Exploring Similarities Among Many Species Distributions

Scott Simmerman, Jingyuan Wang, James Osborne, Kimberly Shook and Jian Huang
University of Tennessee, Knoxville

William Godsoe
University of Canterbury, New Zealand

Theodore Simons
U.S. Geological Survey, North Carolina State University

ABSTRACT

Collecting species presence data and then building models to predict species distribution has been long practiced in the field of ecology for the purpose of improving our understanding of species relationships with each other and with the environment. Due to limitations of computing power as well as limited means of using modeling software on HPC facilities, past species distribution studies have been unable to fully explore diverse data sets. We build a system that can, for the first time to our knowledge, leverage HPC to support effective exploration of species similarities in distribution as well as their dependencies on common environmental conditions. Our system can also compute and reveal uncertainties in the modeling results enabling domain experts to make informed judgments about the data. Our work was motivated by and centered around data collection efforts within the Great Smoky Mountains National Park that date back to the 1940s. Our findings present new research opportunities in ecology and produce actionable field-work items for biodiversity management personnel to include in their planning of daily management activities.

Categories and Subject Descriptors

D.1.3 [**Software**]: Concurrent Programming—*Parallel Programming*; I.3.2 [**Computing Methodologies**]: Computer Graphics—*Graphics Systems - Remote Systems*

Keywords

HPC, Species Distribution Modeling, Parallel Processing

1. INTRODUCTION

A major challenge in biology is to manage the vast diversity of living things found across the earth. Most geographic regions contain a large number of distinct species, and many species are poorly known. Given this uncertainty, an important pre-requisite for any biodiversity management plan is to predict the geographic distributions of individual species and to summarize this knowledge across many species. To address these problems scientists have created two ever-growing sources of data: (1) records on the occurrences of individual species and (2) geo-referenced datasets on environmental conditions. Using these data, ecologists can leverage machine learning algorithms or statistical models to estimate the probability of presence for individual species. These models—hereafter Species Distribution Models or SDMs—constitute a simple scalable approach to use limited information on the ecology of a species to predict its geographic distribution. Unfortunately ecologists still lack the tools to effectively compare, contrast, and explore trends and relationships in the large number of SDMs that are currently being generated.

To fill this gap we develop tools for summarizing information on the relationship between individual SDMs as well as analyzing the uncertainty in the models. These tools play a crucial role for meeting time-critical management needs of national stewardship agencies with missions such as ensuring long-term security of food production and detecting early stages of irreversible damages at ecosystem-scale.

A scalable data analytics system is needed for such a challenging application. In such a system, the source of data makes no fundamental difference. The data could be collected by experts in the field, as with the ATBI, or by citizens as with other recent data collection projects such as eBird and Natureserve[1]. In this work, we focus on the ATBI data for the following reasons: (1) proximity to the domain experts and final users, (2) availability of both species inventory data and also high-quality geo-referenced environmental data, and (3) availability of a production-quality modeling method, maximum entropy, whose results are acceptable to the bio-inventory stewards at the National Park Service (NPS) and US Geological Survey (USGS).

Our system uses high performance computing (HPC) and has the following features: (1) reliable employment of SDM models to construct species distribution predictions across a large canvas of species, (2) an ability to test, record and analyze varying environmental responses in each individual model, (3) visualization of data quality in support of user exploration, (4) on-demand hierarchical grouping of species by similar distribution as well as similar environmental responses. We enable a user community unfamiliar with HPC

[1] http://ebird.org, http://www.natureserve.org

XSEDE12 2012 Chicago, IL USA

to greatly increase their productivity using their own legacy code, for example burning through 90 CPU hours in just 25 minutes (see Table 2).

With these results, we also enable the testing of hypotheses fundamental to developing management plans, such as: When managing a specific aspect of an environment, which species are the most affected? For a species central to a management goal, do we already have sufficient, quality data? If not, which parts of the data are the most trustworthy? To our knowledge, no systems exist to serve these needs, and especially not on a scale as large as that of GSMNP.

2. BACKGROUND

2.1 Species Distribution Modeling

Our system explores one example of an extensive data set on species occurrences, the All Taxa Biodiversity Inventory (ATBI[2]) of The Great Smoky Mountains National Park (GSMNP). The ATBI project began in 1998 as an effort to document all forms of life in the Park [8]. To date, the inventory has documented 7,391 new species to the Park and 922 species new to science. As with many projects that collect data about the environment, the raw data will always be too sparse to capture the full details of the subjects and the targeted environment. Therefore, it is imperative to use further modeling to make well-informed predictions of species distribution, allowing for more comprehensive study and development of actionable management plans.

GSMNP spans 2,200 km^2 and represents an iconic system to compare observations on biodiversity. The Park is one of the most biologically diverse regions in North America. For this reason it is both an International Biosphere Reserve and a World Heritage Site. This region has been a focus of diversity research for decades notably through Robert H. Whittaker's pioneering work in the 1940's. In a series of papers, Whittaker argued that many species specialize in locations with similar elevations [14] and that these responses to elevation varied dramatically from one species to another.

GSMNP management has used these iconic features to pioneer creating and maintaining scientifically sound biodiversity inventories. In their latest ATBI database [8], the Park has amassed data for over 17,000 species in the Park. In addition to the geo-referenced species occurrence data, the Park has collected environmental data such as geology, soil, terrain, vegetation and climate which enables the modeling of species distributions across many disparate taxa.

Since Whittaker's time researchers have developed several methods to model species' distributions. Many of these can be thought of as estimates of the probability that a species is present. SDM methods can be divided into those for presence-only data vs. those for presence as well as absence data. Our data is presence-only, and in this category there are still many modeling options such as envelope similarity models, ENFA (ecological niche factor analysis), GARP (genetic algorithms for rule-set prediction), logistic regression models, spatial point-process models and MaxEnt (maximum entropy) models.

