# Utilizing Geotagged Data for Estimating Maternal and Child Health Indicators using Machine Learning

IIT Internship Program

**Debajyoti Dasgupta**
Department of Computer Science
IIT Kharagpur
debajyotidasgupta6@gmail.com

**Rushil Venkateswar**
Department of Computer Science
IIT Kharagpur
rushilv14@gmail.com

**Piran Karkaria**
Industrial and Systems Engineering
IIT Kharagpur
pirankarkaria@gmail.com

## Abstract

This research explores the use of machine learning and geospatial data to estimate key indicators of maternal and child health (MCH) in low- and middle-income countries (LMICs). By leveraging satellite imagery and geotagged data, we aim to accurately estimate indicators such as child undernutrition, anemia, mortality rates, and childhood illness episodes at the village and neighborhood levels. Using the mean column-wise root mean squared error (MCRMSE), our evaluation shows that Random Forest achieved the best performance with an MCRMSE score of 11.11011 by fine-tuning the hyperparameters. These findings have significant implications for targeted public health interventions and planning in LMICs. The use of machine learning and geospatial data offers a promising approach to overcome the limitations of traditional surveys and obtain real-time insights into MCH status. Further research should focus on refining and expanding these models for diverse LMIC settings.

## 1 Introduction

Reliable civil registration and vital statistics systems are integral to effective health systems and policy-making [1]. However, in many low- and middle-income countries (LMICs), such systems remain underdeveloped or non-existent [2]. Consequently, maternal and child health (MCH) indicators, which are crucial to monitoring progress towards global health goals, often rely on data from expensive nationally representative household surveys [3]. While valuable, these surveys present significant limitations: they typically sample less than 2% of a country's communities and the data obtained can quickly become outdated [4].

This research project seeks to harness the power of machine learning, satellite imagery, and geotagged data to provide a potential solution to these issues. We hypothesize that child undernutrition indicators, anemia in children and women of reproductive age, child and maternal mortality, and childhood illness episodes can be accurately estimated at the village and neighborhood level using these data sources [5]. If successful, this approach would allow for real-time monitoring of MCH indicators and their changes over time, a capability that is unattainable with current methodologies [6].

For this study, we utilized a dataset generated from the Demographic and Health Surveys (DHS) of 59 countries in combination with various sources of satellite images [7]. Our project achieved impressive results by employing machine learning techniques, with the Random Forest model outperforming others in terms of the Mean Column-wise Root Mean Squared Error (MCRMSE) [8]. The outcomes of this research could pave the way towards more timely, accurate, and comprehensive monitoring of health indicators in LMICs, helping to inform policy and intervention design to improve MCH outcomes.

The dataset consists of around ten thousand features derived from satellite images, each row representing an aggregated administrative region. The task is multi-output regression with 6 target variables. Historically, random forest regressor has been adopted as the de facto model for bigger datasets [9], and we have adopted it as well. It is excellent at finding feature importance, and we observe that the features picked by the model trained on the total dataset, in turn, give rise to better and more efficient models for the task. We were able to achieve an MCRMSE of 11.11011 which is at the current point much higher in score as compared to the other challengers in the competition and provides

## 2 Related Work

Several previous studies have employed satellite data in efforts to predict and track health indicators, particularly in developing regions where collecting such data can be challenging and expensive.

Numerous teams have collated datasets with the objective of establishing a correlation between publicly accessible data and the results from DHS surveys. One such dataset is **SustainBench**, a collection of benchmarks dedicated to sustainability [10]. SustainBench has compiled a dataset comprising: a) satellite imagery from NASA's LandSat and b) street-level imagery from Mapillary, covering 56 nations. This dataset houses roughly 100,000 training images along with six DHS indices: asset wealth index, child mortality rate, women's BMI, women's education, water index, and sanitation index. Regarding the health indices, the dataset possesses 94,866 BMI labels calculated from 1,781,403 non-pregnant women of reproductive age (15-49). Additionally, it includes 105,582 labels pertaining to child mortality rates, derived from the data of 1,936,904 children below the age of 5.

