects (n=63) the average rate is -1.33 m/yr while for accretional transects (n=19) the average rate is 0.31 m/yr. The LRR analysis clearly shows that erosion is occurring at 4 times the rate of accretion on our study site.

Figure 10- Results of Linear Regression Rates model (left) and Net Shoreline Movement model (right). Shorelines are located on the west side of each panel. Baselines are located on the east side of each panel. The imagery used in this analysis is from 1994-2019.

|  |  |  |
| --- | --- | --- |
| Range (LRR) | Count | Percentage of total transects (n=82) |
| -4.0 < LRR <= -3.0 | 2 | 2.4% |
| -3.0 < LRR <= -2.0 | 7 | 8.5% |
| -2.0 < LRR <= -1.0 | 32 | 39.0% |
| -1.0 < LRR <= -0.5 | 12 | 14.6% |
| -0.5 < LRR <= 0.5 | 23 | 28.0% |
| 0.5 < LRR <= 1.0 | 6 | 7.3% |
| 1.0 < LRR <= 2.0 | 0 | 0% |
| 2.0 < LRR <= 3.0 | 0 | 0% |
| 3.0 < LRR <= 4.0 | 0 | 0% |

|  |  |  |
| --- | --- | --- |
| Range (NSM) | Count | Percentage of total transects (n=82) |
| -91.8 <NSM <= -68.5 | 9 | 11.0% |
| -68.5 <NSM <= -45.7 | 15 | 18.3% |
| -45.7 <NSM <= -22.9 | 21 | 25.6% |
| -22.9 <NSM <= -10.0 | 11 | 13.4% |
| -10.0 <NSM <= 3.5 | 16 | 19.5% |
| 3.5 <NSM <= 7.0 | 4 | 4.9% |
| 7.0 <NSM <= 10.5 | 5 | 6.1% |
| 10.5 <NSM <= 11 | 1 | 1.2% |

Table 5 - Count statistic of the range, transect count of that range, and percentage occurring of that particular range off DSAS calculations from 1994-2019 LRR rates (left) and NSM distance (right).

|  |  |
| --- | --- |
| total number of transects | 82 |
| average distance | -29.1 |
| number of transects with negative distance | 67 |
| percent of all transects that have a negative distance | 81.70% |
| maximum negative distance | -91.71 |
| maximum negative distance transect ID | 12 |
| average of all negative distances | -36.83 |
| number of transects with positive distance | 15 |
| percent of all transects that have a positive distance | 18.29% |
| maximum positive distance | 10.91 |
| maximum positive distance transect ID | 44 |