In general, predictions made by different SDM methods will produce quite different patterns. Our application scientists' favorite method is MaxEnt, which works by constraining the estimated probability of presence to resemble the observed probability of presence. To do so, MaxEnt constructs a model consisting of a number of features such as descriptions of how environmental variables might affect the probability of presence. It then performs model optimization to minimize the information contained in the residuals. MaxEnt assumes that presence-only samples are random with respect to the species of interest, and that covariates of species occurrence probabilities are distinct and independently distributed from covariates of species detection probabilities [11].

In this work, we direct our primary attention to the input, output and important controlling arguments of the models, and treat the internals of the modeling process as a "black box". In the following we make no further distinction between SDM methods because our data analytics system is generally applicable to all models.

2.2 Complexity of Data Analysis and Visualization

Many ecologists have studied the utility of SDM models. However, these efforts still do not allow for a comprehensive understanding of the choices necessary to make useful SDMs. The current limit is our ability to understand the consequences of the numerous input choices available to apply machine learning algorithms. Some choices are easy to interpret such as the particular combination of variables included as model input. Others are much more abstract such as regularization parameters that penalize overly complex models and threshold values that distinguish predicted presences from absences. Moreover these factors may interact in unexpected ways making it important to consider the joint influence of each variable. There is a pressing need for appropriate ways to compare models and effective visualization methods augmented by computation power to rigorously investigate a large parameter space.

Though SDMs are widely applied, the sources of error inherent in such analyses are poorly understood. Over the last few years a tremendous amount of environmental data has become available for analyzing species distributions in the form of interpolated maps with environmental measurements. However, the best way to use this ever-growing bank of data is unclear, as ecologists face two seemingly contradictory problems. First, species distributions are an amalgam of many facets of the environment and these facets interact non-linearly. For the foreseeable future we will not have measurements of many of these processes and as a result we must interpret SDMs that fail to incorporate much of what we know about organisms [3]. Second, we have only a finite amount of data on the distribution of any one species. As a result, there is a serious danger of generating overly complex models that do not generalize well. Biodiversity management will remain extremely challenging until tools become available for visualizing uncertainties due to the internals of a model and the quality of input data.

The conclusions from SDMs are also susceptible to sampling error. When fitting complex models with multiple interacting sources of uncertainty, it is difficult to predict how sensitive conclusions will be due to changes in the data used to run the analyses. For this reason it is important to determine whether biological conclusions are robust to changes in sampling protocols.

The Park's natural resource managers need to visualize and understand advanced biological analyses of virtually all

[2] http://www.dlia.org/atbi, http://www.nps.gov/grsm

of the Park's species, and, unfortunately, this capability is currently lacking. The ATBI project has been in existence since 1998. No other protected reserve possesses such a comprehensive set of biodiversity data (though there are special purpose data repositories at similar scales for birds, for example). The resource management and science staff at GSMNP need a synthesis of these data to be able to answer critical conservation questions, such as: What is the distribution of species in the Park? Where are the most species-rich sites? What natural factors in the environment affect these distributions? Where do un-natural stressors likely have the greatest impact on rare or vulnerable species and biodiversity in general? These questions are becoming more critical every day because of the wide variety in the Park's natural diversity, non-native invasive species, air pollutants, habitat loss, and possibly rapid climatic changes.

We must note that other leading researchers have reported recent success on using analytics for other pressing applications, most notably, a powerful birdVis [2] system for the eBird project, analytics for epidemic monitoring [1] and analytics for maritime resource allocation [7]. We find the primary differences between our work and theirs to be different application needs. Our needs use HPC to take SDM-based studies from tens of species at a time to now hundreds or thousands at a time, and from previously using environmental data at 1 kilometer resolution to now 30-meter resolution. In addition, we work on visualizing similarities between hierarchically organized groups as well as the uncertainties involved, but do not focus on visualizing events, especially the temporal aspects of events as in [1, 2, 7]. Our ultimate goal is to create a system for production use by non-technical end users such as bioinventory managers and park rangers. As future consumers of HPC technology, these users in general lack the background or the desire to understand the technical intricacies. Instead, their daily job function involves problem-solving in their own domain. In this respect, they differ from the researchers who are typically the main users of scientific gateways which provide a set of tools allowing remote computation-intensive study without geographic limitations [15]. Although we realize this fundamental difference, we do notice that our cyberinfrastructure requirements such as central data repositories and data analytic capabilities, are similar to other gateway efforts such as geospatial modeling gateways [5], climate modeling environments [16], and the iPlant Collaborative [6].

3. ADDRESSING LARGE PROBLEMS USING HPC

Software tools for biologists and ecologists are just recently coming to terms with very large data and parameter spaces that call for parallel processing with HPC. With current modeling methods which seek to quantify and predict species habitat, computing power is often a major limitation. The work by Webster et.al in 2008 [13] represents the state of the art in this popular research direction.

Our collaboration with biologists and ecologists using MaxEnt for species modeling is just one example of a common occurrence where a popular "black box" software tool needs to scale up to address a larger problem size. The challenge is to develop quick, simple ways for new users to begin this scaling process in an HPC environment, using existing, familiar tools. Just leveraging computing power, however, is only the start of the solution. Interpreting the large volumes of computation results and creating useful and semantically meaningful abstractions to describe groups of coexisting species presents a challenge of its own.

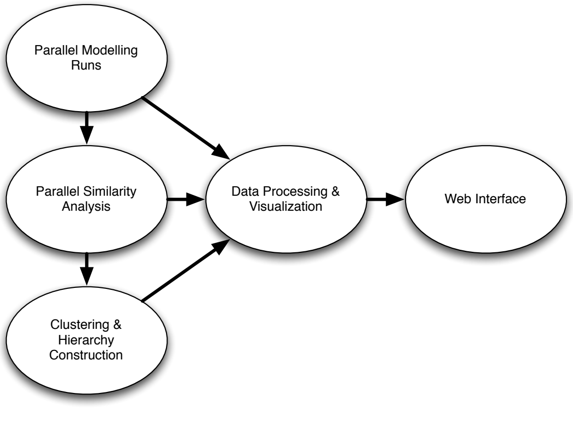

Parallel Modelling Runs
Parallel Similarity Analysis
Clustering & Hierarchy Construction
Data Processing & Visualization
Web Interface

Figure 1: Overall workflow of our system incorporating HPC in species distribution modeling.

