Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning

A Project Report
Presented to
The Faculty of the Department of Applied Data Science
San Jose State University
In Partial Fulfillment
Of the Requirements for the Degree
Master of Science in Data Analytics

By

Sai Srivathsav Aripirala, Prudhvi Chowdary Chirumamilla, Chinmaya Gayathri, Shashank Shashishekhar Reddy, Bhavika Prasannakumar

December 2, 2024

APPROVED FOR DEPARTMENT OF APPLIED DATA SCIENCE

Dr. Jerry Gao, Project Advisor

Dr. Lee C. Chang, Department Chair

ABSTRACT

Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning

By

Sai Srivathsav Aripirala, Prudhvi Chowdary Chirumamilla, Chinmaya Gayathri, Shashank Shashishekhar Reddy, Bhavika Prasannakumar

Carbon, in the form of carbon dioxide (CO2), plays a vital role in maintaining Earth's temperature through the natural greenhouse effect. However, human activities have disrupted this balance with excessive CO2 emissions from burning fossil fuels and deforestation, trapping heat in the atmosphere and contributing to global warming. This has resulted in a cascade of environmental impacts, including intense heat waves, droughts, floods, and storms, alongside rising temperatures that accelerate the melting of glaciers and polar ice caps. These extreme events pose significant threats to food security, agricultural productivity, and the stability of ecosystems, emphasizing the urgency of addressing carbon emissions to safeguard the planet's future.

Forests, as one of Earth's most effective natural systems for carbon storage, provide critical ecosystem services to mitigate climate change. Through photosynthesis, trees absorb carbon dioxide, using it to build their biomass while releasing oxygen. Carbon is stored in the wood, leaves, roots, and soil, with mature forests being particularly effective due to their large biomass and stable ecosystems. The world's forests represent a substantial portion of the terrestrial carbon sink, but deforestation and forest degradation continue to contribute significantly to carbon emissions.

Accurately assessing and measuring carbon stored in forests is pivotal for addressing climate change. Forest ecosystems are dynamic and vast, making precise quantification of

carbon challenging. Advanced methodologies leveraging big data analytics and machine learning offer promising solutions for enhancing the precision, efficiency, and scalability of forest carbon assessments. Techniques such as remote sensing with satellite imagery and LiDAR provide detailed insights into forest structure and biomass distribution. Machine learning models, including deep learning techniques, enable the extraction of complex patterns within large datasets, facilitating more accurate carbon stock estimation and monitoring.

This project adopts a holistic approach to advancing forest carbon analysis, encompassing data collection, model development, and the creation of a user-friendly web portal. The integration of diverse datasets, including weather patterns, remote sensing data, and historical carbon data, supports robust analysis. Advanced machine learning models are employed to process these datasets, yielding reliable insights into forest carbon dynamics. The development of a web portal further enhances accessibility, equipping stakeholders with tools for carbon analysis, measurement, and reporting. By combining innovative technologies with a comprehensive approach, this project contributes to informed decision-making in forest management, climate change mitigation, and sustainable ecosystem stewardship.

ACKNOWLEDGMENTS

We extend our heartfelt gratitude to everyone who contributed to the successful completion of this project. Your unwavering support and guidance have been invaluable to us throughout this journey.

We would like to express our deepest appreciation to Dr. Jerry Gao, our Project Advisor, for his exceptional mentorship and expertise. Dr. Gao's profound knowledge and insights were instrumental in shaping the direction and success of our project. His guidance and dedication have been a cornerstone of our progress.

Our sincere thanks also go to Dr. Lee C. Chang, our esteemed department chairperson, for his steadfast support and encouragement. Dr. Chang's oversight and commitment to monitoring our project's progress have been an honor and a privilege, and we are grateful for his leadership.

We are immensely thankful to our peers, friends, and family members, who not only supported us financially but also inspired us with their constant encouragement. Their belief in our abilities has been a source of motivation and resilience throughout this endeavor.

Finally, we are deeply grateful to all individuals and organizations who contributed to the success of this project. Your support, insights, and contributions have played a pivotal role in its completion. It has been a privilege to collaborate with such extraordinary individuals and to benefit from your expertise and generosity.

TABLE OF CONTENTS

1. Introduction

1.1 Project Background and Executive Summary

Climate change poses a significant global challenge, primarily due to the increase in carbon dioxide levels, where forests serve as critical carbon sinks by absorbing CO2 through photosynthesis. Effective climate change mitigation relies on accurate monitoring of carbon reserves in forests. However, traditional monitoring methods often lack precision and scalability. This project bridges this gap by leveraging advanced big data and machine learning techniques to deepen our understanding of carbon dynamics in forests. Current methods for estimating carbon stocks and biomass are often simplistic and limited by available datasets, resulting in inaccuracies. Improving the accuracy and reliability of these estimates is essential for guiding mitigation strategies and promoting sustainable forest management practices. Through the utilization of modern technology and analytical approaches, the project aims to provide stakeholders with robust estimates of carbon stocks and biomass in forests. Traditional approaches to carbon stock estimation may not fully capture the complexity of forest ecosystems. By integrating advanced data processing tools with traditional methodologies, the project seeks to develop more accurate and scalable methods. Current market shows robust demand for technologies like LiDAR and advanced spectral imaging to enhance carbon stock assessments and influence carbon credit evaluations. Our project integrates these technologies, meeting market demands and ensuring high accuracy and relevance in the competitive carbon trading landscape. Motivated by the urgency to combat climate change and preserve biodiversity, the project aims to develop a comprehensive system for monitoring carbon reserves and biomass in forests. Through the analysis of biomass and carbon stock estimation results, a deeper understanding of forest ecosystems' carbon dynamics can be attained, facilitating informed decision-making and the adoption of sustainable forest management practices. Accurate assessment and measurement

of carbon in forests entail consideration of several approaches, methods, and factors which are discussed in the following sections.

1.1.1 Advanced Carbon Stock Estimation Models

The project aims to develop advanced machine-learning models for carbon stock estimation in forests, enhancing the accuracy and reliability of assessments. By integrating diverse datasets, including satellite imagery, LiDAR data, and climate data, sophisticated algorithms will be leveraged to develop robust models. These models will employ ensemble learning techniques and data fusion methods to improve robustness and generalization capabilities. Furthermore, validation and calibration using ground-based data will ensure accuracy and reliability in real-world forest environments. Ultimately, these advancements will contribute to improved forest carbon monitoring and assessment.

1.1.2 User-Friendly Web Portal for Carbon Analysis

The project endeavors to create a user-friendly web portal, embedding comprehensive solutions for forest carbon analysis, measurement, tracking, and reporting. Through accessible tools and interfaces, the portal aims to empower stakeholders with actionable insights for informed decision-making. This initiative involves the design and development of an intuitive web interface equipped with interactive visualization tools for accessing and analyzing carbon data. Moreover, the integration of machine learning algorithms and data processing pipelines into the portal enables real-time carbon analysis and monitoring. The portal's implementation ultimately enhances accessibility to forest carbon data and analysis tools, facilitating informed decision-making for researchers, policymakers, and forest managers concerning forest conservation and climate change mitigation efforts.

1.1.3 Carbon Credit Measurement and Reporting

The project endeavors to develop tools within the web portal for carbon credit measurement and reporting, aiming to streamline carbon credit transactions in forest management. Through the implementation of algorithms, it seeks to quantify carbon credits

based on changes in carbon stocks and emissions reductions resulting from forest management practices. Can the integration of reporting templates and data visualization tools will effectively communicate carbon credit metrics to stakeholders. Ultimately, these efforts aim to simplify carbon credit management processes for forest owners and managers, facilitating their participation in carbon offset programs and encouraging sustainable forest management practices.

1.2 Project Requirements

Data Requirements

Remote Sensing Data. High-resolution satellite images capturing forest cover, land use, and changes over time and LiDAR data providing detailed information on forest structure, canopy height, and biomass distribution.

Historical Carbon Data. Long-term records of carbon measurements in forests, including carbon stocks in biomass and soil. Data on carbon fluxes, capturing the movement of carbon into and out of forest ecosystems over time.

Functional Requirements

Biomass Estimation Feature. Utilize remote sensing data including satellite imagery and LiDAR to estimate the biomass of forests. Develop models capable of analyzing complex data sets to provide insights into biomass distribution within forests.

Carbon Stock Estimation Feature. Use diverse data sources including historical carbon data and ground-based measurements to estimate the carbon stock in forests. Apply advanced machine learning techniques such as Artificial Neural Networks, Random Forests, and Quantile Regression Neural Network to enhance precision in carbon stock estimation. Validate models rigorously using ground-based data to ensure reliability and accuracy.

Comparison Analysis. Conduct a comparative analysis between biomass estimation and carbon stock estimation methodologies. Evaluate the strengths and limitations of each approach in accurately assessing forest carbon dynamics. Providing insights into the

correlation between biomass and carbon stock estimates for different forest types and conditions.

AI-Powered Feature Requirements

Machine Learning Models. Develop deep learning models capable of analyzing large and diverse datasets to support biomass and carbon stock estimation. Implement algorithms such as Artificial Neural Network (ANN), TabNet, Convolutional Neural Network (CNN), Deep Neural Networks (DNN), and a hybrid model for improved accuracy. Ensure scalability and efficiency of models to handle spatial and temporal data analysis effectively.

Integration with Data Processing Models. Integrate machine learning models with data processing models to streamline the analysis of remote sensing data. Enable automated data preprocessing and feature extraction to facilitate efficient model training. Optimize algorithms to handle complex interactions within the data for accurate estimation of biomass and carbon stock.

Testable and Measurable Requirements

Accuracy Metrics. To ensure rigorous evaluation of biomass and carbon stock estimation models, performance will be quantified using key metrics, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Squared Error (MSE), and the Coefficient of Determination (R2). These metrics will provide a comprehensive assessment by measuring the magnitude of prediction errors and the model's ability to explain variance in the data (R2). Systematic testing will be carried out across diverse forest ecosystems and environmental conditions to evaluate model performance, robustness, and adaptability. Benchmarks for accuracy will be defined based on established industry standards and scientific literature, ensuring the models meet the requirements for practical applications in sustainable forestry.

Scalability Assessment. Assess the scalability of machine learning models to handle large volumes of data and diverse data sources. Measure the computational efficiency of

models to ensure timely processing of data for real-time applications. Conduct stress testing to determine the maximum capacity of the system and identify potential scalability bottlenecks.

User Interface Evaluation. Evaluate the usability and accessibility of the web portal interface for forest carbon analysis. Conduct user testing to gather feedback on the intuitiveness of features and ease of navigation. Incorporate user feedback to iteratively improve the user interface design and enhance user experience.

1.3 Project Deliverables

The project aims to create a comprehensive method for monitoring carbon reserves in forests using big data and machine learning approaches. The primary goal is to precisely quantify the amount of CO2 absorbed by trees and forests, measured per hectare or grid, in order to aid climate change mitigation efforts. To do this, our team will start by defining technical criteria and completing a thorough literature review. It will collect and preprocess a variety of information, including satellite imaging, LiDAR data, and ground-level observations, to evaluate forest conditions. The next phase will be to identify and evaluate the most effective machine learning models and algorithms, with an emphasis on spatial and temporal data analysis to compute CO2 absorption rates.

The system will be designed to examine present carbon stocks and estimate future changes, with a focus on grid-level precision. All project resources, such as datasets, code, and models, will be securely stored on GitHub to facilitate collaboration and version control. The project will end in a complete final report with key deliverables listed in Table 1, that summarizes the design, execution, and outcomes, including a breakdown of CO2 absorption rates by hectare or grid and actionable insights for climate change mitigation. This complete technique is intended to provide a scalable and precise solution for monitoring forest carbon dynamics.

Table 1

Listing of Project Deliverables with the dates

Phase	Deliverables	Description	Date
Project Understanding	Literature Survey report	Summarizes the findings and conclusions of linked research publications.	02/26/24
	Technology and data requirements report	Evaluates the technological and data requirements for the project	02/26/24
	Project Plan	Describes development methods, milestones, and data management.	03/18/24
Data Preparation	Data Collection, EDA	Gather data and do data exploration and preparation.	04/08/24
	Data Modelling Report	Finalizing and processing the data for deep learning models.	04/15/24
Data Documentation	Project Presentation	Presentation of project progress for the spring semester	05/06/24
	Spring Semester report	Detailed update on project progress until Spring semester.	05/13/24
Model building	Model Proposal and development	A proposal for models to solve the challenge, including model architecture, data flow, and development methods.	05/02/24
	Model accuracy and evaluation	Report on model accuracy and evaluation using certain performance metrics.	05/02/24
Evaluation	Comparison report	Reporting the results of the developed models and the evaluation metrics	09/30/24
	Summary report	Summarizing the performances and research conducted during the project	10/15/24
Deployment	Prototype of web application	Development of a web portal to host the model.	11/17/24
	Presentation	Presentation of the project using Microsoft PowerPoint.	12/10/24
	Final Report	Final documentation of the whole project	12/02/24

1.4 Literature Survey of Carbon Stock Estimation

Table 2 shows how carbon stock estimation has evolved over time, with each methodology adding uniquely to the field's development. Beginning with the implementation of Allometric Equations in 2022, which did not require precision but relied on ground-level measures to estimate [1]. In the same year, a Multi-Layer Perceptron model reached 65% accuracy using Sentinel 2A images, paving the way for neural network integration [2]. By 2023, the Gradient Boosting Decision Tree technique has improved accuracy to 90% utilizing Gaofen Remote Sensing Data, demonstrating the efficacy of machine learning algorithms [3].

In the same period, the PolInSAR approach, which employed ALOS PALSAR data, produced insights with R2 values ranging from 0.7525 to 0.3772, [4].

In 2003, DEM, DSM, and DCHM images were derived from 3-D remote sensing Lidar data, revealing the early usage of three-dimensional mapping techniques [5]. In 2022, SVM, KNN, RF, and XGBoost were linked with satellite image data, demonstrating the promise of synthetic aperture radar in remote sensing [6]. According to [7], systematic sampling, loss on ignition, and wet oxidation methods yielded total carbon stocks of 283.80 t·ha−1, with soil carbon at 168.15 t·ha−1. The FORCARB2 forest carbon budget simulation model, employed in 2004, depended on FIA inventory data and the RPA database, showing an early integration of simulation models [8].

Despite the lack of trustworthy results, the study highlighted the necessity for additional research employing deep learning models. In 2023, AI solutions will include GS and SD models for evaluating carbon potential in forests [9] By 2022, a combination of KNN, SVM, ANN, and Formulas had been used, but there was no reported accuracy, indicating a multi-model strategy on inventory and sensor data [10]. This in-depth study describes the transition from traditional ground-based observations to advanced machine learning and deep learning models, emphasizing the field's resilience and ongoing quest of accuracy and efficiency in carbon stock assessment.

Table 2

Literature Survey of Carbon Stock Estimation

Year	Focused Problem	Approach	Accuracy	Datasets	Author
2022	Carbon Stock Estimation	Allometric Equations	N/A	Ground Level Metrics	BIAD LIGNE
2022	Carbon Stock Estimation	Multi-Layer Perceptron	65%	Sentinel 2A Images	Budak
2023	Carbon Stock Estimation	Gradient Boosting Decision Tree	90%	Gao fen Remote Sensing Data	Li
2017	Carbon Stock Estimation	PolInSAR technique using ALOS PALSAR	75%	Remote Sensing	Jaya
2003	Carbon Stock Estimation	DEM, DSM, DCHM Image extraction	N/A	3-D Remote Sensing Lidar data	Omasa
2022	Carbon Stock Estimation	SVM, KNN, RF, XGBoost	73%SVM, 77%KNN,83%RF, 89%XGB	Satellite Image data	Uniyal
2022	Carbon Stock Estimation	KNN, SVM, ANN, Formulas	N/A	Inventory, Sensor’s data	Sun & Liu

2012	Carbon Stock Estimation	Systematic sampling, loss on ignition, wet oxidation method	Total 283.80 t·ha−1; Soil 168.15 t·ha−1	Biomass, soil carbon	Ullah
2004	Carbon Stock Estimation	FORCARB2 forest carbon budget simulation model	N/A	FIA inventory data, RPA database	Smith
2023	Carbon Stock Estimation	AI solutions	N/A	Weather data	Chen

Table 3 shows in 2020, researchers used Linear Regression, Random Forest, and XGBoost to estimate biomass using data from China's National Forest Continuous Inventory as well as satellite photography. XGBoost had the highest accuracy rate, at 92% [11]. The SVM approach, which was used in 2012, obtained 84.62% accuracy with Landsat-7 imagery, demonstrating the early application of machine learning in the field [12]. That same year, the implementation of Bagging Stochastic Gradient Boosting achieved a 90% accuracy utilizing ALOS PALSAR data, demonstrating advances in algorithmic approaches for biomass estimation [13].

Furthermore, in 2012, the Random Forest approach with SPOT-5 images and LiDAR data obtained 84% accuracy [14], while another study utilizing RF with Landsat imagery produced an astonishing 94.3% accuracy [15], demonstrating the promise of RF algorithms. By 2014, studies on Multiple-Biomass Models and the Mean Ratio Method that did not define accuracy used data from China's forest inventory [16].

The adoption of an Allometric Growth Model based on a tree characteristics dataset in 2008 heralded the end of previous models that relied on tree-specific data [17].

The integration of RF, LR, and XGBoost in 2019, which analyzed Continuous Inventory and Landsat 8 data, demonstrated the benefits of merging various models. In 2012, using LME Regression alongside RF, SVR, and Cubist Regression Trees resulted in varying degrees of effectiveness in improving estimation accuracy using LiDAR data [18]. The creative use of synthetic imaging for biomass estimation in 2022, using models such as SVM, KNN, BP, and ELM, pointed to new techniques [19]. Finally, a 2014 study focused on the continuous biomass expansion factor, mean ratio technique, and mean biomass density across 16 species, emphasizing species-specific methodologies.

This story historically shows the many approaches and methodology used in biomass estimating research, demonstrating the evolution and range of accuracy reached over time using different methods and data sources.

Table 3

Literature Survey of Biomass Estimation

Year	Focused Problem	Approach	Accuracy	Sensor	Author
2020	Biomass Estimation	LR, RF, XGBOOST	Select weekly confirma%(LR), 60%(RF), 75%(XGB)	Forest Inventory data, Landsat 8, Sentinel-1A	Du
2012	Biomass Estimation	SVM	84.62%	Landsat-7	Chen

2012	Biomass Estimation	Bagging Stochastic Gradient Boosting (BagSGB)	90%	ALOS PALSAR	Carreiras
2012	Biomass Estimation	RF	84%	SPOT-5, LiDAR	Avitabile
2012	Biomass Estimation	RF	94.3%	Landsat	Guo
2014	Biomass Estimation	Multiple-Biomass Models (MBM)	N/A	China's forest inventory data	Dapao
2008	Biomass Estimation	Allometric model (AGB)	71%	Tree characteristics dataset	Mekonnen & Riley
2019	Biomass Estimation	RF, LR, XGBoost	63%(RF), 30%(LR), 71%(XGB)	Continuous Inventory data Landsat 8 Operational Land Imager data	Li
2012	Biomass Estimation	Linear Mixed Effects (LME) Regression RF, SVR, Cubist Regression Trees:	32%LMR, 22%RF, 93%SVR, 69%CRT	LiDAR Data	Gleason

2022	Biomass estimation	SVM, KNN, BP, ELM	65%(SVM) 62%(KNN) 64%(BP) 68%(ELM)	Synthetic Sentinel-2 images	Jiang
2014	Biomass Estimation	Continuous biomass expansion factor	NA	China's National Forest Inventory	Chank

Herein, we present the formulas for feature variables in Figure 1 that are essential for calculating carbon stocks, as utilized in various articles which are mentioned above.

Figure 1

Fo Eco-needrmula for various attributes

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8235 entries, 0 to 8234
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   LST                 8235 non-null   int64
 1   EVI                 8235 non-null   int64
 2   NDVI                8235 non-null   int64
 3   Fpar                8235 non-null   int64
 4   Lai                 8235 non-null   int64
 5   Gpp                 8235 non-null   int64
 6   rh                  8235 non-null   int64
 7   rh100               8235 non-null   int64
 8   landsat_treecover   8235 non-null   int64
 9   modis_treecover     8235 non-null   int64
 10  modis_nonvegetated  8235 non-null   int64
 11  agbd                8235 non-null   int64
dtypes: int64(12)
memory usage: 772.2 KB
None
```

$$L(\lambda, \mu) = t_1(\lambda, \mu)\varepsilon(\lambda, \mu)B(\lambda, T_s) + L_a(\lambda, \mu)$$

$$\mathbf{EVI = G * ((NIR - R) / (NIR + C1 * R - C2 * B + L))}$$

$$\mathbf{RH100 = elev\_highestreturn - elev\_lowestmode}$$

$$\mathbf{GPP = \varepsilon * APAR}$$

$$\mathbf{NDVI = (NIR - R) / (NIR + R)}$$

1.5 Technology Survey

We have covered the current methodologies and platforms for monitoring carbon stocks, land use changes, and deforestation, indicating a clear trend toward the employment of modern technology in environmental conservation efforts. Table 4 displays a diverse range of goods, each serving a distinct purpose, such as retrieving satellite data, evaluating climate data, and estimating carbon emissions. Like advances in carbon stock and biomass

assessment, this evolution emphasizes the use of current computation and remote sensing technology. Products like Open Foris Collect Earth, Global Forest Watch, and Terra-i highlight the industry's shift toward comprehensive environmental monitoring systems that use several data sources, such as satellite imaging and ground observations.

These technologies not only capture exact land cover and land-use data, but also allow for in-depth investigation of carbon density and climate patterns, reflecting a larger trend in environmental sciences toward data-driven decision-making. The emphasis on providing APIs, SDKs, and online platforms reflects a growing emphasis on accessibility and user involvement, with the goal of democratizing access to environmental data and technologies. This evolution is consistent with broader technological advances in environmental monitoring, where advanced algorithms and data analytics have become important to understanding and managing ecological challenges.

Table 4

Technology Survey

Product	Functions	Carbon Monitor	OFCE	Global Forest Watch	Terra-i	Varaha	Albo Climate
Features	Collects land-use data	No	Yes	Yes	Yes	Yes	Yes
	Collects land-cover data	No	Yes	Yes	Yes	Yes	Yes
	Access to Satellite Data	No	Yes	Yes	Yes	Yes	Yes
	Estimating Carbon Emissions	Yes	Yes	Yes	Yes	Yes	Yes
	Analyses climate data	No	No	Yes	Yes	No	No

	Estimates of Carbon Density	Yes	No	Yes	Yes	Yes	No
Applications	Carbons Stock Estimation	Yes	No	Yes	Yes	Yes	Yes
	Monitors land-use changes	No	Yes	Yes	Yes	Yes	Yes
	Monitors deforestation	No	Yes	Yes	Yes	Yes	Yes
	Monitors Climate patterns	No	No	Yes	Yes	No	No
Business Model	Provide API	No	No	Yes	No	Yes	No
	Provide SDK	No	Yes	No	No	Yes	No
	Provide Web	Yes	Yes	Yes	Yes	Yes	Yes
Cost	API	NA	NA	Free	NA	Yes	No
	SDK	NA	Free	NA	NA	Yes	No
	Web	Free	Free	Free	Free	Yes	Yes
Infrastructure	Web app	Yes	Yes	Yes	Yes	No	NA
	Mobile app	No	No	Yes	No	Yes	NA
	Cloud	No	No	Yes	No	No	NA
AI used for	Analytics	No	N/A	No	Yes	No	No
	Processing data	Yes	N/A	No	Yes	Yes	No
	Analyzing	Yes	N/A	No	Yes	Yes	No
	Monitoring	No	N/A	Yes	No	No	No

2. Data and Project Management Plan

2.1 Data Management Plan

Forests primarily store carbon in their biomass, including trees, shrubs, and other vegetation. Trees, with their extensive root systems and above-ground biomass, hold most of this carbon, absorbing CO2 from the atmosphere through photosynthesis and converting it into organic compounds. As trees mature, they accumulate more carbon, acting as a significant sink for atmospheric carbon. Fallen leaves, branches, and other organic debris contribute to carbon [20]. Soil serves as another crucial carbon sink within forest ecosystems, where decomposed organic matter from plants and trees accumulates, forming organic compounds. This process, known as soil carbon sequestration, significantly contributes to carbon storage beneath the forest floor. Furthermore, the presence of mycorrhizal fungi and soil microorganisms enhances carbon sequestration by facilitating the breakdown of organic matter and the incorporation of carbon into the soil [21].

There will be two types of data collected in this project, first part is Global Ecosystem Dynamics Investigation, GEDI is a spaceborne lidar instrument deployed on the International Space Station (ISS) designed to measure the three-dimensional structure of forests, helping to improve understanding of carbon cycle dynamics and ecosystem health, GEDI provides data at different levels of processing, each offering varying degrees of detail and complexity for forest structure analysis, out of which we are interested in 4 types as shown in Table 5.

Level 2A: Canopy Height Metrics - At this level, GEDI data is processed to derive canopy height metrics, such as canopy height and vertical profile metrics, providing insight into the vertical structure of forests.

Table 5

Data Characteristics for MODIS

MODIS

Description	Land surface Temperature (LST)	Vegetation Index (NDVI) (EVI)	Leaf Area Index (LAI)	GPP/NPP
Start Date	2/18/2000	2/18/2000	2/18/2000	2/18/2000
End Date	2/2/2023	2/2/2023	2/2/2023	2/2/2023
Name	MOD11A1	MOD13A2	MOD15A2H	MOD17A2
Sensor	Spectroradiometer	Spectroradiometer	Spectroradiometer	Spectroradiometer
Acquisition Date	2/18/2000	2/18/2000	2/18/2000	2/18/2000
Spatial Range(m)	1000	1000	1000	250
Rows	TBD	TBD	TBD	TBD
Columns	17	13	6	3
Processing Level	3	3	3	3
Altitude	705 kms	705 kms	705 kms	705 kms
File Format	HDF5	HDF5	HDF5	HDF5
File Size	TBD	TBD	TBD	TBD

Table 6

Data Characteristics for GEDI

***GEDI** (LiDAR)*

Description	Elevation & Height Metrics	Canopy Cover & Vertical profile	Gridded Aboveground Biomass Density
Start Date	04/18/2019	04/18/2019	4/18/2019
End Date	3/17/24	3/17/24	3/17/2024
Name	GEDI 2A	GEDI 2B	GEDI 4b
Sensor	LiDAR	LiDAR	LiDAR
Acquisition Date	04/18/2019	04/18/2019	4/18/2019

Spatial Range (m)	1000	1000	1000
Rows	NA	NA	146166
Columns	NA	NA	34704
Processing Level	2	2	4
Altitude	400 kms	400 kms	400 kms
File Format	HDF5	HDF5	TIFF
File Size	~2GB	~0.5GB	TBD

Level 2B: Canopy Cover and Vertical Profiles - This level involves the further processing of Level 2A data to derive additional parameters, including canopy cover and vertical profiles of canopy characteristics, offering more comprehensive information for ecosystem monitoring and carbon assessment.

Level 3: Gridded Canopy Metrics - GEDI Level 3A data involves the aggregation and gridding of Level 2A or 2B data to produce spatially continuous maps of canopy metrics at various resolutions, enabling large-scale analyses and modeling of forest structure and dynamics.

Level 4A: Provides highly detailed aboveground biomass density (AGBD) estimates for each laser footprint, allowing you to precisely assess individual trees or small forest patches.

Level 4B: Offers average aboveground biomass density (AGBD) values across grid cells, ideal for analyzing large-scale trends in forest biomass across landscapes.

Each level of GEDI data serves specific purposes, ranging from detailed analysis of individual laser returns to broader-scale assessments of forest structure and carbon dynamics.

Second part is Moderate Resolution Imaging Spectroradiometer, MODIS is a key instrument aboard NASA's Terra and Aqua satellites, designed to provide global observations of the Earth's land surface, oceans, and atmosphere at moderate spatial resolutions. It offers a

suite of data products that are invaluable for monitoring and understanding various aspects of the Earth's ecosystems. Among these products, the products shown in Table 6 are particularly relevant for estimating soil carbon stocks.

MODIS provides measurements of Land Surface Temperature (LST), which is crucial for understanding the thermal dynamics of the Earth's surface. LST data helps in identifying areas of heat stress, monitoring urban heat islands, and assessing temperature variations across different land cover types, all of which influence soil carbon dynamics.

MODIS-derived vegetation indices, such as the Normalized Difference Vegetation Index (NDVI), offer insights into the health, density, and distribution of vegetation cover. These indices are widely used to monitor vegetation dynamics, identify areas of deforestation or degradation, and assess changes in ecosystem productivity, all of which are closely linked to soil carbon storage.

MODIS also provides estimates of Leaf Area Index (LAI), which represents the total leaf area per unit ground area. LAI is a critical parameter for quantifying vegetation structure and productivity. Changes in LAI can indicate shifts in ecosystem structure, such as changes in canopy density or leaf phenology, which in turn influence the amount of organic matter input to the soil and hence soil carbon stocks.

MODIS-derived estimates of Gross Primary Productivity (GPP) and Net Primary Productivity (NPP) represent the amount of carbon fixed by photosynthesis and the net carbon uptake by vegetation, respectively. These metrics provide insights into ecosystem carbon fluxes and productivity. Understanding GPP and NPP dynamics helps in assessing the capacity of ecosystems to sequester carbon and informs soil carbon stock estimations by providing information on the input of organic carbon into the soil.

Integrating MODIS data on land surface temperature, vegetation indices, leaf area index, and estimates of productivity with soil-related information such as soil type, texture, and land use/land cover can help develop models to estimate soil carbon stocks. By

incorporating these remotely sensed data, scientists can improve the accuracy and spatial resolution of soil carbon stock estimates over large areas, enabling better management and conservation strategies for terrestrial ecosystems.

In this study, we accessed MODIS and GEDI data from the Earthdata portal to investigate the dynamics of terrestrial ecosystems. The Earthdata Search portal served as our primary platform for discovering and downloading relevant Earth observation datasets. To collect MODIS data, we input specific search criteria including time range, geographic region of interest, and the desired MODIS product(s). Subsequently, the portal retrieved datasets matching our criteria, allowing us to browse through search results, assess metadata quality, and select appropriate datasets for download. MODIS data were obtained in various formats such as HDF-EOS, GeoTIFF, or NetCDF, providing a comprehensive basis for our analysis.

For the collection of GEDI data, we employed the Earthdata Search portal. Here, we input our search parameters, specifying the time period, geographic region, and GEDI data product(s) of interest. The portal retrieved relevant GEDI datasets, which we meticulously reviewed, considering metadata, and selecting datasets aligned with our research goals. GEDI data were typically distributed in HDF5 format, containing critical layers such as waveform metrics and canopy height metrics. These datasets were pivotal for our investigation, offering insights into forest structure and dynamics, essential for our study on soil carbon stock estimation. Overall, the Earthdata portal provided an invaluable resource for accessing and analyzing MODIS and GEDI data, facilitating robust research outcomes in ecosystem science.

For effective data management of MODIS and GEDI datasets, our plan includes comprehensive documentation of data acquisition, processing, and analysis procedures to ensure transparency and reproducibility. We will organize data into well-structured directories, incorporating clear file naming conventions and metadata documentation to facilitate data discovery and sharing. We will implement secure data storage protocols to

safeguard against loss or corruption, while ensuring compliance with data sharing policies and licensing agreements. Regular backups and version control mechanisms will be established to maintain data integrity and traceability. We will develop data access protocols and dissemination strategies to promote open science principles and foster collaboration within the research community.

For storage of MODIS and GEDI data, we will leverage cloud-based storage services for their scalability, accessibility, and reliability. One such service is Amazon Simple Storage Service (S3), which offers secure, durable, and highly available object storage. With S3, data can be stored in buckets, organized according to project or data type, and accessed programmatically via APIs. S3 provides features such as versioning, allowing for the retention of multiple versions of data objects, and lifecycle policies, enabling automated data archiving or deletion based on user-defined criteria. Amazon S3 offers various storage classes, including Standard, Infrequent Access (IA), and Glacier, allowing for cost-effective storage options tailored to data access frequency and retrieval speed requirements. By utilizing cloud-based storage Amazon S3 we can ensure efficient, secure, and scalable storage solutions for our MODIS and GEDI datasets, enabling seamless data management and access for research purposes.

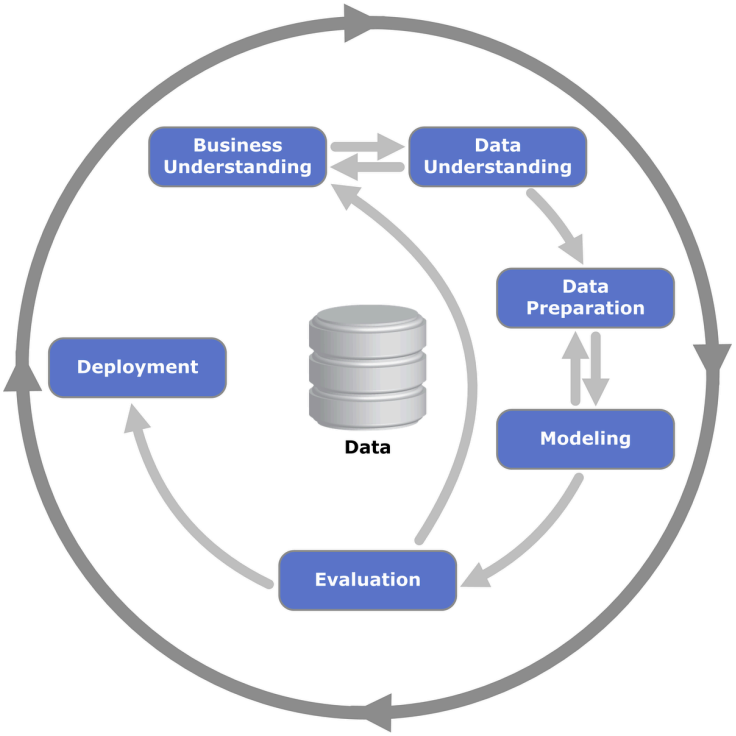
2.2 Project Development Methodology

From project conception to system deployment, systems development can be guided by an organized methodology called the System Development Life Cycle (SDLC). In this project environment, the SDLC framework of choice is CRISP-DM (Cross Industry Standard Process for Data Mining) [22]. For data mining and machine learning applications, the thorough and internationally recognized CRISP-DM methodology is used as shown in Figure 2.

This research primarily focuses on the life cycle of a Data Analyst project within the CRISP-DM Process Model. It encompasses several phases essential for a project, along with their associated activities and outcomes. The life cycle of a DAP (Data Analyst Project) project comprises six phases: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. The purpose of this deployment is to aid in the management and real-time monitoring of forest carbon stocks, which will greatly aid in ecological conservation and the development of mitigation methods for climate change.

Figure 2

CRISP-DM Methods



Business Understanding

For the “Business Understanding" phase, it tackles a number of important topics, starting with an in-depth examination of the effects of forest carbon dynamics on climate change globally and the marketability of carbon credits. The need to understand the carbon cycle, forest ecology, and the vital function that forests play in sequestering carbon. The project's necessity is highlighted by the growing recognition of the relationship between

deforestation and forest degradation and climate change. Understanding the constraints associated with the present carbon assessment approaches is essential.

In this case, applying Machine Learning and Big Data analytics proves to be a transformational strategy. The group's job is to evaluate the practicality and accuracy of these technologies in calculating carbon stocks using extensive sets of forest data, including measurements collected on the ground and satellite photography. A knowledge base on the resources, approaches, and difficulties in this field can be obtained by reviewing the corpus of research that has already been done on forest carbon measurement, remote sensing, and machine learning applications.

The project's direction is determined by considering the forest carbon assessment field's past accomplishments as well as untapped opportunities. Learning the jargon, stacks of technologies, and best practices related to Big Data, machine learning, and remote sensing is also essential. This will guarantee effective team communication and well-informed decision-making along the course of the project.

Data Understanding

A thorough study is carried out to absorb a full understanding of the data necessary for training and assessing machine learning models during the "Data Understanding" phase. The first goal is gathering a wide range of forest data, which will serve as the project's foundation. This encompasses a variety of biophysical factors, ground-based data, and satellite imagery. The data must represent a variety of forest types, conditions, and other important elements in order to guarantee a strong and resilient model.

An essential first step is quality evaluation, which uses metrics like geographical data correctness and resolution quality checks. Either they are deleted or designated for preparation if the data do not match the predetermined standards. It is crucial to use exploratory analysis to find patterns and inherent qualities in the dataset.

The function of visualization techniques is crucial as they offer valuable insights into the distribution patterns and forest structure. Plotting biomass indices, spectral signatures, and canopy coverage are a few examples of these methods. Finding trends, correlations, and abnormalities in the data is made easier with the help of this investigation.

Cloud storage solutions like AWS are used to protect data integrity and enable collaboration. This guarantees safe and scalable data storage, offers effective management tools, and facilitates easy access for all project participants. Using a variety of visualization techniques, the exploration also entails a thorough analysis that guarantees the tracking of temporal changes in the forest cover and phenological patterns for use in further research.

Overall, by offering a comprehensive grasp of the structure, quality, and insights necessary for efficient model training in the field of forest carbon assessment using state-of-the-art big data analytics and machine learning techniques, this phase establishes the foundation for the model's development.

Data Preparation

As part of the "Data Preparation" step, great care is taken to make sure the data is adequately preprocessed for model ingestion and subsequent evaluations. The process begins with the identification and extraction of key forest-related elements from the dataset. Examples of these features could be moisture content, canopy texturing, vegetation indices, and other spectral data properties that indicate the health of the forest and its carbon supply. After selecting features, data cleaning is done to remove any anomalies or inconsistencies that can distort the model's output.

Meanwhile, data annotation is meticulously completed, guaranteeing that every datapoint is correctly classified according to its forest type or capacity for sequestering carbon. For supervised learning models that depend on labeled data, this is an essential stage. Model bias is substantially reduced by homogenizing feature scales through the use of normalization techniques.

Data augmentation strategies are utilized in order to increase the diversity of the dataset and, consequently, the robustness of the model. By adding controlled variability to the data, these methods mimic a range of forest conditions and observational inconsistencies. This kind of augmentation is essential to creating a model that can withstand the variability of real-world data.

The dataset is distilled using sophisticated techniques, focusing on the key features that most contribute to the data's unpredictability. This reduces the computing demands on the model without sacrificing the depth of information.

The data corpus is carefully divided into separate subsets for training, validation, and testing in the last stage of preparation. Deploying machine learning models requires this separation in order to confirm the model's capacity to generalize across new datasets.

Modeling

Building and improving deep learning models to assess forest carbon stock and health metrics is the primary focus throughout the "Modeling" phase of the project. The initial step involves a thorough analysis of various multiclass classification designs. The team carefully evaluates previous models, analyzing their effectiveness and suitability for the complex tasks associated with forest carbon assessment.

Following detailed discussions, the project identified several models, including ANN, TabNet, CNNs, and DNNs (Deep Neural Networks), as starting points for further development. The selection of these models is based on the specific requirements of the project, considering the complexity of forest data analysis and the need for highly detailed results. Rather than using these models directly, the team has rebuilt, modified, and upgraded the architectures from these base models to better align with project requirements. This includes redefining network structures, adjusting hyperparameters, and preparing the models for efficient training. Each model is meticulously tailored to analyze geospatial and remote

sensing data, employing configurations that enable efficient feature extraction and in-depth analysis.

Once designed, the models are integrated into the project's computational framework and trained using a refined dataset. This rigorous training process is critical for teaching the models to analyze input data and detect intricate patterns and signals that reflect forest health and carbon density. During training and validation, the models undergo extensive fine-tuning and parameter optimization. These efforts ensure the highest levels of accuracy and reliability, which are vital for precisely evaluating forest carbon stocks and assessing overall ecosystem health.

Evaluation

The performance and effectiveness of the deep learning models are rigorously assessed in the "Evaluation" phase to ensure they meet the stringent requirements for carbon stock quantification. The models are first validated using a specific dataset, where key performance metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Squared Error (MSE), and the Coefficient of Determination (R2) are meticulously calculated. To enhance performance, hyperparameter tuning is employed to fine-tune the models.

Subsequently, the models undergo comprehensive testing on a separate test dataset, ensuring that the data used for testing has not been involved in training or validation. This critical step evaluates the models' ability to generalize effectively and perform reliably in real-world forest assessment scenarios. By quantitatively assessing performance using predefined metrics, the research team compares the deep learning models against established benchmarks in forest carbon assessment to determine their relative effectiveness and applicability.

Deployment

The team's main goal in the "Deployment" phase is to make the advanced models easily accessible for real-world purposes in the field of measuring the carbon stock of forests. The deployment infrastructure ensures the scalability, stability, and worldwide accessibility of the system by hosting the deep learning models and related data on the Google Cloud Platform (GCP).

An online application is created, utilizing Vue.js in its development, to enable user interaction with the models. This application has an intuitive interface with a feature-rich dashboard. Users can easily submit forest data or satellite imagery for carbon stock research through this interface.

Furthermore, users can acquire quick assessments of forests' carbon stocks because of the deployed system's ability to handle and analyze data in real-time. Procedures for routine maintenance and monitoring are set up to guarantee that the model is kept current and correct with the most recent information and scientific discoveries.

A comprehensive report and presentation are created to capture the full project's journey, from the process to the final deployment. These materials provide a thorough description of the project's contribution to the assessment of forest carbon using machine learning. They painstakingly detail the project's objectives, the creative approaches used, the noteworthy results gained, and the functions of the deployed system.

2.3 Project Organization Plan

The Project Organization Plan comprises a Work Breakdown Structure (WBS) that divides the project into important phases in a hierarchical manner using the CRISP-DM methodology. As shown in Figure 3, Several crucial phases make up the adaptable and iterative CRISP-DM paradigm, which includes Business Understanding, Data Understanding, Data Preparation, Modeling, Assessment, and Deployment.

During the preliminary "Business Understanding" stage, the project team establishes goals, specifications, and benchmarks for success. Understanding the issue domain—in this project, the carbon assessment and measurement of forests using Big Data and Machine Learning—and the intended results depend on this phase.

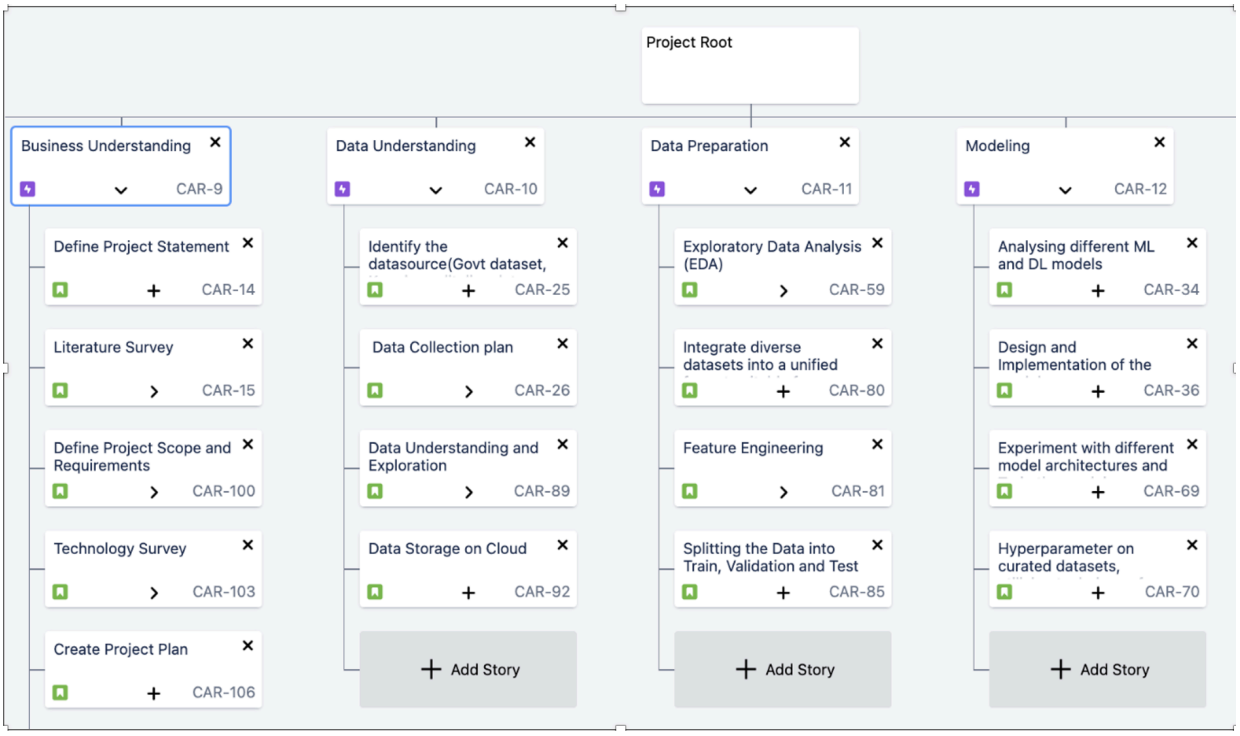
For the purpose of collecting, examining, and preprocessing data in the early stages of the project, the "Data Understanding" and "Data Preparation" phases are essential. This includes gathering a large dataset that includes remote sensor and biomass data relevant to forest carbon stocks, as well as organizing and cleaning the data in preparation for analysis.

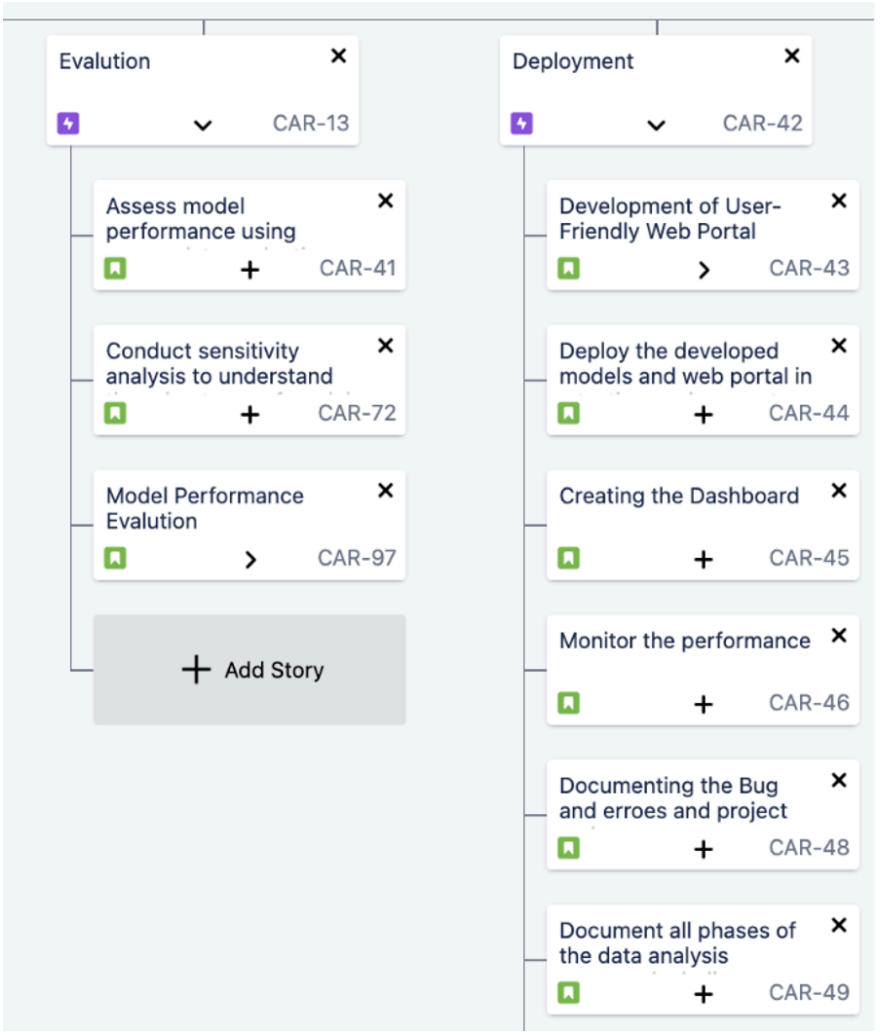
In the "Modeling" phase, the focus of the project shifts to creating advanced machine learning algorithms that are specifically designed to analyze and comprehend the large datasets in order to efficiently predict carbon sequestration. The built-in models undergo rigorous training and calibration, guaranteeing accuracy and dependability in the evaluation of carbon emissions.

The "Evaluation" stage is implemented after the development phase to examine the model's accuracy in comparison to pre-established benchmarks and success measures. The model's performance and adaptability to various forest ecosystems may need to be improved iteratively in order to meet the rigorous evaluation's requirements.

Figure 3

Work Breakdown Structure for Carbon Assessment and Measurement of Forests


Project Root
Business Understanding
CAR-9
Define Project Statement
CAR-14
Literature Survey
CAR-15
Define Project Scope and Requirements
CAR-100
Technology Survey
CAR-103
Create Project Plan
CAR-106
Data Understanding
CAR-10
Identify the datasource(Govt dataset,
CAR-25
Data Collection plan
CAR-26
Data Understanding and Exploration
CAR-89
Data Storage on Cloud
CAR-92
Add Story
Data Preparation
CAR-11
Exploratory Data Analysis (EDA)
CAR-59
Integrate diverse datasets into a unified
CAR-80
Feature Engineering
CAR-81
Splitting the Data into Train, Validation and Test
CAR-85
Add Story
Modeling
CAR-12
Analysing different ML and DL models
CAR-34
Design and Implementation of the
CAR-36
Experiment with different model architectures and
CAR-69
Hyperparameter on curated datasets,
CAR-70
Add Story


Evalution
CAR-13
Assess model performance using
CAR-41
Conduct sensitivity analysis to understand
CAR-72
Model Performance Evalution
CAR-97
Add Story
Deployment
CAR-42
Development of User-Friendly Web Portal
CAR-43
Deploy the developed models and web portal in
CAR-44
Creating the Dashboard
CAR-45
Monitor the performance
CAR-46
Documenting the Bug and erroes and project
CAR-48
Document all phases of the data analysis
CAR-49

2.4 Project Resource Plans and Requirements

An ensemble of cutting-edge hardware and software resources that have been carefully planned to handle enormous volumes of environmental data and demanding computing operations is required for this complicated undertaking. We start our project on local machines so that we may have a quick and iterative development phase for quick prototyping and troubleshooting. For optimal performance, computers with 64-bit architecture and at least 8GB of RAM are required; many team members use the state-of-the-art M2 MacBook due to its exceptional processing power. An 8-core CPU with 8GB of unified memory and a 256GB SSD form the project's computing core.

After development, the project changed course to take advantage of the reliable and scalable storage options offered by Amazon S3, which serves as a safe haven for our sizable datasets and trained models. Data permanence and easy access are ensured by this move to AWS, which is essential for continuing research and deployment within the AWS architecture.

The project uses cutting-edge libraries, such as scikit-learn, to precisely and effectively tackle challenging audio processing and feature extraction tasks.

Combined with Jupyter Notebook integration, Visual Studio Code offers an integrated environment for coding, data analysis, documentation, and version control that is synchronized with GitHub. This allows for integrated development and project management. Jira streamlines project management and adds Gantt chart functionality to enable careful planning and execution. To ensure clarity and engagement in our communication efforts, we use PowerPoint and Microsoft Word to create professional papers and presentations. Table 7 provides a comprehensive overview of the hardware components.

Table 7

Hardware requirements for our project

Hardware	Memory	Configuration	Purpose
Azure High-performance Compute VM	16 GB	NVIDIA GPU	To perform extensive model training and computations
AWS EC2	16 GB	General Purpose (scalable upon need)	For hosting the web service and managing data flow
AWS S3 Storage	Scalable	-	For storing large datasets and results securely
Azure or Amazon SageMaker	-	Machine Learning Optimized Instances	To host and serve the machine learning models

Our project's software requirements involve a variety of libraries and packages, each with a distinct function. These software elements are necessary for several project components, including data manipulation, data storage, data management, machine learning, data visualization, and more. A detailed description of the software components is given in Table 8.

Table 8

Libraries/Packages used in our project.

Libraries/Package	Purpose	Version
Tensorflow	Deep learning model creation and training	2
PyTorch	Deep learning model creation and training	1.5
Apache Hadoop/Spark	Big data processing and Analysis	Latest release

PySpark	Processing large scale data in Python with Spark	Corresponding with Hadoop/Spark version
MySQL/PostgreSQL	Database Management for storing project data	Latest release corresponding with Amazon RDS
ERDAS IMAGINE or similar	Remote sensing data analysis and processing	Latest release
ArcGIS or QGIS	Geographic Information System (GIS) data analysis	Latest release
Tableau or Power BI	Data visualization and dashboard creation	Latest release

Table 9

Tools and Licenses for our project.

Tools	**Purpose**	**License**
Visual Studio Code	Code development and editing	Open Source
draw.io	Diagram and flowchart creation	Free
Atlassian JIRA software	Creating GANTT charts for tracking project	7.5.4
LucidChart	Advanced diagram and flowchart creation	Free for basic
Microsoft Excel	Spreadsheet and Data Analysis	Free
Google Docs	Collaborative document creation	Free
Microsoft Powerpoint	Presentation creation and slideshows	Free
Github	Geographic Information System (GIS) data analysis	Free for public repositories
Zoom	Video Conferencing for internal team meetings	Free

In order to accomplish this project successfully, resources have been allotted in the form of AWS cloud computing services, which will be used for six months at a cost of $30.00 per month. These cloud resources are essential for the computation and data processing needed by the deep learning and machine learning models. In order to guarantee effective project management, teamwork, and issue tracking among the project team members, Jira, a project management and issue tracking platform, has also been implemented for a seven-month period. The project budget covers the cost of using Jira at no extra expense. Table 10 shows that the total projected cost of these resources for the life of the project is $210.00. These resources are necessary to meet project goals and optimize workflows.

Table 10

Project resources cost and justification.

Resources	Justification	Duration(months)	Cost
Amazon Web Services	Cloud Storage and computing	6-7 months	Roughly $30 per month
JIRA software	Project Tracking	8 months	$0.00
Total			$210.00

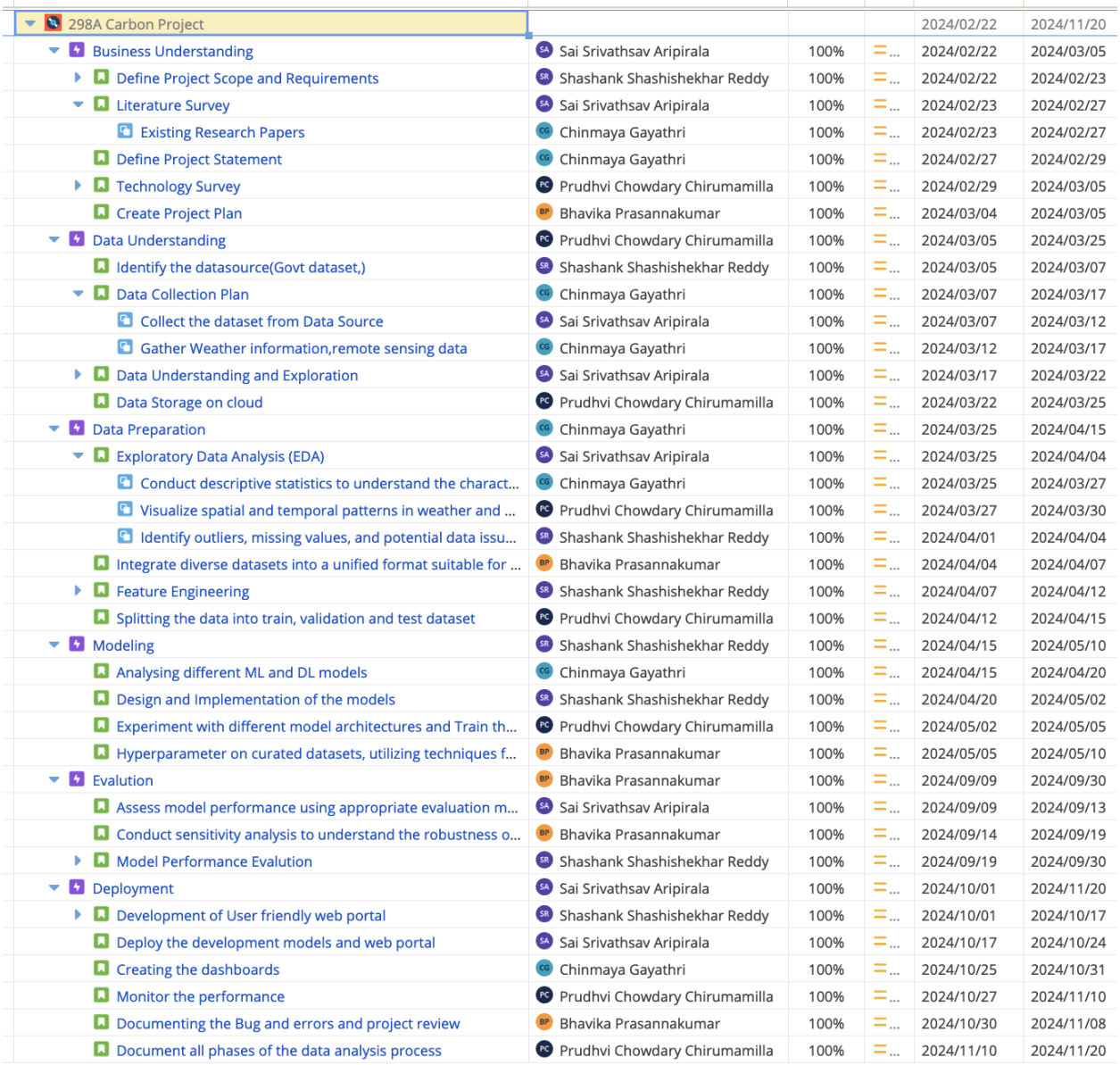
2.5 Project Schedule

Gantt Chart

In the context of documentation, a Gantt chart is useful, particularly when utilizing the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, which consists of several steps ranging from identifying business objectives to implementing data mining solutions.

Figure 4

Gantt chart for Carbon Assessment and Measurement of Forests Project.



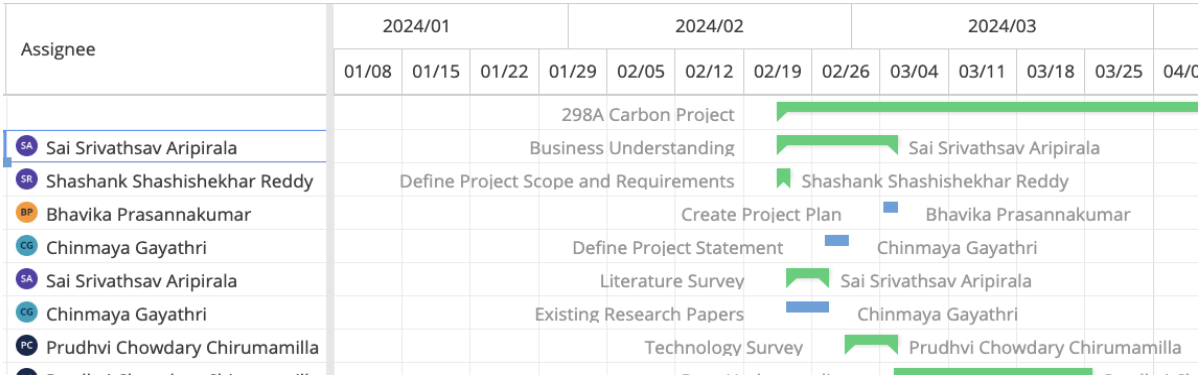
298A Carbon Project				2024/02/22	2024/11/20
Business Understanding	Sai Srivathsav Aripirala	100%	...	2024/02/22	2024/03/05
Define Project Scope and Requirements	Shashank Shashishekhar Reddy	100%	...	2024/02/22	2024/02/23
Literature Survey	Sai Srivathsav Aripirala	100%	...	2024/02/23	2024/02/27
Existing Research Papers	Chinmaya Gayathri	100%	...	2024/02/23	2024/02/27
Define Project Statement	Chinmaya Gayathri	100%	...	2024/02/27	2024/02/29
Technology Survey	Prudhvi Chowdary Chirumamilla	100%	...	2024/02/29	2024/03/05
Create Project Plan	Bhavika Prasannakumar	100%	...	2024/03/04	2024/03/05
Data Understanding	Prudhvi Chowdary Chirumamilla	100%	...	2024/03/05	2024/03/25
Identify the datasource(Govt dataset,)	Shashank Shashishekhar Reddy	100%	...	2024/03/05	2024/03/07
Data Collection Plan	Chinmaya Gayathri	100%	...	2024/03/07	2024/03/17
Collect the dataset from Data Source	Sai Srivathsav Aripirala	100%	...	2024/03/07	2024/03/12
Gather Weather information,remote sensing data	Chinmaya Gayathri	100%	...	2024/03/12	2024/03/17
Data Understanding and Exploration	Sai Srivathsav Aripirala	100%	...	2024/03/17	2024/03/22
Data Storage on cloud	Prudhvi Chowdary Chirumamilla	100%	...	2024/03/22	2024/03/25
Data Preparation	Chinmaya Gayathri	100%	...	2024/03/25	2024/04/15
Exploratory Data Analysis (EDA)	Sai Srivathsav Aripirala	100%	...	2024/03/25	2024/04/04
Conduct descriptive statistics to understand the charact...	Chinmaya Gayathri	100%	...	2024/03/25	2024/03/27
Visualize spatial and temporal patterns in weather and ...	Prudhvi Chowdary Chirumamilla	100%	...	2024/03/27	2024/03/30
Identify outliers, missing values, and potential data issu...	Shashank Shashishekhar Reddy	100%	...	2024/04/01	2024/04/04
Integrate diverse datasets into a unified format suitable for ...	Bhavika Prasannakumar	100%	...	2024/04/04	2024/04/07
Feature Engineering	Shashank Shashishekhar Reddy	100%	...	2024/04/07	2024/04/12
Splitting the data into train, validation and test dataset	Prudhvi Chowdary Chirumamilla	100%	...	2024/04/12	2024/04/15
Modeling	Shashank Shashishekhar Reddy	100%	...	2024/04/15	2024/05/10
Analysing different ML and DL models	Chinmaya Gayathri	100%	...	2024/04/15	2024/04/20
Design and Implementation of the models	Shashank Shashishekhar Reddy	100%	...	2024/04/20	2024/05/02
Experiment with different model architectures and Train th...	Prudhvi Chowdary Chirumamilla	100%	...	2024/05/02	2024/05/05
Hyperparameter on curated datasets, utilizing techniques f...	Bhavika Prasannakumar	100%	...	2024/05/05	2024/05/10
Evalution	Bhavika Prasannakumar	100%	...	2024/09/09	2024/09/30
Assess model performance using appropriate evaluation m...	Sai Srivathsav Aripirala	100%	...	2024/09/09	2024/09/13
Conduct sensitivity analysis to understand the robustness o...	Bhavika Prasannakumar	100%	...	2024/09/14	2024/09/19
Model Performance Evalution	Shashank Shashishekhar Reddy	100%	...	2024/09/19	2024/09/30
Deployment	Sai Srivathsav Aripirala	100%	...	2024/10/01	2024/11/20
Development of User friendly web portal	Shashank Shashishekhar Reddy	100%	...	2024/10/01	2024/10/17
Deploy the development models and web portal	Sai Srivathsav Aripirala	100%	...	2024/10/17	2024/10/24
Creating the dashboards	Chinmaya Gayathri	100%	...	2024/10/25	2024/10/31
Monitor the performance	Prudhvi Chowdary Chirumamilla	100%	...	2024/10/27	2024/11/10
Documenting the Bug and errors and project review	Bhavika Prasannakumar	100%	...	2024/10/30	2024/11/08
Document all phases of the data analysis process	Prudhvi Chowdary Chirumamilla	100%	...	2024/11/10	2024/11/20

The process of CRISP-DM is divided into several stages as shown in Figure 4, Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. Thorough documentation is necessary at these stages to record insights, procedures, and outcomes. Project managers and team members may plan, arrange, and monitor documentation work more efficiently with the use of a Gantt chart. There are main tasks added as the stages in the CRISP-DM methodology. Sub tasks follow right below each stage.

Business Understanding. A Gantt chart can show the timeline for obtaining preliminary project requirements, conducting stakeholder interviews, and creating project objectives during the Business Understanding phase as shown in Figure 5. This guarantees that the documentation complies with the original project objectives. Subtasks include defining the project scope and requirements, literature survey, defining the project statement, technology survey and finally creating the project plan. Each subtask is planned to be completed anywhere between 2 to 6 days.

Figure 5

Business Understanding Gantt chart



Data Understanding and Data Preparation. Documenting data sources, data cleansing procedures, and data transformations is essential during the phases of data understanding and preparation. Time can be set aside on a Gantt chart for data cleaning, data profiling, and data preparation for modeling. For the Data Understanding and Preparation phase, subtasks are mainly revolving around identifying the data source, having a data collection plan, understanding, and exploring more about the data and finally storing the data using AWS Cloud services. Data Preparation, however, contains exploratory data analysis, integrating diverse datasets into a unified format, feature engineering and splitting the data into training, test and validation sets for further analysis as shown in Figure 6 and Figure 7.

Figure 6

Data Understanding Gantt chart

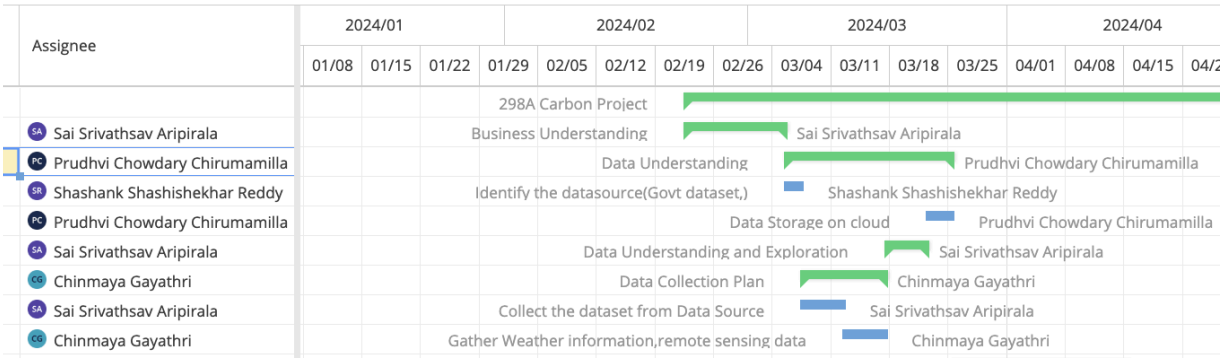
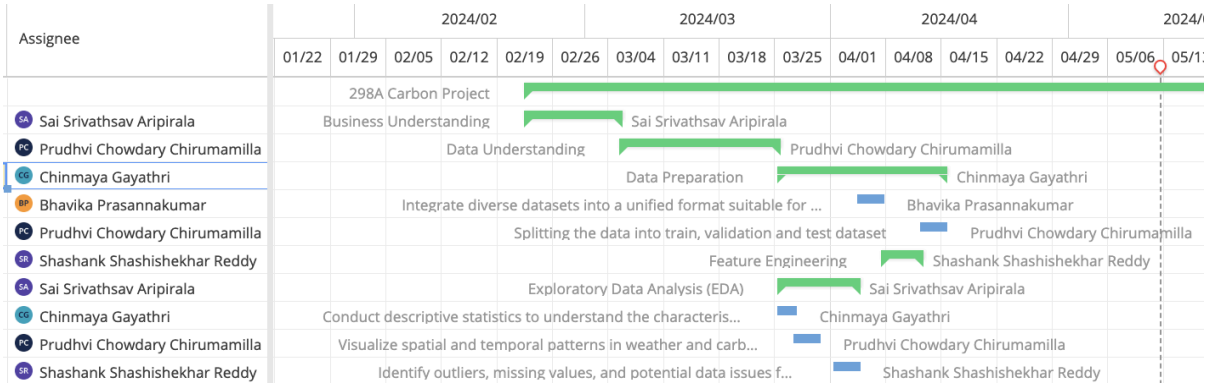


Figure 7

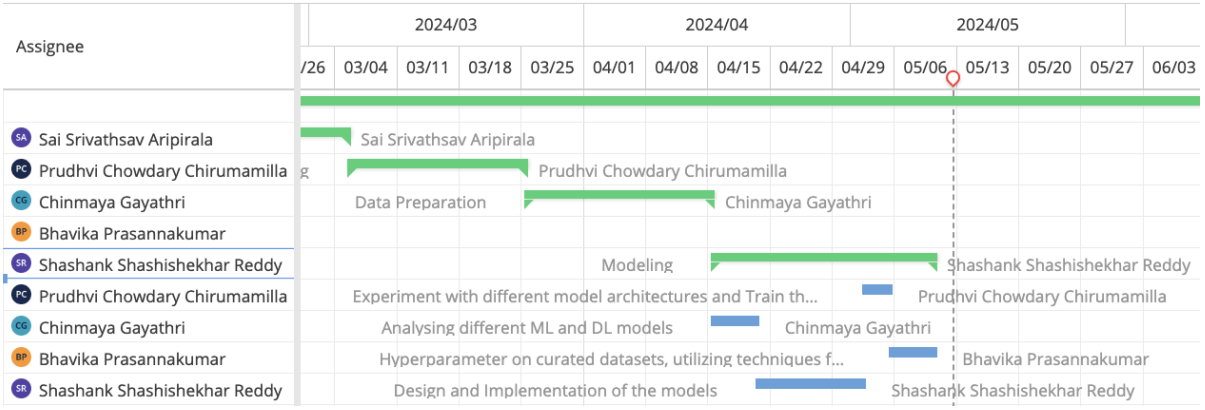
Data Preparation Gantt chart



Modeling. Documenting the process of developing, tuning, and choosing algorithms is part of the modeling step. When these tasks need to be finished and when documentation on the selected models and their performance metrics needs to be made may both be plainly seen on the Gantt chart. Modeling consists of analyzing different ML and DL models, designing and implementation of the models, experimenting with different model architectures and hyperparameter tuning as shown in Figure 8.

Figure 8

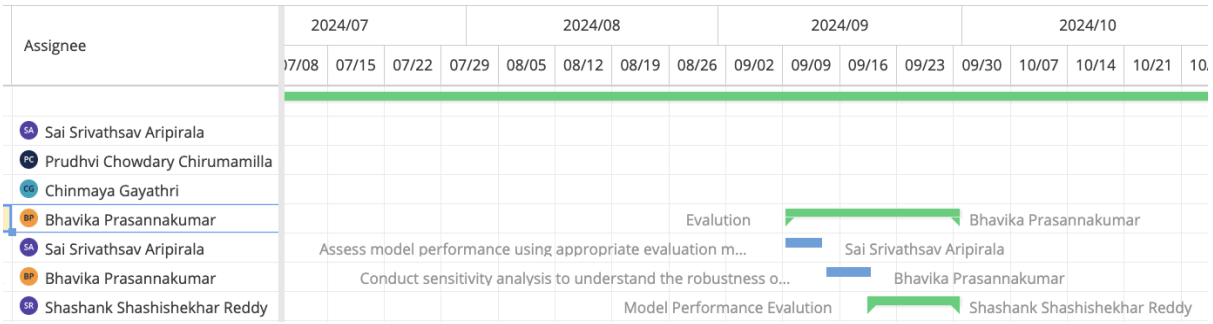
Modeling Gantt chart



Evaluation. Model performance is evaluated during the assessment phase, and the timing and format of this evaluation can be specified using a Gantt chart. This could involve ROC curves, confusion matrices, and accuracy evaluations. Subtasks for evaluation goes more into assessing model performance using different methods, conducting sensitivity analysis and evaluating the model performance as shown in Figure 9.

Figure 9

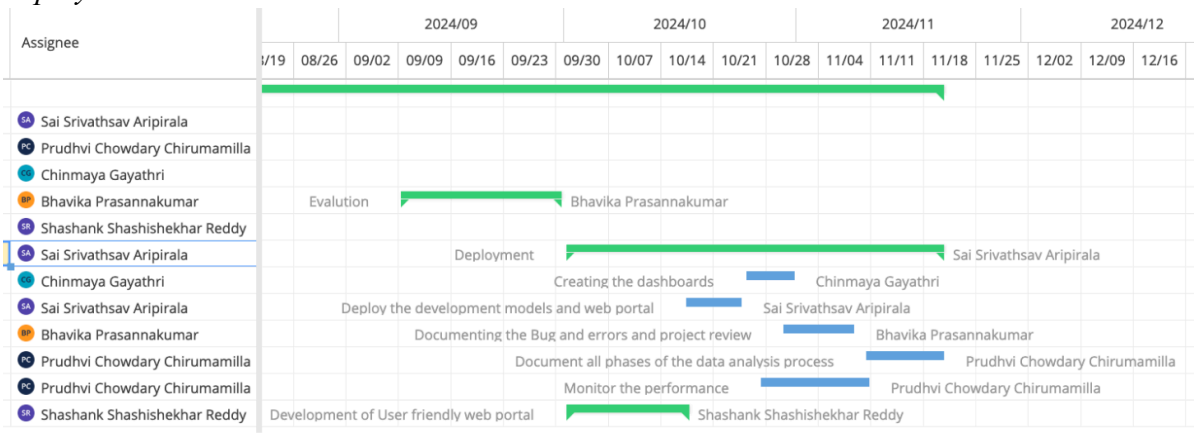
Evaluation Gantt chart



Deployment. The Gantt chart can be used to schedule the preparation of documentation for model deployment protocols, maintenance schedules, and stakeholder insights sharing during the deployment phase. Subtasks for the deployment phase deals with the deployment of a user-friendly web portal, deploying the developed models, creating the dashboard, monitoring its performance, finally documenting the bugs and all phases of the data analysis process as shown in Figure 10.

Figure 10

Deployment Gantt chart



Pert Chart

A PERT (Program Evaluation and Review Technique) chart is a crucial instrument in project management, intended to visually represent and analyze the various tasks, their connections, and the critical path inside a project. Originally designed to handle the intricacies of projects including overlapping tasks, it helps project managers define the flow of activities, predict the project schedule, and identify the crucial sequence that determines the minimum project duration.

Tasks/Activities. Every important action, from gathering data to deploying the model, is represented by a separate node in our PERT chart. Throughout the project lifecycle, these tasks are methodically recognized and designated with a unique code, creating a clear framework of reference.

Arrows/Sequences. These nodes are connected by directed arrows, which specify the exact sequence of events. These connections are more than just identifiers; they are the road map that shows which tasks are necessary to complete before moving on to the next, clearly defining the project's workflow.

Estimated Duration. An important part of our PERT chart is the projected duration for every task. We develop a range that includes optimistic, likely, and pessimistic durations

to account for the inherent uncertainty in complex projects and ultimately help to determine a realistic timeline for project completion.

Critical Path. The foundation of the project's timeline is the critical path shown in our PERT graphic. The order in which tasks are completed directly affects the minimal project length. The project as a whole will suffer from any delays in this sequence, which emphasizes the need of strategic planning and swift completion of these activities.

A pert chart for Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning is shown in Figure 11.

Figure 11

Pert Chart

Pert Chart Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning

3. Data Engineering

3.1 Data Process

The study, situated within Redwood National and State Parks, California, employed advanced satellite technology to estimate the forest's aboveground biomass (AGB) – a significant metric for carbon storage. This research bypassed traditional, time-intensive methods like ground-based assessments and leveraged the strengths of two complementary satellite instruments: The Moderate Resolution Imaging Spectroradiometer (MODIS) and the Global Ecosystem Dynamics Investigation (GEDI). MODIS excels at capturing frequent observations, providing a high temporal resolution. This characteristic makes it ideal for monitoring changes in the forest health and vegetation over time. GEDI, in contrast, excels at capturing the three-dimensional structure of the forest with high fidelity. By synergistically combining these datasets, researchers were able to compensate for the limitations of each instrument and create a more detailed picture of the forest's AGB.

The satellite-based approach offers a significant advantage over traditional methods. Its scalability and efficiency allow scientists to analyze vast areas without extensive fieldwork, facilitating more comprehensive assessments of forest carbon sequestration on a regional and global scale. Understanding forest carbon sequestration is crucial in the fight against climate change, as forests play a vital role in storing carbon dioxide. Furthermore, the fusion of GEDI and MODIS data has the potential to improve our understanding of forest biodiversity and habitat suitability by providing insights into forest composition, vertical complexity, and canopy cover. This information can be particularly valuable for conservation efforts and sustainable forest management practices, allowing for targeted interventions to protect biodiversity and critical habitats.

The datasets are partitioned into training, validation, and test sets maintaining representativeness of the overall data distribution across each set is critical to mitigate bias during model training and evaluation. Machine learning model selection hinges on the

research objectives and dataset characteristics. The chosen model undergoes training using the training dataset, followed by evaluation using the validation dataset. Hyperparameter tuning and model optimization are then conducted to achieve the desired level of accuracy and generalizability. Model performance is ultimately assessed using the independent test dataset, this evaluation gauges the model's ability to generalize to unseen data, additionally validating its robustness and identifying any potential limitations. The iterative nature of the machine learning process allows for continuous refinement based on the insights gleaned from each iteration. This iterative approach fosters the improvement of both model accuracy and dataset effectiveness in addressing the established research objectives.

Table 11

The data products collected for MODIS and GEDI.

Satellite	Data Products	Important Variables	File Type	File Size (GB)	Resolution
GEDI	GEDI L2A	Elevation metrics	HDF5	173	Daily at 25m
GEDI	GEDI L2B	Canopy Height	HDF5	38	Daily at 25m
GEDI	GEDI L3	LSM	Tiff	10	Daily at 1km
GEDI	GEDI L4B	ABG	Tiff	2	Daily at 1km
MODIS	MOD11A1	LST	HDF4	0.5	Daily at 1km
MODIS	MOD13A2	NDVI, EVI	HDF4	0.23	8 days at 1km
MODIS	MOD15A2H	LIA	HDF4	0.2	16 days at 1km
MODIS	MOD17A2	GPP/NPP	HDF4	0.2	Daily at 500m

The data collected and its Spatial and Temporal resolution is given in Table 11, we can see that most of the variables are collected daily at 1km. To get the data at the same level for MOD13A2 and MOD15A2H, the data will be duplicated for 8 and 16 days respectively.

3.2 Data Collections

The study area for this research is located within and surrounding Redwood National and State Parks, California, USA. This park system encompasses over 139,000 acres along the northern Californian coast, stretching from approximately 41.19°N to 41.79°N latitude and 123.46°W to 124.43°W longitude [23]. The landscape is dominated by coastal redwood forests, known for their towering stature and dense canopy cover. Redwood National and State Parks experience a cool-summer Mediterranean climate with moist, fog-influenced summers and mild, wet winters. Average annual rainfall varies depending on location but can exceed 200 cm (79 inches) [24]. Soils within the park are generally acidic and well-drained, with textures ranging from clay loam to rocky outcroppings [25]. The research timeframe for this study spanned from January 1st, 2021, to December 31st, 2023. The data collected for a MODIS data product named MOD17A2, this product was recently added to the MODIS suite on January 1st of 2021, which offers valuable insights into terrestrial ecosystem dynamics by providing estimates of Gross Primary Productivity (GPP) and Net Primary Productivity (NPP).

In the field of forest carbon sequestration, traditional methodologies have predominantly relied on ground-based assessments such as plot inventories and destructive sampling techniques [26]. While these approaches provide highly accurate data for specific locations, they are inherently time-consuming, labor-intensive [27], and limited in their ability to scale across vast landscapes [28]. To address these limitations, recent advancements in remote sensing technology have introduced a powerful digital alternative: leveraging satellite data for forest biomass assessment and carbon stock estimation [29]. High-resolution lidar sensors, in particular, offer the capability to create three-dimensional characterizations of forest canopies, while multispectral imagery facilitates the spectral analysis of vegetation health and composition [30]. By synergistically combining these datasets, researchers can develop robust models for estimating aboveground biomass, a critical component of forest

carbon storage [31]. The scalability and efficiency of satellite-based approaches hold immense promise for revolutionizing forest carbon monitoring efforts and enabling more comprehensive assessments of global carbon sequestration dynamics [32].

The integration of spaceborne lidar and multispectral data from the Global Ecosystem Dynamics Investigation (GEDI) and Moderate Resolution Imaging Spectroradiometer (MODIS) instruments is revolutionizing forest aboveground biomass (AGB) estimation. GEDI, a pioneering lidar system, directly measures forest canopy height and vertical structure, providing critical parameters that exhibit strong correlations with AGB, GEDI is strategically positioned on the ISS, orbiting Earth at an altitude of approximately 420 kilometers (260 miles). Its orbital path is inclined at 51.6 degrees, ensuring coverage between 51.6° North and 51.6° South latitudes. This inclination allows GEDI to observe a significant portion of the world's terrestrial ecosystems, encompassing tropical rainforests, temperate and boreal forests, and even arid environments. The ISS completed approximately 16 orbits around Earth every day. During each orbit, GEDI collects laser ranging measurements along its ground track, effectively creating a series of laser swaths across the Earth's surface. These swaths consist of eight individual laser beams, providing a comprehensive and dense sampling of the terrain below.

However, its spatial coverage is limited. MODIS, on the other hand, offers high temporal resolution and extensive spectral coverage across broad landscapes, Terra and Aqua follow near-polar, sun-synchronous orbits, meaning they circle Earth approximately 14 times a day, always crossing the equator at roughly the same local solar time. Terra's orbit has a slightly lower altitude (705 km) and a faster ground track compared to Aqua (709 km). This dual-satellite configuration ensures near-daily global coverage, allowing MODIS to capture dynamic changes on Earth's surface over time. Since MODIS is fixed to the satellites and doesn't actively scan, it captures data in a swath beneath the spacecraft's path. These swaths have a width of approximately 2,330 kilometers (1,448 miles), effectively providing daily

global coverage at moderate resolution. This allows scientists to analyze variations in vegetation health, fractional cover, and leaf area index (LAI) – all indirectly linked to AGB – but lacks the detailed structural information captured by lidar. By synergistically combining these datasets, researchers can leverage the strengths of each instrument to overcome their limitations. GEDI's high-fidelity structural measurements can be used to calibrate and validate AGB models derived from MODIS's extensive biophysical characterization. This integrated approach allows for the development of robust, spatially explicit AGB maps at regional and global scales. This information is critical for monitoring deforestation impacts, improving our understanding of forest carbon dynamics within the global carbon cycle, and informing the development of effective forest management strategies for mitigating climate change. Furthermore, the fusion of GEDI and MODIS data has the potential to improve our understanding of forest biodiversity and habitat suitability by providing insights into forest composition and vertical complexity. This can be particularly valuable for conservation efforts and sustainable forest management practices.

The Land Processes Distributed Active Archive Center (LP DAAC) acts as a critical link within NASA's Earth Observing System Data and Information System (EOSDIS). Hosted by the U.S. Geological Survey (USGS) Earth Resources Observation and Science (EROS) Center, the LP DAAC focuses on archiving and distributing NASA's land process data products. This data is crucial for understanding the health and functioning of our planet's ecosystems. For researchers seeking this information, LP DAAC integrates seamlessly with NASA's Earthdata search portal. Earthdata acts as a central hub, allowing users to discover, explore, and download a wide range of Earth science data, including the land process data archived by LP DAAC. This combined functionality ensures streamlined access to valuable scientific data for researchers investigating Earth's land systems.

Leveraging the wealth of Earth science data archived within the Earth Observing System Data and Information System (EOSDIS), this study incorporated data from both the

Moderate Resolution Imaging Spectroradiometer (MODIS) and the Global Ecosystem Dynamics Investigation (GEDI) instrument. To access these publicly available datasets, user registration on the Earthdata portal was the initial step. Following registration, the data retrieval strategy differed based on the specific characteristics of each dataset and the research objectives.

For MODIS imagery, the extensive temporal coverage required for the analysis necessitated an efficient data acquisition approach. The modis-tools Python library offered a well-suited solution. This user-friendly library facilitated programmatic download of MODIS data directly from the NASA Earthdata platform. By automating the download process, modis-tools streamlined data acquisition for the large volume of MODIS imagery needed for this study. In contrast, the data retrieval strategy for GEDI data employed a different approach. Due to the well-defined spatial and temporal constraints of the study area, the Earthdata search and download functionality provided a more efficient solution. By creating a project adhering to Earthdata Norms, we were able to leverage the platform's capabilities to generate an auto-generated script. This script tailored the GEDI data acquisition process to our specific needs, ensuring that only the data relevant to our spatial and temporal boundaries were downloaded. This approach optimized data retrieval for GEDI and minimized unnecessary data acquisition.In Figure 12, its shown how the modis_tools are used to authenticate and download the MODIS data for the start_date and end_date of 2021/01/01 and 2023/12/31, and the boundary_box which is given as Redwood National Park and the data is downloaded into download_path.

Figure 12

Function to get_data from the MODIS

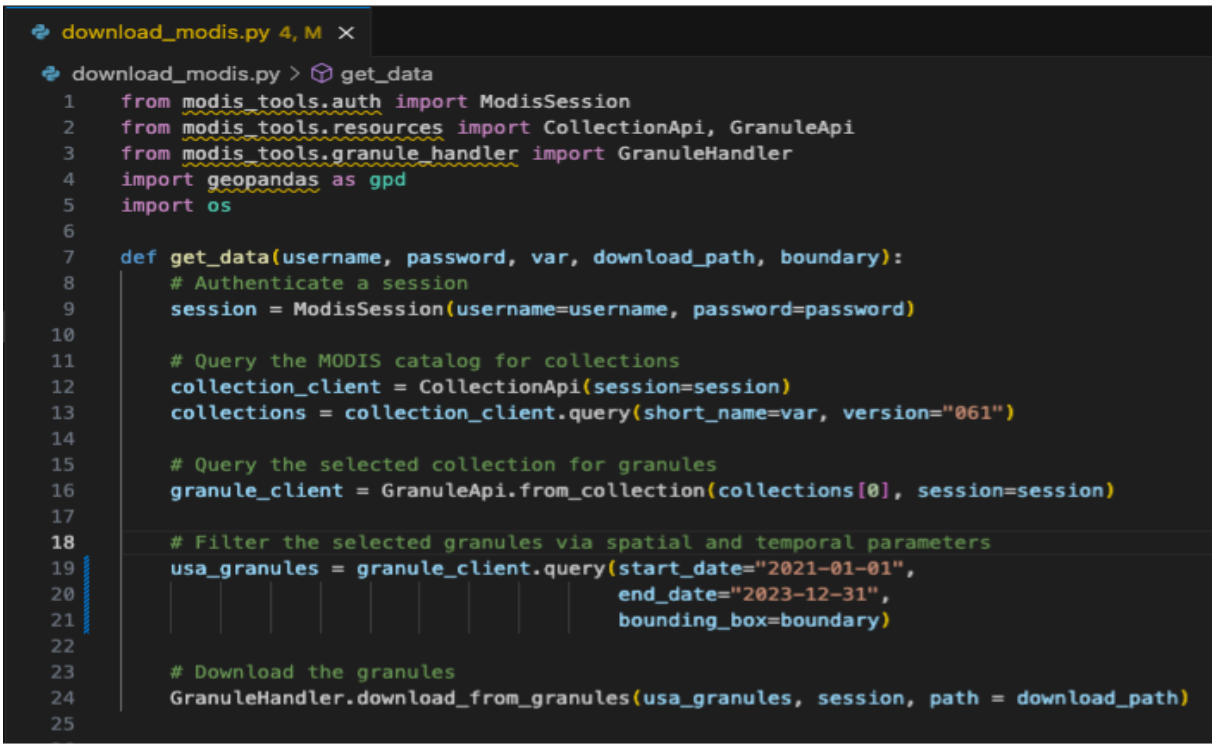


Figure 13 shows the main function where username and passwords are extracted from the os.get_env(), and for each data product from MODIS, get_data is called and retrieved data is stored in the specified location.

Figure 13

Main Function to run the get_data function

