Software for Automated Habitat Modeling 1.0.4

(SAHM)

User Help

DRAFT

**For help with SAHM Contact:**

**Jeff Morisette (970\226-9144;** [**morisettej@usgs.gov**](mailto:morisettej@usgs.gov?subject=SAHM%20Question)**)**

**Catherine Jarnevich (970\226-9439;** [**jarnevichc@usgs.gov**](mailto:jarnevichc@usgs.gov?subject=SAHM%20Question)**)**

**Updated 4/13/2010**

This guide provides instructions for using the SAHM command-line interface to run predictive models. Please review the following table of contents to obtain a general understanding of the information covered in this guide.

Table of Contents

[Overview 3](#_Toc258923856)

[Requirements for Field Data and Environmental Layers 3](#_Toc258923857)

[Overview of the Statistical Models 5](#_Toc258923858)

[Reports and Predictive Maps Created 5](#_Toc258923859)

[Installation 6](#_Toc258923860)

[Prerequisites 6](#_Toc258923861)

[Install R Base Software 6](#_Toc258923862)

[Run the SAHM Installer 7](#_Toc258923863)

[Create the Project Directory 10](#_Toc258923864)

[Copy Files to the Project Directory 11](#_Toc258923865)

[Data Preparation 13](#_Toc258923866)

[FieldDataQuery 13](#_Toc258923867)

[Resampler 15](#_Toc258923868)

[LowValsToNaN 16](#_Toc258923869)

[Model Generation Process 18](#_Toc258923870)

[Step 1: Create a Merged Data Set 18](#_Toc258923871)

[Step 2: Run the Models 19](#_Toc258923872)

[Step 3: Create the Predictive Map 20](#_Toc258923873)

[Step 4: Generate a Report 20](#_Toc258923874)

[Model Generation Process Using Quick Map 21](#_Toc258923875)

[Appendix: Summary of SAHM Modeling Techniques 22](#_Toc258923876)

[Generalized Linear Models (GLM ) 22](#_Toc258923877)

[Random Forest (RF) 22](#_Toc258923878)

[Boosted Regression Trees (BRT) 22](#_Toc258923879)

[Multivariate Adaptive Regression Splines (MARS) 22](#_Toc258923880)

[Maximum Entropy (MAXENT) 22](#_Toc258923881)

## Overview

SAHM (Software for Automated Habitat Modeling) is a decision tool which combines predictor layers such as NASA satellite data with user collected field sampling measurements for a particular species. The program then uses these data to analyze habitat requirements of the species of interest and predicts the potential distribution based on habitat suitability. Its outputs have the potential to help users generate color coded maps to aid in predicting and managing the spread of invasive species. To accomplish this SAHM uses **Field Data** which has been gathered from on-the-ground surveys and **Environmental Layers** in the form of imagery of the study area. Then, SAHM runs **Statistical Models** (that are either part of SAHM or provided by the user) with this data to create predictive maps or reports for users.

### Requirements for Field Data and Environmental Layers

**Field Data** used in SAHM have the following requirements:

* The coordinate projection system must match what is used for the environmental layers. Options are: Universal Transverse Mercator (UTM) or geographic (decimal degrees). The datum must be WGS 84.
* Must be in a comma-separated value (CSV) format with the file extension “csv”.
* The first line in the file is for labels.
* The first three labels must be
  + x: UTM easting or decimal longitude
  + y: UTM northing or decimal latitude
  + response:binary: presence or presence/absence at that location
* Additional columns are optional.
* Subsequent lines contain one location defined in the x and y columns with 1 for presence and 0 for absence in the response:binary column.



