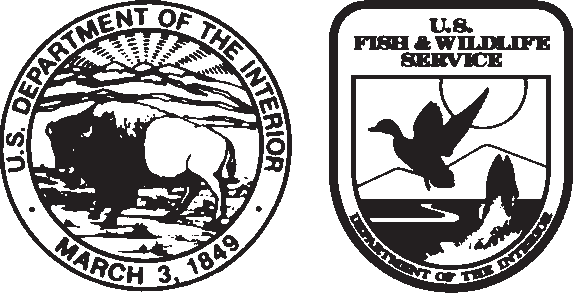
Text box: National Conservation Training Center



**Segmentation and Classification Basics with eCognition**

A training for: USFWS NCTC

**Developed in partnership with:**

Geospatial Technology and Applications Center (GTAC) | US Forest Service

**2022 Edition**

Kutz K., Vermillion ML, Segmentation and Classification Basics with eCognition, 2022 edition. U.S. Fish and Wildlife Service, National Conservation Training Center, Shepherdstown, West Virginia, USA.

This product is for educational purposes only

This is on the back of the cover page

[Cite your source here.]

**Module 2: Classification in eCognition**

**COURSE LEADER NOTE:**

It is suggested you use a functioning table of contents like the one pictured below. We suggested Arial or Time New Roman font because they are 508 compliant.

Directions:

1. Put your cursor where you want to add the table of contents.
2. Go to References
3. Select Table of Contents
4. Choose an automatic style.
5. **If you make changes to your document that affect the table of contents, update the table of contents by right-clicking the table of contents and choosing Update Field.**
6. Note you will have to use headings in the styles area above (normal, Body text, list paragraph etc.). You can also put a table of contents in manually.
7. There are numerous YouTube videos available if you get stuck.

**Table of Contents**

**Type chapter title (level 1) 1**

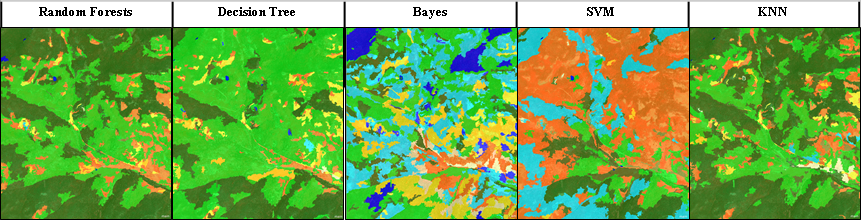
Type chapter title (level 2) 2

Type chapter title (level 3) 3

**Type chapter title (level 1) 4**

Type chapter title (level 2) 5

Type chapter title (leve



**Module 2: Classification in eCognition**

Introduction

eCognition offers more advanced workflows that can produce even better results because the classier runs on the image segments instead of pixel-wise. Additionally, eCognition makes it easy to include multiple layers, to include vector data, and to create attributes for each image-object, like the mean NDVI to be used in the classification. eCognition provides access to Bayes, Decision Tree, k-Nearest Neighbors, Support Vector Machines, and Random Trees (i.e., Random Forests) classifiers. In this exercise you will use the data from Module 1, including the NDVI that was created, then using training data to classify the image and examine the results.

# Objectives:

By the end of the session, you should be able to:

* Classify and image in eCognition
* Understand object-based image classification

# Create Image Segments

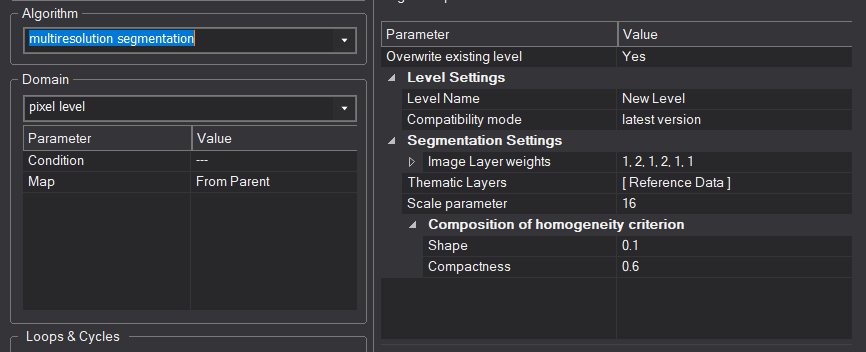
Keep the same project as used in Module 1, we will use the same data to preform an image classification.

## Add Multi-Resolution Segmentation Process

1. From the **Process Tree**, right click and select **Add New Process**
2. Change the **Name** section where it says **“do”** to **Segmentation** and leave the **Algorithm** set to **execute child processes**
3. Leave the rest of the settings at their defaults and click **Ok**
4. Right click on the **Segmentmentation** rule in the **Process Tree** window and select **Insert Child Process**

5. From the **Algorithm** drop down menu, select **multiresolution segmentation**

*Note: If you begin typing the name of the algorithm you would like, the software automatically fills in the options that match. You can use your down arrows to select these - a shortcut compared to scanning and selecting from the full list of options.*

1. Find the **Scale** Parameter and set it to **16**. Then find the **Shape and Compactness** parameters are set to **.1** and **.6** respectively
2. Next set the image layer weights. Set the following image layer weights by selecting the drop-down arrow to the left of the **Image Layer Weights** parameter option.
3. Set **DEM** and **NDVI** to **2** - Leave all other layer weights to **1**. 

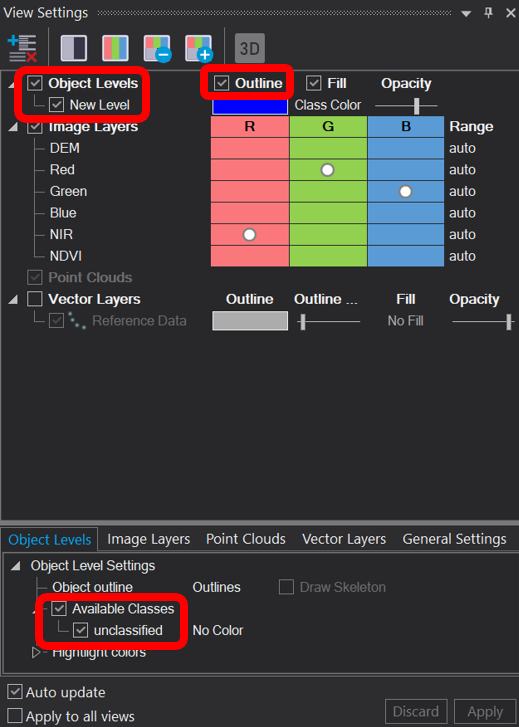
**Note:** There is no limit to setting the layers weights. The difference in numbers is the relative proportion of influence that the different layers will have on the segmentation process. In this instance DEM and NDVI will have twice as much influence on the segmentation process than the other bands. We increased the weights of these bands because it is expected that these variables have the strongest influence on the objects that we are interested in, which is vegetation distribution.

