Using Machine Learning to Predict the First Visibility of the New Crescent Moon

Neil Power
Department of Physics and Astronomy
University of Manchester
Student ID: 10623021
neil.power-2@student.manchester.ac.uk

This project was performed in collaboration with Ezzy Cross,
and supervised by Dr. Emma Alexander and Prof. Anna Scaife.

Abstract

The first visibility of the new crescent Moon has significant cultural and historical importance worldwide. First visibility cannot be calculated exactly, but machine learning models can be used to create predictions. A dataset of new crescent Moon observations was compiled from previous astronomical studies and crowd-sourced observation projects. For each observation, 17 features of the Moon were calculated, and the dataset used to train several machine learning models in order to predict the visibility of the new crescent Moon. Gradient boosted trees using XGBoost were found to be the most accurate model, achieving an accuracy of 92.2 ± 0.8% for predicting visibility by the naked eye. On the same data, the most commonly used mathematical predictor, Yallop's criterion, has an accuracy of 78.8%. The XGBoost model was extended to predict whether visibility was possible by eye or visual aid, such as a telescope, with an accuracy of 83.1 ± 1.0%. From this model, visibility maps were made in order to predict the first visibility of the new crescent Moon.

Contents

1 Introduction **3**

2 Properties of the Moon **3**
2.1 Date and Coordinates . . . 3
2.2 Sunset, Moonset and Lag Time . . . 3
2.3 Moon Age and Best Time . . . 4
2.4 ARCV, ARCL and DAZ . . . 4
2.5 Parallax, Semi-Diameter and Topocentric Width . . . 4

3 Moon Visibility Criteria **5**
3.1 Historical Moon Visibility Criteria . . . 5
3.2 Yallop's Criterion . . . 5

4 Machine Learning for Data Analysis **6**
4.1 Random Forest . . . 6
4.2 Gradient Boosted Forests . . . 7
4.3 Neural Networks . . . 7
4.4 Machine Learning and the Moon . . . 8

5 Observational Data and Data Pre-Processing **9**
5.1 Sources of Data . . . 9
5.2 Distribution of Sightings . . . 9
5.3 Validation . . . 11
5.4 Calculating Properties of the Moon . . . 11
5.5 Labelling Observations . . . 11

6 Evaluation of Yallop's Criterion **11**
6.1 Relevance of ARCV and Width . . . 11
6.2 Accuracy of Yallop's Criterion . . . 12
6.3 Visualising Moon Visibility . . . 13

7 Building Machine Learning Models **13**
7.1 Machine Learning Metrics . . . 13
7.2 Neural Networks . . . 14
7.3 Comparison of Machine Learning Methods . . . 15

8 Results: Testing the Predictive Models **16**
8.1 Feature Importance . . . 16
8.2 Multi-Label Predictions . . . 17
8.3 Cloud Cover and Weather . . . 18
8.4 Visibility Predictions . . . 18

9 Conclusion **19**
9.1 Future Work . . . 19

General Notes **20**

References **20**

Appendix **22**

1 Introduction

The Moon has been used to track the passage of time for thousands of years. A series of excavations dating from 8000 BC in Warren Field are believed to have used the Moon to track lunar months, potentially making it one of the world's oldest calendars [1]. The Moon is of particular significance for the Islamic Hijri calendar, where the visibility of the new crescent typically signifies the start of each new month. As a result, new crescent sightings are important to mark the start and end of Islamic holidays and observations, such as fasting during the month of Ramadan.

The first visibility of the new crescent Moon is the first time the Moon is visible after the new Moon. Unlike the new Moon, the date and time of the first visible crescent cannot be calculated exactly. There are many factors that determine when the first crescent is visible, from astrophysical properties such as the angles between the Sun and the Moon, to atmospheric properties such as the weather at the time of observation.

As a result of its significance to lunar calendars, different criteria have been developed to predict the visibility of the new crescent Moon. Many of these were made from observations of the new crescent, and as a result several datasets of new crescent sightings are available. There are also projects collecting crowd-sourced observational data. The availability of observational data and the lack of a direct mathematical method to predict visibility means that machine learning is a useful approach for this problem. Figure 1 outlines several key objectives in order to investigate this.

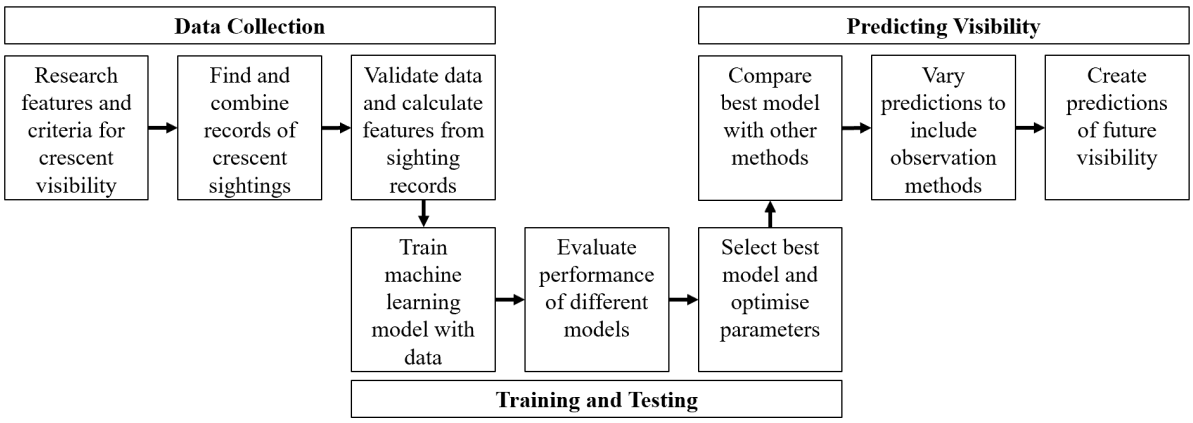


Figure 1: Objectives for this project.

2 Properties of the Moon

This section outlines the features and properties of the Moon, and their relevance when predicting the visibility of the new crescent.

2.1 Date and Coordinates

The visibility of the crescent will vary depending on date, latitude and longitude. The date and time of observation will dictate the position of the Earth, Sun and Moon, whereas the position of the observer will determine the relative angles between the three bodies, as well as the time of sunset and moonset. The Julian date format was used throughout this project, as this allows easy comparison between dates. The ISO 6709 [2] convention was used for latitude and longitude, with latitude ranging between +90°(N) and -90°(S), and longitude ranging between -180°(W) and +180°(E).

As position relative to the horizon is important for visibility, the horizontal coordinate system of altitude and azimuth were used. The position of an object in the sky can be described by its altitude, the angle between the object and the horizon, and its azimuth, the angle of the object around the horizon from north [3] Both of these are measured locally, and therefore depend on the position of the observer.

2.2 Sunset, Moonset and Lag Time

As the Moon must be visible in the sky in order for a new crescent to be seen, the times of moonset and sunset are important. The lag time is defined as the difference between sunset and moonset, with

a positive lag time corresponding to a moonset occurring after sunset. For the first sighting of the new crescent moon, a positive lag time is required, corresponding to an observation made in the evening. The United States Naval Observatory definitions of sunset and moonset were used [4].

2.3 Moon Age and Best Time

Typically, the first visibility of the lunar crescent occurs between 1 and 3 days after the new Moon [5]. The age of the Moon is defined as the number of hours since the new Moon.

The new crescent will typically be visible only for a short period each day. Over this period, there will be a best observational time, when the crescent is at its most visible. Bruin's method [6] was used to calculate the best observation time T_b from the sunset T_s and moonset T_m. This method is defined in Equation 1, and finds the time with the best contrast between the background sky and the Moon, determined from observational records.

$$T_b = \frac{5}{9}T_s + \frac{4}{9}T_m \tag{1}$$

2.4 ARCV, ARCL and DAZ

Figure 2 shows the relative locations of the Sun and Moon in terms of altitude and azimuth. The arc of light (ARCL) is defined as the angular separation between the centres of the Sun and the Moon. The arc of vision (ARCV) is the angular difference in altitude between the centres of Sun and the Moon, whilst the relative azimuth (DAZ) is the angular difference in azimuth [7]. These are used frequently in historical literature, and so it is useful to continue using them for ease of comparison.

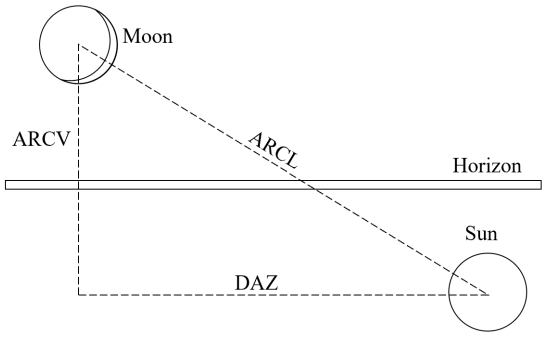


Figure 2: Relative positions of the Sun, Moon and horizon.

2.5 Parallax, Semi-Diameter and Topocentric Width

The geocentric lunar parallax, π is defined as the angle subtended by the Moon over the radius of the Earth [8]. It has been used historically to calculate the distance of the Moon, and can be calculated from the Earth-Moon distance, d_{Moon}, the radius of the Earth, r_{Earth} and the zenith angle of the Moon, z, using Equation 2.

$$\sin\pi = \frac{r_{Earth}}{d_{Moon}}\sin z \tag{2}$$

The semi-diameter of the moon, SD, is half of the angular width of the Moon as observed from Earth [9]. Equation 3 shows how this can be calculated from the lunar parallax.

$$SD = 0.27245\pi \tag{3}$$

The semi-diameter can be converted to a topocentric semi-diameter SD', accounting for the position of an observer, using Equation 4, where h is the Moon's geocentric altitude [5].

$$SD' = SD(1 + \sin h \sin\pi) \tag{4}$$

The topocentric width of the crescent, W', defined as the width of the lit area of the Moon as measured along the Moon's diameter, can be calculated using Equation 5 [6].

$$W' = SD'(1 - \cos(ARCL)) \tag{5}$$

The part of the Moon illuminated (phase), I can be similarly approximated in terms of ARCL [8] using Equation 6 .

$$I = \frac{1}{2}(1 - \cos ARCL) \tag{6}$$

3 Moon Visibility Criteria

3.1 Historical Moon Visibility Criteria

Many different methods have been used to predict the visibility of the new crescent Moon throughout history. Some of the earliest methods date from the Babylonian era, requiring the age of the Moon to be greater than 24 hours, and the lag time to be more than 48 minutes [10]. These persisted for several hundred years, with Hindu texts building on the Babylonian criteria by considering the width of the crescent [6]. In the 9th and 10th centuries, Islamic astronomers also researched the importance of the longitude and altitude separation between the Sun and the Moon, as well as the distances between the Sun, Moon and Earth.

