Machine Learning with Everyone Can Learn Data - Datacamp Challenge

Fouzan Asif Siddiqui[1]

[1]Data Science Department, National University of Computer and Emerging Sciences FAST, Karachi, Sindh, Pakistan. Email: *k238054@nu.edu.pk*

Abstract

Machine learning empowers computers to learn from data and make predictions. This report leverages data science pipeline execution along with detailed Machine Learning on dataset provided by Datacamp for competition "Everyone Can Learn Data Science", a dataset which is rich with historical, anatomical, and regional dinosaur fossil information. Data Cleaning was done using imputation techniques using Simple Imputer for null values in region (42/4951), KNeighborsImputer for null values in diet and type, Random Forest imputation for null values in family attribute, and Linear Regression for null values in length_m attribute. were used to populate missing attributes as all these Imputers were capturing decision boundaries well as learned from experiments. Furthermore, Data visualization and cluster validity analysis (purity, entropy, cohesion, separation) on the "class" attribute shed light on data structure. Also, the visualizations helped understand the correlation and causal relationships between different attributes of data. Modeling explored various classifiers for "family", "region", and hybrid predictions, along with independent and ensembled latitude/longitude modeling for "region" prediction. Heterogenous ensemble Techniques as well has homogenous ensembled techniques like bagging, boosting, and stacking boosted model performance. Generative Adversarial Networks (GANs) and bootstrapping were employed to expand the dataset through sample generation. This report unveils the capabilities of machine learning for analyzing paleontological data, showcasing the power of data cleaning, visualization, clustering, ensemble learning, and data augmentation to extract knowledge and make predictions from the Dinosaur fossil dataset.

1. Introduction

The field of paleontology has traditionally relied on meticulous fossil excavation, anatomical analysis, and historical context to understand the fascinating world of dinosaurs.

	name	diet	type	length_m	max_ma	min_ma	region	lng	lat	class	family
0	Protarchaeopteryx	omnivorous	small theropod	2.0	130.0	122.46	Liaoning	120.733330	41.799999	Saurischia	Archaeopterygidae
1	Caudipteryx	omnivorous	small theropod	1.0	130.0	122.46	Liaoning	120.733330	41.799999	Saurischia	Caudipterygidae
2	Gorgosaurus	carnivorous	large theropod	8.6	83.5	70.60	Alberta	-111.528732	50.740726	Saurischia	Tyrannosauridae
3	Gorgosaurus	carnivorous	large theropod	8.6	83.5	70.60	Alberta	-111.549347	50.737015	Saurischia	Tyrannosauridae
4	Gorgosaurus	carnivorous	large theropod	8.6	83.5	70.60	Alberta	-111.564636	50.723866	Saurischia	Tyrannosauridae

Figure 1: First 5 rows of dataset

However, the ever-growing volume of paleontological data presents a challenge: extracting meaningful insights and uncovering hidden patterns becomes increasingly difficult with traditional methods. This is where machine learning (ML) steps in, offering a powerful set of tools to revolutionize dinosaur research.

Machine learning empowers computers to learn from data and make intelligent predictions. By applying ML techniques to paleontological datasets, researchers can unlock a deeper understanding of dinosaur evolution, diversity, and distribution. This report explores the application of ML to the Everyone Can Learn Data Science: Dinosaurs.csv competition dataset. This rich dataset provides a unique opportunity to delve into the world of dinosaurs through various attributes, including:

Historical context: Data on the minimum and maximum millions of years ago (mya) a particular dinosaur species existed helps us understand the temporal distribution of dinosaurs across different eras. Anatomical features: Information on dinosaur length is a valuable metric for understanding size variations within and across species. Dietary classification: Categorization as carnivore or herbivore sheds light on the feeding ecology of different dinosaur groups. Geographic distribution: Data on the latitude and longitude of fossil discoveries allows us to explore the geographical spread of dinosaur populations. Taxonomic information: The dataset includes classifications for dinosaur class (e.g., Large Theropod) and family (e.g., Tyrannosauridae), providing a crucial framework for understanding evolutionary relationships. However, the raw data itself is not always ready for analysis. Data cleaning, a crucial initial step in any ML project, plays a vital role in ensuring the quality and integrity of the information used for modeling.

TO extract valuable insights from data, in this project, I opted to work with panteological data provided by Datacamp's Competition "Everyone can Learn Data". With a total of 4951 rows, and 12 columns that contain historical, attributional, and geographical aspects of dinosaurs fossils, this dataset was used to execute data science pipeline along with Data Cleaning, Data Augmentation, Data Visualization, Data and Cluster Validity, Modelling, Ensembled Learning, and using GAN to generate data samples. Given below is the dataset description:

Following data cleaning, data visualization techniques are employed to explore the relationships and distributions within the dataset. This visual exploration helps us identify potential patterns and guide further analysis.

Next, the report explores the application of various clustering algorithms to the "class"

S.no	Column	Non-Null	Count	Dtype
1	occurrence_no	4951	non-null	int
2	name	4951	non-null	object
3	diet	3596	non-null	object
4	type	3596	non-null	object
5	length_m	3568	non-null	float64
6	max_ma	4951	non-null	float64
7	min_ma	4951	non-null	float64
8	region	4909	non-null	object
9	lng	4951	non-null	float64
10	lat	4951	non-null	float64
11	class	4951	non-null	object
12	family	3494	non-null	object

Table 1: Dataset Description

	length_m	max_ma	min_ma	lng	lat
count	3568.000000	4951.000000	4951.000000	4951.000000	4951.000000
mean	8.212688	117.518477	106.622270	-37.048675	34.591448
std	6.629887	45.270821	44.395885	84.591106	23.961138
min	0.450000	70.600000	66.000000	-153.247498	-84.333336
25%	3.000000	83.500000	70.600000	-108.258705	36.274439
50%	6.700000	99.600000	89.800000	-96.099998	42.611198
75%	10.000000	155.700000	145.000000	27.383331	47.745138
max	35.000000	252.170000	247.200000	565.000000	78.101875

Figure 2: Describing statistics about our Dataset

and "family" attributes. By grouping dinosaur entries based on shared characteristics, we can gain insights into the natural organization within the data and assess the effectiveness of the clustering through techniques like purity, entropy, cohesion, and separation. This can provide us 2 insights, how correct the data is, how healthy can our analysis be during modelling.

