

Cardiff University

School of Computer Science and Informatics

CM3203 – Individual Project

Predicting Animal Movements Using Collar Data to Help Fight Poaching and Save Animals

Final Report

Author:

Jacob Harkins

Supervisor:

Charith Perera

Acknowledgements

I would like to thank Charith Perara and Naeima Hamed for contributions throughout the project with help and advice. I would also like to thank my family for their constant support.

Abstract

As Bornean elephants are driven to extinction, experts look towards machine learning to innovate patrol routing methods and potentially halt further poaching of the pygmy elephant. The original aim of the project was to explore the dataset with the intention of helping patrollers better understand elephant movement, and thus locate them easily. However, with the prevalence of consistent elephant movement trends in the usage of natural forest corridors, this project aims to develop and train a machine learning regression framework to predict the future GPS locations and movement trends of a Bornean pygmy elephant located in Sabah, Malaysia, in the hopes that it can be used as a secondary tool to base patrol routes around. Results from Linear, Polynomial and Vector Autoregressive regression models are compared in this project – Vector Autoregressive being the selected model that obtained the lowest RMSE, as well as the most precise movement trend prediction on a satellite image. Whilst the yielded results provide a promising foundation for those more experienced to build off, serious concerns regarding extrapolation leaves doubt in the project's ability to formulate predictions in the present.

Contents

Abstract	3
Contents	4
Figures	6
Introduction	8
Background.....	9
Literary review	11
Constraints.....	13
Papers the project is based on	13
Specification & Design.....	15
DGFC dataset.....	15
Language and libraries used.....	16
Data pre-processing.....	18
Selection of primary dataset.....	18
Duplicate records.....	18
Outliers.....	19
Dropping of variables	19
Data Processing.....	20
Averaging – different time scales	20
De-seasonalising and de-trending	20
Adding week number and month number	20
Data Visualisations	20
Machine learning	21
Overfitting.....	21
Selected models.....	22
Evaluating models	22
Comparing models	23
Implementation	24
Data Preprocessing	24
Reading of CSV files and removal of duplicates	24
Checking for ‘dummy’ columns	24
Dropping of variables, the changing of index, datetime conversion.....	24
Outlier Testing	25
Data processing	25

Creation of Data Frames for time frames.....	25
Calculating seasonal and trend values.....	26
Converting to a stationary time series	26
Basic GMap visualisations.....	26
Machine Learning.....	27
Linear Regression.....	27
Polynomial Regression.....	28
Vector Autoregressive model	29
Implementation limitations.....	29
Results & Evaluation	32
Data Pre-processing.....	32
Outlier test	32
Duplicate rows test.....	32
Variable viability tests.....	32
Data processing	33
Seasonality plots on original data	33
ADF tests	34
Stationary plots.....	35
Visualisations	38
Machine Learning.....	38
Time Series Split.....	38
Regression results	39
Predictions vs Actual satellite plots.....	40
Results appraisal.....	42
Future Work.....	43
Extrapolation.....	43
Unrealistic plotted movement patterns	43
Spatial Autoregressive (SAR)	43
Conclusion	44
Reflection	45

Figures

FIGURE 1 – IMAGE DISPLAYING THE IVORY TRAFFICKING ROUTES BETWEEN AFRICA AND ASIA [6]	9
FIGURE 2 – INCIDENCE OF ABSOLUTE POVERTY IN EACH MALAYSIAN STATE, 2019 AND 2020 [13]	10
FIGURE 3 – SATELLITE IMAGE OF DGFC AND THE SURROUNDING AREA [9]	11
FIGURE 4 - STEPS REQUIRED FOR A FINALISED MACHINE LEARNING MODEL	15
FIGURE 5: DATA FLOW DIAGRAM	18
FIGURE 6: DIAGRAM EXPLAINING THE FORWARD CHAINING PROCESS	21
FIGURE 7: DIAGRAM DEMONSTRATING A PREDICTION (BLUE) WHICH HAS A LOW RMSE FROM THE ACTUAL VALUES (RED), BUT A POOR MOVEMENT TREND FORECAST.	22
FIGURE 8: THE IMPORTING OF CSV FILES INTO A DATA FRAME AND THE DROPPING OF DUPLICATE RECORDS	24
FIGURE 9: TESTS FOR VALUES ABOVE ZERO IN POTENTIAL COLUMN DROP CANDIDATES	24
FIGURE 10: THE DROPPING OF VARIABLES, ALONG WITH DATA TYPE CHANGING FOR THE NEW INDEX	24
FIGURE 11: CALCULATING OF THE LOWER AND UPPER QUARTILE, AND THE INTERQUARTILE RANGE	25
FIGURE 12: QUERIES FOR FINDING OUTLIERS IN THE DATA	25
FIGURE 13: THE CREATION OF NEW TABLES FOR DIFFERENT TIME FRAMES	25
FIGURE 14: THE CALCULATION OF SEASONAL AND TREND VALUES	26
FIGURE 15: CONVERSION TO STATIONARITY	26
FIGURE 16: THE CREATION OF A GMAP PLOTTING FUNCTION	26
FIGURE 17: CREATION OF A HOVERTOOL FOR GMAP	26
FIGURE 18: DEFINING THE PROPERTIES FOR THE GMAP FIGURE	27
FIGURE 19: THE PLOTTING OF A DIFFERENT DATA SOURCE ONTO A GMAP	27
FIGURE 20: THE DEFINING OF LINEAR REGRESSION INPUTS	27
FIGURE 21: TRAIN/TEST DATA TIME SERIES SPLITTING	28
FIGURE 22: FITTING OF TRAINING DATA INTO A LINEAR REGRESSION MODEL, SUBSEQUENT PREDICTIONS AND RMSE SCORE	28
FIGURE 23: CREATION OF LINEAR DATA FRAME TO STORE PREDICTIONS	28
FIGURE 24: POLYNOMIAL REGRESSION MODEL	28
FIGURE 25: VAR SELECTED NO. OF PREDICTIONS, WITH SUBSEQUENT DATA SPLIT	29
FIGURE 26: VAR FITTING OF MODEL WITH SELECTED LAG ORDER AND TRAINING DATA	29
FIGURE 27: VAR FORECAST INPUT, RESULTS AND SUBSEQUENT DATAFRAME TO STORE VAR RESULTS	29
FIGURE 28: LONGITUDE PLOT ON ORIGINAL DATA	33
FIGURE 29: LONGITUDE PLOT ON ORIGINAL DATA	33
FIGURE 30: ADF TEST RESULTS	34
FIGURE 31: STATIONARY LONGITUDE PLOT FOR ORIGINAL DATA FRAME	35
FIGURE 32: STATIONARY LATITUDE PLOT FOR ORIGINAL DATA FRAME	35
FIGURE 33: STATIONARY LONGITUDE PLOT FOR WEEKLY DATA FRAME	36
FIGURE 34 : STATIONARY LONGITUDE PLOT FOR WEEKLY DATAFRAME	36

FIGURE 35: : STATIONARY LONGITUDE PLOT FOR MONTHLY DATAFRAME.....	37
FIGURE 36: : STATIONARY LONGITUDE PLOT FOR MONTHLY DATAFRAME.....	37
FIGURE 37: A SATELLITE IMAGE DISPLAYING THE GPS READINGS FOR ORIGINAL DATA FRAME.....	38
FIGURE 38: RESULTS OF FORWARD CHAINING CROSS VALIDATION - LINEAR REGRESSION.....	39
FIGURE 39: RESULTS OF FORWARD CHAINING CROSS VALIDATION - POLYNOMIAL REGRESSION.....	39
FIGURE 40: RMSE RESULTS - LINEAR REGRESSION.....	39
FIGURE 41: RMSE RESULTS - POLYNOMIAL REGRESSION	40
FIGURE 42: RMSE RESULTS - VECTOR AUTOREGRESSIVE	40
FIGURE 43: SATELLITE GPS PLOTS FOR LINEAR REGRESSION, ORIGINAL, WEEKLY, MONTHLY LEFT TO RIGHT	41
FIGURE 44: SATELLITE GPS PLOTS FOR POLYNOMIAL REGRESSION, ORIGINAL, WEEKLY, MONTHLY LEFT TO RIGHT	41
FIGURE 45: SATELLITE GPS PLOTS FOR VECTOR AUTOREGRESSIVE, ORIGINAL, WEEKLY, MONTHLY LEFT TO RIGHT	41

Introduction

Across the globe, wildlife rangers are trapped in an endless conflict against poachers. Year in year out, billions of endangered animals are killed or captured in the name of greed. Despite success in countries such as Uganda, where the elephant population has increased by 600% from the 1980s to the 2010s[1], the resources available to wildlife rangers are often strained by the need for constant patrols. Whilst necessary, it can be a difficult decision to balance budget and the safety of the animals – in an ideal world, the more patrols, the better. However, with the Ugandan Wildlife Authority revealing that 50-90% [2] of its budget is spent on patrols, patrols must be meticulously planned as efficiently as possible – a struggle when animals could be located anywhere within the reserve.

With this problem in mind, the primary aim of this project is to explore the data provided by the Danau Girang Field Centre with the intention of helping rangers better understand how elephants move within the reserve, so that it is easier to locate them on a different date. A sub-aim of this project is to make it easier for rangers to organise patrols within the reserve. Another sub-aim is to make it more difficult for poachers to get close to the animals and potentially harm them. The final sub-aim is to potentially make predictions on where animals are located within the reserve at a given date.

To achieve the aims set, the objectives I have within the project scope are:

- To process and clean the data. There should be no null values and irrelevant variables should be dropped.
- To develop multiple prediction techniques.
- To test each prediction technique so that a measure of accuracy can be obtained.
- To compare each prediction technique with each other so that the most optimal one can be selected.

It is crucial that the aims set are achievable within the scope of the project: to assist wildlife rangers in organising patrols efficiently.

To achieve these, assumptions must be made in the model that the elephants are not driven to other areas but outside interference, which could be anything – deforestation, natural disasters, etc. After all, the model that is being created cannot account for these things – predictions can only be made with the variables present in the data.

The approach I shall be taking in carrying out this project is a Waterfall development approach. As each of the aims listed above relies on the previous one to be implemented, I feel a more linear approach to developing the project is more suitable. As seen on the Gantt chart provided, each section during development will be completed one after another, ideally within a week for each section. However, the project timeline leaves extra time to focus on certain sections if needed.

Background

In the current environmental climate, more species of animals are getting driven to extinction than ever before. Whether it be deforestation, climate change, or the main driver for this project – poaching. The elephant family – an ancestral tree that stretches back over 10 million years [3], has had its Asian population drop by at least 50% in the past three generations [4]. An animal with a rich evolutionary history, at risk of getting slaughtered to extinction for its ivory, which is then used to make jewellery and other vanity items [5]. The largest market for purchasing ivory is in Asia, with massive markets in Japan, China, Hong Kong, and Vietnam – to name a few. Ivory is used to make a variety of items from musical instruments to name seals, ornaments, and jewellery [6].

With the largest markets being located within a proximity to each other, it is of no surprise that Malaysia has emerged as a central transit hub for the illegal smuggling of ivory [7]. Between 2003-2014, 19% of ivory seized globally was linked to Malaysia [8], leading to it being acknowledged as a major transit country at the 15th conference for countries/territories in CITES (Convention on International Trade in Endangered Species of Wild Fauna and Flora).

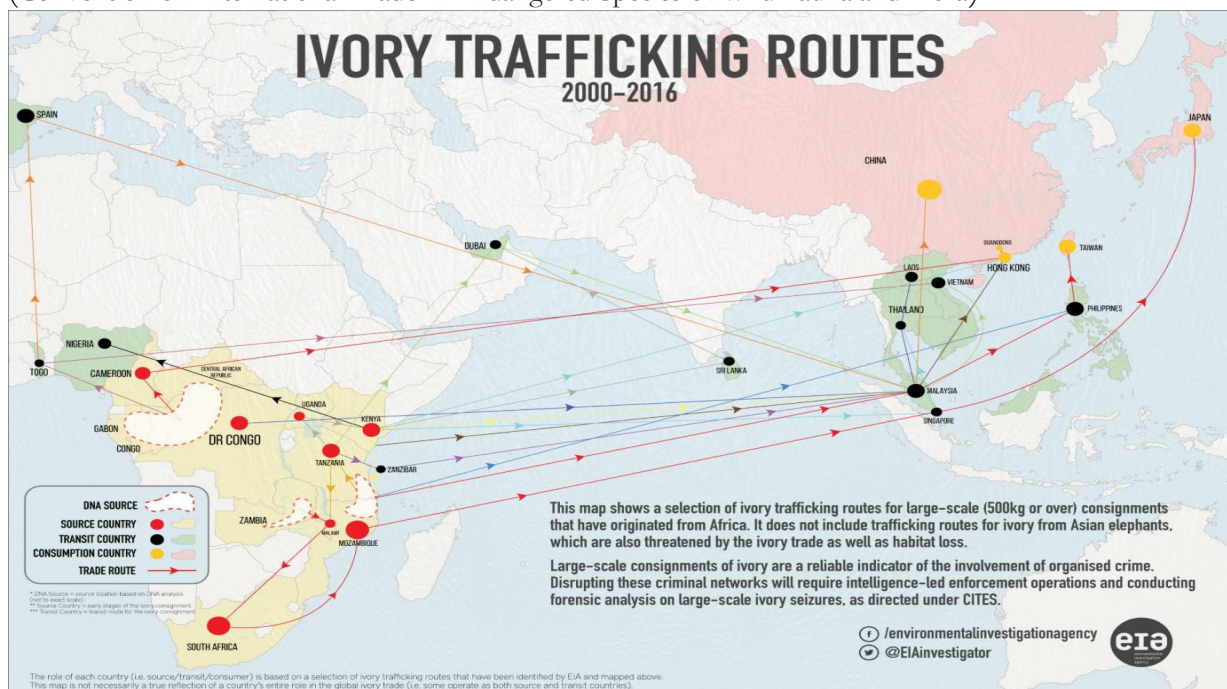


Figure 1 – image displaying the ivory trafficking routes between Africa and Asia [6]

It comes of no surprise that for wildlife rangers, the frontline of this epidemic is an endless war against a tide of poachers – the International Ranger Federation (IRF) & Thin Green Line Foundation (TGLF) reported the number of rangers murdered as 48 out of 107 total deaths in 2017 across Asia and Central Africa [10]. Poachers take advantage of the large reserves that make it difficult for rangers to keep surveillance on elephants. After all, rangers cannot keep an eye on every elephant at once, and even if they gain prior knowledge of poachers in the area, then it is often difficult to locate elephants at a moment's notice with how large and dense reserves can be. Whilst the best way to prevent this would be to have more frequent patrols, with how exhausting it is to maintain patrols on a reserve's budget, the next best solution is to make the patrols as

efficient as possible. Subsequently, with the predicted locations of these animals being made available to these rangers, they may be able to do so.

