Logo, company name

Description automatically generatedDETERMINING THE AGE OF INSECT LARVAE FOR FORENSIC ENTOMOLOGY

MOHIT JAIN

*14th September 2022  
 MSc in Computer Science 2021-22*

SCHOOL OF COMPUTING AND MATHEMATICS   
Keele University   
Keele   
Staffordshire   
ST5 5BG

# **ABSTRACT**

Many forensic entomologists have studied the growth of insect larvae to estimate their ages. Feeding the larvae in the lab over many years at various rearing temperatures can determine the age of any insect larvae. Through this process, they have gathered a variety of data by tracking the changes in the larvae's properties over time and at various temperatures until it transforms into a pupa. Using this data, I need to create an application that uses the characteristics of the larvae as an input to predict the age of insect larvae.

To develop the application for the end-users, I will employ machine learning methods and a few different Python frameworks in this project. To achieve this, I'll use the decision tree algorithm to determine the insect larvae's age with the highest degree of accuracy and better visualise how the algorithm predicts the value.

# **ACKNOWLEDGEMENT**

I want to start by sincerely thanking my supervisor, Dr Charles Day, for his passion for this project, his patience, and for his intelligent criticism and dissertation recommendations. His technical and editorial guidance was crucial to the successful completion of this dissertation, and I have learned a lot from him about how academic research functions. He guided me on various machine learning project methodologies and how to plan each aspect of the project right away. Even some of the tools I might have utilised to create my application were suggested by him.

I also want to thank the people that evaluated my project since they gave me insightful criticism. With their assistance, I can determine what kind of design is most educational and user-friendly.

I want to thank my flatmates who are constant support during this project. They have guided me on what type of format for the report is most interactive and understandable.

TABLE OF CONTENT

[0](#_Toc113984923)

[ABSTRACT 1](#_Toc113984924)

[ACKNOWLEDGEMENT 1](#_Toc113984925)

[TABLE OF FIGURES 3](#_Toc113984926)

[INTRODUCTION 4](#_Toc113984927)

[ MOTIVATION 4](#_Toc113984928)

[ BACKGROUND INFORMATION 5](#_Toc113984929)

[ APPLICATION DEVELOPMENT PROCESS 7](#_Toc113984930)

[ LITERATURE AND DATA COLLECTION 8](#_Toc113984931)

[ PREVIOUS WORK OVERVIEW 10](#_Toc113984932)

[ AIMS AND OBJECTIVE 12](#_Toc113984933)

[DECISION TREE ALGORITHM 13](#_Toc113984934)

[ ID3 DECISION TREE 13](#_Toc113984935)

[ C4.5 DECISION TREE 14](#_Toc113984936)

[ CART 15](#_Toc113984937)

[METHODOLOGY 17](#_Toc113984938)

[IMPLEMENTATION AND TOOL USED 21](#_Toc113984939)

[ PANDAS LIBRARY 21](#_Toc113984940)

[ SCIKIT-LEARN LIBRARY 22](#_Toc113984941)

[ PYTHON FLASK 23](#_Toc113984942)

[ PYTHON GUI TKINTER 24](#_Toc113984943)

[ DTREEVIZ LIBRARY 24](#_Toc113984944)

[EVALUATION 26](#_Toc113984945)

[RESULTS 27](#_Toc113984946)

[CONCLUSION 28](#_Toc113984947)

[ LIMITATIONS 28](#_Toc113984948)

[ FUTURE WORK 28](#_Toc113984949)

[REFERENCES 29](#_Toc113984950)

[APPENDIX 1 33](#_Toc113984951)

[APPENDIX 2 33](#_Toc113984952)

[APPENDIX 3 33](#_Toc113984953)

**TABLE OF FIGURES**

[Figure 1 - The Application Development Process 6](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983665)

[Figure 2 - Isomorphen diagram of the L. sericata - taken from Grass Berger and Reiter (Reiter, 2001) 8](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983666)

[Figure 3 - Neural Network age Clusters 10](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983667)

[Figure 4 - NEURAL NETWORK AGE ERROR FEEDBACK 10](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983668)

[Figure 5 - Graph for the understanding of Decision Tree 12](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983669)

[Figure 6 - Classification Tree Example 15](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983670)

[Figure 7 - Example Dataset, X is a continuous variable, and Y is another continuous variable 18](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983671)

[Figure 8- Example Data Table 18](#_Toc113983672)

[Figure 9- Original Dataset of Insect Larvae Characteristics 18](#_Toc113983673)

[Figure 10 - Example Tree Representation 19](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983674)

[Figure 11 - Tree Visualization of Original Dataset 20](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983675)

[Figure 12 - Display of UI Application for online users 22](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983676)

[Figure 13 - GUI application for Offline users 23](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983677)

[Figure 14 - Tree Visualization of the predicted value 24](file:////Users/mohitjain/Desktop/Final%20Project%20Report.docx#_Toc113983678)

[Figure 15 - Path visualisation of the predicted value 24](#_Toc113983679)

[Figure 16 - Evaluation Ratings 25](#_Toc113983680)

[Figure 17 - Actual and Predicted age of the insect larvae 26](#_Toc113983681)

# **INTRODUCTION**

To better understand and develop a machine that by accessing enough live and previous data, shall be able to predict or estimate quite like a human being. Many developers and scientises are using several kinds of methods in Artificial Intelligence, and Machine learning by providing real-time and previous input data and getting more realistic output/outcomes from that, which can help us to be aware and technically advanced. This project aims to make a machine learning model to predict and determine the age of Insect larvae for Forensic entomologists. This can help the end-user predict the period of the dead body during the Post-mortem Investigation (PMI). This model allows the end-user to fasten their medical examination of the dead body by collecting the larvae samples from the crime scene.

## MOTIVATION

Insect plays a vital role during the crime scene to get a proper estimate of the time when murder or suicide happens. Many undiscovered and unrecognised dead bodies have been found in earlier days, but it takes a huge time for a forensic scientist to get a proper estimation of the time of death. So, to determine the predicted time of the dead body, Forensic Entomologist collected a huge amount of data from insect larvae which was laid down on the body after death and at different raring temperature; they observed the larval stage of the larvae with their change in structure according to their weight and length. By doing this process for years in the laboratory, they noted the time of insect larvae until it will get change the laboratory, they said the time insect larvae until it changes its state to pupae. Now, by using Machine learning techniques, the software developers made some models for them, which helps them to predict the age of the larvae with different input fields on the Neural Network algorithm in the previous development.

The Post-mortem Interval (PMI), often known as the time since death, must be ascertained when a body is found. Depending on the condition of the body and the environment (the more hospitable the settings, the quicker the decomposition process), a medical pathologist can typically provide an accurate PMI up to 72 hours after death (J.B. Butcher, 2012). However, the PMI will become less precise the longer the body is discovered after this window of time, which is where bug evidence can be used frequently and very successfully.

This project is a Machine learning model based on a decision tree algorithm, in which the data is split into different sections according to input properties and moves to the required selection according to the input. This model helps the user not only to provide the user output as a prediction but also the user can visualise how this model comes to this prediction. The existing methods for estimating the PMImin involve two major processes:

1) The oldest insect specimens that have grown on the carrion must be recognised and dated to provide the earliest PMI estimates.

2) The succession patterns of arthropods present on the carrion must be investigated for late PMI calculations (J. Amendt, 2007).

