**MSc Data Science(ABC)**

**Title**

**Bird Species Classification in the United Kingdom (ABC)**

**Using Machine Learning Methods**

**Student \*\*\*(ABC)**

**A Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Data Science at the University of Salford (Dummy Filled by User)**

**Title**

**Bird Species Classification in the United Kingdom (ABC)**

**Using Machine Learning Methods**

**By \*\*\***

**@\*\*\***

**Supervisor: \*\***

**School of Science, Engineering and Environment University of**

**ABC, ABC, United Kingdom**

**Academic Year: \*\***

**Abstract**

Many studies have been published for the automated identification of certain bird species. This research area continues to be studied. In this work, we investigate the identification of many bird species under a set of circumstances using various machine learning algorithms and the distinctive features of the Xeno-Canto bird dataset collection are discussed. We study in depth if and how classification results are affected by the number of species and the different features selection in the subsets supplied to the classifier. More specifically, a data sampling strategy is used to select six months' worth of data from distinct species for repeated classification runs. The collection of 659 bird species extracted from the Xeno-Canto database contains the sounds of the species, with the number of species contained in each subset varying between 10 and 300. We have selected the top recorded from 55 species from the dataset and applied Logistic Regression, K-Nearest Neighbors, Random Forest, Support Vector Machine, and Decision Tree to classify different bird species. Precision, recall, and accuracy are some of the typical metrics used to analyze classification performance. All of these metrics point to an improvement in classification performance however, we also observe a decrease in classification results as the number of species rises owing to an issue with the sparse dataset. We carefully examine this reliance and contrast the calculated findings with an analytical explanation assuming dependencies for a hypothetically idealized flawless classifier. Additionally, we note that the classification performance varied among 20 randomly chosen subgroups depending on the species composition.

Table of Content

[2 List of Figures 5](#_Toc113916270)

[2.1 List of Tables 7](#_Toc113916271)

[1. Chapter 1: Introduction 8](#_Toc113916272)

[2.2 General Overview 8](#_Toc113916273)

[2.3 Aims and Objectives 8](#_Toc113916274)

[2.4 Scope of Research Work 8](#_Toc113916275)

[2.5 Tools for Data Analysis 9](#_Toc113916276)

[3 Chapter 2: Literature Review 10](#_Toc113916277)

[3.1 Introduction 10](#_Toc113916278)

[3.2 Data Mining 10](#_Toc113916279)

[3.3 Machine Learning 12](#_Toc113916280)

[3.3.1 Supervised 12](#_Toc113916281)

[3.3.2 Unsupervised 15](#_Toc113916282)

[3.3.3 Reinforcement learning 15](#_Toc113916283)

[4 Chapter 3: Data Description, EDA, and Data Preprocessing 16](#_Toc113916284)

[4.1 Data Description 16](#_Toc113916285)

[4.2 Exploratory data analysis and cleaning 17](#_Toc113916286)

[5 Chapter 4: Analysis Approach and Implementation 27](#_Toc113916287)

[5.1 Methodology 27](#_Toc113916288)

[5.2 Data Preprocessing 27](#_Toc113916289)

[5.3 Data slicing between dependent and independent variables 35](#_Toc113916290)

[5.4 Splitting Data into Train and Test 37](#_Toc113916291)

[5.5 Support Vector Machine 37](#_Toc113916292)

[5.6 Random Forest 41](#_Toc113916293)

[5.6.1 Classification in random forest. 43](#_Toc113916294)

[5.7 Decision Tree 44](#_Toc113916295)

[5.7.1 Graphical representation of Decision Tree. 44](#_Toc113916296)

[5.7.2 Strength 46](#_Toc113916297)

[5.7.3 Weaknesses. 47](#_Toc113916298)

[5.8 K-Nearest Neighbors (KNN) 47](#_Toc113916299)

[5.8.1 Modeling and classification model 48](#_Toc113916300)

[5.8.2 Model Construction and Classification Process 49](#_Toc113916301)

[5.9 Logistic Regression 51](#_Toc113916302)

[5.9.1 Logistic regression step-by-step 52](#_Toc113916303)

[5.9.2 Odds and probabilities 52](#_Toc113916304)

[5.9.3 Continuous explanatory variables or variables with more than two levels 53](#_Toc113916305)

[5.9.4 Variables inclusion and selection 54](#_Toc113916306)

[5.9.5 Experiments and Results 55](#_Toc113916307)

[6 Chapter 6 Conclusion 64](#_Toc113916308)

[6.1 Results for Xeno-Canto database 64](#_Toc113916309)

[6.2 Conclusion 64](#_Toc113916310)

[7 References 66](#_Toc113916311)

# List of Figures

[Figure 1. Types of Machine Learning 12](#_Toc113916312)

[Figure 2. Different Supervise Learning Models 13](#_Toc113916313)

[Figure 3. Most Frequent Species in North and South of England 18](#_Toc113916314)

[Figure 4. Top 50, 50 Species that sing more frequently in the Morning and Evening 19](#_Toc113916315)

[Figure 5. Distribution of Map of England and mapping the location of the spices 20](#_Toc113916316)

[Figure 6. Top 15 Country having Most Bird Species Recording 21](#_Toc113916317)

[Figure 7. Most Occurred Bird’s Species Percentage 21](#_Toc113916318)

[Figure 8. Most Frequent Recording in Different Seasons of the Year 22](#_Toc113916319)

[Figure 9. Frequent Recording with Respect To Years 22](#_Toc113916320)

[Figure 10. Frequent Recording With Respect To Months 23](#_Toc113916321)

[Figure 11. Species having only 1 Subspecies and Different Subspecies 24](#_Toc113916322)

[Figure 12. Distribution of Bird Sub Species 25](#_Toc113916323)

[Figure 13.Top 10 Uploaders Which Are More Frequent Recording Uploaders 26](#_Toc113916324)

[Figure 14.Recorder That More Frequently Upload in the Evening and Morning 26](#_Toc113916325)

[Figure 15. Distribution of Bird Species on the basis of Time 28](#_Toc113916326)

[Figure 16. Species that sings Have More than 0:50 Second of Times 29](#_Toc113916327)

[Figure 17. Number Years Count for Species Occurrences 30](#_Toc113916328)

[Figure 18. Bird Seen Count W.R.T Countries 31](#_Toc113916329)

[Figure 19. Bar plots for all species in Dataset 32](#_Toc113916330)

[Figure 20. Species Distribution after Filtering the Data 33](#_Toc113916331)

[Figure 21. Color Map with Correlation for Numeric Features. 34](#_Toc113916332)

[Figure 22. Heatmap correlation metric for numerical features of different bird species 35](#_Toc113916333)

[Figure 23. Heatmap Correlation graph between converted and Numerical Features for the Bird Species 36](#_Toc113916334)

[Figure 24. Colormap Correlation graph between converted and Numerical Features for the Bird Species 37](#_Toc113916335)

[Figure 25. Different Hyperplanes 39](#_Toc113916336)

[Figure 26. Separating Features Boundaries on the basis of Linear and Non-Linear Data 41](#_Toc113916337)

[Figure 27. Random Forest Overview 42](#_Toc113916338)

[Figure 28. Workflow of Decision Tree 43](#_Toc113916339)

[Figure 29.Decision Tree Representation 44](#_Toc113916340)

[Figure 30.Entropy Function 45](#_Toc113916341)

[Figure 31. Binary Dataset Consisting Of Squares and Circles Is Scatted In 2-Dimensional Space 47](#_Toc113916342)

[Figure 32. 9 Data Tuples Consisting Of Circles are Included inside the Distinct Line 50](#_Toc113916343)

[Figure 33. Centroid Selection based on Euclidian Distance 50](#_Toc113916344)

[Figure 34. Updated and Improved Centroids 51](#_Toc113916345)

[Figure 35. Confusion Matrix on Logistic Regression 56](#_Toc113916346)

[Figure 36. Testing Accuracy on Different values of K 57](#_Toc113916347)

[Figure 37. Confusion Matrix on Logistic Regression 58](#_Toc113916348)

[Figure 38. Confusion Matrix on Random Forest 59](#_Toc113916349)

[Figure 39.Setting hyperparameters for Grid Search 60](#_Toc113916350)

[Figure 40. Confusion Matrix on SVM 61](#_Toc113916351)

[Figure 41. Confusion Matrix on Decision Tree 62](#_Toc113916352)

[Figure 42. Accuracy comparison of different models 63](#_Toc113916353)

## List of Tables

[Table 1. Description of Dataset 16](#_Toc113916354)

[Table 2. Missing values details. 17](#_Toc113916355)

[Table 3. Statistical Description of Numerical Variables 33](#_Toc113916356)

[Table 4. Correlation table b/w Numerical Features 34](#_Toc113916357)

[Table 5. Logistic Regression Model Values with Different Parameters. 52](#_Toc113916358)

[Table 6. Multivariate Logistic Regression containing on Continuous Variable. 54](#_Toc113916359)

# Chapter 1: Introduction

## General Overview

This chapter covers the synthesis results to achieve the expected output. In this study, our main focus in the field of healthcare by using deep learning techniques for the classification of different bird species classification. Machine learning models are used to classify bird species based on the metadata of the birds. The results have been analyzed using Machine Learning frameworks with different criteria which have been discussed in the methodology chapter. In the last section, sensitivity analysis has been done. This chapter will discuss a detailed overview of the result of the Xeno-Canto database.

## Aims and Objectives

This research work aims to investigate the identification and classification of bird species by incorporating metadata from the Xeno-Canto database by using machine learning algorithms. Improve and address the classification results by handling class imbalance problems in different species class classification problems.

The objectives involved in this study are:

* To design models for the identification of different species by adding non-numerical features collected from the United Kingdom.
* To test different data analysis methods and machine learning techniques for the identification of different bird species.
* To perform different data analyses and perform experiments using machine learning.
* To develop a sound model that can compare the best prediction for bird species classification.
* To explore different factors affecting birds which are in the North and South of England and how the species are more frequent in different circumstances including the area, the sound of the species, and the recorder of the bird voice.

## Scope of Research Work

The analysis of bird species and their many species has gained significant significance in recent years because of advances in machine learning and data science that have propelled research on metadata. The battle to keep the connections between many study fields to their new topic and domain is a significant difficulty. When creating new tactics and treatments to assess data for various bird species, knowledge of bird species is crucial. We will identify the species of birds that were obtained from various sources in the United Kingdom in this thesis. We conduct data analysis and employ a variety of machine learning techniques, such as Logistic Regression, K-Nearest Neighbors, Random Forest, Support Vector Machine, and Decision Tree, to categorize the various bird species that are involved in colonial migration and the movement of bird species in the UK region. We can better comprehend the many bird species by recording their sounds and seeing how they move across various environments. Build a machine learning model that can recognize and categorize certain species from input data. With the help of this analysis, we may reason analytically about the various bird species, which will be covered in the technique section. Using metadata, the model may be used to categorize various bird species.

## Tools for Data Analysis

We have used multiple tools and libraries for data analysis. The language used was python as it is open source and in this, a lot of open-source libraries help us to do data cleaning, analysis, and plotting graphs having the use of full information. We used the google colaboratory as well to implement the different machine learning models as it provides free GPU service to train the model. The libraries that we have used in this for as follows:

* Pandas: it is used for the data reading and filtering of the data using different queries.
* Sklearn: to implement the machine learning model and to use the measuring parameter to evaluate the models
* Matplotlib: The plotting of the graphs has been done with the help of Matplotlib
* Seaborn: It is also used for plotting visual information

# Chapter 2: Literature Review

## Introduction

Many bird species use vocal sound patterns that can be compared to words to communicate. The single most important requirement for this group was discovered (Anon n.d.-a) during a recent session of scientists studying the biology of syntax: an appropriate, shared database. For general recordings of bird and animal sounds, there are several great online databases, such as Xeno-canto (Anon n.d.-b). These databases give users access to recordings and pertinent metadata, but they do not contain annotations about the kinds or orders of the voice units that make up the recordings. It is usually considerably more time-consuming to annotate phrase types than it is to listen to the recordings, hence a shared, freely accessible resource for birds’ metadata syntax is highly required for different analyses The dataset comprises an Indian trial of 26 features that can be analyzed by data mining techniques that are mentioned below.

## Data Mining

The science of data mining is the extraction of knowledge or information from data. It is a task that is frequently carried out on laptops, desktop computers, and cloud computational resources. Big data analytics has a focus on stream mining, which applies mining operations to input streamed data. It has many contemporary uses, especially in light of the development of the Internet of Things (IoT), which produces real-time streamed data from various sources. Because the streamed input data enters continuously and in large volumes, it is not possible to store it locally. As a result, single-pass processing is needed to analyze the data. The main difficulties in mining streamed data are the selection and description of data.

All records in the dataset represent related objects, with each record representing a single entity or object. In this database analogy, the columns list various characteristics or "attributes" of each object. For instance, if the item is a smartphone, attributes might include the device's screen size, internal storage, and operating system. The category or "class" attribute, which selects the category to which each record belongs, is assigned to one of these columns. Using our smartphone as an example once more, the class value or "label" may refer to the phone's target market, such as "entry-level" or "power-user." This kind of dataset is frequently used with classification, a type of data mining. There are two primary functions of classification. It identifies or "learns" first. Of the data mining techniques developed recently, several major kinds of data mining methods, including generalization, characterization, classification, clustering, association, evolution, pattern matching, data visualization, and meta-rule guided mining, are herein reviewed. The techniques for mining knowledge from various kinds of databases, including relational, transactional, object-oriented, spatial, and active databases, as well as global information systems, are also examined.

Data mining is a technique that incorporates machine learning, statistics, and database management systems to find and retrieve patterns from large data sets. Data mining is a discipline of computer science and statistics that aims to retrieve information from a data collection using clever algorithms and organize it into an understandable structure for future use. Data mining, in other words, is the application of efficient techniques for the analysis of extremely large volumes of data and the extraction of pertinent and potentially unexpected patterns in data.

The study of self-learning, data-driven computer algorithms is known as machine learning. It is an artificial intelligence technique that builds a framework from training information to make assumptions or decisions without being programmed separately to do so. In a variety of fields were creating conventional algorithms to perform the necessary tasks is challenging or impossible, such as medicine, email filtering, voice recognition, and computer vision, machine learning algorithms are used.

It is challenging to develop a new concept of machine learning because it is an interdisciplinary field with origins in many different disciplines, including statistics, mathematics, data analysis, and information analytics. However, both machine learning and statistical learning are different from one another. Machine learning statistics that concentrate on computer-aided estimation is a machine learning subset that is closely related. Since it offers tools, hypotheses, and implementation domains, the study of optimization techniques is beneficial to the field of machine learning. In some applications of machine learning, neural nets and data are combined to simulate how a genetic brain works. When used to address business problems, machine learning is also referred to as predictive analytics.

In Machine Learning, according to (Panch, Szolovits, and Atun 2018) computer programmers learn affiliations of predictions from trained data. Most simply, machine learning is the use of machines to apply predictive methods to data. Compared to what is traditionally utilized in medicine, it employs a wider range of statistical methods. Modern methods can hold more complex data because they depend on networks with fewer predictions about the original data lake Deep Learning.

Machine learning offers strategies, tactics, and resources that can assist in resolving prognostic and diagnosis issues across numerous medical specialties. ML is being used to analyze the significance of prognostic factors and their variations for diagnosis, such as the diagnosis of disease advancement, the retrieval of clinical knowledge for clinical assessment, the planning, and support of therapy, as well as the whole management of patients (Magoulas and Prentza 2001).