In the early 1900s, astronomers such as J. K. Fotheringham [7] and E. W. Maunder [11] began to use recordings of historical sightings to predict visibility. They suggested ARCL values of 10° and 11° respectively were required in order for the new crescent to be visible. In 1932, French astronomer A. Danjon proposed the widely used 'Danjon Limit' of 7°, using 76 observations, some the same as Fotheringham and Maunder [12]. Several other ARCL limits have also been proposed, such as M. Ilyas's value of 10.5° for visibility specifically by the naked eye [10]. In 2004, M S. Odeh suggested a Danjon limit of 6.4°, although this was based off a single observation [13].

In order for more detailed analysis to be possible, larger sighting datasets than those used by Danjon and earlier astronomers were needed. In 1988, B. E. Schaefer [14] compiled a dataset of 201 observations from various astronomical literature, and filtered through the observations to remove or update any impossible sightings. Recognising the effects that geographical location has on observations, Schaefer calculated parameters such as ARCL, ARCV and DAZ for a 'best observation time'. Schaefer's dataset, updated in 1996 [15], has formed the basis for many visibility criteria, such as Yallop [5] and Odeh [13].

One of the first numerical criteria to determine and predict the visibility of the new Moon was set out by Fotheringham [7] and Maunder [11]. They calculated values of ARCV and DAZ at sunset for positive and negative sightings, and then used this to calculate values of ARCV needed for values of DAZ in order for the crescent to be visible. In 1997, F. Bruin [6] considered both the width of the lit crescent and ARCV as key factors determining visibility, fitting a cubic polynomial to a plot of W against ARCV to create a visible or invisible prediction.

3.2 Yallop's Criterion

In 1997, Yallop [5] set out one of the most widely used criteria for the visibility of the new crescent Moon, with a 'q-value' criterion based on Bruin's method, with the addition of calculating parameters at a best time. This formula, shown in Equation 7 is the difference between ARCV and a cubic equation of width, found by fitting to values of ARCV from positive observations. Values of q fall within defined ranges, from easily visible (A) to not visible and below the Danjon limit (E), as shown in Table 1. Yallop's criteria is widely used for current predictive models, such as the visibility maps published on the Islamic Crescents Observation Project website [16].

$$q = \frac{1}{10}\left(ARCV - \left(11.8371 - 6.3226W' + 0.7319W'^2 - 0.1018W'^3\right)\right) \tag{7}$$

Table 1: Interpreting q from Yallop's formula

Category	q value	Visibility Code	Description
A	$q > +0.216$	V	Easily visible (ARCL $\geq 12°$)
B	$+0.216 \geq q > -0.014$	V(V)	Visible under perfect conditions
C	$-0.014 \geq q > -0.160$	V(F)	May need optical aid
D	$-0.160 \geq q > -0.232$	I(V)	Will need optical aid
E	$-0.232 \geq q > -0.293$	I(I)	Not visible with a telescope
F	$-0.293 \geq q$	I	Not visible (ARCL $\leq 8°$)

4 Machine Learning for Data Analysis

Machine learning models can be given input data, learn from it, and create predictions without explicit programming. They are a valuable tool in astrophysics [17], where large quantities of data are measured and often have correlations too complex to fit using purely numerical methods. Predicting the visibility of the new crescent Moon is a complex problem, as the Moon has many different properties that can contribute to its visibility. These are difficult to combine into a single formula, as shown by the tens of different criteria developed over the last couple of decades. Machine learning can be used to create a model that can include many more features, and account for more complex relationships between the features and visibility. It is also difficult to differentiate when the Moon is visible by eye or requires visual aid with numerical criteria. A machine learning model can use a large number of real-world observations by different methods to predict the exact method by which the crescent will be visible.

4.1 Random Forest

When evaluating the performance of different machine learning models for new crescent sighting data, Al-Rajab et al. [18] found a random forest machine learning model to be one of the most effective. Random forests were first formulated as a model in 1995 by T. K Ho [19], and then extended by L Breiman [20] in 2001. Random forests are an ensemble method, where they learn by combining the results predicted by many decision trees. A decision tree is a simple machine learning model, in which a tree is constructed out of conditional statements about each of the input variables. The tree consists of an input root node, many internal nodes and branches, and output leaf nodes.

Decision trees are built by using one of several possible algorithms to decide when and where to split the tree in order to make the best decision. Typically, these decisions are based on entropy, information gain and/or Gini impurity [21]. Entropy can be considered as the expectation value of surprise, where surprise is inversely proportional to probability and a metric of how unlikely a selection of data is. From this, the information gain can be calculated, which represents how much information each feature can provide about the overall label. Trees can be built by maximising information gain at each split, in order to select the feature that has the highest impact on the result at each point in the tree. Another way to build decision trees is using Gini impurity. This is the probability of misclassifying a random data point, and the option with the lowest impurity can be used to create a split.

In random forests, many different decision trees are built. Each is built with a random subset of data using bagging, where samples of data that can include multiple instances of the same points are selected [22]. As well as this, random forests have feature randomness, where samples of features are also selected in the same way. Random forest can be used as both a classification algorithm, with discrete labels, or a regression algorithm, with continuous numerical predictions. For classification, the label predicted by the most individual trees is chosen, whereas for regression the output of all the different trees is averaged and this is taken as the overall prediction. Figure 3 shows an example of a simple random forest.

Random forests typically avoid overfitting, which is a common disadvantage of decision tree-based models. Data does not need to be scaled or normalised, but may struggle to fit unbalanced datasets. Due to its random nature, random forests are non-deterministic, so different trees will be built with slightly different results every time the model is trained on the same data.

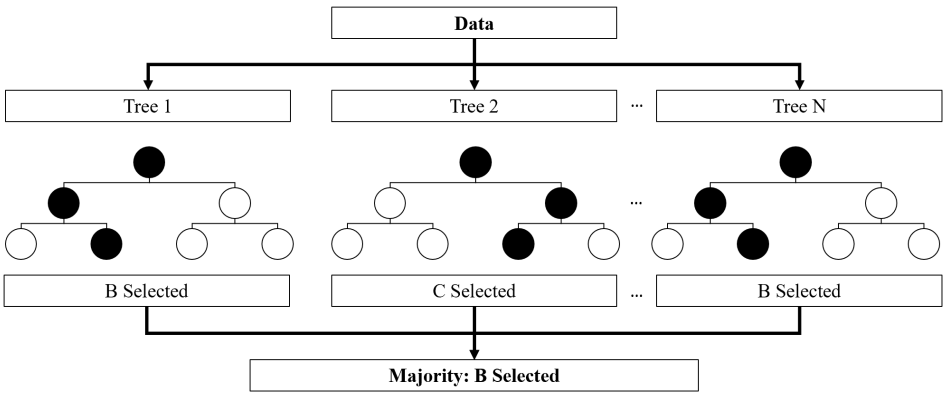


Figure 3: Simple visualisation of a random forest model, showing several decision trees and how the majority prediction is used to create an overall prediction.

4.2 Gradient Boosted Forests

Another form of decision-tree based ensemble methods is gradient boosted forests. These are similar to random forests, as a number of individual decision trees are constructed. However, they are built and optimised in different ways. With gradient boosting, a series of decision trees are constructed, and the residuals of each model are used to train the next one. This process takes place using a gradient descent algorithm, an optimisation algorithm used to find the minima or maxima of a function [23]. The function to be minimised is the loss function, usually one that calculates the error residuals, defined as the difference between the actual label and predicted label.

By minimising the error residuals, optimal values for each tree's splits can be selected. At every step in building a tree, many splits for many features can be analysed, and the one with the lowest residuals selected. The weighted sum of each tree's prediction is used to create an overall prediction. Gradient boosting can support unbalanced datasets, and is less sensitive to hyperparameters, as these only determine the first tree, rather than every tree. It is also less likely to overfit testing data than random forests. Both random forests and gradient boosting methods adapt well to distributed computing, and have similar performance.

Extreme Gradient Boosting, or XGBoost [24], is a specific implementation of gradient boosting, optimised for performance and computational efficiency. XGBoost minimises an objective function, not a loss function in order to build successively improving trees. This objective function weights larger trees lower, essentially discouraging overly complex trees and overfitting. When optimising to minimise residuals, XGBoost also uses both first and second order approximations for the residual function.

4.3 Neural Networks

Artificial neural networks are models that attempt to replicate the way neurons work in the brain, and are increasingly used in astronomy for classification problems. A neural network learns by passing data through neurons, or nodes. Each node applies a complex activation function to the data to determine how the data is passed on to the next nodes, eventually reaching output nodes that indicate the prediction. The parameters of these functions are then adjusted repeatedly as the model learns.

A neural network first applies an activation function to weighted input data [25]. This is the first layer of the network. Depending on the function outputs, the data is then passed to nodes in the next layer, where each node once again applies an activation function before passing the data onto the next layer. The central layer(s) are known as hidden layers. The data will then reach the output layer, which contains one node for each possible label of the data. The output node selected represents the model's prediction. At each stage, the weights of the data and parameters of the nodes in each layer are varied in order to minimise the residuals between the actual data and the neural network's predictions. Activation functions are typically non-linear, in order for complex relationships to be represented over multiple layers. An example of a popular activation function to transform the input layer is a rectified linear unit (ReLU) function, which when given a positive input will return the same input, returning zero otherwise [26].

Figure 4 shows every layer in the network consists of a series of neurons, similar to the nodes in a decision tree. Neurons have weighted connections through layers. The network is trained using backpropagation, where these weights are varied in order to minimise a loss function, which represents the difference between the actual data and predictions [27]. Neural networks typically have a set learning rate, which determines how weights are varied during training. Similarly, the weight decay of a network is a parameter which prevents the weights becoming too complex and overfitting the data. There are many different algorithms used for backpropagation. One is stochastic gradient descent, which approximates the change in gradient needed to minimise loss. Adam [28] is an extension to stochastic gradient descent, varying the learning rates for every individual weight, rather than the entire model. The structure of a neural network, including the number of hidden layers and number of neurons per hidden layer can be varied to tune the network to a problem. Other parameters of a neural network include the number of epochs (number of times that the entire training dataset is passed to the model), and the batch size (number of samples passed through the network before the parameters of the network are changed).