## BACKGROUND INFORMATION

A subfield of forensic science called forensic entomology uses insects and the arthropod relatives of those insects that live in decaying human remains to assist legal investigations. Sung Tu, a Chinese attorney and death investigator, applied forensic entomology for the first time in the 13th century (J. Hajslova, 2011). Between the 13th and 19th centuries, biologists made discoveries, and since then, forensic entomology has developed into a distinct field of study. Although it has been available for a long time, its actual forensic potential has recently begun to be utilised. The first instance of it being used in a trial was in France in 1850 (J. Hajslova, 2011).

The vast amount of information provided by forensic entomology can be divided into three categories (H.J. Kim, 2011):

1) Urban forensic entomology.

2) Medicolegal forensic entomology.

3) Forensic entomology of stored product pests.

Only medicolegal forensic entomology will be considered for this study. Medicolegal forensic entomology is concerned with the necrophagous insects that infest decomposing remains and are implicated in criminal investigations (Larson, 2009). Forensic entomotoxicology is a branch of medicolegal forensic entomology that can be used to determine whether the deceased had taken drugs and toxins prior to death because the growth rate of the larvae can be accelerated or retarded depending on the drug (Sovocool, 2011). Medicolegal forensic entomology is concerned with more than just insect infestations in the deceased. It also includes cases of myiasis, the infestation of live vertebrates with Diptera larvae, which is usually associated with neglect cases where open wounds are colonised (J.Y. Yew, 2008). On the other hand, myiasis is a distinct branch of forensic entomology.

Because of the number of insects that inhabit rotting bodies, they give a range of forensic evidence. The first inhabitants, known as blowflies (J. Hajslova, 2011) can provide information ranging from an estimate of how long a person has been dead [10] to determine whether the deceased had used drugs prior to death (R.B. Cody, 2007). The evidence's quality is dependent on accurate insect identification and a thorough understanding of entomology and blowfly biology (J.B. Butcher, 2009).

A minimum Postmortem Interval (PMImin) can be obtained if a forensic entomologist is aware of the blowfly species present, their respective development times, and how they are affected by temperature and other environmental conditions. The estimation is dependent on correct identification, a high degree of accuracy in the environmental conditions to which the insects are thought to have been exposed, and the dependability and credibility of experimental data used in the calculations (J.B. Butcher, 2009).

Many insects, ranging from flies (Diptera) to wasps (Hymenoptera), will be drawn to a cadaver as it decomposes, a process known as succession (see also 1.3). This pattern of insect succession on decomposing remains has been thoroughly researched, and flies, notably the blowfly (Calliphoridea), are known to be the early colonisers, arriving relatively fast after death (Kohonen, 1990), making them the most popular insects used for PMI assessments.

If the species are incorrectly identified, the PMImin estimation will be either too short or too long, potentially leading to conflicts with other evidence (J.B. Butcher, 2009). To determine the PMImin, the age of the eldest developing blowfly on the body must also be estimated. These are the first steps a forensic entomologist must take when confronting a crime scene.

Female blowflies will lay their eggs in open wounds (if any exist) or in the body's natural and moist orifices, such as the ears, eyes, nose, and genitalia. Because this process occurs in waves, there will be a continuous stream of larvae hatching at various times. Based on the life cycle stages of the insect species found in the body, a forensic entomologist can estimate PMImin. The PMI is a mathematical calculation with many variables (ambient air temperature, mass maggot temperature, etc.); when all of these are considered, a reliable time since death can be estimated.

Businesses can use Machine Learning to automate routine tasks. It also aids in the automation and rapid creation of models for data analysis. Various industries rely on massive amounts of data to optimise their operations and make informed decisions. Machine Learning aids in developing models capable of processing and analysing large amounts of complex data to produce accurate results. These models are precise and scalable, with a shorter turnaround time. Businesses can capitalise on profitable opportunities while avoiding unknown risks by developing such Machine Learning models (team, 2022).

In this Project, I’m going to use Machine Learning to help the end-user predict the insect larvae age from the above-experimented data for the larvae in the laboratory under different rearing temperature conditions. Machine learning appears to be one of the most significant recent developments, enabling automation and artificial intelligence. Other machine learning algorithms aid in allocating and prognosis of high-complexity tasks (Anon., 2019). It offers a notable, extraordinary, albeit highly complex, method for analysing and forecasting actions.

Some algorithms may produce successful results right away, but others do not. As a result, using a specific algorithm is generally regarded as an exploratory rather than an executional method. Machine learning must explain how it learns by experimenting with various shapes and networks. Machine learning can occur in one of two ways. The first method is supervised learning, in which a model is built using known input and output data. The second method, unsupervised learning, occurs when unusual aspects and patterns are discovered solely in the input data.

## APPLICATION DEVELOPMENT PROCESS

Diagram

Description automatically generatedIn practice, designing a classification or regression application is often iterative and involves the consideration of many limitations and criteria. The four critical steps of this iterative approach are depicted in Figure 1. The dotted arrows indicate the iterative nature of the process. The four basic steps are as follows:

Figure 1 – The Application Development Process

1. **We are analysing the problem**, which entails identifying the relevant domain, data, and human aspects.
2. **We are selecting a particular model and algorithm** based on matching the identified problem-specific factors (Step 1) to the general properties of the models and algorithms under consideration.
3. **Examine the test results** produced by the chosen model and algorithm. In most cases, the first attempt at picking an algorithm does not result in the final solution but rather gives additional data for re-analysing the issue objectives and constraints. Various of limitations can be changed for many problems. For example, one may be able to acquire more data, which can modify the problem dimensionality to sample size ratio, influencing the model and algorithm used. Similarly, analysing the results of a specific algorithm can compel one to modify one's goals. For example, whereas model understandability may have been the main goal at first, it may become less important in the face of a considerable loss of prediction accuracy.
4. **Deploying the developed system** in an operational environment. If the model meets the project's overall objectives in an acceptable manner, the procedure is complete. Otherwise, one might need to go back to Step 1 or 2 (Brodley, 1995).

The application in this project is built on the *Decision tree algorithm* in machine learning, which is considered because of its ability to split data according to properties and provide a better visualisation to the user, and after analysing different algorithms with the dataset, the Decision tree algorithm predicted the required data with the most precise prediction of the larvae age.

## LITERATURE AND DATA COLLECTION

The Data which has been used in this project was collected in Frankfurt’s Forensic Entomology laboratory in the form of an excel sheet which contains the columns of changes in the characteristics of weight, length, larvae stage, etc. from the juvenile stages of necrophagous insects (which usually colonise the body very soon after death) till it convert to pupae on ambient rearing temperature. Based on anatomical and morphological changes, there are recognised methods for determining the age of blow fly eggs and larvae. These alterations in eggs can be seen by histological examination or light microscopy (Pais M, 2018). The age of larvae is estimated by measuring their age-dependent length or weight and comparing it to species-specific reference data (Grassberger M, 2001). However, post-feeding larvae migrate from the corpse to pupariate (B, 1991), and thus, due to the fluctuating decrease in size during this stage, length measurement cannot be used to estimate the age of those larvae without incorporating behavioural aspects (such as moving away from and not staying on the body).

If distinct landmarks are formed after pupation, morphological approaches could be employed to estimate age. However, such markers are difficult to define and contain a large degree of subjectivity (Zajac BK, 2012). Several molecular age markers for pupae of flesh flies (Shang Y, 2020) and blow flies (Tarone AM, 2011)have been identified to overcome this. Molecular analysis, which can be standardised, gives quantitative and precise gene expression data. The expression levels of numerous genes were studied throughout pupal development for this purpose. The assessment of the activity of these age-related genes reveals the pupa's age. Differentially expressed genes in flies show promise as a potential age indicator during metamorphosis. The quantitative examination of these transcripts reveals gene up and down regulation. However, due to the often-small quantitative differences in expression within one gene (fold change values 15 between distinct age landmarks) and the considerable interindividual variation, generating solid age determination data is difficult.