Borneo's reputation as one of the most biodiverse places in the world makes Sabah's tourism industry a massive part of the local economy - with an estimated 80000 jobs and in 2019, total tourism receipts were RM8.342bil and RM12 million in tourism tax revenue, with a total of 4.1 million tourist arrivals recorded [11]. However, with the COVID pandemic jeopardising the tourism industry across the globe, many residents in Sabah find themselves unemployed without a source of income – with the percentage of residents living under the national poverty line increasing from 19.5% in 2019 to 25.3% in 2020 [12], as well as paid and self-employed employment dropping 16.1% and 9.7% respectively [12]. Whilst elephants are seen as beloved and iconic creatures to those not in consistent contact with them, some in Sabah see them as pests. Deforestation, due to palm oil and logging, often sends elephants into neighbouring villages and plantations in the search for food, where they may destroy valuable crops and infrastructure in its path [13] - the current economic climate could push a family below the poverty line.

Locals poaching for bushmeat and/or ivory are not the only stakeholders within Sabah however – with 1.54 million hectares of land being used for oil palm plantations, elephants are sometimes shot at or even poisoned to prevent them from eating the fruit of the oil palm tree. One incident in Sabah, 2013 [14], left 14 elephants dead from a single herd – all poisoned along the border of one of these plantations

With these statistics and issues in mind, it is of no surprise that the multitude of animals in the local area paired with the short distance to major markets in neighbouring countries means the potential risk for wildlife is massive – people living locally, or in nearby countries may be tempted by the high selling price of ivory, which can bring in between \$121-\$900 per kilogram [15], as well as the ease of mind from taking care of what is seen as a dangerous beast to some locals.

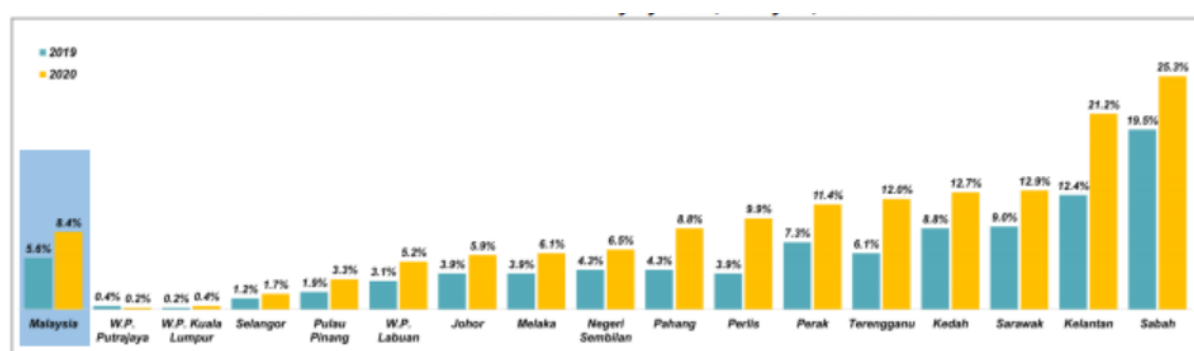


Figure 2 – Incidence of absolute poverty in each Malaysian state, 2019 and 2020 [13]

As a result of this, DGFC has provided all the data from GPS collars previously fitted onto elephants within the Kinabatangan region to the project, in the hopes of finding a way to assist rangers in curating a more effective patrol path and help prevent elephants and other animals that are at risk of being poached.

Danau Girang Field Centre (DGFC) is one of many wildlife research centres in Sabah, Borneo that accommodates researchers and scientists who are trying to discover ways in which to help local wildlife flourish, as well as mitigate risks in the fight against poachers. The DGFC is a collaboration research and training facility between the Sabah Wildlife Centre and Cardiff

University, located a couple of miles from the Kinabatangan River. Their aims are to further scientific research in the area, by contributing to long-term conservation projects and developing a better understanding of the local environment and the animals within it [16]. By utilising technology such as GPS collars, researchers at DGFC can explore survival mechanisms employed by species within the jungle – allowing them to formulate species action plans and landscape management guidelines to help the animal population prosper.

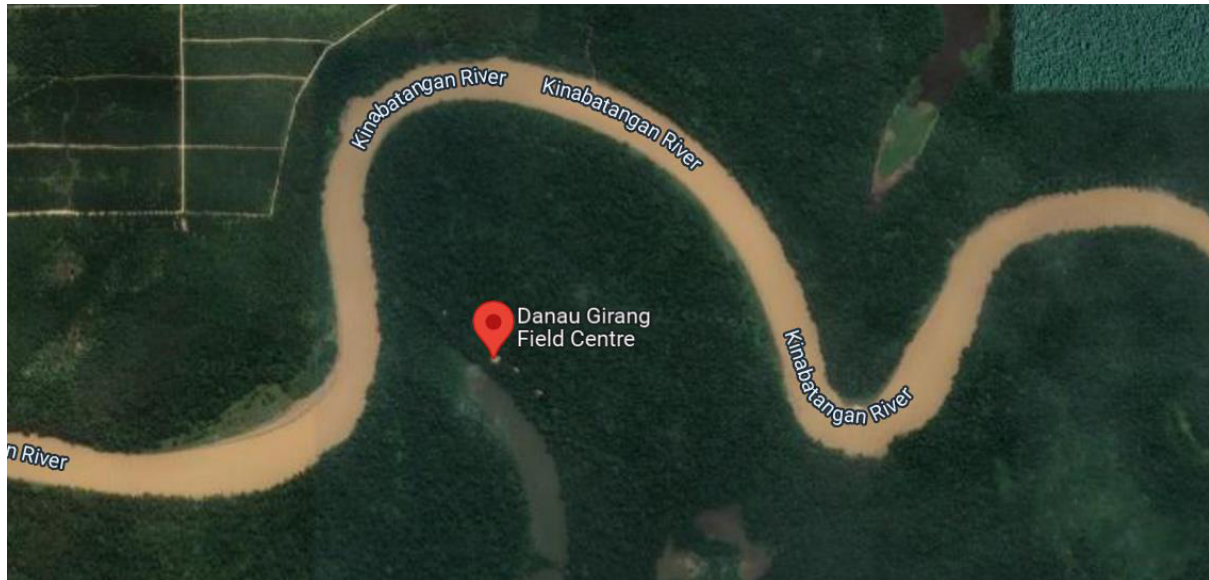


Figure 3– Satellite image of DGFC and the surrounding area [17]

With several wildlife reserves situated close to the research centre, researchers at the DGFC recognise how difficult it can be to monitor local wildlife within the dense vegetation that populates the lower Kinabatangan floodplains, the forests surrounding Kinabatangan River being the largest forest-covered floodplain in Malaysia [18]. A diverse landscape including mangrove forests, and swampland paired with peak temperatures and rainfall of 32°C from May-August [19] and 460mm in December, results in expeditions to locate animals having to be meticulously planned out, due to the physical and mental capacity needed to navigate such a topography.

Literary review

To combat the growing poaching industry, researchers have looked to machine learning to create frameworks to increase the efficiency of patrolling within reserves.

One such method used was an app called PAWS – Protection Assistant for Wildlife Security - designed to optimise human patrol resources and make foot patrols as efficient as possible, using game theory, and raster models paired with geographical data [21].

Whilst PAWS undoubtedly improved the efficiency of patrol routes, which coincides with the aim of this project to help rangers combat poaching and protect elephants, the focus of the project is using machine learning to make predictions on the location of elephants, whereas PAWS uses game theory decision making with probabilities to plot patrol routes. Whilst the overall objective is similar, the methodology is different - this project looks to help rangers by predicting where animals are at a given date, not by plotting routes. Furthermore, topological data that was used as the basis for their methods is not available to be used in Sabah, the dense vegetation making it difficult to distinguish landmarks that are drivers in animal and human distribution.

Nguyen, et al. (2016) made a similar comprehensive anti-poaching tool with temporal and observation uncertainty reasoning (CAPTURE) [22], that had larger success with plotting patrol routes than PAWS, through extension of security games with additional factors such as the defender's imperfect detection of poaching signs, complex temporal dependencies in the poacher's behaviours, and lack of knowledge of numbers of poachers. To train the model, 12 years of data provided from Queen Elizabeth National Park in Uganda.

Whilst the complex methods utilised have shown to be effective at optimising patrol routes, they are not available to be used in this project. Whilst it could be argued that there are relevant similarities between their project and ours regarding mapping out potential future routes, there are too many differences between mapping out human trajectory paths in comparison to those used by elephants for the method to be a feasible venture. Furthermore, the focus of the project is using machine learning to make predictions on future trajectories, whereas the focus of this project is the use of security games to map out optimal patrol routes. Also, their use of AUC as an accuracy metric is useful in classification scenarios, whereas the data provided by DGFC is exclusively continuous, meaning regression accuracy metrics are suited.

Kar et al. designed an interpretable classification ensemble to protect threatened species (INTERCEPT) [23], which differed from the methods used previously, instead designing a behaviour model based on decision trees that predicted poacher attacks and were more interpretable than previous iterations, with rangers observing 10 times the number of findings on patrols than the average. Another advantage it had over previous prediction tools is that the runtime was shorter due to the decision tree-based methods used having a lower complexity than the methods used in CAPTURE and such. 13 years of data was provided from Queen Elizabeth National Park in Uganda – the same as CAPTURE.

Furthermore, to refine the model, 12 years of data was used as a training set, with the last year being used as an evaluation set. As forementioned, the DGFC data set provided does not extend to such a range, meaning that a model designed using these methods is not likely to match up to expectations set by Kar et al., especially regarding how much data is required for an optimal decision tree.

US researchers conducted a study on Louisiana black bears [24], where they hoped to explore the significance that geographical features can have on a bear's movement and behaviour.

They used machine learning classifiers paired with environmental data from the National Land Cover Database to predict the bear's next 'step' (which is two sequential points, each with a step length and angle between them), and whether the bear would be in an 'exploratory' or 'foraging' state, which was concluded from step length and the various environmental features around the location. Seeing which environmental variables would have the greatest impact on a bear's next step and state, for example, distance from a river, allowed them to better understand which new 'steps' are more likely than others, probability wise - the step with the highest probability being inevitably selected.

Using geographical data would boost the accuracy of forecasts; however, there is a shortage of geographical data available in Sabah, thus these methods cannot be implemented. Additionally, the dense vegetation in Sabah makes it difficult to differentiate wildlife corridors, highways, and other potential animal movement drivers. In addition, while classification of animal movement stages may be useful for creating predictions a few days after current data due to its effect on the distance animals walk, the most recent data points in the Seri dataset are from early 2018. Extrapolation may result in less accurate forecasts as the distance between the predictions and the data increases. For instance, predictions for the year 2022 may be futile.

As the nature of our data is GPS measurements of a moving object, the next step was to look at papers which discussed methods in which future trajectories were predicted. Ayele et al published a survey,

which consisted of a library of published research papers that each utilised machine learning methods to predict the next location of people, vehicles etc.

One such paper [25] used deep-learning methods with contextual features to make location predictions on where vehicles would be located next. Firstly, the similarities between candidate locations were mined, then contextual features modelled between trajectories e.g., periodical patterns and dynamic trajectory features. Thirdly, CNN and LSTM were used to predict each trajectory with contextual information.

Elephants frequently re-use migration routes as they move between reserves, therefore analysing the periodic patterns between elephant migration routes could provide information into where they might go next. Observing the relationship between GPS locations and the dates on which they were collected and projecting visualisations of GPS locations over various time periods, such as months or weeks, can help reveal this connection more clearly and comprehensively. However, as previously said, there is a dearth of background.

Constraints

Whilst the forementioned papers were successful in completing their project aims, their approach and methodology left limitations that this project aims to solve:

- Years of ranger patrol data was required in approaches to predict a future trajectory route, which may not be available to a great number of reserves that lack the budget to collect such a range of data and may also lack the infrastructure to run the computationally intensive algorithms that are employed in PAWS, CAPTURE, INTERCEPT etc. These issues are resolved by utilising machine learning models that require less data and have a lower runtime, meaning the methods can be employed at a lower cost and a quicker rate.
- The collective focus on using methods to predict an optimal patrol route for rangers. While efficient patrol routes are the end goal in this project, it will be because of predicted animal locations allowing for more efficient patrol planning instead. By utilising a dataset of GPS data related to elephant movements instead of using past ranger patrol data, an alternative is given to reserves that may only have past animal movement data available to them.
- Reliance on geographical features to have a sufficient prediction means reserves which lack geographical information in a machine-readable format may not be able to make use of such methods. By basing the models used in this project on GPS and temporal data instead, it is made accessible to such reserves.