Bedrock geology	categorical
Digital elevation model	continuous
Leaf on canopy cover	categorical
Slope measured in degrees	categorical
Solar radiation data	continuous
Soil organic type	categorical
Terrain shape index	continuous
Topographic convergence index	continuous
Understory density classes	categorical
Vegetation classes	categorical

Table 1: Ten environmental layers provided by the GSMNP.

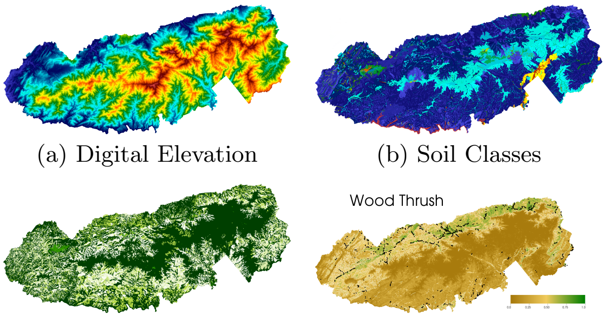
We address these challenges in our overall workflow as shown in Figure 1. On the left side of the workflow diagram, the components include parallel orchestration of serial SDM models for scalability, parallel similarity analysis, and on-the-fly hierarchical abstraction creation through on-demand hierarchical clustering. The results are then amalgamated through parallel data processing and visualization and subsequently are delivered through a fully interactive web interface to stakeholders that include Park natural resource managers as well as ecology researchers.

3.1 The Data

The current standard for modeling species distributions is to use coarse resolution (∼1 kilometer) environmental layers when generating models [4]. Superseding the breadth and depth of this standard, GSMNP provided us with ten environmental layers uniformly set on a common grid at 30-meter resolution (Fig. 2).

These environmental layers include abiotic factors, such as elevation (Fig. 2a) and soil types (Fig. 2b), as well as biotic layers on vegetation cover (Fig. 2c). These data are all scalar variables on a common grid, and each environmental layer can be either a continuous or a categorical variable (Table 1). Each layer amounts to 15–30MB in raw format.

To ensure high quality SDMs we restrict our attention to



(a) Digital Elevation (b) Soil Classes

(c) Vegetation Classes (d) Sample SDM result

Figure 2: Environmental layers and SDM example from the GSMNP.

species with 30 or more distinct records. Each record contains GPS coordinates, species sighted, time of sighting as well as other metadata, and can be geo-referenced with the environmental layers. Since the initial focus on ATBI was heavily directed towards finding rare, special, unique and previously unknown species, our selection criteria trimmed down the species count to around 500, about half of which comprise bird and tree species in the Park.

3.2 Modeling With HPC

Among the many alternative SDM methods for presence-only data, maximum entropy is well understood [11] and has recently gained much popularity, to no small extent due to the availability of a production quality software that implements the method. This software is MaxEnt[3] [10], written in Java. The inputs to MaxEnt consist of species records (coordinates of observed locations) plus environmental layer data. To control the process of fitting a model, there are several parameters such as convergence threshold, maximum iterations, regularization value (β), selection of included features, and selection of environment layer input.

The MaxEnt software is only available as an executable jarfile; it is not open source but free for academic and research use. Originally designed for personal computer platforms, MaxEnt employs inherently serial steps, although Java threads are used when convenient. There are a few key issues to address before we can use this "black box" code in an HPC environment.

First of all, large-scale runs of MaxEnt constitute an atypical HPC usage pattern which calls for a tool to aid in scheduling. Most codes on an HPC platform are designed to run in parallel either using message passing (MPI) or threads (OpenMP or pthreads). It is far from practical to require that a proven machine learning SDM model be rewritten to operate with great internal parallel scalability. We have to explore ways to achieve scalability external to these black box models. From this respect, this need is general to almost all fields that use complex models to interpret data and make predictions.

For this project we developed a tool called Eden which achieves coarse-grained parallelism by managing concurrent runs of serial code. Eden is simply a master-worker framework that allows the user to submit a list of commands to run on a given number of cores. The problem it solves is quite general—run a list of N single-processor jobs on P processors, where N is far greater than P. The single-process jobs can have highly varying running time, can fail due to an array of reasons such as internal bugs, saturating disk I/O channels or saturating runtime limits of concurrent threads.

After exploring several existing tools for managing such concurrency, we found that most did not fit our specific need. Tools such as Swift [4], the Nimrod toolkit [5], Condor [6], and Hadoop [7] are all comprehensive tools designed for running concurrent jobs across many heterogeneous computing resources. In our case, we need to run many small serial jobs on a single system and make this as easy as possible for the user. Therefore we wanted to avoid any added burden to the user such as learning a new language (as with Swift) or requiring installations at the system administrator level (as with Condor). Other barriers to using these tools for our project include runtime errors and lack of support for the Nimrod toolkit, and the dependence on Java for Hadoop. We also explored GNU's Parallel utility [8] but found its command line interface daunting for inexperienced users.

Eden is light-weight and operates entirely at user level, running with very low privileges and definitely not as an admin or daemon process. This is for security and stability reasons for large-scale systems with high concurrencies. Totalling only 450 lines of code, Eden is entirely script-based and requires no compilation or building process. Offered as an open source project [9] since July 2011, Eden works with typical job queuing systems as on Nautilus as well as with any generic multi-processor system or fat node without a job scheduler. For a user wishing to scale their code, Eden provides a two-step process: first trying Eden on their own multiprocessor system (e.g. a desktop machine) and working out any issues with concurrent runs; then using Eden with the same simple interface to scale their code on Nautilus.