## Machine Learning

Machine learning is defined as giving machines the capability to learn and take decisions accordingly. This is done by giving them a massive amount of data which could be in any form be it words, numbers, images, videos, etc. First, the data is preprocessed into a form that can be fed to a machine, and then with the help of an algorithm, it finds a statistical pattern in that massive amount of data and learn through these patterns and act accordingly.

Whereas on the one hand where machine learning is just the learning of a PC program or an algorithm to logically improvise the set of instructions that are given to it. On the other hand, it could be seen as the perspective of hypothetical and numerical demonstrating of how this cycle functions. It is divided into the following categories as shown in Fig. 1 which sums up the numerous approaches to the outline.

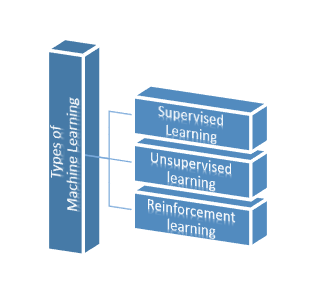


Figure 1. Types of Machine Learning

### Supervised

Supervised learning is the fundamental type of ML, in which we give input and output variables to the machine and use an algorithm to learn it. At the point when the machine is supervised while it is "learning", the preparation type is called supervised learning. Supervised learning has two variables, the input variables are known as input features whereas the output variables are known as the labeled data. In supervised learning, we train our machine with labeled data in other words some correct outputs are fed to it while it is being trained and then it is tested with some new set of instructions. It then analyzes the trained data and produces results from the labeled data.

Supervised learning is further divided into two different types depending upon their applications and merits: regression and classification, and each has its own arrangement of utilization cases and merits which include: Linear and Polynomial regression; Nave Bayes, Trees, etc. as shown in Fig. 2.

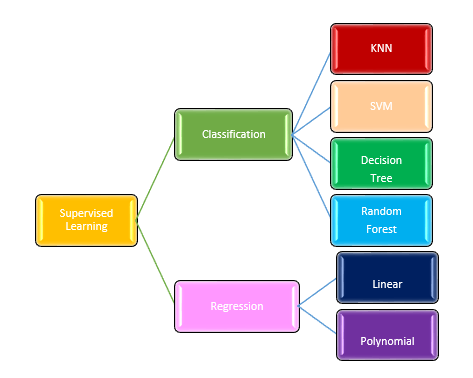


Figure 2. Different Supervise Learning Models

#### Classification

Giving an output variable a class or a category is known as classification. The most common and widely recognized situation is whether the email is spam or not. Another example of classification could be disease, whether the patient has a heart attack or not. The algorithm classifies the given set of information based on how these two classes relate to one another. When dealing with two classes it is known as binary classification, for more than two classes it is considered multi-class classification.

Grouping issues can be unraveled with various measures of calculation. Whichever algorithm you decide to utilize relies upon the information and the circumstance. Given below are a few known algorithms:

* **Support Vector Machines:** In 1992, the Support Vector Machine (SVM) algorithm was developed based on the statistical hypothesis of Vapnik. The algorithm separates two classes from each other by creating a hyper-plane. It uses a function known as kernel which is the focal point and does a greater part of the learning tasks (Boser, Guyon, and Vapnik 1992).
* **Decision Trees:** A decision tree develops models like a tree structure. It breaks the set of information into nodes. The starting node of the tree is called the root node whereas the last is the goal of the tree. It can work on categorical as well as numerical data. It could have a number of branches where the leaf node shows the classification.
* **K-Nearest Neighbor:** It is the simple algorithm used for classification as well as regression. A set of data is given with a query. We find the distance from every single set of objects to the new input; here K is the odd number of results of the smallest distance from the data point to the query. The query is then classified into the larger number of least distant data sets
* **Naive Bayes Classifier:** This classification works based on Bayes’ Theorem. It is a relatively easy algorithm and distinctively very useful for large sets of data. It is not only simple but considered to be the most sophisticated method of classification. It accepts the occurrence of a specific element in a subgroup is random to the occurrence of some supplementary element or also that these properties have a free commitment to the likelihood.

#### Regression

The technique forecast continuous distinct results utilizing trained data.

* **Linear:** The algorithm assumes that there is a direct relation between the input and the output variable from the given information. The Input and the output variables are known as independent and dependent variables. At the point when concealed information is passed to the calculation, it utilizes the capacity, and figures and guides the contribution to a consistent incentive for the yield.
* **Polynomial Regression:** The algorithm predicts discrete qualities for the arrangement of Independent factors of the new information as its yield lies between the scope of 0 and 1 that have been passed to it. It does the forecast by planning the concealed information to the logit work that has been customized into it.

### Unsupervised

It is entirely different from supervised learning where a system is first trained on labeled output and tested to give the correct outcome. In unsupervised learning there are no input or output variables or no such training is given to the system the system itself acts on the given information without being guided and needs to learn all by itself. It is chiefly concerned with the methods that include a grouping of data according to their similarities and differences in them. Suppose a system is fed with images of animals let them be breeds of cats and dogs, now the system has no idea about the breeds and characteristics of cats and dogs, but it can separate them and make a group of images according to the similarities in them.

### Reinforcement learning

When compared with supervised and unsupervised learning, we can find reinforcement learning completely different from them. Because we can’t find any relationship between the three of them as we do in supervised and unsupervised learning; lack and abundance of labeled data, which is why the relationship to the third category of machine learning is not very clear. In reinforcement learning to accomplish an unmistakable goal, the algorithm has to learn through the trial-and-error method. Ideal results are reinforced, and non-ideal results are rebuffed. It is exactly like playing a game on a computer for the first time where you are not familiar with the keys and resulting actions, and you eventually learn when and which key needs to be pressed to do a certain action.

# Chapter 3: Data Description, EDA, and Data Preprocessing

## Data Description

From the start, we have the dataset in JSON format. To apply data analysis and machine learning algorithms we need data in excel or CSV form. Most of the libraries in Python language like pandas, Sklearn, Matplotlib, etc. used the dataset in this format. As a result, we converted the dataset into the commas-separated format CSV. There are 46565 rows × 26 columns of this dataset details of these columns are shown in Table 1.

Table 1. Description of Dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.** | **Column** | **Type** | **Count -Non-Null** | **Description** |
| 1 | id | Int64 | 46565 non-null | The catalog number of the recording on xeno-canto |
| 2 | gen | Object | 46565 non-null | The generic name of the species |
| 3 | sp | Object | 46565 non-null | The specific name (epithet) of the species |
| 4 | ssp | Object | 46565 non-null | The subspecies name (sub specific epithet) |
| 5 | en | Object | 46565 non-null | The English name of the species |
| 6 | rec | Object | 46565 non-null | The name of the recordist |
| 7 | cnt | Object | 46565 non-null | The country where the recording was made |
| 8 | loc | Object | 46567 non-null | The name of the locality |
| 9 | lat | Object | 46567 non-null | The latitude of the recording in decimal coordinates |
| 10 | lng | Object | 46565 non-null | The longitude of the recording in decimal coordinates |
| 11 | alt | Object | 46565 non-null | Alternative text |
| 12 | type | Object | 46565 non-null | The sound type of the recording (e.g. 'call', 'song', etc.). This is generally a comma-separated list of sound types. |
| 13 | url | Object | 46565 non-null | The URL specifying the details of this recording |
| 14 | file | Object | 46565 non-null | The URL to the audio file |
| 15 | file-name | Object | 46565 non-null | The original file name of the audio file |
| 16 | sono | Object | 46565 non-null | An object with the urls to the four versions of sonograms |
| 17 | lic | Object | 46565 non-null | The URL describing the license of this recording |
| 18 | q | Object | 46565 non-null | The current quality rating for the recording |
| 19 | length | Object | 46565 non-null | The length of the recording in minutes |
| 20 | time | Object | 46565 non-null | The time of day that the recording was made |
| 21 | date | Object | 46565 non-null | The date that the recording was made |
| 22 | uploaded | Object | 46565 non-null | The date that the recording was uploaded to xeno-canto |
| 23 | also | Object | 46565 non-null | An array with the identified background species in the recording |
| 24 | rmk | Object | 46565 non-null | additional remarks by the recordist |
| 25 | bird-seen | Object | 46565 non-null | Was the recorded bird visually identified? (yes/no) |
| 26 | playback-used | Object | 46565 non-null | Was playback used to lure the bird? (yes/no) |

## Exploratory data analysis and cleaning

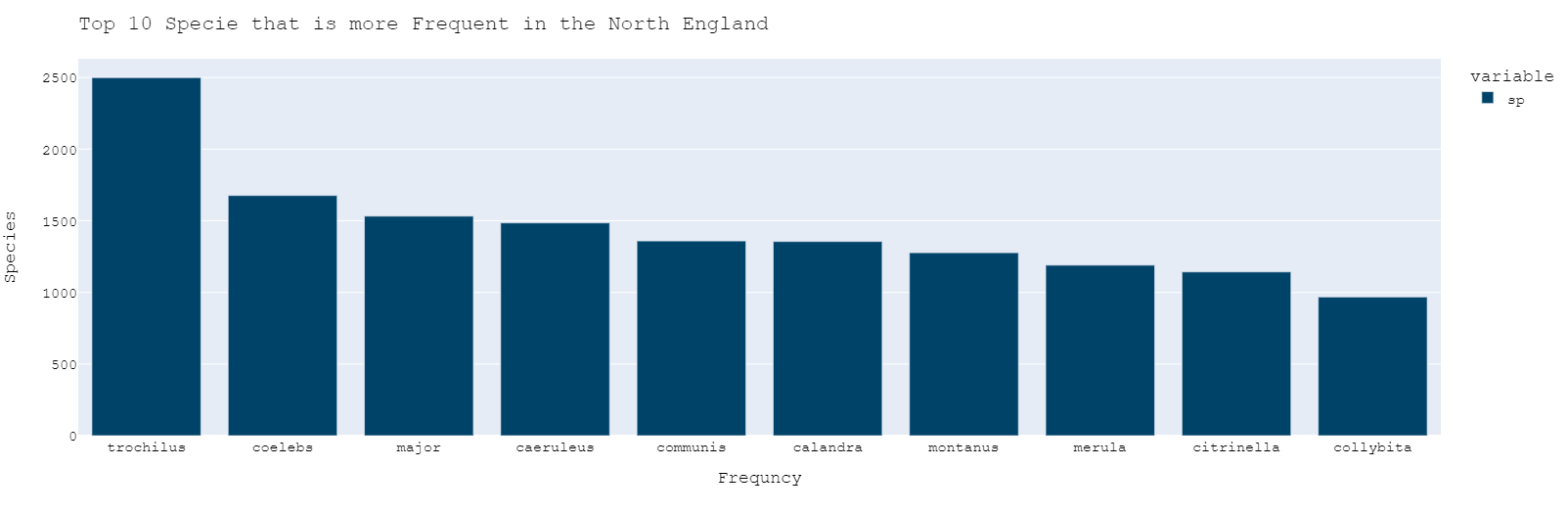
By exploring the dataset, analysis that the dataset contains also null and missing values. We have to deal with all the outliers and missing values that effect accuracy while learning meaning full information. In this data multiple types of garbage data like these "n.a.","?","NA","n/a", "na", "--","-","n.a", "am", '[]', "?:?", "['']", "unknown", "??:??". To apply techniques of missing datasets, we need to remove all kinds of garbage values and replace those with NaN values. Now the data cleaning process can be applied easily. There are multiple NaN values in this dataset as shown in Table 2.

Table 2. Missing values details.

|  |  |  |
| --- | --- | --- |
| Column | Total Null Values | Percentage |
| ssp | 33385 | 71.6955 |
| also | 18453 | 39.6285 |
| rmk | 6775 | 14.5496 |
| playback-used | 2442 | 5.24428 |
| bird-seen | 2015 | 4.32728 |

These missing values have been removed and fixed. Different techniques like the mean and mode of the columns are taken to fill those NaN cells. The date column contains a lot of information, it needs to separate to get more analysis. From the date column, we made three new columns day\_of\_month, month, and year. These new columns help us to do an analysis of data on the base of days, months, and years. All the columns of the dataset were in the object datatype except the id. So type casting of all columns as all data are in object type so we have to convert into their respective data type as per their nature. In the year columns, it contains the values that have from 000,0,00. Filters were applied to these garbage values to make them clean.

“Is there a specie that is more frequent in North England than the south of ENGLAND?” To answer this filters were applied. The data doesn’t contain any information about North and south England. The loc column contains detailed information about the location of the bird species. So we filter every location based on the North and south key word in the "loc" col. First, we find out the top species that are frequent in south England and then north England as shown in Fig. 3. Now we have frequent species in both north and south we again simply filter out “a specie that is more frequent in North England than the south of ENGLAND” by using pandas data frame quires. Trochilus was found 2499 times in the North of England and only two times in the South of England. So, we are concluding our findings that the most Frequent Bird Species in North England are "Trochilus" and "Mystery".



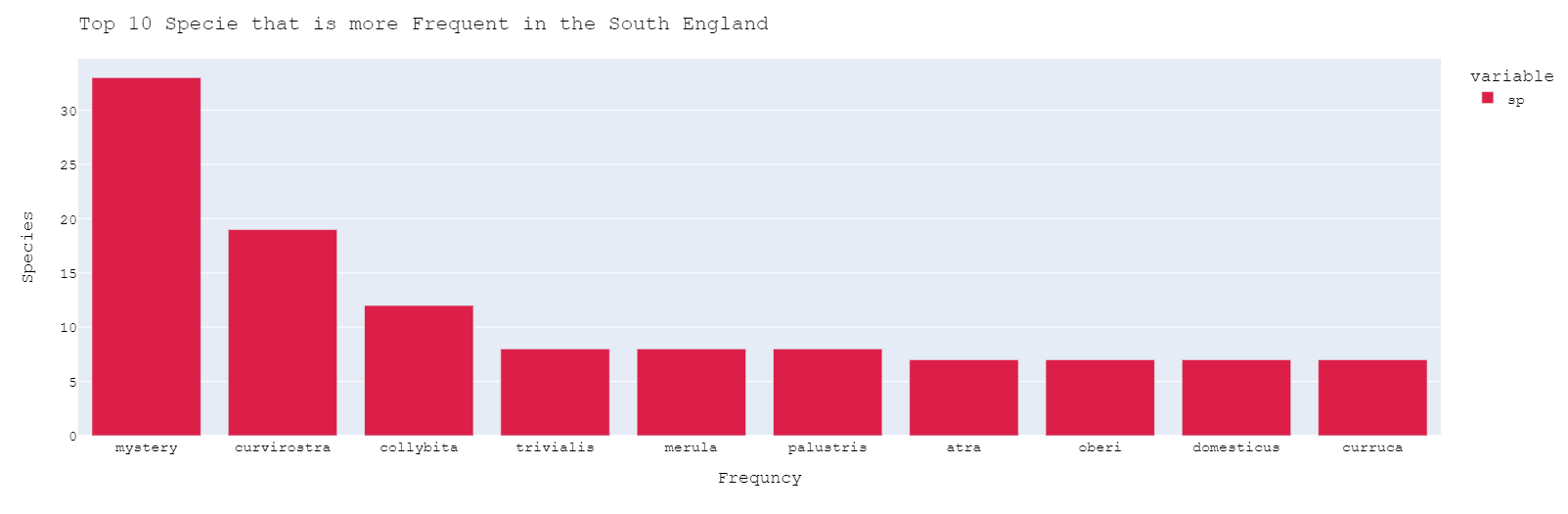


Figure 3. Most Frequent Species in North and South of England

The second object is to find **“Is there a specie that sings more frequently in the morning than in the evening”**. We have time in minutes and seconds in order to convert this time into an hour we should first convert it into hourly data. We created an hour column on the basis of ":" and then we extracted the hour of the data. Now we have an hour column, but we don't have a session of the day e.g. 'Late Night', 'Early Morning', 'Morning', 'Noon', 'Evening', 'Night'. To convert hours into these sessions we have to convert them into different sessions of the day. We convert it on the bases of hourly basis like 24 hours in a day has been converted into chunks of the session then map into their respective session. By applying the query, we get the top 50, 50 Species that sing more frequently in the Morning and Evening as shown in Fig. 4. We filtered the data based on morning and evening time on an hourly basis and create a news column name session. We analyze the column with respect to the most frequent species which occurred in the morning and evening. We notice that Bird Collybita occurred 1158 times in the morning and 36 times in the evening. So we conclude the result for this scenario that Collybita is more frequently sings in Morning than Evening.