Finally, the power of ML modeling comes into play. This report details the exploration of different classification algorithms used to predict the "family" attribute based on the available features. Additionally, the latitude and longitude data are modeled independently and through ensemble methods (bagging, boosting, stacking) to predict the "region" of fossil discovery.

Generative Adversarial Networks (GANs) and bootstrapping techniques are also explored to augment the dataset size by generating new, realistic data points. This data augmentation can potentially improve the generalizability and robustness of the trained models.

Throughout this report, we aim to showcase the effectiveness of various machine learning techniques for analyzing and extracting knowledge from paleontological data. We demonstrate how data cleaning, visualization, clustering, ensemble learning, and data augmentation can be leveraged to uncover hidden patterns and make insightful predictions from the Dinosaurs.csv dataset.

2. Data Cleaning

2.1 Managing Null Values

As observed from table 1, we have a large number of null values in Diet, Type, Length_m, and Family whereas a small number of null values in Region. Figure 3 gives you a detail of missing values.

As seen in the figure 3, mostly null values for diet, type, and length_m occur together. However, for family, the distribution changes. Thus, in this project, "Region", having only 42 nulls, was imputed using Simple Imputer which simply replaces the missing values with the most frequent value. For family attribute, Random Forest Imputation was used since Random forest was capturing the most accurate decision boundaries on our dataset, and family attribute was a sensitive attribute. Furthermore, diet and type were imputed using KNeighborsImputer as for attributes such as diet and type, the closest number of samples hold a priority as they're alike. Lastly, length_m was imputed using Linear Regression. Since the dataset size was very small, we could not have dropped any null values and using simple central tendency wouldn't have solved the problem.

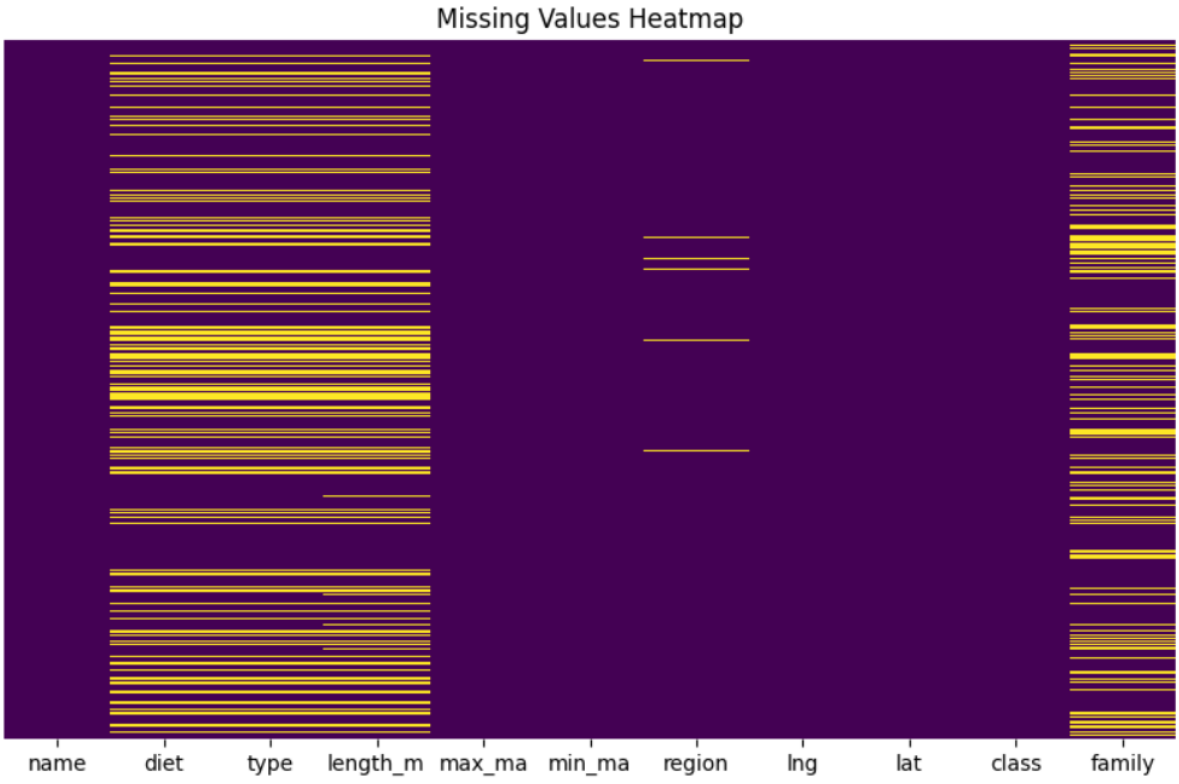

Figure 3: Null Values Map accross the dataset

2.2 Increasing size of dataset

Increasing the size of dataset is important to allow visualization and modelling with moderate variance and moderate bias. Otherwise, problems such as data imbalance, data biasness, underfitting, incorrect decision boundaries will emerge. Since there exist no more null values, increasing the size of dataset can now be performed.

To increase size of dataset, Bootstrapping and Generative Adversarial Networks were used to sustain the quality of existing samples, and add a new small amount of samples using our data. Bootstrapping simply duplicates with shuffling the samples from existing dataset. GANs, however, use a discriminator and a generator to first train the prediction model on entire data, and then train generator model that uses RNN is trained to produce a stream of new samples.

3. Data Visualization

3.1 Relationship of potential target attributes

In this section, we performed 2D and 3D visualizations to analyze the different relationships and patterns between different attributes. Our dataset, however, having multiple dimensions, was converted to 2 and 3 visual components to perform visualization easily. Visualization, however, was done on 2 target attributes i.e. Class and Family attributes.

