

Fort Collins Science Center

Tutorial for the Software for Assisted Habitat Modeling (SAHM) package in VisTrails.

By Marian Talbert, Colin Talbert

Report Series XXXX–XXXX

U.S. Department of the Interior

U.S. Geological Survey

U.S. Department of the Interior

KEN SALAZAR, Secretary

U.S. Geological Survey

Marcia K. McNutt, Director

U.S. Geological Survey, Reston, Virginia: 201x  
Revised and reprinted: 201x

For more information on the USGS—the Federal source for science about the Earth,  
its natural and living resources, natural hazards, and the environment—visit   
<http://www.usgs.gov> or call 1–888–ASK–USGS

For an overview of USGS information products, including maps, imagery, and publications,  
visit <http://www.usgs.gov/pubprod>

To order this and other USGS information products, visit <http://store.usgs.gov>

Suggested citation:  
Author1, F.N., Author2, Firstname, 2001, Title of the publication: Place of publication   
(unless it is a corporate entity), Publisher, number or volume, page numbers; information   
on how to obtain if it’s not from the group above.

Any use of trade, product, or firm names is for descriptive purposes only and does not imply   
endorsement by the U.S. Government.

Although this report is in the public domain, permission must be secured from the individual   
copyright owners to reproduce any copyrighted material contained within this report.

**Contents**

Introduction 5

Getting Ready to Run an Initial Workflow 6

Installation and Configuration of VisTrails and SAHM 6

Selecting a workspace 6

Knowing where to find help 6

About the example used in this Tutorial 7

Running your first workflow 7

Using the CovariateCorrelationAndSelection widget 9

Viewing the model output 13

Model Output 14

Tracking model performance 16

Using the VisTrails History 17

Finding the difference between two history nodes 19

Modifying existing workflows 20

Adding a new predictor to a workflow 21

Modifying our Existing Workflow for Presence-Only Modeling and Examining Related Tools 23

Presence-Only with Random Background Points 24

Presence-Only with a Kernel Density Estimate (KDE) for Background Points 26

Modifying an Existing Workflow for Count Data 27

Applying a model to a new region 27

Applying a model to independent evaluation data 29

Running Models with your own data 30

Gathering your Data 30

Starting a new VisTrails VT file 30

Trouble Shooting 32

References: 34

**Figures**

**Figure 1.** VisTrails graphical user interface (GUI) with the initial Brewer’s Sparrow node open. 8

**Figure 2.** The Covariate Correlation and Selection Widget. 10

**Figure 3.** The Predictor Inspector Widget 11

**Figure 4.** Model run status viewer 12

**Figure 5.** SAHM spreadsheet output 13

**Figure 6.** Viewing model performance in a workspace 16

**Figure 7.** The History view in a VisTrails workflow. 17

**Figure 8.** Difference view between two workflows allows you to quickly identify what has changed. 19

**Figure 9.** Connecting a new Predictor module to our PARC module 22

**Figure 10.** Setting up SAHM to run with Random Background points. 24

**Figure 11.** A workflow for applying a model to a new spatial extent. 28

**Figure 12.** An error encountered in a module and the corresponding error message. 32

**Tables**

**No table of figures entries found.**

# **Introduction**

The purpose of this tutorial is to provide a step-by-step guide to setting up and running a SAHM workflow using the VisTrails Graphical User Interface (GUI) as well as introducing the user to the features and functionality of the Software for Assisted Habitat Modeling (SAHM) package through these examples. This tutorial will use several examples to demonstrate how to input data, how to add and configure models to build a workflow, how to execute a model, how to modify a workflow and how to view the multiple types of model results. These examples will demonstrate the range of niche models, options and response types that are presently supported by the SAHM package. This document is organized to gradually introduce features and demonstrates the steps a user would generally take to build a workflow starting with the tutorial workflow and adding new modules to customize the application to fit their needs. This tutorial is intended to accompany the SAHM user guide (SAHM\_UserGuide[1].docx which is included in the SAHM folder of your download and installation files). The SAHM user guide provides much more detailed information on installation, provides a VisTrails Primer, and documentation for each module in the SAHM package as well as information on interpreting model evaluation metrics. The Software, Tutorial and User Guide may be updated periodically as additional features are added to SAHM or bugs are fixed. Please visit <https://my.usgs.gov/catalog/RAM/SAHM> to download the latest version.

# Getting Ready to Run an Initial Workflow

## Installation and Configuration of VisTrails and SAHM

In order to work through this tutorial you must have downloaded and installed VisTrails and SAHM as well as its dependencies R and the Java program Maxent. Detailed instructions for this installation are available in the SAHM User Guide. It might also be helpful to review the section of the user guide entitled “VisTrails Primer” which provides the basic information about working in VisTrails.

### Selecting a workspace

The workspace is the folder where all your output from the very initial preprocessing through the final model output is stored. When model output is created all evaluation metrics and graphics are saved in your session folder so that you can access and use these as needed. To select a workspace go to the menu bar at the very top of the GUI and select Packages->SAHM->Change Session Folder and then navigate to the desired folder.

### Knowing where to find help

It is worth pointing out at the outset where the user can find help when trouble arises. When any Module is highlighted within a VisTrails workflow the user can click the Documentation button to the left of the workflow canvas. We have provided detailed information on the likely input and output ports for each module as well as the formats required for each parameter. If one desires information on a specific input or output right click on the Input\Output name and click view documentation. The SAHM user guide also provides detailed documentation on each module as well as information on interpreting output. At the end of this tutorial we have included a section on trouble shooting when errors occur which will be periodically updated to include several frequently encountered problems. We have also put together a SAHM Google group (https://groups.google.com/forum/#!forum/vistrails-sahm) which we hope will become a forum where SAHM users can post questions and generate discussions about species distribution modeling and where SAHM developers can provide information on the latest updates to the software.

# About the example used in this Tutorial

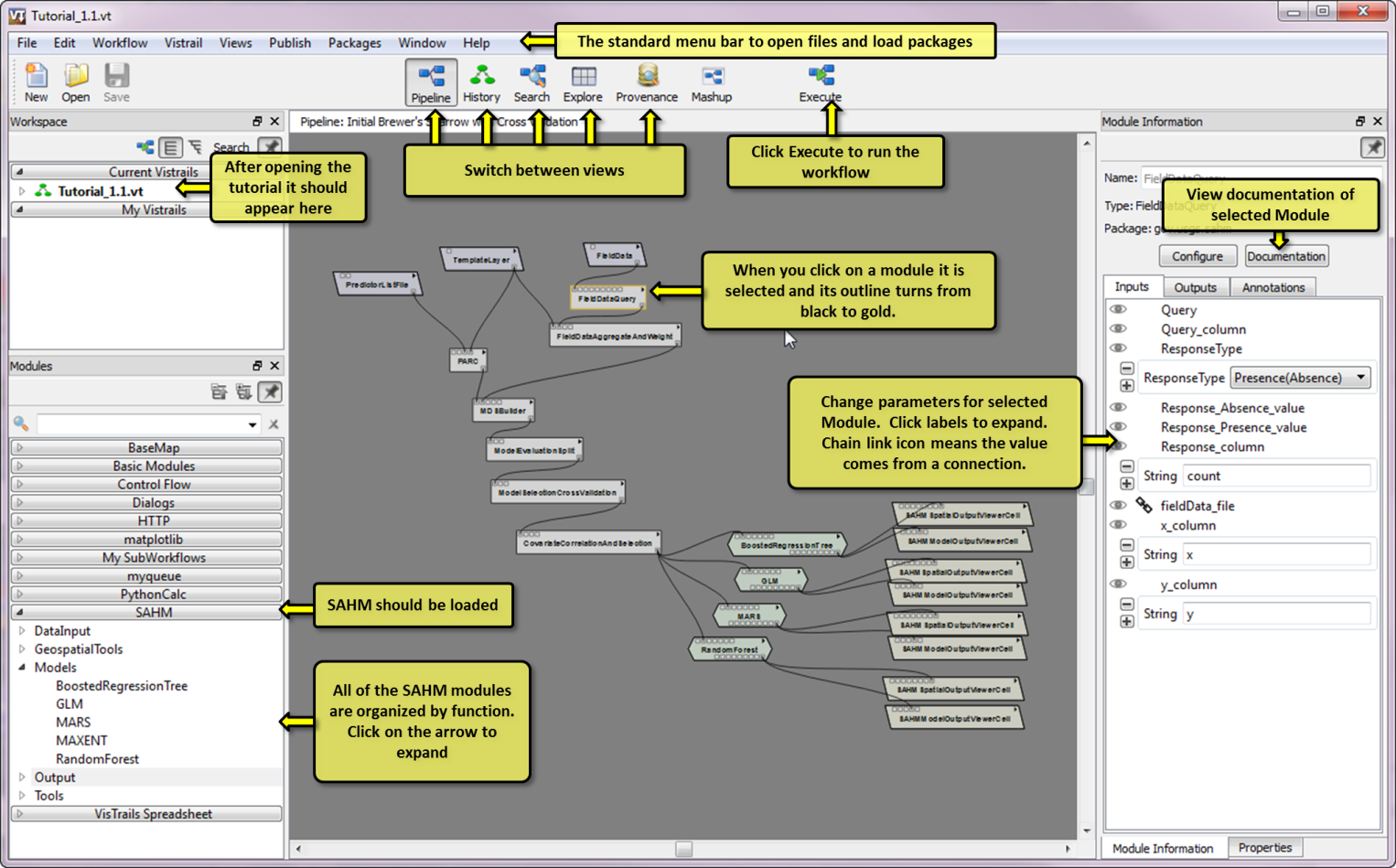
The data set we are looking at consists of count data for Brewer’s sparrow (Spizella brewer) in the Wyoming Basins, USA. The data were downloaded and are freely available from the Sagebrush Ecosystem Conservation and Management: Ecoregional Assessment Tools and Models for the Wyoming Basins website (<http://sagemap.wr.usgs.gov/wbea.aspx#CAdata>). The data were collected as species counts but these are aggregated to presence/absence and presence-only for demonstration purposes in this tutorial.