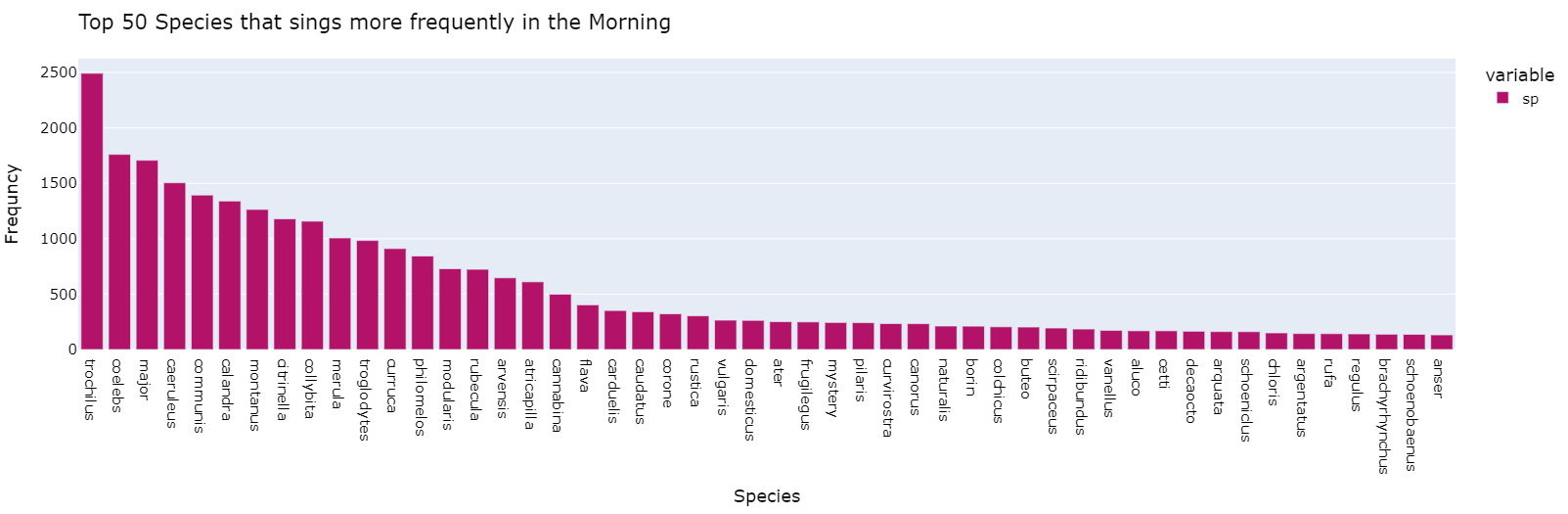
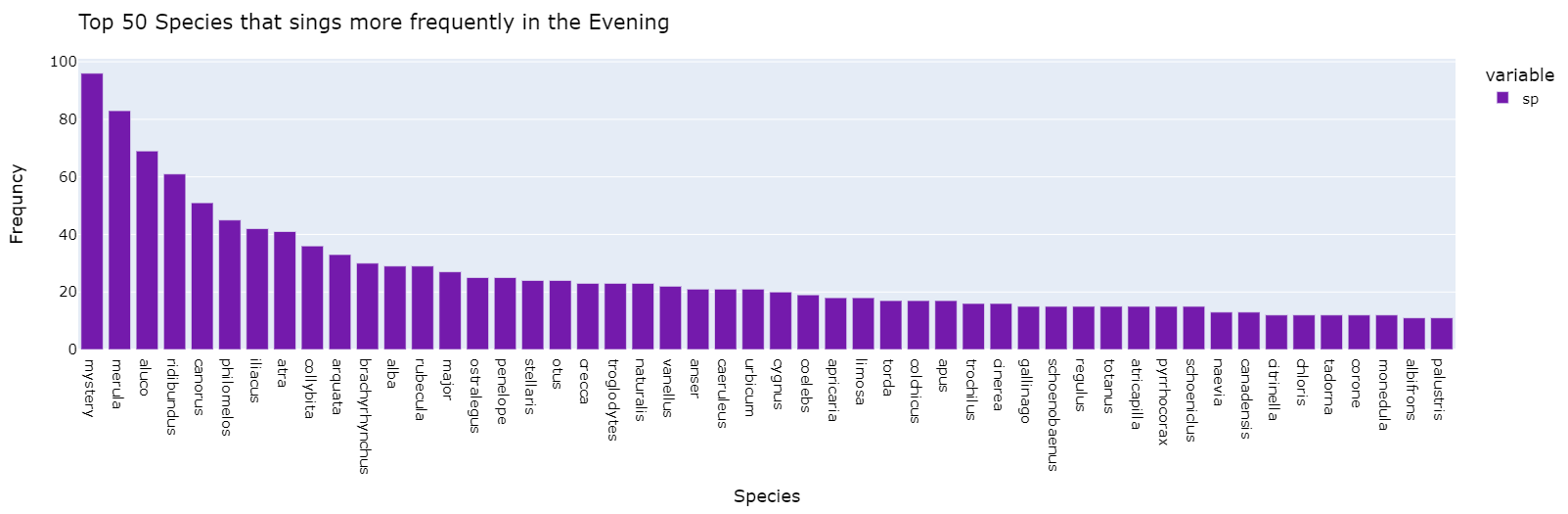


Figure 4. Top 50, 50 Species that sing more frequently in the Morning and Evening

We have to **“Work on the geographical distribution of the bird species”**. Firstly, have to find out where recordings were made. Since I have the latitude and longitude of the location where recordings were made, I'll be plotting the exact locations on the map where the recordings were made. Note: Let's look at the top 15 countries with the most recordings. The majority of recordings are located in the UK, followed by their cities. The initial step is to convert Country/Cities to iso\_alpha format to plot on Map. We have very noisy data in the location of the column, so we have to extract the country name from that loc column. So, we split the "loc" column data by using ",". As the country name is present after "," After that, we extract that name of the country and store it in the new country column. After the creation of the country column, we have to clean the country which has ambiguous data we update the country with respect to their city as we have checked them manually and replaced their name with their states. The country short form is called iso\_alpha so we convert the name of the Main country which is the United Kingdom, so we convert it into short or iso\_alpha code to plot on the map. As per the data, we are gathering information about countries we find that the provided data is from the United Kingdom as there is no other country involved in the latitude and longitude, so we are assuming that we are not adding another database for the bird to this data due to inconsistency. Normal Distribution without Map and as we are seeing that most of the birds are from the UK a very fewer and limited number of birds are from other continents/countries. Distribution of Map of the England and mapping the location of the spices shown in Fig. 5.

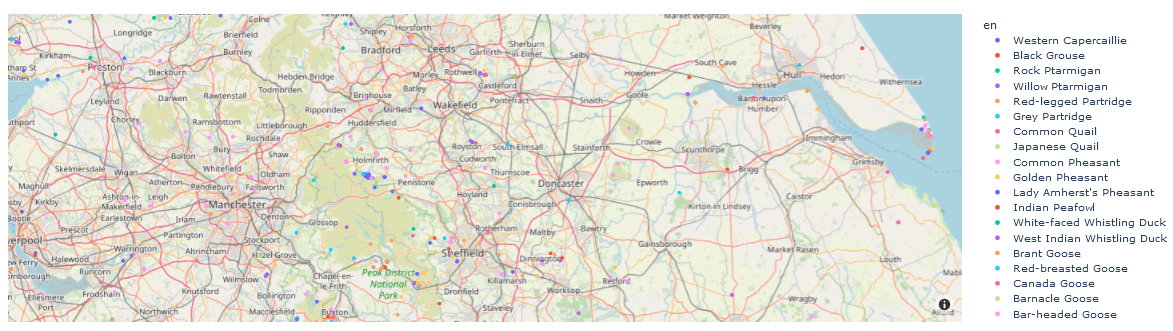


Figure 5. Distribution of Map of England and mapping the location of the spices

Top 15 most common elevations for the bird’s species by counting their occurrence in the newly created country column. Fig. 6. Shows the top 15 bird species and Fig. 7 Percentage most occurred of bird species,

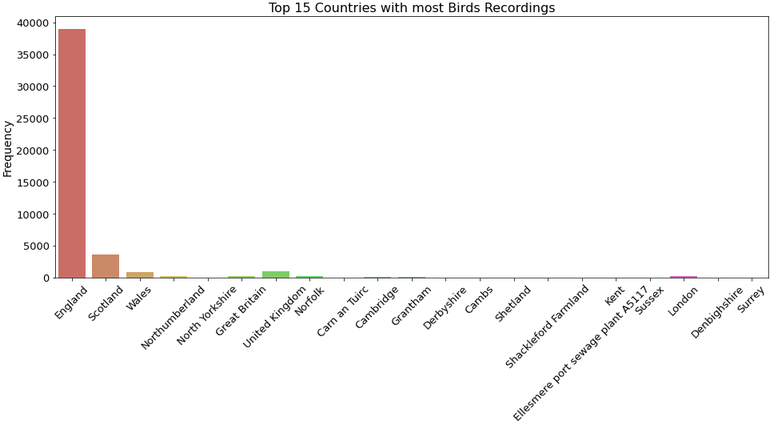


Figure 6. Top 15 Country having Most Bird Species Recording

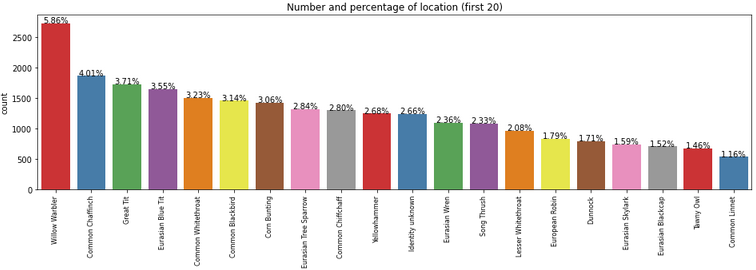


Figure 7. Most Occurred Bird’s Species Percentage

From the above operation of data cleaning and data grouping/aggregation based on another important factor, we decide that the most common type of birds and recording are taken from England, which is shown in Geographical Distribution, and the Most frequent Birds Recording are from the United Kingdom specifically from England. By Further analysis, we extract the city where the most birds are being found in the city of England named “Willow Warbler”.

We have to do an Analysis of the time of the day and potentially look at a different time of the year e.g. Do we have more recordings in the spring than winter? The data contains months as we converted the date column. Now we have to find the different seasons of the year. So, we have to convert the months into their respective session e.g. “Winter", "Spring", "Summer", and “Autumn”. The session is cross-checked with England weather/session. Finds the most frequent uploading of the bird species recordings in the spring session as shown in Fig. 8.

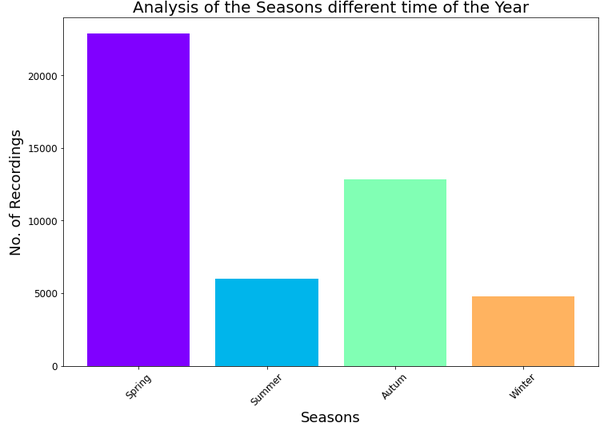


Figure 8. Most Frequent Recording in Different Seasons of the Year

Now analysis of bird recording with respect to Years. Finds the most frequent uploading of the bird species recordings in year columns which are 2020 as shown in Fig. 9.

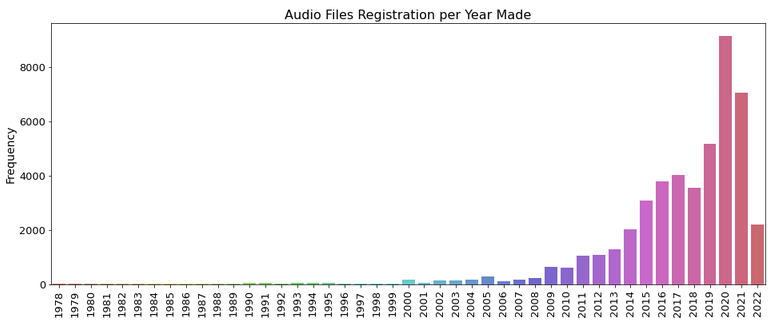


Figure 9. Frequent Recording with Respect To Years

We also did the analysis of bird recording with respect to months as well. Finds the most frequent uploading of the bird species recordings on the basis of months. So, we detect that soring session and April month are more suitable for Bird species recording registration as shown in Fig. 10.

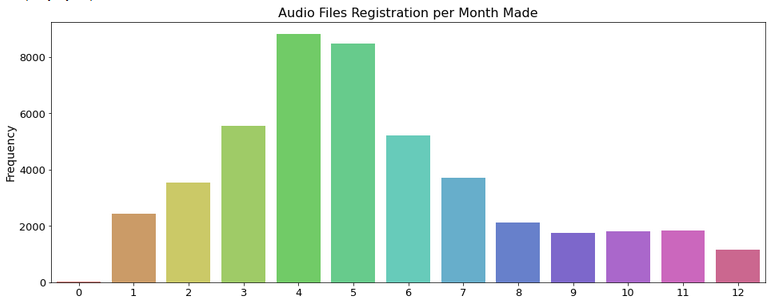


Figure 10. Frequent Recording With Respect To Months

As per the frequencies of the recorded voices months, we have more recordings in the spring than winter as shown in the graph. The most specific month is April when Recorder uploads 8000+ Audio file registration. So, we conclude that Audio file registration is potentially looking at the time spring season of the year in the morning time of the day. Weather conditions in the UK follow as: Spring (March, April, and May) is a time for sudden rain showers, blossoming trees, and flowering plants. Summer (June, July, and August) is the UK's warmest season, with long sunny days, occasional thunderstorms, and, in some years, heatwaves. Autumn (September, October, and November) can be mild and dry or wet and windy.