In figure 5 and 6, we can see that the distribution is overlapping, and class won't be a

```python
desired_size = 12000
current_size = len(df)
additional_samples_needed = desired_size - current_size

bootstrapped_samples = df.sample(n=additional_samples_needed, replace=True, random_state=42)
bootstrapped_df = pd.concat([df, bootstrapped_samples])

display(bootstrapped_df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 12000 entries, 0 to 1055
Data columns (total 11 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   diet      12000 non-null  object
 1   type      12000 non-null  object
 2   length_m  12000 non-null  float64
 3   region    12000 non-null  object
 4   lng       12000 non-null  float64
 5   lat       12000 non-null  float64
 6   family    12000 non-null  object
 7   class     12000 non-null  object
 8   max_ma    12000 non-null  float64
 9   min_ma    12000 non-null  float64
 10  name      12000 non-null  object
dtypes: float64(5), object(6)
```

Figure 4: Bootstrapping to 12000 values

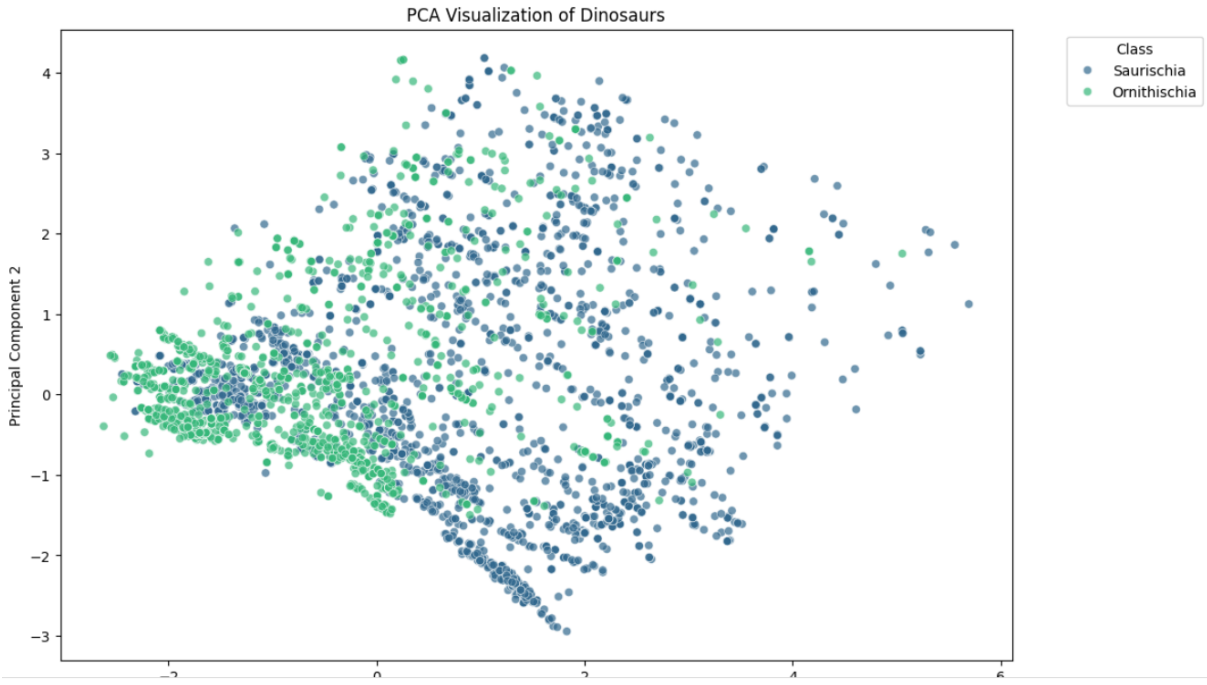

Figure 5: 2D representation of scattered class attributes

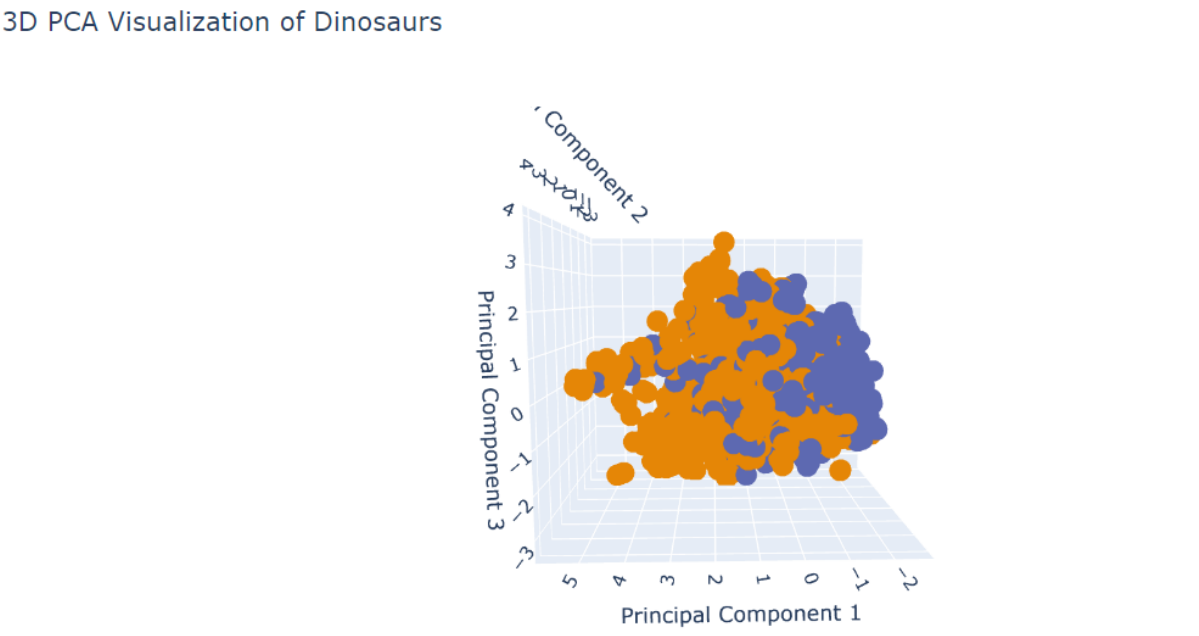

Figure 6: 3D representation of scattered class attributes

straightforward attribute to predict and may need some transformation (probably kernel transformation using Support Vector Machine). In case where class classification is involved, we'll need to cluster these groups into several subgroups to avoid misclassification.