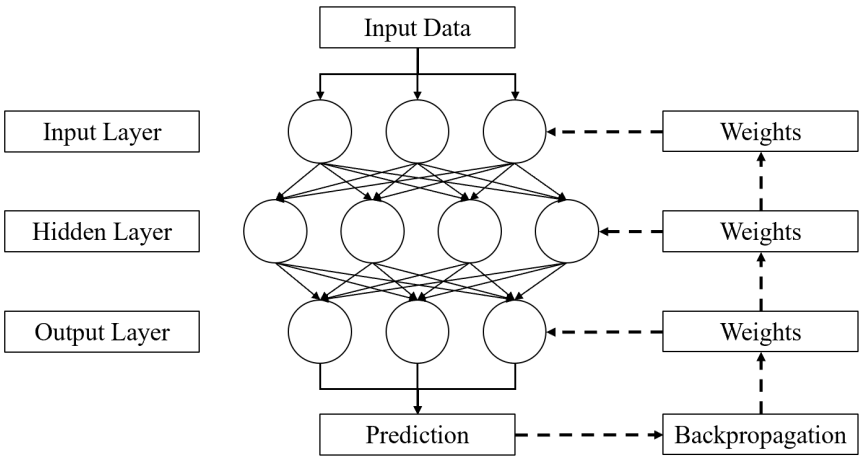


Figure 4: Simple diagram showing the layout of an artificial neural network, with one hidden layer and a loop to repeatedly update node weights and parameters.

4.4 Machine Learning and the Moon

There have been several recent attempts to apply machine learning to the problem of predicting the new crescent Moon. In 2018, T. Ahmed [29] trained several machine learning algorithms on a dataset of 1522 global observations. The age, lag, ARCL, DAZ and phase were calculated for each measurement, and then these were used as features for four different types of supervised machine learning models. Accuracies of 83-86% were obtained. The study considered only observations from the dataset without cloud cover, but included an atmospheric parameter ranging from superb to very hazy. Different methods of observation beyond naked eye were also not considered. The study was one of the first attempts to link together new crescent visibility and machine learning, but acknowledged that more data from more locations was needed.

In 2022, Z. T. Allawi [30] used machine learning in order to create a four-year calendar to predict visibility for Baghdad. A dataset of 366 global observations (most from Schaefer and Odeh) was used. A single hidden layer neural network was used to classify sightings as either visible, probably visible, or visible. Allawi used only observations taken during clear weather, and used W, ARCL and ARCV as input parameters for the model. A test accuracy of 72.7% was achieved, and the model was used to create a calendar of the dates of monthly first visibility in Baghdad for the next four years.

M. Al-Rajab et al. [18] in 2023 evaluated the performance of different machine learning algorithms at predicting crescent visibility. They applied eight algorithms available in the Python `scikit-learn` library [31] to a dataset of 350 crescent Moon sightings. Decision tree algorithms such as random forest were found to be most effective, with an accuracy of 93%. Features used included ARCL, ARCV, Moon altitude, lag and a similar atmospheric factor to Ahmed [29]. The overall dataset used was not balanced, as 70% of the observations were negative, and this was identified as an improvement for future work.

All three studies are limited by the amount of available data, a crucial factor when training a machine learning model. Many of the datasets used also had a significant imbalance between positive and negative sightings, and this can affect how certain machine learning models learn. Combining the sources used, and other available data, into a single large set of observations may yield better results. Typically, the predictions using when creating a predictive model are labels (visible or not visible), whereas observational data usually includes the method of observation (such as the naked eye, or telescope). This could be incorporated into a predictive model. The studies also focused on predicting visibility with clear weather, excluding any positive or negative sightings made during cloudy days. Ahmed also mentions that the atmosphere feature could be better calculated using local temperature and humidity. The weather, specifically clouds, is a significant factor in determining whether the new crescent Moon is visible, and the wide variety of historical meteorological data available from sources such as the Met Office means that weather trends could be incorporated into models in some capacity.

5 Observational Data and Data Pre-Processing

Four different sources of data were used. Each recorded positive and negative observations of the new crescent Moon, useful for building an accurate predictive model.

5.1 Sources of Data

The Islamic Crescents Observation Project (ICOP), part of the International Astronomical Center (IAC) [16] has been collecting observations since 1998. Members can submit positive or negative observations directly to the ICOP website, where they are then published by country for each Islamic month. The 'ICOP' dataset refers to a collection of 2133 observations from 2009 to 2020, digitised by T. Ahmed [29]. This data includes information about the cloud level, as either 'clear' or 'partly cloudy', as well as either a 'true' or 'false' label for visibility by any of the following methods: naked eye, binoculars, telescope and CCD cameras. It also contained some Moon properties, such as lag time and ARCV, but these were recalculated from the date, latitude and longitude.

The 'ICOP23' dataset consists of post-2020 ICOP observations that were not included in Ahmed's original dataset, up to November 2023. 1747 observations were retrieved from the ICOP website [16] using Python-based HTML extraction. The Python `geopy` geocoding module [32] was then used to calculate the latitude and longitude of each observation from its recorded town or city. This data contains the same features as the ICOP dataset, including the same cloud level and visibility labels.

The Islamic Crescents Observation for the UK (ICOUK) was established in 2008, and the 'ICOUK' dataset contains digitised records of 651 crescent observations in the United Kingdom [33]. This dataset contains either a 'clear', 'partly cloudy' or 'totally cloudy' label for cloud. Each observation also has a method label: either not seen, seen by naked eye, seen by binoculars, seen by telescope or a combination of these.

In 2018, Alrefay et al. [34] published a criterion using observations from astronomers at the Saudi Arabia National Centre for Astronomy, part of King Abdul Aziz City for Science and Technology (KACST). The 'Alrefay' dataset contains 476 positive and negative new crescent observations. All observations are from Saudi Arabia, and were only recorded if the sky was clear. Sightings are labelled as visible either by the naked eye or telescope.

A collection of 295 sightings was collated by Schaefer in 1988 [14] and 1996 [15]. These were used by Yallop in 1997 to devise Yallop's criterion. Yallop corrected a number of errors in the data, such as incorrect new Moon dates, and so the dataset published by Yallop [5] was used, rather than Schaefer's original recordings. In this 'Yallop' dataset, positive and negative observations were only recorded during clear skies. This dataset spans a wide time period, from the mid 1800s to the 1990s. Older Moon sightings may be more difficult to calculate parameters for, but are useful in order to ensure models are as general as possible.

The observations used by Schaefer [15] and Odeh [13] were combined into a single dataset of 366 observations by Z. T. Allawi in 2022 [30]. This was not used, as most of this data was duplicated in the ICOP and Yallop datasets.

The ICOP, ICOP23, ICOUK, Alrefay and Yallop datasets were combined, providing a total of 5302 observations as summarised in Table 2 .

Table 2: Sources of data and datasets of new crescent Moon observations.

Identifier	Collated by	Source	Region	Date Range	Count
ICOUK	Direct from source	ICOUK [33]	UK	2015-2023	651
ICOP	Ahmed [29]	ICOP [16]	Global	2009-2020	2133
ICOP23	Direct from source	ICOP [16]	Global	2020-2023	1747
Alrefay	Alrefay et al. [34]	KACST	Saudi Arabia	1988-2015	476
Yallop	Yallop [5]	Schaefer [14], [15]	Global	1859-1996	295

5.2 Distribution of Sightings

Each individual dataset had a different distribution of positive and negative sightings, but when combined the data was well-balanced, with 52% positive and 48% negative. The distribution of sightings by method is shown in Figure 5. Most positive observations were by the naked eye (37% of all observations), with 15% of all observations classed as a positive observation with visual aid (binoculars, telescope or CCD camera).

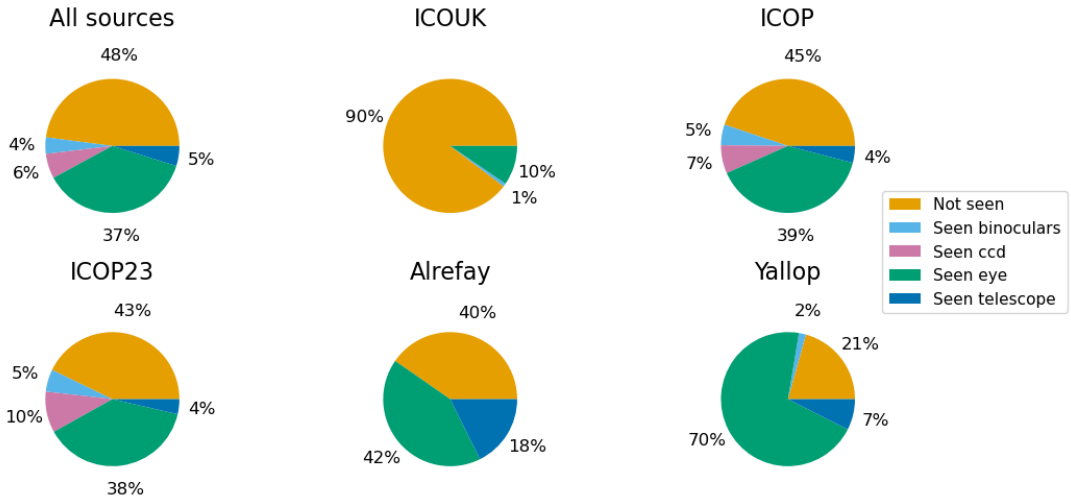


Figure 5: The distribution of new crescent Moon observations by sighting method for each dataset.

The geographic distribution of the observations are shown in Figure 6. The Yallop dataset is predominately from Western Europe and North America, whereas the ICOP datasets mostly include observations from the Middle East and North Africa. The ICOUK data only includes the UK, and the Alrefay dataset only includes Saudi Arabia. Less data is available for the southern hemisphere, including South America and Central/Southern Africa.

