



Forest Service
U.S. DEPARTMENT OF AGRICULTURE

Pacific Northwest Research Station | General Technical Report PNW-GTR-1039 | September 2025

Workflow and Training Materials for Validating Species Detections from Passive Acoustic Monitoring: A Case Study from the Northern Blue Mountains

Ben J. Vernasco, Adam Duarte, Matthew J. Weldy,
Savannah R. Finch, and Jamie Ratliff



In accordance with Federal civil rights law and U.S. Department of Agriculture (USDA) civil rights regulations and policies, the USDA, its Agencies, offices, and employees, and institutions participating in or administering USDA programs are prohibited from discriminating based on race, color, national origin, religion, sex, disability, age, marital status, family/parental status, income derived from a public assistance program, political beliefs, or reprisal or retaliation for prior civil rights activity, in any program or activity conducted or funded by USDA (not all bases apply to all programs). Remedies and complaint filing deadlines vary by program or incident.

Persons with disabilities who require alternative means of communication for program information (e.g., Braille, large print, audiotape, American Sign Language, etc.) should contact the responsible Agency or USDA's TARGET Center at (202) 720-2600 (voice and TTY) or contact USDA through the Federal Relay Service at (800) 877-8339. Additionally, program information may be made available in languages other than English.

To file a program discrimination complaint, complete the USDA Program Discrimination Complaint Form, AD-3027, found online at [How to File a Program Discrimination Complaint](#) and at any USDA office or write a letter addressed to USDA and provide in the letter all of the information requested in the form. To request a copy of the complaint form, call (866) 632-9992. Submit your completed form or letter to USDA by: (1) mail: U.S. Department of Agriculture, Office of the Assistant Secretary for Civil Rights, 1400 Independence Avenue, SW, Washington, D.C. 20250-9410; (2) fax: (202) 690-7442; or (3) email: program.intake@usda.gov.

USDA is an equal opportunity provider, employer, and lender.

Authors

Ben J. Vernasco is an Oak Ridge Institute for Science and Education fellow, U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station, 3625 93rd Avenue SW, Olympia, WA 98512, and a research scientist, Whitman College, Department of Biology, 345 Boyer Avenue, Walla Walla, WA 99362, vernascb@whitman.edu; **Adam Duarte** is a research wildlife biologist, and **Savannah R. Finch** is an ecologist, U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station, 3625 93rd Avenue SW, Olympia, WA 98512, Adam.Duarte@usda.gov and Savannah.Finch@usda.gov; **Matthew J. Weldy** is a doctoral student, Oregon State University, Department of Forest Ecosystems and Society, Corvallis, OR 97331, Matthew.Weldy@oregonstate.edu; **Jamie Ratliff** is a wildlife program manager, U.S. Department of Agriculture, Forest Service, Wallowa-Whitman National Forest, 1550 Dewey Avenue, Suite A, Baker City, OR 97814, Jamie.Ratliff@usda.gov.

Cover photo: Ruby-crowned kinglet perched on tree branch. Photo by Michael Vodianky; FotoRequest, stock.adobe.com.

Abstract

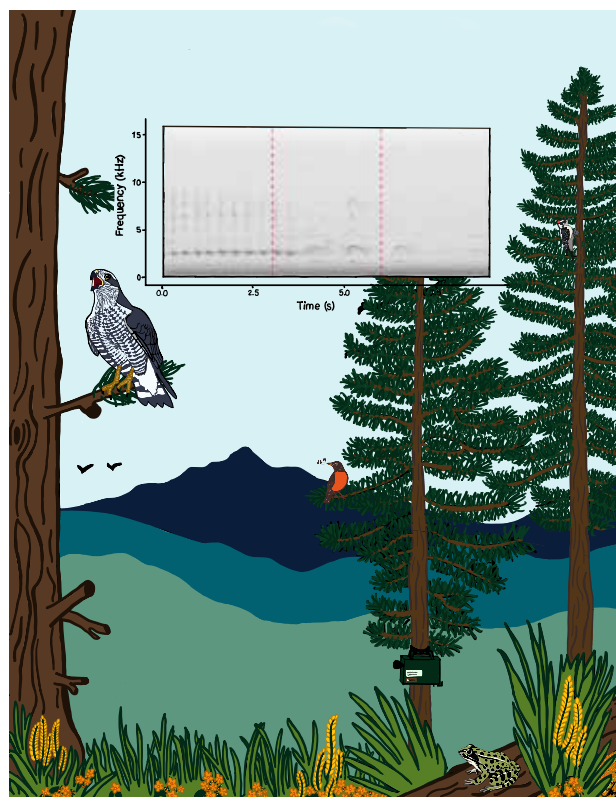
Vernasco, Ben J.; Duarte, Adam; Weldy, Matthew J.; Finch, Savannah R.; Ratliff, Jamie. 2025. Workflow and training materials for validating species detections from passive acoustic monitoring: a case study from the northern Blue Mountains. Gen. Tech. Rep. PNW-GTR-1039. Portland, OR: U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station. 54 p. <https://doi.org/10.2737/pnw-gtr-1039>.

Passive acoustic monitoring (PAM) has revolutionized wildlife monitoring by lowering barriers associated with data collection. Data from PAM can support adaptive management programs aimed at meeting multiple objectives in the face of uncertainty. Pairing PAM with acoustic classifiers capable of identifying sounds produced by wildlife can generate thousands of species detections across managed landscapes. However, such species detections represent both true- and false-positive detections, and human review is required to generate accurate detection data. Although collecting audio recordings is highly cost-effective and scalable, making reliable inferences from audio recordings presents a new challenge for wildlife practitioners faced with processing many audio recordings and reviewing automated species detections. Here, we provide guidance for wildlife practitioners using PAM and acoustic classifiers to monitor wildlife populations. We offer (1) guidance on how to effectively implement sampling designs that support robust inferences and (2) open-source code and training materials to facilitate efficient review of thousands of automated species detections. More specifically, we highlight important considerations for selecting sites to deploy autonomous recording units, the times of day to record, and potential species detections for human review. We also introduce and share open-source code to implement the Simple Passive Acoustic Monitoring protocol, a highly efficient and scalable approach for reviewing species detections generated by PAM and acoustic classifiers. We use managed forests in the northern Blue Mountains of Oregon and Washington as a case study for the techniques described in this report. Overall, this report facilitates the use of

adaptive management programs by providing wildlife practitioners guidance on how to integrate rapidly developing technologies and tools capable of effectively and efficiently monitoring wildlife populations.

KEYWORDS:

Acoustic classifier
Convolutional neural network
Wildlife monitoring
Adaptive management
Simple Passive Acoustic Monitoring protocol



Art by Lydia Morgan

Contents

1	Introduction
4	Sampling Guidelines and Considerations
4	Examples of Sampling Schemes
6	Selecting Sampling Sites
7	Choosing When to Record
8	Selecting Automated Species Detections to Review
14	Example of a Workflow of the Simple Passive Acoustic Monitoring Protocol
16	Creating Clip Extraction File
27	Creating WAV Files of Species Detections
29	Creating Spectrograms of Species Detections
33	Combining Spectrograms and Audio Clips into MP4 Files
34	Creating Species Detection Processing File
39	Sound Library Development and Reviewer Training and Assessment
39	Sound Library Development
40	Assessing Reviewer Accuracy
41	Conclusions
41	Acknowledgments
42	References
47	Appendix 1: Minimum Information Needed to Extract Audio Clips of Species Detections
49	Appendix 2: Example of an Audio Clip Validation Protocol
49	Useful Links for Learning and Identifying Species Vocalizations
49	Validation Protocol
53	Appendix 3: Species List

Introduction

The multiple-use mission of the U.S. Department of Agriculture, Forest Service requires the agency to implement land management strategies that achieve multiple objectives, including those associated with timber production, wildfire, recreation, and fish and wildlife. Land managers often use adaptive management, a decision-support framework, to meet multiple-use objectives because it explicitly incorporates various sources of uncertainty while informing recurrent management decisions (Gannon et al. 2024, Johnson et al. 2011, Moore and Conroy 2006, Temperli et al. 2012). The adaptive management framework, however, is reliant on accurate monitoring data to understand the current state of the ecosystem being managed, to assess how the ecosystem is responding to management, and to update or refine hypotheses related to ecosystem dynamics—often termed “learning by doing” (Larson et al. 2013, Lyons et al. 2008, Peterson and Freeman 2016). Yet, the needed monitoring data have historically been highly demanding to collect in terms of resources and time, particularly when collection occurs at the landscape scale (Witmer 2005).

Modern technologies and rapidly developing quantitative methods are beginning to facilitate the collection of a wealth of data associated with wildlife (Lahoz-Monfort and Magrath 2021, Tosa et al. 2021) and the habitats on which they rely (Barry et al. 2024, Bell et al. 2024, Camarretta et al. 2020). Herein, we focus on the use of passive acoustic monitoring (PAM) and acoustic classifiers to collect, process, and validate wildlife occurrence data. PAM uses relatively simple technology, often referred to as autonomous recording units (ARUs), to record soundscapes for extended durations of time across large geographic areas. PAM is revolutionizing our ability to monitor wildlife populations at large spatial scales (Darras et al. 2019, 2025; Ross et al. 2023). In particular, the equipment and effort required to implement PAM has become less cost prohibitive, particularly relative to hiring field technicians to survey similar amounts of area for the same amount of time (e.g., Borker et al. 2015). Moreover, PAM reduces the impact of monitoring activities on target wildlife species (e.g., eliciting behavioral responses using call-back surveys), while allowing for the collection of monitoring data for the larger wildlife community. Examples of successful PAM applications in the context of wildlife monitoring include monitoring associated with regulatory compliance when implementing land management activities (Appel et al. 2023, Gaylord et al. 2023), identifying areas that support wildlife reproduction (Chronister et al. 2024, Dale et al. 2022, Reid et al. 2022), understanding the impacts of invasive species on native species (Appel et al. 2023, Bota et al. 2024, Hopkins et al. 2022, Rugg et al. 2023), quantifying the effects of wildfire on species distributions (Brunk et al. 2023, 2025; Wood et al. 2024), and detecting the prevalence of illegal activities in protected areas (Astaras et al. 2017, Braulik et al. 2017).

Although amassing thousands of hours of soundscape recordings is now feasible, extracting meaningful information (e.g., species occurrence data) from large databases of sound recordings remains a significant challenge (Hartig et al. 2024). Fortunately, open-source acoustic classifiers, such as BirdNET (Kahl et al. 2021), PNW-Cnet (Ruff et al. 2023), HawkEars (Huus et al. 2025), and Perch (Ghani et al. 2023), have been developed to help tackle this problem.¹ Acoustic classifiers are machine learning algorithms that calculate a confidence score, often ranging 0–1, for discrete time segments within audio

¹ The use of trade or firm names in this publication is for reader information and does not imply endorsement by the U.S. Department of Agriculture of any product or service.

recordings (e.g., 3 seconds) for each sound class the acoustic classifiers can recognize. Sound classes can range from distinct avian calls/songs to anthropogenic noises such as gunshots. Notably, the confidence scores do not represent the probability that a detection is a true-positive detection. Confidence scores are rather unitless values that represent the acoustic classifier's confidence in a detection, and the relationship between the confidence score and probability of correct classification is highly context specific (for more discussion on confidence scores see below and Wood and Kahl 2024). Importantly, the automated detections provided by acoustic classifiers include both true- and false-positive detections. A true-positive detection is a detection that contains enough information that a human reviewer can confirm the audio contains the target sound class. A false-positive detection is a detection that does not contain the target sound class or does not contain enough information for a human reviewer to confidently confirm that the audio segment contains the target sound class. The presence of both true- and false-positive detections requires that some amount of human review of automated detections occurs before inferences can be made.

Wildlife practitioners often use acoustic classifiers to analyze large audio datasets (e.g., >10,000 hours of recordings) and prioritize detections for human review using the resulting confidence scores. In some cases, using confidence scores to prioritize human review lowers the barrier to implementing PAM by reducing the technician time required to review the audio recordings that are used to generate species occurrence data. Despite the potential for confidence scores to guide the manual review process, it is important to take into consideration the process used to review and validate detections. Minor workflow adjustments can lead to substantially reduced or increased technician time to extract meaningful information from the audio recordings when done at scale. Basic principles of sampling design as well as concepts specific to using acoustic classifiers to generate species detections can be considered early in the project development process to maximize efficiency. Guidance related to making informed audio processing decisions after collecting hours of audio recordings is currently limited. Additionally, while open-source software for reviewing species detections to build or further develop acoustic classifiers is available (Balvanera et al. 2025, Lapp et al. 2023), simple, open-source software for processing large quantities of species detections from PAM for the purposes of landscape-scale wildlife monitoring is not readily available.

In this report, we present the Simple Passive Acoustic Monitoring (SPAM) protocol, an efficient workflow for validating target sound class detections from large-scale PAM projects. First, we provide a description of some sampling design considerations important to successful implementation of PAM. Next, we describe the workflow and code associated with implementing the SPAM protocol (fig. 1) to validate automated species detections. The code used to implement the SPAM protocol is written for users with previous experience using the R programming language. The SPAM protocol is open-source and suitable for projects with limited budgets. Major processing steps can also be scaled, using parallel processing, to meet the demands of large-scale PAM projects. The resulting MP4 files, which contain the audio and visual representation (i.e., spectrogram) of species detections, are small files (e.g., about 200–900 KB each), and large numbers of the species detections can easily be shared and reviewed on cloud-based platforms. Overall, the SPAM protocol facilitates simple and efficient review of automated species detections and the generation of species occurrence data from large, landscape-scale PAM projects.

We demonstrate the construction and use of a SPAM protocol with an example of a PAM program that is being implemented on the Umatilla and Wallowa-Whitman National

Forests, within the northern Blue Mountains of Oregon and Washington, USA (Duarte et al. 2024a). This case study uses audio recordings collected by ARUs and automated species detections generated by BirdNET v2.4 (Kahl et al. 2021). We used BirdNET because of its accessibility and ability to classify sounds made by most of the avian community in our study area. We also include a sound library for 107 forest-associated avian species identified in the northern Blue Mountains and describe how we created this diagnostic sound library. Lastly, the reliability of species detection data from PAM depends on accurate human review of automated species detections. We provide a library of up to 50 previously reviewed and annotated species detections for each avian species in the northern Blue Mountains sound library to facilitate the training and quantification of reviewer accuracy. Importantly, the objective of this report is not to catalog or even evaluate every acoustic classifier (or how to use them) or describe all the approaches that can be used to analyze species detection data generated by PAM. The field of PAM is rapidly transforming, and such a review would quickly be outdated. Instead, we aim to provide and demonstrate the use of clear guidelines and accessible code to promote more efficient generation of reliable detection data from PAM efforts.

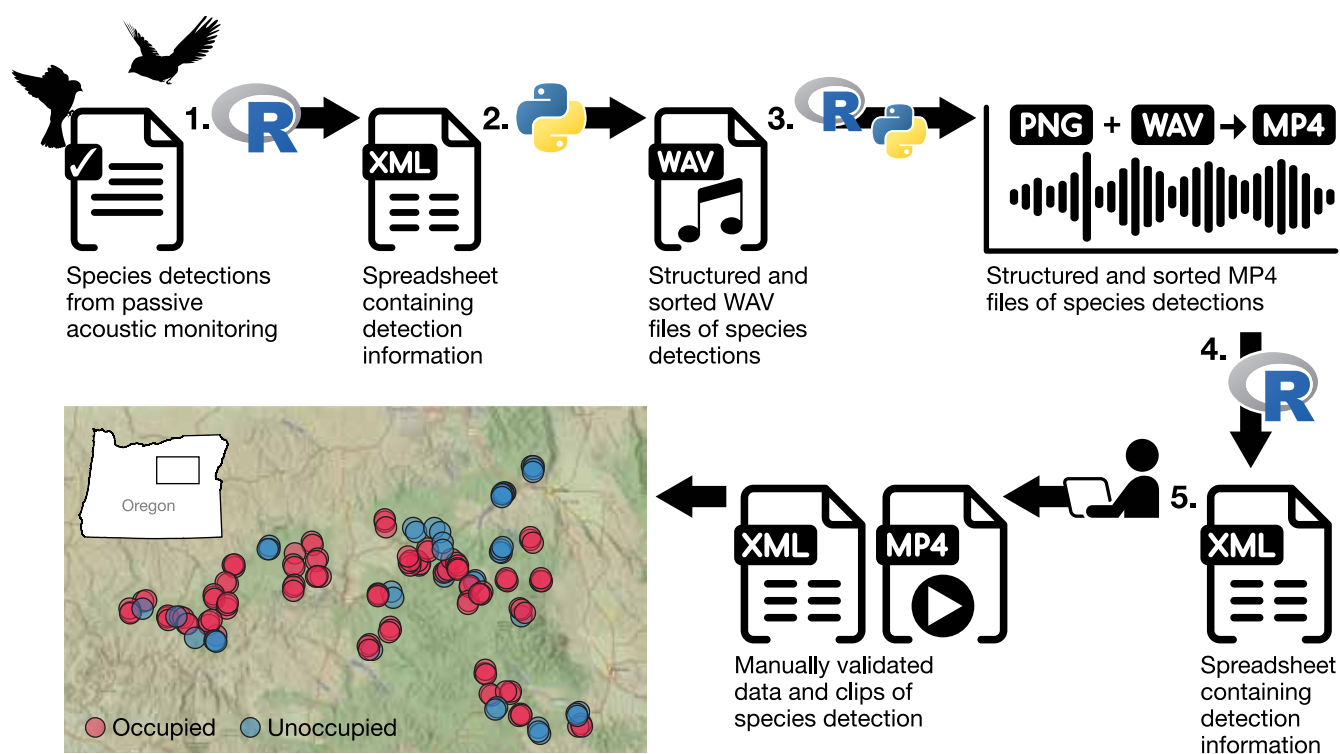


Figure 1—Workflow of Simple Passive Acoustic Monitoring protocol. Arrows connect each step of the protocol and symbols above arrows denote the program(s) used between each step. This protocol begins with a collection of species detections, generated by processing audio recordings with an acoustic classifier, such as BirdNET or PNW-Cnet, in the form of multiple CSV files. (1) Program R is then used to collate and select detections to be reviewed, organize information related to where and when species detections occurred, and create a CSV file containing information needed to extract WAV files of species detections. (2) The CSV file is processed using Python to extract short WAV files containing species detections from raw audio files with user-specified filenames and folder structures. (3) Program R is used to create PNG files containing visual representations of the sound within each WAV file (i.e., spectrograms), and Python is used to combine the WAV and PNG files into MP4 files that facilitate visual and auditory assessment of species detections. (4) The filenames and folder structure of the library of MP4 files are processed in Program R to create spreadsheets for entering information about each species detection during manual review. (5) Species detections are manually reviewed using the MP4 files to generate species occurrence data, exemplified by a map of occupied and unoccupied sites in the northern Blue Mountains, for further statistical analysis. In this map, red circles are places in which the species of interest was detected and blue circles are places where the species of interest was not found.

Sampling Guidelines and Considerations

The scale and relative feasibility of PAM has revolutionized wildlife monitoring. The ease of generating large PAM datasets, however, does not minimize the importance of investing significant time for considering the sampling design, nor does it compensate for a poor sampling design. When conducting PAM, sampling occurs at a minimum of three different levels. First, surveyors need to identify where in the landscape to place the ARU(s). Second, surveyors need to define their sampling window in terms of both days within the year and minutes within a day to collect data. Third, during the postprocessing and analysis stage, surveyors need to consider how detections are sampled for human review and validation. Considering all three of these sampling levels is essential to making accurate inferences from PAM data and correctly informing land management strategies aimed at meeting multiple-use objectives. In this section, we briefly review some aspects of each of these levels of sampling. Arguably, the most important consideration for implementing PAM is that all levels of the sampling design are directly tied to the monitoring objectives and the resources available to collect, process, validate, and analyze the audio data.

Examples of Sampling Schemes

Four common sampling schemes that are used at all three levels of sampling in PAM are simple random sampling, stratified random sampling, systematic sampling, and adaptive cluster sampling (fig. 2). There are excellent resources that review these and other sampling designs in detail (Morrison 2008; Thompson 2004, 2012; Thompson et al. 1998), so we only provide a brief introduction to each here.