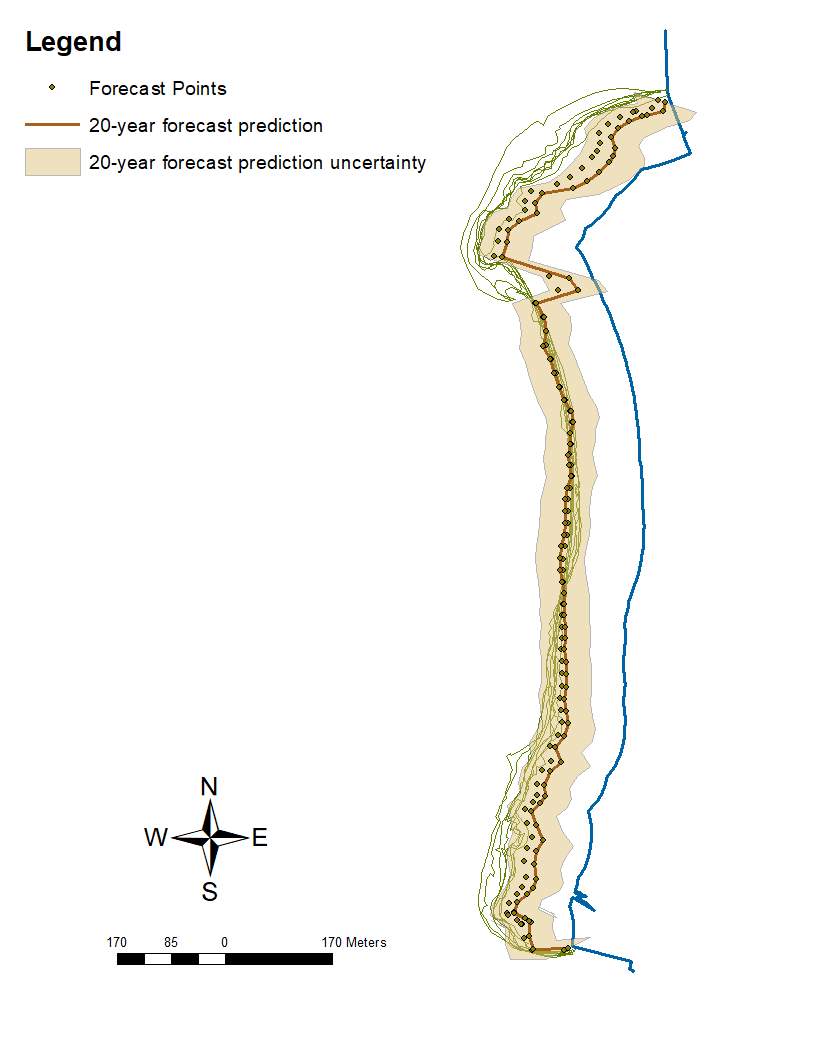
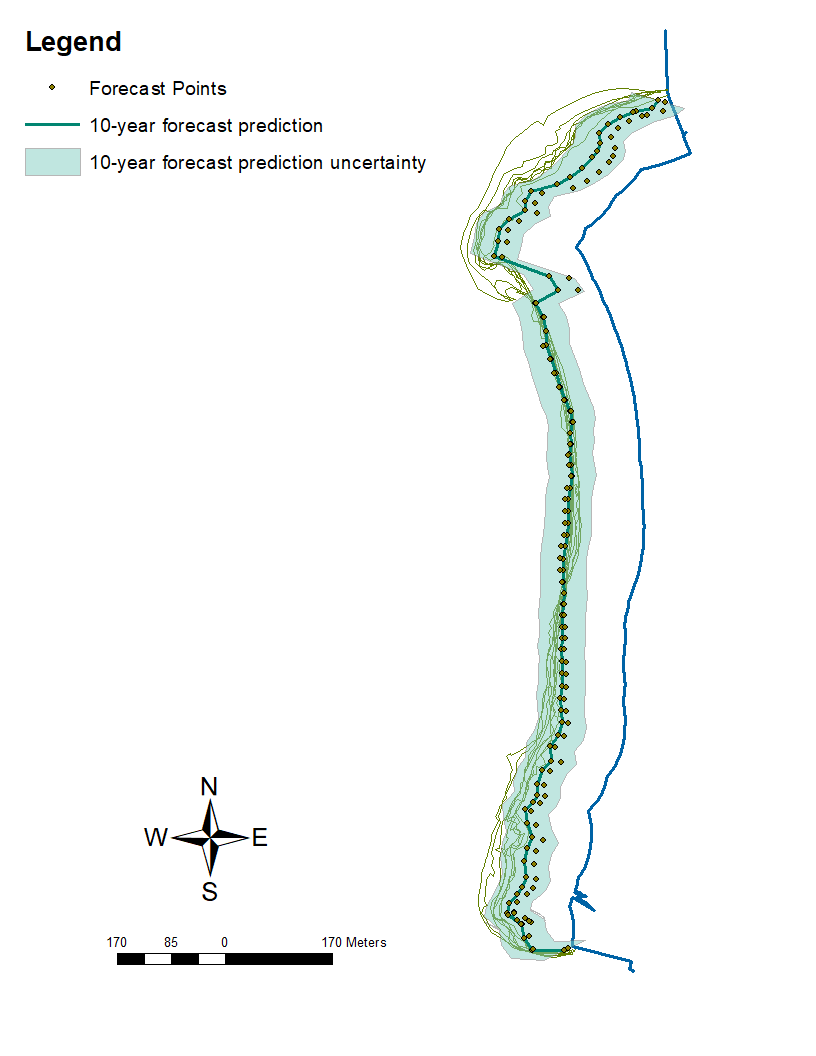
Table 6- Summary statistics calculated by DSAS, DISTANCE: NSM (Net Shoreline Movement, m)

