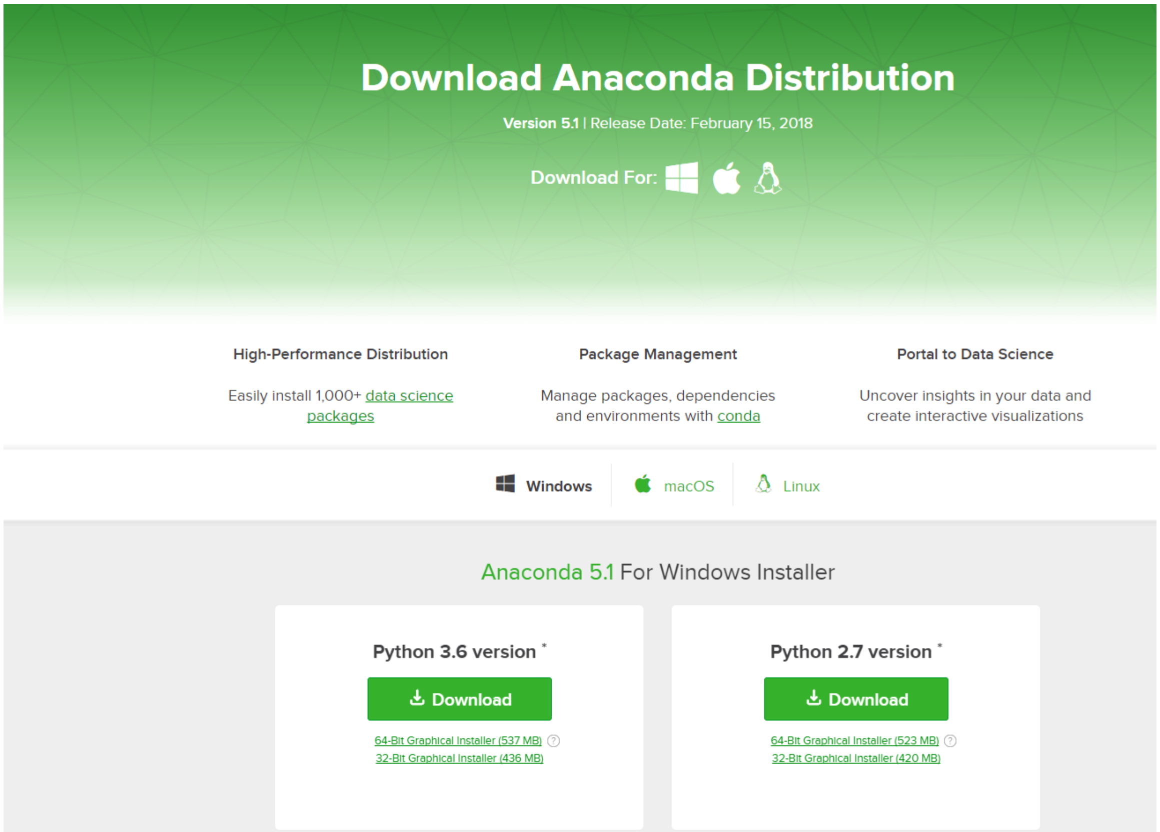
PriSM toolbox: user manual (Windows)

Written by Daniel Buscombe, Daniel.Buscombe@nau.edu. Version 0.1. February 2018

Download and install Python

We recommend using the Anaconda Python distribution because it ships with a lot of libraries that PriSM uses. Anaconda is free, available for all major operating systems, 32 and 64 bit, lots of different versions of Python. Head to www.anaconda.com/download and download the installer.



Download Anaconda Distribution

Version 5.1 | Release Date: February 15, 2018

High-Performance Distribution

Easily install 1,000+ data science packages

Package Management

Manage packages, dependencies and environments with conda

Portal to Data Science

Uncover insights in your data and create interactive visualizations

Anaconda 5.1 For Windows Installer

Python 3.6 version *

64-Bit Graphical Installer (537 MB)
32-Bit Graphical Installer (436 MB)

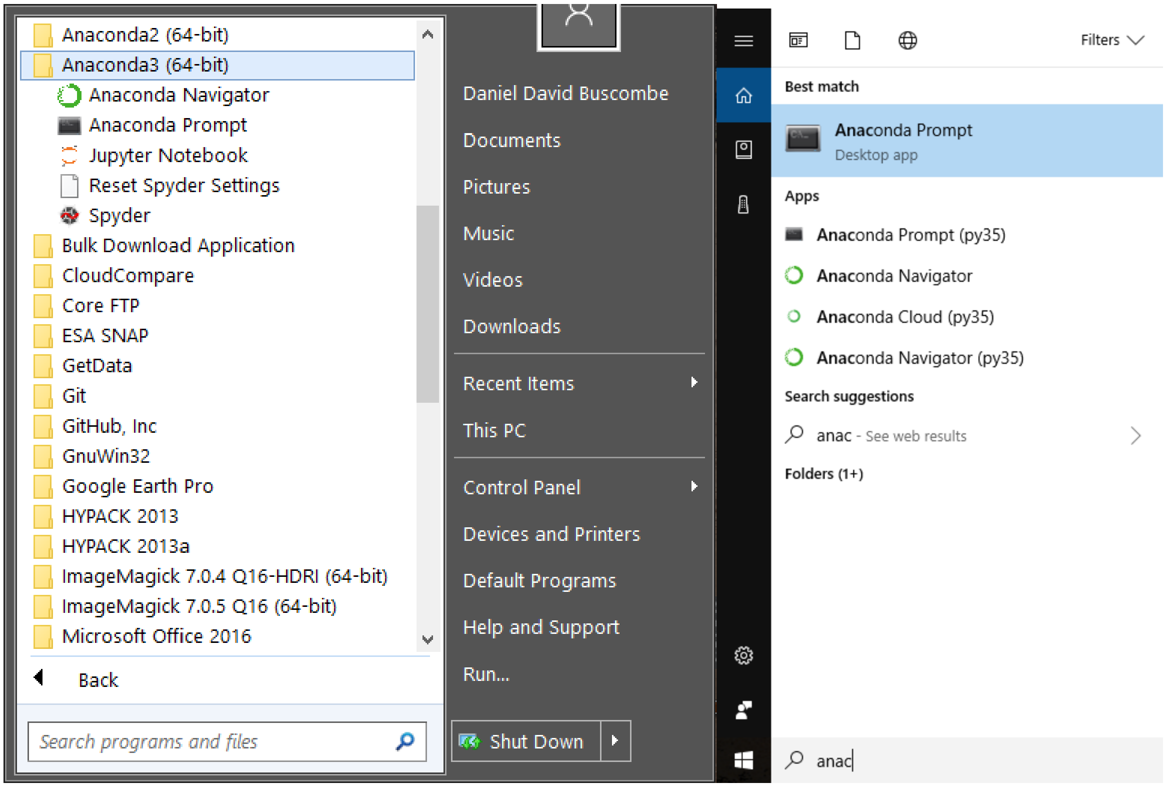
Python 2.7 version *

64-Bit Graphical Installer (523 MB)
32-Bit Graphical Installer (420 MB)

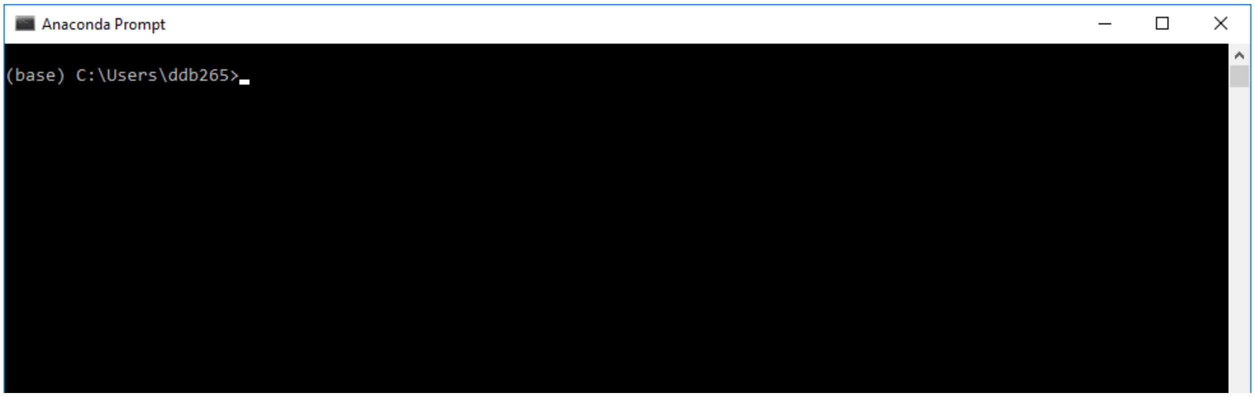
We recommend that you use Python 3. As of Feb. 19 2018, Python 3.6 is the latest version available from Anaconda. Go ahead and install that. For more experienced Python users who may prefer Python 2, you may use PriSM with versions of Python 2 greater than 2.7. For everyone else, Python 3 is the way to go.