Similarly, in figure 7 and **??**, families attributes appear to be mixed as well. So we'll need models with appropriate decision boundaries.

3.2 Correlation Analysis

Correlation analysis of dataset is performed to see the best attributes for modelling (that may have a correlational or causal relational pattern). Selecting correlated attributes can help better generalization as the data is considered to have a pattern. Otherwise, decision boundaries are complex.

4. Cluster Validity Analysis

In this section, we have observed the behaviour of our dataset as in clusters with 2 basic ideas i.e. groups that exist within our class attribute, and clusters that our data can be grouped into. Attached below, is a document that describes cluster validity analysis of our data.

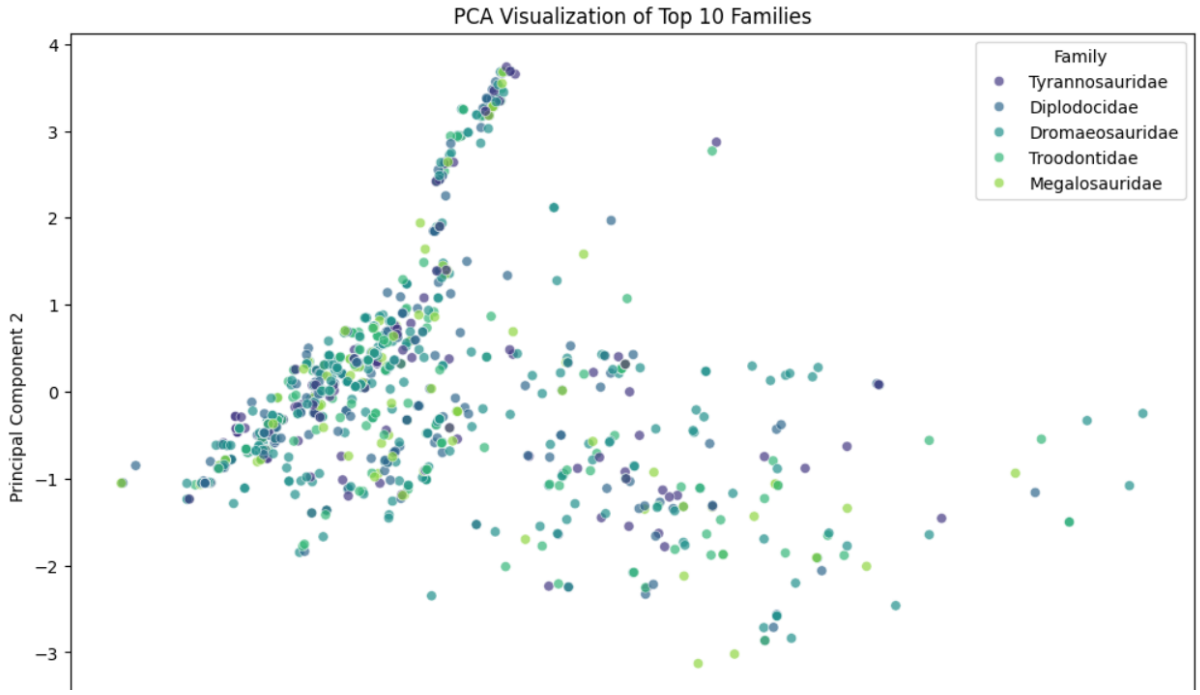

Figure 7: 2D representation of scattered top 10 families

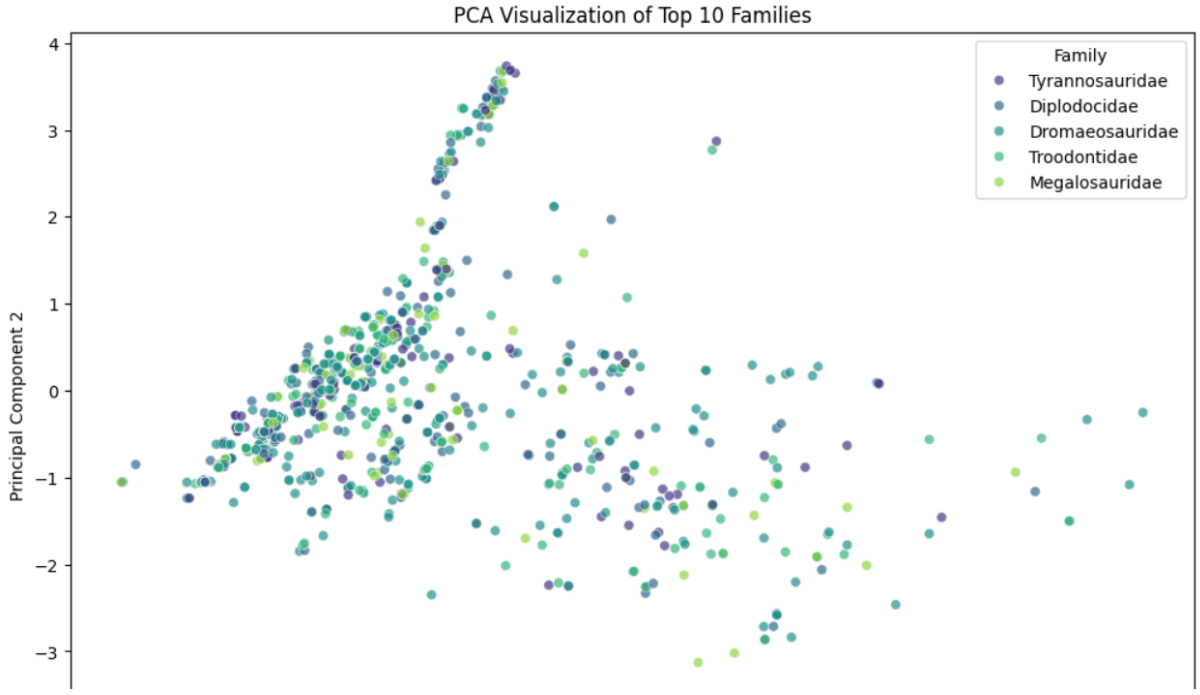

Figure 8: 3D representation of scattered top 10 families