|  |  |
| --- | --- |
| total number of transects | 82 |
| average rate | -0.95 |
| average of the confidence intervals associated with rates | 0.49 |
| reduced n (number of independent transects) | 900.00% |
| uncertainty of the average rate using reduced n | 0.17 |
| average rate with reduced n uncertainty | -0.95 +/- 0.17 |
| number of erosional transects | 63 |
| percent of all transects that are erosional | 76.83% |
| percent of all transects that have statistically significant erosion | 69.51% |
| maximum value erosion | -3.32 |
| maximum value erosion transect ID | 22 |
| average of all erosional rates | -1.33 |
| number of accretional transects | 19 |
| percent of all transects that are accretional | 23.17% |
| percent of all transects that have statistically significant accretion | 10.98% |
| maximum value accretion | 0.62 |
| maximum value accretion transect ID | 44 |
| average of all accretional rates | 0.31 |

Table 7- Summary statistics calculated by DSAS, RATE: LRR (Linear Regression Rate, m/yr)

***3.4 Shoreline analysis for 10 and 20-year prediction***

The DSAS calculations for future shoreline predictions are depicted in Figure 11. The 10-year prediction (left) demonstrates a uniformity of erosion particularity in the south and center of Deer Island. The north end of Deer Island has an acute area right before the shoreline bulge that is projected to be completely eroded by the 10-year prediction. The center of Deer Islands has a slight accretion area, but the majority of the 10-year projection is predicting that the west shoreline of Deer Island will erode. The 20-year prediction is very similar to the 10-year prediction model, but with more drastic erosion in the north end. The most eroded section of Deer Island (toward the north end) is getting close to the baseline.

Figure 11- Shoreline prediction for 10-year (left) and 20-year (right), including uncertainty. Shorelines are located on the west side of each panel. Baselines are located on the east side of each panel. Forecast points were created by DSAS to assist in the prediction model. The thicker black (left) and brown (right) lines depict the DSAS shoreline prediction. The lighter shaded region indicates the uncertainty of the predicted shoreline.

**4. Discussion and conclusion**

Results in this analysis suggest that more shoreline erosion occurred during the 1994- 2007 time frame compared to the later time frame of 2010-2019. The transects results depict more erosion in the NSM analysis (Figure 8, right) than compared to the time frame of 2010- 2019 analysis (Figure 9, right). It is curious for us to think about how and why this seemingly obvious drastic NSM erosion has occurred in the earlier time frame analysis. A year prior to the first imagery in the time series the Storm of the Century hit the Big Bend region. There is evidence during this storm event that sandy coasts were susceptible to shoreline erosion (Goodbred & Hine, 1993). Years of dramatic storm clusters in the Gulf of Mexico (1994- 2015, retreat erosion rate of − 5.49 ± 1.4 m/year) indicate significant morphological changes of the coast and could have possibly delayed natural beach recovery (Sankar et al., 2018).Despite the Storm of the Century happening prior to our shoreline analysis, an abrupt shoreline change due to an intense weather event coupled with SLR might have triggered an unbalance of natural erosion and accretion rates on Deer Island during the 1994- 2007 time frame, especially considering storm clusters encompassed this time frame.

