

Comparing Bayesian and frequentist estimates of lost animal recovery rates in Vancouver

Anne Nguyen

Introduction

Dogs and cats are a popular choice of companion animal in households. Across Canada, the Canadian Animal Health Institute (CAHI) estimates 60% of Canadian households have ownership of at least one dog or cat with total dog populations around 7.9 million and cat populations around 8.5 million (1). Given these numbers, pets are lost from homes regularly however how often they are reunited with their owner is unclear. These animals can return on their own, be returned by other people, or enter animal sheltering systems. A 2012 paper by Weiss et al. estimates recovery rates of 91% for dogs and 75% for cats in the United States based on data collected from surveys. Using animal control data from the city of Vancouver (3), this report aims to compare Bayesian and frequentist estimates of cat and dog recovery rates in Vancouver.

Materials and Methods

The Animal Control Lost and Found data set from the city of Vancouver contains information on animals that are reported lost to the Vancouver Animal Control Office. Variables include the date the animal was lost, the type of animal or best fit breed description, color of the coat, name of animal, sex of the animal, and state of whether they are “lost”, “found”, or “matched”. The original data contains 17628 observations.

For this report, only cats and dogs are of interest and whether they have been found. In this case, an animal was identified as a cat or dog based the breed/type description and indicated with a binary variable where 1 is when the state was labelled “found” or “matched” and 0 otherwise. After processing, there were 17088 observations.

The data is first split where 80% of the data is in the training set and the remaining 20% in the test set. MCMC conducted using Stan was used to compute the Bayesian estimate of the animal recovery rate with 10000 iterations. Bootstrapping was used to compute the frequentist estimate of the animal recovery rate with 10000 replicates.

Data Analysis

The model is specified below for $i \in \{Cat, Dog\}$.

$$a_i \sim Exp(1)$$

$$b_i \sim Exp(1)$$

$$p_i \sim Beta(a_i, b_i)$$

$$Y_i | p_i \sim Binom(n, p_i)$$

As recovery rate is a probability, a beta distribution was selected for the distribution. A uniform prior of $p_i \sim Unif(0, 1)$ was also considered which is equivalent to $p_i \sim Beta(1, 1)$, but the shape parameters were parameterized to $Exp(1)$ for some flexibility. The binomial distribution was selected for the likelihood as each animal can be considered an independent trial. This is with the assumption of the data that an

animal is recorded only once. Separate models for cats and dogs are chosen as recovery rates and methods of searching vary in recorded literature (2).

Following the fit of the model with Stan, trace plots were created to check for non-convergence and insufficient mixing. One of the plots is shown below for the parameter corresponding to the recovery rate for cats. The chains do not differ much from each other and bounce around a stable mean. Trace plots for other parameters are similar. There appears to be no apparent issues.