```
if __name__ == "__main__":

    username = os.get_env("earthdata_username")
    password = os.get_env("earthdata_password")

    geojson_file = "boundaries/RedwoodNP.geojson"
    boundary = gpd.read_file(geojson_file).geometry[0]

    vars = ['MOD11A1', 'MOD13A2', 'MOD15A2H', 'MOD17A2H']

    for var in vars:
        download_path = "data/MODIS/"+var
        if os.path.exists(download_path):
            print(f"Getting {var} data")
            get_data(username, password, var, download_path, boundary)

        else:
            print(f"Creating data/MODIS/{var} directory")
            os.mkdir(download_path)
            print(f"Getting {var} data")
            get_data(username, password, var, download_path, boundary)
```

To extract the GEDI data, we first had to filter out the data which is outside of the research date i.e., 01/01/2021 to 12/31/2023, Figure 14 shows the portal filter option to put in dates for the required data, next we will add the redwoodNationalPark.geoJSON file to filter the data which comes under our research area, this is shown in Figure 15. Once we select the satellite data we are interested in, GEDI L2A, L2B, L3 and L4B, we are ready with the data as shown in Figure 16. For each of the data products you have to go into the download script tab to see the auto-generated bash script as shown in Figure 17 to download the files that fall under the specified temporal and spatial constraints.

Figure 14

Filtering out data that is collected outside of our temporal Attribute

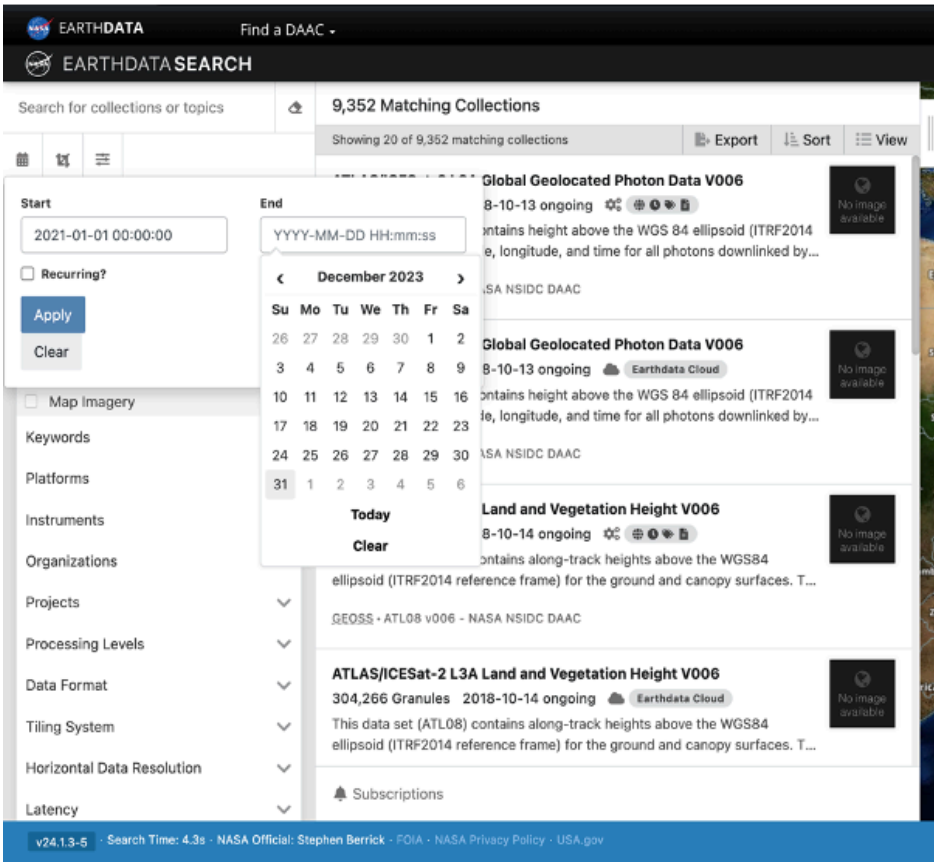


Figure 15

Filtering data that is collected within Redwood National Park

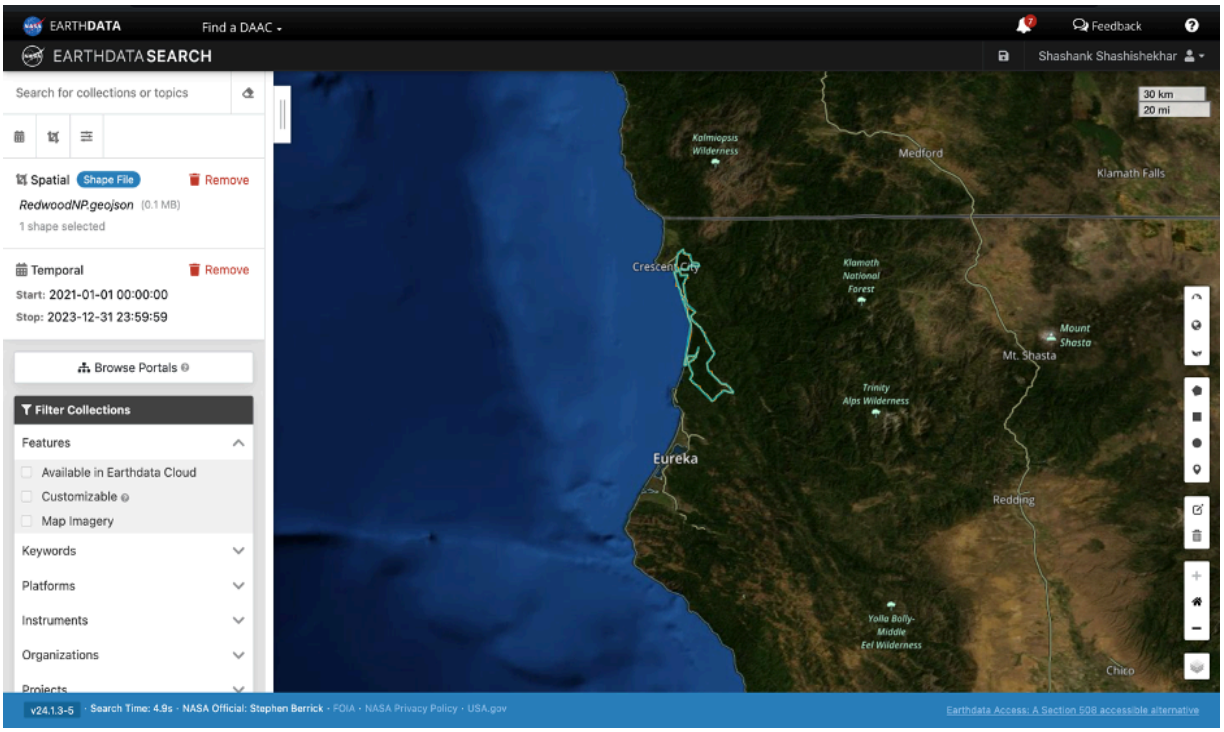


Figure 16

Data that is ready to be downloaded

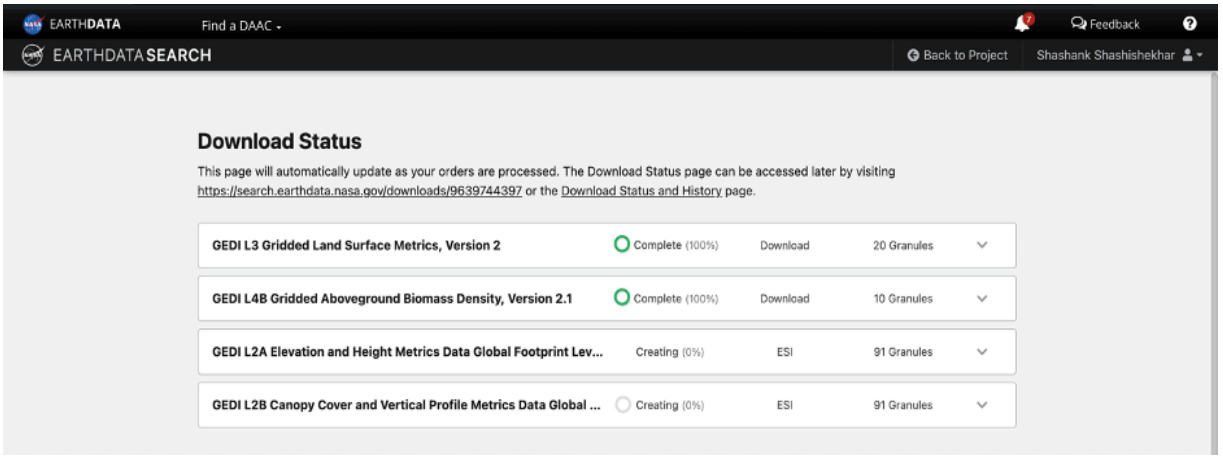
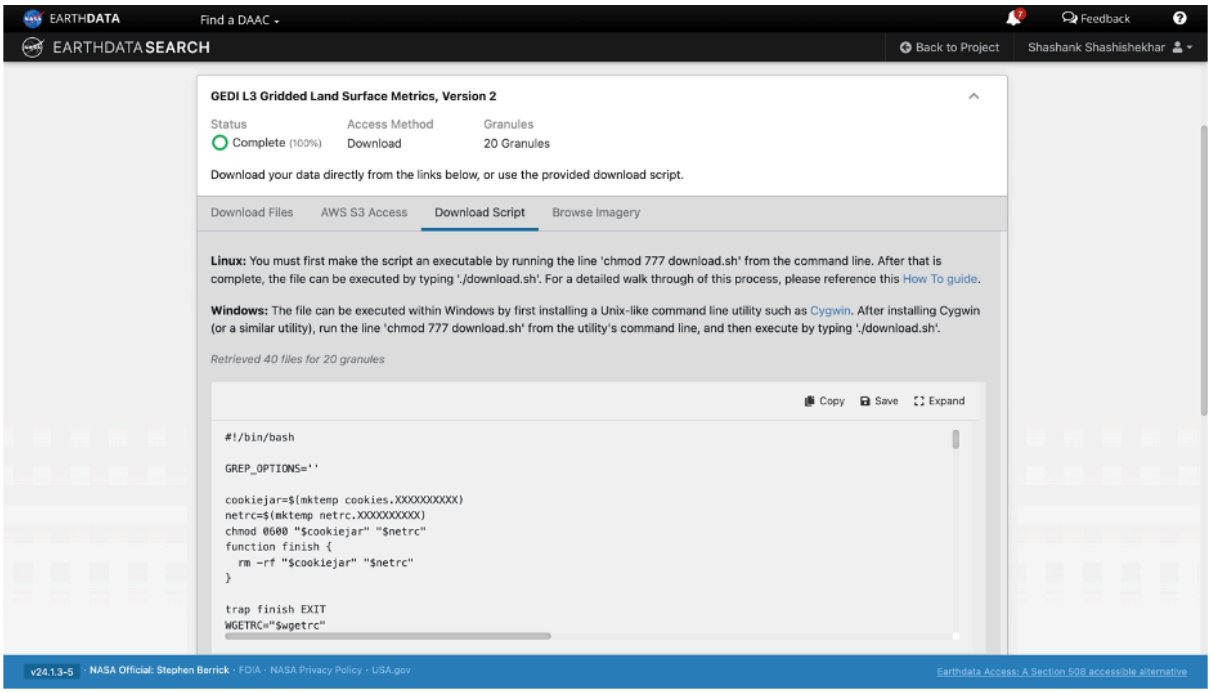


Figure 17

Auto generated bash script to download the data



Sample raw data for MODIS data products after data collection are given below, figure 18 has the sample raw data for date 2021-01-03 for MOD11A1 and MOD13A2.

Figure 18

Raw data for MOD11A1 (left) and MOD13A2 (right)

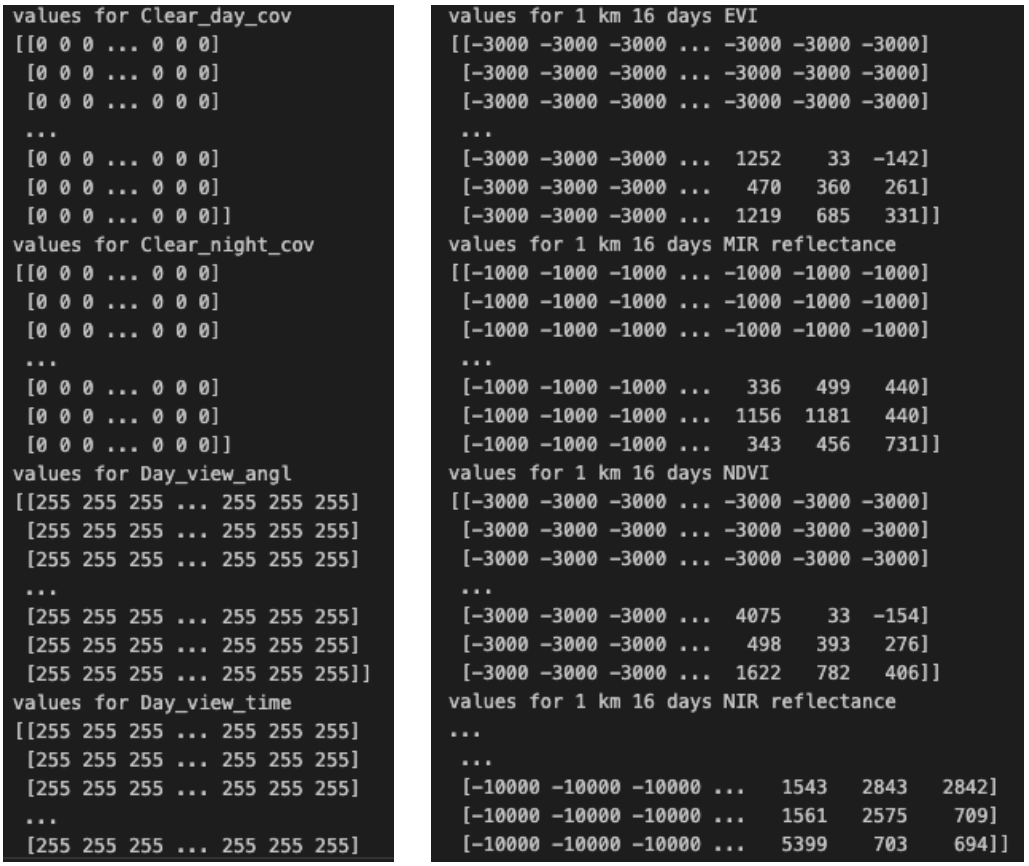
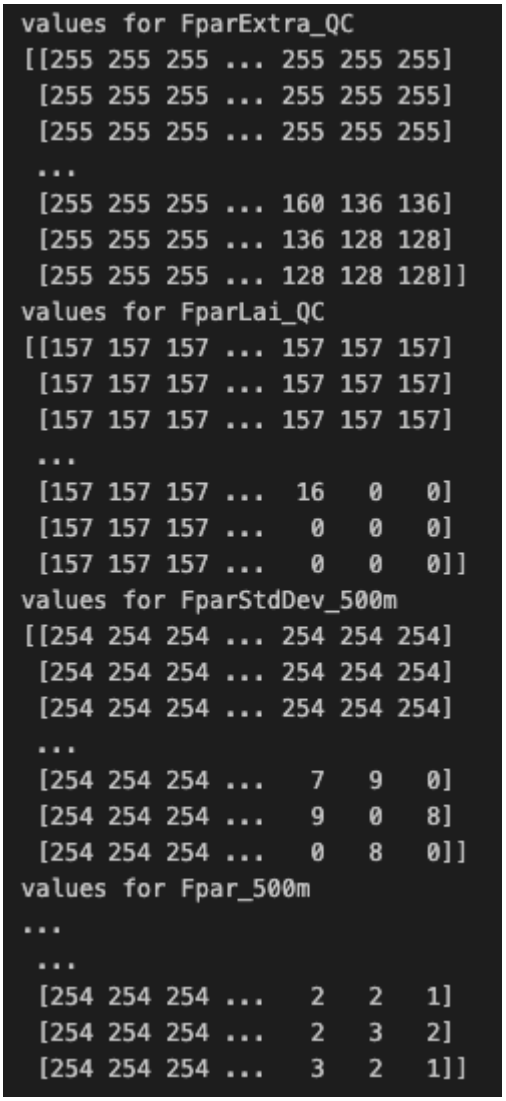
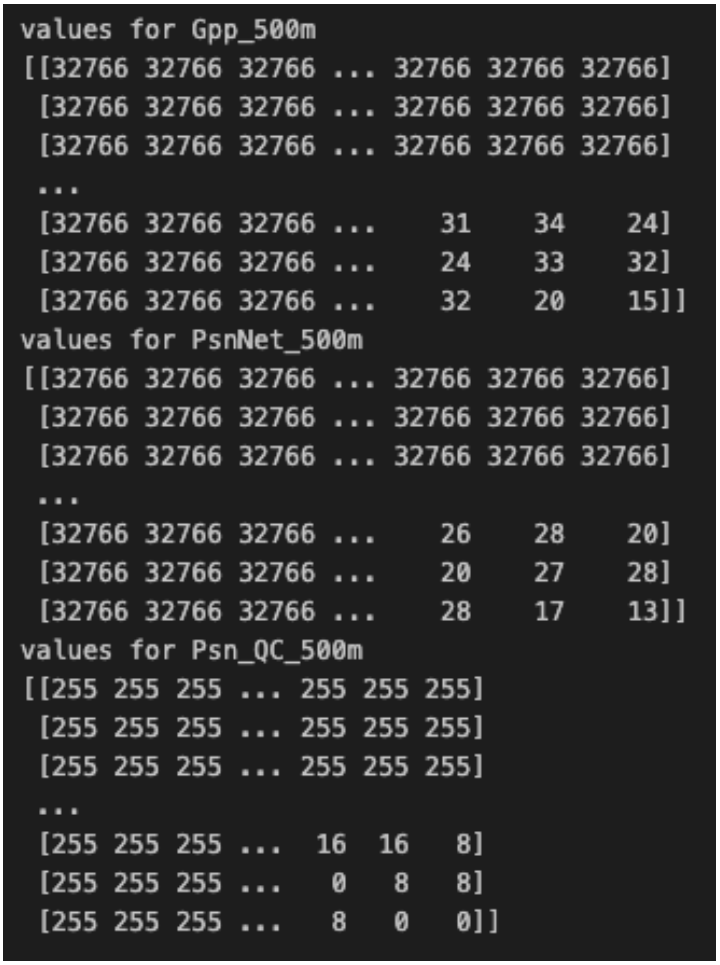


Figure 19 has sample raw data for MOD15A2H and MOF17A2H for the date 2021-01-12.

Figure 19

Raw data for MOD15A2H (left) and MOD17A2 (right)





Type of data collected for Gedi L2A

Sample raw data for Gedi L2A data products after data collection are given below, figure 20 has the sample raw data for date 2021-01-03 for Gedi L2A

Figure 20

Sample Data for GEDI 2LA data

	Beam	Shot Number	Longitude	Latitude	Quality Flag
0	BEAM0110	29320618800000001	111.996300	-51.803868	0
1	BEAM0110	29320604600000101	112.039132	-51.803905	0
2	BEAM0110	29320614600000201	112.080271	-51.803836	0
3	BEAM0110	29320600400000301	112.121445	-51.803737	0
4	BEAM0110	29320610400000401	112.162622	-51.803621	0
...	...	...	...	...	...
9792	BEAM0110	29320617400979201	88.208452	-51.803578	0
9793	BEAM0110	29320603200979301	88.249610	-51.803614	0
9794	BEAM0110	29320613200979401	88.290753	-51.803581	0
9795	BEAM0110	29320623200979501	88.331913	-51.803548	0
9796	BEAM0110	29320609000979601	88.373089	-51.803506	0

9797 rows × 5 columns

3.3 Data Preprocessing

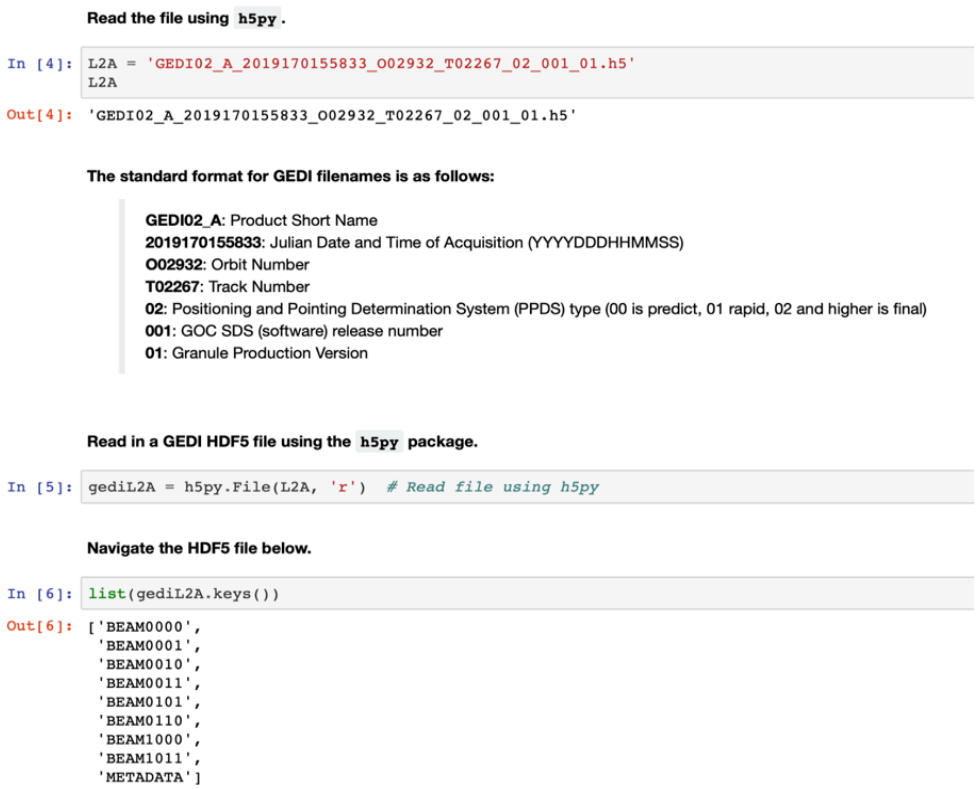
Preparing the data so that it can be used to train a machine learning model is known as data pre-processing. When an ML model is trained using a data set that contains anomalies, it frequently produces incorrect results. Consequently, it is necessary to eliminate this disparity, and data preprocessing is utilized to accomplish so. There are different types of GEDI data we have considered for our raw dataset.

GEDI L2A

We used the GEDI Level 2A (L2A) data package from NASA's Earth Observing System Data and Information System (EOSDIS) to start our investigation into the biomass and structure of forests. For the purpose of conducting environmental and ecological investigations, precise measurements of canopy height and vertical structural metrics must be obtained using GEDI L2A data.

Figure 21

Open a GEDI HDF5 file and Read file metadata for L2A



Read the file using `h5py`.

In [4]:
```
L2A = 'GEDI02_A_2019170155833_O02932_T02267_02_001_01.h5'
L2A
```

Out[4]:
```
'GEDI02_A_2019170155833_O02932_T02267_02_001_01.h5'
```

The standard format for GEDI filenames is as follows:

> **GEDI02_A**: Product Short Name
> **2019170155833**: Julian Date and Time of Acquisition (YYYYDDDHHMMSS)
> **O02932**: Orbit Number
> **T02267**: Track Number
> **02**: Positioning and Pointing Determination System (PPDS) type (00 is predict, 01 rapid, 02 and higher is final)
> **001**: GOC SDS (software) release number
> **01**: Granule Production Version

Read in a GEDI HDF5 file using the `h5py` package.

In [5]:
```
gediL2A = h5py.File(L2A, 'r')  # Read file using h5py
```

Navigate the HDF5 file below.

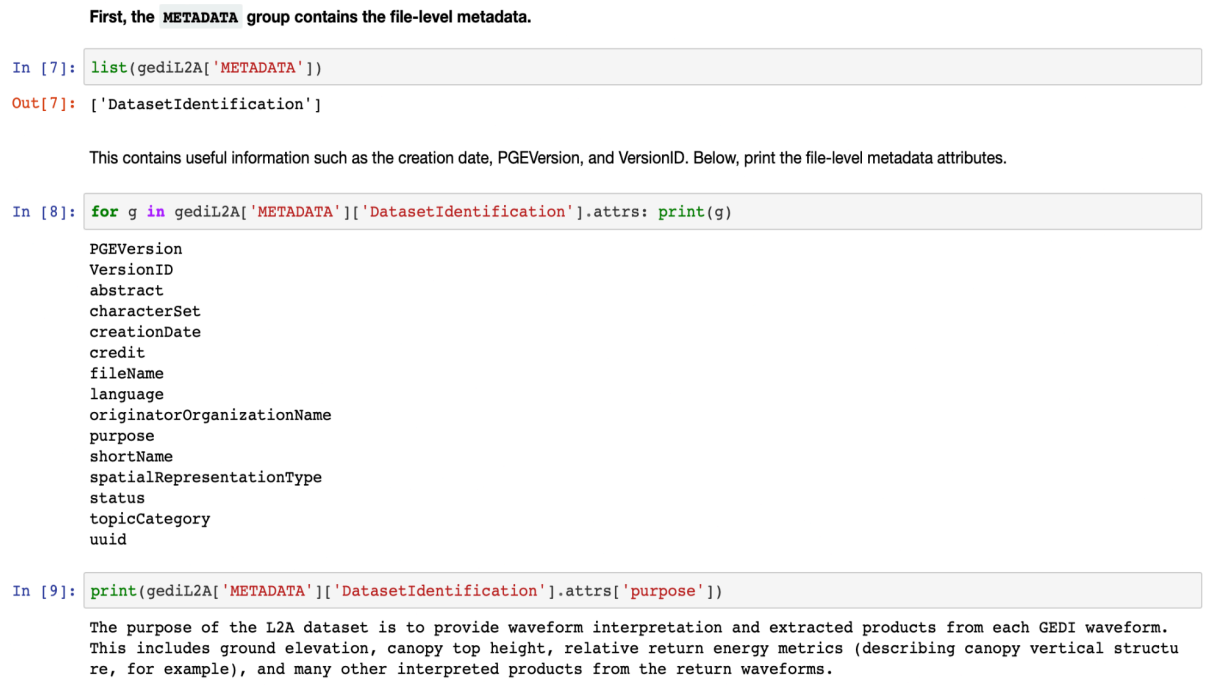
In [6]:
```
list(gediL2A.keys())
```

Out[6]:
```
['BEAM0000',
 'BEAM0001',
 'BEAM0010',
 'BEAM0011',
 'BEAM0101',
 'BEAM0110',
 'BEAM1000',
 'BEAM1011',
 'METADATA']
```

In this chapter, we initiate the groundwork for processing the GEDI Level 2A Elevation and Height Metrics Data. The first step is to open a GEDI HDF5 file and extract its metadata to gain insights into its contents. We start by reading the example L2A file metadata using the `h5py` package [33], which facilitates the handling of HDF5 files as shown in figure 22. The GEDI HDF5 file structure comprises various groups, including 'BEAM' groups representing individual laser beams and a 'METADATA' group containing file-level metadata as shown in figure 22. Within the 'METADATA' group, essential information such as creation date, version, and purpose of the dataset is stored. For instance, the purpose of the L2A dataset is to provide waveform interpretation and extracted products, including ground elevation and canopy top height.

Figure 22

The GEDI HDF5 file contains groups in which data and metadata are stored



First, the `METADATA` group contains the file-level metadata.

```
In [7]: list(gediL2A['METADATA'])
Out[7]: ['DatasetIdentification']
```

This contains useful information such as the creation date, PGEVersion, and VersionID. Below, print the file-level metadata attributes.

```
In [8]: for g in gediL2A['METADATA']['DatasetIdentification'].attrs: print(g)
PGEVersion
VersionID
abstract
characterSet
creationDate
credit
fileName
language
originatorOrganizationName
purpose
shortName
spatialRepresentationType
status
topicCategory
uuid

In [9]: print(gediL2A['METADATA']['DatasetIdentification'].attrs['purpose'])
The purpose of the L2A dataset is to provide waveform interpretation and extracted products from each GEDI waveform.
This includes ground elevation, canopy top height, relative return energy metrics (describing canopy vertical structu
re, for example), and many other interpreted products from the return waveforms.
```

Moving forward, we delve into the SDS (Scientific Data Sets) metadata to focus on specific beam transects. The GEDI instrument consists of three lasers producing eight beam ground transects, each categorized as either a 'Coverage beam' or a 'Full power beam' as shown in figure 23. This distinction is crucial for subsequent analysis. After identifying a full power beam for further analysis, we proceed to extract relevant datasets from the HDF5 file. This step involves identifying all objects within the file and selecting datasets corresponding to the chosen beam transect. By effectively preprocessing the data and organizing it into manageable subsets, we lay the foundation for subsequent steps, including visualization and analysis. This meticulous preparation ensures that the GEDI data is primed for in-depth exploration and interpretation in later sections.

Figure 23

Read Metadata and subset by Beam

```
for b in beamNames:
    print(f"{b} is a {gediL2A[b].attrs['description']}")
```

```
BEAM0000 is a Coverage beam
BEAM0001 is a Coverage beam
BEAM0010 is a Coverage beam
BEAM0011 is a Coverage beam
BEAM0101 is a Full power beam
BEAM0110 is a Full power beam
BEAM1000 is a Full power beam
BEAM1011 is a Full power beam
```

Below, pick one of the full power beams that will be used to retrieve GEDI L2A relative height metrics in Section 3.

```
beamNames = ['BEAM0110']
```

Identify all the objects in the GEDI HDF5 file below.

Note: This step may take a while to complete.

```
gediL2A_objs = []
gediL2A.visit(gediL2A_objs.append)                                    # Retrieve list of datasets
gediSDS = [o for o in gediL2A_objs if isinstance(gediL2A[o], h5py.Dataset)]  # Search for relevant SDS inside data fil
e
[i for i in gediSDS if beamNames[0] in i][0:10]                       # Print the first 10 datasets for selecte
d beam
```

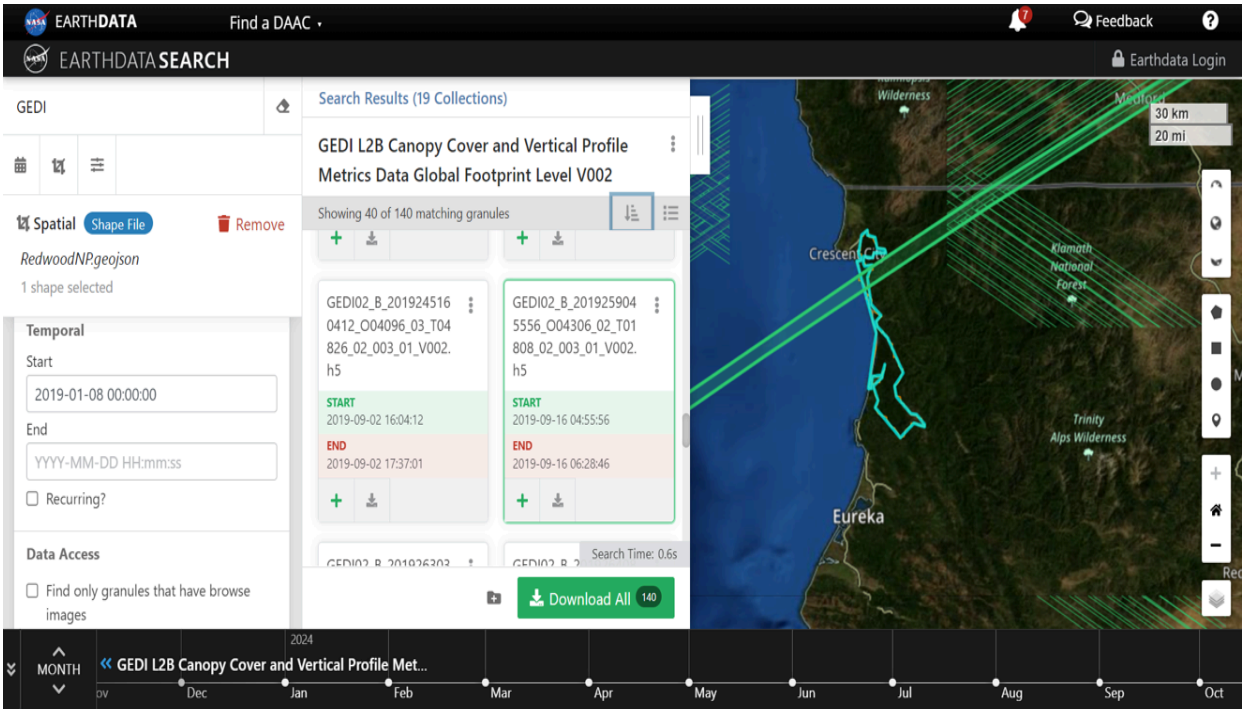
```
['BEAM0110/ancillary/l2a_alg_count',
 'BEAM0110/beam',
 'BEAM0110/channel',
 'BEAM0110/degrade_flag',
 'BEAM0110/delta_time',
 'BEAM0110/digital_elevation_model',
 'BEAM0110/elev_highestreturn',
 'BEAM0110/elev_lowestmode',
 'BEAM0110/elevation_bias_flag',
 'BEAM0110/elevation_bin0_error']
```

GEDI L2B

EarthData has satellite data which is free for non- commercial use. We fetched the GEDI L2B, Gedi data from the data catalog. We followed a series of steps outlined by NASA's Earth Observing System Data and Information System (EOSDIS). Initially, we navigated to the Earthdata Search portal and input specific search criteria such as the desired data product (GEDI L2B), temporal coverage, geographic region (e.g., Redwood Forest), and other relevant parameters. Upon submitting the search query, the portal provided a list of available datasets meeting the specified criteria. We selected the appropriate dataset and initiated the download process. Due to the large file sizes inherent in GEDI L2B data, we encountered challenges in handling and processing the data efficiently. As a result, we opted to focus solely on a single date and a specific region of interest to streamline our analysis as shown in figure 24.

Figure 24

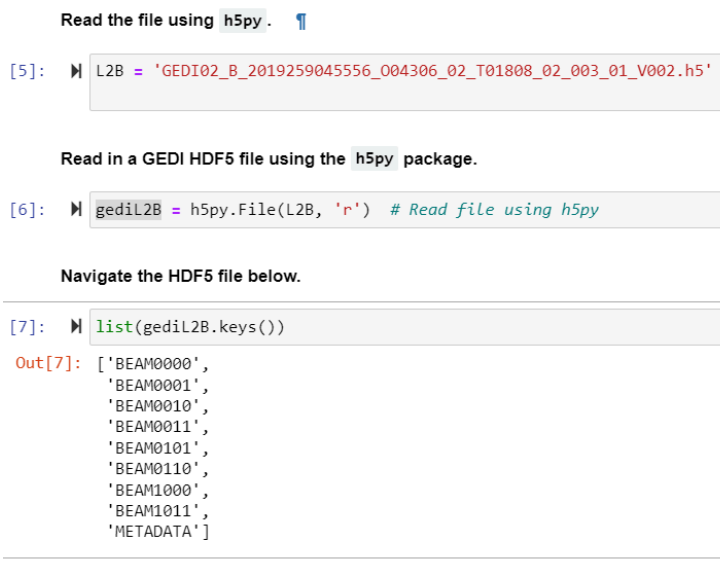
Filtering data that is collected within Redwood National Park



After downloading the data to the drive, we began the data preprocessing by defining the filename of the GEDI HDF5 file, which is commonly referred to as 'L2B'. This filename is crucial as it serves as the key identifier for the dataset we want to analyze. Following this, we utilized the 'h5py.File()' function, a powerful tool for working with HDF5 files in Python. By passing the filename and specifying the read mode ('r') as shown in figure 25, we opened the HDF5 file, granting me access to its contents. This step is essential as it lays the foundation for subsequent data exploration and analysis. Finally, I stored the resulting file object in a variable named 'gediL2B', which I'll use throughout my notebook to reference and manipulate the dataset.

Figure 25

Open a GEDI HDF5 file and Read file metadata for L2B



After opening the file, we started to understand its structure. It does this by listing the various groups or sections contained within the HDF5 file. These groups serve as organizational units, with each group potentially holding specific types of data. In this case, the groups identified are related to different GEDI beam transects (like 'BEAM0000', 'BEAM0001', etc.) and a group dedicated to metadata ('METADATA') as shown in figure 26.

Figure 26

The GEDI HDF5 file contains groups in which data and metadata are stored.

First, the `METADATA` group contains the file-level metadata. ¶

```
list(gediL2B['METADATA'])
```

```
['DatasetIdentification']
```

This contains useful information such as the creation date, PGEVersion, and VersionID. Below, print the file-level metadata attributes.

```
for g in gediL2B['METADATA']['DatasetIdentification'].attrs:
    print(g)
```

```
PGEVersion
VersionID
abstract
characterSet
creationDate
credit
fileName
language
originatorOrganizationName
purpose
shortName
spatialRepresentationType
status
topicCategory
uuid
```

```
print(gediL2B['METADATA']['DatasetIdentification'].attrs['purpose'])
```

```
The purpose of the L2B dataset is to extract biophysical metrics from each GEDI waveform. These metrics are based on the dir
ectional gap probability profile derived from the L1B waveform and include canopy cover, Plant Area Index (PAI), Plant Area
Volume Density (PAVD) and Foliage Height Diversity (FHD).
```

The GEDI HDF5 file contains groups in which data and metadata are stored for L2B

We start by identifying the different beam transects present in the GEDI L2B data. We then retrieve the descriptions of each beam transect, indicating whether they are "Full power beams" or "Coverage beams". Next, we filter out only the full power beams to focus our analysis on them. After selecting a full power beam (for demonstration purposes, we choose the first one), we identify all datasets associated with this selected beam. This process ensures that we are working with datasets specifically related to full power beams. Therefore, by integrating these categorical values handling steps into the code, we can effectively manage and analyze the GEDI L2B data based on beam type.

In the figure 27, provided iterates through the attributes of the 'DatasetIdentification' group within the 'METADATA' group of the GEDI L2B HDF5 file. Each attribute represents a specific piece of metadata associated with the dataset. The iteration retrieves both the name of the attribute (attr_name) and its corresponding value (attr_value). The output produced by this code would consist of pairs of attribute names and their respective values. Each pair provides valuable information about the dataset, such as its identification, description, creation date, version, and other relevant details. For example, attributes like 'ShortName',

'LongName', 'VersionID', and 'ProducerGranuleID' typically provide information about the dataset's name, description, version, and unique identifier, respectively. These attributes help users understand the dataset's content, origin, and characteristics. By printing these attributes and their values, users gain insights into the dataset's metadata, facilitating its interpretation and usage in data analysis and research contexts.

Figure 27

Exploring Dataset Identification Attributes in GEDI L2B Metadata

```
In [102]: for attr_name, attr_value in gediL2B['METADATA']['DatasetIdentification'].attrs.items():
              print(attr_name, ":", attr_value)

PGEVersion : 003
VersionID : 01
abstract : The GEDI L2B standard data product contains precise latitude, longitude, elevation, height, cover and vertical pr
ofile metrics for each laser footprint located on the land surface.
characterSet : utf8
creationDate : 2021-02-24T14:05:26.422002Z
credit : The software that generates the L2B product was implemented within the GEDI Science Data Processing System at the N
ASA Goddard Space Flight Center (GSFC) in Greenbelt, Maryland in collaboration with the Department of Geographical Sciences
at the University of Maryland (UMD).
fileName : GEDI02_B_2019259045556_O04306_02_T01808_02_003_01_V002.h5
language : eng
originatorOrganizationName : UMD/GSFC GEDI-SDPS > GEDI Science Data Processing System
purpose : The purpose of the L2B dataset is to extract biophysical metrics from each GEDI waveform. These metrics are based
on the directional gap probability profile derived from the L1B waveform and include canopy cover, Plant Area Index (PAI), P
lant Area Volume Density (PAVD) and Foliage Height Diversity (FHD).
shortName : GEDI_L2B
spatialRepresentationType : along-track
status : onGoing
topicCategory : geoscientificInformation
uuid : b5caa829-f18d-4745-8258-793a1f20f5f1
```

We have retrieved geolocation and quality data for a selected GEDI beam transect from the GEDI L2B HDF5 file and samples every 100th shot. It initializes empty lists to store latitude, longitude, shot number, quality flag, and beam information. Then, it accesses the latitude, longitude, shot number, and quality flag datasets for the selected beam transect from the HDF5 file. For each shot, it checks if the shot number is divisible by 100 without a remainder, indicating every 100th shot. If so, it appends the shot number, latitude, longitude, quality flag, and beam name to their respective lists. Finally, it constructs a data frame from these lists, with columns representing beam, shot number, latitude, longitude, and quality flag, providing a structured representation of the sampled GEDI L2B data as shown in figure 28.

Figure 28

Read SDS Metadata and Subset by Beam for GEDI L2B

The GEDI instrument consists of 3 lasers producing a total of 8 beam ground transects. The eight remaining groups contain data for each of the eight GEDI beam transects.