Eden's architecture is shown in Figure 3. It uses mostly Bourne shell with a few tcl scripts to handle socket communication. To use Eden, the user must first provide a job list and specify the number of processors to use (P). When Eden is launched, the initial `eden.sh` script configures the run and then launches `adam.sh`, typically on the login/master node. The `adam.sh` script launches P worker processes (`abel.sh`), one on each of the P processors. On a typical PBS queuing system, this involves creating a PBS script and submitting it using the `qsub` command. The `abel.sh` processes then each retrieve a job from `adam.sh` via a socket connection, execute the job, and then request a new job, continuing until the command list is exhausted. For each command completed, stderr, stdout, and timing information are written to individual files on disk. Finally, the `eve.sh` script cleans up and distills these files into a single summary file.

Once we have a way to efficiently run many MaxEnt instances, we next have to deal with I/O issues. MaxEnt by default outputs images of the species distribution maps as well as various other charts and figures relating to the statistical details of the model. This could result in 10 to 12 output files for a single MaxEnt run. When running hun-

[3] http://www.cs.princeton.edu/ schapire/maxent

[4] http://www.ci.uchicago.edu/swift/main/

[5] http://www.messagelab.monash.edu.au/Nimrod

[6] http://research.cs.wisc.edu/condor/

[7] http://hadoop.apache.org/

[8] http://www.gnu.org/software/parallel/

[9] http://sourceforge.net/projects/rdaveden

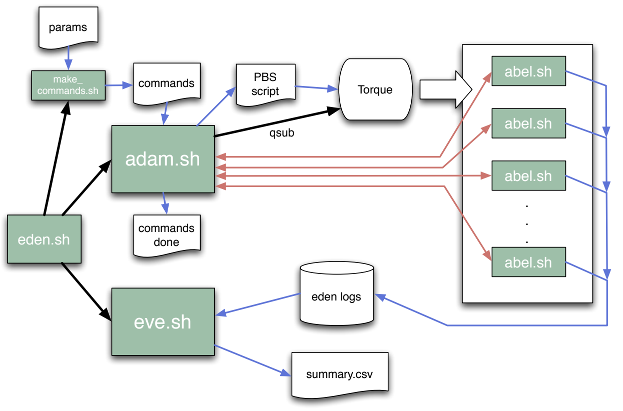


Figure 3: The Eden framework for managing concurrent modeling runs.

dreds or thousands of runs concurrently, this deluge of files would easily overwhelm the file system. By judicious use of command line flags, we restricted the MaxEnt output to the minimum allowable, seven files per run, with the most substantial being the SDM overlaid on a raw ASCII grid of 30 meter resolution. But even with these improvements, we generally had to limit the number of concurrent runs to prevent overwhelming the file system.

Of course, by forfeiting the output of images from MaxEnt, we had need of other means of rendering the SDM. We chose to use a popular scientific data visualization tool VisIt to produce custom maps of the models. VisIt can easily be scripted using Python and run in parallel through a batch system, optimizing the production of these images from the raw SDM data. In our rendering, the black dots indicate the original observation points, dark brown indicates very low probability of presence and green represents high probability of presence. Also, a white contour line marks the threshold value suggested by the MaxEnt model that separates presence prediction from absence prediction. A sample of the rendered results is shown in Fig. 2d.

Finally, we found the need to replace built-in capabilities with custom procedures to increase parallelism. MaxEnt provides a cross-validation capability by performing multiple runs for a particular species, with each run holding back a different subset of samples for testing. This allows for evaluation of the fit of the model. Using MaxEnt's cross-validation feature, unfortunately, led to much longer run times per species for a couple of reasons: MaxEnt has no inherent parallelism (so, for instance, a 10-way cross-validation is equivalent to doing 10 serial runs), and then several summary results are serially created by default. To optimize the opportunities for parallelism, we externally orchestrated cross-validation runs, which involved creating subsets of the sample data, doing independent MaxEnt runs, and making summary plots with our own custom C code.

3.3 Similarity Analysis with HPC

Newer and faster capabilities to handle large SDM runs will not remedy the need for carefully filtering the resulting data and for analyses of interpretational pitfalls. It is always up to the expert users to use new capabilities properly to advance our understanding of SDM output and conservation implications. For this purpose, we use similarity metrics as an indirect way to measure data quality by comparing cross-validation results and also as a direct way to construct groups of species with known relationships to establish basic context and support and guide further exploration.

With SDM models, there are several ways to measure similarity: (1) Sørensen Index, (2) Jaccard Index, and (3) χ^2 Index. Our expert users chose the Sørensen similarity index (SSI) which, when comparing the SDMs of two species, is calculated by the fraction:

$$\frac{2a}{2a+b+c}$$

where a is the number of shared presence locations, b is the number of presence locations for only species 1, and c is the number of presence locations for only species 2. The SSI value will be between 0.0 and 1.0 with 1.0 indicating an exact match. Other measures are plausible, notably Jaccard and the squared distance between observed and expected counts [9]. However, many biological questions correspond to determining if two species will encounter one another, and so we have focused our analyses on Sørensen's similarity.

We use SSI to compare cross-validation runs of a single species and also to compare all possible pairs of species SDMs. Figure 4D shows an example of a cross-validation similarity matrix computed using SSI as the metric. The matrix shows similarity between models from 10 cross-validation runs using ATBI data of the Cedar Waxwing (*Bombycilla cedrorum*). A traditional "hot-cold" color map is used, where red indicates an SSI of 1.0 (exactly similar) and blue indicates and SSI of 0.0 (no commonality). We can see from this example that the cross-validation results moderately vary from one another due to the orange and yellow colors.

3.4 Visualization - MDS and Hierarchical Clustering

Once we have SSI information on how every species compares to every other species we naturally want to see how similar species form clusters. One way to show clustering is through multi-dimensional scaling (MDS). MDS is a method of producing a mapping of objects where the relative positions of the objects provide a graphical representation of their interrelationship. In our case, the objects are SDMs and their interrelationship is the degree of similarity based on the SSI metric. The 2D MDS plot provides a view of similarities among all species in one space. We used the Vegan package in R to perform the MDS mappings. The MDS plot appears in the lower left corner of the interface (Figure 4E). The current species is indicated by a red square and the other points are colored according to the SSI bins.