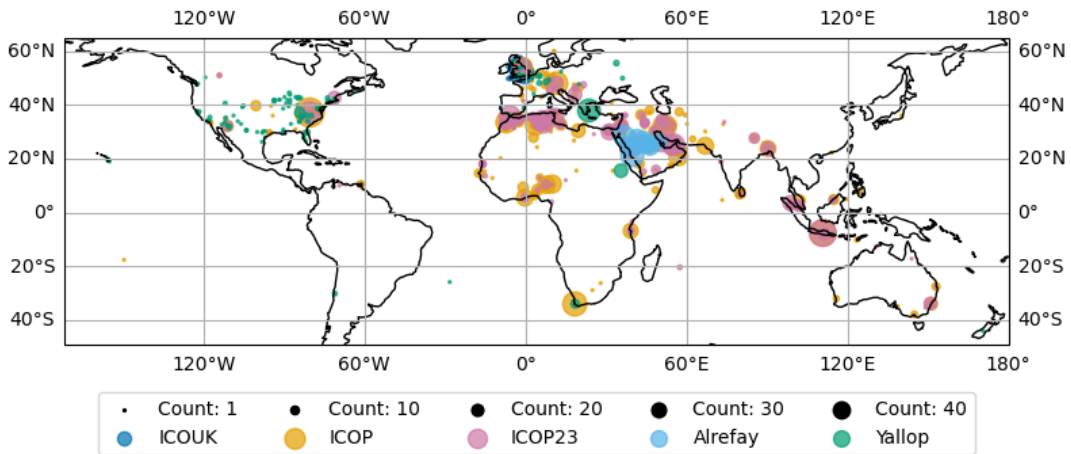


Figure 6: The geographic distribution of observations, with the size of each point proportional to the number of observations.

The dates at which observations took place are shown in Figure 7. The Yallop data spans the widest time range, from 1859 to 1996. Most observations took place in the last 20 years, when observations such as the ICOP and ICOUK started collecting online records of observations. This is useful in order to make future predictions, as there will be many recent observations for models to learn from.

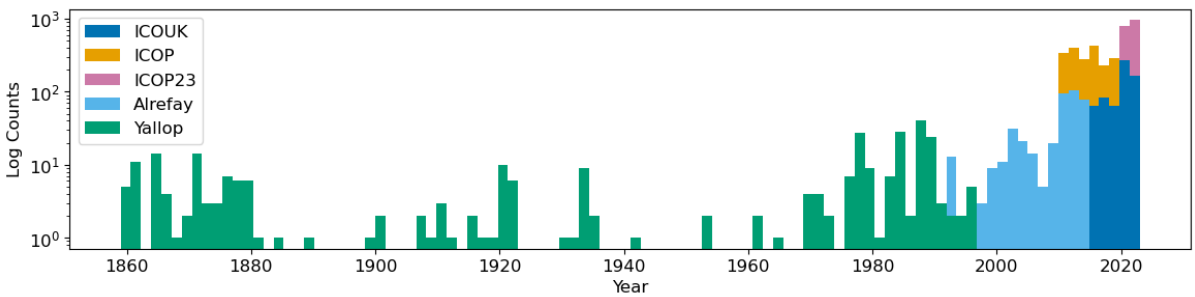


Figure 7: The distribution of observations over time.

5.3 Validation

Each of the five datasets were checked for duplicates against each of the other datasets. This was done by checking for entries with the same date, longitude and latitude to ensure the same sighting was not counted twice. No duplicates were found. There were occasionally observations at the same time and location within the same dataset, but all of these were found to be separate observations at the same location. For example, in the ICOUK dataset, there were often multiple observations from London by different observers on the same day. The observations were plotted on a world map to manually check the locations, and several corrections were made. For example, every ICOUK observation at Saltburn-by-the-Sea, North Yorkshire, was listed with a longitude of 0.9732° (east), when this should have been -0.9732° (west).

Yallop's criterion was also calculated for every data point in the dataset, and 1.8% of positive observations were found to be below the Danjon limit, mostly from the ICOP and ICOUK. Only three of these were observations with the naked eye, and these observations were included as the Danjon limit is not an absolute constraint. Some of the observations will also have been observations of the waning crescent shortly before the new Moon. These can be distinguished by a negative lag time, which approximately 8% of observations had. Whilst this is not the first visibility, many of the same factors apply when predicting visibility, and so these sightings were not removed.

5.4 Calculating Properties of the Moon

When the location of an observation was not given as latitude and longitude, `geopy` [32] was used to look up the coordinates using OpenStreetMap's Nominatim API. The `astropy` [35] library was used for storing and processing times and angles, storing dates and times as Julian dates. Observation dates in the ICOP, ICOUK and Alrefay datasets were dates in UTC, whereas the dates in the Yallop dataset were the local dates as measured in the country of observation. These were converted to Julian dates by using the `timezonefinder` [36] library to find the time zone for each location. The time of sunset and moonset were found using the `skyfield` [37] library's almanac functionality. The DE430 [38] ephemeris (table of object positions over time) was used, covering 1850 to 2150, which spans the entire set of observations. Equation 1, Bruin's best time formula, was used to calculate the best time of observation.

The `skyfield` almanac was also used to calculate the time and date of the last new Moon, and from this the age of the Moon. The `astropy` library was used to calculate the position of the Sun and Moon in horizontal coordinates for a specific time and location. From these, ARCL and DAZ could be calculated as the difference in altitude and azimuth. The angular separation of the Sun and Moon, ARCV, was also calculated using `astropy`, as well as the distances between the Sun, Moon and Earth. Parallax was calculated from the Earth-Moon distance using Equation 2. The semi-diameter of the Moon and its topocentric width were calculated using Equations 4 and 5. The illumination of the Moon was calculated using Equation 6.

5.5 Labelling Observations

Observations in the dataset were labelled by 'Method', corresponding to the method of observation: either eye, binoculars, telescope, CCD camera or not seen. Also included was a 'Methods' label, containing every method that the Moon was visible by. For example, if the Moon was seen by binoculars, it would also be visible by telescope and CCD, but not the naked eye. This made the assumption that a telescope would generally be better than binoculars for viewing the Moon, which should generally apply for all but the worst telescopes and best binoculars. These labels allow the combined dataset to be filtered into a 'naked eye visibility' dataset with 4518 'seen' or 'not seen' observations, or a 'optical aid' dataset that combines this with observations by optical aid as an extra 'seen with aid' category.

6 Evaluation of Yallop's Criterion

Yallop's criterion is one of the most widely used criteria for predicting new crescent visibility, and provides a useful comparison to evaluate machine learning models.

6.1 Relevance of ARCV and Width

The two variables in Yallop's criterion are ARCV and the width of the crescent, W. These were calculated for every observation in the combined dataset, and plotted against each other in Figure 8, with the points

coloured by visibility method. Clusters of positive sightings are visible at higher values of ARCV and W, with negative sightings generally at lower values. However, the data is significantly scattered, suggesting more factors affect the visibility of the new crescent than those considered by Yallop.

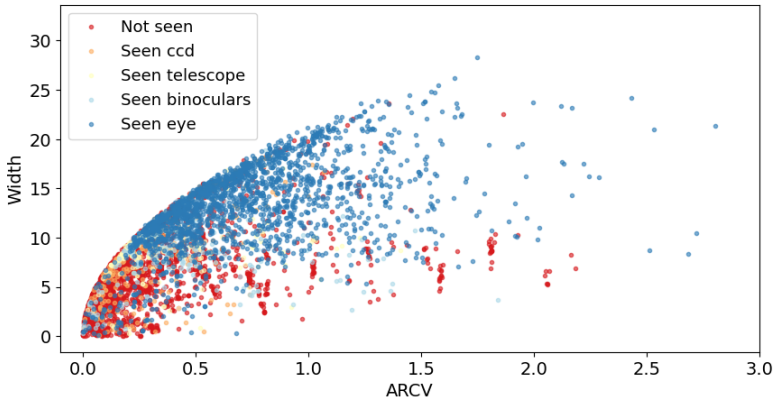


Figure 8: Values of ARCV against topocentric width for observations.

6.2 Accuracy of Yallop's Criterion

The accuracy of Yallop's criterion was estimated by comparing predicted q-values with the combined dataset observations. Initially, the ranges V to V(F) were considered a positive prediction (up to 'may need optical aid'), and I(V) to I as a negative prediction. Predictions were compared with actual observations, and the accuracies shown in Table 3. For a more complex model including sightings with visual aid, range I(F) ('will need optical aid') was considered a positive prediction with aid. The 97.1% accuracy for the Alrefay dataset shows that Yallop's criterion has excellent accuracy under perfect conditions, but lower accuracies occur for datasets with more varied locations and weather conditions. Predictions including sightings with visual aid have a slightly lower accuracy, most likely as predictions of 'may need optical aid' are difficult to account for.

Table 3: Accuracy of Yallop's criterion

Dataset	Accuracy	
	Naked Eye	Naked Eye & Visual Aid
All data	0.788	0.759
ICOUK	0.665	0.551
ICOP	0.802	0.782
ICOP23	0.763	0.741
Alrefay	0.971	0.971
Yallop	0.820	0.807

Figure 9 shows the range of q-values for positive and negative observations. The central 75% of the data for both sightings is distinctly separated. However, there is considerable overlap between the lower quartile of positive sightings and the central quartiles of negative sightings. This overlap suggests that the observers have seen the Moon when a prediction is not theoretically possible by Yallop's criterion. This may either be due to observer error, or suggest that Yallop's criterion may slightly underestimate visibility. An overlap involving the upper quartile of the negative plot is expected, as a theoretically possible observation can still be negative due to factors such as cloud cover.

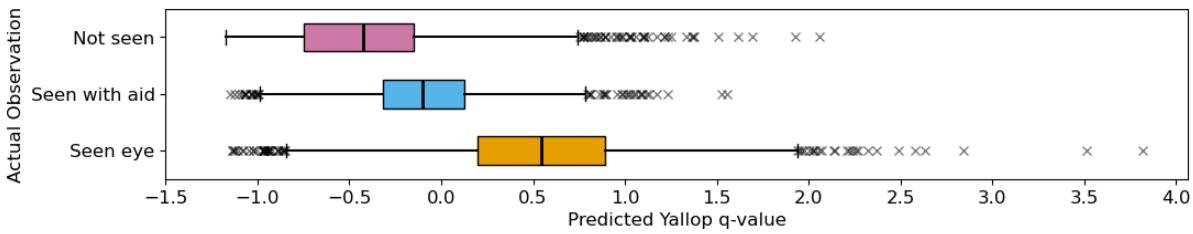


Figure 9: Predicted Yallop's criterion q-values against observations by naked eye or visual aid.

Figure 10 shows predicted q-values for exact observation methods, showing significant overlap between the visual aid methods: binoculars, telescope and CCD. This implies that these categories are less distinct than the three categories of seen, seen with aid and not seen, and much less distinct than a binary seen or not seen model. The amount of aid that binoculars and telescopes (of any brand or type) provide cannot be categorised exactly, and so this may cause issues for models attempting to create predictions by exact method.

