Heat maps portraying bird escapes in New Zealand

Chrystal Foy, Theodora Jin, Shaun Roberts

The University of Auckland, PB 92019, Auckland, New Zealand

Abstract

We generated and commented on a range of heat maps for bird escapes in New Zealand.

- The 5 species with the most escapes were Cockatiels (489), Budgerigars (324), Ring-necked parakeets (124), Lorikeets (86), and Rainbow lorikeets (78).
- Auckland and Canterbury had the most bird escapes with approximately 50% and 16% of all bird escapes respectively.
- Most escapes in Auckland occur in Devonport, Central Auckland, and near the Waitakere ranges.
- Most escapes occur during spring and summer.

---

1. Introduction

Lost pet birds pose a threat, not only to native trees and plant life, but native bird species as well, and if they get established the damage is compounded. In order to explore the distribution of exotic bird escapes in New Zealand, we have generated a range of heat maps. A heat map is a representation of data in the form of a map or diagram in which data values are represented as colours. Below is an example heat map from NIWA which shows the drought index for September and October 2014.

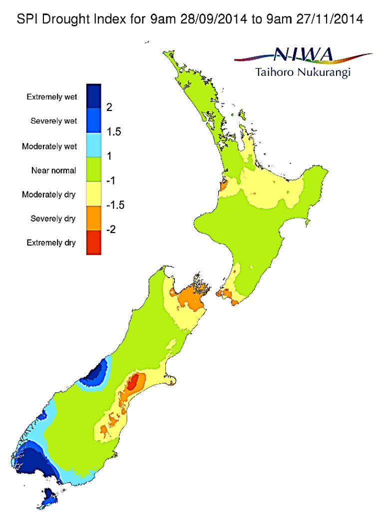

Figure 1: NIWA heatmap of Drought Index for September and October 2014

2. Background: Exotic birds

There are approximately 174 native bird species in New Zealand, comprising 71% of the 245 species of birds breeding in New Zealand (Brockie, 2007). As part of natural selection, species can become endangered or extinct if they cannot survive due to competition for resources (Scoville, 2017). This is often termed survival of the fittest and as it states, the fittest and most adaptable species are the ones that survive. If exotic bird species are introduced and/or established, they can out compete weaker native species for resources which can lead the locals to becoming endangered or even extinct. These invasive species can also spread diseases and have negative effects on crops.

Many bird species became extinct when humans settled New Zealand due to the introduction of new animals (Wodzicki and Wright, 1984). The Maori dog and the Kiore rat were introduced by the Polynesian settlers while European rats, mice, and bird species were introduced by the European settlers. Wodzicki and Wright (1984) describe the effects of the introduction of the bird species as adverse. These birds spread avian diseases and out compete native bird species causing their populations to decrease or even disappear completely.

The introduction of new bird species is still a threat. An example of an invasive bird species is the Rainbow lorikeet (Trichoglossus haematodus). According to the New Zealand Department of Conservation (n.d.), the Rainbow lorikeet was released deliberately and illegally in the 1990s in Auckland and whilst now legal is now a threat to native wildlife. They are invasive, as they out compete native birds for food and nesting sites. They also carry avian diseases that affect native bird species, including endangered native parrots, native honey-eaters, such as the much loved Tui and native cavity nesters such as the Kaka. Additionally, they can affect the commercial fruit industry, as they can damage fruit crops.

Invasive bird species clearly have a negative impact on NZ natives and if invasive populations get established, these effects are compounded. Preventative measures must be taken to ensure this does not occur by identifying risk hot-spots and monitoring them. One way to identify these hot-spots is to look at the location of where birds are lost as pets across New Zealand. If many of the same bird species are lost or released into the wild within close proximity to each other, it can increase the likelihood of a wild population getting established.

The five key bird species of interest are the Ring-necked parakeet, Rainbow lorikeet, Alexandrine parakeet, Sulphur-crested cockatoo and the Galah. We also looked at the top five most lost bird species that we identified from the data.

Our aim was to produce animated heat maps to portray the number of birds lost within Auckland, for each of the five key bird species and the top five lost bird species over a period of time. An overall static heat map of New Zealand showing birds lost was also produced. The purpose of these heat maps was to provide an educational tool to help educate the public about exotic bird escapes and to help identify risk hot-spots in New Zealand.

3. Method

The original data was composed of 1,464 observations, where each observation is a lost bird listing from TradeMe and PetsOnTheNet in New Zealand listed between 1 January 2010 and 27 November 2017 with 29 variables. The data was cleaned for consistency and duplicates including unreasonable dates like the 27th of November 2017 as at the time of analyis this had not come to pass. The data was also prepared for heat map generation by adding additional variables. These were conducted using **R**.

3.1. Data Cleaning

The data was split to only include listings from the 4th of August, 2012 to the 25 of April, 2017. This new dataset had 1,463 lost bird listings.
Originally the species were labelled directly as they would be read from the listings, this means that there were many different names for the same species as people call the same bird species different things. This required knowledge of the different common and scientific names of the species, which we did not have. Josie Galbraith was kind enough to uniformly name the species with their common name, specific name, family, and order. Additionally the date the birds were lost was missing for many observations, as such we replaced it with the date the listings was placed.
The data was probabilistically checked for duplicates using the Fellegi-Sunter record linkage method. The R code to do this can be found in the appendix. There were a few possible duplicates, however, most of these were either escapes of breeding pairs or groups, or were the same bird escaping at different times.

3.2. Number of listings per location

Table 1 portrays the number of lost bird listings by location (city) in New Zealand between 4 August 2012 to 25 April 2017. Auckland has the largest number of bird listings contributing to almost 50% of the total bird listings and is almost three times more than Canterbury, the city with the second largest number of bird listings. Due to this sole reason, the focus and the generation of heat maps were only for Auckland. For a bigger picture and better understanding, a static heat map of New Zealand for all species lost, was also produced.

Table 1: Number of listings per location

Region	Counts	Region	Counts
Auckland	684	Tasman	16
Canterbury	244	Southland	12
Waikato	113	Marlborough	8
Wellington	112	Wairarapa	8
Bay of Plenty	90	Franklin	6
Manawatu	46	Kaipara	5
Hawkes Bay	30	Coromandel	2
Northland	31	Rodney	2
Otago	31	Far North	1
Taranaki	19	West Coast	1

3.3. Data preparation for heat maps

To generate the heat maps, **R** code was kindly provided by Associate Professor Rachel Fewster from the University of Auckland. **Table 2** lists the variables that were kept, added and dropped.

Variables kept were of relevance to our aim, this included the species common name, the date the listed bird was lost, the number of birds lost in one listing, and the record number, a unique number given to each observation (i.e. First observation gets assigned one, the tenth gets ten, etc). The number of birds were originally a categorical variable (*Escape.type*) but were converted into a numeric variable (*NumberBirds*). This provided a weighting for each observation row, to allow correct representation in the heat maps. Irrelevant variables were dropped.

Additional variables were required for the **R** code to work. Including changing the location of the bird escape into a longitude and latitude form. This was generated by using details across many variables (street, suburb, town, region, country, and location) and a package in **R**. Excel was also used to manually input longitude and latitude co-ordinates, when addresses were not provided in a specific manner (e.g. at the junction of two streets). Two other variables (*TrapChecked* and *TrapName*) were also required but were irrelevant to the dataset.

Two sets of the animated heatmaps were generated, one set takes 15 seconds to go through the almost 5 years of bird escapes, the second set take 20 seconds.

Table 2: Data variables

Kept (Renamed)	**Added**	**Dropped**	
Common.name (Species)	Latitude	Additional.Details	Owner.ID
Date.Lost	Longitude	Age	Pet.Name
Escape.type (NumberBirds)	TrapChecked	Breeder	Ph.Number
Record	TrapName	Country	Region
		Date.Found	Sex
		Date.Listed	Source
		Family	Species.name
		Found	Street
		Location	Suburb.Area
		Order	Town

4. Results

4.1. Heat maps

The total number of listings for the five key and five top species in Auckland are tabulated in **Table 3**. These species and escapes were used to generate heat maps over the time period. At the top of each heat map, a scroll bar moves as the time elapses. This scroll bar changes colour based on the seasons to allow for easier interpretation of escapes through seasons (blue is for Spring/Summer, red is for Autumn and orange is for Winter). Due to the Sulphur-crested cockatoo having a sample size of 13 (approximately) over the five year period, there were not enough observations to make an informative heat map.