# Segment the Image

1. From the Process tree, right-click on the Parent process called ‘Segmentation’ and select Execute. This will execute all child processes—in this case, just the segmentation process.

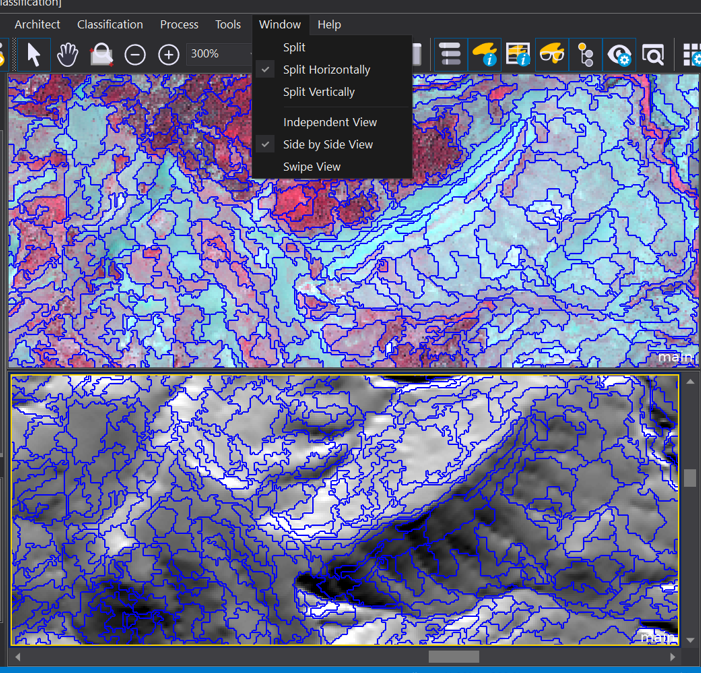
**NOTE:** Since we are working with a large image, it will take a few minutes to complete the process.

1. If the segments are not visible when the segmentation process finishes, make sure the boxes outlined in red (see image below) are checked within the **View Settings** window.



1. Use the **scroll wheel** on your mouse or the plus and minus buttons on the main menu to zoom in and out to **visually evaluate the segments**. Also use the pan-hand to pan around the image and inspect the segments.

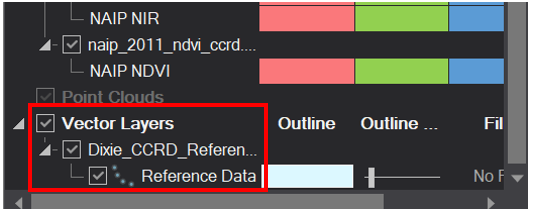
Do the segments effectively delineate the patterns you see in the landscape? You may also want to view the segments over some of the other image layers by changing the image layer mixing or splitting your windows and viewing two different sets of data.



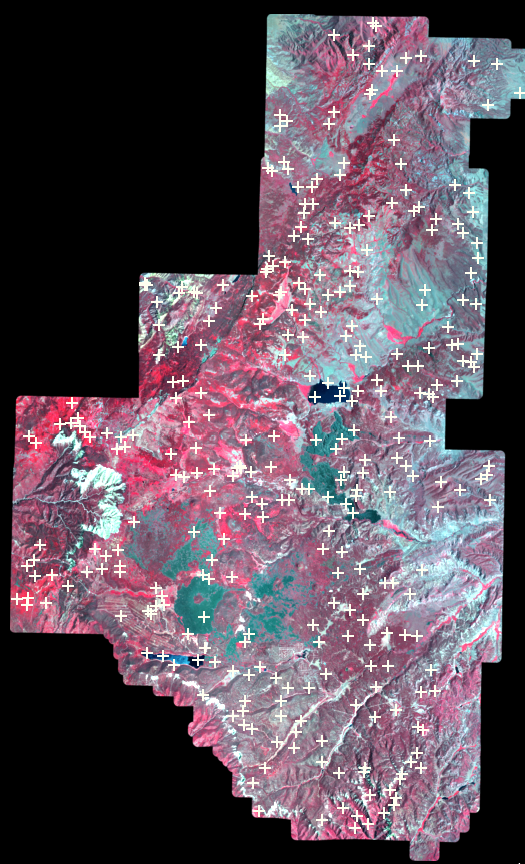
# Create Samples from Training Data

## Examine Training Data

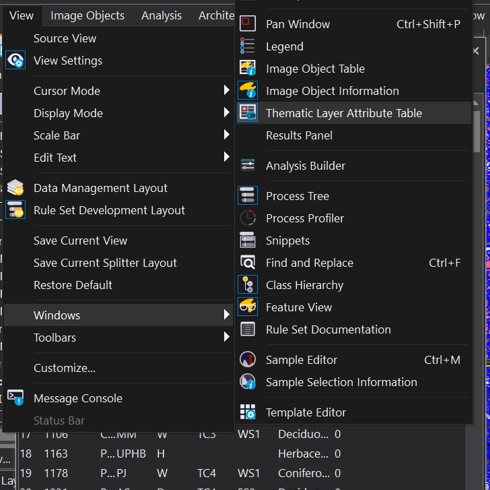
1. **Uncheck** one of the segment boxes within the **View Settings** window to **turn off** the segment boundaries.
2. To display the vector data, we will use to train our classifier, find the **Vector Layers** check box underneath the available layers within the **View Settings** window and make sure all boxes are check (see following graphic).



1. **Zoom** and **pan** to see the locations where field data were collected for this area. (See following graphic. Note, your reference data points might have a different default color system.)