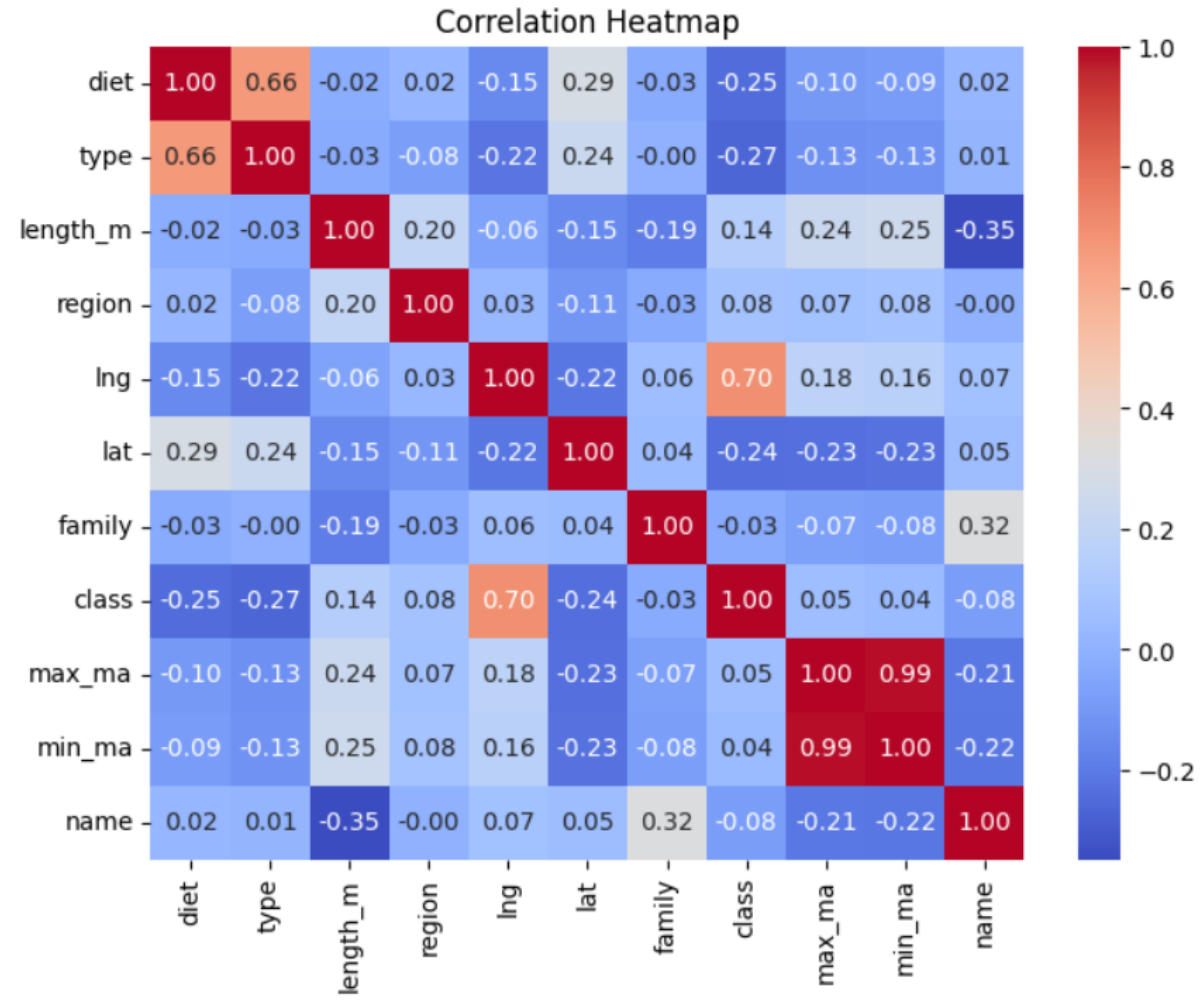

Figure 9: Correlational heat map between attributes

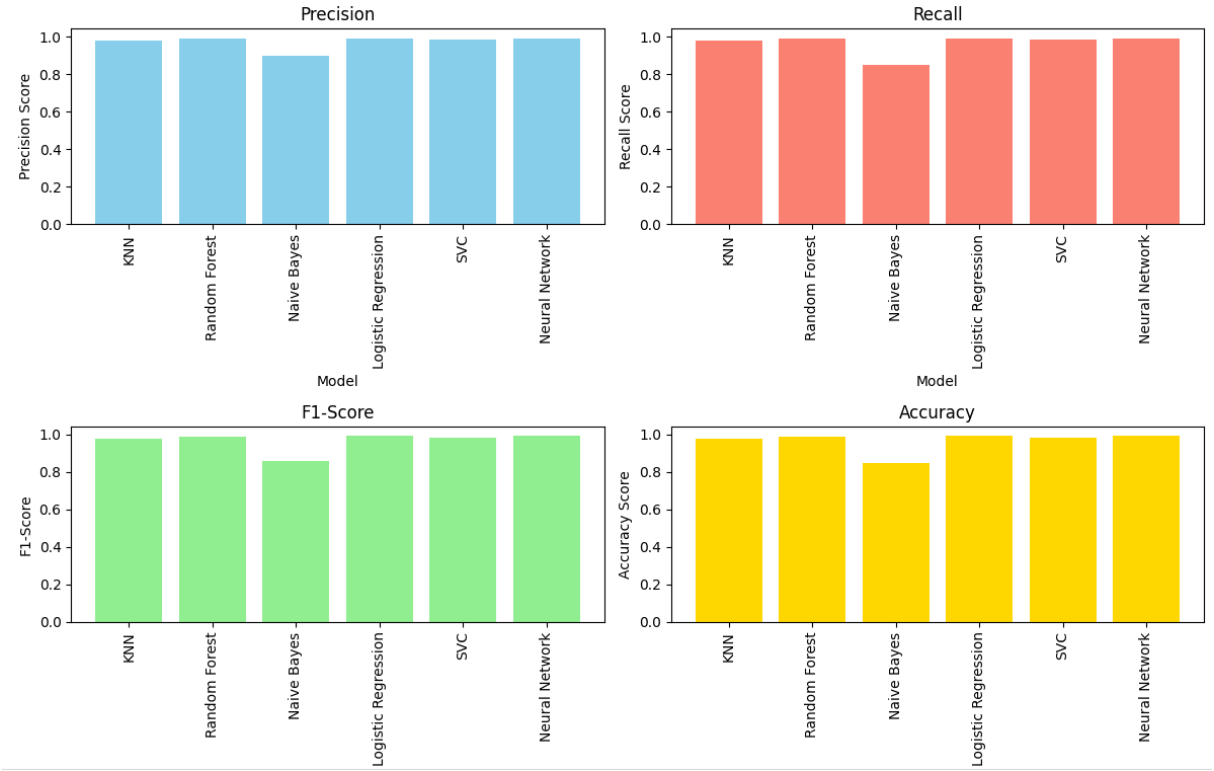

Figure 10: Performance of top 6 models on the dataset class attribute prediction

5. Modelling

5.1 Selecting the right models

Selecting the correct models that effectively capture your decision boundary are significantly important. Given in figure 10, 11, and in tables 2, 3, we can see that almost all models except SVC and Naive Bayes did well in terms of performance metrics (accuracy, precision, recall, f1 score). However, on family attribute, only Decision Tree and Random Forest performed well. The rest underperformed.

Name	Precision	Recall	F1-Score	Accuracy
KNN	0.979085	0.978809	0.978806	0.978809
Random Forest	0.989926	0.989909	0.989888	0.989909
Naive Bayes	0.900497	0.849647	0.857755	0.849647
Logistic Regr.	0.990982	0.990918	0.990926	0.990918
SVC	0.985102	0.984864	0.984836	0.984864
Neural Network	0.993025	0.992936	0.992946	0.992936