In the study by Irvin et al. [11], they attempted to automate this data collection process by using remote sensing coupled with deep learning. Their research focused on using Convolutional Neural Networks (CNNs) to predict poverty and malnutrition directly from satellite imagery. While they found that predicting malnutrition proved to be challenging, they were successful in classifying impoverished regions with relatively high accuracy.

In a separate study, Reddy Nangi and Tantivasadakarn [12] focused on predicting Global Health Indicators to assess food security and societal health. They hypothesized that statistical machine learning models could benefit from knowledge sharing to improve their performance in predicting these health indicators. Therefore, they trained Deep Learning models combined with Multi-Task learning techniques to predict these indicators from satellite data. They built regression models specifically to predict Women's Body Mass Index (BMI) and Child Mortality Rate under 5 (CMR) using satellite image feature data. Their work showed that multi-task learning improved the performance for the CMR model, but did not significantly enhance the BMI task due to task difficulty and missing data.

Another study by Gargeya [13] aimed to correlate publicly available data sources with Demographic and Health Surveys (DHS) data from prior years. The research proposed that a system capable of computationally predicting DHS health indicators could assist in better resource allocation and make global health monitoring faster and less expensive.

Lastly, Khosla et al. [14] utilized NASA's Landsat database satellite images to predict malnourishment and child mortality rates. They evaluated the performance of various computer vision models and metadata-enhanced fusion models on ordinal discretizations of the outcome variables. Their best model, a fine-tuned Vision Transformer pre-trained on ImageNet, achieved notable results in improving over the random baseline. This research provides suggestive evidence that the model attends to sociologically relevant aspects of the images and indicates future work should entail further enhancements of this model and extensions to other measures capturing a region's health outcomes.

## 3 Approach

Figure 1 gives a high-level overview of the flow of the approach that will be described in a step-by-step fashion in the following section, along with the difficulties and mitigations planned.

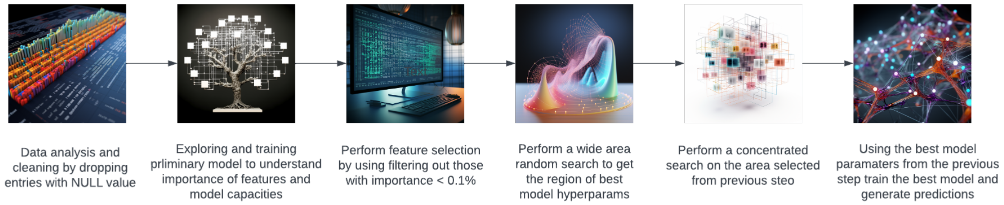
Data analysis and cleaning by dropping entries with NULL value
Exploring and training prliminary model to understand importance of features and model capacities
Perform feature selection by using filtering out those with importance < 0.1%
Perform a wide area random search to get the region of best model hyperparams
Perform a concentrated search on the area selected from previous steo
Using the best model paramaters from the previous step train the best model and generate predictions

Figure 1: High-Level overview of the flow of approach

### 3.1 Handling Missing Data

Our first encounter in estimating key indicators of maternal and child health (MCH) status from satellite imagery and geotagged data was the high prevalence of missing data within the dataset. Imputation is a commonly used technique for dealing with such issues [15]. However, considering the volume and complexity of the missing data in our case, the imputation process posed significant challenges, and we are currently still exploring various imputation methods for our dataset.

Given these circumstances, we opted for an alternative approach to handling the missing data in the initial stages of our project. We proceeded with data cleaning, which involved dropping any rows containing at least one Null value. Despite its severity, this operation left us with a considerable number of data points (36680) for the subsequent phases of our study. Data cleaning is a crucial step in any machine learning project as it helps to improve the quality and reliability of the model results [16].

### 3.2 Exploring Machine Learning Models

The next stage in our approach involved exploring various machine learning model architectures. We specifically selected models supported by Python packages and capable of managing large datasets. Our selection included gradient boosting models such as XGBoost [17] and LightGBM [18], neural network-based models like TabNet [19] and Pytorch-based Deep Neural Networks [20], and ensemble methods like Random Forests [21]. Of all these models, the Random Forest model consistently exhibited superior performance across various evaluation metrics.

