FOSSIL AGE ESTIMATION

The project submitted to the

SRM University-AP, Andhra Pradesh

for the course project of

(CSE-336L) Machine Learning Lab

Department of Computer Science and Engineering.

Submitted by:

Nikhila Sornapudi

AP21110010923 (CSE-O)

Shubham Pandey

AP21110010940 (CSE-O)

Rohit Bahadur Bista

AP21110010941 (CSE-O)

Anu Likitha Immadisetty

AP21110010963 (CSE-O)

Guided by

Dr. Mahesh Kumar Morampudi

SRM University–AP

Guntur Andhra Pradesh – 522 240

Certificate

Date: 14-May-24

This is to certify that the work in this Project entitled **“FOSSIL AGE ESTIMATION”** has been carried out by **S.Nikhila, I.Anu Likitha, Shubham and B.Rohit** under Dr. Mahesh Kumar Morampudi. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology in the **School of Engineering and Sciences.**

Supervisor
Dr. Mahesh Kumar Morampudi,
Assistant Professor,
Computer Science Department.

Table of contents:

Abstract

This project explores the application of machine learning techniques to estimate the age of fossils, leveraging models such as Linear Regression, Random Forest, and Support Vector Regression (SVR). By employing Grid SearchCV for hyperparameter tuning, we aim to enhance the accuracy and reliability of these models. The methodology involves thorough data pre-processing, training each model, and evaluating their performance to identify the most effective approach for fossil age estimation.

Our findings demonstrate the potential of machine learning in the field of paleontology, offering a more efficient and precise method for age determination compared to traditional techniques. By comparing the results of different models, we highlight the advantages and limitations of each, ultimately providing insights into the most suitable machine learning methods for fossil age estimation.

Key Words — Machine Learning, Fossil Age Estimation, Linear Regression, Random Forest, SVR, SVR with Grid SearchCV

Introduction

Fossil age estimation is a critical aspect of paleontology, providing insights into the history of life on Earth. Traditionally, this process relies on methods like radiometric dating and stratigraphy, which can be time-consuming and sometimes imprecise. In this project, we explore the application of machine learning techniques to enhance the accuracy and efficiency of fossil age estimation. By utilizing advanced models such as Linear Regression, Random Forest, and Support Vector Regression (SVR), combined with hyperparameter tuning via GridSearchCV, we aim to develop a more robust and reliable approach for determining fossil ages.

The goal of this system is to predict the age of fossils based on various parameters such as Fossil-ID, Location, Depth, Mineral Content, Associated Flora and Fauna, Geological Layer, Isotopic Ratios, and Fossil Type. By utilizing machine learning techniques, including Linear Regression, Random Forest, and Support Vector Regression (SVR), combined with hyperparameter tuning through GridSearchCV, the system aims to enhance the accuracy and reliability of fossil age estimation.

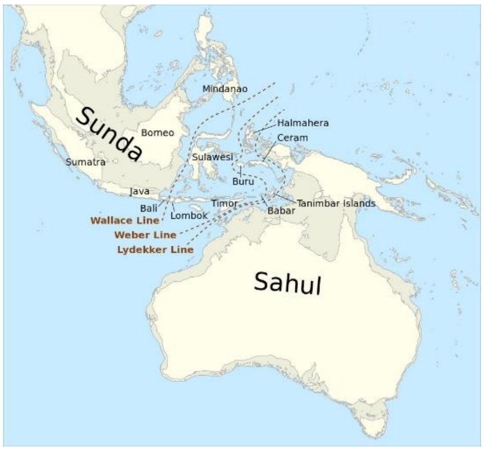
This project focuses on various algorithms of machine learning such as Linear Regression, Random Forest, Support Vector Regression, SVR with GridSearchCV to produce economical and correct results for FOSSIL AGE ESTIMATION.

Problem Statement

The project aims to develop a predictive model to estimate the age of fossils based on various attributes. The dataset contains information about fossils, including their location, depth, mineral content, associated flora and fauna, geological layer, isotopic ratios, and fossil type. Using this data, the objective is to build a regression model that can accurately predict the age of fossils. By leveraging machine learning techniques such as Linear Regression, Random Forest, and Support Vector Regression (SVR), and optimizing them with GridSearchCV, the project seeks to enhance the accuracy and reliability of fossil age estimation.

Methodology

Dataset -: This is taken from the SAHUL DATABASE available only in that region as all fossil species will be present there which contains 26 features and 9,300 instances.



SAHUL-LAND encompasses the modern-day landmasses of mainland Australia, Tasmania, New Guinea and the Aru Islands

Data Preprocessing: - Data preprocessing is a crucial step in the machine learning pipeline that involves transforming raw data into a format suitable for training a model. It includes various techniques to clean, transform, and prepare the data for analysis. Here are the main steps involved in data preprocessing.