Using a de novo transcriptome analysis, quite distinct gene expression levels of C. vicina were recently found during metamorphosis (Zajac BK, 2015). Markers for 15 separate developmental stages of C. vicina pupae were identified, one for each day of pupal development at 15 °C. These indicators have fold change values ranging from 10 to 40,000 when compared to conditions in the very early pupa, indicating a substantially larger shift in gene expression than observed for other age-dependent transcripts. Consequential qPCR assays for each of these indicators, known as A1/ A2-O1/ O2, have been validated for three different continuous breeding temperatures seen in relevant forensic scenarios: 15 °C, 20 °C, and 25 °C.

Except for one, the relative expression patterns of each marker during metamorphosis are similar for each temperature and correlate to transcriptome data, demonstrating that the selected age markers are independent of the constant temperatures studied (Zajac BK, 2018).

However, no studies have been undertaken to assess those markers at changing temperatures, even though this is a more realistic environment, especially in outdoor scenarios. This is significant since it is unknown whether age indicators' gene expression levels are the same under constant and variable temperature circumstances. Several development studies (Warren J-A, 2013)demonstrated the effect of changing versus stable temperature circumstances on the development time of several insects. Thus, different blowflies such as Calliphora vomitoria, Protophormia terraenovae, and Lucilia sericata developed faster under varying temperatures (Davies L, 1994) ,but Aldrichina grahami and C. vicina developed slower relative to stable temperatures (Chen W, 2019).

Diagram

Description automatically generated

Figure 2 - Isomorphen diagram of the L. sericata - taken from Grass Berger and Reiter (Reiter, 2001)

When assessing their development or age, keep in mind that the development time of blow flies is affected by species-specific temperature influences. Greenberg and Kunich (Greenberg B, 2005) assumed that the relationship between development rate and temperature is linear within a specific temperature range where development is possible. As a result, to compare growth at different temperatures, the physiological age is provided in accumulated degree hours (ADH) or accumulated degree days (ADD). ADH or ADD is the sum of a specified amount of heat required for an insect's development over its species-specific lower development threshold. There are signs that the quantity of ADD required for full growth may be temperature sensitive (B, 1991). As previously stated, the ADD necessary for growth is lower than that estimated near the lower temperature threshold and more than that calculated at or beyond the temperature optimum (Ames C, 2003).

As a result, the current study studied the development of C. vicina cultured at different fluctuating and constant temperatures from the lower development threshold to the higher development threshold. Three temperature ranges were studied, reflecting C. vicina's lower, medium, and upper-temperature development ranges. The current study investigated breeding circumstances of 15 °C, 20 °C, and 25 °C, as well as evenly variable temperatures with corresponding mean values, namely 5-15 °C, 15-25 °C, and 25-35 °C. Furthermore, it was studied if the pupae's gene expression patterns are only dependent on physiological age (ADD %-development at a given temperature), independent of whether the external temperature is constant or variable during development.

Furthermore, an R-based statistical tool was developed to estimate age based on comparing a specific gene expression pattern with the pattern of all age indicators at all investigated developmental stages. As a result, as a foundation for the age prediction tool, a crucial reference database, including the gene expression data of the pupae reared under the stated constant and variable temperature settings, was created. The gene expression data of C. vicina pupae from outdoor breeding were used to validate the age prediction algorithm.

This Data was used in Previous work by Machine Learning application experts, which is elaborated on in the next section of the report. Now, the data is passed on for this project to improve the accuracy of predicting the age of insect larvae. After getting this dataset, the data was segregated according to the usefulness of the algorithms, which were tried and tested to examine the best-fit algorithm for the prediction and its visualisation.

## PREVIOUS WORK OVERVIEW

Previously, the attempt of Machine Learning for the age estimation of the insect larvae was carried out by the School of Physical and Geographical Science Researchers of Keele University. They have used an ageing technique of forensically important larvae using cuticular hydrocarbons. They analysed using Gas Chromatography-Mass Spectrometry and statically analysed using Principal Component Analysis and Artificial Neural Networks and were able to successfully age larvae of Calliphora vicina and Calliphora vomitoria (Hannah Moore, et al., 2015).

Artificial neural networks (ANNs) are a machine learning method widely used in pattern recognition and clustering applications. They may learn features from a dataset and apply this knowledge to recognise and categorise new, previously unknown data. ANN learning is accomplished by adjusting the weighted connections between the network's artificial neurons. This weight-changing procedure is repeated until an appropriate error (defined as the difference between the network's actual output and the ground truth goal output) is obtained. ANNs are well-suited to handling noisy non-linear data. They can learn the connection between input and output data, making them well-suited to quick processing huge real-world datasets (e.g., voice recognition (J. Butcher, 2010), robotics (M. Beyeler, 2015), structural engineering (J. Butcher, 2014), and forensic sciences (A. Bianconi, 2010).

The Self Organising Map (SOM) (Kohonen, 1990) is an unsupervised technique that groups training data according to its underlying features, one of several forms of ANNs. A SOM comprises an input and an output layer, both of which include artificial neurons, with each neuron in one layer coupled to every neuron in the next. After training, the SOM maps multidimensional data with comparable features into topologically co-located clusters in its output layer, yielding collections that reflect similar data points in its output layer.

Unsupervised training of a SOM entails presenting each input pattern to the input layer, where the incoming weighted connections of the output neuron that best fits the input pattern (known as the winner neuron) are adjusted. This weight shift leads to the creation of clusters inside the output layer, where comparable input patterns cluster in proximity. For further information about SOM training, please see (Kohonen, 1990) and (C. Day, 2009).

Because of a SOM's topological ordering capabilities, it is highly suited to studying high dimensional data, such as that obtained from blowfly species. This was demonstrated in prior work, in which hydrocarbon data acquired from Lucilia sericata pupae were identified using a SOM, with a classification accuracy of 89% (J. Butcher, 2013).

The study's novelty is that it examines the CHC profiles of larvae from two forensically essential blowfly species found in the UK, Calliphora vicina and Calliphora vomitoria, to determine whether different chemical changes are occurring over time and to test the method's reliability by investigating whether this technique is more generally applicable across species. To analyse the HCs, Gas Chromatography-Mass Spectrometry (GC-MS) was employed, and statistical and artificial intelligence data analysis methods were applied to the gathered datasets to cluster and automatically categorise the data, as well as enhance data visualisation.

Length

Length

Age error feedback

Age error feedback

Age Clusters?

Age Clusters?

length

length

weight

weight

stage

stage

Stage

Stage

Weight

Weight

Figure 3 - Neural Network Age Clusters

Figure 4 – Neural Network Age Error Feedback

Some limitations of the previous work are that as the larvae develop and the ambient temperature decreases, the accuracy of the present NNs decreases. SOM NNs are often less able to predict the precise PMI, or PMI +/-1, than MLP NNs. So yet, there is no indication that these findings can be repeated with other insect species (M. Baqué, 2015).

The strength of this work is that even if accuracy is just marginally improved, a more systematic PMI estimate may be beneficial. There is room for autonomous computational intelligence to help PMI estimate using just length, weight, and stage data. The topological map of SOM is a helpful visualisation tool for qualitatively assessing PMI, whereas MLP is useful for an estimate.

## AIMS AND OBJECTIVE

This project aims to create a software tool to assist forensic entomologists in predicting the age of insect larvae which is collected during the Post-mortem Investigation at a crime scene to predict the time of death of a body.