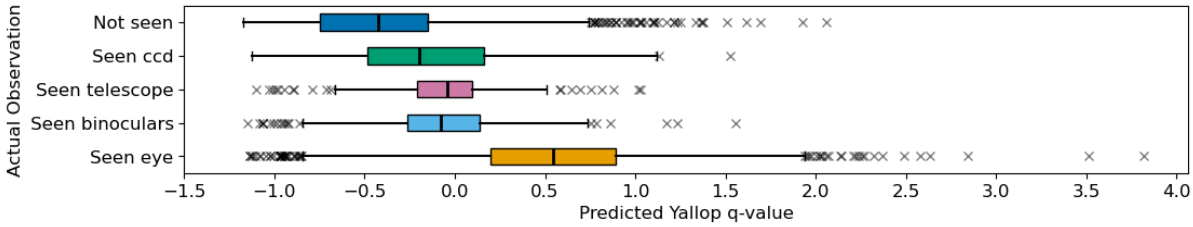


Figure 10: Predicted Yallop's criterion q-values against exact method of observation.

6.3 Visualising Moon Visibility

Contour maps are a popular way to show the predicted visibility of the new crescent Moon. Figure 11 was created by calculating ARCV and W at best time for a 100 by 100 grid of points, from -60° to +60° latitude and -180° to +180° longitude. Yallop's criterion for each point was then calculated using Equation 7. The contours were set to the visibility ranges in Table 1, ranging from easily visible with the naked eye, to invisible even with visual aid. An interactive version of this map was made using the Python `geovista` library [39], allowing a user to zoom and pan over a 3D globe showing satellite imagery.

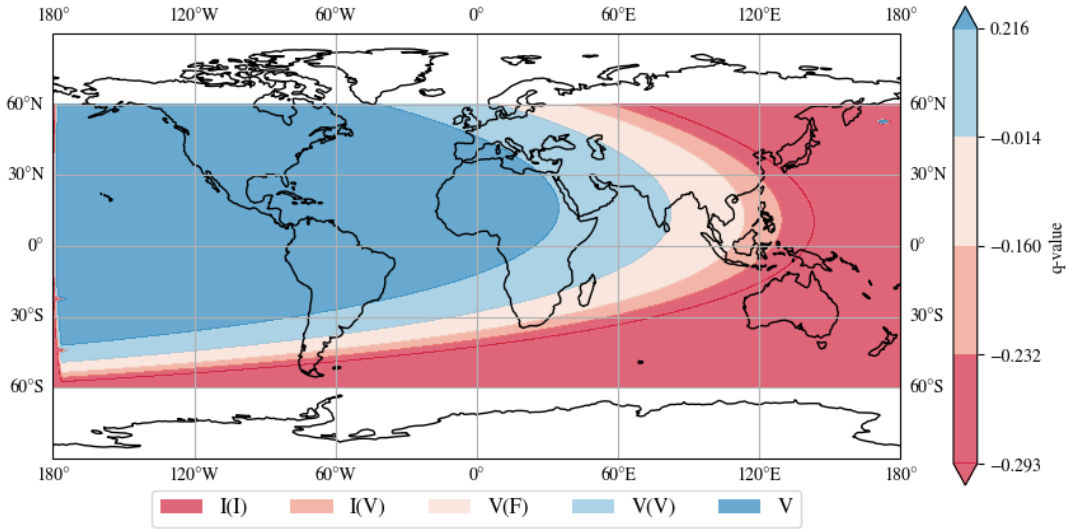


Figure 11: Contour plot for the 22nd of March 2023, showing predicted visibility according to Yallop's criterion by location.

7 Building Machine Learning Models

The dataset of combined and validated observational records of new crescent Moon sightings was then used to train different machine learning models.

7.1 Machine Learning Metrics

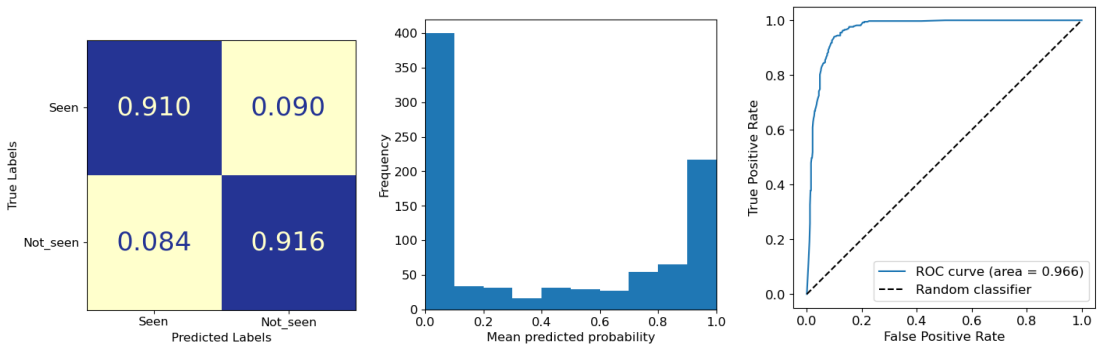
There are a number of different ways to evaluate how 'good' a machine learning model is. The most common is accuracy, or how many of the predicted labels match the true labels on the testing data. When training a model, the dataset is usually split into subsets of training and test data, to ensure the model

is tested on unseen data. If the data selected for training is randomly selected from the dataset, the accuracy of a model will vary slightly every time it is trained. To account for this, models can be trained on the same set of training data each time, and parameters can be set to pre-set the 'random' selection of features and subsets for models such as random forest. Another way is repeated cross validation, where the model is run a number of times, selecting a different set of training and testing data every time. The average accuracy and other metrics can be calculated, along with their standard deviations, showing how much the model varies on a typical run. Some common metrics [40] include:

- Accuracy: the proportion of predicted labels that match the true labels
- Receiver Operating Characteristic (ROC): area beneath a curve of true positive labels against false positive labels
- Recall: the proportion of predicted positive (seen) labels that match the true positive labels
- Precision: the likelihood that positive predictions are actually correct
- F1 score: the harmonic mean of the precision and recall

Every machine learning model has hyperparameters, parameters that set the properties of the model and how it learns. As a initial test, the `scikit-learn` library's implementation of a random forest classifier [31] was used with its default hyperparameters. 80% of the data was selected at random for training the model, with the remaining 20% retained for testing. The dataset included 20 features: date, latitude, longitude, Moon age, sunset, moonset, lag, Moon altitude and azimuth, Sun altitude and azimuth, Moon-Earth distance, Sun-Moon distance, ARCL, ARCV, DAZ, illumination, parallax, width and cloud level. The date, sunset and moonset were removed, leaving 17 features used to train the model. As an initial test, only positive or negative observations by eye were included, reducing the dataset from 5302 to 4518. Either 'seen' or 'not seen' was passed to the model as a label, so the model could learn to classify observations into one of the two categories. An average accuracy of 91.2% $\pm$ 0.8 was found. Attempts to train the model on only the ICOUK dataset (90% negative) were unsuccessful, as decision tree based models typically require a balanced dataset. The combined dataset has a near perfect split of 52% negative and 48% positive sightings, so balancing the data was not necessary.

One way to visualise model predictions is a confusion matrix, as shown in Figure 12a. This is a table of actual labels against predicted labels, with each cell of the table coloured corresponding to the proportion of matching labels. The plot shows how the initial random forests model is able to separate positive and negative observations well, with around 9% of positive sightings misclassified as negative, and 8% for the opposite. Alternatively, the overlap between predictions can be displayed in a probability histogram such as Figure 12b, which plots the mean probability of positive predictions. This shows that most predictions were distinctly classed as not seen (probability of 0), or seen (probability of 1) with a small degree of overlap indicating uncertain predictions. Figure 12c shows the ROC curve for the initial random forest model, where the area of 0.966 shows a very high true positive rate with few false positives.



(a) Confusion matrix (b) Probability histogram (c) ROC curve

Figure 12: Graphs displaying metrics of prediction accuracy for a random forests model

7.2 Neural Networks

An artificial neural network was built using the `PyTorch` library [41]. This network consisted of an input layer, a single hidden layer and an output layer. The Adam optimisation algorithm [28] was used for backpropagation in order to update the weights of the model during training. A rectified linear unit

function (ReLU) was used as an activation function to transform the data between each layer. Initially, a hidden layer with 8 neurons was used, halfway between the number of neurons in the input layer (17) and output layer (2). An initial learning rate of 0.1, and weight decay of 0 were used.

The binary naked eye visibility dataset was split into sections for training (70%), validation (15%) and testing (15%). This three-way split means that validation data can be used to test different parameters of the model, before comparing the final model with the testing dataset. This model did not successfully learn on the data, guessing 'seen' 100% of the time, and so the hyperparameters (variables used to create the network) needed to be adjusted for the training data. Multiple hyperparameters were adjusted simultaneously through Bayesian optimisation using the Weights and Biases framework [42]. A number of sweeps were made over ranges of hyperparameters, with each step slightly varying the hyperparameters based on previous results to minimise test loss.

The hyperparameters were tuned using Bayesian optimisation to minimise test loss, a representation of the errors made by the network when predicting labels for test data. The number of hidden layers (1-3), the number of neurons in the hidden layer (4-32), the number of epochs (50-500), learning rate (0-0.1), weight decay (0-0.1) and batch size (20-260) were all varied. The best model with the lowest test loss was a single hidden layer network, with 32 neurons in the hidden layer, 300 epochs and a batch size of 60. A best learning rate of 0.005 and best weight decay of 0.01 was found. Figure 13 shows the training of this neural network. The neural network achieved a reasonably high accuracy of 89.0 $\pm$ 1.4%, but lower than the random forests model. The neural network is also significantly more computationally intensive, taking minutes rather than seconds to run, as well as being more complex to implement in Python.

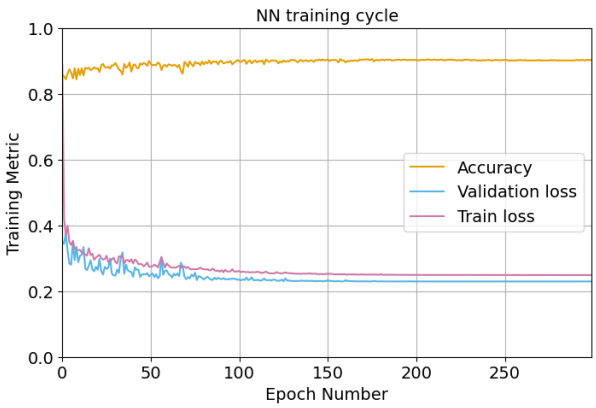


Figure 13: The accuracy, training and validation losses over epochs of the ANN model.

