

RESEARCH ARTICLE



Enhancing Data Analytics and Visualization Support in CSViewer for Analysts – Version 1.1: Access to an Integrative Database and Knowledge Model of Cayo Santiago Rhesus Macaques

Martin Q. Zhao^{1,*}, Cooper Novak¹, Ethan R. Widener¹, Raj A. Patel¹, Rui Gong² , George Francis³ and Qian Wang³

¹Department of Computer Science, Mercer University, USA

²Department of Informatics and Mathematics, Mercer University, USA

³Department of Biomedical Sciences, Texas A&M University College of Dentistry, USA

Abstract: The CSViewer for Analysts app is designated to provide a myriad of graphical user interfaces and analytical tools for researchers to study the valuable datasets collated from the Cayo Santiago Rhesus macaque colony, which has been raised on the “monkey island” of Puerto Rico since 1938. It is a part of the effort to integrate decades of accumulated genealogy and demographic information with newly collected osteology data using the CS derived skeletal sets. When Version 1.0 was first shared among the collaborating teams in early 2023, CSViewer enabled its users to browse the CS population in the forms of matrilineal and patrilineal trees, as well as to visualize skull dimension data and photos when they are available. After choosing a Java-based dataframe library, a planned redesign has been implemented into Version 1.1, which supports data selection for animal subjects, bone measures to include, as well as facilitates more streamlined data management within the system. Various data analytics and visualization features have been added in v1.1, and a workshop was held during AABA 2024 to introduce CSViewer to interested users. Project-related materials have also been used to provide teaching materials in related undergraduate classes at the Computer Science Department of Mercer University to engage students in this NSF-funded project.

Keywords: data management for analytics, application design and development, biomedical informatics, custom regression and visualization, problem-based learning

1. Introduction

The Cayo Santiago Rhesus macaque population has provided abundant data resources for multiple research fields since its founding in 1938 on a small island, Cayo Santiago off Puerto Rico. In addition to genetic, environmental, and age-related information about these free-ranging monkeys that have been well kept by the Caribbean Primate Research Center (CPRC), thousands of skeletal sets with CS genetics are also curated at CPRC museum, which provides a rare collection that can be cross-referenced to each individual’s sex, age, and pedigree [1–8].

However, there has been no direct access to these valuable data sources for the research communities. Datasets can be requested from CPRC, which maintains the CS population and derived data through NIH grants [9–12]. In 2019, an interdisciplinary team started a collaborative project funded by NSF, which aims at (1) collecting bone (in terms of dimension, mineral density, and pathological tracing); (2) combining the new osteology data with existing genealogy and social grouping data into a comprehensive database; and developing a software application for researchers to search and

select datasets of interest from the database; and (3) feed into their hypothesis testing on genealogical and pathological studies.

The team brings together expertise in anthropology, biology, biomedical sciences, and computer sciences. In this collaborative project, bone dimensions, bone mineral density (BMD), body mass, tooth eruption, and observable disease conditions of the rhesus monkey have been incorporated into the database with details on each monkey’s sex, birth and death dates, parentage information, and social rank (when available). A software application code-named CSViewer for Analysts started to take its form in early 2023, based on a conceptual design as described by Zhao et al. [13].

It is worth mentioning that the R&D needs from this project have provided unique teaching and learning opportunities for collaborating institutions. This is regarded as one of the Broad Impacts of this project when it was applied for NSF support. Specifically, MQZ has used problems from this NSF-funded project in his Software Engineering, Database Systems, and Data Science Applications courses [14]. This paper will introduce in certain details how students have been involved in research and development (R&D) activities leading towards the CSViewer for Analysts application.

*Corresponding author: Martin Q. Zhao, Department of Computer Science, Mercer University, USA. Email: zhao_mq@mercer.edu

The long-term goal for CSViewer is to be a searchable and computer-interoperable Knowledge Model based on the integrative Cayo Santiago rhesus health database. Its initial version, v1.0, completed in early 2023, was intended to be a proof-of-concept, for demonstrating its potential to collaborating teams and other researchers using the CS datasets. A redesign was planned since then to adopt a dataframe-based Java library in CSViewer, streamlining data selection and analytics. Major enhancements have been made to the new version 1.1 based on user feedback. A workshop was held (on March 20, 2024) during the 92nd annual conference of the American Association of Biological Anthropologists (AABA 2024 in Los Angeles). Direct feedback on desirable features and formats received from an audience of researchers and educators has been included in the app (Build 1.1.1) during summer of 2024.

2. System Development Methodologies

CSViewer is developed using Java-based technologies following an incremental approach. It employs standard Java features, such as Swing, JCF (Java Collections Framework), and JDBC (Java database connectivity) to facilitate user-friendly interfaces for navigation and data manipulation tasks. In addition, third-party libraries are used to support more sophisticated data analytics and visualization use cases.

Development efforts for database and the graphical user interfaces (GUI) that led to CSViewer Version 1.0 were reported in Zhao et al. [13, 14]. A newer version (v1.1) that utilizes a lightweight Java-based dataframe library to streamline data manipulation and sharing between system modules has been introduced in Zhao et al. [15], with typical use cases (as implemented in Build 1.1.0) described.

CSViewer is the first application of its kind that intends to provide sophisticated GUI interfaces for researchers to examine the abundant data and necessary analytical tools to gain insight from the data. The process is exploratory in nature. One big challenge is to make the app to be functional from the very beginning, while allowing for continuous extension with new data sources being added and new requirements from the users calling for new use cases and up-to-date techniques (such as machine learning, AI, image processing, and visualization).