The objectives of this project are: -

* Develop a software tool that can load and help the user visualise the raw insect larvae data.
* Using selected parts of the raw data, build a machine learning model that can predict the age of an insect larva based on some supplied input data.
* Help the end-user to visualise how the machine learning model has made its prediction(s).
* Going to make two models for both online and offline users because sometimes there are no internet signals at the crime scene.

To achieve these aims and objectives, there are several processes which were carried out during the project building and its research:

1. This project works on some previously collected data:

To predict the immediate age of an insect Forensic Entomologist from Frankfurt carried out some operations on insect larvae during their birth to till it gets coverts into a pupa in the laboratory condition at different rearing temperatures by feeding them periodically to observe the change in their state-like shape, weight, length and how long they are surviving at different temperatures. By recording the behaviour of these larvae for a year or more, they collected this data, which was passed on to the software developer to make some machine learning application for the prediction.

1. Filtering and normalisation of data:

The data filtering is carried out to observe how precise the results are coming by comparing their prediction scores. The prediction scores are kept by changing, adding, or excluding the different characteristics of the provided data. This normalisation of the data helps this project to get the most accurate prediction of the larvae age while working with a *decision tree algorithm.*

1. Training and testing on different algorithms:

Training of the data is carried out with the help of different algorithms in machine learning, and after many activities and testing finally, the project is concluded using the decision tree regression algorithm, which is among the most accurate result-providing algorithms. The most important thing about the decision tree algorithm is that it provides better visualisation of the predicted age to the user of the model.

# **DECISION TREE ALGORITHM**

The Decision Tree algorithm is a member of the supervised learning algorithm family. Unlike other supervised learning algorithms, the decision tree technique may also be used to handle regression and classification problems (Chauhan, 2022). The objective is to build a model that predicts the value of a target variable using basic decision rules derived from data attributes. A tree illustrates a piecewise constant approximation.

In Decision Trees, we begin at the tree's root to anticipate a class label for a record. We compare the values of the root attribute with the record's attribute. We proceed to the next node based on the comparison by following the branch corresponding to that value. In figure 5 below, decision trees learn from data to approximate a sine curve using an if-then-else set of decision rules. The deeper the tree, the more complicated the decision rules and model fit.

Chart

Description automatically generated

Figure 5 - Graph for understanding of Decision Tree

There are several decision-tree algorithms. ID3, CART, C4.5, CHAID, MARS, and conditional inference trees are a few examples here (Pathak, et al., 2018).

## ID3 DECISION TREE

Iterative Dichotomiser 3 (ID3) is the most popular and commonly used decision learning method, developed by Quinlan Ross in 1986. Dichotomisation is the process of separating disparate data into distinct zones. ID3 techniques split the data based on the dataset's optimum information gain, which minimises the entropy of the splits. As demonstrated in equations (1) and (2), information gain is the likelihood of obtaining a comparable value from any picked region divided by the decision tree border.

~~ (1)

~~ (2)

Initially, the entropy of the independent variables is calculated concerning the independent variables in the dataset. The information gain is evaluated by selecting the attribute with the highest information gain as an attribute for that node. This process is repeated until no more details are left, or some stopping criteria are met [3], [4]. A realistic example of ID3 may be found in [2] and [4], as shown below.

Split function for ID3:

1. ID3\_SPLIT\_FUN (Node, {Data set})
2. Decision node for this node A
3. For each value of A, create a new child node.
4. Split training {data set} to create child nodes.
5. For each Child node/subset:
   * If the subset is pure: stop
   * else: ID3\_SPLIT\_FUN (Child\_Node, {Data set})

## C4.5 DECISION TREE

Even though ID3 is a popular and robust algorithm, it has numerous drawbacks. It does not function efficiently; for example, when the dataset is enormous [4], there are some missing characteristics [3] or continuous variables [3]. Quinlan Ross suggested C4.5 in 1993 as an extension to the previous ID3. C4.5, like ID3, uses the notion of information gain in conjunction with gain ratio to determine the appropriate split for a node. As indicated in (3) and (4), the gain ratio normalises the information gain, and the characteristics with the highest gain ratio are picked for splitting that node (4).

~~ (3)

~~ (4)

C4.5 employs the gain ratio because [5] discovered that information gain was more skewed in multivariate splits. While splitting information, the number of data moving into specific daughter splits of a node is represented by Di. C4.5 decision trees can manage difficulties such as missing attributes and pruning trees after they have been created [3]. C4.5 may also be used to forecast continuous variables by establishing a threshold and separating the data if the mean of the data exceeds this threshold.

## CART

This approach is suitable for both classification and regression. The CART algorithm divides a node into sub-nodes using the Gini Index criteria. It starts with the training set as a root node. After successfully splitting it in two, it breaks the subsets using the same logic and then splits the sub-subsets again, recursively, until it finds that further splitting will not give any pure sub-nodes or the maximum number of leaves in a growing tree, which is referred to as tree pruning. Because of its simplicity and dual nature (which could be used for regression and classification issues), the CART tree was proposed by Leo Breiman, Jerome Friedman, Charles J. Stone, and R.A. Olshen in 1984. It is based on the concept of recursive partitioning [1], to enhance homogeneity in subsequent daughter nodes. Unlike ID3 and C4.5 trees, which employ information gain criteria to identify an ideal attribute for the split, CART uses the Gini index, as illustrated in (5) [6].

~~ (5)

CART algorithms are supported by the pruning approach since the natural ending point for CART trees is when each leaf node is homogenous [10], but this causes the additional issue of over-fitting and thus leads to the poor prediction of the created model. Trees are allowed to reach their maximum potential before being pruned from the leaf nodes at each step. Since numerous trees are conceivable, the optimal tree is chosen based on cost and complexity.

Where, CC(T) = Costs Complexity related to tree produced after pruning.

Err(T) = Error related to that tree.

L(T) = Length of the tree.

a = Penalty with each tree.

The most minor complexity of the tree created after pruning is picked from all the cost complexities. Real-world use of the CART algorithm is presented in [7], where a tree is used to categorise whether a person is suitable to donate blood to a blood bank or not.

1. **Classification Decision Trees:**

This method always uses a fixed or category target variable. The next step in the process is to apply the algorithm to determine the "class" that a target variable is most likely to belong. Predictions about who will or won't use a digital platform or graduate from high school are examples of classification-type problems (Sabbarwal, 2020).

Diagram

Description automatically generatedThese are examples of straightforward binary classifications in which the variable can have just one of two possible values, which are mutually incompatible. In other situations, you might need to make predictions based on a range of different variables. As an illustration, you'll need to foresee what kind of smartphone a customer could want to buy. In these circumstances, the variable has many values. Here is how a traditional classification tree appears.

Figure 6 - Classification Tree Example

1. **Regression Decision Trees:**

Decision tree regression trains a model in the form of a tree to predict data in the future and provide useful continuous output by observing the properties of an item. Continuous output denotes the absence of discrete output, i.e., output that is not only represented by a discrete, well-known set of numbers or values.

It is different from the classification Tree because the classification tree is utilised when the dataset must be divided into classes that are a component of the response variable. The courses are frequently either Yes or No. In other words, there are only two of them, and they are incompatible. There may also be quite two classes in some situations, in which case a variation of the classification tree approach is used (Sabbarwal, 2020).

On the other hand, regression trees are utilised when the response variable is continuous. Regression trees are used, for instance, when the response variable is anything like a property's price or the day's temperature.

