Analysis of Aquaculture Data and Prediction Using Machine Learning Algorithm

Perkins Offi - 2114921

11 March 2022

1 Introduction

The Aquaculture industry in Scotland is one of growing importance to the Scottish economy, providing valuable jobs and income. The fishes are grown in cages either in lakes or in the sea. Being an industry that directly affects the environment, there has been an increase in the monitoring of producing farms activites as well as their records on incidents in the farm. The industry is regulated with three main target areas; promoting food safety, compliance with legislation and sustainability.

This project involves the use of two datasets; escapes.csv which contains records of fish escapes into the wild, and analysis.csv which contains the results of water analysis using a number of components. Both datasets have a level of alignment. The goal of this project is to perform analysis on a merged version of the datasets using a learning model to predict the specie of fishes that escape from a fish farm in an escape incident.

Section 2 describes the two datasets including every data preparation mechanism that was used for the proper formatting of the datasets. Section 3 includes the steps taken to prepare the datasets for merging, the merging of the datasets and creation of a .csv file containing the merged dataset. Section 4 contains additional exploratory data analysis carried out on the merged dataset. Section 5 contains the supervised learning experiment and results while Section 6 presents the conclusion drawn from the experiment.

```r
#Loading all required libraries
library(stringr)
library(dplyr)
library(Hmisc)
library(lubridate)
library(corrplot)
library(RColorBrewer)
library(randomForest)
library(MASS)
library(leaps)
library(ISLR)
library(caret)
library(xtable)
```

2 Exploratory Data Analysis

A complete understanding of the datasets is required in order to carry out analysis. This section describes the datasets, provides an initial inspection of the features to establish relationships and also data preprocessing for optimal performance of the learning model.

2.1 Data loading

The datasets which are in csv format was loaded using the read.csv function and saved in two variables called 'data01' and 'data02'.

```
#Set working directories
setwd("C:/Users/perki/Desktop/RGU/Data Science/Coursework/CMM 535 Coursework")

#Load datasets
data01 = read.csv("escapes.csv", header = T, stringsAsFactors = T)
data02 = read.csv("analysis.csv", header = T, stringsAsFactors = T)
```

2.2 Data description

The escapes dataset (data01) is made up of 38 columns and 357 rows (instances) that outline information recorded for fish escape incidents. The analysis dataset (data02) is made up of 9 columns, 6 numerical and 3 categorical. It also contains 351 rows (instances) that outline the results of water analysis carried out at the fish farms.

2.2.1 Attribute Information - data01

- Escape.ID: Unique ID for an escape incident.
- Operator.at.Time.of.Escape: Company operating the fish farm at the time of the escape.
- Escape.Water.Type: Type of water in which incident occurred (f-fresh water, s-sea water).
- Escape.Start.Date: Date in DD-Mon-YY when the escape incident started.
- Escape.Start.Time: Time in 24-hour format when the escape incident started.
- Escape.End.Time: Date in DD-Mon-YY in when the escape incident ended (Incorrect column name).
- Escape.Grid.Reference: Grid reference of the eacape incident.
- Escaped.Species: Name of fish specie that escaped during the incident.
- Stage: Stage of fish growth.
- Age: Age of fish in months.
- Average.Weight: Average weight of escaped fish in grams and kilograms.
- Initial.Date.of.Escape: Initial notification date of escape incident.
- Initial.Number.Escaped: Initial number of fish that escaped.
- Initial.Escape.Reason: Initial cause of the escape incident.
- Final.Date.of.Escape: Final notification date of escape incident.
- Final.Number.Escaped: Final number of fish that escaped.
- Final.Number.Recovered: Number of escaped fish that was recovered.
- Final.Escape.Reason: Final cause of the escape incident.
- Marine.Scotland.Site.ID: Marine Scotland unique ID for the fish farm.

- Date.Registered: Registration date of the site.
- Site.Name: Name of the fish farm site.
- National.Grid.Reference: National Grid Reference of the site.
- Local.Authority: Local Authority at the site location.
- Producing.in.Last.3.Years: Whether or not the farm has been producing in the last three years.
- Site.Address.1: Line 1 of the site address.
- Site.Address.2: Line 2 of the site address.
- Site.Address.3: Line 3 of the site address.
- Site.Post.Code: Postal code of the site address.
- Site.Contact.Number: Contact phone number of the site.
- Aquaculture.Type: Type of aquaculture activity carried out at the site.
- Water.Type: Type of water in which the fish are being grown.
- Health.Surveillance: Frequency of Health Survellance carried out at the site.
- Easting: Longitude component of the site co-ordinate.
- Northing: Latitude component of the site co-ordinate.
- MS.Management.Area: Marine scotland management area.
- Region: Region where the site is located.
- Operator: Current operator of the site.
- Species:Species of fish produced at the site.

2.2.2 Attribute Information - data02

- year: The year when the water analysis was carried out.
- month: The month of the year when the analysis was carried out.
- Site.Name: The name of the site from which the analysed water sample was taken.
- c2: Component 2.
- c3: Component 3.
- c4: Component 4.
- c5: Component 5.
- c6: Component 6.
- c7: Component 7.

2.3 Data preparation and cleaning

In this section, the datasets were preprocessed for appropriate functionality. Preparation carried out includes determining and removing variables not needed for analysis, determining columns of the wrong type and converting them to the correct formats, etc.

2.3.1 Removing non-informative variables

In the first dataset (data01), Escape.Water.Type, Escape.Grid.Reference, Stage, Escape.Start.Time, Escape.End.Time, Initial.Date.of.Escape, Final.Date.of.Escape, Initial.Number.Escaped, Initial.Escape.Reason, Date.Registered, National.Grid.Reference, Local.Authority, Site.Address.1, Site.Address.2, Site.Address.3, Site.Contact.Number, Aquaculture.Type, Easting, Northing, MS.Management.Area, Region, Operator and Species columns were observed to be non-informative in relation to the analysis to be carried out and thus removed from the datset leaving 15 variables.