Papers the project is based on

As the project is dealing with time series and continuous data, regression models are more suited for use in this project. Therefore, papers that were documented to work with similar data as well as similar regression techniques were looked at to get an idea on how to approach the task.

One such paper [27] converted a non-stationary data set comprised of stock prices to make predictions on future closing prices, through the use of various regression models, including an Vector Autoregressive (VAR) model. By calculating the trend and seasonality of the data, they were able to subtract it from each data point – allowing accurate predictions to be made due to a now uniform spread of data, with a constant variance.

Despite the lack of spatial data, the methods used can be applied to this project. Precautionary steps will be taken to convert the time series data provided until stationarity is present. As VAR models base predictions off previous points, it is a suitable model that will be explored further.

In order to demonstrate the achievement of the aims, this project will explore the data in the hopes of finding relationships that will allow machine learning models to make accurate predictions on the location of elephants in the future.

Specification & Design

The aims of this project are to be able to explore the data with the intention of helping law enforcement to better understand how elephants move so they can be located easily, and therefore be able to make it easier to organise patrols. The approach that is being taken by this project is to apply the provided dataset to multiple machine learning models in the hopes of getting predicted GPS locations that will assist rangers in patrol planning. Thus, data processing and cleaning, along with visualisations, must be performed on the data to ensure results are reliable and representative of the data given.

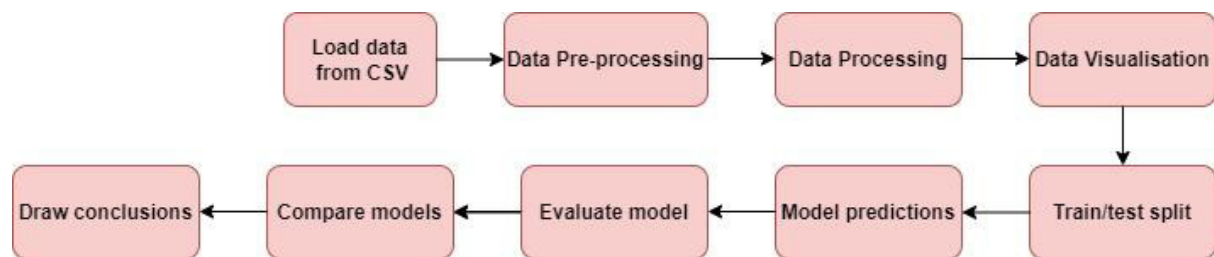


Figure 4 - Steps required for a finalised machine learning model

DGFC dataset

The datasets provided by DGFC have multiple variables with varying data types:

ID

An identifier that is unique for each elephant. Stored as an object.

Local Date

The current local date in Sabah, Malaysia when the GPS collar collects its readings. Stored as an object.

Local Time

The current local time in Sabah, Malaysia when the GPS collar collects its readings. Stored as an object.

GMT Date

The current GMT time in Sabah, Malaysia when the GPS collar collects its readings. Stored as an object.

GMT Time

The current GMT time in Sabah, Malaysia when the GPS collar collects its readings. Stored as an object.

Latitude

The latitudinal coordinate of the elephant in question at the current data collection instant. Stored as float64.

Longitude

The longitudinal coordinate of the elephant in question at the current data collection instant. Stored as float64.

Temperature

The estimate temperature of the elephant in Celsius at the current data collection instant. Stored as float64.

External Temperature

The external temperature around the elephant in Celsius at the current data collection instant. Stored as int64.

Activity

Binary field to see if the animal has moved or not. Stored as int64.

True Speed

The speed of the elephant at current data collection instant. Stored as int64.

Direction

The direction that the elephant is travelling at the current data collection instant. Stored as int64.

Altitude

The altitude of the elephant in metres at the current data collection instant. Stored as object.

Distance

The distance travelled from the previous data collection instant to the present collection, in metres.

Language and libraries used

Python

Python [28] is a popular programming language that available to be installed with Windows, Linux, and Mac OS. Due to its extensive documentation and ease of use, Python is a popular language amongst the AI community. The following Python libraries below will be used in conjunction:

Pandas

Pandas [29] is a software library written for the Python programming language for data manipulation and analysis. It allows csv files to be read and loaded in to a 'dataframe' object – essentially an ordered two-dimensional table consisting of rows and columns. Pandas is particularly helpful, as it automatically detects column names with their corresponding rows, meaning time does not have to be used reformatting data that loads incorrectly. Additionally,

having all the data stored in a dataframe object means manipulating the data is as simple as adding the equated function to the end of the dataframe name, separated by a period.

NumPy

NumPy [30] is a library for the Python programming language, allowing multi-dimensional arrays and matrices to be utilised in a straightforward manner, along with a large library of high-level mathematical functions to operate on such arrays. It was of particular use when performing numerical methods such as the regression methods and box plot tests, as well as calculating root mean squared error (RMSE).

MatPlotLib

Matplotlib [31] is a plotting library for the Python programming language and NumPy. It provides an object-oriented API for embedding plots much more easily than previously. It allowed visualisations to be plotted for results from the various tests ran throughout the project.

Seaborn

Seaborn [32] is a Python data visualization library based on matplotlib. It provides an interface for drawing statistical graphs – in this case, it was used for box plots.

StatsModels

Statsmodels [33] is a Python package that allows users to explore data, estimate statistical models, and perform statistical tests such as Augmented Dickey-Fuller and seasonal decomposition tests. The vector autoregressive model used was implemented from this library.

Bokeh

Bokeh [34] is a Python library for interactive visualization that targets web browsers for representation. It was used in conjunction with a Google Maps API to plot GPS data on satellite images, with the options of colour coding different data sources on the map.

SkLearn

Scikit-learn [35] (formerly scikits.learn and also known as sklearn) is a free software machine learning library for the Python programming language. Both the linear and polynomial regression models used in this project were from this library, as well as the time series train/test split method.

Google Maps JavaScript API [36]

Allows satellite images to be customized using JavaScript, making it compatible with BokehJS – both combining to be the central point of visualisation. GPS values will be plotted on these satellite maps and used as an evaluation method.

Data pre-processing

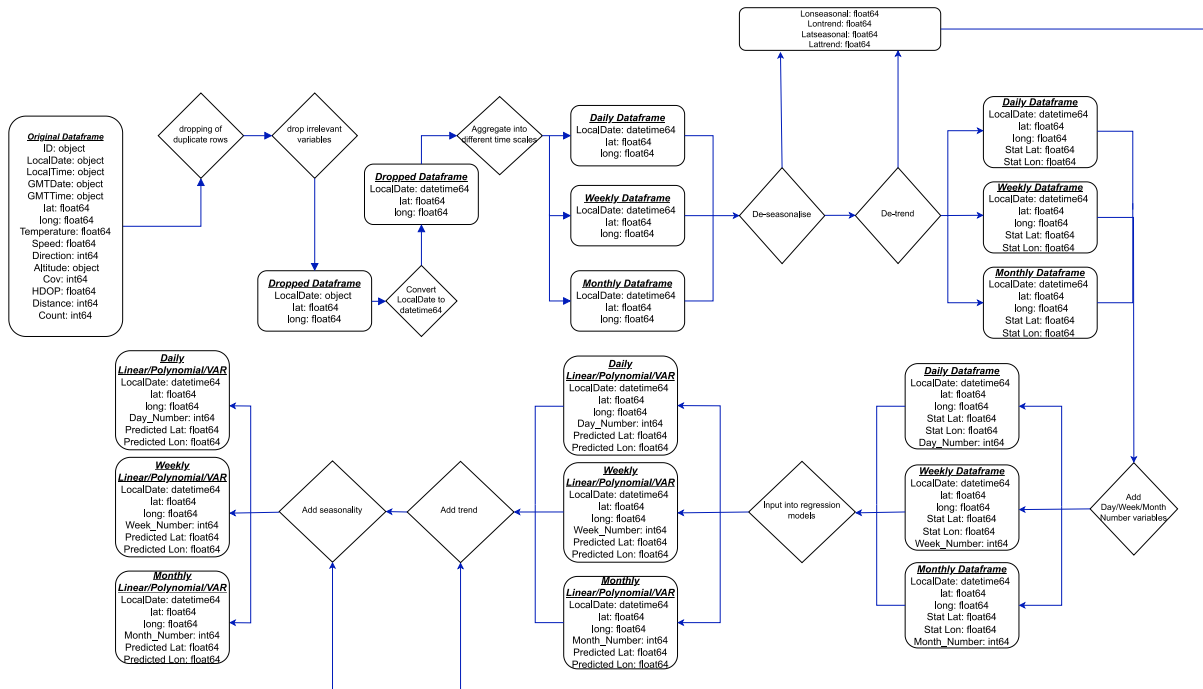


Figure 5: data flow diagram

Selection of primary dataset

Whilst the project was initially intended to use multiple datasets to train and test a prediction model, many of the other GPS data sources from other elephants had to be excluded due to poor data quality. This included missing/null column names and values, as well as inconsistent units for variables between different data sets. Having values that are missing or null can cause errors to occur during the runtime, meaning replacing said values is crucial. Furthermore, it can be very time-consuming to get data with missing column values and inconsistent units into a readable format that pandas data framing can use without issues.

Whilst there were several datasets that did not suffer from these issues, the differing geographical features meant combining multiple datasets to train and test a model was not feasible. This is because of the multitude of variables that our model could not account for in each location having too much of a varying effect on the regression techniques used.

For these reasons, the Seri dataset was selected to be the focus of the project, due to it being the dataset with the greatest timeframe of recorded data. This will allow for more refinement of the model when training, thus giving stronger predictions in the testing phase.

Duplicate records

Duplicate records can occur when the GPS collar malfunctions, causing data collection to occur multiple times. This can shift the mean value and regression lines towards such values, causing a bias in method. Predictions would be weighted towards these values, potentially leading to incorrect conclusions being drawn.

Visualisations would also be negatively affected. The natural migratory behaviour of elephants is to use natural corridors, therefore duplicate results may lead to rarely used paths being erroneously labelled as points of interest. Points of interest that would be included in future patrols – going against the primary aim of making patrols more efficient.

The panda's function will check these columns in every row, and list all the ones that have duplicates. It would then delete all the duplicates, except the first entry.

Outliers

Outliers can be caused by several things with some of the most common reasons being human-animal conflict and deforestation. This is because it can lead to natural migration corridors no longer being available, potentially causing them to migrate in an unexpected manner. Weather conditions could also affect it - if a season is much drier than usual, then elephants may migrate and stay close to rivers and such, an aspect that cannot be accounted for due to the lack of geographical data available for use.

Outliers can negatively impact the results and could potentially lead to the wrong model being selected, as regression lines could be weighted towards such values. This would in turn give predictions that are inaccurate and unrepresentative of the true data.

Whether an outlier is still included or not depends purely on if it is feasible or not. It is easy to verify the validity of the data points because of how frequent the GPS readings are and because the trajectories can clearly be tracked in sequential order. Plotting data onto satellite images allows large jumps between adjacent points to be spotted easily and accounted for.

To test for outliers, NumPy will be used to calculate percentiles, whilst a box plot function imported from the seaborn library will be used to visualise. Designing queries with the calculated percentiles will allow outlier rows to be returned.

Dropping of variables

Whilst the initial raw Seri dataset had a range of different variables, it is likely that a majority of them are fields which hold no relevance to the designed models. Columns with empty values can cause errors to occur during regression, whilst irrelevant variables can cause coefficient estimates to be less precise, subsequently reducing the effectiveness of the model.

Furthermore, having many variables can cause regression models to 'drift', and make it much more likely to be overfitted. This could potentially cause the results to show trends that do not exist.

For these reasons, the pandas' functions will be used on the data frames to check for empty values, which will be removed if present. The rest of the columns will be evaluated based on their relevance to the aim of the project.

Data Processing

Averaging – different time scales

To add more depth to visualisations and predictions, three different time scales were selected to be used: the original timeframe, with readings every two to six hours, as well as the weekly and monthly average timeframes.

A possible problem with plotting average GPS values is that the elephant may move erratically within a single week or month. As a result, the average point could look as if the elephant has been within the dense vegetation, when it would have been using the natural corridors surrounding that space. Furthermore, visualisations using the original timeframe may have larger jumps in movement than expected due to inconsistent GPS collection intervals. However, these issues should have a miniscule impact on predictions and visualisations regardless.

Before pandas functions could convert and average the data, *Local Time* would have to be changed to *datetime64* datatype – the original *object* datatype not being eligible due to its string-like behaviour causing errors.

De-seasonalising and de-trending

The use of non-stationary data alongside machine learning methods is a recipe for disaster, with prediction outputs that are particularly capricious since they are not independent. The regression methods utilised in this project rely on the assumption that each predicted point is autonomous of one another, thus the data would need to be transformed into a stationary format. An ADF test on the data would verify these concerns.

To do so, the *seasonal_decompose* function from *StatsModel* will be applied on the latitude and longitude values for each timeframe separately to avoid complications. The seasonal and trend values for each row will be returned and saved into separate variables depending on the timespan. These seasonal and trend values will be subtracted from their respective columns – the resultant values being saved as new columns in the Data Frames, aptly named ‘*Stat Lon*’ and ‘*Stat Lat*’. A precautionary ADF test and *pyplot()* on these new columns will verify their stationary status before further use.

