Final Project Writeup

Samantha Wavell
UID 806317391

Stats 418 Spring ‘25
University of California, Los Angeles
Department of Statistics & Data Science

FINAL PROJECT WRITEUP

INTRODUCTION

The data for this project are from the public API, xeno-canto.org, which holds a collection of over 952 thousand recordings of over 12 thousand species of birds, grasshoppers, bats, frogs, and land mammals. Specifically, data for this project are from crows (Order: Passeriformes, Family: Corvidae, Genus: Corvus). New recordings are often added to the website. As of June 2, 2025, there were a total of 11,400 recordings of 45 subspecies of crows.

The Shiny app developed for this project predicts the species of crow from which a bird recording originates, based on location (country) and season (fall, winter, spring, summer) of the recording. A random forest classifier was trained on metadata from the xeno-canto database. The app displays species predictions along with audio recordings, model performance metrics, and an interactive map and data table.

Access the live Shiny app here: https://96upvf-samantha-wavell.shinyapps.io/Stats-418-Final-Project-App/

EXPLORATORY DATA ANALYSIS

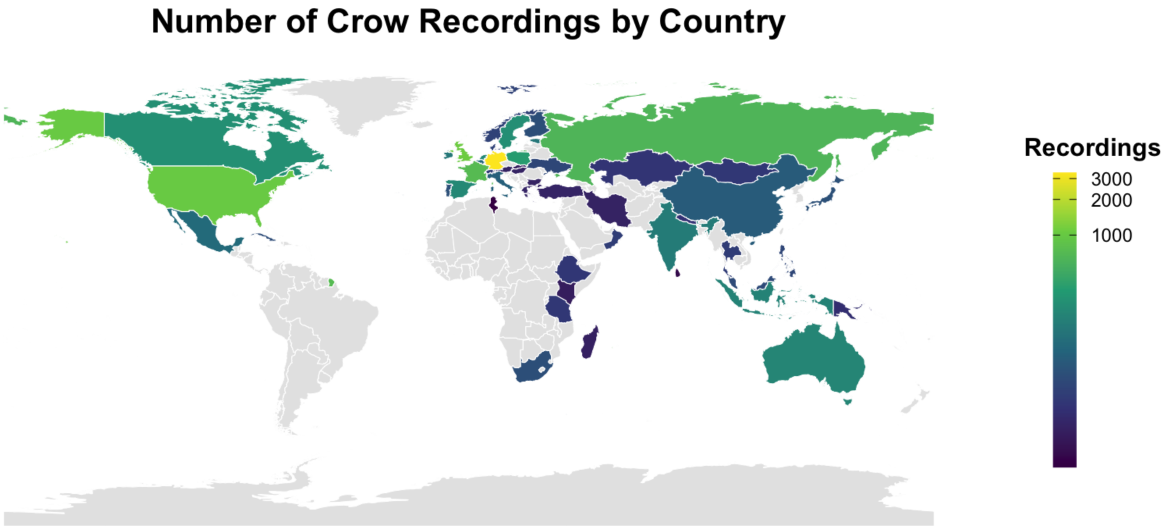
The API query returned a JSON object with the total number of relevant recordings and accompanying data. After filtering the database to exclude country/season combinations with less than 10 recordings, the final API database contains 10,718 rows and 38 data fields, including:

- **id**: the catalogue number of the recording on xeno-canto
- **gen**: the generic name of the species
- **sp**: the specific name (epithet) of the species
- **ssp**: the subspecies name (subspecific epithet)
- **grp**: the group to which the species belongs (birds, grasshoppers, bats)
- **en**: the English name of the species
- **rec**: the name of the recordist
- **cnt**: the country where the recording was made
- **loc**: the name of the locality
- **lat**: the latitude of the recording in decimal coordinates
- **lon**: the longitude of the recording in decimal coordinates
- **alt**: the elevation of the recording in meters
- **type**: the sound type of the recording (e.g., ‘call’ or ‘song’) and additional free text
- **sex**: the sex of the animal