Table 3: Number of listings per Key and Top Species

Common Name	Counts
Ring-necked parakeet	124
Rainbow lorikeet	78
Alexandrine parakeet	52
Galah	36
Sulphur-crested cockatoo	13

(a) Five Key Species

Common Name	Counts
Ring-necked parakeet	124
Rainbow lorikeet	78
Cockatiel	489
Budgerigar	324
Lorikeet	86

(b) Five Most Lost Species

All heat maps for these species and all species are found in **Appendix B**. For the five key and five top species in Auckland, a summary of risk hot-spots and seasonality in escapes per exotic species are in **Table 4**.

Overlaps occurred in five Key invasive species and five Top frequently escaping species, they are Ring-necked parakeet and Rainbow lorikeet. Escapes occur seasonally, predominantly in Central Auckland, Devonport and Waitakere in Spring and Summer. As our data provides a general picture for birds escaping around New Zealand, majority of escapes occur in the North Island, centered at Auckland and spread at Hamilton and Tauranga. There are also large numbers of escapes in Wellington and Christchurch as expected, as they are high population density areas.

Table 4: Summary of heat maps

Heat maps	Category	Location	Seasonality	Additional Comment
Ring-necked parakeet	BOTH	Main band of Auckland, between the harbour and Takanini, Auckland CBD,West Auckland, South Auckland and Devonport	Spring and Summer	
Rainbow lorikeet	BOTH	Auckland CBD, Devonport and Waitakere	Spring and Summer	Escapes more isolated in certain areas
Alexandrine parakeet	KEY	Auckland CBD, Takanini, central west Auckland	Spring and Summer	
Galah	KEY	Auckland CBD, Waitakere, Orakei, North Shore	Spring and Summer	
Sulphur-crested cockatoo	KEY	-	-	Small sample size so no heat map was generated
Cockatiel	TOP	Central CBD, West Auckland and Devonport	June/July and November	
Budgerigar	TOP	Auckland CBD and West Auckland, some in South Auckland	Spring/Summer	
Lorikeet species	TOP	Devonport, Auckland CBD, Bucklands beach and west Auckland		A lot more sporadic
Auckland	ALL	Devonport, central CBD, more spread out in the most northern and southern parts	Spring and Summer	Number increasing over time, may proportion to population density
NZ Static	ALL	Majority in the North Island, Auckland, Hamilton, Tauranga, more observation in Wellington and Christchurch		Many escapes in high population areas, more in Auckland than expected compared to Wellington

Figure 2 is a static heat map of all lost birds in New Zealand. Around half of the escapes occurred in Auckland, followed by Wellington and Christchurch. The majority of the escapes occur in the North Island, centered in Auckland, spread across Hamilton and Tauranga. When comparing the population sizes of locations, more escapees per person are seen in Auckland than expected, compared to Wellington. Less escapes are seen on the West coast of the South Island such as in Queenstown, which is not surprising as escapes are more likely to happen in high population areas.
An animated heat map for all species of lost birds in Auckland can be seen in **Figure 3**. Initially escapes predominantly occur around central Auckland, west Auckland and Devonport, but over time the number of escapes increase and appear to change with the season. There was a high density of escapes in central Auckland that are continuous. Escapes occur as far north as Orewa, and as far south as the Hunua ranges.

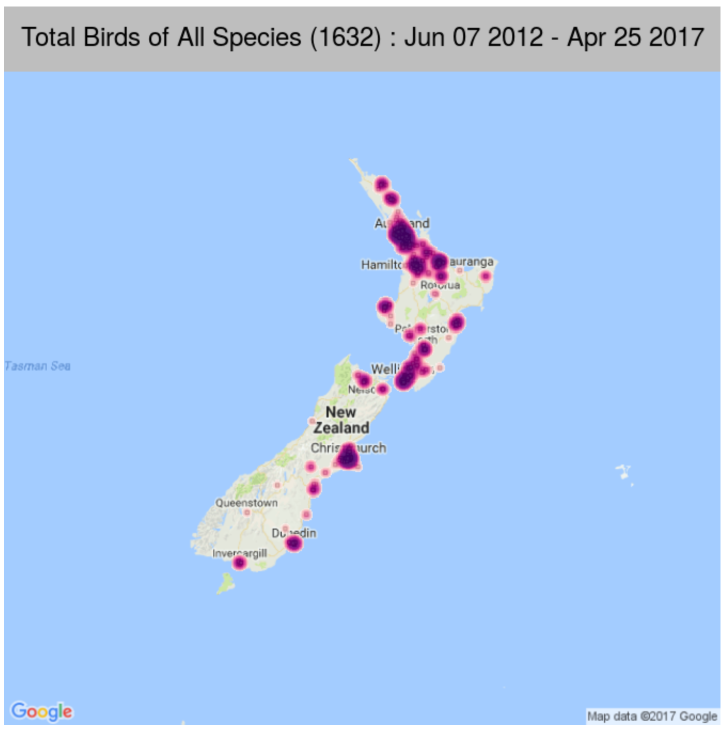

Figure 2: Static heat map of all species in New Zealand

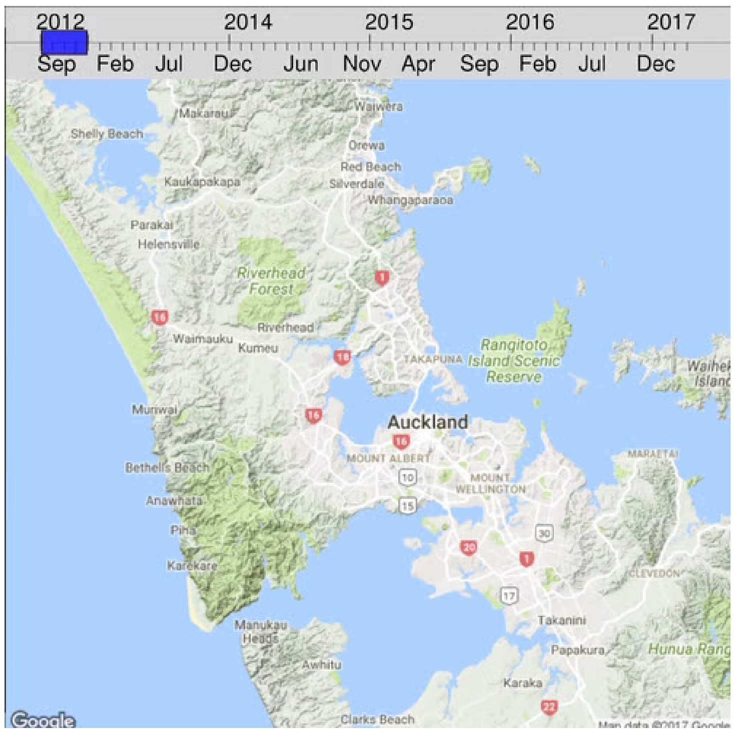

Figure 3: Animated heat map for all species in Auckland

4.2. Time series analysis

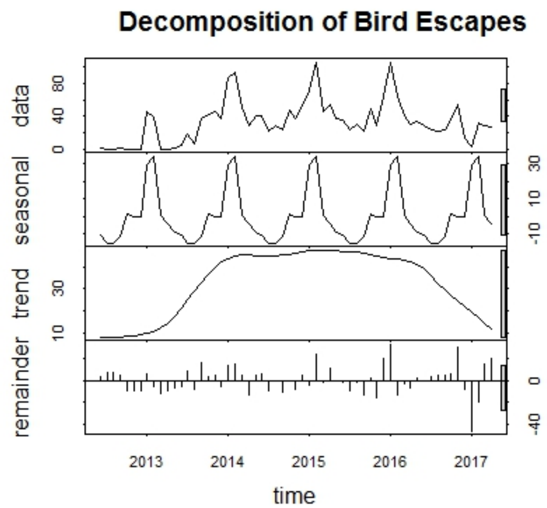

Figure 4: Time series of lost bird listings

To investigate the pattern in bird escapes within Auckland, the total number of bird escapes for all species was plotted on a time series. This can be seen in *data*, the top panel in **Figure 4**. The time series was broken down into three components: *seasonal*, *trend*, and *remainder*.