- **Simple random sampling** is the most straightforward to implement because sample selection does not require additional information (e.g., locations to deploy ARUs, detections to validate). Here, we select a group of samples from all possible samples (i.e., the population) at random in such a way that each sample has an equal chance of being selected. The advantage of this design is that it provides an unbiased set of samples from the population. However, by random chance, the selected samples may not provide coverage in space (i.e., samples may be clustered) or in representation of all possible conditions (e.g., burned and unburned sites, high- or low-confidence score detections), particularly when the number of samples selected is relatively low compared to the size of the population.
- **Stratified random sampling** is similar to simple random sampling, except that the samples are first separated into groups (e.g., based on the land management treatment or confidence score values), and a random selection of samples occurs within each of those groups. This sampling scheme increases the representation of the specific conditions, as defined by the groups, within the selected samples and can facilitate improved inference by generating more precise and accurate parameter estimates. However, stratified random sampling requires prior information about the samples to make the groupings and may lead to spatial or temporal clustering in the samples that are selected. Additionally, the groups may change over time. For example, forest composition and structure will undoubtedly change via forest management, disturbance, and succession, and static sampling sites will influence the extent to which all groups are equally represented over time. Thus, stratified random sampling may not always represent an improvement from simple random sampling.

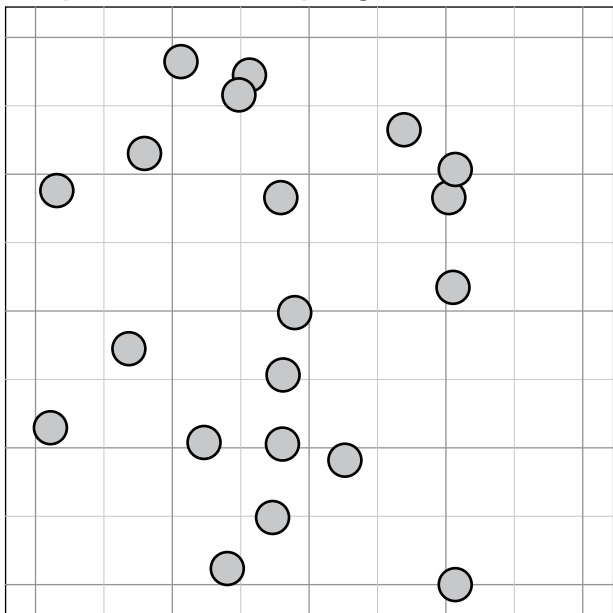
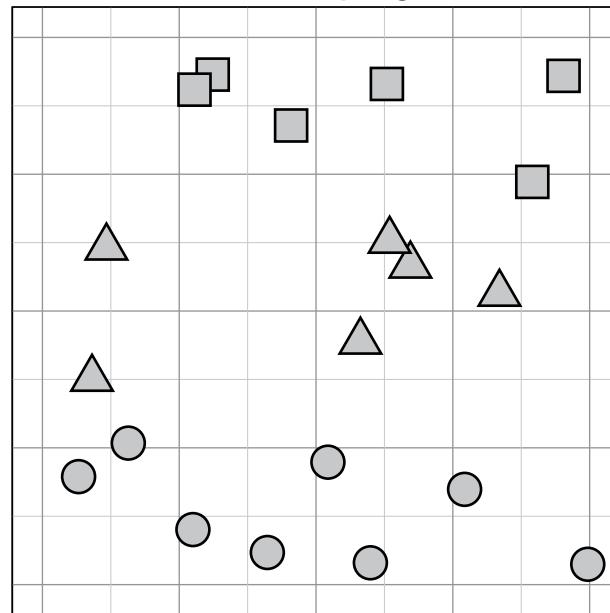
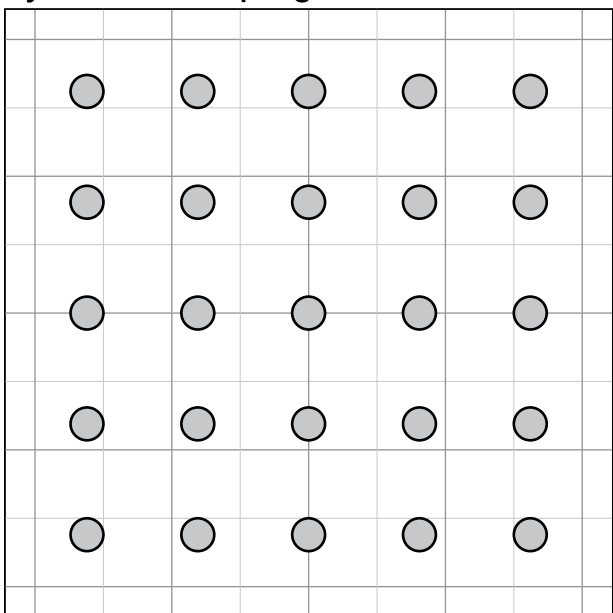
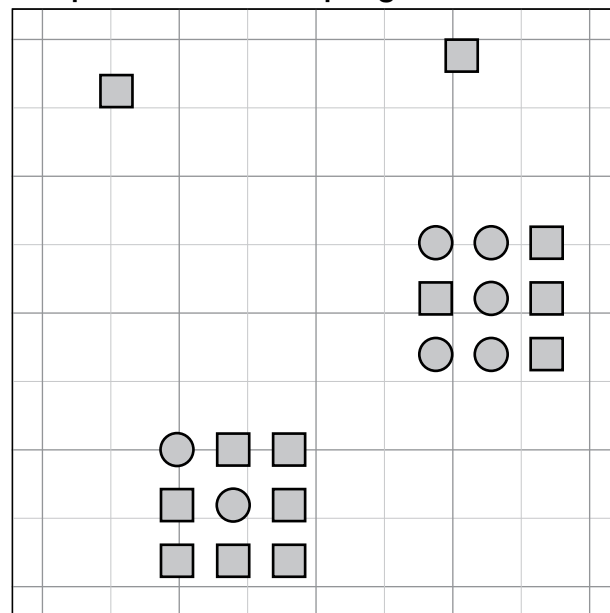
Simple random sampling**Stratified random sampling****Systematic sampling****Adaptive cluster sampling**

Figure 2—Each graph reflects the application of one of the four sampling designs used to select sampling locations, though each sampling design can also be used to select when to sample and which detections to validate. In the graph showing the stratified random sampling, the shape of the points represents the different groups (e.g., land management treatments) used to stratify sites. In the adaptive cluster sampling example, the shape of the points refers to positive (circles) and negative (squares) detections. In this adaptive cluster sampling scheme, sampling around initial positive detections (i.e., the center circle in each cluster) only occurred at adjacent sites and did not continue to expand around subsequent positive detections. Squares not surrounded by point clusters represent initial samples that were not positive detections.

- **Systematic sampling** begins by defining the extent and spacing of a lattice of sampling locations. This lattice forms a sampling grid that is overlaid on the population (e.g., study area). Typically, the sampling grid is anchored in space by randomly selecting a location that is tied to some aspect of the sampling grid (e.g., the center or corner of the grid). The benefit of systematic sampling is that samples are consistently spaced across a geographic area or consistently through time. However, systematic sampling can

require a substantial number of samples depending on the grid spacing and extent. The extent of the grid and spacing of samples also determines the conditions being sampled. Therefore, as with simple random sampling, it is not guaranteed that the samples represent all conditions of interest.

- **Adaptive cluster sampling** is a unique sampling design in which any of the above types of sampling may occur, but there is additional, clustered sampling close to positive outcomes from the sample. For example, if a target species is detected at a survey station (hereafter, station), adaptive cluster sampling adds samples in adjacent sample units until all sample unit surveys yield no positive detections. Like systematic sampling, this sampling scheme can require considerable resources to implement. Species detections, for instance, would be reviewed at regular intervals throughout the survey period, and the review of subsequent samples would be determined based on the initial reviews of detections. Such effort may not be feasible for every monitoring program. It is important to consider, however, that adaptive cluster sampling tends to yield more precise estimates when surveying for rare or clustered target species.

Selecting Sampling Sites

Choosing where to sample can be a difficult decision and is made based on monitoring objectives. Some wildlife practitioners have placed a limited number of ARUs in areas expected to have a target species to help guide subsequent surveys. Although this modified form of adaptive cluster sampling (i.e., where the initial sampling location is not random) may be useful in the short term, such data have little use for making inferences outside of the immediate objective. Examples of uses of such data include confirming species presence in areas where they are suspected present and acting as validation data for species distribution or habitat suitability models (reviewed by Duarte et al. 2025). Others have chosen to restrict ARU deployments to areas planned for management treatments, such as timber harvests or prescribed burns, to effectively fulfill any regulatory monitoring requirements. Although this sampling approach may amass a considerable amount of data over time, how project areas and the survey stations within them are sampled may limit the usefulness of these data for predicting the status and trends of wildlife across the larger landscape (Baumbusch et al. 2025). If the objective is to estimate status and trends in species distributions, to predict species occurrence in unsampled areas of interest across managed forests (e.g., to new areas proposed for timber harvests), or to use the data to answer questions related to the larger wildlife community, a more robust approach is to develop a sampling extent across the forest of interest using one or more of the other sampling schemes outlined above. In the northern Blue Mountains, adaptive cluster sampling, simple random sampling, and stratified random sampling are implemented at different extents. For example, wildlife practitioners have placed ARUs in areas suspected to have American goshawks to help guide where on-the-ground surveys may prove fruitful for finding nest locations. This approach has been successful. However, because of the limitations of this sampling scheme, we implemented a two-scale monitoring design focused on project-level monitoring and landscape-level monitoring (Duarte et al. 2024a).

For landscape-level monitoring, we implemented a design that closely matches the design implemented for northern spotted owls (*Strix occidentalis caurina*) on the west side of the Cascade Range in Oregon and Washington (Lesmeister and Jenkins 2022). More

specifically, we created a 500-hectare hexagon tessellation that spanned the Wallowa-Whitman and Umatilla National Forests and omitted any hexagon containing <50 percent forest-capable lands (as defined by Davis et al. 2015) that are managed by the Forest Service. Because of limited equipment and a desire to help inform active forest management, we omitted any hexagons that predominately intersected with wilderness or experimental forest lands. We then used simple random sampling to select hexagons. We also implemented a modified rotating frame design by designating hexagons as either static or rotating (e.g., Duarte et al. 2020). Static hexagons are sites that we sampled every year, and rotating hexagons are sites that we sampled once every 3 years. Static hexagons allow for the estimation of long-term changes in the wildlife community, and rotating hexagons allow for the estimation of spatial heterogeneity in the wildlife community. The modified rotating frame design reduces the number of ARUs needed per year while surveying across the two national forests.

For project-level monitoring, we collaborated with Wallowa Resources, an organization that currently measures fine-scale vegetation metrics at plots within areas proposed for active forest management. Wallowa Resources selected survey sites using a stratified random sampling design wherein they classified land within project area boundaries by topographic position (e.g., elevation and aspect), vegetation type (e.g., plant association or potential vegetation groups), and planned forest management treatment (e.g., commercial thinning, precommercial thinning, prescribed fire). Plot locations were then randomly selected within those classifications. We aimed to pair our ARU deployments with these vegetation plots. However, ARUs sample a larger area around the station (i.e., the area that sound can travel), and we have a limited number of ARUs. Thus, we randomly sampled a subset of the vegetation plots to colocate ARUs while maintaining a minimum distance of 250 meters among ARUs to minimize the odds of simultaneously recording the same sound at two adjacent stations. Finally, both levels of our sampling design required that ARU deployments be at least 50 meters from roads, trails, and loud streams. This reduced environmental noise, which can make it difficult to detect target sound classes, and minimized the likelihood of tampering with, and theft of, our equipment.

Choosing When to Record

Once the sites are determined for a PAM program, some consideration can be given to the timing of recordings. This occurs at two levels—minutes within a day and days within a year. Given a limited number of ARUs, shorter deployment periods allow ARUs to be moved more frequently and sample a larger number of sites. However, longer deployment periods increase opportunities to detect cryptic or rare species that infrequently produce sounds. Evaluating the tradeoff between sampling a greater number of sites and increasing the probability of detecting a cryptic or rare species if it is present at a given site will depend on the behavior and distribution of the species of interest as well as monitoring objectives. Within each 24-hour day, the recording schedule should include periods when the target species is most likely to produce sounds. In some cases, however, prior information concerning when animals make sounds can be biased by when surveyors were able to access remote forest environments. Furthermore, one of the benefits of using ARUs is the ability to simultaneously collect data on the broader wildlife community, which may not produce sound in concert with target species. We have found that the most significant tradeoffs in the number of minutes within a day and the number of days within a year to record sounds are

in the audio recording storage and processing steps. Obviously, if a PAM program chooses to record for longer durations, the equipment needed to store and backup these audio recordings and the time needed to process the audio recordings increases considerably.

In the northern Blue Mountains, we collect most of our audio data from early May to late June, which aligns with the nesting season for most avian species in the region. This timing of deployment aligns our PAM with when species are available for detection (i.e., active calling related to the reproductive cycle and when migrants are present in the region), and it allows us to meet assumptions of population closure when fitting various statistical models to the validated species detection data (i.e., species space-use patterns tend to be linked to the reproductive cycle). The ARU deployments may begin earlier in the season if access is not limited by snow. We typically do not have ARUs deployed beyond mid- to late September, which aligns with the availability of crews, includes the postfledging period for most avian species in the region, and often occurs before the region's wildfire season. We deploy an ARU at a survey station for a minimum of 2 to 3 weeks at a time, although we often deploy ARUs for longer periods based on the logistics of retrieving them. We chose this deployment duration to maximize our ability to detect rare or cryptic species and to ensure that we had the replication data required to fit statistical models (e.g., occupancy models, N-mixture models) that account for imperfect detection. Furthermore, we use a systematic recording schedule by setting the ARUs to record continuously from 2 hours before until 2 hours after sunrise; 1 hour before until 3 hours after sunset; and 10 minutes at the top of every hour that is not included in the previously described time windows. This matches what was done for northern spotted owls on the west side of the Cascade Range (Lesmeister and Jenkins 2022). This recording schedule enables us to monitor the dawn and dusk chorus as well as small portions of time throughout the day, which allows us to process the data for all forest-associated avian species that make distinct sounds throughout the day.

Selecting Automated Species Detections to Review

Unlike the sampling processes above, there is considerably more variation in how audio recordings made by ARUs are reviewed. Some practitioners may opt to manually listen to and inspect spectrograms of all raw audio recordings or a random selection of segments from audio recordings (e.g., Cameron et al. 2020). These approaches preclude the need for an acoustic classifier, but they can be cost-prohibitive when done at the scale of a large PAM program. Similarly, some practitioners may be tempted to validate all species detections that an acoustic classifier generates. However, depending on the scale of the project, acoustic classifiers can generate thousands of species detections. Validating all the detections would require a substantial investment in technician time and likely impede the timely delivery of information to managers. Moreover, management is rarely, if ever, informed by a complete census. Instead, we typically attempt to implement robust sampling designs that allow us to leverage contemporary quantitative methods to estimate the parameters of interest (e.g., occupancy probability, abundance/density). Thus, in this section, we focus on using acoustic classifiers to help prioritize detections for validation before downstream statistical analyses. In the northern Blue Mountains, we have used a combination of approaches depending on the objectives of the data processing effort and information gained as the project continues to develop and grow. We briefly describe these approaches and others that are currently most commonly used to validate detections with the assistance of acoustic classifiers. However, these methods are a quickly evolving component of PAM.

One of the most common approaches to validating species detections is to identify a confidence score threshold and prioritize human validation efforts by filtering detections to those with a confidence score greater than or equal to the threshold. For example, Appel et al. (2023) restricted manual validation of all potential barred owl calls to detections with a confidence score of ≥ 0.95 . From the reviewed detections, they summed the number of true-positive detections for use as a covariate when quantifying the effects of barred owls on northern spotted owl occupancy and pair status. Duarte et al. (2024b) approached their validation effort similarly when quantifying the intensity of forest habitat use of marbled murrelets (*Brachyramphus marmoratus*); however, they used estimates of the precision of the acoustic classifier (i.e., the proportion of detections confirmed as true positives in a subset of potential calls) to reduce the technician time spent reviewing detections. In particular, they estimated the number of marbled murrelet detections as the number of detections with a confidence score of ≥ 0.95 multiplied by the estimated precision for detections at or above that threshold. An important consideration is that as the threshold used to decide if a species detection is reviewed increases, the number of true-positive detections that are not reviewed can also increase. The threshold implemented for a particular project is therefore best considered in the context of PAM objectives. For example, PAM focused on northern spotted owl often validates all detections with a confidence score of ≥ 0.25 (Appel et al. 2023, Weldy et al. 2023). Although certainly more time-consuming, this low threshold results in a lower chance of missing true-positive detections, which is important when dealing with an endangered species whose presence has direct implications for land management decisions.

The confidence scores are, to some extent, related to the probability that a species detection represents a true- or false-positive detection. There are two approaches that take advantage of this relationship. Both approaches require the review of a random sample of automated detections across a range of confidence scores. Using the first approach, a random sample of automated detections across the full spectrum of confidence scores are selected (Wood et al. 2023a, 2023b, 2024). A logistic regression model is then fit using the mix of true- and false-positive human-validated detections to predict the probability that a detection represents a true-positive detection based on its confidence score. A sum of the number of detections above a priori specified probability of a true positive (e.g., 0.9) for each site is then calculated. While effective, this approach suffers from the fact that it ignores uncertainty in the predicted probability. Moreover, the effectiveness of the approach depends on the strength of the relationship between the confidence score and the probability of a true positive, which is known to be species-specific (Sethi et al. 2024). We provide estimates of this relationship for 63 forest-associated species in the northern Blue Mountains in figure 3.

The second approach aims to account for potential false-positive detections to estimate call density, where call density is the estimated proportion of automated species detections that represent true-positive detections regardless of the confidence score (Navine et al. 2024b). Using this approach, automated detections are stratified into bins defined by logarithmic quantiles based on the distribution of confidence scores (e.g., 0.00–0.5, 0.51–0.75, 0.76–0.875, and 0.876–1.000), and a fixed number of randomly selected samples within each quantile undergo human review. Using this collection of validated detections, which represent both false-positive and true-positive detections, rules of probability are used to quantify a call density estimate (and the associated uncertainty) with a bootstrapping procedure. Navine et al. (2024a) subsequently demonstrated that, among Hawaiian honeycreepers (family: Fringillidae), there is agreement between the call density estimates generated from PAM and bird density estimates measured by point counts and distance sampling.