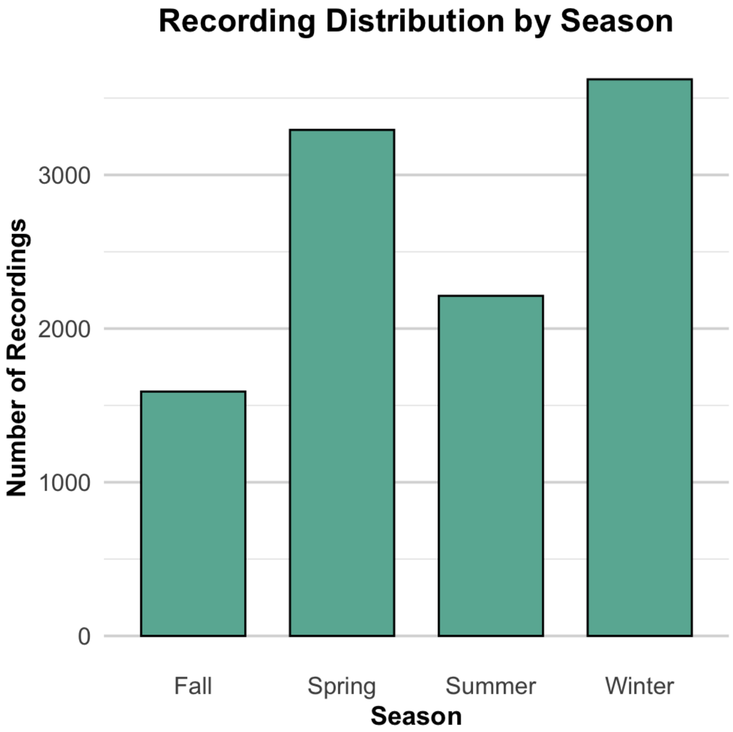
- **stage**: the life stage of the animal (adult, juvenile, etc.)
- **method**: the recording method (field recording, in the hand, etc.)
- **url**: the URL specifying the details of the recording
- **file**: the URL to the audio file
- **file-name**: the original file name of the audio file
- **sono**: an object with urls to the 4 versions of sonograms (small, medium, large, full)
- **osci**: an object with urls to the 3 versions of oscillograms (small, medium, large)
- **lic**: the URL describing the license of the recording
- **q**: the current quality rating for the recording
- **length**: the length of the recording in minutes
- **time**: the time of day that the recording was made
- **date**: the date that the recording was made
- **uploaded**: the date that the recording was uploaded to xeno-canto
- **also**: an array with the identified background species in the recording
- **rmk**: additional remarks by the recordist
- **animal-seen**: was the recorded animal seen?
- **playback-used**: was playback used to lure the animal?
- **temp**: temperature during recording (applicable to specific groups only)
- **regnr**: registration number of specimen (when collected)
- **auto**: automatic (non-supervised) recording?
- **dvc**: recording device used
- **mic**: microphone used
- **smp**: sample rate
- **season**: a conversion of date to season (fall, winter, spring, summer)

Note: the `season` data field is a new variable that was specifically created for this project.

Of the 51 countries with at least 10 recordings in at least 1 season, the majority of recordings are from Germany (3,348 recordings), the United Kingdom (1,085 recordings), the United States (1,017 recordings), France (727 recordings), and Russia (629 recordings).



The majority of recordings were captured in the winter and spring.



Below is a comprehensive list of the number of recordings per species. Note that species with less than 10 recordings were excluded from analysis, as indicated with an asterisk (*).

- American Crow: 482
- Australian Raven: 50
- Banggai Crow: 15
- Bismarck Crow: 11
- *Bougainville Crow: 5
- Brown-headed Crow: 10
- Brown-necked Raven: 19
- Cape Crow: 36
- Carrion Crow: 1,931
- Chihuahuan Raven: 64
- Collared Crow: 13
- Cuban Crow: 35
- Cuban Palm Crow: 28
- Eastern Jungle Crow: 14
- Fan-tailed Raven: 28
- Fish Crow: 136
- Flores Crow: 23
- Forest Raven: 38
- Grey Crow: 24
- Hispaniolan Palm Crow: 21
- Hooded Crow: 3,493
- House Crow: 138
- Indian Jungle Crow: 43
- Large-billed Crow: 291
- Little Crow: 10
- Little Raven: 28
- Long-billed Crow: 23
- *Mariana Crow: 4
- *New Caledonian Crow: 8
- Northern Raven: 2,346
- Palawan Crow: 22
- Pied Crow: 72
- Piping Crow: 28
- Rook: 858
- Sinaloa Crow: 43
- Slender-billed Crow: 95
- Small Crow: 26
- *Somali Crow: 3
- *Tamaulipas Crow: 9
- Thick-billed Raven: 11
- Torresian Crow: 110
- *Violet Crow: 6
- White-billed Crow: 17
- White-necked Crow: 24
- White-necked Raven: 27

METHODOLOGY

The Shiny app uses a random forest classification model trained on recordings of crows from the API to predict the most likely crow species based on two inputs: country and season of the recording. It uses a cross-validation framework to optimize accuracy and outputs the top three predicted species along with their predicted probabilities. To ensure performance and reliability, the model is trained once and cached for use throughout the session.

