LA County Shelter Analysis

Dominique Akinyemi

2024-10-01

Improving Outcomes for High-Risk Dog Breeds in Los Angeles County Shelters

Project Overview

Los Angeles County animal shelters are over capacity. At the time of this project, they are 180% over capacity. This project will analyze shelter data to identify which dog breeds contribute most to overpopulation and have the lowest adoption rates, longest stays, or highest risk of euthanasia.

The goal is to use this analysis to develop actionable strategies for increasing adoption rates and reducing shelter overpopulation.

Skills

- Data cleaning
- Exploratory data analysis
- Trend analysis
- Data visualization

Tools

- R

Data Sources

For this project, I used the "Animal Care PawStats Data" provided by Los Angeles County on their open data website. The data set can be accessed here.

The data is frequently updated and maintained, with the most recent update being September 3, 2024 (as of the project date). It includes over 300,000 records (including all animal types).

The data set includes the following attributes for each intake entry: facility, animal ID, animal type, animal breed, impound no, admission fiscal year, intake type, intake group, outcome fiscal year, outcome type, outcome group, intake date, and outcome date.

Data Cleaning and Preparation

To prepare for cleaning and analysis, I installed and loaded the following R packages:

```
install.packages(c("tidyverse", "lubridate", "janitor", "skimr"))
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lubridate)
library(janitor)
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(skimr)
```

And imported the data set.

```
shelter_animals <- read_csv("/./Volumes/X9 Pro/Shelter Analysis v2/Animal_Care_PawStats_Data.csv")
```

```
## Rows: 313291 Columns: 14
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (13): FACILITY, ANIMAL_ID, ANIMAL_GROUP, PRIMARY_BREED, IMPOUND_NO, Inta...
## dbl  (1): ObjectId
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Then, I performed the following cleaning tasks:

1. Removed cats and other types of animals from data set. It now contains 142,201 dogs.

```
shelter_dogs <- shelter_animals[shelter_animals$ANIMAL_GROUP == "DOGS", ]
```

2. Standardized capitalization of column names and data entries (to lowercase).

```
shelter_dogs <- shelter_dogs %>%
  mutate(across(where(is.character), tolower))
colnames(shelter_dogs) <- tolower(colnames(shelter_dogs))
```

3. Removed redundant columns.
    - animal_group: no longer needed as the only data left in this column is dogs.
    - intake_fiscal_year, outcome_fiscal_year: information needed already contained in date columns.
    - objectid: not needed as rows are numbered.

```
shelter_dogs <- shelter_dogs %>%
  select(-intake_fiscal_year, -outcome_fiscal_year, -objectid)
```

4. Changed data column types from character to date (removing time).

```
shelter_dogs$intake_date <- as.Date(shelter_dogs$intake_date)
shelter_dogs$outcome_date <- as.Date(shelter_dogs$outcome_date)
```

5. Summarized data to identify further cleaning needed.

```
skim(shelter_dogs)
```

Table 1: Data summary

Name	shelter_dogs
Number of rows	142201
Number of columns	11
Column type frequency:	
character	9
Date	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
facility	0	1.00	6	14	0	7	0
animal_id	0	1.00	8	8	0	125844	0
animal_group	0	1.00	4	4	0	1	0
primary_breed	0	1.00	3	18	0	221	0
impound_no	0	1.00	10	10	0	142201	0
intake_type	0	1.00	5	10	0	23	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
intake_type_group	5878	0.96	5	9	0	4	0
outcome_type	0	1.00	3	10	0	23	0
outcome_type_group	30624	0.78	3	10	0	8	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
intake_date	0	1.00	2010-10-21	2024-08-29	2021-07-04	2361
outcome_date	1034	0.99	2007-08-14	2024-08-29	2021-07-08	2253

6. Investigated and resolved 1,034 missing values in outcome_ date column. Removed unknown outcomes and outcome types.

```
shelter_dogs <- shelter_dogs %>%
  filter(!is.na(outcome_date) & outcome_type != "unk9999999")
```

7. Investigated and resolved missing values in intake_type_group and outcome_type_group columns.
    - After inspecting these columns and comparing them to the intake_type and outcome_type columns, I found they were redundant groupings for intake/outcome classifications. I removed these columns from the data.

```
shelter_dogs <- shelter_dogs %>%
  select(-intake_type_group, -outcome_type_group)
```

8. Confirmed that there were no longer any missing values from columns.

```
skim(shelter_dogs)
```

Table 4: Data summary

Name	shelter_dogs
Number of rows	141159
Number of columns	9
Column type frequency:	
character	7
Date	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
facility	0	1	6	14	0	7	0
animal_id	0	1	8	8	0	124973	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
animal_group	0	1	4	4	0	1	0
primary_breed	0	1	3	18	0	221	0
impound_no	0	1	10	10	0	141159	0
intake_type	0	1	5	10	0	23	0
outcome_type	0	1	3	10	0	22	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
intake_date	0	1	2010-10-21	2024-08-29	2021-06-24	2361
outcome_date	0	1	2007-08-14	2024-08-29	2021-07-08	2253

9. Standardized names in intake_type and outcome_type columns.

```
shelter_dogs <- shelter_dogs %>%
  mutate(
    intake_type = recode(intake_type,
                         "dispo req" = "disposal_required",
                         "emer evac" = "emergency_evacuation",
                         "owner sur" = "owner_surrender",
                         "return" = "returned_to_shelter",
                         "pd request" = "police_dept_request",
                         "owner died" = "owner_died",
                         "hospitaliz" = "owner_hospitalized",
                         "trans_int" = "internal_transfer",
                         "arrested" = "owner_arrested",
                         "confiscate" = "confiscated",
                         "ani-safe" = "animal_safety",
                         "cola hi" = "cola_hi_program",
                         "bornincare" = "born_in_care",
                         "finder" = "found",
                         "court case" = "court_case",
                         "danger dog" = "danger_dog",
                         "sn board" = "spay_neuter_board",
                         "trans_ext" = "external_transfer",
                         "cool cen" = "cooling_center"
    ),
    outcome_type = recode(outcome_type,
                         "rto micro" = "returned_to_owner",
                         "rto" = "returned_to_owner",
                         "dead" = "deceased_on_arrival",
                         "disposal" = "deceased_on_arrival",
                         "euth" = "euthanized",
                         "adoption" = "adopted",
                         "rto fldmic" = "returned_to_owner",
                         "foster" = "fostered",
                         "died" = "died_in_care",
                         "rto fld id" = "returned_to_owner",
                         "close2home" = "rescue",
                         "rtn" = "returned_to_owner",
```

```
                          "trans_int" = "internal_tranfer",
                          "trans_ext" = "external_transfer",
                          "rtc" = "returned_to_owner",
                          "aspcapw" = "rescue",
                          "aspcatrans" = "rescue",
                          "cool cen" = "cooling_center"
    )
  )
```

10. Removed dogs already deceased upon intake, and those whose stays were transient/temporary.

```
shelter_dogs <- shelter_dogs %>%
  filter(!intake_type %in% c("disposal_required", "animal_safety", "cola_hi_program", "danger_dog", "spa
shelter_dogs <- filter(shelter_dogs, outcome_type != "deceased_on_arrival")
```

11. Investigated date discrepancies to find errors.
    - Skimr tibble shows min intake date is from 2010 and min outcome date is from 2007. Looking at the data set, these are obvious errors, so I will remove the entries.

```
shelter_dogs <- shelter_dogs[!(shelter_dogs$outcome_date < shelter_dogs$intake_date), ]
shelter_dogs <- shelter_dogs %>%
  filter(intake_date != as.Date("2010-10-21"))
```

12. Compared animal_id and impound_no to resolve duplicates.
    - There are 124,973 unique animal_ids and 141,159 unique impound_no out of 141,159 total data entries. This confirms there are no duplicate values in the impound_no column, but there are duplicate animal_ids.
    - All of the repeated animal_ids have the same primary_breed but different dates, so I can confirm that they are multiple shelter intakes rather than duplicate animal_ids.

```
repeated_animal_ids <- shelter_dogs %>%
  group_by(animal_id) %>%
  filter(n() > 1) %>%
  ungroup()
```

13. Standardized breed names through mapping.

    - I used American Kennel Club's official list as a reference but occasionally chose other breed names that are commonly used or are regionally/internationally recognized.

```
unique_breeds <- unique(shelter_dogs$primary_breed)
print(unique_breeds)
```

```
##   [1] "chihuahua sh"       "staffordshire"      "min pinscher"
##   [4] "lhasa apso"         "labrador retr"      "collie smooth"
##   [7] "amer eskimo"        "maltese"            "poodle min"
##  [10] "parson russ ter"    "germ shepherd"      "shih tzu"
##  [13] "border collie"      "pointer"            "rhod ridgeback"
##  [16] "aust shepherd"      "shiba inu"          "terrier"
```

```
##  [19] "pit bull"           "rottweiler"         "mastiff"
##  [22] "cocker span"        "cairn terrier"      "beagle"
##  [25] "dachshund lh"       "chihuahua lh"       "golden retr"
##  [28] "norfolk terrier"    "pug"                "boxer"
##  [31] "eng bulldog"        "plott hound"        "aust cattle dog"
##  [34] "jack russ terr"     "dachshund"          "bichon frise"
##  [37] "siberian husky"     "cavalier span"      "eng sprngr span"
##  [40] "anatol shepherd"    "boston terrier"     "chinese crested"
##  [43] "rat terrier"        "oldeng sheepdog"    "chow chow"
##  [46] "cane corso"         "amer bulldog"       "yorkshire terr"
##  [49] "alask malamute"     "schnauzer giant"    "american staff"
##  [52] "havanese"           "border terrier"     "schnauzer min"
##  [55] "germ sh point"      "old eng bulldog"    "whippet"
##  [58] "welsh corgi pem"    "poodle stnd"        "catahoula"
##  [61] "basenji"            "doberman pinsch"    "silky terrier"
##  [64] "pomeranian"         "chinese sharpei"    "great pyrenees"
##  [67] "manchester terr"    "basset hound"       "great dane"
##  [70] "french bulldog"     "queensland heel"    "pekingese"
##  [73] "neapolitan mast"    "bull terrier"       "flat coat retr"
##  [76] "fila"               "munsterlander"      "keeshond"
##  [79] "tibetan mastiff"    "alaskan husky"      "west highland"
##  [82] "aust kelpie"        "dachshund wh"       "am pit bull ter"
##  [85] "papillon"           "schnauzer stand"    "eng foxhound"
##  [88] "poodle toy"         "toy fox terrier"    "akita"
##  [91] "eng setter"         "tibetan span"       "tibetan terr"
##  [94] "eng coonhound"      "eng toy spaniel"    "belg malinois"
##  [97] "st bernard rgh"     "wheaten terr"       "bruss griffon"
## [100] "fox terr wire"      "dutch shepherd"     "dogo argentino"
## [103] "bernese mtn dog"    "schipperke"         "welsh terrier"
## [106] "lowchen"            "shetld sheepdog"    "samoyed"
## [109] "carolina dog"       "boerboel"           "jindo"
## [112] "welsh corgi car"    "aust terrier"       "mex hairless"
## [115] "vizsla"             "weimaraner"         "ns duck tolling"
## [118] "greyhound"          "norw elkhound"      "bulldog"
## [121] "dalmatian"          "tenn tr brindle"    "amer foxhound"
## [124] "pharaoh hound"      "irish wolfhound"    "dogue de bordx"
## [127] "eng pointer"        "alask klee kai"     "airedale terr"
## [130] "canaan dog"         "bullmastiff"        "armenian gampr"
## [133] "black/tan hound"    "redbone hound"      "ital greyhound"
## [136] "black mouth cur"    "clumber span"       "bloodhound"
## [139] "newfoundland"       "bluetick hound"     "scot terrier"
## [142] "tr walker hound"    "port water dog"     "wolf hybrid"
## [145] "patterdale terr"    "irish terrier"      "brittany"
## [148] "american bully"     "bearded collie"     "spaniel"
## [151] "shepherd"           "eng cocker span"    "st bernard smth"
## [154] "welsh spr span"     "skye terrier"       "bouv flandres"
## [157] "collie rough"       "belg tervuren"      "leonberger"
## [160] "span water dog"     "boykin span"        "bull terr min"
## [163] "spinone ital"       "germ wh point"      "presa canario"
## [166] "chesa bay retr"     "landseer"           "japanese chin"
## [169] "picardy sheepdg"    "field spaniel"      "coton de tulear"
## [172] "belg sheepdog"      "podengo pequeno"    "sussex span"
## [175] "formosan mtn"       "fox terr smooth"    "german pinscher"
## [178] "saluki"             "finnish spitz"      "irish setter"
```