Figure 3—The relationship between the probability of correct classification and the BirdNET confidence score (see full caption below)

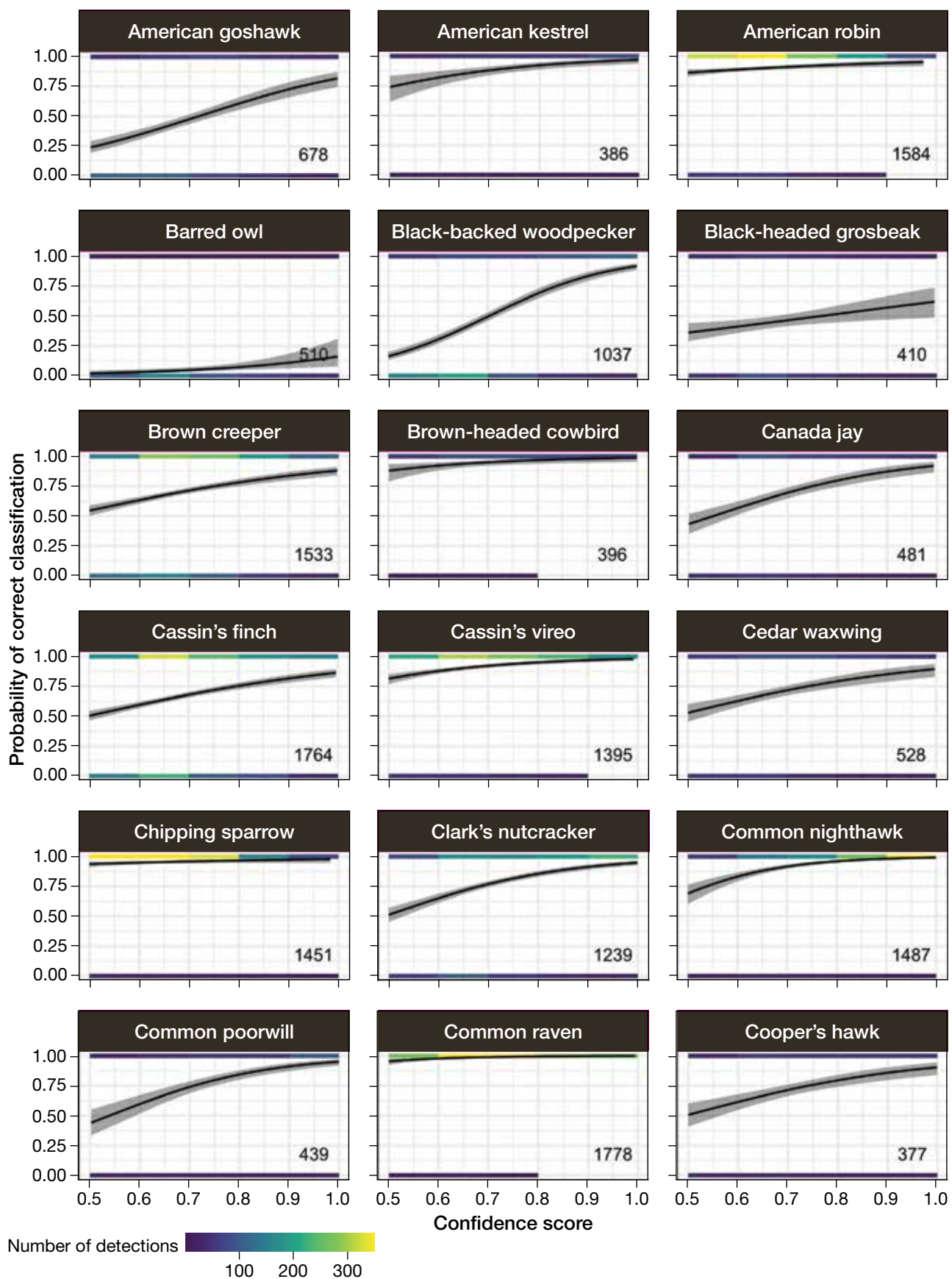


Figure 3 (continued) —The relationship between the probability of correct classification and the BirdNET confidence score (see full caption below)

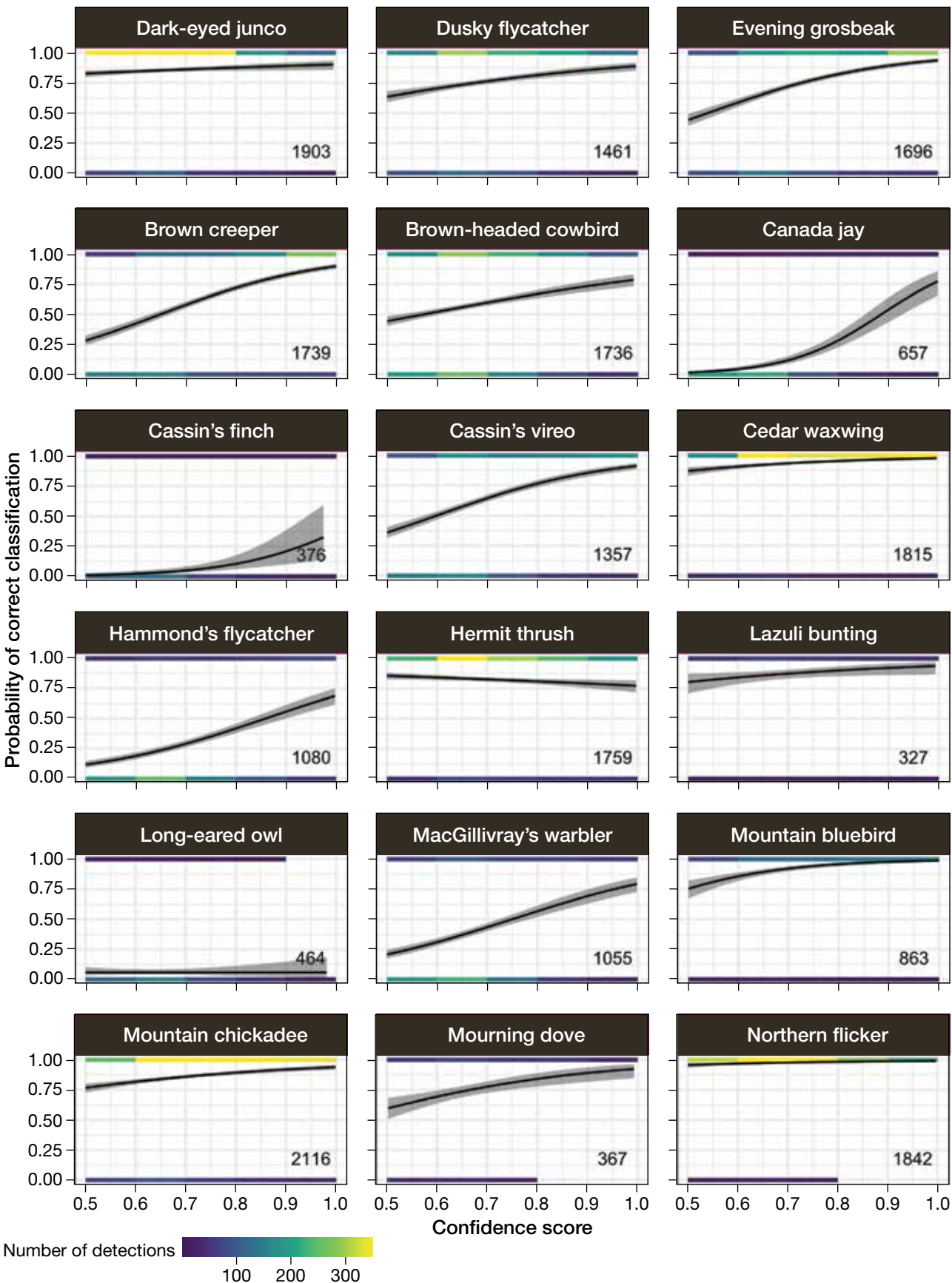
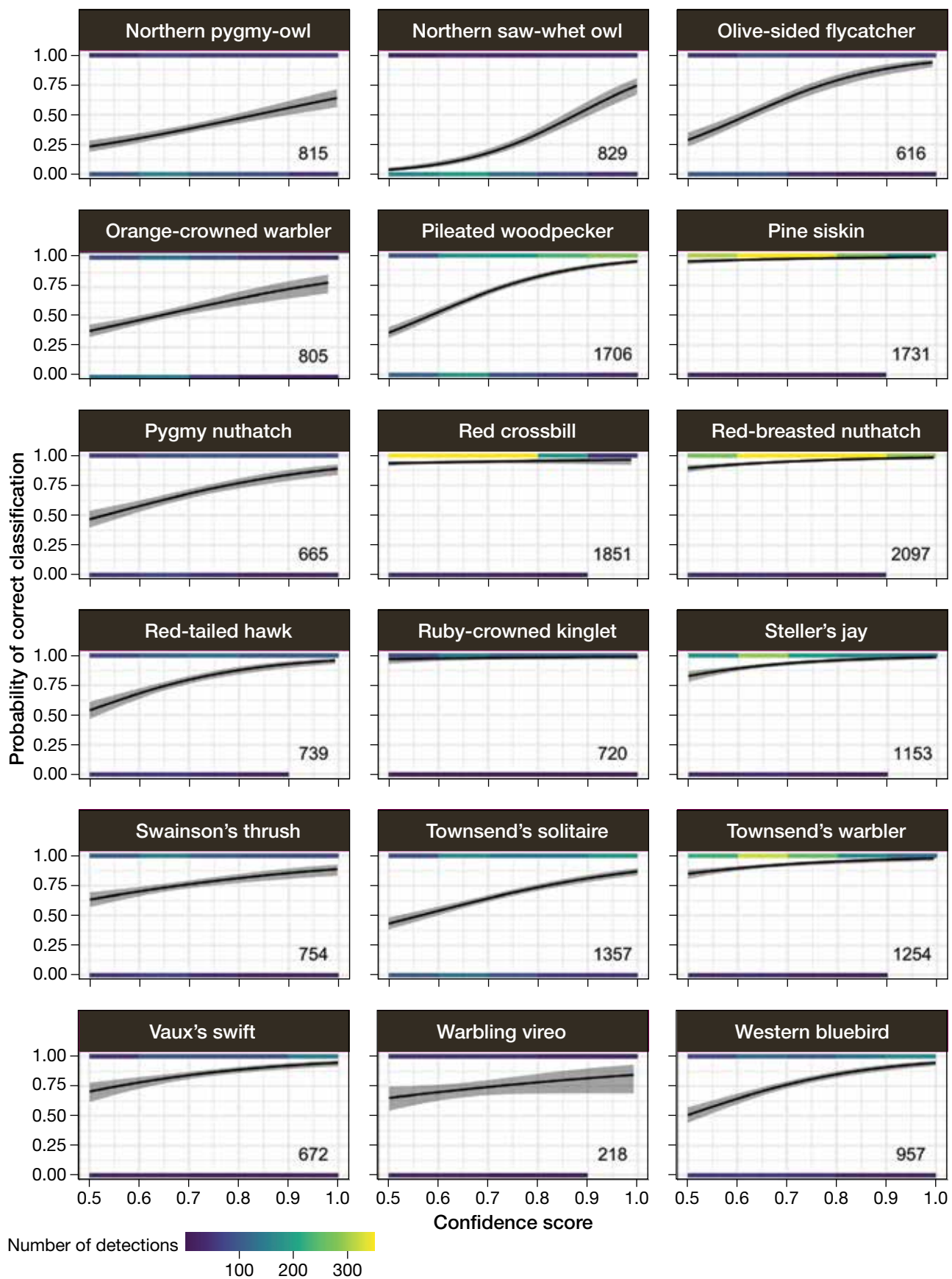


Figure 3 (continued)—The relationship between the probability of correct classification and the BirdNET confidence score (see full caption below)



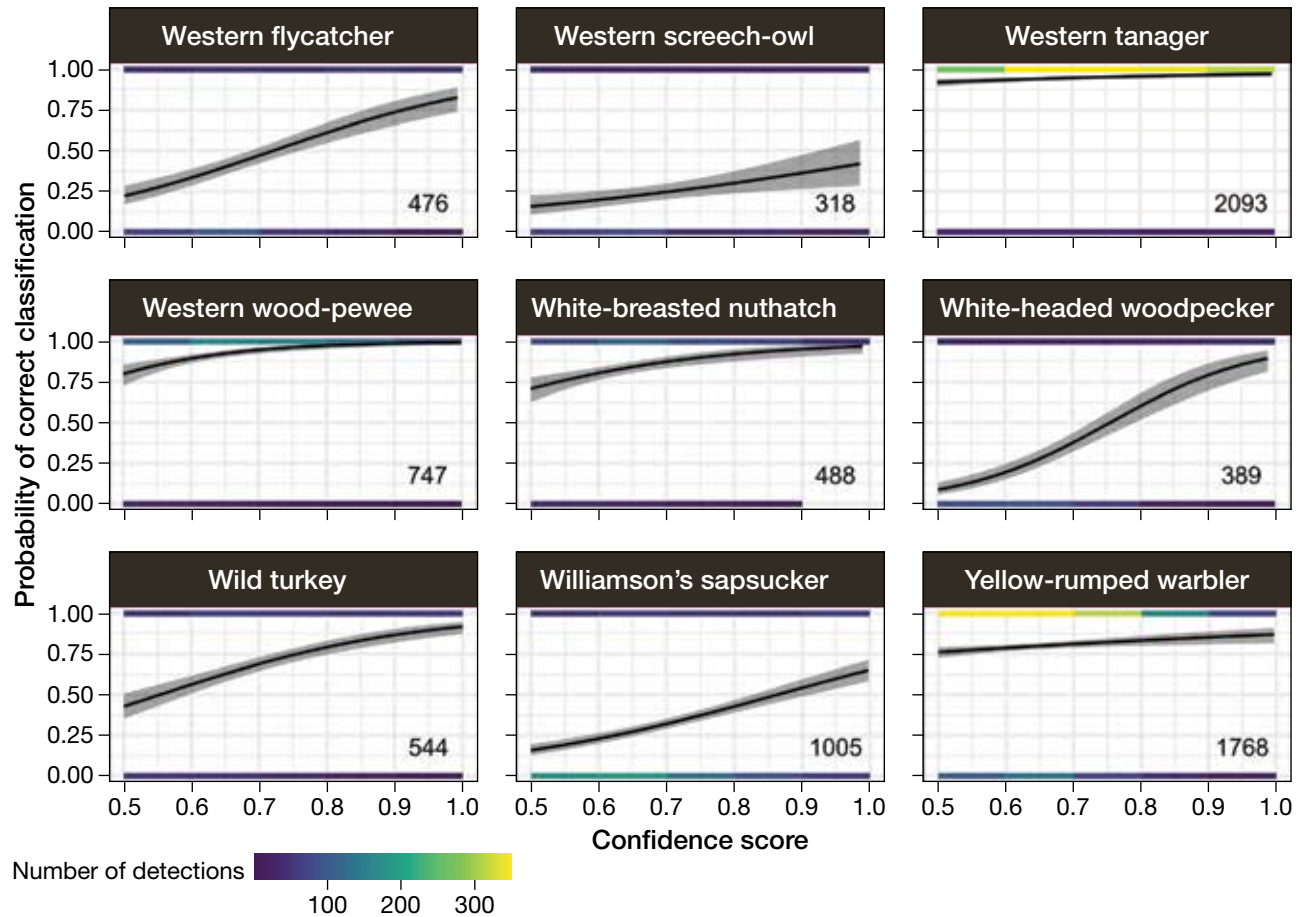


Figure 3—The relationship between the probability of correct classification and the BirdNET confidence score can vary considerably among forest-associated species of birds in the northern Blue Mountains. Correct classification indicates the species detection contained enough information to confidently confirm the target species presence. Black lines show species-specific predicted values, and the ribbon surrounding each line denotes the 95-percent confidence interval, calculated using the visreg package (Breheny and Burchett 2017). Logistic regression models were fitted using the glmmTMB package (Brooks et al. 2017). Colored lines at the top and bottom of each graph denote the number of human-validated species detections present within a given confidence score bin (e.g., 0.5–0.6, 0.6–0.7) for both true- and false-positive species detections. Bins containing greater than 350 detections were given a value of 350 to facilitate readability; 29 of the 736 bins have >350 detections. Inset numbers within each species panel denote the total sample size used to fit the logistic regression model for that species. Before model fitting, species were filtered to those that were forest associated and had a minimum of 200 detections within the northern Blue Mountains dataset. Species detections were from monitoring that occurred in 2022 and were validated using the protocol provided in appendix 2. Only detections with a confidence score greater than 0.5 were reviewed. For each survey week at each station, only the first three reviewed detections for each species were included in the logistic regression analysis. All detections were reviewed by trained observers and only call types present in the provided "diagnostic clips" folder were used to classify a detection as a true positive. See appendix 3 for the scientific names of all species included here.

All the above approaches focus on identifying true-positive detections via direct human validation or by estimation based on a subset of reviewed detections in which practitioners try to remove false-positive detections from their data before they implement subsequent analyses. However, none of the above approaches on their own account for false negatives—when a sound of interest is present but not detected. False negatives can arise as a result of the classifier not detecting a true positive or the sound not being recorded by the ARU (e.g., the recording schedule does not specifically match when a target species vocalizes, ARUs are not deployed long enough to record present rare/cryptic species, or equipment malfunctions). There have been some additional analytical developments that aim to account for both false-negative and false-positive detections.

Some initial efforts used occupancy models that account for false-negative and false-positive detections and required practitioners to classify detection data for a sample period as no detection, uncertain detection (i.e., a PAM detection based on a confidence score threshold from an acoustic classifier), or certain detection (e.g., mist net captures, a subsample of validated detections) (Banner et al. 2018, Clement et al. 2014). Using this approach, only a single potential positive detection for a subset of sampling occasions is needed. Although this method reduces time spent processing and validating audio recordings and seems ideal for PAM efforts that yield very few automated detections (e.g., when monitoring bats), it does not capitalize on the breadth of detail present in long survey periods that are often characteristic of avian PAM studies.

To help fill this need, Chambert et al. (2018) developed a false-positive occupancy model parameterization that makes full use of the validated and uncertain PAM detections in the form of counts to estimate occupancy probabilities. Similar analytical methods with the same data requirements also exist (Kéry and Royle 2021, Rhinehart et al. 2022). Others have developed models that borrow model structure components from N-mixture models to estimate relative abundances. These approaches have some of the same data requirements, and relative abundance is the relative number of species detections that are true-positive detections (Doser et al. 2021, Udell et al. 2024). These approaches represent major advancements in the field but rely on a confidence score thresholding approach and carry forward some of the same limitations mentioned above.

Example of a Workflow of the Simple Passive Acoustic Monitoring Protocol

The following example demonstrates how to use the software associated with the Simple Passive Acoustic Monitoring (SPAM) protocol (fig. 1). The SPAM protocol creates a structured, efficient process for gathering detection information from audio recordings, reviewing species detections, and generating occurrence data in a minimal number of steps.

This guide makes use of sample detections and annotated scripts to facilitate learning and application of the SPAM protocol. Scripts are designed and written to be implemented by users with some previous experience with the R programming language.

To use the provided scripts and datasets in this protocol, the machine being used will need the following open-source programs:

- Python version 3.9 or greater via the Miniconda installer

- Program R version 4.2 or higher and RStudio Version 2023.03.1+446 or higher with the following packages: dplyr (Wickham et al. 2025a), tidyr (Wickham et al. 2025b), lubridate (Spinu et al. 2023), stringr (Wickham 2023), tuneR (Ligges et al. 2024), ggplot2 (Wickham et al. 2024), seewave (Sueur et al. 2023)
- FFMpeg (Tomar 2006)

These programs are available online free of charge for download and, as of press time, are preapproved for use on Forest Service computers.

After ensuring these programs are installed, download the software folder accompanying this report (Vernasco and Duarte 2025). Figure 4 shows the structure of this folder. Before proceeding with the following example of the SPAM protocol, it may be helpful to review the workflow overview in figure 1 and the directory structure in figure 4. Each of the following sections detail how to use their associated scripts that are provided in the “scripts” folder (fig. 4), beginning with instructions on how to create the CSV file that contains the information about where and when species detections occurred.

```

SPAM_Software_Folder/
├── Bioacoustics_GTR_112224.Rproj          # RStudio project file used to run provided SPAM scripts
├── all_hard_drive_details.csv             # Paths and file names for audio files included in software folder
├── example_clip_processing_file.csv       # Example file for entering review information about species detections
├── example_detections_to_be_created.csv   # Example file detailing detections to be extracted from raw audio files
├── birdnet_files/                        # BirdNET detection files organized by hard drive
│   ├── hard_drive_1/
│   ├── hard_drive_2/
│   └── hard_drive_3/
├── raw_audio_files/                      # Audio recordings associated with the provided BirdNET detections
│   ├── hard_drive_1/
│   ├── hard_drive_2/
│   └── hard_drive_3/
├── detections/                          # Folder for storing extracted audio files containing species detections
├── mp4s/                                # Folder for storing MP4 video clips of species detections
├── spectrograms/                        # Folder for storing spectrogram images of species detections
├── scripts/                             # Directory containing scripts for implementing SPAM protocol
│   ├── clip_extraction.py                # Python script for extracting audio clips from audio recordings
│   ├── clip_processing_file_creation.R    # R script to create a clip processing file
│   ├── creating_clip_extraction_file.R    # R script to create a clip extraction file
│   ├── spam_environment.yml              # Conda environment setup file for the SPAM protocol
│   ├── spectrogram_script.R              # R script to generate spectrograms of species detections
│   └── wav_spectrogram_combine.py        # Python script to combine audio clips and spectrograms into .MP4s

```

Figure 4—Structure and descriptions for the software folder provided with the Simple Passive Acoustic Monitoring (SPAM) protocol (Vernasco and Duarte 2025). The contents of this directory can be used to implement the SPAM protocol.