Table 2: Performance Stats for Models on Class Attribute

Also, Random Forest was used to perform several other predictions as given in the figure 12. And the overall accuracy was observed to be high.

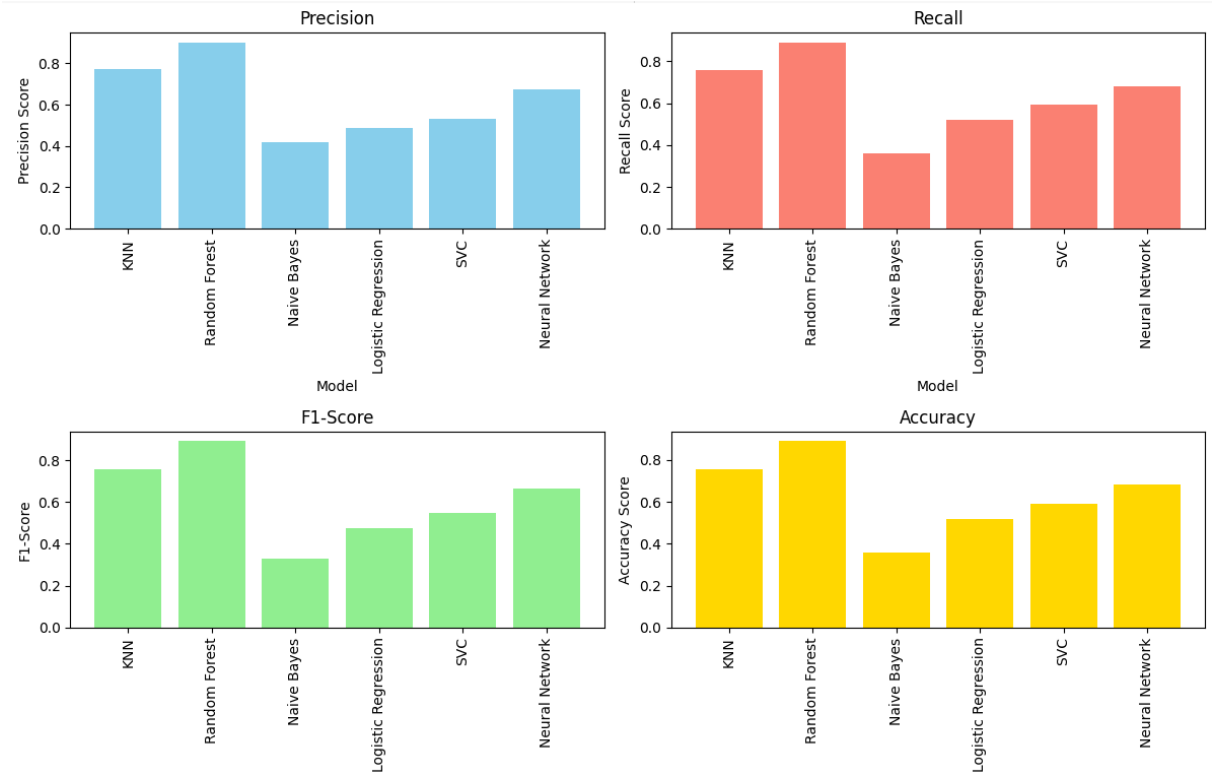

Figure 11: Performance of top 6 models on the dataset family attribute prediction

```
Diet Prediction Accuracy: 0.7907133243606999
Geographical Distribution Analysis Accuracy: 0.9367429340511441
Body Size Estimation Mean Squared Error: 0.014552086137281305
Family Classification Accuracy: 0.8337819650067295
Type Prediction Accuracy: 0.8371467025572006
```

Figure 12: Random Forest's Accuracy in different scenarios

Name	Precision	Recall	F1-Score	Accuracy
KNN	0.771464	0.757820	0.756573	0.757820
Random Forest	0.901982	0.892028	0.892220	0.892028
Naive Bayes	0.419243	0.360242	0.328353	0.360242
Logistic Regr.	0.488575	0.518668	0.473469	0.518668
SVC	0.530460	0.593340	0.547249	0.593340
Neural Network	0.671455	0.681130	0.664453	0.681130