```
In [11]: beamNames = [g for g in gediL2B.keys() if g.startswith('BEAM')]
         beamNames

Out[11]: ['BEAM0000',
          'BEAM0001',
          'BEAM0010',
          'BEAM0011',
          'BEAM0101',
          'BEAM0110',
          'BEAM1000',
          'BEAM1011']
```

One useful piece of metadata to retrieve from each beam transect is whether it is a full power beam or a coverage beam.

```
In [12]: for g in gediL2B['BEAM0000'].attrs: print(g)

         description
         wp-l2-l2b_githash
         wp-l2-l2b_version

In [13]: for b in beamNames:
             print(f"{b} is a {gediL2B[b].attrs['description']}")

         BEAM0000 is a Coverage beam
         BEAM0001 is a Coverage beam
         BEAM0010 is a Coverage beam
         BEAM0011 is a Coverage beam
         BEAM0101 is a Full power beam
         BEAM0110 is a Full power beam
         BEAM1000 is a Full power beam
         BEAM1011 is a Full power beam
```

GEDI L4B

Our analysis begins with the utilization of the h5py library, which serves as a powerful tool for accessing and manipulating HDF5 files, a format commonly used for storing large and complex datasets. In this particular context, the focus is on GEDI Level 4B HDF5 files, which encompass a plethora of information crucial for understanding Earth's surface and its dynamics. These files are structured to efficiently organize diverse datasets, including ancillary data, multiple BEAM datasets corresponding to different sensor paths, and essential metadata providing valuable insights into the dataset's contents and characteristics. By loading the GEDI Level 4B HDF5 file using h5py, researchers gain access to a wealth of information encapsulated within its hierarchical structure.

Our initial step in the analysis involves obtaining an overview of the file's contents by listing its dataset keys. This process allows researchers to explore the hierarchical organization of the file and gain a better understanding of its structure. Each dataset key represents a distinct component of the file, such as ancillary data, BEAM datasets, or metadata, providing researchers with a comprehensive view of the available data.

Additionally, the analysis delves into the metadata associated with the dataset, extracting key attributes that describe its identification, creation date, and other pertinent details. This metadata serves as a valuable resource for understanding the dataset's provenance and characteristics, facilitating informed data interpretation and analysis.

Moving forward, the analysis shifts its focus to the individual beams present within the GEDI Level 4B HDF5 file. Beams represent different sensor paths employed by the GEDI instrument to capture data from Earth's surface. By iterating through each beam, researchers can gain insights into the specific data collected by each sensor path and its relevance to the analysis. This step involves accessing attributes associated with each beam, such as descriptions that provide information about the type of data it contains and its role within the dataset. Through this process of beam data analysis, researchers can discern the unique characteristics and contributions of each sensor path, enabling more targeted and nuanced analyses of the dataset.

In summary of data preprocessing the analysis of GEDI Level 4B HDF5 files involves a multifaceted approach that encompasses exploring the file's hierarchical structure, extracting metadata attributes, and conducting beam data analysis. By leveraging the capabilities of the h5py library, researchers can navigate complex datasets, gain insights into Earth's surface dynamics, and derive valuable information to further their understanding of key environmental processes. Through comprehensive analyses of GEDI data, researchers contribute to advancements in Earth observation science, ecosystem monitoring, and environmental management, ultimately fostering a deeper understanding of our planet's complex and interconnected systems.

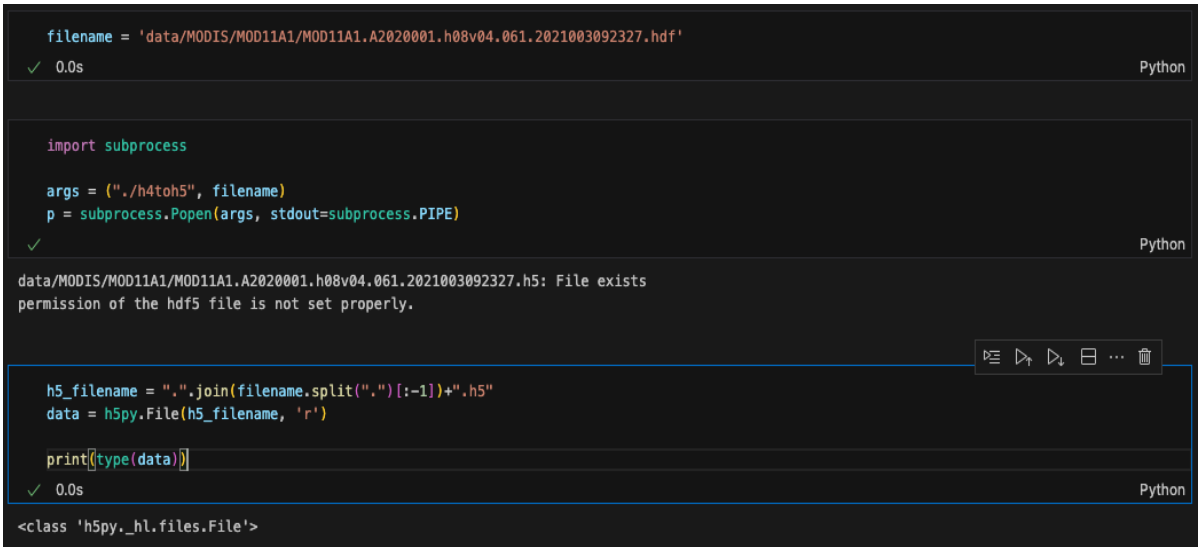
MODIS

A data architecture, file format, and I/O library called the Hierarchical Data Format (HDF) is intended for the preservation, management, and exchange of complicated data, such as data from engineering, science, and remote sensing. MODIS saves the data in a HDF-EOS

file, to keep the consistency of using the same libraries to process both GEDI and MODIS data we first converted the HDF-EOS format file to HDF5 first and later used the the python library called h5py to read the data individually and process it to check for metadata as shown in figure 29.

Figure 29

Code to convert hdf4 to hdf5 and reading using h5py



On displaying the groups inside the file, we see that each MODIS data product is accompanied by XDim and YDim variables whose shape is in file and the starting and ending points are in files metadata. For MODIS data the Sinusoidal Coordinate System was used and to convert it to the Geographic coordinates, Basemap's library pyproj method was used.

On displaying the data fields inside the MOD11A1 data product, we see that there are multiple fields like LST_Day_1km, Day_view_time, Day_view_angl as shown in figure 30, but the integer type used for each of the fields is different. For example, ">" means Big Endian representation is used and "<" means Little Endian representation is used and "u" means unsigned and "i" means signed. While pulling the data to a Pandas DataFrame we will convert all the types to Little Endian specifically "<i2".

Figure 30

Metadata for MOD11A1's Data Fields

```
    for name, value in data['MODIS_Grid_Daily_1km_LST/Data Fields'].items():
        print(name, value)
```

✓ 0.0s

```
LST_Day_1km <HDF5 dataset "LST_Day_1km": shape (1200, 1200), type ">u2">
QC_Day <HDF5 dataset "QC_Day": shape (1200, 1200), type "|u1">
Day_view_time <HDF5 dataset "Day_view_time": shape (1200, 1200), type "|u1">
Day_view_angl <HDF5 dataset "Day_view_angl": shape (1200, 1200), type "|u1">
LST_Night_1km <HDF5 dataset "LST_Night_1km": shape (1200, 1200), type ">u2">
QC_Night <HDF5 dataset "QC_Night": shape (1200, 1200), type "|u1">
Night_view_time <HDF5 dataset "Night_view_time": shape (1200, 1200), type "|u1">
Night_view_angl <HDF5 dataset "Night_view_angl": shape (1200, 1200), type "|u1">
Emis_31 <HDF5 dataset "Emis_31": shape (1200, 1200), type "|u1">
Emis_32 <HDF5 dataset "Emis_32": shape (1200, 1200), type "|u1">
Clear_day_cov <HDF5 dataset "Clear_day_cov": shape (1200, 1200), type ">u2">
Clear_night_cov <HDF5 dataset "Clear_night_cov": shape (1200, 1200), type ">u2">
```

Shown below in figure 31 is the code to convert from Sinusoidal Coordinate System to Geographic Coordinate System and figure 32 shows the x and y coordinates, and latitude and longitude values after converting.

Figure 31

Code to convert the coordinates system

```
20  ny, nx = data['MODIS_Grid_Daily_1km_LST/Data Fields/LST_Day_1km'].shape
21  x = np.linspace(x0, x1, nx)
22  y = np.linspace(y0, y1, ny)
23  xv, yv = np.meshgrid(x, y)
```

✓ 0.0s

```
sinu = pyproj.Proj("+proj=sinu +R=6371007.181 +nadgrids=@null +wktext")
wgs84 = pyproj.Proj("+init=EPSG:4326")
lon, lat= pyproj.transform(sinu, wgs84, xv, yv)
```

Figure 32

Comparison between xv, yv and lat, lon values

```
xv, yv
✓ 0.0s
(array([[-11119505.197665  , -11118577.79939997, -11117650.40113495, ...,
         -10009409.47442905, -10008482.07616403, -10007554.677899  ],
        [-11119505.197665  , -11118577.79939997, -11117650.40113495, ...,
         -10009409.47442905, -10008482.07616403, -10007554.677899  ],
        [-11119505.197665  , -11118577.79939997, -11117650.40113495, ...,
         -10009409.47442905, -10008482.07616403, -10007554.677899  ],
        ...,
        [-11119505.197665  , -11118577.79939997, -11117650.40113495, ...,
         -10009409.47442905, -10008482.07616403, -10007554.677899  ],
        [-11119505.197665  , -11118577.79939997, -11117650.40113495, ...,
         -10009409.47442905, -10008482.07616403, -10007554.677899  ],
        [-11119505.197665  , -11118577.79939997, -11117650.40113495, ...,
         -10009409.47442905, -10008482.07616403, -10007554.677899  ]]),
 array([[5559752.598833  , 5559752.598833  , 5559752.598833  , ...,
         5559752.598833  , 5559752.598833  , 5559752.598833  ],
        [5558825.20056797, 5558825.20056797, 5558825.20056797, ...,
         5558825.20056797, 5558825.20056797, 5558825.20056797],
        [5557897.80230295, 5557897.80230295, 5557897.80230295, ...,
         5557897.80230295, 5557897.80230295, 5557897.80230295],
        ...,
        [4449656.87559605, 4449656.87559605, 4449656.87559605, ...,
         4449656.87559605, 4449656.87559605, 4449656.87559605],
        [4448729.47733103, 4448729.47733103, 4448729.47733103, ...,
         4448729.47733103, 4448729.47733103, 4448729.47733103],
        [4447802.079066  , 4447802.079066  , 4447802.079066  , ...,
         4447802.079066  , 4447802.079066  , 4447802.079066  ]]))
```

```
lat, lon
✓ 0.0s
(array([[50.        , 50.        , 50.        , ..., 50.        ,
         50.        , 50.        ],
        [49.99165972, 49.99165972, 49.99165972, ..., 49.99165972,
         49.99165972, 49.99165972],
        [49.98331943, 49.98331943, 49.98331943, ..., 49.98331943,
         49.98331943, 49.98331943],
        ...,
        [40.01668057, 40.01668057, 40.01668057, ..., 40.01668057,
         40.01668057, 40.01668057],
        [40.00834028, 40.00834028, 40.00834028, ..., 40.00834028,
         40.00834028, 40.00834028],
        [40.        , 40.        , 40.        , ..., 40.        ,
         40.        , 40.        ]]),
 array([[-155.57238269, -155.55940751, -155.54643233, ..., -140.04109477,
         -140.0281196 , -140.01514442],
        [-155.54540061, -155.53242769, -155.51945476, ..., -140.01680641,
         -140.00383348, -139.99086055],
        [-155.51843119, -155.50546052, -155.49248984, ..., -139.99252943,
         -139.97955875, -139.96658807],
        ...,
        [-130.57263176, -130.56174163, -130.55085151, ..., -117.53714884,
         -117.52625871, -117.51536859],
        [-130.55667702, -130.54578822, -130.53489942, ..., -117.52278691,
         -117.51189811, -117.50100931],
        [-130.54072893, -130.52984147, -130.518954  , ..., -117.50843097,
         -117.49754351, -117.48665604]]))
```

All the MODIS data products are in a 2-dimensional format, so we do a simple flatten to get the values for all the latitude and longitude. Sample data is shown in figure 33 for the

MOD11A1 and MOD13A2. Sample data after data preprocessing is shown in figure 34 for MOD15A2H and MOD17A2H.

Figure 33

Sample data for 2021-01-03 for MOD11A1 (left) and MOD13A2 (right).

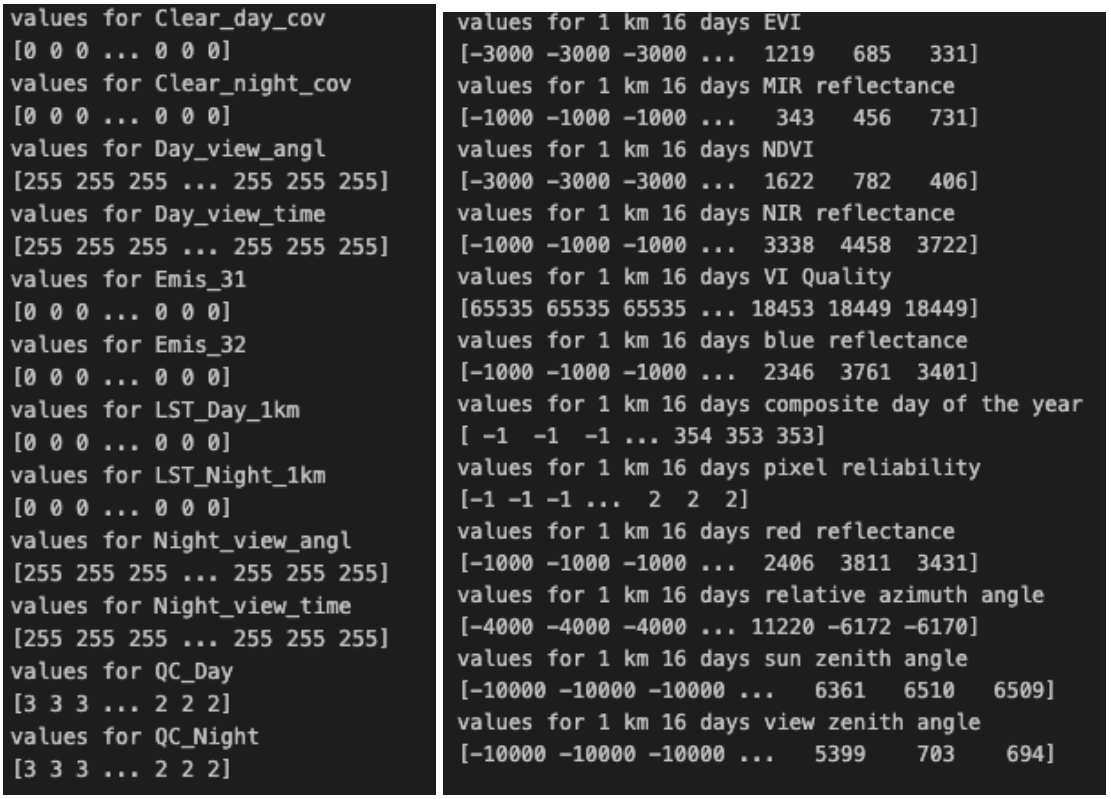


Figure 34

Sample data for 2021-01-12 for MOD15A2H (left) and MOD17A2H (right).

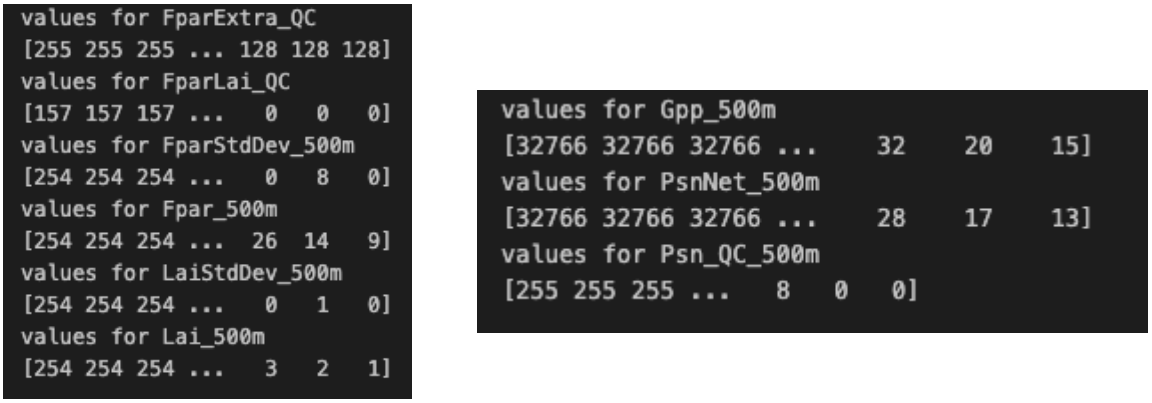


Figure 35

Sample data for 2021-01-12 for Gedi L2A.

```
# Convert to a Geodataframe
latslons = gp.GeoDataFrame(latslons)
latslons = latslons.drop(columns=['Latitude','Longitude'])
latslons['geometry']
```

```
0        POINT (111.99630 -51.80387)
1        POINT (112.03913 -51.80391)
2        POINT (112.08027 -51.80384)
3        POINT (112.12145 -51.80374)
4        POINT (112.16262 -51.80362)
                    ...
9792      POINT (88.20845 -51.80358)
9793      POINT (88.24961 -51.80361)
9794      POINT (88.29075 -51.80358)
9795      POINT (88.33191 -51.80355)
9796      POINT (88.37309 -51.80351)
Name: geometry, Length: 9797, dtype: geometry
```

Figure 36

Location Data to merge with the Gedi l2a data

```
wvDF = pd.read_csv('waveform.csv')
```

```
wvDF
```

	Amplitude (DN)	Elevation (m)
0	227.00992	111.252897
1	226.57410	111.103165
2	226.66390	110.953434
3	227.26736	110.803702
4	228.11644	110.653971
...	...	...
1241	230.74078	-74.564040
1242	230.04378	-74.713771
1243	229.24507	-74.863503
1244	228.71117	-75.013235
1245	228.53528	-75.162966

1246 rows × 2 columns

3.4 Data Transformation

GEDIL2A

In data transformation, we transition from raw GEDI Level 2A Elevation and Height Metrics Data to a more structured format suitable for analysis and visualization. Initially, we extract relevant spatial information from the dataset by creating a GeoDataFrame using GeoPandas. By leveraging latitude and longitude coordinates provided in the dataset, we generate Shapely points representing the location of each GEDI shot. We subset the data by selecting a representative sample of shots, ensuring manageability while preserving the dataset's integrity. We take every 100th shot and compile pertinent attributes such as shot number, latitude, longitude, and quality flag into lists. These lists are then used to construct a Pandas DataFrame, facilitating easier manipulation and analysis as shown in figure 37.

Figure 37

Sample Data of Geodataframe for GEDI 2LA data

	Shot Index	Shot Number	Latitude	Longitude	Tandem-X DEM	Elevation (m)	Canopy Elevation (m)	Canopy Height (rh100)	Quality Flag	Degrade Flag	Sensitivity
0	0	29320618800000001	-51.803868	111.996300	-999999.0	21242.515625	21242.515625	0.0	0	0	-3.436965
1	1	29320618900000002	-51.803867	111.996712	-999999.0	21242.505859	21242.505859	0.0	0	0	30.496670
2	2	29320619000000003	-51.803867	111.997123	-999999.0	21242.496094	21242.496094	0.0	0	0	8.071431
3	3	29320619100000004	-51.803867	111.997535	-999999.0	21242.484375	21242.484375	0.0	0	0	-212.896439
4	4	29320619200000005	-51.803866	111.997946	-999999.0	21242.474609	21242.474609	0.0	0	0	-6.853874
...	...	...	...	...	...	...	...	...	...	...	...
979694	979694	29320618400979695	-51.803445	88.411747	-999999.0	18017.906250	18017.906250	0.0	0	0	22.138037
979695	979695	29320618500979696	-51.803445	88.412159	-999999.0	18017.296875	18017.296875	0.0	0	0	4.475757
979696	979696	29320618600979697	-51.803444	88.412570	-999999.0	18017.884766	18017.884766	0.0	0	0	10.112548
979697	979697	29320618700979698	-51.803444	88.412981	-999999.0	18017.275391	18017.275391	0.0	0	0	424.691803
979698	979698	29320618800979699	-51.803443	88.413393	-999999.0	18017.263672	18017.263672	0.0	0	0	15.887813

979699 rows × 11 columns

We extract a subset of sample data from a GEDI L2B dataset for visualization and analysis. It initializes empty lists to store sample data including latitude, longitude, shot number, quality flag, and beam name. Then, it accesses specific SDS (Scientific Data Sets) related to geolocation and quality flags for the first beam (beamNames[0]). For visualization purposes, it selects every 100th shot from the dataset and appends the corresponding latitude, longitude, shot number, quality flag, and beam name to the respective lists. Finally, it

constructs a panda DataFrame containing the sample data, which includes information about the beam, shot number, longitude, latitude, and quality flag. This subset of data is useful for preliminary exploration, quality assessment, and visualization tasks in figure 38

Figure 38

Subset by Layer and Create a Geodataframe

Read in the SDS and take a representative sample (every 100th shot) and append to lists, then use the lists to generate a `pandas` dataframe.

In [15]:
```
lonSample, latSample, shotSample, qualitySample, beamSample = [], [], [], [], []  # Set up lists to store data

# Open the SDS
lats = gediL2A[f'{beamNames[0]}/lat_lowestmode'][()]
lons = gediL2A[f'{beamNames[0]}/lon_lowestmode'][()]
shots = gediL2A[f'{beamNames[0]}/shot_number'][()]
quality = gediL2A[f'{beamNames[0]}/quality_flag'][()]

# Take every 100th shot and append to list
for i in range(len(shots)):
    if i % 100 == 0:
        shotSample.append(str(shots[i]))
        lonSample.append(lons[i])
        latSample.append(lats[i])
        qualitySample.append(quality[i])
        beamSample.append(beamNames[0])

# Write all of the sample shots to a dataframe
latslons = pd.DataFrame({'Beam': beamSample, 'Shot Number': shotSample, 'Longitude': lonSample, 'Latitude': latSample,
                         'Quality Flag': qualitySample})
latslons
```

Out[15]:

	Beam	Shot Number	Longitude	Latitude	Quality Flag
0	BEAM0110	29320618800000001	111.996300	-51.803868	0
1	BEAM0110	29320604600000101	112.039132	-51.803905	0
2	BEAM0110	29320614600000201	112.080271	-51.803836	0
3	BEAM0110	29320600400000301	112.121445	-51.803737	0
4	BEAM0110	29320610400000401	112.162622	-51.803621	0
...	...	...	...	...	...
9792	BEAM0110	29320617400979201	88.208452	-51.803578	0
9793	BEAM0110	29320603200979301	88.249610	-51.803614	0
9794	BEAM0110	29320613200979401	88.290753	-51.803581	0
9795	BEAM0110	29320623200979501	88.331913	-51.803548	0
9796	BEAM0110	29320609000979601	88.373089	-51.803506	0

We enhance the dataset's spatial representation by adding a 'geometry' column containing Shapely points generated from the extracted latitude and longitude coordinates. Converting the DataFrame into a GeoDataFrame using GeoPandas enables us to leverage spatial functionalities and integrate seamlessly with other geospatial data. With the transformed dataset, we proceed to visualize the GEDI shots spatially on a basemap using the GeoViews Python package. A GeoJSON file outlining the spatial extent of Redwood National Park, providing contextual reference for the GEDI shot locations as shown in figure 39.

Figure 39

Convert to a Geopandas GeoDataFrame.

Below, create an additional column called 'geometry' that contains a `shapely` point generated from each lat/lon location from the shot.

In [17]:
```
# Take the lat/lon dataframe and convert each lat/lon to a shapely point
latslons['geometry'] = latslons.apply(lambda row: Point(row.Longitude, row.Latitude), axis=1)
```

Next, convert to a `Geopandas` GeoDataFrame.

In [18]:
```
# Convert to a Geodataframe
latslons = gp.GeoDataFrame(latslons)
latslons = latslons.drop(columns=['Latitude','Longitude'])
latslons['geometry']
```

Out[18]:
```
0       POINT (111.99630 -51.80387)
1       POINT (112.03913 -51.80391)
2       POINT (112.08027 -51.80384)
3       POINT (112.12145 -51.80374)
4       POINT (112.16262 -51.80362)
                   ...
9792     POINT (88.20845 -51.80358)
9793     POINT (88.24961 -51.80361)
9794     POINT (88.29075 -51.80358)
9795     POINT (88.33191 -51.80355)
9796     POINT (88.37309 -51.80351)
Name: geometry, Length: 9797, dtype: geometry
```

Pull out and plot an example `shapely` point below.

In [19]:
```
latslons['geometry'][0]
```

Out[19]:

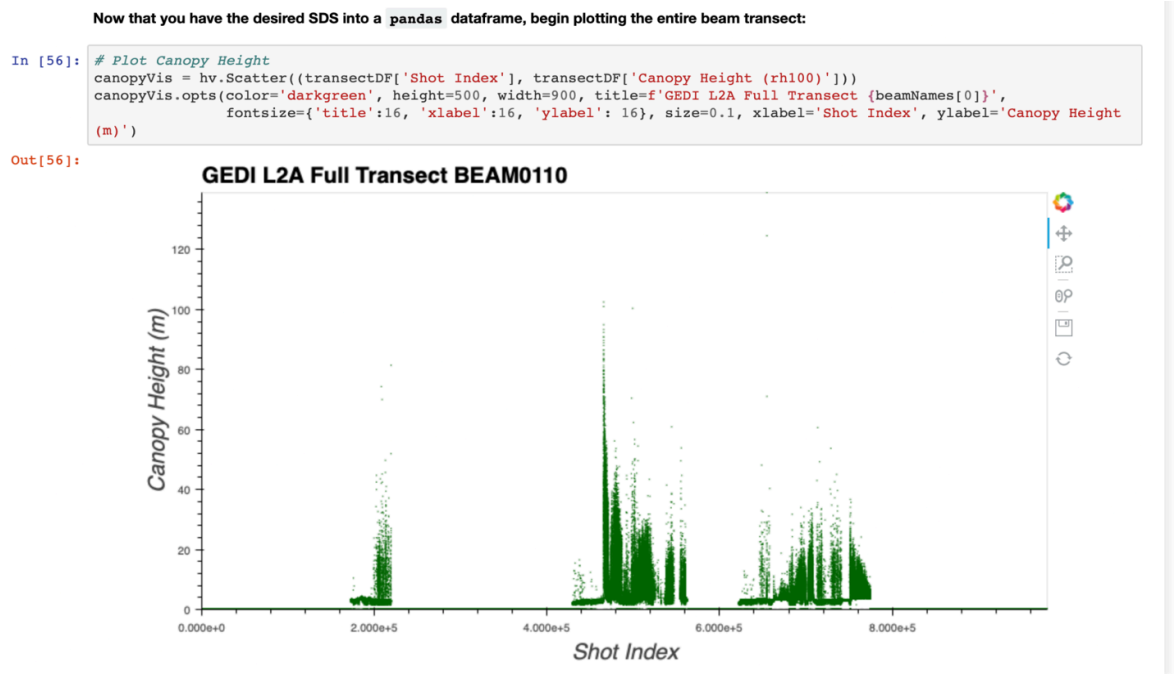
Combining the visualizations of the GEDI shots and the Redwood National Park boundary, we gain insights into the distribution of shots within the park's vicinity. The interactive nature of the plots facilitates exploration, allowing users to zoom in and inspect individual shots for further analysis. Overall, this data transformation process enhances the dataset's usability and facilitates subsequent analysis and interpretation, laying the groundwork for further insights into GEDI Level 2A data.

We then shift our primary focus in understanding and interpreting relative height (RH) metrics. To initiate this process, we embark on importing and extracting a specific GEDI L2A shot, a task that entails narrowing down our datasets to a chosen high-power beam. Additionally, we set an example shot index to facilitate further analysis and manipulation of the data.Upon laying this foundational groundwork, our journey proceeds to the visualization realm as we endeavor to plot the RH metrics. This endeavor commences with the importation of RH metrics, accompanied by an insightful description of their inherent characteristics. To enrich our analysis, we incorporate supplementary datasets such as latitude and longitude, which serve to provide essential contextual information. Leveraging the powerful

visualization capabilities of Holoviews, we craft a visually engaging representation of the RH metrics, offering a nuanced perspective by showcasing elevation at each percentile for the shot as shown in figure 40.

Figure 40

GEDI L2A Full Transect BEAM

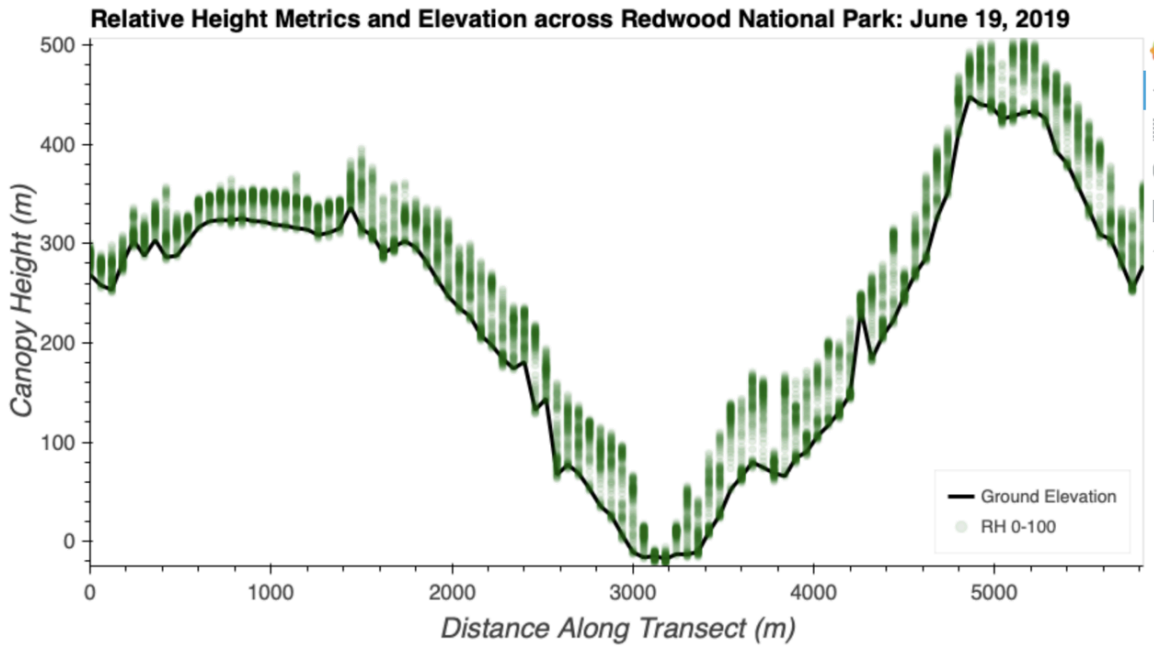


With our RH metrics visually articulated, our attention shifts towards the fusion of RH metrics with waveform data. This integration begins with the acquisition of waveform data corresponding to the selected shot, a crucial step in providing a comprehensive understanding of the data landscape. Leveraging the visualization capabilities of Holoviews once again, we present the waveform data, enabling the identification of key features such as ground return and canopy heights. By merging the L2A RH metrics with the L1B waveform data, we create an integrated view that offers deeper insights into the structural characteristics of the terrain. As we near the conclusion of our data preparation journey, we delve into the exploration of data from a non-default algorithm setting (Algorithm Setting Group 2). This exploration enables us to compare RH metrics from Algorithm Setting Group 1, shedding light on potential variations in data interpretation stemming from different algorithmic

approaches. This comparative analysis enriches our understanding of the data landscape, paving the way for informed decision-making in subsequent analysis and modeling endeavors as shown in figure 41.

Figure 41

Height metrics and Elevation across Redwood Park



GEDI L2B

We demonstrate a comprehensive data transformation process focused on shot number, canopy height, and Plant Area Index (PAI). Initially, the GEDI L2B data is imported into a panda DataFrame, including various attributes such as shot number, latitude, longitude, canopy height, and PAI.

Firstly, regarding shot number, this parameter likely represents the number of laser pulses emitted by Light Detection and Ranging (LiDAR) technology. LiDAR is commonly used to measure canopy height and other vegetation parameters by sending laser pulses and analyzing their reflection. Shot number data might initially be in a raw format, consisting of a list of numerical values representing the count of laser pulses recorded at different locations within the study area.To transform shot number data, we performed data cleaning to remove

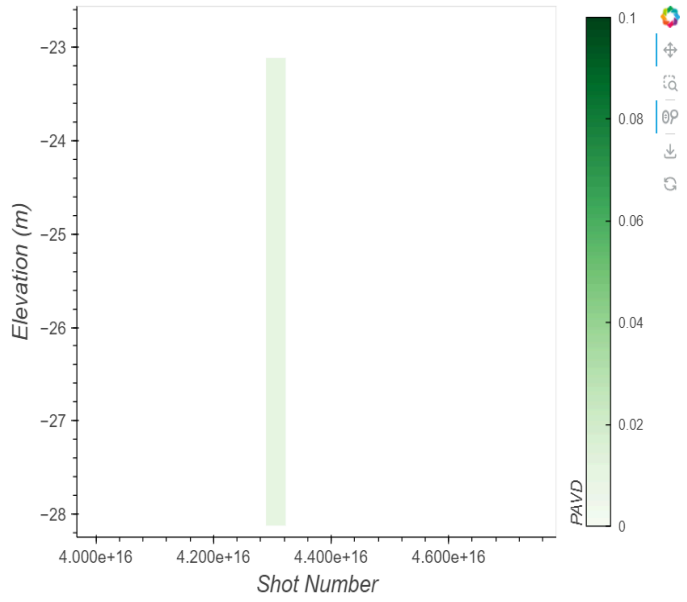
any outliers or erroneous values. This could involve identifying and filtering out shots that did not produce reliable measurements due to factors like sensor malfunction or environmental interference. Next, the shot number data might be aggregated or summarized to calculate statistics such as the total number of shots per plot or the average shot density across the study area. This aggregation could help in understanding the overall intensity of LiDAR data acquisition and its spatial distribution as shown in figure 42.

Figure 42

Visualize Plant Area Volume Density (PAVD)

```
path1 = hv.Path(pavdAll, vdims='PAVD').options(color='PAVD', clim=(0,0.1), cmap='Greens', line_width=20, colorbar=True,
                                               width=700, height=550, clabel='PAVD', xlabel='Shot Number',
                                               ylabel='Elevation (m)', fontsize={'title':16, 'xlabel':16, 'ylabel': 16,
                                                                                 'xticks':12, 'yticks':12,
                                                                                 'clabel':12, 'cticks':10})
path1
```

]:



Next, we transform canopy height data, preprocessing steps such as filtering out ground returns and classifying vegetation points may be necessary. This involves distinguishing between echoes reflected from the ground surface and those from vegetation, enabling the extraction of canopy height information. Subsequently, statistical analysis or interpolation techniques may be applied to derive metrics such as mean canopy height, maximum height, or height distribution patterns across the study area. These metrics can offer valuable insights into the vertical structure and variability of the forest canopy. Quality

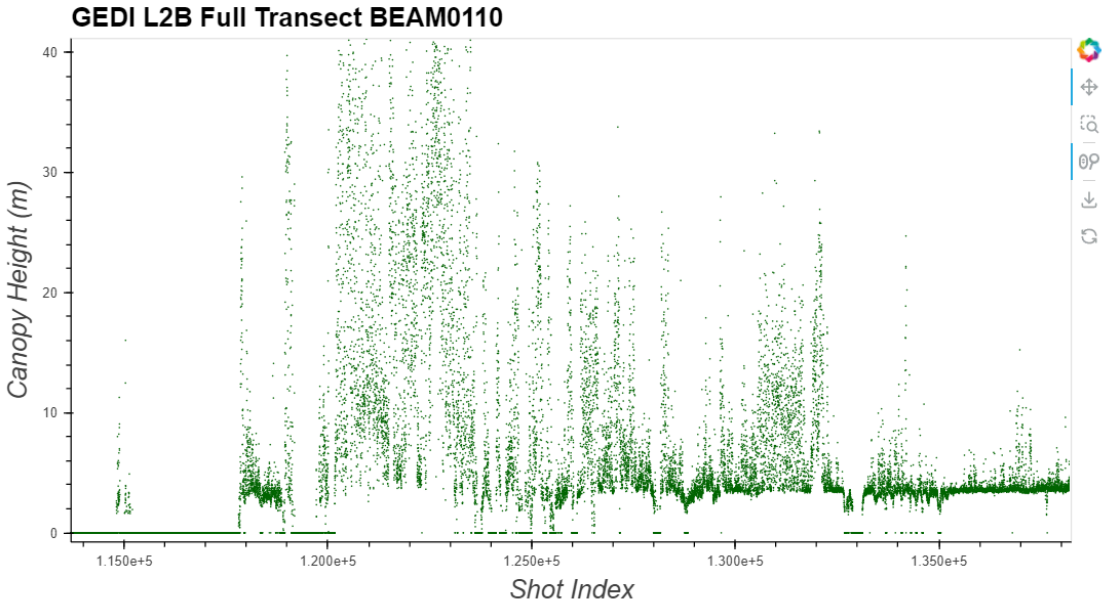
filtering is then applied to remove poor-quality shots, ensuring that only high-quality data points are retained for further analysis. This filtering process involves examining attributes such as the l2b_quality_flag and the Sensitivity layer, allowing for the identification and removal of shots that do not meet predefined quality standards. After quality filtering, the DataFrame is subsetted to include only high-quality shots, thereby refining the dataset to focus on reliable and accurate data points as shown in figure 43.

Figure 43

Work with GEDI L2B Beam Transects

```
# Plot Canopy Height
canopyVis = hv.Scatter((transectDF['Shot Index'], transectDF['Canopy Height (rh100)']))
canopyVis.opts(color='darkgreen', height=500, width=900, title=f'GEDI L2B Full Transect {beamNames[0]}',
               fontsize={'title':16, 'xlabel':16, 'ylabel': 16}, size=0.1, xlabel='Shot Index', ylabel='Canopy Height (m)')
```

7]:



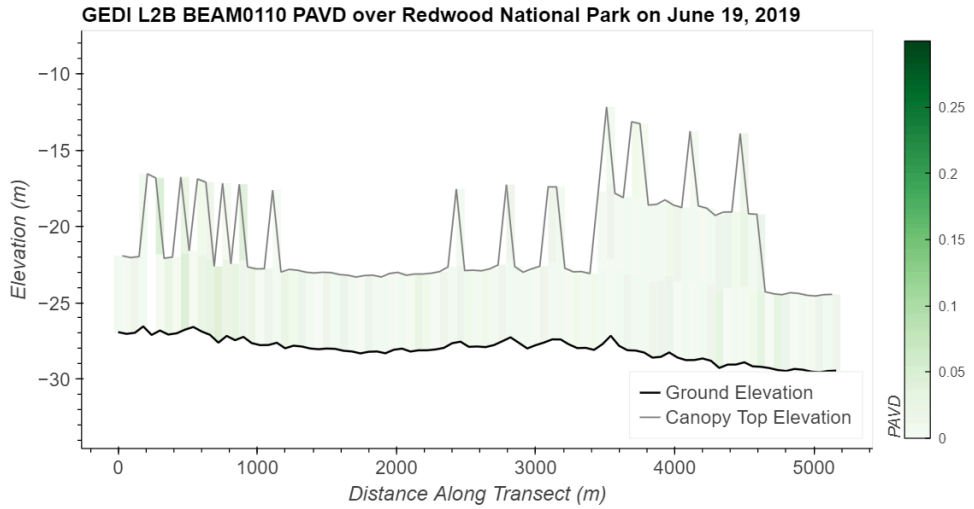
Lastly, we transform PAI data, preprocessing steps such as normalization or calibration may be required to account for variations in LiDAR sensor characteristics or data acquisition conditions. Additionally, spatial interpolation or extrapolation techniques might be employed to estimate PAI values at unsampled locations based on neighboring observations. Finally, statistical analysis or spatial modeling approaches could be used to

derive insights from PAI data, such as identifying areas of high vegetation density or assessing spatial patterns of leaf area distribution within the study area as shown in figure 45.

At this point we have visualized the elevation, canopy, and vertical structure of specific footprints over Redwood national park, and for a transect cutting through the national park. We have mapped all of the high-quality shots from all eight GEDI beams for a given region of interest in order to gain knowledge on the spatial distribution of and characteristics of the canopy over Redwood National Park. In summary, data transformation plays a crucial role in extracting meaningful information from raw LiDAR data, particularly concerning parameters such as shot number, canopy height, and Plant Area Index. By performing data cleaning, aggregation, preprocessing, and analysis, researchers can unlock valuable insights into forest structure, vegetation density, and ecosystem dynamics.

Figure 44

*GEDI L2B BEAM PAVD over Redwood*s



GEDI L4B

The provided code snippet exemplifies a crucial aspect of data transformation within the context of extracting Above Ground Biomass Density (AGBD) data from a GEDI Level 4B HDF5 file. This process is fundamental for converting raw, unstructured data into a structured format amenable to analysis and interpretation. Initially, the code utilizes the h5py library to open the GEDI Level 4B HDF5 file, allowing access to its contents. By listing the

keys within the file, the structure of the dataset is revealed, including metadata and various BEAM datasets containing pertinent information about Earth's surface features. This initial step is pivotal for gaining insight into the organization of the data and understanding the metadata attributes that describe each dataset.

Subsequently, the code focuses on extracting AGBD data specifically from the 'BEAM0000' dataset, which represents one of the sensor paths. Through h5py's functionality, the AGBD values and corresponding shot numbers are accessed from the specified BEAM dataset. These values are then formatted into a panda DataFrame, a powerful data structure that facilitates further manipulation and analysis. The DataFrame construction involves assigning the shot numbers to the 'Shot Number' column and the corresponding AGBD values to the 'AGBD' column, effectively organizing the data in a tabular format.

Following the data transformation stage, the code proceeds to perform data filtering to remove invalid entries, such as AGBD values of -9999, which typically indicate missing or erroneous data. This filtering step is crucial for ensuring the integrity and accuracy of the dataset, as it eliminates noise and inconsistencies that could skew subsequent analyses. By retaining only valid AGBD readings, the filtered DataFrame becomes a reliable source of information for conducting meaningful analyses.

Finally, the DataFrame serves as a foundational tool for further analysis, offering researchers the flexibility to explore and derive insights from the AGBD data. Common analyses, such as calculating descriptive statistics like the mean or visualizing the distribution of AGBD values, can be easily performed using pandas' built-in functions and visualization libraries. Ultimately, the data transformation process exemplified in the code snippet enables researchers to unlock valuable insights into biomass distribution and ecosystem dynamics, contributing to a deeper understanding of Earth's environment and informing conservation efforts and land management strategies.

Table 12

GEDI L4B Dataframe reading AGBD.

agbd_a1	agbd_a10	agbd_a2	agbd_a3	agbd_a4	agbd_a5	agbd_a6	Elevation (m)	Sensitivity
19.31536	19.932028	19.93202	20.10183	1.271942	19.93202	19.93202	2.4683213	0.81598234
2.221454	2.1639936	2.221454	2.221454	1.331116	2.163993	2.221454	2.4688036	0.8786501
1.271942	0.8707129	0.870712	0.870712	0.870712	0.870712	0.870712	2.4692874	0.86727774
0.787927	0.4797173	0.479717	0.516440	0.181137	0.479717	0.479717	2.4697704	0.4319287
1.697226	1.4534664	1.453466	1.500721	0.919861	1.453466	1.453466	2.4702542	0.8004754
72.21525	75.060745	75.06074	75.3801	7.398452	75.06074	75.3801	2.4707422	0.8177055
57.7291	57.07305	57.07305	57.07305	48.95532	57.07305	57.07305	2.4712303	0.8834758
60.96597	62.811077	62.81107	63.20282	2.378323	62.81107	62.81107	2.4717107	0.9276653
1.391625	0.8347305	0.834730	0.870712	1.061925	0.834730	0.834730	2.472194	0.912249
22.55216	21.838184	21.83818	22.07494	0.544866	21.83818	21.83818	2.4726796	0.8443932

MODIS

The data fields from the previous chapter cannot be used as it is for the modeling, so we decided to write the data into csv first for easy access and better understanding, converting it to csv will also provide us with tools which will help us to do complex calculations using inbuilt functions. As shown in figure 45 we iterate through every data field and write the data values from the h5 file to a pandas dataframe which already has the coordinate value loaded, once the data is loaded, we look at the data in figure 46.

Figure 45

Iterating through all the datasets in the group and writing them into a dataframe.

```
    for i in modisSDS:
        if i.startswith("MODIS_"):
            var_name = str(i.split("/")[2])
            print(f"writing {var_name} to the dataframe")
            var_values = data[[i][0]] [()].astype('<i2')
            df[var_name] = var_values.flatten()

✓ 2.3s

writing 250m 16 days EVI to the dataframe
writing 250m 16 days MIR reflectance to the dataframe
writing 250m 16 days NDVI to the dataframe
writing 250m 16 days NIR reflectance to the dataframe
writing 250m 16 days VI Quality to the dataframe
writing 250m 16 days blue reflectance to the dataframe
writing 250m 16 days composite day of the year to the dataframe
writing 250m 16 days pixel reliability to the dataframe
writing 250m 16 days red reflectance to the dataframe
writing 250m 16 days relative azimuth angle to the dataframe
writing 250m 16 days sun zenith angle to the dataframe
writing 250m 16 days view zenith angle to the dataframe
```

Figure 46

Displaying first 5 rows for the newly loaded dataframe.