Because we are using BirdNET and our case study focuses on forest-associated avian species, our target sound classes all represent potential species-level detections (e.g., rather than call types). For clarity, we refer to target sound classes in this document as “species detections” from here forward, but we recognize that not all species detections are true-positive detections and that target sound classes do not necessarily have to be produced by wildlife (e.g., gunshots). Moreover, acoustic classifiers such as BirdNET and PNW-Cnet calculate logit-transformed confidence scores that can range from 0 through 1 (albeit BirdNET omits any detections with a confidence score of <0.1), whereas acoustic classifiers, such as Perch, calculate confidence scores that can range from $-\infty$ to $+\infty$. If desired, Perch confidence scores can be rescaled to 0–1 using a logit transformation. Of course, the opposite is also true. Another consideration is that the detection window duration of acoustic classifiers

can differ. For example, BirdNET uses 3-second detection windows and PNW-Cnet uses 12-second detection windows. Furthermore, in the following example, raw audio recordings are stored in separate folders as would be the case when stored on external hard drives. Differences in the duration of the detection window or storage location of the raw audio files (if applicable) are readily accommodated with minor adjustments to the code that we provide.

To begin working through the provided sample code of the SPAM pipeline, open the .Rproj file in the accompanying software folder. Opening the .Rproj file starts RStudio and sets the working directory as the folder containing the .Rproj file. This process facilitates the use of relative file paths (as opposed to complete/absolute file paths) and ensures that the scripts within the “scripts” folder function without having to change file paths. File paths only need to be changed when specified.

Creating Clip Extraction File

The following code demonstrates the use of the “creating_clip_extraction_file.R” script with a random selection of 100 BirdNET v2.4 output files in the form of CSV files. The product of this script is the clip extraction file, which contains the information needed to extract, label, and organize short audio clips containing automated detections. The extent to which the “creating_clip_extraction_file.R” script can be used for a particular project depends on the acoustic classifier used and the format of the output files from the classifier. For projects using a different classifier with a different output file format than that of BirdNET v2.4, the information and formatting requirements of the CSV file used to extract clips in the next step are provided in appendix 1. Once the clips are extracted, the remainder of the SPAM protocol can be used to process automated species detection data, regardless of the acoustic classifier used.

In this example, the BirdNET output files are saved in the “birdnet_files” folder within the provided software folder. The “birdnet_files” folder has a hierarchical structure that begins with the hard drive on which the original audio files are stored. Within each hard drive folder (e.g., “hard_drive_1”), there are folders representing the sampling structure which, in our project in the northern Blue Mountains, are ARUs deployed at four survey stations within a respective hexagon selected from a sampling grid. The default settings of BirdNET produce the “data” folder within each station folder. Each “data” folder contains one or more CSV files, each of which contains the species detections for a given recording and information about when the detection occurred within the recording. Note that filenames of each audio file contain the date and time of the recording. The ARU that collected the audio recordings determines the time and date. More specifically, in this example, the names of the CSV files are structured as: “name_date_time.BirdNET.results.csv,” wherein “name” refers to the prefix specific to the ARU that created the audio recording. The dates and times in the filenames are formatted as YYYYMMDD (e.g., 20220531 = May 31, 2022) and hhmmss (e.g., 150000 = 15:00:00), respectively. By default, BirdNET replaces the audio file’s original .wav file extension with “.BirdNET.results.csv” when output files containing species detections are made.

To begin the process of creating the clip creation file, it is necessary to prepare the R environment by loading the following packages.

```
library(dplyr)
library(tidyr)
library(lubridate)
library(stringr)
```

The next step is to specify the directories containing important files, including the BirdNET output files and the hard drive directories containing the original audio recordings. The first two directories listed below are relative file paths that begin with the directories in the software folder. The third file path can be replaced with a file path to a directory on an external hard drive, which contains all the audio recordings generated at each sampling location. In this example, hypothetical hard drives in the provided software folder are specified. As a reminder, for this script to run, it is necessary to open R using the .Rproj file in the software directory.

```
birdnet_files <- "./birdnet_files"
base_directory <- getwd()
hd1 <- "./raw_audio_files/2022_hard_drive_1" # path to hard drive 1
folder
hd2 <- "./raw_audio_files/2022_hard_drive_2" # path to hard drive 2
folder
hd3 <- "./raw_audio_files/2022_hard_drive_3" # path to hard drive 3
folder
```

Now that the paths to the file and hard drive directories are specified, all the BirdNET output files (i.e., CSV files) can be combined into a single data frame (df), in which each row of the data represents a species detection, and the columns are the subdirectories in the file paths, with the final column being the file paths of the CSV file containing each detection.

```
# Make an object that contains all CSV files in the directory contain-
ing BirdNET output files.
```

```
csv_files <- list.files(path = birdnet_files, pattern = "\\\\.csv$",
recursive = TRUE, full.names = TRUE)
```

```
# Initialize an empty data frame with the same column names as the
BirdNET CSV output files. The following column names specified (e.g.,
"Start..s.") need to match those specified by the structure of the
files containing the species detection data.
```

```
data_frames <- data.frame(Start..s. = numeric(), End..s. = numeric(),
Scientific.name = character(), Common.name = character(), Confidence =
numeric(), filepath = character(), stringsAsFactors = FALSE)
```

```
# Loop through each CSV file to combine them into one data frame.
Note that files with zero detections are skipped, and the name of any
skipped files is printed during processing.
```

```
for (file in csv_files) {
  # Read the data from the current file.
```

```

current_data <- read.csv(file, header = TRUE)

# Check if the current data frame has zero rows.
if (nrow(current_data) > 0) {
  # Add a new column with the file path.
  current_data$filepath <- file

  # Append the data frame to the list.
  data_frames <- rbind(data_frames, current_data)
} else {
  # Print a message indicating that the file is skipped.
  cat("Skipped file:", file, "because it has zero rows.\n")
}
}

## Skipped file: ./birdnet_files/2022_hard_drive_1/47551/47551_E/Data/
BM_20220706_010000.BirdNET.results.csv because it has zero rows.

## Skipped file: ./birdnet_files/2022_hard_drive_1/47900/47900_B/Data/
BM_20220518_010000.BirdNET.results.csv because it has zero rows.

## Skipped file: ./birdnet_files/2022_hard_drive_2/48281/48281_D/Data2/
SMU04322_20220621_230000.BirdNET.results.csv because it has zero rows.

## Skipped file: ./birdnet_files/2022_hard_drive_2/48504/48504_F/Data/
BM_20220605_170000.BirdNET.results.csv because it has zero rows.

## Skipped file: ./birdnet_files/2022_hard_drive_2/48504/48504_F/Data/
BM_20220613_223902.BirdNET.results.csv because it has zero rows.

## Skipped file: ./birdnet_files/2022_hard_drive_3/48199/48199_C/Data/
COA_20220721_180000.BirdNET.results.csv because it has zero rows.

# Save data_frames as a new object to preserve the original data
frame for future reference as needed.

df <- data_frames

```

At this point, the structure of data frame can be inspected.

```

head(df)
##   Start..s. End..s.      Scientific.name      Common.name
Confidence
## 1         6      9 Psilosops flammeolus Flammulated Owl
0.1628
## 2        36     39 Psilosops flammeolus Flammulated Owl

```

```

0.4917
## 3      63      66 Psilosops flammeolus Flammulated Owl
0.3011
## 4      81      84 Psilosops flammeolus Flammulated Owl
0.2738
## 5      87      90 Psilosops flammeolus Flammulated Owl
0.4954
## 6     114     117 Psilosops flammeolus Flammulated Owl
0.2671
##
filepath
## 1 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
## 2 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
## 3 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
## 4 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
## 5 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
## 6 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv

```

The information contained in data frame can now be formatted to further refine and structure the information needed for each species detection.

```
# Make a new column containing the file path so you can keep the original file path in your data frame for future reference.
```

```
df$path <- df$filepath
```

```
# Separate the duplicated file path column into new columns. Note that the "into" argument within the "separate()" function specifies the new names of the columns that are created. These column names can be updated as necessary.
```

```

df <- df %>%
  mutate(path = str_split(path, "/")) %>% # Split path by "/".
  mutate(

path = sapply(

  path, function(x) paste(x[-(1:2)], collapse = "/")

)

```

```
) %>%

separate(

  path,

  into = c("hd", "hexagon", "station", "data", "fn"),

  sep = "/", fill = "right", extra = "merge"

)

# Create a column containing the name of the WAV file by replacing the
file extension added by BirdNET with the ".wav" file extension.

df$wav_fn <- gsub("BirdNET.results.csv","wav", df$fn)

# Create a column containing the name of the WAV file without the
".wav" file extension.

df$wav_fn1 <- gsub(".BirdNET.results.csv","", df$fn)

# To extract the time and date information from the recording file-
name, while retaining the filename, create a new column containing the
WAV filename to split it into new columns.

df$fn1<-df$wav_fn1

# Separate the filename into its components.

df <- df %>% separate(fn1, c("name","date", "time"), sep = "_")

# Format the date of the recording into a YYYYMMDD format.

df$date <- ymd(df$date)

# Format the time of the recording into a hhmmss format.

df$time <-hms(

  paste(

    substring(gsub(".wav","",df$time),0,2),
```

```

    substring(gsub(".wav","",df$time),3,4),

    substring(gsub(".wav","",df$time),5,6),

    sep = ":"
  )
)

# Create columns needed for clip generation. Note that some columns
are redundant.

# Create an onset column, which represents the number of seconds from
the start of the recording until when the detection appears.

df$onset <- df$Start..s.

# Create a new site column.

df$site <- df$station

# Create a column that describes the minutes and seconds from the
start of the recording to when the detection occurs.

df$mm_ss <- sprintf("%02d %02d", df$onset %/% 60, df$onset %% 60)

# Give each row a random ID for future reference. Note that no two
rows have the same ID.

df$ID <- sample(1:nrow(df),size = nrow(df), replace = FALSE)

# Create a new column containing the species name.

df$species <- df$Common.name

# Create a column that ranks rows by confidence score for each
station and for each species. Add additional groups in the order you
want calls ranked in the group_by() section.

```

```
df <- df %>% group_by(station, species) %>% arrange(desc(Confidence))
%>% mutate(rank = row_number()) %>% ungroup()
```

```
# Inspect df in its new form.
```

```
head(df)
```

```
##   Start...s. End...s.      Scientific.name      Common.name
Confidence
```

```
## 1         6         9 Psilosops flammeolus Flammulated Owl
0.1628
```

```
## 2        36        39 Psilosops flammeolus Flammulated Owl
0.4917
```

```
## 3        63        66 Psilosops flammeolus Flammulated Owl
0.3011
```

```
## 4        81        84 Psilosops flammeolus Flammulated Owl
0.2738
```

```
## 5        87        90 Psilosops flammeolus Flammulated Owl
0.4954
```

```
## 6       114       117 Psilosops flammeolus Flammulated Owl
0.2671
```

```
##
```

```
filepath
```

```
## 1 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
```

```
## 2 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
```

```
## 3 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
```

```
## 4 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
```

```
## 5 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
```

```
## 6 ./birdnet_files/2022_hard_drive_1/47451/47451_D/Data/
BM_20220629_010000.BirdNET.results.csv
```

```
##          hd hexagon station data
fn
```

```
## 1 2022_hard_drive_1  47451 47451_D Data BM_20220629_010000.BirdNET.
results.csv
```

```
## 2 2022_hard_drive_1  47451 47451_D Data BM_20220629_010000.BirdNET.
results.csv
```

```
## 3 2022_hard_drive_1  47451 47451_D Data BM_20220629_010000.BirdNET.
results.csv
```

```
## 4 2022_hard_drive_1  47451 47451_D Data BM_20220629_010000.BirdNET.
results.csv
```

```
## 5 2022_hard_drive_1  47451 47451_D Data BM_20220629_010000.BirdNET.
results.csv
```

```
## 6 2022_hard_drive_1  47451 47451_D Data BM_20220629_010000.BirdNET.
results.csv
```

```
##          wav_fn          wav_fn1 name      date
time Onset
```

```
## 1 BM_20220629_010000.wav BM_20220629_010000  BM 2022-06-29 1H 0M
OS    6
```

```
## 2 BM_20220629_010000.wav BM_20220629_010000  BM 2022-06-29 1H 0M
OS   36
```

```
## 3 BM_20220629_010000.wav BM_20220629_010000  BM 2022-06-29 1H 0M
OS   63
```

```
## 4 BM_20220629_010000.wav BM_20220629_010000  BM 2022-06-29 1H 0M
OS   81
```

```
## 5 BM_20220629_010000.wav BM_20220629_010000  BM 2022-06-29 1H 0M
OS   87
```

```
## 6 BM_20220629_010000.wav BM_20220629_010000  BM 2022-06-29 1H 0M
OS  114
```

```
##      site mm_ss    ID      species
```

```
## 1 47451_D 00 06 4639 Flammulated Owl
## 2 47451_D 00 36 6341 Flammulated Owl
## 3 47451_D 01 03 7113 Flammulated Owl
## 4 47451_D 01 21 12937 Flammulated Owl
## 5 47451_D 01 27 9515 Flammulated Owl
## 6 47451_D 01 54 7949 Flammulated Owl
```

At this point, the file paths can be formatted to the original audio recordings. In this case, the original audio files are present in the accompanying software folder. Note that this process requires a CSV file containing a column denoting the location identifier for deployment locations (stations in this case), the filename of the original audio recordings made by the ARU, and the relative file paths to the audio recordings. The relative file paths should begin with folders stored within the folders specified by “hd1,” “hd2,” etc.

```
# Load the file containing information about the original audio files
and paths to recordings.
```

```
hd <- read.csv(paste(base_directory,"all_hard_drive_details.csv", sep =
"/"))
```

```
# Add the WAV file path to df for clip creation using the station ID
(i.e., the location identifier) and the WAV filename as matching condi-
tions. Note that station IDs must be unique for the filenames to be
specified correctly.
```

```
df$path <- hd[match(paste(df$station,df$wav_fn),paste(hd$station,hd$f-
n)), "path"]
```

```
# Add a new column denoting the full path to the hard drive or folder
containing original WAV files, based on the value in the hard drive
column (i.e., hd1 corresponds to the first hard drive, hd2 the second,
hd3 the third, etc.). Remove or add a "#" from each row depending on
the number of hard drives or folders specified. In the following code,
hd2 and hd3 are skipped because of the presence of the #.
```

```
df <- df %>%
  mutate(hd_path = case_when(
    hd == "hard_drive_1" ~ hd1,
    #hd == "hard_drive_2" ~ hd2,
    #hd == "hard_drive_3" ~ hd3,
    TRUE ~ NA_character_ # Default case, if none of the above match.
  ))
```

```

# Paste together the two file paths to make a complete file path to
the WAV file.

df$full_path <- paste(df$hd_path, substring(df$path, 2, ), sep = "")

# Check if the file exists to make sure your paths are correct; all
values should be TRUE.

df$path_check <- file.exists(df$full_path)

# Count the number of correct (TRUE) and incorrect (FALSE) file paths.

df %>% group_by(path_check) %>% count()

## # A tibble: 1 × 2
## # Groups:   path_check [1]
##   path_check      n
##   <lgl>        <int>
## 1 TRUE        17137

# In this example, all files specified exist because only TRUE is
represented.

# Write out the CSV file for subsequent use.

write.csv(df, "example_clip_creation_file.csv", row.names = FALSE)

```

The clip creation file is now formatted properly and saved as “example_clip_creation_file.csv” in the working directory. Figure 5 is an example of “example_clip_creation_file.csv” output.