**“Do we have more species for which we have only 1 subspecies?”** to find out this we applied different filters using pandas libraries. We get “Number of Species which have Exactly One Specie”, "Number of Species which have Different Species" and "Species for which we have only 1 Subspecies “.Analysis of the Species for which we have only 1 Subspecies or more than 1 Subspecies" is shown in Fig. 11.

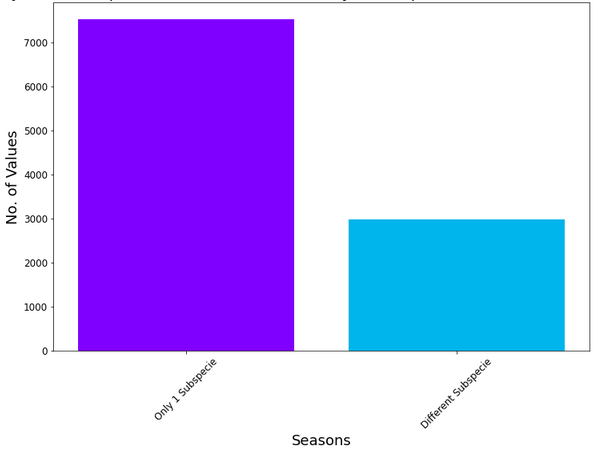


Figure 11. Species having only 1 Subspecies and Different Subspecies

From the above comparison graph, we can clearly see that number of specie that have exactly one subspecies is greater in number than the number of specie that has different subspecies. For reaching this conclusion, we firstly calculate the number of rows that have one specie and in front of that subspecies exactly have 1 one subspecies and similarly, we did vice versa for different subspecies.

**“What is the Most Frequent sub specie and for other Species”?** We have to find the most common frequent subspecies that occurred in the sub-specie column. From Fig. 12 we can conclude that “Scotica” is the most frequent sub-specie.

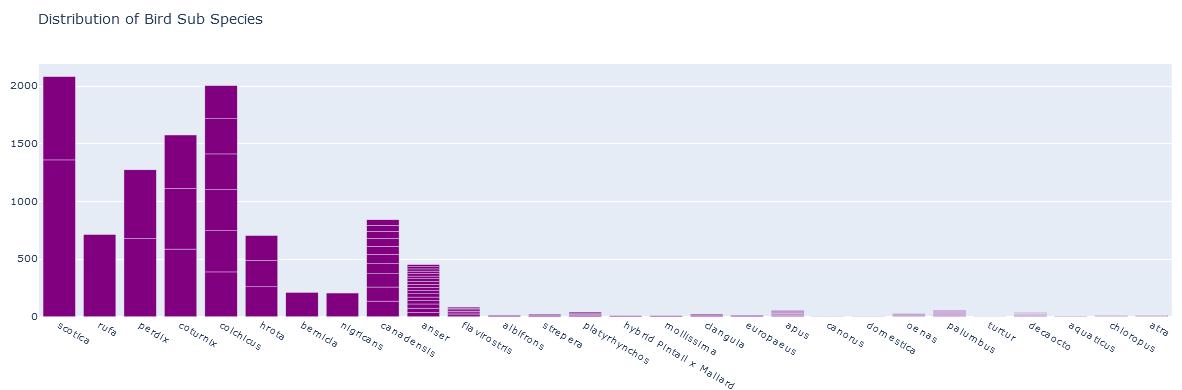


Figure 12. Distribution of Bird Sub Species

Now have found most common frequent species occurred in the spice’s column using filtering. By applying a filter, we get that Scotica is the most frequent subspecies and is the most frequent specie also Curvirostra is the specie.

**“The behavior of 10 people who are frequent uploaders”?** As we already find the frequent audio recording registration per so we will analyze this question on the basis of Uploader Person. First, we will find the frequent uploader of bird species audios with respect to morning and evening. Then after we have to find the behavior of the uploader in which year and which month and which day of the month, they were uploading the recording. In this analysis, we find that the most frequent uploader of the recording is David M having uploads of 27776 as shown in Fig 13. Now we will find the behavior of the Uploader that it may upload mostly at night or morning, or which part of the day or year as shown in Fig 14.

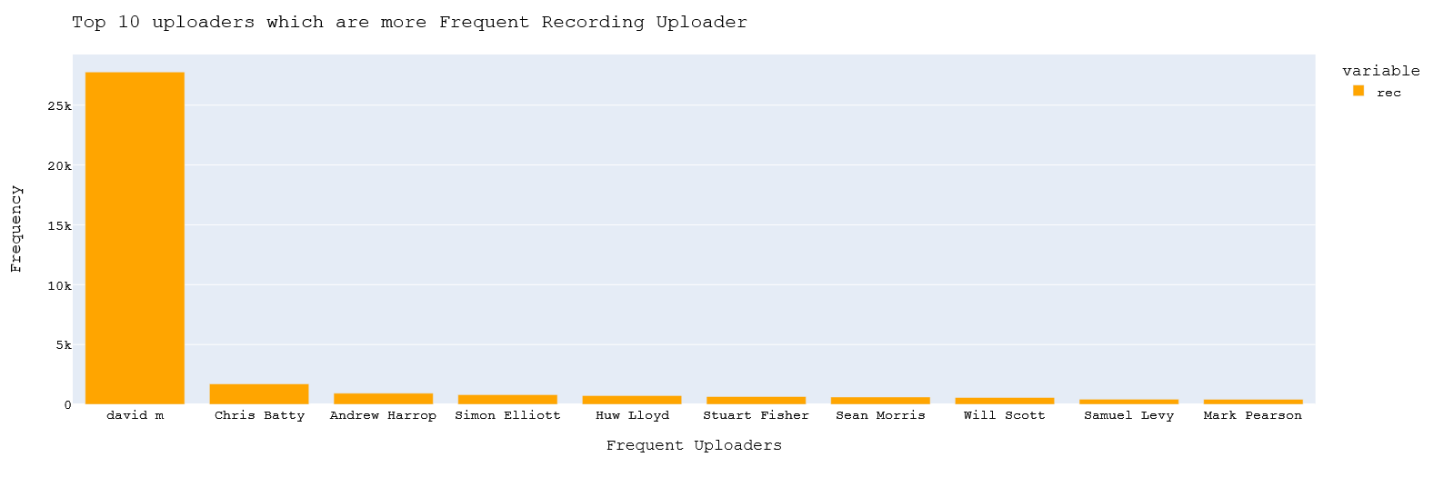
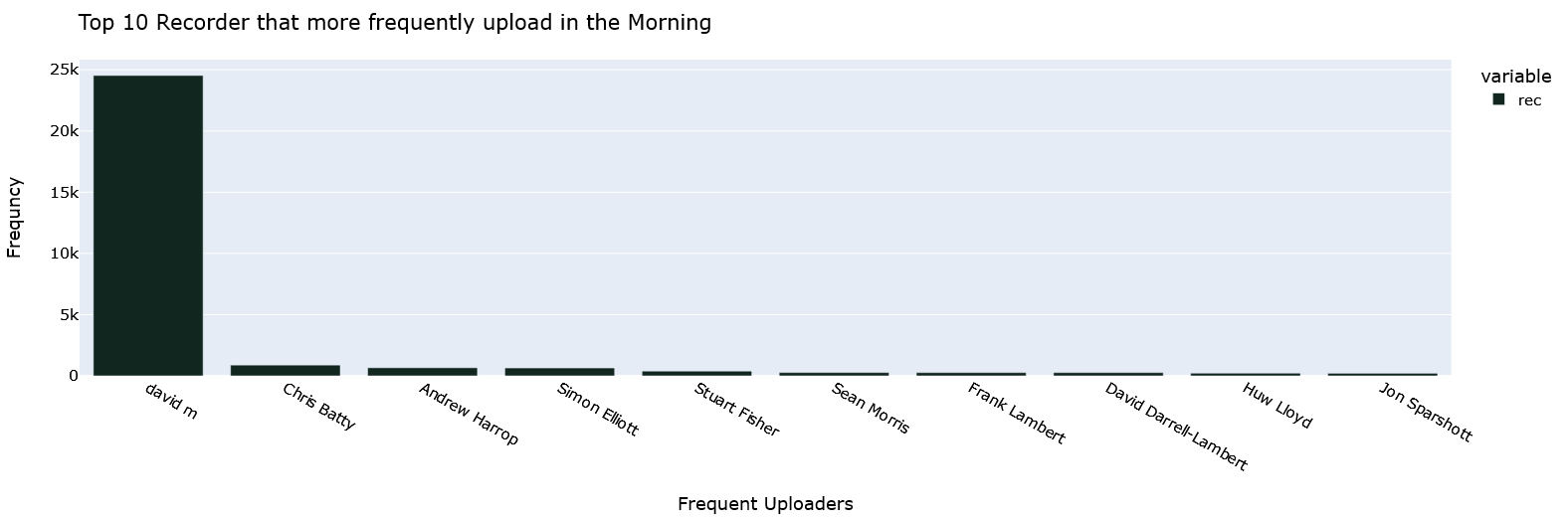


Figure 13.Top 10 Uploaders Which Are More Frequent Recording Uploaders



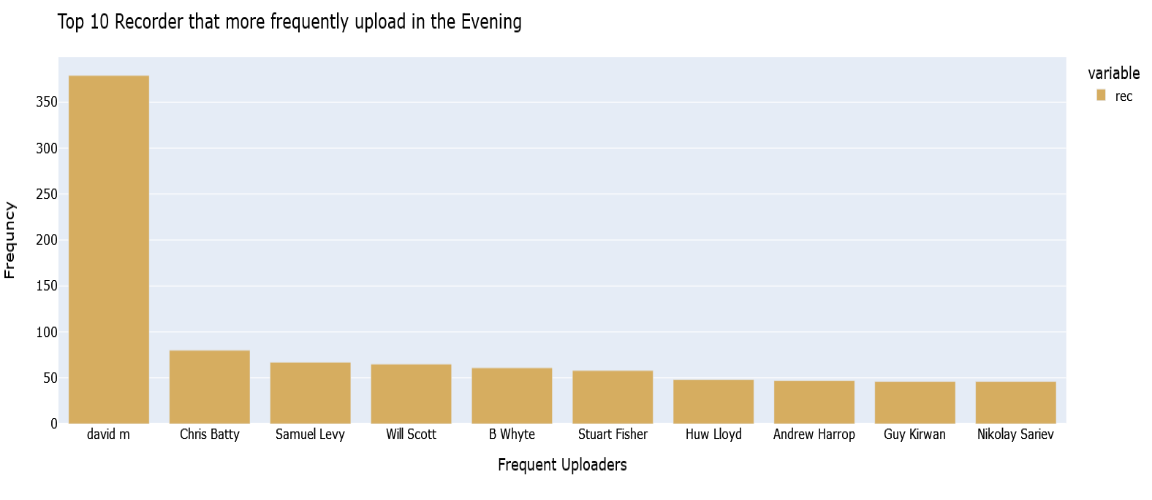


Figure 14.Recorder That More Frequently Upload in the Evening and Morning

# Chapter 4: Analysis Approach and Implementation

## Methodology

Experiments are performed on a real-world dataset of bird species from xeno-canto. Algorithms are implemented in python using colab. Colaboratory, commonly known as "Colab," is a Google Research product. Colab is particularly well suited to machine learning, data analysis, and education. It enables anyone to create and execute arbitrary Python code through the browser. We also import different python libraries such as NumPy, pandas, seaborn, and plotely for graph plotting matplotlib and also for graph plotting. For standard scaler and normalization also used the sklearn library. For train and test validation split we also use the sklearn for preprocessing, we import different models like logistic regression K-nearest neighbor classifier, random forest, random search CV, decision tree classifier, and support vector machine also from sklearn. We also import evaluation metrics from sklearn where we import confusion metrics, classification reports, and cross-validation scores.

## Data Preprocessing

To build a Machine learning model accordingly to the required scenarios, like to find a pattern to identify 4 or 5 specie that have at least 100 recordings that are consistent in length like 50 seconds. We need to do some preprocessing before building the model. For that purpose, first, we read the data from CSV, which we saved after exploratory data analysis (EDA) of bird species data. Then we check how many feature columns we have and their respective counts and data types. Then we change the data type of the length column and filter out the rows that count the value of the length column equal to or greater than 0:50.

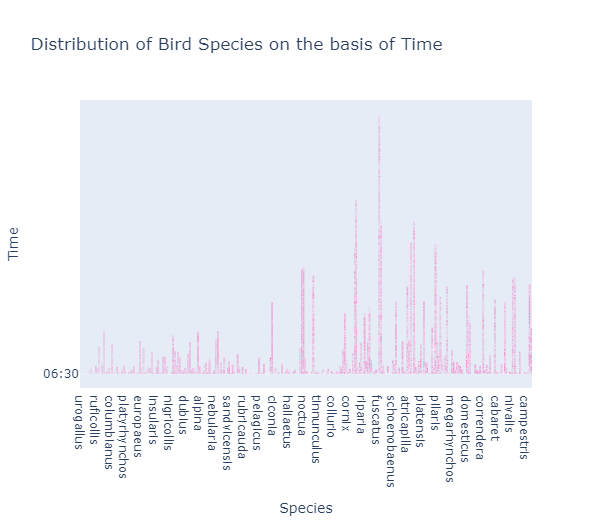


Figure 15. Distribution of Bird Species on the basis of Time

As we see in above Fig. 15, we found many species that have recording lengths greater than 0:50 seconds, but we have to tell 4 to 5 species that have greater recording lengths than 0:50 seconds. For that purpose, we apply another condition where we consider only those species that have been recorded more than 1000 times. In Fig. 16 we can see that TROCHILUS, MERULA, COELEBS, and CALANDRA are those four species that have more frequent recording lengths greater than 0:50 seconds.

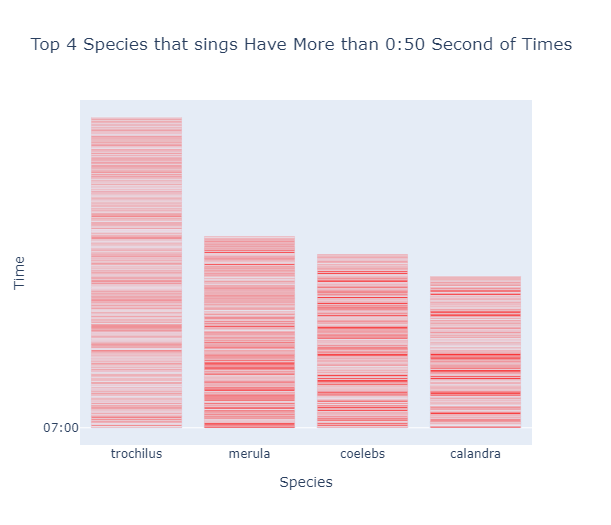


Figure 16. Species that sings Have More than 0:50 Second of Times

The next Scenario we had was to build an algorithm to process 6-month worth of data. To do that, work predominantly on some kind of ML algorithm that has now been. For that, we first find the most frequent upload in the particular year so we can extract the last six months of data accordingly to the most recorded species for that particular year.