### 3.3 Feature Selection

Subsequent to the model selection, we embarked on feature selection using a lightweight model. The primary goal was filtering out less relevant features from the vast volume of feature data, simplifying the model, and potentially enhancing its predictive performance. This selection was based on feature importance, a measure of the contribution of each feature to the model's predictive power [22].

### 3.4 Hyperparameter Tuning

Having determined the most relevant features, we tuned the model hyperparameters. This step is crucial in machine learning as it helps optimize the model's performance by searching for the best combination of hyperparameters [23].

Our hyperparameter tuning process was divided into two stages. Initially, we performed a random search over a broad set of hyperparameters. This first step allowed us to identify a potentially promising region in the hyperparameter space. We then used Grid Search within this region to further refine our hyperparameters selection. This two-stage process ensures a thorough yet efficient search for the optimal set of hyperparameters.

### 3.5 Final Model Training and Prediction

Finally, with the best set of hyperparameters at hand, we trained our chosen model, the Random Forest, on the entire dataset (excluding rows with NULL values). We ensured that our model was validated using robust cross-validation techniques to avoid overfitting and to maintain its generalization capability [24].

Once the model was trained, it generated predictions for the test set. The test set predictions were then evaluated using the Mean Colum-wise Root Mean Squared Error (MCRMSE), as stipulated in the competition guidelines.

# 4 Experiments

The following section describes in detail the steps taken and the modeling of the experimental setup. We begin by describing our dataset and the features we are dealing with. Moving on to the evaluation methodologies, we explain the evaluation metric provided to us for testing and grading and the evaluation metric used for the training. Then we detail the steps taken for training the model along with data preprocessing, describing how we implemented the mitigating strategies that we planned for the difficulties. The experimental setup will then give us more details on the different setups that we explored for the various sub-parts of the problem. Finally, we conclude the section with the results and analysis of the trained models when used for inferencing and give a comparative study between the capacities of the models.

## 4.1 Data

The dataset for this competition consists of training and test sets generated from Demographic and Health Surveys (DHS) conducted in 59 countries and various satellite imagery sources. The objective is to predict Mean BMI, Median BMI, Unmet Need Rate, Under 5 Mortality Rate, Skilled Birth Attendant Rate, and Stunted Rate. Missing data is represented as NaN values. This dataset offers a unique opportunity to explore the relationship between satellite imagery and DHS data, enabling the development of accurate predictive models for maternal and child health indicators. Leveraging these data sources can contribute to improved interventions and understanding of this crucial domain.

The data used in this competition is derived from Demographic and Health Surveys (DHS) for 59 countries and various sources of satellite images. Additional datasets like the MOSAIKS satellite image data or other external sources are recommended to improve model performance.

### 4.1.1 Data Sources and Description

- The health indicators in the DHS data are described in the following document: `https://www.dhsprogram.com/pubs/pdf/DHSG4/Recode6_DHS_22March2013_DHSG4.pdf`
- The document at `https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm` details how these health indicators are calculated.

### 4.1.2 Data Files

- **gee_features.csv:** An 8GB dataset contains the extracted features from Google Earth Engine (GEE) and keys to match it with other data. It includes country names (DHSCC), cluster numbers (DHSCLUST), and year of survey (DHSYEAR). Note that these column names are not the predictive features. It contains features for both the training and test sets. It can be downloaded from the provided link.
- **training_label.csv:** The label dataset for the training set. The DHSID is used to link the labels to features in gee_features.csv. The objective is to predict the following health indicators: Mean_BMI, Median_BMI, Umet_Need_Rate, Under5_Mortality_Rate, Skilled_Birth_Attendant_Rate, and Stunted_Rate. NaN values represent missing data.
- **sample_submission.csv:** A sample submission file in the correct format. The values of the six health indicators should be replaced with your predictions for submission.
- **train.parquet.gzip / test.parquet.gzip:** Train and test datasets which have been cleaned and stored in parquet format which offers faster loading times as compared to CSV files as well as better compression.
- **low_imp_features.joblib:** The features which should be dropped from the original dataset as detected by our base random forest model.