Start...s.	End...s.	Scientific name	Common name	Confidence	filepath	hd	hexagon	station	data	in	wav in	wav fn1	name	date	time	Onset	site	mm_ss	ID	species	rank	path	hd_path	full_path	path_check
576	579	<i>Loxia curvirostra</i>	Red Crossbill	0.6002	7501_F/Data/2022_hard_drive.1	47501	47501_F	Data	1000000_Bird	20623	1000000_Bird	20623_10	BM	6/23/22	10H0M03	576	47501_F	0936	15604	Red Crossbill	1	ata/BM_2022	One Touch/27501/47501	TRUE	
3492	3495	<i>Loxia curvirostra</i>	Red Crossbill	0.5234	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	3492	47551_A	5812	13786	Red Crossbill	1	directory_1/07	One Touch/27551_A/Data/	TRUE	
579	582	<i>Loxia curvirostra</i>	Red Crossbill	0.5114	7501_F/Data/2022_hard_drive.1	47501	47501_F	Data	1000000_Bird	20623	1000000_Bird	20623_10	BM	6/23/22	10H0M03	579	47501_F	0939	1407	Red Crossbill	2	ata/BM_2022	One Touch/27501/47501	TRUE	
2130	2133	<i>Loxia curvirostra</i>	Red Crossbill	0.4599	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	2130	47551_A	3530	1345	Red Crossbill	2	directory_1/07	One Touch/27551_A/Data/	TRUE	
585	588	<i>Loxia curvirostra</i>	Red Crossbill	0.4029	7501_F/Data/2022_hard_drive.1	47501	47501_F	Data	1000000_Bird	20623	1000000_Bird	20623_10	BM	6/23/22	10H0M03	585	47501_F	0945	642	Red Crossbill	3	ata/BM_2022	One Touch/27501/47501	TRUE	
435	438	<i>Loxia curvirostra</i>	Red Crossbill	0.393	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	435	47676_G	0715	12657	Red Crossbill	3	directory_3/07	One Touch/27676_G/Data/	TRUE	
3486	3489	<i>Loxia curvirostra</i>	Red Crossbill	0.3459	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	3486	47551_A	5806	9940	Red Crossbill	3	directory_1/07	One Touch/27551_A/Data/	TRUE	
588	591	<i>Loxia curvirostra</i>	Red Crossbill	0.3118	7501_F/Data/2022_hard_drive.1	47501	47501_F	Data	1000000_Bird	20623	1000000_Bird	20623_10	BM	6/23/22	10H0M03	588	47501_F	0948	10400	Red Crossbill	4	ata/BM_2022	One Touch/27501/47501	TRUE	
3519	3522	<i>Loxia curvirostra</i>	Red Crossbill	0.2837	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	3519	47551_A	5839	13948	Red Crossbill	4	directory_1/07	One Touch/27551_A/Data/	TRUE	
2109	2112	<i>Loxia curvirostra</i>	Red Crossbill	0.2727	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	2109	47551_A	3509	903	Red Crossbill	5	directory_1/07	One Touch/27551_A/Data/	TRUE	
534	537	<i>Loxia curvirostra</i>	Red Crossbill	0.2651	7548_A/Data/2022_hard_drive.1	47548	47548_A	Data	080000_Bird	20702	080000_Bird	20702_08	BM	7/12/22	8H0M0S	534	47548_A	0854	10750	Red Crossbill	1	ata/BM_2022	One Touch/27548/47548	TRUE	
2148	2151	<i>Loxia curvirostra</i>	Red Crossbill	0.25	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	060002_Bird	20624	060002_Bird	20624_06	BM	6/24/22	6H0M2S	2148	47787_A	3548	8085	Red Crossbill	1	directory_5/07	One Touch/2787_A/Data/	TRUE	
531	534	<i>Loxia curvirostra</i>	Red Crossbill	0.231	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	060002_Bird	20624	060002_Bird	20624_06	BM	6/24/22	6H0M2S	531	47787_A	0851	14790	Red Crossbill	2	directory_5/07	One Touch/2787_A/Data/	TRUE	
562	565	<i>Loxia curvirostra</i>	Red Crossbill	0.2201	7501_F/Data/2022_hard_drive.1	47501	47501_F	Data	1000000_Bird	20623	1000000_Bird	20623_10	BM	6/23/22	10H0M03	562	47501_F	0942	1717	Red Crossbill	5	ata/BM_2022	One Touch/27501/47501	TRUE	
375	378	<i>Loxia curvirostra</i>	Red Crossbill	0.2155	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	375	47676_G	0615	87	Red Crossbill	2	directory_3/07	One Touch/27676_G/Data/	TRUE	
3489	3492	<i>Loxia curvirostra</i>	Red Crossbill	0.2148	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	3489	47551_A	5809	4568	Red Crossbill	6	directory_1/07	One Touch/27551_A/Data/	TRUE	
387	390	<i>Loxia curvirostra</i>	Red Crossbill	0.2136	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	387	47676_G	0627	11683	Red Crossbill	3	directory_3/07	One Touch/27676_G/Data/	TRUE	
378	381	<i>Loxia curvirostra</i>	Red Crossbill	0.2038	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	080000_Bird	20703	080000_Bird	20703_08	BM	7/12/22	8H0M0S	378	47787_A	0618	8495	Red Crossbill	3	directory_7/07	One Touch/2787_A/Data/	TRUE	
2127	2130	<i>Loxia curvirostra</i>	Red Crossbill	0.2036	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	2127	47551_A	3527	11093	Red Crossbill	7	directory_1/07	One Touch/27551_A/Data/	TRUE	
2139	2142	<i>Loxia curvirostra</i>	Red Crossbill	0.2032	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	060002_Bird	20624	060002_Bird	20624_06	BM	6/24/22	6H0M2S	2139	47787_A	3536	11525	Red Crossbill	4	directory_5/07	One Touch/2787_A/Data/	TRUE	
432	435	<i>Loxia curvirostra</i>	Red Crossbill	0.1873	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	432	47676_G	0712	2327	Red Crossbill	4	directory_3/07	One Touch/27676_G/Data/	TRUE	
177	180	<i>Loxia curvirostra</i>	Red Crossbill	0.1809	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	177	47551_A	0257	2339	Red Crossbill	8	directory_1/07	One Touch/27551_A/Data/	TRUE	
369	372	<i>Loxia curvirostra</i>	Red Crossbill	0.1774	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	080000_Bird	20703	080000_Bird	20703_08	BM	7/12/22	8H0M0S	369	47787_A	0609	14235	Red Crossbill	5	directory_7/07	One Touch/2787_A/Data/	TRUE	
120	123	<i>Loxia curvirostra</i>	Red Crossbill	0.177	7637_F/Data/2022_hard_drive.1	47637	47637_F	Data	1200000_Bird	20603	1200000_Bird	20603_12	BM	6/3/22	12H0M0S	120	47637_F	0200	2524	Red Crossbill	1	directory_1/07	One Touch/27637_F/Data/	TRUE	
447	450	<i>Loxia curvirostra</i>	Red Crossbill	0.1744	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	080000_Bird	20703	080000_Bird	20703_08	BM	7/12/22	8H0M0S	447	47787_A	0727	8259	Red Crossbill	6	directory_7/07	One Touch/2787_A/Data/	TRUE	
429	432	<i>Loxia curvirostra</i>	Red Crossbill	0.1668	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	429	47676_G	0709	9718	Red Crossbill	5	directory_3/07	One Touch/27676_G/Data/	TRUE	
2154	2157	<i>Loxia curvirostra</i>	Red Crossbill	0.1658	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	060002_Bird	20624	060002_Bird	20624_06	BM	6/24/22	6H0M2S	2154	47787_A	3554	12042	Red Crossbill	7	directory_5/07	One Touch/2787_A/Data/	TRUE	
396	399	<i>Loxia curvirostra</i>	Red Crossbill	0.1577	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	080000_Bird	20703	080000_Bird	20703_08	BM	7/12/22	8H0M0S	396	47787_A	0636	14729	Red Crossbill	8	directory_1/07	One Touch/2787_A/Data/	TRUE	
378	381	<i>Loxia curvirostra</i>	Red Crossbill	0.1507	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	378	47676_G	0618	5938	Red Crossbill	6	directory_3/07	One Touch/27676_G/Data/	TRUE	
2151	2154	<i>Loxia curvirostra</i>	Red Crossbill	0.1355	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	060002_Bird	20624	060002_Bird	20624_06	BM	6/24/22	6H0M2S	2151	47787_A	3551	7196	Red Crossbill	9	directory_5/07	One Touch/2787_A/Data/	TRUE	
531	534	<i>Loxia curvirostra</i>	Red Crossbill	0.1338	7548_A/Data/2022_hard_drive.1	47548	47548_A	Data	080000_Bird	20702	080000_Bird	20702_08	BM	7/12/22	8H0M0S	531	47548_A	0851	7154	Red Crossbill	2	ata/BM_2022	One Touch/27548/47548	TRUE	
426	429	<i>Loxia curvirostra</i>	Red Crossbill	0.1171	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	426	47676_G	0706	9971	Red Crossbill	7	directory_3/07	One Touch/27676_G/Data/	TRUE	
384	387	<i>Loxia curvirostra</i>	Red Crossbill	0.1153	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	384	47676_G	0624	14533	Red Crossbill	8	directory_3/07	One Touch/27676_G/Data/	TRUE	
438	441	<i>Loxia curvirostra</i>	Red Crossbill	0.1142	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	438	47676_G	0718	14117	Red Crossbill	9	directory_3/07	One Touch/27676_G/Data/	TRUE	
2157	2160	<i>Loxia curvirostra</i>	Red Crossbill	0.1123	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	060002_Bird	20624	060002_Bird	20624_06	BM	6/24/22	6H0M0S	2157	47787_A	3557	4233	Red Crossbill	10	directory_5/07	One Touch/2787_A/Data/	TRUE	
471	474	<i>Loxia curvirostra</i>	Red Crossbill	0.1084	787_A/Data/2022_hard_drive.1	47787	47787_A	Data	080000_Bird	20703	080000_Bird	20703_08	BM	7/12/22	8H0M0S	471	47787_A	0751	15329	Red Crossbill	11	directory_7/07	One Touch/2787_A/Data/	TRUE	
3504	3507	<i>Loxia curvirostra</i>	Red Crossbill	0.1052	7551_A/Data/2022_hard_drive.1	47551	47551_A	Data	061102_Bird	20619	061102_Bird	20619_06	BM	6/19/22	6H11M2S	3504	47551_A	5824	2440	Red Crossbill	9	directory_1/07	One Touch/27551_A/Data/	TRUE	
465	468	<i>Loxia curvirostra</i>	Red Crossbill	0.1048	786_D/Data/2022_hard_drive.1	47786	47786_D	Data	1200000_Bird	20604	1200000_Bird	20604_12	BM	6/4/22	12H0M0S	465	47786_D	0745	3688	Red Crossbill	1	directory_3/07	One Touch/2786_D/Data/	TRUE	
381	384	<i>Loxia curvirostra</i>	Red Crossbill	0.1043	7676_G/Data/2022_hard_drive.1	47676	47676_G	Data	070002_Bird	20606	070002_Bird	20606_07	HIA	6/6/22	7H0M2S	381	47676_G	0621	1020	Red Crossbill	10	directory_3/07	One Touch/27676_G/Data/	TRUE	

Figure 5 — Example of a clip creation file used for extracting audio clips containing automated species detections from raw audio recordings collected by autonomous recording units. Columns contain identifying information about each automated species detection, including the time and location of the detection, the name and location of audio recordings, and the common and scientific name of the detected species. The ID column represents a random identification number assigned to each row, which can be used to locate a specific detection and easily confirm that information is entered into the proper row of the data entry sheet. When placed at the beginning of filenames, the ID column can also be used to randomly organize detections before review.

Creating WAV Files of Species Detections

The clip creation file made using Program R contains the relevant information about each detection, including the species, time and location of the detection, the location within the audio file of the detection (i.e., seconds since the start of the recording), and the path to the audio file.

After it is created, the clip creation file and Python can be used to extract short audio files containing each species detection. Shorter audio files of each detection speed up the review process; users don't need to navigate a lengthy audio recording to find detections within the audio recording. Reducing the number of steps to review each detection is the key to efficiency when there is a need to review thousands of species detections.

To begin, open the appropriate command line interface for the machine. If using Microsoft Windows, this would be the command prompt—hold the Windows key and press “R”, and then type “cmd”. For an Apple Mac machine, hold the Command key and press the space bar, then type “Terminal” and press Enter. Once the command line interface is open, set the working directory to the software folder by entering the following code into the command prompt. Note that “file/path/to/scripts/folder” needs to be replaced with the path to the “scripts” folder in the provided software folder.

```
cd /file/path/to/scripts/folder
```

Assuming Python is installed via Miniconda, the provided environment can now be loaded, which installs the various dependencies associated with the SPAM protocol.

```
conda env create -f spam_environment.yml
```

Once the environment has been created, activate the environment within the terminal. Note that “spam_env” is the name of the environment.

```
conda activate spam_env
```

Now that the environment is activated, the clip extraction script can be run.

```
python "clip_extraction.py"
```

After running this code, the user will be prompted to provide the path to a CSV file. In this case, provide the full path to the file created in the previous step (e.g., example_clip_creation_file.csv).

Enter the path to the .csv file:

The full path to the appropriate CSV file is entered. Here is an example file path:

```
Enter the path to the .csv file: /path/to/example_detections_to_be_created.csv
```

The terminal then prompts the user to provide the path to the output directory (i.e., the folder where clips are saved upon creation). At this point, the file path can be copied and pasted to the output directory below.

```
Enter the output directory path:  
/path/to/GTR_software_folder/clips
```

The next prompt asks for the duration of the detection window, which varies based on the acoustic classifier used to generate species detections. As a reminder, BirdNET v2.4 uses 3-second detection windows in which case users enter the number 3. PNW-Cnet uses a 12-second detection window, in which case users enter the number 12. Note that, by default, 0.5 seconds are added to start and end times of the detection window. So, specifying a 3-second clip duration results in 4-second clips and a 12-second clip duration results in 13-second clips.

```
Enter the detection window length in seconds (e.g., 3): 3
```

Next, the terminal asks for a list of species for which longer clips are needed. Longer clips have a 3-second buffer before and after the actual detection window. This option is included because detections of some species require, or are much easier validated, using sounds before or after the detection window. In our experience, such species include owls, doves, and forest-associated raptors. Note that the names that are specified need to exactly match the corresponding names in the “species” column of the clip creation file, and each name is followed by a comma and a space (e.g., American goshawk, barred owl). If there are many species to enter, it can be more efficient to type the list in a word processor, carefully check for typos and correct formatting, and copy and paste the list into the terminal. If longer clips are not needed for any species, simply press the Enter key.

```
Enter a comma-separated list of species for longer duration clips  
(or 'all species' for all species, or press Enter for none): American  
Goshawk, Barred Owl, Mourning Dove
```

Users are now prompted to provide the names of columns in the order that they want them to appear in filenames of the audio clips created. Filenames provide a simple way to organize detections in a particular order, including a random order. Filenames also provide a reliable and easily accessible location for important identification information about a detection. We therefore recommend including in the filename the location identifier, the date, the hour of the day, and the seconds from the start of the clip to when the detection occurred. At the very least, filenames should contain the unique identifier for each detection, as specified in the ID column of the clip creation file.

Filenames can also be used to review clips in a random order by setting the first component of the filename to the random number assigned to each detection (i.e., the ID column). Alternatively, to order clips by the confidence score provided by the classifier, set the first component of the filename to the rank of the clip (see “Creating Clip Extraction File” for details on how to create the rank column).

In the following example, clip names contain (in the corresponding order) the randomly generated ID, the station at which the detection occurred, the filename of the original recording without the .wav extension (which contains date and time of day information), and the number of seconds from the start of the recording to the detection. Note that each of these components is separated by an underscore (_) in the resulting filenames.

Enter the column names in the order they should appear in the output filenames (e.g., 'ID,site,fn,mm_ss'): ID,site,wav_fn1,mm_ss

Next, the script asks for the output path structure, referring to the structure of the folders containing the clips to be created. The structure used depends on the objectives of the projects, and to some extent, user preference. To create encounter histories for single species occupancy modeling, for example, clips should be grouped by the location and then the primary survey periods (e.g., station, week). If multiple species are included in the clip creation file, the species name should be included in the output structure. Using both the folder structure and filenames can further facilitate efficient review of detections based on project objectives by sorting detections and automating the process of choosing which detections to review. In the following example, we make folders grouped by species and then station.

Enter the desired output path structure (use {column names} separated by'/'): {species}/{site}

The script begins to create clips and provides the path to each clip as it is created (i.e., “Output clip path”). Here is an example output from the code provided above.

Clip created: /Users/benvernasco/Documents/GTR_software_folder/clips/
Red Crossbill/47501_F/15804_47501_F_BM_20220623_100000_09 36.wav

Clip created: /Users/benvernasco/Documents/GTR_software_folder/clips/
Red Crossbill/47551_A/13786_47551_A_BM_20220619_061102_58 12.wav

Clip created: /Users/benvernasco/Documents/GTR_software_folder/clips/
Red Crossbill/47501_F/1407_47501_F_BM_20220623_100000_09 39.wav

After the script has completed processing, inspect the specified output directory for the clips. In this example, the audio clips are stored in species-specific folders and sorted by the random ID, which facilitates reviewing clips for each species in random order.

An additional file will be saved called “skipped_clips.csv” within the output directory. This file contains the rows from the CSV input file of clips that could not be created because of issues with the information provided in the CSV file (e.g., `df$path_check = FALSE`). Inspect the “skipped_clips.csv” file to examine clips that were not made.

Creating Spectrograms of Species Detections

After creating the audio clips containing species detections, graphs displaying visual representations of the sound contained in each clip (i.e., spectrograms) can be created for manual review. In these graphs, the *y*-axis represents the frequency of the sound, and the *x*-axis represents time in seconds. The shading of each pixel in the graph represents the amplitude of sound at that frequency and time within the clip. Darker pixels in the spectrogram represent higher amplitude or louder sounds (fig. 6).

The `spectrogram_script.R` searches the indicated directory for WAV files and makes PNG files of each WAV file identified. The PNG files are stored using the same directory structure, preserving the information contained within folder names, such as the station identifier or species name.

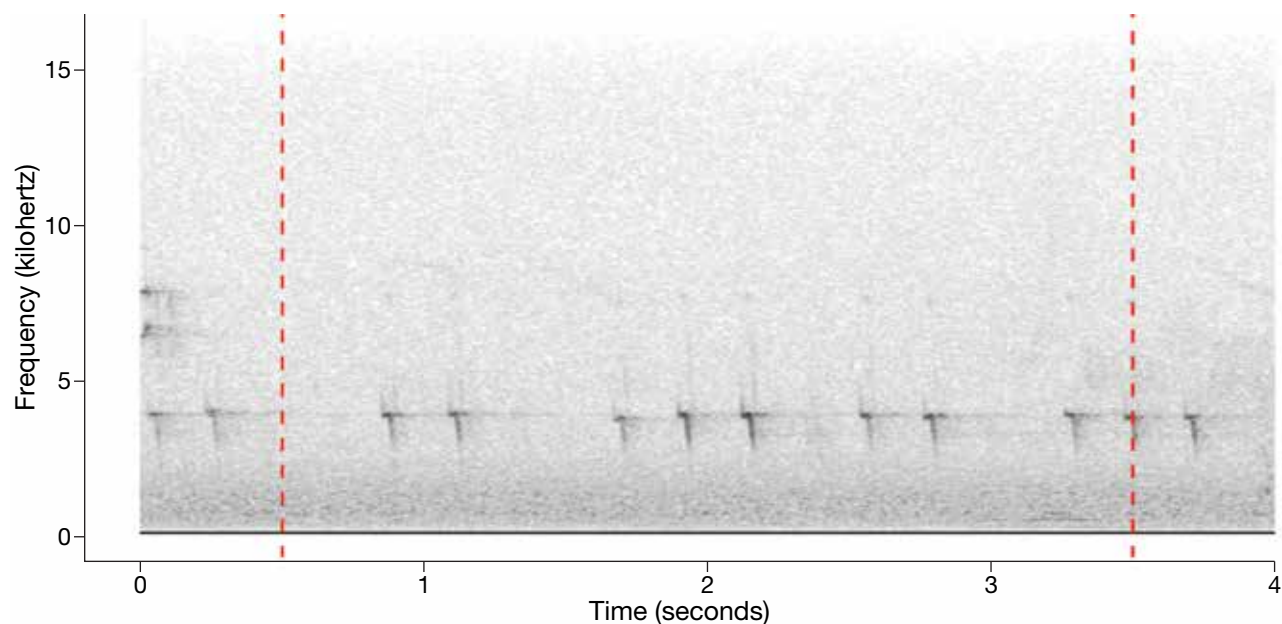


Figure 6—Example of a spectrogram produced by the Simple Passive Acoustic Monitoring protocol. Shading denotes the amplitude of acoustic signals with darker areas having greater amplitude. Red, dashed vertical lines are positioned 0.5 seconds before the start and after the end of the detection window. The space between the dashed lines represents the 3-second BirdNET detection window. This example shows a true-positive detection of a red crossbill. Notably, for some sound classes, confirmation of presence or absence of the sound of interest can be determined by visual assessment alone. Rapid assessment of the visual and auditory information contained in each species detection is facilitated by combining spectrograms with the original audio into an MP4 file.

```
##Loading packages
library(tuneR)
library(ggplot2)
library(dplyr)
library(tidyr)
library(seewave)
```

The next step is to assign an object called “clip_directory” as the directory containing the audio clips of species detections. Assign another object called “base_directory” as the directory from which you are working, which contains the “clip_directory” folder. All WAV files in the “clip_directory” are made into spectrograms and saved in a folder in the “base_directory” folder. In this section of code, the duration of the detection window is specified. As previously mentioned, the duration of the detection window can vary based on the classifier used (e.g., 3 seconds for BirdNET v2.4, 12 seconds for PNW-Cnet v4).

```
clip_directory <- "./clips"
base_directory <- getwd()
window_duration <- 3 # Set the duration window, in number of seconds,
for your classifier.
```

The generate_spectrograms function can now be loaded. This function identifies the file paths associated with all WAV files in “clip_directory” and creates a new folder called “spectrograms” in the “base_directory” folder. The folder structure of all the WAV files is then re-created in the “spectrograms” folder. Each audio file is then read and converted into a spectrogram, which is saved in its corresponding directory using the same filename as the WAV file from which it was created.

```

generate_spectrogram <- function(file_path) {
  # Derive the relative path by removing clip_directory from file_
  path.
  relative_file_path <- gsub(paste0("^", gsub("\\\\", "/", normalize-
  Path(clip_directory)), "/?"), "", gsub("\\\\", "/", normalizePath(file_
  path)))

  # Create the full path for saving the spectrogram.
  spectrogram_path <- file.path(base_directory, "spectrograms", rela-
  tive_file_path)

  # Ensure the subdirectories exist.
  sub_dir <- dirname(spectrogram_path)
  if (!dir.exists(sub_dir)) {
    dir.create(sub_dir, recursive = TRUE)
  }

  # Read the current WAV file.
  audio <- readWave(file_path)

  # Calculate the duration of the audio clip in seconds.
  duration <- length(audio@left) / audio@samp.rate

  # Set the positions for the red vertical lines based on the center
  of the classifier detection window.
  center_time <- duration / 2

  red_line_positions <- c(center_time - (window_duration/2), center_
  time + (window_duration/2))

  # Generate the spectrogram for the current file.
  spect <- spectro(audio, f = audio@samp.rate, w1 = 1024, ovlp = 90,
  plot = FALSE)

  # Convert the spectrogram to a data frame.
  colnames(spect$amp) <- spect$time
  rownames(spect$amp) <- spect$freq
  spect_df <-
    spect$amp |>
    as_tibble(rownames = "freq") |>
    pivot_longer(
      -freq,
      names_to = "time",
      values_to = "amp"
    ) |>

```

```
mutate(
  freq = as.numeric(freq),
  time = as.numeric(time)
)

# Set the dynamic range for the spectrogram.
dyn = -80

# Floor the amplitude values at the dynamic range.
spect_df_floor <-
  spect_df |>
  mutate(
    amp_floor = case_when(
      amp < dyn ~ dyn,
      TRUE ~ amp
    )
  )

# Plot the spectrogram as a raster with red, dashed vertical lines
at the beginning and end of the detection window.

plot <- spect_df_floor %>%
  ggplot(aes(time, freq)) +
  geom_raster(aes(fill = amp_floor)) +
  guides(fill = "none") +
  theme_classic() +
  xlab("Time (s)") +
  ylab("Frequency (kHz)") +
  scale_fill_gradientn(colours = c("white", "grey", "black")) +
  geom_vline(xintercept = red_line_positions, color = "red", line-
type = "dashed")

# Save the plot as a PNG file in the subdirectory.
file_name <- tools::file_path_sans_ext(basename(file_path))
png(file = file.path(sub_dir, paste0(file_name, ".png")),
  width = 6, height = 3, units = "in", res = 100)
print(plot)
dev.off()
}
```

The directory and all its subdirectories within the “base_directory” folder are then searched for WAV files. Each WAV file is then saved to a list in the form of its file path.

```
# Get a list of all WAV files in the "clips" directory and its subdi-
rectories. If your folder that contains WAV files has a different
name, change "clips" to the appropriate name here.
wav_files <- list.files(file.path(base_directory, "clips"), pattern =
"\\.wav$", recursive = TRUE, full.names = TRUE)
```

After identifying the WAV files, the `generate_spectrogram` function is used to process the WAV files as described above. An example of the resulting spectrograms is shown in figure 6.

```
For (file_path in wav_files) {
  generate_spectrogram(file_path)
}
```