# Running your first workflow

If you don’t already have it open, go ahead and open VisTrails. Open the tutorial by either clicking the Open icon or File->Open and navigating to the Tutorial\_1.1.vt file which can be found in ..\examples\SAHM\_example in the folder where you installed VisTrails. You should see several connected boxes appear in the canvas and under Workspace on the Left panel you should see the Tutorial\_1.1.vt in bold. If your view does not look identical to Figure 1 you might be in the wrong node in the history.

At this point you should also see SAHM under the Modules that are available. By clicking on the triangle next to the SAHM button we can see the classes of modules available within SAHM including DataInput, GeospatialTools, Models, Output, and Tools. Clicking the arrow next to each one displays the modules for each class that are currently available. If you don’t see the SAHM Module available click Edit->Preferences. Then click on the Module Packages Tab, highlight sahm in the Disabled packages box (top left), click Enable and close this box. This should load the package.

One of the powers of VisTrails is in the flexibility it allows the user to customize the workflow to suit their needs. You can get a glimpse of some of the control the user has over defining the various components of this workflow by clicking on some of the modules in the canvas. For example if you click on modules at the top of the canvas such as the TemplateLayer, PredictorListFile, or FieldDataQuery you can see different ways to point to input data and to specify several ways it can be used. Clicking on the model modules such as BoostedRegressionTree or RandomForest you can see the many parameters the user can access to define how a model is calibrated and to customize the output to suit the user’s needs. We will go into great detail on how the user can work with these modules to build customized workflows for their specific needs but first we will allow the user to click execute on a complete workflow that we have created in order to demonstrate one of the SAHM packages other strengths (e.g. visualization).



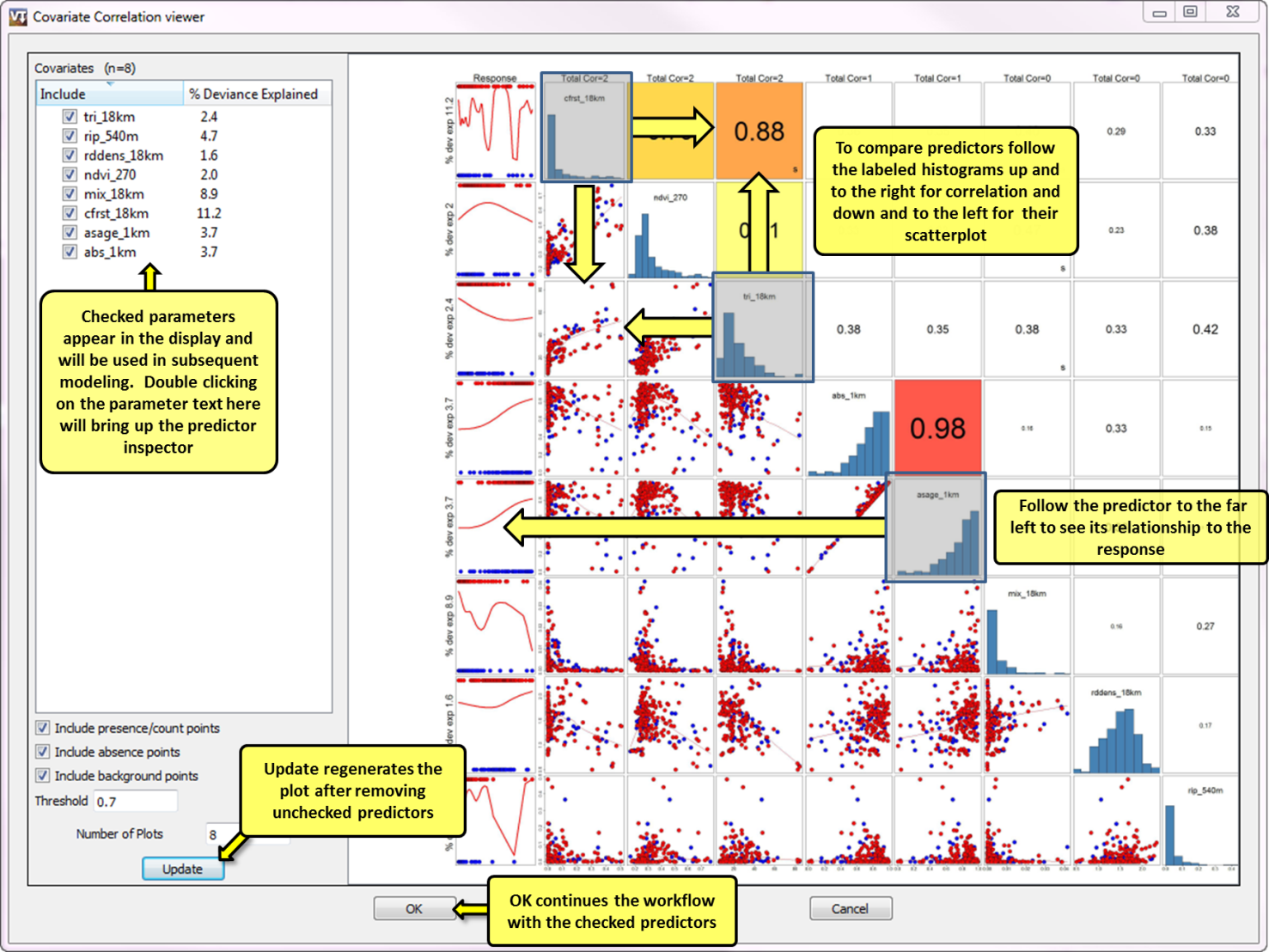
1. VisTrails graphical user interface (GUI) with the initial Brewer’s Sparrow node open.

First, make sure you are displaying the Initial brewer’s sparrow with cross-validation workflow. To do so, click the ‘History’ button in the menu bar directly above the canvas. If the highlighted bubble is labeled ‘Initial Brewer’s Sparrow with Cross Validation’, you don’t need to do anything else. If not, click on this bubble. Once it is selected click the ‘Pipeline’ button to return to the workflow canvas with the intial workflow now displayed. To execute this workflow click the Execute button in the top center of the menu bar. Clicking this will sequentially execute the code behind each of the modules starting from the top and working down. The order they execute in is not specified but all upstream modules must complete before any downstream modules execute. You will notice each module turn green as it finishes successfully. As each module completes successfully, output from that module is saved in the model workspace you selected. When an error is encountered in the workflow this output can often provide an indication of what when wrong. Examining the output from each module can provide you with a better understanding of what each module does. When you get to the CovariateCorrelationAndSelection module, execution will pause and an interactive window displaying the correlation between the covariates used will appear. Sometimes you must hover over the VT icon on the application bar of your computer and select this window to bring it up manually.

## Using the CovariateCorrelationAndSelection widget

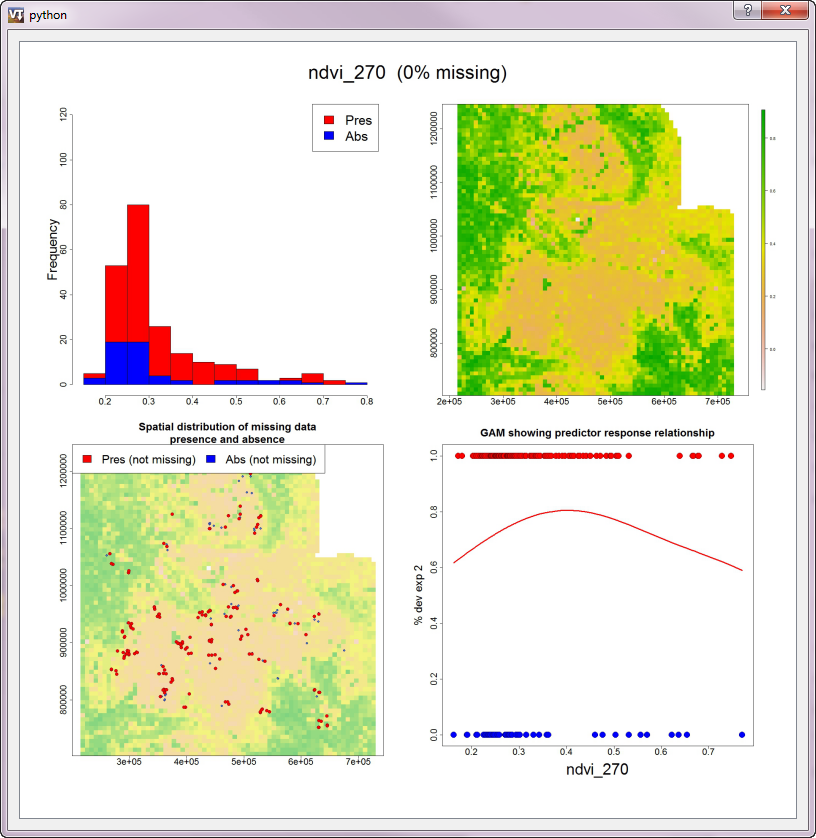
If you are following along this tutorial sequentially you should have the CovariateCorrelationAndSelection widget up now. If not follow the directions in the previous paragraph to get to this point. The widget that you are seeing should appear like Figure 2.

Data inspection is a critically important step in fitting models. Many times there are unexpected problems with data that can be identified by close inspection. Midway through our workflow we use the CovariateCorrelationAndSeleciton module to perform a detailed inspection of the data and to filter out highly correlated predictors which if left in the model can lead to erratic performance for slightly different data and poor ability to predict even within the environmental space used for model calibration. We have implemented an interactive display to allow the user to iterate through the process of covariate filtering. This display works by sorting on the covariates that have the highest number of correlations above some threshold (by default 0.7) with other predictors under consideration. This allows the user to view and remove those predictors in a sequential order which can be extremely useful when the initial number of predictors is very high.