In the second dataset (data02), all variables were observed to be relevant and thus no variable was removed.

```
#Making copy of datasets
data1 = data01
data2 = data02

#Removing non-informative columns from data1
data1$Escape.Water.Type = NULL
data1$Escape.Grid.Reference = NULL
data1$Escape.Start.Time = NULL
data1$Escape.End.Time = NULL
data1$Stage = NULL
data1$Initial.Date.of.Escape = NULL
data1$Final.Date.of.Escape = NULL
data1$Initial.Number.Escaped = NULL
data1$Initial.Escape.Reason = NULL
data1$Date.Registered = NULL
data1$National.Grid.Reference = NULL
data1$Local.Authority = NULL
data1$Site.Address.1 = NULL
data1$Site.Address.2 = NULL
data1$Site.Address.3 = NULL
data1$Site.Contact.Number = NULL
data1$Aquaculture.Type = NULL
data1$Easting = NULL
data1$Northing = NULL
data1$MS.Management.Area = NULL
data1$Region = NULL
data1$Operator = NULL
data1$Species = NULL
```

2.3.2 Cleaning the 'Age' variable

It was observed that the instances under the age variable are supposed to be numbers in months. For analysis to be carried out using the age column, the data type was converted from factor to numeric and the column name changed to 'Age.in.Months'. There were also quite a number of columns with incorrect or missing data which were replaced by the mean of the age distribution. The mean age was used because it is a good representation of what the missing value could be given the sample distribution.

```
#Changing the variable name from Age to Age.in.Months
names(data1)[names(data1) == 'Age'] = 'Age.in.Months'

#Fixing incorrect values in the Age column.
data1["Age.in.Months"][data1["Age.in.Months"] == "2 yrs sw"] = "24 months"
```

```
data1["Age.in.Months"][data1["Age.in.Months"] == "7-8 months"] = "7 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "9-10 month"] = "9 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "50g"] = "9 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "13mths at "] = "13 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "3wks sw"] = "1 month"
data1["Age.in.Months"][data1["Age.in.Months"] == "3 yrs old"] = "36 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "14mths"] = "14 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "24mths sw "] = "24 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "3-4 months"] = "3 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "30 & 42 mo"] = "36 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "6 weeks at"] = "1 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "16 - 17 mo"] = "16 months"
data1["Age.in.Months"][data1["Age.in.Months"] == "2 - 2.5 kg"] = "15 months"

#Removing the 'months' string from the instances and converting observations to numeric data type
data1$Age.in.Months = as.numeric(gsub('[a-zA-Z]', '', data1$Age.in.Months))

#Changing incorrect values to NA in the Age column.
data1["Age.in.Months"][data1["Age.in.Months"] == 20120] = NA
data1["Age.in.Months"][data1["Age.in.Months"] == 1999] = NA
data1["Age.in.Months"][data1["Age.in.Months"] == 2000] = NA

#Replacing NA values with mean value of the Age column.
data1$Age.in.Months = impute((data1$Age.in.Months), mean)
data1$Age.in.Months = round(data1$Age.in.Months, digits = 0)
data1$Age.in.Months = as.numeric(data1$Age.in.Months)
```

2.3.3 Cleaning the 'Average.Weight' variable

It was observed that the instances under the Average.Weight variable were weight measurements in grams and kilograms. For analysis to be carried out using the column, the data type was converted from factor to character and then to numeric. A measurement unit unification was carried out on the instances, making sure all measurements are in kilograms and the column name was changed to 'Average.Weight.Kg'. There were also quite a number of columns with incorrect or missing data which were also replaced by the mean weight value. The mean weight was used because it is a good representation of what the missing value could be given the sample distribution.

```
#Changing the data type from Factor to Character.
data1$Average.Weight = as.character(data1$Average.Weight)

#Fixing incorrect values in the Agerage.Weight column.
data1["Average.Weight"][data1["Average.Weight"] == "9 months"] = "50g"
data1["Average.Weight"][data1["Average.Weight"] == "15 months (in sw)"] = "2.25 kg"
data1["Average.Weight"][data1["Average.Weight"] == "350 - 400 g"] = "375 g"
data1["Average.Weight"][data1["Average.Weight"] == "1.8-2.0 kg"] = "1.9 kg"
data1["Average.Weight"][data1["Average.Weight"] == "17.5 kg"] = "17.5 g"
data1["Average.Weight"][data1["Average.Weight"] == "750g - 2 kg"] = "1.38 kg"
data1["Average.Weight"][data1["Average.Weight"] == "1 - 2.3 kg"] = "1.65 kg"
data1["Average.Weight"][data1["Average.Weight"] == "2-4kg"] = "3 kg"
data1["Average.Weight"][data1["Average.Weight"] == "4-5 kg"] = "4.5 kg"
data1["Average.Weight"][data1["Average.Weight"] == "150-200g"] = "175 g"
data1["Average.Weight"][data1["Average.Weight"] == "0.25 - 2 kg"] = "1.13 kg"
data1["Average.Weight"][data1["Average.Weight"] == "20-25g"] = "22.5 g"
```

```r
data1["Average.Weight"][data1["Average.Weight"] == "450 grams"] = "450 g"
data1["Average.Weight"][data1["Average.Weight"] == "~60g"] = "60 g"
data1["Average.Weight"][data1["Average.Weight"] == "90grams"] = "90 g"
data1["Average.Weight"][data1["Average.Weight"] == "~1.0 kg"] = "1 kg"
data1["Average.Weight"][data1["Average.Weight"] == "8-10 kg"] = "9 kg"
data1["Average.Weight"][data1["Average.Weight"] == "1 lb"] = "0.45 kg"
data1["Average.Weight"][data1["Average.Weight"] == "250-300g"] = "275g"
data1["Average.Weight"][data1["Average.Weight"] == "170-220g"] = "195g"
data1["Average.Weight"][data1["Average.Weight"] == "3.2 kilos"] = "3.2kg"
data1["Average.Weight"][data1["Average.Weight"] == "30-40g"] = "35g"
data1["Average.Weight"][data1["Average.Weight"] == "6.5 & 12 kg"] = "9kg"
data1["Average.Weight"][data1["Average.Weight"] == "500-900g"] = "700g"
data1["Average.Weight"][data1["Average.Weight"] == "150-250 g"] = "200g"
data1["Average.Weight"][data1["Average.Weight"] == "70-140 g"] = "105g"
data1["Average.Weight"][data1["Average.Weight"] == "70-140 g"] = "105g"

#Removing spaces between numbers and units in the Average.Weight column
data1$Average.Weight = gsub('\\s+', '', data1$Average.Weight)

#Changing incorrect values to NA in the Average.Weight column.
data1$Average.Weight[data1$Average.Weight == "unknown" ] = NA

#Creating a function to unify measuresment units
WeightUnit <- function (svalue){
   if(grepl("kg", svalue)){
      kg <- str_locate(svalue,"kg")[1,1]
      num = as.numeric(substr(svalue, 0,kg-1))
      return(as.character(num * 1000))
   } else {return(as.character(svalue))}
}

#Applying the function to the whole column
data1$Average.Weight = sapply(as.character(data1$Average.Weight),WeightUnit)

#Converting observations to numeric data type
data1$Average.Weight = as.numeric(gsub('[a-zA-Z]', '', data1$Average.Weight))

#Converting all observations from grams to kilograms
data1["Average.Weight"][data1["Average.Weight"] == 3.413] = 3413
data1["Average.Weight"][data1["Average.Weight"] == 3.200] = 3200
data1["Average.Weight"][data1["Average.Weight"] == 1.626] = 1626
data1["Average.Weight"][data1["Average.Weight"] == 22.5] = 22500
data1["Average.Weight"][data1["Average.Weight"] == 1.1] = 1100
data1["Average.Weight"][data1["Average.Weight"] == 2.200] = 2200
data1["Average.Weight"][data1["Average.Weight"] == 5.325] = 5325
data1["Average.Weight"][data1["Average.Weight"] == 5.325] = 5325

data1$Average.Weight = data1$Average.Weight / 1000

#Changing the variable name from Average.Weight to Average.Weight.Kg
names(data1)[names(data1) == 'Average.Weight'] = 'Average.Weight.Kg'

#Replacing NA values with mean value of the Average.Weight.Kg column.
```

```
data1$Average.Weight.Kg = impute((data1$Average.Weight.Kg), mean)
data1$Average.Weight.Kg = round(data1$Average.Weight.Kg, digits = 2)
data1$Average.Weight.Kg = as.numeric(data1$Average.Weight.Kg)
```