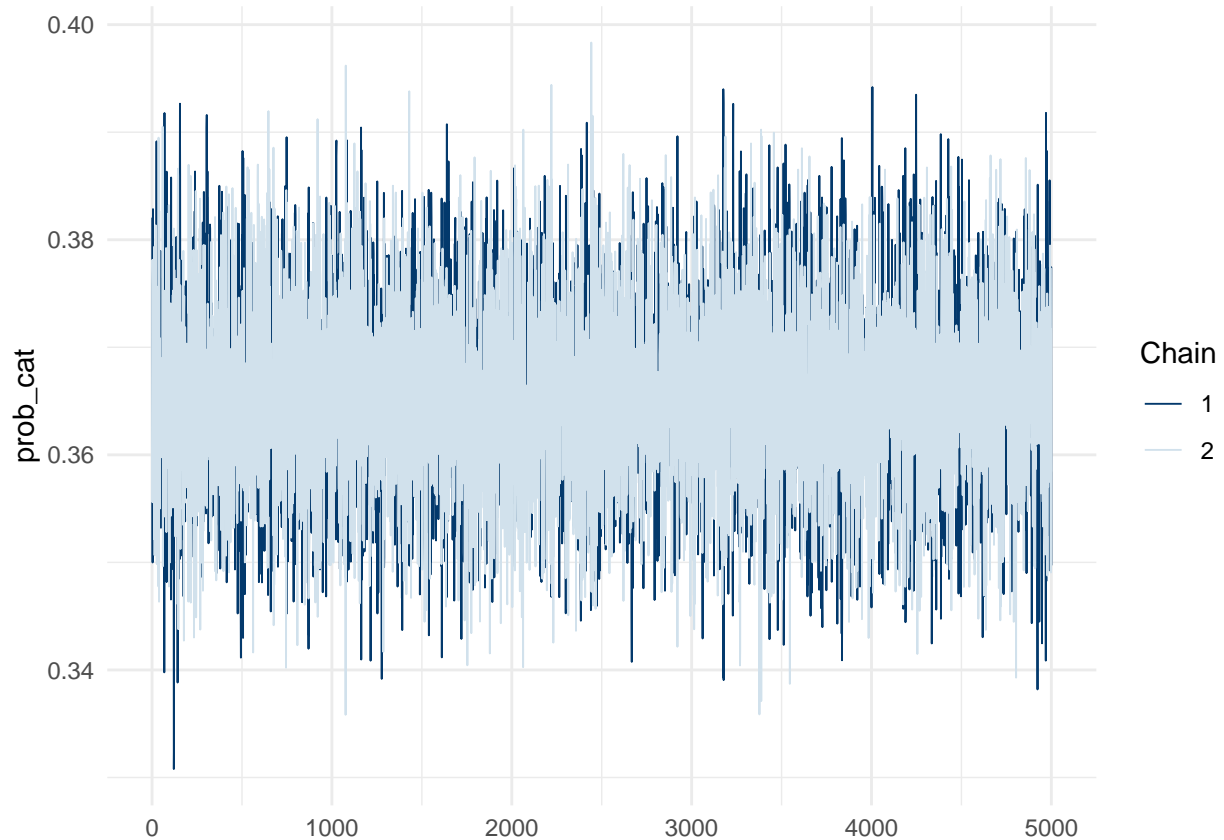


Figure 1: Trace plot for cat recovery rate.

Table 1: MCMC fit summary for model.

	mean	se_mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
prob_cat	0.3656	0.0001	0.0088	0.3484	0.3597	0.3655	0.3715	0.3831	13506	1.0001
prob_dog	0.5380	0.0000	0.0048	0.5285	0.5348	0.5380	0.5413	0.5475	14570	0.9999
a_cat	1.0615	0.0075	0.7934	0.1162	0.4863	0.8726	1.4297	3.0927	11256	0.9999
b_cat	1.3778	0.0097	1.0449	0.1333	0.6059	1.1089	1.8835	3.9962	11601	1.0003
a_dog	1.2634	0.0090	0.9436	0.1360	0.5790	1.0320	1.7096	3.5977	10938	1.0000
b_dog	1.1818	0.0087	0.9014	0.1106	0.5209	0.9647	1.6031	3.4928	10821	1.0000
lp__	-	0.0292	1.8393	-	-	-	-	-	3969	1.0008
	9342.2797			9346.8090	9343.2568	9341.9403	9340.9441	9339.7533		

From the fit summary, the effective sample sizes are larger than the number of iterations indicating the draws may be anticorrelated. Rhat, the potential scale reduction factor on split chains, is around 1 which together with the trace plot supports convergence.

Below is a table containing the 95% credible intervals from MCMC and the 95% confidence intervals from MCMC and bootstrapping.

Table 2: 95% credible intervals (MCMC) and 95% confidence intervals (MCMC and bootstrap).

Method	Animal	Mean	Lower(2.5%)	Upper(97.5%)	Width
Bootstrap	Cat	0.3657	0.3484	0.3828	0.0344
MCMC - Credible	Cat	0.3656	0.3484	0.3826	0.0343
MCMC - Confidence	Cat	0.3656	0.3481	0.3830	0.0349
Bootstrap	Dog	0.5381	0.5286	0.5478	0.0192
MCMC - Credible	Dog	0.5380	0.5285	0.5476	0.0191
MCMC - Confidence	Dog	0.5380	0.5282	0.5478	0.0196

Bootstrapping has higher estimates of the recovery rate than the MCMC results however the estimates for both method are relatively close. MCMC confidence intervals are wider than the intervals for bootstrapping while the credible intervals are the smallest. Using the test data, recovery rate estimates of 0.3668 and 0.5329 for cats and dogs respectively are captured in the intervals.

Conclusion

Both Bayesian and frequentist estimates of recovery rates are similar, with dogs having a recovery rate of around 54% which is higher than that of cats with a rate of 37%. This falls in line with the general trend in previous studies however rates are much lower. This may be due to limitations of the data where only animals reported loss are recorded and some loss of quality due to data entry errors. Further exploration could include methods of searching for cats (4) and dogs (5) and their contribution to recovery rates. Regarding methods, more thorough checks of the model such as prior/posterior predictive check could be done using generated quantities as this report is limited by the computational resources.

References

1. Canadian Animal Health Institute. (2022, September 22). 2022 - Latest Canadian Pet Population Figures Released | Press Releases. Cahi-Icsa.ca. <https://cahi-icsa.ca/press-releases/2022-latest-canadian-pet-population-figures-released>
2. Weiss, E., Slater, M., & Lord, L. (2012). Frequency of Lost Dogs and Cats in the United States and the Methods Used to Locate Them. *Animals*, 2(2), 301-315. <https://doi.org/10.3390/ani2020301>
3. Animal control inventory - lost and found. (2025, March 13). City of Vancouver Public Data. <https://opendata.vancouver.ca/explore/dataset/animal-control-inventory-lost-and-found/information/>
4. Huang, L., Coradini, M., Rand, J., Morton, J., Albrecht, K., Wasson, B., & Robertson, D. (2018). Search Methods Used to Locate Missing Cats and Locations Where Missing Cats Are Found. *Animals*, 8(1), 5. <https://doi.org/10.3390/ani8010005>
5. Lord LK, Wittum TE, Ferketich AK, Funk JA, Rajala-Schultz PJ. Search and identification methods that owners use to find a lost dog. *J Am Vet Med Assoc*. 2007 Jan 15;230(2):211-6. doi: 10.2460/javma.230.2.211. PMID: 17223753.