Launch python

From the search or start menus, find Anaconda. Anaconda3 refers to Python 3. Anaconda2 refers to Python 2. You may install them both and use either. Launch the 'Anaconda prompt'



Below is what the Anaconda prompt looks like. It's like a regular command prompt and has much of the same functionality



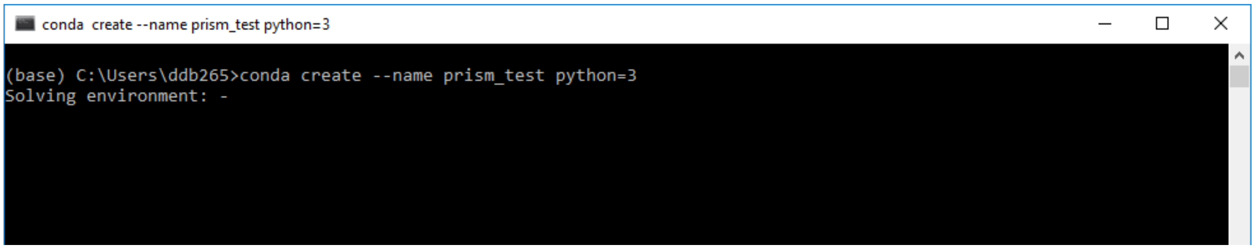
```
(base) C:\Users\ddb265>_
```

Using PriSM in a Python virtual environment

We recommend that you try the PriSM toolbox out first using a conda virtual environment (https://conda.io/docs/user-guide/tasks/manage-environments.html). This is a safe way to run python programs if you already have python installed on your computer. It is also a good way to make sure all the libraries you'll need to install to use PriSM play nicely with each other.

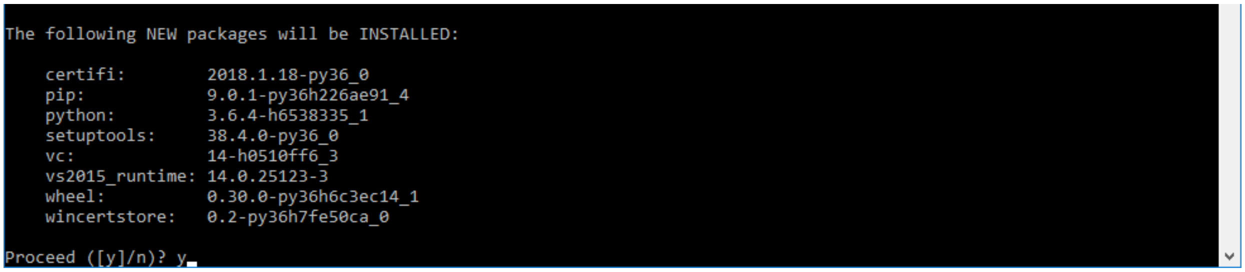
1. Setting up the virtual environment

You may call this environment whatever you like. Below, I called my environment ‘prism_test’. Write the following and press enter



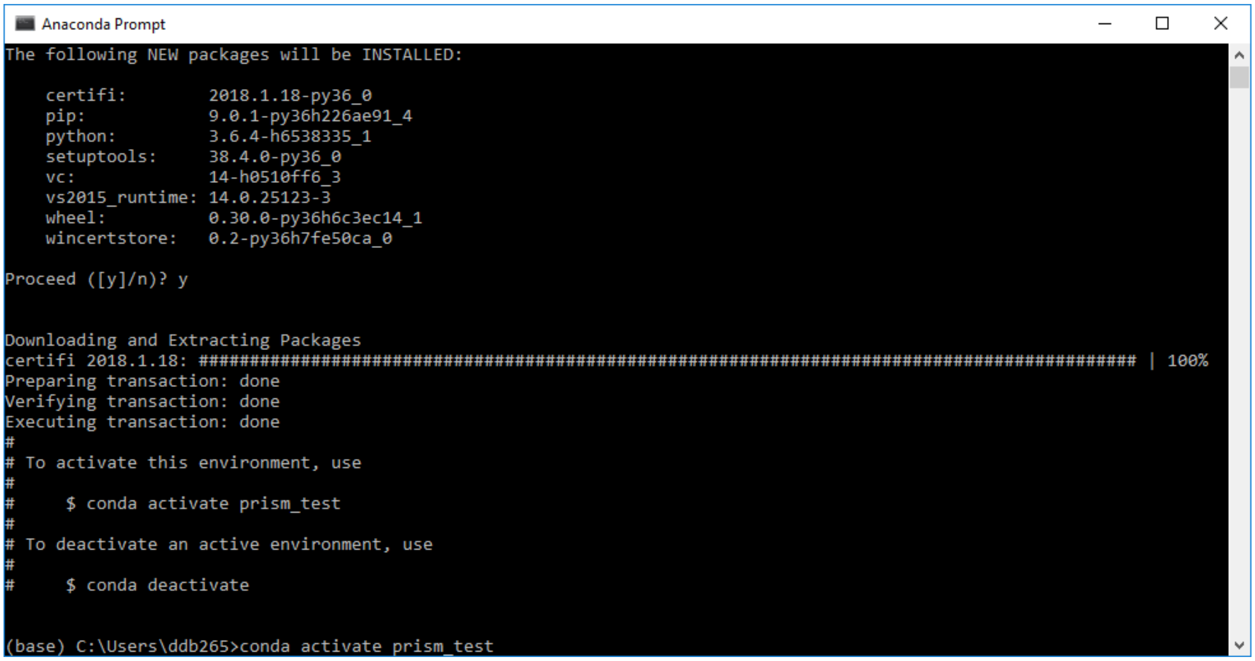
```
(base) C:\Users\ddb265>conda create --name prism_test python=3
Solving environment: -
```

When it asks you to proceed, hit enter (or type y, then press enter)



```
The following NEW packages will be INSTALLED:

    certifi:        2018.1.18-py36_0
    pip:            9.0.1-py36h226ae91_4
    python:         3.6.4-h6538335_1
    setuptools:     38.4.0-py36_0
    vc:             14-h0510ff6_3
    vs2015_runtime: 14.0.25123-3
    wheel:          0.30.0-py36h6c3ec14_1
    wincertstore:   0.2-py36h7fe50ca_0

Proceed ([y]/n)? y
```

Next you need to activate the environment: it tells you what to do (in my case ‘conda activate prism_test’ [Enter])

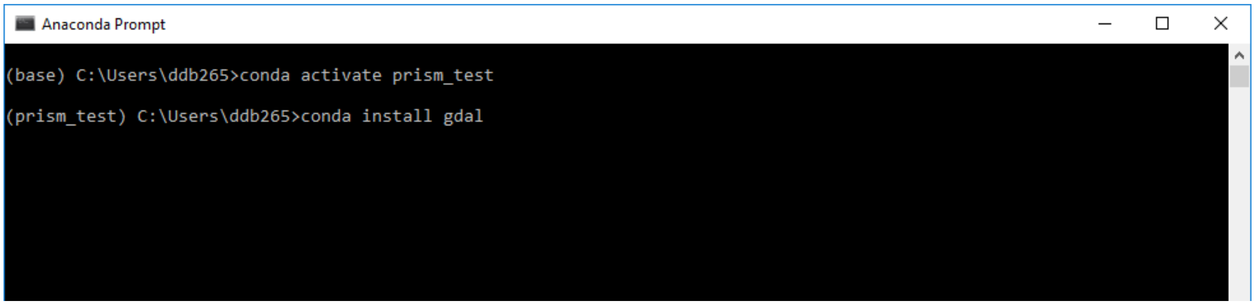


```
The following NEW packages will be INSTALLED:

    certifi:        2018.1.18-py36_0
    pip:            9.0.1-py36h226ae91_4
    python:         3.6.4-h6538335_1
    setuptools:     38.4.0-py36_0
    vc:             14-h0510ff6_3
    vs2015_runtime: 14.0.25123-3
    wheel:          0.30.0-py36h6c3ec14_1
    wincertstore:   0.2-py36h7fe50ca_0

Proceed ([y]/n)? y


Downloading and Extracting Packages
certifi 2018.1.18: ############################################################################################ | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate prism_test
#
# To deactivate an active environment, use
#
#     $ conda deactivate


(base) C:\Users\ddb265>conda activate prism_test
```

2. Installing required libraries

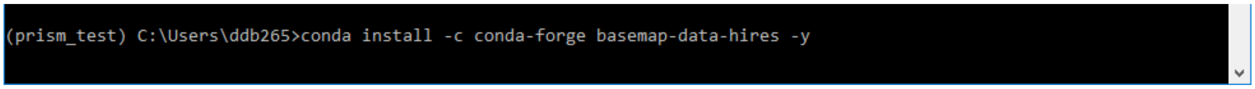
First you need to install the libraries that PriSM needs. There is quite a few, mostly providing us with geospatial and plotting routines but also some related to machine learning and other numerical computations. We'll use conda (https://conda.io/docs/) and pip (https://en.wikipedia.org/wiki/Pip_(package_manager)) to install what we need automatically

1. GDAL (http://www.gdal.org/) is first (conda install gdal [Enter], then hit [Enter] again when it asks you if you'd like to proceed). This takes a few minutes to install