Combining Spectrograms and Audio Clips into MP4 Files

Now that both the clips of species detections and spectrograms have been made, they can be combined into a single video file (i.e., an MP4 file) so that the detections can be viewed and listened to simultaneously. This process uses a Python script. To begin, open the command prompt on the machine as described above and load the SPAM environment. Note that this process requires first loading the environment as described in the clip extraction section and requires that `ffmpeg` is installed on your machine.

```
conda activate spam_env
```

Now set the working directory to the folder containing the scripts by using the `"cd"` command. As before, `"/file/path/to/scripts/folder"` should be replaced with the actual file path.

```
cd /file/path/to/scripts/folder
```

Run the script for creating clips using the following code. Anytime a space is included in a filename or file path, the text should be enclosed by quotes to prevent any errors.

```
python "wav_spectrogram_combine.py"
```

Running this code produces a prompt asking the user for the path to the directory containing the WAV files of species detections. Below is the path to the `"clips"` folder within the shared folder.

```
Enter the directory containing WAV files: /path/to/GTR_software_folder/
clips
```

A prompt asking for the path to the folder containing the spectrogram files (i.e., PNG files) appears next. Below is the path to the “spectrograms” folder within the shared folder.

```
Enter the directory containing WAV files: /path/to/GTR_software_folder/  
spectrograms
```

And finally, a prompt asking for the output directory appears. Below is the path to the “mp4s” folder. This is the location of the resulting MP4 files.

```
Enter the directory containing WAV files: /path/to/  
GTR_software_folder/mp4s
```

If the correct paths are provided, the script states the number of matching audio-image pairs found and then begins building and writing the videos. The text following the stated number of audio-image pairs found is provided for each MP4 file made. The MP4 files are also saved using the same folder structure as the PNG and WAV files but are placed in the “mp4s” directory.

```
Found 39 audio-image pairs.
```

```
Moviepy - Building video /Users/benvernasco/Documents/GTR_software_  
folder/mp4s/Red Crossbill/12433_47548_A_BM_20220702_080000_08 51.mp4.
```

```
MoviePy - Writing audio in 12433_47548_A_BM_20220702_080000_08 51TEMP_  
MPY_wvf_snd.mp4
```

```
MoviePy - Done.
```

Creating Species Detection Processing File

Now that the MP4 files representing species detections are made, the file used for data entry (i.e., the clip processing file) can be made. The “clip_processing_file_creation.R” script creates a CSV file that can be used to record information about each detection during the review process. The script begins by creating a list of all the MP4 files created, each of which represents a detection to be reviewed. Information related to the time and location of each detection is extracted from folder and filenames and added to columns within the file. There are empty columns for storing information about species presence, requirements for additional review, the call type detected, multiple species calling within the clip, and additional notes. To begin the process of making the processing file, prepare a new R session for use:

```
library(dplyr)
```

```
library(tidyr)
```

```
library(lubridate)
```

The next step is to specify the relevant directories. The base directory should contain the “clips,” “spectrograms,” and “mp4s” directories. In this example, it is the current working directory, which is the accompanied shared folder. Then, the directory containing the MP4 files is defined, a list of all the MP4 files in the “mp4s” directory is created, and an empty data frame whose size is determined by the number of MP4 files is created.

```

base_directory <- getwd()
mp4_directory <- file.path(base_directory, "mp4s")

# Set working directory to folder containing MP4 files of detections
to be reviewed

setwd(mp4_directory)

# Make a list of all mp4 files to be reviewed.

mp4_files <- list.files(recursive = TRUE, pattern = "\\\\.mp4$", full.
names = TRUE)

# Create an empty data frame to store the filename parts.

df <- data.frame(matrix(ncol = 0, nrow = length(mp4_files)))

```

Now an object containing the file paths to all the MP4 files is created. These paths can be subdivided to populate the review sheet with identifying information, including information related to the time and location of a detection. The information contained in the folder structure and filenames can now be used to populate a data frame containing identifying information. In this example, the clips are stored in a folder representing the species name.

```

# Check if any mp4 files were found.
if (length(mp4_files) == 0) {
  stop("No .mp4 files found in the specified directory.")
}

# Split each file path into components.
split_paths <- strsplit(mp4_files, "/")

# Ensure that all elements are valid lists of strings.
if (!all(sapply(split_paths, is.character))) {
  stop("Invalid file paths detected.")
}

# Determine the maximum number of components in the paths.
max_parts <- max(sapply(split_paths, length))

# Pad the paths with "NA" (not applicable) to ensure that all have the
same length.
split_paths_padded <- lapply(split_paths, function(x) {
  # Extend to max_parts by adding NA if necessary.
  length(x) <- max_parts

  return(x)
})

# Combine the list of split paths into a data frame.

```

```
df <- data.frame(do.call(rbind, split_paths_padded), stringsAsFactors =
FALSE)

# Assign generic column names.
colnames(df) <- paste0("V", seq_len(ncol(df)))

# Inspect the resulting data frame.
head(df)

##              V1              V2
## 1 Red Crossbill 10387_47551_A_BM_20220619_061102_02 57.mp4
## 2 Red Crossbill 11110_47501_F_BM_20220623_100000_09 36.mp4
## 3 Red Crossbill 11162_47786_D_BM_20220604_120000_07 45.mp4
## 4 Red Crossbill 11439_47787_A_BM_20220624_060002_35 54.mp4
## 5 Red Crossbill 11809_47787_A_BM_20220624_060002_35 51.mp4
## 6 Red Crossbill 12433_47548_A_BM_20220702_080000_08 51.mp4

# Rename the columns appropriately.
colnames(df) <- c("species","filename")
```

In this example, the filenames are structured as “clip ID_hexagon name_station name_file prefix_date of detection_detection time of day_minutes and seconds since start that the detection occurred.” The following code is used to reformat this information and, in some cases, make the information into new columns.

```
# Make a new column with the filename. This will be used to create
columns containing identifying information.
df$fn<-df$filename

# Separate the filename into different components. The labels in the
c() function are the new column names, and there should be one entry
for each character sequence separated by an underscore.
df <- df %>% separate(fn, c("ID","hex","station","name","date","time","mm_
ss"), sep = "_")

# Remove the file extension from the last column.
df$mm_ss<-gsub(".mp4","",df$mm_ss)

# Fix the formatting of the station name.
df$station<-paste(df$hex,df$station, sep = "_")

# Fix the formatting of the date column to YYYYMMDD.
df$date <- ymd(df$date)

# Format the time column to hhmmss.
df$time<-hms(paste(substring(df$time,0,2), substring(df$time,3,4),
substring(df$time,5,6), sep = ":"))

# Reorder columns and remove unnecessary columns. Column names can be
added or removed from the following. Reordering the columns changes
```

the order of the columns in the data frame.

```
df <- df %>% select(filename, hex, station, date, time, mm_ss,
species, ID)
```

Now we have the identifying information about each detection within the data frame. The next step is to add columns to enter information about the detection during the manual review process. The columns to add and their uses are listed here:

- The “positive detection” column can be used to count the number of positive detections within a survey period in cases for which a certain number of positive detections per survey period or sampling location is desired.
- The “focal_species_present” column can be used to record if the species detections represent a true positive (1), a false positive (0), or a potential true positive (2).
- The “needs_further_review” column can be used to indicate if a particular detection needs additional review (1) or does not need additional review (0). Combining this with the “focal_species_present” values can be used to indicate detections which are true positives and do not need additional review (1/0), detections which are false positives (0/0), detections which are potential true positives but need further review (2/1), or detections which are potential false positives but need further review (0/1).
- The “call_type” column can be used to indicate the class-specific sound type present in the true-positive detections (see the “Sound Library Development” section for more details).
- The “multi_species” column can be used to indicate if multiple species beyond the detected species are present (1) or absent (0) in the clip. This can be helpful information for subsequent uses of the detections to retrain acoustic classifiers.
- The “notes” column can be used to record additional information related to the species detection that is worth noting, including the presence of non-animal sounds (e.g., rain, mechanical sounds).

The following code creates and appends the new columns to the data frame.

```
# nc (new columns) contains the names of the new columns for the data
entry sheet used during clip review.
nc <- c("positive detection","focal_species_present","needs_further_
review","call_type","multi_species","notes")
# Add the columns specified in nc and set all values to empty.
df[,nc]<=""
```

The data frame is now complete and can be written as a CSV file. The CSV file can be opened, and the review information can be entered. Reviewing clips is made easier by using additional formatting steps within Microsoft Excel; it is best to resave the CSV file as an Excel workbook (i.e., .xlsx) to maintain these formatting changes. An example of the CSV file created using the above code and additional formatting changes made in Excel can be found in figure 7.

```
setwd(base_directory)
write.csv(df,"example_clip_processing_file.csv", row.names = FALSE)
```

species	filename	ID	hex	station	name	date	time	mm_ss	positive detection	focal_species_present	needs_further_review	true_label	call_type	multi_species	notes
Red Crossbill	M_20220619	10387	47551	47551_A	BM	6/19/22	6H 11M 2S	02:57							
Red Crossbill	M_20220623	11110	47501	47501_F	BM	6/23/22	12H 0M 0S	09:36							
Red Crossbill	M_20220604	11162	47786	47786_D	BM	6/4/22	10H 0M 0S	07:45							
Red Crossbill	M_20220624	11439	47787	47787_A	BM	6/24/22	6H 0M 2S	35:54							
Red Crossbill	M_20220624	11809	47787	47787_A	BM	6/24/22	6H 0M 2S	35:51							
Red Crossbill	M_20220702	12433	47548	47548_A	BM	7/2/22	8H 0M 0S	08:51							
Red Crossbill	A_20220703	12926	47787	47787_A	BM	7/3/22	8H 0M 0S	07:51							
Red Crossbill	A_20220606	13637	47676	47676_G	HJA	6/6/22	7H 0M 2S	06:15							
Red Crossbill	M_20220702	14002	47548	47548_A	BM	7/2/22	8H 0M 0S	08:54							
Red Crossbill	M_20220619	14604	47551	47551_A	BM	6/19/22	6H 11M 2S	58:39							
Red Crossbill	A_20220606	14772	47676	47676_G	HJA	6/6/22	7H 0M 2S	07:15							
Red Crossbill	M_20220703	15213	47787	47787_A	BM	7/3/22	8H 0M 0S	06:36							
Red Crossbill	M_20220703	15792	47787	47787_A	BM	7/3/22	8H 0M 0S	06:18							
Red Crossbill	M_20220703	15988	47787	47787_A	BM	7/3/22	8H 0M 0S	07:27							
Red Crossbill	A_20220606	16217	47676	47676_G	HJA	6/6/22	7H 0M 2S	07:18							
Red Crossbill	M_20220619	16822	47551	47551_A	BM	6/19/22	6H 11M 2S	35:09							
Red Crossbill	M_20220606	1935	47676	47676_G	HJA	6/6/22	7H 0M 2S	07:09							
Red Crossbill	M_20220623	2024	47501	47501_F	BM	6/23/22	10H 0M 0S	09:48							
Red Crossbill	M_20220624	2246	47787	47787_A	BM	6/24/22	6H 0M 2S	08:51							
Red Crossbill	M_20220619	269	47551	47551_A	BM	6/19/22	6H 11M 2S	58:06							
Red Crossbill	M_20220619	32	47551	47551_A	BM	6/19/22	6H 11M 2S	58:24							
Red Crossbill	M_20220703	3499	47787	47787_A	BM	7/3/22	8H 0M 0S	06:09							
Red Crossbill	M_20220624	3561	47787	47787_A	BM	6/24/22	6H 0M 2S	35:57							
Red Crossbill	A_20220606	4176	47676	47676_G	HJA	6/6/22	7H 0M 2S	07:06							
Red Crossbill	M_20220619	4388	47551	47551_A	BM	6/19/22	6H 11M 2S	35:27							
Red Crossbill	M_20220623	4829	47501	47501_F	BM	6/23/22	10H 0M 0S	09:39							
Red Crossbill	M_20220606	498	47676	47676_G	HJA	6/6/22	7H 0M 2S	07:12							
Red Crossbill	M_20220623	5010	47637	47637_E	BM	6/3/22	12H 0M 0S	02:00							
Red Crossbill	M_20220623	5480	47501	47501_F	BM	6/23/22	10H 0M 0S	09:45							
Red Crossbill	M_20220623	5678	47501	47501_F	BM	6/23/22	10H 0M 0S	09:42							
Red Crossbill	M_20220624	5924	47787	47787_A	BM	6/24/22	6H 0M 2S	35:48							
Red Crossbill	A_20220606	7036	47676	47676_G	HJA	6/6/22	7H 0M 2S	06:24							
Red Crossbill	A_20220606	7264	47676	47676_G	HJA	6/6/22	7H 0M 2S	06:27							
Red Crossbill	A_20220606	7861	47676	47676_G	HJA	6/6/22	7H 0M 2S	06:18							
Red Crossbill	M_20220619	8336	47551	47551_A	BM	6/19/22	6H 11M 2S	35:30							
Red Crossbill	M_20220624	8467	47787	47787_A	BM	6/24/22	6H 0M 2S	35:36							
Red Crossbill	A_20220606	8710	47676	47676_G	HJA	6/6/22	7H 0M 2S	06:21							
Red Crossbill	M_20220619	9309	47551	47551_A	BM	6/19/22	6H 11M 2S	58:09							
Red Crossbill	M_20220619	9967	47551	47551_A	BM	6/19/22	6H 11M 2S	58:12							

Figure 7—Example of a clip processing file, created using the SPAM protocol, used for entering information gleaned during manual review of each automated species detection. The first nine columns from the left include identifying information about each detection (e.g., species detected, unique ID, location and time of detection). The remaining columns are used for entering review information, such as if the sound class detected is present, if further review of the detection is needed to confirm presence of the sound class, and the actual identity of the sound detected.

Sound Library Development and Reviewer Training and Assessment

The above sections provide materials for creating a relatively simple and efficient protocol for determining if species detections generated by acoustic classifiers are true or false positives. Generating accurate human-validated species detection data from acoustic classifier outputs, however, depends on knowing which sounds are reliable indicators of species presence and the ability of technicians to accurately recognize diagnostic sounds for species of interest. In this section, we use results from our own PAM efforts in the northern Blue Mountains of Oregon and Washington to demonstrate the process of building a regional library of sounds used for species recognition. We also provide 50 detections for each species in the sound library for the purpose of quantifying reviewer performance.

Sound Library Development

Most species of animals generate a diversity of acoustic signals. For example, woodpeckers make a variety of vocalizations and drums, both of which are conspicuous acoustic signals that may represent a sound class within an acoustic classifier. Vocalizations are often species-specific, but multiple, often co-occurring woodpecker species drum on resonant surfaces using their bill for long-distance communication. With a few exceptions, reliably identifying woodpecker species using drums from PAM data is not possible (Pieplow 2019). Additional examples of unreliable acoustic signals include the chip calls of many sparrows and warblers. Species detections generated by some commonly used acoustic classifiers, such as BirdNET, represent species-level detections. For these detections, the acoustic classifier does not differentiate among the various acoustic signals (hereafter “call types”) that a species produces (Kahl et al. 2021). As such, identifying a species in PAM data depends on understanding which call types produced by the target species can be used to reliably indicate its presence. Examples of species detections containing reliable sounds from ARU recordings are also needed for classifying sounds and training technicians because publicly available field recordings are mostly made using specialized recording equipment (e.g., shotgun microphone with parabolic dish) that creates much higher quality recordings than the recordings generated by the ARUs used in PAM.

To develop a sound library, we first composed a list of species present in the northern Blue Mountains. We began with *Birds of the Umatilla National Forest* (USDA FS, n.d.) and supplemented it with the BirdNET species list for the area and the recommendations of Denny (2023), a local biologist with extensive experience surveying birds in the northern Blue Mountains. The final species list consisted of 207 bird species. Species detections were then generated by processing nearly 50,000 hours of audio recordings collected during the 2022 field season (see Duarte et al. 2024a) with BirdNET v2.4. Technicians familiar with identifying species by sound then conducted a manual review of a random sample of the detections that had a confidence score of >0.5 . Technicians had access to websites containing resources for identifying bird species by sound (see app. 2).

During the review process, technicians annotated detections as true or false positives; true positives were grouped into species-specific call types and labeled using species-specific four-letter codes (The Institute for Bird Populations, n.d.) and a number. For example, the American goshawk alarm call is represented by “AGOS_ct1”. The “ct” stands for call type, and it was assigned as “ct1” because it was the first AGOS (American goshawk) call type discovered in our project. Technicians flagged potential species call types as

they discovered them. All technicians and supervisors who reviewed detections attended weekly group meetings to discuss new call types. If the group agreed that a call type could be used to reliably identify a given species, they added a new call type to a folder with the name of the call type (e.g., AGOS_ct1). Each call type folder was stored in a folder named after the species represented by the call type (e.g., “American Goshawk”). All species folders were stored in a “diagnostic clips” folder, which was accessible during the review process by all technicians reviewing detections. Following the completion of reviewing the 2022 species detection data, the “diagnostic clips” folder contained 107 species and 279 unique, species-specific, call type combinations. The “diagnostic clips” folder resulting from the above process can be found in the data repository accompanying this report (Vernasco and Duarte 2025).

Assessing Reviewer Accuracy

Reviewers of species detections have differing abilities for recognizing diagnostic call types, and this variability can have consequences for inferences made from validated detection data (Iles et al. 2024). Quantifying variability among reviewers can provide insight into the extent of the issue, identify call types on which reviewers should focus their training, and generate confidence in the species detection data. One approach for assessing reviewer accuracy is to have reviewers annotate the same detections independently and then compare the resulting annotations. To facilitate this process, we provide a random selection of up to 50 previously annotated detections, in the form of MP4 files containing the audio and spectrogram of the detection window for each of the 107 species that we detected in the northern Blue Mountains (see the “validation_clips” folder within the accompanying software folder by Vernasco and Duarte 2025). We selected up to 25 true-positive and 25 false-positive detections for each species. We labeled the detections using species four-letter codes and a number (e.g., AGOS_clip_2.mp4). Before relabeling, we randomly sorted the detections such that the labels do not indicate if it is a true- or false-positive detection. The processing sheet within the “validation clips” folder can be filled out using the validation protocol in appendix 2. The call types specified for positive detections are those in the “diagnostic clips” folder that is also provided in the accompanying software folder (Vernasco and Duarte 2025).

To evaluate reviewer accuracy using the provided validation clips, multiple reviewers should validate the same detections. Reviewer scores can then be compared, and call types with high discrepancy between reviewers can be identified. Additional time and training specifically focusing on how to evaluate call types with high discrepancies can facilitate more accurate assessments of clips. Monitoring objectives are more likely to be met if efforts are directed toward achieving comparable degrees of accuracy among all reviewers before beginning the process of collecting data used for inference.

Conclusions

The combination of PAM and acoustic classifiers is revolutionizing wildlife monitoring and providing new challenges for the researchers and managers who use PAM. We offer several forms of guidance for approaching the opportunities and challenges of PAM. We describe some sampling designs that can be used to implement PAM, introduce the SPAM protocol to help process and validate detections, and provide training materials and associated code and data to accomplish these tasks. We show the construction and use of the SPAM protocol with an example of a PAM program that we are implementing on the Umatilla and Wallowa-Whitman National Forests in the Pacific Northwest. Notably, throughout this document, we assumed PAM is an effective way to achieve monitoring objectives for the management program at hand. However, we fully recognize this will not always be the case. Most certainly, PAM is a promising approach to fill many critical information needs, particularly those associated with audible wildlife on managed forest landscapes. PAM is not a monitoring panacea, however, and other approaches may be warranted depending on the target species and monitoring objectives. Still, we hope this report helps further lower the barrier for others to implement PAM across managed lands by providing clear guidelines and accessible code to promote more efficient generation of reliable species detections from PAM efforts.

Acknowledgments

We owe a tremendous amount of gratitude to biologists who helped collect the Blue Mountains PAM data. We thank Casey Kroening, Meredith Mathews, Lila Fried, and Max Maddox for their help validating the species detections associated with this report. We are grateful to the U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station's Bioacoustics Lab for their assistance in establishing the field protocols and logistical support for the collection of PAM data in the northern Blue Mountains. Joshua F. Goldberg, Ivana Mali, and Damon B. Lesmeister provided valuable comments that improved an earlier version of this document. This work was financially supported by the Forest Service, Pacific Northwest Research Station, Collaborative Forest Landscape Restoration Program; the Wallowa-Whitman National Forest; and the Oregon Department of Fish and Wildlife.