**Environmental Layers** (Predictors) used in SAHM have the following requirements:

* Environmental layer projection options are: Universal Transverse Mercator (UTM) or geographic (decimal degrees). The datum must be WGS-84. All layers must be in the same coordinate system.
* All layers must be in GeoTiff format with the file extension “tif”
* All layers must have the same cell size, resolution and extent

### Overview of the Statistical Models

SAHM includes a host of statistical models which are used to produce predictive maps. The installer currently contains five models: BRT, GLM, MARS, MaxEnt, and RF (see appendix for an explanation for each of these models). These are R scripts or Java archives located in the installation directory under Resources/ModelBuilder. They adhere to the input and output requirements of SAHM’s *pluggable* architecture.

To run SAHM (and the models in SAHM), a series of 4 command line applications (MdsBuilder, ModelBuilder, MapBuilder and RptBuilder) must be run by the user (either separately or bundled in QuickMap) in order to create the desired output.

How to run the command line applications (and thus the models) is outlined in the “Model Generation Process” section below.

NOTE: New models may be *plugged* into this architecture, if they conform to the input and output specifications. To run new models, copy them into the ModelBuilder subdirectory of your project directory.

NOTE: The four command line applications are bundled together as the QuickMap application. Both options (running the four command line applications independently or bundled in Quick Map) are detailed in separate sections in this guide.

### Reports and Predictive Maps Created

Depending upon the command line application which is run, SAHM outputs (reports and predictive maps) include:

* A merged data set (MDS) file
* An xml file containing model results
* A predictive map (in GeoTiff and JPG formats)
* An HTML report.

## Installation

NOTE: Depending upon your system’s security requirements running the R, the Java Run-time Engine (JRE) or the SAHM installer may require your System Administrators help.

NOTE: The installation instructions are written for a Windows operating system. Different steps may be required if using an operating system other than Windows.

### Prerequisites

* The Java Run-time Engine (JRE) must be installed.
* The R modeling engine must be installed with the packages gbm, mda, PresenceAbsence, rgdal, sp, randomForest, and XML. The base R installation and these packages are available from the R web site, http://www.r‐project.org/, all of which may be installed according to the R documentation.

NOTE:JRE may already be installed on your system.

NOTE: SAHM Requires R version 2.9.1 or greater.

### Install R Base Software

1. Download from <http://mirrors.ibiblio.org/pub/mirrors/CRAN>

As of 2/18/2010 the steps for downloading R include: Select Windows, select base subdirectory, select Download R <current version>, select Save. (Follow the R installation instructions on the site.)

2. Add R to the system path

a. Start ‐‐> My Computer ‐‐> View System Information ‐‐> Advanced ‐‐> Environment Variables

b. From "System variables", highlight "Path" (As mentioned above – this may require assistance from your system administration team.)

c. Click "Edit"

d. Append ";C\Program Files\R\R‐2.9.1\bin" to "Variable value"

e. Click “OK” to close all Control Panel windows

3. Install R Packages

1. Start --> Programs --> R --> R <current version>

2. From the Menu bar at the top, select Packages --> Install packages

3. Choose download mirror site when asked (first time only)

4. Install the packages gbm, mda, PresenceAbsence, randomForest, rgdal, sp, XML (use ctrl +click to select multiple packages from the list).

### Run the SAHM Installer

Locate and Click on the SAHM file which was sent to you.

The first installer window shows a collection of components (*packs)* for installation. Ensure they are all selected, and click “Next.”



Choose the path in which to install the components, and click “Next.”



The packs will be installed. Wait until the progress bar indicates completion, and click “Next.”



**The following step is important:** behind the scenes, the installer is preparing the components. Wait until the “Next” button is activated, and click it. **Do not click “Quit.”**



From the final window, click “Done.”



Congratulations, you have completed the process for installing R and SAHM. The first step to running SAHM is to create a project directory. This is detailed in the next section.Create the Project Directory

To run a SAHM model, users must start by creating a project directory which will be used to store input and output data. A different project directory must be created each time a new project (a different set of field data) is worked on.