This section will summarize data and system modeling, with certain details regarding how a flexible architectural design and an incremental development process are employed to provide a software solution for an ever-growing feature set. Section 3 will address aspects relevant to the

redesign and extended data analytics and visualization support as reflected in the evolution of CSViewer versions. Section 4 will conclude recent efforts and suggest future directions.

2.1. Data modeling and database management

As described by Zhao et al. [13], a relational data model was originally proposed for the integrative database to manage subject genealogy information of the rhesus families maintained by the CPRC, plus health data obtained in this project (including photos, measurements, and observation data). A relational design like a human health database is used, which shows great extensibility. More tables need to be added to keep track of valid users, access control, user activities, etc., at a later stage.

A conceptual data model for the integrative database is shown in Figure 1, to summarize various data entity types and relationships among them.

Various data sources as integrated into the database are described as follows:

- 1) Each of the 10949 animals was born to a family traced back to one of the 77 founding mothers, as recorded in the census datasets from 2013 and 2020.
- 2) Each animal has a known (behavioral) mother. DNA tracings for dam (mother) and sire (father) were only available since 1975, for some 4800 animals.
- 3) Each founder has an M-code in the format of M1###, with # representing a digit. A Unicode (L- ####) is assigned to each animal to replace the tattoo (cast to the animal as a “personal” identifier) in publications to be compliant with HIPAA requirements.
- 4) Animals are assigned to one of the social groups, which evolved over the past 86 years. An animal may be transferred to one or more groups other than its “birth group”.
- 5) Skeletal sets of more than 2300 CS monkeys were cataloged at a “museum” at the Caribbean Primate Research Center (CPRC). Bone dimension and mineral density data have been collected during this collaborative research project.

Details about data sources and colony-wide statistical results are reported in other papers currently under review.

2.2. System architecture of CSViewer app

A layered architecture, as shown in Figure 2, is adopted for the CSViewer application. It consists of multiple layers for user

Figure 1
Conceptual data model for the Cayo Santiago Rhesus colony database

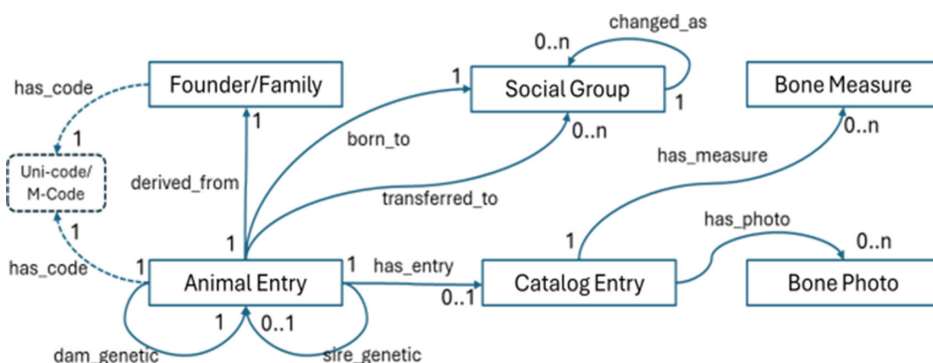
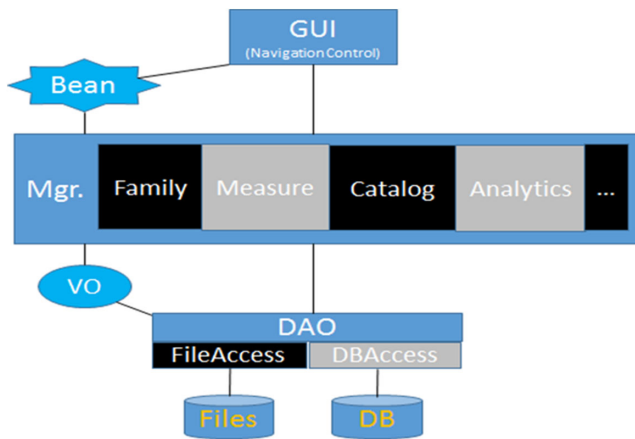


Figure 2
Layered architecture with data accessibility,
business managers, and GUI



interactions, high-level “business” processes (such as generating organizing family trees and analytical results), and accessing data from the database. For versions 1.0 and 1.1 of the CSViewer system, CSV files are used to store the more stable data structures (e.g., matrilineal and patrilineal family trees) that can be bundled with executables for users to download.

As illustrated in Figure 2, the middle layer is divided into partitions, each of which manages a different kind of data (family, measure, image, etc.) and related processing tasks. This design helps make the system flexible such that processing functions can be implemented separately. The managers are responsible for accessing data using the relevant data accessors and organizers (DAOs) and creating value objects (VOs, such as animal or measure objects) that map to the relational data entries. Information prepared by the corresponding manager will then be sent to the related GUI component that displays it to the end user.

The layered architecture with partitions each focuses on one perspective of the functions allows for useful features to be added to the system gradually through (sub)versions. Partitions facilitating family trees and essential analytical results were chosen as the highest priority and hence implemented in v1.0. Partitions for Catalog and Image are deferred to future versions, as described in Section 4.

2.3. Incremental development process

The CSViewer project uses an iterative approach to prioritize core functionalities first. This allows for more detailed design and development of additional features in later stages based on user feedback and evolving needs.

2.3.1. History and purposes of the (sub)versions

Pilot studies and prototyping of modules usable in the intended software application started before 2019 when the proposal to NSF was submitted. Early efforts intended to facilitate essential functions for anthropological studies, including interactive family tree panels, a dialog box for image and text data entry, and panels for showing data analytics results [13]. Test databases were also implemented and hosted on SQL Server instances (in local network and cloud instances).