2.3.4 Cleaning the 'Final Number Escaped' variable

The Final.Number.Escaped variable was observed to be of the factor type. It was converted to character type for formatting and then to numeric. There were quite a number of incorrect observations which were corrected and all NA values replaced with 0. This is because it was assumed that in those instances, there were no recorded number of escaped fish.

```
#Changing the data type from Factor to Character.
data1$Final.Number.Escaped = as.character(data1$Final.Number.Escaped)

#Fixing incorrect values.
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "~200"] = "200"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "ca. 150"] = "150"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "0 (160,000 dead)"] = "0"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "100-200"] = "150"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "0 (13 dead)"] = "0"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == ">500 <1050"] = "775"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "~2,500"] = "2500"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "30-40"] = "35"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "est - 4000"] = "4000"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "70-80,000"] = "75000"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "5000-10000"] = "7500"
data1["Final.Number.Escaped"][data1["Final.Number.Escaped"] == "20 (estimate)"] = "20"

#Removing all non-numeric characters
data1$Final.Number.Escaped = gsub(',', '', data1$Final.Number.Escaped)

#Converting observations to numeric data type
data1$Final.Number.Escaped = as.numeric(gsub('[a-zA-Z]', '', data1$Final.Number.Escaped))

#Replacing NA values with 0.
data1$Final.Number.Escaped [is.na(data1$Final.Number.Escaped)] <- 0
```

2.3.5 Cleaning the 'Final Number Recovered' variable

The Final.Number.Recovered variable was observed to be of the factor type. It was converted to character type for formatting and then to numeric. There were quite a number of incorrect observations which were corrected and all NA values replaced with 0. This is because it was assumed that in those instances, none of the escaped fishes were recovered.

```
data1$Final.Number.Recovered = as.character(data1$Final.Number.Recovered)

#Fixing incorrect values.
data1["Final.Number.Recovered"][data1["Final.Number.Recovered"] == "80 - 100"] = "90"
data1["Final.Number.Recovered"][data1["Final.Number.Recovered"] == "300+"] = "300"
data1["Final.Number.Recovered"][data1["Final.Number.Recovered"] == "1,000 ongoing"] = "1000"
data1["Final.Number.Recovered"][data1["Final.Number.Recovered"] == "500 (dead)"] = "500"
```

```r
data1["Final.Number.Recovered"][data1["Final.Number.Recovered"] == "6578 live,1709 dead"] = "8287"
data1["Final.Number.Recovered"][data1["Final.Number.Recovered"] == "15 live"] = "15"

#Removing all non-numeric characters
data1$Final.Number.Recovered = gsub(',', '', data1$Final.Number.Recovered)

#Converting observations to numeric data type
data1$Final.Number.Recovered = as.numeric(gsub('[a-zA-Z]', '', data1$Final.Number.Recovered))

#Replacing NA values with 0.
data1$Final.Number.Recovered [is.na(data1$Final.Number.Recovered)] = 0
```

2.4 Cleaning the second dataset (data2)

The year and month variables in data2 were of the integer type instead of factors. These variables were converted into factors. The column names were also changed from 'year' and 'month' to 'Year' and 'Month' respectively for future analysis purposes.

```r
#Changing the variable names
names(data2)[names(data2) == 'year'] = 'Year'
names(data2)[names(data2) == 'month'] = 'Month'

#Converting Analysis.Year and Analysis.Month columns to factor data type
data2$Year = as.factor(data2$Year)
data2$Month = as.factor(data2$Month)
```

3 Merging Datasets

In the section, the two datasets (data1 and data2) were integrated to form a merged dataset called 'escapesPlus'. For merging to be possible, both datasets need to have some common variables. Data extraction and new column creation were carried out on data1 to make sufficient key variables available. The merging was done using the common key variables in both datasets: site name, month and year. The merged dataset was then saved in a file called 'escapesPlus.csv'.

3.1 Preparation of datasets for merging

The common key variable 'Site.Name' for merging the datasets had different letter cases in both datasets. In data1, the instances in the Site.Name column are all in lower case while in data2, the instances are in title case (The first letter of each word is in capitals). This was corrected leaving all instances in title case. The data type of Escape.ID was changed from numeric to factor. The data type for Escape.Start.Date was changed from factor to date. Two new columns called 'Year' and 'Month' was then created which contains only the year and month respectively, extracted from the Escape.Start.Date.

```r
#Changing case type of Site.Name variable in both datasets
data1$Site.Name = str_to_title(data1$Site.Name)
data2$Site.Name = str_to_title(data2$Site.Name)

#Changing the data type of Escape.ID to factor
data1$Escape.ID = as.factor(data1$Escape.ID)
```

```r
#Changing the data type of Escape.Start.Date to date
data1$Escape.Start.Date = as.character(data1$Escape.Start.Date)
data1$Escape.Start.Date = as.Date(data1$Escape.Start.Date, "%d-%b-%y")

#Creating a new columns called 'Year' and 'Month' in data1
data1$Year = year(data1$Escape.Start.Date)
data1$Year = as.factor(data1$Year)
data1$Month = month(data1$Escape.Start.Date)
data1$Month = as.factor(data1$Month)
```