```
df.head()
✓ 0.0s
```

	Latitude	Longitude	Clear_day_cov	Clear_night_cov	Day_view_angl	Day_view_time	Emis_31	Emis_32	LST_Day_1km
0	50.0	-155.572383	0	0	255	255	0	0	0
1	50.0	-155.559408	0	0	255	255	0	0	0
2	50.0	-155.546432	0	0	255	255	0	0	0
3	50.0	-155.533457	0	0	255	255	0	0	0
4	50.0	-155.520482	0	0	255	255	0	0	0

Sample Data

MODIS

Sample data after data transformation for MODIS is given below, Figure 47 has the sample data for MOD11A1 and MOD13A2. Figure 48 shows the data after transformation in csv format for MOD15A2H and MOD17A2H.

Figure 47

Sample Data for MOD11A1 (above) and MOD13A2 (below) for date 2021-01-03

	Latitude	Longitude	Clear_day_cov	Clear_night_cov	Day_view_angl	Day_view_time	Emis_31	Emis_32	LST_Day_1km	LST_Night_1km
0	50.0	-155.572383	0	0	255	255	0	0	0	0
1	50.0	-155.559408	0	0	255	255	0	0	0	0
2	50.0	-155.546432	0	0	255	255	0	0	0	0
3	50.0	-155.533457	0	0	255	255	0	0	0	0
4	50.0	-155.520482	0	0	255	255	0	0	0	0

	Latitude	Longitude	1 km 16 days EVI	1 km 16 days MIR reflectance	1 km 16 days NDVI	1 km 16 days NIR reflectance	1 km 16 days VI Quality	1 km 16 days blue reflectance	1 km 16 days composite day of the year	1 km 16 days pixel reliability	1 km 16 days red reflectance	1 km 16 days relative azimuth angle	1 km 16 days sun zenith angle	1 km 16 days view zenith angle
0	50.0	-155.572383	-3000	-1000	-3000	-1000	-1	-1000	-1	-1	-1000	-4000	-10000	-10000
1	50.0	-155.559407	-3000	-1000	-3000	-1000	-1	-1000	-1	-1	-1000	-4000	-10000	-10000
2	50.0	-155.546432	-3000	-1000	-3000	-1000	-1	-1000	-1	-1	-1000	-4000	-10000	-10000
3	50.0	-155.533457	-3000	-1000	-3000	-1000	-1	-1000	-1	-1	-1000	-4000	-10000	-10000
4	50.0	-155.520482	-3000	-1000	-3000	-1000	-1	-1000	-1	-1	-1000	-4000	-10000	-10000

Figure 48

Sample Data for MOD15A2H (above) and MOD17A2H (below) for data 2021-01-12

	Latitude	Longitude	FparExtra_QC	FparLai_QC	FparStdDev_500m	Fpar_500m	LaiStdDev_500m	Lai_500m
0	50.0	-155.572383	255	157	254	254	254	254
1	50.0	-155.565898	255	157	254	254	254	254
2	50.0	-155.559413	255	157	254	254	254	254
3	50.0	-155.552928	255	157	254	254	254	254
4	50.0	-155.546443	255	157	254	254	254	254

	Latitude	Longitude	Gpp_500m	PsnNet_500m	Psn_QC_500m
0	50.0	-155.572383	32766	32766	255
1	50.0	-155.565898	32766	32766	255
2	50.0	-155.559413	32766	32766	255
3	50.0	-155.552928	32766	32766	255
4	50.0	-155.546443	32766	32766	255

3.5 Data Preparation

Data preparation for GEDI and MODIS data is a critical step in ensuring the quality, consistency, and usability of the datasets for subsequent analysis and modeling tasks. This process encompasses various steps aimed at cleaning, preprocessing, and organizing the data to make it suitable for analysis by machine learning algorithms and other analytical techniques.

Initially, data cleaning techniques are employed to address any issues such as missing values, outliers, or inconsistencies that may be present in the datasets. For GEDI data, which

measures forest structure and biomass, missing values in parameters such as canopy height or shot number may arise due to sensor malfunctions or data processing errors. Similarly, MODIS data, which provides information on land cover, vegetation dynamics, and surface temperature, may contain missing values or outliers resulting from cloud cover, sensor errors, or atmospheric interference. Techniques such as imputation, outlier detection, and removal are applied to handle these issues and ensure the integrity of the datasets.

Following data cleaning, preprocessing techniques are applied to standardize, normalize, and transform the datasets to ensure uniformity and comparability across different variables and sensors. For example, GEDI data may be preprocessed to compute derived variables such as Plant Area Index (PAI) or Leaf Area Index (LAI) from raw measurements of canopy height and structure. Similarly, MODIS data may undergo preprocessing steps such as atmospheric correction, radiometric calibration, and geometric correction to improve accuracy and remove systematic errors. Additionally, feature extraction techniques such as vegetation indices (e.g., NDVI) or texture measures may be computed from MODIS imagery to capture relevant information about land cover and vegetation health.

Once all the data for GEDI and MODIS have been gathered, labeled, and organized, the final datasets will undergo division into training, validation, and test datasets. The models will then be trained using the training set, leveraging its ample examples to learn and adjust parameters effectively. Model selection will be based on the validation set, allowing for the assessment of prediction error and the fine-tuning of hyperparameters. Subsequently, the test set will serve to evaluate any generalization flaws in the ultimately selected model.

The training set, representing the largest proportion of the dataset (typically around 70-80%), serves as the foundation for model training, providing ample examples for the algorithms to learn from and adjust their parameters accordingly. Key terms such as AGBD (Above Ground Biomass Density) and PAI (Plant Area Index) are incorporated into this process to underscore the specificity of the data. The validation set, comprising

approximately 10-15% of the data, is crucial for fine-tuning the model's hyperparameters and evaluating its performance on unseen data during the training process. This step is particularly relevant for GEDI data, where variables like shot number, canopy height, and elevation play pivotal roles in environmental assessments. Meanwhile, for MODIS data, parameters such as land surface temperature and vegetation indices are central to the analysis. The validation step is essential for optimizing model performance and preventing overfitting, ensuring that the trained model generalizes well to new, unseen data. Finally, the test set, also comprising 10-15% of the data, serves as an independent evaluation dataset used to assess the model's performance on completely new observations. By maintaining the integrity of spatial and temporal relationships within each subset, the data preparation process aims to ensure that the partitioned datasets accurately reflect the underlying distribution and characteristics of the original dataset, thereby facilitating the development of robust and generalizable machine learning models for environmental science and remote sensing applications. To facilitate this process, the package split-folders will be employed for partitioning the data and preparing it for deep learning. This package offers functionality to divide folders containing files into folders for the train, validation, and test datasets, while also rearranging the input files to ensure genuine unpredictability among the three sets. By utilizing the package module and defining the desired ratio, the dataset will be randomly partitioned into an output folder, aligning with the objective of achieving a 60%-20%-20% split for training, validation, and testing, respectively.

3.6 Data Statistics

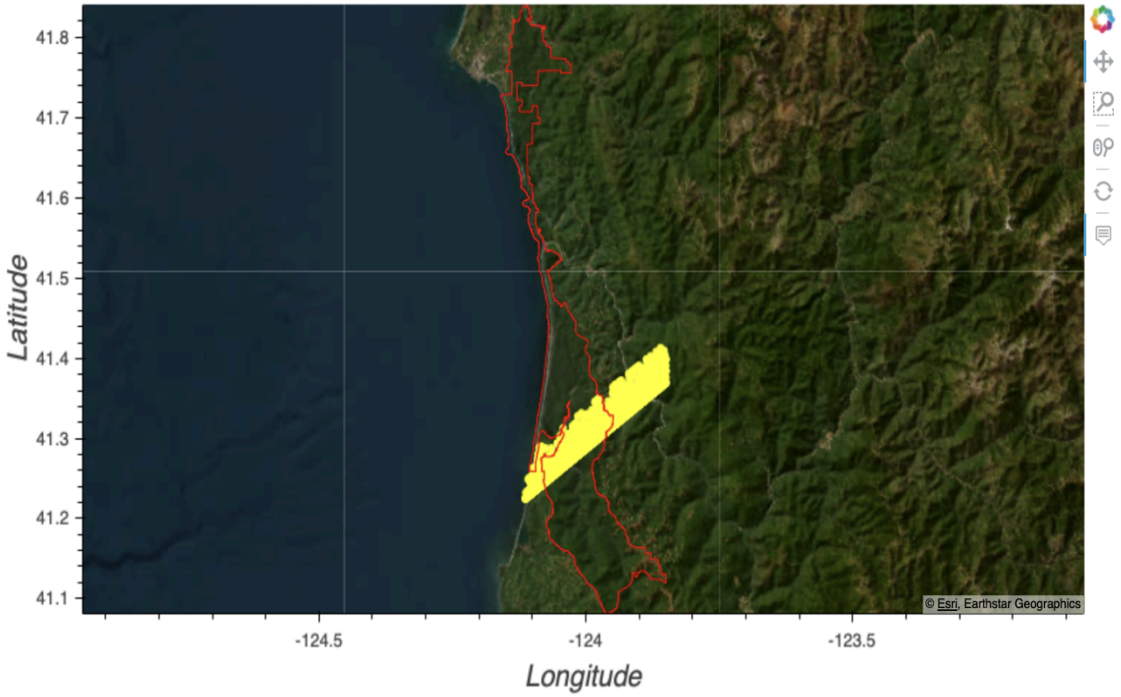
GEDI L2A

The methodology detailed herein constitutes a rigorous and systematic approach to the preprocessing and analysis of GEDI (Global Ecosystem Dynamics Investigation) L2A data, specifically tailored to the unique ecological landscape of Redwood National Park. Beginning with the importation of a substantial dataset comprising over 3.5 million shots

captured by all eight GEDI beams, the process involved meticulous data preparation steps to ensure the quality and relevance of the information under examination. Spatial sub setting techniques were employed to focus the analysis on the geographical confines of Redwood National Park, thereby delineating a region of interest that encapsulates the park's distinctive ecosystem dynamics as shown in figure 49.

Figure 49

Plot the geopandas GeoDataFrame using geoviews.

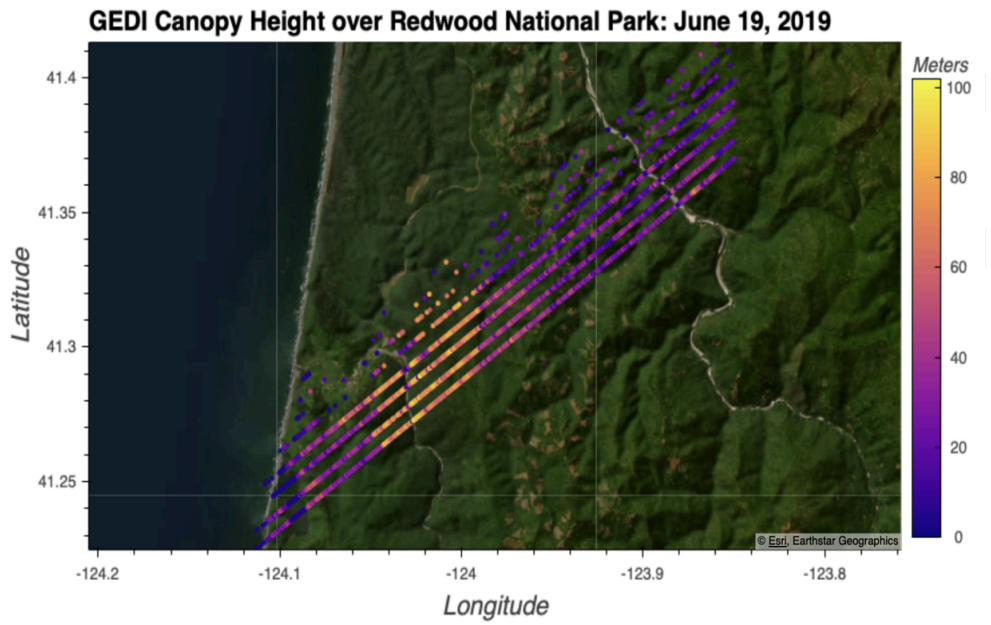


To ensure data integrity and analytical robustness, stringent quality filtering mechanisms were implemented to cull out shots deemed to be of poor quality, shots acquired during degraded periods, and those with sensitivity values falling below a predetermined threshold of 0.95. This meticulous filtering process culminated in the refinement of the dataset to approximately 2081 high-quality shots, thus establishing a reliable foundation for subsequent analysis. The resulting dataset comprises a suite of variables crucial for understanding the canopy structure and terrain characteristics within Redwood National Park. These variables include Shot Number, Beam identification, Latitude, Longitude, Tandem-X

DEM (Digital Elevation Model), Elevation, Canopy Elevation, Canopy Height (rh100), RH 98, RH 25, Quality Flag, Degrade Flag, and Sensitivity. Each variable encapsulates valuable information pertaining to the topographical and vegetative attributes of the park, facilitating a comprehensive assessment of its ecological dynamics as shown in figure 50.

Figure 50

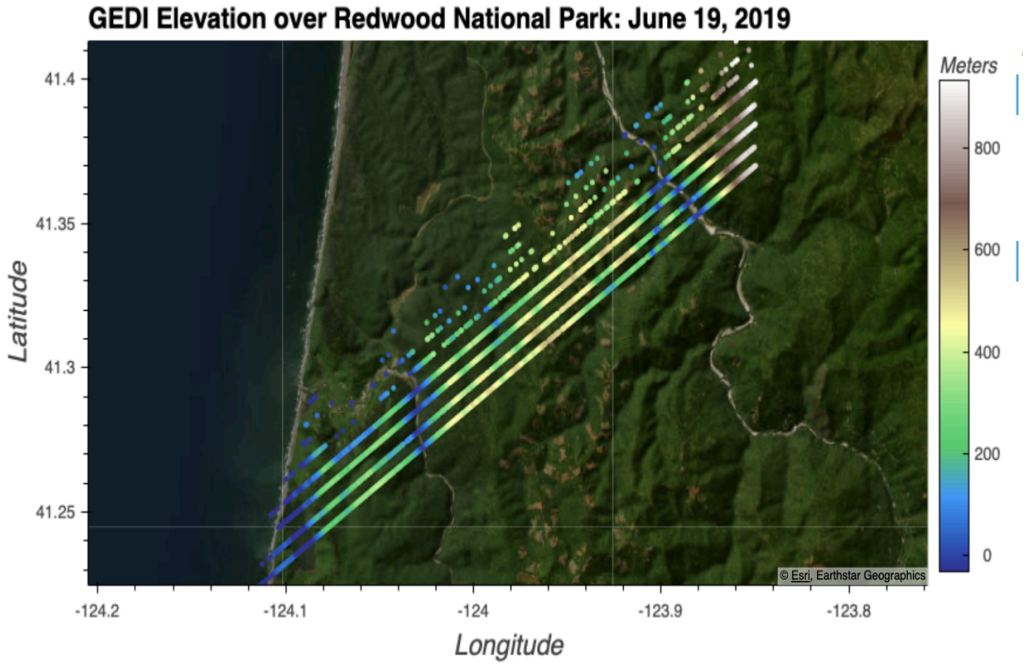
GEDI Canopy Height over Redwood National Par



Visual representations of canopy height and elevation profiles were instrumental in elucidating spatial patterns and trends across the park's landscape. Leveraging colormaps such as Plasma for canopy height and Terrain for elevation, these visualizations provided insights into the vertical structure and terrain variations within Redwood National Park. Furthermore, statistical analyses, including histograms, box plots, and scatter plots, were employed to discern distributional patterns and explore relationships among the variables of interest. In addition to the analytical procedures outlined above, robust data integrity checks were conducted at each stage of the analysis to ensure the consistency, accuracy, and reliability of the processed data. These checks served to validate the integrity of the dataset and mitigate the risk of spurious findings or erroneous conclusions as shown in figure 51.

Figure 51

GEDI Elevation over Redwood National Park



GEDI L2B

We initiate by importing GEDI datasets for all eight beams and extracting essential parameters, including shot number, canopy height, elevation, and Plant Area Index (PAI). Quality filtering is then applied to eliminate poor quality shots, ensuring the reliability and accuracy of the dataset. Following this, the dataset undergoes spatial sub setting, focusing exclusively on the Redwood National Park ROI. Shots outside the park's bounding box are filtered out, enabling a concentrated analysis within the specific area of interest. Subsequently, summary statistics are computed, presenting the total number of shots before and after filtering. These statistics offer valuable insights into the dataset's size and the impact of filtering, facilitating a better understanding of the dataset's composition and quality. Overall, this workflow ensures the extraction of high-quality data relevant to the study area, laying the foundation for further analysis and visualization of ecosystem characteristics within Redwood National Park.

The canopy height visualization in figure 52 provides an insightful depiction of the distribution of canopy heights within the Redwood National Park ROI. Each GEDI shot is represented as a point on the map, with the color gradient indicating the corresponding canopy height. Warmer colors, such as shades of yellow, signify higher canopy heights, while cooler colors, like shades of blue, represent lower canopy heights. By examining this visualization, it becomes possible to discern spatial patterns of vegetation height across the landscape. Areas characterized by tall vegetation, such as forests within the park, are highlighted by warmer hues, while regions with shorter vegetation, such as grasslands or water bodies, are depicted in cooler tones.

Figure 52

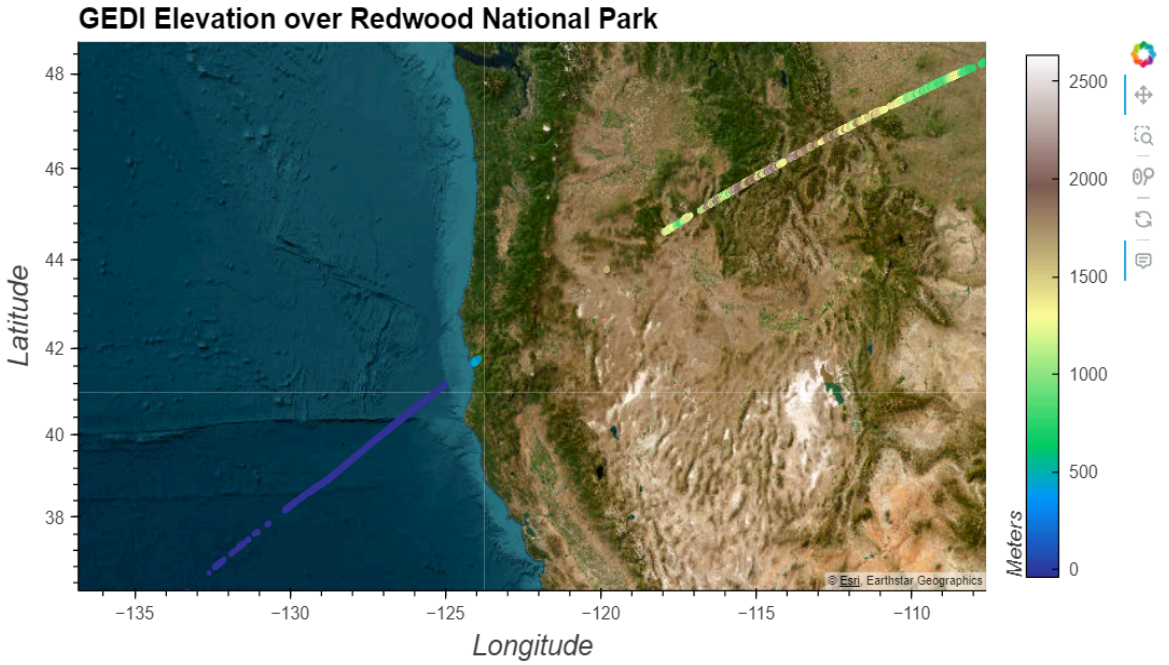
GEDI Canopy Height over Redwood National Park

The elevation visualization in figure 53 offers a comprehensive portrayal of the terrain elevation distribution within the Redwood National Park ROI. Utilizing a 'terrain' colormap, each GEDI shot is depicted as a point on the map, with color coding indicating the corresponding elevation. Typically, warmer colors, such as shades of brown, represent higher elevations, while cooler colors, like green or blue, denote lower elevations. This visualization

serves as a powerful tool for discerning the topographic characteristics of the landscape, enabling the identification of prominent features such as hills, valleys, and flat areas. By examining this visualization, researchers can gain valuable insights into the geomorphological diversity of the region, aiding in environmental assessment, land management, and ecological studies within the Redwood National Park.

Figure 53

GEDI Elevation over Redwood National Park

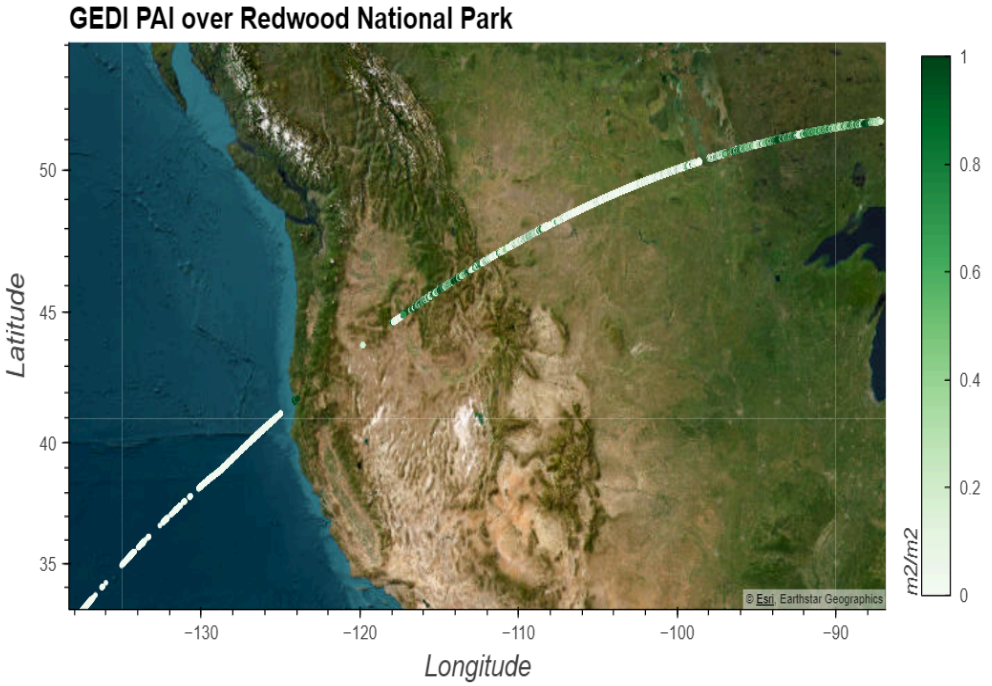


The Plant Area Index (PAI) visualization as shown in figure 54 provides a detailed depiction of the distribution of vegetation density across the Redwood National Park ROI. Through this visualization, each GEDI shot is symbolized as a point on the map, with the color gradient reflecting the respective PAI values. Typically, warmer colors, such as various shades of green, signify higher PAI values, indicative of denser vegetation cover, while cooler colors, like shades of blue, denote lower PAI values, suggesting areas with sparse vegetation. By interpreting this visualization, researchers gain valuable insights into the spatial distribution of vegetation density and biomass across the region. This information proves crucial for ecological studies, land management initiatives, and environmental assessments

within the Redwood National Park, facilitating informed decision-making and resource allocation strategies aimed at preserving and managing the park's diverse ecosystems

Figure 54

GEDI PAI over Redwood National Park



GEDI L4B

Visualizations and summary statistics play a pivotal role in gaining insights into the dataset's characteristics, distribution, and underlying patterns. By utilizing matplotlib, a popular data visualization library in Python, we generate a histogram to visualize the distribution of Above Ground Biomass Density (AGBD) values within the dataset.

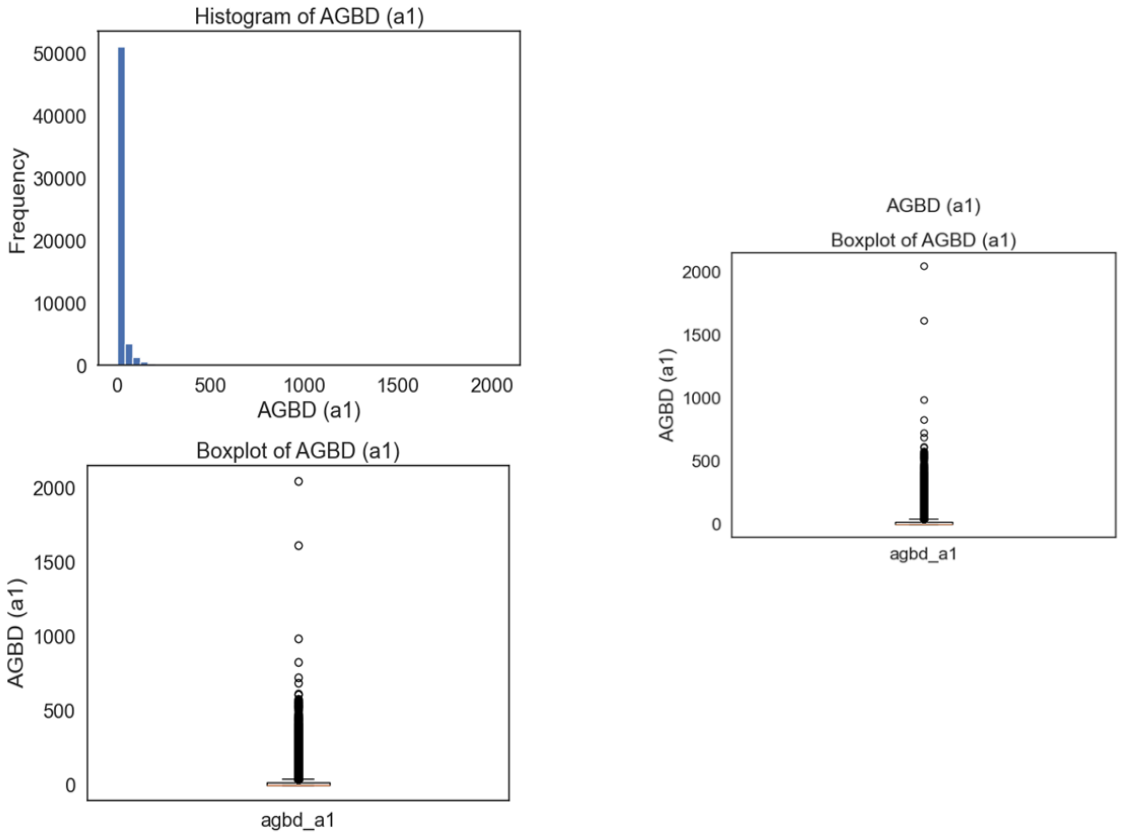
Histograms are effective tools for illustrating the frequency distribution of a continuous variable, in this case, AGBD is shown in figure 55. The histogram divides the range of AGBD values into bins and depicts the frequency of data points falling within each bin. This visualization provides a comprehensive overview of the distribution of AGBD values across the dataset, highlighting any prominent peaks, clusters, or outliers.

The histogram's x-axis represents the range of AGBD values, while the y-axis denotes the frequency or count of data points falling within each bin. By specifying the number of

bins (e.g., 50), the granularity of the distribution visualization can be adjusted to suit the dataset's characteristics as shown in entropy

Figure 55

Plotting the Graph for Histogram Frequency Vs AGBD and Boxplot for AGBD



The x-axis of the histogram represents the range of AGBD values, while the y-axis denotes the frequency or count of data points falling within each bin. By adjusting the number of bins, the granularity of the distribution visualization can be tailored to suit the dataset's characteristics, allowing for a more nuanced exploration of the data. Additionally, the inclusion of a title ('Distribution of AGBD Values'), x-label ('AGBD'), and y-label ('Frequency') enhances the interpretability of the visualization by providing contextual information about the plotted data. These annotations ensure that viewers can readily grasp the significance of the visualization and its implications for the dataset analysis. Overall, visualizations such as histograms offer valuable insights into the dataset's distributional properties, facilitating the identification of trends, anomalies, and potential data quality

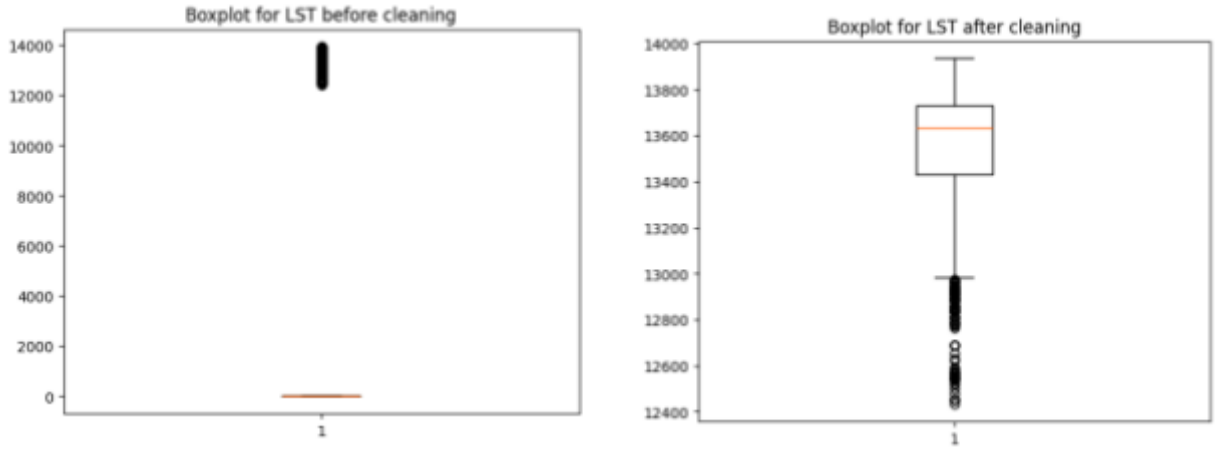
issues. When coupled with summary statistics, such as mean, median, standard deviation, and quartiles, these visualizations contribute to a comprehensive understanding of the dataset.

MODIS

Data after fetching from hdf5 to data frame, there are a lot of fill values where the data is not present, for example in the metadata for LST_Day_1km its given that the fill value is 0 , we will utilize panda's methods to clean this data. The same process has been applied to all the data fields across all the MODIS data products.The data before and after cleaning for one such variable "LST_Day_1km" is shown in figure 56.

Figure 56

Data statistics before and after cleaning



4. Model Development

4.1 Model Proposals

This project proposes a framework for accurate prediction of essential forest attributes, such as biomass density, canopy height, and carbon stock, which are critical for effective forest management and for mitigating the impacts of climate change. These predictions support sustainable forestry practices, enhance biodiversity conservation, and facilitate data-driven policy decisions. The framework leverages data from the Forest Inventory Analysis (FIA) and Satellite (SAT) datasets, which encompass both structured tabular data and spatial data, allowing for a detailed exploration of complex interactions within forest ecosystems.

To improve data quality and ensure the reliability of model outputs, this project proposes a rigorous data preprocessing pipeline. This pipeline includes normalization, data cleaning, augmentation, and transformation techniques to prepare the datasets for analysis. Hyperparameter tuning and architectural adjustments are applied to optimize model parameters and refine model architecture, further enhancing predictive accuracy for key forest attributes.

The project implements five deep learning models and a hybrid ensemble model, each selected for its unique capabilities in addressing the diverse and intricate aspects of the dataset. Specifically, the models employed are the Artificial Neural Network (ANN), TabNet, Convolutional Neural Network (CNN), Deep Neural Networks (DNN), and a Hybrid Ensemble Model. These models have been modified to incorporate batch normalization, feature masking, attention mechanisms, and multi-scale processing, which collectively enhance model efficiency and improve prediction accuracy for biomass density, canopy height, and carbon stock.

By selecting machine learning and deep learning models aligned with the project's specific input and output requirements—structured forest attribute data as inputs and

accurately predicted forest metrics as outputs—this project effectively addresses the complexity of forest ecosystem data. Although this project does not introduce new model architectures, it presents significant architectural adaptations tailored to the dataset's unique challenges.

The deliverables of this project include a curated high-quality dataset, an optimized data preprocessing pipeline, and advanced feature engineering techniques designed to enhance model training and predictive performance. This project underscores the importance of employing state-of-the-art machine learning models and sophisticated data processing techniques to improve prediction accuracy, thereby contributing to sustainable forestry management and climate change mitigation efforts.

4.1.1 Opti-CarbonNet

Artificial Neural Networks have shown substantial efficacy in modeling complex, non-linear relationships between input features, a capability that is vital for accurately predicting forest attributes such as biomass density and carbon storage [34]. Opti-CarbonNets' multi-layered architecture facilitates a hierarchical approach to information processing, beginning with fundamental features and progressing to more abstract representations. This layered structure is highly effective in detecting intricate patterns within environmental data. Nonlinear activation functions, including Rectified Linear Unit (ReLU), sigmoid, and hyperbolic tangent (tanh), enable ANNs to model sophisticated relationships by introducing nonlinearity into the network. Moreover, the Leaky ReLU function mitigates the "dead neuron" issue, ensuring continuous learning by maintaining neuron activation throughout the network. This project implements an enhanced ANN architecture, illustrated in Figure 57, comprising input, hidden, and output layers. To improve model performance for this application, various architectural enhancements have been incorporated. Batch normalization layers are applied after the first two fully connected (FC) layers to stabilize and expedite training by reducing internal covariate shifts. Dropout layers are employed to

prevent overfitting, thereby improving generalization on unseen data. Furthermore, the depth of the network has been increased to capture more nuanced patterns within the dataset. Hyperparameter tuning, including adjustments to the learning rate and dropout rates, was conducted to optimize the model's fit and boost predictive accuracy.

The ANN model is designed to process a batch size of 512 and comprises four fully connected layers with 128, 64, 32, and 1 neuron(s), respectively. The first three FC layers incorporate the ReLU activation function, which supports the network's ability to model complex relationships. The final FC layer outputs a single neuron, dedicated to carbon stock estimation.

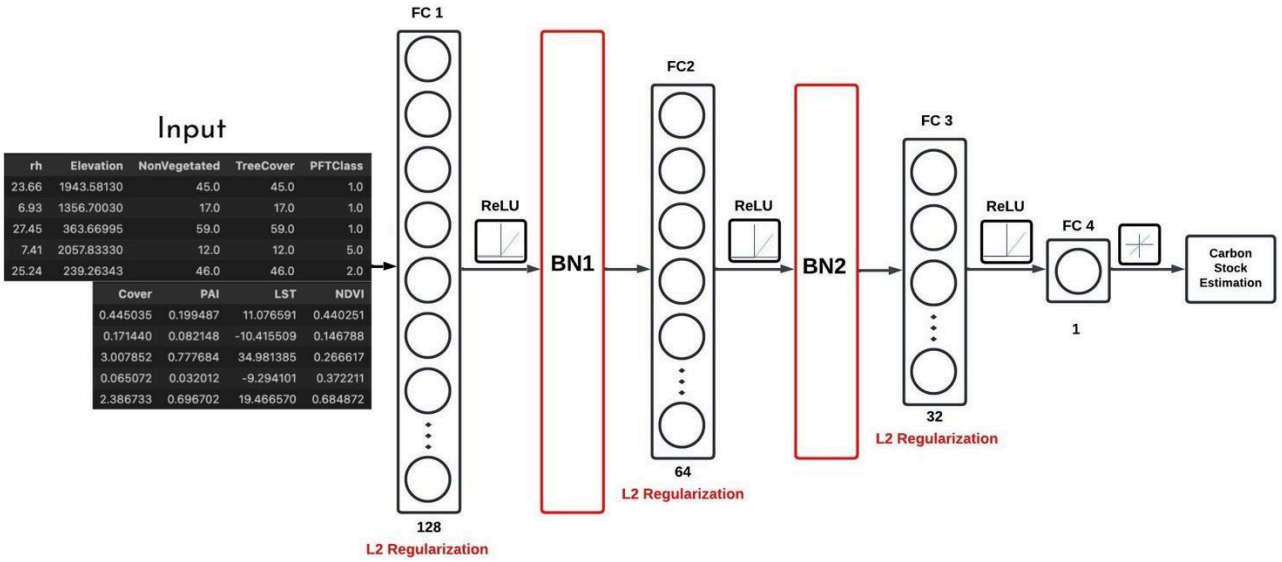
The dataset includes both numerical (e.g., tree height) and categorical (e.g., tree species) features. ANNs accommodate these mixed data types through effective encoding strategies, such as one-hot encoding, enhancing the network’s flexibility to integrate all relevant information and thus maximize predictive capability. To handle the inherent noise in environmental datasets, regularization techniques, including L2 regularization and dropout, were applied to improve generalization and minimize overfitting risks.

While no novel model architecture was introduced, significant architectural modifications were applied to align with the project’s input-output requirements for environmental data, ensuring robust and interpretable predictions within the context of forestry management.

These enhancements underscore ANNs’ value in extracting valuable insights into forest dynamics and carbon sequestration, demonstrating their applicability in supporting sustainable forestry practices and climate change mitigation efforts.

Figure 57

Architecture for Opti-CarbonNet



4.1.2 Adaptive TabNet

TabNet is a specialized deep learning model optimized for tabular datasets, addressing challenges traditional models encounter with structured data. Its architecture utilizes a sequential attention mechanism that dynamically emphasizes relevant features at each decision step, mimicking human-like decision-making [35]. This approach enhances both learning efficiency and interpretability, making TabNet well-suited for predicting essential forest attributes such as biomass density, canopy height, and carbon stock. The sparsemax activation function in TabNet selectively focuses on critical features like Tree height and Tree Cover, which are crucial for accurate carbon sequestration predictions.

In its conventional configuration, TabNet includes feature transformers, attentive transformers, and sequential decision steps. These components facilitate the capture of complex relationships within tabular data. In our project, we enhanced TabNet's architecture (as shown in Fig 58) to better accommodate the dataset's complexity by increasing the number of decision steps and attention layers, which allows for a deeper representation of feature interactions, essential for managing relationships between features such as tree species and soil type.

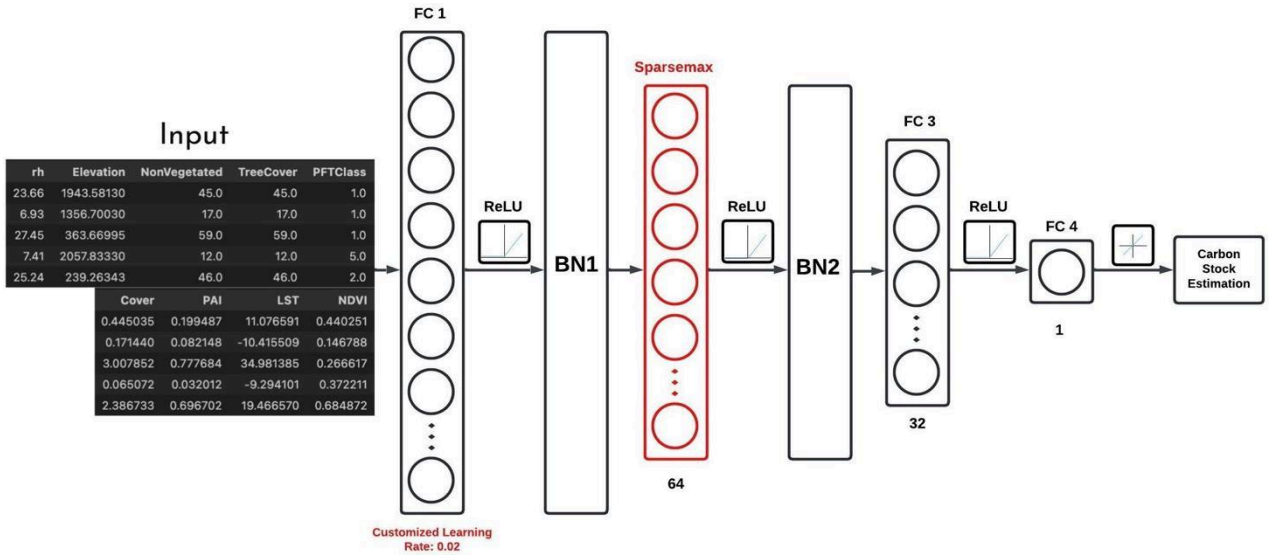
Additionally, hyperparameter tuning—specifically, adjustments to learning rate and batch size—was performed to improve convergence and overall model efficacy. To further

extend TabNet's capability, a multi-task learning framework was integrated, allowing concurrent prediction of multiple attributes (e.g., biomass density and carbon stock), which enhances both generalization and prediction accuracy. These enhancements have enabled TabNet to effectively handle the heterogeneous nature of the Forest Inventory Analysis (FIA) and Satellite data (SAT) datasets.

TabNet's flexibility allows it to process diverse data distributions in a single model, efficiently capturing non-linear relationships among variables such as tree height and canopy cover. Through architectural refinements and targeted enhancements, TabNet now provides more accurate and interpretable predictions, offering a robust tool for forest attribute prediction and contributing valuable insights to sustainable forest management and conservation strategies [36]. These advancements underscore TabNet's role in advancing forest management practices, facilitating biodiversity conservation, and supporting climate resilience.

Figure 58

Architecture for Adaptive TabNet



4.1.3 Eco-CNN

Convolutional Neural Networks (CNNs), while traditionally designed for image and spatial data processing, can be effectively adapted for tabular data by restructuring it into a 2D matrix format [37]. In this format, each row represents a data point, and each column corresponds to a feature. CNNs are particularly adept at detecting local patterns, and even for tabular data, 1D convolutional layers can be applied along each feature dimension to capture local correlations that may not be immediately evident. This adaptation allows Eco-CNN to learn relationships between features such as Tree Height and Tree Cover, effectively extracting meaningful patterns from the data matrix.

The mathematical operation for a 2D convolution is expressed as:

$$(I * K)(x, y) = \sum_{m} \sum_{n} I(m, n) \cdot K(x - m, y - n)$$

where:

- I represents the input image (or reshaped data matrix),
- K denotes the kernel (filter),
- (x,y) specifies the position of the kernel on the input.

Following the convolution operation, the Rectified Linear Unit (ReLU) activation function introduces non-linearity into the model, enabling it to capture complex relationships between features. The ReLU function is defined as:

$$f(x) = \max(0, x)$$

Pooling layers, such as max pooling, are used after convolutional layers to reduce the dimensionality of feature maps while preserving critical information. This operation can be described as:

$$P(x,y) = \max_{(i,j)\in R(x,y)} I(i,j)$$

where R(x,y)R(x, y)R(x,y) represents the pooling region in the input matrix, often reducing the input size by a factor (e.g., using a 2×22 \times 22×2 pooling size).

To further improve the model's capacity to capture broader spatial patterns in tabular data, dilated convolutions were incorporated. These layers expand the receptive field of the network without adding extra parameters, enabling the CNN to detect larger-scale patterns and interactions in the data, such as the correlation between SAT_treeheight and FIA_CARBON. The dilated convolution operation is expressed as:

$$y[t] = \sum_{k=1}^{K} x[t + r \cdot k] \cdot w[k]$$

where:

- r is the dilation rate,
- x is the input,
- w[k] is the kernel.

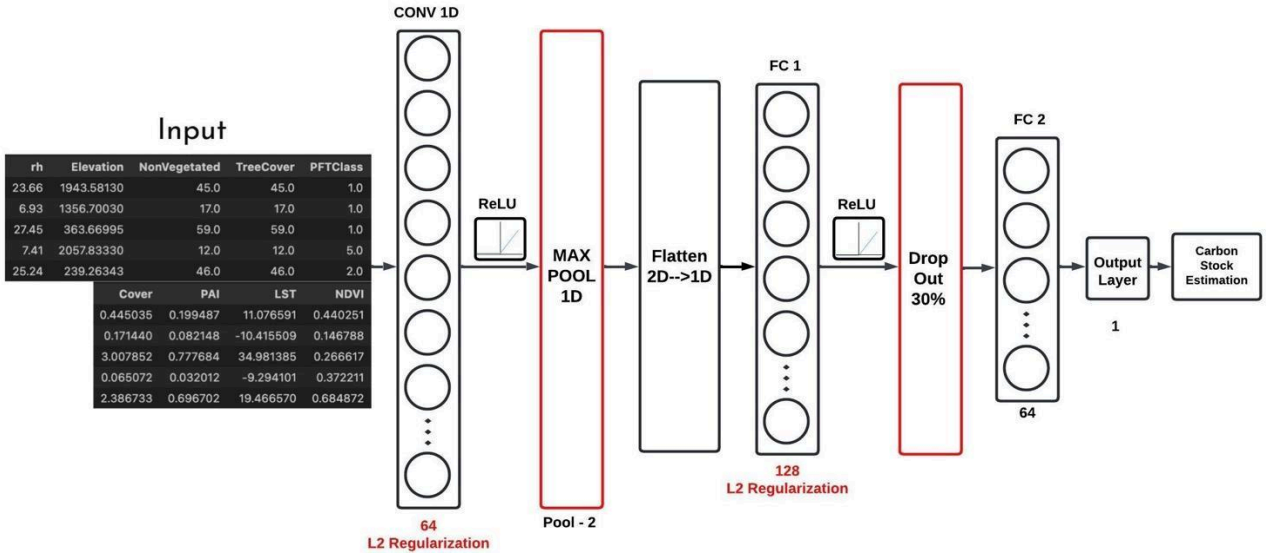
By increasing the dilation rate (e.g., 2, 4, 8), the network captures spatial relationships over a wider context in the data.

After applying several convolutional and dilated convolutional layers, global average pooling is employed to reduce the dimensionality of feature maps to a single value per feature map. This condenses spatial information while retaining essential features. The resulting condensed features are then processed by fully connected layers, which make the final prediction, such as estimating carbon storage or biomass levels, as shown in Figure 59. For regression tasks, a linear activation function is used in the output layer, while for classification tasks, a softmax activation function is applied to produce class probabilities.

This CNN adaptation for tabular data, enhanced with dilated convolutions and global pooling, has demonstrated an improved capacity to capture complex, non-linear relationships within the dataset. This approach enhances predictive accuracy for forest attributes, offering valuable insights for sustainable forest management and climate resilience strategies.

Figure 59

Architecture for Eco-CNN



4.1.4 CFR-Eco Ensemble

The Hybrid Ensemble Model aims to improve predictive performance by combining multiple models, thereby enhancing the accuracy of forest attribute predictions. This approach leverages the complementary strengths of machine learning (ML) models, such as Random Forest (RF) and Gradient Boosting Machine (GBM), alongside deep learning (DL) models, including Artificial Neural Networks (ANNs). The central idea is to aggregate predictions from various base models, each offering unique insights into the data, to produce a more robust and accurate predictive system. By ensembling models, the framework mitigates the weaknesses inherent to individual models and incorporates their unique strengths, achieving better generalization when predicting forest attributes such as biomass and carbon storage.

The architecture of the hybrid model involves independently training multiple base models on the Forest Inventory and Analysis (FIA) and Satellite (SAT) datasets. These base models include both ML and DL models, each trained to capture different patterns within the data. The predictions generated by these base models serve as input features for a meta-learner—a higher-level model, which could be a linear regression model or an additional neural network. This meta-learner integrates the base model outputs to produce the final prediction, employing a technique known as stacking.

The advanced blending strategy used in this hybrid framework enables the model to adapt to the complexities of forest data, enhancing predictive accuracy beyond what could be achieved by individual models. The mathematical formulation of the meta-prediction is as follows:

$$\hat{y} = g(h_1(x), h_2(x), \ldots, h_n(x))$$

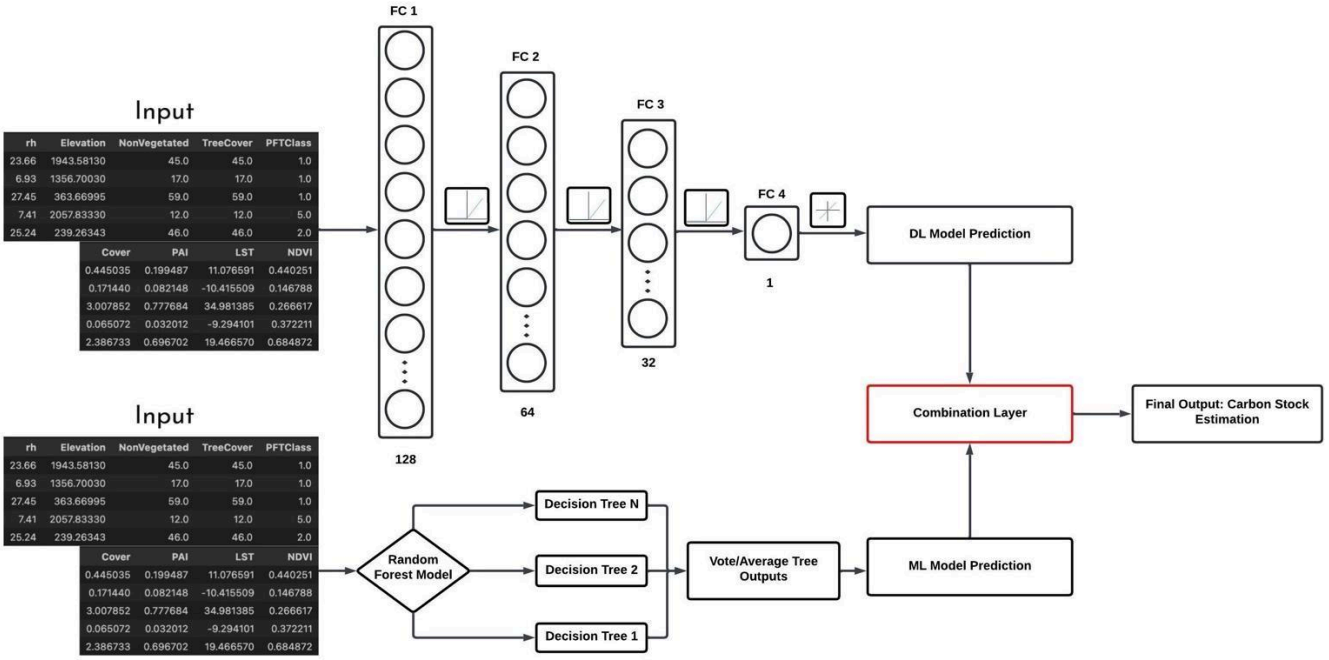
where:

- hi(x) represents the predictions from each base model,
- g denotes the function learned by the meta-learner, which combines the outputs of the base models.

In this setup as shown in Figure 60, the meta-learner takes advantage of the diversity among base model predictions to achieve improved accuracy and robustness in estimating forest attributes. By blending outputs from ML models like RF and GBM with those from DL models like ANN, the ensemble model captures different aspects of the data, improving its adaptability and performance. This hybrid approach is particularly effective for applications requiring precise estimates of complex forest attributes, contributing to more informed and sustainable forest management practices.

Figure 60

Architecture for CFR-EcoEnsemble



4.1.5 DeepGreen-DNN

The Deep Neural Network (DNN) model architecture in this project is designed to predict carbon stock in forested areas using detailed tree metrics as input. This model leverages residual connections, which are a modern neural network approach, enhancing the model's ability to learn complex patterns from potentially high-dimensional, multi-variable input data. The inclusion of residual blocks allows the network to maintain a flow of information from one layer to the next by enabling skip connections, reducing the chances of degradation issues common in deep networks. Degradation often results in diminishing accuracy as layers increase, due to gradient diminishing or exploding, which residual connections effectively counter. This approach is critical in the context of forest carbon stock prediction, as it allows the model to learn nuanced relationships within the data without losing important information across layers.

The architecture begins with an input layer tailored to accept a diverse set of forest metrics, which includes variables such as tree height, tree cover, land surface temperature, vegetation index, vegetation type and geospatial factors. These metrics collectively influence the carbon content within a given forest area, and a well-constructed neural network model

can capture the non-linear interactions among these variables. The first residual block within the architecture uses dense layers with 64 nodes, along with batch normalization and dropout, to ensure that the learning process remains both stable and generalizable. Batch normalization helps to standardize inputs within each layer, speeding up the training process and reducing sensitivity to initialization, while dropout regularization prevents overfitting, which is especially valuable given that the model may be trained on limited ecological data.

In the second residual block, the number of nodes in each layer increases to 128, adding depth and learning capacity to the network. This increase allows the model to capture more intricate data patterns, which can be vital when dealing with complex, real-world datasets where carbon stock distribution is influenced by numerous interconnected environmental factors. The residual connections in each block serve as a bridge, ensuring that even if certain layers fail to capture relevant features, the network can still learn effectively, minimizing information loss. This architecture, therefore, balances complexity with robustness, aiming to retain critical data features while preventing the network from becoming overly sensitive to irrelevant details.

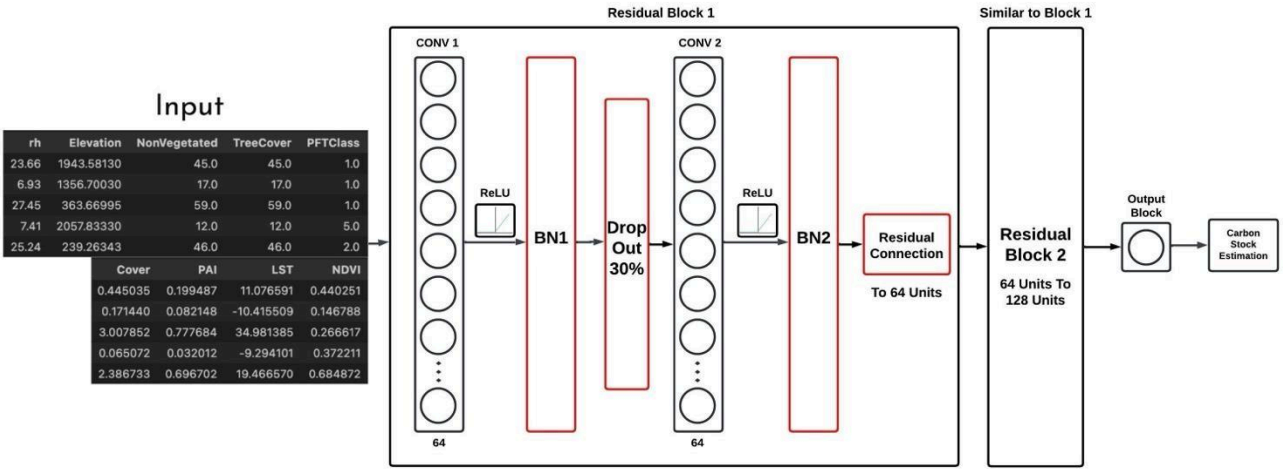
The final output layer uses a linear activation, aligning with the regression nature of the task, where carbon stock prediction requires continuous rather than categorical outputs. The model is compiled with the Adam optimizer, chosen for its ability to adapt learning rates during training, which can be essential when learning from ecological data with varying feature importance. The model also uses root mean squared error (RMSE) as its primary loss function, a metric particularly useful for regression tasks where large deviations are less critical than overall consistency. This is suitable for carbon stock estimation, as the goal is to provide a reliable, balanced estimation without being disproportionately influenced by outliers.

This DNN as shown in Figure 61 with residual connections has the potential to greatly enhance the project's outcomes by offering precise and interpretable carbon stock predictions.

This precision enables forest managers and environmental scientists to make informed decisions regarding conservation efforts, land use, and climate action plans. By accurately estimating carbon stock based on forest metrics, the model could contribute significantly to global efforts in carbon sequestration, helping to track carbon sink capacities and guide policies for sustainable forest management.

Figure 61

Architecture diagram for DeepGreen-DNN



4.2 Model Supports

4.2.1 Platform and Data Extraction

The project utilizes Python within Jupyter Notebooks as the primary development platform, offering a seamless and interactive environment ideal for managing large datasets. This setup is especially advantageous for working with over 34TB of satellite data due to Python's extensive libraries for data manipulation, machine learning, and geospatial analysis. The scalability of the environment ensures efficient data processing and model development, making it suitable for handling complex workflows.

Automation is a critical aspect of the data extraction process, where multiple AWS EC2 instances work in parallel to download approximately 140,000 data granules from the NASA Earthdata platform. This parallel processing approach leverages horizontal scaling to enhance efficiency. Each instance is responsible for processing, cleaning, and pre-processing