Hierarchical clustering is another way to examine clusters of similar species. When comparing between two species, there are potentially three metrics.

Metric 1: The pair-wise SSI metric that distinctly states how similar the two species are distributed.

Metric 2: The environmental responses. Together with each MaxEnt prediction, a feature vector is also provided to show the weight given to each environmental layer. In our case with 10 environmental layers, the environmental response vector is 1×10. Exploring this vector allows users to hypothesize about the environment's role in modeling a particular distribution.

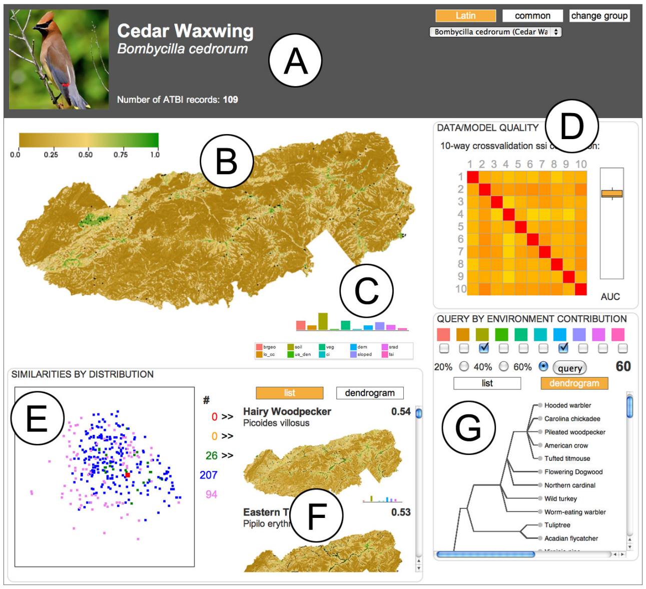


Figure 4: The web interface.

Metric 3: Similar coexisting pattern with the species across all taxa. Given a particular species and a total of S species in the study, we can form a $1 \times S$ vector where each element is the SSI value between the current species and one other species from the study. By comparing these vectors among species, scientific users can hypothesize about different kinds of species coexistence.

Users can decide, on the fly, which metric to use to form the feature vector. In the subsequent hierarchical clustering step we cluster by Euclidean distance between the feature vectors. We start the clustering process with each species being treated as a single cluster. We recursively cluster together species that are similar (i.e. feature vectors within some threshold distance). The threshold value increases at each level so that clusters are gradually merged together. The stopping criteria is when no new clusters can be found within some predetermined threshold distance. A virtual root node is always added in the last step so that the forest of cluster hierarchies becomes a tree that is easier to manage.

To ensure biologically meaningful results, we add an additional stopping criteria—no clusters can have an average of pair-wise SSI below a given threshold. That is, all of the species within a cluster should demonstrate basic similarity as a group. This threshold is typically set at a very loose SSI value of 0.2.

As part of our interface, we visualize cluster hierarchies as dendrograms and aim to show the detailed merging of hierarchies among the clusters. In the dendrogram layout, edges simply mean a parent-child relationship in the tree hierarchy. Users can choose to see the current grouping of similar species as a list of SDM maps or as a dendrogram.

3.5 Remote Visualization - Web Interface

Modeling and subsequent analysis results are accessed via a web browser interface. The interface provides immediate access to all results from the MaxEnt modeling runs allowing the user to visually compare SDMs as well as explore clustering results. The stakeholders involved in the mission of data-driven natural resource management are geographically dispersed, and it is difficult to deploy highly synchronous visualizations to managers, scientists, and team leads in the field. Also, it is unique to this field that a large number of non-profit groups are collaborating on this mission, either with each other or directly with the national parks. Given such a dynamic environment, we opted to use asynchronous remote visualization and utilize a web browser as the delivery tool on the user end. Dynamic information visualization is still implemented on the user end, but only after the data has been heavily reduced to a small set (e.g. less than 100 species). Our working website is located at http://seelab.eecs.utk.edu/alltaxa. It is recommended for use with Safari, Chrome, or Firefox browsers and has known issues with Internet Explorer.

When accessing the website, the user first chooses a particular group to explore. Currently, we have species divided into 10 taxonomic groups suggested by our expert users. The user is then presented the screen shown in Figure 4.

The sections of the web interface are labeled as follows:

A species picture, scientific name and common name, and total number of distinct records in ATBI

B plot of SDM model result – this is an average taken over 10 cross-validation runs

C bar chart of environment layer contributions, again averaged over 10 runs

D data quality metrics including a matrix comparing SSI values over 10 cross-validation runs and a Tukey boxplot of AUC scores for the 10 runs.

E multidimensional scaling (MDS) plot showing species similarity (Metric 1) over entire group

F ability to view different groups of species based on distribution similarity, either by list or dendrogram (Metric 1 or 3)

G ability to query for similar environment response (Metric 2) and view results as list or dendrogram

For each individual species, the user can view the overall results from a 10-way cross-validation MaxEnt run. The SDM is shown as a map (Fig. 4B), averaged over the 10 runs, where dark brown indicates low probability of presence and dark green indicates high probability. The black dots show the original sample points. Environment contributions are shown in the bar chart below the map (Fig. 4C).

Information relating to data and model quality for an individual species is shown in the top right panel (Fig. 4D). Here, the user can get a sense of the outcome from the 10-way cross-validation runs by viewing an SSI matrix relating to the current species. If the 10×10 matrix contains mostly red, that indicates that there was very little variability between the 10 SDMs—meaning a consistent model was produced no matter what subset of the data was set aside for testing. On the other hand, lots of variability in the matrix with colors ranging toward orange and yellow indicates inconsistency across the 10 cross-validation runs and might indicate a data quality issue. For example, in Fig. 4D, cross-validation run 4 stands out as having lower SSI values compared with the others. This indicates that the 10% of the data withheld in that run has a substantial leverage on the results and calls for closer inspection.