Adding week number and month number

Due to requirements for an integer or float datatype for regression with *SkLearn* and *StatsModels*, a new variable would have to be defined for the number of days, weeks, or months for each Data Frame instead of using the datetime index. *Pandas*’ functions allow these values to be taken from the index and saved into new variables.

Data Visualisations

The data now sufficiently cleaned, visualisations were the key to unearthing any potential trends within the data that would potentially yield results that were more accurate and reliable. With natural forest corridors being the primary migration routes utilised by elephants, the most probable method of pinpointing these paths was to plot the GPS points on satellite image.

Basic visualisations revealed seasonal trends in elephant movement routes, with the same natural corridors being used frequently over the time period. For this reason, machine learning models were selected as the method to harness these trends and produce GPS predictions.

Machine learning

To make sure the most effective machine learning model is selected, numerous different models have been tested against one another. The metric for differentiating between various prediction methods is the root mean squared error – which is the standard deviation of how far the predicted points are from the regression line.

Overfitting

However, picking a model is not as simple as choosing the one with the smallest RMSE. A common problem when fitting regression models is overfitting – when the finished model performs well on the training data set, but performs significantly worse when introduced to new, unseen data. To stop this from occurring, forward chaining cross validation will be used to split the data into segments.

The standard procedure with machine learning models is to simply split the data into training and testing data in a 70%/30% split. However, with the nature of time series, future data cannot be used to predict past data. Especially regarding GPS data, as patterns that may occur in future data may be picked up by the model and used to make predictions on earlier data. This can result in skewed data that has a bias towards these future values. Furthermore, in a real-world scenario where the finished model is used, there will only be past data to draw conclusions from, so to use an inexperienced model that has not been trained in this manner would not be suitable. This is also why k-folds cross validation split is not applicable.

Forward chaining cross validation is unique compared to other methods, as only past data is used to train the model to make predictions. Across all folds, the RMSE score should display consistency if there is no overfitting present.



Figure 6: diagram explaining the forward chaining process

Selected models

With the nature of time series data, several different regression models were used with the dataset. Regression models were chosen to be used as they are particularly useful in this scenario where a continuous dependent variable is being predicted from a few independent variables. Furthermore, with how erratic GPS data can be, selecting a model to use is not as straight forward as choosing the most complex ones. Simple models have been shown to perform just as well, the lower complexity often meaning a lower runtime as well as a base to branch off.

The first regression model to be used was linear regression – the most basic regression model available. The results are not expected to be as accurate as other regression models due to the unlikelihood that the relationship between latitude/longitude and temporal data displays linearity. However, linear regression is still an effective intermediate step on which to draw conclusions off in selecting other models as it is easily interpretable and is a frequent basis for other regression methods, such as the following ones used.

The second regression model used was polynomial regression. As polynomial regression is nonlinear, the RMSE is expected to be lower than the one obtained from linear regression. The relationship between the variables is also more likely to be nonlinear. Whilst linear regression does not have any parameters to adjust, selecting a degree that the polynomial model will have can affect results massively as each regression line drawn will differ massively from one another. A well-documented problem to be careful of when using polynomial regression is that increasing the degree can give the impression of the RMSE dropping lower. But in reality, the regression line is plotting itself to follow the course of the training data as finely as possible. This can be tested by using the model on unseen data and seeing if the RMSE is greater than seen in the previous phase.

A Vector autoregressive model was chosen as the final regression method as it captures the relationship between different time series and how they influence one another – using this relationship to make predictions based on previous values. This makes VAR a multivariate method in contrast to the last two univariate models, which could potentially give a different result's perspective. Selecting the optimal lag order is crucial to prevent autocorrelation from occurring which can cause forecasted values to being higher or lower than expected.

Evaluating models

Each model will be evaluated separately based on the RMSE score it has for each timeframe. RMSE is the choice over others such as mean absolute error since it is the average difference between the predicted value and its actual value. This means that it has the same units as the predicted variable, making it more interpretable. RMSE is also a quadratic scoring metric, meaning outliers also have a large weight – a desirable trait as vastly inaccurate predictions could have serious repercussions in the project's problem area.

However, with the nature of GPS data, numerical data cannot be the only evaluation metric. After all, predictions can potentially have a low RMSE score but demonstrate a movement trend that is inconclusive and incorrect.

For this reason, predicted GPS values will be plotted on satellite images along with the true GPS test values to see how closely the predictions mirror the elephant movement trends.

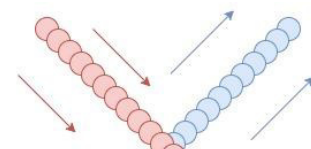


Figure 7: diagram demonstrating a prediction (blue) which has a low RMSE from the actual values (red), but a poor movement trend forecast.

Comparing models

The average RMSEs' of latitude and longitude for each separate timeframe and model will be compared with one another, along with the respective visualisation plots. The model that performs the best along with the satellite image plot that correctly predicts the movement trend will be selected.

Implementation

Data Preprocessing

Reading of CSV files and removal of duplicates

```
df = pd.read_csv('Seri.csv')
df = df.drop_duplicates(subset=['Local Date', 'Local Time', 'Lat', 'Lon'], keep='first')
```

Figure 8: the importing of CSV files into a data frame and the dropping of duplicate records

The Seri.csv file has its column names and row values processed and put into a DataFrame called 'df'. It is then dropped of all rows which share the same *Local Date*, *Local Time*, *Latitude* and *Longitude* - only the first row being kept.

Checking for 'dummy' columns

```
print((df['True Speed (km/h)']>0).value_counts())
print((df['Activity']>0).value_counts())
print((df['Dir']>0).value_counts())
print((df['Ext Temp (deg C)']>0).value_counts())
```

Figure 9: Tests for values above zero in potential column drop candidates

This code checks the potential drop candidates for values above zero.

Dropping of variables, the changing of index, datetime conversion

```
dropped_df = df.drop(['GMT Date', 'GMT Time', 'Local Time', 'ID', 'Temp (deg C)', 'Ext Temp (deg C)', 'Activity', 'True Speed (km/h)', 'I
dropped_df['Local Date'] = pd.to_datetime(dropped_df['Local Date'], dayfirst=True)
dropped_df.index = dropped_df['Local Date']
dropped_df.drop('Local Date', axis=1, inplace=True)
```

Figure 10: the dropping of variables, along with data type changing for the new index

Unwanted variables are dropped from the DataFrame using pandas 'drop' function, followed by *Local Date* being converted into datetime64 datatype before being defined as the index of the DataFrame – the duplicate column subsequently being removed from the table.

For the machine learning algorithms, the exact same format is followed, whether it is weekly or monthly, longitude or latitude – only variable names are changed.

Outlier Testing

```
LonLQ = np.percentile(df.Lon, 25)
LonUQ = np.percentile(df.Lon, 75)
LonIQR = LonUQ - LonLQ
```

Figure 11: Calculating of the lower and upper quartile, and the interquartile range

Lower and upper quartiles were calculated using NumPy, which was then used to calculate the interquartile range.

```
df_HighOutlierLon=df['Lon']>=LonUQ + (1.5*LonIQR)
filtered_df = df[df_HighOutlierLon]
print(filtered_df)

df_LowOutlierLon=df['Lon']<=LonLQ - (1.5*LonIQR)
filtered_df = df[df_LowOutlierLon]
print(filtered_df)
```

Figure 12: Queries for finding outliers in the data

The previously saved variables were used as query filters, the `filtered_df` returning all outliers in the data frame.

Data processing

Creation of Data Frames for time frames [38]

```
weeklydf = dropped_df.resample('W').mean()
weeklydf['Week_Number'] = weeklydf.index.week

weekly2016df = weeklydf[weeklydf.index.year == 2016]
weekly2017df = weeklydf[weeklydf.index.year == 2017]

monthlydf=dropped_df.resample('M').mean()
monthlydf['Month_Number'] = monthlydf.index.month

monthly2016df = monthlydf[monthlydf.index.year == 2016]
monthly2017df = monthlydf[monthlydf.index.year == 2017]
```

Figure 13: the creation of new tables for different time frames

A code segment which showcases the mean averaging of the dropped Data Frame into new tables for days, weeks, and months. The `resample()` function paired with `mean()` calculates the average across each of the row's within each chosen interval, the letters within `resample()` corresponding to a daily, weekly and monthly frequency. As the *Local Time* index is in datetime format, each part of the date can be accessed and taken as an integer, the weekly and monthly

numbers being saved as new columns in their respective Data Frames. Separate Data Frames were created for 2016 and 2017 – both with a full year’s worth of data.

Calculating seasonal and trend values

```
result = seasonal_decompose(dropped_df['Lon'], model='additive', period=48)

dailyLontrend = result.trend
dailyLonseasonal = result.seasonal
```

Figure 14: the calculation of seasonal and trend values

The `seasonal_decompose` function is called from `statsModels`, which allows new variables to be made for the respective trend and seasonality from the input column.

Converting to a stationary time series

```
dropped_df['Stat Lon'] = dropped_df['Lon']
dropped_df['Stat Lon'] = dropped_df['Lon'] - result.trend
dropped_df['Stat Lon'] = dropped_df['Stat Lon'] - result.seasonal
```

Figure 15: Conversion to stationarity

Previously mentioned variables are subtracted from the longitude – a new column being created that holds the stationary values.

Basic GMap visualisations [39]

```
Lat, Lon = 5.135470, 118.507174
def plot(lat, lng, zoom=10, map_type='roadmap'):
    gmap_options = GMapOptions(lat=lat, lng=lng,
                               map_type=map_type, zoom=zoom)
```

Figure 16: the creation of a GMap plotting function

Code shown for the creation of satellite image visualisations. A GPS coordinate is given to be the centre point of the figure, which is used as the input for the plotting function defined, along with the default zoom and image type on the generated image.

```
hover = HoverTool(
    tooltips = [
        ('Date', '@{Local Date}{%c}'),
        ('Lat', '@Lat'),
        ('Lon', '@Lon'),
    ],
    formatters={'@{Local Date}': 'datetime'}
)
```

Figure 17: Creation of a HoverTool for GMap

A hover tool is defined for the figure that allows local date, latitude, and longitude to be displayed when a cursor is placed over a data point, which is defined in the *tooltips* section - the label shown on the left, the column name containing the values on the right - selected using `@`. The use of `{}` and `%c` allowing the Local Date to be acknowledged. The formatter section passes the corresponding format to Bokeh.

```
p = gmap(api_key, gmap_options, title='Malaysia',
        width=bokeh_width, height=bokeh_height,
        tools=[hover, 'reset', 'wheel_zoom', 'pan'])
source = ColumnDataSource(weeklydf_test)
center = p.circle('Lon', 'Lat', size=4, alpha=0.5,
                  color='yellow', source=source)

return p
```

Figure 18: defining the properties for the GMap figure

The satellite image figure is created, with the previously shown options. The data is defined as a data source – the properties of each circle plotted from the source being established: the column names for the longitude and latitude values being given, along with the size, the opacity, colour and data source being named. The final figure is then returned.

```
source1 = ColumnDataSource(data = dict(latitude = weeklydf_forecast['Lat'].values, longitude = weeklydf_forecast['Lon'].values))
p.circle(x = "longitude", y = "latitude", width = 18800, size = 4, color = "red", fill_alpha = 0.5, source = source1)
show(p)
```

Figure 19: the plotting of a different data source onto a GMap

With the figure already defined and created, a new data source, if needed, shows different points on the graph – the latitude and longitude being defined in a dictionary with the corresponding dataframe. The circle is drawn on the satellite image with selected characteristics.

Whilst in this case, the satellite image plots predicted GPS points from the regression models against the actual values, the exact same format was followed for prior visualisations.

Machine Learning

Linear Regression [40]

The process for making predictions on a test set with linear regression is shown above.

```
day = dropped_df['Day_Number'].values.reshape(-1, 1)
long = dropped_df['Stat Lon'].values
```

Figure 20: the defining of Linear Regression inputs

Before the *week number* and *longitude* values can be used with the *LinearRegression()* function imported from *SkLearn*, the week number is converted into a 2-d array.


```
tscv = TimeSeriesSplit(n_splits=5)
for train_index, test_index in tscv.split(day):
    day_train, day_test = day[train_index], day[test_index]
    long_train, long_test = long[train_index], long[test_index]
```

Figure 21: train/test data time series splitting

The number of splits is chosen, the 'for' loop iterating through each data split. Depending on the split number, a different proportion of the data is used in the training/testing sets through indexing, until all the data is being used in the final fold.

```
linear_regressor.fit(day_train, long_train)
long_pred = linear_regressor.predict(day_test)
linear_reg_rmse = np.sqrt(mean_squared_error(long_test, long_pred))
print(linear_reg_rmse)
```

Figure 22: fitting of training data into a Linear Regression model, subsequent predictions and RMSE score

Training data is fitted with the linear regression model, the fitted model then being used to make predictions. The RMSE between the prediction and the true value is calculated and printed.

```
dailyLinearPredictions = dropped_df.iloc[:,[0,1]].tail(len(day_test))
dailyLinearPredictions['Predicted Lon'] = long_pred
```

Figure 23: creation of Linear data frame to store predictions

A new data frame is made – the *iloc* function taking the latitude and longitude values from *dropped_df*. The 'Predicted Lon' column is made from the previous predictions.

This exact layout is copied for each time frame, with slight variable name changes.