3.2 Merge

The datasets were merged using the merge function. The common key variables used for the merge were Site.Name, Month and Year. The merged dataset had 357 rows and 23 columns. The rows from data1 with no corresponding rows to merge with from data2 were then removed to enable smooth analysis, thus reducing the number of rows to 351. The merged dataset was then saved as a .csv file.

```r
#Merging the datasets
escapesPlus = merge(data1, data2, by=c("Year", "Site.Name", "Month"), all = T)
escapesPlus = escapesPlus[!duplicated(escapesPlus$Escape.ID), ]


#Replacing NA values
escapesPlus$Final.Escape.Reason [is.na(escapesPlus$Final.Escape.Reason)] <- "unknown - unk"
escapesPlus$Site.Post.Code [is.na(escapesPlus$Site.Post.Code)] <- "n/a"
escapesPlus$Health.Surveillance = as.character(escapesPlus$Health.Surveillance)
escapesPlus$Health.Surveillance [is.na(escapesPlus$Health.Surveillance)] <- "not applicable"

#Removing rows with no values for water analysis results
escapesPlus = escapesPlus[complete.cases(escapesPlus), ]

#Saving merged dataset to file called escapesPlus.csv
write.csv(escapesPlus,"C:/Users/perki/Desktop/RGU/Data Science/Coursework/CMM 535 Coursework\\escapesPl
```

4 Exploratory Data Analysis of Merged Dataset

In this section, additional exploratory data analysis was carried out on the merged dataset. Visualizations and summary statistics are used to better understand the relationships between variables and also obtain interesting information.

4.1 Correlation of numeric variables

Figure 1 is a correlation plot of the numeric variables in the dataset. It was observed that there is generally little or no corellation between the numeric variables.The highest correlation recorded was between the contaminants; there was a strong positive linear correlation between c2 and c3, c2 and c4, c2 and c7, c3 and c4, c3 and c7, and c4 and c7. There is a weak positive linear correlation between the Age of the fish and their average weight, this would be expected as older fish tend to weigh more. However, it is worthy of note that the corr function performs linear analysis on the variables and the results may be different for other models.

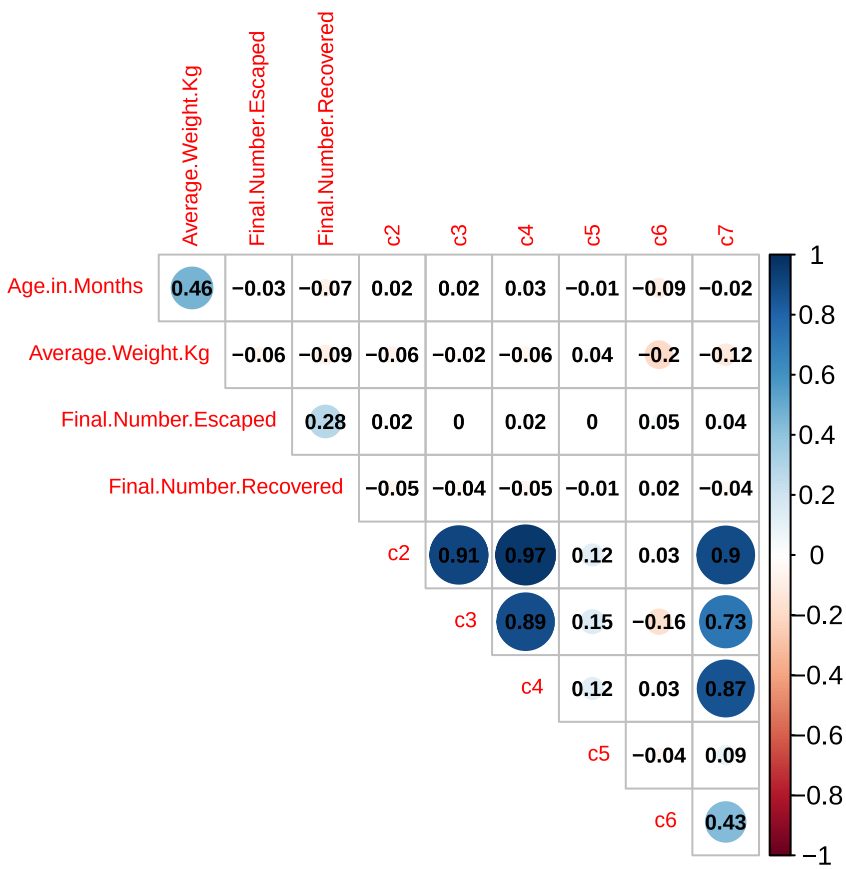

Figure 1: Correlation plot of numeric variables

4.2 Looking at categorical variables

A frequency count of the escaped fish species showed that, out of the 351 escape incidents, 273 involved the Atlantic Salmon specie, 72 involved the Rainbow Trout specie, while the other species had extremely low number of occurrences. A bar plot of this frequency count was also carried out (Figure 2).

```
#Frequency count of Escaped.Species instances.
summary(escapesPlus$Escaped.Species)
```

```
##          atlantic salmon brown trout and sea trout                       cod
##                      273                         1                         1
##                  halibut                lumpsucker             rainbow trout
##                        2                         1                        72
##                   wrasse
##                        1
```