References

- Appel, C.L.; Lesmeister, D.B.; Duarte, A. [et al.]. 2023.** Using passive acoustic monitoring to estimate northern spotted owl landscape use and pair occupancy. *Ecosphere*. 14(2): e4421. <https://doi.org/10.1002/ecs2.4421>.
- Astaras, C.; Linder, J.M.; Wrege, P. [et al.]. 2017.** Passive acoustic monitoring as a law enforcement tool for Afrotropical rainforests. *Frontiers in Ecology and the Environment*. 15(5): 233–234. <https://doi.org/10.1002/fee.1495>.
- Balvanera, S.M.; Mac Aodha, O.; Weldy, M.J. [et al.]. 2025.** Whombat: an open-source audio annotation tool for machine learning assisted bioacoustics. *Methods in Ecology and Evolution*. 16(1): 19–28. <https://doi.org/10.1111/2041-210X.14468>.
- Banner, K.M.; Irvine, K.M.; Rodhouse, T.J. [et al.]. 2018.** Improving geographically extensive acoustic survey designs for modeling species occurrence with imperfect detection and misidentification. *Ecology and Evolution*. 8(12): 6144–6156. <https://doi.org/10.1002/ece3.4162>.
- Barry, B.R.; Holbrook, J.D.; Vogeler, J.C. [et al.]. 2024.** Using spaceborne LiDAR to reveal drivers of animal demography. *Ecological Applications*. 34(8): e3048. <https://doi.org/10.1002/eap.3048>.
- Baumbusch, R.; Duarte, A.; Peterson, J.T. 2025.** Invisible hand of sampling for management: underlying needs to survey a threatened seabird can bias aggregated data. *Journal of Biogeography*. 52(3): 699–711. <https://doi.org/10.1111/jbi.15068>.
- Bell, D.M.; Gregory, M.J.; Yang, Z. 2024.** Hindcasting and updating Landsat-based forest structure mapping across years to support forest management and planning. *Forest Ecology and Management*. 572: 122239. <https://doi.org/10.1016/j.foreco.2024.122239>.
- Borker, A.L.; Halbert, P.; Mckown, M.W. [et al.]. 2015.** A comparison of automated and traditional monitoring techniques for marbled murrelets using passive acoustic sensors. *Wildlife Society Bulletin*. 39(4): 813–818. <https://doi.org/10.1002/wsb.608>.
- Bota, G.; Manzano-Rubio, R.; Fanlo, H. [et al.]. 2024.** Passive acoustic monitoring and automated detection of the American bullfrog. *Biological Invasions*. 26(4): 1269–1279. <https://doi.org/10.1007/s10530-023-03244-8>.
- Braulik, G.; Wittich, A.; Macaulay, J. [et al.]. 2017.** Acoustic monitoring to document the spatial distribution and hotspots of blast fishing in Tanzania. *Marine Pollution Bulletin*. 125(1–2): 360–366. <https://doi.org/10.1016/j.marpolbul.2017.09.036>.
- Breheny, P.; Burchett, W. 2017.** Visualization of regression models using visreg. *The R Journal*. 9(2): 56–71. <https://journal.r-project.org/archive/2017/RJ-2017-046/index.html>. (June 18, 2025).
- Brooks, M.E.; Kristensen, K.; van Benthem, K.J. [et al.]. 2017.** glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling. *The R Journal*. 9(2): 378–400. <https://journal.r-project.org/archive/2017/RJ-2017-066/index.html>. (June 18, 2025).
- Brunk, K.M.; Goldberg, J.F.; Maxwell, C. [et al.]. 2025.** Bioregional-scale acoustic monitoring can support fire-prone forest restoration planning. *Frontiers in Ecology and the Environment*. Early View: e2843. <https://doi.org/10.1002/fee.2843>.
- Brunk, K.M.; Gutiérrez, R.J.; Peery, M.Z. [et al.]. 2023.** Quail on fire: changing fire regimes may benefit mountain quail in fire-adapted forests. *Fire Ecology*. 19: 19. <https://doi.org/10.1186/s42408-023-00180-9>.
- Camarretta, N.; Harrison, P.A.; Bailey, T. [et al.]. 2020.** Monitoring forest structure to guide adaptive management of forest restoration: a review of remote sensing approaches. *New Forests*. 51(4): 573–596. <https://doi.org/10.1007/s11056-019-09754-5>.

- Cameron, J.; Crosby, A.; Paszkowski, C.; Bayne, E. 2020.** Visual spectrogram scanning paired with an observation–confirmation occupancy model improves the efficiency and accuracy of bioacoustic anuran data. *Canadian Journal of Zoology*. 98(11): 733–742. <https://doi.org/10.1139/cjz-2020-0103>.
- Chambert, T.; Waddle, J.H.; Miller, D.A.W. [et al.]. 2018.** A new framework for analysing automated acoustic species detection data: occupancy estimation and optimization of recordings post-processing. *Methods in Ecology and Evolution*. 9(3): 560–570. <https://doi.org/10.1111/2041-210X.12910>.
- Chesser, R.T.; Billerman, S.M.; Burns, K.J. [et al.]. 2024.** Checklist of North American Birds. <https://checklist.americanornithology.org/>. (June 23, 2025).
- Chronister, L.M.; Larkin, J.T.; Rhinehart, T.A. [et al.]. 2024.** Evaluating the predictors of habitat use and successful reproduction in a model bird species using a large-scale automated acoustic array. *Ecography*. 2025(3): e06940. <https://doi.org/10.1111/ecog.06940>.
- Clement, M.J.; Rodhouse, T.J.; Ormsbee, P.C. [et al.]. 2014.** Accounting for false-positive acoustic detections of bats using occupancy models. *Journal of Applied Ecology*. 51(5): 1460–1467. <https://doi.org/https://doi.org/10.1111/1365-2664.12303>.
- Cornell Lab. 2025a.** All about birds. Cornell University. <https://www.allaboutbirds.org/guide/>. (August 27, 2025).
- Cornell Lab. 2025b.** Macaulay Library. Cornell University. <https://www.macaulaylibrary.org/>. (August 27, 2025).
- Cornell Lab. 2025c.** Peterson field guide to bird sounds. Cornell University. <https://academy.allaboutbirds.org/peterson-field-guide-to-bird-sounds/>. (August 27, 2025).
- Dale, S.S.; Jenkins, J.M.A.; Ruff, Z.J. [et al.]. 2022.** Distinguishing sex of northern spotted owls with passive acoustic monitoring. *Journal of Raptor Research*. 56(3): 287–299. <https://doi.org/10.3356/JRR-21-33>.
- Darras, K.; Batáry, P.; Furnas, B.J. [et al.]. 2019.** Autonomous sound recording outperforms human observation for sampling birds: a systematic map and user guide. *Ecological Applications*. 29(6): e01954. <https://doi.org/10.1002/eap.1954>.
- Darras, K.F.A.; Rountree, R.A.; Van Wilgenburg, S.L. [et al.]. 2025.** Worldwide soundscapes: a synthesis of passive acoustic monitoring across realms. *Global Ecology and Biogeography*. 34(5): e70021. <https://doi.org/10.1111/geb.70021>.
- Davis, R.J.; Ohmann, J.L.; Kennedy, R.E. [et al.]. 2015.** Northwest Forest Plan—the first 20 years (1994–2013): status and trends of late-successional and old-growth forests. Gen. Tech. Rep. PNW-GTR-911. Portland, OR: U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station. 112 p. <https://doi.org/10.2737/pnw-gtr-911>.
- Denny, M. 2023.** Personal communication. Biologist, Blue Mountain Audubon Society, PO Box 1106, Walla Walla, WA 99362-0022.
- Doser, J.W.; Finley, A.O.; Weed, A.S.; Zipkin, E.F. 2021.** Integrating automated acoustic vocalization data and point count surveys for estimation of bird abundance. *Methods in Ecology and Evolution*. 12(6): 1040–1049. <https://doi.org/https://doi.org/10.1111/2041-210X.13578>.
- Duarte, A.; Peterson, J.T.; Pearl, C.A. [et al.]. 2020.** Estimation of metademographic rates and landscape connectivity for a conservation-reliant anuran. *Landscape Ecology*. 35(6): 1459–1479. <https://doi.org/10.1007/s10980-020-01030-8>.
- Duarte, A.; Spaan, R.S.; Peterson, J.T. [et al.]. 2025.** Bayesian networks facilitate updating of species distribution and habitat suitability models. *Ecological Modelling*. 501: 110982. <https://doi.org/https://doi.org/10.1016/j.ecolmodel.2024.110982>.
- Duarte, A.; Vernasco, B.J.; Weldy, M.J. [et al.]. 2024a.** Northern Blue Mountains wildlife monitoring, 2022–2023. Olympia, WA: U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station. 45 p. <https://research.fs.usda.gov/sites/default/files/2024-12/pnw-duarte-et-al-northern-blue-mountains-wildlife-monitoring2022-2023.pdf>. (June 23, 2025).

- Duarte, A.; Weldy, M.J.; Lesmeister, D.B. [et al.]. 2024b.** Passive acoustic monitoring and convolutional neural networks facilitate high-resolution and broadscale monitoring of a threatened species. *Ecological Indicators*. 162: 112016. <https://doi.org/10.1016/j.ecolind.2024.112016>.
- Gannon, J.J.; Grant, T.A.; Vacek, S.C. [et al.]. 2024.** Crisis on the prairies revisited. *Ecological Restoration*. 42(1): 64–76. <https://doi.org/10.3368/er.42.1.64>.
- Gaylord, M.A.; Duarte, A.; McComb, B.C.; Ratliff, J. 2023.** Passive acoustic recorders increase white-headed woodpecker detectability in the Blue Mountains. *Journal of Field Ornithology*. 94(4): art 1. <https://doi.org/10.5751/jfo-00330-940401>.
- Ghani, B.; Denton, T.; Kahl, S.; Klinck, H. 2023.** Global birdsong embeddings enable superior transfer learning for bioacoustic classification. *Scientific Reports*. 13(1): 22876. <https://doi.org/10.1038/s41598-023-49989-z>.
- Hartig, F.; Abrego, N.; Bush, A. [et al.]. 2024.** Novel community data in ecology-properties and prospects. *Trends in Ecology & Evolution*. 39(3): 280–293. <https://doi.org/10.1016/j.tree.2023.09.017>.
- Hopkins, J.M.; Edwards, W.; Schwarzkopf, L. 2022.** Invading the soundscape: exploring the effects of invasive species' calls on acoustic signals of native wildlife. *Biological Invasions*. 24(11): 3381–3393. <https://doi.org/10.1007/s10530-022-02856-w>.
- Huus, J.; Kelly, K.G.; Bayne, E.M.; Knight, E.C. 2025.** HawkEars: a regional, high-performance avian acoustic classifier. *Ecological Informatics*. 87: 103122. <https://doi.org/10.1016/j.ecoinf.2025.103122>.
- Iles, D.T.; Francis, C.M.; Smith, A.C. [et al.]. 2024.** Two listeners detect slightly more birds than a single listener when interpreting acoustic recordings. *Ornithological Applications*. 126(4): duae030. <https://doi.org/10.1093/ornithapp/duae030>.
- The Institute for Bird Populations. [N.d.].** Standardized 4- and 6-letter bird species codes. <https://www.birdpop.org/pages/birdSpeciesCodes.php>. (July 2, 2025).
- Johnson, F.A.; Breininger, D.R.; Duncan, B.W. [et al.]. 2011.** A Markov decision process for managing habitat for Florida scrub-jays. *Journal of Fish and Wildlife Management*. 2(2): 234–246. <https://doi.org/10.3996/012011-JFWM-003>.
- Kahl, S.; Wood, C.M.; Eibl, M.; Klinck, H. 2021.** BirdNET: a deep learning solution for avian diversity monitoring. *Ecological Informatics*. 61: 101236. <https://doi.org/10.1016/j.ecoinf.2021.101236>.
- Kéry, M.; Royle, J.A. 2021.** Applied hierarchical modeling in ecology: analysis of distribution, abundance and species richness in R and BUGS. Volume 2: dynamic and advanced models. Netherlands: Academic Press. <https://doi.org/10.1016/C2015-0-04070-9>.
- Lahoz-Monfort, J.J.; Magrath, M.J.L. 2021.** A comprehensive overview of technologies for species and habitat monitoring and conservation. *BioScience*. 71(10): 1038–1062. <https://doi.org/10.1093/biosci/biab073>.
- Lapp, S.; Rhinehart, T.; Freeland-Haynes, L. [et al.]. 2023.** OpenSoundscape: an open-source bioacoustics analysis package for Python. *Methods in Ecology and Evolution*. 14(9): 2321–2328. <https://doi.org/10.1111/2041-210X.14196>.
- Larson, A.J.; Belote, R.T.; Williamson, M.A.; Aplet, G.H. 2013.** Making monitoring count: project design for active adaptive management. *Journal of Forestry*. 111(5): 348–356. <https://doi.org/10.5849/jof.13-021>.
- Lesmeister, D.B.; Jenkins, J.M.A. 2022.** Integrating new technologies to broaden the scope of northern spotted owl monitoring and linkage with USDA forest inventory data. *Frontiers in Forests and Global Change*. 5: 966978. <https://doi.org/10.3389/ffgc.2022.966978>.
- Ligges, U.; Krey, S.; Mersmann, O. [et al.]. 2024.** tuneR: analysis of music and speech. <https://cran.r-project.org/package=tuneR>. (June 23, 2025).

- Lyons, J.E.; Runge, M.C.; Laskowski, H.P.; Kendall, W.L. 2008.** Monitoring in the context of structured decision-making and adaptive management. *The Journal of Wildlife Management*. 72(8): 1683–1692. <https://doi.org/10.2193/2008-141>.
- Moore, C.T.; Conroy, M.J. 2006.** Optimal regeneration planning for old-growth forest: addressing scientific uncertainty in endangered species recovery through adaptive management. *Forest Science*. 52(2): 155–172. <https://doi.org/10.1093/forestscience/52.2.155>.
- Morrison, M.L.; Block, W.M.; Strickland, M.D. [et al.]. 2008.** *Wildlife study design*. 2nd ed. New York: Springer. 375 p. <https://doi.org/10.1007/978-0-387-75528-1>.
- Navine, A.K.; Camp, R.J.; Weldy, M.J. [et al.]. 2024a.** Counting the chorus: a bioacoustic indicator of population density. *Ecological Indicators*. 169: 112930. <https://doi.org/10.1016/j.ecolind.2024.112930>.
- Navine, A.K.; Denton, T.; Weldy, M.J.; Hart, P.J. 2024b.** All thresholds barred: direct estimation of call density in bioacoustic data. *Frontiers in Bird Science*. 3: 1380636. <https://doi.org/10.3389/fbirs.2024.1380636>.
- Peterson, J.T.; Freeman, M.C. 2016.** Integrating modeling, monitoring, and management to reduce critical uncertainties in water resource decision making. *Journal of Environmental Management*. 183(2): 361–370. <https://doi.org/10.1016/j.jenvman.2016.03.015>.
- Pieplow, N. 2019.** *Peterson field guide to bird sounds of western North America*. New York: Houghton Mifflin Harcourt. 648 p.
- Reid, D.S.; Wood, C.M.; Whitmore, S.A. [et al.]. 2022.** Breeding status shapes territoriality and vocalization patterns in spotted owls. *Journal of Avian Biology*. 2022(8): e02952. <https://doi.org/10.1111/jav.02952>.
- Rhinehart, T.A.; Turek, D.; Kitzes, J. 2022.** A continuous-score occupancy model that incorporates uncertain machine learning output from autonomous biodiversity surveys. *Methods in Ecology and Evolution*. 13(8): 1778–1789. <https://doi.org/10.1111/2041-210X.13905>.
- Ross, S.R.P.-J.; O’Connell, D.P.; Deichmann, J.L. [et al.]. 2023.** Passive acoustic monitoring provides a fresh perspective on fundamental ecological questions. *Functional Ecology*. 37(4): 959–975. <https://doi.org/10.1111/1365-2435.14275>.
- Ruff, Z.J.; Lesmeister, D.B.; Jenkins, J.M.A.; Sullivan, C.M. 2023.** PNW-Cnet v4: automated species identification for passive acoustic monitoring. *SoftwareX*. 23: 101473. <https://doi.org/10.1016/j.softx.2023.101473>.
- Rugg, N.M.; Jenkins, J.M.A.; Lesmeister, D.B. 2023.** Western screech-owl occupancy in the face of an invasive predator. *Global Ecology and Conservation*. 48: e02753. <https://doi.org/10.1016/j.gecco.2023.e02753>.
- Sethi, S.S.; Bick, A.; Chen, M.-Y. [et al.]. 2024.** Large-scale avian vocalization detection delivers reliable global biodiversity insights. *Proceedings of the National Academy of Sciences*. 121(33): e2315933121. <https://doi.org/10.1073/pnas.2315933121>.
- Spinu, V.; Grolemond, G.; Wickham, H. 2023.** lubridate 1.9.4. <https://lubridate.tidyverse.org>. (June 23, 2025).
- Sueur, J.; Aubin, T.; Simonis, C. 2023.** seewave: an R package dedicated to sound analysis and synthesis. <https://doi.org/10.1080/09524622.2008.9753600>.
- Temperli, C.; Bugmann, H.; Elkin, C. 2012.** Adaptive management for competing forest goods and services under climate change. *Ecological Applications*. 22(8): 2065–2077. <https://doi.org/10.1890/12-0210.1>.
- Thompson, S.K. 2012.** *Sampling*. 3rd ed. Hoboken, NJ: John Wiley & Sons. 436 p. <https://doi.org/10.1002/9781118162934>.
- Thompson, W.L. 2004.** *Sampling rare or elusive species: concepts, designs, and techniques for estimating population parameters*. Washington, DC: Island Press. 429 p. <https://doi.org/10.1017/S0376892906262921>.

- Thompson, W.L.; White, G.C.; Gowan, C. 1998.** Monitoring vertebrate populations. San Diego: Academic Press. 365 p.
- Tomar, S. 2006.** Converting video formats with FFmpeg. *Linux Journal*. 146: 10.
- Tosa, M.I.; Dziedzic, E.H.; Appel, C.L. [et al.]. 2021.** The rapid rise of next-generation natural history. *Frontiers in Ecology and Evolution*. 9: 698131. <https://doi.org/10.3389/fevo.2021.698131>.
- Udell, B.J.; Straw, B.R.; Loeb, S.C. [et al.]. 2024.** Using mobile acoustic monitoring and false-positive N-mixture models to estimate bat abundance and population trends. *Ecological Monographs*. 94(4): e1617. <https://doi.org/10.1002/ecm.1617>.
- U.S. Department of Agriculture, Forest Service [USDA FS]. [N.d.].** Birds of the Umatilla National Forest. F14-SO-08-03 [Brochure]. Pendleton, OR: Umatilla National Forest.
- Vernasco, B.J.; Duarte, A. 2025.** Files and code for the Simple Passive Acoustic Monitoring (SPAM) protocol [Database]. Version 1.0.0. <https://doi.org/10.5281/zenodo.15115814>.
- Weldy, M.J.; Lesmeister, D.B.; Yackulic, C.B. [et al.]. 2023.** Long-term monitoring in transition: resolving spatial mismatch and integrating multistate occupancy data. *Ecological Indicators*. 146: 109815. <https://doi.org/10.1016/j.ecolind.2022.109815>.
- Wickham, H. 2023.** stringr: simple, consistent wrappers for common string operations. R package version 1.5.1. <https://stringr.tidyverse.org>. (June 23, 2025).
- Wickham, H.; Chang, W.; Henry, L. [et al.]. 2024.** ggplot2: create elegant data visualisations using the grammar of graphics. <https://ggplot2.tidyverse.org>. (June 23, 2025).
- Wickham, H.; François, R.; Henry, L. [et al.]. 2025a.** dplyr: a grammar of data manipulation. R package version 1.1.4. <https://dplyr.tidyverse.org>. (July 22, 2025).
- Wickham, H.; Vaughan, D.; Girlich, M. 2025b.** tidyr: tidy messy data. R package version 1.3.1. <https://tidyr.tidyverse.org>. (July 22, 2025).
- Witmer, G.W. 2005.** Wildlife population monitoring: some practical considerations. *Wildlife Research*. 32(3): 259–263. <https://doi.org/10.1071/WR04003>.
- Wood, C.M.; Barceinas Cruz, A.; Kahl, S. 2023a.** Pairing a user-friendly machine-learning animal sound detector with passive acoustic surveys for occupancy modeling of an endangered primate. *American Journal of Primatology*. 85(8): e23507. <https://doi.org/https://doi.org/10.1002/ajp.23507>.
- Wood, C.M.; Kahl, S. 2024.** Guidelines for appropriate use of BirdNET scores and other detector outputs. *Journal of Ornithology*. 165: 777–782. <https://doi.org/10.1007/s10336-024-02144-5>.
- Wood, C.M.; Kahl, S.; Barnes, S. [et al.]. 2023b.** Passive acoustic surveys and the BirdNET algorithm reveal detailed spatiotemporal variation in the vocal activity of two anurans. *Bioacoustics*. 32(5): 532–543. <https://doi.org/10.1080/09524622.2023.2211544>.
- Wood, C.M.; Socolar, J.; Kahl, S. [et al.]. 2024.** A scalable and transferable approach to combining emerging conservation technologies to identify biodiversity change after large disturbances. *Journal of Applied Ecology*. 61(4): 797–808. <https://doi.org/10.1111/1365-2664.14579>.
- Xeno-canto. 2025.** Xeno-canto. Xeno-canto Foundation. <https://xeno-canto.org/>. (August 27, 2025).