Table 3: Performance Stats for Models on Family Attribute

```
Random Forest CV Scores: [0.97578204 0.97777778 0.98383838 0.98989899 0.97171717]
Random Forest Mean CV Accuracy: 0.9798028723154859
Random Forest Test Accuracy: 0.9899091826437941

Logistic Regression CV Scores: [0.98082745 0.99292929 0.98686869 0.9959596  0.97373737]
Logistic Regression Mean CV Accuracy: 0.9860644793036316
Logistic Regression Test Accuracy: 0.9909182643794148

K-Nearest Neighbors CV Scores: [0.96266398 0.97272727 0.97979798 0.95959596 0.96969697]
K-Nearest Neighbors Mean CV Accuracy: 0.968896431520044
K-Nearest Neighbors Test Accuracy: 0.9788092835519677

Support Vector Machine CV Scores: [0.97880928 0.98484848 0.98787879 0.99090909 0.97575758]
Support Vector Machine Mean CV Accuracy: 0.9836406445891814
Support Vector Machine Test Accuracy: 0.9848637739656912
```

Figure 13: Cross Validation Performance on different models

5.2 Modelling with Cross Validation

Simultaneously, I tried using Cross Validation on different models to see if the decision boundaries on the data improved. The figure 13 shows the results achieved.

5.3 Regressors

Until now, we have used Classification with Cross Validation. However, in this section, I have used 5 regressors to predict numerical values - Longitude and Latitude. The 5 regressors include Linear Regressor, Gradient Boosting Regressor, Random Forest Regressor, Support Vector Regressor, and KNN Regressor. The results are provided in table

S.no	Models	MSE Longitude	MSE Latitude
1	Random Forest Regressor	0.386889	0.454425
2	Gradient Boosting Regressor	0.531717	0.597956
3	Support Vector Regressor	0.795400	0.869126
4	k-Nearest Neighbors Regressor	0.422673	0.486455
5	Linear Regression	0.938532	0.890386

Table 4: Cross Validation with Regressors

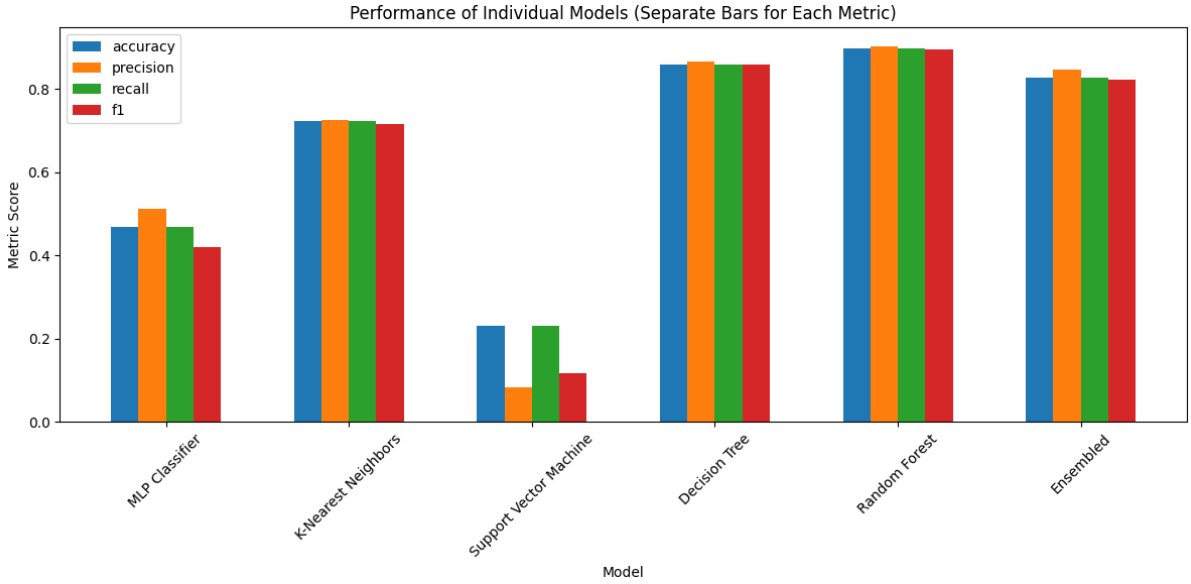

Figure 14: Overall performances by models vs Ensemble

6. Ensembled Learning

In this section, we have used 5-6 different scenarios to execute ensemble to increase performance and predicting power for our dataset.

6.1 Simple Heterogenous Ensemble

For creating a simple heterogenous ensemble, I opted for 5 different models namely KNN, Neural Network, SVM, Decision Trees and Random Forest and combines them to ensemble using VotingClassifier. As observed in the figure 14, performance of ensemble can be affected when if a model is highly underperforming, in our case, SVC on Families attributes (as creating 75 hyperplanes would be horrendous).

6.2 Heterogenous Ensemble with Cross Validation

Cross Validation combined with Ensembled Learning augments the generalization of the model and decreases the bias and variance greatly. Thus, in this section, I have used 5 StratifiedKFolds to train all models individually on different splits (1 split per model), and combine them together to ensemble using VotingClassifier. Results are shown in figure 15

6.3 Multi-level Heterogenous Ensemble

In this section, I have explored how to use ensemble to predict 2 different variables, and then consequently use them to predict a third variable. In this case, I have used the dataset to predict longitude and latitude of a dinosaur's fossil given its family, class, type, dies, length in meters, max_ma, and min_ma. Using all the predicted longitude and