```
## [181] "swiss hound"        "karelian bear"      "affenpinscher"
## [184] "kangal"             "komondor"           "dutch sheepdog"
## [187] "puli"               "norwich terrier"    "caucasian mountain"
## [190] "entlebucher"        "dandie dinmont"     "otterhound"
## [193] "harrier"            "afghan hound"       "ibizan hound"
## [196] "eskimo"             "briard"             "treeing cur"
## [199] "akbash"             "gr swiss mtn"       "hovawart"
## [202] "lakeland terr"      "blue lacy"          "norw buhund"
## [205] "eng shepherd"       "sealyham terr"      "glen of imaal"
## [208] "swed vallhund"      "beauceron"          "maremma sheepdg"
## [211] "pbgv"               "polish lowland"     "kuvasz"
## [214] "tosa"               "curlycoat retr"     "eurasier"
```

```
breed_mapping <- tibble(
  original_breed = c("chihuahua sh","staffordshire","min pinscher","lhasa apso","labrador retr","collie

  standardized_breed = c("chihuahua","bull_terrier","miniature_pinscher","lhasa_apso","labrador_retriev
)
```

```
shelter_dogs <- shelter_dogs %>%
  left_join(breed_mapping, by = c("primary_breed" = "original_breed")) %>%
  mutate(primary_breed = coalesce(standardized_breed, primary_breed)) %>%
  select(-standardized_breed)
```

```
unique_breeds <- unique(shelter_dogs$primary_breed)
```

14. Organized unique breeds and counts in a data frame.

```
unique_breeds <- data.frame(primary_breed = unique_breeds, stringsAsFactors = FALSE)
unique_breeds <- shelter_dogs %>%
  count(primary_breed, name = "counts") %>%
  arrange(desc(counts))
```

15. Backed up cleaned data set in a CSV file before beginning analysis.

```
write.csv(shelter_dogs, file = "/./Volumes/X9 Pro/Shelter Analysis v2/shelter_dogs.csv", row.names = FA
```

Exploratory Analysis of Shelter Data Set

To begin my analysis, I answered the following questions to get an overview of the data.

Which breeds have the largest populations in shelters?

```
head(unique_breeds, 10)
```

```
## # A tibble: 10 x 2
##    primary_breed         counts
##    <chr>                  <int>
##  1 bull_terrier           20075
##  2 german_shepherd        18251
```

```
##  3 chihuahua              15016
##  4 terrier                10936
##  5 siberian_husky          9594
##  6 labrador_retriever      5938
##  7 miniature_poodle        3323
##  8 bulldog                 2607
##  9 maltese                 2485
## 10 australian_cattle_dog   1831
```

Intake and Outcome Types

What are the most common intake reasons?

```
intake_types <- shelter_dogs %>%
  group_by(intake_type) %>%
  summarise(count = n()) %>%
  arrange(desc(count))
head(intake_types, 5)
```

```
## # A tibble: 5 x 2
##   intake_type         count
##   <chr>               <int>
## 1 stray               74699
## 2 owner_surrender     29506
## 3 returned_to_shelter  3995
## 4 transfer             3514
## 5 police_dept_request  1571
```

Note: After strays, dogs surrendered by their owners are the most common intakes in shelters. Why are so many owners surrendering their dogs?

What are the most common outcomes?

```
outcome_types <- shelter_dogs %>%
  group_by(outcome_type) %>%
  summarise(count = n()) %>%
  arrange(desc(count))
head(outcome_types, 5)
```

```
## # A tibble: 5 x 2
##   outcome_type      count
##   <chr>             <int>
## 1 adopted           49394
## 2 returned_to_owner 22775
## 3 rescue            20903
## 4 euthanized        16986
## 5 transfer           4757
```

Outcome Rates

What percentage of dogs were adopted?

```r
adoption_rate <- (sum(shelter_dogs$outcome_type == "adopted") / nrow(shelter_dogs)) * 100
print(adoption_rate)
```

```
## [1] 41.74294
```

What percentage of dogs were euthanized?

```r
euthanasia_rate <- (sum(shelter_dogs$outcome_type == "euthanized") / nrow(shelter_dogs)) * 100
print(euthanasia_rate)
```

```
## [1] 14.35489
```

What percentage of dogs were sent to rescues?

```r
rescue_rate <- (sum(shelter_dogs$outcome_type == "rescue") / nrow(shelter_dogs)) * 100
print(rescue_rate)
```

```
## [1] 17.66515
```

What percentage of dogs were returned to owner?

```r
returned_to_owner_rate <- (sum(shelter_dogs$outcome_type == "returned_to_owner") / nrow(shelter_dogs)) *
print(returned_to_owner_rate)
```

```
## [1] 19.24718
```

Rates By Breed

Adoption rates by breed?

```r
breed_adoption_rates <- aggregate(outcome_type ~ primary_breed, data = shelter_dogs,
                                  function(x) mean(x == "adopted") * 100)
names(breed_adoption_rates)[names(breed_adoption_rates) == "outcome_type"] <- "adoption_rate"
```

Euthanasia rates by breed?

```r
breed_euthanasia_rates <- aggregate(outcome_type ~ primary_breed, data = shelter_dogs,
                                    function(x) mean(x == "euthanized") * 100)
breed_euthanasia_rates <- breed_euthanasia_rates %>%
  rename(euthanasia_rate = outcome_type)
```

Rescue rates by breed?

```r
breed_rescue_rates <- aggregate(outcome_type ~ primary_breed, data = shelter_dogs,
                                function(x) mean(x == "rescue") * 100)
breed_rescue_rates <- breed_rescue_rates %>%
  rename(rescue_rate = outcome_type)
```

What are the returned to owner rates by breed?

```
breed_returned_rates <- aggregate(outcome_type ~ primary_breed, data = shelter_dogs,
                                  function(x) mean(x == "returned_to_owner") * 100)
breed_returned_rates <- breed_returned_rates %>%
  rename(returned_to_owner_rate = outcome_type)
```

I combined the rates for the top 5 outcome types along with breed counts into one tibble.

```
unique_breeds <- unique_breeds %>%
  left_join(breed_adoption_rates, by = "primary_breed") %>%
  left_join(breed_euthanasia_rates, by = "primary_breed") %>%
  left_join(breed_rescue_rates, by = "primary_breed") %>%
  left_join(breed_returned_rates, by = "primary_breed")
head(unique_breeds, 5)
```

```
## # A tibble: 5 x 6
##   primary_breed   counts adoption_rate euthanasia_rate rescue_rate
##   <chr>            <int>         <dbl>           <dbl>       <dbl>
## 1 bull_terrier     20075          34.3           27.2         14.2
## 2 german_shepherd  18251          44.2           17.7         13.8
## 3 chihuahua        15016          43.6            9.25        23.8
## 4 terrier          10936          46.0            7.14        26.0
## 5 siberian_husky    9594          44.0           11.9         10.9
## # i 1 more variable: returned_to_owner_rate <dbl>
```

Length of Stay

Average number of days in shelter?

```
shelter_dogs$stay_length <- as.numeric(difftime(shelter_dogs$outcome_date, shelter_dogs$intake_date, uni

average_stay <- mean(shelter_dogs$stay_length)
round(average_stay)
```

```
## [1] 15
```

Average per breed?

```
breed_average_stay <- shelter_dogs %>%
  group_by(primary_breed) %>%
  summarise(average_stay = mean(stay_length)) %>%
  arrange(desc(average_stay))
head(breed_average_stay)
```

```
## # A tibble: 6 x 2
##   primary_breed              average_stay
##   <chr>                             <dbl>
## 1 munsterlander                     128.
## 2 irish_terrier                      68.6
## 3 greater_swiss_mountain_dog         39
## 4 newfoundland                       38.6
## 5 wolfdog                            38.6
## 6 akbash                             36
```

```r
unique_breeds <- unique_breeds %>%
  left_join(breed_average_stay, by = "primary_breed")
```

```r
unique_breeds <- unique_breeds %>%
  mutate(average_stay = round(average_stay))
```

Trends Over Time

First, I created month/year columns

```r
shelter_dogs <- shelter_dogs %>%
  mutate(
    intake_month = month(intake_date, label = TRUE, abbr = TRUE),
    outcome_month = month(outcome_date, label = TRUE, abbr = TRUE),
    intake_year = year(intake_date),
    outcome_year = year(outcome_date)
  )
```

```r
shelter_dogs <- shelter_dogs %>%
  mutate(
    intake_month_year = format(as.Date(intake_date), "%m-%Y"),
    outcome_month_year = format(as.Date(outcome_date), "%m-%Y")
  )
```

Intakes

How many intakes have occurred per year?

```r
yearly_intakes <- shelter_dogs %>%
  group_by(intake_year) %>%
  summarise(intake_count = n()) %>%
  arrange(intake_year)
print(yearly_intakes)
```

```
## # A tibble: 8 x 2
##   intake_year intake_count
##         <dbl>        <int>
## 1        2017            9
## 2        2018        14316
## 3        2019        24914
## 4        2020        13699
## 5        2021        13705
## 6        2022        17978
## 7        2023        20164
## 8        2024        13544
```

I will exclude 2017 and 2024 from yearly analysis due to incomplete data.