To the right of the SSI matrix (still in Fig. 4D), the AUC side bar shows a Tukey plot. AUC is the "area under curve" metric given by MaxEnt for measuring quality of fit of the model. The Tukey plot shows min, max, median, and the bottom and top quartile of test AUC scores for the 10 runs. The box surrounding the Tukey plot indicates the full dynamic range of AUC scores, 0.0 to 1.0.

Besides viewing modeling information for a single species, the interface provides three ways to examine similarities among species through panels E, F and G in Fig. 4. Panel E shows the 2D MDS plot showing overall species similarities. The current species is indicated by a red square a little larger than the surrounding squares. Individual points are colored corresponding to the partitioning of the SSI values into five bins. The topmost bin, colored red, denotes SSI values between 0.8 and 1.0. The other bins span intervals of 0.2 in descending order of orange, green, blue and pink. The number of species falling into each bin is displayed on the right of panel E and is clickable. In the example of Fig. 4, the top two bins (SSI of 0.8–1.0 and 0.6–0.8) are empty, and 26 species have an SSI between 0.4 and 0.6 compared with the current species of Cedar Waxwing (*Bombycilla cedrorum*). The most similar species, the Hairy Woodpecker (*Picoides villosus*), appears at the top of the list with an SSI of 0.54.

Panel G (Fig. 4G) allows the user to perform queries based on environmental responses. Users can ask, for example, "Which species models have elevation and vegetation contributing at least 60 percent to the model?" Any combination of environment layers can be chosen along with three options for a contribution threshold. Results are shown as either a sorted list or dynamically generated dendrogram. Fig. 4G shows the dendrogram view.

4. PERFORMANCE AND SCALABILITY

For our computation and analysis, we used the supercomputer Nautilus at the NSF Center for Remote Data Analysis and Visualization (RDAV) at the National Institute for Computational Sciences, University of Tennessee. Nautilus[10] is an SGI Altix 1000 machine with 1024 cores and 4 TB of global shared memory in a single system image. Because of having a single operating system image running on a large number of cores, Nautilus is particularly well-suited to such atypical HPC jobs as running hundreds of instances of serial code like MaxEnt. Nautilus allows the use of shared libraries so that languages such as Python and Java can be used on the entire machine. Often, large capability HPC systems such as Kraken or Jaguar do not allow these types of languages due to restrictions of the light-weight OS instances running on compute nodes.

Table 2 lists timings for the various workflow components of a representative run involving 530 species. Using the Eden tool to manage concurrent runs on 256 cores, we were able to perform 10-way cross-validation MaxEnt runs on 530 species (totaling 5300 individual runs) in 25 minutes wall time. These runs included converting the MaxEnt output from ASCII to binary. Each MaxEnt run takes, on average, 61 seconds, and ranges between 37–147 seconds. Summing up the wall clock times of each individual model amounts to 90 hours of CPU time. This corresponds to a parallel utilization of 84.4%. We also experimented with concurrently running two Edens using 128 cores each to perform half of the SDM runs. The total running time is no different from using a single Eden run with 256 cores.

The input for the runs consists of species sample files and environmental layer data, both totaling less than 80 MB after compression. Our Eden run automatically decompresses data before feeding data into MaxEnt. This design has greatly reduced the input part of the I/O bottleneck. The output from 5300 MaxEnt runs totals to 167 GB in 37100 files. We convert the ASCII output grids to binary, reducing the overall file output to 90 GB. Due to this large amount of I/O from the runs and the limitation of the filesystem on Nautilus, we were limited to using only a small portion of the machine, usually 256 cores, to reduce the number of concurrent reads and writes.

Other steps in our overall workflow shown in Table 2 involved using Eden to run scripts in parallel for such things as data pre-processing (wall time of 3 minutes), rendering of SDM maps using VisIt with Python (wall time of 4 minutes), and aggregating cross-validation results and summary statistics (wall time of 3 minutes). For performing pair-wise comparisons on the SDMs, we developed custom C code using pthreads. Nautilus was particularly well-suited for this task because of its large global shared memory, allowing all SDMs to be resident in memory at the same time for comparisons. Our code could produce an SSI matrix for 530 maps (139,920 comparisons of 2899×1302 grids) in just 26 seconds wall time using 256 cores on Nautilus. With all of the miscellaneous runs, most of the wall clock time was spent in reading data. For future studies on continuously updated data of 8,000–10,000 species, we will have to employ advanced techniques in parallel I/O. However, because the data is continuously updated by the data collectors, it is very unlikely to be in a format ideal for parallel I/O. Precomputing to reorganize the input data will be unavoidable.

Finally, hierarchical clustering on 530 species takes less than 3 seconds. Dendrograms are computed on the fly using Javascript in the web browser. Depending on how many species are chosen in a query, these functionalities can range from highly interactive to having a lag of a few seconds.

We considered three kinds of scalability during our design stage: in terms of input data, in terms of model analysis, and in terms of access to the analysis. Not all are related to computing power and the priorities would vary significantly for different management purposes. For the scope of this work, which is very basic compared to that of an overall long-term project in support of natural resource management by the National Park Service, we could only address the scalability of model analysis and the access to the analysis.

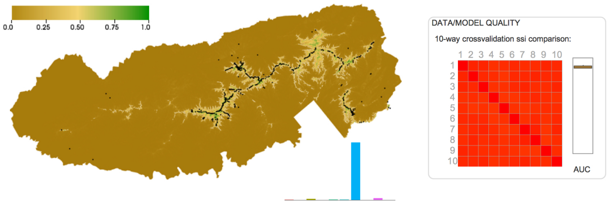
In orchestrating large scale modeling, our system is stable and the running time increases linearly as species are added. Assuming that a monthly run schedule would be sufficient and the set of species will number around 10,000 , our modeling running time (including cross-validation) would be approximately 7.8 hours, which is acceptable. The SSI similarity analysis requires comparing every species pair which leads to a big-O squared complexity. This corresponds to a running time of $\sim$ 3 hours for comparing 10,000 species.