Chart, histogram

Description automatically generated

Figure 17. Number Years Count for Species Occurrences

Fig. 17 shows that in the year 2021 we found most recordings of specie. The next step is to build a model for our problem. So, for that purpose, we have to readjust the columns for dependent and independent variables. Also, we change the date column to a string so we can apply different conditions according to our required scenario. We are selecting the most entries for the year 2020 and selecting six-month data from 2020. Then we extract the data from January 2020 to June 2020 and save it in a separate data frame. Then we do exploratory data analysis for the newly separated data frame to find useful insights and patterns. Before deep diving, in EDA we first drop non-important features which are non-significant in classification and have no impact on final output such as gen, en, rec, lat, lng, alt, year, month, day of the month, hour, etc. We also find the most frequent country for specific specie and found that most of them are found in England and Scotland as shown in Fig. 18.

Shape

Description automatically generated

Figure 18. Bird Seen Count W.R.T Countries

After that, we extract the data then we should find how many classes for this data like calculating independent variables. We found that we have 196 classes for the last six months of data for the year 2020 so we should check the distribution of the classes if we have classes conflict then we should remove them. First for further complication in modeling we plot species type count, so we have an idea about the number of values counts of each specie as shown in Fig. 19.

A picture containing chart

Description automatically generated

Figure 19. Bar plots for all species in Dataset

We noticed that our classes are very imbalance, so apply some selective sampling methods to adjust the distribution of classes. For that purpose, we use the counter like we should remove all those classes which have 20 or fewer entries per classes so that we can overcome the class imbalance and other complications in the model. After that, we checked the number of remaining classes so that we should encode labels for standardization and other processing methods. We see that from 196 classes we reduce it to 55 which means that we were having a large number of entries which have between the range of 1 and 20. Then, we checked again data imbalance and class distribution to check imbalance classes to balance by different methods as shown in Fig. 20.

Chart, histogram

Description automatically generated

Figure 20. Species Distribution after Filtering the Data

After that once again checking for Null Value in the columns, we found it again then removed it by just dropping it and saving the data that we have cleaned and filtered from the original data for future use. Then we find a statistical description of numerical values of data as shown in Table. 3.

Table 3. Statistical Description of Numerical Variables

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | lat | lng | alt | year | month | day\_of\_month | hour |
| count | 4913.000000 | 4913.000000 | 4913.000000 | 4913.0 | 4913.000000 | 4913.000000 | 4913.000000 |
| mean | 53.807475 | -1.051497 | 88.027885 | 2020.0 | 4.242215 | 16.091797 | 7.765317 |
| std | 1.825977 | 1.660947 | 52.059047 | 0.0 | 1.339679 | 8.883035 | 5.536200 |
| min | -28.289900 | -6.587600 | -50.000000 | 2020.0 | 1.000000 | 0.000000 | 0.000000 |
| 25% | 52.877300 | -1.392776 | 70.000000 | 2020.0 | 4.000000 | 8.000000 | 5.000000 |
| 50% | 54.122600 | -0.547300 | 80.000000 | 2020.0 | 4.000000 | 17.000000 | 6.000000 |
| 75% | 54.125200 | -0.542800 | 100.000000 | 2020.0 | 5.000000 | 24.000000 | 8.000000 |
| max | 60.800100 | 86.005000 | 440.000000 | 2020.0 | 6.000000 | 31.000000 | 23.000000 |

Correlation table between different numerical features so we get a clear understanding, of which features are highly correlated and those are not as shown in Table. 4.

Table 4. Correlation table b/w Numerical Features

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | lat | lng | alt | year | month | day\_of\_month | hour |
| lat | 1.000000 | -.602040 | -.028790 | NaN | -.027299 | -0.007091 | -.025152 |
| lng | -.602040 | 1.000000 | -.129397 | NaN | -.130516 | -0.011163 | -.096040 |
| alt | -.028790 | -.129397 | 1.000000 | NaN | 0.040901 | 0.035417 | -.009846 |
| year | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| month | -.027299 | -.130516 | 0.040901 | NaN | 1.000000 | -0.113391 | 0.019661 |
| day\_of\_month | -.007091 | -.011163 | 0.035417 | NaN | -.113391 | 1.000000 | 0.015420 |

As we have seen in the above Table. 4, we have two kinds of values ranging from positive and negative. Actually, we have two kinds of correlations between any two features it should be highly positive correlated or highly negative correlated. So, we have to consider both kinds of correlations for our scenario. We also plot the correlation metrics for visualization as shown in Fig. 21.

Chart

Description automatically generated

Figure 21. Color Map with Correlation for Numeric Features.

Graphical user interface, application, Teams

Description automatically generated

Figure 22. Heatmap correlation metric for numerical features of different bird species

## Data slicing between dependent and independent variables

First, we slice data variables between dependent and independent variables, and we convert all those columns for which we have non-numeric data into numeric. So, we can visualize and draw a correlation and then standardize the values. For that purpose, we have to label and encode them. We have used two methods like one hot encoding and label encoding and we will use only one of them that is more suitable for ML algorithms. The popular encoding method for categorical data is label encoding. According to this method, an individual number is given to each label depending on its alphabetical order. But label encoding has some limitations and problems for example take an example that we are trying to convert a column that includes countries' names. The country names in the aforementioned situation do not have a hierarchy or rank. However, when label encoding is done, the alphabets are used to order the nation names. As a result, there is a very high likelihood that the model accurately depicts the relationship between nations like India, Japan, and the US. If we don't want this, for that purpose researchers introduced the idea of one-hot encoding. Another widely used method for handling categorical values is one-hot encoding. It merely generates new features based on how many distinct values there are in the categorized feature. The category's individual values will all be included as features. Making dummy variables is the procedure of one-hot encoding. Each category is represented as a one-hot vector in this encoding method. One-Hot Encoding creates a Dummy Variable Trap because the values of one variable can be readily anticipated using the information from the other variables. A scenario known as the "Dummy Variable Trap" occurs when variables have a high correlation. The Dummy Variable Trap causes the multi-collinearity issue. When there is a reliance between the independent characteristics, multi-collinearity arises. Machine learning models like linear regression and logistic regression suffer greatly from multi-collinearity. In order to solve the multi-collinearity issue, one of the dummy variables must be removed. So, both have pros and cons so it depends on us which one is more suitable for our scenario. So, according to our scenario label encoding is far better than one hot encoding because we are going to used logistic regression and the same kind of ML algorithms on our dataset which suffer greatly from multi-collinearity. We import label encoding from sklearn preprocessing and transform our non-numeric class/ predict column. Then we do the same for input features and apply encoding for non-numeric input features such as gen, rec, en, and country. Then, we find the correlation between converted and overall columns as shown in Fig. 23.

Graphical user interface, application, Teams

Description automatically generated

Figure 23. Heatmap Correlation graph between converted and Numerical Features for the Bird Species

Graphical user interface, application, table, treemap chart

Description automatically generated

Figure 24. Colormap Correlation graph between converted and Numerical Features for the Bird Species

## Splitting Data into Train and Test

Train and test are useful for testing machine learning algorithms applicable to prediction-based algorithms/applications. It is a quick and easy method that allows us to compare our machine learning model results with others. We used a 75:25 ratio for the train test split which means we used 75% of our data for training purposes and 25% of data for testing purposes. Now for training data, we have 3684 rows and 11 features and similarly, for testing, we have 1229 rows and 11 features to train our machine learning models. We have applied Logistic Regression, K-Nearest Neighbors, Random Forest, Support Vector Machine, and Decision Tree to classify different bird species. Each of the models is elaborated on in the below sections.

## Support Vector Machine

In 1992 Vapnik designed a support vector machine, in supervised machine leaning algorithms Support Vector Machine (SVM) is amongst the most efficient methods. It was utilized for a wide range of applications including regression, classifying objects, and selecting features. SVM uses the theory of margin as its principal to evaluate the optimal dividing hyperplane for classification. These margins are also known as the distance from the nearest points to the hyperplane which can be maximized for better generalization. An exchange is made between reducing the incorrectly classified number of samples and increasing the margins. The capacity and performance of the model are governed by some constraints. The bias and variance of the model can be balanced through this exchange. Later in this section formulation for SVM-based classification will be discussed for both linearly separable and non-separable datasets. (Burbidge et al. n.d.) (Dietterich and Kong 1995)

#### Linearly Separable Classification

Assume the number of training examples is N, which are precisely linearly separable, where a characteristic is possessed by each point and associates to either one or both classes having +1 and -1 values respectively. The data of the training is transformed into a format (xi, yi) here i = 1... N, yi ∈ {+ 1, −1}, x ∈ RK. It implies that yi = + 1 if xi ∈ class T, and yi = −1 if xi ∈ class F. To separate the two classes’ a hyperplane can be created having the following properties in equations 1,2,3,4.

**(1)**

“w” here is the perpendicular normal vector to the plane, and bias is denoted by b which defines the point is located in relation to the origin. Two parallel bounding planes can be used to demarcate the two categories in this instance.

**(2)**

**(3)**

The rule to provide decision is given by ƒ(x) = sign ( wTx + b). Combining the above two boundary conditions results:

**(4)**

Support vectors are the points residing on the plane wTx + b = ± 1.

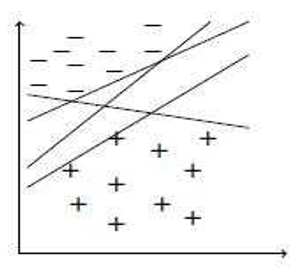


Figure 25. Different Hyperplanes

Fig 25 shows numerous hyperplanes which can be created to separate the categories present in the training set. But even so, only one among these can correctly classify by obtaining the largest distance and maximum separation. For improved generalization, the SVM model aims to correctly categorize the points as well as move them away from one another. This shows that points carry no information which does not support vectors. The SVM model seeks out the dividing plane by increasing which is the same as decreasing

Consequence to direct Quadratic Programming (QP) optimization (EECS, University, and 2006 n.d.) (Lee et al. n.d.) in equation 5.

**(5)**

Several datasets consisting of curved decision separating boundaries are present in real life which are inseparable from linear decision boundaries. Linearly inseparable classes or datasets that are partially separable can be dealt with by advancing the SVM. The model is allowed to drop some examples on the incorrect side of the hyperplane to handle such cases. Soft margin is included so the SVM is customized to accomplish the task of allowing a few points to fall on the incorrect side without impacting the outcome or infringing the constraints. Data points falling on the incorrect hyperplane with the soft margin have a penalty depending upon the distance from it to the margin boundary. To calculate this soft margin, non-negative nonlinear variables are included that make up for the possibility that some points maybe wrongly classified. The boundary constraint violation is measured by the quantity of the slack variables.

**(6)**

**(7)**

The two equations can be merged as:

**(8)**

The goal is to accurately classify the data points having the maximum distance from the margin and reduce the quantity of misclassified points. The following formulation fits the purpose.

**(9)**

Here C is the positive parameter that directs the change between the penalty of the slack variable and the margin. The smaller value of C will make more data fall on the incorrect side and may inadequately fit the training data likewise it will over fit the data if the value is large which results in faulty generalization (Burbidge et al. n.d.)(Lee et al. n.d.)

#### Kernel Trick

Most of the datasets are nonlinear and inseparable but mapping them into the high-dimensional region through nonlinear mapping can make them linearly separate. For this purpose, kernels are utilized to transfer the trained dataset from its input region Rn to high dimensional space through a nonlinear function ɸ. Set of features utilized in categorizing the training dataset constitute the feature space. The feature space has l dimensionality in which the training data xin present in the input space Rn will be transformed into . The dot production of the inner rates helps acquire the kernel, which resultantly defines the mapping. This shows in inner space Rn hyperplane will be nonlinearly compared to the linear separating hyperplane in the feature region F. To identify inner products in the feature space by mapper inputs, kernels proved to be helpful by using a nonlinear mapping, casting a function into high-dimensional space again. Thanks to the kernel trick SVM can build nonlinear boundaries. Many kernel functions are available to be used, depending upon the case suitable kernel is chosen by the test set through trial and error as shown in Fig. 26.

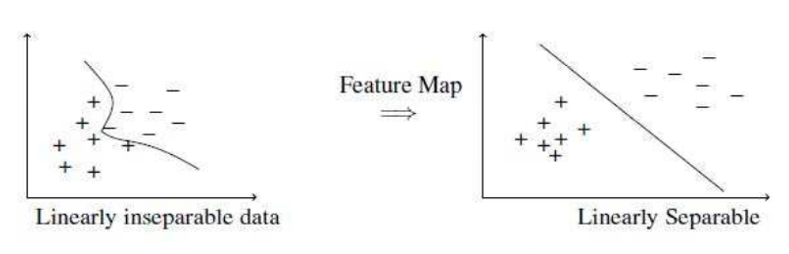


Figure 26. Separating Features Boundaries on the basis of Linear and Non-Linear Data

Support Vector Machine is comparatively easy to train which the primary benefit of the algorithm is. On high dimensional data sets, it performs moderately well. It can easily control the exchange between complexity and error. SVM is designed to deal with categorical as well as continuous data. The nonlinearity of the data can be detected easily. Being a non-parametric method, no assertions regarding data sets are needed. Its prediction rate is very high and shows generalization performance very well. Having a convex optimization problem provides a particular solution with a distinctive minimum value. For erroneous datasets, SVM is more capable and reliable. If the features are not interpretable, SVM cannot interpret them which appears to be a significant drawback of the model. An excellent kernel function is required which can be expensive computationally. The obtained results are also opaque because it is a non-parametric method.

## Random Forest

Another machine learning model is the random forest, it can be used for handling both classification and regression problems. It makes use of ensemble learning, a method for solving complicated problems by combining several classifiers. The random forest consists of various decision trees. Using various decision trees, a forest if created that learns using bagging and bootstrap aggregation. The efficiency of machine learning techniques is increased by bagging, an ensemble meta-algorithm. Depending on the assumptions by the decision trees, the result of the random forest is determined. It makes predictions by averaging or taking the mean of the output of the different trees. The accuracy of the result and the number of trees are directly proportional.

The limitations of the decision tree model are eliminated by a random forest. It improves efficiency and decreases the overfitting of the data. Predictions are generated without demanding numerous packages of configuration such as sci-kit- learn. Compared to the decision tree classifier, it is more reliable. It offers a practical method for dealing with missing data. Hyper-parameter tuning is not required for generating a reasonable prediction. It fixes the overfitting problem with decision trees. At the splitting point of the node in each random forest tree, a subset of attributes is randomly chosen as shown in Fig. 27.

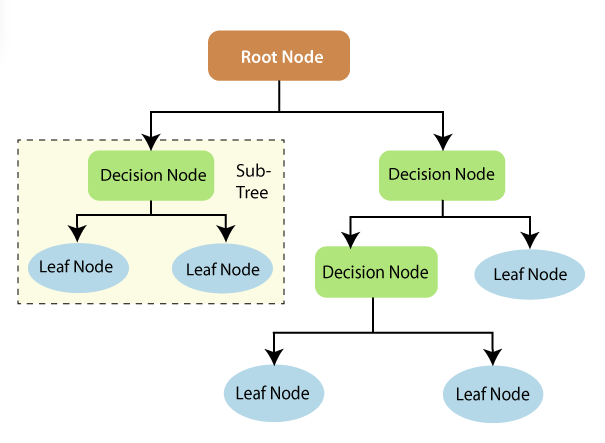


Figure 27. Random Forest Overview

Decision trees act as a core component of random forests. A decision analysis method that has a tree-like architecture is called a decision tree. The characteristics which are utilized to assess the result are represented by the nodes of the tree. Decision tree nodes help to connect the leaves of the trees. The nodes with three various types in a decision tree are depicted in the diagram below. The scientific method can shed lighter on the working of decision trees. The foundation of a decision tree is based on information gain and entropy. A summary of these basic premises will help us better comprehend the construction of decision trees. Entropy is used to measure uncertainty. Information gain measures the degree to which uncertainty from the target variable is decreased considering several independent variables. The idea behind information gain (class) shows that independent features are used to learn more regarding the target variable. The information gain is calculated using the entropy of the target attribute (Y) along with the conditional entropy of Y (given X). In this instance, the entropy of Y is reduced by conditional entropy. The training makes use of information gained. Using information gain decision trees can be trained. It aids in getting the uncertainty of the tree significantly reduced. A high information gain denotes the removal of a large amount of uncertainty. Splitting branches of the tree, a crucial step in the creation of decision trees, depends on entropy and information gain. Take a look at a straightforward decision tree instance. Let's say a prediction is needed whether or not a customer will buy a mobile phone. His decision is based on the phone's functionalities. A decision tree model can be used to display this analysis. The above-mentioned phone qualities are represented by the decision's root of the tree and decision nodes. Whether a phone is purchased or not depends upon the outcome of the node. The internal storage, RAM, and price are the primary criteria for selection.