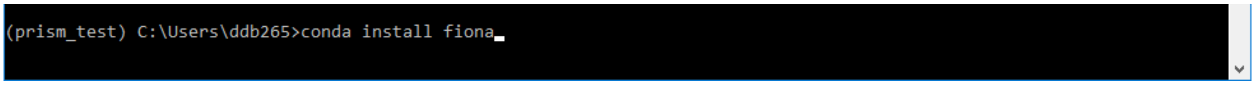
```
(base) C:\Users\ddb265>conda activate prism_test

(prism_test) C:\Users\ddb265>conda install gdal
```

2. basemap (https://matplotlib.org/basemap/)



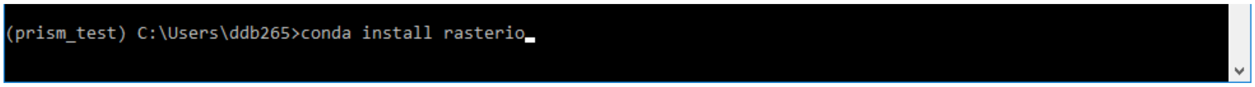
```
(prism_test) C:\Users\ddb265>conda install -c conda-forge basemap-data-hires -y
```

3. Fiona (http://toblerity.org/fiona/)



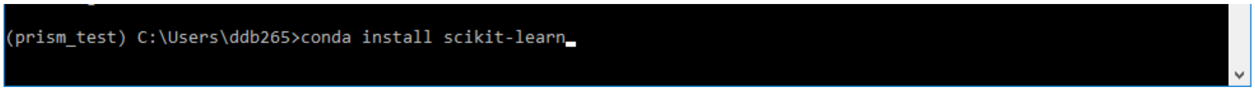
```
(prism_test) C:\Users\ddb265>conda install fiona
```

4. rasterio (https://mapbox.github.io/rasterio/)



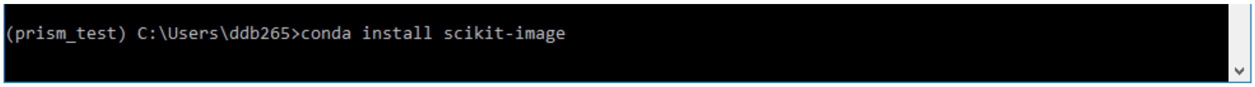
```
(prism_test) C:\Users\ddb265>conda install rasterio
```

5. scikit-learn (http://scikit-learn.org/stable/)



```
(prism_test) C:\Users\ddb265>conda install scikit-learn
```

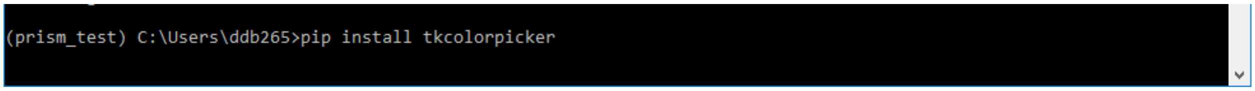
6. scikit-image (http://scikit-image.org/)



```
(prism_test) C:\Users\ddb265>conda install scikit-image
```

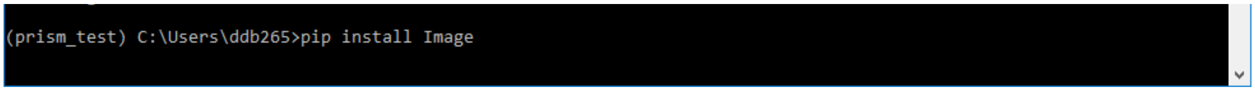
Finally, we install four libraries using pip.

First, tkcolorpicker (https://pypi.python.org/pypi/tkcolorpicker)



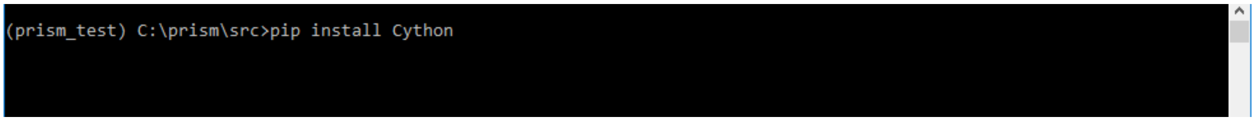
```
(prism_test) C:\Users\ddb265>pip install tkcolorpicker
```

Then, the Imaging library (https://pillow.readthedocs.io/en/latest/)



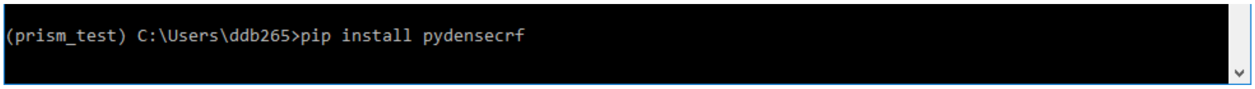
```
(prism_test) C:\Users\ddb265>pip install Image
```

Then Cython (http://cython.org/)



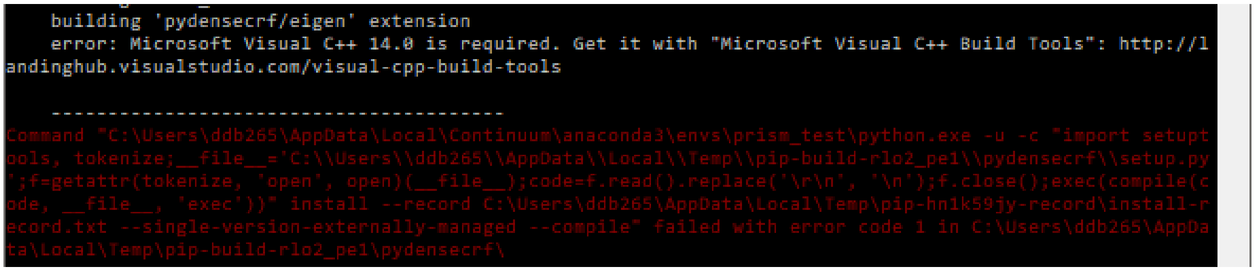
```
(prism_test) C:\prism\src>pip install Cython
```

Finally, pydensecrf (https://github.com/lucasb-eyer/pydensecrf)



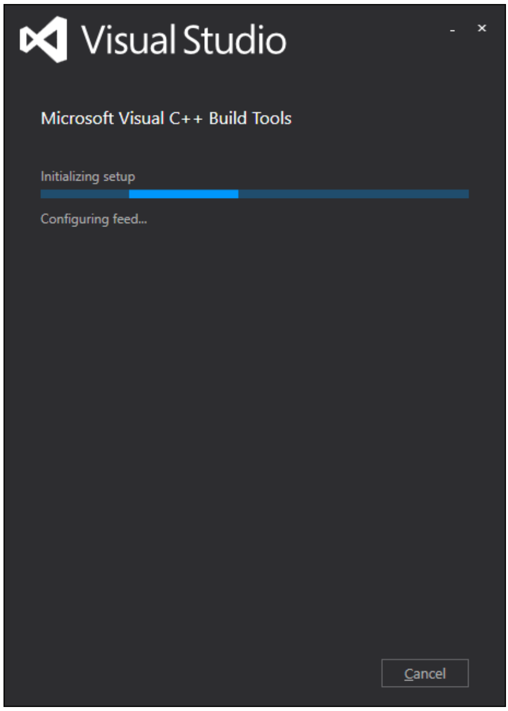
```
(prism_test) C:\Users\ddb265>pip install pydensecrf
```

If you get an error that looks like this:



```
    building 'pydensecrf/eigen' extension
    error: Microsoft Visual C++ 14.0 is required. Get it with "Microsoft Visual C++ Build Tools": http://l
andinghub.visualstudio.com/visual-cpp-build-tools

    ----------------------------------------
Command "C:\Users\ddb265\AppData\Local\Continuum\anaconda3\envs\prism_test\python.exe -u -c "import setupt
ools, tokenize;__file__='C:\\Users\\ddb265\\AppData\\Local\\Temp\\pip-build-rlo2_pe1\\pydensecrf\\setup.py
';f=getattr(tokenize, 'open', open)(__file__);code=f.read().replace('\r\n', '\n');f.close();exec(compile(c
ode, __file__, 'exec'))" install --record C:\Users\ddb265\AppData\Local\Temp\pip-hn1k59jy-record\install-r
ecord.txt --single-version-externally-managed --compile" failed with error code 1 in C:\Users\ddb265\AppDa
ta\Local\Temp\pip-build-rlo2_pe1\pydensecrf\
```

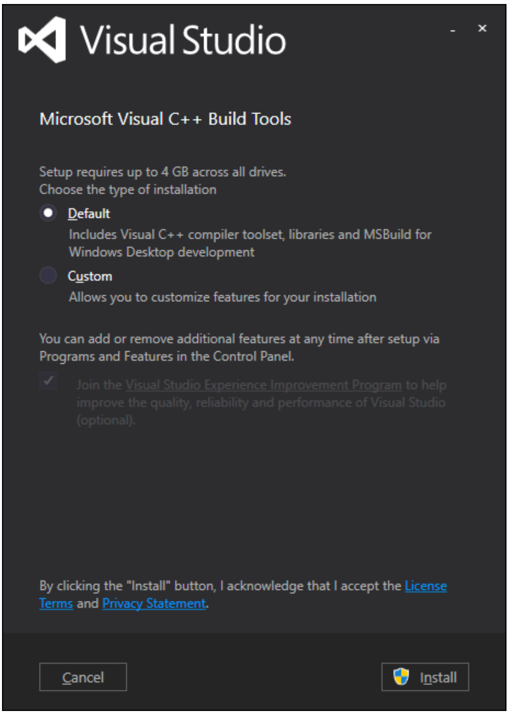
This means you'll need to install a C++ compiler. First, go here: http://landinghub.visualstudio.com/visual-cpp-build-tools and download Visual C++ 2015 Build Tools



Microsoft Visual C++ Build Tools

Initializing setup

Configuring feed...