With the help from a student team in MQZ’s Software Engineering class in Fall of 2022, the main window for CSViewer

(Version 0) was integrated from four individual working modules. Version 1.0 was then completed in early 2023, after adding appropriate menus and a tabbed structure for the main content panel. Additional modules and features have been gradually integrated into subsequent subversions of CSViewer after passing functional tests.

Version 1.0 was mostly meant to be a proof-of-concept, by providing a helpful tool for the collaborating teams to explore the CS families and look for vital data of individual animals and related measure data as available. Valuable feedback from user experiences had been utilized in prioritizing development efforts for Version 1.1.

Version 1.1 was targeted to be a functional application that empowers researchers to (1) create a project (folder) and manage related data and result files; (2) explore the colony-wide statistics (i.e., several “big picture” charts) to look for promising data for a specific study; (3) select animal sample and bone measure data; and (4) carry out analytics and visualization tasks as supported in the CSViewer app. An initial build (1.1.0) was demonstrated at a workshop held during 92nd annual conference of the American Association of Biological Anthropologists (AABA) in March 2024 in Los Angeles.

Details about use cases of v1.1 and visuals revealing a holistic view of the CS population are reported in other papers currently under review. In this paper, we will focus on data analytics and visualization features recently included in Build 1.1.1, which are highlighted by a custom regression model as proposed for fitting skeletal development data for the CS animals. Interesting system model transformation and codebase refactoring practices will be described in Section 3.

In the next two subsections, we will further elaborate on the design of the GUI, MGR, and DAO layers and the collaborations among the layers, in a context shared in v1.0 and v1.1. The selection and inclusion of a dataframe library and its impacts on design model transformations are discussed in the next section.

2.3.2. Graphical user interfaces

The CSViewer window (Figure 3) employs a commonly used menu structure to accommodate various visualization and analytical functions. Information in various aspects that can be displayed in four major panels is described as follows.

The main content panel is used to display the main results of the current task. As shown in Figure 3, matrilineal family trees are shown in the main content panel. More tabs will be added to the main content panel when other tasks (such as animal set selection, bone measure selection, and visualization) are carried out next. The tabbed structure can hold any number of tabs and allow for easy navigation between the tabs.

A summary pane is used below the tabbed pane structure to display animal summary information about an animal selected in the family tree tab. Additional information (when available) will be displayed to the panels on the right, including bone measurement in table format, and/or photos relating to the selected subject.

2.3.3. Data management and manipulation

Traditionally, custom “bean” and “value” classes are needed to encapsulate relevant data, as illustrated in Figure 2. In v1.0 design, custom Java classes (AnimalFamilyInfo and MeasureBean) were used to encapsulate animal and bone measure data. Designated “manager” classes (such as SearchMgr) are responsible for accessing data in the database with the help of the corresponding data accessor and organizer objects (such as AnimalFamilyInfo

Figure 3
CSViewer v1.1: main window layout

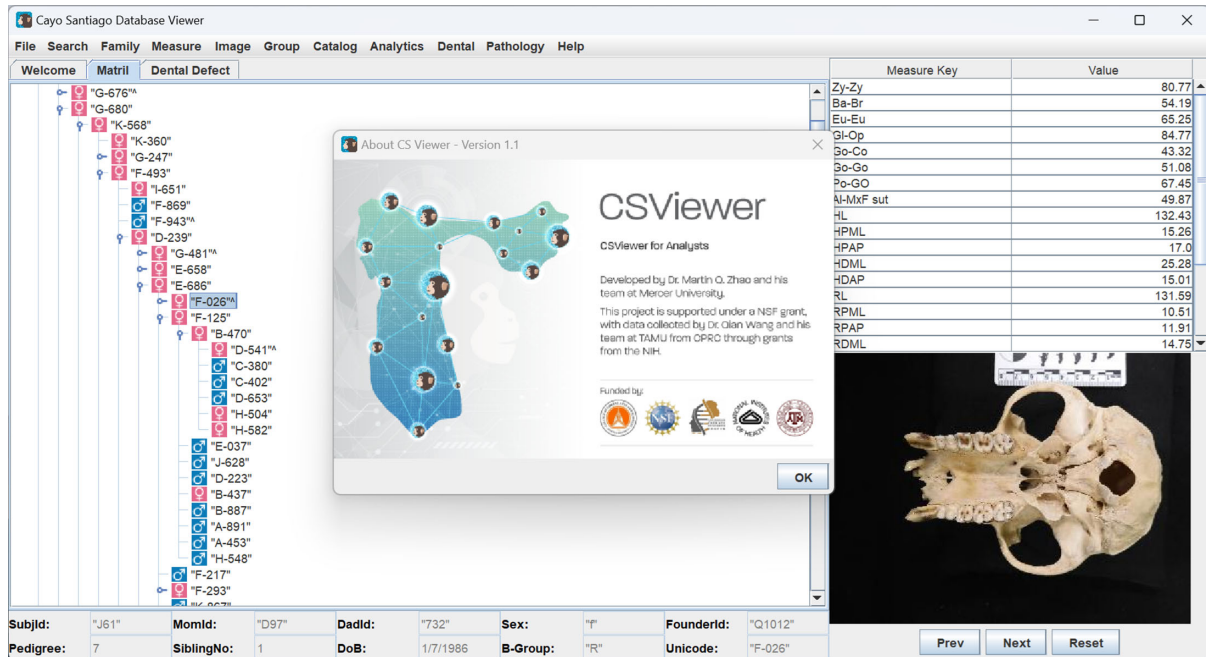
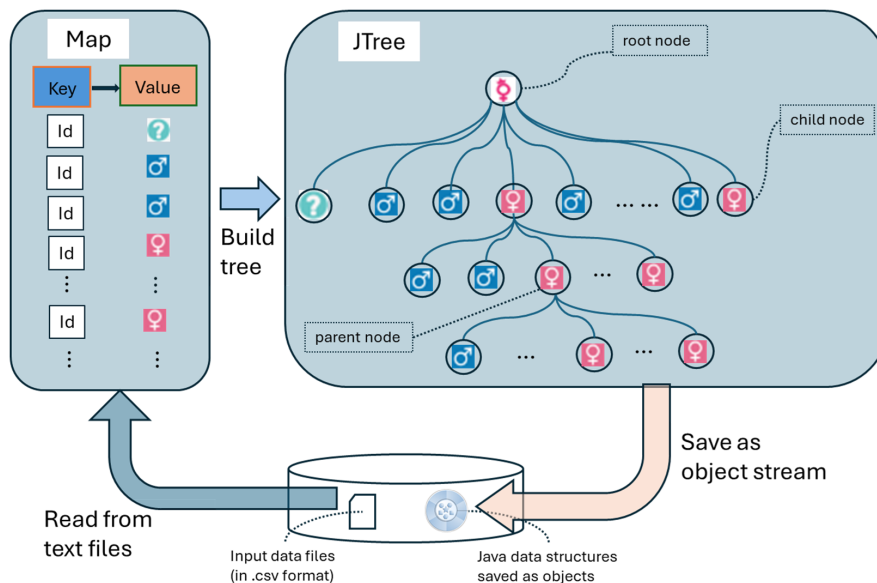