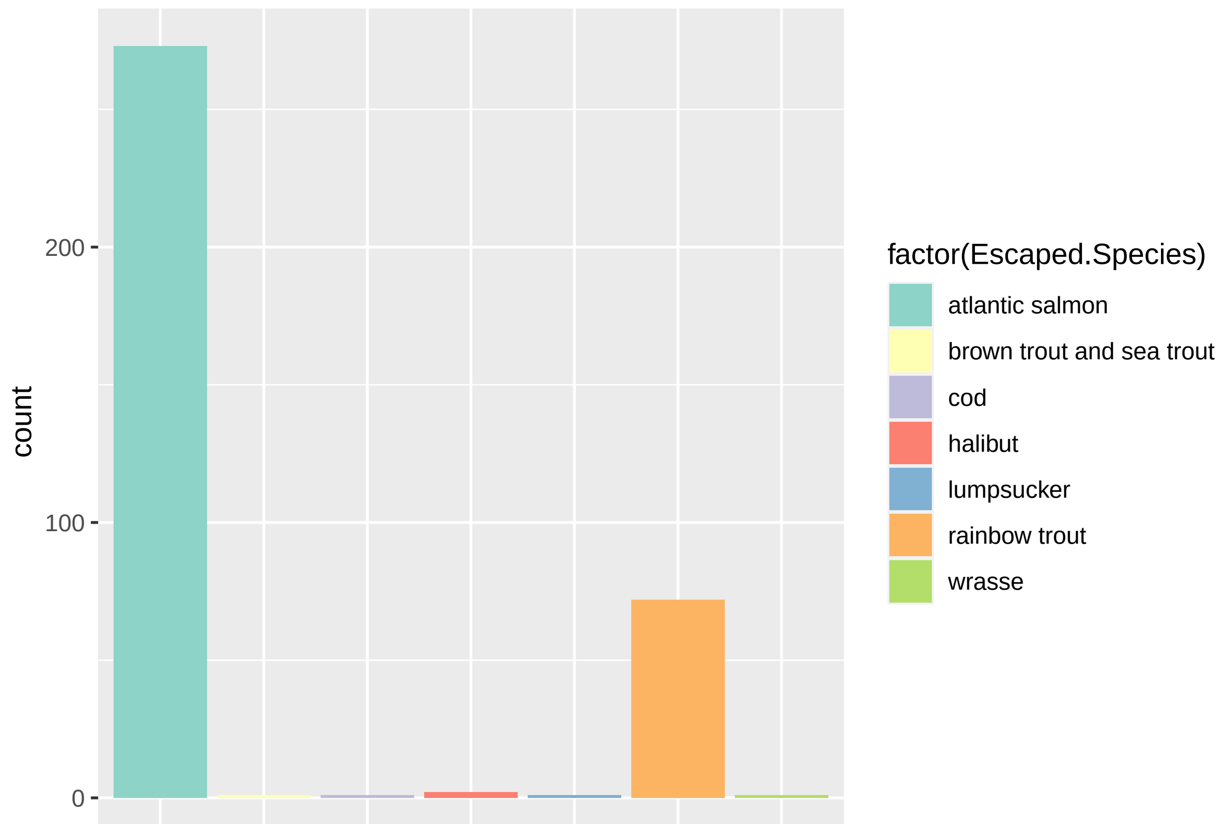

Figure 2: Frequency of Escaped Species

A frequency count was also carried out to find out the most frequent reason for escape incidents. It was observed that predators accounted for the most escape incidents (71), followed by holes in nets (57), human error (49) and weather (43). Other reasons had very low frequencies.

```
#Frequency count of Final.Escape.Reason instances.
summary(escapesPlus$Final.Escape.Reason)
```

```
##                        chafe/ snag - cha                   equipment damage - eqd
##                                        3                                       19
##           equipment wear and tear - eqw                          flooding - fld
##                                        7                                        5
##                        hole in net - hol                       human error - hum
##                                       57                                       49
##          inappropriate equipment - eqi                    mooring failure - moo
##                                        2                                        2
## net failure (not including hole) - net          no actual escape of fish - nes
##                                        2                                       54
##                              other - oth                       pen failure - pen
##                                        2                                        1
##                           predator - prd                    screen failure - scr
##                                       71                                        4
##            transfer pipe failure - trp                           unknown - unk
##                                        7                                       15
##                          vandalism - van                           weather - wth
##                                        8                                       43
```

Another frequency count was done to observe how many escape incidents have occurred in each water type. It was observed that a majority of the incidents occured in seawater (267), compared to fresh water(83).

```r
#Frequency count of Water.Type instances.
summary(escapesPlus$Water.Type)
```

```
##              freshwater freshwater and seawater                seawater
##                      83                       1                     267
```

A frequency count of escape incidents that occurred in specific months of the year was done to find out if there was any relationship between time of the year and fish escape incidents. The incident counts were generally evenly distributed across the months of the year with only January having a relatively higher number of occurences than the others. A bar plot was made to show the distribution (Figure 3).

```r
#Frequency count of Month instances.
summary(escapesPlus$Month)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12
## 46 21 30 24 34 23 21 30 33 27 32 30
```

4.2.1 Relationship between categorical and numeric variables

The dataset was further explored to investigate the relationships between categorical variables, as well as between categorical and numeric variables. The relationship between the species of the escaped fishes and the water type was investigated. The Atlantic Salmon specie was observed to have been involved in more escape incidents in sea water than in fresh water, while the Rainbow Trout specie was involved in more escape incidents in fresh water than in sea water.

```r
#Escaped.Species vs Water.Type
ESWT = xtabs(~ Escaped.Species+Water.Type, data=escapesPlus)
ESWT
```