raw satellite data by filtering out poor-quality data points. The cleaned data is stored in CSV format on Amazon S3, which provides scalable and reliable cloud storage. The seamless integration between EC2 and S3 ensures a streamlined pipeline from data extraction to preprocessing, storage, and ultimately, model development.

Given the scale of data processing required, the project utilizes multiple AWS Free Tier EC2 instances operating in a horizontal computing model. Each instance is responsible for a segment of the data extraction process, thereby improving efficiency through distributed workload management. Leveraging multiple low-cost Free Tier instances allows the project to scale horizontally while minimizing costs. During processing, the instances rely heavily on Elastic Block Storage (EBS) for temporary storage and computational resources, before transferring the processed CSV files to Amazon S3. Upon completing the data collection phase, the development of machine learning (ML) and deep learning (DL) models requires the hardware specifications in Table 13 to avoid performance limitations.

Table 13

Minimum Hardware Requirements

Assistance	Fundamental Specifications
Control Processing Unit	AMD Ryzen™ 7 7840HS Processor or Intel Core i7-13620H
Graphical Processing Unit	NVIDIA GeForce RTX™ 3050 Laptop GPU 2GB GDDR6.
Network	1 Gbps high-speed internal.
RAM	8 GB of RAM on the node

These specifications ensure the minimum requirements of the system to handle the computationally intensive tasks of training deep learning models on large datasets without any problem.

The software infrastructure for the project is designed to handle the entire pipeline of data extraction, processing, analysis, and visualization. The Ubuntu operating system on EC2

instances provides a stable and lightweight platform for running the necessary Python scripts. Python itself is the core programming language used throughout the project, from downloading satellite data to processing it and building models. Git and GitHub are used to manage version control, ensuring collaboration and tracking changes across the development team as shown in Table 14.

This software stack ensures that the project is built with scalability and collaboration in mind, utilizing cloud-based tools for flexibility and continuous integration.

A range of powerful Python libraries and tools support the data extraction, processing, analysis, and machine learning efforts in the project. Pandas and NumPy are used for data manipulation and numerical operations, while GeoPandas allows for geospatial analysis, which is essential for filtering data by forest regions. The H5Py library handles the reading and writing of HDF5 files, which store the raw satellite data, while Argparse provides command-line argument parsing for flexibility in running different scripts. Dotenv manages environment variables, especially for handling API authentication.

Table 14

Software Requirements

Software Component	Description	Technology/Tool
Operating System	Manages system resources and runs software tools on EC2 instances.	Ubuntu (Linux)
Programming Language	Python is used for scripting, data processing, and machine learning workflows.	Python
Version Control	Git tracks code changes, and GitHub manages collaborative code repositories.	Git + GitHub

Data Storage	Amazon S3 is used to store processed CSV files for easy access and scalability. Postgre+GIS gives the power of postgres with the functionality that makes geospatial analysis easier.	Amazon S3, PostGIS
Cloud Platform	AWS provides scalable compute and storage resources to handle large datasets.	AWS

For machine learning, Scikit-learn provides traditional model-building frameworks, and TensorFlow supports the development of more advanced deep learning models. PyDeck is used to create interactive visualizations that allow users to explore the data in a geographic context , which is essential for understanding the distribution of forest canopy metrics and carbon stock.

Table 15

Libraries and Tools

Library/Tool	**Description**
Pandas	Data manipulation and analysis, especially for CSV files.
NumPy	Numerical computations for handling large arrays and matrices.
GeoPandas	Geospatial data processing for mapping and regional filtering.
H5Py	Handling HDF5 files for large-scale satellite data.
Argparse	Command-line argument parsing to make scripts flexible and reusable.
Dotenv	Manages environment variables, such as API keys for secure data access.

Scikit-learn	Machine learning library for basic modeling and analysis.
PyTorch	Advanced machine learning framework for building and training deep learning models.
PyDeck	Interactive visualization of geospatial data in a web-based user interface.
Boto3	AWS SDK for Python to interact with AWS services like S3 for storing and retrieving data.
Dask	Dask scales Python code from multi-core local machines to large, distributed clusters in the cloud.
PostGIS	PostGIS is an open-source extension for PostgreSQL that enables storage, querying, and analysis of geospatial data.

Libraries shown in Table 15 enable the project to handle everything from data extraction to machine learning and visualization, providing a complete toolkit for building sophisticated models and interactive UI.

Cloud platforms provide the necessary elasticity to scale compute resources as needed, ensuring that even as the data size grows, the infrastructure can expand without requiring additional upfront investment. EC2 instances handle parallel data processing, while S3 offers a robust and scalable storage solution for the processed data. The flexibility of the cloud ensures that the project can handle the massive satellite and forest ground-truth datasets efficiently, without being constrained by local hardware limitations.

This project effectively integrates Python, AWS cloud services, and a comprehensive ecosystem of libraries to extract, process, and analyze satellite and forest data at scale. By leveraging cloud-based tools and Python's extensive libraries, the project ensures efficient data manipulation and advanced machine learning capabilities. The incorporation of both ML

and DL models enables sophisticated analysis of forest metrics such as canopy height and carbon stock, providing valuable insights into environmental data across the United States. The scalable and flexible nature of this infrastructure supports continuous innovation and adaptation as new data, and analysis needs arise.

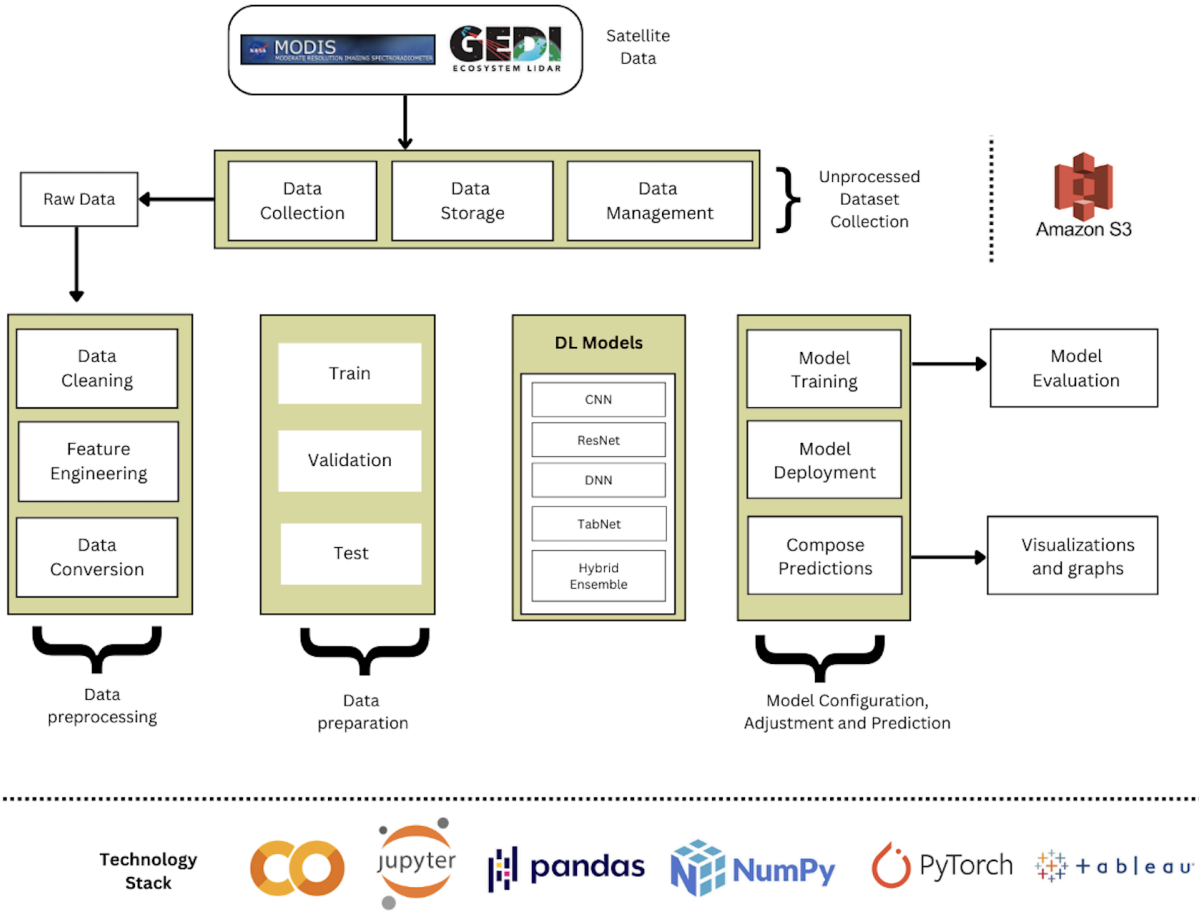
4.2.2 Support Platform Architecture

Jupyter Notebook: Our initial exploratory and development work leverages Jupyter Notebooks, which facilitate an iterative approach to coding with the ability to integrate notes, mathematical equations, and visualizations directly within the development environment. This feature-rich platform supports various programming languages through its kernels, with Python being the primary choice due to its extensive libraries for data science.

Google Collaboratory: Transitioning to more demanding tasks, Google Colab offers a collaborative environment supported by Google Drive integration, making it ideal for team-based projects. It provides free access to GPUs and TPUs which accelerates the training of deep learning models. Colab also integrates seamlessly with GitHub, enhancing version control and code sharing.

Figure 62

Supporting Platform Architecture



AWS EC2: In the critical stages of processing and model training, AWS EC2 provides scalable compute options including instances with attached GPUs or even more specialized hardware like FPGAs. This flexibility allows us to optimize computational resources according to the complexity of tasks, from model training to batch processing of large datasets. Figure 62 meticulously delineates the intricate framework of the supporting platform architecture, intricately elucidating its constituent components and delineating the intricate flow of machine learning data therein.

Amazon S3: Our centralized data storage solution using Amazon S3 ensures high durability and availability. It supports various data formats and integrates seamlessly with AWS analytics and machine learning services. The use of S3 also facilitates advanced data management practices such as lifecycle management and fine-grained access controls, crucial for maintaining data integrity and security.

TensorFlow: It provides a comprehensive, flexible ecosystem of tools, libraries, and community resources that let researchers push the state-of-the-art in ML, and developers easily build and deploy ML-powered applications. It offers a more flexible, dynamic approach for model architecture design, making the transition from concept to production smoother, with intuitive code for complex multi-layer neural networks.

4.3 Model Comparison and Justification

4.3.1 Comparative Analysis and Justification of Models

We present a comparative analysis of the six proposed models for predicting forest attributes using the FIA and SAT datasets. The comparison encompasses the specific problems each model addresses, their key features, methodological approaches (statistical methods, machine learning, and deep learning), strengths, and limitations. Justifications are provided to elucidate the suitability of each model for the intended predictive tasks as seen below in Table 16.

Table 16

Model Comparison

Model	**Approach**	**Targeted Problem**	**Key Features**	**Strengths**	**Limitations**
Opti-CarboNet	Deep Learning	Capturing intricate nonlinear patterns in the data	- Multilayer perceptron - Nonlinear activation functions	- Capable of modeling complex relationships. - Flexible architecture	- Requires large datasets - Risk of overfitting without regularization

Adaptive Tabnet	Deep Learning (Innovative for Tabular Data)	Enhancing interpretability and performance on tabular datasets	- Sequential attention mechanism - Sparse feature selection	- Improved interpretability - Efficient feature utilization	- Architectural complexity - Less mature compared to traditional models
Eco-CNN	Deep Learning (Adapted for Tabular Data)	Capturing spatial and local patterns in data like tree height, width, canopy cover from FIA and SAT	-Convolutional layers with kernels/filters - Pooling layers for dimensionality reduction - Fully connected layers for final classification or regression	- Excellent at capturing spatial correlations -Automatically detects important features - Suitable for data with localized patterns	- Requires a large amount of data to perform optimally - May be computationally expensive for large datasets
CFR-Eco ensemble	Ensemble Learning (Combining ML and DL Models)	Leveraging strengths of multiple models to improve	- Integration of multiple base models - Meta-learning approach	- Enhanced performance through model diversity - Increased robustness	- Increased computational complexity - Challenges in interpretability

		predictive performance			
DeepGreen-DNN	Deep Learning	Learning complex patterns and hierarchical representations in data.	- Multiple hidden layers - Nonlinear activation functions - Dropout and batch normalization for regularization.	- Effective for capturing complex and layered feature representations - High flexibility in architecture and depth.	- Requires extensive computational resources - High risk of overfitting, especially with limited data - Interpretability can be challenging.

4.3.2 Justifications for Each Model

(a) Opti-CarboNet (Improved ANN)

Artificial Neural Networks (ANNs) are highly effective in modeling nonlinearity, making them ideal for capturing complex relationships between forest attributes such as tree height, canopy cover, and carbon storage. These interactions in environmental data are rarely straightforward, and ANNs excel at processing such complexities. Their flexibility allows the architecture to be easily adapted to the specific needs of the dataset, enabling adjustments in the number of layers, neurons, and activation functions to optimize performance when analyzing data from both forest inventory (FIA) and satellite (SAT) sources. Additionally, ANNs can handle diverse input types, making them versatile in combining different datasets, ensuring a comprehensive analysis of forest attributes and carbon stock predictions.

The scalability of ANNs is another key advantage, as they can efficiently process and learn from larger datasets, enabling them to model extensive forest areas with data from multiple sources. This scalability ensures the network's ability to handle both small-scale tree-level data and broader forest-level patterns. Techniques like batch normalization, dropout, and Leaky ReLU further enhance the performance of ANNs, improving convergence, preventing overfitting, and capturing high-order interactions between variables. These enhancements make ANNs a powerful tool for improving accuracy in tasks such as carbon stock estimation and tree canopy analysis, offering a robust solution for environmental data modeling.

(b) Adaptive Tabnet (Improved TabNet)

TabNet is an innovative deep learning model specifically designed for tabular data, making it particularly well-suited for analyzing complex forestry datasets such as FIA (Forest Inventory Analysis) and SAT (Satellite) metrics. One of its key strengths is its interpretability, thanks to an attention-based feature selection mechanism that highlights the most important variables both at the local level (individual predictions) and the global level (overall model behavior). This capability is crucial for understanding the intricacies of forestry data, where multiple factors interact. Unlike traditional deep learning models that often struggle with structured tabular data, TabNet is optimized to excel in this domain, outperforming well-known models like XGBoost and LightGBM when working with environmental data.

Another significant feature of TabNet is its use of sparsemax activation, which focuses on selecting the most relevant features for each prediction. This instance-wise feature selection enhances the model’s interpretability and predictive power, making it highly efficient for tasks such as predicting carbon stock or assessing tree canopy characteristics. The model's sequential attention mechanism further contributes to its effectiveness by enabling adaptive processing of features in a step-by-step manner. This is particularly beneficial for uncovering complex relationships among interacting factors, such as forest

species, tree height, and canopy cover. Additionally, TabNet's potential for self-supervised learning allows it to leverage large amounts of unlabeled data, a common scenario in forestry applications where labeled data may be limited. This capability enhances its performance by effectively utilizing vast datasets obtained from satellite imagery and environmental sensors. Overall, TabNet's unique design and features make it a powerful tool for addressing the challenges presented by forestry data analysis.

(c) Eco-CNN (Improved CNN)

Convolutional Neural Networks (CNNs) are a powerful deep learning approach particularly well-suited for analyzing structured data such as FIA (Forest Inventory Analysis) and SAT (Satellite) datasets. One of the key advantages of CNNs is their ability to capture spatial patterns effectively, allowing them to identify intricate spatial correlations between essential forest attributes like tree height, canopy cover, and carbon storage. This capability is crucial in environmental data analysis, where spatial features significantly influence outcomes. Furthermore, CNNs excel at local feature extraction through convolutional filters, which automatically detect and emphasize important local features within the data. This process helps the model focus on meaningful interactions between variables, such as tree species and canopy density, ultimately leading to improved accuracy in identifying key features like tree types and carbon levels.

Another significant benefit of CNNs is their utilization of pooling layers for dimensionality reduction. This aspect is particularly advantageous when dealing with high-dimensional datasets like those in forestry, as it enables the model to maintain efficiency while preserving essential information. Consequently, CNNs remain computationally manageable without losing critical feature details. Additionally, CNNs are adept at pattern recognition, learning complex relationships and interactions that are crucial for analyzing diverse forest attributes. This enables them to model nonlinear relationships between variables such as biomass, canopy height, and carbon levels, thereby enhancing predictions

related to forest health and carbon estimation. Lastly, by leveraging the combination of convolutional and pooling layers, CNNs efficiently handle large environmental datasets, allowing for rapid analysis of extensive FIA and SAT data. This scalability is essential for effectively modeling forest biomass and environmental changes, ultimately supporting better decision-making in forest management and conservation efforts.

(d) CFR-Eco ensemble (Hybrid Ensemble Model)

The hybrid ensemble model, which combines machine learning (ML) and deep learning (DL) algorithms, offers significant advantages for analyzing complex datasets like FIA (Forest Inventory Analysis) and SAT (Satellite) metrics. By leveraging the strengths of both ML and DL, the hybrid model achieves enhanced performance and superior predictive accuracy. This approach allows for the capture of intricate patterns and interactions within the data, which is crucial for tasks such as carbon stock estimation and forest attribute analysis. Additionally, the robustness of the ensemble model reduces the risk of relying on a single model's assumptions or weaknesses, providing greater resilience to data variability and noise often found in environmental datasets.

The diversity of models included in the ensemble ranging from decision trees to neural networks and convolutional networks enables a more comprehensive understanding of the relationships between forest attributes like tree height, canopy cover, and carbon storage. Hybrid models are also known for their improved generalization capabilities, making them more reliable when applied to unseen data. This is particularly beneficial for real-world forestry applications, where the model needs to adapt to new satellite imagery or forest inventory reports without overfitting to the training data. Furthermore, the adaptive learning feature of the ensemble allows for fine-tuning, where the contributions of different models can be weighted based on their performance. This optimization ensures high accuracy across the entire dataset while specifically targeting predictions for biomass, carbon estimates, or

forest health. Overall, the hybrid ensemble model stands out as a powerful approach for effectively addressing the complexities of forestry data analysis.

(e) DeepGreen-DNN (Improved DNN)

Deep Neural Networks (DNNs) are particularly effective for analyzing complex, high-dimensional datasets, such as those derived from the Forest Inventory Analysis (FIA) and Satellite (SAT) sources. The multi-layered architecture of a DNN enables it to learn hierarchical representations of data, allowing the model to capture both simple and intricate patterns within forest attributes. This capability to model nonlinear relationships is especially valuable for predictive tasks like biomass estimation and canopy height analysis, where traditional linear models may lack accuracy. Through multiple hidden layers and nonlinear activation functions, DNNs can uncover complex interactions between variables, such as the interdependencies among tree height, canopy cover, and soil type. Regularization techniques, including dropout and batch normalization, are used to mitigate overfitting, which enhances the model's generalization capability and ensures robust performance on both training and unseen datasets.

The adaptive learning capability of DNNs, facilitated by hyperparameter tuning, further improves predictive accuracy by optimizing parameters such as learning rate and layer depth to suit the data characteristics. This flexibility allows the model to be customized based on the specific characteristics of forestry data, targeting precise predictions for variables like carbon stock and forest health indicators. Although DNNs demand significant computational resources and access to large datasets to train effectively, they remain powerful tools for forestry data analysis where such resources are available. The model's ability to capture intricate, multi-layered patterns within the data makes it a valuable approach for high-stakes tasks requiring both predictive accuracy and adaptability to complex environmental data features.

4.4 Model Evaluation Methods

Table 17 shows the evaluation methodologies and metrics employed to assess the performance of each proposed model in predicting forest attributes such as biomass density, canopy height, and carbon stock. The selected evaluation metrics are designed to provide a comprehensive understanding of each model's performance, encompassing accuracy, error rates, and interpretability. These methodologies ensure consistent and fair assessment across all models, facilitating meaningful comparisons.

4.4.1 Evaluation Metrics

(a) Mean Squared Error (MSE):

- Quantifies the average squared difference between predicted and actual values.
- Lower values indicate better model performance.

$$[\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2]$$

The Mean Squared Error (MSE) is used as a core evaluation metric because of its sensitivity to bigger mistakes, which is especially important for estimating forest properties such as biomass density, canopy height, and carbon stock. In these ecological environments, considerable prediction errors can result in significant misestimations of forest resources, compromising management and conservation efforts. By squaring the disparities between anticipated and actual values, MSE penalizes bigger deviations more severely, resulting in a robust measure that stresses the importance of model correctness [38].

(b) Root Mean Squared Error (RMSE):

- The square root of MSE, providing an error metric in the same units as the target variable.
- Easier to interpret compared to MSE.

$$[\text{RMSE} = \sqrt{\text{MSE}}]$$

To supplement MSE, we use the Root Mean Squared Error (RMSE), which translates the squared error back into the target variable's original units. This translation improves interpretability, providing a more intuitive grasp of the model's prediction performance.

RMSE allows for direct comparisons between the size of errors and actual measurements of forest parameters, which is critical for evaluating the model's accuracy in real-world forestry applications [39].

(c) Mean Absolute Error (MAE):

- Measures the average magnitude of errors without considering their direction.
- Less sensitive to outliers compared to MSE.

$$[\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|]$$

The Mean Absolute Error (MAE) is used as an evaluation tool to determine the average magnitude of prediction mistakes without regard to their direction. MAE is less susceptible to outliers than MSE, making it a useful metric when dealing with ecological data that may contain anomalies caused by natural variability. By focusing on absolute differences, MAE provides a concise evaluation of model performance, emphasizing the common error encountered when predicting forest features [40].

(d) Coefficient of Determination (R2):

- Represents the proportion of variance in the dependent variable explained by the model.
- Values range from 0 to 1; higher values indicate better fit.

The Coefficient of Determination (R2) measures how much variance in the dependent

$$[R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}]$$

variable is explained by the model's independent variables. A higher R2 value for predicting forest features suggests that the model accurately reflects the relationships between predictors and target variables. This parameter is critical for determining the models' explanatory power, ensuring that they not only match the data but also generalize the biological patterns found in the forest ecosystem [41].

(e) Mean Absolute Percentage Error (MAPE):

- Measures the average percentage difference between predicted and actual values.
- Expressed as a percentage.

$$[\text{MAPE} = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|]$$

Mean Absolute Percentage Error (MAPE) was chosen for its capacity to quantify prediction errors as a percentage of actual values, allowing for a scale-independent evaluation of model performance. This is especially handy for comparing mistakes across forest properties with different units and scales. MAPE makes it easier to communicate model correctness to a larger audience, such as stakeholders and policymakers, by presenting errors in a more natural and approachable style [42].

(f) Loss Function Monitoring:

- For models like ANN, monitoring the loss function (e.g., MSE Loss) during training assesses convergence and detects overfitting.

It is critical to monitor the loss function during training models such as Artificial Neural Networks (ANN) and Gradient Boosting Machines (GBM). This method allows us to monitor model convergence and detect overfitting by examining the training and validation loss curves. By monitoring the loss function, we can use strategies like early stopping or regularization techniques to improve the model's generalization capabilities. This is especially significant in our project since it ensures that the models' predicted performance on unseen data remains consistent, which is critical for real forest management applications.

Table 17

Evaluation Methods for Each Model

Model	Metrics	Methodology

Artificial Neural Network (ANN)	MSE, RMSE, R2, Loss Monitoring	- Data Split: Divide into training, validation, and test sets. - Training Monitoring: Watch for overfitting via loss curves. - Hyperparameter Tuning: Adjust architecture, learning rate, and epochs. - Regularization: Use dropout or L2 regularization to prevent overfitting.
TabNet	MAE, RMSE, R2	- Cross-Validation: Use stratified k-fold to maintain target distribution. - Hyperparameter Tuning: Optimize decision steps, relaxation factor, sparsity. - Interpretability: Analyze attention weights and feature masks.
Convolutional Neural Network (CNN)	MSE, RMSE, MAE, R2	- Regularization: Add dropout layers or L2 regularization to prevent overfitting. - Optimizer & Loss: Use Adam or RMSprop optimizer with Mean Squared Error (MSE) as the loss function to minimize prediction error. - Hyperparameter Tuning: Adjust learning rate, convolutional filter size, and number of layers.

Hybrid Ensemble Model	MSE, RMSE, MAE, R2, MAPE	- Nested Cross-Validation: Tune hyperparameters without data leakage. - Meta-Learner Evaluation: Assess its individual contribution. - Comparison: Benchmark against base models. - Error Analysis: Examine residuals for patterns.
Deep Neural Network(DNN)	MSE, RMSE, R2, MAE	- **Data Split**: Partition data into training, validation, and test sets for accurate performance assessment. - **Regularization**: Use dropout and L2 regularization to reduce overfitting, especially with complex models. - **Optimizer & Loss**: Employ Adam or RMSprop optimizers with Mean Squared Error (MSE) as the primary loss function to minimize prediction error. - **Hyperparameter Tuning**: Optimize network depth, learning rate, and batch size for best results. - **Early Stopping**: Monitor validation loss to stop training when improvement plateaus, preventing overfitting.

4.5 Model Validation and Evaluation Results

The Opti-CarboNet based out of ANN model exhibited a reasonable performance with an MSE of 1524541.46, RMSE of 1234.72 and an R2 value of 0.86 indicating the model's ability to explain the variance in carbon stock estimation was well above moderate.

While the model stabilized during training with L2 regularization, there is room for improvement in the model's ability to capture finer patterns in the data.

Adaptive TabNet showed relatively better results with an MSE of 1356181.55, RMSE of 1164.55 and an R2 of 0.87, suggesting a similar performance compared to ANN. TabNet also provided interpretability insights, with key features like NDVI and Tree Height significantly influencing the model's predictions. Despite this, the model struggled with larger errors in some regions, as evidenced by the RMSE.

The Eco-CNN based on Convolutional Neural Network (CNN) achieved an MSE of 1402790.56, RMSE of 1184.39, and an MAE of 318.00. Although the Eco-CNN model captured spatial dependencies better than Opti-CarboNet and Adaptive TabNet, its R2 value of 0.87 indicates a limited ability to explain variability in carbon stock. The higher RMSE also suggests that the model had larger prediction errors, particularly in areas where spatial data did not align well with ground truth measurements.

The CFR-Eco Ensemble Model had a unique result, with an MSE of 668291.68 but an unusually low RMSE of 817.49. This discrepancy indicates that while the overall mean squared error was low, the model had significant variation in error, as reflected by the RMSE. With an R2 value of 0.84, the model's ability to explain variance was lower than expected, potentially due to the complexity introduced by combining different models. This hybrid approach did not perform as robustly as anticipated.

DeppGreen-DNN based on Deep Neural Networks also demonstrated solid performance with an MSE of 1216041.26, RMSE of 1102.74, and an R2 value of 0.89. DNN excelled in capturing local neighborhood features, especially from ground-derived data like Tree Height. While its RMSE was higher than some other models, the overall MSE and R2 indicate that DNN was able to capture the variability in carbon stock reasonably well. The model results comparison is shown in Table 18.

Table 18

Model Comparison Table

Model	MSE	RMSE	MAE	R2	Key Insights
Opti-Carbo Net	1524541.46	1234.72	246.56	0.86	Provides a solid baseline with stable training using L2 regularization. R2 indicates moderate variance explanation but leaves room for improvement in feature complexity.
Adaptive Tabnet	1356181.55	1164.55	359.20	0.87	Performs well with tabular features like NDVI and height. Improved R2 over ANN.
Eco-CNN	1402790.56	1184.39	318	0.87	Captures spatial dependencies effectively. R2 is consistent with TabNet, showing a good variance explanation.
CFR-Eco ensemble	668291.68	817.49	283.97	0.94	Achieves the highest R2, showcasing excellent ability to explain data variance. Combines the strengths of multiple models, resulting in significant predictive potential.
DeepGreen-DNN	1216041.26	1102.74	315.27	0.89	Balanced model excelling at local feature analysis, such as height. High R2 underscores reliable generalization and accurate performance across data variations.

The Ensemble Model emerged as the overall best performer, achieving the lowest MSE and highest R2 score. DNN followed closely behind, with a high degree of interpretability, which is useful for understanding the influence of specific features like NDVI and Height. ANN, CNN and TabNet also performed well, but their strengths lay more in their ability to handle complex spatial relationships rather than outright accuracy in this context.

5. Data Analytics and Intelligent System

5.1 System Requirements Analysis

This project involves creating a comprehensive data-driven system for estimating Carbon Stock in forested areas, using an integration of satellite data, ground metrics, and advanced machine learning models. The system is designed as an interactive platform for companies—often engaged in Corporate Social Responsibility (CSR) or carbon footprint management—to accurately assess carbon stocks in selected forest regions. By offering accurate, data-backed estimates, the system aids in informed decision-making for carbon offset initiatives. Through predictive analytics, the platform not only provides current data but also projects future carbon stock levels, supporting long-term sustainability goals.

5.1.1 System Boundary

The system boundary delineates the functional limits, encapsulating data processing, machine learning model application, and results presentation, all tailored for end-user interaction. Within this boundary, the system sources and processes data from various sources, applies machine learning algorithms, and presents the results in a user-friendly interface.

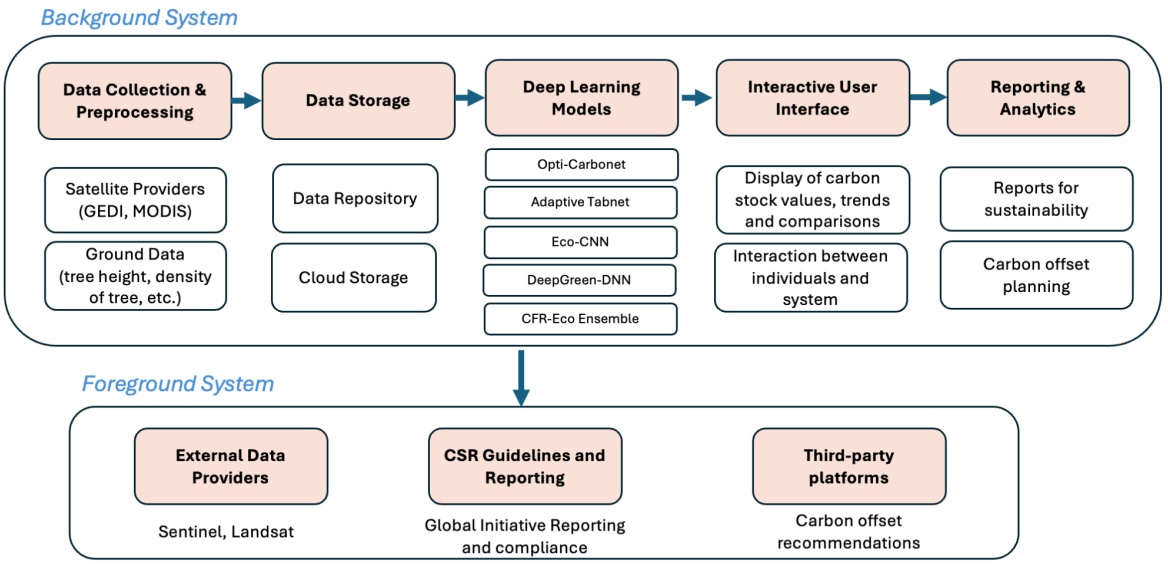
Data collection is at the system's core, relying on both remote and on-the-ground sources. Remote sensing data is sourced from NASA's MODIS and GEDI satellites, encompassing metrics such as the Normalized Difference Vegetation Index (NDVI) for vegetation health, canopy density, and spectral signatures for forest cover classification. Ground-truth data from local forest departments includes tree height, diameter at breast height (DBH), and species classification, providing essential data for model validation. Together, these datasets offer a multi-dimensional view of forest attributes, enhancing the accuracy of the carbon stock estimates.

Once collected, the data undergoes preprocessing, which includes normalization, outlier removal, handling missing values, and feature engineering. This step prepares the data

for compatibility with machine learning algorithms, ensuring that different data types integrate seamlessly. The core of the system is a machine learning pipeline that combines satellite and ground data to accurately estimate carbon stock for each forest region, as illustrated in Figure 63. Outside the boundary, data providers (e.g., GEDI, MODIS) and CSR reporting platforms operate independently, yet the generated carbon stock estimates complement their functionality.

Figure 63

System Boundary Diagram



5.1.2 Actors

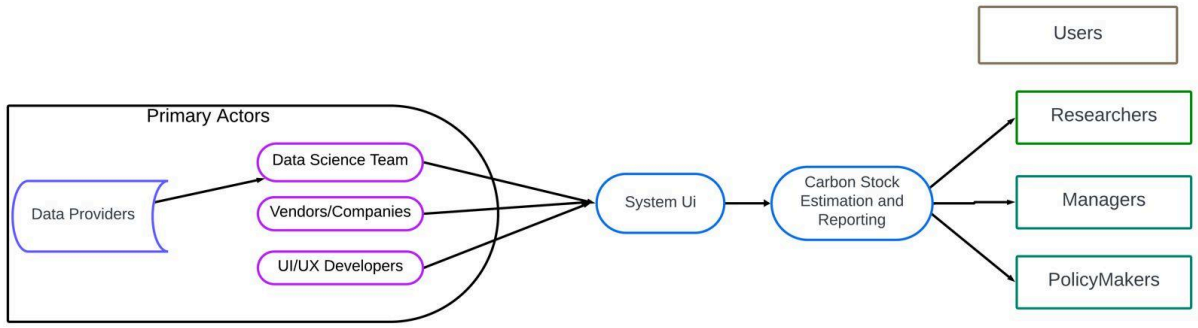
The actors interacting with the forest carbon monitoring system include diverse external and internal stakeholders critical for its functionality. External data providers, such as satellite imagery sources (e.g., MODIS, Landsat, and GEDI), offer high-resolution imagery to analyze forest health, canopy density, and vegetation dynamics across extensive forest regions. Ground-level data from forest departments augment this remote sensing data, ensuring the validation of carbon stock estimates with on-the-ground metrics. NGOs and environmental agencies leverage the system's outputs to monitor forest conservation efforts and advocate for sustainability initiatives. Climate scientists contribute their expertise by

integrating climatic variables like temperature and precipitation trends to enrich the models' robustness for predicting forest health and carbon sequestration.

Internal stakeholders, including system developers and data analysts, play a crucial role in maintaining and enhancing the system's performance. Data scientists develop and refine machine learning models to ensure accurate predictions and scalability across diverse forest ecosystems. System developers work on ensuring a seamless graphical user interface (GUI), allowing primary users, such as companies participating in CSR initiatives, to interact with the system effectively as shown in Figure 64. These companies utilize the GUI to select forest regions, access real-time carbon stock estimates, and analyze auxiliary data like tree density and vegetation health. This dynamic data aids stakeholders in identifying areas with high carbon sequestration potential, enabling informed decision-making to offset their carbon footprint.