```r
yearly_intakes <- yearly_intakes %>%
  filter(!(intake_year %in% c(2017, 2024)))
print(yearly_intakes)
```

```
## # A tibble: 6 x 2
##   intake_year intake_count
##         <dbl>        <int>
## 1        2018        14316
## 2        2019        24914
## 3        2020        13699
## 4        2021        13705
## 5        2022        17978
## 6        2023        20164
```

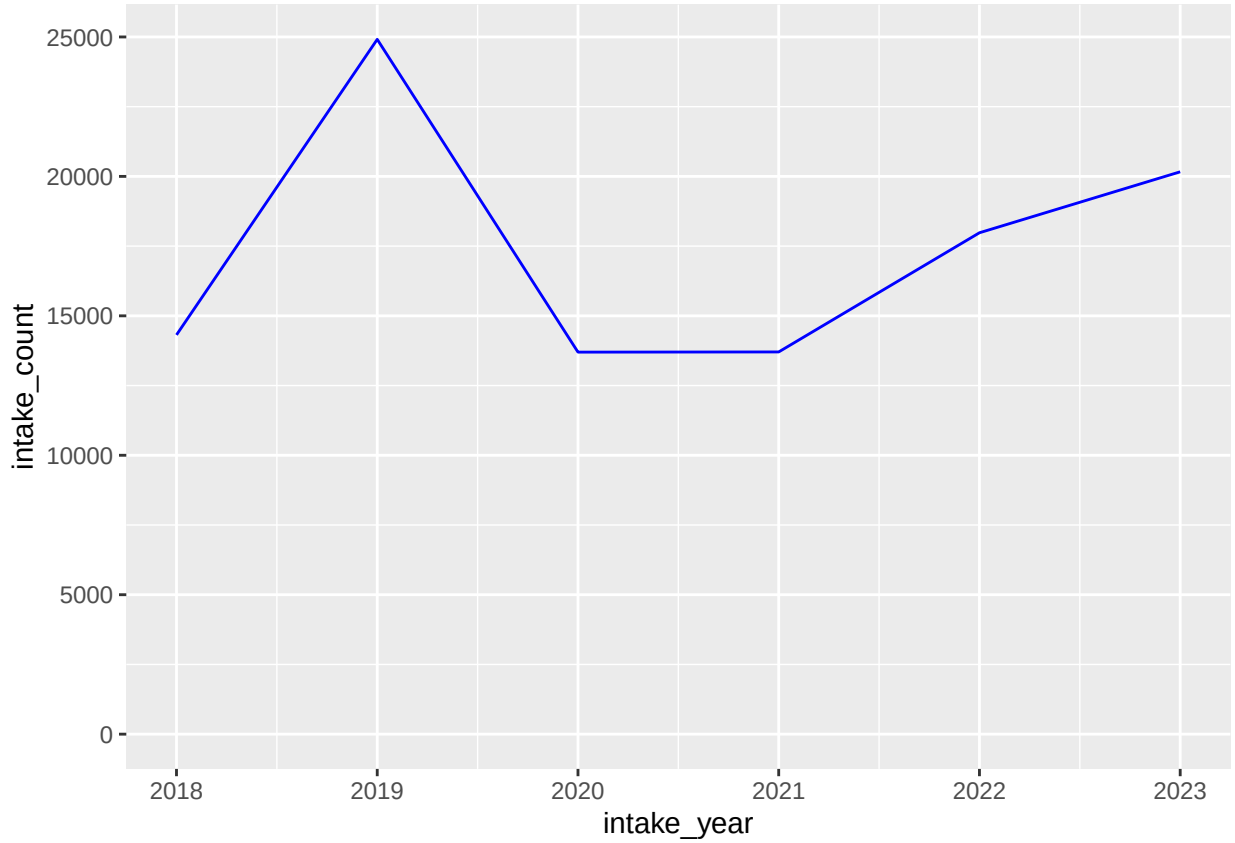
Average yearly intakes?

```r
average_yearly_intakes <- yearly_intakes %>%
  summarise(average_intakes = mean(intake_count))
print(average_yearly_intakes)
```

```
## # A tibble: 1 x 1
##   average_intakes
##             <dbl>
## 1          17463.
```

Yearly intakes visualized with a line chart:

```r
ggplot(yearly_intakes, aes(x = intake_year, y = intake_count)) +
  geom_line(color = "blue") +
  expand_limits(y = 0)
```

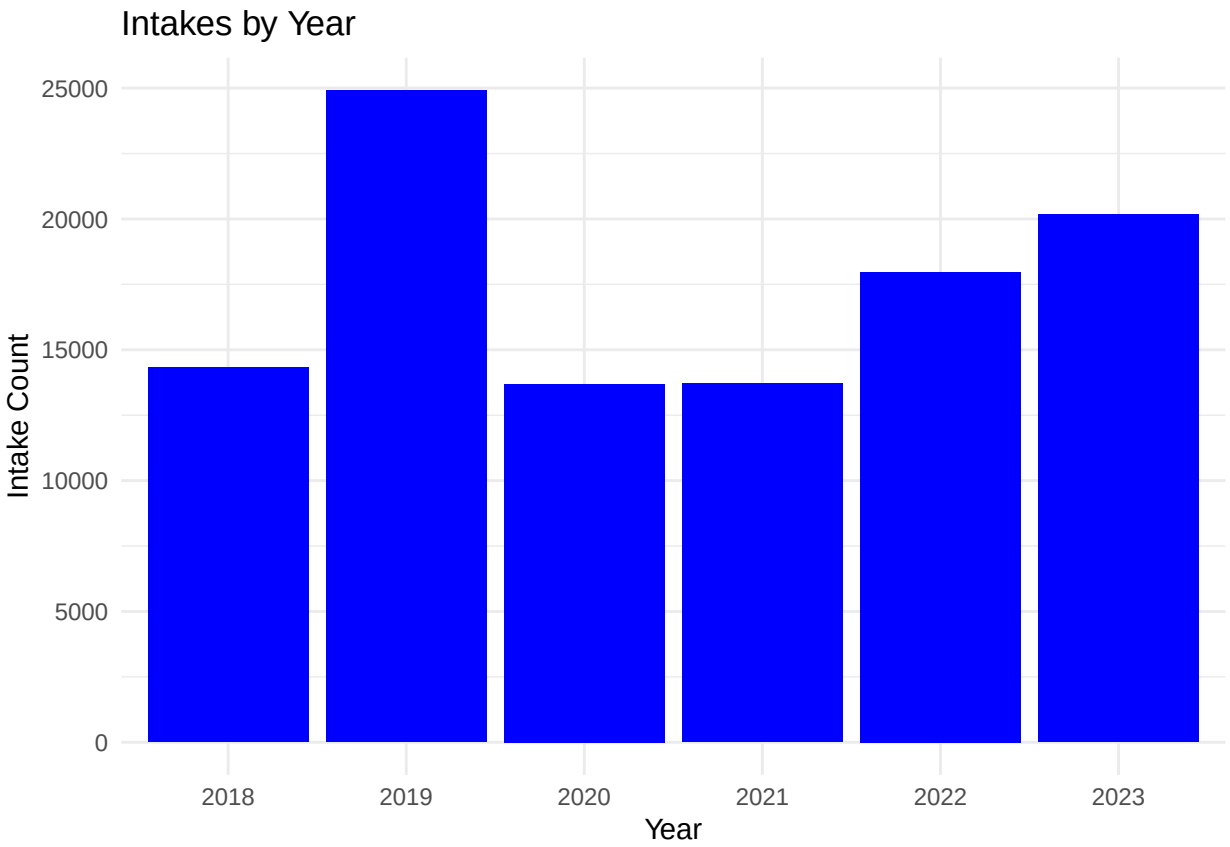



```
  labs(title = "Total Intakes per Year", x = "Year", y = "Intake Count") +
  theme_minimal()
```

```
## NULL
```

Visualized with a bar chart:

```
ggplot(yearly_intakes, aes(x = as.factor(intake_year), y = intake_count)) +
  geom_bar(stat = "identity", fill = "blue") +
  labs(x = "Year", y = "Intake Count", title = "Intakes by Year") +
  theme_minimal()
```



- Intakes peaked in 2019, normalized in 2020-2021, then increased in following years.

How many intakes have occurred per month from 2018-2023?

```
monthly_intakes <- shelter_dogs %>%
  group_by(intake_year, intake_month) %>%
  summarise(intake_count = n())
```

```
## `summarise()` has grouped output by 'intake_year'. You can override using the
## `.groups` argument.
```

```
monthly_intakes <- monthly_intakes %>%
  filter(!(intake_year %in% c(2017, 2024)))
```

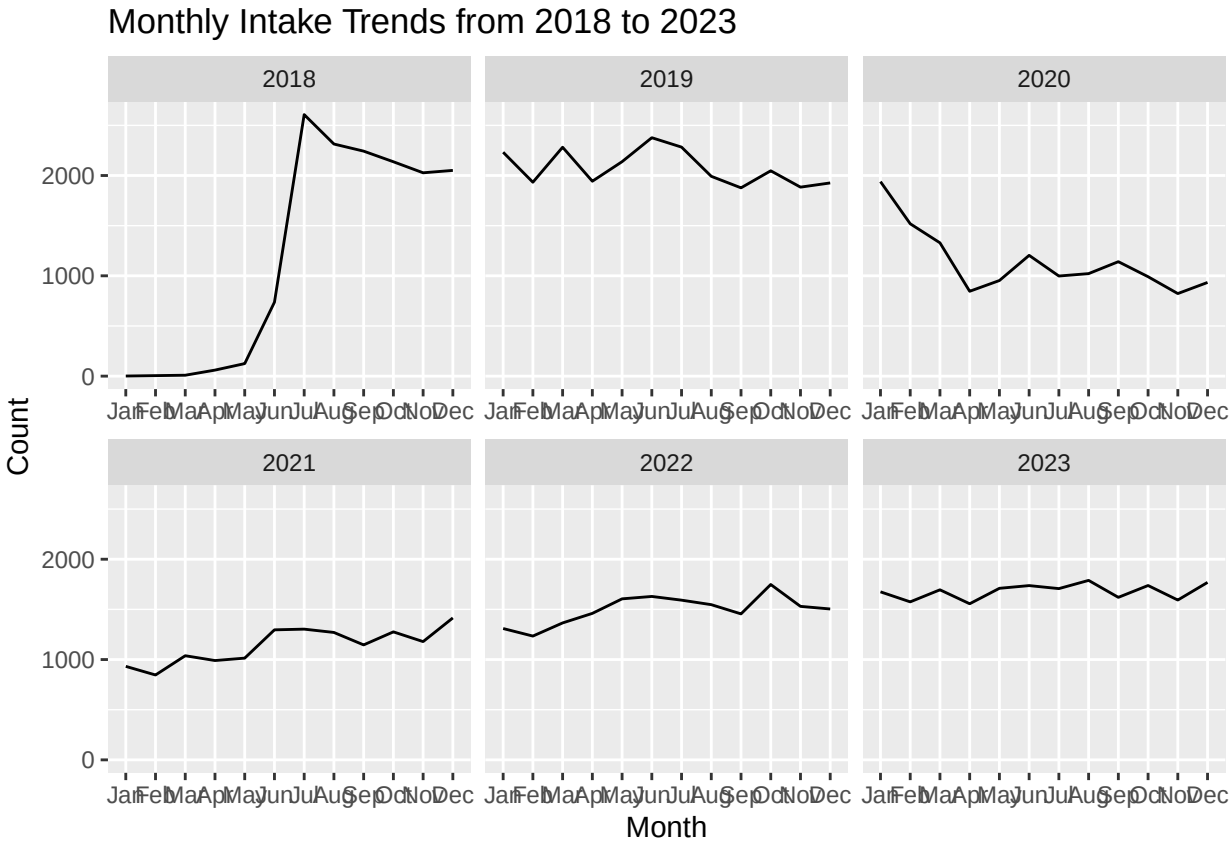
Average monthly intakes?

```
average_monthly_intakes <- monthly_intakes %>%
  ungroup() %>%
  summarise(average_intakes = mean(intake_count))
print(average_monthly_intakes)
```

```
## # A tibble: 1 x 1
##   average_intakes
##             <dbl>
## 1           1455.
```

Visualized monthly intakes:

```
ggplot(monthly_intakes, aes(x = intake_month, y = intake_count, group = intake_year)) +
  geom_line() +
  facet_wrap(~ intake_year, scales = "free_x") +
  labs(title = "Monthly Intake Trends from 2018 to 2023", x = "Month", y = "Count")
```



Outcomes

First, I counted total outcomes by year.

```
yearly_outcomes <- shelter_dogs %>%
  group_by(outcome_year, outcome_type) %>%
  summarise(outcome_count = n())
```

```
## `summarise()` has grouped output by 'outcome_year'. You can override using the
## `.groups` argument.
```

```
yearly_total_outcomes <- yearly_outcomes %>%
  group_by(outcome_year) %>%
  summarise(total_outcomes = sum(outcome_count))
```

Also, excluding 2017 and 2024

```r
yearly_outcomes <- yearly_outcomes %>%
  filter(!(outcome_year %in% c(2017, 2024)))
yearly_total_outcomes <- yearly_total_outcomes %>%
  filter(!(outcome_year %in% c(2017, 2024)))
```

How many adoptions occurred each year?