Appendix

Repository

https://github.com/a-nguyen-1/Stat447C_Project

Data processing

```
# Read in data
data <- read.csv("animal-control-inventory-lost-and-found.csv", sep = ";")

# Key strings to ID cats
cat = "cat|DSH|DLH|DMH|tabby|hair|domestic|siam|calico|coon|russian|persian|abyss|tux|kitt|norwegian|ab

# Key strings to ID birds
bird = "bird|cocka|budg|dove|parrot|finch|conure|congo|ringneck|pigeon|duck|hen|keet|peacock"

# Key strings to ID dogs
dog = "poo|dog|terr|coll|pit|dog|mix|hound|shep|bern|dood|malt|york|tzu|beag|husk|lab|aki|corg|dach|ame

# Key strings to ID reptiles (combined since there are few)
reptile = "snake|reptile|frog|lizard|boa|dragon|python|turtle"

# Identify animal type
data2 = data %>%
  mutate(Animal = case_when(grepl(cat, Breed, ignore.case=TRUE) ~ "Cat",
                             grepl(bird, Breed, ignore.case=TRUE) ~ "Bird",
                             grepl(reptile, Breed, ignore.case=TRUE) ~ "Reptile",
                             grepl("rabb|lop", Breed, ignore.case=TRUE) ~ "Rabbit",
                             grepl("ferr", Breed, ignore.case=TRUE) ~ "Ferret",
                             grepl("hamster", Breed, ignore.case=TRUE) ~ "Hamster",
                             grepl("X", Breed) ~ "Dog",
                             grepl(dog, Breed, ignore.case=TRUE) ~ "Dog",
                             .default = "Unknown"),
         Found = ifelse(State == "Matched" | State == "Found", 1, 0))

# Filter only cats and dogs
cat_data = data2 %>%
  filter(Animal == "Cat")

dog_data = data2 %>%
  filter(Animal == "Dog")

#Create training and test sets
set.seed(447)
sample <- sample(c(TRUE, FALSE), nrow(cat_data), replace=TRUE, prob=c(0.8,0.2))
cat_train <- cat_data[sample, ]
cat_test <- cat_data[!sample, ]

sample <- sample(c(TRUE, FALSE), nrow(dog_data), replace=TRUE, prob=c(0.8,0.2))
dog_train <- dog_data[sample, ]
dog_test <- dog_data[!sample, ]
```

Stan code

```

data {
  int<lower=0> n1; // Number of cats reported lost to the Vancouver Animal Control Office
  int<lower=0> n2; // Number of dogs reported lost to the Vancouver Animal Control Office
  int<lower=0,upper=n1> k1; // Number of cats found or matched
  int<lower=0,upper=n2> k2; // Number of dogs found or matched
}

parameters {
  real<lower=0,upper=1> prob_cat_found; // Probability of cat being found or matched
  real<lower=0,upper=1> prob_dog_found; // Probability of dog being found or matched
  real<lower=0> a_cat; // Shape parameter alpha (cat)
  real<lower=0> b_cat; // Shape parameter beta (cat)
  real<lower=0> a_dog; // Shape parameter alpha (dog)
  real<lower=0> b_dog; // Shape parameter beta (dog)
}

model {
  // prior
  a_cat ~ exponential(1);
  b_cat ~ exponential(1);
  a_dog ~ exponential(1);
  b_dog ~ exponential(1);
  prob_cat_found ~ beta(a_cat,b_cat);
  prob_dog_found ~ beta(a_dog,b_dog);

  // likelihood
  k1 ~ binomial(n1, prob_cat_found);
  k2 ~ binomial(n2, prob_dog_found);
}

```

Model fit

```

# Fit model
fit = stan(
  seed = 447,
  file = "animal2.stan",
  chains = 2,
  data = list(n1 = nrow(cat_train),
              n2 = nrow(dog_train),
              k1 = sum(cat_train$Found == 1),
              k2 = sum(dog_train$Found == 1)),
  iter = 10000
)

summary(fit)

```

Bootstrap

```

set.seed(447)
# Bootstrap with 10000 replicates for each animal
b_cat = boot(data = cat_train$Found,
             statistic = function(x,i) mean(x[i]), R = 10000)
cat_ci = boot.ci(b_cat)

```

```
b_dog = boot(data = dog_train$Found,  
             statistic = function(x,i) mean(x[i]), R = 10000)  
  
dog_ci = boot.ci(b_dog)
```

Intervals