Polynomial Regression [41]

The code below shows the process for polynomial regression.

```
week = weeklydf['Week_Number'].values.reshape(-1, 1)
long = weeklydf['Lon'].values

poly = PolynomialFeatures(degree=4, include_bias=False)
week_poly = poly.fit_transform(week)
pol_reg = LinearRegression()

tscv = TimeSeriesSplit(n_splits=5)
for train_index, test_index in tscv.split(week_poly):
    week_train, week_test = week_poly[train_index], week_poly[test_index]
    long_train, long_test = long[train_index], long[test_index]
    pol_reg = LinearRegression()
    pol_reg.fit(week_train, long_train)
    long_predicted = pol_reg.predict(week_test)
    poly_reg_rmse = np.sqrt(mean_squared_error(long_test, long_predicted))
    print(poly_reg_rmse)
```

Figure 24: Polynomial regression model

Much of the process for fitting a polynomial regression model is like the one followed for linear regression. The difference being that in order for the model to function, *week_number* is fitted with *PolynomialFeatures()*, which converts it into a polynomial array of the specified degree, and *include_bias = false* leaving out the polynomial result of degree 0.

Vector Autoregressive model [42]

Code showing the fitting of a VAR model before it is used to forecast future GPS positions.

```
obs = 18
week_train, week_test = weeklydf.iloc[:,[3,4]][0:-obs], weeklydf.iloc[:,[3,4]][-obs:]
```

Figure 25: VAR selected no. of predictions, with subsequent data split

The desired number of predictions is chosen, which is also used to split the ‘Stat Lon’ and ‘Stat Lat’ values taken from the *weeklydf*.

```
model = VAR(week_train)
modelCriteria = model.select_order()
print(modelCriteria)
model_fitted = model.fit(2)
```

Figure 26: VAR fitting of model with selected lag order and training data

The training set is added to the model, the lag order being selected based on results from *.select_order()*. A lag order of two is used to fit the model.

```
forecast_input = week_train.values[-2:]

fc = model_fitted.forecast(y=forecast_input, steps=obs)
weeklyVARPredictions = pd.DataFrame(fc, index=weeklydf.index[-obs:], columns=['Predicted Lon', 'Predicted Lat'])
weeklyVARPredictions['Lat'] = weeklydf['Lat'].tail(len(week_test))
weeklyVARPredictions['Lon'] = weeklydf['Lon'].tail(len(week_test))
```

Figure 27: VAR forecast input, results and subsequent dataframe to store VAR results

The forecast input is selected from the last two values in the training set. The fitted model makes several predictions based on the training data, input values and number of predictions selected. The following is put into a new Data Frame, which then has the original *Lat* and *Lon* values added to it.

Implementation limitations

Looking back at the initial plan timetable, the difference between theory and reality is startling. Various problems throughout the development cycle of this project arose that took long periods of time to move past.

A disproportionate amount of time was spent reading numerous different research papers about predicting animal movements and trajectories, in the search for a paper that utilised similar regression methods. However, there were a severe lack of information on the subject. To combat this issue, the search scope was widened – even if papers used different methods, the general approach was noted to see if it could be applied to my own project.

Going into the project, my initial idea was to use the correlation that temperature and the time of year has with latitude and longitude to train the machine learning models to make more accurate predictions. Considering the impact that temperature has on an elephant’s migration path, I felt

like this paired with detailed land survey data could be utilised to develop an effective model. However, I quickly realised this was not plausible.

Firstly, the temperature in Borneo is consistent throughout the year, meaning that with the data available, there does not seem to be enough of a deviation from the average temperature each year to cause a change in an elephant's migration route. An Initial plot of temperature against latitude and longitude separately, seemed to show no correlation between the variables. Sequential months would be on opposite sides of the graph.

Secondly, searching for detailed land survey information on Seri's surrounding area had no success. Maps and satellite images were available, but not detailed to the point where it could be implemented into Python as usable data. This was particularly an issue, as water sources have a large influence on movement paths, which is why it is such a common variable in countless different papers modelling animal movement. Whilst satellite images are still helpful in visualising the animal routes, as the large tree canopies block the forest floor, there is no way to see the different vegetation densities present throughout the rainforest. This makes it difficult to see natural footpaths that are commonly used by elephants when the plant density is low.

The datasets provided also had several different issues that made it difficult to generate accurate predictions.

An inconsistent data collection interval on the GPS collars meant the elephant would appear to have a sudden change in position that was a cause for concern, until further investigation saw that whilst most of the time, data readings were collected every two hours, various problems with the GPS coverage meant that readings were not able to be collected until signal improved enough. Average GPS readings were affected, as when calculating the mean values for different time scales (e.g., days, weeks, months), the difference in the number of observations is not considered. Within the whole dataset however, the number of occurrences was not great enough to warrant time spent fixing it.

Furthermore, there was a problem with the amount of relevant data. Whilst there were many different elephants, each with their own corresponding dataset to use, a majority were only fitted with GPS collars for an amount of time that made it difficult to use machine learning models. Over half of the datasets have under two years' worth of data, with a few having less than 6 months' worth. Nine of the datasets also were not able to be used due to their poor quality, each having their own issues whether it was outliers, missing/null data, missing column names and more.

As all machine learning models rely on good training data to make quality predictions, having datasets that have readings for two and a half years at most makes it difficult to find reoccurring trends that can be used to improve the model. With a larger dataset, visualisations on the satellite image could be refined, with migration corridors being more apparent. It would also allow for more training data, giving a greater degree of accuracy.

At first, to show a visualisation of the data, MovingPandas was used, a library which allows GPS points to be plotted as connected trajectories. Whilst helpful, the lack of documentation due to the development team consisting of one person meant some time was wasted on troubleshooting. It also lacked the definition desired for GPS visualisations. To solve this problem, a Google Maps API was obtained, which with Bokeh, was used to plot predictions and other values.

Overly ambitious project aims were also an issue, as being able to get predictions that are accurate enough to be useful was more difficult and time consuming than intended. This is partially because of a mistake in initial ADF testing, which led to the assumption that the dataset was already stationary. As expected, predictions were wildly inaccurate – an issue which I spent many hours trying to get to the root of. Performing the test correctly revealed that the dataset needed to be converted to stationarity. After which, the predictions were far more precise.

Results & Evaluation

The aim of the project has been achieved, the refined regression models all performed admirably on unseen data, with low RMSE scores across the board. Furthermore, plotting of predicted GPS values with actual values onto satellite images displayed undeniable evidence that the future movement trend of Seri was successfully modelled. Whilst many models have been curated to predict future movement trajectories, there are few that utilise simple regression machine learning models to anticipate the movement of animals. This project hopes to lay a foundation on which those more knowledgeable can further refine.

Data Pre-processing

Outlier test

Using the methods displayed in the previous implementation section, queries were run on the latitude and longitude columns. Only empty data frames were returned, meaning that there are no outliers present through the Seri dataset.

Duplicate rows test

The number of duplicate rows returned was 860.

Variable viability tests

Using the test methods displayed earlier in implementation, *Speed*, *direction*, *activity*, and *external temperature* returned zero counts for true, when queried for values above zero. These columns were dropped. *'Back to Catalog'*, a hyperlink, was incorrectly taken as a column name along with the empty column next to it, which pandas has left as *'Unnamed'* – all of which were subsequently dropped immediately.

Coverage and *HDOP* are metrics related to the transmission and measurement quality of the GPS collections, neither of them reaching values of which action will have to be taken. *Count* is simply the number of readings. All were dropped.

Whilst temperature was initially promising, the low variance in Sabah's temperature throughout the year meant its effect on elephant migration was negligible. This rendered it useless to the project.

Whilst the potential relationship with *distance* and *altitude* that had been displayed in previous works, that is only through alternate methods – the manner in which regression is being used in this project makes these metrics unneeded.

GMT Date and GMT Time are metrics for the UK; irrelevant to the problem area of Sabah, whereas including Local Time as an independent variable adds too much complexity to the predictions, as it requires the data to be in its original time scale, with readings every two-six hours.

ID is the unique identifier for Seri – irrelevant considering Seri is the only elephant in the dataset.

Data processing

Seasonality plots on original data

Daily

Lon

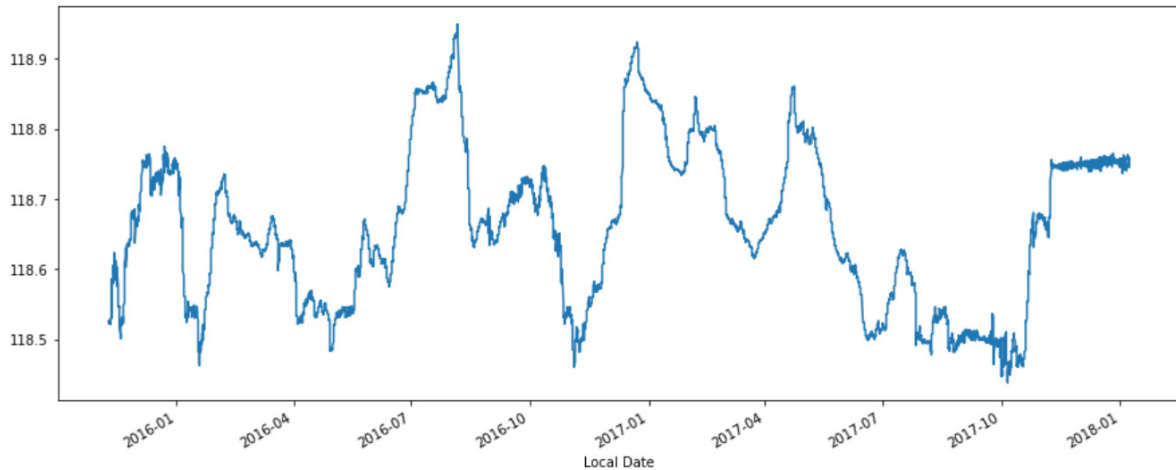


Figure 28: Longitude plot on original data

Lat

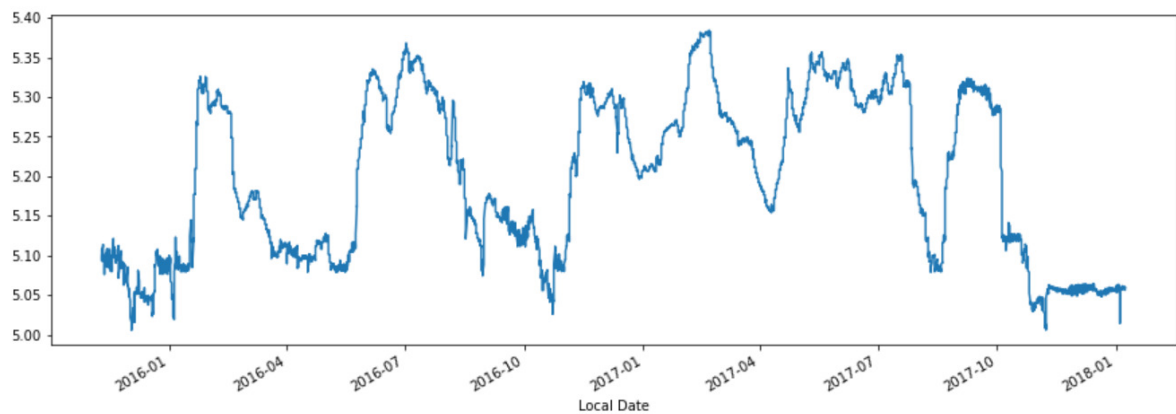


Figure 29: Longitude plot on original data

Whilst ADF tests are required for verifying that stationarity is not present, characteristics are displayed in the graphs above that help indicate. Periodic fluctuations are observed which are signs of seasonality. Furthermore, trends can be observed, with long-term increases followed by decreases.

ADF tests

Initiating a regression model on non-stationary data frequently yields results that are false and wildly inaccurate – in a phenomenon called ‘spurious regression’, a relationship in which multiple variables are associated but not causally related, often due to the presence of unknown factors. Stationarity is the basis assumption that formulates many statistical methods – without it, it is nigh impossible to distinguish clear trends. To combat this, Augmented Dickey-Fuller (ADF) tests were performed on the Seri data set in its initial form, for both latitude and longitude.

For a time-series data set to be considered stationary, test statistic < critical 5% value, and $p < 0.05$.

ADF Tests				
Longitude				
	Initial Dataset	Converted Daily	Converted Weekly	Converted Monthly
Test Statistic	-3.14	-18.77	-7.19	-1.54
P-value	0.02	0.00	0.00	0.51
Critical Value (5%)	-2.86	-2.86	-2.89	-3.07
Test Stat < critical value 5% ?	Yes	Yes	Yes	No
P-value < 0.05 ?	Yes	Yes	Yes	No
Latitude				
	Initial Dataset	Converted Daily	Converted Weekly	Converted Monthly
Test Statistic	-2.52	-20.30	-6.90	-4.80
P-value	0.11	0.00	0.00	0.00
Critical value (5%)	-2.86	-2.86	-2.89	-3.03
Test Stat < critical value 5% ?	No	Yes	Yes	Yes
P-value < 0.05 ?	No	Yes	Yes	Yes

Figure 30: ADF test results

Testing both latitude and longitude from the initial Seri dataset returned peculiar results – within the conditions of ADF, longitude was considered stationary and latitude was not considered stationary. On the contrary, the longitude plot displays seasonality and trends through the period – regardless of the results, which narrowly pass the ADF criteria. For this reason, it was in the best interest of the project to convert latitude and longitude data nevertheless, by subtracting the trend and seasonality from every value for each time period.