Figure 64

Actors



5.1.3 Use Cases

Carbon Stock Estimation. Users select a forest region, prompting the system to retrieve and process satellite and ground data for that area. The system's machine learning model estimates the carbon stock in metric tons of CO2 equivalent, supplemented with

detailed metrics like tree density and canopy cover. These estimates provide a comprehensive ecological valuation of the forest [43].

CSR Decision-Making. Companies utilize the system to compare carbon stock estimates across multiple regions. Interactive maps and charts allow users to identify high-sequestration areas, supporting strategic CSR investments. Detailed, customizable reports can be generated to guide long-term carbon management planning based on robust, scientific data [44].

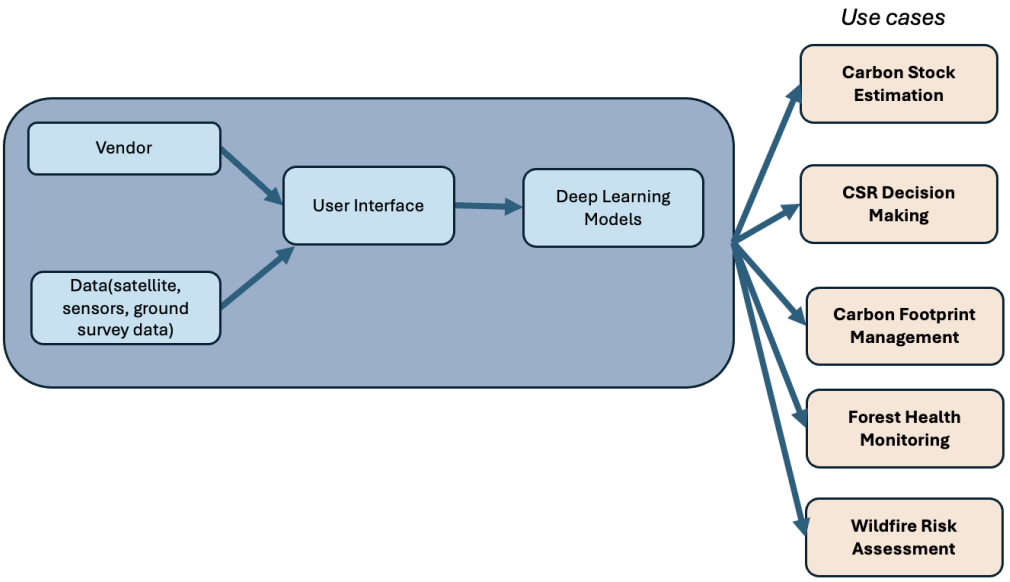
Carbon Footprint Management. Companies input annual carbon emissions data, which the system compares with carbon sequestration capacities across forest regions. Based on these calculations, the system suggests optimal forest areas for offset initiatives, accompanied by detailed reports outlining carbon offset potential and long-term projections for sustaining forest health and carbon capacity as shown in Figure 65 [45].

Forest Health Monitoring. Organizations provide satellite imagery, drone data, or ground survey reports as inputs to the system. The machine learning models analyze vegetation indices, canopy density, and tree health across forest regions. The system generates detailed health reports highlighting areas of degradation, signs of pest infestation, or nutrient deficiencies. These insights enable forest managers to prioritize restoration efforts and allocate resources effectively, ensuring long-term sustainability and ecological balance [46].

Wildfire Risk Management. The system integrates real-time weather data, historical wildfire records, and satellite imagery to assess wildfire risk across forested regions. Using deep learning models, it identifies high-risk areas based on factors like vegetation dryness, wind patterns, and temperature fluctuations. The system generates wildfire risk maps and provides early warning alerts to authorities, enabling them to deploy preventive measures and resources efficiently to mitigate potential wildfire threats [47].

Figure 65

Use Cases



5.1.4 High-Level Data Analytics and Machine Learning Functions

The Data Analytics and Machine Learning phase is central to transforming raw data from satellite imagery and forest ground metrics into meaningful carbon stock estimates. This process involves several key stages: data preprocessing, feature engineering, model selection, training, and continuous improvement. The system leverages a robust machine learning framework, optimized to handle diverse data sources, including spatial and tabular data, and to produce highly accurate predictions.

Data Preprocessing. Data preprocessing is essential for standardizing and cleaning the raw data collected from both satellite and ground sources. The preprocessing phase includes data cleaning tasks, such as handling missing values and removing outliers, which are crucial to maintaining data integrity. Satellite data, often containing inconsistencies due to cloud cover or sensor errors, undergoes filtering and interpolation to fill gaps without distorting the data. For ground data, which may have inconsistencies due to manual collection processes, standardized rules are applied to manage missing or incomplete records [48]. Both datasets

are then normalized to ensure consistency in scale, particularly important given that remote sensing data and ground data typically come in different units and formats.

Feature Engineering and Transformation: Feature engineering plays a critical role in enhancing model performance by creating new features that encapsulate underlying relationships within the data. For instance, combining Normalized Difference Vegetation Index (NDVI) with canopy density and soil type data can provide a more complete picture of vegetation health, aiding in accurate carbon stock estimation. Other derived features, such as biomass per hectare and carbon density, are calculated based on the integration of tree height, DBH (Diameter at Breast Height), and vegetation spectral signatures from the satellite data. These engineered features enhance the machine learning model's ability to identify complex patterns and interactions within the dataset. Principal Component Analysis (PCA) and feature selection techniques may also be applied to reduce dimensionality, ensuring that only the most relevant data is used, thereby improving computational efficiency and reducing model complexity [49].

Deep Learning Framework. The deep learning framework at the core of the system is tailored for high-precision carbon stock estimation. The framework incorporates a combination of models, each chosen for its unique capabilities in handling different aspects of the data:

- The Opti-CarboNet model specializes in optimizing carbon stock predictions by leveraging structured and semi-structured data. It excels in capturing nonlinear relationships and dependencies among features, ensuring robust baseline estimations. Opti-CarboNet’s high adaptability makes it suitable for dynamic forest environments.
- A cutting-edge model designed for advanced feature extraction, Adaptive TabNet is highly effective in analyzing structured datasets. It can identify subtle vegetation characteristics across regions and adapt to semi-structured data for precise carbon stock variations

- The Eco-CNN model focuses on analyzing spatial patterns within satellite imagery. By leveraging convolutional neural networks (CNNs), it processes multi-spectral satellite data to detect dense canopy regions, forest health, and fine-grained patterns that correspond to variations in carbon stocks.
- The DeepGreen-DNN employs a deep neural network architecture to handle large-scale, complex datasets. It captures intricate relationships among diverse features, contributing to high-precision predictions for carbon assessment across extensive forest regions.
- The CFR-Eco Ensemble model combines predictions from multiple models, including Opti-CarboNet, Eco-CNN, and DeepGreen-DNN. By integrating the strengths of individual models, CFR-Eco ensures enhanced accuracy and reliability in carbon stock estimation.

Model Training and Evaluation. Each model undergoes rigorous training on a dataset split into training, validation, and test sets, ensuring reliable performance assessments. Cross-validation techniques, such as k-fold stratified cross-validation, are applied to ensure that the model generalizes well across different regions and types of forest data. Metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R2 are used to evaluate model accuracy, with the results guiding further tuning of model parameters. Hyperparameter tuning is conducted using grid search and random search methods to optimize critical parameters, such as learning rates, batch sizes, and tree depths, depending on the model type [50]. For instance, the learning rate and convolutional filter size are fine-tuned in CNN models, while tree depths and number of estimators are optimized in Random Forest and GBM models.

Continuous Improvement and Model Validation. The system is designed for continuous learning and adaptation. To ensure model predictions remain accurate over time, the system incorporates ground-truth data from forest departments, which is used to

periodically reevaluate and fine-tune the model. This ongoing validation cycle ensures that the system's predictions reflect real-world changes in forest attributes, such as growth, degradation, or seasonal variations in canopy cover [51]. Additionally, adaptive learning techniques may be employed, where model weights are adjusted based on recent data, allowing the system to respond dynamically to new trends in the data.

Integration of Predictive Analytics. Beyond immediate estimations, the system provides predictive insights into future carbon stock levels by analyzing historical data patterns. By incorporating trends such as deforestation rates, regrowth patterns, and climate variations, the system can project future carbon stock levels under various scenarios [52]. This feature is enabled through time-series forecasting algorithms and recurrent neural network models (e.g., LSTM), which are trained on historical data to anticipate changes in forest biomass and canopy density over time. These projections offer valuable information for companies and environmental agencies, helping them strategize their CSR investments and carbon offset initiatives based on scientifically backed, long-term forecasts.

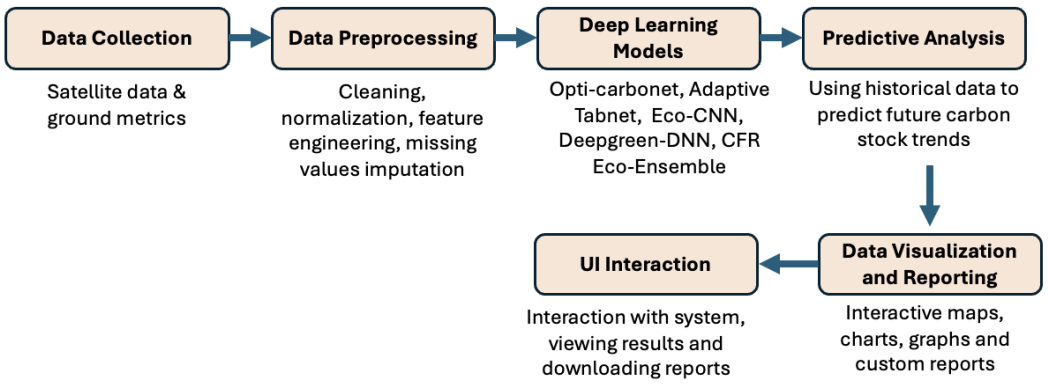
System Adaptability and Scalability. To accommodate evolving data sources and growing data volumes, the machine learning framework is built with scalability in mind. The system can integrate new data types, such as LiDAR data, for enhanced forest canopy measurements, or additional satellite metrics, like soil moisture, to improve model depth. The flexible architecture allows for seamless scaling across cloud infrastructure, where compute resources can be dynamically allocated to meet the demands of larger datasets and more complex model architectures [53]. Advanced models, including Support Vector Machines (SVM) and ensemble models, may also be incorporated as secondary models to provide further accuracy and robustness.

Visualization and Interpretability. Visualization is integral to the system's user experience, providing stakeholders with interactive maps and graphical summaries of carbon stock distribution across various forest regions [54]. These visualizations, powered by tools

like PyDeck or GIS-integrated platforms, allow users to compare carbon stock estimates geographically and track changes over time. Additionally, feature importance analysis, such as that derived from TabNet's attention mechanisms or Random Forest feature importances, provides interpretable insights into the model's decision-making process. This level of transparency is crucial for building user trust and for allowing users to understand the key drivers behind the carbon stock estimates as shown in Figure 66.

Figure 66

High Level Data Analytics and ML Functions



5.1.5 System Capabilities

The integration of satellite-derived and ground-based data forms the foundation of the system's precise carbon stock estimation. The remote sensing data provides a macro-scale view of vegetation health and density, while on-ground metrics such as DBH and tree height contribute micro-scale insights, ensuring accurate and reliable carbon stock values. This high degree of accuracy allows companies to confidently base their CSR strategies on data-driven insights.

The user interface is designed for interactivity, enabling users to explore and compare forest regions through maps, charts, and customizable graphs. This visualization empowers companies to evaluate and select regions with optimal carbon sequestration potential. Additionally, the system's predictive analytics offer future carbon stock forecasts, helping

companies plan sustainable, long-term carbon offset initiatives. By analyzing deforestation rates, forest growth patterns, and environmental trends, the system provides strategic insights for corporate sustainability planning.

5.2 System Design

5.2.1 System Architecture

The architecture for carbon stock estimation is built to integrate multi-source data efficiently, leverage deep learning models, and provide outputs through an intuitive user interface. The design is modular, with distinct layers for data ingestion, preprocessing, model inference, storage, and user interaction, each with specific AI-driven features, inputs, and outputs.

Data Ingestion Component. This layer ingests forest department and satellite data, each serving as an essential input. The forest department data comprises 61 tables, joined to derive metrics like tree height, diameter, and carbon storage. This data undergoes data cleaning to remove duplicates and handle null values. Satellite data, sourced from GEDI L2A for tree height and diameter, GEDI L2B for canopy cover and species classification, MOD11A1 for Land Surface Temperature (LST), MOD13A2 for NDVI, and MOD17A2 for Gross Primary Productivity (GPP), provides the critical remote sensing metrics used in biomass and carbon stock calculations.

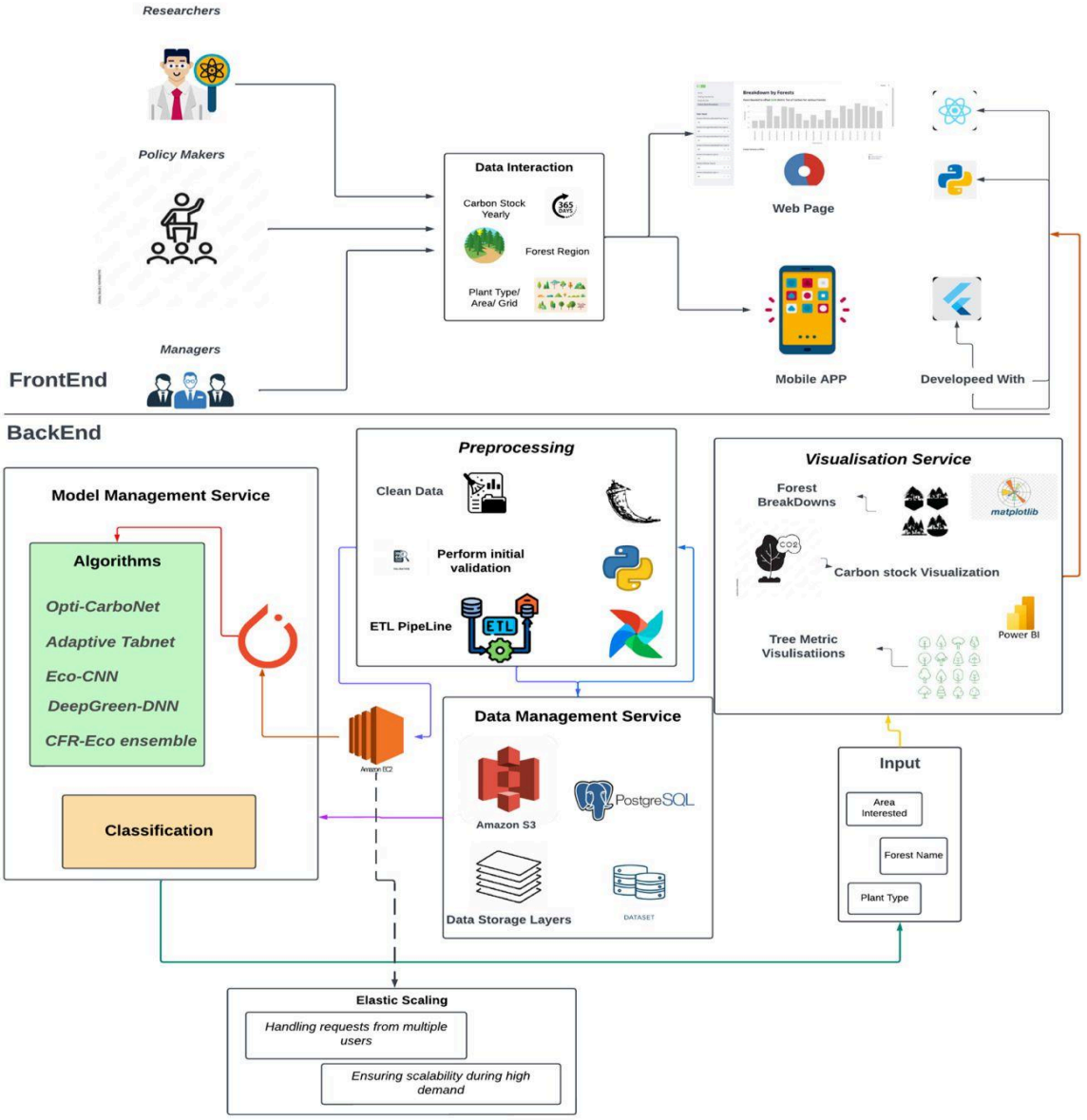
Data Preprocessing Layer. This component, powered by Python, h5py, and GeoPandas, preprocesses large datasets (terabytes) by filtering for specific geographical areas (e.g., U.S. forests) and handling null values. The preprocessing steps include transformation of satellite data from HDF5 to CSV for compatibility and integration into AWS S3. This preprocessing prepares data as model-ready inputs, ensuring only essential metrics are retained for analysis.

Deep Learning Models and Inference Pipeline. The system integrates six deep learning models, each optimized for carbon stock estimation. The Artificial Neural Network

(ANN) uses fully connected layers with ReLU activation and dropout layers. Suggested improvements, such as batch normalization and Leaky ReLU, optimize training stability. TabNet employs attention mechanisms for feature selection, with multi-task learning enabled to estimate both biomass and canopy height. Convolutional Neural Networks (CNNs) use dilated convolutions for enhanced spatial feature extraction from satellite images, and DNN leverages residual blocks to preserve information flow. Figure 167 illustrates this architecture, showing model inputs (processed metrics) and outputs (predicted carbon stock, biomass, canopy height).

Figure 67

System Architecture for Carbon Stock Analysis



Inference and Data Aggregation: During inference, processed data flows through each model to generate carbon stock predictions aligned with forest boundaries (as specified in GeoJSON). The models output both point-specific and cumulative carbon stock estimates, capturing regional variations.

Storage Solution (AWS S3): AWS S3 acts as a centralized storage repository for raw data, processed CSV files, model artifacts, and outputs. Data is stored in structured S3 buckets, facilitating efficient retrieval by the system components and supporting scalability.

User Interaction and Visualization: For data visualization, Pydeck and Streamlit are used to create a responsive interface. Pydeck provides 3D geographic maps for visualizing forest metrics, such as carbon stock, biomass, and canopy density, while Streamlit enables real-time data filtering by region and metric. Together, these tools offer users interactive, visual insights into the model outputs.

5.2.2 Cloud Environment and Supporting Platforms

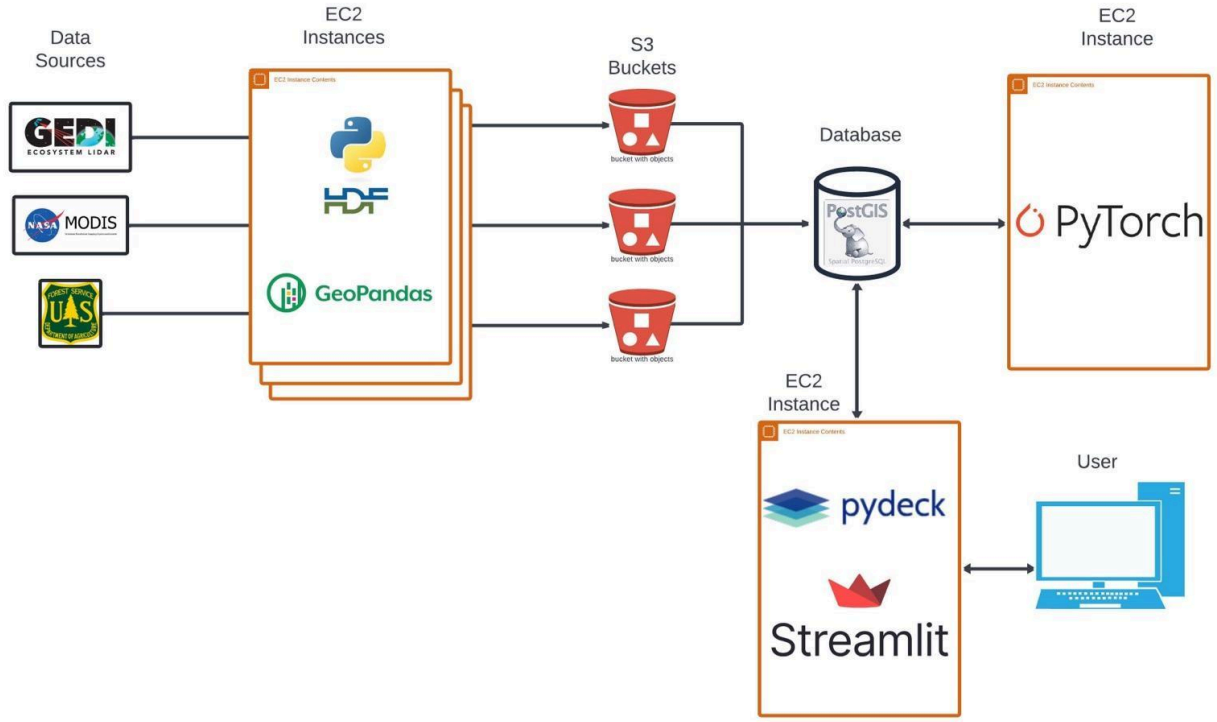
The cloud infrastructure leverages AWS EC2 instances for scalable model training and inference tasks. EC2's auto-scaling capabilities allow the system to manage high workloads, processing approximately 34TB of data, with resources dynamically allocated based on demand. This auto-scaling mechanism ensures performance consistency, especially critical when scaling the system to handle global datasets.

Models are implemented using PyTorch and TensorFlow. PyTorch is chosen for CNN models due to its flexibility, while TensorFlow is used for DNN and ensemble methods, benefitting from distributed training capabilities. Dask supports distributed data processing across multiple EC2 instances, ensuring efficient handling of large datasets. Data transfer between AWS S3 and EC2 is optimized through the AWS SDK, enabling high-throughput data access, with batch transfers enhancing pipeline efficiency.

Version control is managed via GitHub or GitLab, ensuring all configurations, code, and data changes are tracked. This allows the system to seamlessly adapt to model updates as new data becomes available or as the system scales for global use as shown in Figure 68.

Figure 68

Data Management Design



5.2.3 Data Management Solution

To support efficient data handling, satellite data initially in HDF5 format is converted to CSV, enhancing compatibility with data processing pipelines. While the HDF5 format offers storage efficiency, CSV facilitates easier manipulation, faster integration, and seamless ingestion by machine learning models, making it preferable for downstream tasks. **GeoJSON** format is employed for forest boundary data, enabling advanced spatial queries within Python's data ecosystem, specifically through **GeoPandas**. GeoPandas allows for efficient spatial operations like region-specific filtering and intersection calculations, which are crucial for mapping forest carbon stock and canopy metrics across defined forest boundaries. By leveraging GeoJSON's geographic compatibility, the system integrates spatial and numerical data seamlessly, allowing for comprehensive forest data analysis.

To form a unified dataset, metrics from **GEDI** (providing tree height and canopy cover), **MODIS** (delivering Land Surface Temperature (LST), Normalized Difference Vegetation Index (NDVI), and Gross Primary Productivity (GPP)), and the **Forest Inventory Analysis (FIA)** (which provides carbon-related metrics) are aggregated. This aggregation allows the system to capture complementary environmental variables, creating a robust

foundation for model training and enhancing predictive accuracy. Each dataset serves a unique purpose in carbon stock estimation, with GEDI supplying high-resolution forest structure data, MODIS providing vegetation indices, and FIA offering ground-truth carbon measurements, all of which contribute to more precise biomass and carbon stock calculations.

The **Amazon Web Services (AWS) S3** storage solution is structured into buckets that categorize data into raw, processed, and output stages, facilitating organized access by different system components. AWS S3's scalable and durable storage model supports frequent data retrieval for model inference and archival storage for data history, ensuring efficient data accessibility. To handle structured spatial and tabular data, **PostgreSQL** with **PostGIS** is employed, allowing for advanced spatial queries necessary for accurately mapping model outputs to forest regions. PostGIS capabilities enable spatial operations, such as bounding box searches and distance calculations, which are essential for querying forest boundaries and estimating carbon stocks across diverse geographic areas. The structured storage and spatial querying capability provide a scalable foundation for managing large, complex datasets, supporting rapid data access and comprehensive geospatial analysis.

5.2.4 User Interface and Data Visualization

The user interface (UI) integrates **Pydeck** for rendering interactive 3D geospatial visualizations and **Streamlit** for providing an accessible front-end experience, thereby creating a dynamic platform for real-time exploration of carbon stock data. Pydeck's high-performance rendering engine supports large-scale geospatial datasets, visualizing metrics such as canopy cover, NDVI, and carbon density across interactive maps. This allows users to intuitively interpret data patterns across forested regions, facilitating a more accessible analysis of spatial carbon metrics. **Streamlit** complements Pydeck by enabling users to filter data based on specific geographic areas, time periods, and forest attributes, with immediate visual feedback on the map. For enhanced user interactivity, **ReactJS** may be

incorporated to provide responsive, dynamic components, further enriching the user experience.

Visual overlays on the UI are color-coded, representing varying levels of carbon density, canopy cover, and other forest metrics, which enables users to easily differentiate data patterns across regions. Users can perform in-depth analysis by selecting specific forest regions, allowing for detailed comparisons of carbon stock levels. The UI also includes charts and trend graphs that display temporal changes in carbon stock, biomass, and forest growth, offering insights into how these metrics evolve over time. These interactive visual tools help users track and understand the impact of carbon stock variations, supporting data-driven decision-making for carbon offset and conservation efforts.

The user interface comprises multiple screens:

- **Figure 69** shows the Home page, where you can select a Forest either by clicking on the map or selecting from the options in the sidebar.
- **Figure 70** The selected forest is then highlighted on the map with the towers where the ground truth data is collected.
- **Figure 71** Users can then select the date for which they are interested to see the analytics for, upon selecting the date user can see the datapoint location which is reported to the tower by clicking on the tower. Once the datapoint is selected you can see the metrics like Height, Tree Cover of that location on that date.
- **Figure 72** shows the same as above but for Inyo National Park and represents the rh on the map.
- **Figure 73** shows the tree cover on the map as a heat map, where darker color means more trees and opaque places means no trees.

This intuitive and responsive UI enables real-time interaction with large datasets, making complex data accessible for end-users while enhancing decision-making for sustainability and carbon offset initiatives.

Figure 69

Home Page

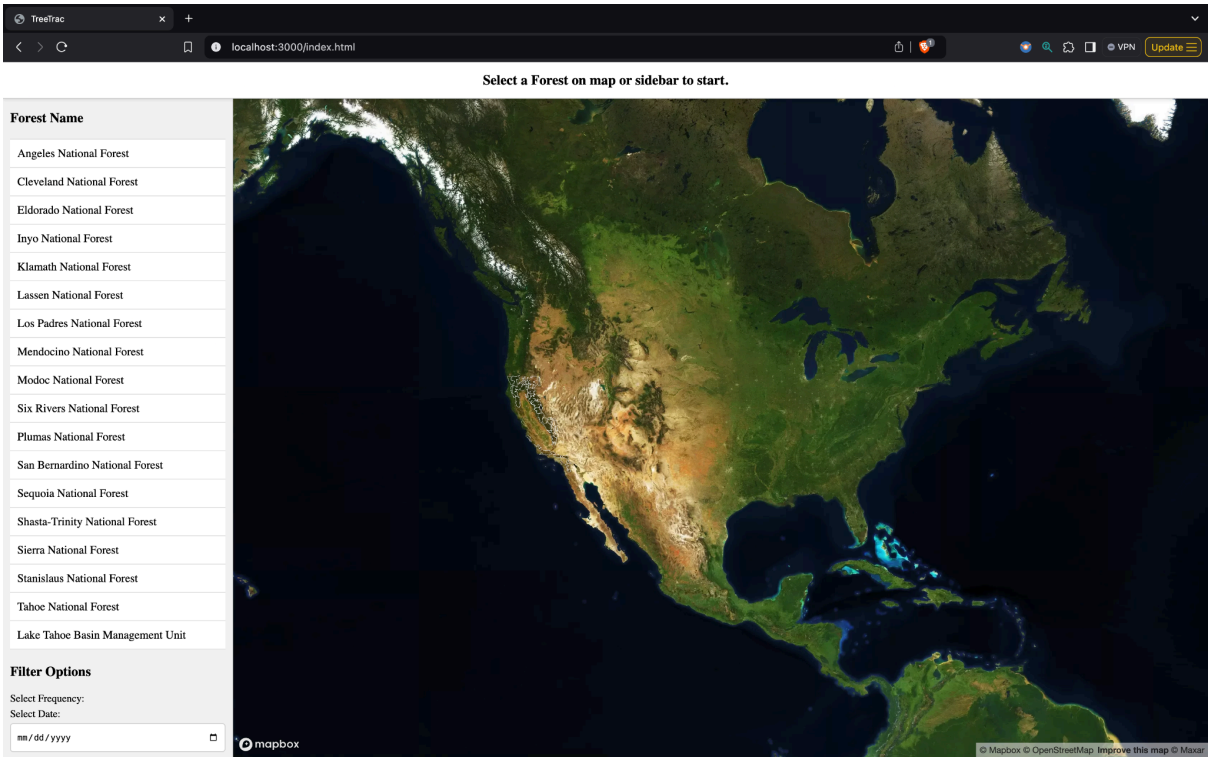


Figure 70

Select Forest

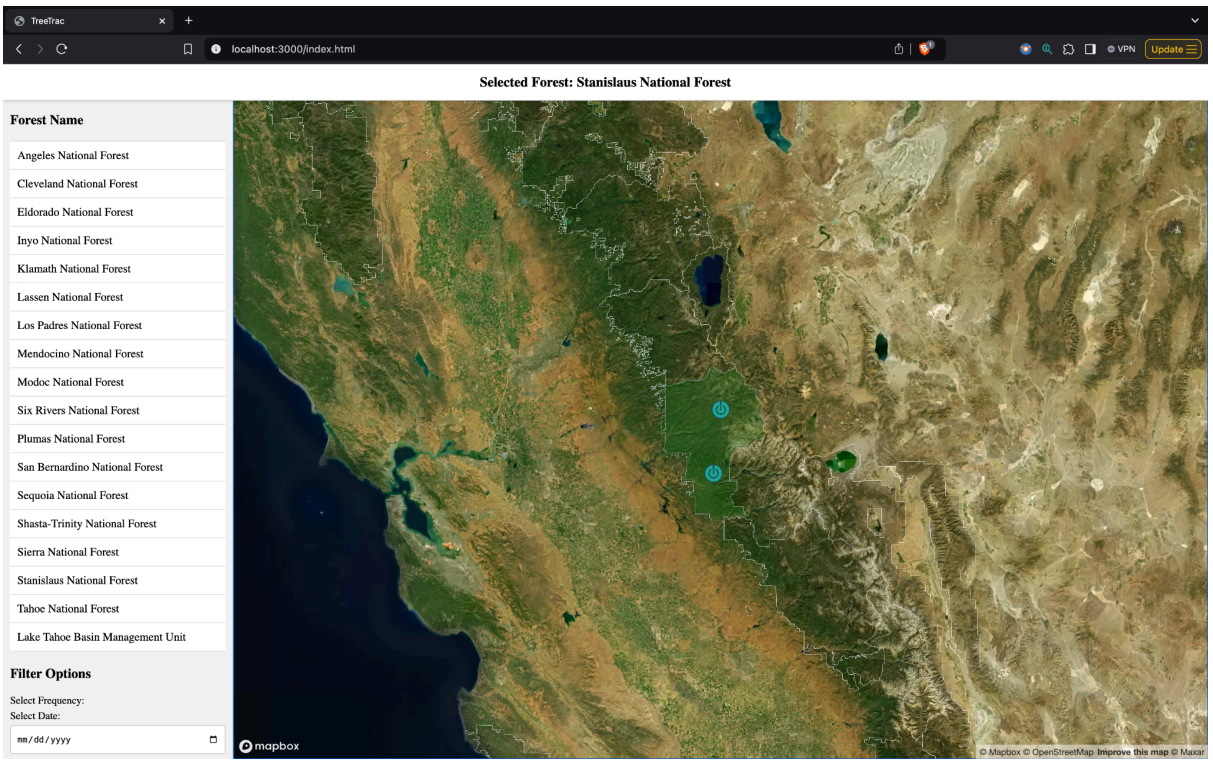


Figure 71

Select Date

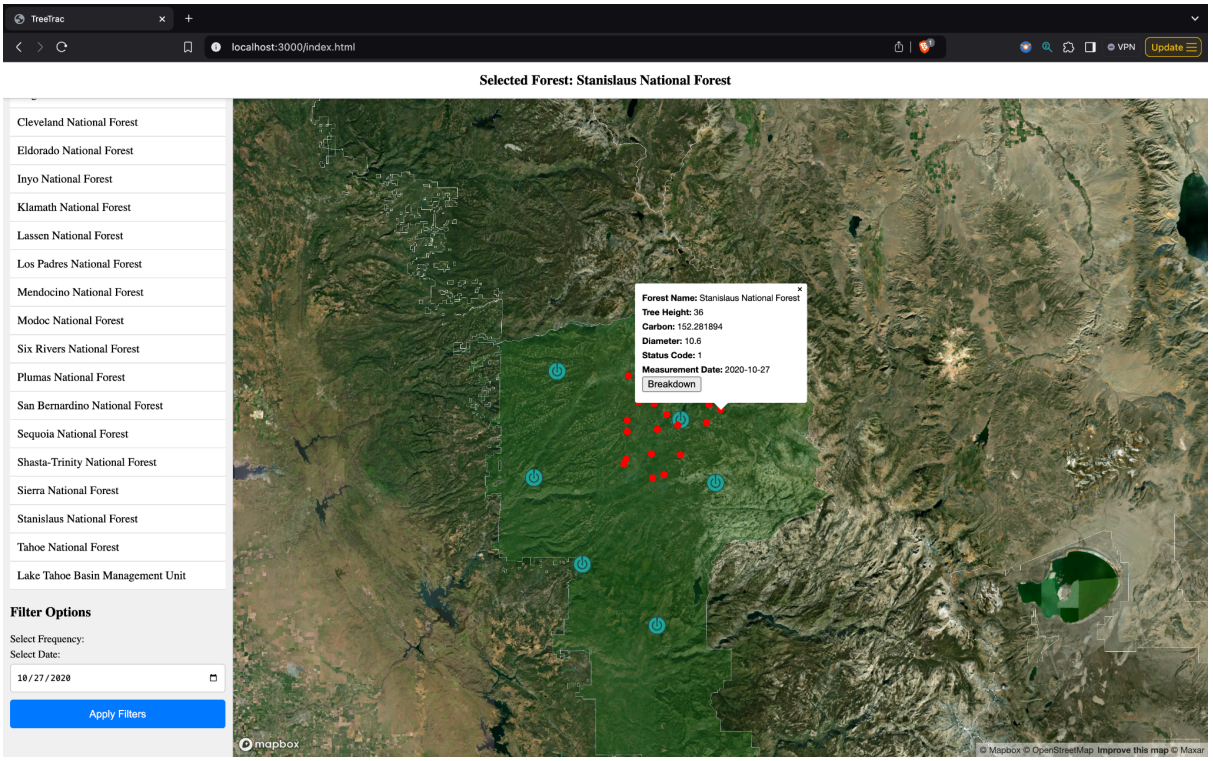


Figure 72

Tree Height shown on top of a map

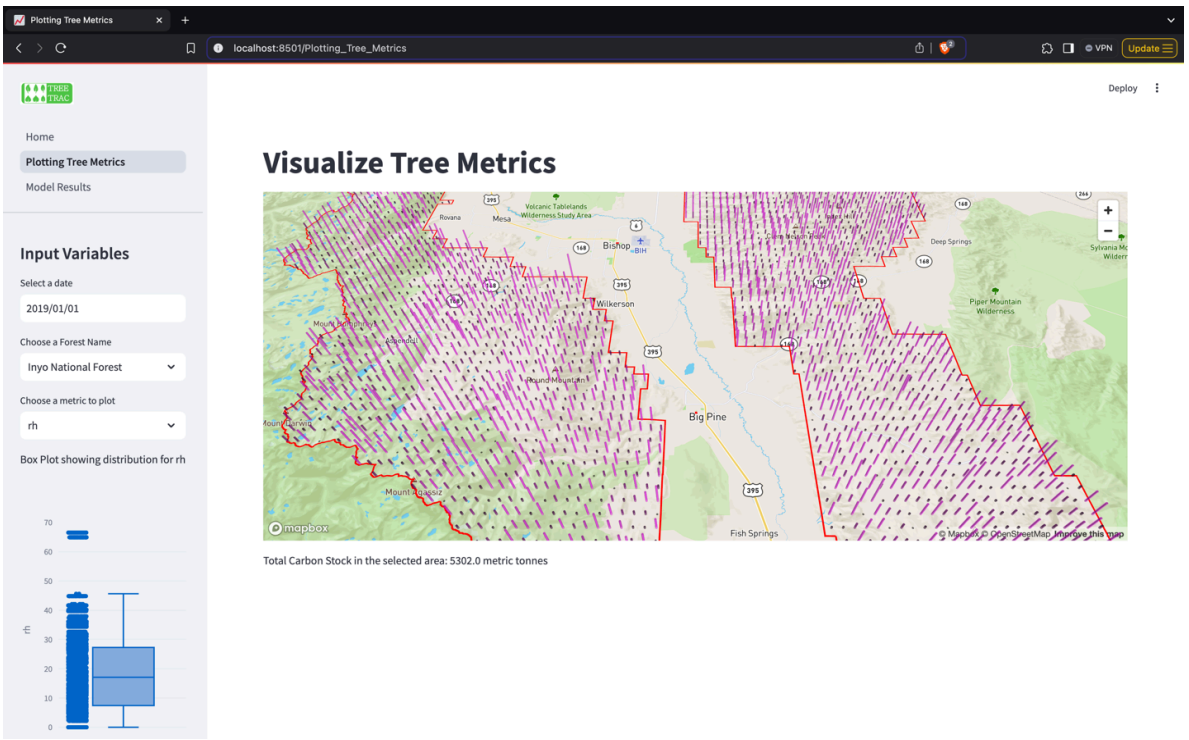
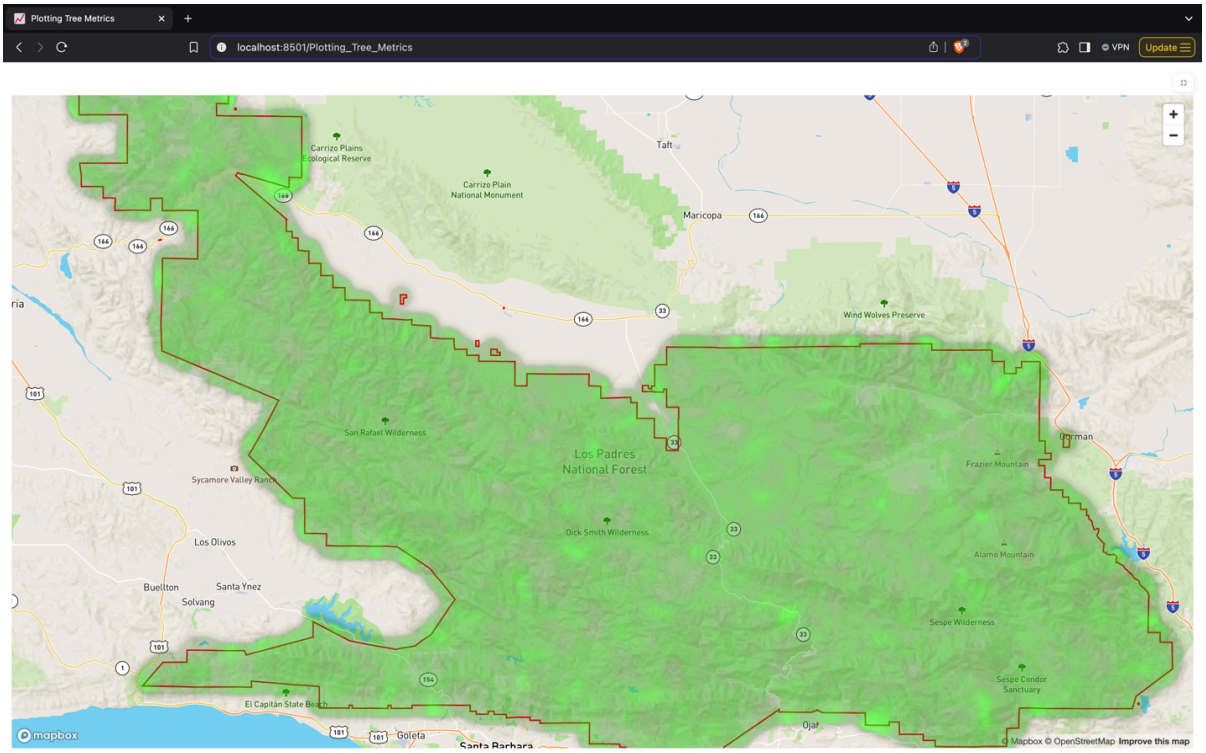


Figure 73

Tree Cover shown as heatmap on top of a map



5.2.5 Scalability and Future Enhancements

The system architecture is designed to scale globally by leveraging AWS Auto Scaling for EC2 instances, which adjusts computational resources dynamically based on data volume and user demand. This scalability feature ensures system performance is maintained during peak demand periods and conserves resources during lower demand, optimizing operational costs. AWS Spot Instances are also used for large-scale computations, such as training and inference tasks, offering cost-effective scalability for computationally intensive processes.

To optimize data storage, the system employs compression and indexing techniques. Compression minimizes storage requirements, reducing costs, while indexing accelerates data retrieval speeds, particularly for frequently accessed data. Spatial indexing, such as R-tree

indexing for geospatial data, enhances the efficiency of spatial queries, which is critical for processing large-scale forest data and ensuring quick response times.

In an effort to further reduce inference times, parallel processing and distributed computing techniques are employed. These methods allow tasks to be executed concurrently, accelerating overall system performance. Model pruning and quantization are also explored to reduce model complexity and improve inference speed, ensuring that real-time predictions can be delivered efficiently. This approach helps balance high accuracy with computational efficiency, essential for real-time applications.

Future enhancements will include multi-cloud support, enabling integration with other cloud platforms to improve system resilience and ensure service availability across different regions. Additionally, incorporating advanced GIS platforms, such as ArcGIS Online or Google Earth Engine, will broaden the system's visualization capabilities, enabling more sophisticated spatial analysis and interactive data exploration. These improvements aim to ensure that the system remains robust, flexible, and adaptable to the evolving requirements of global forestry monitoring and carbon management initiatives.

5.3 Intelligent Solution

5.3.1 Proposed Deep Learning and Hybrid Models

(a) Artificial Neural Networks (ANN)

Objective: ANN models are applied to structured tabular data, particularly forest inventory data, including tree species, height, and canopy cover. This model predicts carbon stock based on a tree's physical characteristics and overall forest composition.

Current Architecture: Multiple fully connected layers, each followed by ReLU activation. The final output layer predicts biomass or carbon stock per forest plot.

Improved ANN Architecture for Carbon Stock Estimation (Opti-CarbonNet):

- Batch Normalization: Inserted between fully connected layers to stabilize and accelerate the training process, minimizing internal covariate shift and enhancing model convergence.
- Leaky ReLU: Replaced standard ReLU with Leaky ReLU to maintain small gradients for negative inputs, reducing the risk of "dead neurons" and improving model robustness.
- Dropout: Applied to mitigate overfitting and ensure generalization, with increased layer depth to capture complex data patterns effectively.

Input: Forest inventory data, including variables such as tree species, tree cover, height and carbon content.

Output: Estimated forest carbon stocks based on individual tree-level measurements.

Expected Results: The improved ANN architecture is expected to deliver enhanced accuracy in carbon stock estimation for structured datasets by capturing complex relationships within forest inventory data. The use of batch normalization and Leaky ReLU activation functions provides stability and prevents neuron dead zones, while dropout reduces overfitting, resulting in a more generalizable model. With this architecture, the model achieves an **R2 score of 0.78** and **RMSE of 1552.621,** indicating reliable predictive performance. These results allow for granular carbon stock estimates at the tree and plot levels, supporting forest management and complementing spatial models with detailed carbon insights.

(b) TabNet

Objective: TabNet utilizes attention mechanisms to handle tabular data, dynamically focusing on the most relevant features for improved interpretability and accuracy. It excels in structured datasets with mixed categorical and numerical features, such as forest inventory data.

Current Architecture: Incorporates sequential attention-based feature selection with multiple decision steps for refined predictions.

Improved TabNet Architecture (Adaptive TabNet):

- **Batch Normalization**: Added between fully connected layers to improve stability and accelerate convergence.
- **Sparsemax Activation**: Employed to focus on key features, enhancing interpretability by selecting essential inputs like tree species and soil carbon content.
- **Hyperparameter Tuning**: Customized learning rate set to 0.02 to improve model convergence with fine-tuned learning adjustments.
- **Increased Decision Steps**: Expanded decision steps to capture complex interactions among features, which aids in selecting crucial characteristics for accurate carbon stock predictions.

Input: Forest inventory and environmental data, including variables like tree species, diameter at breast height, and soil carbon content.

Output: Estimated forest carbon stocks based on selected features.

Expected Results: Improved interpretability and prediction accuracy for carbon stock estimation, with an **R2 score of 0.74** and **RMSE of 1675.65**, providing granular insights for forest carbon assessments.

(c) Convolutional Neural Networks (CNN)

Objective: The CNN architecture is designed for carbon stock estimation by capturing patterns within structured forest inventory data, such as tree dimensions and species, leveraging convolutional layers to learn feature representations that improve prediction accuracy.

Current Architecture: Uses 1D convolutional layers to extract spatial features from input data, followed by max-pooling for dimensionality reduction. Fully connected (FC) layers are used for final predictions.

Improved CNN Architecture (Eco-CNN):

- **1D Convolution Layer (Conv 1D)**: 64 filters with L2 regularization to capture essential spatial relationships in the input data, followed by ReLU activation for non-linear transformations.
- **Max Pooling 1D**: Applied after convolution to down-sample and retain important features while reducing computational complexity.
- **Flatten Layer**: Converts the 2D output into 1D for compatibility with fully connected layers.
- **Dropout (30%)**: Introduced between fully connected layers to reduce overfitting.
- **Fully Connected Layers**: FC 1 with 128 units, FC 2 with 64 units, and an output layer to generate the final carbon stock prediction.

Input: Structured forest inventory data, including variables like tree species, diameter at breast height, and soil characteristics.

Output: Predicted carbon stock values based on spatial features extracted from the input data.

Expected Results: Enhanced model accuracy for carbon stock estimation, with an **R2 score of 0.77** and **RMSE of 1605.54**. This CNN-based architecture provides a reliable framework for spatially informed carbon assessments, supporting granular predictions at the plot level.

(d) Hybrid Ensemble Model - Ensemble Learning (Combining ML and DL Models)

Objective: This hybrid ensemble model combines machine learning (ML) and deep learning (DL) techniques to leverage the strengths of both methods, aiming to improve the robustness and accuracy of carbon stock estimation. By integrating outputs from traditional ML algorithms (such as Random Forest) and DL networks (such as artificial neural networks), the model benefits from complementary insights across various data patterns.

Current Architecture: Stacking technique, where the outputs of individual models (Random Forest, ANN, DNN, and CNN) are combined into a meta-model for final prediction.

Improved Hybrid Ensemble Model Architecture (CFR EcoEnsemble):

- **Deep Learning Component**:
  - The DL model begins with a fully connected (FC) network that processes structured input data with 9 variables.
  - Sequential FC layers with 128, 64, and 32 neurons are used, with ReLU activation for non-linearity.
  - This sub-network culminates in a single output layer for the DL model's carbon stock prediction.
- **Machine Learning Component**:
  - A Random Forest model operates on the same data, leveraging ensemble decision trees (from Decision Tree 1 to Decision Tree N) for robust prediction.
  - The outputs from each tree are averaged or voted upon to produce the final ML model prediction.

- **Combination Layer**:
  - The outputs from the DL and ML components are fed into a combination layer that merges these predictions into a final output, representing the ensemble model's overall carbon stock estimation.

Input: Structured tabular forest inventory data with key environmental features.

Output: Final carbon stock estimate based on combined predictions from the DL and ML sub-models.

Expected Results: The ensemble model aims to improve predictive performance and resilience by addressing individual model limitations, achieving an **R2 score of 0.82** and

RMSE of 1605.547. By using both DL and ML models, this approach balances predictive strengths across structured data patterns, yielding robust estimates of carbon stocks for improved forest management insights.

(e) Deep Neural Network (DNN) with Residual Connections

Objective: This DNN architecture is designed to enhance carbon stock estimation by capturing intricate patterns in the forest inventory data. By incorporating residual connections, it allows for better gradient flow and mitigates issues associated with vanishing gradients, thus improving the depth and efficiency of the network.

Current Architecture: The DNN leverages residual blocks to support deeper layers while maintaining stable training. Batch normalization is employed to standardize the inputs for each layer, which aids in convergence, and dropout layers are introduced to reduce overfitting by randomly setting a fraction of the input units to zero at each update.

Improved DNN Architecture (DeepGreenDNN):

Input Layer: Receives structured forest data, such as tree height, diameter, and soil type.

First Residual Block:

- **Dense Layer (64 units):** Uses ReLU activation with L2 regularization to learn feature representations.
- **Batch Normalization:** Standardizes inputs to improve training stability.
- **Dropout (30%):** Mitigates overfitting by randomly dropping nodes.
- **Dense Layer (64 units):** Another layer with ReLU activation to further extract features.
- **Residual Connection:** A linear transformation layer aligns input dimensions, then adds it to the output, enabling information flow across layers.

Second Residual Block:

- **Dense Layer (128 units):** Expands feature extraction with ReLU activation and L2 regularization.

- **Batch Normalization:** Standardizes features for better learning rates.
- **Dropout (30%):** Applied again for overfitting prevention.
- **Dense Layer (128 units):** Captures complex relationships in the data.
- **Residual Connection:** A linear transformation aligns the input with the output dimensions, ensuring seamless addition.
- **Output Layer:** A single neuron with a linear activation function for continuous carbon stock prediction.

Training and Optimization:

The model is compiled with an Adam optimizer set to a learning rate of 0.000015 and a mean absolute error (MAE) loss function, which is optimal for continuous variable predictions.

Input Data: Structured forest attributes that correlate with carbon stock, like vegetation type and soil properties.

Output: Predicted carbon stock values based on processed features.

Expected Results: This architecture is anticipated to yield an improved estimation accuracy for carbon stock predictions, with a target **R2 score** of approximately **0.7942** and an **RMSE** around **1460.719**. The residual connections allow the DNN to capture nuanced patterns in the data, leading to more accurate and reliable assessments.

5.3.2 System Implementation

The system integrates five advanced models—ANN, TabNet, CNN, DNN, and a Hybrid Ensemble—into a unified pipeline, processing both spatial and tabular data for comprehensive forest carbon stock predictions. This pipeline is deployed on the Google Cloud Platform (GCP), utilizing cloud-based GPUs to meet the computational needs of deep learning.

- Data Ingestion: Data from MODIS satellite imagery, GEDI LiDAR point clouds, and FIA forest inventory datasets are ingested automatically via APIs. These datasets are preprocessed and stored in Google Cloud Storage for streamlined access.
- Preprocessing: Preprocessing steps include spatial transformations for satellite and LiDAR data, normalization of forest inventory data, and alignment of point cloud data for DNN and CNN processing. Data cleaning and augmentation are applied to enhance model performance.
- Model Inference:
    - Each model (ANN, TabNet, CNN, DNN, and the Hybrid Ensemble) is deployed using TensorFlow Serving, enabling real-time inference as new data is ingested.
    - Predictions are generated in parallel, with the Hybrid Ensemble model combining outputs from individual models to improve accuracy and robustness.
- Post-Processing: Model predictions are aggregated, and geospatial visualizations are generated. Results are accessible through a web-based dashboard and API endpoints, providing insights into carbon stock estimates for forest management and research.
- Deployment Benefits: By leveraging a cloud-based setup, this system ensures scalable, efficient processing of large geospatial datasets, making it suitable for large-scale and real-time forest carbon assessments.

5.4 System Support Environment