Seasonal components are regular and predictable patterns that occur over a same time period. Seasonality can be seen in the second panel called *seasonal* where the number of bird escapes fluctuate up and down as a regular pattern every year. Peak escapes occur during Spring and Summer with low escapes occurring during Winter.

Trend components show an increasing or decreasing pattern over time. Shown in the third panel called *trend*, bird escapes appeared to have increased from 2012 to 2015, plateauing for a year before decreasing in mid-2016.

Remainder components are any remaining random variation after taking into account seasonal patterns and trend. The fourth panel called *remainder* looks reasonably random except for a large residual in early 2017. This large residual and the decrease in trend from mid-2016 could be due to the way the data was collected.

5. Discussion

5.1. Overall results

Identified risk hot-spots of bird escapes in Auckland include central Auckland, west Auckland and Devonport. Overall in New Zealand, most escapes occur in Auckland, followed by Canterbury, Waikato and Wellington. Escapes appear to be seasonal, occurring predominantly during Spring and Summer.

These lost bird escapes include a wide range of species including the Cockatiel, the Budgerigar, and the Rainbow lorikeet. These three species are the most dominant exotic species to be lost within Auckland and can be very concerning as they can be very invasive. For example, Rainbow lorikeets are known to be pests as they damage fruit crops. More importantly, they can drive New Zealand native birds into endangerment or extinction. They compete against the native Kaka for cavity nests and the native tui for nectar, putting their reproductive capabilities and survival at risk.

With the high negative impact exotic birds have on New Zealand native birds, preventative measures should be taken by using the identified risk hot-spots and seasonal patterns of escaping.

5.2. Speculations

Why are we seeing the patterns mentioned above? The high number of escapes in Devonport that appear in all the heatmaps could be because Devonport has an older population and parrots and other exotic birds make great pets for the elderly as they are not physically demanding and provide company. Additionally, pet birds generally don't take up much space and are therefore reasonable pets to have in apartments like those in the inner city. The high number of escapes around the centre of west Auckland could be explained by the Waitakere ranges, as the birds have somewhere to go. Additionally, those who live near the Waitakere ranges probably enjoy nature and are more likely to have pet birds. The seasonality can be attributed to when the breeding seasons for many pet birds are. For many exotic species breeding season is during spring and summer and the drive to reproduce encourages them to escape and find a mate.

5.3. Limitations

Both the data source and technical analysis aspects provide limitations to conclusions that can be drawn.

Many birds escaped without a lost date, so an approximation was made using the listing date, reducing the precision of our analysis. Also, the data source came from only two websites, TradeMe and PetsOnTheNet. As not all lost birds are listed, we do not know the real population of escaping birds in New Zealand. Hence the generalisations we could make were limited.
When rerunning the duplication checking code, we found that there were five duplicate listing which we did not find with our initial duplicate checking. However, as they were three Cockatiels and two Budgies this was only 0.006% more observations for each Cockatiels and Budgies.
Lastly, our aim was to create an educational tool for the public, about the impact of lost birds establishing populations, therefore minimal technical analyses were conducted and results should be interpreted with care.

6. Conclusion

Just under half the escapes occurred in Auckland, followed by Canterbury with around 16% of the total escapes. The rest of the regions had very few escapes. The bird species with the most escape events were Cockatiels (489), Budgerigars (324), Ring-necked parakeets (124), Lorikeets (86), and Rainbow lorikeets (78) respectively. These escapes were seasonal with more during spring and summer, and less during autumn and winter. In Auckland most of these escapes occurred around the Waitakere ranges, central Auckland, and Devonport.

Acknowledgements

The authors are grateful to Rachel Fewster and Peter Mullins for their support and guidance throughout this project.

References

[1] Brockie, B. (2007, Sep 24). Native plants and animals - overview - Species unique to New Zealand, Te Ara - the Encyclopedia of New Zealand. Retrieved from https://www.teara.govt.nz/en/native-plants-and-animals-
overview/page-1. (Accessed 7 October 2017).

[2] Department of Conservation. (n.d.). Rainbow lorikeet. Retrieved from http://www.doc.govt.nz/nature/pests-and-threats/animal-pests/rainbow-lorikeet/. (Accessed 7 October 2017).

[3] Scoville, H. (2017, Apr 12). Survival of the Fittest? Retrieved from https://www.thoughtco.com/survival-of-the-fittest-1224578. (Accessed 7 October 2017).

[4] Wodzicki, K. and Wright, S. (1984). Introduced birds and mammals in New Zealand and their effect on the environment. Tuatara, 27(2). Retrieved from http://nzetc.victoria.ac.nz/tm/scholarlytei-Bio27Tuat02-t1-body-d1.html. (Accessed 7 October 2017).

Appendices

Appendix A: Tables

Table 5: Summary of bird escape events in descending order

Species	**Counts**	**Species**	**Counts**	**Species**	**Counts**
Cockatiel	489	Bourke's parrot	11	Senegal parrot	2
Budgerigar	324	Moustached parakeet	9	Amazon parrot	1
Ring-necked parakeet	124	Red-rumped parrot	9	Australian ringneck parrot	1
Lorikeet spp	86	Eclectus parrot	5	Bengalese finch	1
Rainbow lorikeet	78	Sun conure	5	Domestic duck	1
Alexandrine parakeet	52	Conure spp	4	Dove spp	1
Domestic canary	47	Domestic pigeon	4	Eastern rosella	1
Parrot spp	45	Grass parrot spp	4	Fischer's lovebird	1
Galah	36	Monk parakeet	3	Goose spp	1
Lovebird spp	34	Turquoise parrot	3	Lord Derby's parakeet	1
Superb parrot	17	Finch spp	2	Muscovy duck	1
Sulphur-crested cockatoo	13	Macaw spp	2	Peach-fronted conure	1
Australian king parrot	12	Red-collared lorikeet	2	Rosella spp	1
African grey parrot	11	Red-throated parrotfinch	2	Scaly-breasted lorikeet	1
Barbary dove	11	Red-winged parrot	2	Zebra finch	1

Table 6: Summary of bird escape events in alphabetical order

Species	**Counts**	**Species**	**Counts**	**Species**	**Counts**
African grey parrot	11	Eastern rosella	1	Peach-fronted conure	1
Alexandrine parakeet	52	Eclectus parrot	5	Rainbow lorikeet	78
Amazon parrot	1	Finch spp	2	Red-collared lorikeet	2
Australian king parrot	12	Fischer's lovebird	1	Red-rumped parrot	9
Australian ringneck parrot	1	Galah	36	Red-throated parrotfinch	2
Barbary dove	11	Goose spp	1	Red-winged parrot	2
Bengalese finch	1	Grass parrot spp	4	Ring-necked parakeet	124
Bourke's parrot	11	Lord Derby's parakeet	1	Rosella spp	1
Budgerigar	324	Lorikeet spp	86	Scaly-breasted lorikeet	1
Cockatiel	489	Lovebird spp	34	Senegal parrot	2
Conure spp	4	Macaw spp	2	Sulphur-crested cockatoo	13
Domestic canary	47	Monk parakeet	3	Sun conure	5
Domestic duck	1	Moustached parakeet	9	Superb parrot	17
Domestic pigeon	4	Muscovy duck	1	Turquoise parrot	3
Dove spp	1	Parrot spp	45	Zebra finch	1