7.3 Comparison of Machine Learning Methods

Allawi et al. [30] found that decision tree based models were effective for predicting new crescent sightings. To determine the best model for our dataset, nine different models were trained and tested against each other. Five decision tree models were used in order to determine the best subtype, with three non-decision tree models and a neural network for comparison. The models were trained and tested on the naked eye visibility dataset with 'seen' by naked eye or 'not seen' labels. All models apart from XGBoost and the neural network were implemented using the `scikit-learn` library, using a 80-20 train-test split. All models also used their default hyperparameters, with the exception of the neural network which required the tuning outlined in Section 7.2 in order to learn. The models used include:

- RandomForest: a random forests classifier, made up of an ensemble of decision trees with different features and subsets of data.
- XGBoost: a gradient boosted decision tree, using the `XGBoost` library.
- NeuralNetwork: an artificial neural network, using the `PyTorch` library.
- DecisionTree: a simple decision tree classifier, built by minimising Gini impurity.
- ExtraTrees: a decision tree ensemble classifier similar to random forests, except the trees are selected completely randomly rather than calculating an optimised selection of features.
- AdaBoost: a boosted decision tree classifier similar to XGBoost, except successive trees are built by improving misclassification, rather than error residuals

- NearestNeighbours: this algorithm creates predictions for a point by considering what labels the point's neighbours have.
- NaiveBayes: this creates predictions by assigning probabilities to each label based on the features of the data, naively assuming that each feature independently affects the probability.
- GaussianProcess: this predicts labels by using Gaussian probability distributions to represent the functions.

Table 4 shows the average accuracy, ROC score, precision, recall and F1 score calculated for the nine models. Metrics for Yallop's criterion are included for comparison. Each model was run 20 times and the average metrics calculated. The XGBoost model had the highest average accuracy, at 92.2 ± 0.8%. Random forests and AdaBoost also performed well, whereas non-decision tree based methods achieved lower accuracies. The neural network performed better than non-tree based methods, but worse than tree based methods.

Table 4: Average accuracies and metrics for the different models

Method	Accuracy	ROC	Precision	Recall	F1
XGBoost	**0.922 ± 0.008**	**0.975 ± 0.004**	**0.899 ± 0.015**	**0.926 ± 0.015**	**0.912 ± 0.009**
RandomForest	0.913 ± 0.009	0.969 ± 0.005	0.887 ± 0.015	0.915 ± 0.012	0.901 ± 0.010
DecisionTree	0.890 ± 0.009	0.888 ± 0.010	0.878 ± 0.022	0.866 ± 0.017	0.872 ± 0.011
NearestNeighbours	0.882 ± 0.012	0.944 ± 0.008	0.852 ± 0.018	0.884 ± 0.017	0.868 ± 0.013
NaiveBayes	0.848 ± 0.008	0.920 ± 0.005	0.860 ± 0.012	0.782 ± 0.014	0.819 ± 0.010
AdaBoost	0.901 ± 0.010	0.961 ± 0.006	0.882 ± 0.018	0.893 ± 0.019	0.887 ± 0.012
GaussianProcess	0.882 ± 0.009	0.888 ± 0.008	0.869 ± 0.018	0.860 ± 0.017	0.864 ± 0.012
ExtraTrees	0.909 ± 0.008	0.966 ± 0.006	0.886 ± 0.017	0.912 ± 0.015	0.899 ± 0.009
Neural Network	0.890 ± 0.014	0.890 ± 0.014	0.890 ± 0.021	0.865 ± 0.024	0.877 ± 0.017
Yallop	0.788 ± 0.000	0.787 ± 0.000	0.778 ± 0.000	0.831 ± 0.000	0.804 ± 0.000

The XGBoost model performs best out of all of the models, with the highest average accuracy, ROC score, precision, recall and F1 score. As a result, XGBoost was selected as the algorithm to use for predicting visibility. As with the neural network, the hyperparameters of XGBoost can be varied to tune the model to the data. XGBoost is very computationally efficient, and as a result large sweeps covering hundreds of different values of several hyperparameters could be measured in order to maximise accuracy. However, hyperparameter optimisation was not found to have a significant impact on accuracy, and the optimisation routine frequently failed to converge on a particular set of hyperparameters. XGBoost is not typically very sensitive to hyperparameters, as many of the hyperparameters only set the initial tree. The default hyperparameters, listed in Table 5, were found to result in the highest and most consistent accuracies.

Table 5: Key hyperparameters of XGBoost and the ranges varied during their optimisation.

Parameter	Description	Range Tested	Default
n_estimator	number of rounds of trees used in model	10-200	100
learning_rate	rate of reduction of step size	0.01-0.5	0.3
max_depth	number of splits the initial tree can have	1-10	6
subsample	proportion of dataset each iteration of a tree uses	0.1-1	1

8 Results: Testing the Predictive Models

The performance of a trained XGBoost model was evaluated with different variations of the datasets.

8.1 Feature Importance

Models assign different levels of importance to the features used to predict labels. These can provide an insight into how the model is being fitted, and also the physical importance of the parameters. The weighted importance of each input feature towards the output of a model can be represented by Shapely values [43], a concept originating from game theory. Figure 14 shows the mean Shapely values for each dataset feature, created using the `shap` library [44].

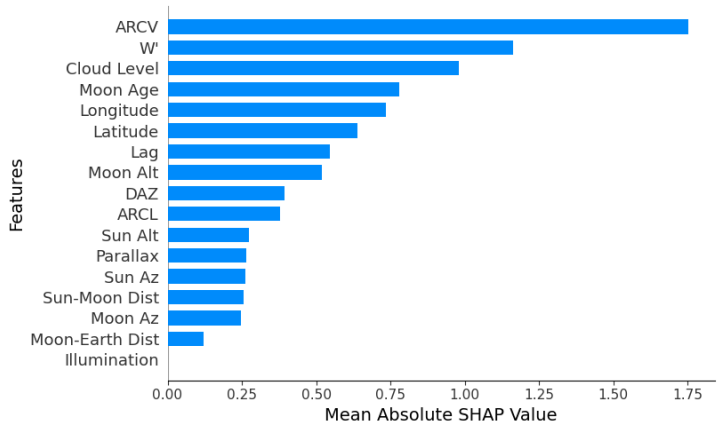
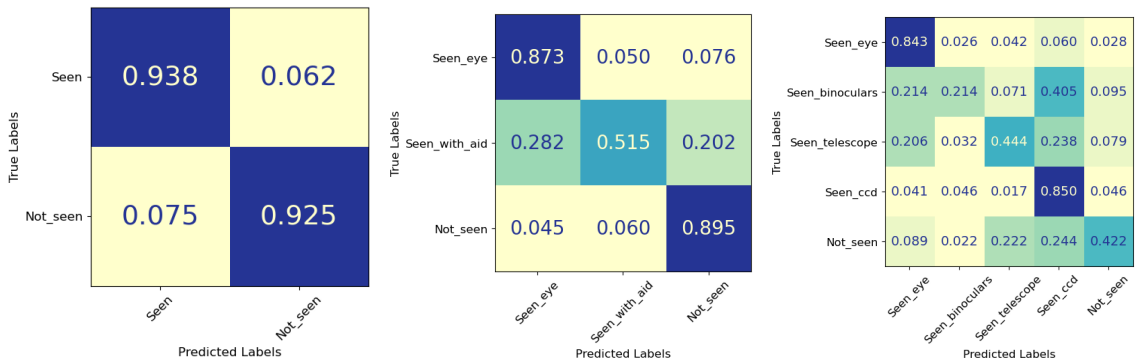


Figure 14: Importance of features of the dataset used to train an XGBoost model.

Features are ranked highest to lowest, for an XGBoost model trained on the binary seen or not seen dataset. ARCV and width are ranked as the most important features when predicting crescent visibility, as expected given their key role in Yallop's criterion. Cloud level is also very important, again expected as cloud cover will determine whether the new crescent is actually visible, even if it is theoretically likely. The age of the Moon is also an important feature as expected - at very young ages, the crescent will be less visible. Longitude as an important feature is reasonably unexpected, but it determines a number of other features. The importance of ARCL is also lower than expected given it is used in indicators such as the Danjon limit. Illumination, with ARCL the only parameter in its calculation, did not have any significant impact on the model.

8.2 Multi-Label Predictions

Initially, the model was trained and tested on a dataset only containing observations made by the naked eye. Each observation had either a seen or not seen label. This achieved a test accuracy of 92.2 ± 0.8%. A confusion matrix is shown in Figure 15a, showing low proportions of false positive and false negative predictions.



(a) Seen with naked eye. (b) Seen with eye or visual aid. (c) Exact method.

Figure 15: Confusion matrices showing the difference between predicted and actual labels for an XGBoost model for different method labels.

Multi-label classification is where the classifier model can predict one of a series of labels, not just a binary classification, and is natively supported by XGBoost. An XGBoost model was trained on the entire combined dataset, with observations labelled as not seen, seen with the naked eye or seen with visual aid (either binoculars, a telescope or a camera). Approximately 15% of the dataset was therefore a sighting with aid. A test accuracy of 83.1 ± 1.0% was found, significantly lower than predicting visibility only by the naked eye. The confusion matrix shown in Figure 15b indicates that the model generally performs well for naked eye visibility and negative sightings, but struggles to distinguish between eye, aid and not seen. This may result from the lack of a set definition of when a sighting is visible with aid, as this will vary by conditions and by observer.

Finally, the model was trained on the combined dataset using the exact method of observation: either naked eye, binoculars, telescope or CCD camera. This needed to take into account that a sighting with the naked eye also implies visibility by telescope, and so on. To do this, multi-output classification was used, where a machine learning model can predict any number and combination of labels. This was also supported by XGBoost, where a binary array can be passed as a label instead of a single string. The test accuracy was found to be 77.3 ± 0.7%, with the confusion matrix shown in Figure 15c. This was not particularly successful, with the model failing to distinguish between the different methods.