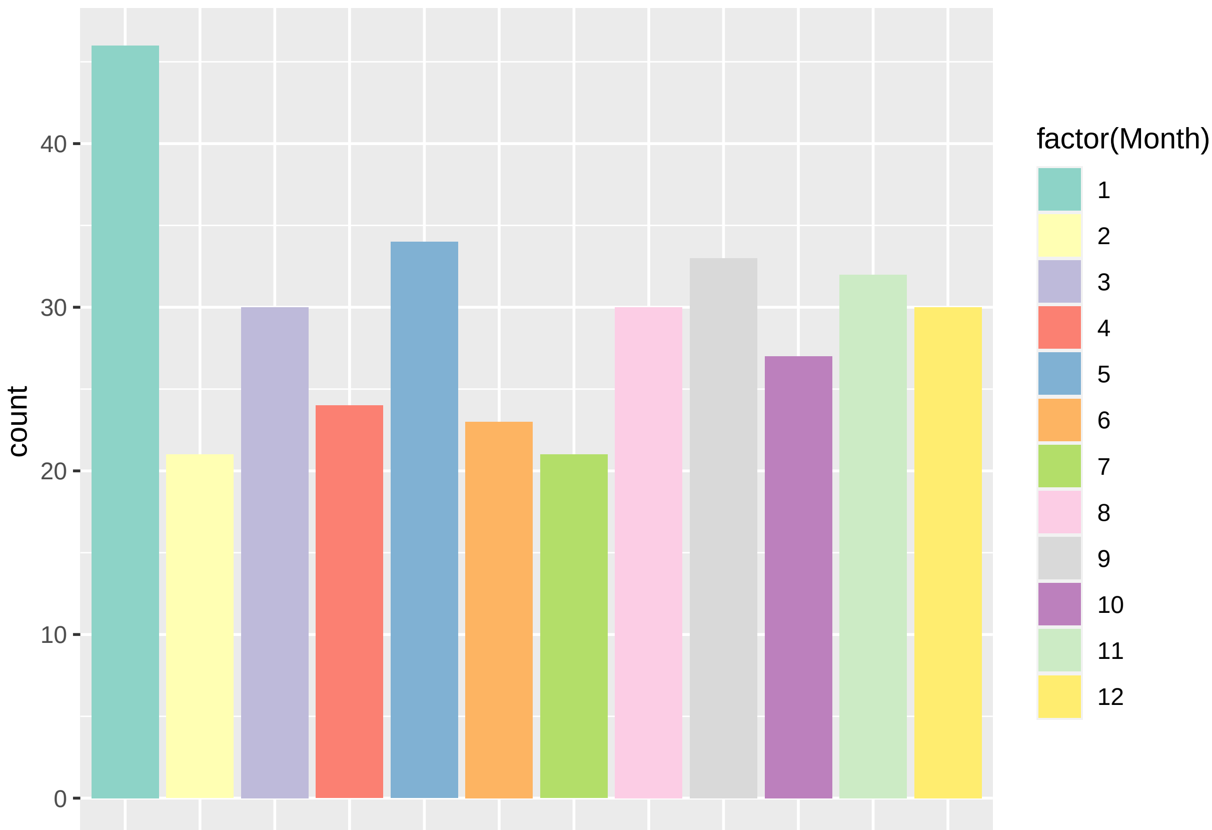

Figure 3: Frequency of Escape Incidents Per Month

```
##                                Water.Type
## Escaped.Species              freshwater freshwater and seawater seawater
##   atlantic salmon                    35                       1      237
##   brown trout and sea trout           0                       0        1
##   cod                                 0                       0        1
##   halibut                             0                       0        2
##   lumpsucker                          0                       0        1
##   rainbow trout                      48                       0       24
##   wrasse                              0                       0        1
```

The relationship between the Escaped Species and the final number of escaped fish was also investigated. A two-way table and point plot (Figure 4) was used to show the relationship. The highest number of escaped fishes was observed to be in the Atlantic Salmon, while the second highest was the Rainbow Trout. Other species had relatively low occurrences.

```
#Number of escaped fish per specie
g = aggregate(escapesPlus$Final.Number.Escaped,
              by=list(Escaped.Specie=escapesPlus$Escaped.Species), FUN=sum)

colnames(g) <- c("Specie of Fish", "Total Number Escaped")

g[order(g$`Total Number Escaped`, decreasing = T),]
```

```
##              Specie of Fish Total Number Escaped
## 1           atlantic salmon              3175077
## 6             rainbow trout               309747
## 4                   halibut                19187
## 2 brown trout and sea trout                18000
## 3                       cod                15800
## 7                    wrasse                  493
## 5                lumpsucker                  283
```

An investigation was also carried out to find out the relationship between escape reason and the number of escaped fishes. It was observed that the highest number of escaped fish was caused by the weather, while holes in nets and predators make up the other two in the top reasons with the highest number of escaped fish.

```
#Number of escaped fish per reason for escape
a = aggregate(escapesPlus$Final.Number.Escaped, by=list(Reason.For.Escape=escapesPlus$Final.Escape.Reas

colnames(a) <- c("Reason For Escape", "Total Number Escaped")

a[order(a$`Total Number Escaped`, decreasing = T),]
```

```
##                          Reason For Escape Total Number Escaped
## 18                      weather - wth              1953534
## 5                   hole in net - hol               417766
## 13                     predator - prd               328879
## 8                mooring failure - moo              203403
## 2               equipment damage - eqd              134282
## 6                   human error - hum               107982
## 4                      flooding - fld               106917
```

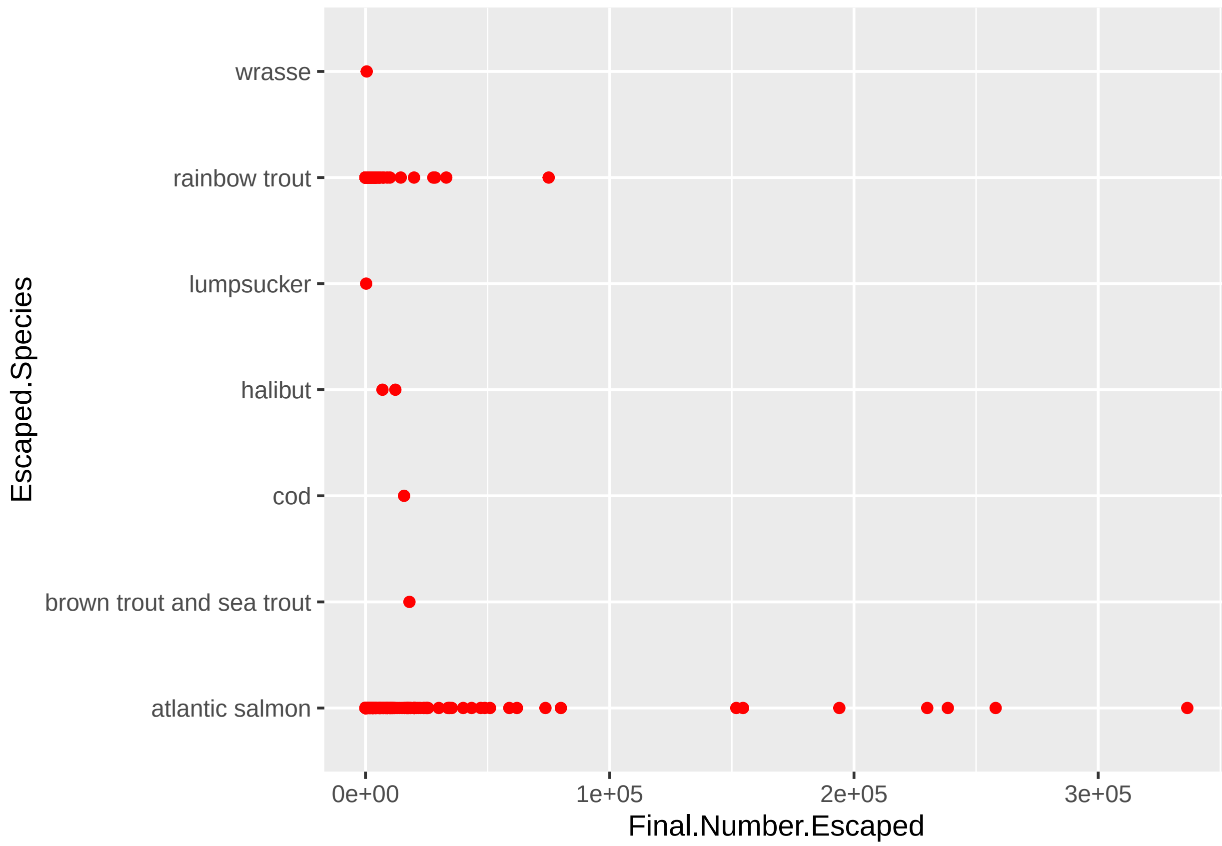

Figure 4: Number of Escaped Fishes Per Specie

```
## 16                           unknown - unk                 91251
## 12                       pen failure - pen                 73684
## 17                         vandalism - van                 36946
## 3            equipment wear and tear - eqw                 32146
## 9  net failure (not including hole) - net                 24020
## 15             transfer pipe failure - trp                  9237
## 1                       chafe/ snag - cha                  8163
## 7            inappropriate equipment - eqi                  7172
## 14                    screen failure - scr                  3165
## 11                             other - oth                    40
## 10          no actual escape of fish - nes                     0
```