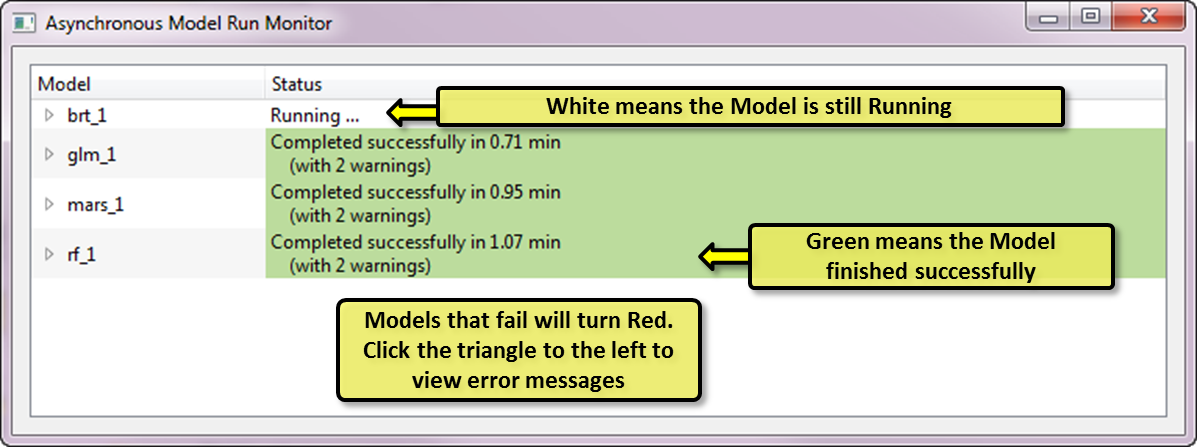
1. The Covariate Correlation and Selection Widget.

From this display we can easily see several useful items. The relationship between each predictor and the response are displayed in the far left column, with left most labels giving the percent deviance explained in a univariate model (with just one predictor and the response). We can see across the top row the total correlations each predictor has above our selected threshold (0.7 by default) with other predictors. If any observations are missing for a given predictor, this information will be displayed across the top as well. The upper right portion displays the highest of the Pearson, Spearman, and Kendal correlation coefficients between each pair of predictors and the bottom right portion shows the scatterplot relationship between each pair of predictors. A standard rule of thumb is to remove any predictors with a correlation with another predictor above 0.7. Decisions about which predictor to remove should be based primarily on a biological understanding of what is important to the organism but other factors such as patterns in missing data or the relationship between each predictor and the response can also be taken into consideration. The predictor inspector widget can help you with this. Double clicking on any of the predictor names in the covariate list on the left will provide a more detailed view of the frequency of the response (e.g. presence or absence), the spatial pattern of the predictor, where in space the observations are located and graphics to identify if a spatial pattern in missing observations might be non-random. Try double clicking on a predictor name, wait a few seconds and you will get an image that looks something like Figure 3. For larger covariates this can take as long as a few minutes.



1. The Predictor Inspector Widget.

You can close out of the predictor inspector widget by clicking on the “x” in the top right corner. Removing a predictor is accomplished by unchecking the box next to the predictor name. To refresh the Covariate Correlation after having removed predictors click “Update”. Once you are satisfied with the set of predictors to include in further analysis click “OK” and the workflow will continue through model fitting. Because of the way SAHM utilizes multiple cores on a single computer the models turn green immediately even though model fitting has not yet completed. An Asynchronous Model Run monitor as shown in Figure 4 should appear which monitors the model run progress and lets you know when a model has completed and/or any errors that were encountered. If the monitor does not appear you can get tot it by going to Packages->SAHM->Check Asynchronous model runs.

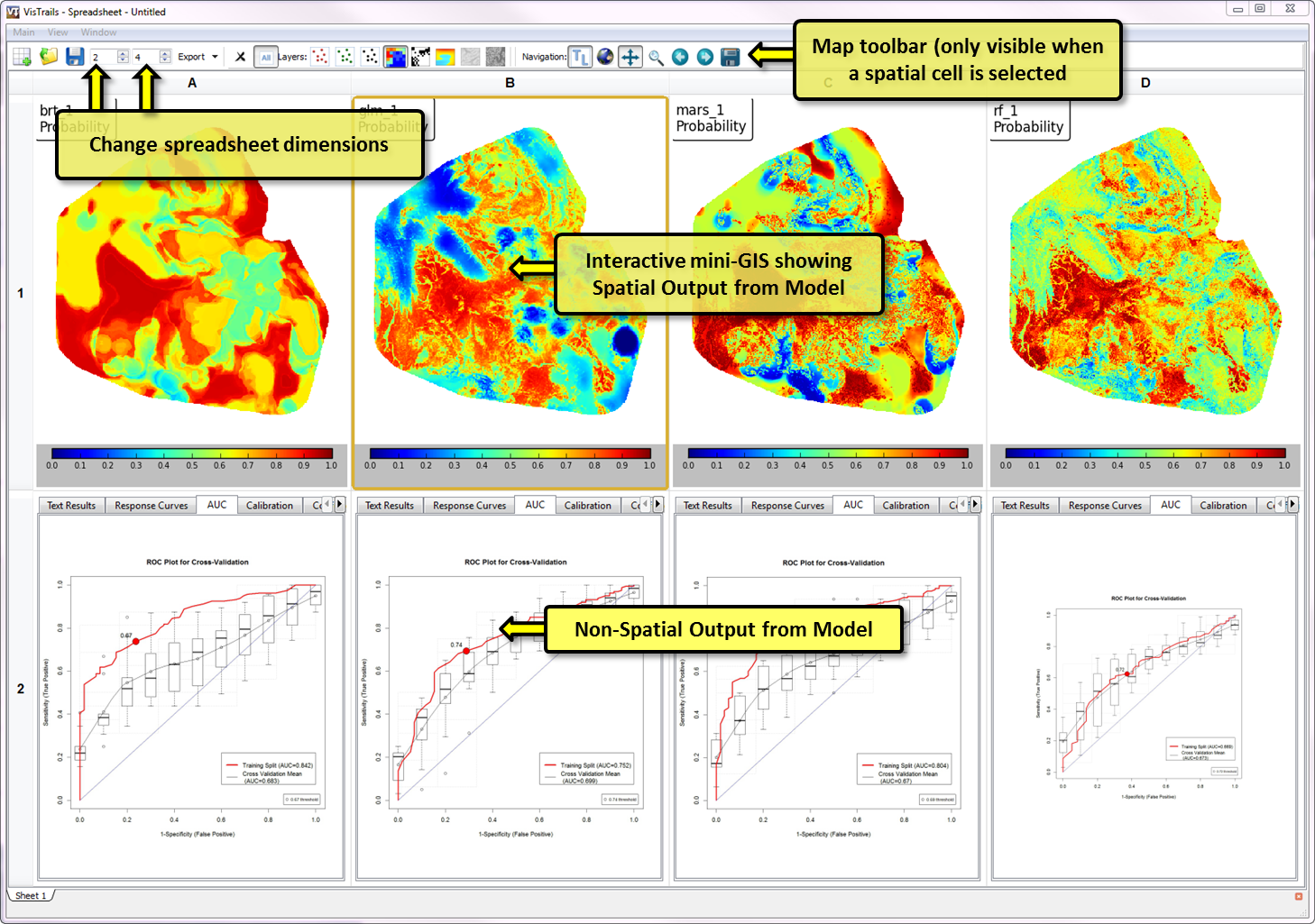


1. Model run status viewer.

After all models have turned green, close this popup window and click ‘Execute’ again to display the model results. The spreadsheet display will likely appear in a two row, three column configuration. Since we want to view eight separate cells (two for each model) we need to add a fourth column. Do this by changing the ‘3’ to a ‘4’ in the number of columns box on the main toolbar of the spreadsheet. Click ‘Execute’ again to refresh the display and fill the new column. Your spreadsheet should look like Figure 5.

## Viewing the model output

The built-in method for viewing output in VisTrails is the spreadsheet. Conceptually it is similar to any other spreadsheet except that instead of the cells containing numbers each contains a visual representation of a particular output. In the case of SAHM each output cell in the spreadsheet contains either a mini-Geographic Information System (GIS) of the predictive maps (SAHMSpatialOutputViewerCell) or a widget that displays various charts used to analyze model performance (SAHMModelOutputViewerCell). Organizing output in this fashion using the spreadsheet allows us to quickly, and in great depth, compare the myriad of comparable output inherent to these modeling techniques produced from multiple model runs. The VisTrails:SAHM framework allows the more tedious and error prone aspects of file management to be escaped to some degree.



1. SAHM spreadsheet output.

The top row in this spreadsheet contains the spatial output and the bottom row contains the model output, including charts and other quantitative model output. If you click on one of the maps, a mini-GIS toolbar appears above the spreadsheet which allows you to interact with the maps. For example, you can turn on or off the presence/absence points, switch between probability, binary, or other map outputs, or change the cursor from zooming or panning modes. Note that when zooming or panning all maps will track together. More information about the specifics of functionality is included in the User Guide. The model output row includes tabs to switch between the different charts, textual, and other quantitative output.

## Model Output

Our software produces several graphics that can be used to compare various aspects of model evaluation. All of these are included as jpeg, geotiff, or text files within your model workspace directory as well as being viewable in the spreadsheet. The interpretation of these is discussed in great detail in the user manual and will be covered only briefly here. The ROC curve and related AUC value are probably the most commonly reported (model output AUC tab). The plot shows the relationship between sensitivity and specificity as the threshold for conversion of the continuous probability to binary presence/absence prediction is varied between 0 and 1. The information this provides relates to the ability of the model to discriminate between presence and absence.