```r
yearly_adoptions <- yearly_outcomes %>%
  filter(outcome_type == "adopted") %>%
  rename(adoption_count = outcome_count) %>%
   select(-outcome_type)
print(yearly_adoptions)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year adoption_count
##          <dbl>          <int>
## 1         2018           5347
## 2         2019          10349
## 3         2020           5500
## 4         2021           5115
## 5         2022           7748
## 6         2023           8767
```

How many dogs were euthanized each year?

```r
yearly_euthanasia <- yearly_outcomes %>%
  filter(outcome_type == "euthanized") %>%
  rename(euthanasia_count = outcome_count) %>%
  select(-outcome_type)
print(yearly_euthanasia)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year euthanasia_count
##          <dbl>            <int>
## 1         2018             1555
## 2         2019             2906
## 3         2020             1495
## 4         2021             1658
## 5         2022             2989
## 6         2023             3773
```

How many dogs were returned to owners each year?

```r
yearly_returns <- yearly_outcomes %>%
  filter(outcome_type == "returned_to_owner") %>%
  rename(return_count = outcome_count) %>%
  select(-outcome_type)
print(yearly_returns)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year return_count
##          <dbl>        <int>
## 1         2018         2894
## 2         2019         5353
## 3         2020         3218
## 4         2021         2906
## 5         2022         3330
## 6         2023         3111
```

How many dogs were sent to rescues each year?

```
yearly_rescues <- yearly_outcomes %>%
  filter(outcome_type == "rescue") %>%
  rename(rescue_count = outcome_count) %>%
  select(-outcome_type)
print(yearly_rescues)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year rescue_count
##          <dbl>        <int>
## 1         2018         2980
## 2         2019         4959
## 3         2020         3067
## 4         2021         2554
## 5         2022         2494
## 6         2023         2724
```

I combined the counts for each outcome into one table and calculated their rates.

```
yearly_outcome_rates <- yearly_total_outcomes %>%
  left_join(yearly_adoptions, by = "outcome_year") %>%
  left_join(yearly_euthanasia, by = "outcome_year") %>%
  left_join(yearly_rescues, by = "outcome_year") %>%
  left_join(yearly_returns, by = "outcome_year")
yearly_outcome_rates <- yearly_outcome_rates %>%
  mutate(
    adoption_rate = (adoption_count / total_outcomes) * 100,
    euthanasia_rate = (euthanasia_count / total_outcomes) * 100,
    rescue_rate = (rescue_count / total_outcomes) * 100,
    return_to_owner_rate = (return_count / total_outcomes) * 100
  ) %>%
  select(-adoption_count, -euthanasia_count, -rescue_count, -return_count)
print(yearly_outcome_rates)
```

```
## # A tibble: 6 x 6
##   outcome_year total_outcomes adoption_rate euthanasia_rate rescue_rate
##          <dbl>          <int>         <dbl>           <dbl>       <dbl>
## 1         2018          13545          39.5            11.5        22.0
## 2         2019          24917          41.5            11.7        19.9
```

```
## 3         2020          14007          39.3           10.7        21.9
## 4         2021          13430          38.1           12.3        19.0
## 5         2022          17832          43.4           16.8        14.0
## 6         2023          20080          43.7           18.8        13.6
## # i 1 more variable: return_to_owner_rate <dbl>
```

Outcomes by month?

```
monthly_outcomes <- shelter_dogs %>%
  group_by(outcome_month_year, outcome_type) %>%
  summarise(outcome_count = n())
```

```
## `summarise()` has grouped output by 'outcome_month_year'. You can override
## using the `.groups` argument.
```

```
monthly_total_outcomes <- monthly_outcomes %>%
  group_by(outcome_month_year) %>%
  summarise(total_outcomes = sum(outcome_count))
```

```
monthly_outcomes <- monthly_outcomes %>%
  filter(!grepl("2017|2024", outcome_month_year))
monthly_total_outcomes <- monthly_total_outcomes %>%
  filter(!grepl("2017|2024", outcome_month_year))
```

Adoptions by month:

```
monthly_adoptions <- monthly_outcomes %>%
  filter(outcome_type == "adopted") %>%
  rename(adoption_count = outcome_count) %>%
   select(-outcome_type)
```

Euthanasia by month:

```
monthly_euthanasia <- monthly_outcomes %>%
  filter(outcome_type == "euthanized") %>%
  rename(euthanasia_count = outcome_count) %>%
   select(-outcome_type)
```

Returns to owners month:

```
monthly_returns <- monthly_outcomes %>%
  filter(outcome_type == "returned_to_owner") %>%
  rename(return_count = outcome_count) %>%
   select(-outcome_type)
```

Rescues by month:

```
monthly_rescues <- monthly_outcomes %>%
  filter(outcome_type == "rescue") %>%
  rename(rescue_count = outcome_count) %>%
   select(-outcome_type)
```

All outcome rates:

```
monthly_outcome_rates <- monthly_total_outcomes %>%
  inner_join(monthly_adoptions, by = c("outcome_month_year")) %>%
  inner_join(monthly_euthanasia, by = c("outcome_month_year")) %>%
  inner_join(monthly_returns, by = c("outcome_month_year")) %>%
  inner_join(monthly_rescues, by = c("outcome_month_year"))

monthly_outcome_rates <- monthly_outcome_rates %>%
  mutate(
    adoption_rate = (adoption_count / total_outcomes) * 100,
    euthanasia_rate = (euthanasia_count / total_outcomes) * 100,
    rescue_rate = (rescue_count / total_outcomes) * 100,
    return_to_owner_rate = (return_count / total_outcomes) * 100
  ) %>%
  select(-adoption_count, -euthanasia_count, -rescue_count, -return_count)
print(monthly_outcome_rates)
```

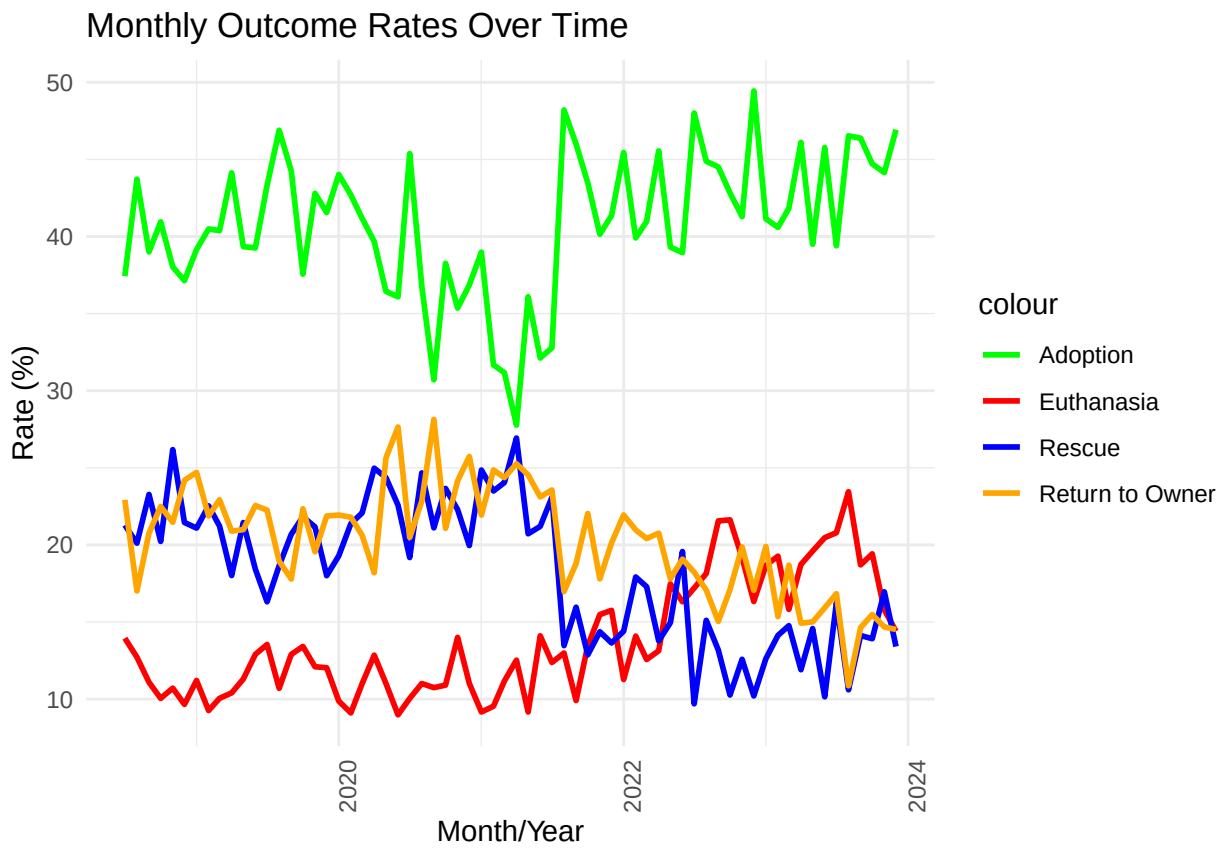
```
## # A tibble: 66 x 6
##    outcome_month_year total_outcomes adoption_rate euthanasia_rate rescue_rate
##    <chr>                       <int>         <dbl>           <dbl>       <dbl>
##  1 01-2019                      2186          39.2           11.2         21.1
##  2 01-2020                      1965          44.0            9.87        19.3
##  3 01-2021                      1026          39.0            9.16        24.9
##  4 01-2022                      1340          45.4           11.3         14.4
##  5 01-2023                      1553          41.1           18.7         12.6
##  6 02-2019                      1966          40.5            9.26        22.5
##  7 02-2020                      1670          42.7            9.10        21.4
##  8 02-2021                       881          31.7            9.53        23.5
##  9 02-2022                      1278          39.9           14.1         17.9
## 10 02-2023                      1557          40.6           19.3         14.1
## # i 56 more rows
## # i 1 more variable: return_to_owner_rate <dbl>
```

Outcome Visualization

Visualized all monthly outcome rates over time:

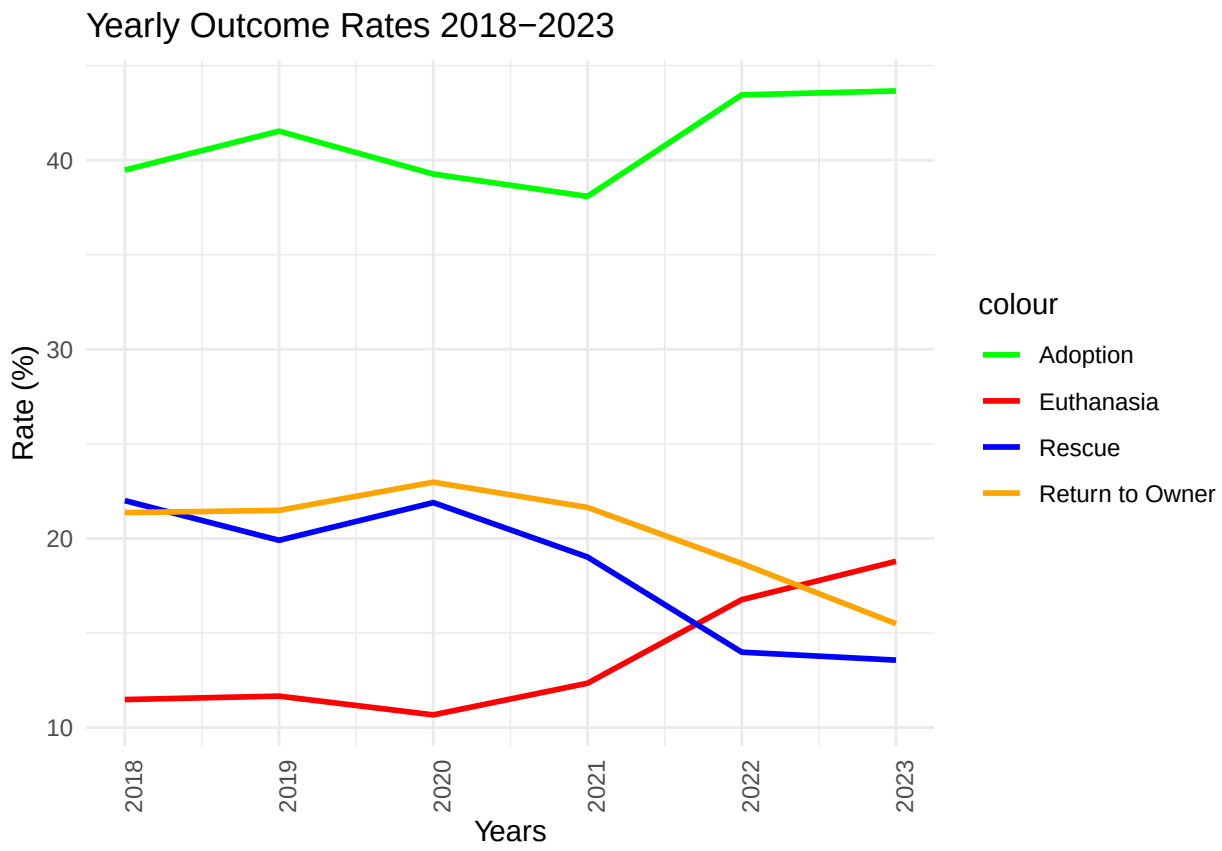
```
monthly_outcome_rates$outcome_month_year <- parse_date_time(monthly_outcome_rates$outcome_month_year, "
```