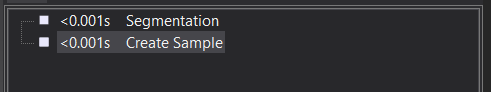


1. Now let’s open the Attribute table of our Reference Data. Go to **View**, then **Windows**, then **Thematic Layer Attribute Table** this will open the attribute table in a new panel in your project

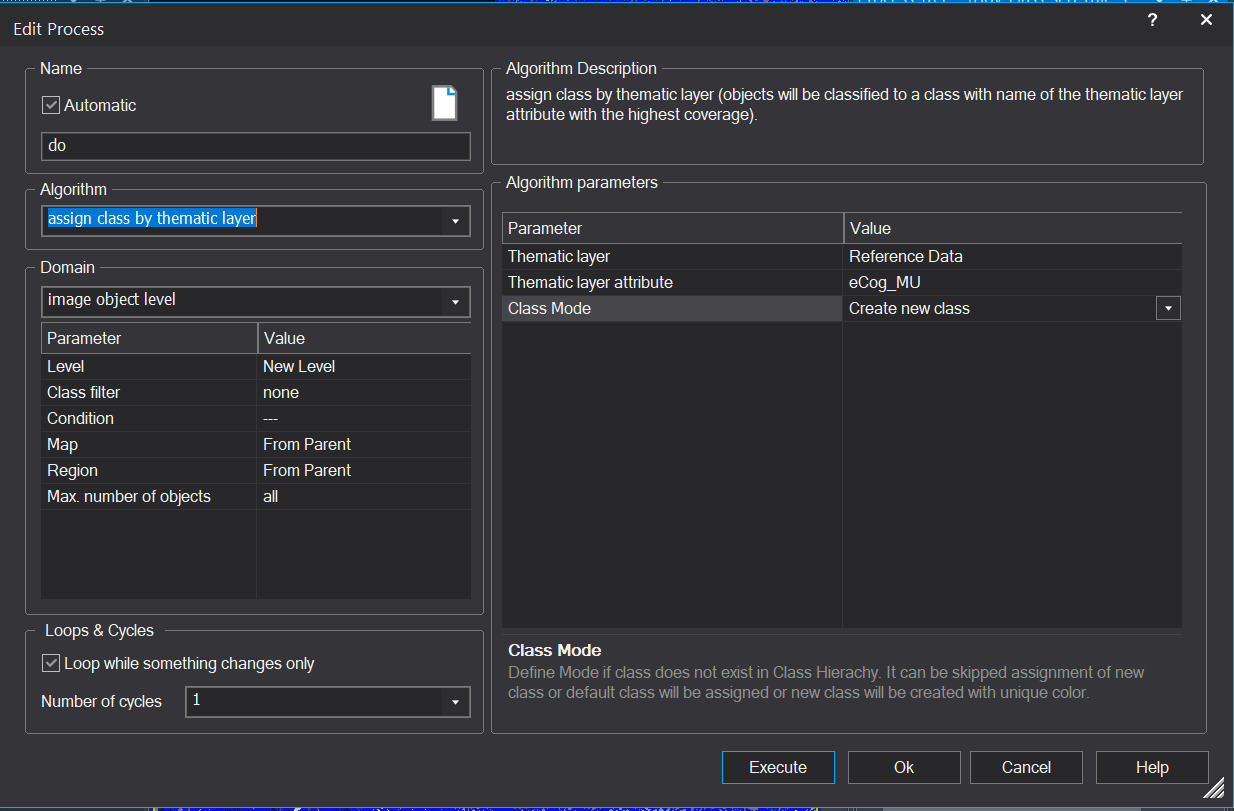


1. The field **eCog\_MU** contains each point’s class name
2. To view the record of a point, use the Normal Cursor (white arrow in main menu field of icons). Click on a point to highlight the record in the table.
3. Alternatively, you can use the Normal Cursor to highlight a record in the table and change the display of the associated data point in the map.
4. Close the table/Thematic Layer Attribute Table (click on the ‘x’ in the upper right hand corner of the table).
5. To turn the shapefile off, uncheck the box within the **View Settings** window that you previously checked to turn on the shapefile

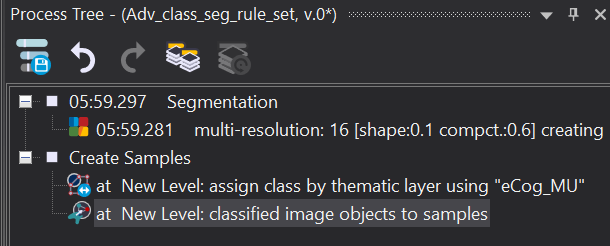
## Create Training Samples

1. Add a **new parent process** in the Process Tree by **right clicking** the **Segmentation** parent process and selecting **Add New Process**.
2. **Name** the process **Create Samples** (this new Parent process is appended at the same “level” as the Segmentation process). Then Click **Ok**
3. Right click on the **Create Samples** and select **Insert** a **child process** in the **Algorithm** field, select **assign class by thematic layer** **Thematic layer field: Reference Data**

* **Thematic layer attribute**: Select **eCog\_MU**.
* **Class Mode**: **Create new class**
* Click **OK** to close the dialog

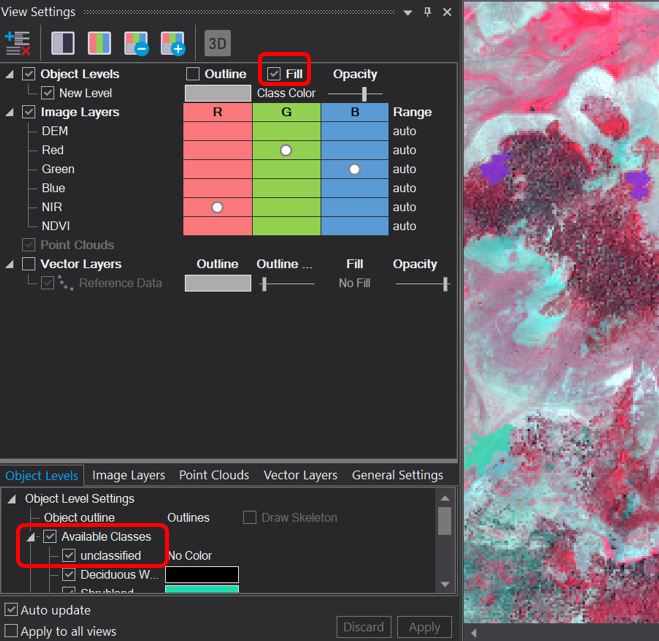


1. Right click on the **Assign class by thematic layer child process** and select **Add New Process**
2. For the **Algorithm** field select **classified image objects to samples**
3. Click **OK** to close the dialog box

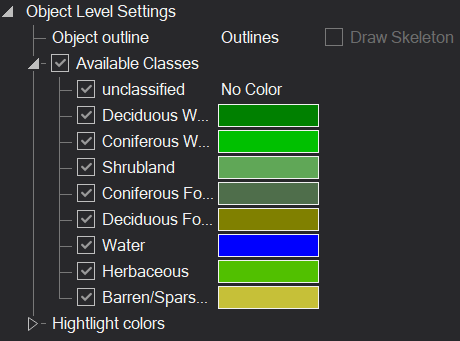


The ‘assign class by thematic layer’ process uses the thematic layer (i.e., **Dixie\_CCRD\_Reference\_Points.shp**) to classify the segments that the points intersect using the class specified in the shapefile attribute field **eCog\_MU**. When Create new class is selected for Class Mode, the software automatically creates classes (i.e., your Class Hierarchy gets created) with the names specified in the shapefile so you don’t have to create them manually. You can use either a point or polygon shapefile to create training samples.