Microsoft Visual C++ Build Tools

Setup requires up to 4 GB across all drives.
Choose the type of installation

● Default
Includes Visual C++ compiler toolset, libraries and MSBuild for Windows Desktop development

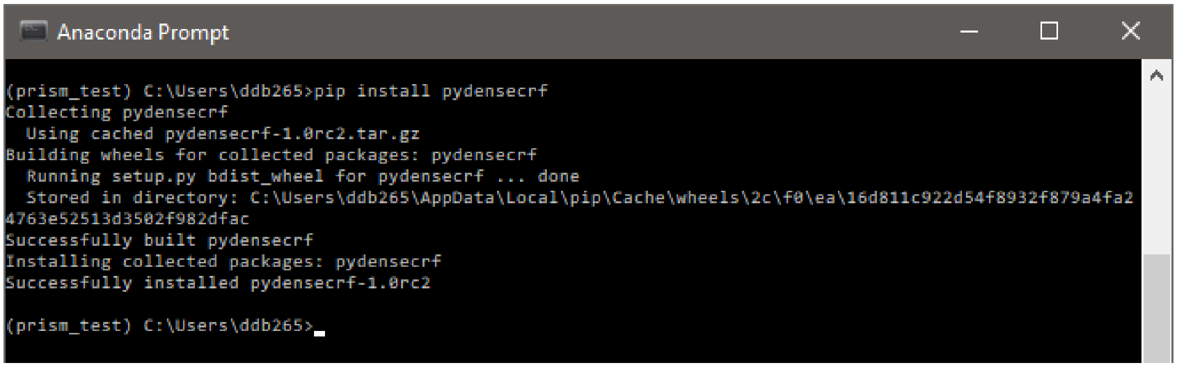
○ Custom
Allows you to customize features for your installation

You can add or remove additional features at any time after setup via Programs and Features in the Control Panel.

☑ Join the Visual Studio Experience Improvement Program to help improve the quality, reliability and performance of Visual Studio (optional).

By clicking the "Install" button, I acknowledge that I accept the License Terms and Privacy Statement.

When it is finished, try to install pydensecrf again – it should now work

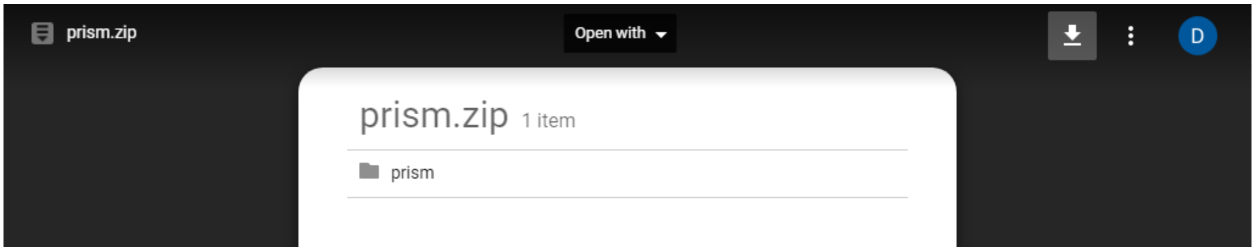


```
(prism_test) C:\Users\ddb265>pip install pydensecrf
Collecting pydensecrf
  Using cached pydensecrf-1.0rc2.tar.gz
Building wheels for collected packages: pydensecrf
  Running setup.py bdist_wheel for pydensecrf ... done
  Stored in directory: C:\Users\ddb265\AppData\Local\pip\Cache\wheels\2c\f0\ea\16d811c922d54f8932f879a4fa2
4763e52513d3502f982dfac
Successfully built pydensecrf
Installing collected packages: pydensecrf
Successfully installed pydensecrf-1.0rc2

(prism_test) C:\Users\ddb265>_
```

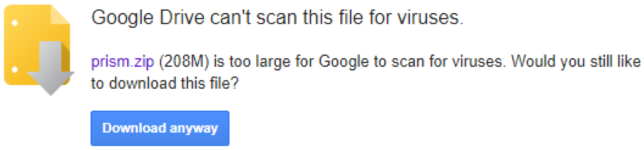
Ok, that should be all the libraries you need. Now you're ready to download the toolbox and start using it.

3. Download PriSM

Go to www.danielbuscombe.com/prism. Hit 'Download program', then "download" (downward arrow)



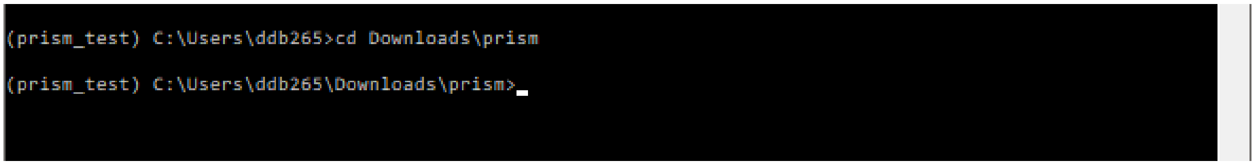
Then 'download anyway'



Google Drive can't scan this file for viruses.

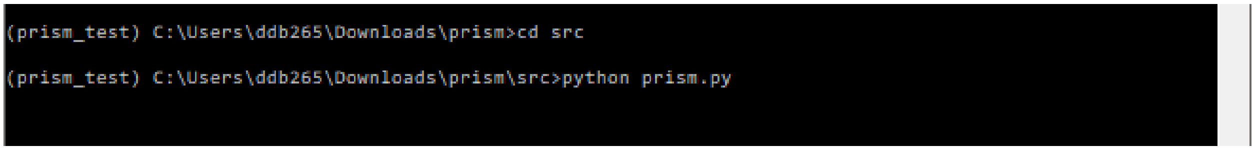
prism.zip (208M) is too large for Google to scan for viruses. Would you still like to download this file?

Unzip / extract to somewhere on your filesystem. Navigate there from the Anaconda prompt using the 'cd' (change directory) command



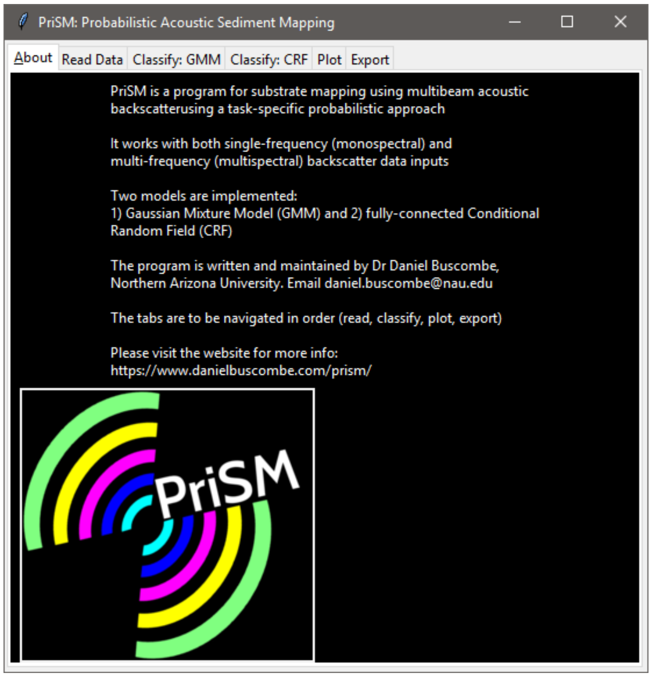
```
(prism_test) C:\Users\ddb265>cd Downloads\prism

(prism_test) C:\Users\ddb265\Downloads\prism>_
```

Navigate to the src directory. Now you can run the program by typing: python prism.py



```
(prism_test) C:\Users\ddb265\Downloads\prism>cd src

(prism_test) C:\Users\ddb265\Downloads\prism\src>python prism.py
```

And you should see this screen after a few moments



PriSM is a program for substrate mapping using multibeam acoustic backscatteringusing a task-specific probabilistic approach

It works with both single-frequency (monospectral) and multi-frequency (multispectral) backscatter data inputs

Two models are implemented:
1) Gaussian Mixture Model (GMM) and 2) fully-connected Conditional Random Field (CRF)

The program is written and maintained by Dr Daniel Buscombe, Northern Arizona University. Email daniel.buscombe@nau.edu

The tabs are to be navigated in order (read, classify, plot, export)