```
ggplot(monthly_outcome_rates, aes(x = outcome_month_year)) +
  geom_line(aes(y = adoption_rate, color = "Adoption"), linewidth = 1) +
  geom_line(aes(y = euthanasia_rate, color = "Euthanasia"), linewidth = 1) +
  geom_line(aes(y = rescue_rate, color = "Rescue"), linewidth = 1) +
  geom_line(aes(y = return_to_owner_rate, color = "Return to Owner"), linewidth = 1) +
  scale_color_manual(values = c("Adoption" = "green", "Euthanasia" = "red", "Rescue" = "blue", "Return
  labs(title = "Monthly Outcome Rates Over Time",
       x = "Month/Year", y = "Rate (%)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Yearly Outcome Rates:

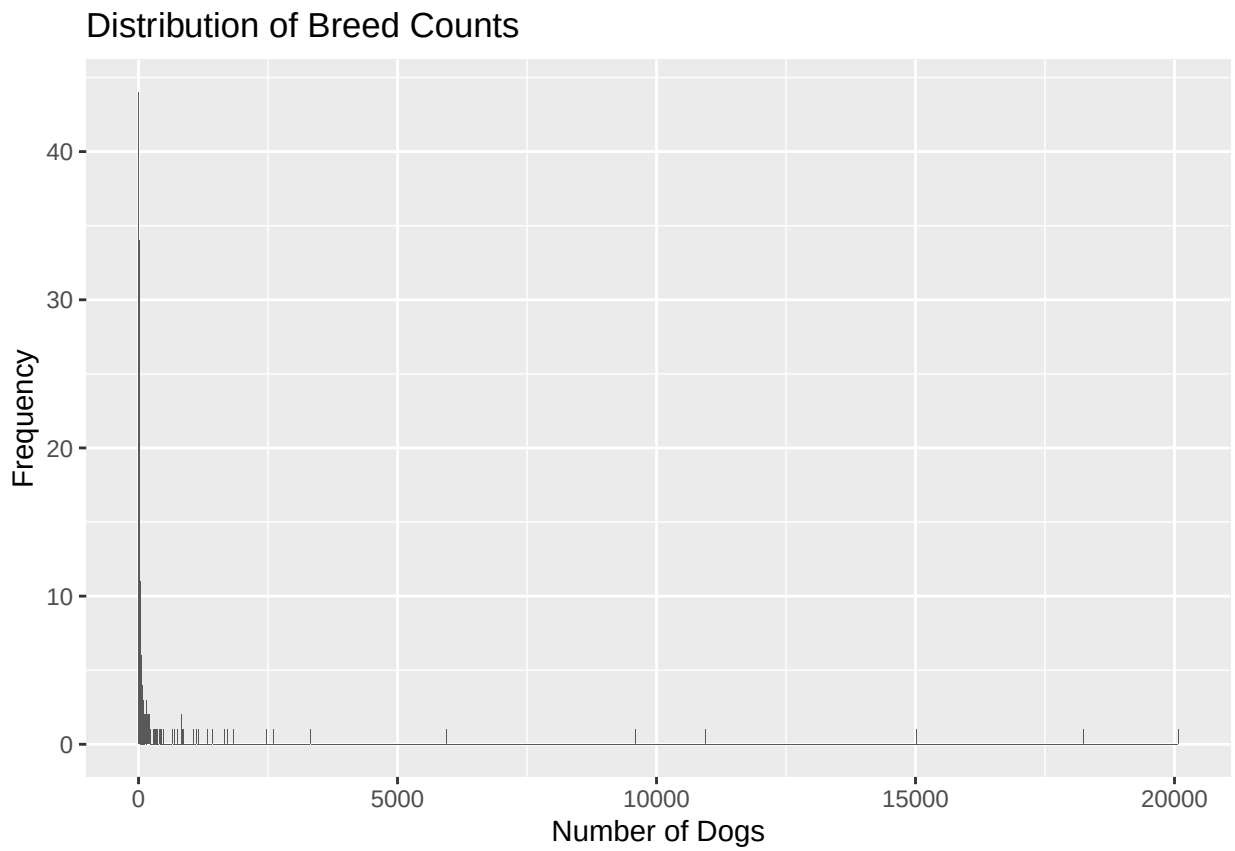
```
ggplot(yearly_outcome_rates, aes(x = outcome_year)) +
  geom_line(aes(y = adoption_rate, color = "Adoption"), linewidth = 1) +
  geom_line(aes(y = euthanasia_rate, color = "Euthanasia"), linewidth = 1) +
  geom_line(aes(y = rescue_rate, color = "Rescue"), linewidth = 1) +
  geom_line(aes(y = return_to_owner_rate, color = "Return to Owner"), linewidth = 1) +
  scale_color_manual(values = c("Adoption" = "green", "Euthanasia" = "red", "Rescue" = "blue", "Return
  labs(title = "Yearly Outcome Rates 2018-2023",
       x = "Years", y = "Rate (%)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



High-Risk Dog Breed Analysis

Pre-Filtering Statistics

```
ggplot(unique_breeds, aes(x = counts)) +
  geom_histogram(binwidth = 10) +
  labs(title = "Distribution of Breed Counts", x = "Number of Dogs", y = "Frequency")
```



The data is skewed by a large number of breeds with few instances in the data and a few number of breeds with many instances. The breeds with abnormally large populations will be included under the classification of "high-risk" breeds since they create a significant drain on shelter resources even if they do not have high rates of euthanasia.

Limiting Focus to Specific Breeds

First, I defined criteria for filtering breeds. Breeds are selected if:

- Breed makes up more than 10% over overall population, contributing to shelter overpopulation OR
- Breed has a statistically significant population percentage (greater than 1%) AND one or more of the following:
  - Adoption Rate - Lower than average adoption rate of 41.69%
  - Euthanasia Rate - Higher than average euthanasia rate of 14.18%
  - Length of Stay - Higher than average of 15 days

Then I calculated population percentages:

```
total_dog_count <- sum(unique_breeds$counts)
unique_breeds <- unique_breeds %>%
  mutate(population_percentage = round((counts / total_dog_count) * 100, 2))
```

And limited the data set to breeds with a population larger than 1% of the population. This limited the number of breeds from 192 to only 15.

```r
filtered_breeds <- unique_breeds %>% filter(population_percentage >= 1.00)

filtered_dogs <- shelter_dogs %>%
  filter(primary_breed %in% filtered_breeds$primary_breed)
```

Then, I filter using my criteria for high risk dogs.

```r
risk_breeds <- filtered_breeds %>%
  filter(
    adoption_rate < 41.74 |
    euthanasia_rate > 14.35 |
    population_percentage > 10 |
    average_stay > 15
    )
print(risk_breeds)
```

```
## # A tibble: 10 x 8
##    primary_breed    counts adoption_rate euthanasia_rate rescue_rate
##    <chr>             <int>         <dbl>           <dbl>       <dbl>
##  1 bull_terrier      20075          34.3           27.2         14.2
##  2 german_shepherd   18251          44.2           17.7         13.8
##  3 chihuahua         15016          43.6            9.25        23.8
##  4 siberian_husky     9594          44.0           11.9         10.9
##  5 miniature_poodle   3323          38.7            8.34        23.1
##  6 bulldog            2607          36.9           13.0         17.7
##  7 maltese            2485          40.1            8.65        23.1
##  8 shih_tzu           1723          36.7            8.88        17.6
##  9 boxer              1660          36.8           15.2         22.0
## 10 rottweiler         1418          36.2           26.1         13.3
## # i 3 more variables: returned_to_owner_rate <dbl>, average_stay <dbl>,
## #   population_percentage <dbl>
```

I excluded those breeds which had low adoption rates that can be explained by high return to owner rates, along with low euthanasia rates and small population. These breeds are not part of my focus.

```r
risk_breeds <- risk_breeds %>%
  filter(
    !(returned_to_owner_rate > 19.24 & euthanasia_rate < 14.35 & average_stay < 15)
  )
```

Final Filtering of Data Set:

```r
risk_dogs <- shelter_dogs %>%
  filter(primary_breed %in% risk_breeds$primary_breed)
```

The final high-risk breeds list includes: bull terriers, german shepherds, chihuahuas, huskies, boxers, and rottweilers.

Trends for High-Risk Breeds

Now I will compare trends in intakes and outcomes for high risk breeds with those of the whole population.

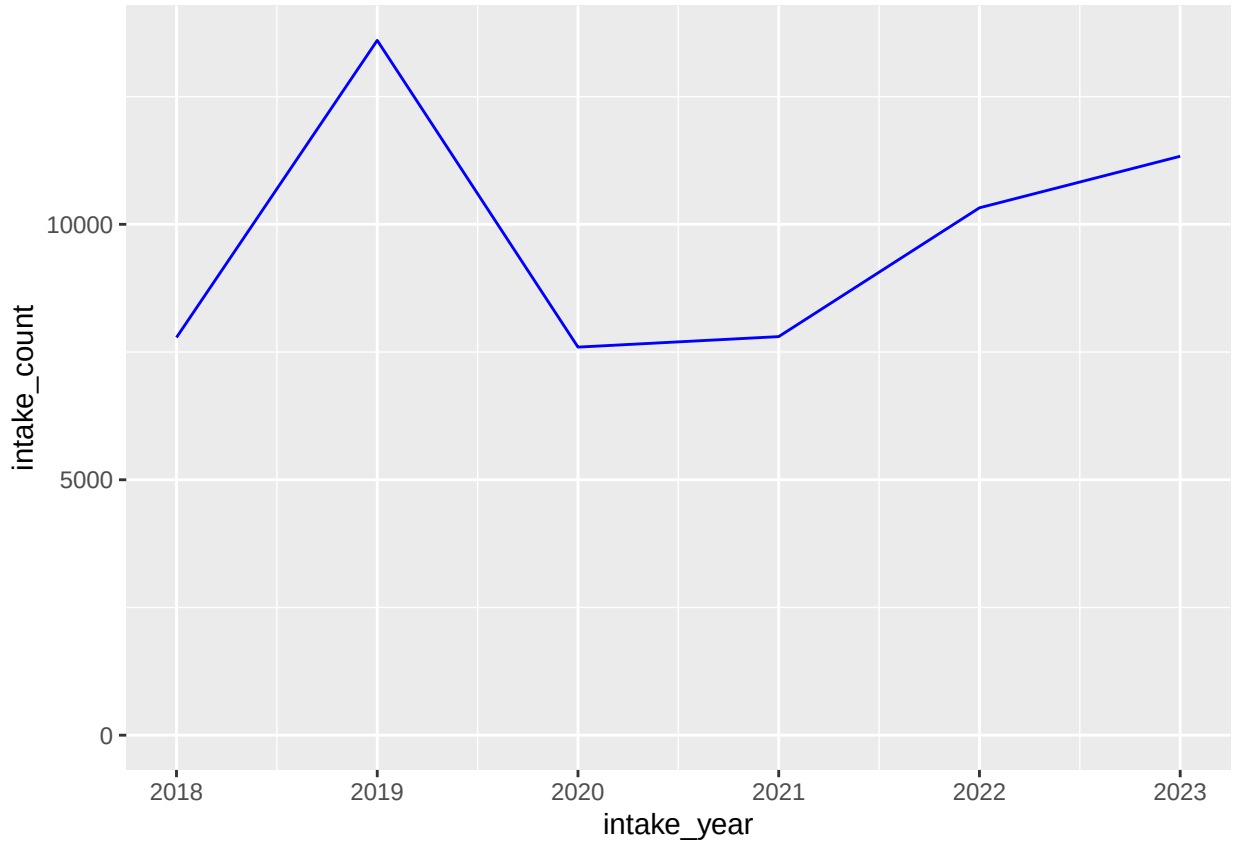
Intakes

Yearly