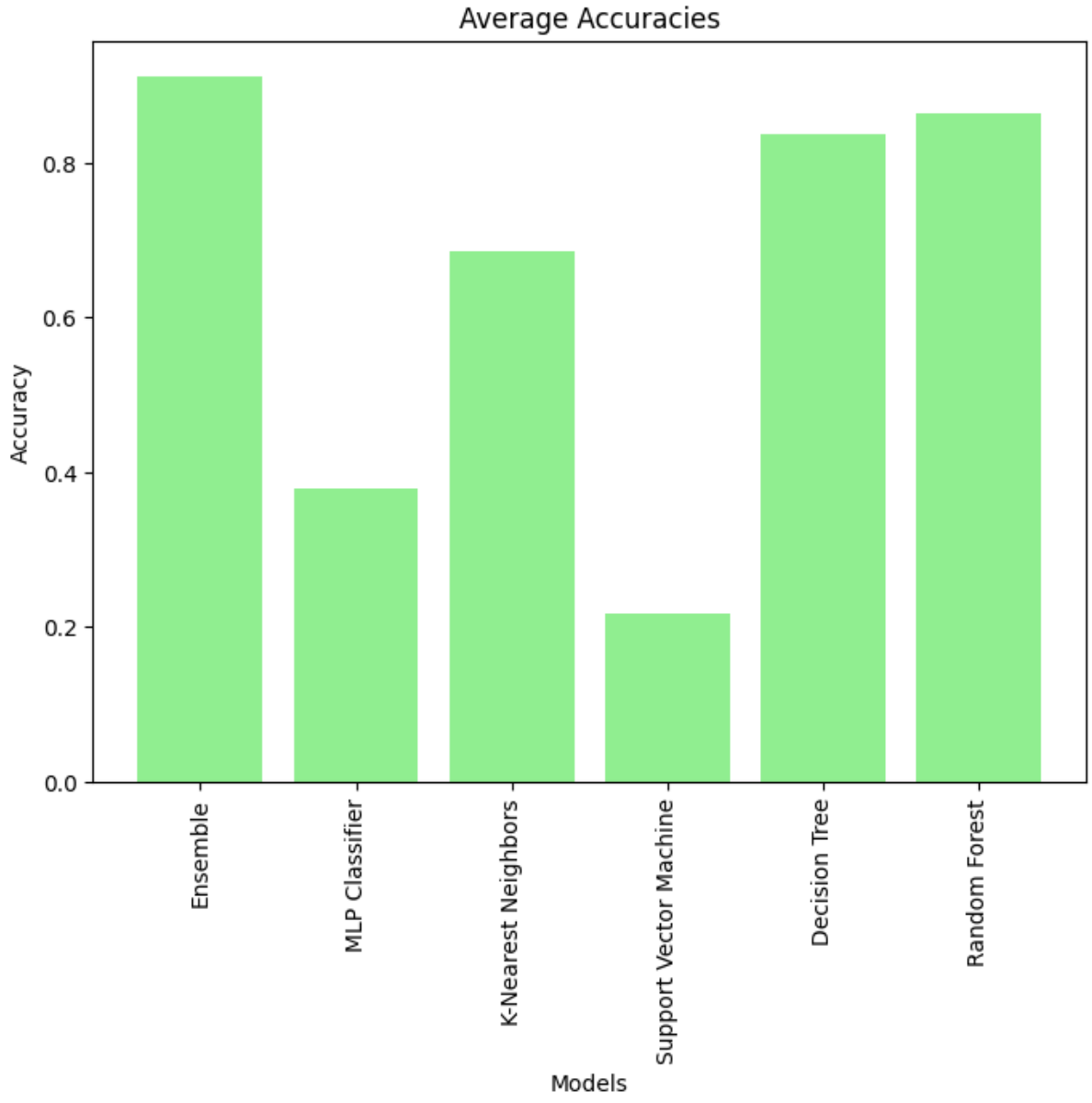

Figure 15: Ensemble and Individual Models with Cross Validation

latitude, the accuracy of model with new longitude and latitudes was compared to the one with previously created model.

Accuracy of Region Prediction on new Longitude and Latitude: 0.773966 Accuracy of Region prediction on ensembled Longitude and Latitude: 0.619576

We observe that when the Longitude and latitude themselves were ensembled using Voting Classifier, the accuracy decreases (mainly due to type of models used). In first case, only Random Forest Regressor was used to predict Longitude and Latitude, whereas in 2nd case, Linear Regression, Random Forest, ExtraTreesRegressor, and SVR were used which has led to the decrease in the accuracy in 2nd case.

6.4 Homogeneous Ensemble - Bagging Technique

In bagging, the model is preserved whereas the data changes. As we observed, Random Forest Classifier had 89% accuracy on Family Attribute. So it was kept as base estimator. However, since we're using transformed data now where labels are LabelEncoded, let's analyze the results.

Accuracy of Bagging ensemble: 0.847 Accuracy of Normal classifier: 0.816 Out-of-Bag score: 0.834

The aforementioned results reveal that Bagging is performing better than actual Random Forest even on scaled data.

6.5 Homogeneous Ensemble - Boosting Technique

In this section, AdaBoost Classifier was used to consider the misclassified samples and focus them more the next time. However, the results of AdaBoost on our dataset weren't soothing.

Accuracy of Boosting ensemble: 0.5913

In comparison with the accuracy of actual Random Forest on our scaled dataset i.e. 81.6%, Boosting's 59.1% can be explained as duplication of samples, and class imbalance.

6.6 Stacking Ensemble

In stacking, multiple models are combined in such a way that model to another input is the predictions of one input. To serve this purpose, StackingClassifier from sklearn.ensemble was used with Logistic Regression, Random Forest, and Gradient Boosting Classifier. However, Stacking obtained accuracy higher (97.59%) than that of bagging, boosting, and almost equal to other heterogeneous ensembles.

7. Conclusion

This report has elegantly demonstrated the power of machine learning (ML) in unlocking insights from paleontological data. By leveraging techniques like data cleaning, visual-

ization, clustering, and ensemble learning on the Dinosaurs.csv dataset, we were able to extract valuable knowledge about dinosaur evolution, distribution, and classification. The exploration of generative models for data augmentation holds promise for further enhancing model generalizability.

In conclusion, this work highlights the potential of ML as a transformative tool for paleontological research. As data collection continues to grow, ML will play an increasingly crucial role in unraveling the mysteries of the dinosaur era.