1. If the **segments** are on,turn them **off** using the **View Settings** window
2. If the **reference points** are on, turn them **off** using the **View Settings** window
3. **Execute** the first child process,then **check** the boxes corresponding to the outlined checkboxs in the next image (**Fill** and **Available Classes**). You will notice that the first process classifies the segments that the shapefile points intersect and creates classes in the **Class Hierarchy window**



1. **(Optional):** Edit the classes and change the colors to make them more meaningful (e.g., Barren as tan, Water as blue, etc)



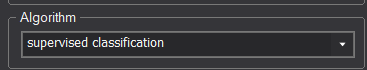
1. Execute the second process to creates samples from the classified image segments
2. There should be no difference in the view window when comparing your classified objects to your samples. Although there is no visual difference, eCognition treats classified objects and sample objects as two separate entities

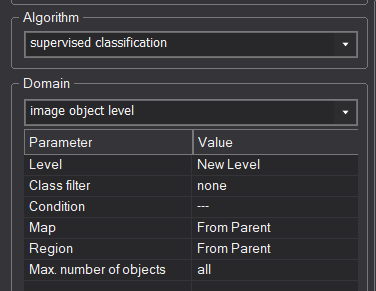
NOTE: When creating a vegetation map, the segments should ideally be created before the field data are collected. Field crews can take maps with the segments to the field and collect the field data to characterize the segments. When created after the field data are collected, the segments may or may not represent the observed conditions due to the variable size, shape, and positioning of the segments. You may want to examine the training samples created in this exercise to see if they appear to represent the assigned class. If not, you may want to eliminate some of the samples and possibly photo-interpret additional ones.

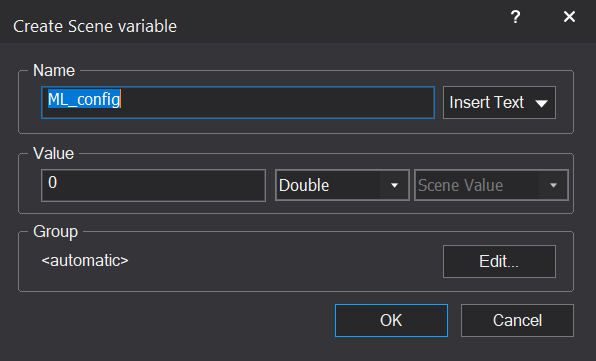
# Setup & Run a Classifier

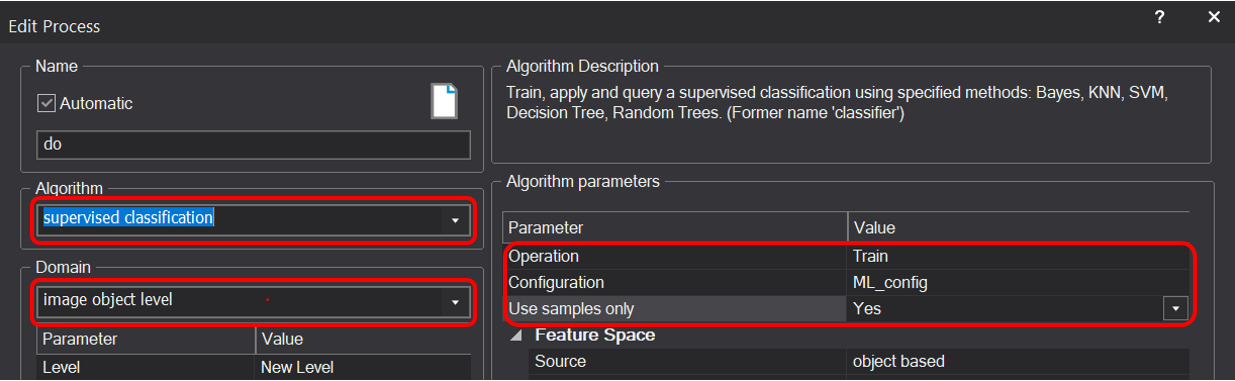
## Setup the ‘Training’ Process

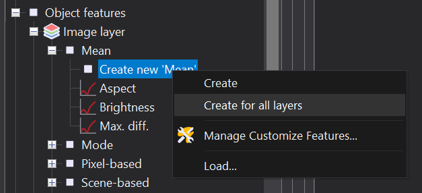
1. Create another parent process and name it **Supervised Classification**. If the process appears in the wrong sequence or at the wrong level in the process tree, practice what you learned from the Intro to eCognition course about dragging and dropping the processes to move it to the right location.
2. **Insert** a child process under the **Supervised Classification** parent process.
3. In the **Edit Process dialog**, select **Supervised Classification**



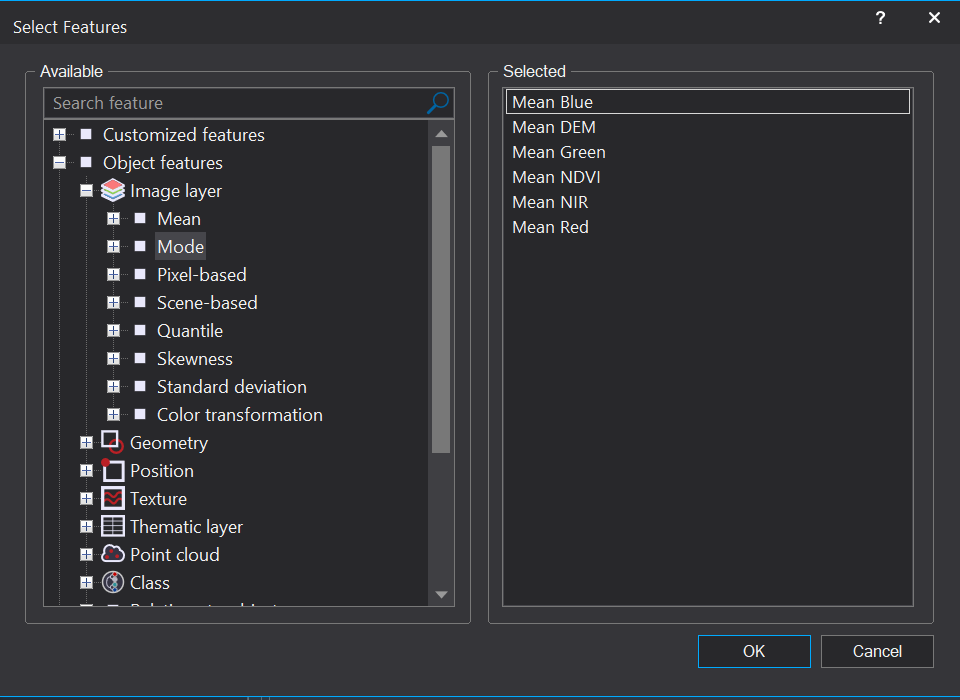
1. Under **Domain** (left side of the Edit Process dialog box), select **image object level** - this tells the classifier that we will classify objects rather than pixels
2. Select **Train** for the **Operation** field (right side of the dialog box). This setting tells the software to develop a rule or model using training or reference data. The rule will be applied later.
3. In the **Configuration** field under Algorithim Parameters, you will define a new variable that stores the model that the **Supervised Classification** algorithim develops. In this field, type “**ML\_config**” (any name will work, but for this exercise use ML\_config).
4. Hit **Enter** and the **Create Scene Variable dialog box** will appear to create this variable (see following graphic)