Randomly selecting the root nodes or grouping the nodes is the primary difference between the decision tree and random forest. Bagging is utilized by random forests to produce the necessary estimation. Bagging entails utilizing multiple training data samples instead of a single sample. Different features and observations from the training set are used to make predictions. Based on the training set that the random forest receives, a variety of outputs for the decision tree are generated. Among them, the output with the highest rank will be chosen to be the final output.

### Classification in random forest.

The Ensemble technique is used to achieve classification results in random forest. Different decision trees are trained through the training set. Randomly chosen features and observations while dividing the nodes are included in the dataset.

A random forest uses various decision trees including their leaf, root, and decision nodes. The leaf node among them depicts the output of a particular decision tree. The final output is chosen by the majority-voting method. The result of the majority of decision trees is chosen to be the final output of the decision tree classifier as shown in Fig. 28.

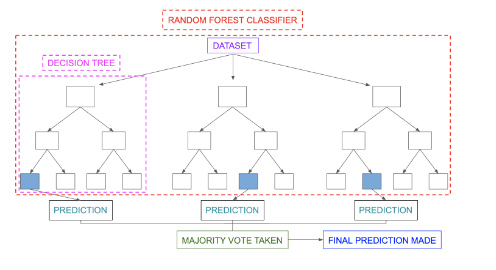


Figure 28. Workflow of Decision Tree

## Decision Tree

Among the classifiers of machine learning and statistics decision tree is one of the most popular in them. It is a hierarchal model that uses the partitioning method. It can be utilized for classification as well as regression problems because the decision tree is a nonparametric model. A decision tree can be transformed into a basic if and then set of rules directly. The presentation of the decision tree makes it easy to comprehend and perceive the output of the model. The fundamental components of the model are covered below (Alpaydin 2020) (learning and 1997 n.d.) (Myles et al. 2004)

### Graphical representation of Decision Tree.

Quinlan presented the ID3 model for generating decision trees in 1986. (Quinlan 1986). In 1993, the C4.5 algorithm was introduced as an upgraded and refined version of the ID3 model(Salzberg 1994). A decision tree is based on a top-down greedy approach. The decision tree has splits which are recurred in basic steps. Its structure is based on terminal leaves and nodes in its network. Usually, a pair of attribute values show its input data. The category of branches and leaves at every instance is determined by a particular test function. The procedure is started from the root of the tree moving towards its nodes and branches corresponding to the value of the attribute. The process is carried out at every node until it reaches the leaf. At that point, the label is determined by its predicted class. (Alpaydin 2020) (learning and 1997 n.d.) (Podgorelec et al. 2002). Fig. 29 is shown as an example to determine the job of an applicant on the bases of their experience and qualification.

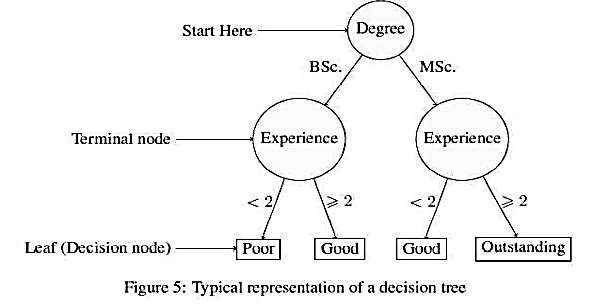


Figure 29.Decision Tree Representation

From the graph above it can be inferred that the performance of an applicant will be considered poor if the or she doesn’t have experience of nearly two years and holds only a bachelor’s degree. The decision tree model can work with multiple classes though the graph only shows binary. Till the final terminal node from the root, the finest features are chosen which makes the algorithm greedy. Starting from the root, the goal is to determine the finest attribute which divides the data at every level accurately. The chosen attribute should be most beneficial for categorizing the data with the greatest differentiation. A statistical method called information gain assesses how effectively the attribute divides the data. Beginning with discussing entropy before actually defining the criterion required for information gain (Quinlan 2014). According to the theory of information, entropy is the lowest amount of bits required to encode the type of an example. Additionally, known as the impurity measure. Suppose there is a collection of S samples and the number of samples that accessed node m is denoted by Sm. The number of samples that belong to j input classes are with m nodes having. The j category has probability, offered with x that attain m node is,

**(10)**

The calculated entropy is given as in equation 11;

**(11)**

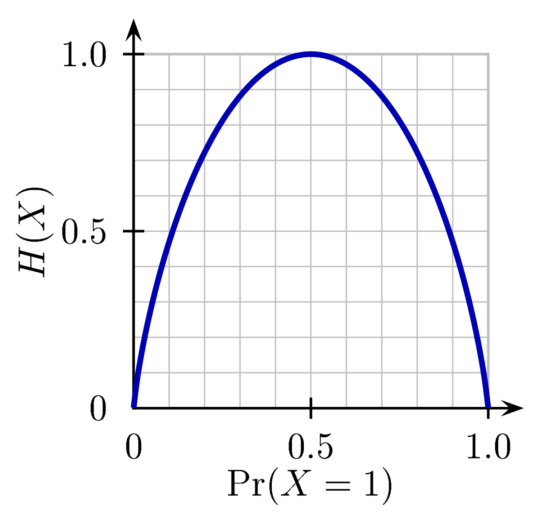


Figure 30.Entropy Function

Fig. 30 shows that the probability of one side will be equal to 1 and the other probability, as well as entropy, will be 0 if every individual data belong to a single class. In contrast, entropy is equal to 1 with 0.5 probability, if samples are divided equally into two classes. If its entropy is lower, the attribute is more helpful (Alpaydin 2020)(learning and 1997 n.d.). Using the information gain criterion, the efficiency of the attribute can be determined, the given entropy is shown as.

**(12)**

In the above equation (A) represents all the possible outputs an attribute it can take, attribute A with value v is gathered in a subset of the whole entropy as Sv The whole entropy before actually dividing the dataset is the first expression in the formula, and the entropy following attribute-based instance dividing is the second expression. An anticipated decrease in entropy following the attribute's value is known as information gain.

The decision tree may over fit, which generally has an adverse effect on generalization and raises the test set error if we keep expanding the tree till every node represents the smallest impurity. Pruning the tree is one method for avoiding over fitting. The tree can be pruned in one of two ways:

* Post pruning is a method that allows decision trees to overfit and grow completely. It later prunes the branches and nodes of the tree which causes over fitting.
* Pruning the tree before its growth and over fitting is known as pre-pruning

Post-pruning is more effective compared to pre-pruning, which is a lot quicker, but post-pruning not only lags in time but also fact that the programmer had no idea when to restrict the growth of the tree. Subsets of the branches are pruned through a pruning set in post pruning. This pruning set differs from the pruning set utilized at the time of training and is also utilized to assess the efficiency of the branches. Subtrees are kept unpruned unless the leaf node outperforms them on the pruning set; in that case, the subtree is pruned and replaced through the leaf node

### Strength

* The graphical representation of the decision tree is simple and straightforward enough to comprehend by any nonprofessional.
* The viewer can usually understand a list of rules transformed by a decision tree.
* Being a non-parametric approach no functional configurations are needed.
* A decision tree can easily handle missing values known as outliers.

### Weaknesses.

* The computational cost of the model is very high.
* Decision Trees are prone to overfitting, but there are several tools such as post and pre-pruning available in practice to prevent this.
* Decision trees are more frequently used in practice for categorization than for calculation based on regression.

## K-Nearest Neighbors (KNN)

KNN is a case-relevant algorithm that while categorizing retains all its data during training. It is a sloppy algorithm and learns very slowly which is why not permitted in most of its implementing areas, such as web mining of a huge repository. Finding some models to show the complete training data set for classification is one method for boosting its effectiveness, i.e. utilizing a training dataset to create an inductive learning approach and its representations for classification. Countless present existing models including neural networks and decision trees were originally developed to construct this model. Criteria to analyze different algorithms is to evaluate them. We are inspired to develop a method for KNN to increase its efficiency by keeping its accuracy as well because it is a straightforward but useful method for classifying data and it is persuading to be among the most efficient strategies on the Reuters corpus of newswire stories in document classification. As seen in Fig. 31 a binary dataset consisting of squares and circles is scatted in 2-dimensional space

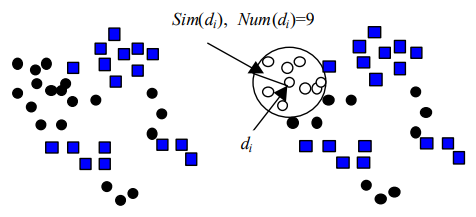


Figure 31. Binary Dataset Consisting Of Squares and Circles Is Scatted In 2-Dimensional Space

Using Euclidean distance to evaluate the similarity measure it can be seen that data points with identical class labels lie close to one another as per distance in several neighborhoods. The number of observation points dI has is given by num (di), including some additional information the data point at the center in this area may serve as an ideal observation of this specific region. The number of data points needed for categorization will be drastically decreased if we use these points as a model to depict the entire training dataset. This will increase the effectiveness of the process of classification. It goes without saying that a new data point will be classified using the label of the class representative if it is encased by it. Or else, we measure the distance between the raw data point closest boundary of both classes, utilizing either boundary as a data point, before classifying the raw data point in accordance with KNN.

During the model development phase, an individual data point owns a huge neighborhood consisting of the largest amount of data points owning the identical labels of that class. The biggest local neighborhood (also known as the biggest global neighborhood) could be determined for every cycle relying on such local neighborhoods. The global neighboring point can be considered as the representative of the entire data it encompasses. Till every uncovered data, the point is covered by some representative the whole procedure is repeated. Naturally, we don't have to select a particular k for our technique during the model-building process; instead, the ideal k is taken as the total amount of data points covered by the representative, but this value may differ with the representative. During the process of building the framework, the k is randomly generated. Additionally, classifying utilizing a list of selected representatives not simply decreases the amount of information required, but also substantially boosts its effectiveness. Out of this perspective, our suggested method fixes the two issues the KNN algorithm acquired.

### Modeling and classification model

The following is a description of the precise model's development

1. Using the provided training dataset, a similarity measure is chosen to generate a similarity matrix.
2. Each data tuple is changed to "ungrouped."
3. Locate the global local neighborhood for every "ungrouped" data tuple that includes the most neighbors who fall into the same class.
4. Generate a representative such as Cls (di), Sim(di), or Num for the data tuple di that has the biggest global neighborhood Ni out of the entire local neighborhoods (di), Rep(di)> into the M to depict the entire data tuples enclosed by Ni, changed the label for complete data tuples enclosed by Ni to "grouped" afterward.
5. Continue doing steps 3 and 4 till the entire training data tuples have changed to a "grouped" setting.
6. The M model is made up of all the participants in the aforementioned training process.

M stands for the developed model in the aforementioned algorithm. The class label di depicts the representative, the enclosed data tuple in Ni with minimum resemblance to di. If more than one neighborhood has the identical maximum number of neighbors in step (4), we select the neighborhood with the lowest value of Sim (di) or the neighborhood that has the highest concentration, as the representative. The following is a description of the classification algorithm:

1. Determine the latest data tuple's resemblance to each representative with in model M before classifying it.
2. Suppose dt is only represented through one representative, and the distance measure between dt and dj is less than Sim (dj), then dt is categorized under dj.
3. Dt is categorized dt by the class representative with the highest Num(dj), that is, the neighborhood includes the most data tuples at the learning stage if dt is encased through at least two representatives of various categories.
4. Dt is categorized by the class with the class representative whose boundary is nearest to dt when no representative from the M model includes dt.

The difference between the Euclidean measure from a representative di to its closest boundary is the Euclidean measure of dt from Sim subtracting (di). Two distinct pruning techniques are added to the KNN algorithm to increase the accuracy of the categorization. One approach entails removing the M model's representatives which only contain a small subset of the data tuples by the training dataset as well as the pertinent data tuples which these representatives conceal. The second approach entails changing step 3 of the model development technique to enable each biggest local neighborhood to contain r data tuples of various groups to the dominant class in this neighborhood (referred to as error-tolerant degree). With this reconfiguration, the pruning is integrated into the model development stage. The following section will present the findings of the experiments.

### Model Construction and Classification Process

The best way to understand this concept is through an instance, so we use visual representations to show how models are built and classifications are made. 36 data tuples are included in a training dataset that is split into two categories, square and circle. Figure 32 depicts the arrangement of data tuples in a two-dimensional data space.

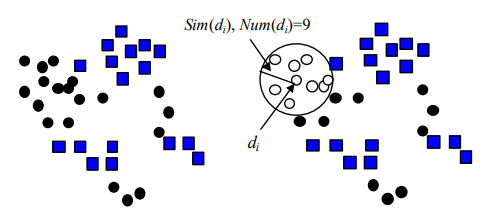


Figure 32. 9 Data Tuples Consisting Of Circles are Included inside the Distinct Line

Fig. 32 shows, that 9 data tuples consisting of circles are included inside the distinct line (Num (di)=9) that share the target class "di-circle" (first pruning technique was utilized in this instance, i.e., we ascribe 0 to r). The first cycle includes the highest number of neighbors that share the same class tag. The Euclidean space covering the farthest distance from di to Ni consisting of data tuple is shown by Sim(di)

The first representative is obtained from the first cycle, which is incorporated with the M model, and moves on to the second cycle. By concluding the second cycle, include an additional representative is included in the M model depicted in Fig. 32. till the complete data tuples are changed to "grouped" from the training set, the process is continued (illustrated vacant circle and square). From the training set, the ten representatives as seen in Fig. 33 are taken and preserved to the M model. from these ten, seven contain over two data tuples, which are indicated by a distinct line circle, the rest three each contain just one data tuple, which is indicated through bold line circle.

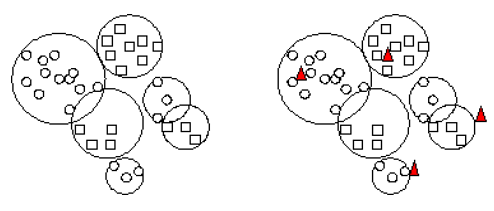


Figure 33. Centroid Selection based on Euclidian Distance

Pruning throughout this work can be accomplished by eliminating representatives from model M (such as Num (di) 2) that only consist of a small number of data tuples. The data tuples included in representatives from the training set will also be eliminated. The model is rebuilt by updating the training set and the final model M is created by the following implementation and pruning for visual representation shown in Fig. 34.