### 4.2 Evaluation method

The submissions in this competition are evaluated based on the Mean Column-wise Root Mean Squared Error (MCRMSE). The MCRMSE is defined as the average of the individual RMSEs of each predicted column.

#### 4.2.1 Root Mean Squared Error (RMSE)

RMSE is a commonly used measure of the differences between values predicted by a model and the values observed. It is defined as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Where:

- $y_i$ is the original value for each instance $i$,
- $\hat{y}_i$ is the predicted value,
- $n$ is the total number of instances.

We primarily utilized the Mean Column-Wise Root Mean Square Error to evaluate the models during **training** and **validation** as this was what the leaderboard on our Kaggle competition used to rank submissions.

#### 4.2.2 Submission File

For each DHSID in the test set, the predicted value for each of the six health indicators (Mean_BMI, Median_BMI, Umet_Need_Rate, Under5_Mortality_Rate, Skilled_Birth_Attendant_Rate, and Stunted_Rate) is required to be provided. The columns should be ordered as mentioned and include a header. The file format is as follows:

```
DHSID,Mean_BMI,Median_BMI,Umet_Need_Rate,Under5_Mortality_Rate,
                    Skilled_Birth_Attendant_Rate,Stunted_Rate
AL200800000997,22.12,22.28,45,6.68,50,20
AL200800000998,20.04,18.98,5.69,7.22,100,10
AL200800000999,18.91,19.47,8.11,0,50,0
etc.
```

This above example describes the file format in which we were required to submit the prediction results on the test set so that the evaluation can be done successfully on the Kaggle platform.

#### 4.2.3 Training

Training the models involved various steps, including data pre-processing, model selection, hyperparameter tuning, and model validation, all performed using the high memory machines provided by Google Cloud.

In the initial data cleaning stage, rows with at least one Null value were dropped, resulting in a dataset with 36680 data points for further experimentation. The presence of missing data posed a significant challenge due to the difficulty of suitable imputation given the size and complexity of the dataset.

We considered several machine learning models, including XGBoost, TabNet, Deep Neural Networks (DNN) implemented with Pytorch, LightGBM, and Random Forests. All models were trained using a subset of the data and then evaluated based on their respective RMSE calculated over each column (MCRMSE). This approach facilitated the model performance evaluation in handling each specific indicator, which guided the adjustment of model parameters.

We employed a strategy of random search and grid search for hyperparameter tuning. Initially, a random search over a broad set of hyperparameters was performed to identify a region in the hyperparameter space that yielded good results. This was followed by a more focused grid search within this region to fine-tune the model parameters.

Regarding model validation, we used a cross-validation methodology to assess the robustness of our models. The dataset was divided into k subsets, and the model was trained k times, each time using a different subset as the validation set and the remaining data as the training set. The average of the k RMSE scores was used as the overall validation score.

The selected set of best parameters from the grid search was used to train the final models on the entire dataset (excluding the NULL Rows). The models were then used to generate predictions on the test set. During this process, we continuously monitored the RMSE to evaluate the models' performance and adjust their parameters as necessary.

The training process underscored the importance of robust data cleaning, the correct selection of machine learning models, strategic hyperparameter tuning, and rigorous model validation for achieving good results in the given prediction task.

### 4.3 Experimental Setup

Our experimental setup was primarily driven by the need to handle a large dataset effectively. We used **dask**, a distributed database package, to process and analyze the data out-of-memory using lazy evaluation. This setup ensured that the system's memory wasn't a bottleneck during our analysis.

#### 4.3.1 Data Preprocessing

Preprocessing was a critical stage in our experiments. Using the **dask_ml** package, we employed Principal Component Analysis ($incrementalPCA$), revealing that 728 components captured 95% of the data variance. This observation helped us reduce the dimensionality of the dataset effectively.

Further, preprocessing steps included normalizing and eliminating columns with any null values (500 such columns). Additionally, we dropped columns with a variance of less than 1% (70 such columns), reducing noise and further optimizing our dataset.