Figure 4
JTree used for organizing family tree structures

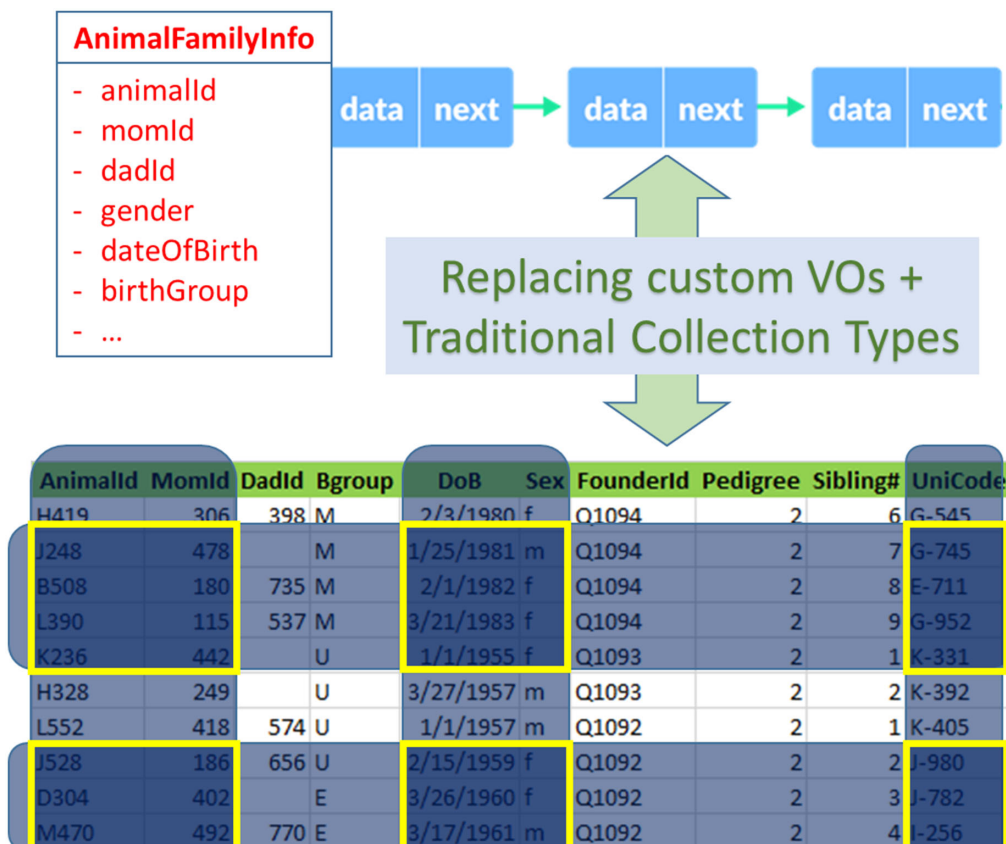


and MeasureBean). The same AnimalFamilyInfo class is also used by the FamilyTreeMgr class to generate and organize matrilineal and patrilineal family tree structures, with the help of its DAO classes. To minimize custom design efforts and streamline data manipulation and sharing among system modules, a redesign by adopting a Java dataframe library was planned while v1.0 was under development. Section 3 will compare the “old Java objects” (OJO) style and the dataframe-based style in certain detail.

The manager classes use collection classes in the standard Java collections framework (JCF) to store data and prepare bean objects to feed to various content panels using standard Java Swing classes, such as JTree and JTable. The JTree view and related data model, a Java-specific implementation of an n-tree data structure, plays a key role in rendering family trees (as shown in Figure 4).

The JTree class helps organize animal data using a hierarchical structure. Matrilineal lines (family lines traced through mothers) are

Figure 5
Comparison of design using “old Java object” (OJO) and DataFrame



represented as expandable trees, with founders as root nodes. Their offspring are displayed beneath them. Mothers within these lines can be further expanded to show their children, allowing for complete navigation of the dataset and exploration of relationships.