As a result, I'm going to utilise a regression decision tree for this project since the output is dependent on the characteristics of the insect larvae and their environments, such as temperature, weight, length, and developmental phases, rather than on binary values. Even though the output is not binary, it changes significantly as the input does. When we are forecasting the outcome, regression trees are utilised. So, based on my research, this method is appropriate for this project.

Additionally, with the aid of this algorithm, I can give users a clearer and more comprehensible visual representation of the expected results and how this algorithm predicts value. This algorithm fits with this dataset with an 88% accuracy score due to the accuracy factor in the algorithm.

# **METHODOLOGY**

First, raw data on insect larval behaviour at various rearing temperatures is gathered. With the aid of several Python tools and modules, this data is filtered and pre-processed. In this study, machine learning is used to determine the larvae's age. In this study, a machine learns from several larval traits, including weight, length, and developmental stages, to estimate the age of the larvae. The decision tree method has been applied to achieve this. All the characteristics of the algorithm are initially utilised to train the algorithm using the dataset, after which the features are decreased, and the behaviour of the algorithm in predicting the age of the larvae is examined (Pothuganti Manvitha, 2019).

1. **Data Gathering**

The data was collected by Forensic Entomologist in laboratory conditions and was passed on to the software developer to build an application with help of this data. Now, I got this dataset from my supervisor to make a model with a different approach which is more precise and user-friendly. This data set contains different characteristics of the insect larvae, which helps us to predict the age of the larvae.

1. **Primary Processing of data**

This process is very important to get the most precise output because sometimes the data is in the raw format, which is not feasible for the analysis. The technique is to clean the data from its raw form. So, firstly I have done some attribute selection in which some of the attributes in the initial dataset that are not pertinent (relevant) to the experimental goals were ignored. The attributes id, series, run, cup, and batch are not used. The main attributes used for this study are hours, weight, length, temperature, and larval stage.

Then, the process of cleaning the values has been applied because, in some cases, the dataset contains some missing values, but we can’t remove the whole row for that missing value because we may lose some data with that. So, I have used a library for this task which is Scikit Learn pre-processing. It contains a class called Imputer which will help us to take care of the missing data.

1. **Processing with decision tree regression algorithm**

Regression occurs when a goal value yields a numerical result rather than a predetermined set of labels. Predicting new target values based on training data is the major issue here. The classification method used with decision trees and the regression methodology is extremely similar. The splitting measurement is the primary difference. Regression trees employ mean squared error in place of impurity (MSE). MSE is steadily decreased until it is at its lowest level by questions. The average target value derived from all samples in the node serves as its outcome. MSE is determined as follows:

Another way to think about the decision tree regression algorithm is as an ensemble machine learning approach. A collection of records with characteristics serves as the input to a regression algorithm. The information is divided up into a few subgroups. A decision tree regression will be built for each subset that was formed. The algorithm that employs the majority vote approach will choose the final class of a test record. The decision tree regression algorithm uses the out-of-bag error method.

The algorithm below is used to build each tree:

1. Let N represent the total number of training examples, and M represent the total number of classifier variables.
2. The amount number of input variables (m) to be considered in deciding at a tree node is given to us; m should be much less than M.
3. Choose N times with replacement from the N available training instances to create a training set for this tree (i.e., take a bootstrap sample). By anticipating their classes, use the remaining examples to estimate the tree's error.
4. Randomly select m variables to use as the decision-making basis for each node in the tree. Based on these m training-set characteristics, determine the optimal split.
5. Every tree is mature and unpruned (as may be done in constructing a standard tree classifier)

After fitting, the model assigns the average target value and directs a new object to the appropriate branch. Simple yet effective. A regression tree has been finished.

1. **Building of Regression tree**

To understand how a regression tree works and makes the decision, let’s consider a random dataset where we have two variables first X, which in this project, temperature, weight, length and larval stage and the second variable is Y which we have to predict as an output, in this project it implies as hours or the age of larvae, as shown in below fig. 6 (Prasad, 2021):

Chart, scatter chart

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Weight | Temperature | Larval stage | Length | Hours |
| 0.0005 | 15 | 1 | 0.20 | 0 |
| 0.0005 | 15 | 1 | 0.20 | 0 |
| 0.0005 | 15 | 1 | 0.21 | 0 |
| 0.0005 | 15 | 1 | 0.22 | 0 |
| 0.0005 | 15 | 1 | 0.23 | 0 |
| 0.0005 | 15 | 1 | 0.31 | 0 |
| 0.0005 | 15 | 1 | 0.34 | 0 |
| 0.0005 | 15 | 1 | 0.34 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Y | 1 | 1.2 | 1.4 | 1.1 | 1 | 5.5 | 6.1 | 6.7 | 6.4 | 6 | 6 | 3 | 3.2 | 3.1 |

Figure 7 - Example Dataset, X is a continuous variable and Y is another continuous variable

Figure 8- Example Data Table

Figure 9- Original Dataset of Insect Larvae Characteristics

By the help of the example dataset which shown in fig. 8, I am providing an understanding of how regression tree works with following steps because the original dataset is too large to explain. A small part of data set is shown in the above fig. 9.

**Step 1**

Data is initially sorted according to X in the first phase (In this case, it is already sorted). Next, calculate the average of the first two rows in variable X, which, given the dataset, is (1+2)/2 = 1.5. Separate the dataset into two equal pieces (Parts A and B) by the distances x 1.5 and X 1.5.

The first row (1,1) is the lone point in Part A now; all other points are in Part B. Take the averaging of all the Y values in Parts A and B individually at this point. These two numbers represent the decision tree's anticipated results for the values of x 1.5 and x 1.5, respectively. Calculate the mean square error and record it using the original and anticipated values.

**Step 2**

In step 1, we divided the dataset based on the average for the first two numbers of the sorted X and computed the predictions. Then, using the same procedure a second time, we determine the standard of the second and third sorted X numbers ((2+3)/2 = 2.5). Then, we divide the dataset into Parts A and B based on X 2.5, forecast the results, and calculate the mean square error as demonstrated in step 1. This procedure is repeated until the third, fourth, fifth, sixth, seventh, and n-1st two numbers (where n is the number of records or rows in the dataset).

**Step 3**

We must now determine the point at which to divide the sample after computing the n-1 mean squared errors. The location that caused the lowest mean squared error when splitting at it is that one. In this instance, x=5.5 is the point. Therefore, the tree will be divided into two sections. x<5.5 and x≥ 5.5. The Root node is chosen in this method, and the data points that point to the Root node's left and right children are then recursively subjected to the same procedure to split them further.

The brief idea of what this regression algorithm is doing is to find the point in the independent variable where the data set should be divided into two pieces so that the mean squared error is minimised. There is a fundamental notion underlying the procedure. This is done by the algorithm repeatedly, creating a structure like a tree.

A picture containing diagram

Description automatically generatedThe regression tree from the example data set works is shown in fig. 10, and part of the original tree visualisation is shown in fig. 11 with the actual dataset.



Figure 10 - Example Tree Representation

Diagram, engineering drawing

Description automatically generatedSimilarly, by the understanding of the above example (fig. 10), which is with 2 variables, I have applied this algorithm on 4 variables where the split starts from the weight attribute (fig. 11). The process is similar; at each node, all the 4 variables would go through the same process as what X went through. The data would be sorted based on the 4 variables separately. For each of the four variables, the positions that minimise the MSE are determined. The variable with the least mean square error would be picked from the three given by their estimated points.