Appendix B: Heatmap

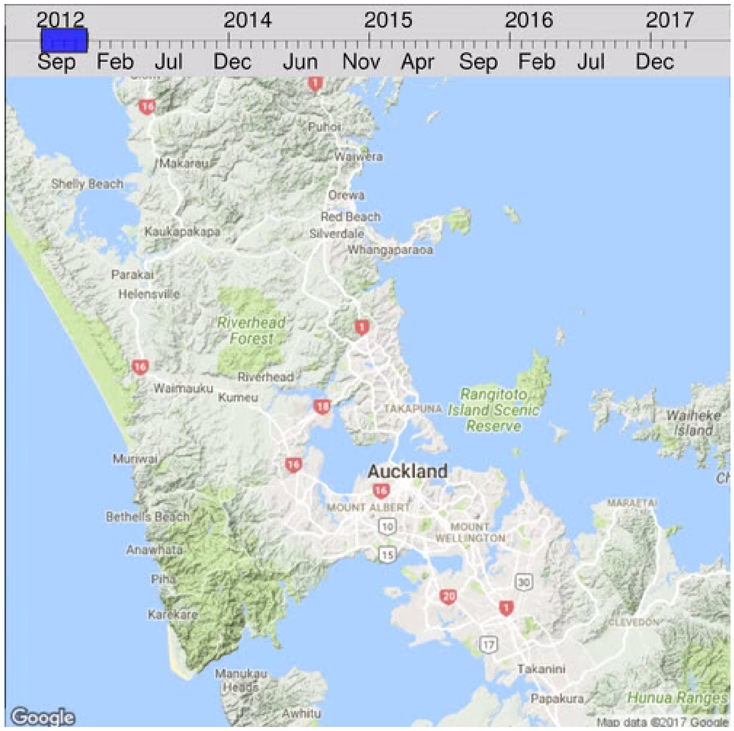

Figure .5: Animated heat map for the Ring-necked parakeet in Auckland

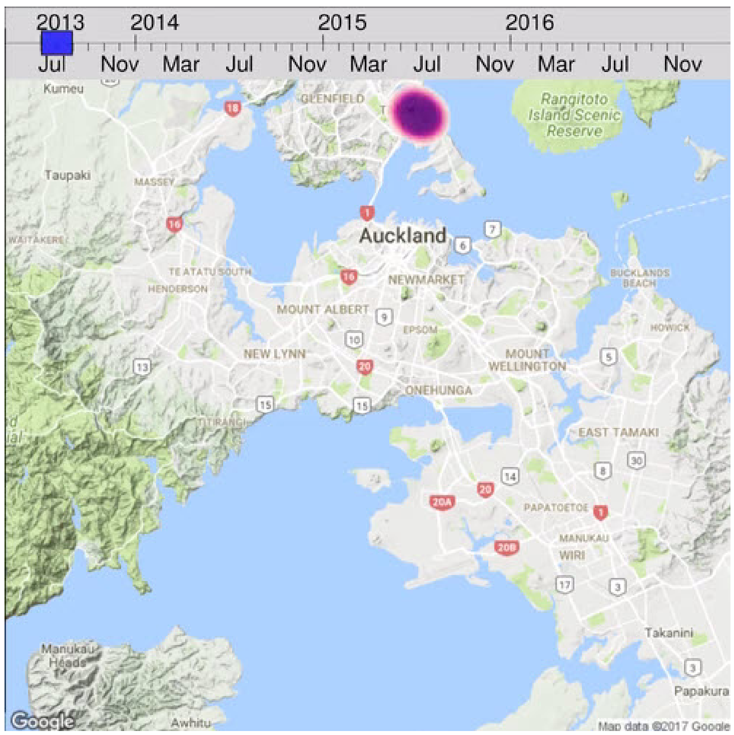

Figure .6: Animated heat map for the Rainbow lorikeet in Auckland

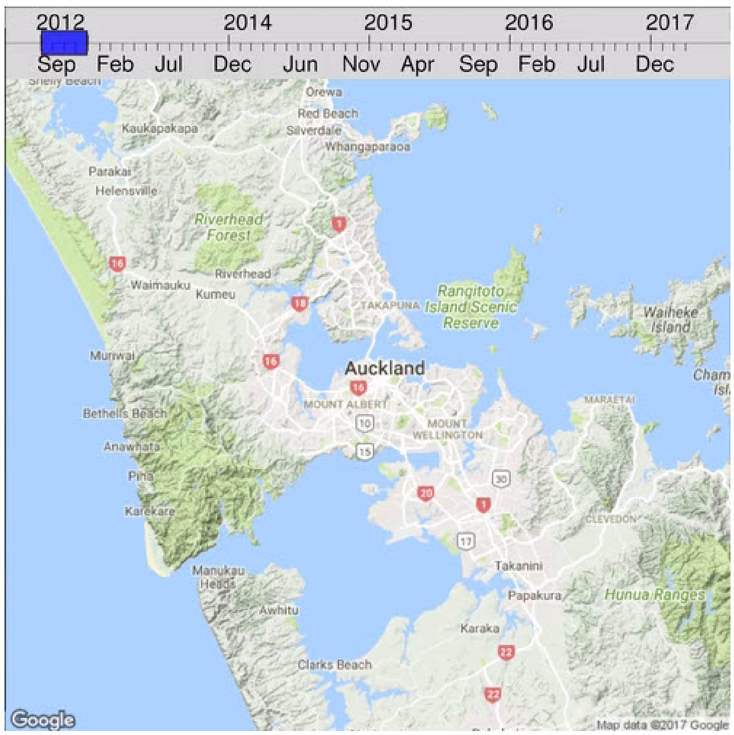

Figure .7: Animated heat map for the Alexandrine parakeet in Auckland

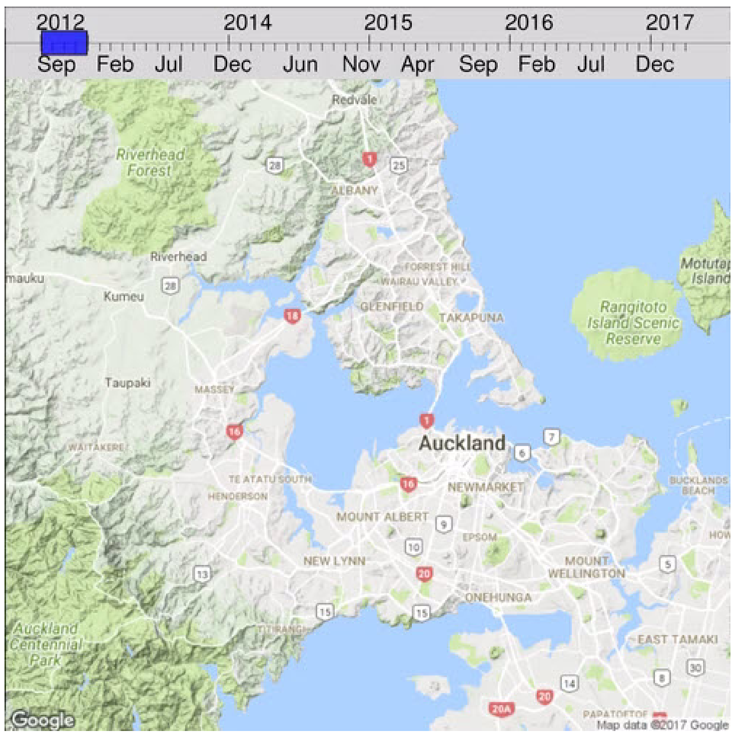

Figure .8: Animated heat map for the Galah in Auckland

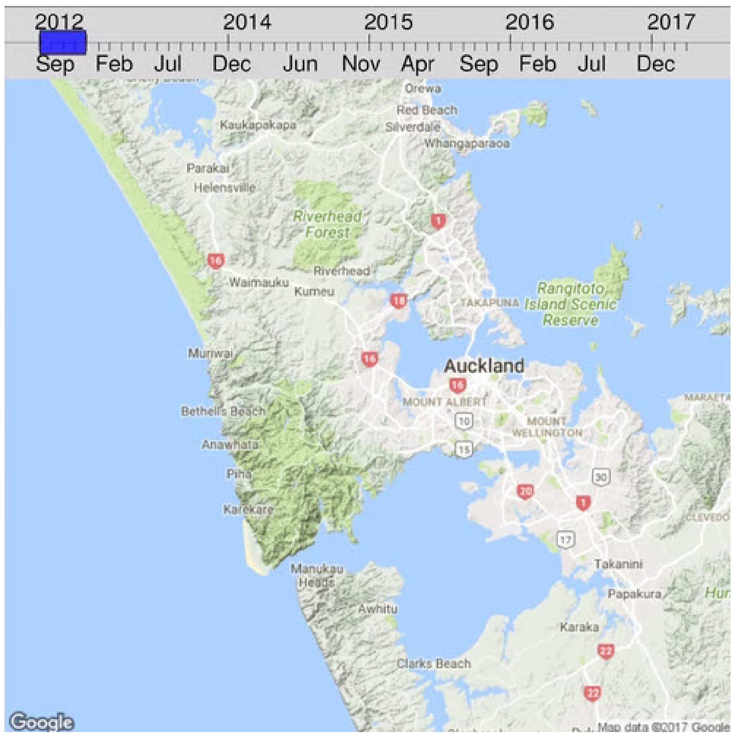

Figure .9: Animated heat map for the Cockatiel in Auckland

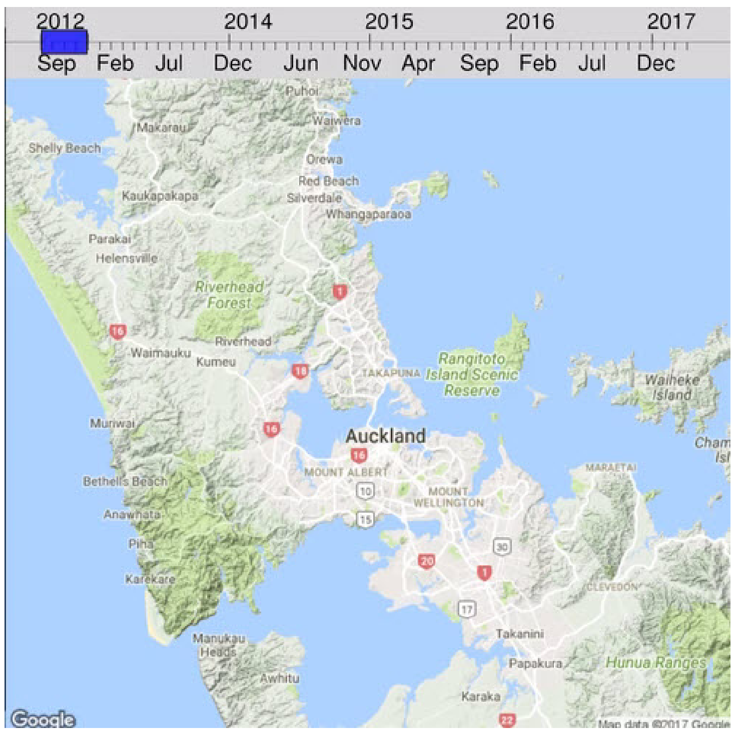

Figure .10: Animated heat map for the Budgerigar in Auckland

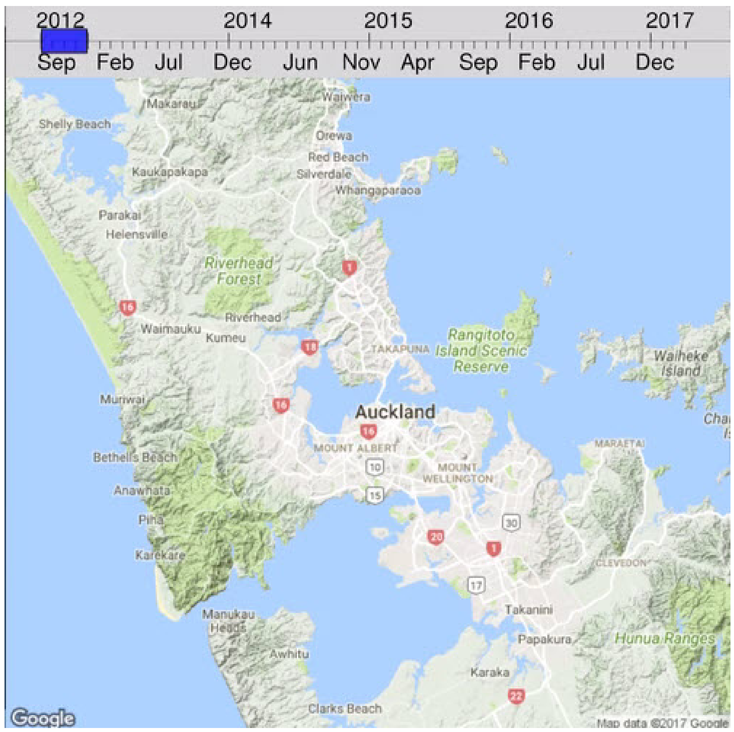

Figure .11: Animated heat map for the Lorikeet species in Auckland

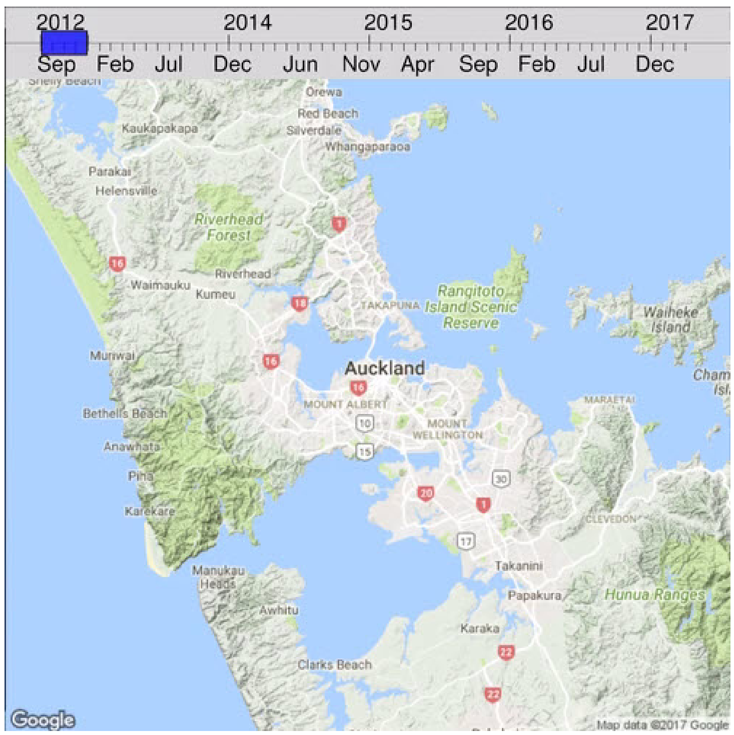

Figure .12: Animated heat map for all species in Auckland

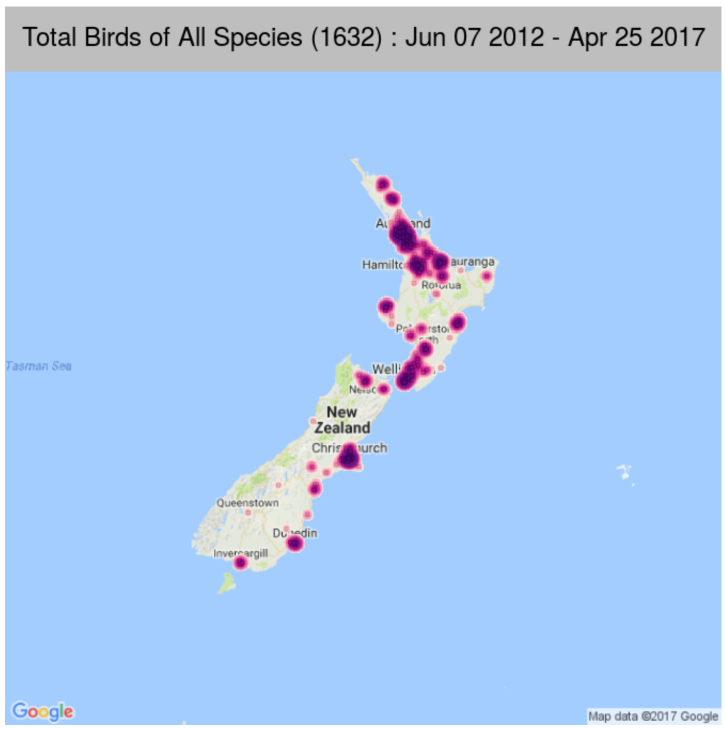

Figure .13: Static heat map for all species in New Zealand

Appendix C: R Code

Variable consistency: Species and Escapes

```r
# read data in

knitr::opts_chunk$set(echo = TRUE)
library(plyr)

library(xlsx)
# dates as integer
# character variables treated as factors

birds = read.xlsx("Birddata.xlsx", 2, header=TRUE, startRow=2,
    colIndex=1:23, colClasses=c("character", "integer", "integer"
    , "character", "integer", rep("character", 18)),
    stringsAsFactors = TRUE)

# escape - changing levels

birds2 = data.frame(birds, stringAsFactors = TRUE)

birds2$Escape.type = as.character(birds2$Escape.type)
is.character(birds2$Escape.type)

x = unique(birds2$Escape.type)
x

# frequency table of escapes for all original levels
table = count(birds2$Escape.type)

# individual escapes
birds2$Escape.type[birds2$Escape.type == x[1]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[4]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[5]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[6]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[10]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[11]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[14]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[15]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[16]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[16]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[19]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[22]] <- "1"
birds2$Escape.type[birds2$Escape.type == x[25]] <- "1"

```

```
# double escapes
birds2$Escape.type[birds2$Escape.type == x[2]] <- "2"
birds2$Escape.type[birds2$Escape.type == x[13]] <- "2"
birds2$Escape.type[birds2$Escape.type == x[24]] <- "2"

# 3 escapes
birds2$Escape.type[birds2$Escape.type == x[9]] <- "3"
birds2$Escape.type[birds2$Escape.type == x[23]] <- "3"

# 4 escapes
birds2$Escape.type[birds2$Escape.type == x[8]] <- "4"
birds2$Escape.type[birds2$Escape.type == x[17]] <- "4"

# 5, 6, 7, 12, 20 escapes
birds2$Escape.type[birds2$Escape.type == x[3]] <- "8"
birds2$Escape.type[birds2$Escape.type == x[7]] <- "12"
birds2$Escape.type[birds2$Escape.type == x[12]] <- "20"
birds2$Escape.type[birds2$Escape.type == x[18]] <- "5"
birds2$Escape.type[birds2$Escape.type == x[20]] <- "6"

x = unique(birds2$Escape.type)
x

# convert Escape.type to numeric
birds2$Escape.type = as.numeric(birds2$Escape.type)
sort(unique(birds2$Escape.type))