ADF tests were repeated on each converted dataset to ensure that stationarity was achieved. Daily and weekly datasets passed for both latitude and longitude by a greater proportion than seen previously, cementing their stationarity. However, the longitude monthly dataset failed the test, and latitude narrowly passed in comparison to daily and weekly results. It is likely that this is due to the small data set available.

Stationary plots

Daily

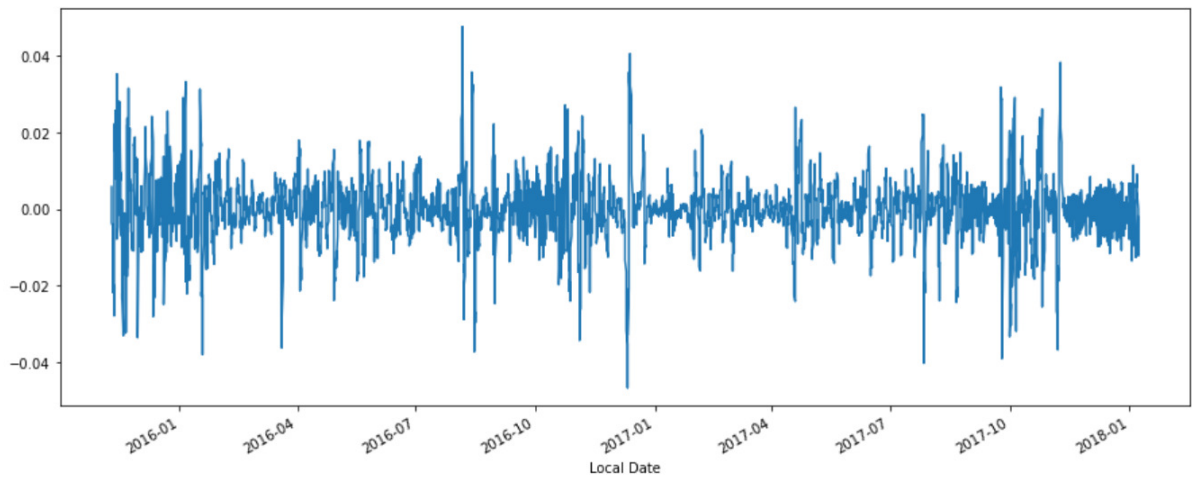


Figure 31: Stationary longitude plot for original data frame

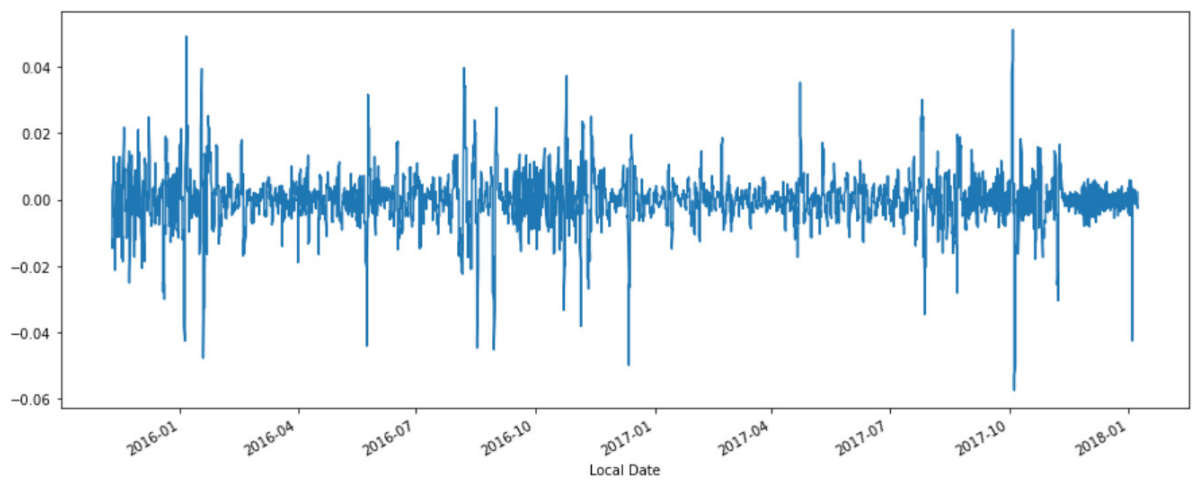


Figure 32: Stationary latitude plot for original data frame

Weekly

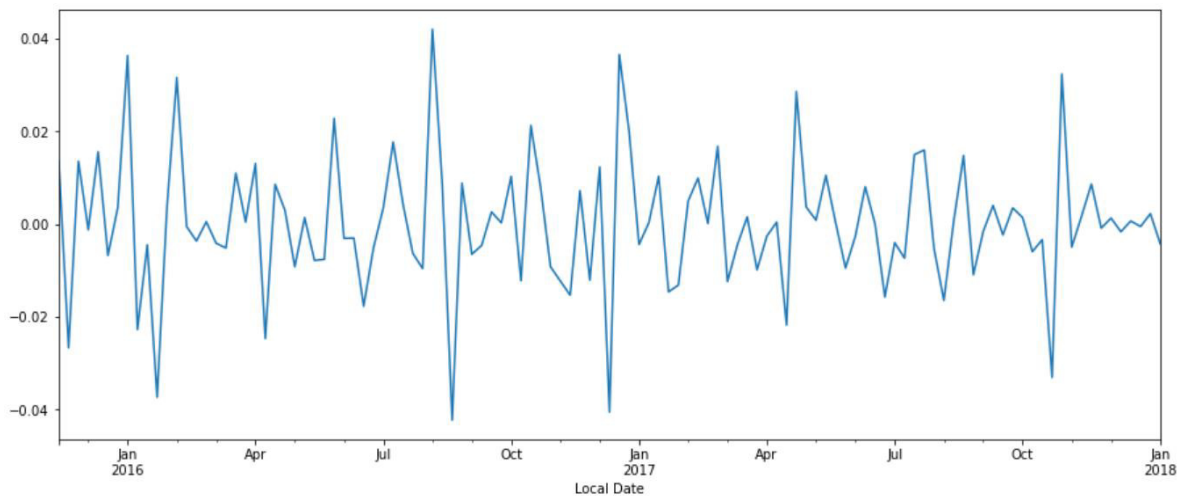


Figure 33: Stationary longitude plot for weekly data frame.

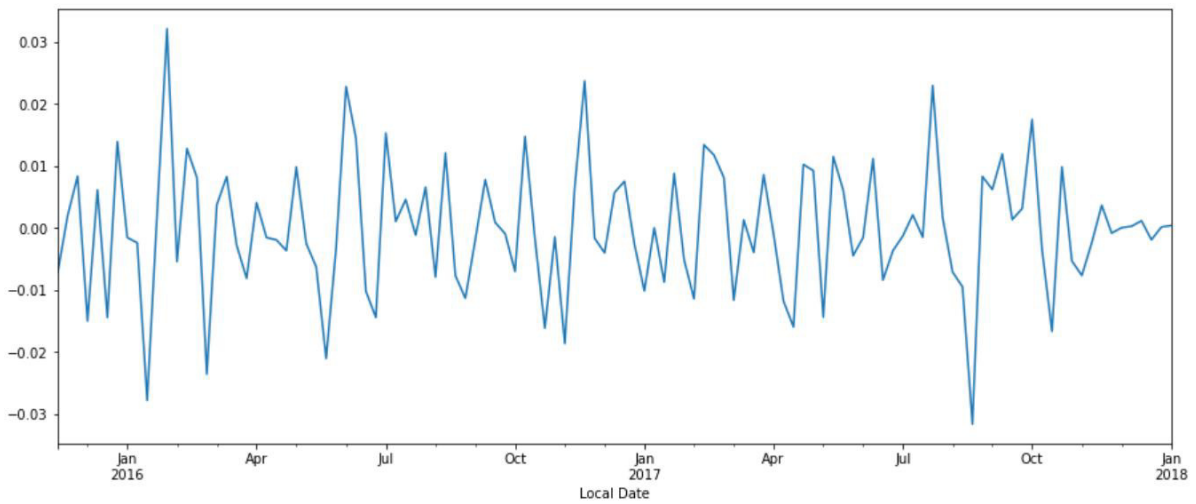


Figure 34: : Stationary longitude plot for weekly dataframe.

Monthly

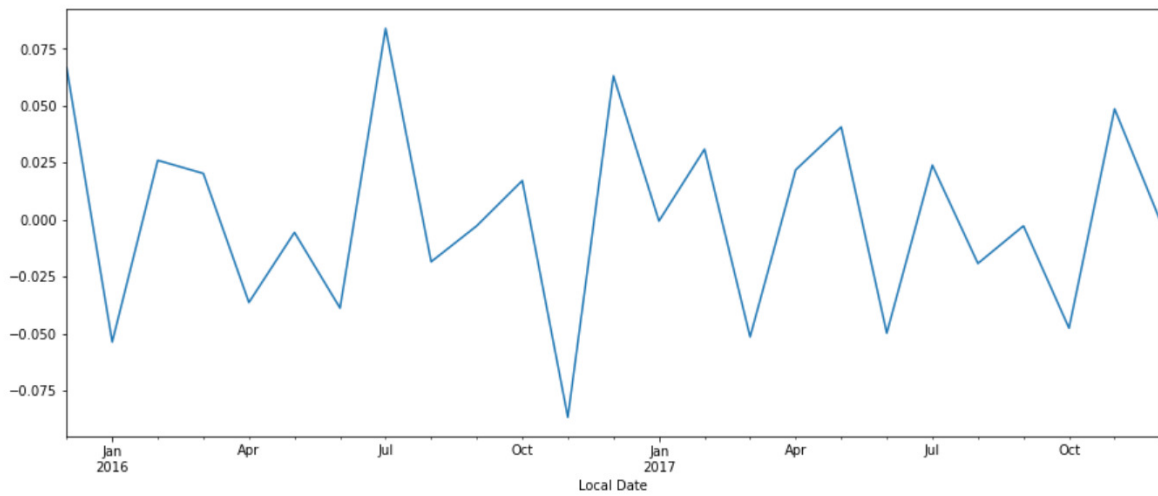


Figure 35: : Stationary longitude plot for monthly dataframe.

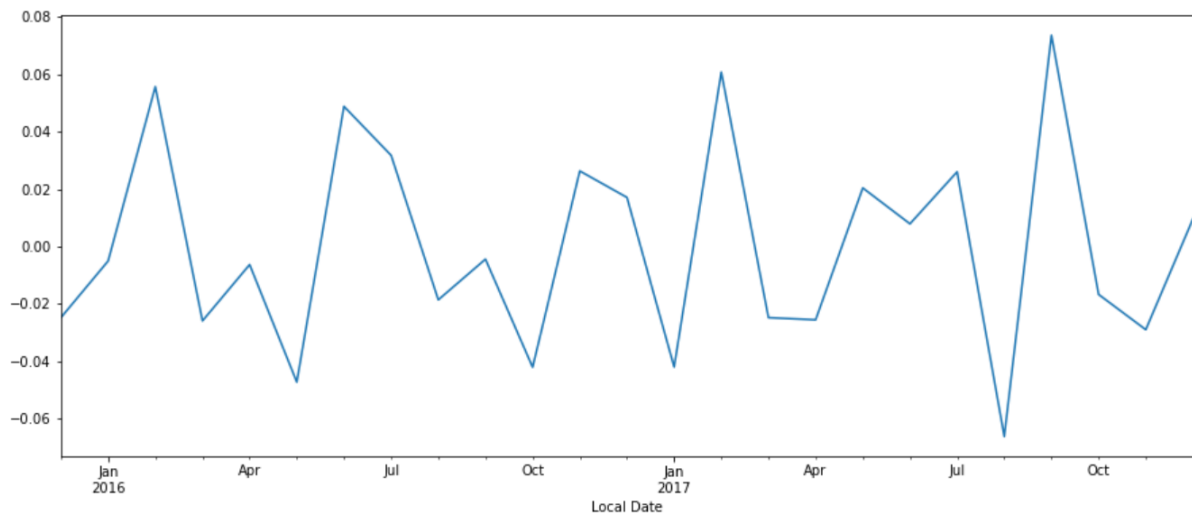


Figure 36: : Stationary longitude plot for monthly dataframe.

The graphs above confirm the ADF tests results. Daily and weekly timeframes show lack of trend, with a lack of variance and seasonality. Monthly graphs show more consistency than previously, but the lack of data makes the graph shape non-uniform.