Despite analyzing such brief time periods on a small shoreline, many changes have occurred. Note that in Figure 8, a small hook shoreline feature (on the north end) can be observed and is completely gone by the time period of Figure 9. Even erosion of small features such as that hook like shoreline can make an impact on the available habitats for animals. Many species depend on shorelines for food, nesting, and shelter (O'Connell et al., 2005).Shorebirds rely on shorelines for feeding habitats during migration in the winter months. Habitat loss, due to erosion, limits the availability of food and resources for these shorebirds, possibly resulting in increased competition. This increased competition may exclude individuals from a foraging site, increase mortality rates for these excluded shorebirds, and ultimately lead to limitations in numbers (Galbraith et al., 2005).The Big Bend region of Florida is already experiencing low shorebird species richness and population abundance, implying that an area already struggling with species biodiversity, despite the lack of human impact, will at least have negative shorebird impacts because of consistently eroding shorelines. During a high erosion storm event, many sandy-shore animals may also be washed up to shore, stranded up shore, or left to die due to exposure. Sandy- shore creatures naturally are able to survive storm events due to their defense mechanisms but are not always able to survive in the event of significant shoreline erosion (Brown & McLachlan, 2002). Whether shorelines erode slowly, but constantly, or in a storm event, extreme shoreline erosion negatively impacts animal species.

It is interesting to note, that although the overall shoreline experienced erosion, there is evidence accretion might have occurred in the middle of the shoreline during 2010-2019 (Figure 9). Table 6 notes that only a total of 10.91 meters was gained in accretion. Accretion for our study site can only come from intense meteorological events since there is a scant supply of sand being dispersed by the Suwannee River (Goodbred et al., 1998). However, it is unclear how much accretion can occur with the perpetuation of sea- level rise consistently stressing the sandy shoreline substrate. Sea-level rise has the second greatest effect on shoreline change on the east side of Florida, but has very similar effects on the west side. There is a possibility for Florida to provide beach nourishment to areas where erosion is evident, but with increasing sea-level rise competing, it may be difficult to evaluate shoreline change (Houston, 2015). Currently there is no schedule to provide beach nourishment to our study site.

During this study, one main source of error arises with the missing imagery years 2007- 2008 and 2011- 2012. If those missing years were available for analysis, it would provide a closer interpretation of the true erosion differences between the two 12 to 13-year time periods. Since our study site is uninhabited and remote, it is not surprising to see that NAIP is not contracted to fly over this area every year. Overall, we are able to see that erosion has occurred through the majority of the shoreline. Another source for possible errors are the digitization of each shoreline. Since each available imagery was used to digitize the years’ shoreline, the digitization of each shoreline might not be exact. However, the resolution of each image was at least 1-meter resolution, which may be considered “high” resolution in comparison to 30-meter resolution from Landsat 7 and 8 (Fisher et al., 2018), which Landsat imagery can also be used for analysis. The higher the resolution is, the more likely the digitized shorelines are accurate. Errors can arise within individual variability while digitizing.

The prediction models are based on a linear regression rate calculated by DSAS, termed Kalman filter (Kalman, 1960). The Kalman filter conducts an analysis to minimize the error between the observed and modeled shoreline position to develop the forecast where the rate and uncertainties are considered (Long & Plant, 2012). Our prediction models project that more shoreline erosion is to be expected (Figure 11). This be can concerning since currently our study site is not impacted by human development, however that may change in the future if people do decide to build residential or commercial properties. The prediction models may be used as a reliable source of information for land management directors who seek to protect uninhabited shorelines along the Big Bend.

This study has revealed brief historical trends of coastal evolution along an undeveloped sandy shoreline. This study may enhance the database of historical shoreline analysis in Florida. The shoreline statistics revealed elevated rates of erosion during the first-time frame 1994-2007. Storm and storm clusters may significantly impact barrier island morphology. Long term sea-level rise and sediment supply are considered major factors that stimulate shoreline erosion and/or accretion (Sankar et al., 2018),which may be contributing to the consistent erosion of our study site. This research has proven that sandy shorelines are susceptible to rates of high erosion that may lead to permanent shoreline loss.