# table of frequencies for cleaned levels of escapes

# by order of 'escape type'
escapes = count(birds2$Escape.type)
colnames(escapes) = c("Escape type", "Frequency")
escapes

# by order of 'frequency numbers'
escapetable = escapes[order(-escapes[,2]),]
escapetable

#Checking the different species names.
levels(birds2$Species)
```

rcode/birdescape.R

Variable consistency: Date Listed and Lost

```
#Finding the indices of the values in Date.Lost that are na:
index_date_lost = which(is.na(birds2$Date.Lost))
birds2$Date.Lost[index_date_lost] = birds2$Date.listed[index_date_lost]
```

rcode/DateLostFixer.R

Duplicate Checking Code:

```
library(jsonlite)
library(xml2)
library(xlsx)

birds2 = read.xlsx("H:/780Proj/nzgeocode.xlsx", sheetIndex = 1, header = T)
birds3 = birds2
head(birds2)
colnames(birds2)

#Removing the last 2 observations
birds3 = birds3[(1:(length(birds3$Record.) -2)), ]

#Removing the variables we do not want to use.
birds3$NA..1 = NULL
birds3$NA. = NULL
birds3$Record.= NULL
birds3$FOUND. = NULL
birds3$Date.found = NULL
birds3$Specific.name = NULL
birds3$Family = NULL
birds3$Order = NULL
birds3$Escape.type = NULL
birds3$Street = NULL
birds3$Suburb.Area = NULL
birds3$Town = NULL
birds3$Region = NULL
birds3$Source = NULL
birds3$Breeder. = NULL
birds3$Ph.Number = NULL
birds3$Additional.Details = NULL
birds3$Location = NULL
birds3$Country = NULL
birds3$Species = NULL

#Attaching numbers to the
birds3 = cbind(1:dim(birds3)[1], birds3)
```

```
#Finding the indices of the values in Date.Lost that are na:
index_date_lost = which(is.na(birds3$Date.Lost))
birds3$Date.Lost[index_date_lost] = birds3$Date.listed[index_
    date_lost]

#Fixing date listed nas
index_date_listed = which(is.na(birds3$Date.listed))
birds3$Date.listed[index_date_listed] = birds3$Date.Lost[index_
    date_listed]

#Removing observation missing both date listed and date lost.
birds3 = birds3[-82,]


colnames(birds3)
#Setting our m values
#Date listed:
mdli = 0.5
#Date lost:
mdlo = 0.5
# #Species
msp = 0.95
#Common name
mcom = 0.99
#Sex
msex = 0.9
#Age
mage = 0.9
#Pet Name
mna = 0.99
#Owner ID
mown = 0.99
#Long
mlong = 0.8
#Lat
mlat = 0.8

mvec = c(mdli, mdlo, msp, mcom, msex, mage, mna, mown, mlong,
    mlat)
#Creating the mu vector:
#mvec = c(mdli, mdlo, mcom, msex, mage, mna, mown, mlong, mlat)

#Creating the u vector
#Date listed:
udli = mean(outer(birds3$Date.listed, birds3$Date.listed, "==")),
```

```
    na.rm = T)
#Date lost:
udlo = mean(outer(birds3$Date.Lost, birds3$Date.Lost, "=="), na.rm = T)
#Owners id
uown = mean(outer(birds3$Owner..ID, birds3$Owner..ID, "=="), na.rm = T)
# #Species
usp = mean(outer(birds3$Species, birds3$Species, "=="), na.rm = T)
#Common name
ucom = mean(outer(birds3$Common.name, birds3$Common.name, "=="), na.rm = T)
#Pet Name
una =  mean(outer(birds3$Pet.Name, birds3$Pet.Name, "=="), na.rm = T)
#Sex
usex = mean(outer(birds3$Sex, birds3$Sex, "=="), na.rm = T)
#age
uage = mean(outer(birds3$Age, birds3$Age, "=="), na.rm = T)
#Longitude
ulong = mean(outer(birds3$longitude, birds3$longitude, "=="), na.rm = T)
#Latitude
ulat = mean(outer(birds3$latitude, birds3$latitude, "=="), na.rm = T)

Ruvec = c(udli, udlo, usp, ucom, usex, uage, una, uown, ulong, ulat)
#no species
#uvec = c(udli, udlo, ucom, usex, uage, una, uown, ulong, ulat)

#Finding the agreement and disagreement weights:
agweight = log(mvec/uvec)
disagweight = log((1 - mvec)/(1 - uvec))