Open a terminal window (start >run> type cmd in the box) navigate to the directory in which you installed SAHM (cd <path to project directory>) (the project directory is SAHM in the example below) and run the following command:

java –jar CreateWorkingDir.jar



This creates the project directory structure. To see the project directory structure, return to your “My Computer” page and locate the project directory:



The project directory (WD-2009-12-17-14-32-30 in the above example) is named with a simple time stamp. (It is possible to name the project directory with a different name other than the time stamp.) It includes the following sub directories:

The **ancillaryOutput** directory which is a repository for model output.

The **layers** and **layers/categorical** directories will contain the environmental layers for the current project directory. Place layers with continuous data values in the top-level directory and layers with categorized data values in the categorical sub directory. All layers may contribute to the models. By adding or removing layers in this directory, you will add or remove layers to the model, thus experimenting with your models.

The **models** directory will contain model source code for the current project directory, thus selecting which models will be run. All models in this directory contribute to the final model output. By adding or removing models in this directory, you can experiment with different combinations of models.

### Copy Files to the Project Directory

From a data repository, copy the field data file for this model run to the project directory (here WD-2009-12-17-14-32-30):



From a data repository, copy the continuous data environmental layers for this model run to the “layers” subdirectory of the project directory and the categorical data environmental layers to the layers\categorical subdirectory:



From the Resources/ModelBuilder subdirectory where SAHM is installed, copy the implementation files for the desired model(s) to the “models” subdirectory of the project directory:



Although you are now ready to run the model, it is possible the field data or environmental layers will need to be prepared (or cleaned up). The next section, Data Preparation, will address how this can be done.

## Data Preparation

Data preparation is an important first step in running Species Distribution Models. SAHM contains Utility Command Line Programs, which assist in preparing (or properly organizing) both the data collected in the field as well as any raster datasets the user wished to consider as predictive layers. These tools help clean up and geospatially reference the data and values that will be used to run the Models. These include:

* **FieldDataQuery** - Aggregates field data locations so that there is only one field data location per pixel.
* **Resampler** - Ensures that all environmental layers have the same cell size and spatial extent as required by SAHM.
* **LowValsToNan** - Attempts to indentify and then convert a layer’s lowest value to NaN (not a number.)

### FieldDataQuery

In many instances data collected in the field can be redundant, both spatially and temporally. When running species distribution models (SDMs) such as those contained in SAHM, spatial issues need to be addressed in order to avoid introduction of pseudo-replication. For instance, considering multiple field data observations which are all spatially located in the same pixel will generate replicate values or redundant information. When running a model this redundancy causes pseudo-replication, and can negatively influence model development. The FieldDataQuery tool helps aggregate field data locations so only one field data observation is represented per pixel. Similarly, field data is often collected temporally through time, where observations of the same location are often recorded annually. The tool additionally allows the user to account for the temporal resolution of the field data by preserving multiple field data observations falling in the same pixel, but collected in different years.

**EDIT the siteConfig File**

SDM research is carried out on many spatial scales, and a user needs the ability to define the extent and resolution of each study area or Area of Interest (AOI). The siteConfig file provides SAHM with the geospatial reference information for each AOI, which is manually defined by the user. Editing the siteConfig file to add a new AOI serves as the reference template defining both the geographic reference and associated spatial resolution of the AOI.

From the Resources/Sites subdirectory in the SAHM installation folder, copy the siteConfig.xml file to the project directory (to ensure the master copy is not worked from or changed).



Modify the siteConfig.xml file to create a site that matches the desired environmental layers. To do this:

* From within the existing siteConfig.xml copy the contents of an existing site (a site is defined as a block of text that begins with a <SiteInfo> tag and ends with a corresponding </SiteInfo> tag). In the example below, you would copy the text highlighted in blue and insert it into the same file below the existing site.
* Modify the information in the text that you just inserted to customize it for your new site - this includes creating a name for the site, modifying the EPSG number (see <http://spatialreference.org/ref/epsg/?page=1>) to match the coordinate system used, changing the height and width that correspond to the number of rows and columns, and modifying the bounding box llLat (bottom), llLon (left), urLat (top), urLon (right).