An investigation to observe the relationship between level of health surveillance at fish farms and the number of escaped fishes recorded was also done. As expected, it was observed that the farms with low health surveillance recorded the most number of escaped fishes, while farms with high health surveillance level recorded the least number of escaped fishes.

```
#Number of escaped fish per level of health surveillance
ag = aggregate(escapesPlus$Final.Number.Escaped, by=list(Health.Surveillance.Level=escapesPlus$Health.Su

colnames(ag) <- c("Health Surveillance Level", "Total Number Escaped")

ag[order(ag$`Total Number Escaped`, decreasing = T),]
```

```
##   Health Surveillance Level Total Number Escaped
## 2                       low              1728520
## 3                    medium              1303174
## 1                      high               343607
## 4            not applicable               163286
```

The relationship between the total number of escaped fish and the water type was also explored. The most escaped fishes have been recorded in sea water compared to fresh water.

```
#Number of escaped fish per water type
agg = aggregate(escapesPlus$Final.Number.Escaped, by=list(Water.Type=escapesPlus$Water.Type),
                FUN=sum)

colnames(agg) <- c("Water Type", "Total Number Escaped")

agg[order(agg$`Total Number Escaped`, decreasing = T),]
```

```
##                Water Type Total Number Escaped
## 3                seawater              2778099
## 1              freshwater               759998
## 2 freshwater and seawater                  490
```

4.2.2 Summary statistics of dataset

The internal structure of the escapesPlus dataset is shown below. Of the 23 variables, 10 are factors, 2 are characters, 1 is a date, while the rest are numeric. The target variable, Final.Number.Escaped, is a float type.

```
## 'data.frame':    351 obs. of  23 variables:
##  $ Year                        : Factor w/ 25 levels "1995","1997",..: 3 3 3 3 3 4 4 4 4 4 ...
##  $ Site.Name                   : chr  "Aird" "Ardyne" "Ardyne" "Camas An Eilean" ...
##  $ Month                       : Factor w/ 12 levels "1","2","3","4",..: 12 7 9 12 5 9 1 12 12 1 ...
##  $ Escape.ID                   : Factor w/ 357 levels "2000001","2000023",..: 234 232 233 235 231 242 2
##  $ Operator.at.Time.of.Escape: Factor w/ 79 levels "abbey st. bathans trout farm",..: 27 54 52 27 11
##  $ Escape.Start.Date           : Date, format: "1998-12-15" "1998-07-01" ...
##  $ Escaped.Species             : Factor w/ 7 levels "atlantic salmon",..: 1 1 1 1 6 1 1 1 6 6 ...
##  $ Age.in.Months               : num  15 15 15 15 15 15 15 15 15 15 ...
##  $ Average.Weight.Kg           : num  2.16 2.16 3.1 2.16 2.16 2.16 1 2.5 0.2 2.16 ...
##  $ Final.Number.Escaped        : num  10000 30000 10000 17000 0 6000 0 4000 0 2000 ...
##  $ Final.Number.Recovered      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Final.Escape.Reason         : Factor w/ 18 levels "chafe/ snag - cha",..: 2 5 6 3 16 6 18 5 18 18 .
##  $ Marine.Scotland.Site.ID     : Factor w/ 190 levels "fs0007","fs0011",..: 83 78 78 6 89 78 145 13 23
##  $ Producing.in.Last.3.Years : Factor w/ 2 levels "no","yes": 2 2 2 1 2 2 1 2 2 2 ...
##  $ Site.Post.Code              : Factor w/ 93 levels "dg10 9lg","dg13 0aw",..: 40 51 51 49 72 51 91 92
##  $ Water.Type                  : Factor w/ 3 levels "freshwater","freshwater and seawater",..: 3 3 3 3
##  $ Health.Surveillance         : chr  "medium" "medium" "medium" "medium" ...
##  $ c2                          : num  1.31 1.838 0.977 2.337 2.003 ...
##  $ c3                          : num  0.422 0.369 0.165 0.624 0.465 0.209 0.178 0.478 0.544 0.59 ...
##  $ c4                          : num  1.06 1.25 0.6 2.07 1.45 ...
##  $ c5                          : num  0.0166 0.0174 0.0193 0.0198 0.02 ...
##  $ c6                          : num  0.0833 0.1044 0.0988 0.0633 0.0853 ...
##  $ c7                          : num  0.131 0.224 0.115 0.194 0.211 ...
```

5 Supervised Learning Experiment

In this section, one supervised learning model is executed and evaluated. The learning was carried out to see how accurately an algorithm can predict the specie of fish that escape from a farm using a given set of predictors. A modified version of the escapesPlus dataset was used. The model used was:

- Random forests (RF).

Before running the model, a number of variables were deleted as they were not required for the learning experiment, mainly because most of them had zero variance. Cross validation resampling method was performed in line with the model with the aid of the 'caret' package in order to make sure the model is robust and the best parameter combinations is determined.

The Accuracy and Kappa are the output metrics of the model for comparison and evaluation. By default, the model selects the best tuning based on the highest Accuracy value.

5.1 Prerequisites

5.1.1 Dropping irrelevant predictors

The following variables which were deemed to be irrelevant to the learning were removed.

```
escapesPlus$Operator.at.Time.of.Escape = NULL
escapesPlus$Escape.ID = NULL
escapesPlus$Marine.Scotland.Site.ID = NULL
```

```r
escapesPlus$Site.Post.Code = NULL
escapesPlus$Site.Name = NULL
escapesPlus$Escape.Start.Date = NULL
escapesPlus$c2 = NULL
escapesPlus$c3 = NULL
escapesPlus$c4 = NULL
escapesPlus$c5 = NULL
escapesPlus$c6 = NULL
escapesPlus$c7 = NULL
```

5.1.2 Control specification

The 10-fold cross validation control mechanism (resampling method) was used for the model.

```r
trControl = trainControl(method="cv", number=10)
```