Calibration plots and statistics describe the goodness-of-fit between the predicted values and the actual observations (Calibration tab). These are especially useful for identifying problems with overfitting or underfitting when separate data is used for model fitting and model evaluation. These plots and statistics can be used to determine how reliably the model will predict if a site is occupied or unoccupied (Pearce and Ferrier 2000). If how well the predictions correspond to a probability is not a concern these metrics can be ignored.

The confusion matrix shows the percent of predicted and observed values in each of the presence and absence classes (Confusion tab). For predicted values these percentages are dependent on the threshold used to discretize the predicted values. The rule used to calculate the threshold is one of the input parameters available in each of the model modules. Several other common evaluation metrics are based on the discretization of the continuous predictions and could be seen as accompanying this plot. These include percent correctly classified, sensitivity, specificity, Cohen’s kappa, and the true skill statistic, which are all reported below the plot.

Model response curves show the relationship between each predictor included in the model and the fitted values, while holding all other predictors constant at their means (Response Curves tab). These surfaces can be examined to assure that the relationships agree with the biological understanding of the organism under study but please keep in mind that these do not take into account interactions between predictors which might have been included in the model. Each individual response curve can be displayed or an image displaying all response curves for predictors in the model at one time can be displayed by selecting from the drop-down list in the viewer cell.

Model residual plots show the spatial relationship between the model deviance residuals (Residuals tab). Most models assume residuals will be independent thus spatial pattern in the deviance residuals can be indicative of a problem with the model fit and inference based on the fit.

Variable importance plots provide a model independent comparison of the relative influence of predictors in each model (Variable Importance tab).

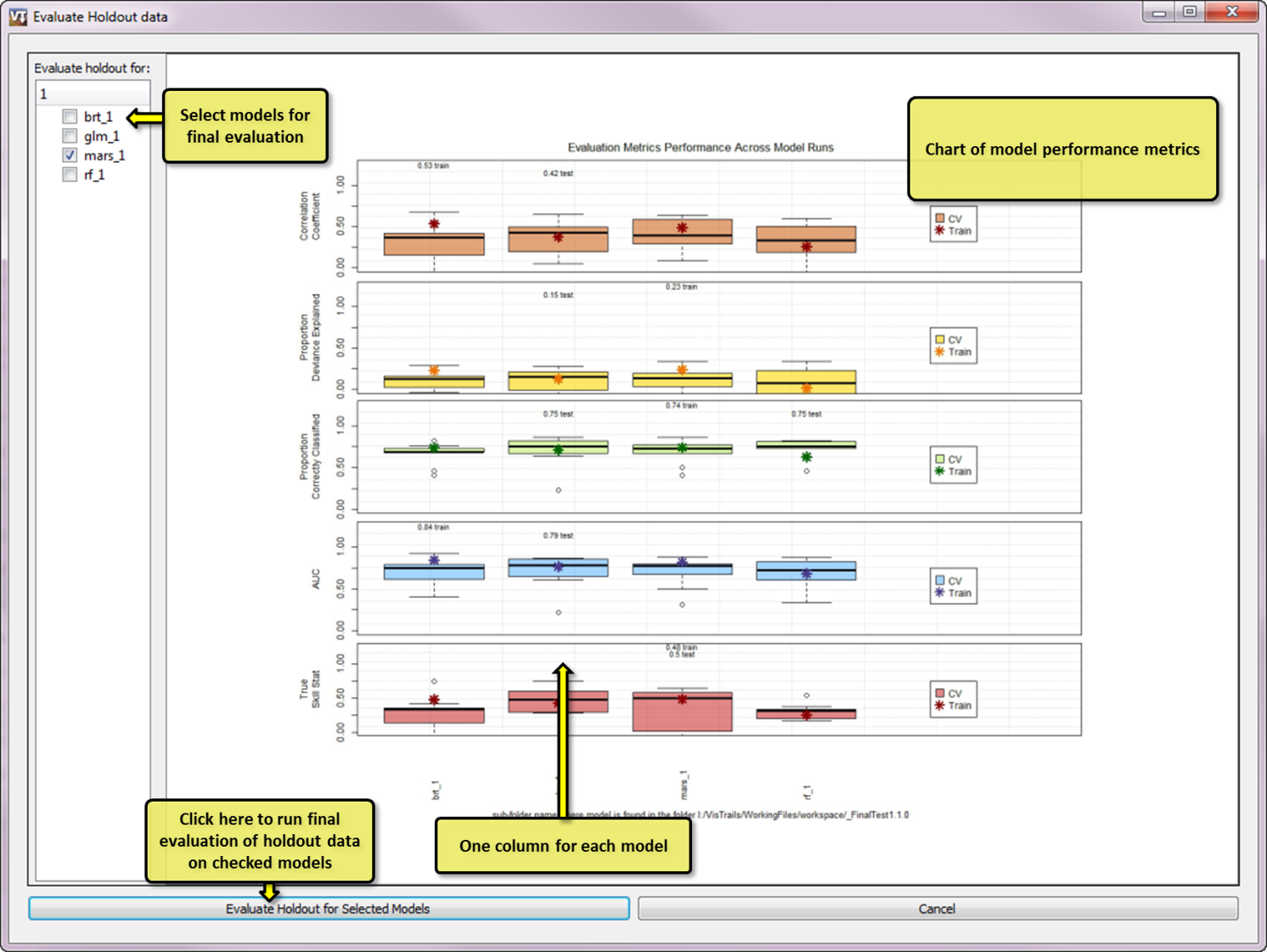
Other optional outputs for each model include different types of spatial output. In this workflow, probability and binary maps were selected. The probability map is the default map that first appears in the spatial output cell. The probability map is a surface of predicted values produced based on the predictor layers (in tiff format) selected in the CovariateCorrelationAndSelection module and the fitted model. (All models in SAHM except Random Forest have a variables selection process within the modeling fitting). These maps can, but do not always, indicate the probability of finding the species at a given site. If calibration is poor then they might not correlate well with actual probabilities, but rather would indicate relative suitability of locations. Also keep in mind that probability is defined in the sampling design. For example, a probability map might indicate the probability of observing the organism in a 100m patch if monitored for 10 minutes if that is how the presence data was collected. It becomes even more complicated for presence only studies, where at best this map will be proportional to occurrence.

The binary map is produced by discretizing the probability map using the same threshold rule used to calculate the threshold dependent metrics in the Confusion tab of the model output cell.

The MESS surface is the multivariate environment similarity surface and shows how well each point fits into the univariate ranges of the points for which the model was fit. Negative values in this map indicate that the point is out of the range of the training data. The MoD map is related and indicates which variable was furthest from the range of the observations used for model training. These optional outputs are not produced unless one checks the ‘makeMesMap’ parameter on the model.

# Tracking model performance

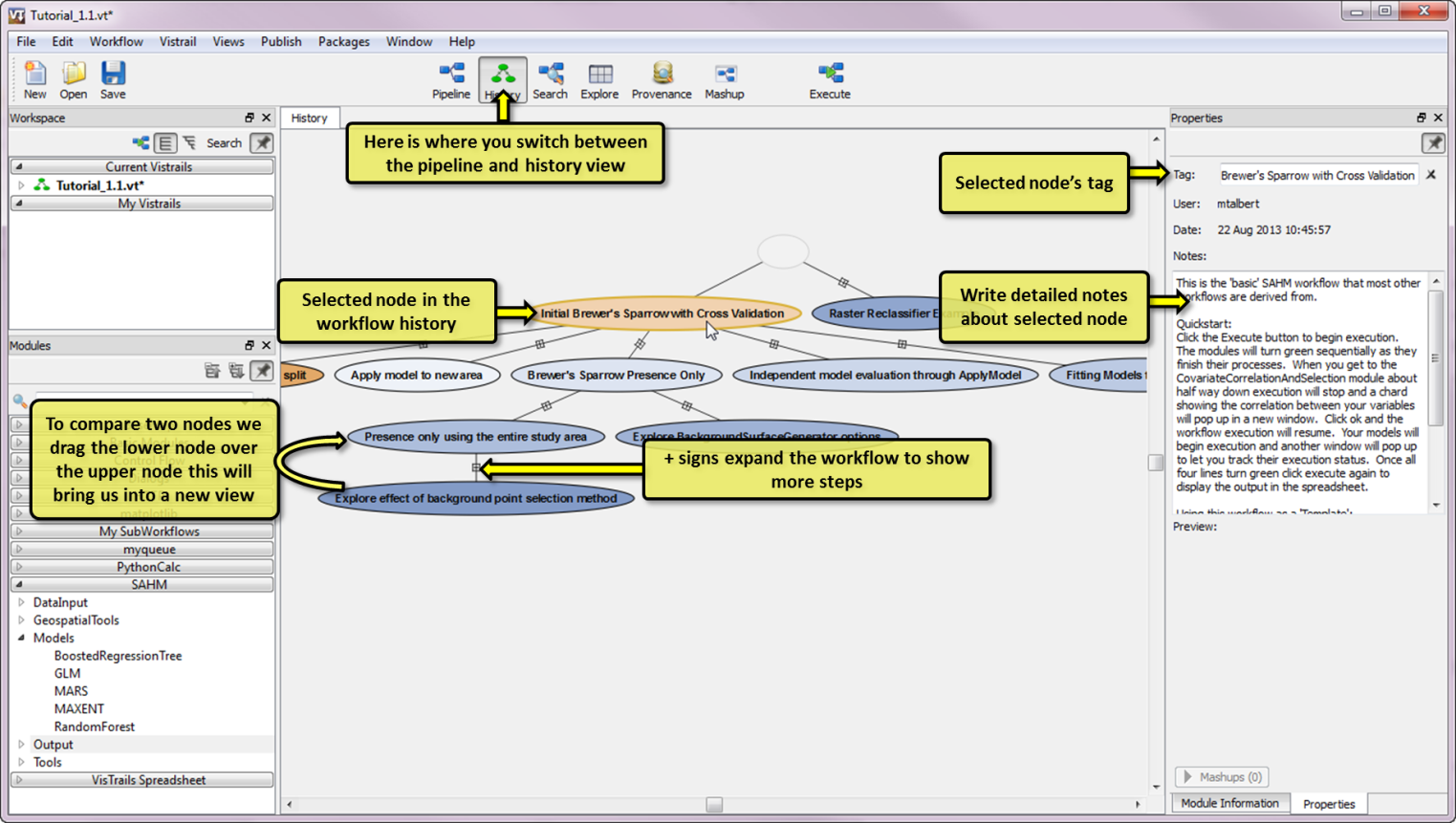
Within a workspace SAHM will compile a graphic describing model performance. So for example if you run a workflow and then iteratively change parameters or try different groups of parameters or covariates, you will be able to track your model performance throughout this process. Click on ‘Packages’->’SAHM’->’Select and Test the Final Model’ and a graphic appears showing model performance for all models with the given response and data splitting options in the workspace allowing the user to see how different decisions affected the model fit. Because this initial workflow included the model evaluation split, you can also choose to evaluate your best models on the hold out data withheld by this split from this screen see figure 6.