```
yearly_risk_intakes <- risk_dogs %>%
  group_by(intake_year) %>%
  summarise(intake_count = n()) %>%
  filter(!(intake_year %in% c(2017, 2024))) %>%
  arrange(intake_year)
print(yearly_risk_intakes)
```

```
## # A tibble: 6 x 2
##   intake_year intake_count
##         <dbl>        <int>
## 1        2018         7785
## 2        2019        13601
## 3        2020         7596
## 4        2021         7802
## 5        2022        10323
## 6        2023        11332
```

```
ggplot(yearly_risk_intakes, aes(x = intake_year, y = intake_count)) +
  geom_line(color = "blue") +
  expand_limits(y = 0)
```



```
  labs(title = "Total High Risk Intakes per Year", x = "Year", y = "Intake Count") +
  theme_minimal()
```

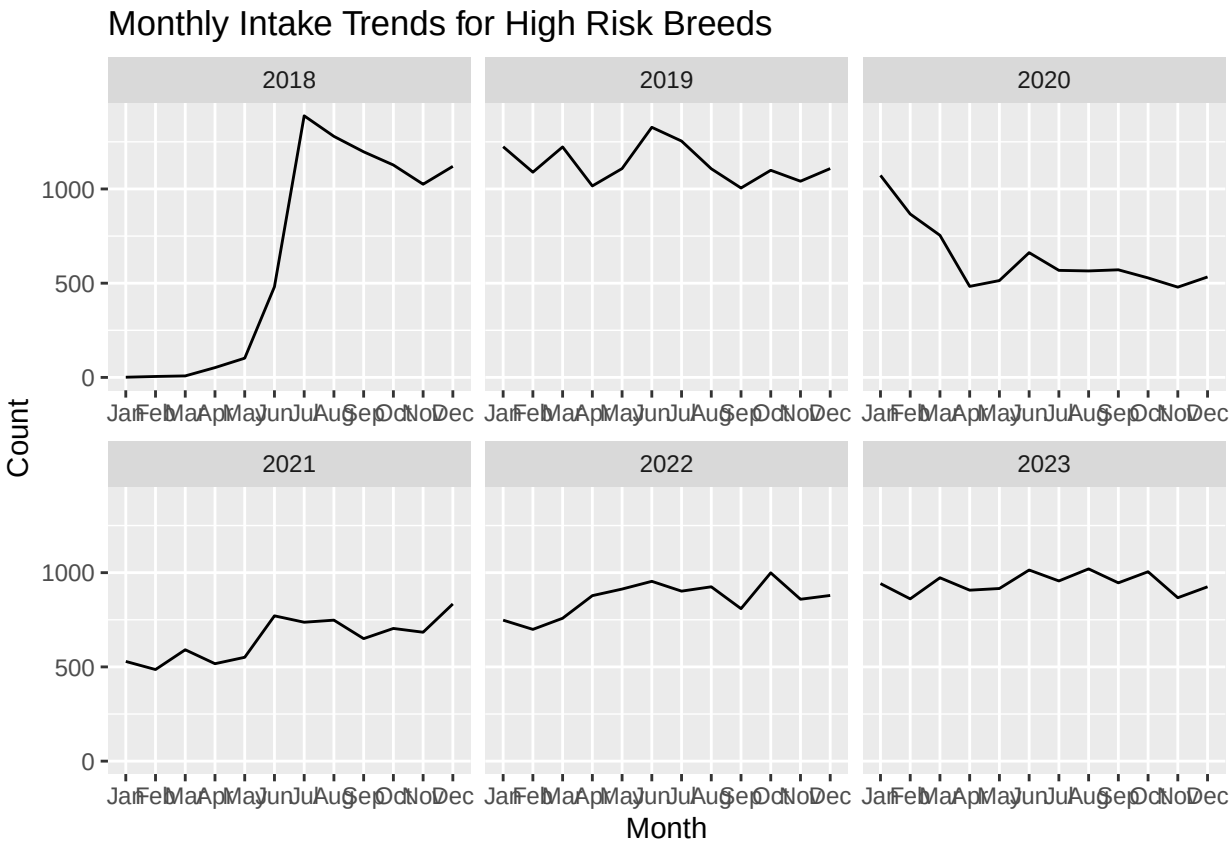
```
## NULL
```

Monthly

```
monthly_risk_intakes <- risk_dogs %>%
  group_by(intake_year, intake_month) %>%
  summarise(intake_count = n()) %>%
  filter(!(intake_year %in% c(2017, 2024)))
```

```
## `summarise()` has grouped output by 'intake_year'. You can override using the
## `.groups` argument.
```

```
ggplot(monthly_risk_intakes, aes(x = intake_month, y = intake_count, group = intake_year)) +
  geom_line() +
  facet_wrap(~ intake_year, scales = "free_x") +
  labs(title = "Monthly Intake Trends for High Risk Breeds", x = "Month", y = "Count")
```



Outcomes

Yearly - All outcomes

```r
yearly_risk_outcomes <- risk_dogs %>%
  group_by(outcome_year, outcome_type) %>%
  summarise(outcome_count = n()) %>%
  filter(!(outcome_year %in% c(2017, 2024)))
```

```
## `summarise()` has grouped output by 'outcome_year'. You can override using the
## `.groups` argument.
```

```r
yearly_total_risk_outcomes <- yearly_risk_outcomes %>%
  group_by(outcome_year) %>%
  summarise(total_outcomes = sum(outcome_count)) %>%
  filter(!(outcome_year %in% c(2017, 2024)))
```

Yearly adoptions

```r
yearly_risk_adoptions <- yearly_risk_outcomes %>%
  filter(outcome_type == "adopted") %>%
  rename(adoption_count = outcome_count) %>%
   select(-outcome_type)
print(yearly_risk_adoptions)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year adoption_count
##          <dbl>          <int>
## 1         2018           2807
## 2         2019           5714
## 3         2020           3050
## 4         2021           2925
## 5         2022           4283
## 6         2023           4614
```

Yearly euthanasia

```r
yearly_risk_euthanasia <- yearly_risk_outcomes %>%
  filter(outcome_type == "euthanized") %>%
  rename(euthanasia_count = outcome_count) %>%
  select(-outcome_type)
print(yearly_risk_euthanasia)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year euthanasia_count
##          <dbl>            <int>
## 1         2018             1074
## 2         2019             1993
## 3         2020              980
## 4         2021             1079
## 5         2022             2091
## 6         2023             2758
```

Yearly returns

```
yearly_risk_returns <- yearly_risk_outcomes %>%
  filter(outcome_type == "returned_to_owner") %>%
  rename(return_count = outcome_count) %>%
  select(-outcome_type)
print(yearly_risk_returns)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year return_count
##          <dbl>        <int>
## 1         2018         1467
## 2         2019         2756
## 3         2020         1699
## 4         2021         1565
## 5         2022         1768
## 6         2023         1661
```

Yearly rescues

```
yearly_risk_rescues <- yearly_risk_outcomes %>%
  filter(outcome_type == "rescue") %>%
  rename(rescue_count = outcome_count) %>%
  select(-outcome_type)
print(yearly_risk_rescues)
```

```
## # A tibble: 6 x 2
## # Groups:   outcome_year [6]
##   outcome_year rescue_count
##          <dbl>        <int>
## 1         2018         1419
## 2         2019         2316
## 3         2020         1623
## 4         2021         1344
## 5         2022         1334
## 6         2023         1391
```

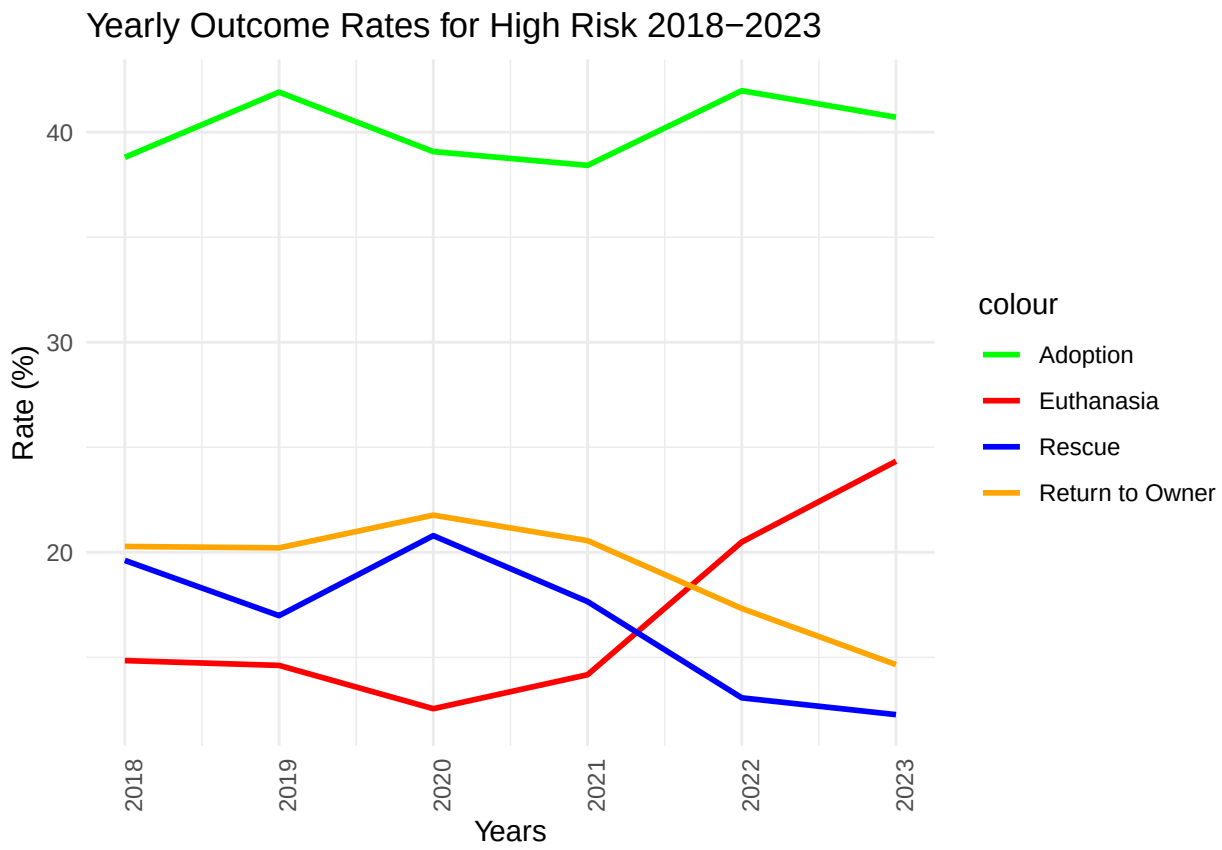
Combined yearly rates