**Bibliography**

1. Alarid-Escudero, F., Krijkamp, E. M., Pechlivanoglou, P., Jalal, H., Kao, S. Y. Z., Yang, A., & Enns, E. A. (2019). A Need for Change! A Coding Framework for Improving Transparency in Decision Modeling. PharmacoEconomics, 37(11), 1329–1339. https://doi.org/10.1007/s40273-019-00837-x
2. Archmiller, A. A., Johnson, A. D., Nolan, J., Edwards, M., Elliott, L. H., Ferguson, J. M., Iannarilli, F., Vélez, J., Vitense, K., Johnson, D. H., & Fieberg, J. (2020). Computational Reproducibility in The Wildlife Society’s Flagship Journals. Journal of Wildlife Management, 84(5), 1012–1017. <https://doi.org/10.1002/jwmg.21855>
3. Barchard, K. A., & Pace, L. A. (2011). Preventing human error: The impact of data entry methods on data accuracy and statistical results. Computers in Human Behavior, 27(5), 1834–1839. <https://doi.org/10.1016/j.chb.2011.04.004>
4. Barone, L., Williams, J., & Micklos, D. (2017). Unmet Needs for Analyzing Biological Big Data: A Survey of 704 NSF Principal Investigators. <https://doi.org/10.1101/108555>
5. Blischak, J. D., Davenport, E. R., & Wilson, G. (2016). A Quick Introduction to Version Control with Git and GitHub. PLoS Computational Biology, 12(1). https://doi.org/10.1371/journal.pcbi.1004668
6. Brown, A. C., & McLachlan, A. (2002). Sandy shore ecosystems and the threats facing them: Some predictions for the year 2025. *Environmental Conservation*, *29*(1), 62–77. <https://doi.org/10.1017/S037689290200005X>
7. Cazenave, A., & Cozannet, G. le. (2014). Sea level rise and its coastal impacts. *Earth’s Future*, *2*(2), 15–34. <https://doi.org/10.1002/2013ef000188>
8. Czech, B., Krausman, P. R., & Devers, P. K. (2000). Economic associations among causes of species endangerment in the United States: associations among causes of species endangerment in the United States reflect the integration of economic sectors, supporting the theory and evidence that economic growth proceeds at the competitive exclusion of nonhuman species in the aggregate. BioScience, 50(7), 593-601.
9. Desantis, L. R. G., Bhotika, S., Williams, K., & Putz, F. E. (2007). Sea-level rise and drought interactions accelerate forest decline on the Gulf Coast of Florida, USA. *Global Change Biology*, *13*(11), 2349–2360. <https://doi.org/10.1111/j.1365-2486.2007.01440.x>
10. Finkl, C. W., & Charlier, R. H. (2003). Sustainability of Subtropical Coastal Zones in Southeastern Florida: Challenges for Urbanized Coastal Environments Threatened by Development, Pollution, Water Supply, and Storm Hazards. *Journal of Coastal Research*, *19*(4), 934–943.
11. Fisher, J. R. B., Acosta, E. A., Dennedy-Frank, P. J., Kroeger, T., & Boucher, T. M. (2018). Impact of satellite imagery spatial resolution on land use classification accuracy and modeled water quality. *Remote Sensing in Ecology and Conservation*, *4*(2), 137–149. <https://doi.org/10.1002/rse2.61>
12. Galbraith, H., Jones, R., Park, R., & Herrod-Julius. (2005). *Global Climate Change and Sea Level Rise: Potential Losses of Intertidal Habitat for Shorebirds*. https://doi.org/10.1675/1524
13. Geselbracht, L., Freeman, K., Kelly, E., Gordon, D. R., & Putz, F. E. (2011). Retrospective and prospective model simulations of sea level rise impacts on Gulf of Mexico coastal marshes and forests in Waccasassa Bay, Florida. *Climatic Change*, *107*(1), 35–57. <https://doi.org/10.1007/s10584-011-0084-y>