#Checking the agreement and disagreement weights are reasonable. i.e. agreements are positve and disagreements are negative.
agweight
disagweight

############Creating our truth vectors
#Date listed:
date_listed = as.vector(outer(birds3$Date.listed, birds3$Date.listed, "=="))
```

```
#Date lost:
date_lost = as.vector(outer(birds3$Date.Lost, birds3$Date.Lost, "=="))
#Owners id
owner_id = as.vector(outer(birds3$Owner..ID, birds3$Owner..ID, "=="))
# #Species
species = as.vector(outer(birds3$Species, birds3$Species, "=="))
#Common name
common_name = as.vector(outer(birds3$Common.name, birds3$Common.name, "=="))
#Pet Name
pet_name =  as.vector(outer(birds3$Pet.Name, birds3$Pet.Name, "=="))
#Sex
sex = as.vector(outer(birds3$Sex, birds3$Sex, "=="))
#age
age = as.vector(outer(birds3$Age, birds3$Age, "=="))
#Longitude
longitude = as.vector(outer(birds3$longitude, birds3$longitude, "=="))
#Latitude
latitude = as.vector(outer(birds3$latitude, birds3$latitude, "=="))

true_truth_matrix = cbind(date_listed, date_lost, owner_id, species, common_name, pet_name, sex, age, longitude, latitude)
#No species
#Creating a truth matrix and removing any observations with weights of NA:
true_truth_matrix = cbind(date_listed, date_lost, owner_id, common_name, pet_name, sex, age, longitude, latitude)
true_truth_matrix[which(is.na(true_truth_matrix))] = 0
length(true_truth_matrix)
length(na.omit(true_truth_matrix))

#Creating a false truth matrix
false_truth_matrix = 1 - cbind(date_listed, date_lost, owner_id, species, common_name, pet_name, sex, age, longitude, latitude)
#No species
false_truth_matrix = 1 - cbind(date_listed, date_lost, owner_id, common_name, pet_name, sex, age, longitude, latitude)
false_truth_matrix[which(is.na(false_truth_matrix))] = 0
```

```
length(false_truth_matrix)
length(na.omit(false_truth_matrix))

#Finding the weights:
weights = sapply(true_truth_matrix*agweight + false_truth_matrix*disagweight, FUN = sum)
#Checking that there are no 'na' values
sum(is.na(weights))

#Plotting the weights:
plot(density(weights))

#Creating our cutoff points and finding the birds that these weights are related to:
nA = nrow(birds3)
lc = 5
first_match = (( which(weights>lc)-1 ) %% nA ) + 1
second_match = floor(( which(weights>lc)-1) /nA) + 1

matches = data.frame(birds3[first_match,], birds3[second_match,])
#Attaching the weights to each observation
matches = cbind(weights[which(weights>lc)], matches)
non_na_matches = matches[-which(is.na(matches$X1.dim.birds3..1..1)),]

non_na_first = non_na_matches[,1:(length(agweight)+2)]
non_na_second = non_na_matches[,(length(agweight)+3):dim(non_na_matches)[2]]

#Massive number of possible duplicates.
#Removing those where the owner ID's are different
matches2 = matches[which(matches$Owner..ID == matches$Owner..ID.1),]
#Removing those where the observatoin numbers are the same:
matches2 = matches2[which(matches2$X1.dim.birds3..1. != matches2$X1.dim.birds3..1..1),]
#Removing matches where species are not the same:
matches2 = matches2[which(matches2$Common.name == matches2$Common.name.1),]
#Removing those where the sex is not the same:
matches2 = matches2[which(matches2$Sex == matches2$Sex.1),]
#Remvong those where the bird names are different:
matches2 = matches2[which(matches2$Pet.Name == matches2$Pet.Name.1),]
```

```
#Now have 64 possible matches, good to check these for duplicates
```