1. Viewing model performance in a workspace

# Using the VisTrails History

One of the more powerful features of VisTrails is its ability to record a detailed history, also called provenance, of a complex, iterative, or evolutionary process. To view a schematic of this history click on the History button in the menu bar above the canvas. In the history view you should see several labeled ovals as in Figure 7. Each oval is referred to as a history node and represents a snapshot of the various modules and parameters specified at a specific point in the history. Clicking on one returns the workflow canvas exactly to the way it was set up when that node was labeled. Try clicking on a few of the nodes and switching back and forth between the history and pipeline views. You can see that by clicking on different nodes in the history and then clicking back on the Pipeline at the top we move very easily from one pipeline to another. Try navigating between a few of the history nodes and switching back and forth between the History and Pipeline views to get comfortable moving around VisTrails.

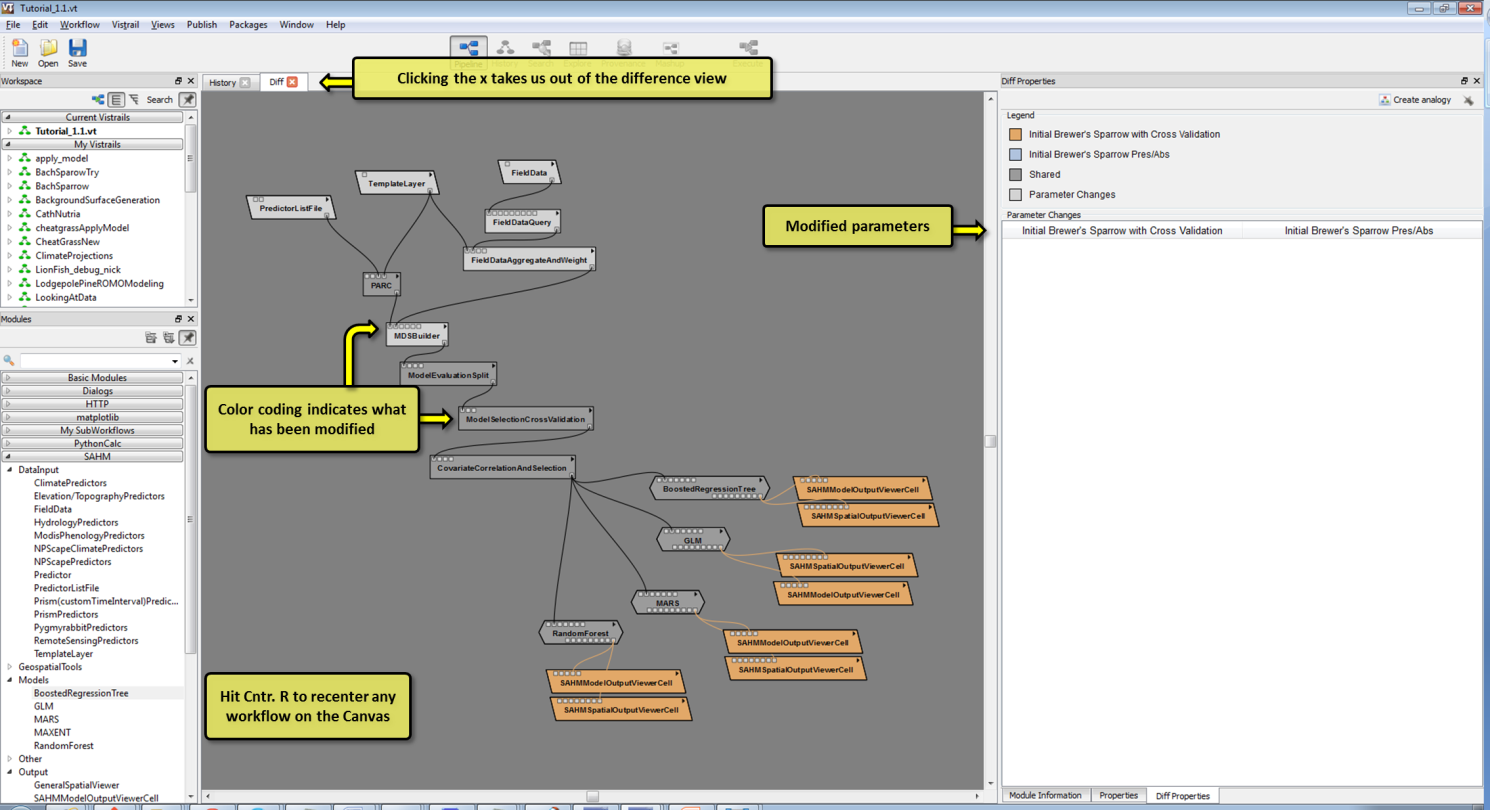


1. The History view in a VisTrails workflow.

Unfortunately, VisTrails has no way of knowing which points in the history of your workflow development are significant and it records a new history node for every change to the modules, connections, or parameters. To keep this level of detail from overwhelming the history view, and making it less helpful to you, it is very important that you give significant nodes a meaningful name. It is also quite useful to include more detailed notes about each workflow. Significant in this context might mean ‘this workflow ran’, ‘this workflow produces the model used in our publication’, ‘this work flow uses the field data from the 2010 season’, etc. To change the name of a history node you can either double click inside the oval and type the new name or type the new name in the Tag box on the right side of the screen when the desired node is encircled in gold. Additional notes can be kept in the Notes box on the right side of the GUI. Minor history changes that weren’t given names are folded up in the plus sign along the line that connects two named nodes. To expand and view these changes click on this plus sign so we can see the steps we took to get from one node to another. In general one would only navigate between the named nodes.

## Finding the difference between two history nodes

If we want to know exactly what was changed between two workflows (history nodes) we can click on a node and drag it on top of another node, which brings us to a difference view that looks like a pipeline but actually highlights the differences in the modules, connections and parameters between the two selected history nodes see Figure 8.



1. Difference view between two workflows allows you to quickly identify what has changed.

# Modifying existing workflows

Modules are added to a workflow by dragging the item from the Available Modules list onto the Workflow Canvas. You will notice on each module a series of small squares on the top and bottom of its shape. These represent input and output 'ports' or parameters for the module. The inputs and parameters are along the top and outputs are on the bottom. When hovering over a text box that gives the port name and data type will appear. See the ‘Knowing where to find help’ section of this tutorial for information on module and port documentation.

Modules are added into a workflow by clicking and dragging from the output port of one module into the appropriate input port of the next module. There are numerous valid ways of connecting modules depending on the processing required. When starting out it can be difficult to know how each module should connect to other elements in the workflow. Please note that while we have limited the ports that will connect to those of the appropriate data type, a common error occurs when two ports are incorrectly connected. It is generally useful at first to open and modify an existing workflow instead of starting from scratch. The module and port documentation as well as the package documentation found in the user guide can be informative as well. Modules or connections that are no longer needed can be removed by selecting them and clicking the 'Delete' key.

As workflows become larger you might find it helpful to resize and navigate around the workflow canvas. The scroll wheel and sideways scroll wheel action move around (pan) the workflow canvas. Panning can also be accomplished by holding shift and the left mouse button and moving the mouse. By holding down the right mouse button and moving the mouse up or down you can zoom in and out. Pressing control+r will re-center the canvas. While you can create a workflow from scratch it has been our experience that a user almost always produces a workflow through modifications to an existing workflow. For that reason we have included several nodes in this tutorial demonstrating example workflows that we will describe. You can of course modify these to suit your own needs.