Patrilineal trees (family lines traced through fathers) are built based on each animal’s father and traced back to the earliest identified male in each lineage (confirmed by DNA screening). To avoid duplication of information, the construction process utilizes animal data from the matrilineal tree input file.

The project also employs JCF map types to track connections between animals and their corresponding catalog entries. These links can then be used to access associated measurement and image data, if available.

3. Enhanced Analytics Features in V1.1

CSVViewer was updated to Build 1.1.0, when it was demonstrated to a mixed audience of seasoned researchers, graduate students, and educators in a workshop during AABA 2024. Functions made available in that version, such as browsing the matrilineal and patrilineal family trees, selecting interested animals and their bone measures, data analytics and visualization features, as well as project management, are detailed in a series of papers submitted to a special issue of American Journal of Primatology (AJP).

The adoption of a Java dataframe library led to a smooth incorporation of unified data management in CSVViewer v1.1, which supports essential data retrieval and selection, as well as subsequent analytics and visualization features in this section. The

processes of selecting the Joinery’s DataFrame API will be described, and the impacts on structuring the “Manager” and “DAO” classes for the Analytics partition are presented. As a result, enhanced analytics features, such as slicing-and-dicing the data (as shown in Figure 5) and custom regression, will also be demonstrated.

3.1. Selection of a Java dataframe library for v1.1

A redesign to streamline data analytic support in v1.1 was set off once v1.0 started to take its shape. One key factor is to find a third-party Java library that supports dataframe-like operations as commonly used in data science tasks. Such APIs (application programming interfaces) allow software developers to organize data into tables and manipulate those tables in ways that core Java does not support. Given the vast amount of data involved in this project, as well as the myriads of categories the data falls under which often intersect and need to be cross-referenced with one another, an underlying table structure is preferred. Dataframe-like operations will allow the data to be placed into large tables that can then be cut down into relevant subsets without designing additional custom Java classes.

Our early research was focused on Tablesaw, which provides the most complete feature sets and is compliant with many data formats and extended visualization features (as in XChart). Tablesaw was eventually not selected because it depends on about two dozen additional third-party APIs, which are not necessary for CSVViewer and help add burdens to deployment. Instead, the Joinery API [16] is selected, which is simple and provides a

DataFrame class with clean interfaces for row- or column-wise selection, table join, aggregation, etc. Components from the Apache Commons project [17] (such as those for CSV, Math, etc.) can be used for accessing CSV files and conducting statistical modeling.

3.2. Refactoring and transforming the analytics partition

An agile process is adopted in the development of CSViewer, which emphasizes using the simplest design for the current build and relies on creating “spike” solutions [18] to test with new features. In v1.0, the Analytics “partition” was implemented with only one AnalyticsMgr class, which interacts with other “manager” classes and generates xy-plot in limited ways. During the redesign, several test classes were created to facilitate a “spike” solution so that the intended chart types and regression modeling options can be supported.

Both refactoring (such as reassigning existing functions to a new class) and model transformation (i.e., restructuring with new classes to provide additional functions), as described in Bruegge and Dutoit [19], were performed in implementing v1.1. The AnalyticsMgr class was extended to a module (or partition) with classes in a gui.analytics package (for generating charts) and a mgr.analytics package (for analytics operations such as regression). The analytics manager does not need its own DAO classes, but requests data from other managers that are in charge. In the scenario of selecting bone measure and displaying the data versus animal’s estimated age at death (EAAD), data will be retrieved through the MeasureMgr class in the mgr. measure package.

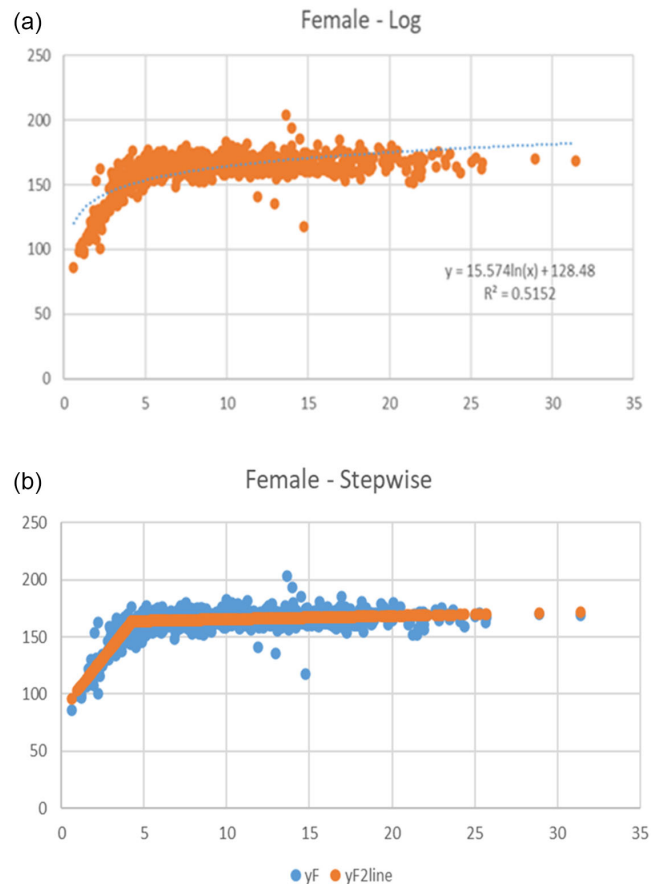
3.3. Extending v1.1 with custom regression and visualization

This subsection presents an example of how user feedback was used in including new features into a new version. Since v1.0 was provided to the collaborating teams to use, GF (who collected the bone measure and imagery data) provided many requirements about analytic features to include and specific ways to visualize results. Supporting line fitting features in xy-plot of bone measure vs. EAAD was requested from the beginning; during the AABA 2024 workshop, additional requests for selecting animal subjects by attributes such as sex and family were also added to the wish list.