rcode/Fellegi_Sunter_Method_Duplicate_Checking.R

New Zealand Geocoding: Longitude and Latitude

```
# all New Zealand geocode


# replicated from previous
library(jsonlite)
library(xml2)
require(xlsx)
library(readxl)

setwd("H:/Bsc(Hon)/stats 780/New folder")
psuedo_list<- read_excel("H:/Bsc(Hon)/stats 780/New folder/Bird_data_tidy.xlsx",
                         col_types = c("numeric", "date", "date",
                                       "text", "date", "text", "text", "text",
                                       "text", "text", "text", "text", "text",
                                       "text", "text", "text", "text", "text",
                                       "text", "numeric", "text", "text",
                                       "text"))

dim(psuedo_list)
psuedo_list$Country <-"New Zealand"



class(psuedo_list)

table(psuedo_list$Region) # check the frequncy
region_list<-unique(psuedo_list$Region)



psuedo_list$Location <- paste(psuedo_list$Street,psuedo_list$Suburb.Area,psuedo_list$Town,psuedo_list$Region, psuedo_list$Country,sep=",", collapse = NULL)
```

```r
#has NA,trying to remove
psuedo_list$Location <- gsub("NA,","",psuedo_list$Location)
class(psuedo_list$Location)
sort(table(droplevels(psuedo_list$Species)),decreasing = T) #
    done!


#convert location to geocode
library(ggmap)
psuedo_list$geocode <-lapply(psuedo_list$Location,geocode)

# separate lon-lat

for (i in 1:length(psuedo_list$geocode)){
  psuedo_list$longitude[i]<-psuedo_list$geocode[[i]][1]
  psuedo_list$latitude[i]<-psuedo_list$geocode[[i]][2]
}

#trying to fix date
library(lubridate)
psuedo_list$Date.listed1<-as.Date(psuedo_list$Date.listed)
ts$EXAMDATE <- as.Date(ts$EXAMDATE)


psuedo_list1<-psuedo_list[,-c(26)]
write.xlsx(psuedo_list1 , "H:/Bsc(Hon)/stats 780/New folder/
    nzgeocode.xlsx")

table(psuedo_list1$longitude,useNA = "ifany")
psuedo_list2<-na.omit(psuedo_list1$longitude)
```

rcode/nzgeocode.R

Auckland Geocoding: Longitude and Latitude

```r
#unstall package so that

library(jsonlite)
library(xml2)
library(xlsx)
library(xtable)
library(readxl)



#Bird_data_tidy  = read.xlsx("H:/Bsc(Hon)/stats 780/New folder/
```

```
    Bird_data_tidy.xlsx", sheetIndex = 1, header = T)
mydata<- read_excel("H:/Bsc(Hon)/stats 780/New folder/Bird_data_
    tidy.xlsx",
                      col_types = c("numeric", "date", "date",
                                    "text", "date", "text", "text"
    , "text",
                                    "text", "text", "text", "text"
    , "text",
                                    "text", "text", "text", "text"
    , "text",
                                    "text", "numeric", "text", "
    text",
                                    "text"))

class(mydata)

region_list<-unique(mydata$Region) # clustering in regions
# Auckland region


# let's play for a while...

Auckland <- subset(mydata,Region=='Auckland')
Auckland$Location <- paste(Auckland$Street,Auckland$Suburb.Area,
    Auckland$Town,Auckland$Region, sep=",", collapse = NULL)
#has NA,trying to remove
Auckland$Location <- gsub("NA,","",Auckland$Location)
class(Auckland$Location)
sort(table(droplevels(Auckland$Species)),decreasing = T) #done!


#convert location to geocode
library(ggmap)
Auckland$geocode <-lapply(Auckland$Location,geocode)

# separate lon-lat

for (i in 1:length(Auckland$geocode)){
  Auckland$longitude[i]<-Auckland$geocode[[i]][1]
  Auckland$latitude[i]<-Auckland$geocode[[i]][2]
}

#trying to fix date
library(lubridate)
Auckland$Date.listed1<-as.Date(Auckland$Date.listed)
```

```r
ts$EXAMDATE <- as.Date(ts$EXAMDATE)


library(tidyr)

#convert to real location
Auckland$Location1 <- unite(Auckland,Street::Region, sep=",")


#### tesing if this map really works
geocode("Toronto Ontario", output = "latlon" , source = "google")

geocode("Findlay Road,Invercargill,Auckland", output = "latlon", source = "google")
geocode(Auckland$Location[1], output = "latlon" , source = "google")



# Auckland geocode
Auckland1<-Auckland[,-c(25)]

#save them
write.xlsx(Auckland1, "H:/Bsc(Hon)/stats 780/New folder/aklgeocode.xlsx")



###
```