Visualisations

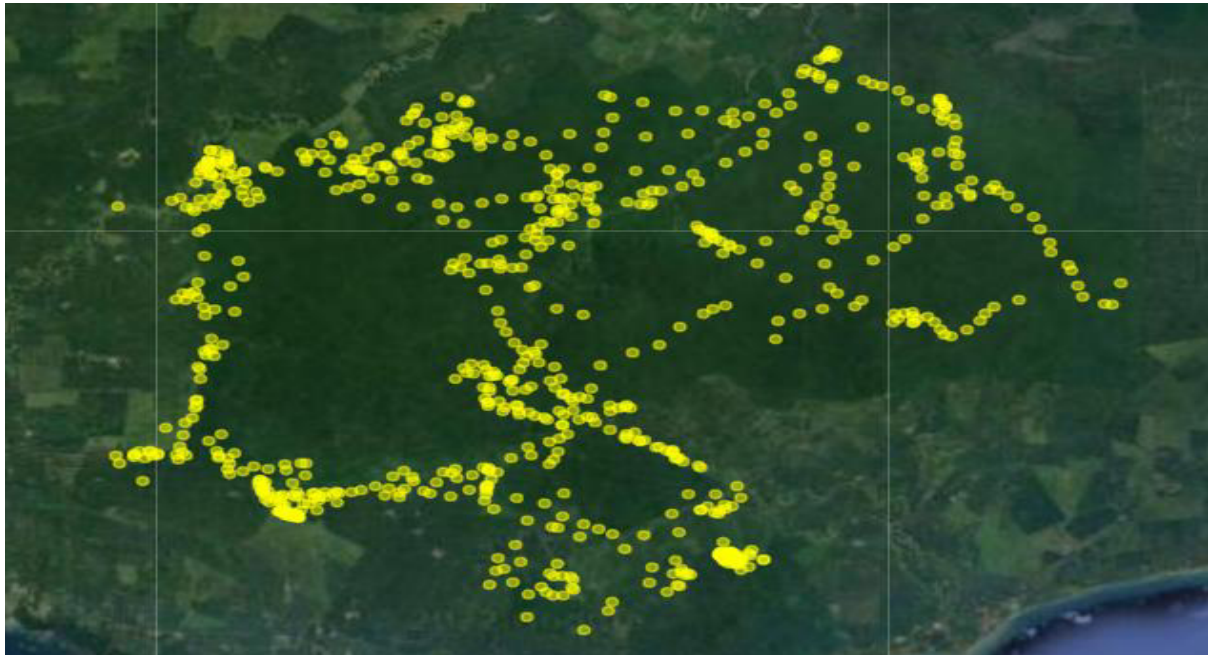


Figure 37: a satellite image displaying the GPS readings for original data frame.

A plot of the initial data reveals the presence of natural forest corridors used by Seri, that are used multiple times through the whole dataset.

Machine Learning

Time Series Split

To inhibit this occurrence, time series splitting was utilised to verify consistent results across each fold, and spot signs of overfitting.

Linear Regression - Daily					
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Long	0.006586448	0.008966769	0.006796756	0.007441409	0.008674107
Lat	0.005986261	0.010049156	0.004958771	0.006958466	0.007496143
Linear Regression - Weekly					
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Long	0.017088002	0.017172762	0.01220162	0.009823004	0.011538096
Lat	0.011145399	0.009994986	0.009473576	0.01152976	0.007344807
Linear Regression - Monthly					
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Long	0.048694786	0.069229029	0.034924836	0.037026136	0.035553441

Lat	0.057842513	0.045423578	0.046952192	0.060045685	0.055345159
-----	-------------	-------------	-------------	-------------	-------------

Figure 38: results of Forward Chaining cross validation - Linear Regression

Polynomial Regression - Daily					
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Long	0.018658087	0.00897438	0.00680566	0.007485135	0.008676975
Lat	0.006036533	0.010383775	0.004969315	0.006960374	0.007511941
Polynomial Regression - Weekly					
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Long	0.019240029	0.016776605	0.013190469	0.009760556	0.011682876
Lat	0.01206106	0.010028755	0.009544908	0.01152094	0.007428603
Polynomial Regression - Monthly					
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Long	0.048694786	0.069229029	0.034924836	0.037026136	0.035553441
Lat	0.057842513	0.045423578	0.046952192	0.060045685	0.055345159

Figure 39: results of Forward Chaining cross validation - Polynomial Regression

For all the different regression datasets, both longitude and latitude predictions produced results that were rational and within the realms of possibility for all folds, ensuring the validity of each fitted models' predictions.

Regression results

Linear Regression - Daily		
	RMSE	RMSE mean
Longitude	0.019576032	0.019331742
Latitude	0.019087451	
Linear Regression - Weekly		
	RMSE	RMSE mean
Longitude	0.032975793	0.032162699
Latitude	0.031349605	
Linear Regression - Monthly		
	RMSE	RMSE mean
Longitude	0.075648295	0.077809569
Latitude	0.079970843	

Figure 40: RMSE results - Linear Regression

Polynomial Regression - Daily		
	RMSE	RMSE mean
Longitude	0.019617198	0.019328578
Latitude	0.019039958	
Polynomial Regression - Weekly		
	RMSE	RMSE mean
Longitude	0.034488008	0.032919322
Latitude	0.031350636	
Polynomial Regression - Monthly		
	RMSE	RMSE mean
Longitude	0.078653204	0.079171446
Latitude	0.079689687	

Figure 41: RMSE results - Polynomial Regression

Vector AutoRegression - Daily		
	RMSE	RMSE mean
Longitude	0.019566871	0.01927508
Latitude	0.01898329	
Vector AutoRegression - Weekly		
	RMSE	RMSE mean
Longitude	0.032869618	0.033959793
Latitude	0.035049967	
Vector AutoRegression - Monthly		
	RMSE	RMSE mean
Longitude	0.076779802	0.081280196
Latitude	0.085780591	

Figure 42: RMSE results - Vector AutoRegressive

Overall, all the models performed similarly – the lowest RMSE across all models being the VAR performed on the original daily dataset. However, both linear and polynomial regression were not far behind for the same period – the linear model performing better with a weekly dataset but worst overall for the daily dataset.

Predictions vs Actual satellite plots

Whilst RMSE is a serviceable benchmark for evaluating model performance, it is inadequate alone as numerical prediction values are difficult to visualise in the context of spatial GPS data. Accordingly, predicted GPS values were plotted against the actual values on a Google Maps satellite image in the interest of contextualising the prediction accuracy in a geographical setting.

Linear plots

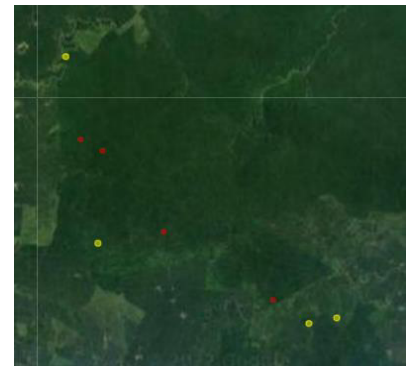
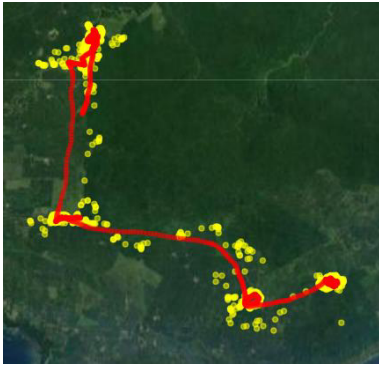


Figure 43: Satellite GPS plots for Linear Regression, original, weekly, monthly left to right

Polynomial plots

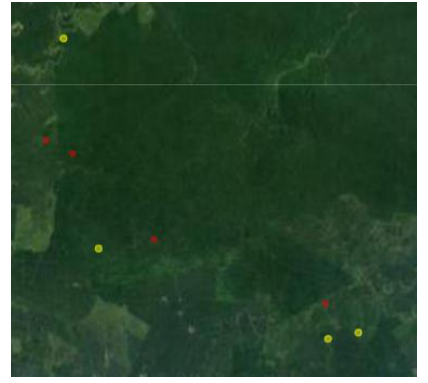
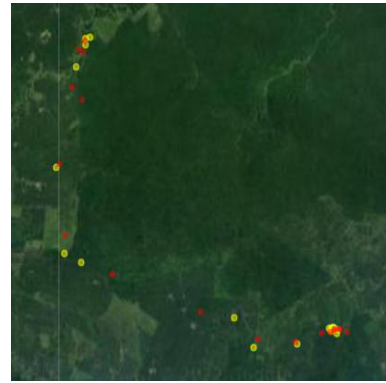
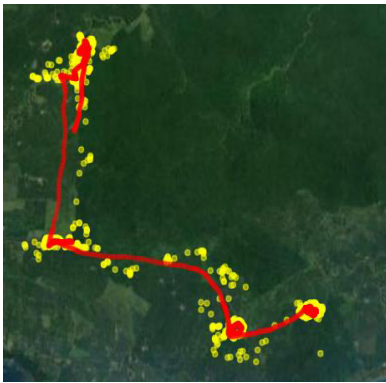


Figure 44: Satellite GPS plots for Polynomial Regression, original, weekly, monthly left to right

VAR plots

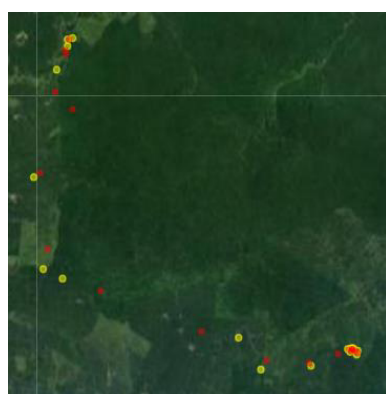
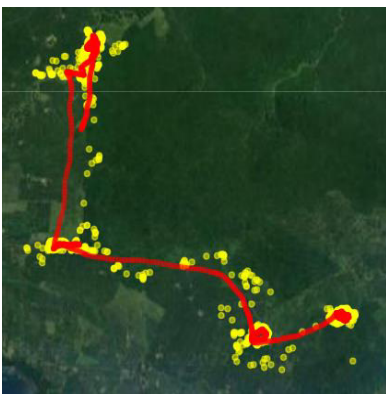


Figure 45: Satellite GPS plots for Vector Autoregressive, original, weekly, monthly left to right

Results appraisal

As expected from the RMSE results, the satellite plots for daily and weekly data displayed accurate prediction points as well as precise movement trend predictions of Seri, with high density points approximated correctly. Predictions for monthly data were much more inconclusive, likely due to the lack of data – with predicted points being placed within dense vegetation instead of following the natural movement corridors as shown in the daily and weekly predictions. Based on the RMSE and satellite predictions, VAR is the selected model that has performed the best.

Despite the accuracy of the results, the use of regression lines as the basis of this project leads something to be desired, which is displayed by the difference between the shape of the actual values against the shape of the predicted values. Animal movement is often erratic and non-uniform, something that is clearly displayed on the daily satellite images. In contrast, the predicted points drawn from the regression line are sequentially placed – a continuous chain of points from start to finish with a constant distance between each of them.

Furthermore, the importance of a large data set is made apparent by the plots for each timeframe, the daily and weekly plots allowing more interpretable conclusions to be drawn regarding animal position and movement in comparison to the monthly plot. Whilst the same training/testing ratio has been utilised across all timeframes, with 8612 and 113 total observations for daily and weekly respectively, monthly data only has 25 observations – a huge drop in the training quality.

A probable issue that is not a factor in this project is extrapolation. The testing dataset used to evaluate the models are temporally situated adjacent to the training set data points, subsequently allowing for predictions to follow trends that have already been displayed recently. However, in real world applications, prediction quality would suffer greatly. The latest temporal data point is in early 2018, meaning if models were used in the present day, unseen variables that cannot be accounted could potentially render results worthless. Future potential patrol routes curated because of forecasts based on past seasons have the capacity to lead rangers amiss in the search of Seri.

Future Work

Although the aim of the project has been achieved, there is still much room for improvement within the project.

Extrapolation

As mentioned previously, extrapolation is an issue that will occur with the use of future data, primarily due to unknown factors changing over time. Instituting geographical data would reduce the effects of extrapolation, as most landmarks such as rivers are unlikely to change over the course of a few years. Furthermore, their effect on elephant migration routes would allow more refined prediction, as high probability areas could be used as the basis for forecasts. Searches for geographical data in Sabah were undertaken in the initial research phase but proved fruitless. With a greater amount of project time, applications could potentially be filed to organisations that withhold such information in the hopes of getting a usable database.

Unrealistic plotted movement patterns

As mentioned earlier, the use of regression lines to forecast points displays an unrealistic uniform movement trajectory for Seri with the daily dataset. Whilst the previous issue was due to time constraints, this issue is simply expected when plotting a significant number of sequential points onto a regression line. Perhaps by constituting classification animal behaviour states for foraging and exploratory, like methods seen in previous works, the GPS points could be plotted in succession based on the classification prediction, which in turn, affects the step length from the previous point. An exploratory state would cause a greater step length between two adjacent points, whereas a foraging state would have a series of smaller steps, in a higher concentration within one area. This could be further developed with a raster model design for the reserve area, with the next cell location being decided based on what cells are within step range, and which has the highest probability based on previously mentioned geographical landmarks. Cell values that are lacking in information e.g., altitude could have missing values interpolated through kriging. The model could perform this iteratively, starting from the last training GPS point.

Spatial Autoregressive (SAR)

Whilst VAR was implemented, spatial autoregressive model was not considered as one of the final models due to time constraints. Numerous issues with the models that I had already began to develop into the project meant that once the troubleshooting phase was over, there was a short amount of time left to spend time on correctly fitting SAR into the project. Time that I instead spent on refining the models, as I felt comfortable using them at this point after hours of tinkering.

Like VAR, SAR allows the spatial impact between variables to be considering when making predictions on data. Spatial correlation could be measured prior to fitting, by using Moran's I. If spatial correlation is detected, geographical variables could be added as dependent variables. The model could use the spatial correlation between these geographical landmarks and the GPS values in the training phase to make accurate predictions in the testing phase.

Conclusion

As this project concludes, it is worth returning to the original aims of the project and critically evaluating whether they were completed. To recapitulate, the aims of the project in the initial plan were:

- To be able to explore the data with the intention of helping patrollers better understand how animals move, so animals can be located easier.
- To be able to make it easier for patrollers to organise patrols.
- To make it harder for poachers to get closer to animals.

Constructing regression models on the data set was not part of the achievable tasks set out for this project initially, yet early visualisations on the Seri dataset displayed consistent migration patterns through natural wildlife corridors. Such a relationship was compelling from an analytical standpoint – these seasonal trends could potentially be a fundamental foundation on which animal movements could be predicted. The results from the selected machine learning configurations demonstrate this, with predicted GPS values exhibiting a high degree of accuracy as well as precise forecasting of the direction of movement trends.