Appendix 1: Minimum Information Needed to Extract Audio Clips of Species Detections

The “creating_clip_extraction_file.R” script uses output from BirdNET v2.4 to create a CSV file containing the information needed to extract and organize audio clips containing species detections. To facilitate using the Simple Passive Acoustic Monitoring (SPAM) protocol with output from other acoustic classifiers, figure A1.1 shows the information needed to run the “clip_extraction.py” script, which creates shorter audio clips that contain species detections. The following three columns are required:

- The “full_path” column contains the path to the raw audio recording made by an autonomous recording unit, or ARU, that contains the respective species detection.
- The “onset” column contains the number of seconds from the start of the recording to the occurrence of the species detection.
- The “species” column denotes the species or sound class detected.

We found it more efficient to organize the extracted audio clips when the following columns are also included:

- The “wav_fn” column contains the names of the original audio files without the extension. Including the original filenames of the extracted audio clips containing species detections will allow the raw audio recordings to be easily matched programmatically.
- The “mm_ss” column contains the minutes and seconds from the start of the audio recording to the occurrence of the species detection. While redundant with the “onset” column, appending the “mm_ss” column to the end of filenames of the extracted audio clips makes manually locating the detection in the raw audio recordings much easier.
- The “site” column denotes the sampling location identifier. The sampling location identifier does not have to be included in the filenames of the extracted audio clips, but it can be helpful to include as one of the folders for organizing the extracted audio clips that contain species detections.

full_path	onset	species	wav_fn	mm_ss	site
/Volumes/One Touch/2022_ARUs/47501/47501_F/Data/BM_20220623_100000.wav	576	Red Crossbill	BM_20220623_100000	0936	Site 1
/Volumes/One Touch/2022_ARUs/47551/47551_A/Data/subdirectory_1/BM_20220619_061102.wav	3492	Red Crossbill	BM_20220619_061102	5812	Site 2
/Volumes/One Touch/2022_ARUs/47501/47501_F/Data/BM_20220623_100000.wav	579	Red Crossbill	BM_20220623_100000	0939	Site 1
/Volumes/One Touch/2022_ARUs/47551/47551_A/Data/subdirectory_1/BM_20220619_061102.wav	2130	Red Crossbill	BM_20220619_061102	3530	Site 2
/Volumes/One Touch/2022_ARUs/47501/47501_F/Data/BM_20220623_100000.wav	585	Red Crossbill	BM_20220623_100000	0945	Site 3
/Volumes/One Touch/2022_ARUs/47676/47676_G/Data/subdirectory_3/HJA_20220606_070002.wav	435	Red Crossbill	HJA_20220606_070002	0715	Site 2
/Volumes/One Touch/2022_ARUs/47551/47551_A/Data/subdirectory_1/BM_20220619_061102.wav	3486	Red Crossbill	BM_20220619_061102	5806	Site 1
/Volumes/One Touch/2022_ARUs/47501/47501_F/Data/BM_20220623_100000.wav	588	Red Crossbill	BM_20220623_100000	0948	Site 1
/Volumes/One Touch/2022_ARUs/47551/47551_A/Data/subdirectory_1/BM_20220619_061102.wav	3519	Red Crossbill	BM_20220619_061102	5839	Site 3
/Volumes/One Touch/2022_ARUs/47551/47551_A/Data/subdirectory_1/BM_20220619_061102.wav	2109	Red Crossbill	BM_20220619_061102	3509	Site 3

Figure A1.1— Structure of the CSV file needed to extract clips using the “clip_extraction.py” script.

Appendix 2: Example of an Audio Clip Validation Protocol

In this protocol, we provide an example of how to validate species detections produced by the BirdNET analyzer using recordings from automated recording units. In this example, we use recordings for multiple weeks from multiple locations during the breeding season. We quantify species occurrence for each week of recording to develop species-level encounter histories for occupancy modeling. This allows for the development of species distribution models while accounting for imperfect detection. For each detection, we record additional information related to the presence of other species and the species-specific vocalization type in the datasheet for the development and improvement of automated species recognizers. See figure A2.1 for an example of the datasheet used to validate species detections.

Observers may differ in their ability to recognize certain species or call types of a given species. While this can be measured by having all observers complete standardized assessments, statistical techniques can also be used to quantify and control for observer differences. In the following examples, we accomplish this by having observers validate species detections for each observation period (in this case, each week of recording for each station). We then include a variable denoting the observer identity for each week in statistical models analyzing species occurrence.

Useful Links for Learning and Identifying Species Vocalizations

- All About Birds (Cornell Lab 2025a): <https://www.allaboutbirds.org/guide/>
- Macaulay Library (Cornell Lab 2025b): <https://www.macaulaylibrary.org/>
- Peterson Field Guide to Bird Sounds (Cornell Lab 2025c): <https://academy.allaboutbirds.org/peterson-field-guide-to-bird-sounds/>
- Xeno-canto (2025): <https://xeno-canto.org/>

Validation Protocol

1. Open the folder containing the MP4 files of detections to be validated and the weekly detection spreadsheets used for data entry. Open the next available data entry spreadsheet and navigate to the corresponding station and the week within that station that contains the MP4 files representing the species detections.
2. Append the data sheet filename by adding your initials to the end of the filename to track which observer validated the detections.
3. Listen to the clip for any sounds made by the species in the “species” column. If you are not familiar with the sounds of the respective species, use the “Useful Links for Learning and Identifying Species Vocalizations” above to find relevant examples. In this example, BirdNET creates species-level detections that are not specific to the different sounds a given species can make; validation requires familiarity with the multitude of vocalizations of each species.
4. After listening to the clip, ensure that the value in the “ID” column (fig. A2.1) matches the ID included in the filename of the clip you listened to. If the observer is 100 percent confident that the focal species indicated in the “species” column is present in the audio clip, a “1” should be entered into the “focal_species_present” column. If the observer is

date	time	mm_ss	Species	Station	Week	ID	positive_detection	focal_species_present	needs_further_review	true_label	call_type	multi_species	notes
5/29/22	11	4M 33S	American Dipper	47664_C	1	12180		2	1				
5/30/22	5	7M 48S	American Dipper	47664_C	1	16133		2	1				
5/30/22	19	56M 36S	American Dipper	47664_F	1	4411		0	1				
5/27/22	5	16M 9S	American Kestrel	47664_F	1	6123	1	1	0		ct1	1	sm, wind
5/26/22	14	2M 54S	American Kestrel	47664_F	1	16652	2	1	0		ct1	1	sm, wind
5/26/22	4	8M 33S	American Robin	47664_B	1	1366		2	1				
5/25/22	19	1M 24S	American Robin	47664_B	1	1735	1	1	0		ct4	1	airplane
5/25/22	19	1M 42S	American Robin	47664_B	1	1832	2	1	0		ct4	1	airplane, insect
5/25/22	19	2M 6S	American Robin	47664_B	1	6470	3	1	0		ct2	0	
5/26/22	4	9M 15S	American Robin	47664_B	1	8755							
5/25/22	19	1M 57S	American Robin	47664_B	1	11689							
5/25/22	19	1M 36S	American Robin	47664_B	1	14696							
5/25/22	19	1M 30S	American Robin	47664_B	1	15741							
5/25/22	19	1M 12S	American Robin	47664_B	1	16128							
5/25/22	19	1M 18S	American Robin	47664_B	1	16175							
5/26/22	19	45M 57S	American Robin	47664_C	1	2321	1	1	0		ct2	0	wind

Figure—A2.1. Example of a data sheet used to record information about species detections. The “date” column contains the date the detection occurred. The “time” column contains the hour of the day the detection occurred. The “mm_ss” column contains the minutes and seconds from the start of the recording to the occurrence of the detection. The “Species” column contains the name of the species detected. The “Station” column contains the name of the recording station at which the detection occurred. The “Week” column contains the week of the monitoring period in which the detection occurred. The “ID” column contains the unique identifier of the detection. The “positive_detection” column is used to keep track of the number of positive detections for each species during each week at each station. The “focal_species_present” column indicates if the species is present (1), absent (0), or if the observer is unsure (2). The “needs_further_review” column indicates if the detection needs additional review, “1” for needed and “0” for not needed. The “true_label” column indicates the true identity of the species present. The “call_type” column indicates the species sound type used for identification. The “multi_species” column indicates if multiple species are present in the recording, “1” for present and “0” for absent. The “notes” column is used to record any additional information about each detection. See the “Validation Protocol” section in this appendix for additional information about the columns.

100 percent confident that the focal species indicated in the “species” column is absent in the audio clip, enter a “0” into the “focal_species_present” column. If the observer is not 100 percent confident, then a “2” should be entered into the “focal_species_present” column. If the audio in the clip contains a sound that is unclassifiable (i.e., further review will not result in a definitive classification), a “3” should be entered into the “focal_species_present” column. If the sound can only be identified to a higher taxonomic classification that includes the species for which the clip was classified (e.g., hummingbird sp., gull sp., sapsucker sp., woodpecker sp.), enter a “4” into the “focal_species_present” column. See table A2.1 for additional details.

5. In this example of a validation protocol, we wanted three positive detections of each species for each sampling week. We used the “positive_detection” column to keep track of positive detections for each species during a given week at a given station. After identifying three positive detections, the observer did not validate the remaining species detections for that sampling week at the station being reviewed. For each species, the observer entered a “1” into the “positive_detection” column upon identifying the first positive detection, a “2” for the second, and a “3” for the third.
6. If a “0” or a “2” was entered into the “focal_species_present” column because the observer was not 100 percent sure if the focal species is or is not present in the audio file, then a “1” should be entered into the “needs_further_review” column. A “0” should be entered into the “needs_further_review” column if the observer is confident in the assessment of the clip, including cases in which the observer entered a “3” into the “focal_species_present” column. See table A2.1 for more information. Subsequent reviews of detections can then be focused on the rows with a “1” in the “needs_further_review” column as needed.

Table A2.1—Review of all six possible combinations for values in the “focal_species_present” and “needs_further_review” columns and the definition of each combination

	Species	focal_species_present	needs_further_review	Definition
1	American Coot	1	0	The focal species is present and further review is not needed.
2	American Coot	0	0	The focal species is absent and further review is not needed.
3	American Coot	0	1	The focal species may be absent but further review is needed.
4	American Coot	2	1	The focal species may be present but further review is needed.
5	American Crow	3	0	The clip contains audio that is unclassifiable, and further review will not change the classification. A “3” in the “focal_species_present” column requires a justification in the notes column as to why the sound is unclassifiable.
6	Calliope Hummingbird	4	0	The clip contains a sound that cannot be identified to the species level, but can be identified as a higher taxonomic classification that includes the species detected. Use “4” for all woodpecker/sapsucker drums. Note that a detection should only be classified as “4” if there is a corresponding folder in the “diagnostic clips” folder (e.g., gull sp., hummingbird sp., sapsucker sp., woodpecker sp.). A “4” does not qualify as a positive detection and should not count toward the number of positive detections in a given week.

Note: The “Species” column shows the detected species. Reviewers enter values in the “focal_species_present” and “needs_further_review” columns based on their evaluation of the detection. There are six possible combinations for values in the “focal_species_present” and “needs_further_review” columns, and the definition of each is provided in the “Definition” column.

7. If BirdNET's identification is incorrect and the actual identity of the species is recognized, the four-letter code (The Institute for Bird Populations, n.d.) for of the true identity of the species is entered into the "true_label" column. The "true_label" column is left blank if the identification of the species in the recording is uncertain.
8. If the detection was classified as a positive detection, enter the label of the sound type from the species-specific folders (e.g., ct1, ct2) within the "diagnostic_clips" folder into the "call_type" column. If a call type is not currently included in a species' "diagnostic_clips" folder, enter "no_ct" into the "call_type" column. The clip can then be copied into a new folder for further review and consideration for inclusion in the species-specific folder within the "diagnostic_clips" folder.

Table A2.2—Codes used to describe miscellaneous background sounds that are not produced by birds

Code	Description
airplane	
bear	
canid	Dog, wolf, coyote
car	Car or truck driving
cow	
frogs	
gs	Gunshot
helicopter	
hm	Heavy machinery such as chainsaws, yarders
horn	Horn from automobile
human	Distant person talking or yelling
insect	
other	Unknown sound, describe if possible
rain	
rn	Rustling noises such as animals walking
sensor	Sound from autonomous recording unit glitching or malfunctioning, sensor sound
siren	
sm	Squirrels, chipmunks, pikas, or other small mammals
speech	Person talking close to recorder
thunder	
ts	Tree sounds, such as branches knocking or tree creaking
ung	Ungulates such as deer or elk
water	Creek or stream
wind	

Note: The codes should be entered into the "notes" column of the data sheet when the sounds in the "Description" column are present in clips. Blank cells in the description column accompany codes that are self-descriptive.

9. If there are sounds from multiple species of birds in the clip, enter a "1" in the "multi_species" column. If there are only sounds belonging to the focal species, nonavian sounds, or no sounds at all, enter a "0" in the "multi_species" column.
10. The final column on the spreadsheet is the "notes" column. This is a general purpose location where any notes of interest about the sounds in the audio file can be recorded. The codes described in table A2.2 can be entered to indicate the presence of specific sounds that are not produced by birds. There are intentionally few limitations to the type of information that can be recorded in the notes. For example, if an audio/image/MP4 file is missing from the folder containing detections to be reviewed, this can be indicated by writing "file missing" in the "notes" column. Distinct notes about a given audio clip and each sound type listed in table A2.2 that is present in a given audio clip should be separated by commas (e.g., cow, rain) to facilitate programmatic separation of independent notes about, or noises present in, that audio clip.

Appendix 3: Species List

Common and scientific names and four-letter codes of all species from the Blue Mountains that are mentioned in the report

Common name	Scientific name	Code
American goshawk	<i>Accipiter atricapillus</i>	AGOS
American kestrel	<i>Falco sparverius</i>	AMKE
American robin	<i>Turdus migratorius</i>	AMRO
Barred owl	<i>Strix varia</i>	BAOW
Black-backed woodpecker	<i>Picoides arcticus</i>	BBWO
Black-headed grosbeak	<i>Pheucticus melanocephalus</i>	BHGR
Brown creeper	<i>Certhia americana</i>	BRCR
Brown-headed cowbird	<i>Molothrus ater</i>	BHCO
Canada jay	<i>Perisoreus canadensis</i>	CAJA
Cassin's finch	<i>Haemorhous cassinii</i>	CAFI
Cassin's vireo	<i>Vireo cassinii</i>	CAVI
Cedar waxwing	<i>Bombycilla cedrorum</i>	CEDW
Chipping sparrow	<i>Spizella passerina</i>	CHSP
Clark's nutcracker	<i>Nucifraga columbiana</i>	CLNU
Common nighthawk	<i>Chordeiles minor</i>	CONI
Common poorwill	<i>Phalaenoptilus nuttallii</i>	COPO
Common raven	<i>Corvus corax</i>	CORA
Cooper's hawk	<i>Accipiter cooperii</i>	COHA
Dark-eyed junco	<i>Junco hyemalis</i>	DEJU
Dusky flycatcher	<i>Empidonax oberholseri</i>	DUFL
Evening grosbeak	<i>Coccothraustes vespertinus</i>	EVGR
Flammulated owl	<i>Psiloscops flammeolus</i>	FLOW
Golden-crowned kinglet	<i>Regulus satrapa</i>	GCKI
Gray flycatcher	<i>Empidonax wrightii</i>	GRFL
Great gray owl	<i>Strix nebulosa</i>	GGOW
Great horned owl	<i>Bubo virginianus</i>	GHOW
Hairy woodpecker	<i>Dryobates villosus</i>	HAWO
Hammond's flycatcher	<i>Empidonax hammondi</i>	HAFL
Hermit thrush	<i>Catharus guttatus</i>	HETH
Lazuli bunting	<i>Passerina amoena</i>	LAZB
Long-eared owl	<i>Asio otus</i>	LEOW
MacGillivray's warbler	<i>Geothlypis tolmiei</i>	MGWA
Mountain bluebird	<i>Sialia currucoides</i>	MOBL
Mountain chickadee	<i>Poecile gambeli</i>	MOCH
Mourning dove	<i>Zenaida macroura</i>	MODO
Northern flicker	<i>Colaptes auratus</i>	NOFL
Northern pygmy-owl	<i>Glaucidium gnoma</i>	NOPO

Common name	Scientific name	Code
Northern saw-whet owl	<i>Aegolius acadicus</i>	NSWO
Olive-sided flycatcher	<i>Contopus cooperi</i>	OSFL
Orange-crowned warbler	<i>Leiothlypis celata</i>	OCWA
Pileated woodpecker	<i>Dryocopus pileatus</i>	PIWO
Pine siskin	<i>Spinus pinus</i>	PISI
Pygmy nuthatch	<i>Sitta pygmaea</i>	PYNU
Red crossbill	<i>Loxia curvirostra</i>	RECR
Red-breasted nuthatch	<i>Sitta canadensis</i>	RBNU
Red-tailed hawk	<i>Buteo jamaicensis</i>	RTHA
Ruby-crowned kinglet	<i>Corthylio calendula</i>	RCKI
Steller's jay	<i>Cyanocitta stelleri</i>	STJA
Swainson's thrush	<i>Catharus ustulatus</i>	SWTH
Townsend's solitaire	<i>Myadestes townsendi</i>	TOSO
Townsend's warbler	<i>Setophaga townsendi</i>	TOWA
Vaux's swift	<i>Chaetura vauxi</i>	VASW
Warbling vireo	<i>Vireo gilvus</i>	WAVI
Western bluebird	<i>Sialia mexicana</i>	WEBL
Western flycatcher	<i>Empidonax difficilis</i>	WEFL
Western screech-owl	<i>Megascops kennicottii</i>	WESO
Western tanager	<i>Piranga ludoviciana</i>	WETA
Western wood-pewee	<i>Contopus sordidulus</i>	WEWP
White-breasted nuthatch	<i>Sitta carolinensis</i>	WBNU
White-headed woodpecker	<i>Dryobates albolarvatus</i>	WHWO
Wild turkey	<i>Meleagris gallopavo</i>	WITU
Williamson's sapsucker	<i>Sphyrapicus thyroideus</i>	WISA
Yellow-rumped warbler	<i>Setophaga coronata</i>	YRWA

Sources: Names are from the American Ornithological Society's *Checklist of North American Birds* (Chesser et al. 2024); four-letter codes are from The Institute for Bird Populations (n.d.)

Pacific Northwest Research Station

Website	https://research.fs.usda.gov/pnw
Telephone	(503) 808-2100
Publication requests	(503) 808-2138
FAX	(503) 808-2130
E-mail	sm.fs.pnw_pnwpubs@usda.gov
Mailing address	Publications Distribution Pacific Northwest Research Station USDA Forest Service 1220 SW 3rd Avenue Portland, OR 97204



Federal Recycling Program
Printed on Recycled Paper

Pacific Northwest Research Station
USDA Forest Service
1220 SW 3rd Ave., Suite 1400
Portland, OR 97204

Official Business
Penalty for Private Use, \$300