Our Eden framework is a general purpose tool and has proven to be beneficial for several other science projects on Nautilus including tornado simulation analysis (Java and Matlab), building energy efficiency modeling (Python and

[10] http://rdav.nics.tennessee.edu/nautilus

	CPU Time *hh:mm:ss*	Wall Time *hh:mm:ss*	Type of Parallelism	Num. Cores	Parallel Utilization (%)
Pre-processing Species Records	00:19:05	00:03:17	scripts via Eden	64	9.1
MaxEnt Runs	90:01:00	00:25:23	legacy code MaxEnt via Eden	256	84.4
Aggregating Cross-validation	01:21:30	00:02:48	scripts/C code via Eden	64	45.5
VisIt Rendering of SDMs	01:46:00	00:04:27	VisIt and Python via Eden	128	18.6
SDM Comparisons	01:15:42	00:00:26	pthreads	256	68.2
Hierarchical Clustering	00:00:03	00:00:03	–	1	–

Table 2: CPU time incurred by each workflow component vs. wall-clock time with parallel acceleration.



(a) Distribution Prediction

(b) Cross Validation

Figure 5: (a) SDM distribution prediction for Fraser Fir and the environment responses, and (b) the corresponding cross validation result.

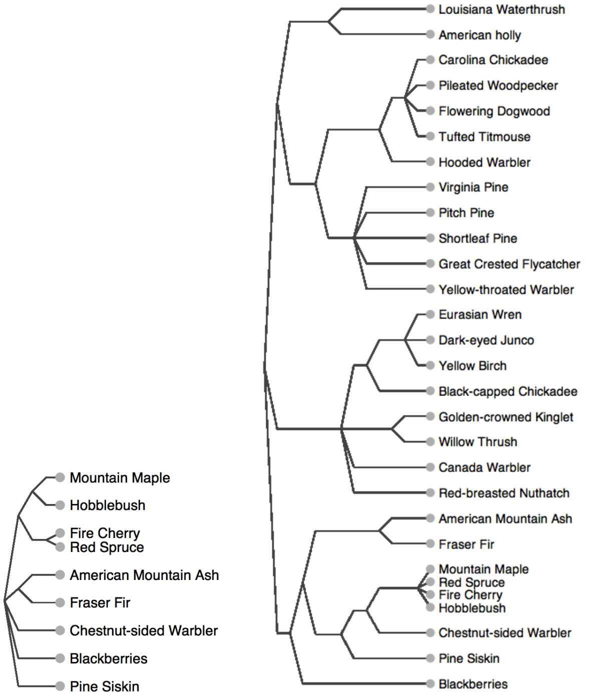
Matlab), and ecology parameter sweep studies (Python and R). Using Eden, four different projects on Nautilus have burned at least 42798 CPU hours over the past year [12].

5. SAMPLE USER EXPLORATION PROCESS

We illustrate the utility of our visualization approach using an example of a species of conservation concern in GSMNP. Fraser fir (*Abies fraseri*) is a tree species only found in mountains in the southeastern United States. There are extensive populations of this species within GSMNP, however over the last few years it has suffered population declines due to attack by the Balsam wooly adelgid (*Adelges piceae*).

When faced with a declining species, one of the first problems is to understand the extent of its geographic distribution. As an example, we query Fraser fir as part of the overall subset of species including bird and plant taxonomic groups, as we plan to subsequently compare this species to other well-sampled taxa. We select "Fraser fir" from the interface species menu and find that this species has been extensively sampled with 358 records. By examining the SDM (Figure 5a) we note that most of these records (black dots) correspond to a narrow band in the eastern portion of the park. The SDM for this species predicts a high probability of presence in this region of the park (beige and green portions of the map) and a low probability of presence (dark brown shading) in other regions.

After noting that our SDM predicts presence in a narrow band of habitat it is useful to determine our confidence in this conclusion. Examining the 10-way cross-validation results, we note that there is substantive agreement between each of the individual runs (Figure 5b). To see this, note that within the SSI matrix most of the individual squares are bright red or orange indicating very high similarity between runs. The Tukey plot to the right of the matrix indicates that the median AUC score from these runs was quite high and that there is little variation among the 10 runs. These results reinforce our conclusion that Fraser fir is restricted to a narrow band of habitat within GSMNP.



(a) by Distribution

(b) by Environment Response

Figure 6: Dendrograms that recursively group species by similar predicted presence vs. similar environmental influences.

Next we explore the community of species with similar distributions, both to identify species that may be susceptible to similar threats and to find species likely to respond to changes in the population size of Fraser fir. We select the list option to find species with similar SSI scores. We first set the threshold to SSI of 0.8, indicating the top most similar species, and find the American Mountain Ash, another tree species with a similarly restricted distribution. When we expand our list out by lowering the SSI threshold to 0.6, we get a list of seven additional species with similar distributions. We then switch to the dendrogram view, which is shown in Figure 6a. The dendrogram includes Fraser Fir, the current species, and also 8 species similar to it with SSI greater than 0.6. In this list we see Red Spruce, another tree susceptible to the same insect.

Finally, we may investigate the environmental variables which predict the distribution of Fraser fir. Note the bar chart to the lower right of the map listing the percent contribution of each environmental variable to the SDM for Fraser fir (Figure 5a). This chart shows that the DEM (Digital Elevation Model) is the highest contributor. We can query this information to find species that are likewise best modeled by elevation. To do this we click on the tick box below the blue square and the "60%" option to query for SDMs in which DEM made at least a 60% contribution. From the list option we obtain a number of species. We may then switch to the dendrogram view to obtain groups of species best predicted by elevation, with Fraser fir and similar species constituting a cluster at the bottom of our graph (Figure 6b).