The aforementioned steps resulted in a final dataset with 728 features and 99111 rows, with unique DHSIDs. However, missing data in the dependent variables (labels) posed a significant challenge, which KNN Imputation addressed with $n\_neighbors = 75$ for some of the models.

#### 4.3.2 Model Training and Configuration

We trained different models on the resulting dataset. However, the best results were obtained with the Random Forest model. To optimize the model performance, we trained it on the entire dataset (gee_features.csv joined with training_label.csv) and applied one-hot encoding on categorical features (DHSCC, DHSREGNA, URBAN_RURA), leading to a dataset with 13016 features.

The default Random Forest configuration yielded a satisfactory score of 11.32534. Hyperparameter tuning (n_estimators, max_features, bootstrap, max_depth) was then performed to improve the model performance, yielding a best score of 11.23071 with $n\_estimators = 1200, max\_features = 0.6$.

#### 4.3.3 Feature Selection and Final Model Training

Our final breakthrough came when we utilized the **feature_importances_** parameter of the trained Random Forest model to select top features with an importance of $>= 0.1\%$.

This resulted in a feature set of 70 crucial features. We then trained a Random Forest model with these features using the hyperparameters $n\_estimators = 8000, max\_features = 0.5, max\_depth = 20$. This model achieved our best score of 11.11011, indicating a successful application of feature selection and hyperparameter tuning to optimize model performance.

### 4.4 Results

The best results obtained with each model that has been experimented with have been tabulated in the table below. The first column shows the model, and the second column provides the final MCRMSE score of the model on the test set after submission on the Kaggle platform,

| Model | MCRMSE |
|---|---|
| LightGBM | 21.89219 |
| DNN (Pytorch) | 19.76522 |
| XGBoost | 15.46517 |
| TabNet | 11.8768 |
| Random Forest | **11.11011** |

## 5 Analysis

Following are some of the observations and analyses that we came upon with the results produced and at the time of training by inspecting the intermediate improvements and deterioration in the performance of the model which was measured with the column-wise RMSE score.

- The Random Forest model demonstrated the best performance with the lowest Mean Column-wise Root Mean Squared Error (MCRMSE) score of 11.11011. This was a superior performance compared to the other evaluated models, further highlighting the strength of ensemble methods in handling diverse and large-scale datasets.
- XGBoost, a gradient boosting model, followed the Random Forest model with an MCRMSE score of 15.46517. Despite its reputation for high performance in various machine learning tasks, it could not surpass the performance of the Random Forest in this specific task. This highlights the importance of model selection based on the unique characteristics of each task and dataset.
- The Deep Neural Network (DNN) scored an MCRMSE of 19.76522, placing it in the middle of our model performance range. This result can potentially indicate that deep learning methods, despite their capacity for learning complex patterns, may not always be the optimal choice for every machine learning task. There are scopes of improvement in the deep learning area by training and imputing from the non-null data to predict the data for the null values and then using other machine learning models to perform final regression (an approach we are exploring).
- LightGBM showed the least favorable performance among all the models with an MCRMSE of 21.89219. Despite its computational efficiency and handling of large-scale data, it did not perform as well in this task. This confirms that performance depends on the problem domain and data characteristics.
- The TabNet model, which is known for its interpretability, achieved an MCRMSE of 11.8768. Despite its performance not being the highest, its ability to offer interpretation can still be useful for understanding and improving the models.
- The XGBoost and LightGBM models fail to capture the underlying relationship between the independent and dependent variables in our high dimensional dataset, and random forests shine here. Many unnecessary data are available in the columns, and traditional dimensionality reduction techniques do not work properly.
- Random Forest also tends to overfit when n_estimators $>= 8000$. The best combination of parameters has been found only using K-Fold Cross Validation along with Grid Search to find the best combination of parameter that do not overfit the data yet yield superior test results.