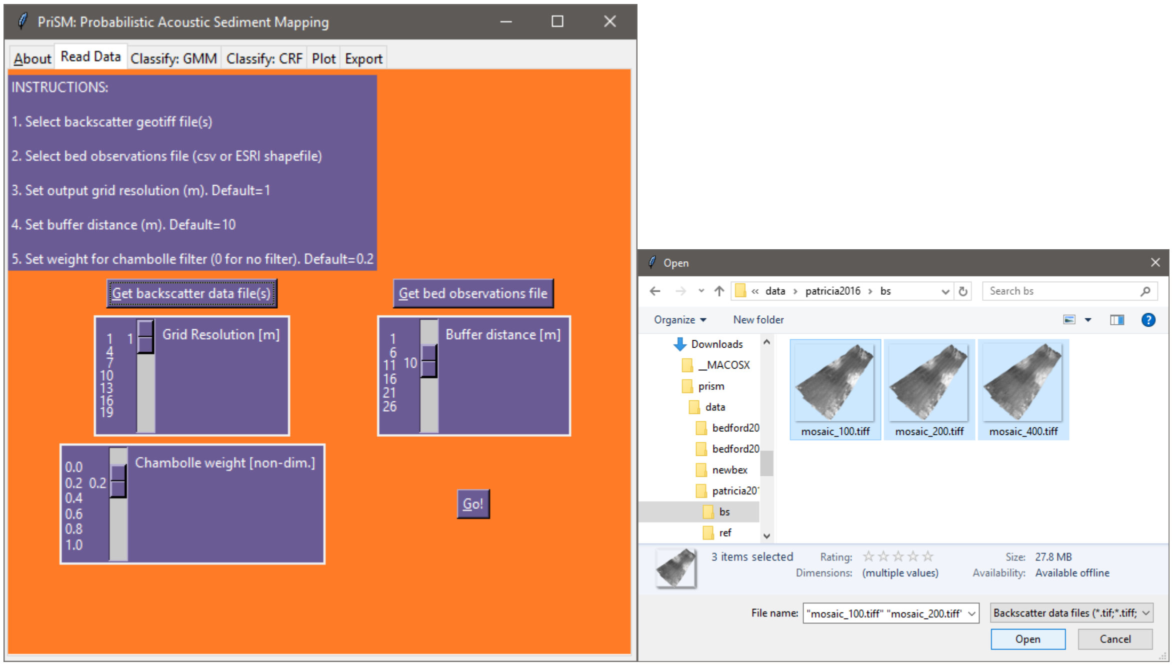
Please visit the website for more info:
https://www.danielbuscombe.com/prism/

4. Running the program: Quickstart guide

We will demonstrate the functionality of the program using the Patricia Bay dataset shipped with the program. All data were collected by R2Sonic for use with the R2Sonic 2017 Multispectral Backscatter competition.

Read module

From the ‘Read Data’ tab, select backscatter data by hitting the ‘Get backscatter data file(s)’ button, then navigate to the data directory → Patrcia2016 → select all three tiff files → Open



INSTRUCTIONS:

1. Select backscatter geotiff file(s)

2. Select bed observations file (csv or ESRI shapefile)

3. Set output grid resolution (m). Default=1

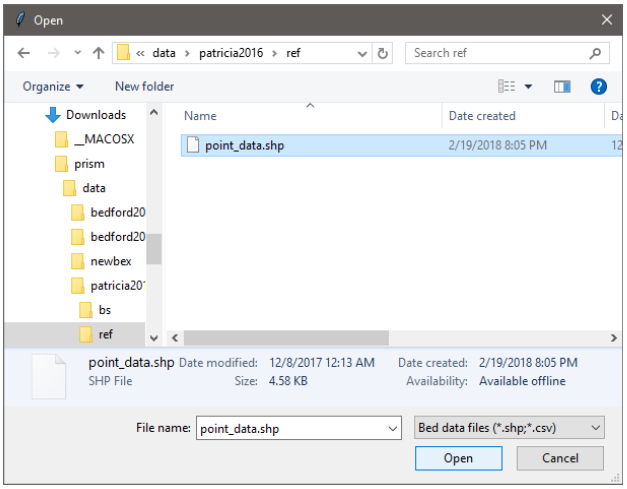
4. Set buffer distance (m). Default=10

5. Set weight for chambolle filter (0 for no filter). Default=0.2

3 items selected Rating: ☆☆☆☆☆ Size: 27.8 MB
Dimensions: (multiple values) Availability: Available offline

File name: "mosaic_100.tiff" "mosaic_200.tiff"
Backscatter data files (*.tif;*.tiff;

Do the same for the 'bed observations file' → select the shapefile in the 'ref' directory → Open

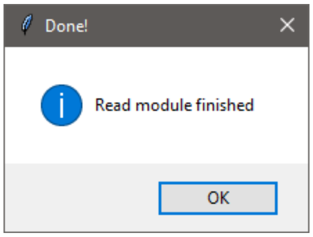


point_data.shp 2/19/2018 8:05 PM

point_data.shp Date modified: 12/8/2017 12:13 AM Date created: 2/19/2018 8:05 PM
SHP File Size: 4.58 KB Availability: Available offline

File name: point_data.shp
Bed data files (*.shp;*.csv)

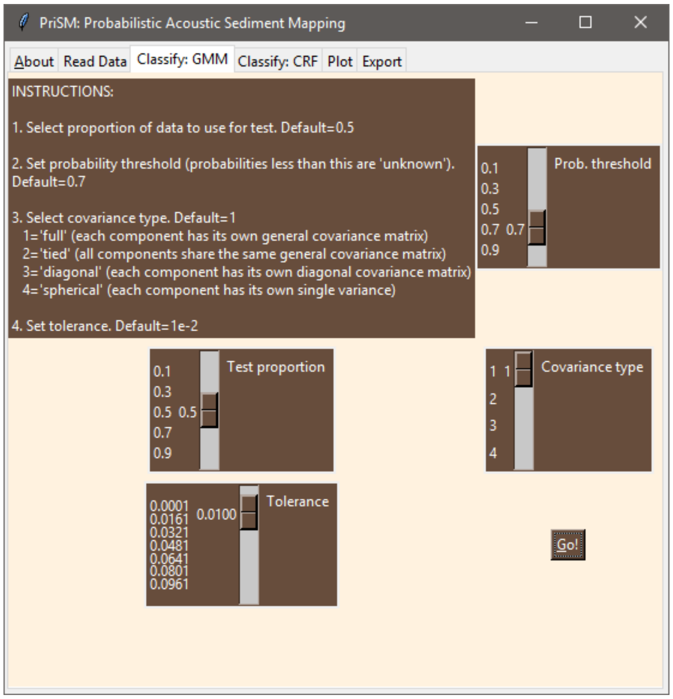
You may use the sliders to change the grid size (default is 1 m), buffer size (the distance in meters that grids are ascribed to each bed observation, default is 10 m), and a filter weight. The filter is for removing high-frequency (speckle) noise. For now, we recommend using the defaults. Hit the 'Go' button. After a few seconds, the following message should appear



Read module finished

Hit OK and move onto the 'Classify: GMM' tab

Classify: GMM



INSTRUCTIONS:

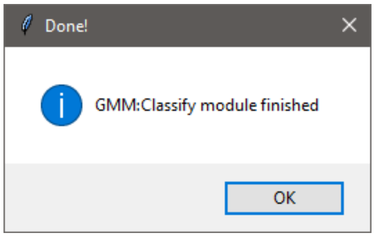
1. Select proportion of data to use for test. Default=0.5

2. Set probability threshold (probabilities less than this are 'unknown'). Default=0.7

3. Select covariance type. Default=1
   1='full' (each component has its own general covariance matrix)
   2='tied' (all components share the same general covariance matrix)
   3='diagonal' (each component has its own diagonal covariance matrix)
   4='spherical' (each component has its own single variance)

4. Set tolerance. Default=1e-2

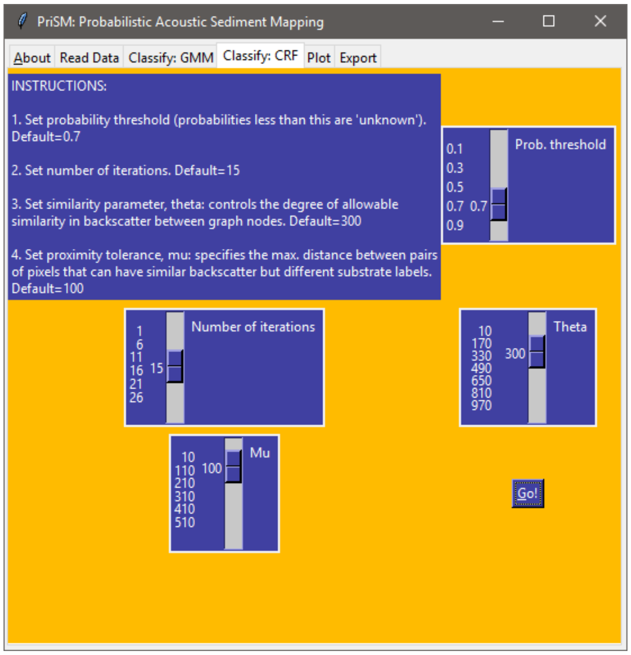
You may use the sliders to change the parameters. ‘Prob. Threshold’ is the threshold probability below which, a classification is considered indeterminate (‘unknown’). The higher this number, the fewer grid cells are classified but the greater confidence in those grid cells that are classified. ‘Test proportion’ is the amount of the data used for testing the model. The remaining proportion is used to train the model. Covariance type is explained in the instructions within the tab. ‘Tolerance’ is a numerical value that determines how the model decides when to finish (very generally, smaller = more accurate model). To run the GMM classification with all the defaults (recommended), simply hit ‘Go’. After a few seconds, you should see this screen