These results suggest that reserve rangers can potentially utilise past GPS data to predict where elephants can be at different times of the year. Consequently, patrol routes can be organised more efficiently, making it difficult for poachers to approach elephants at risk.

However, whilst these conclusions can be drawn from the findings, the reality is that extrapolation and unseen variables within Sabah makes applying the finalised models in the present time an arduous task. Future predictions are likely to be inaccurate without the trends and seasonality used as the basis for this project. Furthermore, whilst regression techniques have served well in this scenario, the elephant movement paths are made to appear consistent and uniform, when it is the opposite. For this reason, the general area surrounding prediction points as well as the direction of movement should be considered as a secondary tool to reinforce evidence such as recorded recent elephant sightings, instead of taking centre stage.

The inclusion of geographical data doubled with spatial regression methods may be a potential avenue for those who are able to obtain such resources. Doing so could possibly mitigate the risks of extrapolation – the fixed landmarks being a basis on which predictions are built around.

Reflection

The "Reflection"

We believe in the concept of "lifelong learning". One of the principles applied throughout the assessment during your studies is that of the value of reflection. We believe that it is important that we reflect upon our performance in order to identify "transferable learning", that can be carried over into future activities. Reflection should focus on what Argyris calls "double loop learning"; this is where we identify, not relatively "simple skills", such as the mastery of a new programming language, but the impact of what we have done on the assumptions, concepts and ideas we used to make decisions about our work. For example, a "reflective practitioner" would try to identify the characteristics of the problem that has been addressed, and consider whether assumptions or decisions about the relevant approach to solving that problem had been appropriate, in order to make a better decision in relation to problems that might be encountered in the future.

Over the course of this project, the development process has shown to me that the original aim of the project can change depending on conclusions unearthed through the methods used. In this instance, the occurrence of trends in the basic visualisation stage revealed seasonality in the movement patterns of Seri, which lead to the use of regression machine learning. In general, I would say that a core lesson has been learnt in relation to the problem area and my own solution. This is that whilst results shown can verify that predicative modelling could benefit wildlife welfare, there is still a big discrepancy between virtual models and the real-life problem area. Problems such as unseen variables and extrapolation cannot be accounted for, so whilst the designed models are promising, application in real life must be done cautiously.

In conclusion, I would say that the approach undertaken has been appropriate considering the limitations regarding data quality – an elephant's route has been successfully forecasted, the implications of which being that patrollers can use such a tool as a secondary input on future patrols. Predictive models in general are a cheaper option for underfunded reserves such as the ones in Sabah, thus this project has relevance in the problem area.

Reference

1. Howard, B., 2015. *Why Elephants Are Recovering in Uganda as They Decline Overall*. [online] National Geographic. Available at: <https://www.nationalgeographic.com/animals/article/150622-uganda-poaching-wildlife-crime-elephants-ambassador-wonekha#:~:text=Elephants%20in%20Uganda%20have%20increased,and%20the%20Uganda%20Wildlife%20Authority>.
2. Snow, J., 2016. *Rangers Use Artificial Intelligence to Fight Poachers*. [online] National Geographic. Available at: <https://www.nationalgeographic.com/animals/article/paws-artificial-intelligence-fights-poaching-ranger-patrols-wildlife-conservation>
3. Palkopoulou, E., Lipson, M., Mallick, S., Nielsen, S., Rohland, N., Baleka, S., Karpinski, E., Ivancevic, A., To, T., Kortschak, R., Raison, J., Qu, Z., Chin, T., Alt, K., Claesson, S., Dalén, L., MacPhee, R., Meller, H., Roca, A., Ryder, O., Heiman, D., Young, S., Breen, M., Williams, C., Aken, B., Ruffier, M., Karlsson, E., Johnson, J., Di Palma, F., Alfoldi, J., Adelson, D., Mailund, T., Munch, K., Lindblad-Toh, K., Hofreiter, M., Poinar, H. and Reich, D., 2018. A comprehensive genomic history of extinct and living elephants. *Proceedings of the National Academy of Sciences*, [online] 115(11). Available at: <https://www.pnas.org/doi/10.1073/pnas.1720554115>
4. World Wildlife Fund. 2018. *The status of Asian elephants*. [online] Available at: <https://www.worldwildlife.org/magazine/issues/winter-2018/articles/the-status-of-asian-elephants>
5. En.wikipedia.org. (2022a). *Ivory trade - Wikipedia*. [online] Available at: https://en.wikipedia.org/wiki/Ivory_trade#Elephant_ivory
6. En.wikipedia.org. (2022b). *Ivory - Wikipedia*. [online] Available at: https://en.wikipedia.org/wiki/Ivory#Controversy_and_conservation_issues
7. Krishnasamy, K., 2016. MALAYSIA'S INVISIBLE IVORY CHANNEL. *Traffic Report*, [online] Available at: <https://portals.iucn.org/library/sites/library/files/documents/Traf-149.pdf>
8. Zhou, X., Wang, Q., Zhang, W., Jin, Y., Wang, Z., Chai, Z., Zhou, Z., Cui, X. and MacMillan, D., 2018. Elephant poaching and the ivory trade: The impact of demand reduction and enforcement efforts by China from 2005 – 2017. *Global Ecology and Conservation*, 16, p.e00486.
9. Davies, C., 2016. *The tangled routes of global elephant ivory trafficking - ELA*. [online] Eia-international.org. Available at: <https://eia-international.org/blog/tangled-routes-global-elephant-ivory-trafficking/>
10. Davis, E., 2018. *New Survey Finds One in Seven Wildlife Rangers Have Been Seriously Injured in the Line of Duty Over the Past Year*. [online] World Wildlife Fund. Available at: <https://www.worldwildlife.org/press-releases/new-survey-finds-one-in-seven-wildlife-rangers-have-been-seriously-injured-in-the-line-of-duty-over-the-past-year#:~:text=The%20results%2C%20part%20of%20the,up%20from%20101%20last%20year.>>
11. Goh, H., 2021. Strategies for post-Covid-19 prospects of Sabah's tourist market – Reactions to shocks caused by pandemic or reflection for sustainable tourism?. *Research in Globalization*, [online] 3. Available at: <https://www.sciencedirect.com/science/article/pii/S2590051X21000216>
12. Dosm.gov.my. 2021. *Department of Statistics Malaysia Official Portal*. [online] Available at: https://www.dosm.gov.my/v1/index.php?r=column/cthemeByCat&cat=493&bul_id=VTNHRkdZkFzenBNd1Y1dmg2UUlRZz09&menu_id=amVoWU54UTl0a21NWmdhMjFMMWcyZz09
13. Othman, N., Goossens, B., Cheah, C., Nathan, S., Bumpus, R. and Ancrenaz, M., 2019. Shift of paradigm needed towards improving human–elephant coexistence in monoculture landscapes in Sabah. *International Zoo Yearbook*, [online] 53(1), pp.161-173. Available at: <https://zslpublications.onlinelibrary.wiley.com/doi/full/10.1111/izy.12226>
14. Hance, J., 2013. *14 Bornean elephants found dead, likely poisoned*. [online] Mongabay Environmental News. Available at: <https://news.mongabay.com/2013/01/14-bornean-elephants-found-dead-likely-poisoned/> [Accessed 19 May 2022].
15. Wyler, L. and Sheikh, P., 2008. International Illegal Trade in Wildlife: Threats and U.S. Policy. [online] Available at: <https://apps.dtic.mil/sti/pdfs/ADA486486.pdf>

16. Cardiff University. n.d. *Danau Girang Field Centre*. [online] Available at: <<https://www.cardiff.ac.uk/danau-girang-field-centre>>
17. Google, 2022. *Picture of Danau Girang Field Centre*. [image] Available at: <<https://www.google.com/maps/place/Danau+Girang+Field+Centre/@5.4136911,118.037672,2741m/data=!3m1!1e3!4m5!3m4!1s0x0:0x47f4b4f6fc957d2f8m2!3d5.4136911!4d118.037672>>
18. Indran, D., n.d. Kinabatangan case study. [online] Available at: <https://www.gwp.org/globalassets/global/toolbox/case-studies/asia-and-caucasus/malaysia-kinabatangancasestudy_256.pdf>
19. World Weather & Climate Information. 2022. *Climate and average monthly weather in Lahad Datu (Sabah), Malaysia*. [online] Available at: <<https://weather-and-climate.com/average-monthly-Rainfall-Temperature-Sunshine,lahad-datu-sabah-my,Malaysia>>
20. Saikim, F., Prideaux, B., Mohamed, M. and Hamzah, Z., 2016. Using Tourism as a Mechanism to Reduce Poaching and Hunting: A Case Study of the Tidong Community, Sabah. *Advances in Hospitality and Leisure*, [online] Available at: <https://www.researchgate.net/publication/311555268_Using_Tourism_as_a_Mechanism_to_Reduce_Poaching_and_Hunting_A_Case_Study_of_the_Tidong_Community_Sabah>
21. Fang, F. and H. Nguyen, T., 2016. PAWS – A Deployed Game-Theoretic Application to Combat Poaching. *Harvard Journal*, [online] Available at: <https://projects.iq.harvard.edu/files/teamcore/files/2017_1_teamcore_aim_paws_0929.pdf>
22. Fang, F., Nguyen, T., Sinha, A., Gholami, S., Plumtre, A., Joppa, L., Tambe, M., Driciru, M., Wanyama, F., Rwetsiba, A., Critchlow, R. and Beale, C., 2017. Predicting poaching for wildlife Protection. *IBM Journal of Research and Development*, 61(6), pp.3:1-3:12.
23. Kar, D., Ford, B. and Gholami, S., 2017. Cloudy with a Chance of Poaching: Adversary Behavior Modeling and Forecasting with Real-World Poaching Data. *International Conference on Autonomous Agents and Multi-Agents Systems*, [online] 16. Available at: <https://www.researchgate.net/publication/316861069_Cloudy_with_a_Chance_of_Poaching_Adversary_Behavior_Modeling_and_Forecasting_with_Real-World_Poaching_Data> [Accessed 27 May 2022].
24. Clark, D., Shaw, D., Vela, A., Weinstock, S. and Santerre, J., 2021. Using Machine Learning Methods to Predict the Movement Trajectories of the Louisiana Black Bear. *SMU Data Science Review*, [online] 5(1). Available at: <<https://scholar.smu.edu/cgi/viewcontent.cgi?article=1173&context=datasciencereview>> [Accessed 7 April 2022].
25. COLLECTION OF TRAJECTORIES
26. Wan Mohd Jaafar, W., Said, N., Abdul Maulud, K., Uning, R., Latif, M., Muhmad Kamarulzaman, A., Mohan, M., Pradhan, B., Saad, S., Broadbent, E., Cardil, A., Silva, C. and Takriff, M., 2020. Carbon Emissions from Oil Palm Induced Forest and Peatland Conversion in Sabah and Sarawak, Malaysia. *Forests*, [online] 11(12). Available at: <<https://www.mdpi.com/1999-4907/11/12/1285/htm>>.
27. Torous, W., Valkanov, R. and Yan, S., 2004. On Predicting Stock Returns with Nearly Integrated Explanatory Variables. *The Journal of Business*, 77(4), pp.937-966.
28. En.wikipedia.org. 2022. *Python (programming language) - Wikipedia*. [online] Available at: <[https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))>
29. En.wikipedia.org. 2022. *NumPy - Wikipedia*. [online] Available at: <<https://en.wikipedia.org/wiki/NumPy>>
30. En.wikipedia.org. 2022. *Matplotlib - Wikipedia*. [online] Available at: <<https://en.wikipedia.org/wiki/Matplotlib>>
31. Seaborn.pydata.org. 2022. *seaborn: statistical data visualization — seaborn 0.11.2 documentation*. [online] Available at: <<https://seaborn.pydata.org/>>
32. En.wikipedia.org. 2022. *pandas (software) - Wikipedia*. [online] Available at: <[https://en.wikipedia.org/wiki/Pandas_\(software\)](https://en.wikipedia.org/wiki/Pandas_(software))>
33. Statsmodels.org. 2022. *Introduction — statsmodels*. [online] Available at: <<https://www.statsmodels.org/stable/index.html>>

34. Ven, B., 2022. *Bokeh*. [online] Bokeh.org. Available at: <<https://bokeh.org/>>
35. En.wikipedia.org. 2022. *scikit-learn - Wikipedia*. [online] Available at: <<https://en.wikipedia.org/wiki/Scikit-learn>>
36. Statsmodels.org. 2022. *Introduction — statsmodels*. [online] Available at: <<https://www.statsmodels.org/stable/index.html>>
37. Google Developers. 2022. *Overview | Maps JavaScript API | Google Developers*. [online] Available at: <<https://developers.google.com/maps/documentation/javascript/overview>>
38. The Data Frog. 2022. *Show your Data in a Google Map with Python*. [online] Available at: <<https://thedatafrog.com/en/articles/show-data-google-map-python/>>
39. python, S. and Shrivastava, A., 2022. *Simple prediction using linear regression with python*. [online] Stack Overflow. Available at: <<https://stackoverflow.com/questions/29623171/simple-prediction-using-linear-regression-with-python>>
40. Data36. 2022. *Polynomial Regression in Python using scikit-learn (with example)*. [online] Available at: <<https://data36.com/polynomial-regression-python-scikit-learn/>>
41. Machinelearningplus.com. 2022. [online] Available at: <<https://www.machinelearningplus.com/time-series/vector-autoregression-examples-python/>>