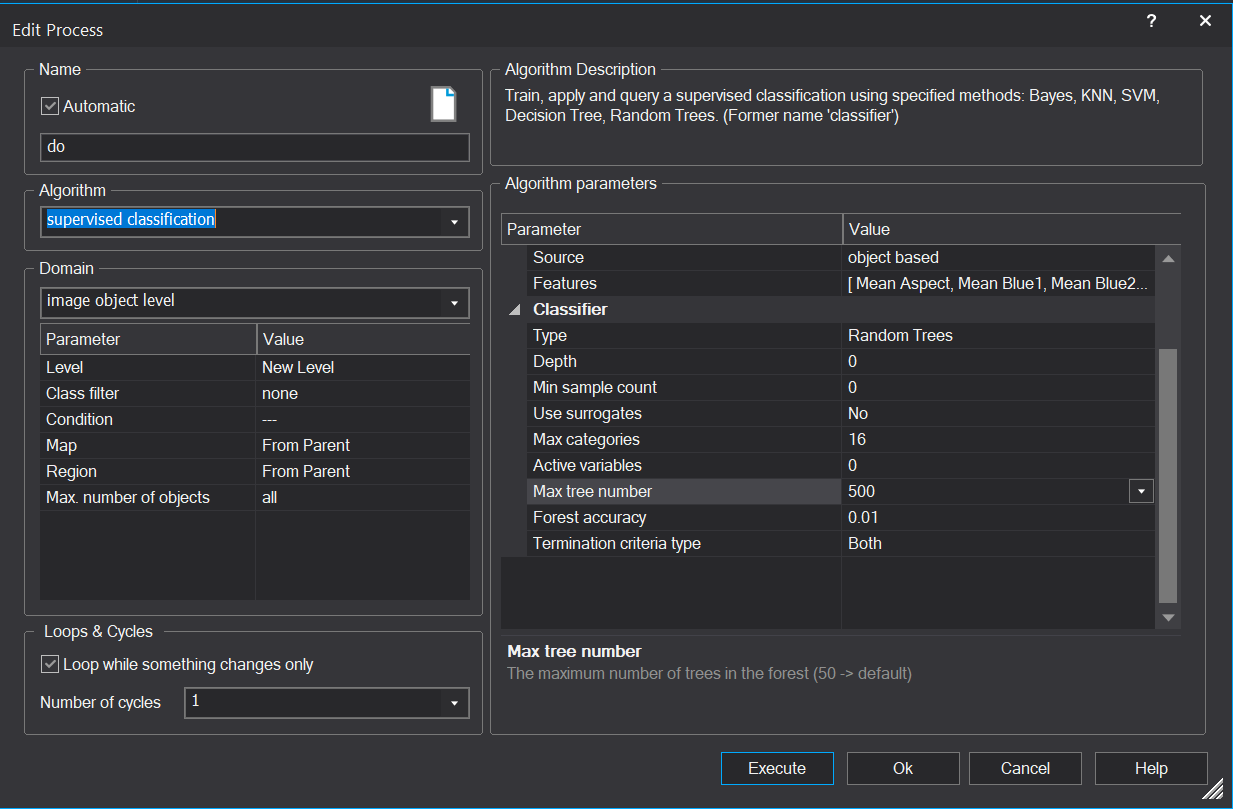
1. Leave the default settings and click **OK** to close it
2. Back in the **Edit Process** dialog select **Yes** in the **Use samples** **only** 
3. Under **Feature Space, Source** select **object based** if it is not already set as object-based
4. Place your cursor in the empty box to the right of **Features** in the **Feature Space**, use the drop down menu to **Select features…**.
5. After the **Select Features** dialog box opens, select which features you want the classifier to use to classify the image.
6. Navigate to **Object features**, then **Image Layer**, then **Mean.** Here you will select the various predictor layers that will be used to classify the area. However, the layers may not be visible yet (e.g., there should be one for each band or layer, such as, Aspect, Blue1, Blue2, DEM, etc.).
7. If you don’t see all of the layers you loaded, right click **Create new ‘Mean’** and select **Create for all layers** (see following graphic). All layers should now be visible



1. Next, *double click* **Mean** to move all the features to the **Selected** window on the right side of the dialog box. The Selected window shows the features that you have chosen to be used as predictor layers for the classification. (see following graphic).



1. In the **Selected** pane in the Select Features window, double clickonthe **Brightness** and **Max. diff**. features to remove each of them.
2. Click **OK** to close the dialog.
3. Back in the **Edit Process** dialog box, you will select which **classification algorithm** you would like to use. You will have the following options: Bayes (Bayesian), KNN (K-nearest neighbor), SVM (support vector machine), decision trees, and random trees (random forest).
   1. For each of the classifiers, there are different sets of parameters. For the purposes of this exercise, you can accept the default parameters.
4. Choose to the **Random Trees** algorithm, often called Random Forest, by selecting it under the **Classifier** then **Type,** your dialog box should now look like the following graphic:

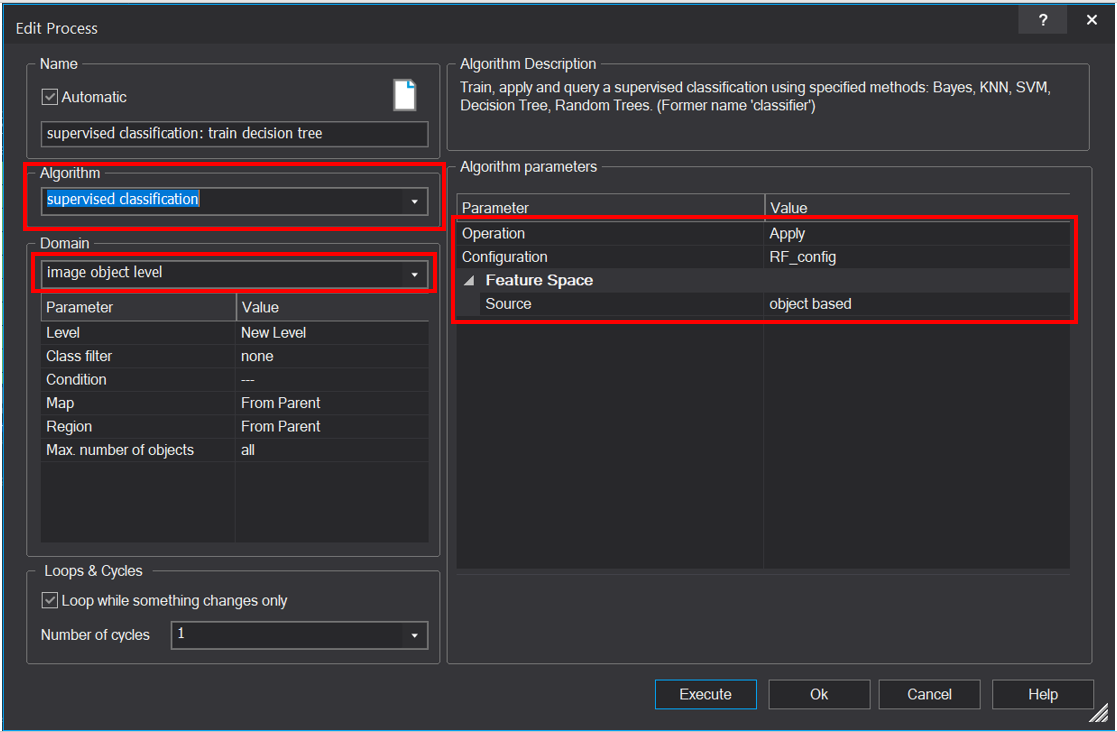


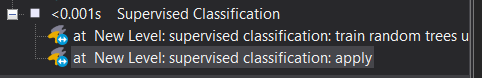
1. Click **OK** to close the dialog box.

Unlike the other classification processes in eCognition, which require just a single process to run, the advanced classifiers, such as random trees, require two Supervised Classification processes. The first Supervised Classification process is used to train the classifier (i.e., develop the classification model). The second one is used to apply the model to the imagery to create a classified map.

## Setup the Supervised Classification ‘Apply’ Process

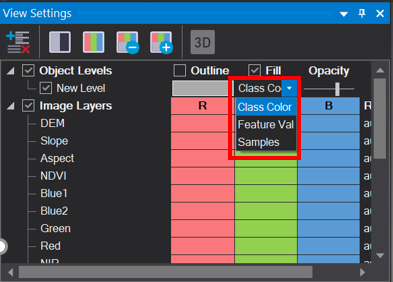
1. **Right click** on the child process at **New Level: supervised classification: train…** process and select **Add** **New Process** to add another child process to the **Supervised Classification** parent process (below the train classifier process).
2. In the **Edit Process** dialog, under **Algorithm** select the **Supervised Classification** process again and select **image object level** for the **Domain** (left side of the box).
3. On the right side of the dialog box for **Operation** select **Apply.** This setting tells the software to apply the model, saved as **ML\_config**, that was generated from the train random trees classifier specified in the previous process
4. In the **Configuration** field, select the variable that holds the model parameters you created in the previous process, **ML\_config**.
5. Select **object based** in the **Feature Space, Source field**



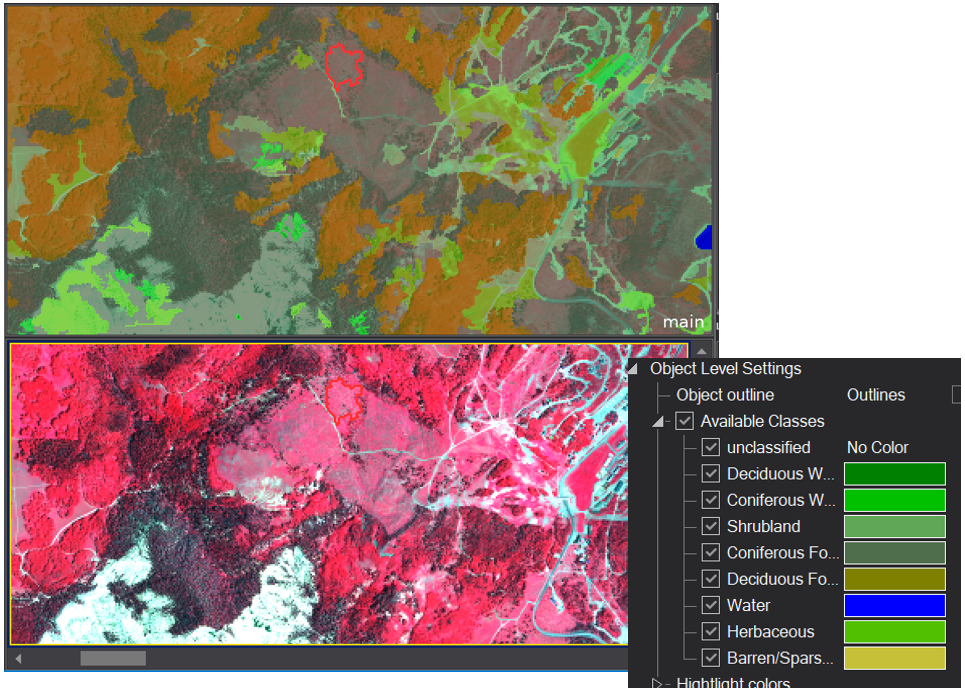
1. Click **OK** to close the dialog box.
2. You may need to click, drag, and drop one of the two processes that you just created so that they appear in the right order. The train process must appear first and the apply process second

# Run the Supervised Classification Process

1. **Right click** the Supervised Classification parent process and select **Execute** to run both the **train** and **apply** classifierrules.
2. Examine the classification results. You may need to change the drop-down menu within the View Settings window from **Sample** to **Class Color**



1. You may want to **split** the view window and **swipe** the classification over the imagery to examine how well the classifier performed. Hint:
2. Activate **Window, Split Horizontally** from the toolbar.
3. Change the view in the bottom window to just viewing the imagery and toggle.
4. Then activate **Window**, **Swipe View** from the toolbar.
5. Now you can move the divider line between the two views to swipe up and down.