Figure 11 - Tree Visualization of Original Dataset

# **IMPLEMENTATION AND TOOL USED**

To implement the above algorithms different python libraries are being used. With use of decision tree regression algorithm, I have built two model for both online and offline users because if the end-users using this application at the crime scene where there is no internet connection even though they can have the prediction with available input from the dead body instantly. To implement both this model, I have used the following libraries and tools:

## PANDAS LIBRARY

The functions offered by the PANDAS library are numerous. In addition to offering a wide range of features for practically any activity, it aids in data processing. The user is given a descriptive statistical overview of all the dataset's characteristics. Even if the data may be understood with it, there aren't many functionalities available. With many features and customization options, it allows report production for the dataset by PANDAS profiling.

In this project, the PANDAS library is used to fetch the data from the dataset, which is in excel format, to the PANDAS data frame. After which, this data was filtered according to the required attributes from the whole dataset, where the irrelevant columns were dropped from the data frame and replaced some of the temporary numbers with their original number. For example, the temperature is numbered with 1, 2, and 3 values and was replaced with 25, 20, and 15, respectively.

## SCIKIT-LEARN LIBRARY

The assessment and validation of models is one of the most important components of supervised machine learning. It's crucial that the procedure be impartial when we assess our model's capacity for prediction. Our dataset may be divided into subsets using train\_test\_split() from the data science toolkit scikit-learn to reduce the possibility of bias throughout the evaluation and validation process (Robinson, 2022).

The coefficient of determination, root-mean-square error, mean absolute error, and other comparable metrics are often used in regression analysis. We frequently use accuracy, precision, recall, F1 score, and related measures for classification issues.

1. **Preparing the Data**

In this part, we will separate our data into characteristics and labels and then into training and test sets using the resulting data. This allows us to train our algorithm on one piece of data while testing it on another set of data that it hasn't seen before. This gives you a more precise idea of how your trained algorithm will work in practice.

By executing the following code in Appendix 1(lines 1-2) below, we divide the data into attributes and labels.

With dropping all the irrelevant columns and the "Hours" column, which serves as the label, in this case, is the only column from the dataset that isn't included in the X variable. The "Hours" column's values are included in the y variable. Our attribute set is represented by the X variable, and the labels are included in the Y variable.

The separation of our data into training and test sets is the last pre-processing step. We'll use the train\_test\_split function from the Scikit-Learn model selection package to randomly divide the data into training and testing groups. To achieve this, we need to run the following code in Appendix 1(lines 3-4).

The test\_size argument in the code above defines the ratio of the test set, which we use to divide the data into a test set of 33% and a training set of 67%.

1. **Training and Making Prediction**

The decision tree algorithm will then be trained on this data and predictions will be made when the data has been split into the training and testing sets. The tree library, which is part of Scikit-Learn, has built-in classes and methods for several decision tree algorithms. We will use the DecisionTreeRegressor class in this project since we will be doing a regression operation. To train the algorithm using the training data supplied as an argument to the fit function, this class's fit method is invoked. After running the script, which is shown in Appendix 1(lines 5-7). Now, the comparison between predicted and actual data is shown in the result section.

## PYTHON FLASK

I created an application using the Python Flask framework that allows users to quickly verify an insect larvae’s predicted age when the user is connected to the internet. After testing out a few different tools for the model, the flask is the best-fitting framework, in my opinion, which provides a better user experience. In Appendix 2, the various tools that were tested and their limitations are displayed.

A lightweight WSGI web application framework is Flask. With the ability to scale up to complex applications, it is meant to be easy to get started with. It was once only a simple wrapper for Werkzeug and Jinja, but it has since developed into one of the most well-liked Python online application frameworks.

Deployment of the machine learning model is the best usage of the Flask framework. Depending on whether the package argument resolves to an actual Python package (a folder containing a \_\_init .py file) or a standard module, the name of the package is used to resolve resources from inside the package or the folder the module is housed in (just a .py file) (Projects, 2022).

Graphical user interface, text

Description automatically generatedDisplay of the webpage is shown in fig. 12 below:

Figure 12 - Display of UI Application for online users

## PYTHON GUI TKINTER

I have created another application model for offline users because if the user wants to have a prediction instantly at the crime scene where there is no internet connection. They can be able to have a predicted age of the insect larvae. To do so, I have implemented my decision tree algorithm with the very known python library GUI (Graphical User Interface) Tkinter.

Tkinter is Python's standard GUI framework and is frequently included with the language. It is well-known for its ease of use and graphical user interface. It's a free source and distributed under the Python License (Foundation., 2022).

Graphical user interface, application

Description automatically generatedThe display of the webpage is shown in fig. 13 below:

Figure 13 - GUI application for Offline users

## DTREEVIZ LIBRARY

The decision tree models' algorithms function by continually dividing the data into various sub-spaces, ensuring that the results in each final sub-space are as homogenous as feasible. Technically, this method is known as recursive partitioning. The programme attempts to divide the data into subsets that are as pure or homogenous as feasible.

Therefore, with dtreeviz, we can see how the feature space is divided at decision nodes, how training samples are dispersed in leaf nodes, how the tree produces predictions for a single observation and more. These processes are crucial for understanding how classification or regression decision tree algorithm.

So, the dtreeviz library helps the end-user to visualise the prediction and which path is followed by the algorithm to reach the predicted value by making decisions for every split.

Diagram

Description automatically generatedTo understand how the visualisation looks when the user clicks on the link, which is shown in fig. 12 and 13, one example of the tree and path visualisation is shown in fig. 14 and 15, respectively.

Figure 14 - Tree Visualization of the predicted value

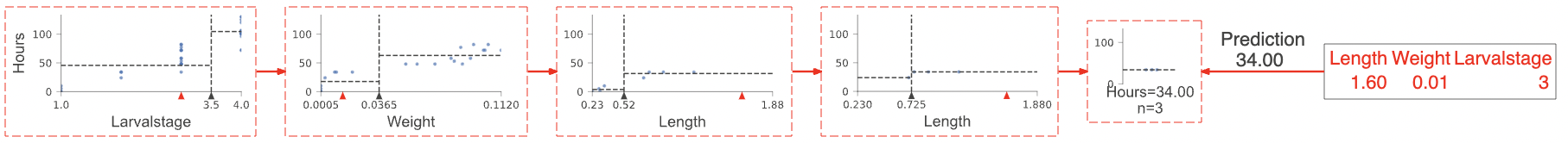


Figure 15 - Path visualisation of the predicted value

# **EVALUATION**

In this section, the evaluation of the project is described, which was evaluated by the Computer Science department students. By providing brief information and knowledge of my project to the evaluator, I have got some valuable feedback and ratings on my application.

The evaluator feedback was mixed; some of the feedback about the UI design are:

* The UI is minimalistic and easy to understand.
* The offline model design could be more informative, but the understandability is fine.
* The project is capable to explain the desired objectives.

Some feedback on the implementation and working of the application.

* The loading time after clicking the submit button is high and there is no loader icon when the page is calculating the output.
* Amazing details and unique application of decision tree.
* The choice of visualisation structure is good but can be minimalistic.
* The implementation of this project is meticulous. Visualisation can be able to understand how the algorithm predicts the age of the larvae.