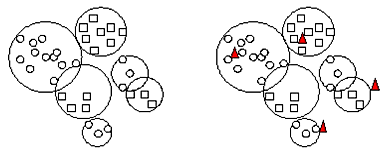


Figure 34. Updated and Improved Centroids

Four triangles from the test data tuples are shown in Fig. 34 These four triangles from the test data are labeled, from left to right, circle, square, circle, based on the classification algorithm previously explained.

## Logistic Regression

Logistic regression though gives a binomial response function very similarly to linear regression. The ability to use constant multiple explanatory variables and the ease with which they can be managed at once is its biggest advantage over Mantel-Haenszel OR. While it may seem insignificant, this final trait is crucial when examining how different explanatory variables affect the response variable. The covariance between variables is ignored if we examine different explanatory variables separately.

Output in the logistic regression models will be based on the light of specific characteristics. The algorithm is a chance ration by which the actual quantity is modeled given as:

**(13)**

The probability that an event will occur is denoted by π here, and βi represents the regression coefficients that are connected with the reference group as well as xi explanatory variables. A significant aspect needs to be emphasized at this point. The members of the comparison group, denoted by β0, are those who define the reference level for the entire variables in x1...m.

### Logistic regression step-by-step

To understand the working and evaluation of the logistic regression let’s consider the previous example. Let's develop a model with all the explanatory factors in it such as age and treatment using logistic regression. With a large sample size and a limited range of variables incorporated, a complete model, also known as a "saturated model," is the ideal place to start. The output of the model is shown in Table 4 below.

Table 5. Logistic Regression Model Values with Different Parameters.

|  |  |  |  |
| --- | --- | --- | --- |
| Term | Estimate β | Standard error | P value |
| Intercept β0 | -2.131 | 0.421 | <0.001 |
| β1 | 0.54 | 0.207 | 0.028 |
| β2 | 1.43 | 0.293 | <0.001 |

All that remains now is for us to perceive this output. Starting with β0's corresponding coefficient of determination. The mean odds for people who died in the reference class are obtained by using the exponent of β0. In other words, the probability of people who are older and have acquired additional treatment and have died is exp (0) = exp (-2.131) = 0.13. Moving on to the following coefficients, there is a slight variation in how they are interpreted. Youngsters who got new treatment likewise have a mean risk of dying that is 1.58 times greater than compared to reference people (exp (1) = exp (0.54)). Similarly, older patients received standard care and had a mean exp (2) = exp (1.43) = 3.89 percent higher mortality risk than the reference group. However, what if the patients were youngsters getting standard care? The mean change of reference people must then be multiplied by exp (1+2) = exp (1.97) = 6.07.

### Odds and probabilities

The fundamentals of interpreting logistic regression look like this. The analysis does bring up some problems, though, and answers aren't always simple to find. We will talk about how to handle them in the following section after this.

It is crucial to first comprehend those odds and probabilities, despite occasionally being used interchangeably, are distinct concepts. The probability is the proportion of events that lead to a particular outcome for all events whether it will occur or not. Conversely, odds are the ratio of two probabilities, such as the chances that an event will be desirable to a result and the chances that an event will not occur. This is significant because a high odds ratio (OR) could even indicate a low probability, and likewise. Let's revisit the earlier instance to better illustrate this idea. The reference group, which consisted of older patients getting new treatment, displayed a death probability of about 0.12 percent. Utilizing:

**(14)**

This could be demonstrated here that the group's average mortality risk is 0.11 percent. From the above equation, 3 possibilities of an individual dying in this group could be calculated as 1.58 x 0.12 = 0.19 or, as a mortality probability equal to 0.16, given that patients who're young with an average possibility of death after getting new treatment are 1.58 higher than the average possibility of morality in the reference group.   Related to this, an older person getting standard care has an average risk of dying that is 3.79 times higher than that of the comparison group, translating to a possibility of dying equal to 3.79 x 0.12 = 0.45 or a probability of dying equivalent to 0.31. Finally, younger patients treated with standard care have a mortality risk equivalent to 5.97 x 0.11.

Although the assessment of outcomes from multivariate explanatory variables is simple and corresponds to that of binomial explanatory variables, it is a little more challenging to comprehend continuous variables. A constant variable's exp(β) signifies the increase in the probability of an occurrence associated with every unit increase in the explanatory variable. As shown, a big OR just indicates that a specific group has a significantly higher chance of occurring than the reference group, as in the example, the variable "the Age" in the earlier example. However, the low probability of the reference group, with a large OR might still show a low probability.

### Continuous explanatory variables or variables with more than two levels

Let’s consider the non-binomial explanatory variable. Whenever an explanatory variable has more than one variable, it is needed to create n-1 binary variables (also known as dummy variables) to be added In simple. In simple words, a dummy variable is something that will take on the value of one when the point falls under the desired class or else will be zero. For example, take x1 and x2 as two dummy variables that must be used in the model to show a variable called "satisfaction" consisting of three levels (Low, "Medium," and "High"). Equation 14a states that the respondents at the reference level, suppose "Low," shows zeros in all dummy variables, while for respondents at "Medium" the value of satisfaction will one in x1 with zero in the rest shows 14b.

Low ----------- **(14a)**

Medium ------ **(14b)**

High ----------- **(14c)**

People with "High" satisfaction levels will experience the reverse (Equation 14c). Typically, statistical software takes care of it instantly, so the viewer isn't required to. While the assessment of results from multivariate regression explanatory variables is simple and corresponds to that of binomial variables, it is slightly more challenging to comprehend continuous variables. A continuous variable's expr (β) symbolizes the increase in the probability of an occurrence associated with every unit increase in the explanatory variable. Table 6 contains the explanatory

Table 6. Multivariate Logistic Regression containing on Continuous Variable.

|  |  |  |  |
| --- | --- | --- | --- |
| Term | Estimate β | Standard error | P value |
| Intercept β0 | 9.041 | 1.514 | <0.001 |
| β1 | -0.295 | 0.041 | <0.001 |
| β2 | 2.230 | 0.298 | <0.001 |

### Variables inclusion and selection

The choice of variables is a significant challenge in logistic models in search of anything "significant," researchers typically gather as many variables as they can before putting them inside the algorithm. As a result of this strategy, two situations are more likely to arise. First, even though multiple variables turn out statistically "significant," the analyst lacks a theory to connect them to the modeled interest event, functioning with samples, which can result in false outcomes. A model having multivariable tends to have lower statistical power, which is the second scenario. As a result, models which are saturated (those that consist of all potential explanatory variables) are dumb and not smart enough to detect suppose a connection between the event that occurred and one explanatory variable. Therefore, the analyst should be extremely careful when choosing the variables to incorporate into the model.

Regression can be started either with a complete (saturated) model or with the null (empty) model that begins with just the intercept term. In the first scenario, variables must be eliminated one at a time, ideally starting with the least important one. This approach is favored simply as it is simpler to manage, whereas the second calls for testing all nominee variables at every stage to determine which the best option should include.  In contrast, as discussed earlier significant variables may be eliminated because of low statistical power when many variables are added all at the same time in a statistical model. Generally, if the size of the sample is large, suppose there are ten entrants for each variable, as a general rule all the explanatory variables can be added. Pre-selection is recommended, though, the size of the sample is small in comparison with the number of variables. Testing all variables first is one way with univariate models which only consider a single explanatory variable at a time and later incorporate all the variables into the multivariate model with relaxed P-values such as P 0.25. No inference will come because of preselecting values it is unnecessary to stress for a strict p-value criterion The P-value factor will enable decreasing the number of initial variables in the system, decreasing the possibility of significant factors being missed.

### Experiments and Results

We applied five Machine learning states of algorithms and compared their results based on accuracy and other evaluation metrics. Following are the machine learning algorithms we implemented on the bird species dataset.

Logistic regression is a statistical model that is often used for predictive analysis and classification. Based on independent variables of the data frame, logistic regression calculates the likelihood that an event will occur, such as voting or not voting. It predicts binary outcomes such as yes or no, win or lose, etc. based on stored observations of a dataset. By examining the correlation between one or more already present independent variables, a logistic regression model forecasts a dependent data variable. We do hyperparameter tuning for logistic regression by changing the value of ‘C’ and ‘Plenty’. An example of a "hyperparameter" is C. The hyper-parameters give the model guidance on how to select parameters, whereas the parameters are numerical values that inform the model on what to do with the characteristics. Extreme parameters will be penalized by regularization, while overfitting is caused by extreme values in the training data. So, the model takes the value of C between 10 and 60 with the step of 10. Another hyperparameter we used is the penalty. With penalized logistic regression, the logistic model is penalized for having too many variables. The less important variables' coefficients eventually approach 0 as a result of this. Another name for this is regularization. We set the value of penalty l2 and l1. In cross-validation or CV, we set a value of 5. Stratified K-Folds are utilized as the standard cross-validation generator. The number of folds utilized is the integer specified if one is provided.

After hyperparameter tuning and applying logistic regression classifier to our train dataset and by comparing the prediction we get on the training dataset with the test dataset. We get overall accuracy of 46%. We also generate confusion metric on the logistic regression model as shown in Fig. 35.

Background pattern

Description automatically generated

Figure 35. Confusion Matrix on Logistic Regression

The k-nearest neighbor’s algorithm, sometimes referred to as KNN or K-NN, is a supervised learning classifier that employs proximity to producing classifications or predictions about the grouping of a single data point. It tries to cluster a large number of unlabeled points into "K" clusters. It is unsupervised since there is no external categorization applied to the points. The "k" in "K-means" stands for the desired number of clusters at the conclusion.

Chart, line chart

Description automatically generated

Figure 36. Testing Accuracy on Different values of K

We test model accuracy by setting different values of k from the range of 5 to 25 and with the step of 5. We get different accuracy values for different values of k, such as when k is 5 we got 73%, on 10 got 69.4%, on 15 67.12%, on k 20 we got 65.7%, and on 25 we got the same 65.74 almost. As shown in Fig. 36, clearly the testing accuracy graph on different values of K. From Fig.36, we can see that the best accuracy we are getting on the value of K is 73.79% when K is set to be at 5. The confusion metric of K Nearest Neighbors is shown in Fig.37.

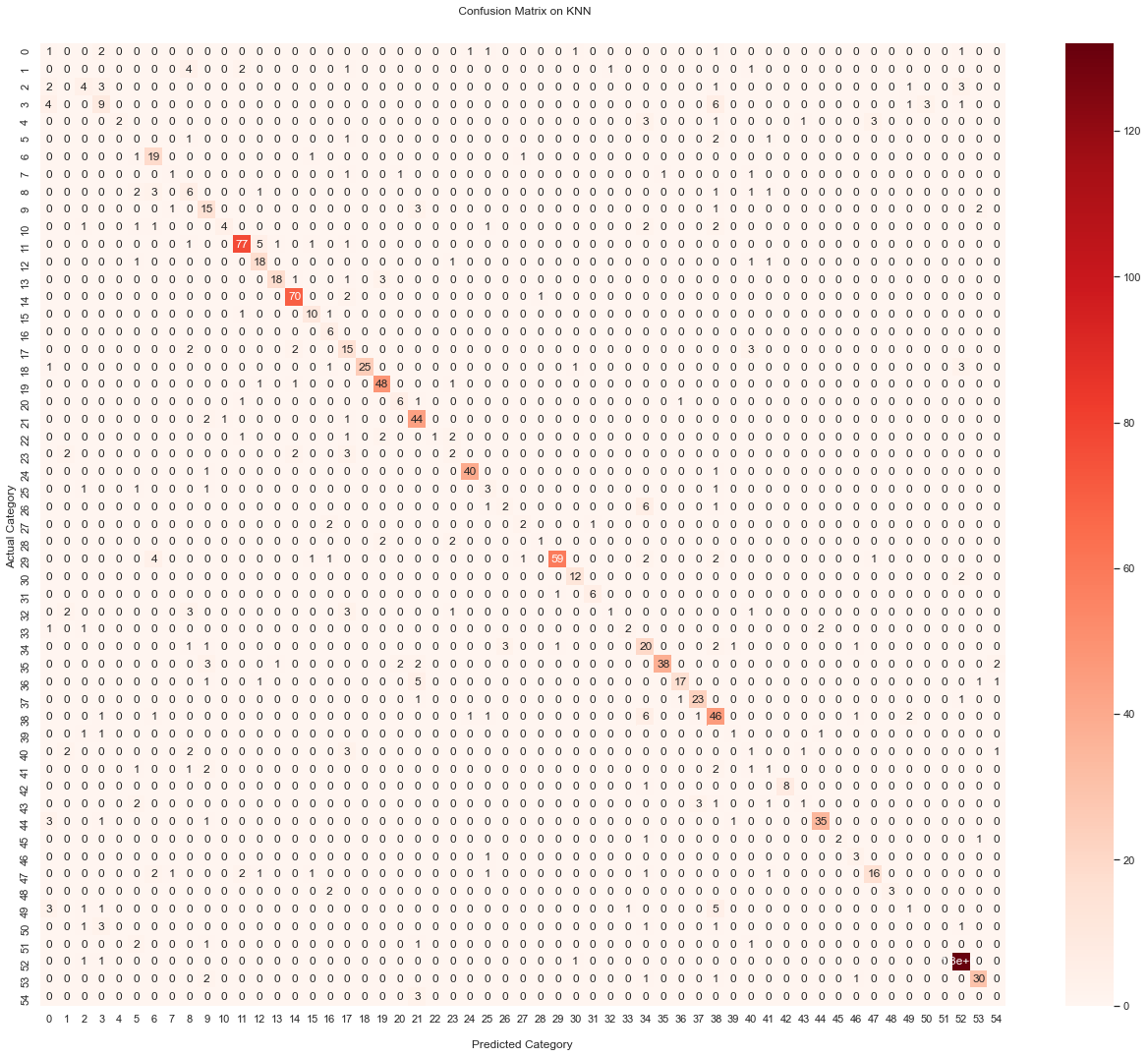


Figure 37. Confusion Matrix on Logistic Regression

A large number of decision trees are built during the training phase of the random forests or random decision forests ensemble learning approach, which is used for classification, regression, and other tasks. The class that the majority of the trees choose is the output of the random forest for classification problems. The mean or average forecast of each tree is returned for regression tasks. The tendency of decision trees to overfit their training set is corrected by random decision forests. Although they frequently outperform decision trees, gradient-enhanced trees are more accurate than random forests. However, their effectiveness may be impacted by data peculiarities. It is frequently used to solve classification and regression issues. On various samples, it constructs decision trees and uses their average for classification and majority vote for regression. The accuracy we got on the random forest classifier is 98.73%. In Fig. 38 Confusion Matrix on Random Forest is shown.