14. Goodbred, S. L., Hine, A. C., & Stumpf, R. (1993). Sediment distribution patterns and the development of the marsh system rimming a shallow-water shelf embayment: Waccasassa Bay, Levy Co., FL. Geological Society of America, Abstracts with Programs;(United States), 25(CONF-9304188--).
15. Goodbred, S. L., Wright, E. E., & Hine, A. C. (1998). SEA-LEVEL CHANGE AND STORM-SURGE DEPOSITION IN A LATE HOLOCENE FLORIDA SALT MARSH. In *JOURNAL OF SEDIMENTARY RESEARCH* (Vol. 68, Issue 2). <http://pubs.geoscienceworld.org/sepm/jsedres/article-pdf/68/2/240/2812177/240.pdf>
16. Hampton, S. E., Strasser, C. A., Tewksbury, J. J., Gram, W. K., Budden, A. E., Batcheller, A. L., Duke, C. S., & Porter, J. H. (2013). Big data and the future of ecology. In Frontiers in Ecology and the Environment (Vol. 11, Issue 3, pp. 156–162). <https://doi.org/10.1890/120103>
17. Heidorn, B. (2008). Shedding Light on the Dark Data in the Long Tail of Science. In Library Trends (Vol. 57, Issue 2).
18. Holling, C. S. (1978). Adaptive environmental assessment and management. John Wiley & Sons.
19. Houston, J. R. (2015). Shoreline Response to Sea-Level Rise on the Southwest Coast of Florida. *Journal of Coastal Research*, *314*, 777–789. <https://doi.org/10.2112/jcoastres-d-14-00161.1>
20. Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.
21. Jones, Z. M. (2013). Git/GitHub, Transparency, and Legitimacy in Quantitative Research.
22. Lefcheck, J. S. (2016). piecewiseSEM: Piecewise structural equation modelling in r for ecology, evolution, and systematics. Methods in Ecology and Evolution, 7(5), 573–579. https://doi.org/10.1111/2041-210X.12512
23. Li, W., & Gong, P. (2016). Continuous monitoring of coastline dynamics in western Florida with a 30-year time series of Landsat imagery. *Remote Sensing of Environment*, *179*, 196–209. https://doi.org/10.1016/j.rse.2016.03.031
24. Long, J. W., & Plant, N. G. (2012). Extended Kalman Filter framework for forecasting shoreline evolution. *Geophysical Research Letters*, *39*(13). <https://doi.org/10.1029/2012GL052180>
25. Lowndes, J. S. S., Best, B. D., Scarborough, C., Afflerbach, J. C., Frazier, M. R., O’Hara, C. C., Jiang, N., & Halpern, B. S. (2017). Our path to better science in less time using open data science tools. Nature Ecology and Evolution, 1(6). <https://doi.org/10.1038/s41559-017-0160>
26. Martinelli, M., Guicciardi, S., Penna, P., Belardinelli, A., Croci, C., Domenichetti, F., Santojanni, A., & Sparnocchia, S. (2016). Evaluation of the oceanographic measurement accuracy of different commercial sensors to be used on fishing gears. Ocean Engineering, 111, 22–33. <https://doi.org/10.1016/j.oceaneng.2015.10.037>
27. Michener, W. K., & Jones, M. B. (2012). Ecoinformatics: Supporting ecology as a data-intensive science. In Trends in Ecology and Evolution (Vol. 27, Issue 2, pp. 85–93). Elsevier Ltd. <https://doi.org/10.1016/j.tree.2011.11.016>
28. Mislan, K. A. S., Heer, J. M., & White, E. P. (2016). Elevating the status of code in ecology. Trends in ecology & evolution, 31(1), 4-7.
29. McLachlan, A. (1990). Sandy beach ecosystems,[in:] Ecology of sandy shores.
30. McKinney, M. L. (2006). Urbanization as a major cause of biotic homogenization. *Biological Conservation*, *127*(3), 247–260. https://doi.org/10.1016/j.biocon.2005.09.005