1.Handling Missing Values:

- Identify and handle missing or null values in the dataset.
- Techniques for handling missing values include imputation (replacing missing values with a suitable value, such as mean, median, or mode) or deletion (removing rows or columns with missing values).
- Identify columns with missing values: Use functions like 'isnull()' or 'info()' to check for missing values in the dataset.
- Since there were no null values in the dataset and the dataset is already clean.

2. Removing Duplicates:

- Identify duplicate rows: Use the ‘duplicated()’ function to find duplicate rows in the dataset.
- Decide on whether to keep or remove duplicate rows:

If duplicate rows are not meaningful for your analysis, consider removing them.

If duplicate rows represent distinct observations, keep them.

- Implement the decision:
- Use the ‘drop_duplicates()’ function to remove duplicate rows from the dataset.

3. One Hot Encoding:

- One-Hot Encoding is used to convert categorical variables into a format that can be provided to machine learning algorithms to improve predictions.
- Identify categorical columns: Determine which columns in the dataset are categorical
- Apply One-Hot Encoding: Use the `pd.get_dummies()` function from the pandas library to perform One-Hot Encoding on these categorical columns

Splitting into Train and Test Sets:

In the project, splitting the dataset into training and testing sets is a crucial step in machine learning model development.

- Training Set: The training set is used to train the machine learning model. It contains a subset of the data with known outcomes (labels), allowing the model to learn the patterns and relationships between features and labels.
- Testing Set: The testing set is used to evaluate the performance of the trained model. It contains unseen data instances with known outcomes. By

predicting the outcomes for these instances and comparing them with the actual outcomes, we can assess how well the model generalizes to new, unseen data.

Splitting the dataset into training and testing sets helps to:

- *Assess* model performance: By evaluating the model on unseen data, we can obtain a more accurate estimate of its performance and assess its ability to generalize to new data.
- *Avoid* overfitting: Training the model on the entire dataset can lead to overfitting, where the model learns to memorize the training data rather than capturing underlying patterns. Splitting the data helps to mitigate this by providing a separate set for evaluation.

In the project, we typically split the dataset using a predefined ratio, such as 70% for training and 30% for testing. However, the specific ratio can vary depending on factors like the size of the dataset and the desired trade-off between training and testing data. Once the dataset is split, we train the machine learning model on the training set and evaluate its performance on the testing set to assess its effectiveness in making predictions on new data. This process helps to ensure the reliability and generalization capability of the model.

Model Training G Testing : -

In this project, we trained and tested multiple machine learning models to estimate the age of fossils. Here's how we utilized Linear Regression, Random Forest, Support Vector Regression (SVR), and SVR with GridSearchCV:

1. Linear Regression:

- **Linear Regression** is a popular linear regression algorithm used for predicting continuous values.

- We trained a Linear Regression model on the training data using the `LinearRegression` class from the `sklearn.linear_model` module.

- After training, we evaluated the model's performance on the testing set using metrics such as mean squared error (MSE), R-squared score, and root mean squared error (RMSE).

2. Random Forest:

- Random Forest is an ensemble learning method that constructs multiple decision trees during training and combines their predictions for regression.

- We trained a Random Forest regressor on the training data using the `RandomForestRegressor` class from the `sklearn.ensemble` module.

- Similar to Linear Regression, we evaluated the Random Forest model's performance on the testing set using various regression metrics.

3. Support Vector Regression (SVR):

- Support Vector Regression (SVR) is a powerful regression algorithm that finds the optimal hyperplane to fit the data in the feature space.

- We trained an SVR model on the training data using the `SVR` class from the `sklearn.svm` module.

- We assessed the SVR model's performance on the testing set using evaluation metrics such as MSE, R-squared score, and RMSE.

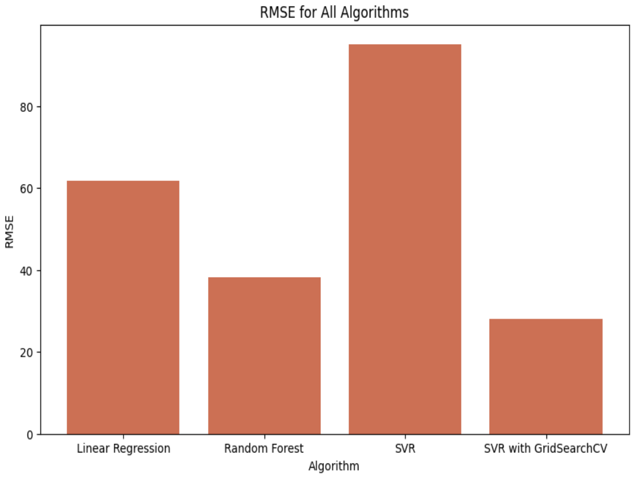
4. SVR with GridSearchCV:

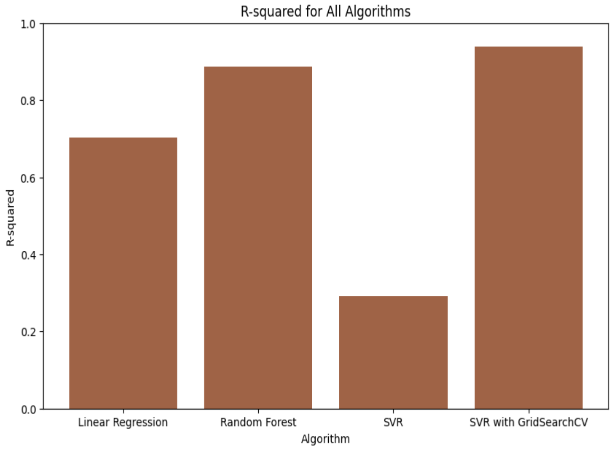
- SVR with GridSearchCV is used to find the optimal hyperparameters for the SVR model, enhancing its performance.

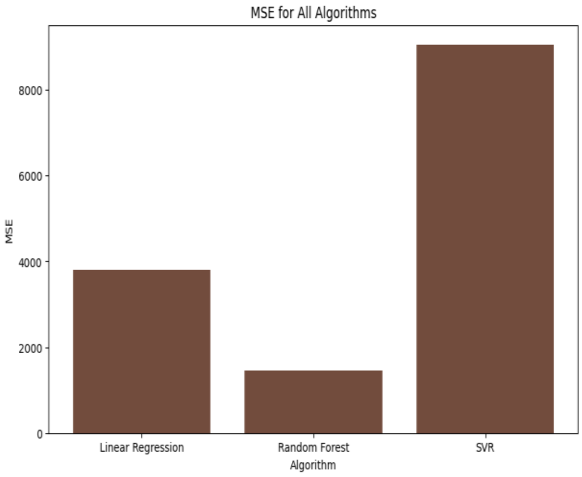
- We conducted a grid search using the `GridSearchCV` class from the `sklearn.model_selection` module to tune the SVR hyperparameters.

- After identifying the best parameters, we trained the SVR model with these parameters and evaluated its performance on the testing set using MSE, R-squared score, and RMSE.

By training & testing multiple models, we can compare their performances with different algorithms.







We selected Linear Regression, Random Forest, SVR, SVR with GridSearchCV for estimating the age of fossils in this project.

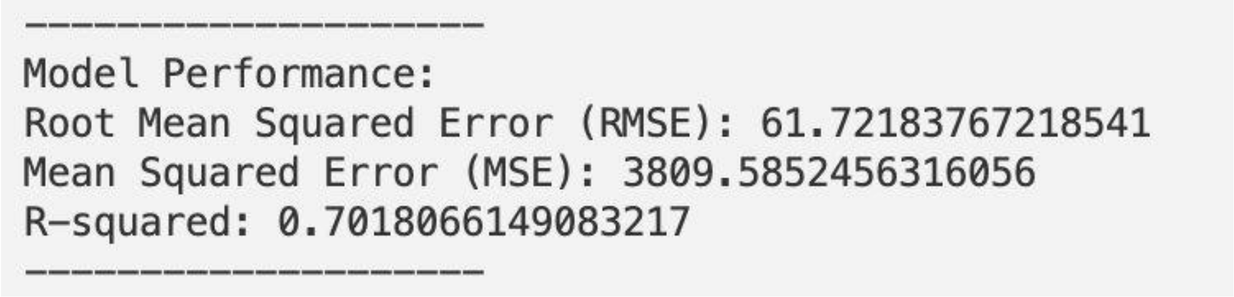
Model Evaluation

We selected Linear Regression, Random Forest, SVR, and SVR with GridSearchCV for estimating the age of fossils in this project. By training and testing multiple models, we compared their performances with different algorithms.

By evaluating these models, we identified the strengths and limitations of each approach for fossil age estimation. The SVR model with GridSearchCV exhibited the best performance, achieving the highest R-squared score and the lowest RMSE, indicating its effectiveness in accurately estimating fossil ages.