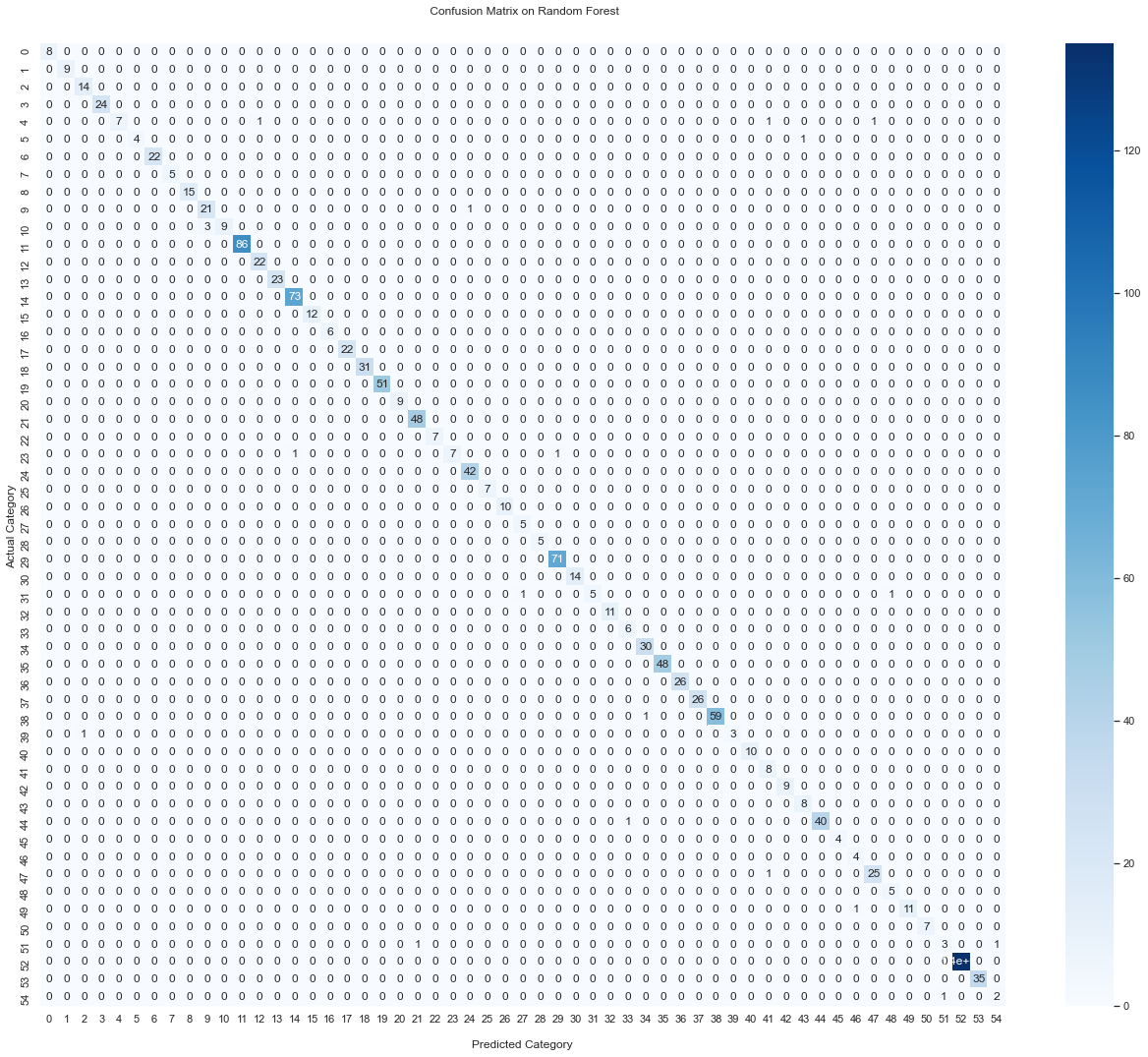


Figure 38. Confusion Matrix on Random Forest

Support vector machines (SVMs) are a group of supervised learning techniques for classifying data, performing regression analysis, and identifying outliers. Support vector machines provide the following benefits: Effective in high-dimensional spaces. Still useful in situations where the number of dimensions exceeds the number of samples. For implementing SVM, first, we Normalize the Predictor (Feature Set) for SVM training. Then separated Predictors and Outcome values from train and test sets. Scaling the Train and Test feature set using a standard scaler. We also did hyperparameter tuning using grid search and cross-validation and create the parameter grid based on the results of a random search. The RBF kernel, which is the default kernel in the sklearn SVM classification method, contains the following formula: where gamma can be manually changed and must be > 0. We set kernel “rbf” and gamma 1e-3 and 1e-4. Set the values of C from 1, 10, 100, and 1000. How much you want to prevent misclassifying each training sample is specified by the C parameter for the SVM optimization. If a smaller-margin hyperplane performs better at accurately classifying all of the training points for high values of C, the optimization will select it. A single training example's effect is determined by the gamma parameter, with low values denoting "far" and large values denoting "near." Models with lower gamma values are equally as accurate as those with higher gamma values. We set the same values for kernel “linear”. While in kernel linear, the term "linearly separable data" refers to data that can be divided into two groups using only a single straight line. Linear SVM is used to classify such data, and the classifier utilized is known as the Linear SVM classifier. When the data can be split using a single line, or when it is linearly separable, a linear kernel is utilized. It is one of the most often utilized kernels. It is often employed when a given data set has a sizable number of features. Performing CV to tune parameters for the best SVM fit is shown in Fig. 39.

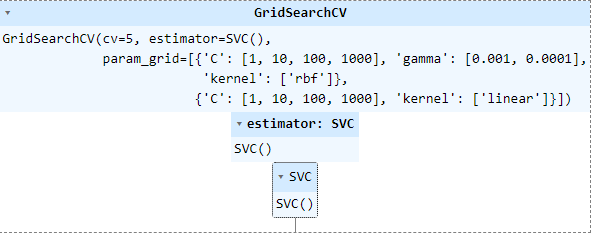


Figure 39.Setting hyperparameters for Grid Search

The best score for training data: 97.69%, setting C at 1000 on best kernel “linear” and gamma in scale. Overall training set the score for SVM is 99.91%, the testing set score for SVM is 98.94% and the accuracy using SVM is 98.94%. The confusion metric for SVM is shown in Fig. 40.

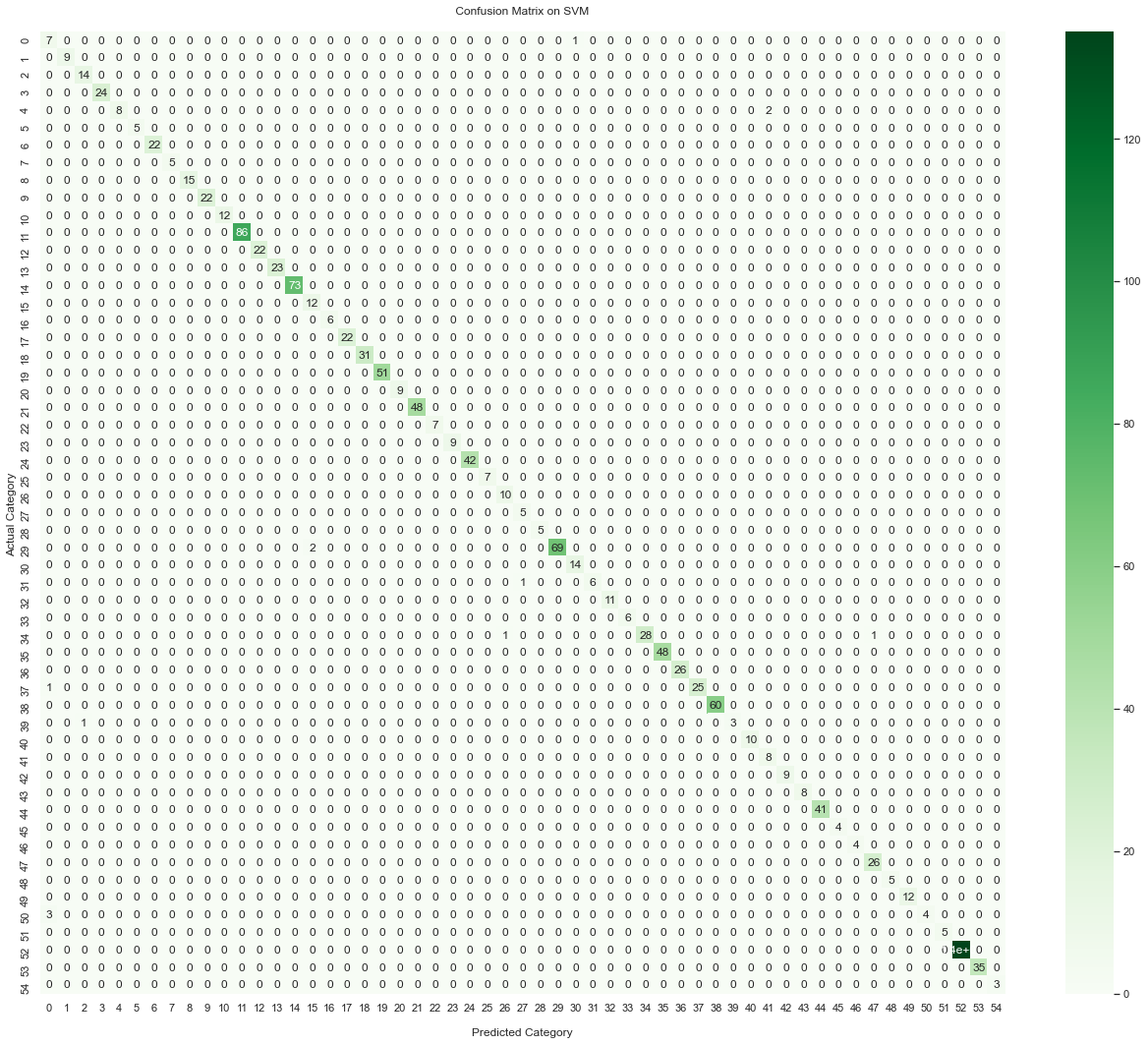


Figure 40. Confusion Matrix on SVM

The non-parametric supervised learning approach used for classification and regression applications is the decision tree. It is organized hierarchically and has a root node, branches, internal nodes, and leaf nodes. We got an accuracy of 99.91% on the Decision tree algorithm and the confusion metric of the decision tree is shown in Fig. 41.

Chart, scatter chart

Description automatically generated

Figure 41. Confusion Matrix on Decision Tree

After Implementing five machine learning algorithms we draw a comparison graph of their accuracy as shown in Fig. 42. We can see that the accuracy of Random Forest, SVM, and Decision Tree is very good on our dataset and have a very small accuracy margin between them. But the Decision Tree outperforms all other Machine Learning models and gives us the best accuracy of 99.1%.

Chart, bar chart

Description automatically generated

Figure 42. Accuracy comparison of different models

# Chapter 6 Conclusion

## Results for Xeno-Canto database

In this section, achieving sustainable results has been discussed on the metadata of bird species from the Xeno-Canto database. A machine learning-based model was used in this study to identify and classify different bird species with different features of the birds. The confusion matrix and training and testing data demonstrate the effectiveness of this model. This study serves as a further stimulus for utilizing different machine learning-based models for the analysis of bird species metadata.

## Conclusion

The significance of this research is in examining how the number of bird species classified affects categorization performance. Additionally, the goal is to show how these categorization outcomes substantially depend on the makeup of bird species subgroups. As a result, we used unbalanced subsets of bird sounds for n species, selecting the species at random from a larger dataset made up of 410 species, with n ranging from 10 to 300. To provide an accurate estimation of the performance given a specific number of bird species, we 20 times iterated the entire process (assembly of the subset, training of the classifier, and testing).

Machine learning methods, including Logistic Regression, K-Nearest Neighbors, Random Forest, Support Vector Machine, and Decision Tree, were used to categorize various bird species. These models were trained using 11 previously computed bird species attributes. As previously demonstrated, we employed several machine learning models to carry out our study, principally due to their model simplicity, reduced computing costs, and the very little data needed to train such models. An easy-to-use model that can be trained using specially created bird Meta features was what we intended to use to compare classification performance and carry out our study. In addition to thoroughly analyzing their reliance on n, we examined the classification performance using some standard metrics for classification success. Even when many distinct species were included in the datasets under analysis and utilizing comparatively less data, we saw that the classification performance was still rather excellent. This is a significant outcome considering that many contemporary methods likewise rely on machine learning models that have been trained on significantly bigger quantities of unprocessed raw data. This shows that machine learning models trained on already-computed birds' various attributes can likewise offer a reliable method for classifying bird species while also using little data and requiring little computing effort.

Regarding the robustness of the technique, we discovered that as the number of species contained in the subset rose, all indicators of classification performance exhibited a reduction in value. When considering the n-dependence of the confusion matrix and assuming the behavior of an imagined perfect classifier, it is possible to analytically explain why some of these measurements have declined.

Furthermore, we note that the makeup of the individual bird subgroups influences classification performance and classification outcomes might differ dramatically depending on the species used in the research. Because of this, it appears that broad statements about how well an algorithm performs for, say, n species of non-randomly drawn species, should not be understood as a measure of performance for all n species. Even though the species are taken from the same dataset, the categorization findings could not be generalized to another collection of n species.

# References

Alpaydin, Ethem. 2020. *Introduction to Machine Learning*.

Anon. n.d.-a. “Understanding Bird Songs: The Role for Automated Speech and Audio Processing.” Retrieved September 12, 2022 (https://projects.iq.harvard.edu/understanding\_bird\_songs).

Anon. n.d.-b. “Xeno-Canto :: Sharing Bird Sounds from around the World.” Retrieved September 12, 2022 (https://xeno-canto.org/).

Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. “Training Algorithm for Optimal Margin Classifiers.” *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory* 144–52. doi: 10.1145/130385.130401.

Burbidge, R., B. Buxton-Keynote papers, young OR12, and undefined 2001. n.d. “An Introduction to Support Vector Machines for Data Mining.” *Cc.Gatech.Edu*.

Dietterich, Thomas G., and Eun Bae Kong. 1995. “Machine Learning Bias, Statistical Bias, and Statistical Variance of Decision Tree Algorithms.”

EECS, V. Jakkula-School of, Washington State University, and undefined 2006. n.d. “Tutorial on Support Vector Machine (Svm).” *Course.Ccs.Neu.Edu*.

learning, TM Mitchell-Machine, and undefined 1997. n.d. “Artificial Neural Networks.” *Cs.Cmu.Edu*.

Lee, YJ, YR Yeh, HK Pao-National Taiwan University of, and undefined 2010. n.d. “An Introduction to Support Vector Machines.” *Jupiter.Math.Nycu.Edu.Tw*.

Magoulas, George D., and Andriana Prentza. 2001. “Machine Learning in Medical Applications.” *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 2049 LNAI:300–307. doi: 10.1007/3-540-44673-7\_19/COVER.

Myles, Anthony J., Robert N. Feudale, Yang Liu, Nathaniel A. Woody, and Steven D. Brown. 2004. “An Introduction to Decision Tree Modeling.” *Wiley Online Library* 18(6):275–85. doi: 10.1002/cem.873.

Panch, Trishan, Peter Szolovits, and Rifat Atun. 2018. “Artificial Intelligence, Machine Learning and Health Systems.” *Journal of Global Health* 8(2):20303. doi: 10.7189/JOGH.08.020303.

Podgorelec, Vili, Peter Kokol, Bruno Stiglic, and Ivan Rozman. 2002. “Decision Trees: An Overview and Their Use in Medicine.” *Journal of Medical Systems* 26(5):445–63. doi: 10.1023/A:1016409317640.

Quinlan, J. R. 1986. “Induction of Decision Trees.” *Machine Learning* 1(1):81–106. doi: 10.1007/BF00116251.

Quinlan, JR. 2014. *C4. 5: Programs for Machine Learning*.

Salzberg, SL. 1994. “C4. 5: Programs for Machine Learning by j. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993.”