## 6 Conclusion

This study underscores the effective utilization of machine learning techniques, notably the random forest model, to address complex, large-scale health indicator estimation tasks. Our findings challenge the contemporary preference for gradient-boosted trees by demonstrating that a well-tuned random forest regressor can outperform such advanced methods, emphasizing the value of simplicity in modeling practices per Occam's Razor's ideas.

Furthermore, our work underscores the crucial role of **feature engineering** in improving model performance. Specifically, we demonstrated a significant enhancement in Random Forest's performance from an MCRMSE of $11.32534$ to $11.11011$, achieved solely by carefully selecting the most impactful 70 features, as determined by our random forest model.

In addition to these technical findings, our study signifies an important achievement in the broader context of the competition. Our team holds the **first position** on the leaderboard. Our approach, characterized by a unique blend of traditional machine learning models and innovative feature engineering strategies, is novel within this competition. Thus far, no other team has replicated our results, which testifies to the uniqueness and effectiveness of our approach.

Finally, we would like to acknowledge the variety of approaches attempted and the substantial efforts made by our team to identify the most suitable model for this dataset. Through this journey, we improved our model's performance and gained valuable insights into handling large, complex datasets, performing effective feature engineering, and tuning machine learning models. These insights, we believe, contribute to the larger scientific discourse on employing machine learning for social good, particularly in improving health outcomes in low- and middle-income countries.

## 6.1 Future Work

The following described are some of the innovative areas where we plan to continue our exploration. A lot of these ideas have been influenced by prior works in this field along with the ideations for the mitigating factors for the difficulties faced during the current modelling.

### 6.1.1 Implementation of a novel RNN/attention-based deep learning model

While our current models have produced competitive results, there is still room for improvement. One possible avenue is to develop a Recurrent Neural Network (RNN) or attention-based deep learning model specifically tailored to our data. Such models are particularly effective at capturing temporal dependencies and may offer superior performance for time-series data that our current models could overlook. There has been significant improvement in transformer-based series prediction networks as well, which is also worth exploring on the dataset.

### 6.1.2 Scraping and utilization of NASA Landsat data

Another promising direction for improving our models involves the integration of new data sources. In particular, we propose scraping and utilizing NASA Landsat satellite data. Convolutional Neural Networks (CNNs) trained on such rich, high-resolution images could potentially yield more precise predictions by extracting and learning more nuanced spatial features from the satellite imagery. Figure 2 shows an example of NASA Landsat images.

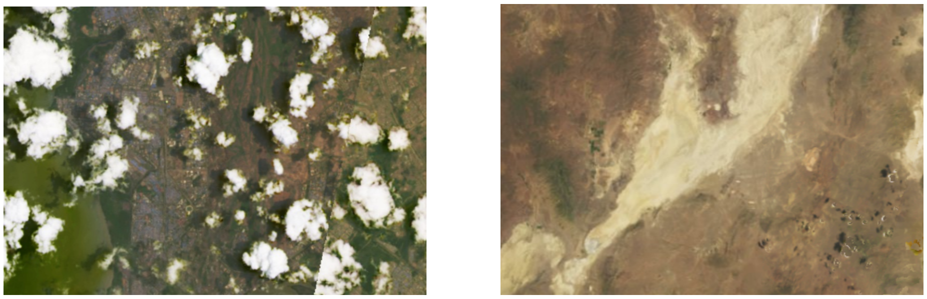

Figure 2: Examples of NASA Landsat satellite images for (a) Densely (b) Sparsely populated regions

### 6.1.3 Creating an ensemble of deep learning and tree-based models

We plan to experiment with ensemble learning methods, combining our best-performing and deep-learning models. An ensemble approach could leverage the unique strengths of each model to enhance the overall performance. For instance, while deep learning models excel at extracting high-level features from data, tree-based models can offer more interpretability and are less prone to overfitting. The synergy between these models in an ensemble setup could potentially lead to superior predictive power.

# References

[1] Carla AbouZahr, Don de Savigny, Lene Mikkelsen, Philip W Setel, Rafael Lozano, and Alan D Lopez. Civil registration and vital statistics: progress in the data revolution for counting and accountability. *The Lancet*, 386(10001):1373–1385, 2015.