Together, this information gives a concise summary on the current state of knowledge on the Fraser fir (*Abies fraseri*), a species of concern. We know that this species has been well sampled but that it has a comparatively narrow distribution within the park. A number of other species have similarly restricted distributions including some that are susceptible to the same environmental threats. Also, the comparatively narrow distribution of this species is best predicted by the elevation of a location.

The above describes the process that our ecology co-author used to study the Fraser fir, a species iconic to the southeast US that is currently in danger. This system has already been provided to the park managers, but so far it has not been put into use for management purposes.

6. CONCLUDING REMARKS

This paper reports early progress on a long-term project for supporting the decision making process of natural resource managers. Their needs include finding correlations as predicted by current data, understanding the uncertainties with the predictions, devising plans to improve data or performing small scale field work to verify predictions, designing and carrying out intervention plans, and continuing monitoring of results. This process is cyclic and comes with time pressure of many kinds. Our work is in essence only a pilot study on the way to meeting these goals. Our future plans include extending the web interface capabilities so that student users as well as researchers and biodiversity managers can more efficiently conduct their own studies and experiments based on sophisticated modeling using HPC resources. The interface will allow non-technical users to effectively leverage HPC resources to answer key questions in their field. We hope to fully automate our workflow so that researchers can easily configure their own MaxEnt runs to be performed on Nautilus and then explore the results interactively through the web interface. We also plan to extend the multivariate visual analytic capabilities of the interface.

Acknowledgment

University of Tennessee computer science undergraduate students James White and Adam Haney worked on reliably running the MaxEnt model in parallel while maintaining system-wide stability. We thank Sean Ahern for insightful discussions. We also owe professional gratitude to Keith Langdon and Thomas Colson of the National Park Service, Todd Witcher of Discover Life in America (DLIA), and Charles Parker of USGS for their insights in the application domain. William Godsoe was supported by a post-doctoral fellowship at the National Institute for Mathematical and Biological Synthesis (NIMBioS), an institute sponsored by the National Science Foundation, the U.S. Department of Homeland Security, and the U.S. Department of Agriculture through the National Science Foundation Award EF-0832858, with additional support from the University of Tennessee, Knoxville. This work was supported primarily by the Office of Cyber Infrastructure of the NSF under Award Number ARRA-NSF-OCI-0906324 for NICS-RDAV center and NSF-OCI-1136246 for support of undergraduate research in the NICS-RDAV center.

7. REFERENCES

[1] S. Afzal, R. Maciejewski, and D. Ebert. Visual analytics decision support environment for epidemic modeling and response evaluation. In *IEEE Conf on Visual Analytics Science & Technology*, 2011.

[2] N. Ferreira, L. Lins, D. Fink, S. Kelling, C. Wood, J. Freire, and C. Silva. Birdvis: Visualizing and understanding bird populations. *IEEE Transactions on Visualization and Computer Graphics*, 17:2374–2383, 2011.

[3] W. Godsoe. I can't define the niche but I know it when I see it: a formal link between statistical theory and the ecological niche. *Oikos*, 119:53–60, 2010.

[4] R. Hijmans and C. Graham. Testing the ability of climate envelope models to predict the effect of climate change on species distributions. *Global Change Biology*, 12:2272–2281, 2006.

[5] H. Lee, L. Zhao, G. J. Bowen, C. C. Miller, A. Kalangi, T. Zhang, and J. B. West. Enabling online geospatial isotopic model development and analysis. In *Proceedings of the 2011 TeraGrid Conference: Extreme Digital Discovery*, pages 1–8, 2011.

[6] A. Lenards, N. Merchant, and D. Stanzione. Building an environment to faciliatate discoveries for plant sciences. In *Proceedings from Gateway Computing Environments 2011 at Supercomputing11*, 2011.

[7] A. Malik, R. Maciejewski, B. Maule, and D. Ebert. A visual analytics process for maritime resource allocation and risk assessment. In *IEEE Conf. on Visual Analytics Science & Technology*, 2011.

[8] B. Nichols and K. Langdon. The smokies all taxa biodiversity inventory: History and progress. *Southeastern Naturalist*, 6(sp2):27–34, 2007.

[9] A. Peterson, J. Soberon, and V. Sanchez-Cordero. Conservatism of ecological niches in evolutionary time. *Science*, 285(1265-1267):3–27, 1999.

[10] S. J. Phillips, R. P. Anderson, and R. E. Schapire. Maximum entropy modeling of species geographic distributions. *Ecological Modelling*, 190:231–259, 2006.

[11] C. T. Rota, R. J. Flectcher, J. M. Evans, and R. L. Hutto. Does accounting for imperfect detection improve species distribution models. *Ecography*, 34:659–670, 2011.

[12] A. Szczepanski, T. Baer, Y. Mack, J. Huang, and S. Ahern. Usage of a hpc data analysis and visualization system. *IEEE Computer (accepted, in press)*, ?(?), 2012.

[13] R. A. Webster, K. H. Pollock, and T. R. Simons. Bayesian spatial modeling of data from avian point count surveys. *J. Agricultural, Biological and Environmental Statistics*, 13(2):121–139, 2008.

[14] R. H. Whittaker. A study of summer foliage insect communities in Great Smoky Mountains National Park". *Ecological Monographs*, 22(1):1–44, 1952.

[15] N. Wilkins-Diehr, D. Gannon, G. Klimeck, S. Oster, and S. Pamidighantam. Teragrid science gateways and their impact on science. *IEEE Computer*, 41(11), 2008.

[16] L. Zhao, C. Song, C. Thompson, H. Zhang, M. Lakshminarayanan, C. DeLuca, S. Murphy, K. Saint, D. Middleton, N. Wilhelmi, E. Nienhouse, and M. Burek. Developing an integrated end-to-end teragrid climate modeling environment. In *Proceedings of the 2011 TeraGrid Conference: Extreme Digital Discovery*, 2011.