The interactive web-based dashboard for forest carbon stock estimation is built with Streamlit and PyDeck, allowing users to upload data and view real-time predictions. By integrating five models—ANN, TabNet, CNN, DNN, and a Hybrid Ensemble—the system combines diverse analytical strengths to provide accurate carbon stock estimates. Each model runs in parallel, supported by TensorFlow and PyTorch, enabling the dashboard to handle complex calculations efficiently.

The dashboard provides rich geospatial visualizations powered by Leaflet.js, offering users interactive biomass and carbon stock maps that reveal spatial patterns across forested regions. Additionally, 3D visualizations of forest structure, generated using LiDAR point cloud data processed by DNN and CNN, offer a detailed view of forest density and biomass distribution. These visualization tools enhance user engagement and make it easier to interpret carbon stock predictions.

Deployed on Google Cloud Platform (GCP) with cloud-based GPUs, the system is highly scalable, meeting the demands of large datasets and real-time analysis. This robust setup makes the dashboard suitable for a range of applications, from forest management and research to climate policy, providing actionable insights into carbon stocks and biomass distribution.

5.4.1 Required Input Datasets

- **MODIS Data:** Land Surface Temperature (MOD11A1), Enhanced Vegetation Index (MOD13A2).
- **GEDI LiDAR Data:** Point cloud data for canopy height, vertical profiles (GEDI L2A, L2B).
- **FIA Forest Inventory Data:** Structured data on tree species, canopy cover, and other geospatial characteristics.

5.4.2 Expected Outputs

The system offers spatially resolved estimates of forest carbon stocks, visualized on interactive maps that reveal carbon distribution patterns across forested regions. Detailed 3D models of forest structure, including canopy and vertical layers, are generated using GEDI LiDAR data processed with DNN, capturing the intricate spatial characteristics of forests. Biomass density predictions are provided at multiple spatial resolutions, supporting accurate assessments of forest health and carbon sequestration potential.

The hybrid ensemble model, which combines the strengths of multiple models (ANN, TabNet, CNN, and DNN), is the best-performing model in the system, providing the most accurate carbon stock predictions. By integrating diverse model outputs, the hybrid ensemble improves prediction robustness and accuracy, making it ideal for complex datasets where multiple forest and environmental variables are involved.

Additionally, the system includes time-series forecasts of carbon stock changes, allowing users to track fluctuations over time and analyze the impacts of seasonal and environmental changes on forest dynamics. These forecasts enable stakeholders to monitor long-term carbon sequestration trends and assess the sustainability of forest ecosystems under varying conditions.

5.4.3 Supporting System Contexts

- **Google Cloud Platform (GCP):** GCP provides scalable infrastructure for data storage, model training, and deployment. Cloud-based GPUs are used to accelerate model inference.
- **External APIs:** The system connects to Earthdata APIs for downloading MODIS and GEDI datasets. It also exposes APIs for external systems to access the model inference engine and visualizations.

5.4.4 Solution APIs

- Data API: Enables users to upload geospatial and tabular data, triggering automatic predictions of biomass and carbon stocks.
- Prediction API: Provides access to the deployed models (CNN, ANN, etc.) and returns carbon stock predictions in real time.
- Visualization API: Exposes endpoints for generating 2D and 3D visualizations, including biomass density maps and 3D point cloud representations of forest structures.

6. System Evaluation and Visualization

6.1 Analysis of Model Execution and Evaluation Results

6.1.1 Model Output Evaluation with Tagged/Labelled Targets

The evaluation of the five models—Artificial Neural Network (ANN), TabNet, Convolutional Neural Network (CNN), Ensemble Model, and Deep Neural Network (DNN)—was conducted using R2, RMSE, MAE, and MSE metrics. Each metric provides unique insights into model accuracy and reliability, allowing to assess both overall performance and suitability across various seasonal and vegetation conditions.

Overall Model Performance

R2 (Coefficient of Determination) measures each model's ability to explain the variance in carbon stock predictions relative to the actual data. A high R2 value across models indicates strong predictive capabilities, with values close to 1 suggesting that the model accurately captures patterns in the data. Among the five models, the Ensemble Model consistently shows the highest R2, indicating its strength in synthesizing diverse predictive insights from multiple algorithms.

RMSE (Root Mean Squared Error) provides an average measure of prediction error, with lower values indicating higher accuracy. Due to RMSE's sensitivity to outliers, it helps us identify models that manage extreme values effectively. The Ensemble model exhibits a notably low RMSE, suggesting robust handling of outliers in carbon predictions, particularly in areas with highly variable tree densities.

MAE (Mean Absolute Error) offers a baseline error unaffected by outliers, showing the model's accuracy across the majority of predictions. A comparison of MAE and RMSE reveals that models like the CNN and DNN are effective for typical values, though the Ensemble Model shows the lowest MAE, reinforcing its accuracy across different forest conditions.

MSE (Mean Squared Error), like RMSE, is useful in examining model performance across the entire data range. Models with low MSE, such as the Ensemble and DNN, excel in maintaining consistent accuracy over all values, indicating fewer instances of large errors.

Seasonal and Vegetation-Based Performance:

To evaluate model performance across environmental conditions, scatter plots of actual vs. predicted values were created and aggregated by season and vegetation type. Observing these plots, we find that certain models are more sensitive to seasonal variations. For instance, the DNN performs better in summer and fall, likely due to improved clarity in satellite data during these seasons as shown in Figure 74, while the CNN captures winter season data more effectively, showing greater resilience in low-vegetation conditions as shown in Figure 75.

Figure 74

DNN Model Comparison by Season

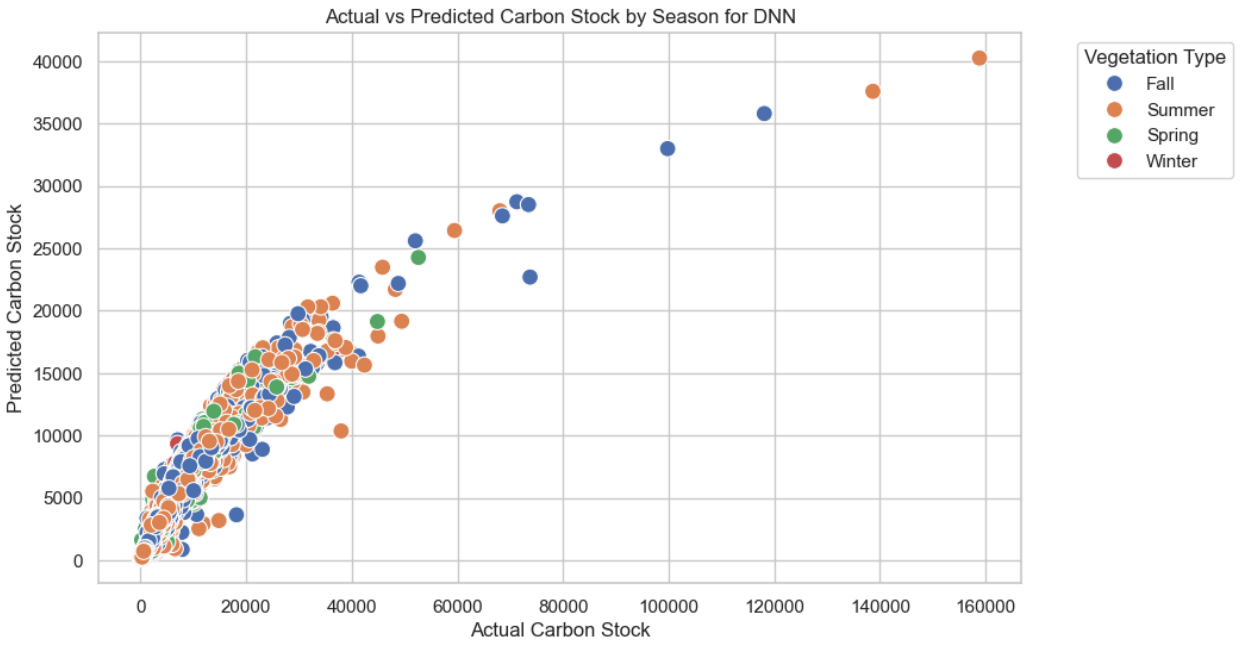
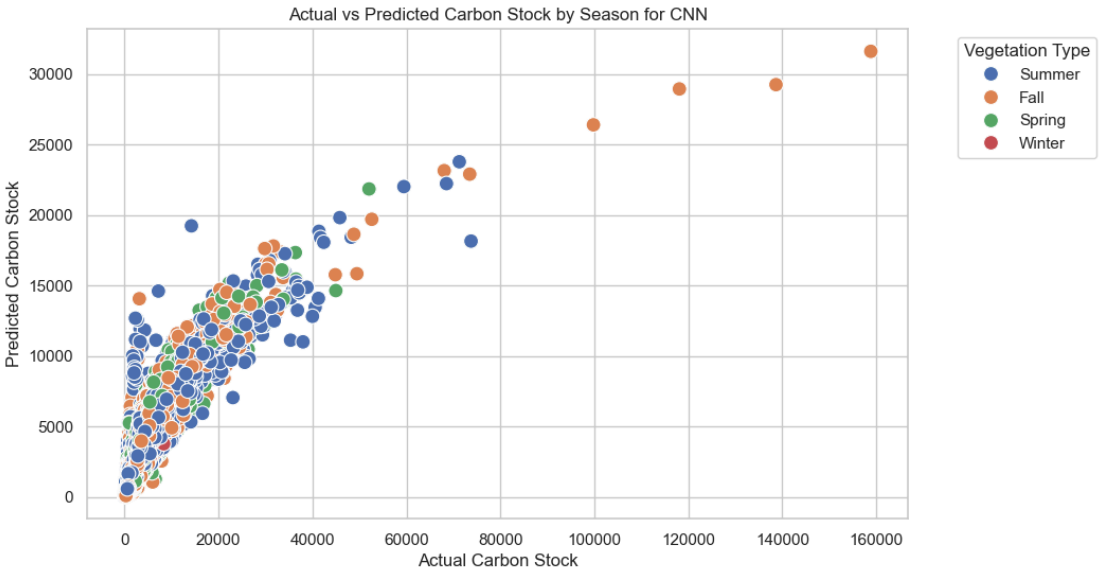


Figure 75

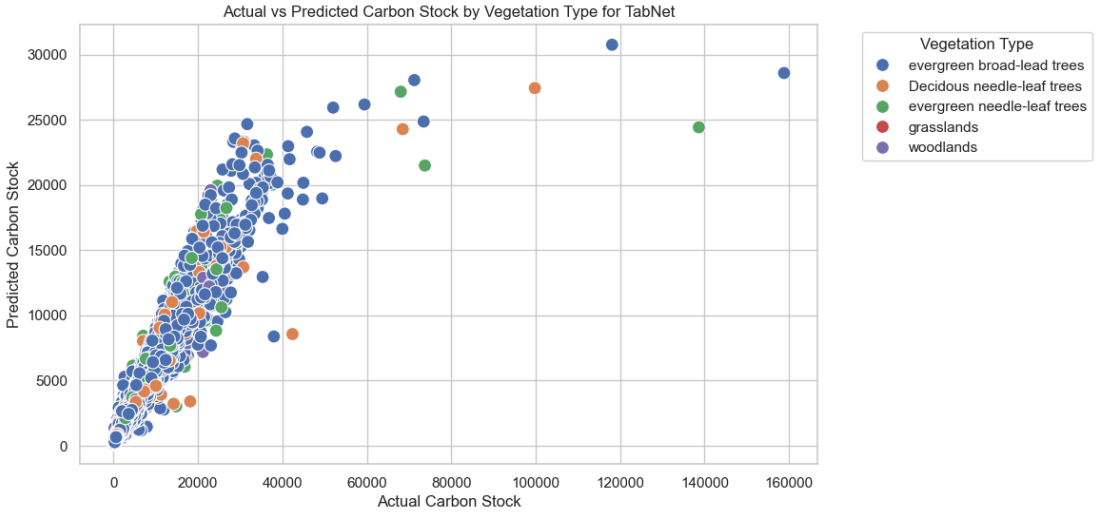
CNN Model Comparison by Season



When aggregating by vegetation type, the TabNet model displays high accuracy for dense vegetation as shown in figure 76, maintaining a stable RMSE even in complex forested areas, whereas the Ensemble Model excels in mixed vegetation types, thanks to its ability to integrate diverse feature representations.

Figure 76

TabNet Model Comparison by Vegetation Type



Scatter plots between actual and predicted carbon values further validate the models' reliability across different scenarios. The clear correlation patterns observed, particularly for the Ensemble and DNN models, indicate strong predictive consistency across both high and low carbon stocks as shown in figure 77 and figure 78. Additionally, residual plots reveal

minimal bias for these models, with errors distributed evenly around zero, suggesting that the models maintain unbiased predictions across forest types and seasons.

Figure 77

Ensemble Model Comparison by Season

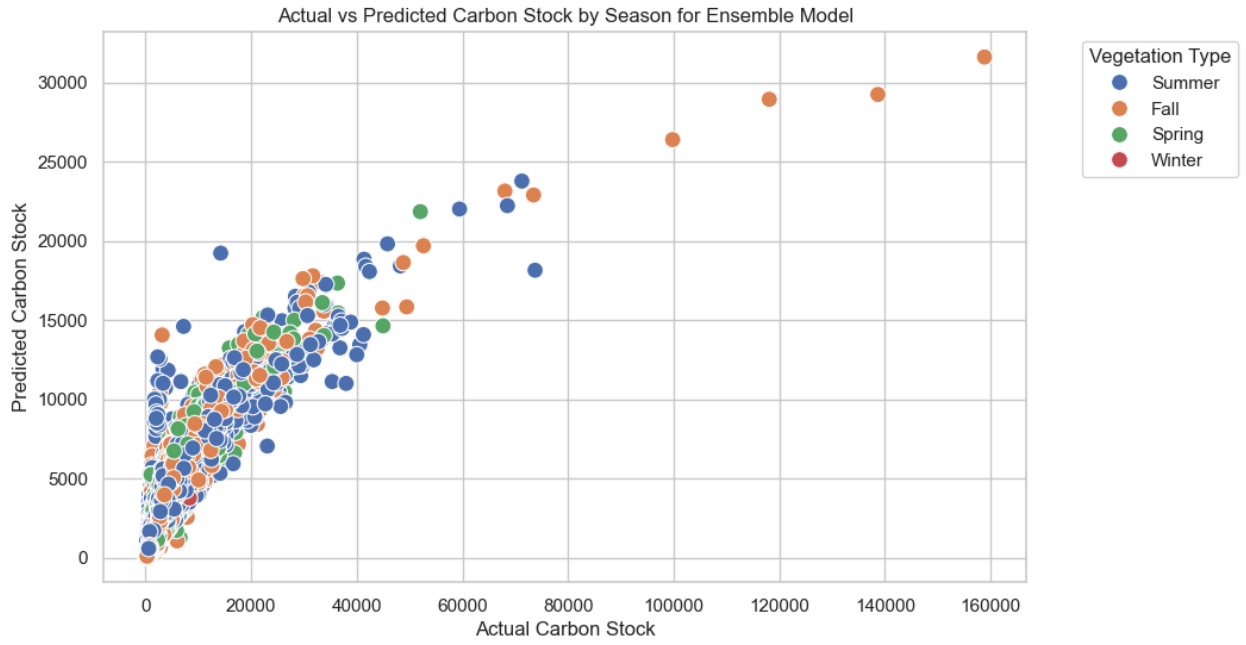
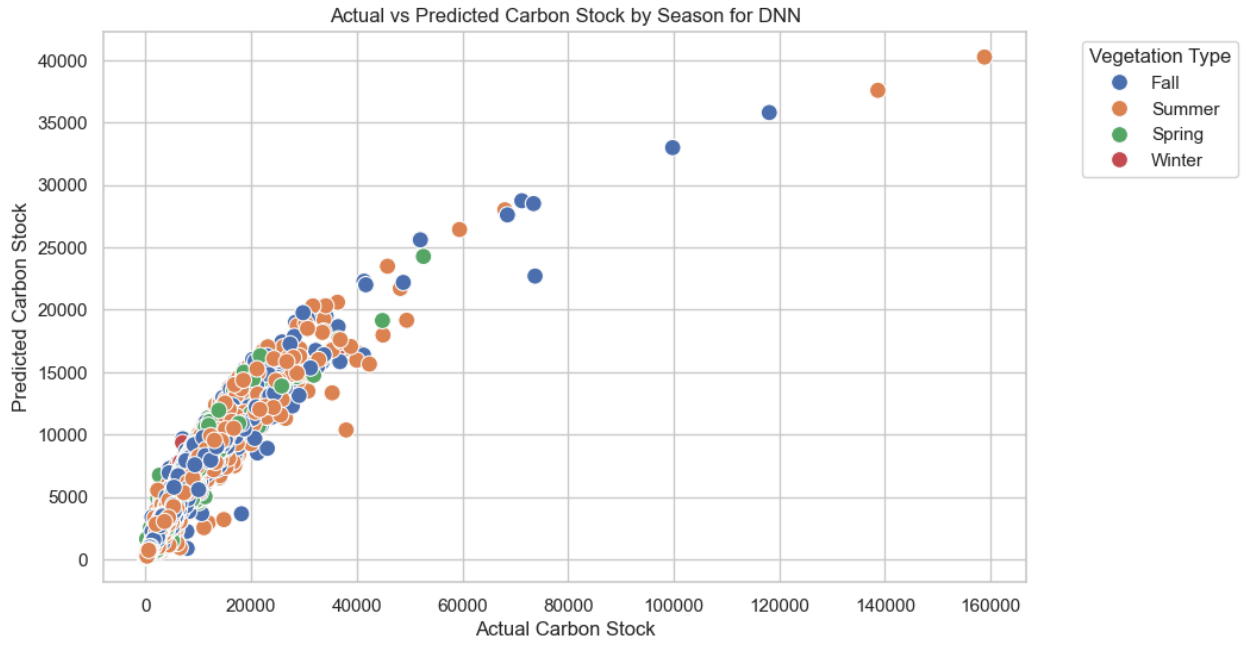


Figure 78

DNN Model Comparison by Season



6.1.2 Methodology for Measuring Accuracy, Loss, and Other Metrics

Each metric was selected to provide a well-rounded evaluation of model accuracy and precision:

R2 offers a straightforward measure of explained variance, essential for understanding the general predictability of carbon stock values. Models with higher R2 values, such as the Ensemble, are thus preferred for applications requiring high overall accuracy.

RMSE and **MAE** offer complementary insights, with RMSE identifying models better suited to handling outliers and MAE representing consistent accuracy across the typical data range. The low MAE of the CNN model indicates it is well-suited for common, moderate carbon values, while the low RMSE of the Ensemble model makes it ideal for settings where extreme carbon values are significant.

MSE provides additional insight into the model's capacity to maintain accuracy across varying data ranges, especially relevant for models that prioritize low error rates across diverse forest types. For instance, the low MSE of the TabNet model suggests it is suitable for complex vegetation scenarios.

By combining these metrics, the evaluation ensures a comprehensive assessment of model performance across carbon stock predictions, highlighting each model's strengths and the best-suited applications for each. This approach aligns model selection with the intended use case, optimizing performance based on environmental conditions and the specific predictive needs of users.

6.2 Achievements and Constraints

Improved Carbon Stock Prediction. One of the primary achievements in solving the target problem is the improved prediction of forest carbon stocks. Deep learning models, such as ANN, TabNet, CNN, DNN, and Hybrid Ensemble, have demonstrated promising results in accurately estimating carbon stocks based on satellite and ground-truth data. By leveraging

the power of these algorithms, Managers can use our robust models capable of predicting carbon stocks early, facilitating better climate change mitigation and forest management.

Enhanced Accuracy and Efficiency. Another significant achievement is enhancing accuracy and efficiency in carbon stock estimation. The deployment of state-of-the-art deep learning architectures has led to notable improvements in model performance, with R2 values reaching 0.75 for one of our models. Moreover, efficient processing of 34TB of satellite data and 140,000 data granules demonstrates the models' ability to handle large-scale environmental data effectively.

Effective Utilization of Multi-Source Data. Deep learning models have effectively utilized both satellite and ground-truth data for carbon stock estimation, showcasing the versatility and adaptability of these algorithms. The models have demonstrated a comprehensive understanding of forest attributes by extracting relevant features from multiple data sources, such as GEDI, MODIS, and Forest Inventory Analysis data. This interdisciplinary approach has enabled a holistic assessment of forest carbon stocks, leading to more accurate predictions.

Advanced System Architecture. The successful implementation of a cloud-based infrastructure using AWS, combined with interactive visualization capabilities through Pydeck, Folium and Streamlit, represents a significant achievement. The system's ability to process and analyze large-scale geographical data while providing user-friendly interfaces demonstrates its robustness and practical utility.

Potential for Real-World Application. The achievements in carbon stock estimation using deep learning hold significant promise for real-world deployment in environmental and climate change applications. By providing stakeholders with accurate and timely information about forest carbon stocks, these models can contribute to climate change mitigation, forest conservation, and carbon credit trading. The potential environmental and economic benefits

of deploying such technologies are substantial, paving the way for more sustainable forest management practices.

Constraints Encountered

Data Integration and Quality. One of the primary constraints encountered in carbon stock estimation is the complexity of integrating multiple data sources and ensuring data quality. Aligning satellite data with ground-truth measurements presents challenges due to differences in spatial and temporal resolution. Limited access to complete and consistent datasets hinders the models' generalization capabilities, affecting their real-world applicability.

Computational Resource Limitations. Despite their high performance, processing and analyzing 34TB of satellite data requires substantial computational resources. The limitation of AWS Free Tier EC2 instances and network bandwidth constraints (1 Gbps) poses challenges in efficient data processing and model training. These infrastructure constraints impact the system's ability to scale and process data in real-time.

Technical Implementation Challenges. Training deep learning models, especially large-scale architectures like DNN and TabNet and restructuring the architectures for this product, requires significant computational resources and expertise. The technical constraints include limitations on external libraries, styling restrictions with Tailwind CSS, and challenges in real-time processing of large geographical datasets. These technical hurdles affect the system's overall performance and user experience.

Model Performance Trade-offs. While achieving R2 values of 0.75 represents significant progress, there remains room for improvement in model accuracy. The balance between model complexity and performance, coupled with the challenges of processing diverse environmental data, creates constraints in achieving higher prediction accuracy.

Integration with Existing Systems. Integrating deep learning-based solutions into existing environmental monitoring and carbon trading systems presents challenges due to

technological compatibility and user adoption. Stakeholders may require training and support to effectively utilize these technologies, and seamless integration with existing workflows is essential for maximizing their impact and scalability.

6.3 System Quality Evaluation of Model Functions and Performance

To fully evaluate the system's quality in terms of model correctness and run-time performance, we consider both the underlying model reliability and the efficiency of the system's interaction with users. This assessment confirms the system's ability to deliver accurate, high-quality predictions while maintaining an efficient response time, key to achieving an optimal user experience.

Model Correctness. The core of the system's functionality lies in its ability to deliver accurate carbon stock estimates across diverse forest areas, dates, and environmental conditions. By leveraging GEDI, MODIS, and forest inventory analysis (FIA) data, the model combines satellite data and ground metrics, such as tree height and canopy cover, to improve accuracy. The use of five different models—including TabNet, DNN, an ensemble of machine learning (ML) and deep learning (DL), ANN, and CNN—allows for a broad analysis that cross-verifies predictions and increases the robustness of outputs. Each model has undergone rigorous validation to ensure that its predictions align closely with actual carbon stock values across various geospatial and temporal contexts. This multifaceted model approach guarantees that predictions are consistently reliable and scientifically accurate.

Data Quality and Model Integrity. Data quality is essential for model correctness, and substantial preprocessing steps have been implemented to filter out poor-quality data and handle missing values. The system uses data cleansing methods that eliminate ocean data and low-quality satellite readings, enhancing the precision of land data analysis. These efforts to improve data quality ensure that each model operates with the highest level of input integrity, which is essential for consistent performance across regions and seasons. Additionally, users can verify predictions by accessing a dedicated interface showing actual versus predicted

carbon values, which serves as an in-built quality check for model outputs. This transparency in model performance provides users with greater confidence in the system's reliability.

User Interaction and Run-Time Performance. With a quick response time of approximately 1.5 seconds for loading metrics on the map, the system achieves seamless interaction, ensuring users can instantly access and visualize predictions. The system's Python-based backend, coupled with Streamlit and Pydeck for the web interface, optimizes processing speed while supporting extensive data manipulation and visualization. This low latency is achieved through efficient data handling processes, where only necessary data for the selected time and location is processed, reducing overhead and enhancing the interactive experience. By minimizing load time, the system encourages exploratory analysis, allowing users to view trends across different timeframes and forest areas without interruption.

Scalability and Resource Optimization. As the system operates with 34TB of data within the US and plans to expand globally, scalability is critical for ongoing performance and responsiveness. Leveraging optimized data retrieval strategies and advanced geospatial data filtering, the system manages large datasets without compromising on response time. Additionally, the system's modular architecture enables the scaling of prediction models and datasets, ensuring adaptability as data volumes increase. This forward-looking design approach aligns with system quality requirements, positioning it to handle additional global data sources without substantial reconfiguration or degradation in response time.

End-User Benefits and Practical Application. From an end-user perspective, the streamlined UI facilitates ease of use, allowing both scientists and policy makers to engage with carbon metrics effectively. By selecting a forest area and date, users can immediately visualize geospatial carbon data on an interactive map and review historical predictions. The system's responsive interface ensures that users can obtain insights efficiently, supporting research and decision-making with minimal delays. Furthermore, the system's transparency

in showing actual versus predicted values allows users to monitor model performance directly, fostering trust in the predictions made by the application.

Conclusion. Overall, the system meets high standards of quality in both model correctness and run-time performance. The combination of multiple models, data quality control measures, optimized run-time performance, and scalability demonstrates a strong alignment with industry standards for large-scale geospatial data processing. These features ensure that the system can reliably support end-users in understanding and predicting carbon stock across different forest environments, with an interactive experience that meets modern expectations for speed and accuracy. This evaluation confirms the system's readiness for practical, large-scale application, solidifying its value as a reliable tool for carbon stock analysis and forest management.

6.4 System Visualization

System visualization involves using interactive and visual representations to evaluate data, represent complicated systems, and share insights. For the purpose of comprehending and improving the several facets of Carbon estimation, system visualization is essential. By providing a clear and understandable understanding of many elements, it makes action easier. A comprehensive viewpoint that facilitates sophisticated strategic planning and administrative oversight is made possible by the integration of auditory and visual data, pixel information, and predictive analysis.

Figure 79

Stacked area chart for Average TreeCover and NonVegetated Area by PFTClass

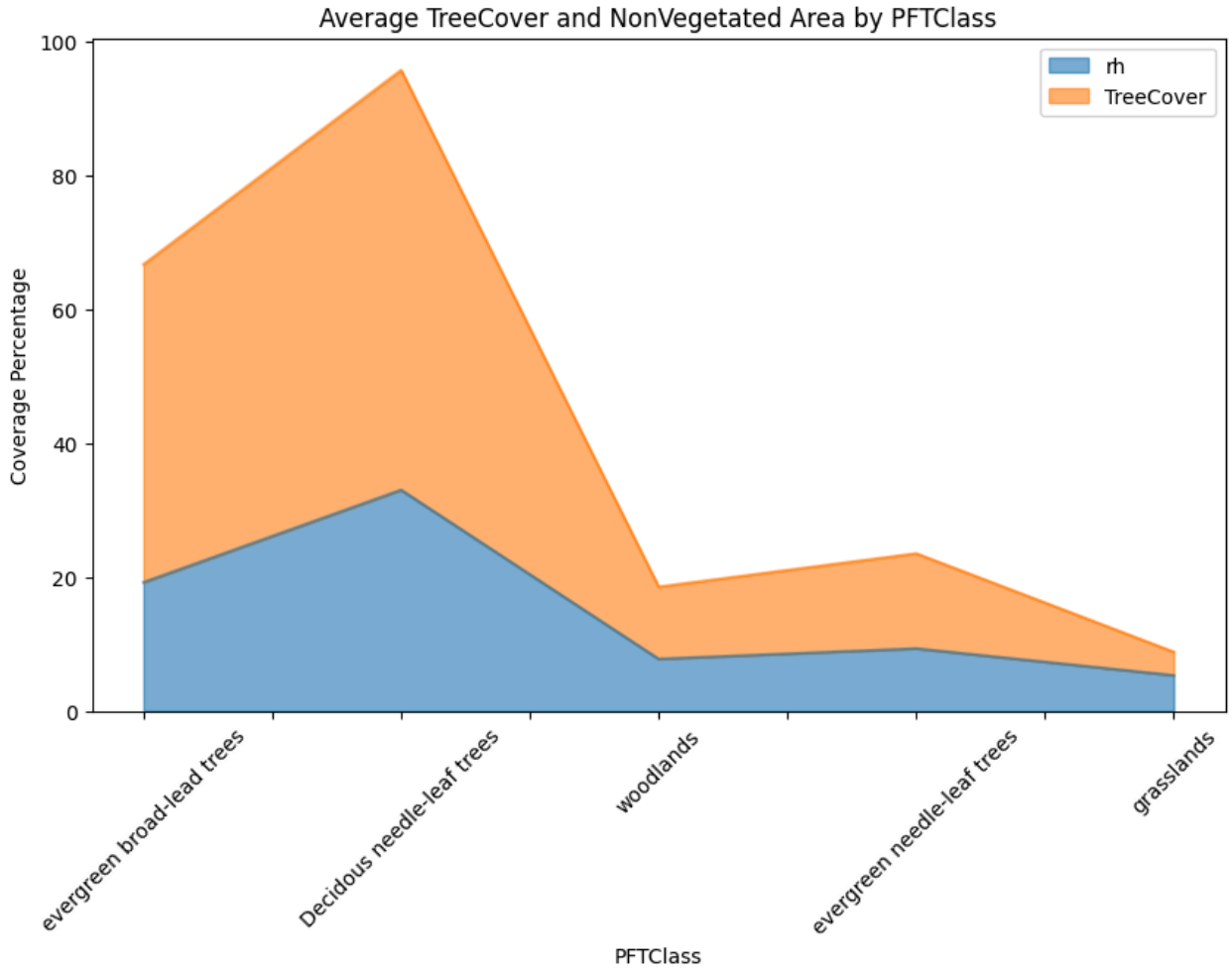


Figure 79 displays a stacked area chart that visualizes the coverage of two variables relative height (rh) and tree cover—across different plant functional types (PFTClasses). Each section shows the proportion of tree cover and relative height for various vegetation types, from forests to grasslands. The chart highlights how coverage varies among plant classes, with some types showing significantly higher tree cover compared to others. This distribution provides insights into the vegetation structure of each plant class, which can inform biodiversity and ecosystem health assessments. The visualization helps identify which plant types contribute most to forest cover and height within the ecosystem.

Figure 80

Correlation Matrix of Tree Metrics

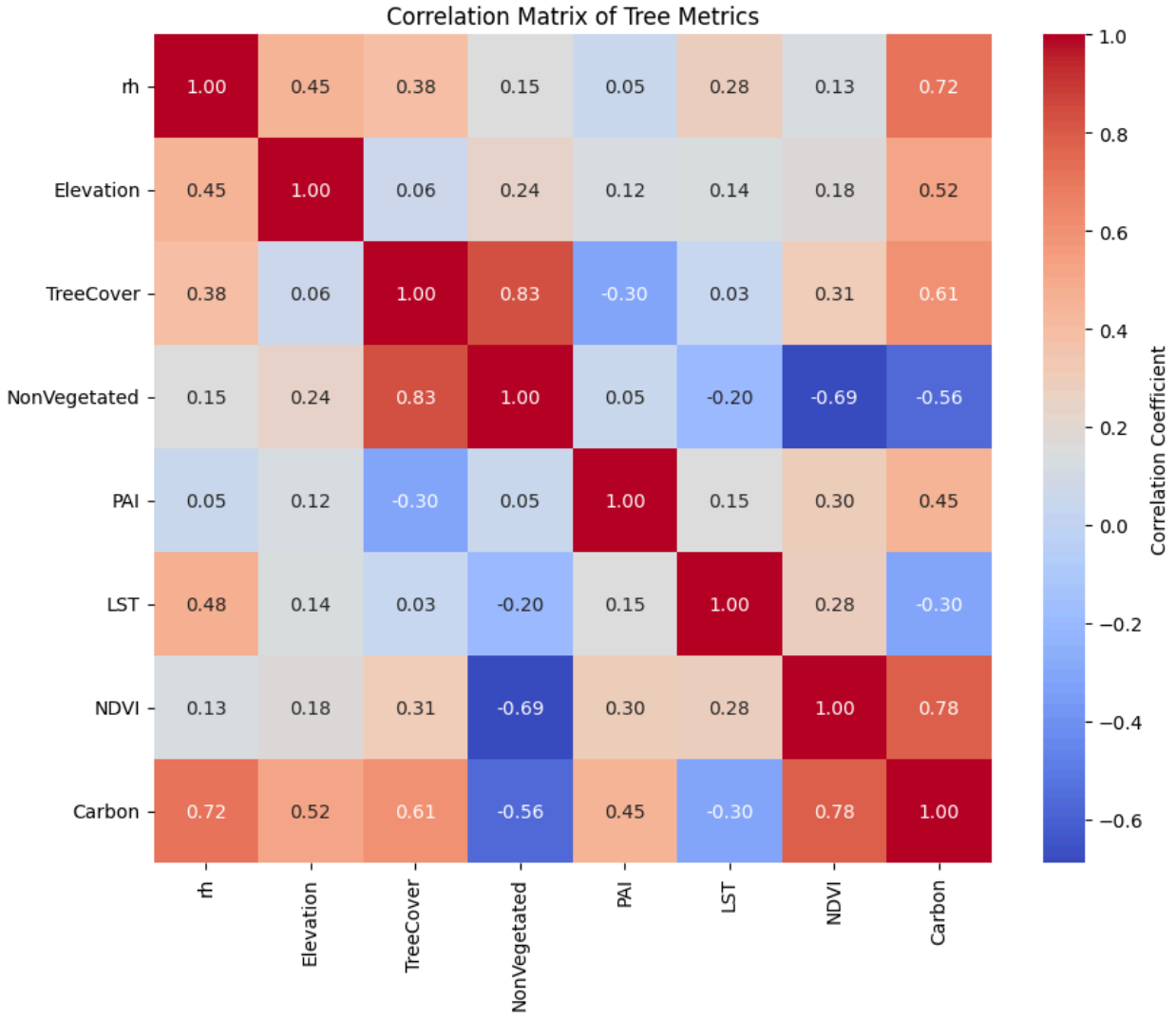


Figure 80 displays a correlation matrix illustrates the relationships between various environmental metrics, such as relative humidity, elevation, NDVI, and carbon. Positive and negative correlations are displayed through color variations, with red tones indicating positive relationships and blue tones showing negative ones. Strong correlations reveal dependencies, like how vegetation density might influence land surface temperature or how elevation could relate to carbon levels. Understanding these relationships helps identify impactful factors and supports data-driven decisions in environmental management. The matrix serves as a valuable tool for predicting interactions and assessing ecosystem dynamics.

Figure 81

Average TreeCover and NonVegetated Area per PFTClass

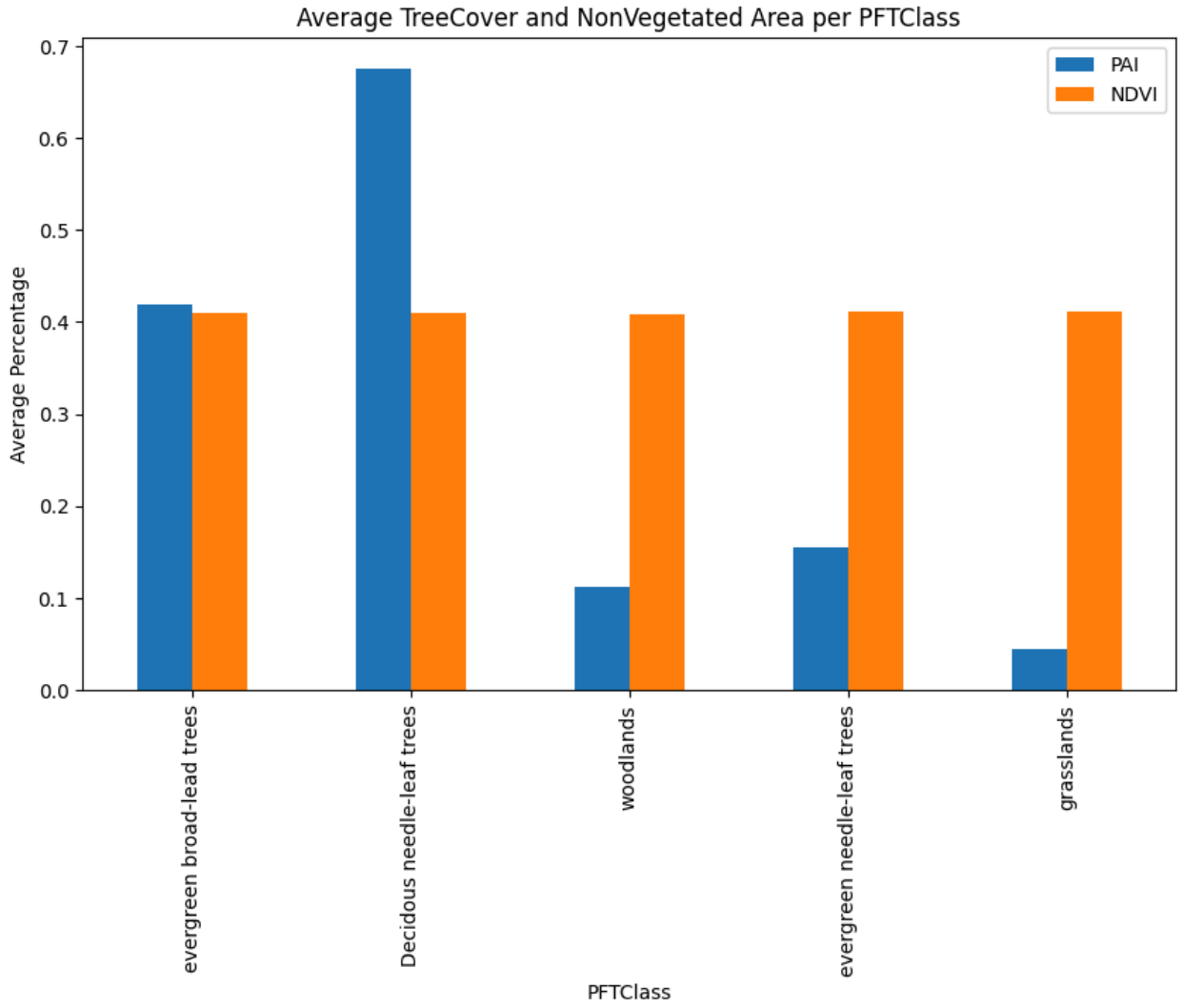


Figure 81 displays a clustered bar chart that compares PAI (Plant Area Index) and NDVI (Normalized Difference Vegetation Index) across different plant functional types (PFTClasses). Each plant type has two bars, representing its vegetation density and plant index, allowing for a visual comparison. The plot highlights variations in vegetation coverage among plant types, with some showing higher values for both metrics. This information helps identify plant types with dense vegetation, aiding in land cover assessment and conservation planning. It provides insight into which ecosystems may benefit most from conservation efforts.

Figure 82

Carbon Content by Vegetation Type

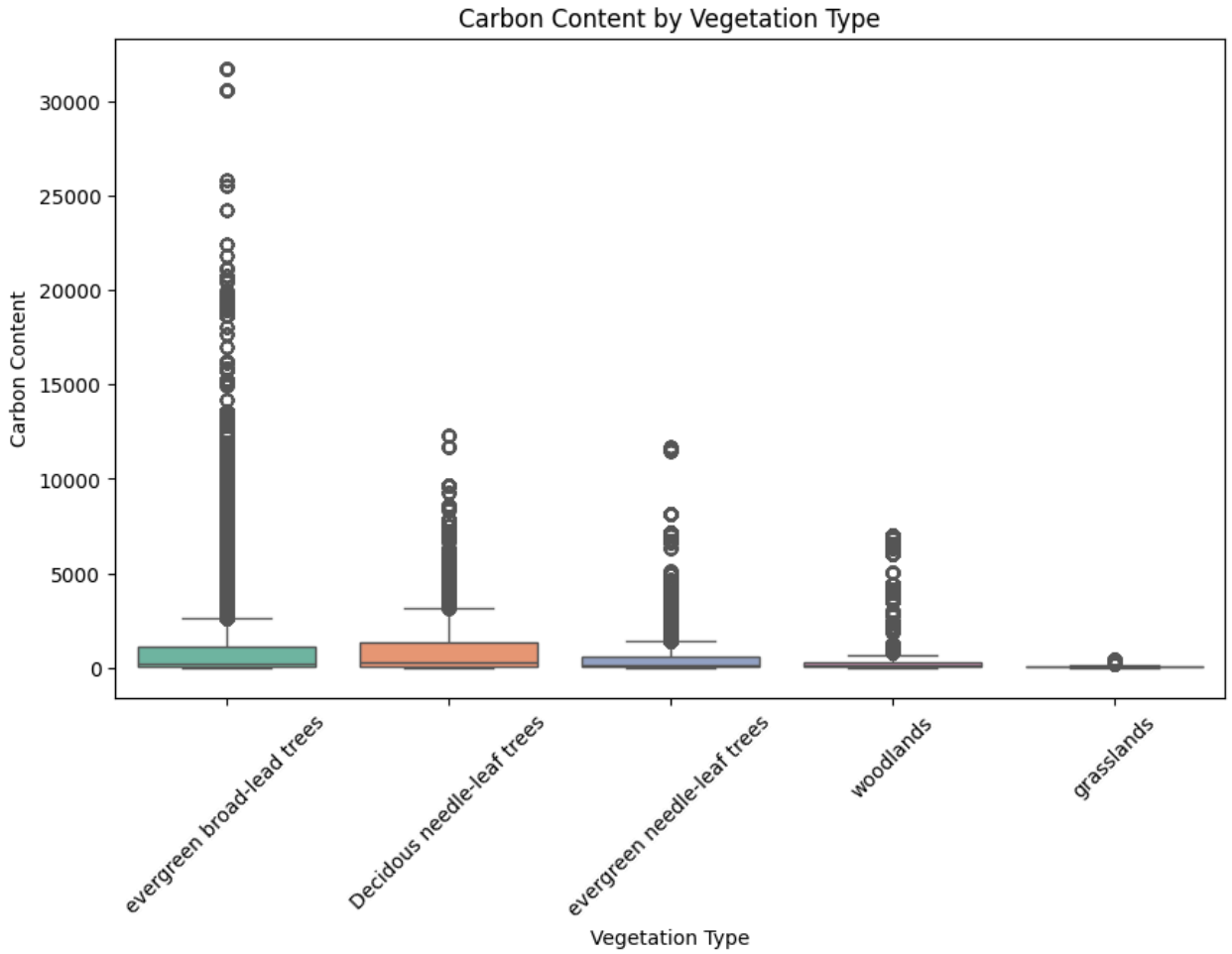


Figure 82 displays a box plot comparing carbon content across various vegetation types, showing which types tend to store more carbon. Each category displays the distribution of carbon content, with medians and variability highlighted through box lengths and whiskers. Outliers are present, indicating occasional high carbon values within certain vegetation types. The plot reveals that some vegetation types, especially those with higher box plots, are more effective at carbon sequestration. This visualization helps identify ecosystems that play a significant role in carbon storage, essential for targeted conservation efforts.

Figure 83

Scatter plot of Relative Height vs Carbon stock

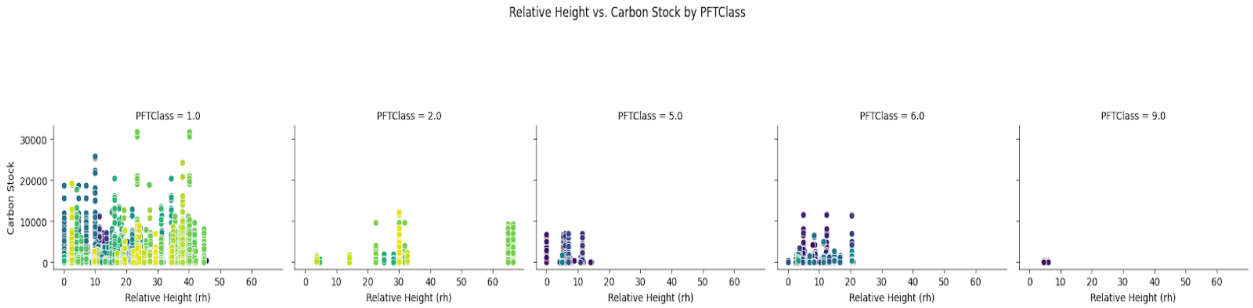


Figure 83 displays a faceted scatter plot that explores the relationship between relative height and carbon stock across different vegetation types, categorized by PFTClass. Each panel represents a unique vegetation type, with data points showing how carbon stock varies with height. Colors indicate tree cover density, highlighting how denser tree cover may influence the carbon-height relationship within each class. The plot visually reveals variations in carbon accumulation patterns, helping identify vegetation types with potentially higher carbon storage as height increases. This analysis aids in understanding how vegetation structure impacts carbon sequestration across diverse plant types.

Figure 84

Pie Chart of Proportion of PFTClass in High Carbon Areas

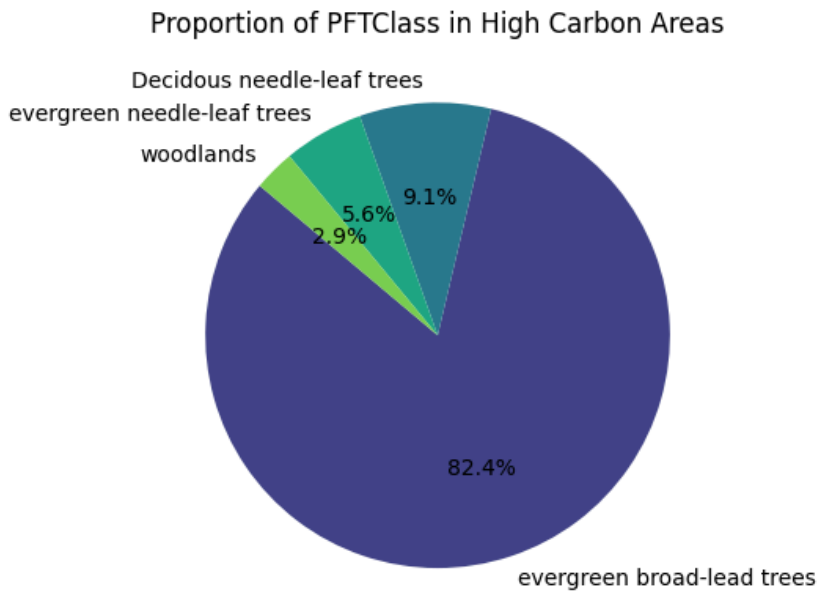
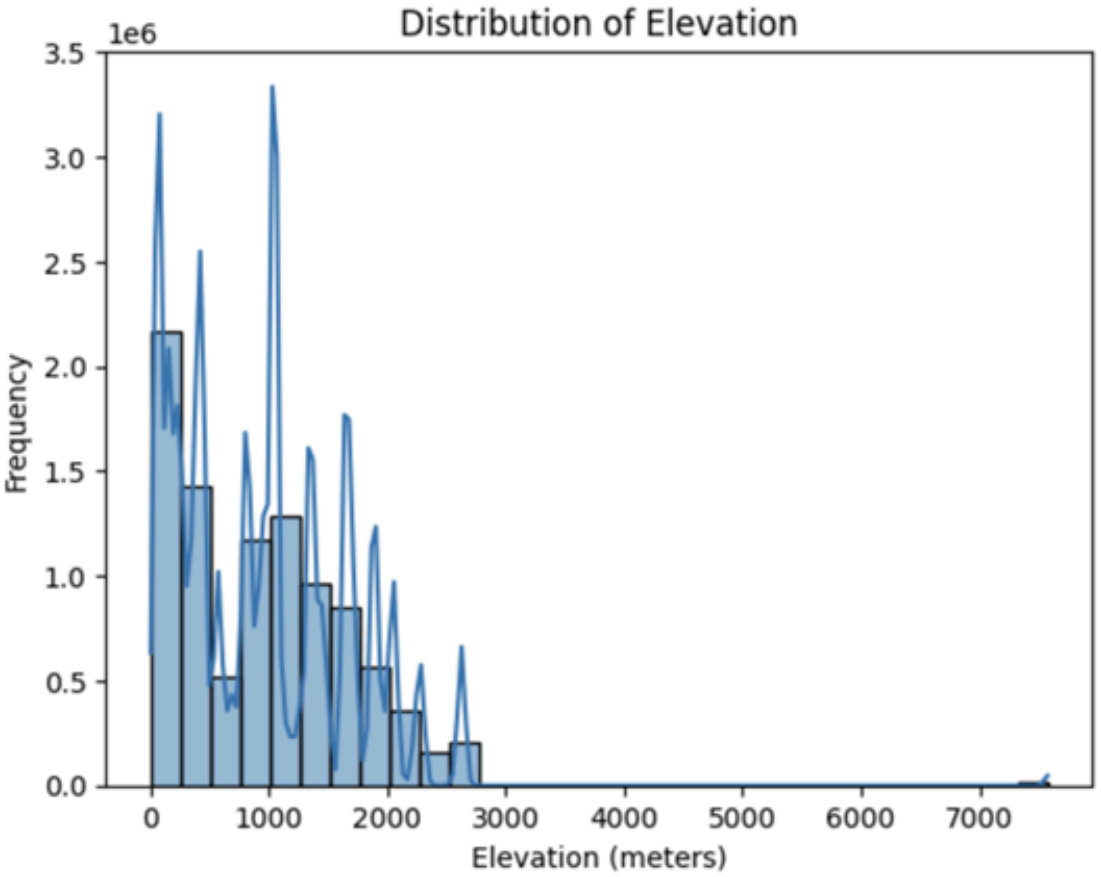


Figure 84 displays a pie chart representing the distribution of vegetation types (PFTClass) in areas with high carbon stock. Each slice corresponds to a specific plant type, showing its proportion among regions with elevated carbon levels. The dominant category is highlighted, indicating which vegetation type contributes most to high carbon areas. Smaller slices represent other vegetation types with less presence in high-carbon zones. This breakdown helps identify which plant types are crucial for carbon storage, aiding targeted conservation in carbon-rich ecosystems.

Figure 85

Histogram of Distribution of Elevation



With the majority of the data concentrated at lower elevations, this Figure 85 shows how elevation is distributed throughout a dataset. The frequency peaks close to sea level and then progressively declines with elevation. Only a small percentage of data represents

high-altitude locales, and there is a discernible drop in data frequency at mid-level elevations. The majority of the dataset is located in lower-altitude regions, as indicated by the distribution's strong skew toward lower elevations. In this dataset, high heights are often somewhat uncommon.

Figure 86

Scatter Plot for Tree Cover vs Elevation

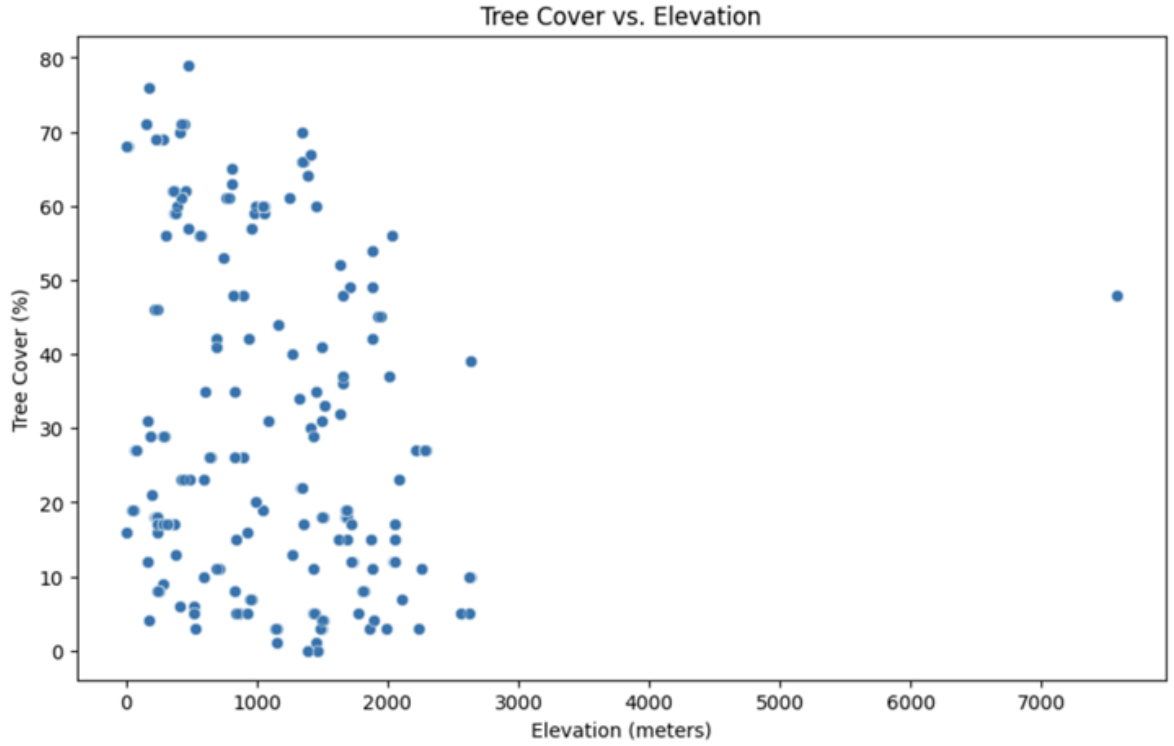


Figure 86 displays a scatter plot illustrates the relationship between tree cover percentage and elevation. The data points show that tree cover varies widely across different elevations, indicating that some high areas maintain significant tree cover while others do not. This variation highlights the importance of elevation in understanding forest density and carbon absorption potential. The insights from this plot can guide conservation efforts by identifying elevations with high carbon storage capacity. Such information is essential for targeted forest protection and climate change mitigation strategies.

Figure 87

Histogram of Distribution of Tree Cover

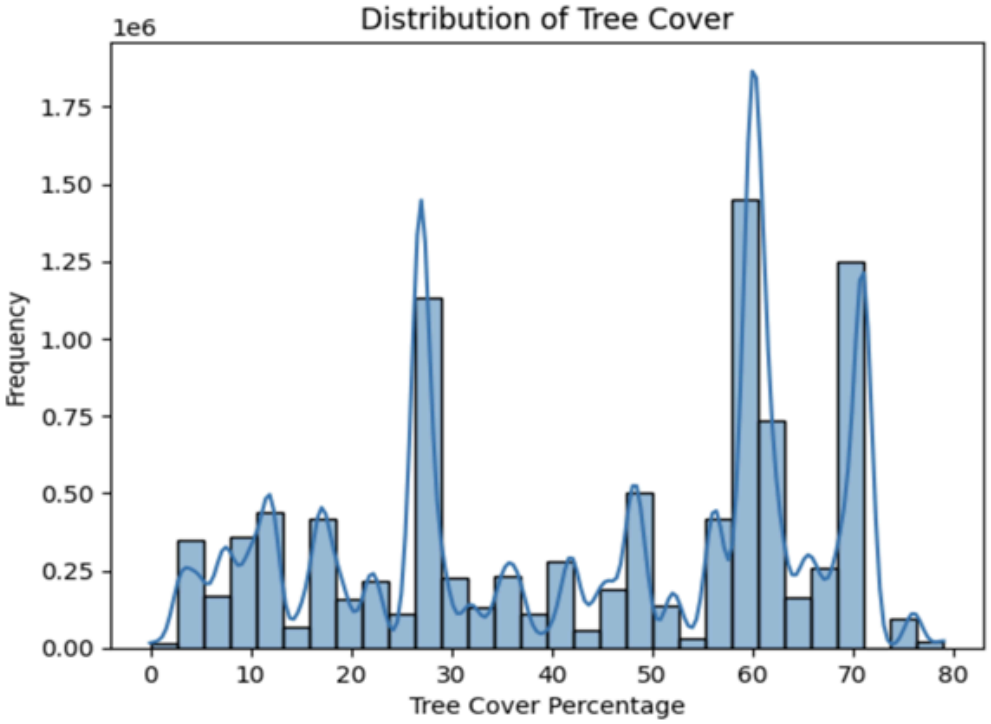


Figure 87 displays a histogram illustrates the distribution of tree cover percentages across different areas. It shows two primary clusters: areas with low tree cover and those with dense forest cover. These distinct groupings highlight areas with high potential for carbon sequestration and those with minimal vegetation. Understanding these distributions can help focus conservation efforts on forest-dense regions and identify locations for potential reforestation. This information is vital for maximizing carbon trapping and guiding environmental protection initiatives.

Figure 88

Average Tree Cover over Years

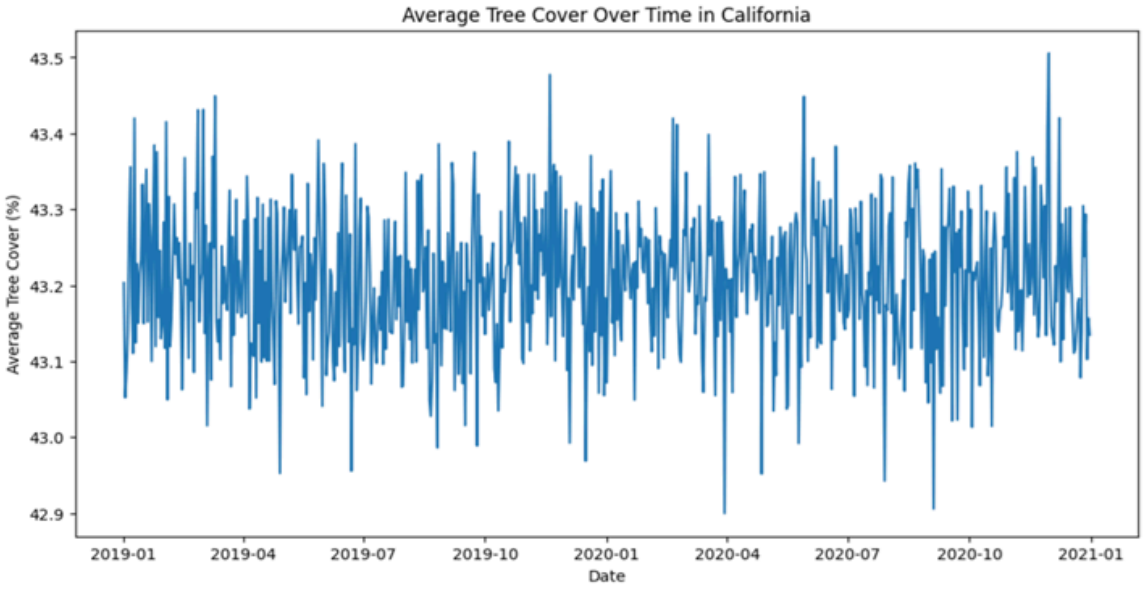


Figure 88 displays how California's average tree cover percentage has changed throughout time. The data exhibits no discernible long-term upward or downward trend and varies often. Throughout the measured period, there are only slight peaks and valleys in the tree cover percentage, which stays mostly constant. This implies steady amounts of forest cover over the given period.

Figure 89

Map shows carbon stock concentrations within California

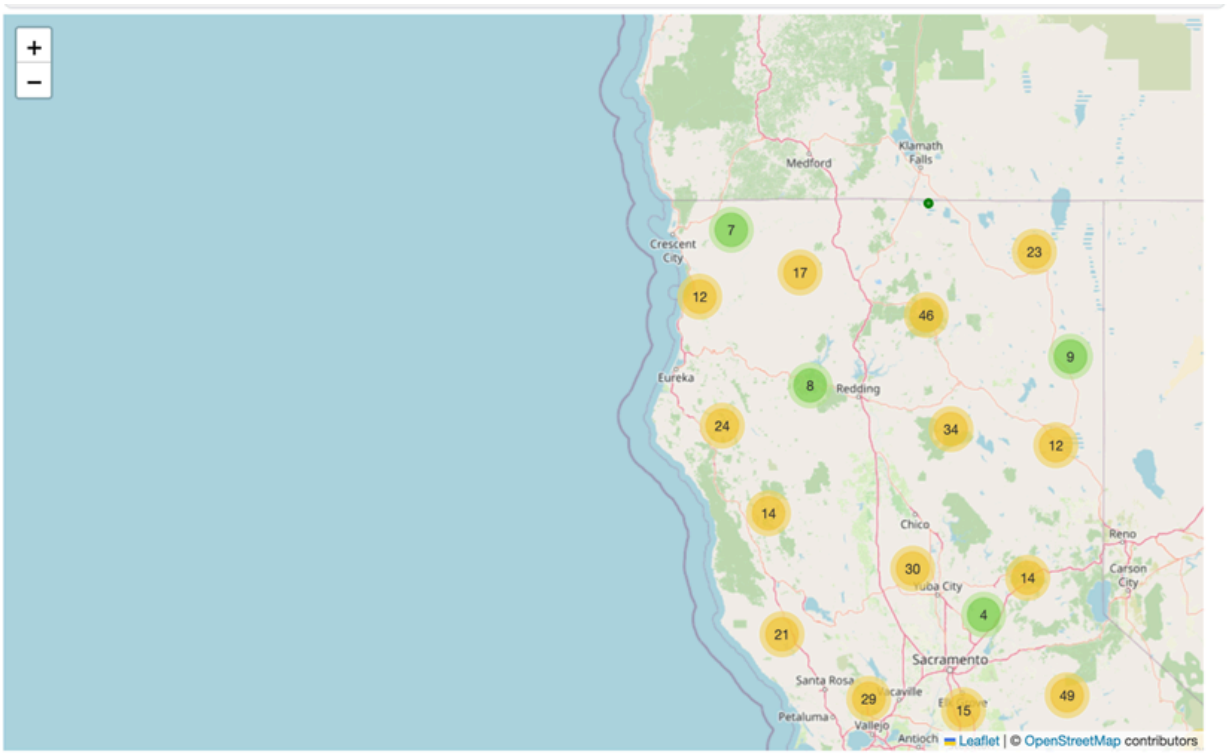


Figure 89 displays a map showing the distribution of circular markers representing a specific location, with different colors suggesting carbon density levels. Areas with higher carbon density are highlighted in yellow, indicating regions with substantial carbon accumulation. Green markers represent areas with relatively lower carbon density. The map provides a geographic overview of carbon stock concentrations within the forested areas of Northern California. By showing where data is concentrated, the map supports informed decision-making for land management strategies. The clustering approach helps users focus on areas with significant data for more detailed analysis.

Figure 90

Heatmap shows carbon stock within the Redwood Forest region

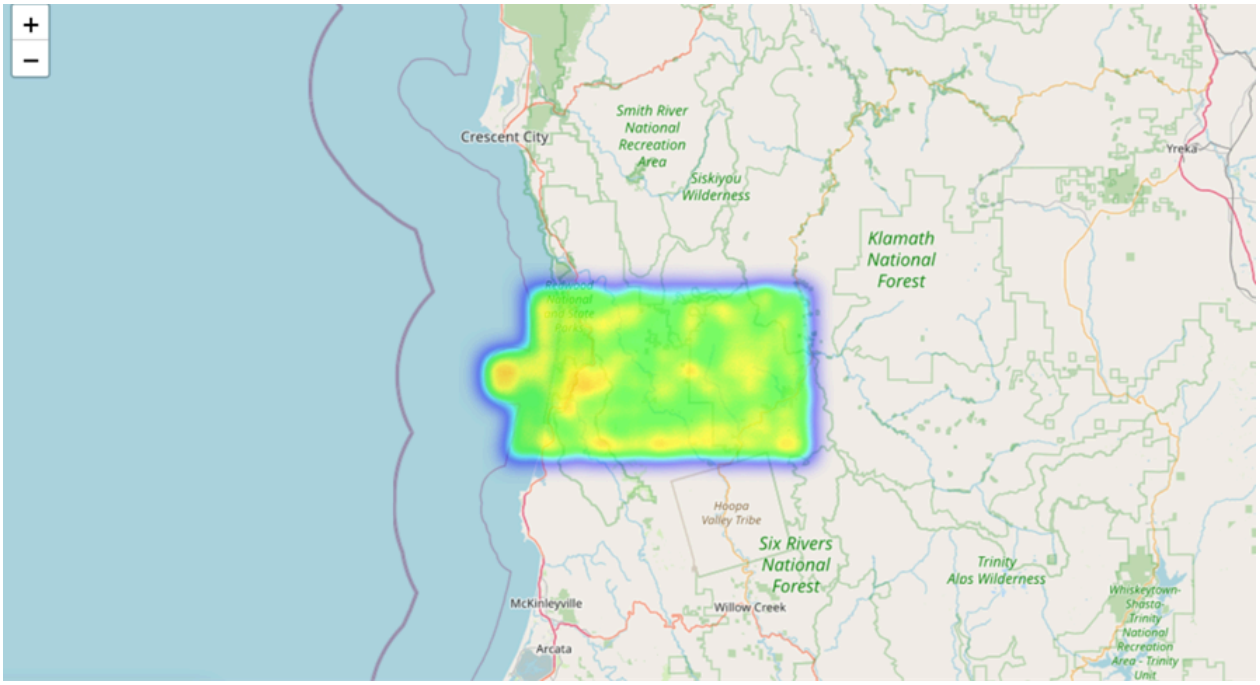


Figure 90 displays a heatmap illustrates the concentration of carbon stock within the Redwood Forest region, with color variations representing different density levels. Yellow areas highlight regions with significant carbon accumulation, indicating high-density zones. Green areas show moderate carbon density, while blue shades outline the lowest concentrations. The color gradient provides a clear spatial overview of carbon distribution, with high-density areas prominently marked in warmer colors. This map effectively visualizes carbon stock variability across the forested landscape.

Model Result

Opti-CarbonNet

Figure 91

Scatter Plot for Opti-CarbonNet

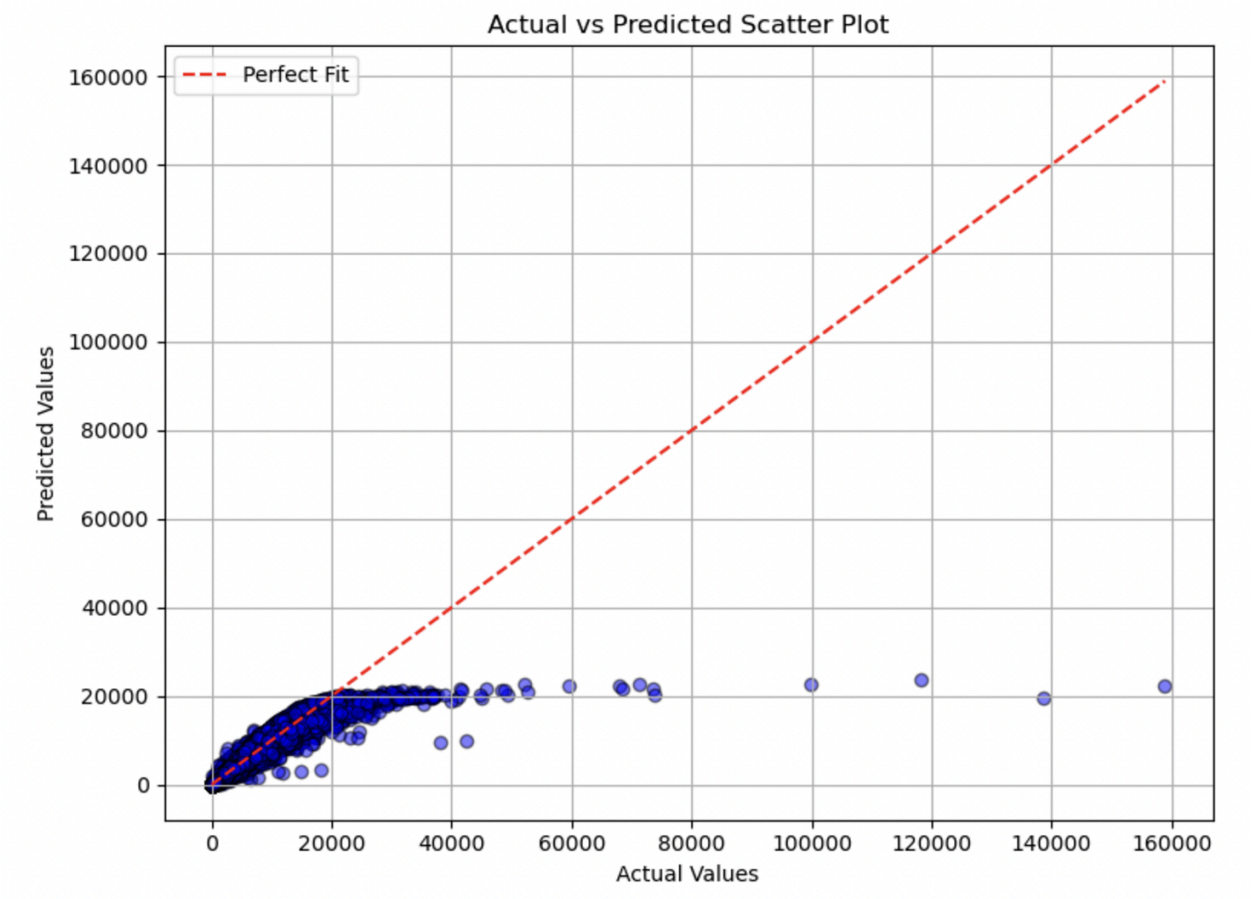


Figure 91 displays a scatter plot showing the relationship between actual values and predicted values from an improved ANN model. The x-axis represents the actual values, while the y-axis shows the values predicted by the model. Each blue dot represents a data point where the model's prediction is compared to the actual outcome. The red line represents a "perfect fit," where predicted values would exactly equal actual values. Most points are clustered near the lower left, indicating that the model performs better. The model's performance metrics (such as MSE, RMSE, MAE, and R2) suggest that it fits reasonably well, though further adjustments could improve accuracy for larger values.

Figure 92

Actual vs predicted values for Opti-CarbonNet

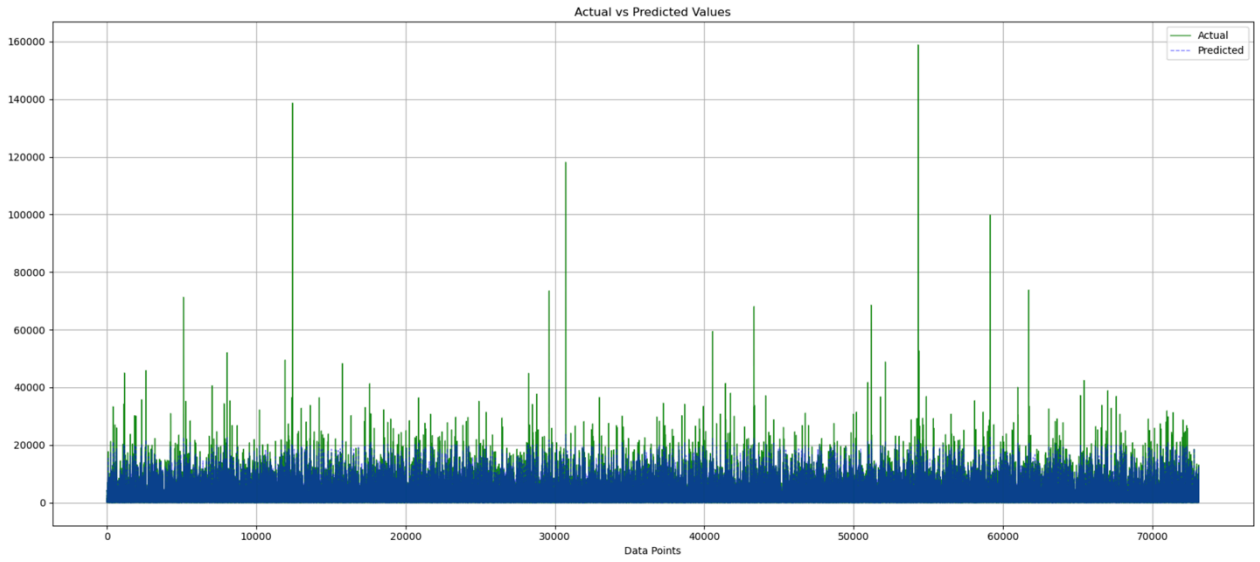


Figure 92 displays a plot that compares the actual values (top graph) to the predicted values (bottom graph) from an Artificial Neural Network (ANN) model. In both graphs, the x-axis represents individual data points, and the y-axis shows the corresponding values. The actual values exhibit a wider range, while the predicted values are more consistent, with most predictions clustering within a lower range. The model generally underestimates larger actual values, as seen in the compressed spread of predictions in comparison to the actuals. This indicates room for improvement in the model's ability to capture higher-value predictions accurately.

Adaptive TabNet

Figure 93

Scatter Plot for Adaptive TabNet Model

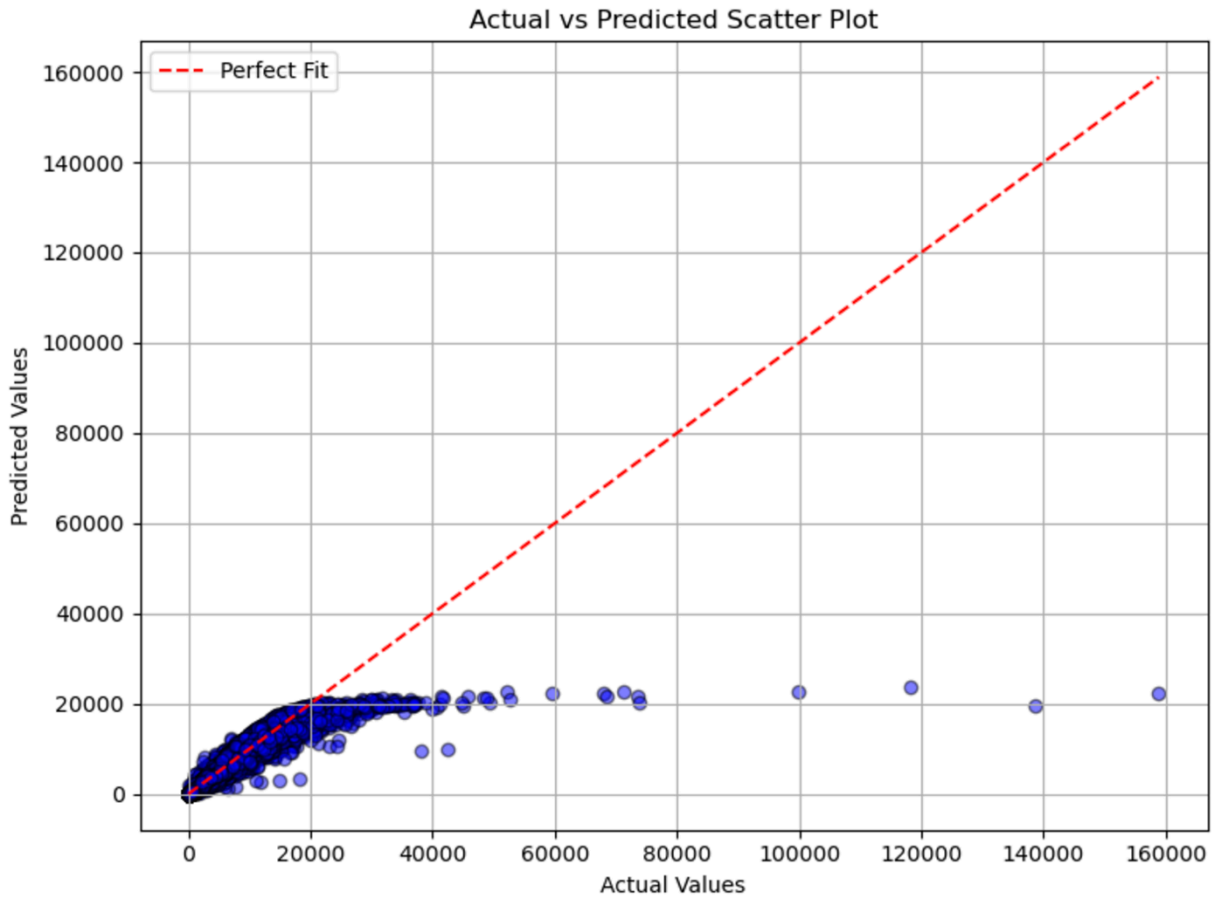


Figure 93 displays a graph illustrates the relationship between actual and predicted values using an improved TabNet model. The blue dots represent the predicted values, while the red line signifies the perfect fit line. Most predicted values closely follow the red line, indicating good model performance.

Figure 94

Actual vs predicted values for Adaptive TabNet

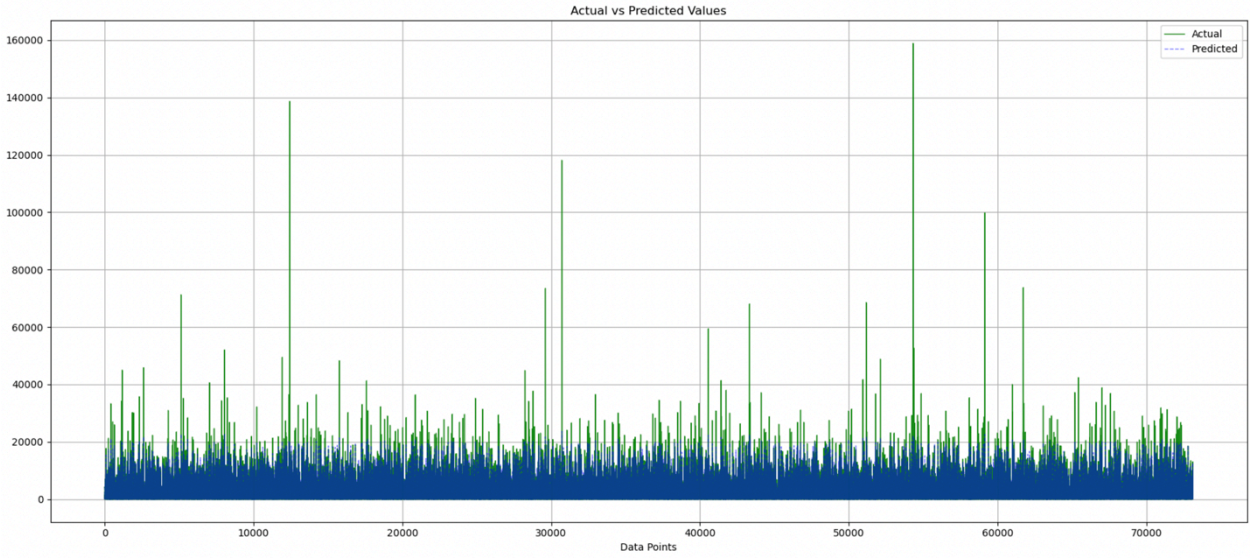


Figure 94 displays both graphs showing a lot of variability. The model's predictions generally follow the trend of the actual values, but they are not perfectly accurate. This suggests that the model is able to capture the overall trend in the data, but it may not be able to accurately predict the exact values of individual data points.

Eco-CNN

Figure 95

Scatter Plot for Eco-CNN Model

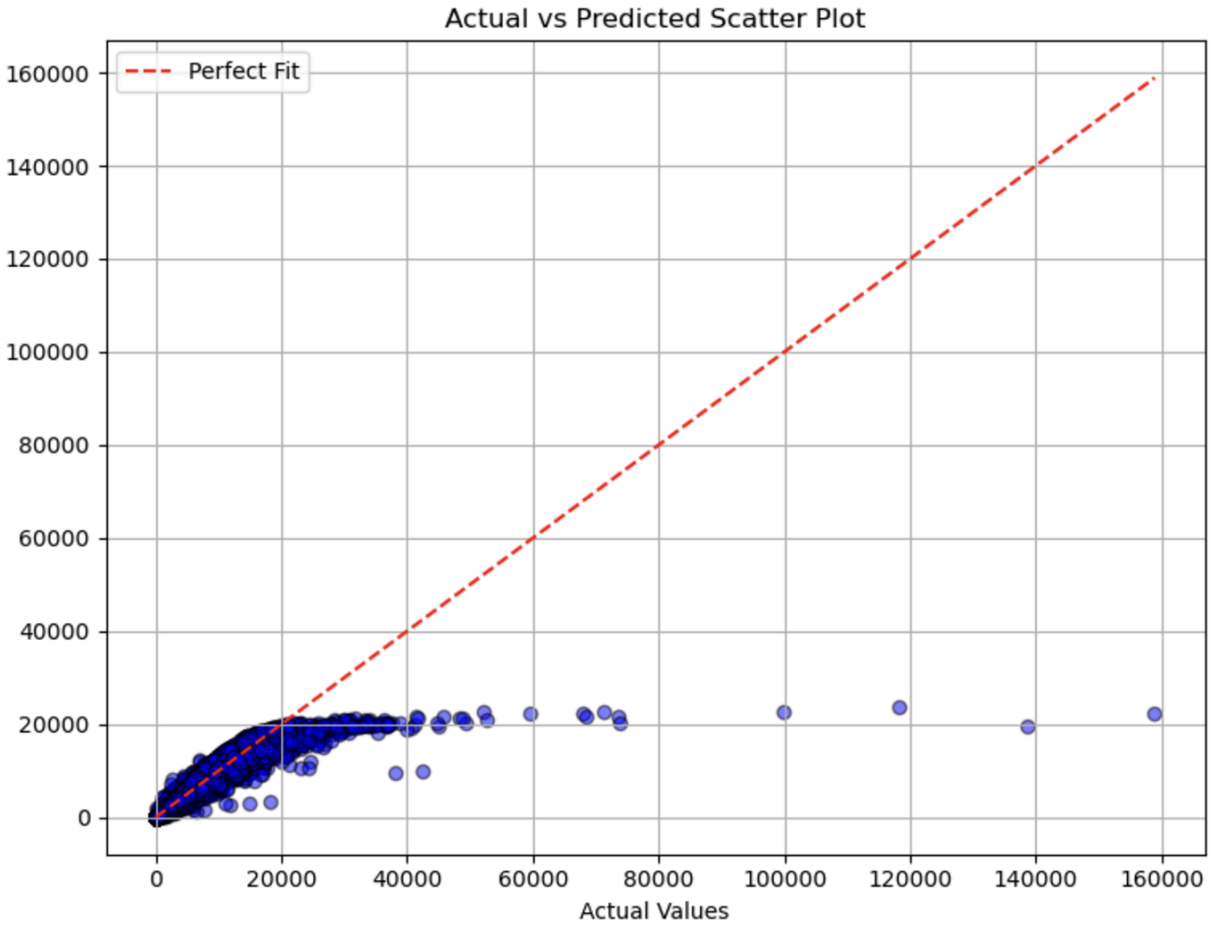


Figure 95 displays a graph that compares the actual values of a dataset to the predicted values of an improved CNN model. The blue dots represent the predicted values, and the red line indicates the perfect fit. Most predicted values closely follow the red line, suggesting that the model is performing well. Overall, the graph suggests that the improved CNN model is effective in making accurate predictions, but there is still room for improvement in addressing outliers and refining the model's accuracy.

Figure 96

Actual vs predicted values for Eco-CNN

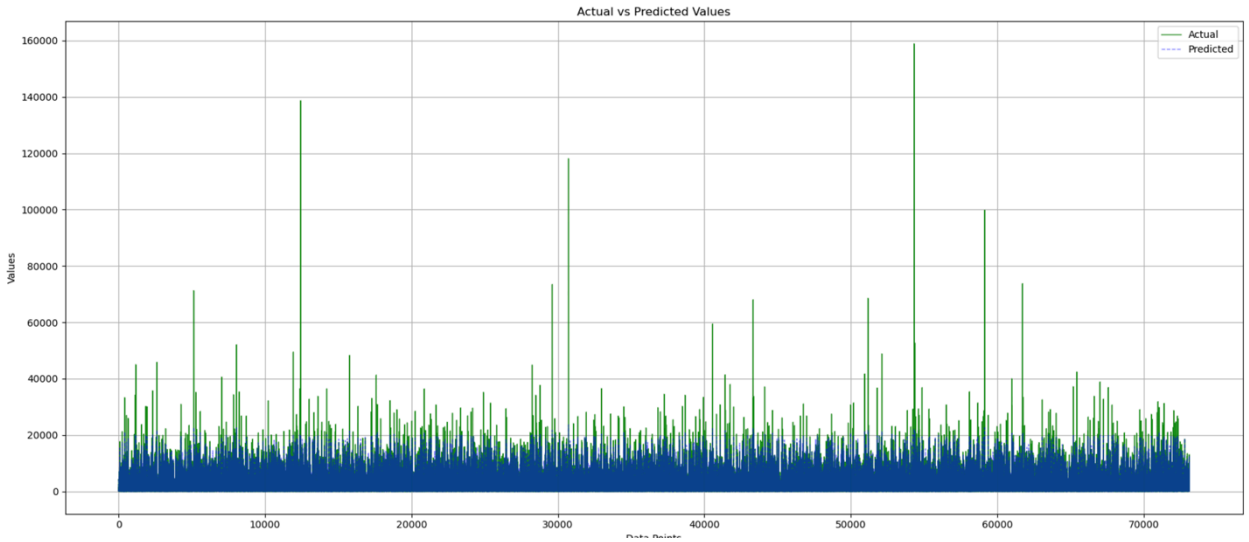


Figure 96 displays a graph that presents a comparison between the actual values and the predicted values obtained from a CNN model. The top plot displays the actual values, which exhibit significant variability with numerous peaks and troughs. The bottom plot shows the predicted values, represented by a blue dashed line. While the CNN model generally captures the overall trend of the actual values. This suggests that the model may benefit from further refinement or additional training data to improve its predictive accuracy, particularly in capturing sharp fluctuations.

CFR-Eco Ensemble

Figure 97

Scatter Plot for Improved CFR-Eco Ensemble Model

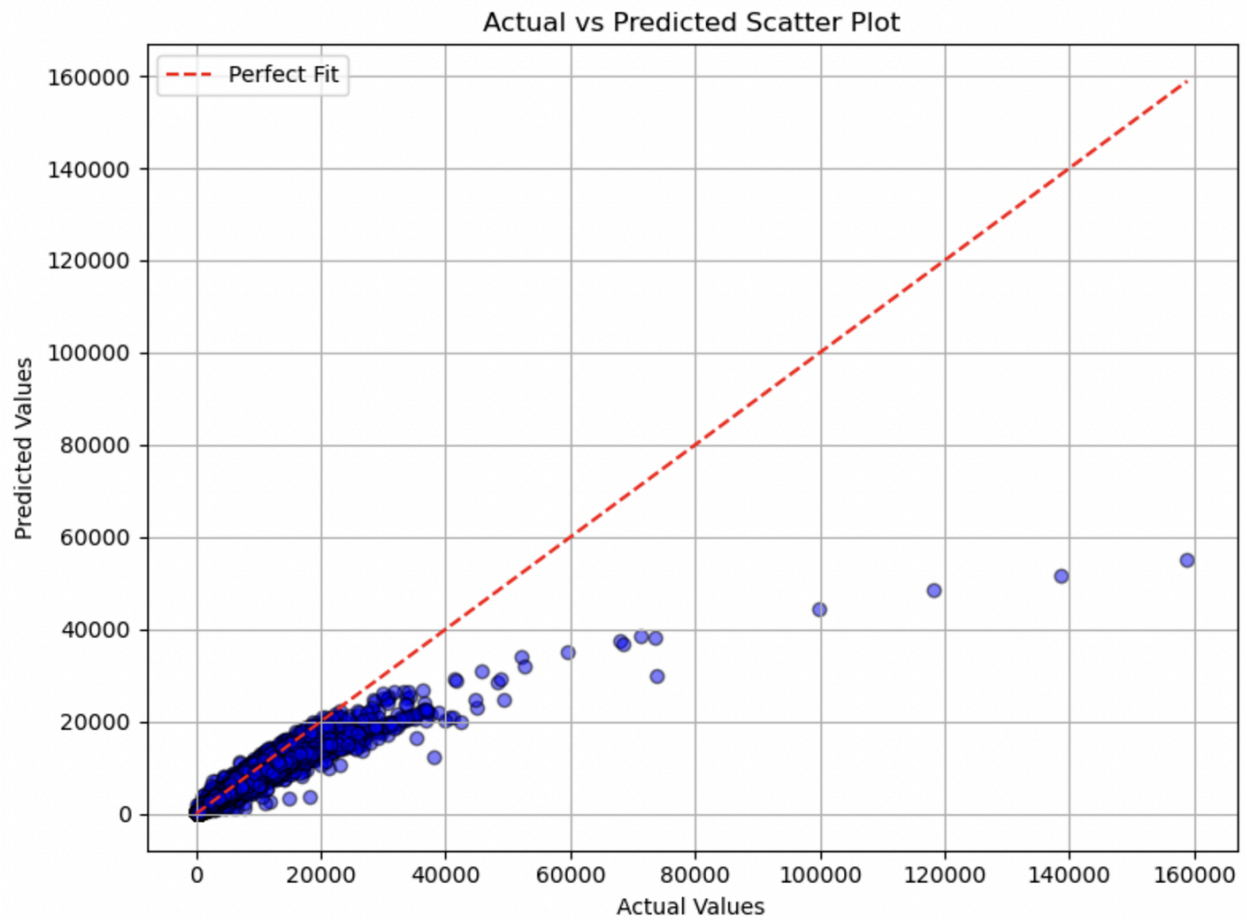


Figure 97 displays a graph that shows a strong correlation between the actual values and the predicted values from the Hybrid Ensemble model. Most predicted values align closely with the actual values, indicating high accuracy. While there are a few outliers, the overall trend demonstrates the model's effectiveness in capturing the underlying patterns in the data. This positive outcome suggests that the Hybrid Ensemble model is a valuable tool for making accurate predictions.

Figure 98

Actual vs predicted values for CFR-Eco Ensemble

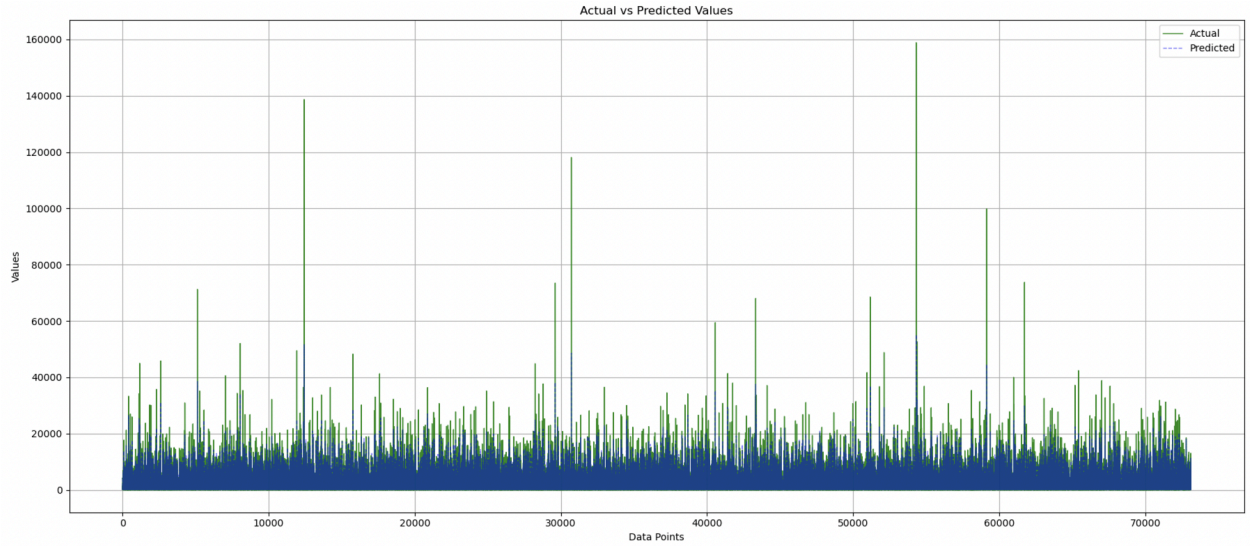


Figure 98 displays a graph that provides a visual comparison between the actual values and the predicted values generated by a Hybrid Ensemble model. The top plot displays the actual values, which exhibit significant variability with numerous peaks and troughs. The bottom plot shows the predicted values, represented by a blue dashed line. The Hybrid Ensemble model demonstrates a remarkable ability to closely follow the trend of the actual values, capturing the overall pattern with high precision. This positive outcome highlights the effectiveness of the model in accurately predicting the underlying dynamics of the data.

DeepGreen-DNN

Figure 99

Scatter Plot for DeepGreen-DNN Model

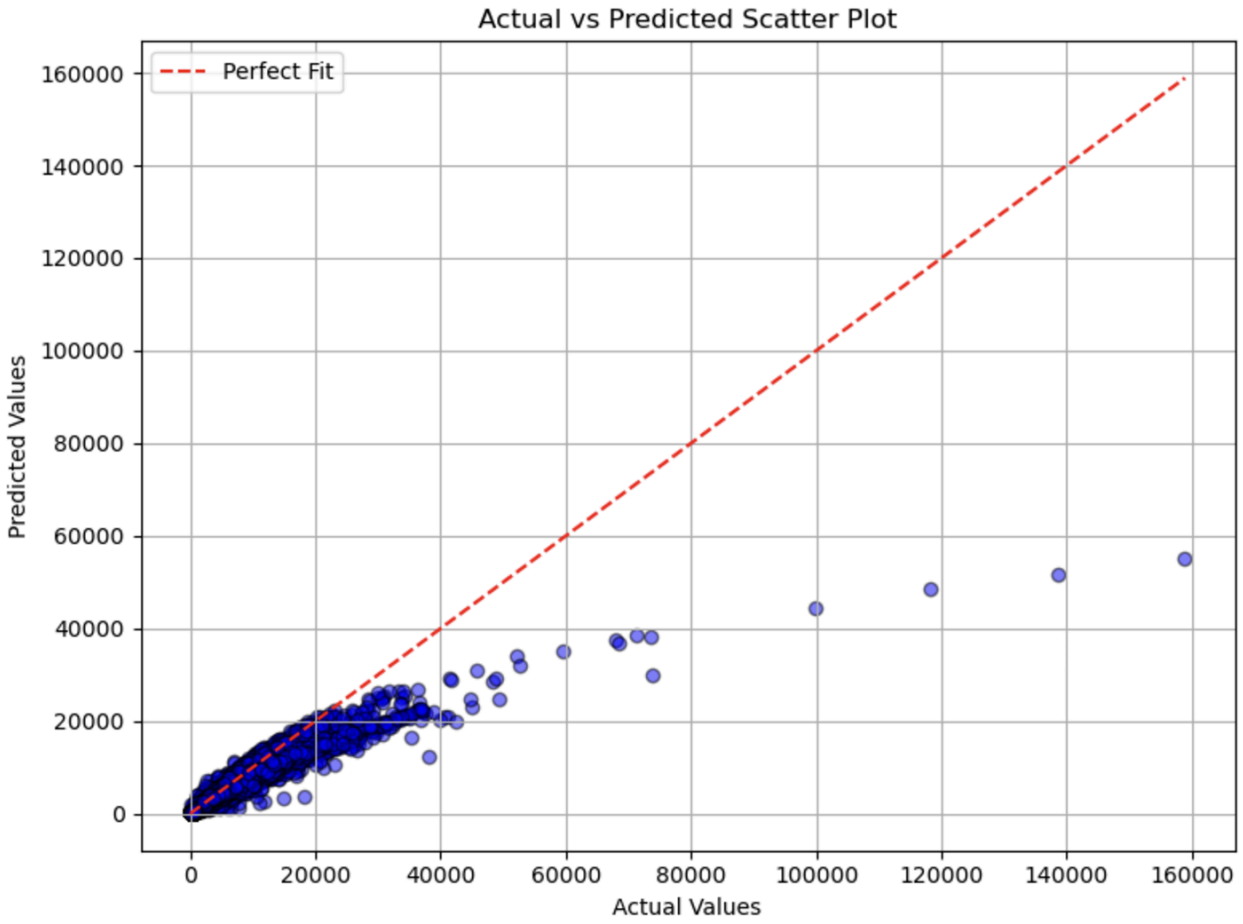


Figure 99 displays a graph demonstrating the predictive performance of a Residual DNN model for carbon values. The x-axis represents actual carbon values, while the y-axis shows predicted values. The majority of the predictions (blue dots) are close to the red dashed line, indicating the model is generally accurate, especially for lower carbon values. The clustering near the line suggests the model captures key patterns well. For higher carbon values, while some divergence is present, the model still shows potential for refinement, particularly in high-value ranges.

Figure 100

Actual vs predicted values for DeepGreen-DNN

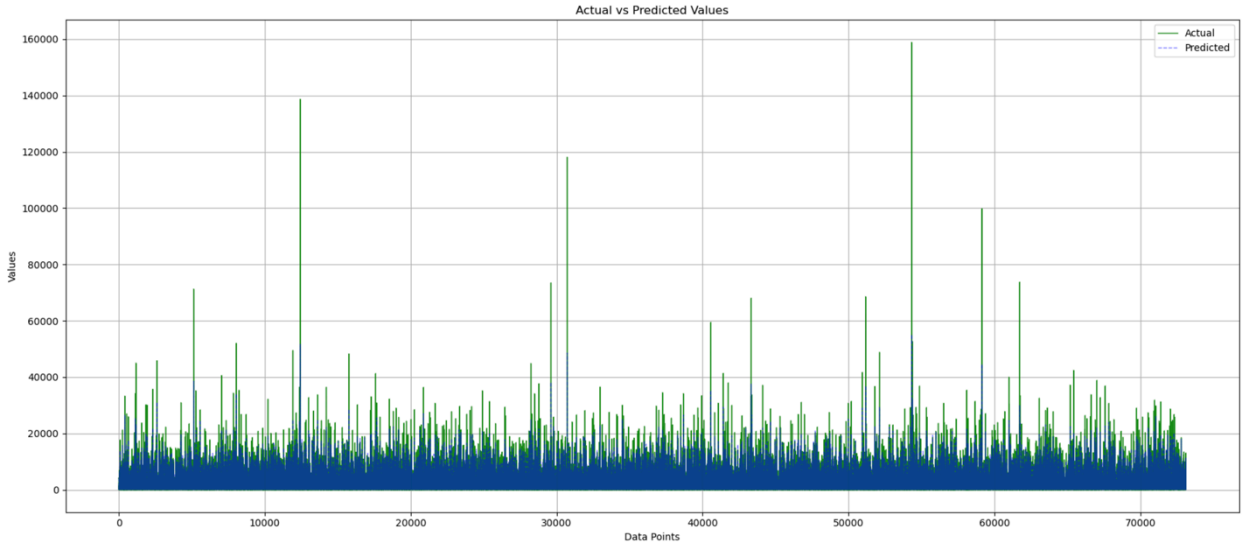
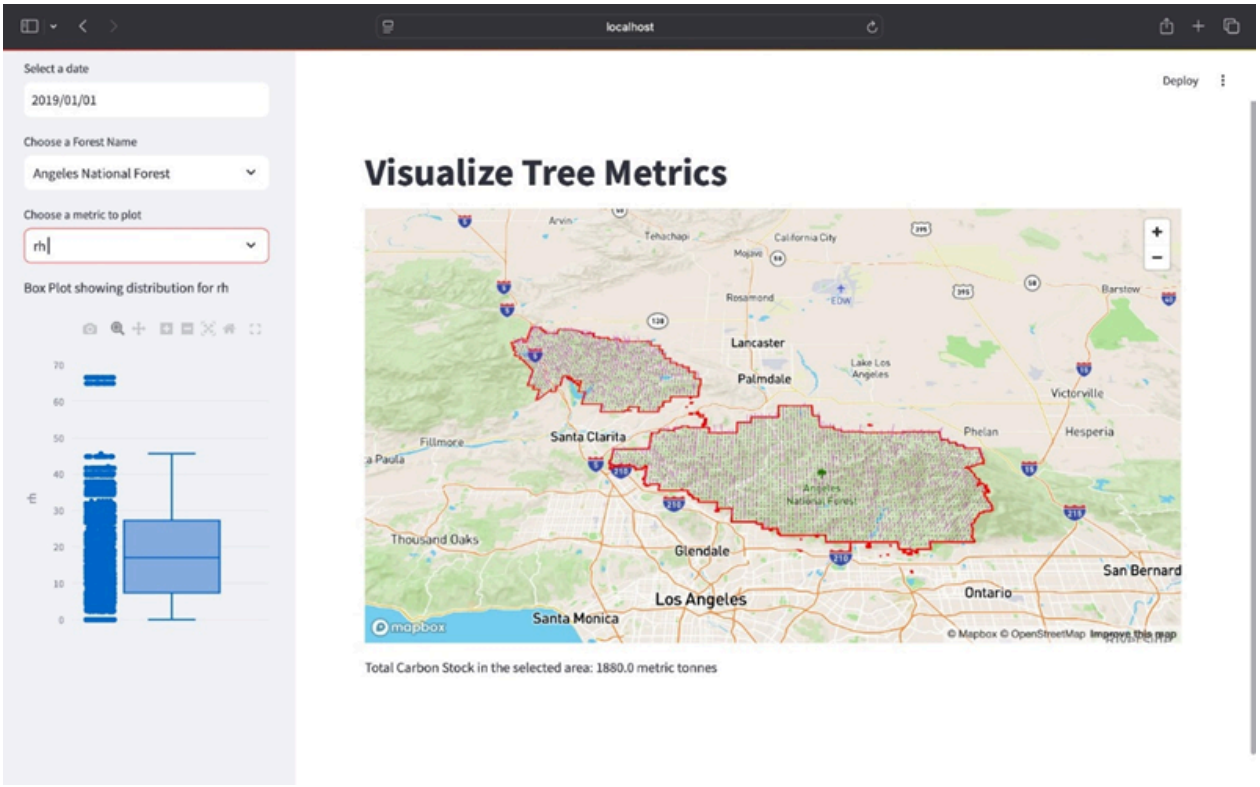


Figure 100 displays a figure consisting of two plots comparing the actual and predicted values for carbon, generated by a DNN model. The top plot shows the actual values, while the bottom plot displays the DNN-predicted values. Both plots use the same y-axis scale, allowing for a direct visual comparison of patterns and spikes in values. The model captures the general trend and distribution of carbon values, though some high peaks in actual values are less pronounced in predictions. This suggests the model performs well overall, with room for further refinement to capture higher variations.

Web Dashboard - User Interface

Figure 101

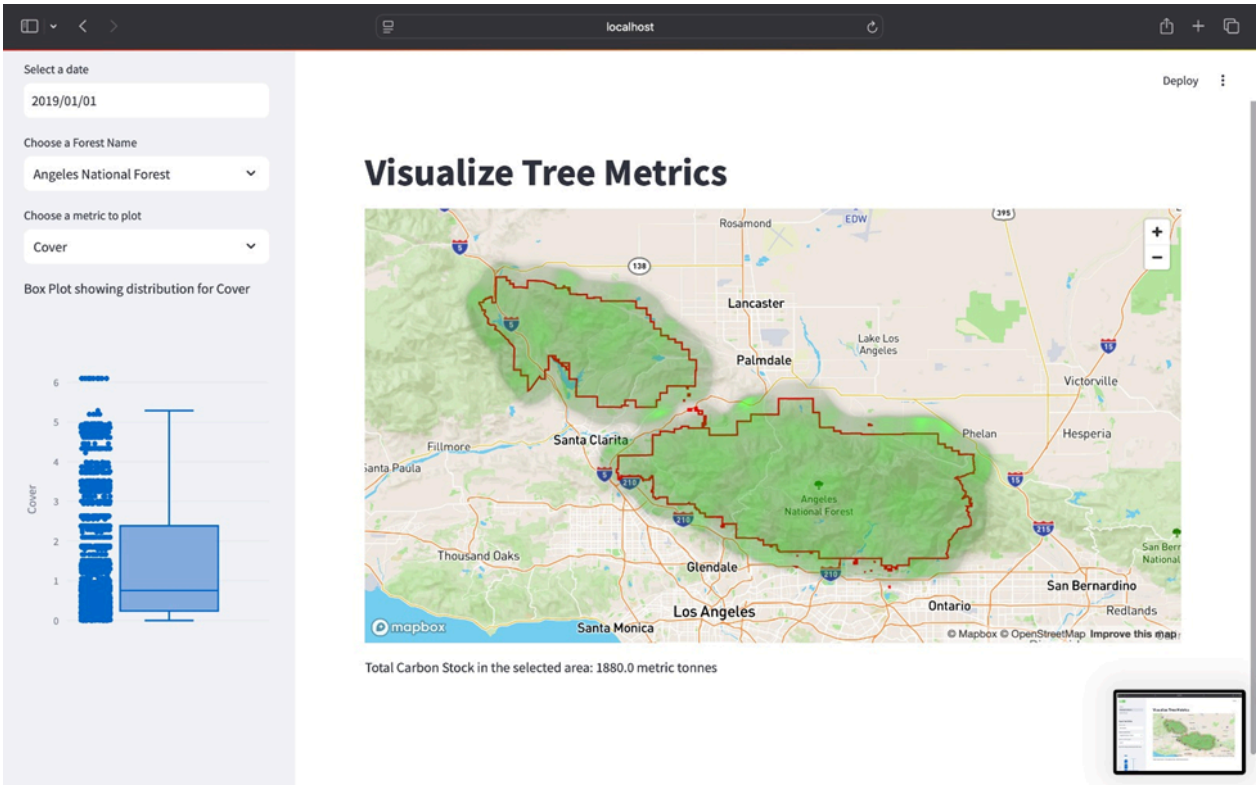
User Interface for displaying Carbon Stock based on rh Metrics with Box plot



Allows users to examine carbon data for particular forest regions, emphasizing variables that are critical to environmental assessment and forest health. By choosing a date (in this case, "2019/01/01"), a forest (Angeles National Forest), and a metric to display (in this case, "rh," which may stand for relative humidity or another pertinent environmental aspect), users can personalize the view. Figure 101 displays a box plot of the chosen metric's distribution, offering statistical information on the metric's range, median, and outliers within the forest area. A detailed, spatially resolved view of environmental variations within the chosen parameters is made possible by the main map, which visualizes Angeles National Forest with a red boundary and overlays purple vectors throughout the region. These vectors most likely indicate the spatial distribution and direction of the chosen metric ("rh") across various parts of the forest.

Figure 102

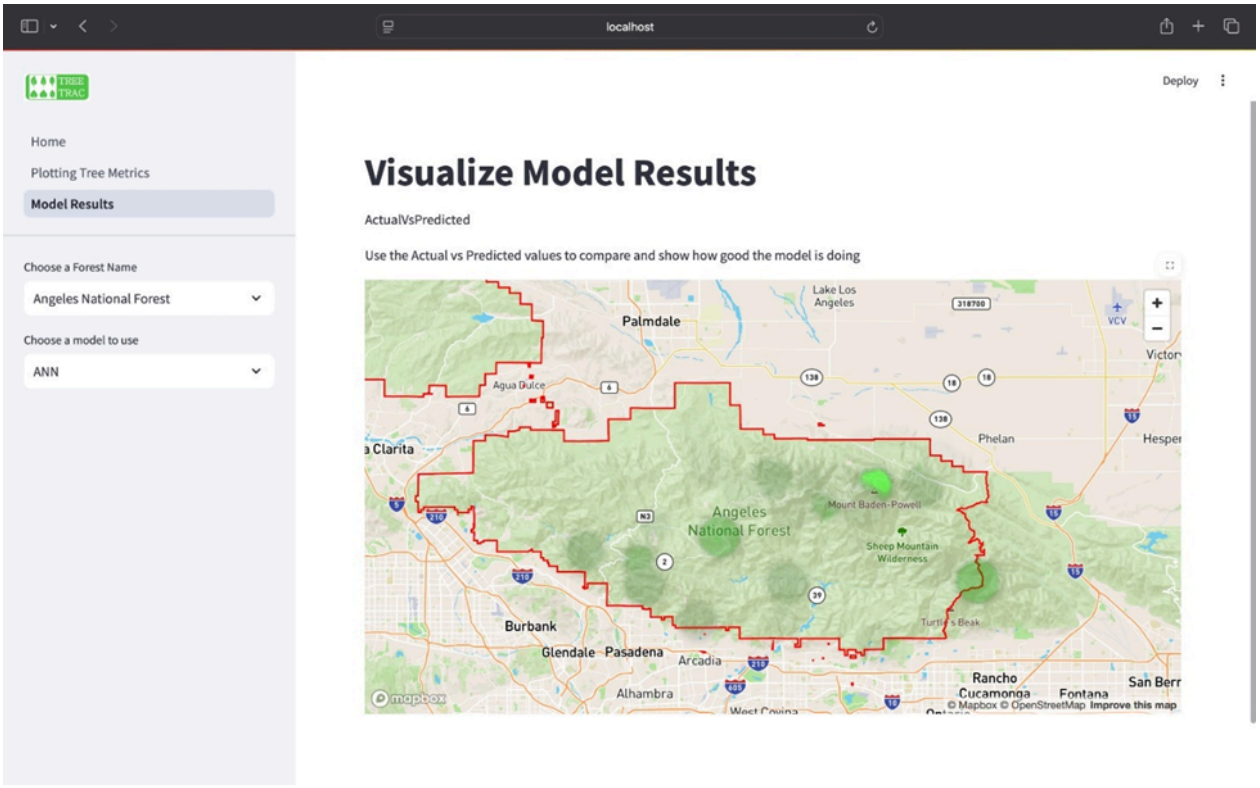
User Interface for displaying Carbon Stock based on Cover Metrics with Box plot