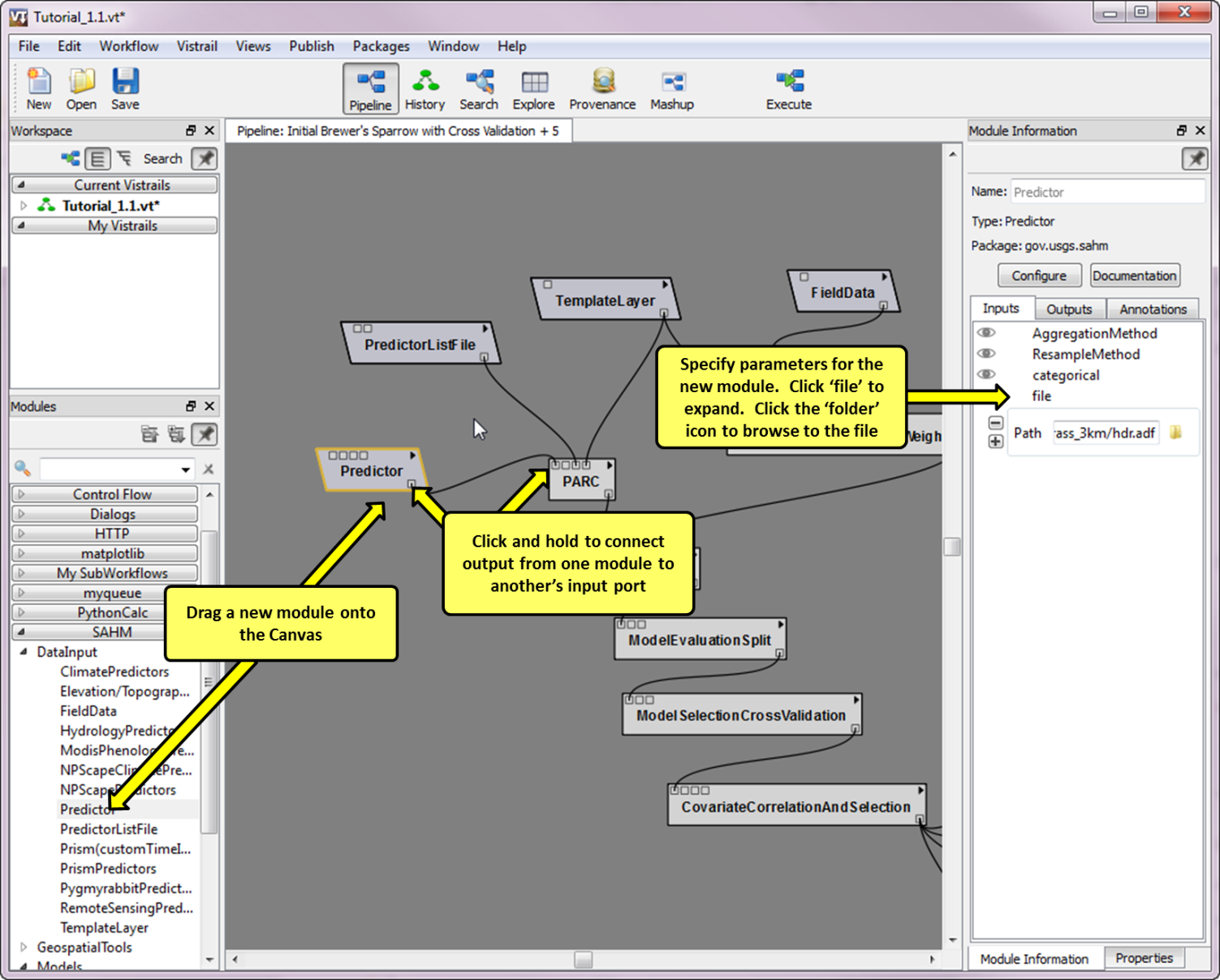
## Adding a new predictor to a workflow

In this section we will be taking an existing workflow and modifying it to add an additional predictor (covariate grid). This demonstrates the fundamental way that we extend and explore the models we are investigating. Start by going back to the ‘Initial Brewer’s Sparrow with Cross Validation’ workflow. All of the covariates that we have included are specified in a PredictorListFile. Let’s say that a collaborator gives us an additional covariate that he thinks is an important determinate of this species habitat. We do not want to start a workflow from scratch, but merely want to add that covariate to the existing workflow.

To accomplish this we need to drag a new module onto the canvas, set some parameters in it, and connect it to our workflow. Start by finding the ‘Predictor’ module in the DataInput group in the SAHM module list found on the bottom left hand side of the application in a pane called Modules. You might have to expand the SAHM list as well as the DataInput list to find this module. To assist with finding modules you can also start typing ‘Predictor’ in the search box at the top of the Modules pane. Click and hold the Predictor module in this list and drag it onto our canvas.

This Predictor module represents a single file and a few parameters that are needed to include it in our workflow. The most important of these parameters is the location of the file we want to include. To set this click the’ file’ parameter in the module information tab on the right of the application. Now click the little folder icon to navigate to the file we want to use. For this example we have supplied a file which can be found at ..\SAHM\_example\additionalCovariateGrid\grass\_3km\hdr.adf. Note that when specifying an ESRI grid you can either specify the folder (grass\_3km) or the file hdr.adf (grass\_3km\hdr.adf) but because of the way in which the file browser in VisTrails is set up you will need to click the file to close the browser. Do not worry about the other parameters in the module for now as the defaults will be fine.

To connect this new module to our workflow click the small box on the lower right of the new Predictor module (also known as an output port) and drag it onto the small box on the upper left of the PARC module (also known as an input port). Do note that there can be multiple input and output ports for each module and that each of these has a specific type. For example, a module might output a file, a string and integer each specified with a separate output port box. When connecting output ports to input ports of subsequent modules take care that you are connecting the appropriate ports. You will get a tooltip which specifies the name and type of an input or output port when you hover over it with your mouse. You can also tell what input port you have connected to by looking for an image of a chain link next to a parameter in the input list for a selected module. Figure 9 show what this new connection will look like. After connecting the new module, click execute to run the workflow with this new predictor included. When the model processing finishes click execute again to see the output. The history node ‘Adding new predictor’ should be nearly identical to the one you just created with the exception of the path to the new predictor. Verify this by dragging your new node over the ‘Adding new predictor’ node in the history view.



1. Connecting a new Predictor module to our PARC module

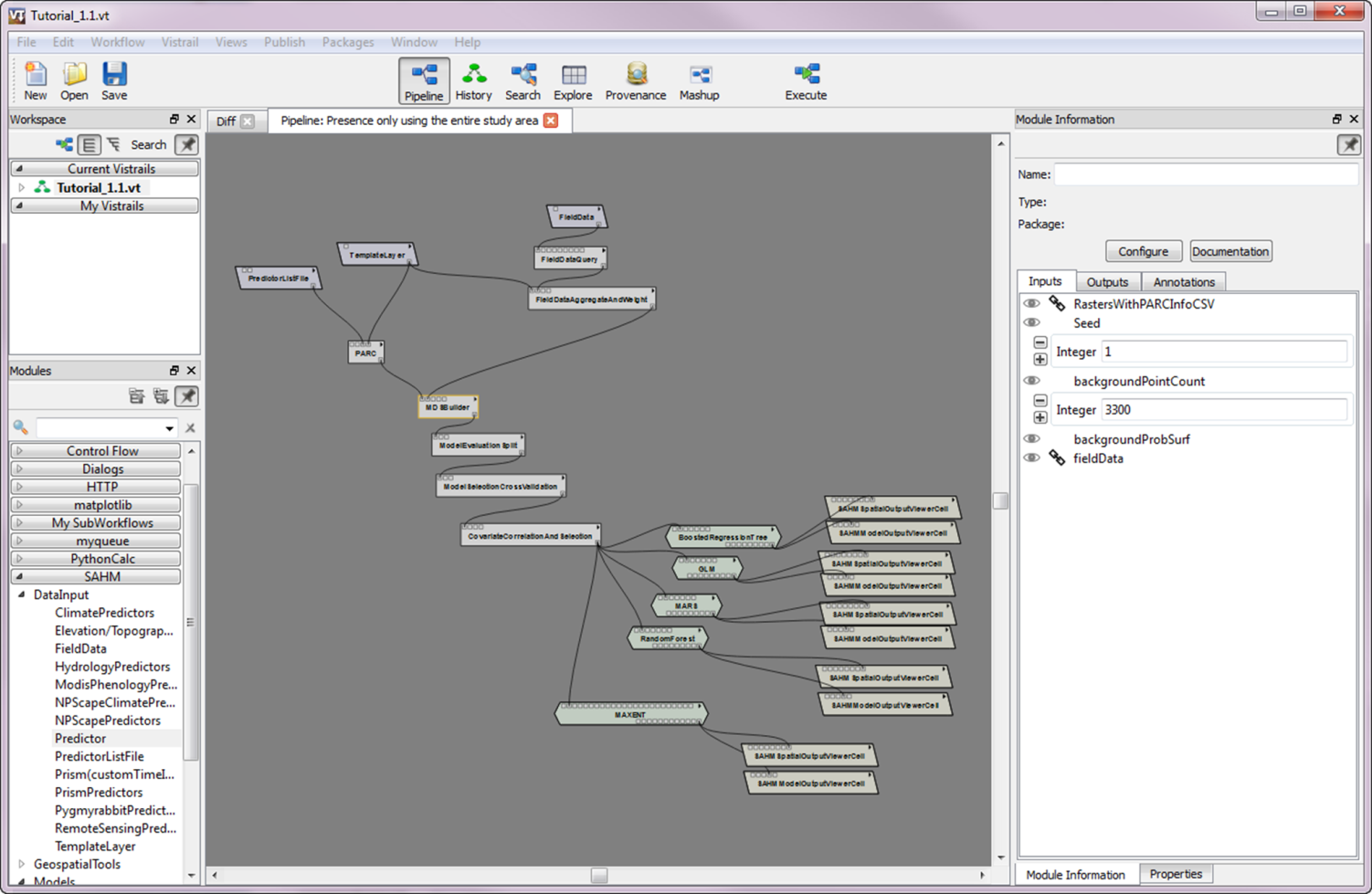
## Modifying our Existing Workflow for Presence-Only Modeling and Examining Related Tools

It is important to mention that when absence data are available it should in general be used. For the purpose of illustration, we will remove the absences from our dataset and demonstrate how one would generate background points to model when only presence data is available.

## Presence-Only with Random Background Points

The easiest way to generate background points is randomly within the template layer. This will honor any masked areas in the template file, e.g. NoData values in the template will be ignored. Modifying our ‘Initial Brewer’s sparrow with Cross validation’ workflow only requires a few steps for this simplest example.