1. **Save** your project

## Assess the Model Performance

In this step, you will assess how well the model classified the samples used to train it.

1. **Select Tools**, then **Accuracy Assessment**… from the main menu.
2. In the **Accuracy Assessment** dialog box, under the **Image object level** select **New Level**. This tells the software which image object level was classified. In this exercise, New Level is the only option since we only have one level of segments.

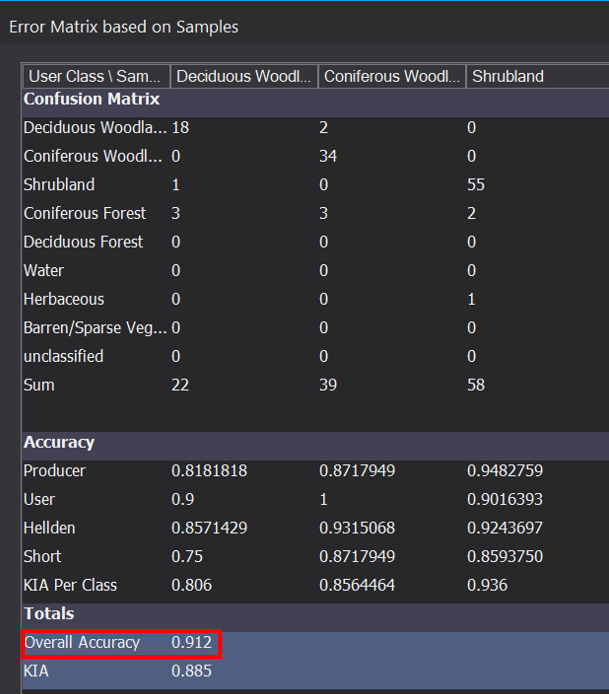
**Note:** The error matrix we are producing shows

**\* how well the classifier classified the samples used to train it\***.

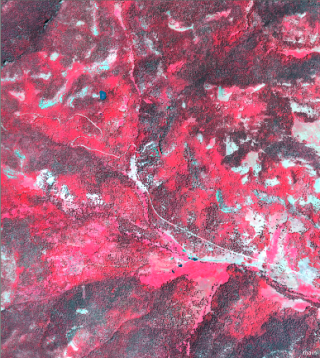
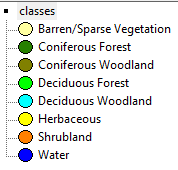
The overall accuracy will **almost always be higher for the training data than for an independent random sample** (a validation data set that is not used to train the classifier).

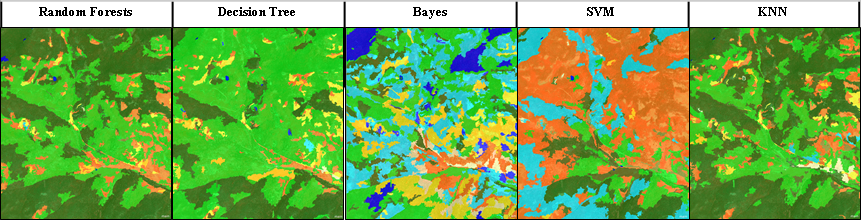
Be aware that this is NOT a true accuracy assessment because it is based on the training data (we will learn how to do a true accuracy assessment later in this exercise). A true accuracy assessment is based on a set of random samples not used to train the classifier. However, this matrix can be used as a preliminary indicator to assess output between different statistical models (random trees, Bayes, etc) in eCognition.

1. Next, select **Error Matrix based on Samples** in the **Statistic** type field
2. Click the **Show Statistics** button to show the error matrix
3. Find the **Overall accuracy** located near the bottom left of the matrix, given the stochastic (random) nature of the modeling process, and depending on the algorithm you choose – your overall accuracy number may not match the numbers below. The Overall Accuracy tells the level of agreement between all the samples in the project and their respective classification. The error matrix also shows the level of agreement for individual classes and reports traditional producer’s and user’s accuracies.

It is important to understand that none of the classifiers will always yield the most accurate results. Some classifiers will perform better with certain data sets than others. It is best to experiment with different classifiers to find one that works well for your data.





Also, be aware that the parameters you set for a classifier will have a significant impact on how well it performs. If you think the model did not perform well, try removing some of the predictor layers. For example, try doing the classification using only Landsat bands. You can also try changing some of the parameters to see if you get better results.

Unfortunately, explaining how the classifiers work and how to set the parameters is beyond the scope of this training. However, if image classification is a routine part of your job, we strongly recommend that you learn how the classifiers work, whether through online information, a textbook, or advanced training, so that you understand how to set the parameters.

# Perform a ‘True’ Accuracy Assessment

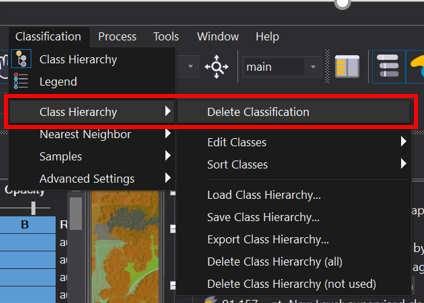
## Load a Validation Shapefile

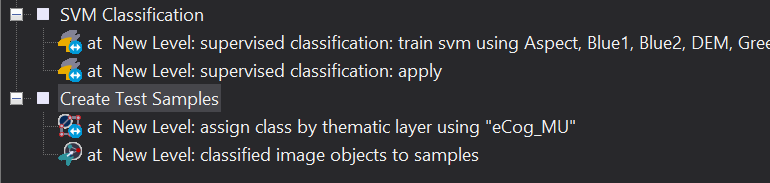
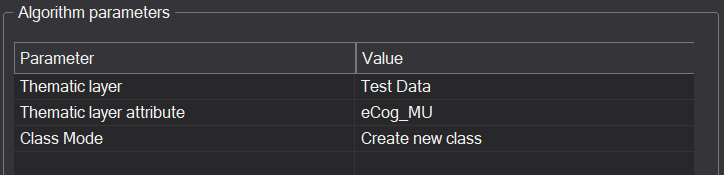
1. From the main menu bar, **click File, Modify Open Project**… to launch the Modify Project dialog box
2. **Click** the **Insert button** next to the **thematic layer box** (below the image layer box)
3. Navigate to the course data shapefile directory (\USFWS\_eCog\_workshop\Shapefile) and select **Dixie\_CCRD\_Validation\_Points.shp**
4. Click **OK**.
5. Double click **“Thematic Layer 2”** which was just added and change the layer alias to **Test Data**.
6. Click **OK** to close the Layer Properties dialog. There will now be two shapefiles loaded in the thematic layer box: **Reference Data** and **Test Data**.
7. Click **OK** to close the Create Project dialog box.