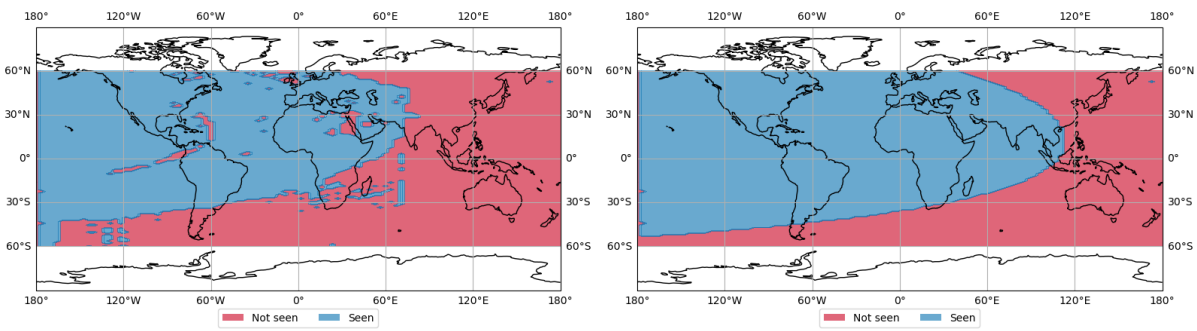
8.3 Cloud Cover and Weather

Weather, especially cloud levels, can significantly affect the visibility of the new crescent Moon. With smaller sets of data, it is sensible to exclude cloudy sightings, as the model may not be able to separate an astronomically possible observation that was not seen due to cloud, and an astronomically impossible observation. With a larger dataset of 5000 observations, cloudy data was included to investigate how this would affect the accuracy of the model. All observations contained some basic information about the cloud cover, typically a category such as 'clear', 'partly cloudy' or 'totally cloudy'. These were converted into a numerical scale, with 0 as clear, 0.5 as partly cloudy and 1 as total cloud cover. Just under half (48%) of observations had clear skies, with 42% partially cloudy and 10% totally cloudy. Generally, high accuracies (92.0 ± 1.7%) can be achieved with cloud level included as a feature. If observations with total cloud cover are excluded (reducing the dataset to 3991 observations), test accuracy decreases to 91.1 ± 0.9%. If only observations with clear skies are included (reducing the dataset to 2081 observations), accuracy increases slightly to 92.8 ± 1.1%.

The datasets only included observer-reported cloud cover, with wide-ranging labels such as 'partial'. In meteorology, cloud cover is measured in oktas, which correspond to the number of eighths of the sky covered by clouds [45]. One okta is a cloudless sky, whereas 8 is total cloud cover. Ideally, all observations would be updated with the cloud cover at time of observation in oktas. However, historical cloud cover data was difficult to obtain. Limited cloud cover from 2019 to 2023 was available from the WeatherOnline service [46], measured at a number of weather stations worldwide. Out of the 5302 observations, 2691 were from 2019 or later. Out of these, only 810 were in an area where WeatherOnline had cloud level data. When the `meteostat` library [47] was used to find the nearest weather station listed on WeatherOnline, many observations did not have a nearby weather station. Only 551 observations had a weather station that measured cloud cover within 100km. An accuracy of 90.6 ± 2.1% was found, lower than observer-reported cloud cover, but the much smaller dataset is likely a factor.

8.4 Visibility Predictions

With high accuracies verified against test data, the models can be used to predict visibility for specific dates and create visibility maps similar to other criteria. The grid of features for 22/03/2023 used in Section 6.3 was given as an input to the binary XGBoost model. Yallop's criterion values, date, longitude and latitude were removed, and cloud levels set to 0, representing clear skies as no weather data was available. Figures 16a and 16b show the visibility maps by the XGBoost model and Yallop's criterion respectively. Yallop's criterion codes A to C were classified as visible, and D to F were not visible. There is an 84% similarity between the two graphs, and the XGBoost model predicts the Moon will not be visible at higher longitudes such as most of India, whereas the visibility range for Yallop's criterion extends more easterly.

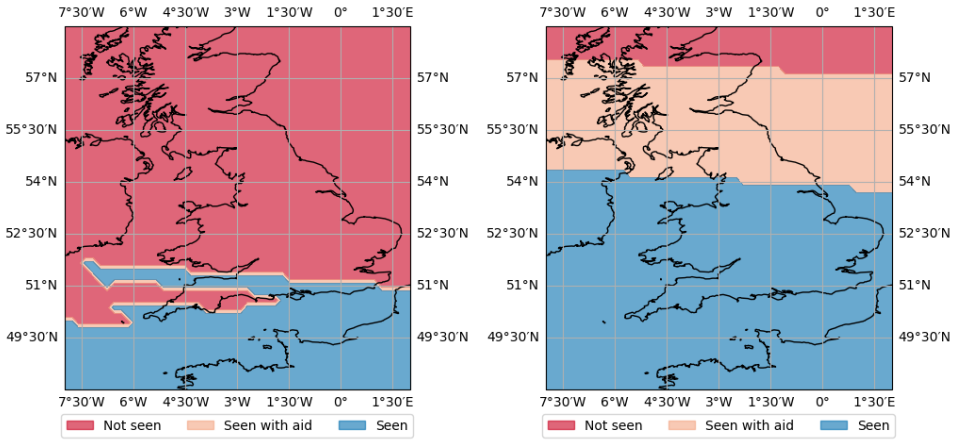


(a) XGBoost predictions

(b) Yallop's criterion predictions

Figure 16: Predicted visibility maps for 22/03/2023.

To test regional accuracy, the model was used to predict the visibility of the new crescent Moon in December 2023 for the UK. The new Moon was on the 12th of December, and so visibility maps were generated for the 13th, 14th and 15th. For the UK on the 14th, Yallop's criterion predicted that the new crescent would be visible with the naked eye in the southern half of the UK, and then visible with aid in the northern half, with no visibility in northern Scotland. However, the XGBoost model predicted no visibility across the majority of the UK. Both models predicted the new crescent would be easily visible on the 15th. According to the ICOUK [48], there were 8 negative sightings and one positive sighting on the 14th. However, the weather was generally cloudy and so it is difficult to evaluate accuracy for specific locations from one month's prediction.



(a) XGBoost predictions (b) Yallop's criterion predictions

Figure 17: Predicted UK visibility maps for 14/12/2023.

9 Conclusion

This project successfully collated a dataset of new crescent Moon sightings and used these to train several machine learning models to predict the first visibility of the new crescent with high accuracy.

A dataset of 5302 Moon sightings was created, the largest used for training machine learning models to predict new crescent Moon visibility. This dataset featured a range of observations from the 1850s to the present day. Positive and negative observations from a variety of locations by several methods were included. Twenty features of the Moon were calculated for each observation, including the angles and distances between the Sun and the Moon.

Several machine learning models were trained on this dataset. Ensemble decision-tree based models were found to be particularly effective, achieving high accuracies and supporting the findings of Allawi [30]. The XGBoost implementation of gradient boosted decision trees had the highest accuracy of 92.2 $\pm$ 0.8%, putting it just ahead of a random forests model and an artificial neural network. When the XGBoost model was trained only on observations during clear weather, this accuracy increased to 92.8 $\pm$ 1.1%. Almost all of the models achieved a higher accuracy than Yallop's criterion at 78.8%, one of the most common mathematical predictions.

The XGBoost model could be used to predict either positive or negative visibility by the naked eye at a high degree of accuracy. This could also be adapted to predict visibility with visual aid, such as a telescope, with a decrease in accuracy of approximately 10%. Attempting to predict the exact method of sighting was less successful, likely due to imbalanced data and the overlap between similar methods such as telescopes and binoculars. The trained model was used to create predictive visibility maps, which were similar to those generated using Yallop's criterion. Overall, the model is one of the most accurate to date for predicting the visibility of the new crescent Moon, and the first machine learning model to predict methods of observation.

9.1 Future Work

More observational data would be useful to further improve the accuracy of models. In particular, there were few observational records from South America and southern Africa. Towards the end of this project, a new dataset was made available by Faid et al. [49], containing 8100 observational records. Many are

from sources already used in this project, but there are still approximately 3000 additional observations that could be included in future projects. The number of observations available from the ICOP and ICOUK will also continue to increase as more observations are added each month.

In order to successfully account for the effect that weather has on visibility, a larger dataset of historical and future predicted cloud cover is needed. The UK Climate Projection 2018 (UKCP18) dataset [50] includes historical records and predictions of cloud cover from 1900 to 2100 at 60km resolution. This was not used due to the prohibitively large download size of the dataset and lack of tools to easily parse the data, but may be a useful resource for future studies. Some sighting records also included information about atmospheric conditions at the time of sighting, and this could be investigated. Additionally, the models used (and most visibility criteria) are unable to create predictions for very high and low latitudes, as concepts of daily sunset and moonset break down. More specific research could be done to create models for these latitude extremes, although finding sighting data may be an issue.

General Notes

Data used for this project is available at https://github.com/EmmaAlexander/mphys-moon, along with the Python code and Jupyter notebooks used. The `pandas` [51] and `numpy` [52] libraries were used for data manipulation. Graphs were produced using a colourblind-friendly colour palette developed by Okabe and Ito [53].

References

[1] V Gaffney et al. "Time and a Place: A luni-solar 'time-reckoner' from 8th millennium BC Scotland." In: *Internet Archaeology* 34 (2013).

[2] International Organization for Standardization. ISO 6709:2022 - Standard representation of geographic point location by coordinates. 2022. URL: https://www.iso.org/standard/75147.html.

[3] H Bradt. *Astronomy Methods: A Physical Approach to Astronomical Observations*. Cambridge University Press, 2004. ISBN: 9780521535519.

[4] United States Naval Observatory. Rise, Set, and Twilight Definitions. URL: https://aa.usno.navy.mil/faq/RST_defs.

[5] B D Yallop. "A Method for Predicting the First Sighting of the New Crescent Moon". In: *NAO Technical Notes* (1997).

[6] Frans Bruin. "The first visibility of the lunar crescent". In: *Vistas in Astronomy* 21 (Jan. 1977), pp. 331–358. ISSN: 00836656. DOI: 10.1016/0083-6656(77)90021-6.

[7] J K Fotheringham. "On the Smallest Visible Phase of the Moon". In: *Monthly Notices of the Royal Astronomical Society* 70 (7 1910), pp. 527–532.

[8] Nautical Almanac Office and United States Naval Observatory. *Explanatory Supplement to the Astronomical Almanac*. University Science Books, 2006, p. 125. ISBN: 9781891389450.

[9] Cesare Barbieri and Ivano Bertini. *Fundamentals of Astronomy*. 2nd. CRC Press, Jan. 2020, pp. 137–138. ISBN: 9780429287305. DOI: 10.1201/9780429287305.