Return to the terminal window (or open it again) and navigate to the project directory (here WD-2009-12-17-14-32-30). Run ***one*** of the following commands:

java -jar FieldDataQuery.jar -c “path to siteConfig.xml” -n “site name in siteConfig.xml” -f “path to field data csv file” -o “path to save new field data csv file” –p

java -jar FieldDataQuery.jar -c “path to siteConfig.xml” -n “site name in siteConfig.xml” -f “path to field data csv file” -o “path to save new field data csv file” –y

Where:

-p: aggregate field data so there is only one field data location per pixel

-y: aggregate field data so there is only one field data location per year per pixel

(Other arguments for this command are detailed in the screen below and can be listed by using the –h parameter.)



### Resampler

Data products used in environmental research are produced from different sources at different scales and in many different spatial resolutions. Specifically raster data sets contain ranges of values and spatial resolutions that can influence the validity of a SDM if not properly taken into consideration. The resampler provides the user with the ability to resample all raster data sets to one specific AOI of the same resolution using the SiteConfig file. Resampling allows the user to consider various types of raster datasets at the same scale. It organizes all the raster data into the same cell size (spatial resolution) containing the same number of rows, number of columns, and coordinate information. This tool can be used to ensure that all of the environmental layers have the same cell size and spatial extent as required by SAHM.

From the terminal window run the following command:

java -jar Resampler.jar -c “path to siteConfig.xml” -n “site name in siteConfig.xml” -i “path to layers directory”

(Other arguments for this command are detailed in the screen below and can be listed by using the –h parameter.)



This also requires modifying the siteConfig.xml file as described above for FieldDataQuery.jar.

LowValsToNaN

Environmental data products deal with unprocessed or excluded values differently. Some products report a NaN (not a number) value, while others will reassign unprocessed pixels to values outside the possible product range, such as -9999. In these instances, values that are considered “Lower” than the possible range need to be identified and reassigned a value of NaN. This processing of reassigning null pixels assures that the SDM does not consider these values during the model runs.

* LowValsToNaN attempts to identify and then convert a layer’s lowest value to NaN (not a number). It is available because identifying null pixels is important to models and GeoTiff does not support null values, but it supports NaN
* LowValsToNaN uses the mean and standard deviation to create a normal distribution. It computes the cumulative probability of the lowest value to determine:

if it is an outlier: Chauvet’s Criteria

if it is indeterminate, it will say so and do nothing

* LowValsToNaN can be run manually (on a single image or a directory of images) OR on the fly (during MdsBuilder).

From the terminal window run the following command:

java -jar LowValsToNan.jar -f “path to GeoTiff file”

(Other arguments for this command are detailed in the screen below and can be listed by using the –h parameter.)

Model Generation Process



This next section will take you through the command line applications which create SAHM output. These include:

* MdsBuilder
* ModelBuilder
* MapBuilder
* RptBuilder

NOTE: These applications can be run separately (as detailed in the next 4 steps) or can be bundled together into one step through the use of Quick Map (detailed in the section “Model Generation Process Using Quick Map”).

### Step 1: Create a Merged Data Set

From the terminal window - navigate to the project directory. Run MdsBuilder.jar with no arguments:

cd <path to project directory>

java –jar <path to installation>/MdsBuilder.jar

This creates a file in the project directory named after the field data file with the “mds” extension. Seki\_brte\_utm.mds, for this example.



The merged data set that is created might look like this:



### Step 2: Run the Models

From the terminal window - navigate to the project directory, and run ModelBuilder.jar with no arguments:

cd <path to project directory>

java –jar <path to installation>/ModelBuilder.jar

This runs one or more models as defined by the files in the models subdirectory of the project directory and creates a model output file in the project directory named after the field data file with the “xml” extension. This file contains tagged output representing the results of the modeling and conforms to the SAHM model output specification. The ancillaryOutput directory contains very detailed individual model specific results that are not part of the SAHM specification.