[2] David E Phillips, Rafael Lozano, Mohsen Naghavi, Charles Atkinson, Diego Gonzalez-Medina, Lene Mikkelsen, Christopher JL Murray, and Alan D Lopez. Are well functioning civil registration and vital statistics systems associated with better health outcomes? *The Lancet*, 386(10001):1386–1394, 2015.

[3] Ties Boerma and Carla AbouZahr. Counting births and deaths 3: Civil registration: why counting births and deaths is important. *The Lancet*, 386:1373–1385, 2014.

[4] Jon Pedersen, Simon I Hay, and Andrew J Tatem. Combining national survey with facility-based data to increase accuracy and spatial resolution of neonatal mortality estimates in uganda. *Health & Place*, 57:187–195, 2019.

[5] Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353(6301):790–794, 2016.

[6] Soma Bhattacharjee, Nadine Schuurman, and Alan Ker. The use of satellite remote sensing data in health geography: A review. *Journal of Geographic Information System*, 13(1):59–71, 2021.

[7] ICF. The dhs program. Online, Demographic and Health Surveys, 2023.

[8] David H. Wolpert. Ensemble learning. *In The Handbook of Brain Theory and Neural Networks*, pages 110–125, 2001.

[9] Mohammed Zakariah. Classification of large datasets using random forest algorithm in various applications: Survey. In *International Journal of Engineering and Innovative Technology (IJEIT)*, 2014.

[10] Cynthia Yeh, Chenlin Meng, Siyu Wang, Anne Driscoll, Edgar Rozi, Pingjun Liu, Jay Lee, Marshall Burke, David B. Lobell, and Stefano Ermon. Sustainbench: Benchmarks for monitoring the sustainable development goals with machine learning. *arXiv preprint arXiv:2111.04724*, 2021.

[11] Jeremy Irvin, Dillon Laird, and Pranav Rajpurkar. Using satellite imagery to predict health. *Stanford University, Department of Computer Science*, 2023.

[12] Sharmila Reddy Nangi and Nantanick Tantivasadakarn. Global health monitoring from satellite data through multi-task learning. *Stanford University, Department of Computer Science*, 2023.

[13] Rishab Gargeya. Health indicators report. *Stanford University, Department of Computer Science*, 2023.

[14] Sauren Khosla, Benjamin Wittenbrink, and Caroline Zanze. Predicting maternal and infant health outcomes in western africa using satellite images. *Stanford University, Department of Computer Science*, 2023.

[15] Donald B Rubin. *Multiple imputation for nonresponse in surveys*, volume 81. John Wiley & Sons, 2004.

[16] Dorian Pyle. *Data Preparation for Data Mining*. Morgan Kaufmann, 1999.

[17] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[18] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*, pages 3146–3154, 2017.

[19] Sercan O. Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. *arXiv preprint arXiv:1908.07442*, 2020.

[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[21] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[22] Isabelle Guyon and Andre Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[23] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. In *Journal of Machine Learning Research*, volume 13, pages 281–305, 2012.

[24] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, volume 14, pages 1137–1145, 1995.

## 7 Team contributions

**Debajyoti Dasgupta:** Started with exploratory data analysis and analyzing some models to figure out the best-performing models. Primarily worked with Random Forest Regressor and Deep Neural Networks. Extended on the Random and Grid Search for Random forest regressor hyperparameter tuning. Additionally explored NASA datasets and web scraping to collect image-based data. I also was involved in creating a preliminary CNN-based model and ideation for present and future approaches. Majorly designed, organized, and completed the majority of the report.

**Rushil Venkateswar:** Preformed dataset cleaning and exploration of Dask methodology and other distributed data packages along with the ideation and model creation and testing for XGBoost, TabNet, and Random Forest model. Significantly contributed to the feature reduction step and efficient data storage with parquet-based storage methods. Also worked on performing Grid search and performing hand tuning hyperparameter tuning to improve the performance of the model.

**Piran Karkaria:** Helped with the exploration and detailed description of the dataset and exploring other sources. Provided insights on data cleaning and writing a first draft of the report.