[10] M Ilyas. "Lunar Crescent Visibility Criterion and Islamic Calendar". In: *Quarterly Journal of the Royal Astronomical Society* 35 (1994), pp. 425–461.

[11] E W Maunder. "On the Smallest Visible Phase of the Moon". In: *Journal of The British Astronomy Association* 8 (21 1911), pp. 355–362.

[12] A Danjon. "Jeunes et Vieilles Lunes". In: *L'Astronomie* 46 (1932), pp. 57–66.

[13] Mohammad Sh Odeh. "New criterion for lunar crescent visibility". In: *Experimental Astronomy* 18 (1-3 Jan. 2004), pp. 39–64. ISSN: 09226435. DOI: 10.1007/S10686-005-9002-5/METRICS.

[14] Bradley E Schaefer. "Visibility of the lunar crescent". In: *Jl R. astr. Soc* 29 (1988), pp. 511–523.

[15] Bradley E Schaefer. "Lunar crescent visibility". In: *Quarterly Journal of the Royal Astronomical Society* 37 (4 Jan. 1996).

[16] International Astronomical Center. Islamic Crescents Observation Project (ICOP). 1998. URL: https://www.astronomycenter.net/icop.html.

[17] José Víctor Rodríguez, Ignacio Rodríguez-Rodríguez, and Wai Lok Woo. "On the application of machine learning in astronomy and astrophysics: A text-mining-based scientometric analysis". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 12 (5 Sept. 2022), e1476. ISSN: 1942-4795. DOI: 10.1002/WIDM.1476.

[18] Murad Al-Rajab, Samia Loucif, and Yazan Al Risheh. "Predicting new crescent moon visibility applying machine learning algorithms". In: *Nature* (2023). DOI: 10.1038/s41598-023-32807-x.

[19] Tin Kam Ho. "Random decision forests". In: vol. 1. IEEE Comput. Soc. Press, 1995, pp. 278–282. ISBN: 0-8186-7128-9. DOI: 10.1109/ICDAR.1995.598994.

[20] Leo Breiman. "Random Forests". In: *Machine Learning* 45 (1 2001), pp. 5–32. ISSN: 08856125. DOI: 10.1023/A:1010933404324.

[21] Suryakanthi Tangirala. "Evaluating the Impact of GINI Index and Information Gain on Classification using Decision Tree Classifier Algorithm". In: *IJACSA) International Journal of Advanced Computer Science and Applications* 11 (2 2020).

[22] Leo Breiman. "Bagging Predictors". In: *Machine Learning* 24 (1996), pp. 123–140.

[23] T Hastie, R Tibshirani, and J H Friedman. *The Elements of Statistical Learning*. 12th. Springer, 2017, pp. 337–367.

[24] Tianqi Chen and Carlos Guestrin. "XGBoost". In: ACM, Jan. 2016, pp. 785–794. ISBN: 9781450342322. DOI: 10.1145/2939672.2939785.

[25] David E Rumelhart, Bernard Widrow, and Michael A Lehr. "The basic ideas in neural networks". In: *Communications of the ACM* 37 (3 Jan. 1994), pp. 87–92. ISSN: 0001-0782. DOI: 10.1145/175247.175256.

[26] Andrea Apicella et al. "A survey on modern trainable activation functions". In: *Neural Networks* 138 (Jan. 2021), pp. 14–32. ISSN: 0893-6080. DOI: 10.1016/J.NEUNET.2021.01.026.

[27] Shun-ichi Amari. "Backpropagation and stochastic gradient descent method". In: *Neurocomputing* 5 (4-5 Jan. 1993), pp. 185–196. ISSN: 09252312. DOI: 10.1016/0925-2312(93)90006-O.

[28] Diederik P Kingma and Jimmy Lei Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (Jan. 2014).

[29] Tafseer Ahmed. "Predicting the Visibility of the First Crescent". In: *KIET Journal of Computing and Information Sciences* 3 (2 Jan. 2018), p. 10. ISSN: 2710-5075. DOI: 10.51153/kjcis.v3i2.52.

[30] Ziyad T Allawi. "A Pattern-Recognizer Artificial Neural Network for the Prediction of New Crescent Visibility in Iraq". In: *Computation* 10 (10 Jan. 2022), p. 186. ISSN: 20793197. DOI: 10.3390/COMPUTATION10100186/S1.

[31] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (85 2011), pp. 2825–2830. ISSN: 1533-7928.

[32] Pilar Lopez Gonzalez-Nieto et al. "Design and Development of a Virtual Laboratory in Python for the Teaching of Data Analysis and Mathematics in Geology: Geopy". In: vol. 1. IATED, Jan. 2020, pp. 2236–2242. ISBN: 978-84-09-17939-8. DOI: 10.21125/inted.2020.0687.

[33] Islamic Crescents Observation for the UK (ICOUK). Islamic Crescents Observation for the UK (ICOUK). 2008. URL: https://www.moonsighting.org.uk/.

[34] Thamer Alrefay et al. "Analysis of Observations of Earliest Visibility of the Lunar Crescent". In: *The Observatory* 138 (Jan. 2018), p. 267.

[35] Adrian M Price-Whelan et al. "The Astropy Project: Building an open-science project and status of the v2. 0 core package". In: *The Astronomical Journal* 156 (3 2018), p. 123.

[36] Jannik Michelfeit. timezonefinder: A python package for looking up the corresponding timezone for given coordinates on earth entirely offline. URL: https://github.com/jannikmi/timezonefinder.

[37] Brandon Rhodes. Skyfield: High precision research-grade positions for planets and Earth satellites generator. Jan. 2019.

[38] W Folkner et al. "The Planetary and Lunar Ephemerides DE430 and DE431". In: *Interplanet. Netw. Prog. Rep* 196 (Jan. 2014).

[39] C Sullivan and Alexander Kaszynski. "PyVista: 3D plotting and mesh analysis through a streamlined interface for the Visualization Toolkit (VTK)". In: *Journal of Open Source Software* 4 (37 Jan. 2019), p. 1450. ISSN: 2475-9066. DOI: 10.21105/joss.01450.

[40] Guy S Handelman et al. "Peering Into the Black Box of Artificial Intelligence: Evaluation Metrics of Machine Learning Methods". In: *American Journal of Roentgenology* 212 (1 Jan. 2019), pp. 38–43. ISSN: 0361-803X. DOI: 10.2214/AJR.18.20224.

[41] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: Curran Associates, Inc., 2019, pp. 8024–8035.

[42] Lukas Biewald. Experiment Tracking with Weights and Biases. Software available from wandb.com. 2020. URL: https://www.wandb.com/.

[43] L S Shapley. 17. A Value for n-Person Games. Ed. by Harold W Kuhn and Albert W Tucker. Jan. 1953. DOI: 10.1515/9781400881970-018.

[44] Scott M Lundberg, Paul G Allen, and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 4765–4774.

[45] Meteorological Office. *Observer's Handbook*. 4th ed. UK Meteorological Office, 2000, pp. 29–30. ISBN: 0861803531.

[46] WeatherOnline Ltd. WeatherOnline Archive Tables. URL: https://www.weatheronline.co.uk/weather/maps/forecastmaps?LANG=en&CONT=ukuk&R=151.

[47] Christian Sebastian Lamprecht. meteostat: Access and analyze historical weather and climate data with Python. URL: https://github.com/meteostat/meteostat-python.

[48] Islamic Crescents Observation for the UK (ICOUK). December 2023 Observation Report. 2023. URL: https://www.moonsighting.org.uk/moon/sighting-reports/report-1445-06.html.

[49] M S Faid, M S A Mohd Nawawi, and M H Mohd Saadon. "Analysis tool for lunar crescent visibility criterion based on integrated lunar crescent database". In: *Astronomy and Computing* 45 (Jan. 2023). ISSN: 22131337. DOI: 10.1016/j.ascom.2023.100752.

[50] Met Office Hadley Centre. UKCP18 Global Projections at 60km Resolution for 1900-2100. 2018. URL: https://catalogue.ceda.ac.uk/uuid/97bc0c622a24489aa105f5b8a8efa3f0.

[51] Wes McKinney. "Data structures for statistical computing in python". In: *Proceedings of the 9th Python in Science Conference* 445 (2010), pp. 51–56.

[52] Charles R Harris et al. "Array programming with NumPy". In: *Nature* 585 (7825 Jan. 2020), pp. 357–362. ISSN: 14764687. DOI: 10.1038/S41586-020-2649-2.

[53] Masataka Okabe and Kei Ito. Color Universal Design (CUD) - How to make figures and presentations that are friendly to Colorblind people. 2008. URL: https://jfly.uni-koeln.de/color/.

Appendix

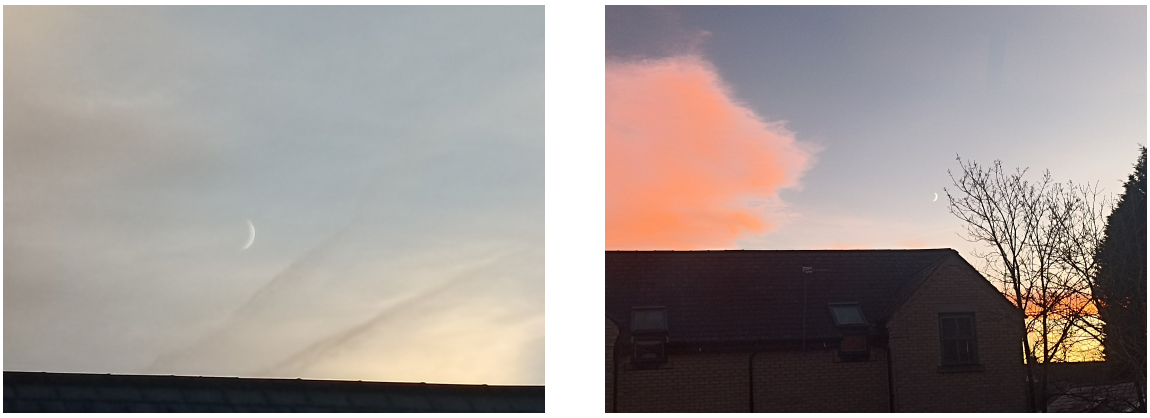


Figure A1. Photographs of the new crescent Moon, taken in Manchester on 16/12/2023. This was four days after the new Moon, and two days after first visibility in the UK [48], so the crescent is clearly visible even in the bright winter sky.