The average rating of the application is shown in fig. 16 below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S. No.** | **Points for the Rating** | **Ev-1 (1-5)** | **Ev-2 (1-5)** | **Ev-3 (1-5)** | **Ev-4 (1-5)** | **Avg.**  **(1-5)** |
| 1 | Online Website Design | 4 | 5 | 5 | 3 | 4 |
| 2 | How Interactive the Online Website | 3 | 4 | 5 | 2 | 3.5 |
| 3 | Showcase of the Input on Online Website | 4 | 3 | 4 | 2 | 3.25 |
| 4 | Showcase of the Output on Online Website | 5 | 5 | 5 | 5 | 5 |
| 5 | Output Tree Visualization on Online Website | 5 | 4 | 5 | 4 | 4.5 |
| 6 | Output Path Visualization on Online Website | 5 | 4 | 3 | 4 | 4 |
| 7 | Offline GUI model Design | 4 | 5 | 5 | 3 | 4 |
| 8 | Showcase of the Input on Offline GUI model | 3 | 3 | 4 | 3 | 3.25 |
| 9 | Showcase of the Output on Offline GUI model | 3 | 5 | 5 | 3 | 3.5 |
| 10 | Output Tree Visualization on Offline GUI model | 3 | 4 | 3 | 2 | 3.25 |
| 11 | Output Path Visualization on Offline GUI model | 3 | 5 | 5 | 2 | 4 |
| 12 | Rate the Understanding of the Project | 5 | 5 | 5 | 5 | 5 |

Figure 16 - Evaluation Ratings

# **RESULTS**

In this section, after applying several different approaches for application, the final two models are created. One is an online model when there is an internet connection, the end-user can check the age of an insect larva by putting the characteristics of the larvae, and this model is made on the Python Flask framework. Another model which is helps the end-user to get the prediction anywhere because it is an offline model for the prediction which is built on the Python Tkinter GUI framework.

The algorithm which is used in this model is the decision tree regression algorithm, and to get the final prediction, we need to follow the following tools, libraries, and steps, which were described in the Implementation section above.

The cross-validation of the dataset by splitting the train and test set by 67% and 33%, respectively, I got 88% accuracy in my prediction. To understand the difference between the actual and predicted output for the age of insect larvae in this application, some of the values are shown below in fig. 17.

|  |  |  |
| --- | --- | --- |
|  | **ACTUAL**  **(hrs)** | **PREDICTED**  **(hrs)** |
| 88 | 58 | 82 |
| 145 | 106 | 106 |
| 11 | 5 | 2.5 |
| 74 | 53 | 58 |
| 173 | 130 | 125 |
| 150 | 120 | 101 |
| 151 | 120 | 101 |
| 97 | 77 | 82 |
| 90 | 72 | 72 |
| 96 | 77 | 72 |
| 147 | 106 | 101 |
| 65 | 48 | 58 |
| 139 | 101 | 120 |

Figure 17 - Actual and Predicted age of the insect larvae

# **CONCLUSION**

This project concludes with the main aim of the project, which is to build an application to estimate the age of insect larvae for forensic entomology. They can predict the time of death of a wounded, dead body by the larvae they get from the wounds of that body after analysing several different approaches and algorithms for the prediction. I have finalised the decision tree algorithm for the application. I have built my model with 88% accuracy of the predicted value for a better end-user experience.

## LIMITATIONS

There are a few limitations in the dataset part of the project, the dataset might not be based on the latest collected dataset, but according to the given set, the prediction is entirely appropriate.

In the implementation, I haven’t used the most advanced frameworks, and the UI design is quite simplistic and can be more workable and interactive for the end-users.

The visualisation of the output does not fit the desktop screen because the tree is based on enormous data. Therefore, there might be some libraries which can be used to get more user-friendly visualisation.

## FUTURE WORK

From a future work perspective, I suggest that one should go through the latest research on the dataset. As this project is built in python language, there is quite a scope for better UI design and better visualisation of the output. The improvement of the samples taken by law enforcement personnel depends equally or even more on the promotion of forensic entomology among its end-users and on the education of the officers or medical examiners who collect insect evidence on death scenes. Although we should look for more optimal guidelines for insect sampling on a death scene, this is a scientific task. But with the help of this type of new research, one should get a better estimate of the output. One can even try the Weka tool, which is based on JAVA language and can directly contact the desired application on whatever required algorithm they want. Some of the descriptions are shown in Appendix 2 for WEKA software.

# **REFERENCES**

A. Bianconi, C. V. Z. A. S. J. G., 2010. Artificial neural networks: A novel approach to analysing the nutritional ecology of a blowfly species. Chrysomya megacephala, Journal of Insect. Science, Volume 10, p. 2010.

Ames C, T. B., 2003. Low temperature episodes in development of blowflies: implications for postmortem interval estimation.. *Med Vet Entomol ,* pp. 178-186.

Analytics, N., 2022. *16 personalities.* [Online]   
Available at: https://www.16personalities.com/free-personality-test

Anon., 2019. *MatWorks.* [Online]   
Available at: https://www.mathworks.com/help/stats/machine-learning-in-matlab.html  
[Accessed 22 Auguest 2022].

Associates, B., n.d. *Belbin.* [Online]   
Available at: https://www.belbin.com/about/belbin-team-roles

B, G., 1991. Flies as forensic indicators. *J Med Entomol,* Volume 28(5), pp. 565-577.

B, G., 1991. Flies as forensic indicators.. *J Med Entomol ,* pp. 565-577.

Brodley, C. E., 1995. *The process of applying machine learning algorithms,* West Lafayette, IN : ACADEMIA.

C. Day, J. A. J. B. P. H., 2009. Element-specific determination of x-ray transmission signatures using neural networks. *NondisT&E Int,* Volume 42(5), pp. 446-451.

Chauhan, N. S., 2022. *KDnuggets.* [Online]   
Available at: https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html  
[Accessed 02 09 2022].

Chen W, Y. L. R. L. e. a., 2019. mpact of constant versus fluctuating temperatures on the development and life history parameters of Aldrichina grahami (Diptera: Calliphoridae). *Insects,* Volume 10(7).

Davies L, R. G., 1994. Development rates of some pre-adult stages in blowflies with reference to low temperatures.. *Med Vet Entomol,* p. 245–254.

Foundation., P. S., 2022. *Python Documentation.* [Online]   
Available at: https://docs.python.org/3/library/tkinter.html  
[Accessed 30 08 2022].

Grassberger M, R. C., 2001. Effect of temperature on Lucilia sericata (Diptera: Calliphoridae) development with special reference to the isomegalen- and isomorphen-diagram.. *Forensic Sci Int,* pp. 32-36.

Greenberg B, K. J., 2005. Entomology and the law: flies as forensic indicators, First digitally printed in paperback. *Cambridge University Press.*

H.J. Kim, W. B. a. Y. J., 2011. Identification of ambiguous cubeb fruit by DART- MS-based fingerprinting combined with principal component analysis. *Food Chemistry,* Issue 129, pp. 1305-1310.

Hannah Moore, P. et al., 2015. *Age estimation of Calliphorida (Diptera: Calliphoridae) larvae using cuticular hydrocarbon analysis and Artificial Neural Networks,* Keele: Keele Univeristy.

J. Amendt, C. C. E. G. C. R. H. L. a. M. H., 2007. Best practice in forensic entomology-standards and guidelines, International Journal of Legal Medicine 121. pp. 90-104.

J. Butcher, D. V. B. S. C. D. a. P. H., 2010. Pruning reservoirs with random static projections. *IEEE workshop on Machine Learning for Signal Processing,* pp. 250-255.

J. Butcher, D. V. B. S. C. D. P. H., 2014. Defect detection in reinforced concrete using Reservoir Computing and Extreme Learning Machines. *Computer-Aided Civil and Infrastructure Engineering,* Volume 29, pp. 191-207.

J. Butcher, H. M. C. D. C. A. a. F. D., 2013. Artificial Neural Network analysis of hydrocarbon profiles for the ageing of Lucilia sericata for Post Mortem Interval estimation. *Forensic Science International,* Volume 232, pp. 25-31.