While working on a “spike” for integrating the new regression module into CSViewer, a stepwise model was found to be more effective than the “one-piece” alternatives (such as linear and log models) commonly used in fitting due to simplicity. Additional research efforts led to a better understanding of the nature of the skeletal development patterns, while design of the related software modules will be described next.

As shown in Figure 6, bone dimensions tend to increase rapidly at an early age and then plateau at the “end of ontogeny”. Conventionally, a log model [20, 21] is used to fit the growth pattern (Figure 6(a)), while a stepwise model (with two straight line segments as shown in Figure 6(b)) appears to go through the middle of the band. Classes that are available in the JFreeChart [22] library can only handle linear and polynomial models. To support this new feature, the Apache Commons Math library [17] is included to define the stepped function. A StepwiseFitter class is implemented (in the mgr.analytics package) as a subclass of the AbstractCurveFitter class provided in the

Figure 6
Comparison of regression models used for bone dimension fitting. (a) Trendline with a log model and (b) trendline with a stepwise model



Apache API, which defines a stepwise function with two different growth rates before and after the age of maturity. Test results coincided well with those calculated using *Microsoft Excel*'s solver.

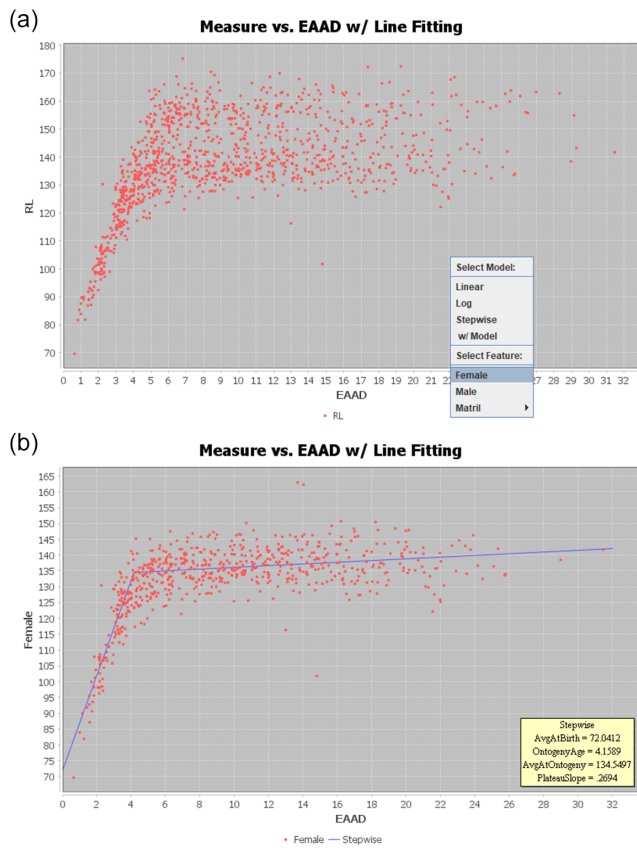
A CsXyPlotChartWithLineFitting class is added to the gui.analytics package to generate the chart, along with a LineFittingMenu class to create a popup menu for the user to select modeling options.

As shown in Figure 7, a user can select from linear, log, and stepwise models from the popup menu. With a closer look into the dots shown in Figure 7(a), one can recognize two “strands” in the plateau: a lower one for females and an upper one for males. It is requested by end user to differentiate sex (or some other attributes, such as matrilineal lineage) and fit each subset separately.

With dots for male subjects removed (as shown in Figure 7(b)), the stepwise model identifies the age at maturity for females as 4.27 years (versus 5.50 years for males) regarding femur length (FL) measures. The parameters of the selected regression model can be displayed by choosing the “w/ Model” option on the popup menu. Multiple models can be selected and compared in the same chart.

Thorough parametric studies are underway to compare the stepwise model and its alternatives for of the CS population. Initial results revealed promising potential for a better understanding of skeletal development of CS Rhesus and similar primates. Regression support as described above is now included in build 1.1.1. Interested readers can visit CSViewer.org for updates.

Figure 7
Screenshots of the newly added attribute selection and regression features. (a) Popup menu for regression and (b) trend line and parameters



Related R&D topics of this project have been used to generate course materials in related classes (including software engineering, database systems, data science applications, and programming) offered by the Computer Science Department at Mercer University. Undergraduate students in those courses have been involved through lectures, hands-on exercises, and contributing to building parts of the system. Teams of students were involved in analyzing the client’s needs, designing a solution using third-party APIs, and applying what they learned into building real-world software applications.

The long-term goal for CSViewer is to be a searchable and computer-interoperable Knowledge Model based on the integrative Cayo Santiago rhesus health database. On the one hand, capabilities for data accessibility and cross-referencing will be enhanced. Convenient functions for skeletal catalog lookup, derived imagery data indexing and processing, pathological annotation information management will be developed. On the other hand, support for data analytics and hypothesis-based research will also be enhanced.