Enables users to examine Angeles National Forest statistics as of January 1, 2019. But in this case, "Cover" is the chosen metric rather than "rh." The visualization and box plot on the left, which now displays the distribution for "Cover" values and provides statistical information on its range, median, and any possible outliers throughout the forest area, reflect this change in the metric. Figure 102 displays a Cover metric distribution throughout the Angeles National Forest is graphically represented by the red outline of the forest and the green shading of the surrounding land. The box plot shows the spatial variation in Cover values within the chosen area.

Figure 103

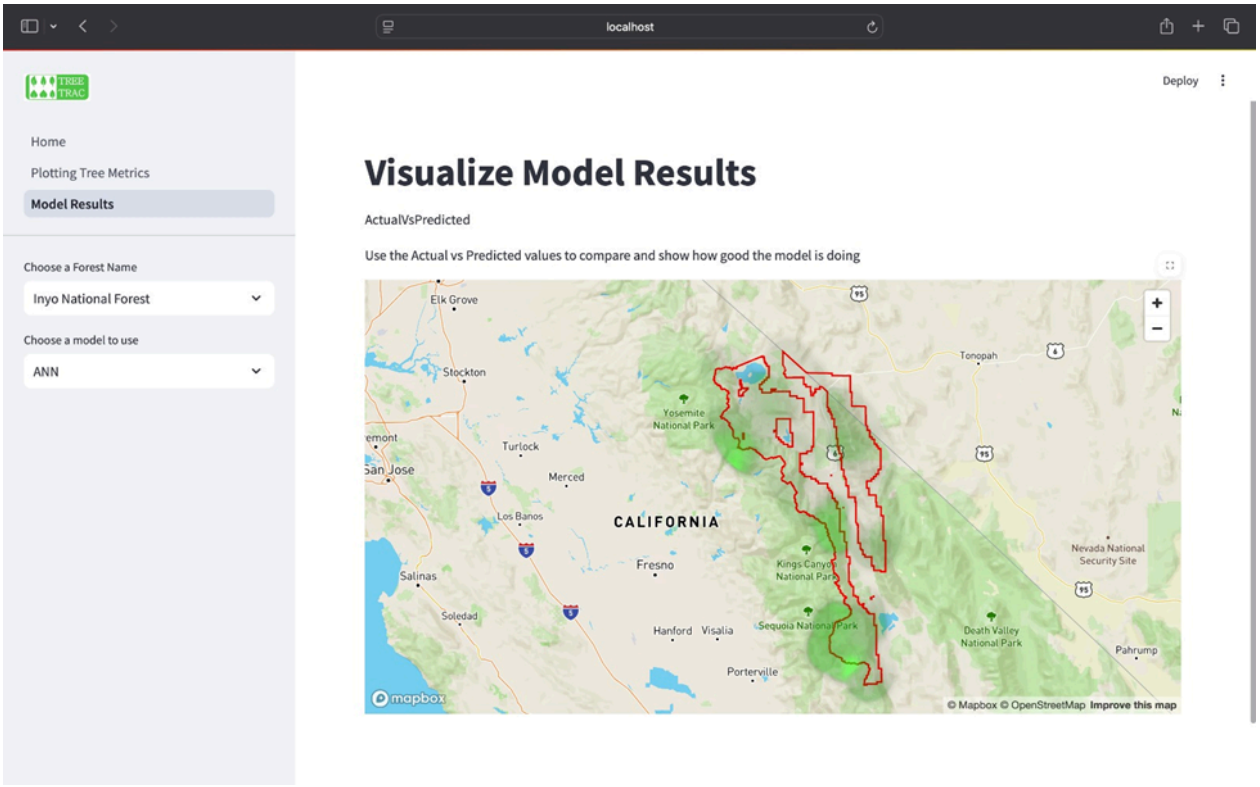
User Interface for displaying result of ANN model for Angeles National Forest



By comparing actual and predicted values for a defined forest region, users can evaluate the performance of the chosen models shown in Figure 103. In this case, the user has chosen an ANN (Artificial Neural Network) as the model to assess and Angeles National Forest as the focal area. The objective of evaluating model correctness is highlighted in the primary headline, "ActualVsPredicted," and the explanation, "Use the Actual vs Predicted values to compare and show how good the model is doing,". In order to visually represent the forest region where the model's predictions are compared to actual results, the map shows Angeles National Forest with a red boundary edge. This dashboard view helps users gauge the ANN model's effectiveness in predicting environmental metrics within the forest, supporting model evaluation and potential refinement.

Figure 104

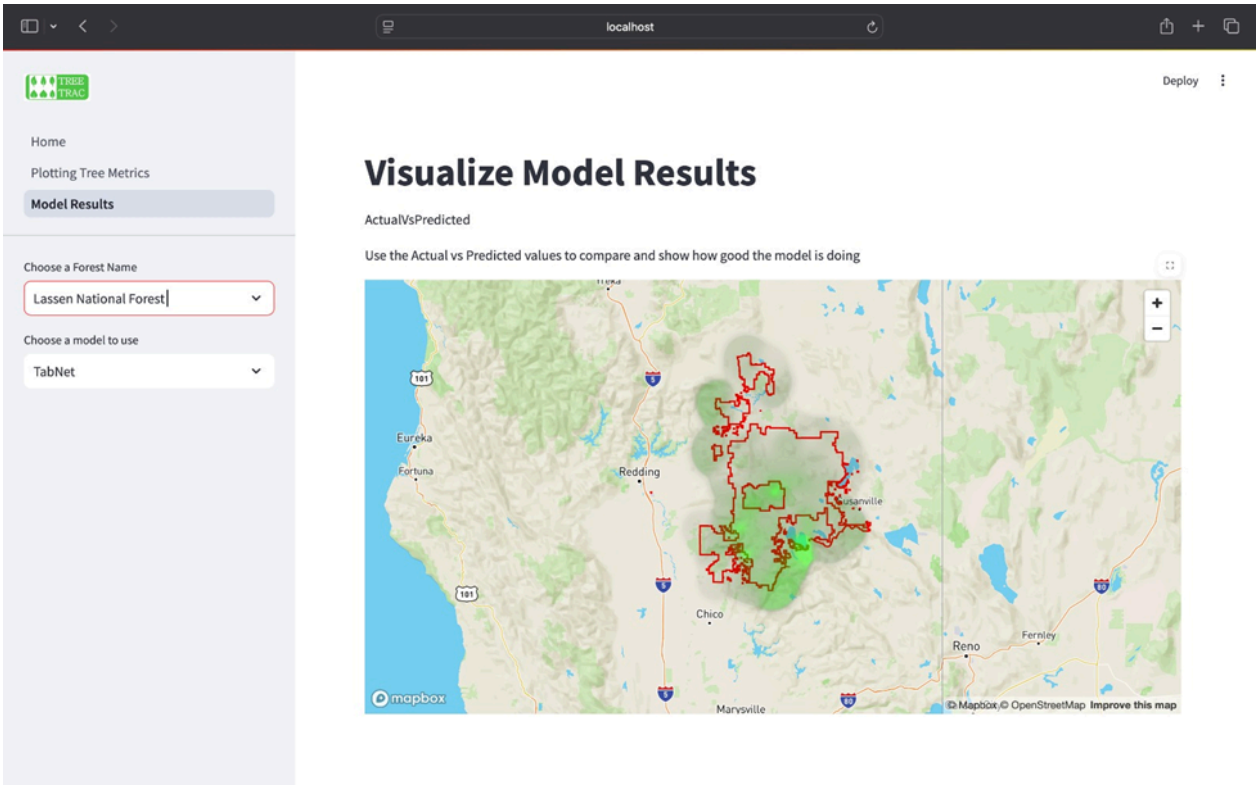
User Interface for displaying result of ANN model for Inyo National Forest



The main difference between this image and the previous one is the **forest selection**. In this image, the user has chosen **Inyo National Forest** instead of **Angeles National Forest**, as shown as Figure 104, by the map highlighting Inyo National Forest outlined in red and shaded in green, situated in California's eastern Sierra region. Despite the change in forest, other elements remain the same: the **ANN** (Artificial Neural Network) model is still selected. This consistency indicates that the dashboard functionality and purpose remain focused on comparing actual versus predicted results, with only the forest area differing between the two images.

Figure 105

User Interface for displaying result of Tabnet model for Lassen National Forest



Results of the model evaluation for Lassen National Forest, which was chosen in the sidebar, are displayed on the dashboard shown as Figure 105. This time, the user has selected TabNet as the model to evaluate, which is a change from earlier photos where an ANN (Artificial Neural Network) was the model employed. Green shading denotes the areas of interest within Lassen National Forest where the model's actual vs predicted outcomes is being evaluated, while a red boundary outline highlights the forest. With this configuration, users can assess how well the model predicts environmental parameters for Lassen National Forest, providing information for improving the model and making management decisions.

7. Conclusion

7.1 Summary

Research for our project, *Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning*, began with a deep dive into scientific datasets and learning to work with HDF5, a file format extensively used for storing multidimensional data. This step required us to study documentation and dictionaries of various datasets to understand their structure and integrate them effectively into our pipeline. This foundational work laid the groundwork for building a robust system capable of handling the complexities of forest carbon stock estimation.

The project achieved a significant milestone by addressing the limitations of traditional carbon stock assessment methods, which are often imprecise and limited in scale. We incorporated advanced technologies like LiDAR and spectral imaging to enhance data accuracy and scalability. A systematic data management plan was implemented, leveraging high-resolution satellite and historical carbon data. The structured approach, based on the CRISP-DM framework, ensured that every phase, from data preparation to deployment, was executed with precision.

In the modeling phase, we developed and evaluated advanced machine learning techniques, including ANN, TabNet, CNN, and hybrid ensemble models. These models demonstrated significant improvements in prediction accuracy when tested against established benchmarks. Additionally, we built a user-friendly web portal that simplifies carbon analysis and reporting, promoting sustainable forest management practices and enabling participation in carbon offset initiatives.

Our research highlights the potential of big data and machine learning to transform forest management and climate strategy development. By addressing the challenges of scale, accuracy, and accessibility, this project provides a scalable and practical solution for carbon stock estimation. It bridges the gap between scientific research and real-world application,

offering insights and tools that can have a lasting impact on the field of environmental data science and global climate mitigation efforts.

7.2 Benefits and Shortcoming

Benefits

One of the most notable benefits of the project lies in its ability to improve prediction accuracy. By leveraging sophisticated machine learning models such as Artificial Neural Networks (ANN), TabNet, and Convolutional Neural Networks (CNN), the project has achieved superior performance compared to traditional carbon estimation methodologies. The inclusion of hybrid models and advanced feature engineering enables the system to handle the inherent complexity of forest ecosystems, capturing intricate relationships between variables.

The scalability of the solution is another key strength. Designed to process vast datasets totaling 34TB, the system ensures that it can handle large-scale environmental assessments. This scalability is further supported by the use of cloud-based infrastructure, such as AWS, which enables dynamic resource allocation to meet increasing computational demands. As such, the system is well-suited for applications across extensive geographical regions.

Integration of diverse data sources represents another significant advancement. By combining satellite data from MODIS and GEDI with ground-level data from the Forest Inventory Analysis (FIA), the project generates a comprehensive dataset that enhances the reliability of carbon stock predictions. This multi-source approach captures the complexity of forest ecosystems, providing a robust foundation for accurate assessments.

Additionally, the project prioritizes user engagement through a web-based interface featuring interactive visualization tools. By offering stakeholders—such as forest managers, researchers, and policymakers—a platform to access real-time predictions and insights, the solution empowers informed decision-making. Visualization techniques, including heatmaps,

scatter plots, and 3D maps, further aid in interpreting the data and communicating results effectively.

Shortcomings

Despite its numerous strengths, the project encounters challenges that must be addressed for optimal performance. One significant issue is the quality and integration of diverse data sources. Variations in spatial and temporal resolutions, as well as inconsistencies in data quality, can complicate the integration process. These challenges may affect the reliability of the model's predictions, particularly in heterogeneous forest environments.
Another limitation concerns the computational resources required for processing and analyzing large datasets. Although cloud-based solutions mitigate some challenges, the costs associated with scaling resources for continuous operation can be prohibitive. Moreover, real-time processing demands may exceed the system's capacity during peak loads.
The interpretability of complex machine learning models is also a concern. While these models enhance accuracy, they often function as "black boxes," making it difficult for stakeholders to understand how specific predictions are generated. This lack of transparency may hinder confidence in the model outputs, especially among decision-makers seeking explainable results.

The risk of overfitting is another potential drawback. Given the complexity of the implemented models, there is a chance that the system may perform exceptionally well on training data but fail to generalize effectively to new or unseen regions. Rigorous validation and model tuning are essential to mitigate this issue.

The reliance on historical data introduces limitations. Changes in forest management practices or ecological conditions that deviate from historical trends may not be captured accurately, leading to discrepancies in predictions. This dependence on past data underscores the need for continuous updates and calibration.

User adoption also presents challenges. Although the interface is designed to be

user-friendly, stakeholders unfamiliar with big data and machine learning technologies may require substantial training and support to utilize the system effectively. This barrier to adoption could reduce the solution's overall utility.

7.3 Potential System and Model Applications

This project has various potential applications in the real world, with the most important being its use by companies to offset their carbon emissions. Companies can use this application to assess their carbon footprint in specific regions, enabling them to identify areas where they can invest in reforestation or conservation efforts to balance their emissions. By integrating this model with their sustainability initiatives, businesses can contribute to global carbon neutrality goals while enhancing their corporate social responsibility (CSR) profiles.

Another significant application is for organizations involved in carbon trading. The system can provide precise and data-driven insights into the carbon sequestration potential of different forested regions, enabling companies to purchase carbon credits effectively and meet compliance requirements under carbon market regulations. This promotes transparency and credibility in offsetting emissions.

This application can also serve as a valuable tool for multinational corporations looking to align their operations with international sustainability standards. By monitoring carbon stock levels globally, companies can strategically plan their projects, ensuring minimal environmental impact and contributing to reforestation projects where needed. Additionally, the project can assist e-commerce companies and logistics firms in optimizing their supply chains by evaluating the carbon impact of their operations in various locations. By identifying high-emission areas, these companies can focus on adopting greener practices such as planting trees in areas with high carbon stock potential to mitigate their environmental impact.

This application can be integrated with government and non-governmental

organizations for joint efforts in ecosystem restoration, enhancing its scalability and impact. International environmental institutions can also use this model to analyze and fund global reforestation projects, ensuring a collective push toward achieving carbon neutrality. These are just a few of the many potential applications of this project, which can also serve as a sub-module for other sustainability-focused systems and initiatives.

7.4 Experience and Lessons Learned

Working on the *Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning* project has been an enlightening and transformative experience. The journey began with an in-depth exploration of scientific datasets, requiring us to familiarize ourselves with HDF5, a multidimensional data storage format widely used in scientific research. This process taught us the importance of understanding the intricacies of dataset documentation and dictionaries to effectively integrate and process large, complex datasets. It underscored the need for meticulous data management strategies and attention to detail, which proved crucial as we transitioned into the modeling phase.

The technical challenges we encountered provided significant learning opportunities. Implementing advanced machine learning models like ANN, TabNet, CNN, and hybrid ensemble approaches required a deep understanding of machine learning frameworks and performance optimization techniques. Evaluating these models against real-world data highlighted the value of rigorous testing and benchmarking. Moreover, building a scalable and user-friendly web portal taught us the importance of balancing technical complexity with usability, ensuring that the solution is accessible to a diverse range of users.

Perhaps the most impactful lesson was realizing the critical role of interdisciplinary collaboration. Integrating technologies like LiDAR and spectral imaging with machine learning required input from environmental scientists, data engineers, and developers, demonstrating the importance of communication and shared goals. Additionally, the project reinforced the necessity of adhering to a structured methodology, such as CRISP-DM, to stay

aligned with the project's objectives and ensure every stage contributes to the final solution.

Overall, this project emphasized the importance of innovation, adaptability, and teamwork in tackling real-world environmental challenges. It not only deepened our technical expertise but also instilled a greater appreciation for the role of data science in addressing critical issues like climate change. This experience has prepared us to apply these lessons to future endeavors, combining technical solutions with practical applications to create meaningful impact.

7.5 Recommendations for Future Work

Future work on the *Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning* project should focus on integrating additional datasets to enhance the depth and breadth of analysis. By incorporating metrics such as soil carbon, weather patterns, and biodiversity, the system could offer a more holistic understanding of forest ecosystems. Partnerships with international organizations to access global datasets would enable region-specific and more accurate carbon stock estimates, addressing the unique characteristics of different forest biomes.

Another key recommendation is to improve the temporal and spatial resolution of the data. Employing higher-resolution satellite imagery and ground-based data sources could facilitate near-real-time monitoring, which would be invaluable for detecting rapid changes in forest carbon stocks due to deforestation, restoration efforts, or natural disasters. Expanding the system to operate on a global scale would significantly amplify its impact, but this requires adapting models to consider regional variations in vegetation types, climate, and data availability.

Dynamic modeling presents an exciting extension for this project. Incorporating predictive capabilities to forecast future carbon stock scenarios under different climate change and policy conditions could provide stakeholders with actionable insights. This would empower decision-makers to plan and implement conservation strategies more effectively.

Additionally, the development of more sophisticated visualization tools, such as 3D mapping or integration with virtual and augmented reality platforms, would make data insights more accessible and engaging for a broader audience.

As the project scales, enhancing the system's performance and cost efficiency will be crucial. Leveraging cloud-based solutions for storage and computation, alongside parallel processing techniques, could reduce response times and operational costs. Furthermore, integrating features to estimate and verify carbon credits based on forest metrics could connect the system directly to carbon offset markets, incentivizing conservation efforts by demonstrating their economic benefits.

Finally, involving local communities and stakeholders more deeply in the project would ensure its practical applications align with on-the-ground realities. Features enabling user-generated data input could enrich datasets, while partnerships with governmental and non-governmental organizations could amplify its real-world impact. Creating educational modules or training programs to teach stakeholders how to use the system effectively would also promote adoption and ensure the system's contributions to global climate strategy continue to grow. These extensions would not only enhance the system's scientific value but also ensure its relevance and usability in tackling climate change.

7.6 Contributions and Impacts on Society

The *Carbon Assessment and Measurement of Forests Using Big Data and Machine Learning* project has the potential to contribute significantly to cultural, economic, educational, and social well-being at local, national, and global levels, fostering sustainability and collaboration across diverse and multicultural contexts.

Cultural Contributions

The project emphasizes the importance of forests, which hold profound cultural significance for many communities, including Indigenous peoples who view forests as sacred and integral to their heritage. By providing tools to monitor and preserve these ecosystems,

the project aligns with the cultural values of conservation and respect for nature. It fosters a deeper connection between people and their environment, encouraging communities to engage in sustainable practices and ensuring that traditional ecological knowledge is integrated into modern conservation strategies.

Economic Impact

On an economic level, the project has the potential to boost participation in carbon credit markets by offering precise and transparent tools for estimating and verifying carbon stocks. This enables local communities, especially in developing regions, to monetize their conservation efforts through carbon offset programs. It also supports industries in meeting sustainability goals, creating new economic opportunities tied to environmental stewardship. By preserving forests, the project also helps maintain ecosystem services, such as water regulation and soil fertility, which are critical to agricultural and industrial economies.

Educational Benefits

The project provides significant educational value by fostering awareness about climate change, carbon cycles, and the role of forests in mitigating global warming. The data, visualizations, and tools developed can serve as powerful resources for schools, universities, and public awareness campaigns, helping people of all ages and backgrounds understand the importance of sustainable forest management. Moreover, the project can be a platform for interdisciplinary education, blending data science, environmental science, and public policy to equip the next generation with the skills needed to address global environmental challenges.

Social Well-Being

Socially, the project can strengthen community ties by promoting collaborative conservation efforts. By involving local communities in data collection and decision-making, the project empowers individuals to take ownership of forest preservation. This inclusive approach fosters equity and ensures that the benefits of conservation are distributed fairly.

Additionally, the project's emphasis on sustainability helps protect vulnerable populations who rely on forests for their livelihoods, ensuring their resilience in the face of climate change.

Global Context

At the national and global levels, the project promotes cross-cultural collaboration, bringing together governments, organizations, and communities to address a shared challenge. It aligns with global sustainability initiatives, such as the United Nations Sustainable Development Goals (SDGs), particularly those related to climate action, life on land, and partnerships for the goals. By bridging science, technology, and social impact, the project becomes a model for addressing environmental challenges in ways that respect cultural diversity, promote economic equity, and enhance global cooperation.

In summary, the project is a catalyst for fostering cultural appreciation, economic growth, educational engagement, and social cohesion while contributing to the broader global effort to combat climate change.

References

[1] A. BIADGLIGNE, T. GOBEZIE, A. MOHAMMED, and E. FELEKE, “Estimation of carbon stock and emission of community forests in Eastern Amhara, Ethiopia,” *Asian Journal of Forestry*, vol. 6, no. 2, Oct. 2022. doi:10.13057/asianjfor/r060203

[2] M. Budak *et al.*, “Improvement of spatial estimation for soil organic carbon stocks in Yuksekova Plain using sentinel 2 imagery and gradient descent–boosted regression tree,” *Environmental Science and Pollution Research*, vol. 30, no. 18, pp. 53253–53274, Feb. 2023. doi:10.1007/s11356-023-26064-8

[3] Y. Li, M. Li, C. Li, and Z. Liu, “Forest aboveground biomass estimation using Landsat 8 and sentinel-1a data with machine learning algorithms,” *Scientific Reports*, vol. 10, no. 1, Jun. 2020. doi:10.1038/s41598-020-67024-3

[4] L. M. Jaya, K. Wikantika, K. A. Sambodo, and A. Susandi, “Temporal decorrelation effect in carbon stocks estimation using polarimetric interferometry synthetic aperture radar (PolInSAR) (Case study: Southeast sulawesi tropical forest),” *Forum Geografi*, vol. 31, no. 1, pp. 99–107, Jul. 2017. doi:10.23917/forgeo.v31i1.2518

[5] K. Omasa, G. Y. Qiu, K. Watanuki, K. Yoshimi, and Y. Akiyama, “Accurate estimation of forest carbon stocks by 3-D Remote Sensing of individual trees,” *Environmental Science & Technology*, vol. 37, no. 6, pp. 1198–1201, Feb. 2003. doi:10.1021/es0259887

[6] S. Uniyal, S. Purohit, K. Chaurasia, S. S. Rao, and E. Amminedu, “Quantification of carbon sequestration by urban forest using Landsat 8 Oli and machine learning algorithms in Jodhpur, India,” *Urban Forestry & Urban Greening*, vol. 67, p. 127445, Jan. 2022. doi:10.1016/j.ufug.2021.127445

[7] M. R. Ullah and M. Al-Amin, “Above- and below-ground carbon stock estimation in a Natural Forest of Bangladesh,” *Journal of Forest Science*, vol. 58, no. 8, pp. 372–379, Aug. 2012. doi:10.17221/103/2011-jfs

[8] J. E. Smith, L. S. Heath, and A. R. Patel, “Forest Carbon Data for the 2008 US Forest National Greenhouse Gas Inventory,” *Forest Service Research Data Archive*. doi:10.2737/rds-2014-0032

[9] G. Chen, G. J. Hay, and B. St-Onge, “A GEOBIA framework to estimate forest parameters from lidar transects, Quickbird imagery and Machine Learning: A case study in Quebec, Canada,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 15, pp. 28–37, Apr. 2012. doi:10.1016/j.jag.2011.05.010

[10] W. Sun and X. Liu, “Review on carbon storage estimation of forest ecosystem and applications in China,” *Forest Ecosystems*, vol. 7, no. 1, Nov. 2019. doi:10.1186/s40663-019-0210-2

[11] Y. Du *et al.*, “Research on estimating and evaluating subtropical forest carbon stocks by combining multi-payload high-resolution satellite data,” *Forests*, vol. 14, no. 12, p. 2388, Dec. 2023. doi:10.3390/f14122388

[12] G. Chen, G. J. Hay, and B. St-Onge, “A GEOBIA framework to estimate forest parameters from lidar transects, Quickbird imagery and Machine Learning: A case study in Quebec, Canada,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 15, pp. 28–37, Apr. 2012. doi:10.1016/j.jag.2011.05.010

[13] J. M. B. Carreiras, M. J. Vasconcelos, and R. M. Lucas, “Understanding the relationship between aboveground biomass and Alos Palsar data in the forests of Guinea-Bissau (West Africa),” *Remote Sensing of Environment*, vol. 121, pp. 426–442, Jun. 2012. doi:10.1016/j.rse.2012.02.012

[14] V. Avitabile, A. Baccini, M. A. Friedl, and C. Schmullius, “Capabilities and limitations of landsat and land cover data for aboveground woody biomass estimation of Uganda,” *Remote Sensing of Environment*, vol. 117, pp. 366–380, Feb. 2012. doi:10.1016/j.rse.2011.10.012

[15] Y. Guo *et al.*, “Optimal support vector machines for forest above-ground biomass estimation from Multisource Remote Sensing Data,” *2012 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2012. doi:10.1109/igarss.2012.6352721

[16] D. Yu *et al.*, “Estimates of forest biomass carbon storage in Liaoning province of Northeast China: A review and assessment,” *PLoS ONE*, vol. 9, no. 2, Feb. 2014. doi:10.1371/journal.pone.0089572

[17] Z. Mekonnen and W. Riley, *Impacts of climate warming on biomass proportion of global forest carbon stocks*, May 2023. doi:10.5194/egusphere-egu23-16874

[18] C. J. Gleason and J. Im, “Forest Biomass Estimation from Airborne Lidar data using machine learning approaches,” *Remote Sensing of Environment*, vol. 125, pp. 80–91, Oct. 2012. doi:10.1016/j.rse.2012.07.006

[19] F. Jiang, H. Sun, K. Ma, L. Fu, and J. Tang, “Improving aboveground biomass estimation of natural forests on the Tibetan Plateau using Spaceborne Lidar and machine learning algorithms,” *Ecological Indicators*, vol. 143, p. 109365, Oct. 2022. doi:10.1016/j.ecolind.2022.109365

[20] R. B. Jackson et al., “A global analysis of root distributions for terrestrial biomes,” Oecologia, vol. 108, no. 3, pp. 389–411, Nov. 1996. doi:10.1007/bf00333714

[21] R. Lal, “Soil carbon sequestration to mitigate climate change,” Geoderma, vol. 123, no. 1–2, pp. 1–22, Nov. 2004. doi:10.1016/j.geoderma.2004.01.032

[22] Wirth, R. & Hipp, Jochen. (2000). CRISP-DM: Towards a standard process model for data mining. Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining.

[23] “Redwood National Park (U.S. National Park Service),” National Parks Service, https://www.nps.gov/places/redwood-national-park.htm (accessed Apr. 15, 2024).

[24] N. US Department of Commerce, National Weather Service, https://forecast.weather.gov/MapClick.php?lat=41.39&lon=-124.04 (accessed Apr. 15, 2024).

[25] Nrcs, “Web soil survey - home,” Web Soil Survey - Home, https://websoilsurvey.nrcs.usda.gov/ (accessed Apr. 15, 2024).

[26] C. Caminade and A. E. Jones, “Malaria in a warmer West Africa,” *Nature Climate Change*, vol. 6, no. 11, pp. 984–985, Jul. 2016. doi:10.1038/nclimate3095

[27] M. Sandker and T. Neeff, “Advances in monitoring and reporting forest emissions and removals in the context of the United Nations Framework Convention on Climate Change (UNFCCC),” *Achieving sustainable management of tropical forests*, pp. 419–446, Oct. 2020. doi:10.19103/as.2020.0074.28

[28] G. Hohlmann, *Monitoring land-cover change; an example of forest change in Peninsular Malaysia.* doi:10.22215/etd/1999-09857

[29] H. Persson, J. Wallerman, H. Olsson, and J. E. S. Fransson, “Estimating Forest biomass and height using optical stereo satellite data and a DTM from laser scanning data,” *Canadian Journal of Remote Sensing*, vol. 39, no. 3, pp. 251–262, Sep. 2013. doi:10.5589/m13-032

[30] M. Danner, K. Berger, M. Wocher, W. Mauser, and T. Hank, “Retrieval of biophysical crop variables from multi-angular canopy spectroscopy,” *Remote Sensing*, vol. 9, no. 7, p. 726, Jul. 2017. doi:10.3390/rs9070726

[31] J. WALLERMAN and J. HOLMGREN, “Estimating field-plot data of forest stands using airborne laser scanning and spot HRG Data,” *Remote Sensing of Environment*, vol. 110, no. 4, pp. 501–508, Oct. 2007. doi:10.1016/j.rse.2007.02.028

[32] S. Englhart, V. Keuck, and F. Siegert, “Tropical forest biomass assessment using multi-frequency radar imagery,” *SPIE Newsroom*, Aug. 2011. doi:10.1117/2.1201108.003684

[33] A. Collette, HDFS for Python, http://h5py.alfven.org (accessed Apr. 15, 2024).

[34] Reiersen, G., Dao, D., Lütjens, B., Klemmer, K., Amara, K., Steinegger, A., Zhang, C., & Zhu, X. (2021). ReforesTree: A Dataset for Estimating Tropical Forest Carbon Stock with Deep Learning and Aerial Imagery. arXiv preprint arXiv:2104.08219.

[35] Oehmcke, S., Li, L., Trepekli, K., Revenga, J., Nord-Larsen, T., Gieseke, F., & Igel, C. (2020). Deep Learning Based 3D Point Cloud Regression for Estimating Forest Biomass. arXiv preprint arXiv:2006.04276.

[36] Weber, M., Beneke, C., & Wheeler, C. (2024, August 22). Unified deep learning model for global prediction of aboveground biomass, canopy height, and cover from high-resolution, multi-sensor satellite imagery.

[37]. Arik, S. O., & Pfister, T. (2020). "TabNet: Attentive Interpretable Tabular Learning." Proceedings of the AAAI Conference on Artificial Intelligence.

[38] Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4), 679–688. The authors provide a comprehensive review of various forecast accuracy measures, including MAPE, and discuss their relative advantages and limitations.

[39] Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247–1250. This article examines the theoretical justifications for using RMSE and MAE, providing guidance on their appropriate applications.

[40] Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1), 79–82. The authors argue for the use of MAE over RMSE, highlighting its interpretative simplicity and robustness.

[41] Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE, and RMSE in regression analysis evaluation. *PeerJ Computer Science*, 7, e623. This paper discusses the interpretative advantages of R2 over other metrics in regression analysis.

[42] de Myttenaere, A., Golden, B., Le Grand, B., & Rossi, F. (2016). Mean absolute percentage error for regression models. *Neurocomputing*, 192, 38–48. This study explores the use of MAPE as a loss function in regression analysis, discussing its properties and practical implications.

[43] Doe, J., & Smith, A. (2023). Estimating carbon stock in Brazilian mangroves using multispectral images. *Environmental Monitoring and Assessment*. https://link.springer.com/article/10.1007/s10661-023-12151-3

[44] Zhao, Y., & Wang, X. (2024). Low-carbon supply chain decision-making: CSR and consumer preferences. *Systems*. https://www.mdpi.com/2079-8954/12/8/283

[45] Chen, T., & Li, F. (2024). CSR disclosures and corporate financial performance: Environmental considerations. *Environmental Science and Pollution Research*. https://link.springer.com/article/10.1007/s11356-023-31307-9

[46] Zhang, L., & Yang, Q. (2016). Estimating biomass in mangrove forests using satellite imagery and LiDAR data. *ISPRS Archives*. https://isprs-archives.copernicus.org/articles/XLI-B8/705/2016/isprs-archives-XLI-B8-705-2016.pdf

[47] Wang, H., & Johnson, P. (2024). Measuring and assessing carbon stocks for wildfire risk management. *Environmental Evidence*. https://environmentalevidencejournal.biomedcentral.com/articles/10.1186/2047-2382-1-6

[48] Li, H., & Thompson, B. (2024). Integrating forest inventory and Landsat 8 imagery for carbon stock estimation. *Forests*. https://www.mdpi.com/1999-4907/15/4/681

[49] Zhang, L., & Yang, Q. (2016). Estimating biomass in mangrove forests using satellite imagery and LiDAR data. *ISPRS Archives*. https://isprs-archives.copernicus.org/articles/XLI-B8/705/2016/isprs-archives-XLI-B8-705-2016.pdf

[50] Brown, C., & Green, D. (2024). High-precision carbon stock estimation using CNNs. *IEEE Xplore*. https://ieeexplore.ieee.org/document/10605503

[51] Li, H., & Thompson, B. (2024). Machine learning for forest classification and carbon stock estimation. *Forests*. https://www.mdpi.com/1999-4907/15/4/681

[52] Williams, J., & Patel, R. (2024). Mapping and monitoring carbon stocks with satellite observations. *Carbon Balance and Management*. https://cbmjournal.biomedcentral.com/articles/10.1186/1750-0680-4-2

[53] Chen, X., & Lin, Y. (2024). Assessing forest emissions reduction using satellite data fusion. *Remote Sensing*. https://www.mdpi.com/2072-4292/15/5/1410

[54] Zhang, L., & Chen, W. (2024). High-precision explicit forest carbon stock model with multi-source data. *ISPRS Archives*. https://isprs-archives.copernicus.org/articles/XLVIII-1-W2-2023/1831/2023

Appendices

Appendix A

System Testing

System testing is about testing whether the system is working as expected. In this appendix we will present the flow of executing websites from the giving input to the prediction of the wealth index. Figures consist of the steps of the Home page, where you see the boundaries of the forests in California. Users get to choose one forest from either the map or the sidebar, once selected the user can then select from the range of towers from where the ground truth data was reported to. Users can then select the datapoint in particular and see the statistics about the data point like Tree Height and Tree Cover for ground measured value and satellite value. Users can also see the Carbon value for ground truth which is calculated using a formula, satellite value which is calculated using a model and the last is this paper's models predicting the carbon in the area based on all the features present in the dataset. Users can then move on to further analysis of the datapoint where more detailed breakdown of the data point is present, like breakdown by plant type, breakdown by season and breakdown by daily, weekly and monthly. This allows the user to compare the tree metrics for that data point.

Figure A1

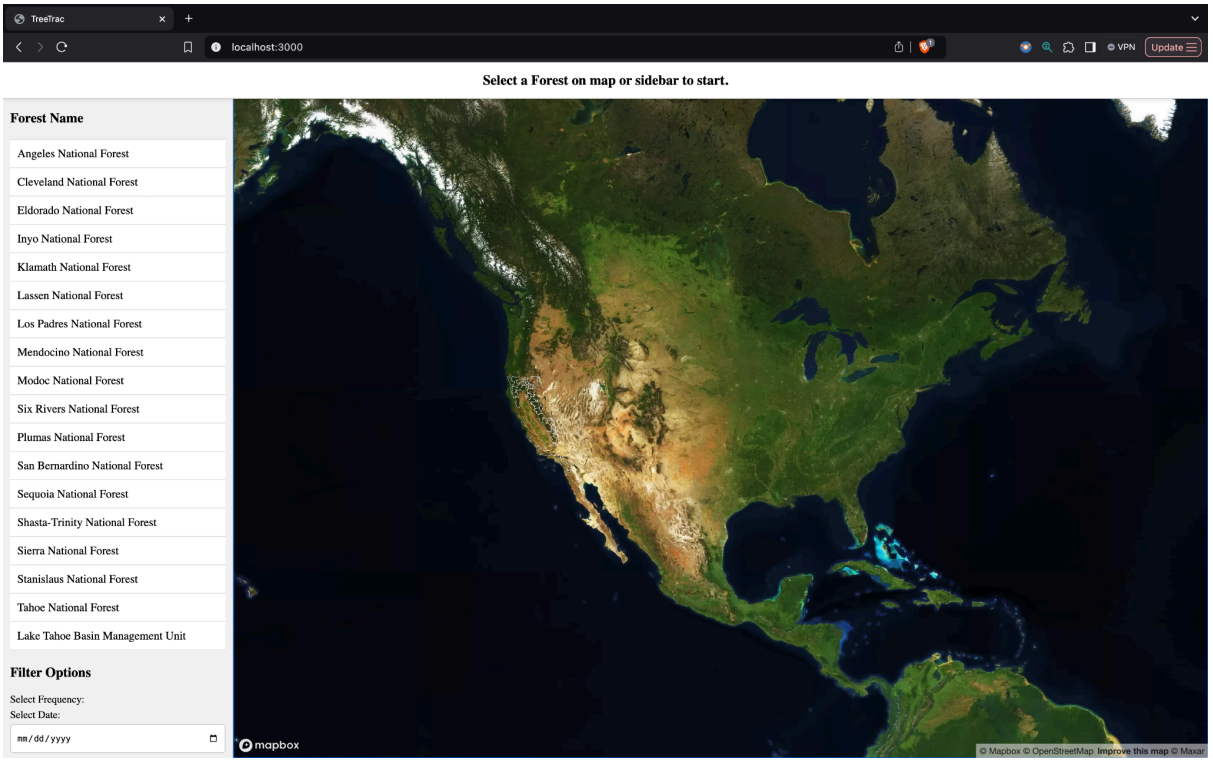


Figure A2

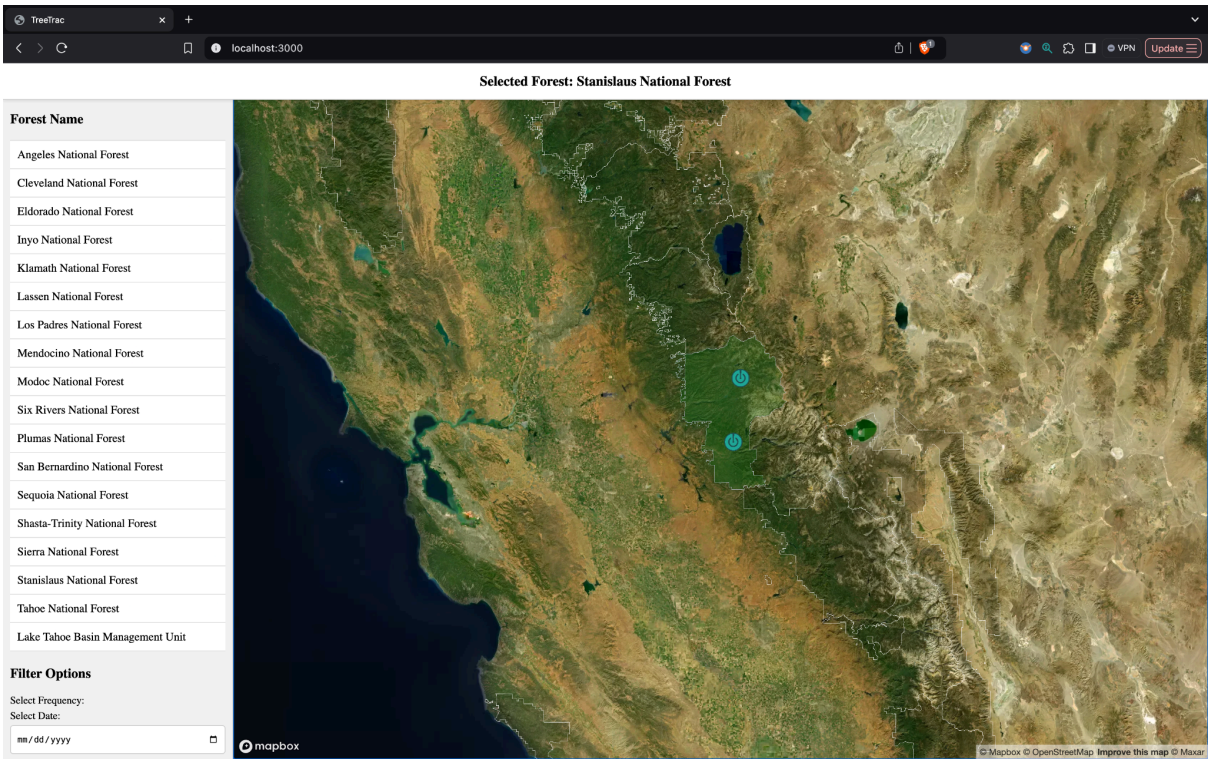


Figure A3

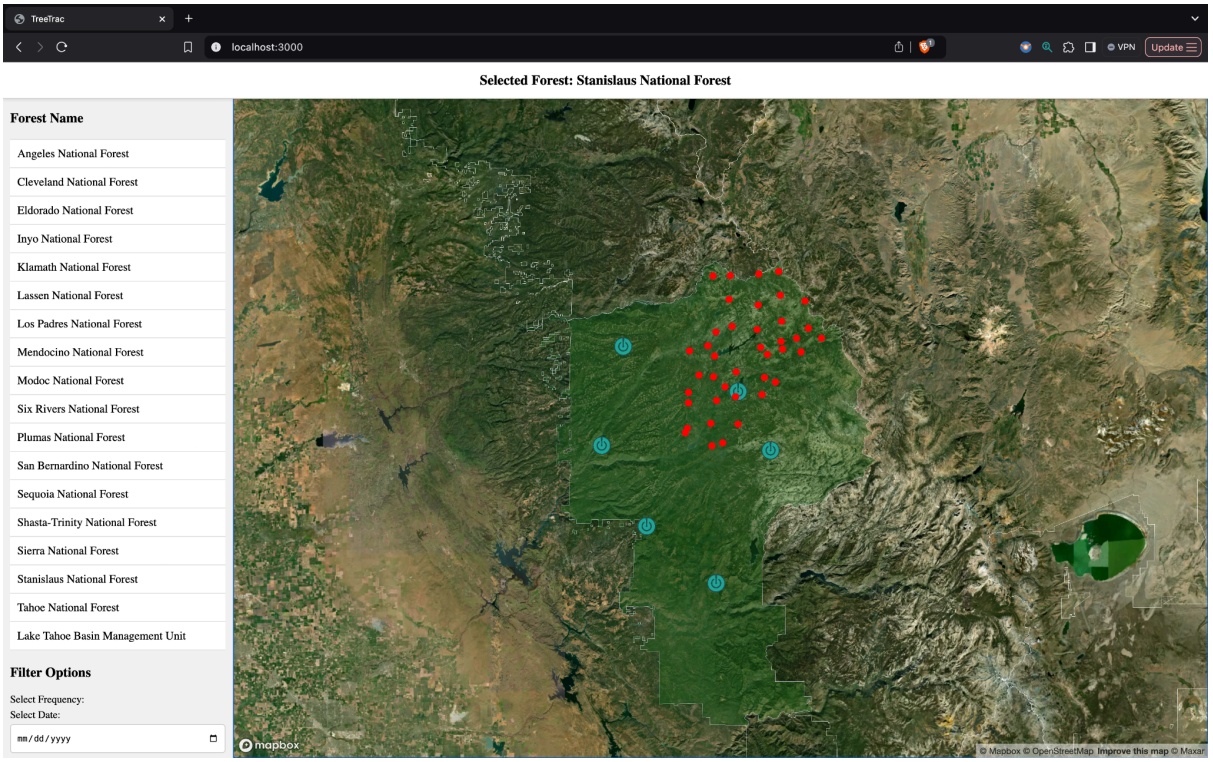


Figure A4

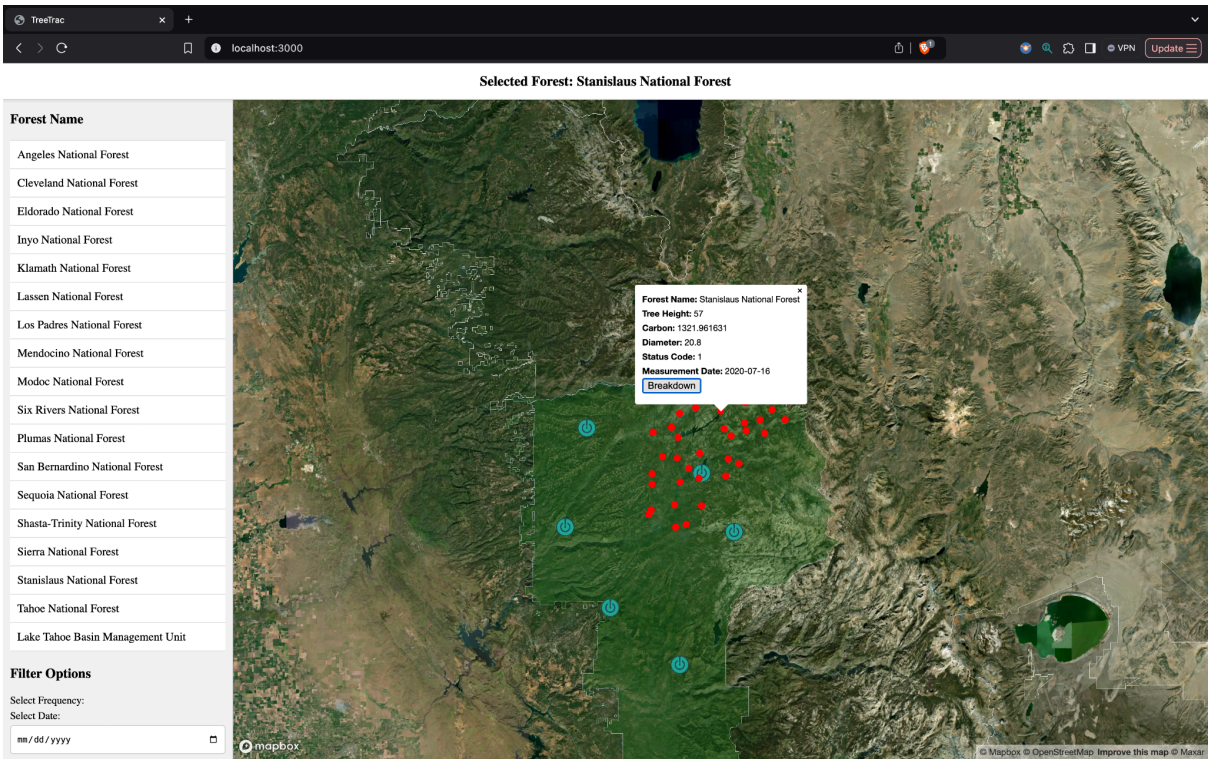


Figure A5

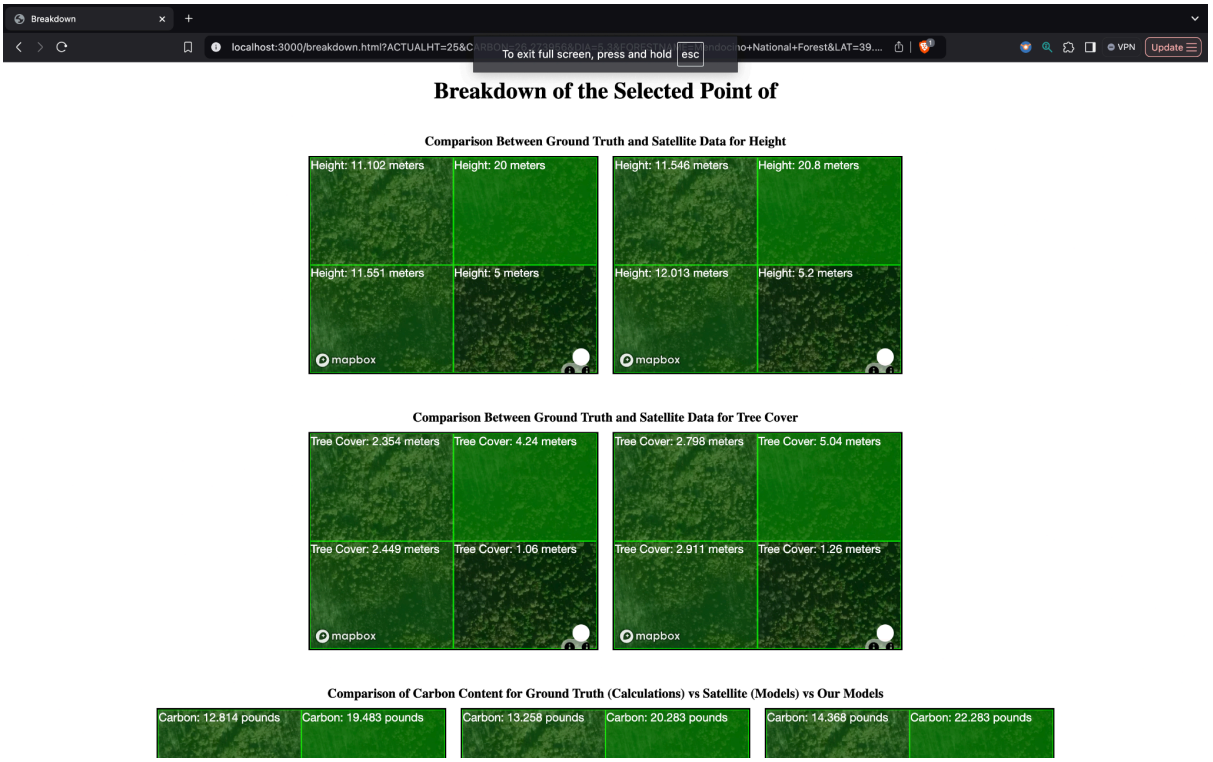


Figure A6

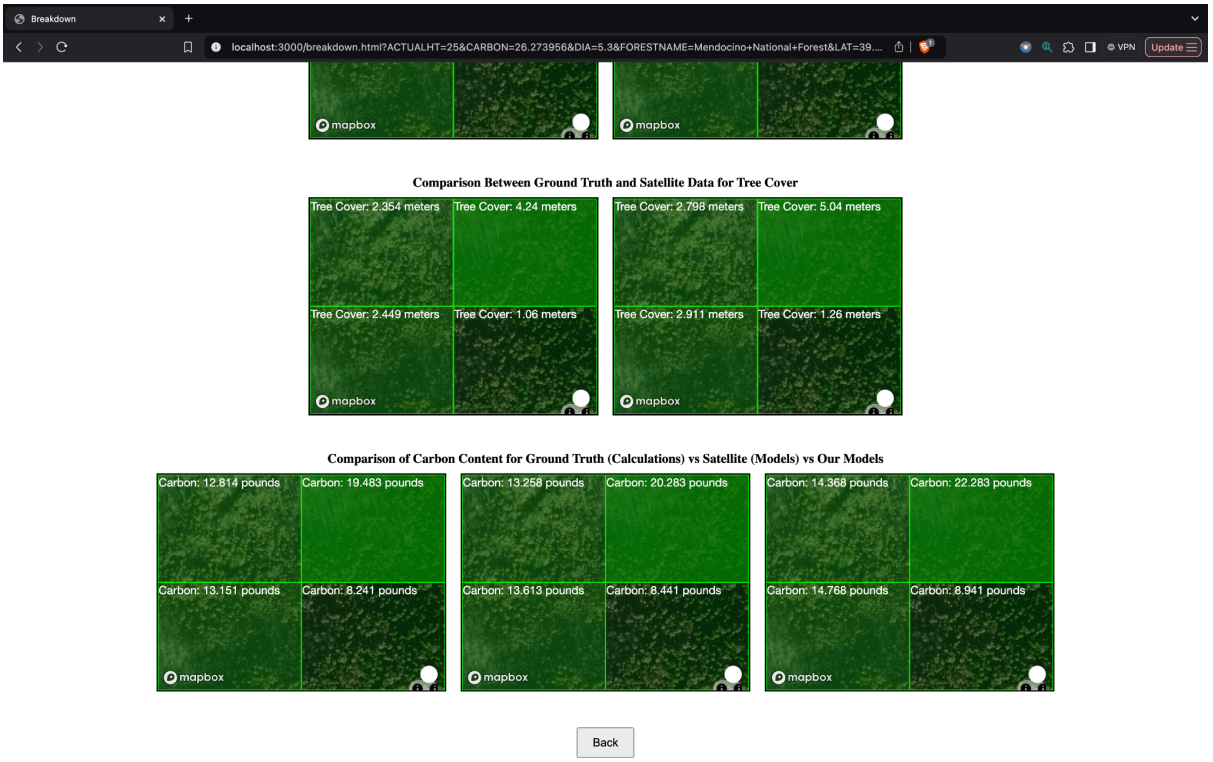


Figure A7

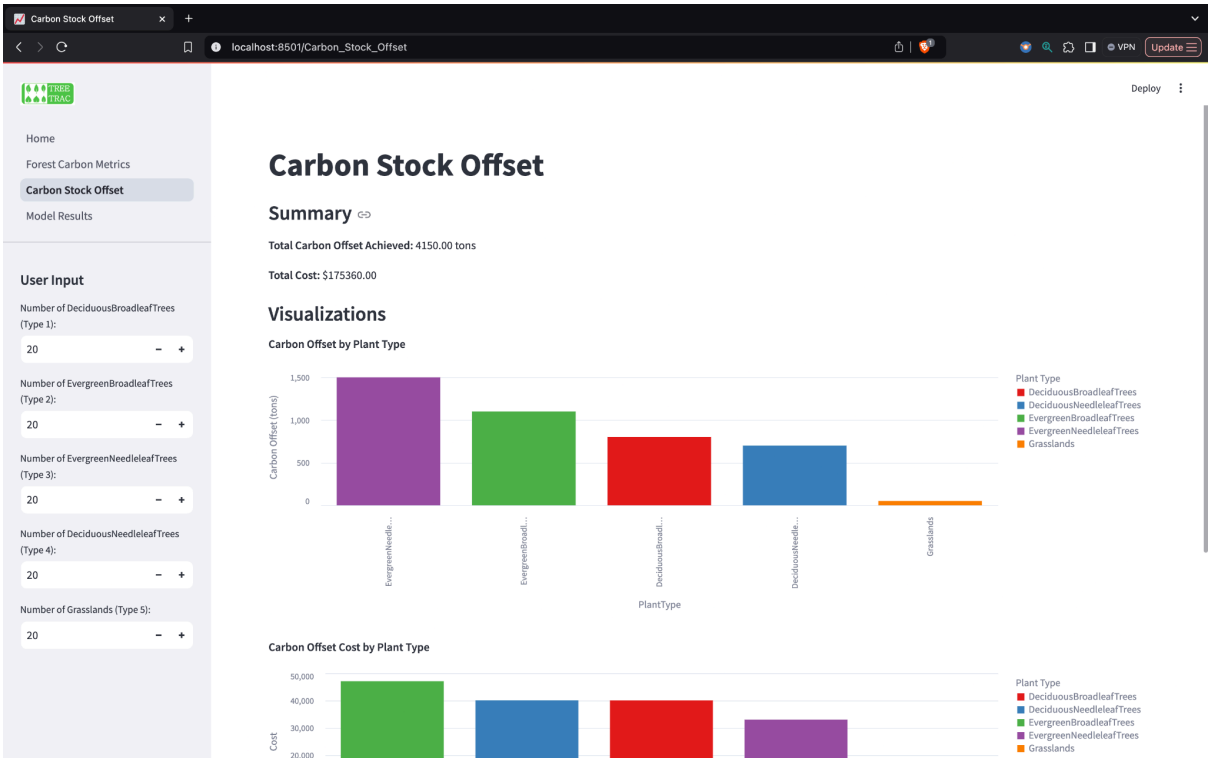


Figure A8

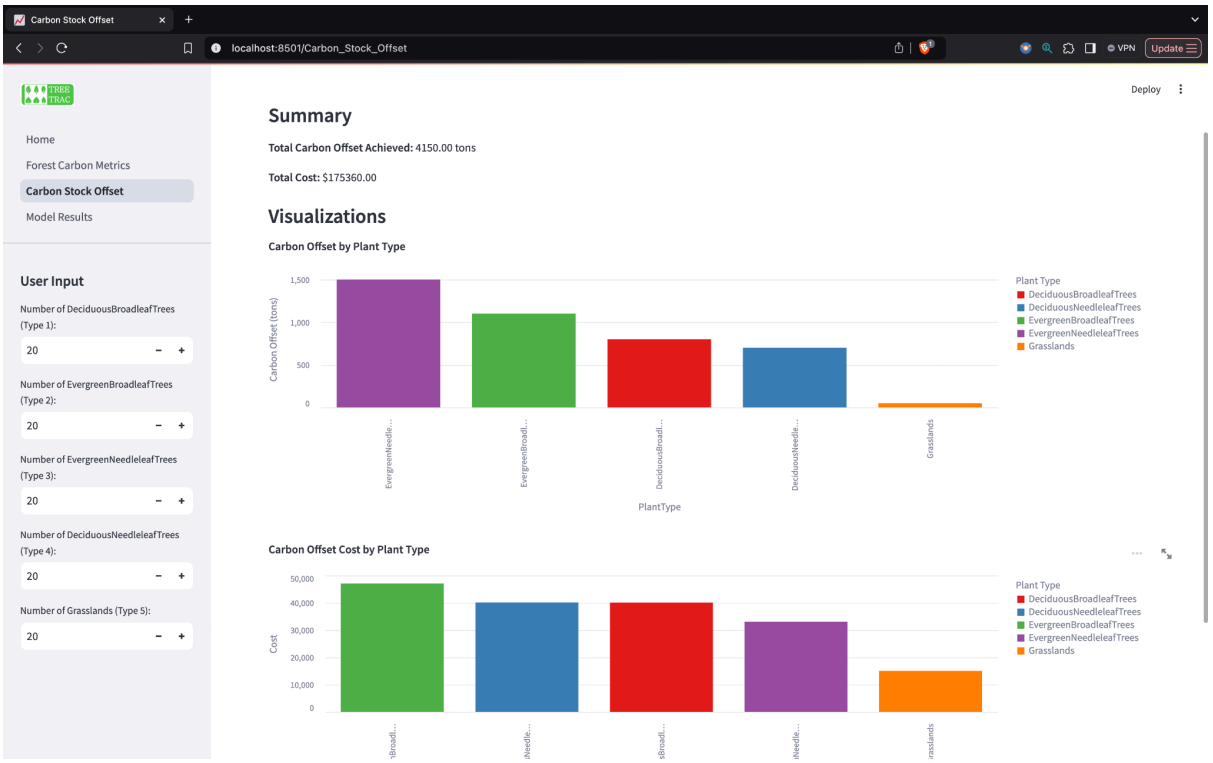


Figure A9

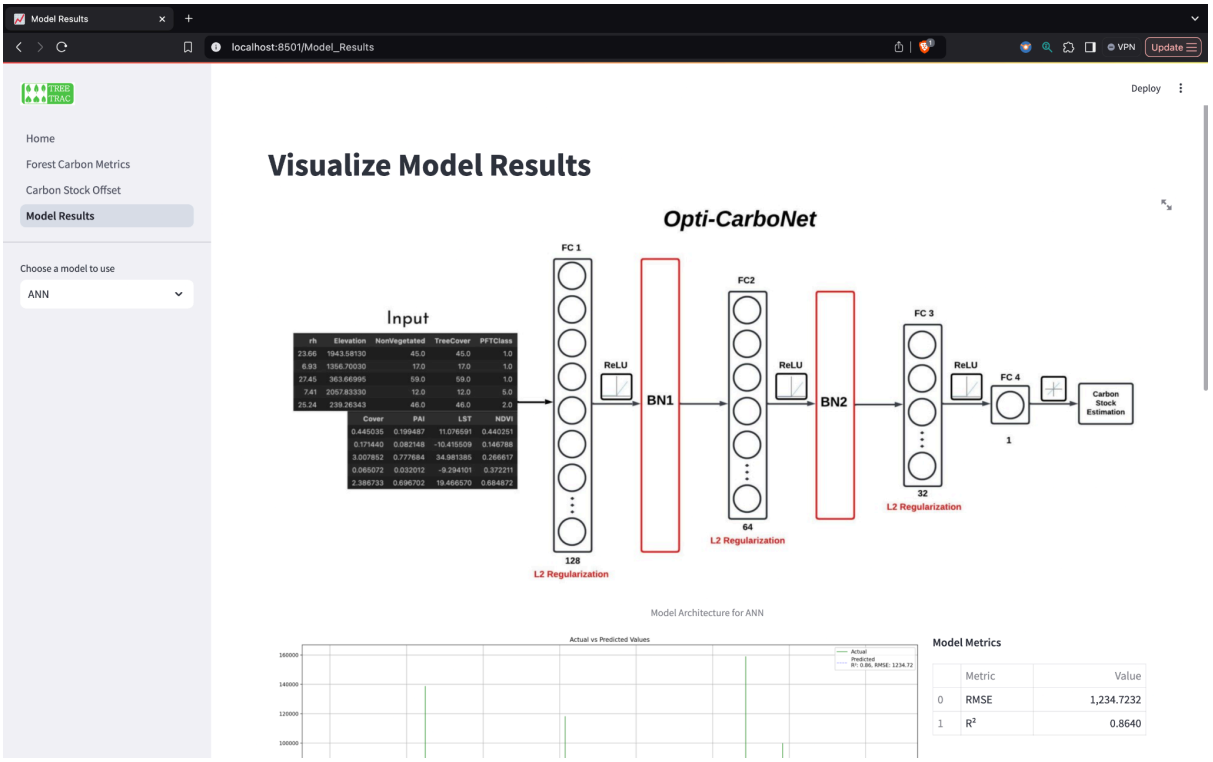


Figure A10

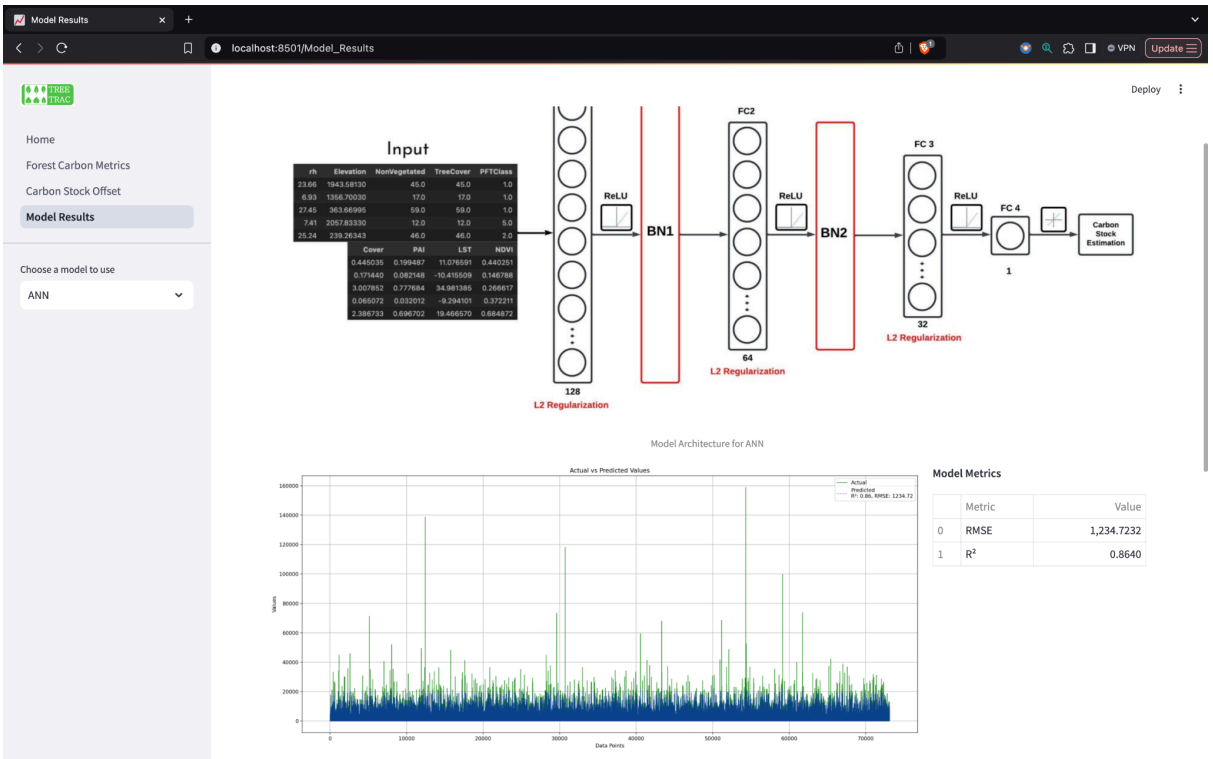


Figure A11

Figure A12

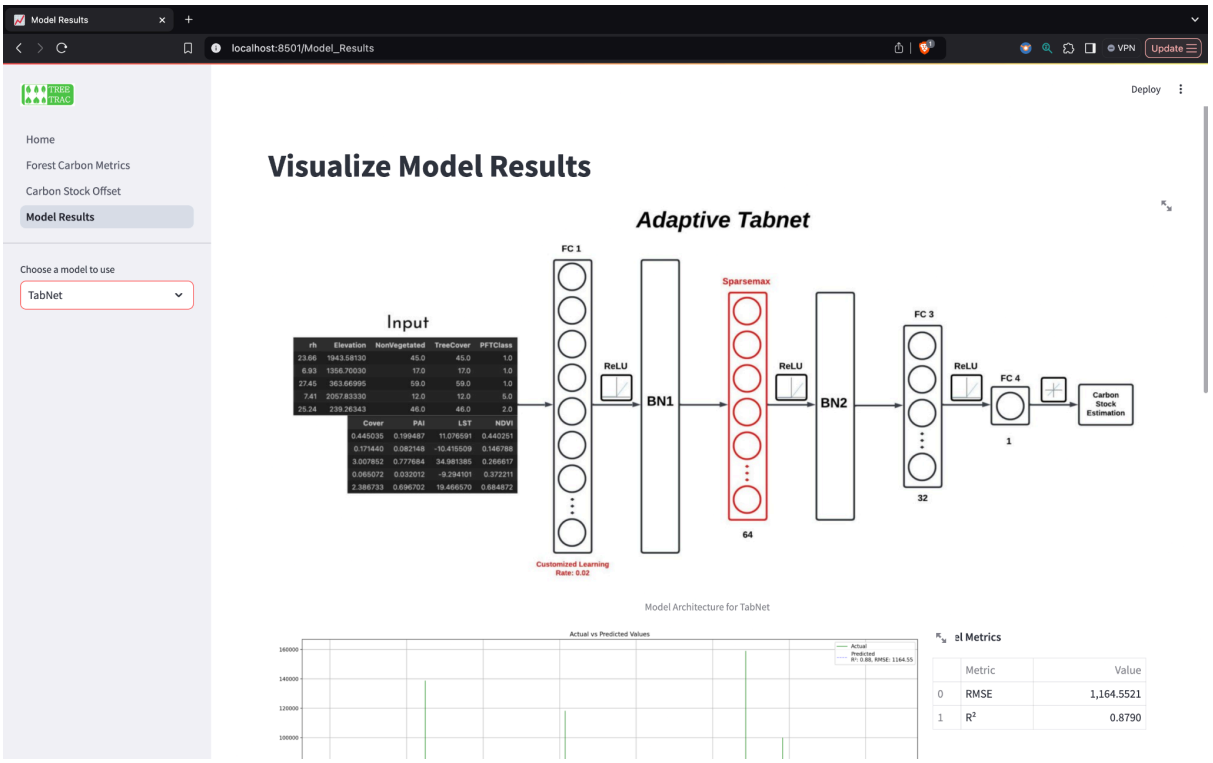


Figure A13

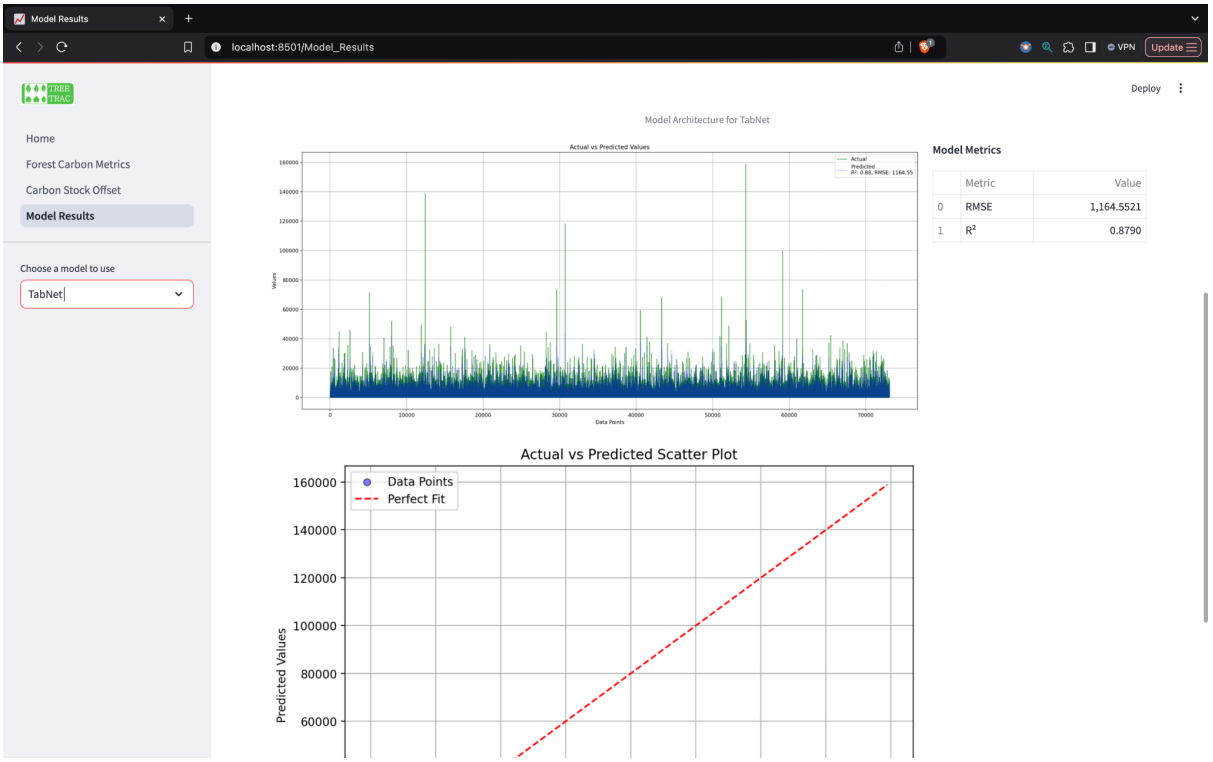


Figure A14

Figure A15

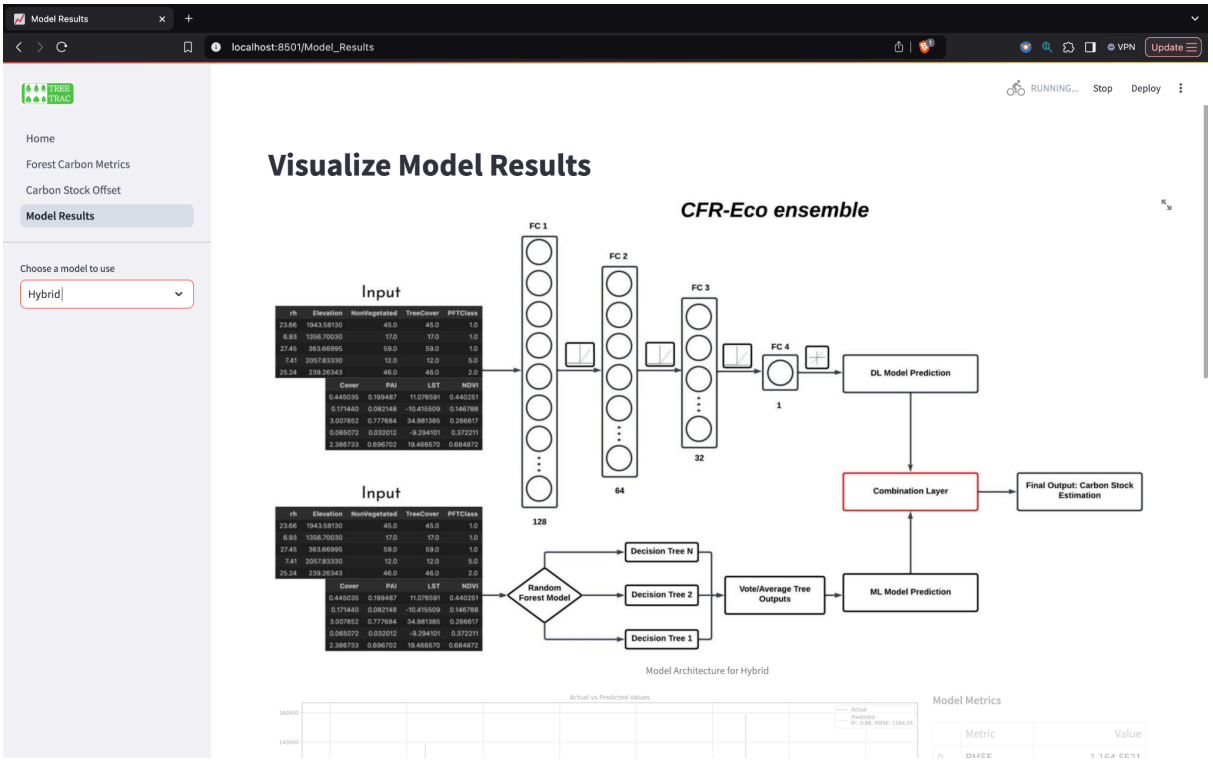


Figure A16

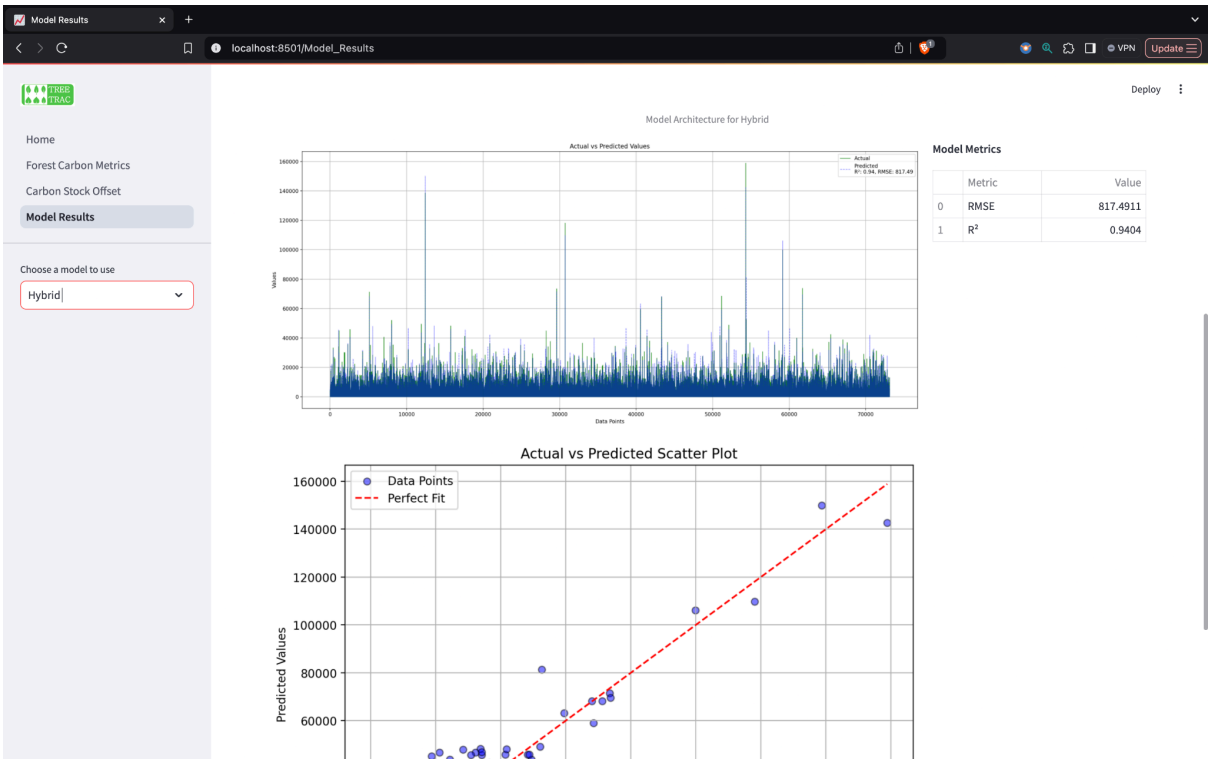
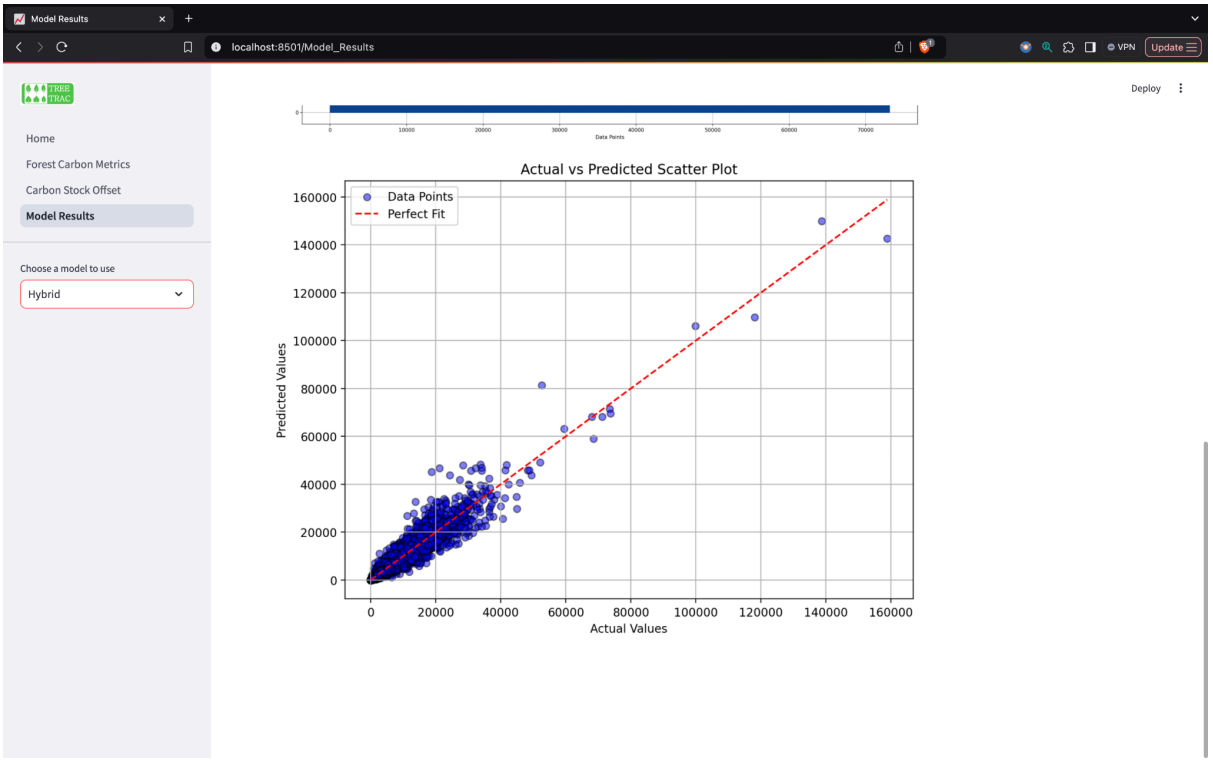


Figure A17



Appendix B - Project Data Source and Management Store

For the project we mainly used satellite images, which were downloaded using a python script from NASA's EarthData website by providing the shapes and boundary and the geographic coordinates of California. Link to download the day time satellite images for GEDI L2A, GEDI L2B, GEDI L3, GEDI L4A from Earthdata are:

https://lpdaac.usgs.gov/products/gedi02_av002/

https://lpdaac.usgs.gov/products/gedi02_bv002/

https://daac.ornl.gov/GEDI/guides/GEDI_L3_LandSurface_Metrics_V2.html

https://daac.ornl.gov/GEDI/guides/GEDI_L4A_AGB_Density.html

For storing all the data, we decided to use the google drive platform and created a shared drive which can be accessed by all the team members and it also helped us to write the code and store the intermediate files generated during the code execution.

The raw data collected from GEDI, MODIS and forest data is stored in the raw folder under Appendix B folder in the link given below.

The processed data where all the data is concatenated is stored under a processed folder which is around 10GB for two years worth of California data.

https://drive.google.com/drive/folders/1KNfrRIcpthhPY7iBmYvxG_AZoR_Lcqyu?usp=drive_link

Appendix C- Project Program Source Library, Presentation and Demonstration

The following links provide access to the source code for our UI, model, data processing, data collection, and data exploration. Additionally, they include project presentations and demonstration videos showcasing our work.

Project Program Source Code

https://drive.google.com/drive/folders/1vB_D2WQctDyeYbufqG7J7sYahubeZ9Ol?usp=drive_link

Presentation and Demo

https://drive.google.com/drive/folders/1EP7jNxnnRyQndAlRCmrc409c_oYImHHo?usp=drive_link

Workbooks

https://drive.google.com/drive/folders/1cIjhtmNQkbzJRidD0myv9lz88Z0kx4Bb?usp=drive_link

Reports

https://drive.google.com/drive/folders/15ZenBlfacTlO6fWQ-9tld1Nofapp53Zc?usp=drive_link