5.2 Random Forest

Random forests is a decision tree based model that draws from the simple idea of the wisdom of the multitude. It is an Ensemble Learning technique in which an aggregate of the results of multiple predictors are used to give a better prediction than the best individual predictor. Random Forests is a widely used model mainly due to its ability to reduce overfitting compared to some other models. The parameter mtry which is the number of randomly sampled variables for splitting at each node is tuned accordingly to get the most appropriate model. The target variable for this model is the Escaped Specie.

```r
#specify tuning parameters
mtry <- c(2,3,4,5,6,7,8,9)
tunegrid <- expand.grid(mtry=mtry)

#run model
set.seed(12345)
RF.fit = train(Escaped.Species ~ .,data = escapesPlus, trControl = trControl,
method = "rf", preProc = c("center","scale", "nzv"),
tuneGrid=tunegrid) #train and test model

RF.fit
```

```
## Random Forest
##
## 351 samples
##  10 predictor
##   7 classes: 'atlantic salmon', 'brown trout and sea trout', 'cod', 'halibut', 'lumpsucker', 'rainbo
##
## Pre-processing: centered (33), scaled (33), remove (29)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 315, 317, 317, 317, 314, 316, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##   2     0.8307323  0.2911768
##   3     0.8512605  0.4320715
```

```
##   4     0.8758703  0.5501782
##   5     0.8674670  0.5180576
##   6     0.8674670  0.5234759
##   7     0.8756303  0.5524764
##   8     0.8756303  0.5534666
##   9     0.8756303  0.5534666
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 4.
```

The result of the RF training and testing process is shown above. An mtry of 4 produces the best result with the highest Accuracy of 87.59% and Kappa of 0.5502, indicating a good performance of the model.

The variable importance for the model is shown below. Seawater from the water type variable is observed to be the most important variable, closely followed by Average Weight.

```
#Checking the variable importance for the model.
varImp(RF.fit)
```

```
## rf variable importance
##
##   only 20 most important variables shown (out of 33)
##
##                                      Overall
## Water.Typeseawater                   100.000
## Average.Weight.Kg                     96.521
## Age.in.Months                         56.045
## Final.Number.Escaped                  54.212
## Final.Number.Recovered                37.974
## Health.Surveillancelow                17.438
## Producing.in.Last.3.Yearsyes          15.986
## Month5                                15.833
## Month6                                13.624
## Health.Surveillancemedium             12.368
## Final.Escape.Reasonpredator - prd      9.130
## Year2007                               8.913
## Month7                                 8.033
## Year2005                               7.516
## Final.Escape.Reasonhole in net - hol   6.633
## Year2008                               6.447
## Final.Escape.Reasonhuman error - hum   6.182
## Month3                                 5.970
## Year2009                               5.947
## Month4                                 5.491
```

Figure 5 shows variable importance plots for the Random Forest model. The predictor 'Water.Type' has the highest importance in the model.

6 Conclusion and recommendation

In this report, regression was performed on a Scottish Aquaculture dataset of escaped fishes and water analysis using the random forest model. The target variable was Specie of Escaped Fish which consists of

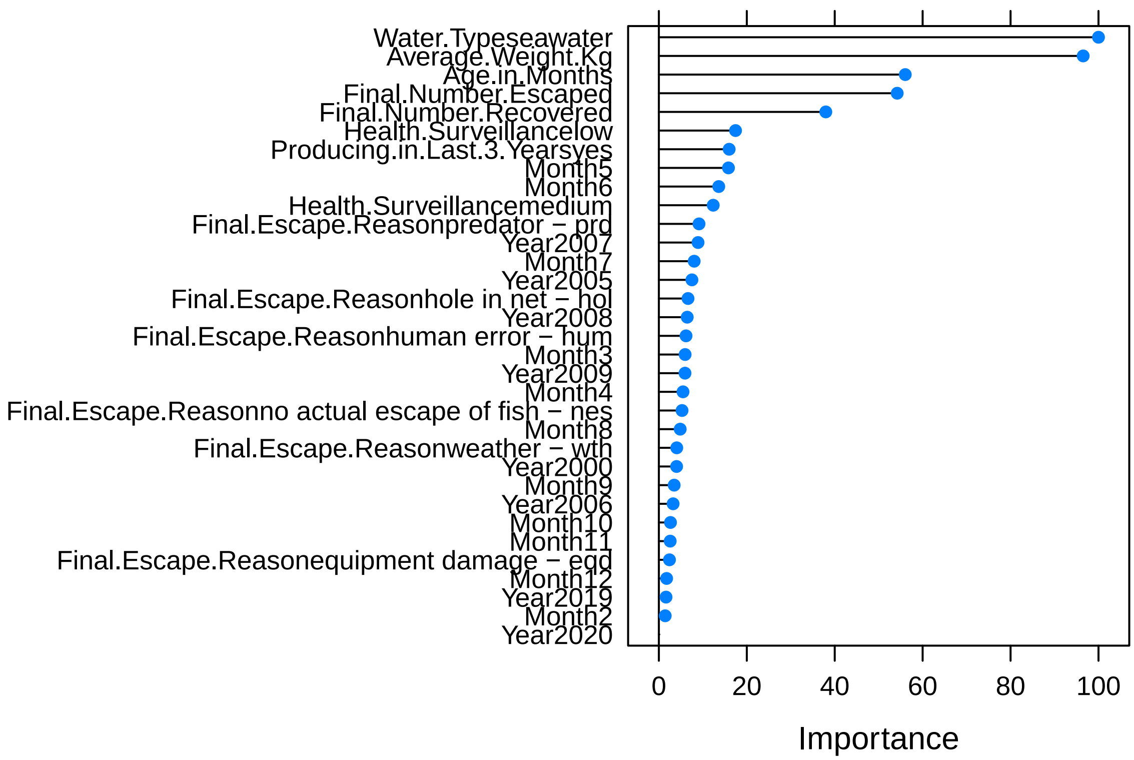

Figure 5: Variable Importance

7 categories. In the end, the result of the model was quite good with an accuracy of 87.59% in predicting the target variable. It is recommended that other learning models such as Generalized Linear Model, Support Vector Regression and Convolutional Neural Networks are applied on the data as there is always the possibility of getting a more reliable model.

7 References

DATA NOVIA TEAM, 2018. Data Manipulation in R. [Online]. Data Novia. Available at: https://www.datanovia.com/en/lessons/identify-and-remove-duplicate-data-in-r/. Accessed [11 March 2022].

YIHUI XIE, CHRISTOPHE DERVIEUX & EMILY RIEDERER, 2022. R Markdown Cookbook. [Online]. Chapman & Hall/CRC. Available at: https://bookdown.org/yihui/rmarkdown-cookbook/. Accessed [12 March 2022].