J. Hajslova, T. C. a. L. V., 2011. Challenging applications offered by direct analysis in real time (DART) in food-quality and safety analysis. *TrAC Trends in Analytical Chemistry,* Issue 30, pp. 204-218.

J.B. Butcher, H. M. C. D. F. D. a. C. A., 2012. Artificial Neural Network analysis of hydrocarbon profiles for the ageing of Lucilia sericata for Post Mortem Interval estimations.

J.B. Butcher, M. L. C. D. a. P. H., 2009. A low frequency electromagentic probe for detection of corrosion in steel-reinforced concrete. *Grantham, C. Majorana, and V. Salomoni, eds., Concrete Solutions, CRC Press, NL,* p. 417–424.

J.Y. Yew, R. C. a. E. a. K., 2008. Cuticular hydrocarbon analysis of an awake behaving fly using direct analysis in real-time time-of-flight mass spectrometry. *Proceedings of the National Academy of Sciences of the United States of America,* Issue 105, pp. 7135-40.

Kohonen, T., 1990. Self-organising map. *Proceedings of the IEEE,* p. 1464–1480.

Kohonen, T., 1990. The self-organising map. *Proceedings of the IEEE,* Volume 78(9), pp. 1464-1480.

Larson, R. S. a. R., 2009. Validation of the direct analysis in real time source for use in forensic drug screening. *Journal of forensic sciences,* Issue 54, pp. 612-22.

M. Baqué, J. A. J. B. C. D., 2015. *Progress towards the automated analysis of Calliphora vicina larval morphology and temperature data for age estimation.* Frankfurt, Germany, Keele University.

M. Beyeler, N. O. N. D. a. J. K., 2015. A GPU-accelerated cortical neural network model for visually guided robot navigation. *Neural Networks,* Volume 72, pp. 72-87.

Maheshwari, N., n.d. *nmg.* [Online]   
Available at: https://nmgtechnologies.com/blog/requirement-gathering-solve-biggest-problems-consulting.html

Pais M, A. M., 2018. Histological age estimation of the eggs of Calliphora vicina Robineau Desvoidy. *Forensic sciences research,* Volume 3(1), pp. 40-51.

Pathak, S., Mishra, I. & Swetapadma, A., 2018. *An Assessment of Decision Tree based Classification and Regression Algorithms,* Coimbatore, India: IEEE.

Point, T., 2022. *Tutorial Point simply easy learing.* [Online]   
Available at: https://www.tutorialspoint.com/weka/weka\_quick\_guide.htm  
[Accessed 02 09 2022].

Pothuganti Manvitha, N. S., 2019. *Campus Placement Prediction Using Supervised Machine Learning Techniques ,* Chennai, India: Research India Publications.

Prasad, A., 2021. *Regression Trees | Decision Tree for Regression | Machine Learning.* [Online]   
Available at: https://medium.com/analytics-vidhya/regression-trees-decision-tree-for-regression-machine-learning-e4d7525d8047  
[Accessed 25 08 2022].

Projects, P., 2022. *Flask API application object.* [Online]   
Available at: https://flask.palletsprojects.com/en/1.1.x/api/#flask.Flask  
[Accessed 29 08 2022].

R.B. Cody, J. L. J. N. a. H. D., 2007. Direct Analysis in Real Time (DART) Mass Spectrometry. *JEOL.*

Reiter, M. G. a. C., 2001. Effect of temperature on Lucilia sericata (Diptera: Calliphoridae) development with special reference to the isomegalen- and isomorphen- diagram. *Forensic Science International,* Volume 120, pp. 32-6.

Robinson, S., 2022. *Decision Trees in Python with Scikit-Learn.* [Online]   
Available at: https://stackabuse.com/decision-trees-in-python-with-scikit-learn/  
[Accessed 27 08 2022].

Sabbarwal, M., 2020. *What is Classification & Regression Trees?.* [Online]   
Available at: https://www.codingninjas.com/blog/2020/12/02/what-is-classification-regression-trees/  
[Accessed 25 08 2022].

Shang Y, R. L. Y. L. e. a., 2020. Differential gene expression for age estimation of forensically important Sarcophaga peregrina (Diptera: Sarcophagidae) intrapuparial.. *J Med Entomol,* Volume 1, pp. 65-77.

Sovocool, A. G. a. G., 2011. Detection of illicit drugs on surfaces using direct analysis in real time (DART) time-of-flight mass spectrometry. *Rapid communications in mass spectrometry : RCM,* Issue 25, p. 1271–81.

Tarone AM, F. D., 2011. Gene expression during blow fly development: improving the precision of age estimates in forensic entomology. *J Forensic Sci.*

team, G. L., 2022. *Great Learning.* [Online]   
Available at: https://www.mygreatlearning.com/blog/what-is-machine-learning/#3  
[Accessed 20 08 2022].

Teamwork, n.d. *Teamwork management.* [Online]   
Available at: https://www.teamwork.com/project-management-guide/project-management-methodologies/

Warren J-A, A. G., 2013. Effect of fluctuating temperatures on the development of a forensically important blow fly, Protophormia terraenovae (Diptera: Calliphoridae). *Environ Entomol,* p. 167–172.

Zajac BK, A. J., 2012. Bestimmung des Alters forensisch relevanter Fliegenpuppen. *Rechtsmedizin,* pp. 456-465.

Zajac BK, A. J. H. R. e. a., 2015. De novo transcriptome analysis and highly sensitive digital gene expression profiling of Calliphora vicina (Diptera: Calliphoridae) pupae using MACE (Massive Analysis of cDNA Ends). *Forensic Sci Int Genet,* p. 137–146.

Zajac BK, A. J. V. M. e. a., 2018. Dating pupae of the blow fly Calliphora vicina Robineau-Desvoidy 1830 (Diptera: Calliphoridae) for post mortem interval-estimation: validation of molecular age markers. *Genes.*

# **APPENDIX 1**

LINE 1 - X = df.drop(["Hours","ID","Series","Run","Cup","Batch","Temperature"], axis=1).values

LINE 2 - y = df["Hours"]

LINE 3 - from sklearn.model\_selection import train\_test\_split

LINE 4 - X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.33, random\_state=100)

LINE 5 - from sklearn.tree import DecisionTreeRegressor

LINE 6 - regressor = DecisionTreeRegressor()

LINE 7 - regressor.fit(X\_train, y\_train)

# **APPENDIX 2**

Here are some ambulation studies during the project.

**WEKA Software:** WEKA is an open-source programme that offers tools for preparing data, implementing several machine learning algorithms, and providing visualisation tools so that you may create machine learning methods and use them to solve real-world data mining issues (Point, 2022). A vital tool for building machine learning models is WEKA. It offers the implementation of several of the most popular ML algorithms. You may also pre-process the data before these methods are applied to your dataset. The Classify, Cluster, Associate, and Select properties are used to categorise the many available algorithms. A stunning and potent visual representation may be used to visualise the outcome at different processing stages. A Data Scientist will find it simpler to quickly apply several machine learning approaches to his dataset, evaluate the results, and develop the best model for the intended purpose.

# **APPENDIX 3**

To access Model one (Online webpage model on Flask)

* Follow the Path and Run: - Flask Model for Online User/myApplication.py.
* After that, go to your browser and paste <http://127.0.0.1:5000/viz> and enter.

To access Model two (Offline model on GUI Tkinter)

* Follow the Path in the folder: - GUI model for Offline Users/main.py file.