One research topic that emerged from this version upgrade effort is to thoroughly explore skeletal development patterns of the CS rhesus population. The new stepwise model will be compared with other commonly used alternatives. It will then be applied to various bone pieces measured and construct a whole picture of CS rhesus skeletal development. The CSViewer app can be generalized to process similar data from other public sources and to help researchers explore family lineage and developmental/pathological trends. In that sense, CSViewer will become a useful tool for all Common Species. A CSViewer for Education edition will be provided in which a sequential Unicode is used for individual identities in lieu of tattoo numbers and other recognizable attributes per CPRC HIPAA regulations. The Education edition can facilitate numerous bioscience courses offered in colleges and high schools (such as Biostatistics, Anthropology, etc.) to provide real-world datasets and tools for analytics and visualization.

4. Conclusions and Future Work

CSViewer for Analysts has been developed to provide a designated app to access the valuable database of the Cayo Santiago rhesus colony. Now in Version 1.1, CSViewer extended its functions from various family trees browsing in its initial version, to providing animal set and bone measure data selection, as well as enhanced analytics and visualization. Build 1.1.0 has been demonstrated during a recent workshop at AABA 2024 to a small number of users. A new build 1.1.1 with features based on feedback from the workshop is now ready for download by interested researchers. Several visuals (in the forms of heatmap, ridge chart, and chord chart) are included in its menu structure to provide retrospective views of the CS population via multiple dimensions (family lineage, social group, and population dynamics).

Extensions to the analytics and visualization modules are ongoing, with more chart types (column charts for categorical data and histograms for numeric data) to be included in future subversions. Various regression models (linear, logarithm, stepwise, etc.) as discussed in this paper, paired up with finetuned sample selection by gender, family, and/or pedigree have now been included in Build 1.1.1. A CSViewer.org website is to be put online, to provide a portal for app download and access related resources such as user guidelines and publications.

Acknowledgements

Dr. Melween I. Martinez Rodriguez, Dr. Carlos Sariol, Mr. Bonn V Liong Aure, Dr. Angelina Ruiz-Lambides, Dr. Alyssa Arre, Ms. Nahiri Rivera Barreto, and other CPRC staff members for their support and help. Special thanks are extended to the following CPRC staff members: Edgar Davila, Myrna Reyes, and Alberto Clemente for performing skeletal preparations; and Mrs. Terry B. Kensler, Dr. Matthew Kessler, Dr. Rich Rawlins, Dr. Jean Turnquist, Mrs. Myriam Viñales, the late Dr. Nancy Hong, Dr. John Cant, and Dr. Donald Dunbar for their curatorial work or contributions in maintaining the skeletal collection. We also gratefully acknowledge the CPRC-NYU Collection and staff at NYU. The collection is maintained by: NYU Center for the Study of Human Origins, NYU Anthropology, and NYU Arts & Science. Its development and curation were supported by grants from the National Science Foundation BCS-1648676 (to Antón, Higham and Williams), BCS-1800558 (to Higham and Antón), BCS-1754024 (to Higham, Antón, and Williams), and the L.S.B. Leakey Foundation (to Higham, Antón, and Williams) and a supplement to R01-AG060931 (to Snyder-Mackler, Brent, and Higham). We are also grateful to Drs. Debbie Guatelli-Steinberg and Luci Kohn for describing use cases and providing feedback. Andres Boulosa, Adrian Faircloth, Jesus Rijo, and Zaina

Khutliwala worked on the team together with E.R.W. that contributed to CSViewer v1.0; while Philomina N. Ekezie, Katherine Duke, Andrew Ellington, and Alexander Gass contributed to v1.1. Rajwol Chapagain and Prakriti Sapkota helped in graphics design, testing, and maintaining download bundles. Dr. Robert Allen, Dr. Andy Digh, Jesse Sowell, and the students at Mercer University's Computer Science Department provided their support and contributions. Special thanks go to Pegasus Vertex, Inc. in Houston, TX for providing graphics used in Version 1.0 and suggestions for Version 1.1.

Funding Support

The Cayo Santiago macaque colony and derived skeletal collection housed at University of Puerto Rico Medical Center which are supported through the Cayo Santiago Primate Research Center by National Institutes of Health NIH contracts NIH 5P40OD012217. This project is supported by NSF grants to M.Q.Z. and Q.W. (#1926402, #1926601).

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data are available on request from the corresponding author upon reasonable request.

Author Contribution Statement

Martin Q. Zhao: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Project administration, Funding acquisition. **Cooper Novak:** Software, Validation, Writing – review & editing, Visualization. **Ethan R. Widener:** Software, Writing – original draft. **Raj A. Patel:** Validation, Writing – review & editing. **Rui Gong:** Methodology, Formal analysis, Writing – review & editing, Visualization. **George Francis:** Methodology, Validation, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Qian Wang:** Conceptualization, Resources, Project administration, Writing – original draft, Writing – review & editing, Funding acquisition.