```
yearly_risk_rates <- yearly_total_risk_outcomes %>%
  left_join(yearly_risk_adoptions, by = "outcome_year") %>%
  left_join(yearly_risk_euthanasia, by = "outcome_year") %>%
  left_join(yearly_risk_rescues, by = "outcome_year") %>%
  left_join(yearly_risk_returns, by = "outcome_year")
yearly_risk_rates <- yearly_risk_rates %>%
  mutate(
    adoption_rate = (adoption_count / total_outcomes) * 100,
    euthanasia_rate = (euthanasia_count / total_outcomes) * 100,
    rescue_rate = (rescue_count / total_outcomes) * 100,
    return_to_owner_rate = (return_count / total_outcomes) * 100
  ) %>%
  select(-adoption_count, -euthanasia_count, -rescue_count, -return_count)
print(yearly_risk_rates)
```

```
## # A tibble: 6 x 6
##   outcome_year total_outcomes adoption_rate euthanasia_rate rescue_rate
##          <dbl>          <int>         <dbl>           <dbl>       <dbl>
## 1         2018           7234          38.8            14.8        19.6
## 2         2019          13635          41.9            14.6        17.0
## 3         2020           7805          39.1            12.6        20.8
## 4         2021           7613          38.4            14.2        17.7
## 5         2022          10204          42.0            20.5        13.1
## 6         2023          11333          40.7            24.3        12.3
## # i 1 more variable: return_to_owner_rate <dbl>
```

Visualization

```
ggplot(yearly_risk_rates, aes(x = outcome_year)) +
  geom_line(aes(y = adoption_rate, color = "Adoption"), linewidth = 1) +
  geom_line(aes(y = euthanasia_rate, color = "Euthanasia"), linewidth = 1) +
  geom_line(aes(y = rescue_rate, color = "Rescue"), linewidth = 1) +
  geom_line(aes(y = return_to_owner_rate, color = "Return to Owner"), linewidth = 1) +
  scale_color_manual(values = c("Adoption" = "green", "Euthanasia" = "red", "Rescue" = "blue", "Return
  labs(title = "Yearly Outcome Rates for High Risk 2018-2023",
       x = "Years", y = "Rate (%)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



When the trends in outcome rates for high risk dogs are compared to trends in overall population, the following can be observed:

- While the overall population saw a stable adoption rate from 2022 to 2023, high risk dogs saw a decrease in adoptions.
- While both have seen an increase in euthanasia since 2021, rates or high risk breeds have increased more steeply.
- Rescue involvement has decreased for both the overall population and high risk population.

Monthly - All Outcomes

```
monthly_risk_outcomes <- risk_dogs %>%
  group_by(outcome_month_year, outcome_type) %>%
  summarise(outcome_count = n()) %>%
  filter(!grepl("2017|2024", outcome_month_year))
```

```
## `summarise()` has grouped output by 'outcome_month_year'. You can override
## using the `.groups` argument.
```

```
monthly_total_risk_outcomes <- monthly_risk_outcomes %>%
  group_by(outcome_month_year) %>%
  summarise(total_outcomes = sum(outcome_count)) %>%
  filter(!grepl("2017|2024", outcome_month_year))
```

Monthly Adoptions

```
monthly_risk_adoptions <- monthly_risk_outcomes %>%
  filter(outcome_type == "adopted") %>%
  rename(adoption_count = outcome_count) %>%
   select(-outcome_type)
```

Monthly Euthanasia

```
monthly_risk_euthanasia <- monthly_risk_outcomes %>%
  filter(outcome_type == "euthanized") %>%
  rename(euthanasia_count = outcome_count) %>%
   select(-outcome_type)
```

Monthly Returns

```
monthly_risk_returns <- monthly_risk_outcomes %>%
  filter(outcome_type == "returned_to_owner") %>%
  rename(return_count = outcome_count) %>%
   select(-outcome_type)
```

Monthly Rescues

```
monthly_risk_rescues <- monthly_risk_outcomes %>%
  filter(outcome_type == "rescue") %>%
  rename(rescue_count = outcome_count) %>%
   select(-outcome_type)
```

Monthly Combined Rates

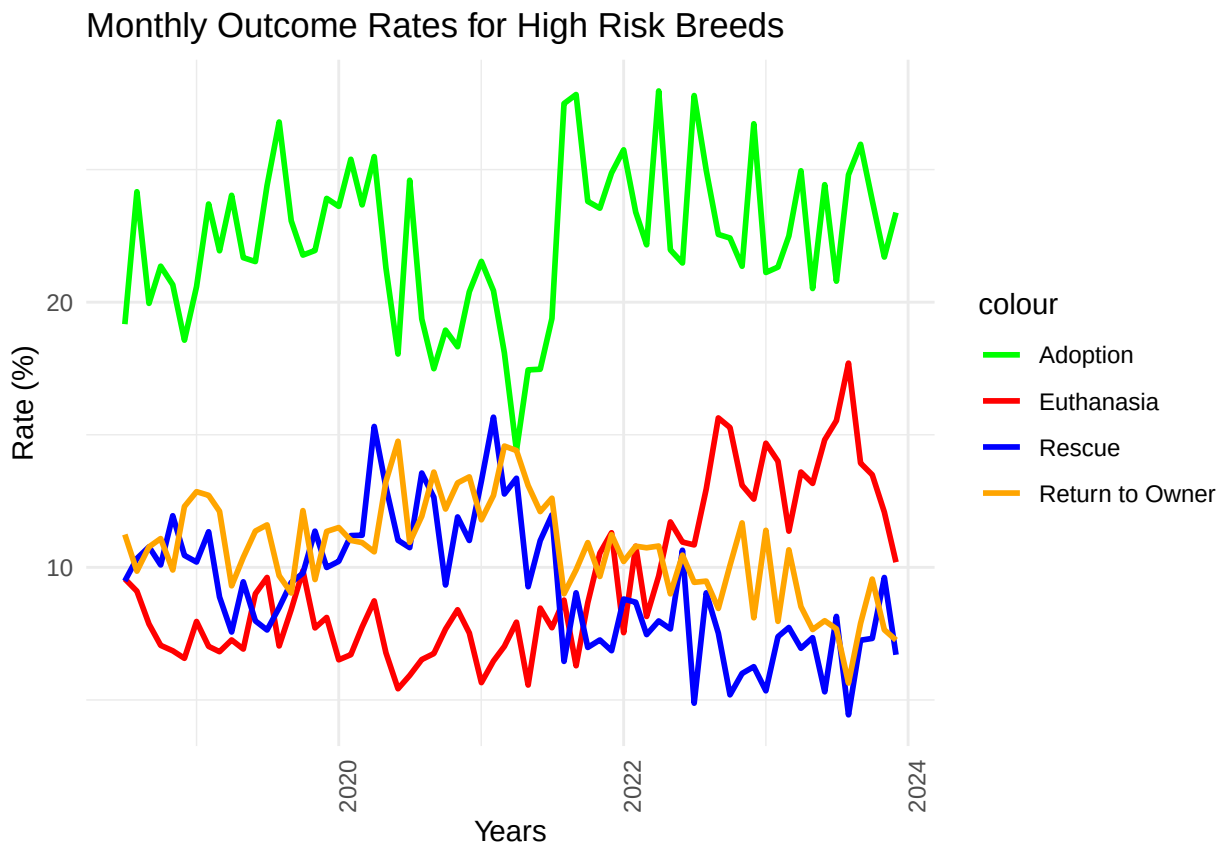
```r
monthly_risk_rates <- monthly_total_outcomes %>%
  inner_join(monthly_risk_adoptions, by = c("outcome_month_year")) %>%
  inner_join(monthly_risk_euthanasia, by = c("outcome_month_year")) %>%
  inner_join(monthly_risk_returns, by = c("outcome_month_year")) %>%
  inner_join(monthly_risk_rescues, by = c("outcome_month_year"))

monthly_risk_rates <- monthly_risk_rates %>%
  mutate(
    adoption_rate = (adoption_count / total_outcomes) * 100,
    euthanasia_rate = (euthanasia_count / total_outcomes) * 100,
    rescue_rate = (rescue_count / total_outcomes) * 100,
    return_to_owner_rate = (return_count / total_outcomes) * 100
  ) %>%
  select(-adoption_count, -euthanasia_count, -rescue_count, -return_count)
```

```r
monthly_risk_rates$outcome_month_year <- parse_date_time(monthly_risk_rates$outcome_month_year, "my")
```

Monthly Visualization

```r
ggplot(monthly_risk_rates, aes(x = outcome_month_year)) +
  geom_line(aes(y = adoption_rate, color = "Adoption"), linewidth = 1) +
  geom_line(aes(y = euthanasia_rate, color = "Euthanasia"), linewidth = 1) +
  geom_line(aes(y = rescue_rate, color = "Rescue"), linewidth = 1) +
  geom_line(aes(y = return_to_owner_rate, color = "Return to Owner"), linewidth = 1) +
  scale_color_manual(values = c("Adoption" = "green", "Euthanasia" = "red", "Rescue" = "blue", "Return
  labs(title = "Monthly Outcome Rates for High Risk Breeds",
       x = "Years", y = "Rate (%)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

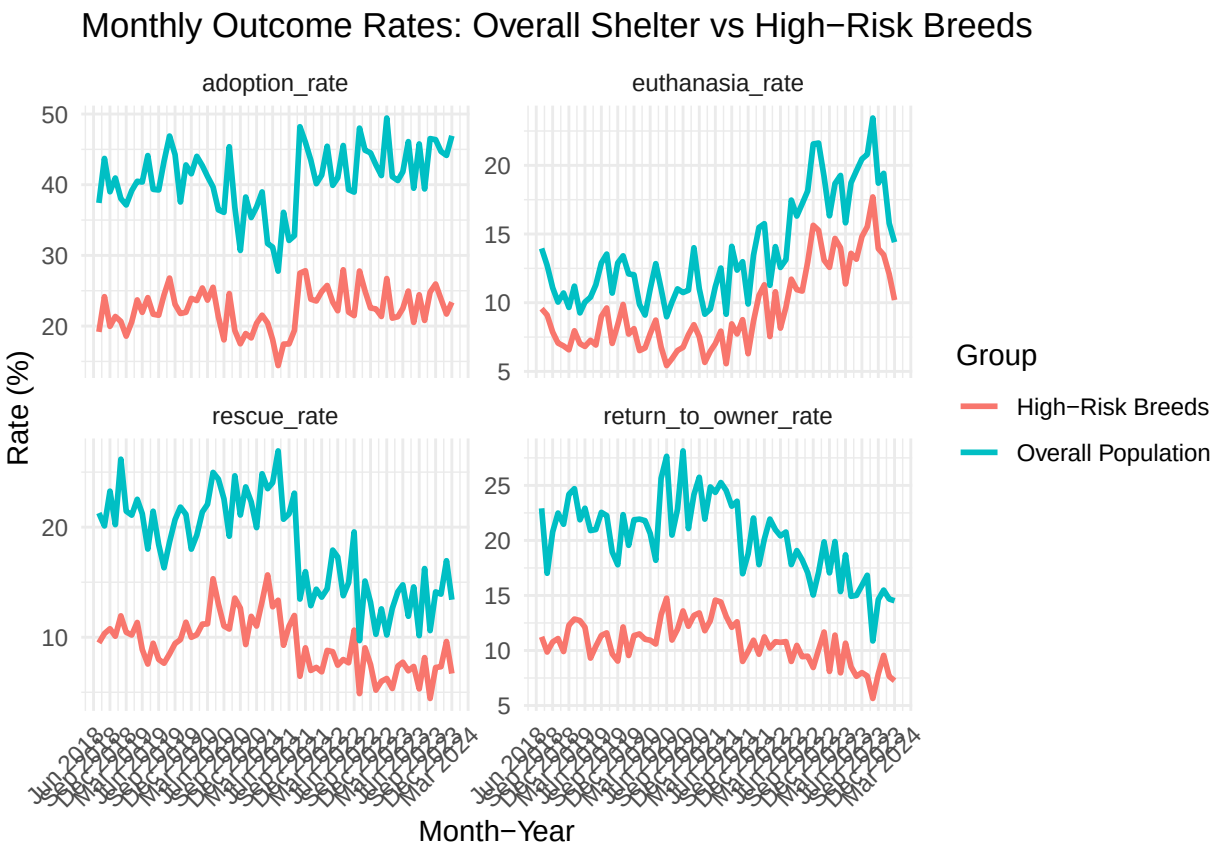


Comparing High Risk with Overall Population

Monthly