31. Morton, R. A., Miller, T. L., & Moore, L. J. (2004). *National Assessment Of Shoreline Change: Part 1 Historical Shoreline Changes And Associated Coastal Land Loss Along The U.S. Gulf Of Mexico*.
32. Nelson, J. R., & Grubesic, T. H. (2018). The implications of oil exploration off the Gulf Coast of Florida. Journal of Marine Science and Engineering, 6(2). <https://doi.org/10.3390/jmse6020030>
33. Nie, M. A., & Schultz, C. A. (2012). Decision-Making Triggers in Adaptive Management. Conservation Biology, 26(6), 1137–1144. <https://doi.org/10.1111/j.1523-1739.2012.01915.x>
34. Noble, W. S. (2009). A quick guide to organizing computational biology projects. In PLoS Computational Biology (Vol. 5, Issue 7). https://doi.org/10.1371/journal.pcbi.1000424
35. Nordstrom, K. F., Jackson, N. L., Smith, D. R., & Weber, R. G. (2006). Transport of horseshoe crab eggs by waves and swash on an estuarine beach: Implications for foraging shorebirds. *Estuarine, Coastal and Shelf Science*, *70*(3), 438–448. https://doi.org/10.1016/j.ecss.2006.06.027
36. O'Connell, M. T., Franze, C. D., & Spalding, E. A. (2005). SPECIAL ISSUE NO. 44. SAVING AMERICA’S WETLAND: Strategies for Restoration of Louisiana’s Coastal Wetlands and Barrier Islands. In *Poirrier Source: Journal of Coastal Research*.
37. Pahl-Wostl, C. (2007). Transitions towards adaptive management of water facing climate and global change. Water Resources Management, 21(1), 49–62. <https://doi.org/10.1007/s11269-006-9040-4>
38. Perez-Riverol, Y., Gatto, L., Wang, R., Sachsenberg, T., Uszkoreit, J., Leprevost, F. da V., Fufezan, C., Ternent, T., Eglen, S. J., Katz, D. S., Pollard, T. J., Konovalov, A., Flight, R. M., Blin, K., & Vizcaíno, J. A. (2016). Ten Simple Rules for Taking Advantage of Git and GitHub. In PLoS Computational Biology (Vol. 12, Issue 7). Public Library of Science. https://doi.org/10.1371/journal.pcbi.1004947
39. Pinelli, J.-P., Roueche, D., Kijewski-Correa, T., Plaz, F., Prevatt, D., Zisis, I., Elawady, A., Haan, F., Pei, S., Gurley, K., Rasouli, A., Refan, M., Rhode-Barbarigos, L., & Moravej, M. (2018). *Overview of Damage Observed in Regional Construction during the Passage of Hurricane Irma over the State of Florida*.
40. Piwowar, H. A., Day, R. S., & Fridsma, D. B. (2007). Sharing detailed research data is associated with increased citation rate. PLoS ONE, 2(3). <https://doi.org/10.1371/journal.pone.0000308>
41. Prana, G. A. A., Treude, C., Thung, F., Atapattu, T., & Lo, D. (2019). Categorizing the Content of GitHub README Files. Empirical Software Engineering, 24(3), 1296–1327. <https://doi.org/10.1007/s10664-018-9660-3>
42. Prlić, A., & Procter, J. B. (2012). Ten Simple Rules for the Open Development of Scientific Software. In PLoS Computational Biology (Vol. 8, Issue 12). https://doi.org/10.1371/journal.pcbi.1002802
43. Purtlebaugh, C. H., & Allen, M. S. (2010). Relative Abundance, Growth, and Mortality of Five Age-0 Estuarine Fishes in Relation to Discharge of the Suwannee River, Florida. *Transactions of the American Fisheries Society*, *139*(4), 1233–1246. <https://doi.org/10.1577/t09-180.1>
44. Rahman, M. M., & Roy, C. K. (2014). An insight into the pull requests of GitHub. 11th Working Conference on Mining Software Repositories, MSR 2014 - Proceedings, 364–367. <https://doi.org/10.1145/2597073.2597121>
45. Ram, K. (2013). Git can facilitate greater reproducibility and increased transparency in science (Vol. 8). http://www.scfbm.org/content/8/1/7