1. Start on the “Initial Brewer's Sparrow with Cross Validation” node of the history and click on the Pipeline.
2. We must modify the query in the FieldDataQuery to remove the absence. This is done by clicking on the FieldDataQuery module and in the ‘Query’ parameter box type “[count] > 0” without the quotation marks. Change the ‘Query\_column’ to “count” without the quotation marks.
3. We need to also set the MDSBuilder to generate our background points. Do this by changing the backgroundPointCount parameter to 3300.
4. Finally, now that we are using background points, we can also use the MAXENT Model. Go to SAHM modules on the left and Expand SAHM clicking the arrow, expand Models in the same way, Click and hold MAXENT and drag it onto the Canvas. To connect this new module to our workflow click the small box on the lower right of CovariateCorrelationAndSelection (also known as an output port) onto the small box on the upper left of the MAXENT module (also known as an input port). This tells MAXENT that it will be receiving its input data from the CovariateCorrelationAndSelection output port.
5. Next, go back to SAHM Modules on the left and drag in the SAHMSpatialOutputViewerCell and the SAHMModelOutputViewerCell onto the Canvas. Connect the Maxent output port ‘modelworkspace’ to the input port on these modules. For each of these output modules you will also want to change the input parameter ‘column’ to 5 and the input parameter ‘row’ to 1 for SAHMSpatialOutputViewerCell and 2 for SAHMModelOutputViewerCell. This will allow the new output to appear in a predefined location on your spreadsheet. Your workflow should now look like Figure 10.



1. Setting up SAHM to run with Random Background points.

You can compare your work to the tutorial workflow by clicking on the History icon and dragging your new workflow over the node labeled “Presence only using the entire study area”. If there are no differences between the two then everything has been set up correctly. You can switch back to the pipeline and click execute to run the new workflow, which will again pase at the covariate viewer. When you get to this point, notice by looking at the top row of the graphic display that some of the predictors are missing 10% of data. Double click on the name of one of these with missing data under Covariates and you’ll find an interesting pattern of missing data. In this case the problem was likely caused by a mismatch in buffers of the the input data. You can resolve this issue by changing the template to one of our layers that has nodata values in this buffer area.

## Presence-Only with a Kernel Density Estimate (KDE) for Background Points

With presence-only data there is often a bias in how the data were collected. Generally efforts will be disproportionately near urban areas or roads for example. When models are fit to such datasets without consideration for this issue it is impossible to disentangle the extent to which you are modeling the species habitat and the extent to which you have modeled where people look f or the species. Unfortunately it is very difficult to quantify these biases so often we are only left with the option to explore how such an uncertainty might affect model output. For this reason we have incorporated several tools to create surfaces from which background points can be generated that use a variety of assumptions about what the sampling effort might be. From the workflow developed to fit presence-only models with random background points we only need to add one module to change the Background Surface.

1. Start on the “Presence only using the entire study area” node of the history and click on the Pipeline.
2. Drag in the BackgroundSurfaceGenerator moduel which is found under the “Tools” heading in the SAHM module listings.
3. The BackgroundSurfaceGenerator must know where the presence points are and must have a template in order to understand the spatial region where background points are to be generated. Connect the output from the TemplateLayer and the FieldDataAggregateAndWeight modules into the input ports of the BackgroundSurfaceGenerator.
4. Connect the BackgroundSurfaceGenerator to the MDSBuilder which we know from the last section is where background points are generated.
5. Click on the BackgroundSurfaceGenerator to see that the method we’re using. The default is KDE using a 95% isopleth and the continuous option turned off. This essentially draws a blob around the presence locations, but has the ability to ignore outlying points.

You can now click execute on this workflow or drag it over the node labeled “Brewer’s Sparrow Presence Only” to see how your modifications compare to ours for the same workflow.

## Modifying an Existing Workflow for Count Data

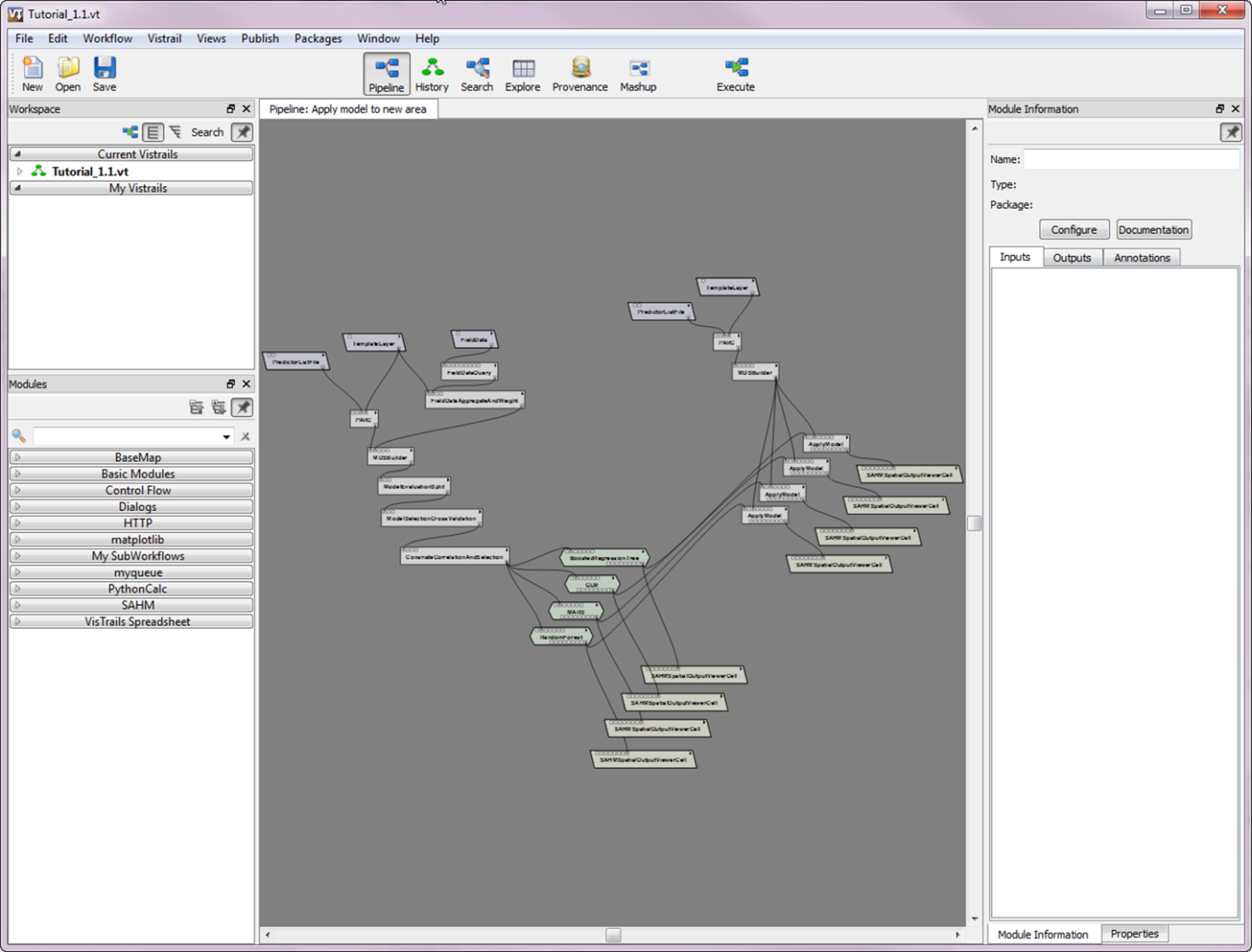
We originally set up our FieldDataQuery for presence/absence data, but when information on counts is available it is generally preferable to make use of all information contained in the response. In order to modify our workflow to handle a count response simply go into the FieldDataQuery and change the response type to Count. Unfortunately , in our experience, we seldom come across examples of count data. As such, the tools available for handling presence/absence or presence-only response are much more well developed. This workflow should be thought of as a direction of development which will continue with future versions of SAHM. Here, we only fit the simplest Poisson model but in the future we plan to add tools for zero inflated distributions as well as negative binomial which are more flexible and generally more appropriate for count data collected in species distribution studies.

## Applying a model to a new region

SAHM has the capability to save a model and apply it to a new spatial or temporal extent. Please note that there are several caveats that apply when projecting models to completely novel areas or to different temporal extents. The workflow should not be undertaken without a good deal of consideration regarding the various sources of uncertainty (see Beale and Lennon 2012 and Diniz-Filho et al 2009 for a discussion of the sources of uncertainty and a method of modeling in the face of large uncertainties). On the purely technical side, to do this we have to line up our new rasters. The new rasters must match the original rasters in name (e.g. bio1), but could be for a new location or time. You will need to specify a template for setting the extent, projection, and the mask and then provide the ApplyModel module with this raster information as well as the information from the model we would like to project.

1. Start on the “Initial Brewer’s Sparrow with Cross Validation” node of the history and click on the Pipeline.
2. By clicking just above the PredictorListFile and holding down the left mouse button we can drag a box over the PredictorListFile, TemplateLayer, PARC module and the MDSBuilder. You should see the outlines for all these modules as well as their connections turn gold. You can then hit Ctrl-c and Ctrl-v to copy and paste these items. Drag the new group to an empty part of your screen.
3. It turns out the original PredictorListFile contained a much larger extent than the template used for the original analysis. These areas were clipped to match the template so by just switching the template layer to the new extent we can apply the model to the larger region. Go into the newly pasted TemplateLayer and under the input field value navigate to the file ..\examples\SAHM\_example\BrewersSparrowData\Predictors\900m\rip\_540m.
4. Next drag in four instances of the ApplyModel modules which are available under Tools. Each of these must receive as input the output from one of the four models we ran earlier and the MDS file that we just built. We will then look at the maps compared to the maps produced from the original models. Drag in four SAHMSpatialOutputViewerCells and set these up to receive output from the ApplyModel.
5. Delete the SAHMModelOutputViewer from the original models and check to see that the new OutputViewerCells are pointing to the same columns as the corresponding models but that the original model output appears in row 1 and ApplyModel output appears in row 2.

Your workflow at this step should look something like Figure 11.