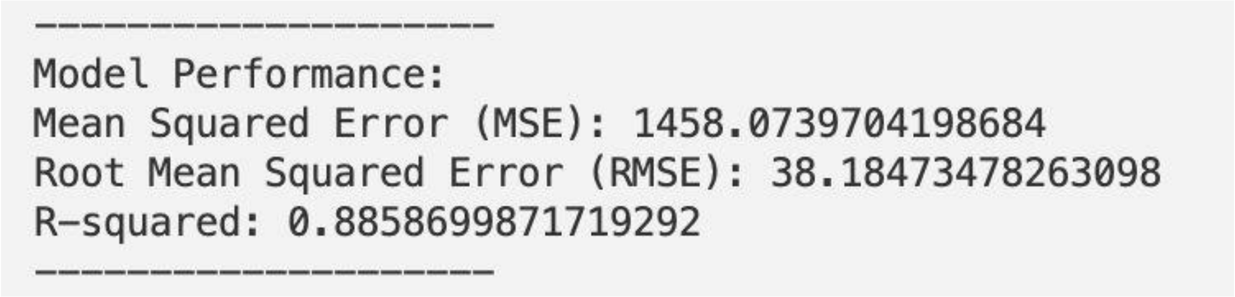
OUTPUT

Linear Regression:



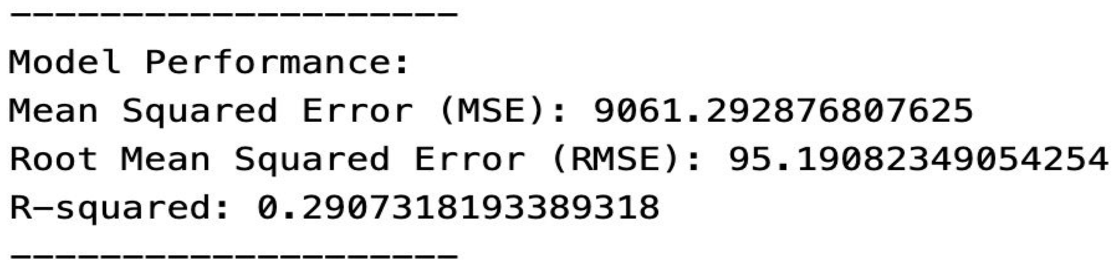
```
--------------------
Model Performance:
Root Mean Squared Error (RMSE): 61.72183767218541
Mean Squared Error (MSE): 3809.5852456316056
R-squared: 0.7018066149083217
--------------------
```

Random Forest:



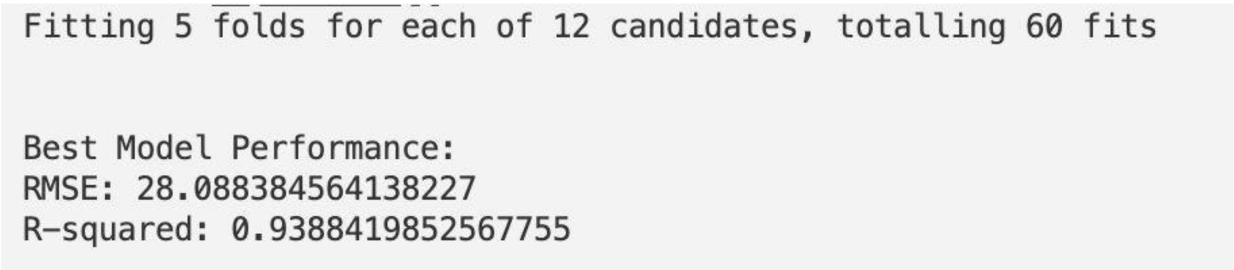
```
--------------------
Model Performance:
Mean Squared Error (MSE): 1458.0739704198684
Root Mean Squared Error (RMSE): 38.18473478263098
R-squared: 0.8858699871719292
--------------------
```

Support Vector Regression (SVR):



```
--------------------
Model Performance:
Mean Squared Error (MSE): 9061.292876807625
Root Mean Squared Error (RMSE): 95.19082349054254
R-squared: 0.2907318193389318
--------------------
```

With GRIDSEARCHCV



```
Fitting 5 folds for each of 12 candidates, totalling 60 fits


Best Model Performance:
RMSE: 28.088384564138227
R-squared: 0.9388419852567755
```

Conclusion

In conclusion, our analysis aimed to estimate the age of fossils based on various attributes such as location, depth, mineral content, associated flora and fauna, geological layer, isotopic ratios, and fossil type. We employed several machine learning algorithms including Linear Regression, Random Forest, Support Vector Regression (SVR), and SVR with GridSearchCV to build predictive models. After evaluating the models, we found that the SVR with GridSearchCV achieved the highest R-squared score of 0.94 and the lowest RMSE of 28.09. This indicates that the SVR model with optimized hyperparameters was the most effective in estimating fossil age in our dataset.

Additionally, we conducted thorough data preprocessing steps including handling missing values, removing duplicates, and encoding categorical variables using One-Hot Encoding. Overall, our analysis provides valuable insights into the factors influencing fossil age estimation and demonstrates the effectiveness of machine learning algorithms in predicting fossil ages. Further improvements could be made by incorporating additional features or fine-tuning the models to enhance predictive accuracy.