References

- [1] Sade, D. S. (1986). *Basic demographic observations on free-ranging rhesus monkeys*. USA: Human Relations Area File.
- [2] Rawlins, R. G., & Kessler, M. J. (1986). *The Cayo Santiago macaques: History, behavior, and biology*. USA: SUNY Press.
- [3] Kessler, M. J., & Rawlins, R. G. (2016). A 75-year pictorial history of the Cayo Santiago rhesus monkey colony. *American Journal of Primatology*, 78(1), 6–43. <https://doi.org/10.1002/ajp.22381>
- [4] Widdig, A., Kessler, M. J., Bercovitch, F. B., Berard, J. D., Duggleby, C., Nürnberg, P., . . . , & Schmidtke, J. (2016). Genetic studies on the Cayo Santiago rhesus macaques: A review of 40 years of research. *American Journal of Primatology*, 78(1), 44–62. <https://doi.org/10.1002/ajp.22424>
- [5] Benderlioglu, Z., & Guatelli-Steinberg, D. (2019). Is there a link between matriline dominance rank and linear enamel hypoplasia? An assessment of defect prevalence and count in Cayo Santiago rhesus macaques (*Macaca mulatta*). *International Journal of Primatology*, 40, 263–275. <https://doi.org/10.1007/s10764-019-00084-z>
- [6] Lawrence, J., Stojanowski, C. M., Paul, K. S., Seidel, A. C., & Guatelli-Steinberg, D. (2021). Heterogeneous frailty and the expression of linear enamel hypoplasia in a genealogical population. *American Journal of Physical Anthropology*, 176(4), 638–651. <https://doi.org/10.1002/ajpa.24288>
- [7] Newman, L. E., Testard, C., DeCasien, A. R., Chiou, K. L., Watowich, M. M., Janiak, M. C., . . . , & Higham, J. P. (2023). The biology of aging in a social world: Insights from free-ranging rhesus macaques. *Neuroscience & Biobehavioral Reviews*, 154, 105424. <https://doi.org/10.1016/j.neubiorev.2023.105424>
- [8] Romero, A. N., Dickinson, E., Turcotte, C. M., & Terhune, C. E. (2023). Skeletal age during hurricane impacts fluctuating asymmetry in Cayo Santiago rhesus macaques. *Ecology and Evolution*, 13(8), e10425. <https://doi.org/10.1002/ece3.10425>
- [9] Milich, K. M., Ruiz-Lambides, A., Maldonado, E., & Maestriperi, D. (2020). Age negatively impacts reproduction in high-ranking male rhesus macaques on Cayo Santiago, Puerto Rico. *Scientific Reports*, 10(1), 13044. <https://doi.org/10.1038/s41598-020-69922-y>
- [10] Pavez-Fox, M. A., Negron-Del Valle, J. E., Thompson, I. J., Walker, C. S., Bauman, S. E., Gonzalez, O., . . . , & Brent, L. J. (2021). Sociality predicts individual variation in the immunity of free-ranging rhesus macaques. *Physiology & Behavior*, 241, 113560. <https://doi.org/10.1016/j.physbeh.2021.113560>
- [11] Testard, C., Larson, S. M., Watowich, M. M., Kaplinsky, C. H., Bernau, A., Faulder, M., . . . , & Brent, L. J. (2021). Rhesus macaques build new social connections after a natural disaster. *Current Biology*, 31(11), 2299–2309. <https://doi.org/10.1016/j.cub.2021.03.029>
- [12] Gonzalez, S. J., Sherer, A. J., & Hernández-Pacheco, R. (2023). Differential effects of early life adversity on male and female rhesus macaque lifespan. *Ecology and Evolution*, 13(11), e10689. <https://doi.org/10.1002/ece3.10689>
- [13] Zhao, M. Q., Maldonado, E., Kensler, T. B., Kohn, L. A., Guatelli-Steinberg, D., & Wang, Q. (2021). Conceptual design and prototyping for a primate health history knowledge model. In *Advances in Computer Vision and Computational Biology: Proceedings from IPCV'20, HIMS'20, BIOCAMP'20, and BIOENG'20*, 509–520. https://doi.org/10.1007/978-3-030-71051-4_40
- [14] Zhao, M. Q., Widener, E. R., Francis, G., & Wang, Q. (2023). Building a knowledge model of Cayo Santiago rhesus macaques: Engaging undergraduate students in developing graphical user interfaces for NSF funded research project. In *Proceedings of the Second International Conference on Innovations in Computing Research*, 344–353. https://doi.org/10.1007/978-3-031-35308-6_29
- [15] Zhao, M. Q., Kaur, M., Kundu, S., & Wang, Q. (2023). Data exploration, preparation, and pilot studies for building a knowledge model of the Cayo Santiago rhesus monkeys. In *2023 International Conference on Computational Science*

- and Computational Intelligence, 633–638. <https://doi.org/10.1109/CSCI62032.2023.00111>
- [16] Joinery. (2024). *Joinery: Data frames for Java*. Github. <https://github.com/cardillo/joinery>
- [17] Apache Commons. (2024). *Commons math: The apache commons mathematics library*. <https://commons.apache.org/proper/commons-math/>
- [18] Beck, K., & Andres, C. (2004). *Extreme programming explained: Embrace change* (2nd ed., The XP series). USA: Addison-Wesley.
- [19] Bruegge, B., & Dutoit, A. H. (2010). *Object-oriented software engineering: Using UML patterns and Java*. USA: Prentice Hall.
- [20] Huang, M. (2020). Theory and implementation of linear regression. In *2020 International Conference on Computer Vision, Image and Deep Learning*, 210–217. <https://doi.org/10.1109/CVIDL51233.2020.00-99>
- [21] Wang, P., Chen, S., & Yang, S. (2022). Recent advances on penalized regression models for biological data. *Mathematics*, 10(19), 3695. <https://doi.org/10.3390/math10193695>
- [22] JFreeChart. (2021). *The JFreeChart project*. <https://www.jfree.org/jfreechart/>

How to Cite: Zhao, M. Q., Novak, C., Widener, E. R., Patel, R. A., Gong, R., Francis, G., & Wang, Q. (2025). Enhancing Data Analytics and Visualization Support in CSViewer for Analysts – Version 1.1: Access to an Integrative Database and Knowledge Model of Cayo Santiago Rhesus Macaques. *Journal of Data Science and Intelligent Systems*, 3(4), 282–290. <https://doi.org/10.47852/bonview.JDSIS42023383>