NOTE: New models may be *plugged* into this architecture, if they conform to the input and output specifications. To run new models, copy them into the ModelBuilder subdirectory of your project directory.



The xml file that is created might look like this:



### Step 3: Create the Predictive Map

From the terminal window - navigate to the project directory, and run MapBuilder.jar with no arguments:

cd <path to project directory>

java –jar <path to installation>/MapBuilder.jar

This combines the binary output images from each model into a composite predictive map   
(GeoTiff and JPG versions).



The predictive map that is created might look like this:



### Step 4: Generate a Report

From the terminal window - navigate to the project directory, and run RptBuilder.jar with no arguments:

cd <path to project directory>

java –jar <path to installation>/RptBuilder.jar

This takes the XML output file generated by the ModelBuilder application (step 2) and uses it to generate an HTML file (with the same name but with the “html” extension). This file contains the model output as well as a link to the JPG version of the predictive map that was generated by the MapBuilder application (step 3) and can be viewed via a web browser.



## Model Generation Process Using Quick Map

Instead of running each process separately, they can be bundled into one step by running QuickMap. Prior to running QuickMap, it is still necessary to create the project directory, copy the files to the project directory, and complete any needed data preparation.

From the terminal window - navigate to the project directory, and run QuickMap.jar with no arguments.

cd <path to project directory>

java –jar <path to installation>/QuickMap.jar

## Appendix: Summary of SAHM Modeling Techniques

For the SAHM application, 5 different modeling techniques to predict invasive species presence have been employed.

### Generalized Linear Models (GLM)

This is basically linear regression adapted to binary presence-absence data. We used a bidirectional stepwise procedure to select covariates to be used in the model. That is, we began with a null model and calculated the AIC (Akaike Information Criterion) score for each covariate which could be added to the model. AIC is a measure of how well the model fits the data with a penalty based on the number of covariates in the model. In the first step, we add the covariate with the best AIC score. In the next step we calculate AIC scores for all two-covariate models and again add the covariate that most improves the AIC, and so on. At each step, we also look at the change in AIC from dropping each covariate currently in the model. The stepwise procedure ends when no additions or removals result in an improvement in AIC.

### Random Forest (RF)

Random forest is a machine learning ensemble classifier. Numerous (in our case, 1,000) decision trees are computed using random subsets of the covariates. Each tree gets one vote, and whichever class gets the best vote “wins”. The relative importance of each covariate is assessed by the change in a fit statistic, on average, for trees that include it. Random forest models automatically model interactions and nonlinear relationships.

### Boosted Regression Trees (BRT)

BRT is also based on numerous decision trees. BRT starts with a single decision tree, then adds a tree that best explains error in the first tree, and so on. Like random forest, BRT models automatically model interactions and nonlinear relationships, and are robust to missing observations. Our implementation makes approximately 1,000 trees, and incorporates advanced algorithms for tuning the model settings, simplifying the model using a cross-validation technique, and for detecting important interactions between covariates.

### Multivariate Adaptive Regression Splines (MARS)

MARS is a non-parametric technique that builds flexible models by fitting piecewise logistic regressions. In effect, it is similar to GLM except that rather than fitting a straight line response to each predictor, piecewise functions of each predictor are fit, which allows MARS to better accommodate nonlinear response to predictors. The model is deliberately overfit and then pruned back.

### Maximum Entropy (MAXENT)

Maxent is a general-purpose machine learning method with a simple and precise mathematical formulation, and it has a number of aspects that make it well-suited for species distribution modeling. Maxent is different from the other models in that it uses a ‘background’ sample as opposed to absence data. It is also different in that it is implemented via a stand-alone java application rather than being run in R.