46. Sankar, R. D., Donoghue, J. F., & Kish, S. A. (2018). Mapping Shoreline Variability of Two Barrier Island Segments Along the Florida Coast. *Estuaries and Coasts*, *41*(8), 2191–2211. https://doi.org/10.1007/s12237-018-0426-3
47. Sassaman, K. E., Wallis, N. J., McFadden, P. S., Mahar, G. J., Jenkins, J. A., Donop, M. C., Monés, M. P., Palmiotto, A., Boucher, A., Goodwin, J. M., & Oliveira, C. I. (2017). Keeping Pace With Rising Sea: The First 6 Years of the Lower Suwannee Archaeological Survey, Gulf Coastal Florida. *Journal of Island and Coastal Archaeology*, *12*(2), 173–199. https://doi.org/10.1080/15564894.2016.1163758
48. Thomas, K., Kvitek, R. G., & Bretz, C. (2002). Effects of human activity on the foraging behavior of sanderlings Calidris alba. *Biological Conservation*, *109*(1), 67–71. <https://doi.org/10.1016/S0006-3207(02)00137-4>
49. Tompkins, E. L., & Adger, W. N. (2004). Does Adaptive Management of Natural Resources Enhance Resilience to Climate Change? And Society, 9(2). https://doi.org/10.2307/26267677
50. U.S. Geological Survey, 1955, USGS 1:24000-scale Quadrangle for Cedar Key, FL 1955: U.S. Geological Survey. (https://www.sciencebase.gov/catalog/item/5a8a3ffbe4b00f54eb3ec75e)
51. Vitale, N. E. (2019). *HABITAT CHANGE, PREDATORS, AND DISTURBANCE: FACTORS INFLUENCING PRODUCTIVITY OF AMERICAN OYSTERCATCHERS (Haematopus palliatus) NESTING IN FLORIDA’S BIG BEND*.
52. Walters, C. J. (1986). Adaptive management of renewable resources. Macmillan Publishers Ltd.
53. Weimer, T., Williams, B. K., Szaro, R. C., Shapiro, C. D., Adamcik, R., Boatman, M., Bransom, S., Casterson, J., Fay, J., Florence, S., Growitz, D., Hermans, C., Johnson, F. A., Kendall, J., Kubly, D., Mayer, M., Moyer, S., Pattison, M., Peterson, R., … Rodriguez, V. (2007). Adaptive Management The U.S. Department of the Interior Technical Guide Lead Authors Other Contributors Book Design.
54. White, E., Baldridge, E., Brym, Z., Locey, K., McGlinn, D., & Supp, S. (2013). Nine simple ways to make it easier to (re)use your data. Ideas in Ecology and Evolution, 6(2). https://doi.org/10.4033/iee.2013.6b.6.f
55. Withers, K. (2002). Shorebird use of coastal wetland and barrier island habitat in the Gulf of Mexico. *TheScientificWorldJournal*, *2*, 514–536. <https://doi.org/10.1100/tsw.2002.112>
56. Wright, E. E., Hine, A. C., Goodbred, S. L., & Locker, S. D. (2005). The Effect of Sea-Level and Climate Change on the Development of a Mixed Siliciclastic-Carbonate, Deltaic Coastline: Suwannee River, Florida, U.S.A. *Journal of Sedimentary Research*, *75*(4), 621–635. <https://doi.org/10.2110/jsr.2005.051>
57. Yenni, G. M., Christensen, E. M., Bledsoe, E. K., Supp, S. R., Diaz, R. M., White, E. P., & Morgan Ernest, S. K. (2018). Developing a modern data workflow for living data. https://doi.org/10.1101/344804
58. Yu, K., Hu, C., Muller-Karger, F. E., Lu, D., & Soto, I. (2011). Shoreline changes in west-central Florida between 1987 and 2008 from Landsat observations. *International Journal of Remote Sensing*, *32*(23), 8299–8313. <https://doi.org/10.1080/01431161.2010.535045>
59. Zedler, J. B. (2017). What’s New in Adaptive Management and Restoration of Coasts and Estuaries? Estuaries and Coasts, 40(1). https://doi.org/10.1007/s12237-016-0162-5
60. Zhang, K., Douglas, B. C., & Leatherman, S. P. (2004). *GLOBAL WARMING AND COASTAL EROSION*. www.ipcc.ch