rcode/aklgeocode.R

Data preparation: Adding variables and formatting Date

```r
# remove unwanted columns
# copy dataframe as birds3
birds3 = birds2
head(birds2)
colnames(birds2)

# remove the last two observations where date listed and lost is BOTH NA
birds3 = birds3[(1:(length(birds3$Record.) -2)), ]

# set unwanted columns as NULL to delete them
```

```
birds3$NA..1 = NULL
birds3$NA. = NULL
birds3$Record.= NULL
birds3$FOUND. = NULL
birds3$Date.found = NULL
birds3$Specific.name = NULL
birds3$Family = NULL
birds3$Order = NULL
birds3$Street = NULL
birds3$Suburb.Area = NULL
birds3$Town = NULL
# keep region
# birds3$Region = NULL
birds3$Source = NULL
birds3$Breeder. = NULL
birds3$Ph.Number = NULL
birds3$Additional.Details = NULL
birds3$Location = NULL
birds3$Country = NULL
birds3$Species = NULL
birds3$stringAsFactors = NULL
birds3$Owner..ID = NULL
birds3$Pet.Name = NULL
birds3$Age = NULL
birds3$Sex = NULL

#Finding the indices of the values in Date.Lost that are na:
index_date_lost = which(is.na(birds3$Date.Lost))
birds3$Date.Lost[index_date_lost] = birds3$Date.listed[index_
    date_lost]

#Fixing date listed nas
index_date_listed = which(is.na(birds3$Date.listed))
birds3$Date.listed[index_date_listed] = birds3$Date.Lost[index_
    date_listed]

# removes observation 82 as it has NA for both date listed and
    lost
birds3 = birds3[-82,]

# drop date.listed
birds3$Date.listed = NULL

# change observation number 1329 for region (Brooklands = in New
    Plymouth/Taranaki)
```

```
birds3$Region[1328] <- unique(birds3$Region)[1]
unique(birds3$Region)

# edit "Region" to be consistent
regions <- unique(birds3$Region)
regions

# Tasman = 15
birds3$Region[which(birds3$Region == regions[19])] = "Tasman"
birds3$Region[which(birds3$Region == regions[25])] = "Tasman"

# Northland
birds3$Region[which(birds3$Region == regions[23])] = "Northland"

regions <- unique(birds3$Region)
regions <- droplevels(regions)


# rename columns
names(birds3)
# "Common.name" -> "species"
names(birds3)[names(birds3) == "Common.name"] <- "Species"

# "Escape.type" -> "NumberBirds"
names(birds3)[names(birds3) == "Escape.type"] <- "NumberBirds"

#### just to tidy up
names(birds3)[names(birds3) == "Record."] <- "Record"
names(birds3)[names(birds3) == "latitude"] <- "Latitude"
names(birds3)[names(birds3) == "longitude"] <- "Longitude"
names(birds3)

# add columns
TrapChecked <- rep("Y", nrow(birds3))
TrapName <- 1:nrow(birds3)

birds4 <- cbind(birds3, TrapChecked, TrapName)

names(birds4)

# checking regions again
regionb4 <- unique(birds4$Region)
regionb4 <- droplevels(regionb4)

# change the date
```

```r
# change date format to dd-mm-yyyy [still character string]
birds4 <- cbind(birds3, TrapChecked, TrapName)
birds4$Date.Lost <- as.Date(birds4$Date.Lost, "%Y-%m-%d")
birds4$Date.Lost <- format(birds4$Date.Lost, "%d-%m-%Y")

birds4$Date.Lost[1]
class(birds4$Date.Lost)

# remove observation in 2010
birds4 <- birds4[-which(birds4$Date.Lost == "17-01-2010"), ]
birds4 <- birds4[-which(birds4$Date.Lost == "03-10-2010"), ]

# removed bird listing dated 27/11/2017 (has not occured yet at date of collection)
birds4 <- birds4[-which(birds4$Date.Lost == "27-11-2017"), ]

# export data.frame birds4 as csv file
write.csv(birds4, "bird_dataset_for_heatmap.csv")

# table of species
# by order of 'Species'
species = count(birds4$Species)
colnames(species) = c("Species", "Frequency")

# by order of 'frequency numbers'
speciestable = species[order(-species[,2]),]

# density
plot(density(birds4$Latitude))
plot(density(birds4$Longitude))
```

rcode/finadata.R

Time Series Generation

```r
# read final dataset csv file
birds2 = read.csv("bird_dataset_for_heatmap.csv", header = T)
colnames(birds2)
date = birds2$Date.Lost

# create empty vectors ready for our monthly escapes per year
cool2012 = c()
cool2013 = c()
cool2014 = c()
cool2015 = c()
cool2016 = c()
```

```r
cool2017 = c()

# function to get number of escapes per month and year
for (i in 1:12) {

  #2012
  #check2012 <- grep(paste(i,-2012, sep = ""), date)
  if (length(grep(paste(i, -2012, sep = ""), date)) == 0) {
    cool2012[i] = 0
  } else {
    cool2012[i] = sum(birds2$NumberBirds[grep(paste(i, -2012,
   sep = ""), date)])
  }
  cool2012

  # 2013
  #check2013 <- grep(paste(i,-2013, sep = ""), date)
  if (length(grep(paste(i, -2013, sep = ""), date)) == 0) {
    cool2013[i] = 0
  } else {
    cool2013[i] = sum(birds2$NumberBirds[grep(paste(i, -2013,
   sep = ""), date)])
  }
  cool2013

  # 2014
  #check2014 <- grep(paste(i,-2014, sep = ""), date)
  if (length(grep(paste(i, -2014, sep = ""), date)) == 0) {
    cool2014[i] = 0
  } else {
    cool2014[i] = sum(birds2$NumberBirds[grep(paste(i, -2014,
   sep = ""), date)])
  }
  cool2014

  # 2015
  #check2015 <- grep(paste(i,-2015, sep = ""), date)
  if (length(grep(paste(i, -2015, sep = ""), date)) == 0) {
    cool2015[i] = 0
  } else {
    cool2015[i] = sum(birds2$NumberBirds[grep(paste(i, -2015,
   sep = ""), date)])
  }
  cool2015
```

```r

  # 2016
  #check2016 <- grep(paste(i,-2016, sep = ""), date)
  if (length(grep(paste(i, -2016, sep = ""), date)) == 0) {
    cool2016[i] = 0
  } else {
    cool2016[i] = sum(birds2$NumberBirds[grep(paste(i, -2016, sep = ""), date)])
  }
  cool2016

  # 2017
  #check2017 <- grep(paste(i,-2017, sep = ""), date)
  if (length(grep(paste(i, -2017, sep = ""), date)) == 0) {
    cool2017[i] = 0
  } else {
    cool2017[i] = sum(birds2$NumberBirds[grep(paste(i, -2017, sep = ""), date)])
  }
  cool2017
}

##### create data.frame with year/month/# of listings
# column for years
year <- 2012:2017
year <- rep(year, each = 12)

# column for month
month <- rep(1:12, 6)

# column for all listings
listings <- c(cool2012, cool2013, cool2014, cool2015, cool2016, cool2017)
listings.df <- as.data.frame(cbind(year, month, listings))

# remove month-year at beginning and ending of time series that has 0 listings
# 1:5 = 0
# 65:72 = 0
listings.df <- listings.df[-c(1:5, 65:72),]

# starts june 2012
# ends april 2017

# create time series object
```

```r
lists <- listings.df$listings

listings.df$series = ts(lists, frequency=12, start=c(2012,6), end=c(2017,4))
listings.ts = ts(listings.df$series, frequency=12, start=c(2012,6), end=c(2017,4))

# plot the listings
plot(listings.ts, main="Bird escapes", xlab = "Year", ylab = "Number of Lost Bird Escapes")

# decompose time series into seasonal, trend and irregular components
decomp.birds.stl <- stl(listings.ts, s.window = "periodic")
plot(decomp.birds.stl)
title(main = "Decomposition of Bird Escapes", line = 2.5, cex.main = 1.5)


# checking autocorrelation of residuals
# few slight significant autocorrelations at lag 3 and 12 but not of large concern
acf(decomp.birds.stl$time.series[,3], main = "ACF of Residuals")

# checking autocorrelation of all listings
# see clear pattern in original data -> seasonal component
acf(lists, main = "ACF of Bird listings (original data)")


#############################################

#cool#### is a vector of the number of listings per month for the year ####
for(i in 1:12){
  cool2012[i] = length(grep(paste(i,-2012, sep = ""), date))
  cool2013[i] = length(grep(paste(i,-2013, sep = ""), date))
  cool2014[i] = length(grep(paste(i,-2014, sep = ""), date))
  cool2015[i] = length(grep(paste(i,-2015, sep = ""), date))
  cool2016[i] = length(grep(paste(i,-2016, sep = ""), date))
  cool2017[i] = length(grep(paste(i,-2017, sep = ""), date))
}
```

rcode/birdtimeseries.R