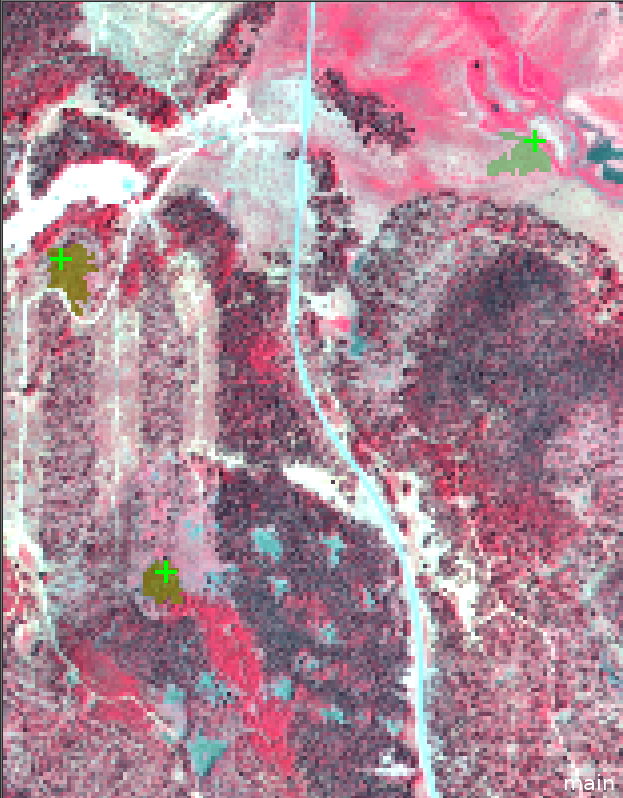
The shapefile Dixie\_CCRD\_Validation\_Points.shp represents a third of all the data that were collected for the study area. Two thirds were used earlier in this exercise to classify the image and the remaining third of the data were reserved for an accuracy assessment. Although the data collected for the accuracy assessment were not collected in an entirely random fashion, they will provide a much better estimate of the true accuracy of the classification than the samples used to train the classifier. To determine the true accuracy of a classified map, the test data should be collected randomly.

## Create Test Samples from the Test Data Shapefile

1. From the main menu located at the top of the main eCognition window, click **Classification**, **Class Hierarchy, Delete Classification** (see graphic below) to launch the Delete Classification dialog. **IMPORTANT:** You must select the **second Class Hierarchy** option in the **Classification** menu. If you clicked the first option, the **Class Hierarchy** window that shows your classes will have disappeared. If this is the case, click the first Class Hierarchy option again to restore it, then, go back into the menu and click the second Class Hierarchy option as instructed above.



1. In the **Delete Classification** dialog that appears, click **Delete** to delete the classification.
2. From the main menu, click **Classification, Samples, Delete All Samples**. Click **Yes** in the popup window to delete the existing samples.
3. **Right click** on the **Create Samples** parent process in the **Process Tree** and select **Copy**
4. **Right click** on the last parent process in the Process Tree (probably the Supervised Classification parent process) and select **Paste** to paste a copy of the Create Samples processes at the bottom of the Process Tree (see following graphic).
5. 
6. Double click the Create Samples parent process. In the Edit Process dialog box, rename this process to **Create Test Samples**.
7. Double click the assign class by thematic layer child process to launch the Edit Process dialog box.
8. **Under** the Algorithm parameters section of the box (right hand side), select **Test Data** for the Thematic layer field (see following graphic).
9. 
10. **Accept** all other values and click **OK** to close the dialog box.
11. **Turn** the segment boundaries off using the **View Settings** window check boxes if they are not already turned off.
12. **Right click** the **Create Test Samples** parent process and select **Execute** to run the two child processes.
13. **View** the **Classification** and **Samples** from the View Settings window to see sample segments intersected by the Test Data shapefile (see following graphic).



1. From the main menu, **click Classification, Class Hierarchy, Delete Classification** to launch the **Delete Classification** dialog. Again, be sure to select the second Class Hierarchy option in the Classification menu.
2. In the **Delete Classification** dialog that appears, click **Delete** to delete the classification created by the **Test Data samples**.
3. **Right click** on the **apply** process of the supervised classification process and select **Execute** (see graphic below). **IMPORTANT**: Do not execute the **parent or train** process- only the **apply** process.
4. **View** the classification by checking the appropriate check boxes within the **View Settings** window.

By executing just the apply process of your classifier, you are using the model that was created last time that used the reference points. If you were to execute the parent process, the software would create a new classification model based on the Test Data, which you do not want.

1. Select **Tools**, **Accuracy Assessment…** from the main menu.
2. In the Accuracy Assessment dialog box, select **New Level** in the **Image object level field** (This tells the software which image object level was classified. In this exercise, New Level is the only option).
3. Next, select Error Matrix based on Samples in the Statistics type field.
4. **Click** the **Show Statistics** button to show the error matrix.
5. **Scroll** to the bottom of the **error matrix**. You will notice that the overall accuracy is lower than that which was reported when you assessed the model performance using the training samples.
6. **Click Close** to close the error matrix.
7. **Click Close** to close the Accuracy Assessment dialog box.

# Save the Project and Rule Set for Later Use

1. **Save** your project (Hint: **File, Save Project**)
2. The rule set you have created can be reused for other image files or simply saved for reference. **Right click** in the Process Tree and select **Save Rule Set**.
3. In the **Save Process dialog**, navigate to a location where you would like to save the rule set. Give the rule set a name and click **Save**.

When you want to use the rule set you saved, simply add your image data then add the rule set into the process tree. Caution: if the image and vector layers and aliases are not the same as those used when the rule set was created, you will need to open any processes that reference the image or vector layers (e.g., multiresolution segmentation and the classifier train processes) and change them to use the proper aliases. When creating new project files and adding old rule sets, sometimes unused aliases will appear that were used by the rule set, but not by your current image and/or vector layers. You can delete these by going to Process, Edit Aliases, Image Layer Aliases… or Process, Edit Aliases, Thematic Layer Aliases… and removing the unused aliases.

**Congratulations**, you have completed the Advanced Classification in eCognition exercise!

**Module developed by:**

Author, Author’s agency

Last name first initial middle initial. 20XX. Module title. Module ## in Author 1 (last name first initial middle initial i..e. Smith JC), Author 2, Author 3, eds. Course title, 20xx edition. U.S. Fish and Wildlife Service, National Conservation Training Center, Shepherdstown, West Virginia, USA.