A Python script (`rf_model_training.py`) was created to train a random forest classifier on the crow recordings. The random forest model was used with 500 decision trees and a fixed random seed for reproducibility. Prior to training, all categorical input variables (country, season, and species name) were encoded. Recordings missing values for any of these key features were excluded from the training set, and species with fewer than 10 observations were removed to reduce noise and improve model stability.

The final dataset was split into training and test sets using stratified sampling to preserve the relative distribution of species, with `random_state = 42` for reproducibility. The model was trained to predict the encoded species label based on encoded country and season inputs. Key evaluation metrics, including accuracy, Cohen's Kappa, No Information Rate (NIR), and 95% confidence intervals, were computed on the test set. Additionally, sensitivity (true positive rate) and specificity (true negative rate) metrics were computed for each species using `classification_report` and `multilabel_confusion_matrix`, respectively. All trained artifacts, including the model, label encoders, and evaluation outputs, were serialized using `joblib` and `pickle` to enable deployment via the API.

The application backend (`api.py`) serves as the interface between the trained machine learning model and the Shiny frontend, allowing for species predictions, performance metric retrieval, and metadata access. When the API is successfully running, it returns a the message `"Corvus API is up"`. The `/predict POST` endpoint accepts a JSON request containing a `"cnt"` (country) and `"season"` value. The API applies label encoding to the inputs, uses the preloaded random forest model to generate class probabilities, and returns the top three predicted species with associated probabilities. If fewer than three classes have nonzero probabilities, only the relevant predictions are returned. The `/lookup GET` endpoint returns a dictionary mapping encoded species labels to their full names. The `/metrics GET` endpoint returns model performance metrics computed during training. The `/metadata GET` endpoint returns a cached snapshot of the full dataset used to train the model. This data is used by the Shiny app for the interactive map and data table.

The API is deployed via Google Cloud Run. The backend is built from a `Dockerfile` that defines the runtime environment and installs all Python dependencies as listed in `requirements.txt`. The image is then deployed using the `gcloud` CLI, enabling public, secure, and scalable access from the Shiny frontend.

Lastly, a Shiny app (`app_R.R`) was developed to visualize the dataset, interact with the API, and provide an interface for users to explore and predict crow species. The app is reactive, using `reactiveVal()` objects to cache metadata and `observeEvent()` blocks to respond to user inputs. The main features of the app include:

1. A map, rendered with `ggplot2` and `plotly`, that shows the number of crow recordings by country. Users can hover over specific countries to view the corresponding number of recordings.

2. A scrollable, sortable table built with the `DT` package that allows users to explore key metadata fields, including ID, Scientific Name, Common Name, Recordist, Country, Location, Type, URL, Length, Date, and Season. Playable audio clips are embedded using HTML `<audio>` tags, and URLs are rendered as clickable hyperlinks.

3. Dropdown menus allow users to select a country and season. The options for the season automatically update after a country is selected to display only valid country/season combinations based on the available data. Upon clicking the prediction button, the app sends a `POST` request to the `/predict` endpoint of the API. The API returns the top three predicted species with associated probabilities, which the app displays alongside species images (fetched directly from a GitHub-hosted image folder). Predictions with 0% probability are omitted for clarity.

4. The `/metrics` endpoint of the API provides model evaluation statistics, including true accuracy, 95% confidence interval, No Information Rate (NIR), corresponding p-value, and Cohen's Kappa. Additionally, a confusion matrix is displayed, and a summary table presents sensitivity and specificity scores for each species.

The trained random forest model is cached for reuse across sessions. All data displayed in the app is retrieved dynamically from the API or cached locally to minimize latency and redundant requests. The app was tested locally and deployed via shinyapps.io.

RESULTS

The random forest model achieved a classification accuracy of 64.4% on the test set. The true model accuracy is expected to fall between 62.6% and 66.2% with 95% confidence. This performance is significantly better than random guessing, as the No Information Rate is 32.7%. To account for agreement due to chance, Cohen's Kappa was calculated at 0.55, which indicates moderate agreement. This suggests that the model captures real patterns in the data beyond what would be expected from random guessing alone.

To better understand how the model performs across different classes, sensitivity and specificity were computed for each crow species. Overall, the model demonstrates high specificity across nearly all species, meaning that it rarely misclassifies other species as a given one. However, sensitivity varies, particularly for less common species. While the White-billed Crow is identified with perfect sensitivity and specificity, many rare or underrepresented species in the training set are not detected well. These results highlight the model's strong performance for common and well-represented species, and the need for additional data to improve classification performance for rare or underrepresented classes.