Bioinformatics Research Project Log

Chandrashekar CR
Supervisor: Dr. Eran Elhaik
Lund University

March 31, 2025

Contents

March 31, 2025

Tasks for the Day

- Understand the mGPS algorithm from the R code and implement the preprocessing steps. (Done)
- Set up the working environment and install all required libraries. (Done)
- Begin tracking the project. (Done)

Notes and Observations

- I am currently using the `ai_env` environment from my previous projects and accessing resources on the bioinformatics server.
- I am still struggling to understand the R code, but I expect to progress faster once I grasp the underlying logic.

April 1, 2025

Tasks for the Day

- Understand and implement Recursive Feature Elimination followed by the XGBoost machine learning algorithm with the correct hierarchical steps. (Done)
- Initialize Git for version control and push code to GitHub. (Done)
- Gain a deeper understanding of cross-validation.

Notes and Observations

- I have understood the general workflow and added comments to the MetaSUB preprocessing script in R.
- I learned about cross-validation and its importance, though I have not yet implemented it.

Pending Tasks

- Git tracking for the project has not been set up yet. (Done)
- Preprocessing steps are still in progress.

April 2, 2025

Tasks for the Day

- Ask Vignesh how to access the LUNARC server. (Done)
- Reimplement the XGBoost machine learning algorithm with the correct hierarchical steps from the previous day. (I don't need to do that)

Notes and Observations

- I initialized Git for the project and started tracking all files except data files and research papers.

- The `metasub_global_git.csv` file contains all the Geographically Informative Taxa (GITs) required for the XGBoost model.

- I realized that I do not need to recreate the exact model; my primary objective is to understand how the input data for XGBoost is preprocessed.

- The key questions to answer: What is the shape of the input data? What are we predicting?

Pending Tasks

- Gain access to the LUNARC server. (Done)

April 3, 2025

Tasks for the Day

- Implement basic neural network architectures in PyTorch. The dataset contains n data points, but hyperparameter tuning has shown that 200 GITs are sufficient for accurate predictions. (Done)

- Integrate Ray parallel processing into the hyperparameter optimization process, reducing the search time (currently estimated at 4-5 hours). (Done)

- Preprocess the data into numerical format by converting categorical variables (continents and cities) using one-hot encoding or label encoding. One-hot encoding seems to be the safer option. (Done)

- Implement stratified K-fold cross-validation before proceeding with a simple neural network model. (Will do it later.)

Notes and Observations

- I successfully logged into the LUNARC server's login node, but I still need to figure out how to access and use the GPUs for training neural networks.

- Authentication and login are complete, but I still need assistance in understanding the following:

  - **Where my allocated storage is located?**
  - **How to submit jobs using SBATCH.**
  - **I need to learn the basics of working on HPC?**

- After performing Recursive Feature Elimination (RFE), the final dataset was created with a shape of 4070×204. Out of the 204 columns, 200 represent the selected features, while the remaining 4 correspond to the target variables.

- Each row in the dataset contains 200 features (GITs) representing the relative sequence abundance (RSA) of microorganisms. The 4 target columns include the continent, city, latitude, and longitude of the sample collection site.

- The dataset comprises samples collected from 40 unique cities across 7 continents.

- Categorical variables (continent and city) were encoded using `sklearn`'s `LabelEncoder`, while latitude and longitude were standardized using `StandardScaler`.

- Initially, stratified cross-validation was considered for training the neural network. However, this approach was temporarily replaced with `train_test_split` for initial model development. Once the basic neural network is functional, stratified cross-validation will be reconsidered for improved model performance.

Neural Network Architecture

- The initial model is a simple feedforward neural network inspired by the hierarchical structure used in the previous study with XGBoost.

- The first version of the network includes an input layer with 200 nodes (corresponding to the 200 selected features), followed by two hidden layers with 400 nodes each. This is followed by a smaller hidden layer with 2 nodes, and finally, an output layer with 7 nodes representing the 7 target continents.

- The plan is to first train this model to predict the continent. Then, I will use the predicted probabilities of the continents as additional features by concatenating them with the original 200 input features. This combined input will be used in a second neural network (with a similar architecture) to predict the city.

- I am still unsure how to best handle the latitude and longitude values within the neural network, and this remains an open issue to resolve.

- One major challenge I've encountered is the long runtime during training. As I am currently working on the bioinformatics server, which only supports CPU-based computation, training is slow. I require access to GPUs hosted on the LUNARC cluster and need guidance on how to proceed with that setup.

April 4, 2025

Tasks for the Day

- Finish presentation today for the weekly lab meeting.

- Get GPUs access and understand HPC working architecture.

- Start the pre-processing for marine and soil data. The pre-processing for metasub, marine and soil data should be written in a modular format for easier exceection of the scripts.

Notes and Observations

Read about different implemetation of neural networks. Also did some reading and understanding about HPC.

April 7, 2025

Tasks for the Day

- Write a proper working neural netowrk that predicts all the target variables (continent, city, latitiude and longitude).

- Finish preprocessing steps for marine dataset and soil dataset.

- Start modularizing all the code so that it can be easily run on the HPC.

Notes and Observations

- Wrote a working script for processing metasub data. I need to do error handling and add function to check if the file format is correct. For now the script works and the preprocessing step also works.

- Wrote a working script to get valid information from the entire dataset using recursive feature elimination, with the base model as random forest.

- Started working the HPC. I understood the basics and now I have started to modularize the script for running them on the HPC GPU compute nodes.