1. A workflow for applying a model to a new spatial extent.

## Applying a model to independent evaluation data

The best test of a model comes from applying it to a completely independent data set. When these are available we can set up the workflow to fit the model on one set of data and evaluate it on the new dataset to produce unbiased evaluation metrics. We have set up a node in our workflow to demonstrate how the user would do this. Unfortunately we don’t have independent evaluation data. To get this node to work the user would point to their independent data and the rasters that cover that extent in the three input boxes on the top right of the canvas.

# Running Models with your own data

## Gathering your Data

In order to run models with your own data you will need the following input data. More detailed information about each of these can be found in the SAHM user manual.

1. A Field Data file: a .csv file which contains X and Y coordinates and a response column. The response can be presence, presence/absence, or count. You must know the coordinate system/projection/datum the x and y are in. The exact formatting of this file is flexible as long as it is in CSV format and the data are arranged in columns. You can use the FieldDataQuery module to reformat it into the format expected by the rest of the workflow.
2. A series of geospatial raster layers: These must cover the extent of your field data locations. Many common raster formats are supported including geotif, ascii, image, and ESRI grids. While these do not need to be in the same projection, datum, or resolution, they all must have defined projections. Some unusual projections might cause problems in the automatic reprojection and resampling algorithm used in PARC.
3. A Predictor List File: a .csv file containing a list of the locations on your computer where the rasters corresponding to each covariate can be located, whether they are categorical, as well as a few other optional resampling details that are discussed in the user guide. Alternatively you can specify these directly in VisTrails by using individual Predictors modules.
4. A Template Layer : a raster which is used to define the spatial extent, resolution, and areas to be masked out in all subsequent analysis. This raster must have defined projection information.

## Starting a new VisTrails VT file

While it is possible to construct a complete workflow from scratch on a blank canvas, we have found it more efficient to use an existing workflow as a starting point or template. The history nodes in this tutorial can be used as such. Since you will not want your new VisTrail to be cluttered with the extraneous history present in this tutorial create a new, blank VisTrail by clicking the ‘New’ icon in the upper left or clicking File->New. You will notice that the canvas is now blank and a new VisTrail named Untitled.vt appeared in bold under your ‘Current Vistrails’ pane. Click save and select a meaningful file name and location.

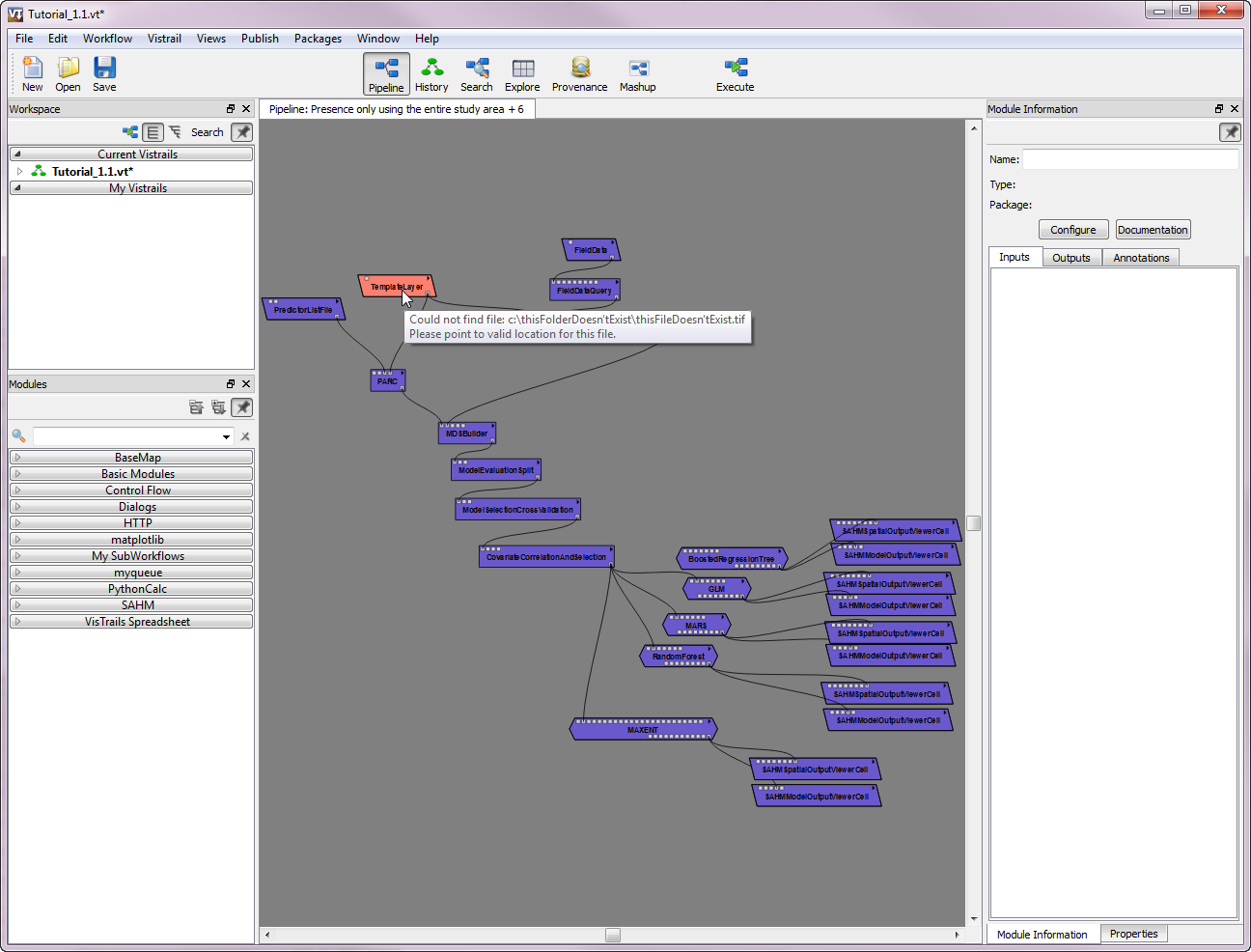
Now we will populate the initial node of this new Vistrail using one of the history nodes of the tutorial vt as a template. Select the Tutorial\_1.1.vt from the Current Vistrails pane and select the History view. Select the node in this workflow that most closely matches the modeling you are going to be doing. For example, if you have presence-only data select the node called ‘Brewer’s Sparrow Presence Only’. Switch back to the Pipeline view. Now select all the modules and connections on the canvas by drawing a select box around them (click and hold the left mouse button towards the upper left of the canvas and drag to the lower right). Alternatively you can select everything with a ‘Ctrl-a’. Copy the selected items with a ‘Ctrl-c’. Now select the Vistrail you just created by clicking it in the Current Vistrails pane. Click in the canvas and paste your copied modules with a ‘Ctrl-v’. The above technique can also be used to move portions of workflows between history nodes and different Vistrails.

Now that we have a template to work with we can modify it to our data and run it. Point the PredictorListFile, TemplateLayer, and FieldData modules file parameters to your specific versions. Change any other parameters that your model requires. For example Response\_column, x\_column, and y\_column in the FieldDataQuery will likely need to be modified. Click execute and if everything is set up correctly your models will run correctly. Switch back to the history node and label the last node with a name and notes of your choosing. From here you will be iteratively modifying this workflow and examining the output in a process similar to what we have done in the tutorial example above.

# Trouble Shooting

As with any complex piece of software there are a myriad of problems that can arise when using SAHM. Some of these are actual software bugs, but many are the result of various user errors or unanticipated (and thus not handled correctly) combinations of inputs and parameters. For many of the more common errors we have added checks that return a sensible message to alert the user what has occurred and what to do to fix it. Examples of this type of error include pointing to files that do not exist, malformed or nonstandard files, spatial data that does not have a defined projection, and specifying nonsensical parameters. When a workflow encounters one of these problems execution stops and the module that encountered the problem turns red. Hovering over the red module will display the error message as in figure 12.

In order to demonstrate this in a controlled manner we have added a node to the tutorial workflow to walk you through fixing errors. Click on the troubleshooting node and click Execute. Try to identify and correct the introduced errors. Hint: the workflow is the same as ‘Presence only using the entire study area’ with three introduced errors.



1. An error encountered in a module and the corresponding error message.

Although many errors can be identified in the process described above many require much more in depth investigation. In order to assist with this type of troubleshooting, we have included a series of log files in the output session folder. At the root of the session folder is a file called ‘sessionLog.txt’ which contains a detailed recording of exactly what was run and when, the command lines used to run external processes, error messages, etc. Additionally, the standard output and error generated by external processes is captured in separate text files. At the root of the session folder is one called errorLogFile.txt, and within each individual model folder (e.g.. rf\_1, mars\_2, etc.) you will find stdErr.txt and stdOut.txt. These can sometimes be used to deduce the problems and possible solutions.

As with most open source software projects the developers of SAHM are not funded to provide assistance to users of the software. When encountering problems using SAHM it is up to the user to investigate the issue fully themselves. This includes reading the built-in documentation and the user manual, comparing the problem workflow and inputs to those provided in this template and searching previous posts to the user group. If the problem remains unsolved the user should post the problem to the SAHM Google group (<https://groups.google.com/forum/#!forum/vistrails-sahm>). Other users and the developers will respond to these posts as able. As a last resort, and only concerning true bugs in the software should a user contact the developers of the software.

# References:

Diniz-Filho, J. A., L. M. Bini, T. F. L. B. Rangel, R. D. Loyola, C. Hof, D. Nogue´s-Bravo, and M. B. Arau´ jo. 2009. Partitioning and mapping uncertainties in ensembles of forecasts of species turnover under climate changes. Ecography 32:1–10.

Beale, C., and Lennon, J. (2012) Incorporating uncertainty in predictive species distribution modelling. *Phil Trans. R. Soc*, 367, 247-258.

Morisette, J. T., et al. 2013. VisTrails SAHM: visualization and workflow management for species habitat modeling. Ecography 36(2): 129-135.