```
compared_rates <- bind_rows(
  monthly_outcome_rates %>% mutate(source = "Overall Population"),
  monthly_risk_rates %>% mutate(source = "High-Risk Breeds")
)
compared_rates_long <- compared_rates %>%
  pivot_longer(cols = c(adoption_rate, euthanasia_rate, rescue_rate, return_to_owner_rate),
               names_to = "outcome_type", values_to = "rate")
compared_rates_long <- compared_rates_long %>%
  mutate(outcome_month_year = as.Date(outcome_month_year))
```

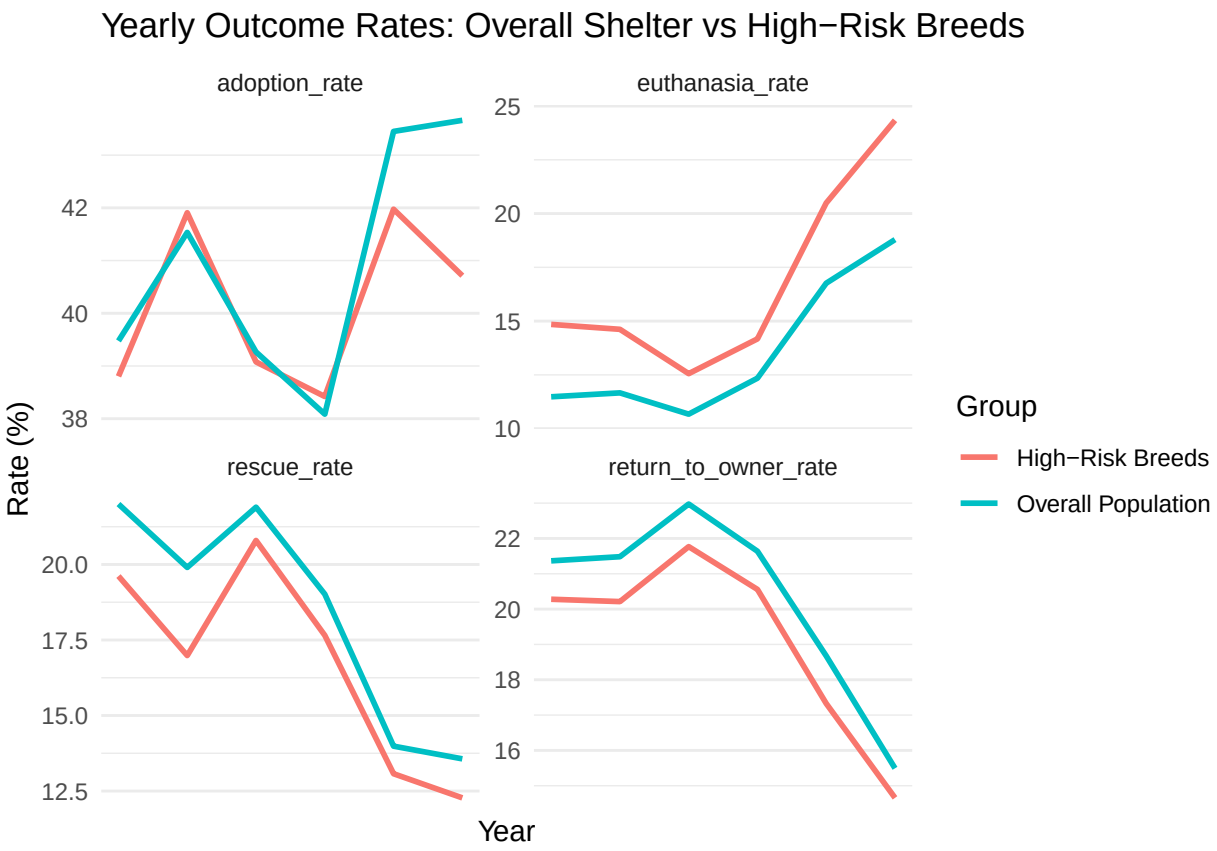
```
ggplot(compared_rates_long, aes(x = outcome_month_year, y = rate, color = source)) +
  geom_line(linewidth = 1) +
  facet_wrap(~ outcome_type, scales = "free_y") +
  labs(title = "Monthly Outcome Rates: Overall Shelter vs High-Risk Breeds",
       x = "Month-Year",
       y = "Rate (%)",
       color = "Group") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_x_date(date_breaks = "3 months", date_labels = "%b %Y")
```



Yearly

```
yearly_compared_rates <- bind_rows(
  yearly_outcome_rates %>% mutate(source = "Overall Population"),
  yearly_risk_rates %>% mutate(source = "High-Risk Breeds")
)
yearly_rates_long <- yearly_compared_rates %>%
  pivot_longer(cols = c(adoption_rate, euthanasia_rate, rescue_rate, return_to_owner_rate),
               names_to = "outcome_type", values_to = "rate")
yearly_rates_long <- yearly_rates_long %>%
  mutate(outcome_year = as.Date(outcome_year))
```

```
ggplot(yearly_rates_long, aes(x = outcome_year, y = rate, color = source)) +
  geom_line(size = 1) +
  facet_wrap(~ outcome_type, scales = "free_y") +
  labs(title = "Yearly Outcome Rates: Overall Shelter vs High-Risk Breeds",
       x = "Year",
       y = "Rate (%)",
       color = "Group") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_x_date(date_breaks = "1 year", date_labels = "%b %Y")
```



Results and Findings

The following is a summary of my analysis findings:

All Los Angles County Shelters

- Most intakes in shelters are strays. After strays, owner surrenders are the most common. These two types make up a majority of all intakes.
- Based on the data set, only around 42% of all dogs were adopted.
- Around 14 percent of dogs were euthanized, around 18% were sent to rescue shelters, and around 19% were returned to their original owners.
- The average amount of time spent in shelters is 15 days.
- Overall intake counts for all shelters peaked in 2019, normalized in 2020-2021, then increased in following years.
  - 2018 and 2019 saw much higher than average monthly intakes
  - 2020-2022 saw lower intakes
  - 2023 saw slightly higher monthly intakes
- While intakes peaked in 2019, adoptions also peaked.
- **Adoptions were highest from 2022-2023 — overall, they are trending upwards!**

- **Unfortunately, euthanasia has also increased drastically from 2022-2023.**
- The number of dogs sent to rescues has decreased in recent years.

All Breeds Represented in Data Set

- There are 197 primary breeds that have been through an LA County shelter since 2017.
  - A large number of breeds have only a few occurrences, but a small number of breeds have excessively large populations in shelters.
    * Only 5 make up $> 5\%$ while 182 have less than 1%
    * The breeds with abnormally large populations will be included under the classification of "high-risk" breeds since they create a significant drain on shelter resources even if they do not have high rates of euthanasia.

High Risk Breeds

This project aims to identify those breeds that have excessive numbers in shelters, face low adoptions, and/or are at higher risk for euthanasia or lengthy stays (which can result in trauma and behavioral issues).

The data set was filtered to isolate the following breeds:

- Makes up more than 10% over overall population, contributing to shelter overpopulation OR
- Has a statistically significant population percentage (greater than 1%) AND one or more of the following:
  - Adoption Rate - Lower than average adoption rate of 41.69%
  - Euthanasia Rate - Higher than average euthanasia rate of 14.18%
  - Length of Stay - Higher than average of 15 days

The high risk breeds were determined to be

1. **Bull Terriers** (including Pit Bull Terriers and Staffordshire Bull Terriers)
   - Highest count across all shelters — 20,075 total. or 17% of data set
   - Adoption rate much lower than average at 34%
   - Extremely high euthanasia rate! Highest at 27%
   - Longer than average stay of 25 days
2. **German Shepherds**
   - Made up 15% of all population
   - Higher than average euthanasia rate of 18%
   - Longer than average stay of 19 days
3. **Chihuahuas**
   - Made up 13% of all population
4. **Siberian Huskies** (including Alaskan Huskies)
   - Made up 8% of all population

- Longer than average stay of 17 days

5. **Boxers**
    - Adoption rate much lower than average at 36%
    - Longer than average stay of 17 days
6. **Rottweilers**
    - Adoption rate much lower than average at 36%
    - Extremely high euthanasia rate! Second highest at 27%
    - Longer than average stay of 17 days

High-risk breed trends compared to overall population trends:

- While the overall population saw an increase in adoptions from 2022 to 2023, high-risk dogs saw a decrease in adoptions.
- While both have seen an increase in euthanasia since 2021, rates or high risk breeds have increased more steeply. Euthanasia rates are consistently higher for high-risk than for the overall population.
- Rescue involvement has decreased for both the overall population and high risk population.

Recommendations

Based on this analysis, I recommend the following actions:

- **Targeted adoption campaigns for high-risk breeds.**
    - Developing breed-specific outreach and marketing campaigns can help highlight the positive qualities of these breeds.
    - Increasing public awareness and education combats negative stereotypes/preconceptions about breeds that hinder their adoption. It also allows for more successful dog ownership.
    - High-risk breeds should be highlighted at shelter adoption events.
    - Consider discounted or waived adoption fees for high-risk breeds.
- **Strengthen relationships with breed-specific rescue groups.**
    - Collaborate on media campaigns for breed education.
    - Send more high-risk dogs to rescues for adoption rather than euthanize them.
        * Rescues often have the expertise and resources to rehabilitate dogs and prepare them for adoption.
- **Manage shelter populations by reducing intakes.**
    - Strays and owner surrenders are two most common intakes.
        * Consider launching a low-cost or free micro-chipping program for dogs not adopted from shelters, to reduce the risk of these dogs ending up in shelters as strays.
    - Dogs surrendered by owners make up 19% of all intakes. Reducing this number would help lessen the drain on shelter resources.
        * Educating owners on breed specific care can reduce surrenders and returns.

Conclusion

The ultimate goal is to increase positive outcomes (adoptions and rescues) while decreasing negative outcomes (euthanasia and long shelter stays) for the high-risk breeds identified in this project. By specifically dealing with the breeds that are prevalent and vulnerable, LA County shelters can better manage overpopulation and improve outcomes for all dogs in their care.

Limitations

- Scope: This project will limit it's focus to dogs (excluding cats from the original data) in the Los Angeles area only. The data set used contains animal intake records for LA County Shelters only and is provided by the city's open data website. This analysis will not include data from City of LA animal shelters. While the City of LA does have some adoption data available publicly, it is not comprehensive and lacks data on breeds. LA county's database is cleaner, more complete, and covers a larger time period/area, so it was the sole focus of this analysis.
- No age or gender data included in data set.

References

1. PawStats Data Source Info
2. American Kennel Club Official Dog Breed List