GMM:Classify module finished

Hit OK and move onto the ‘Classify: CRF’ tab

Classify: CRF



INSTRUCTIONS:

1. Set probability threshold (probabilities less than this are 'unknown'). Default=0.7

2. Set number of iterations. Default=15

3. Set similarity parameter, theta: controls the degree of allowable similarity in backscatter between graph nodes. Default=300

4. Set proximity tolerance, mu: specifies the max. distance between pairs of pixels that can have similar backscatter but different substrate labels. Default=100

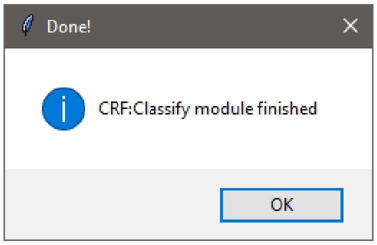
Prob. threshold 0.7

Number of iterations 15

Theta 300

Mu 100

You may use the sliders to change the parameters. ‘Prob. Threshold’ is the threshold probability below which, a classification is considered indeterminate (‘unknown’). The higher this number, the fewer grid cells are classified but the greater confidence in those grid cells that are classified. ‘Number of iterations’ is the number of times the model iterates through before returning a solution. ‘Theta’ and ‘Mu’ parameters are explained in the instructions within the tab. To run the CRF classification with all the defaults (recommended), simply hit ‘Go’. When the model is finished, you should see this screen



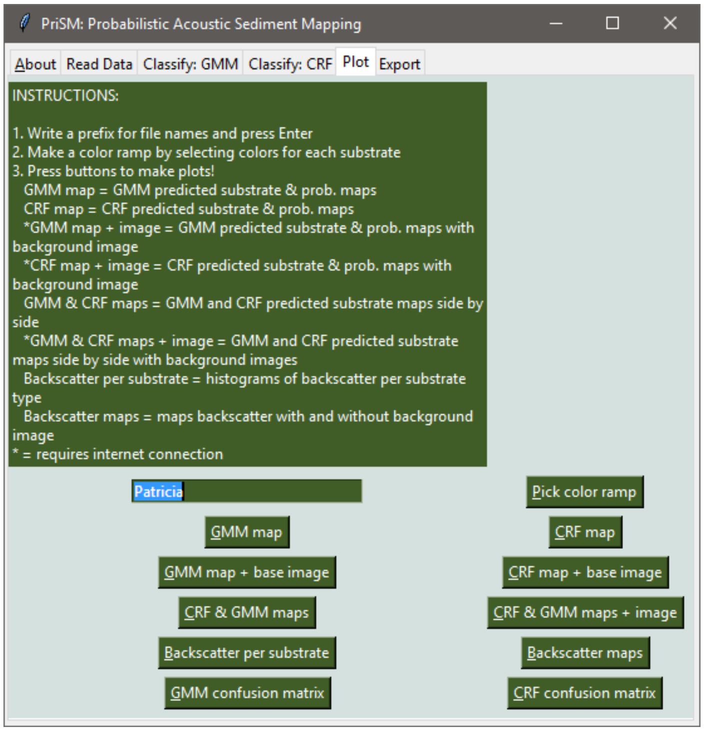
CRF:Classify module finished

Hit OK and move onto the ‘Plot’ tab

Plot

This tab is for generating figures and maps of the results. The files are written to the ‘outputs’ folder within the program

First, you need to provide a file prefix. Below I used ‘Patricia’. Remember to hit [Enter] after typing – the prefix should be highlighted, as below

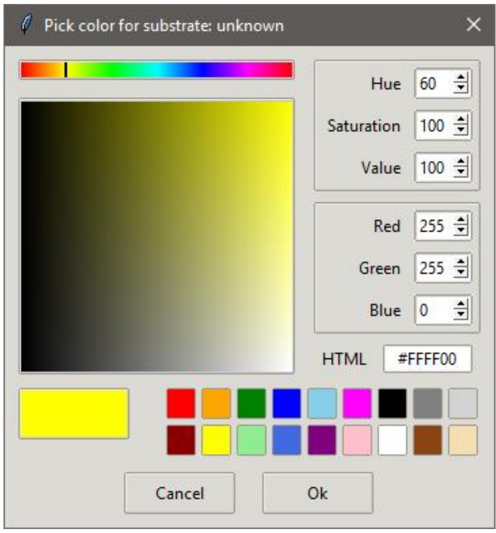


INSTRUCTIONS:

1. Write a prefix for file names and press Enter
2. Make a color ramp by selecting colors for each substrate
3. Press buttons to make plots!
   GMM map = GMM predicted substrate & prob. maps
   CRF map = CRF predicted substrate & prob. maps
   *GMM map + image = GMM predicted substrate & prob. maps with background image
   *CRF map + image = CRF predicted substrate & prob. maps with background image
   GMM & CRF maps = GMM and CRF predicted substrate maps side by side
   *GMM & CRF maps + image = GMM and CRF predicted substrate maps side by side with background images
   Backscatter per substrate = histograms of backscatter per substrate type
   Backscatter maps = maps backscatter with and without background image
* = requires internet connection

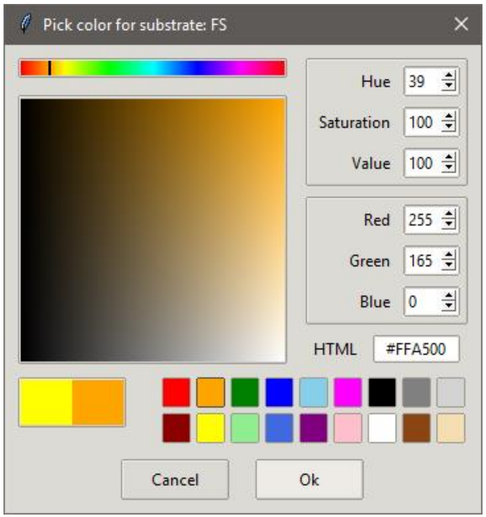
Patricia

Then you’ll need to select colors for each of the substrates by selecting the ‘Pick color ramp’ button. You’ll see a dialog like this:


Pick color for substrate: unknown

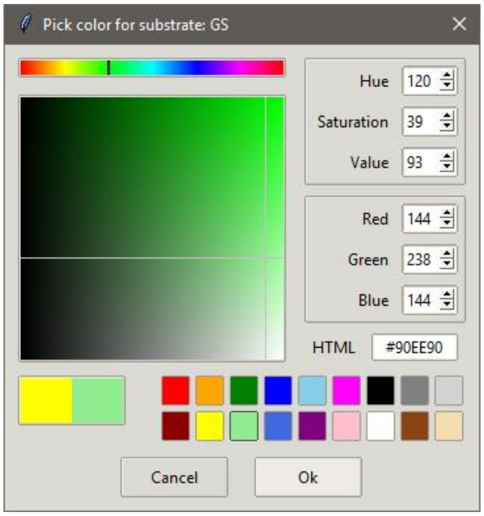
Hue 60
Saturation 100
Value 100
Red 255
Green 255
Blue 0
HTML #FFFF00

You are being asked to select a color for the ‘unknown’ substrate category. I tend to use gray. Cycle through each of the substrates assigning a color to each:



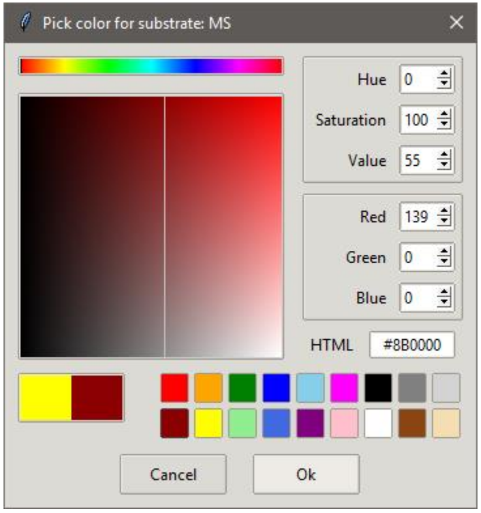
Pick color for substrate: FS

Hue 39
Saturation 100
Value 100
Red 255
Green 165
Blue 0
HTML #FFA500



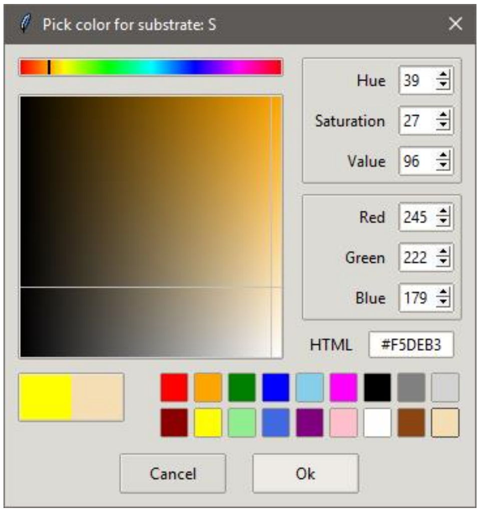
Pick color for substrate: GS

Hue 120
Saturation 39
Value 93
Red 144
Green 238
Blue 144
HTML #90EE90



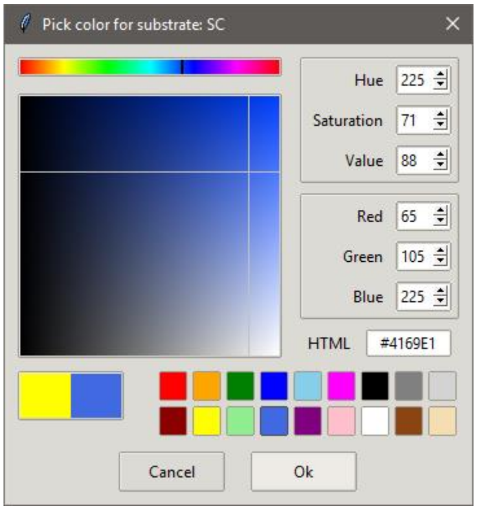
Pick color for substrate: MS

Hue 0
Saturation 100
Value 55
Red 139
Green 0
Blue 0
HTML #8B0000



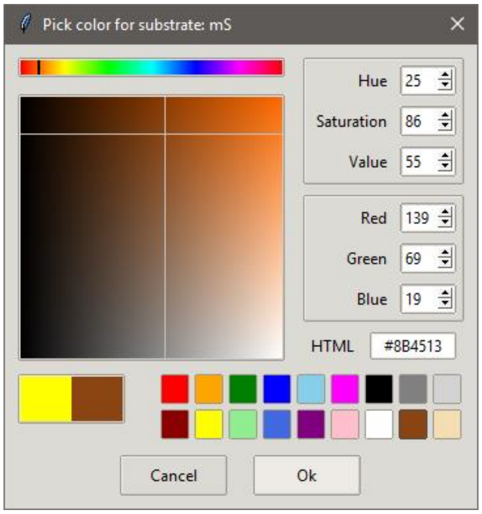
Pick color for substrate: S

Hue 39
Saturation 27
Value 96
Red 245
Green 222
Blue 179
HTML #F5DEB3



Pick color for substrate: SC

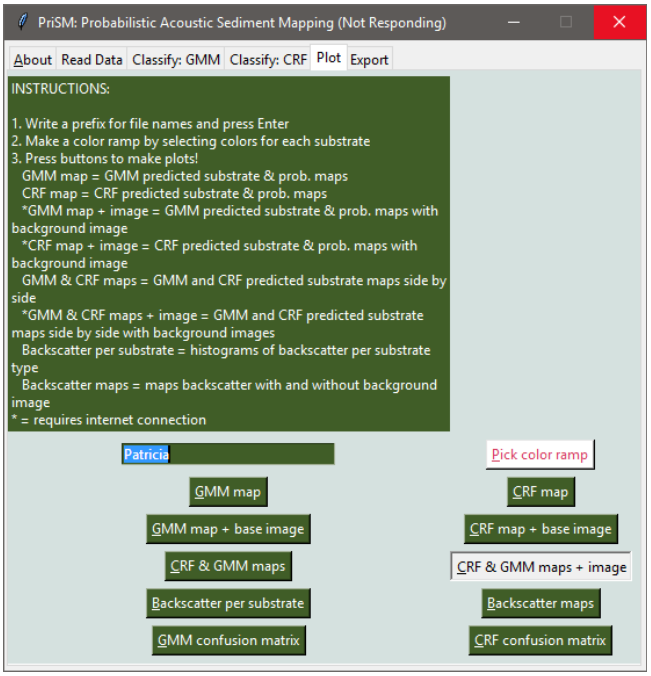
Hue 225
Saturation 71
Value 88
Red 65
Green 105
Blue 225
HTML #4169E1



Pick color for substrate: mS

Hue 25
Saturation 86
Value 55
Red 139
Green 69
Blue 19
HTML #8B4513

Now you're ready to make some plots. The instructions within the tab explain what each button creates. Since we ran both the GMM and CRF modules, we can hit the 'CRF & GMM maps + images' button that will generate lots of outputs from both models:

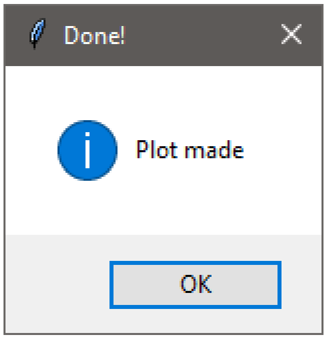

PriSM: Probabilistic Acoustic Sediment Mapping (Not Responding)
About
Read Data
Classify: GMM
Classify: CRF
Plot
Export

INSTRUCTIONS:

1. Write a prefix for file names and press Enter
2. Make a color ramp by selecting colors for each substrate
3. Press buttons to make plots!
   GMM map = GMM predicted substrate & prob. maps
   CRF map = CRF predicted substrate & prob. maps
   *GMM map + image = GMM predicted substrate & prob. maps with background image
   *CRF map + image = CRF predicted substrate & prob. maps with background image
   GMM & CRF maps = GMM and CRF predicted substrate maps side by side
   *GMM & CRF maps + image = GMM and CRF predicted substrate maps side by side with background images
   Backscatter per substrate = histograms of backscatter per substrate type
   Backscatter maps = maps backscatter with and without background image
* = requires internet connection

Patricia

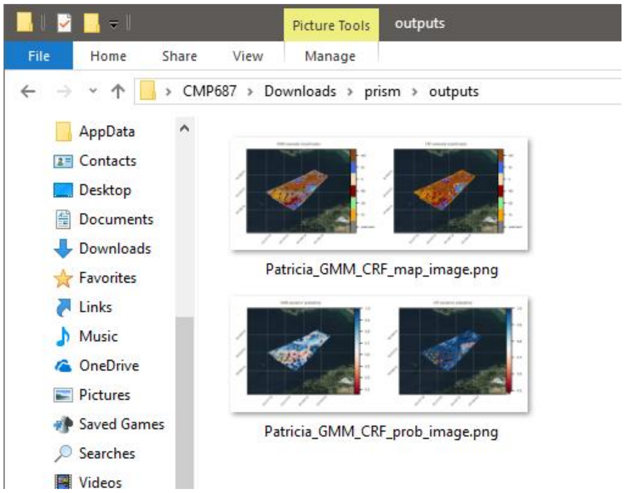
After a little while, you'll get a message:



Done!

Plot made

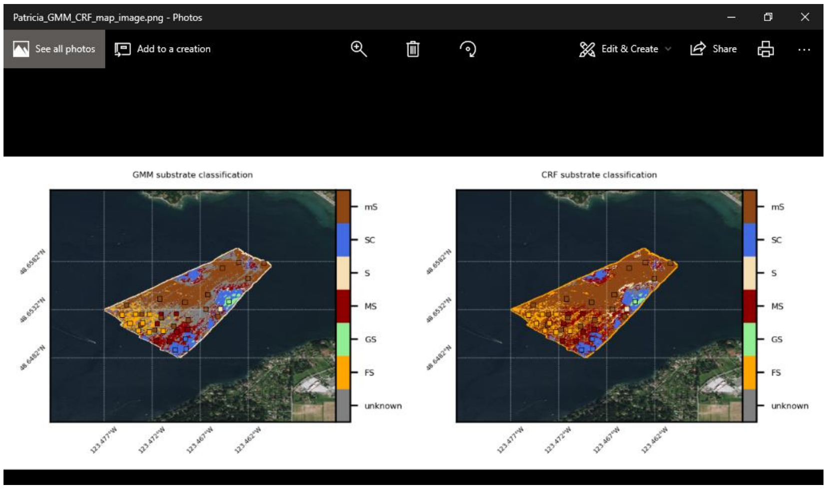
and you can navigate to your browser window (prism → outputs) to see the generated plots



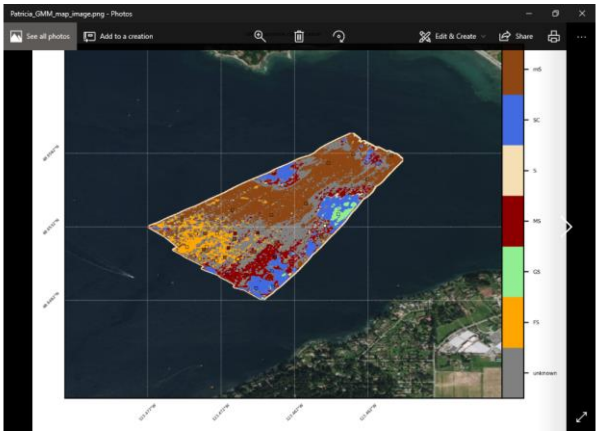
Patricia_GMM_CRF_map_image.png

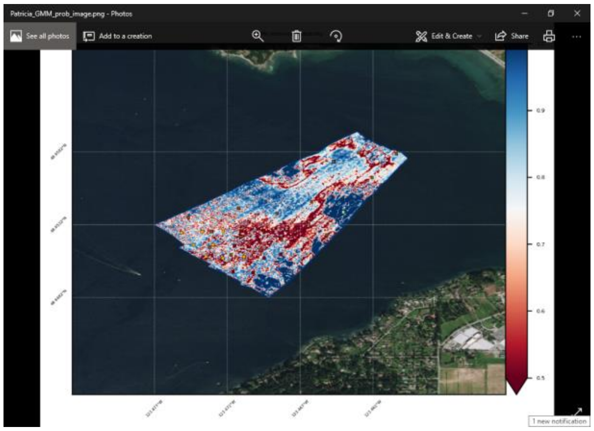
Patricia_GMM_CRF_prob_image.png

And of course you can view the png files in your image viewer:

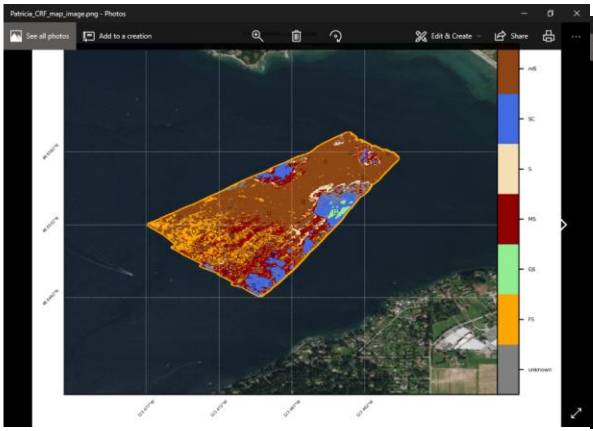


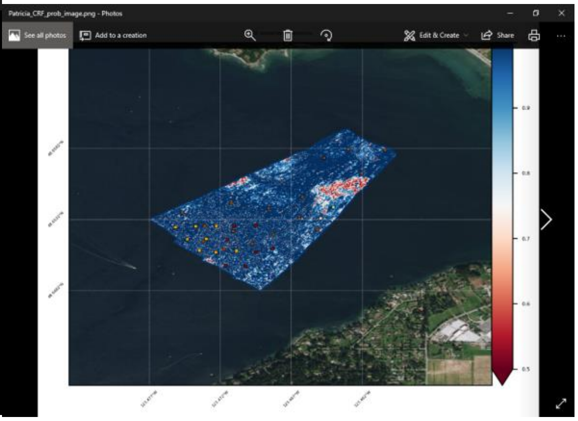
These are substrate maps for GMM and CRF outputs side-by-side, overlain onto an aerial image downloaded from the internet. You may generate other plots for GMM and CRF models individually. For example, ‘GMM map + base image’ will generate substrate and ‘posterior probability of substrate’ maps:



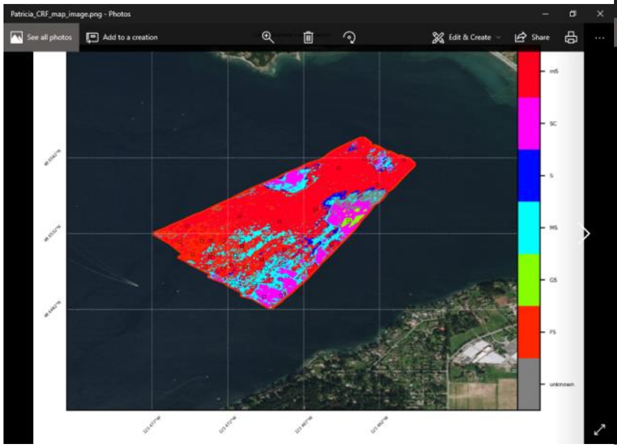


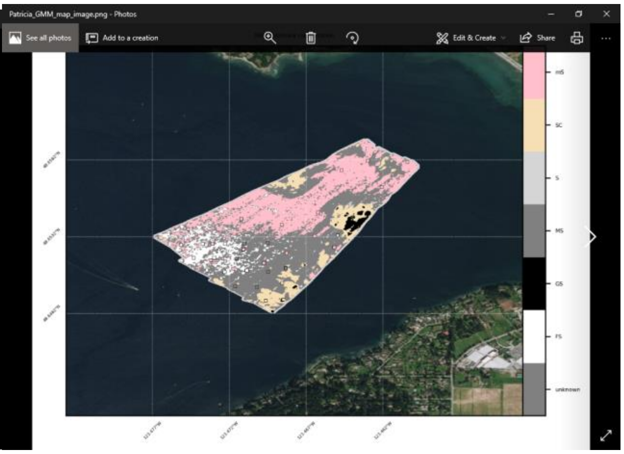
And ‘CRF map + base image’ will create the same things from the CRF outputs:



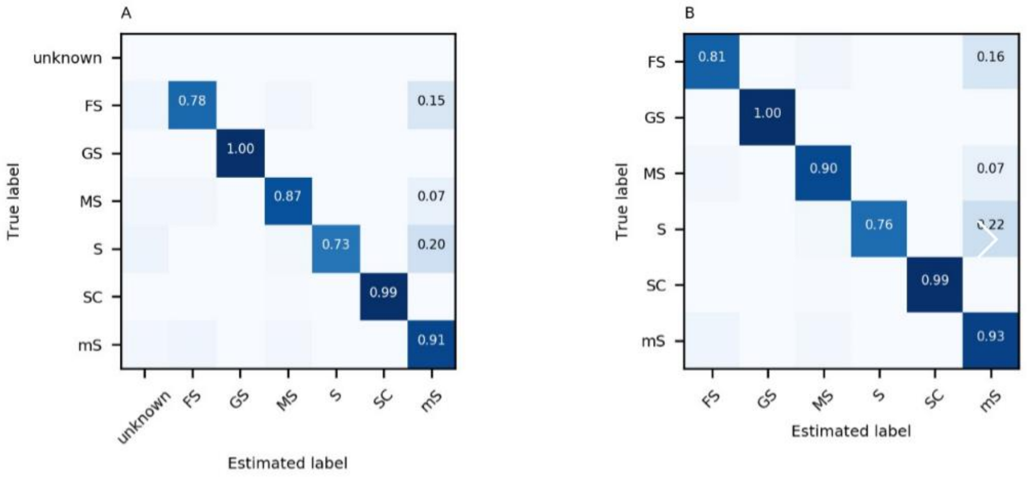


At any point, you may choose a different color scheme and redo the plots, for example:



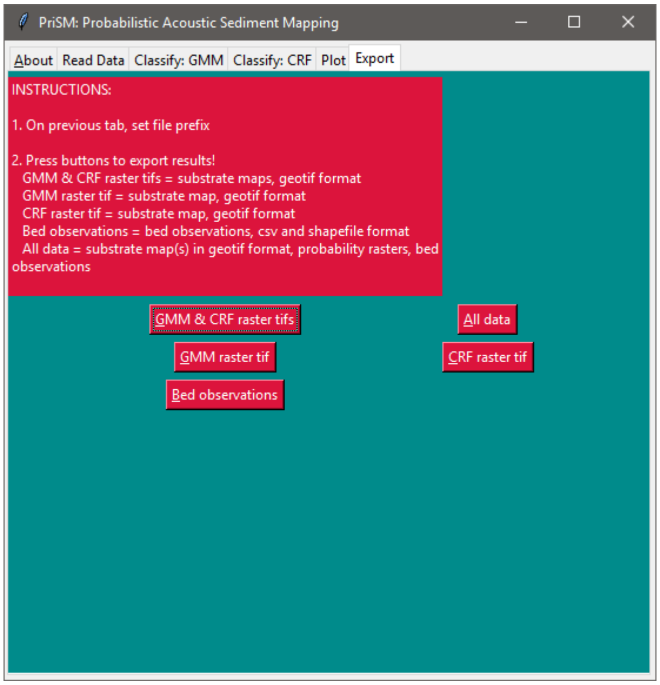


You can even generate plots of confusion matrices to see how well the classifier worked:



Export

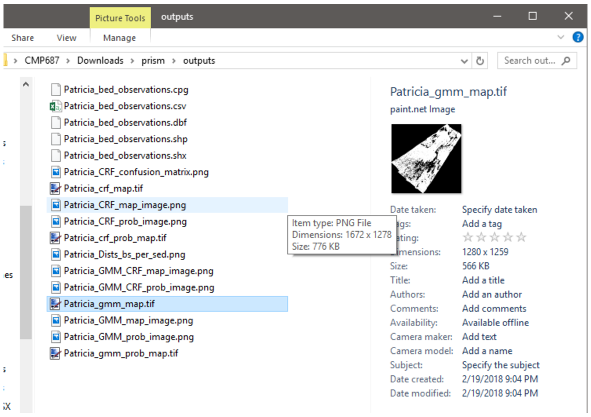
Finally, this tab is for generating outputs that you may use in GIS or another program. The files are written to the ‘outputs’ folder within the program. It is relatively simple: you may generate maps in geoTIFF format (that may be loaded into a GIS), you may export bed observations within the surveyed multibeam extent as a shapefile (again, that is compatible with GIS) or csv, or you may do both!



INSTRUCTIONS:

1. On previous tab, set file prefix

2. Press buttons to export results!
   GMM & CRF raster tifs = substrate maps, geotif format
   GMM raster tif = substrate map, geotif format
   CRF raster tif = substrate map, geotif format
   Bed observations = bed observations, csv and shapefile format
   All data = substrate map(s) in geotif format, probability rasters, bed observations



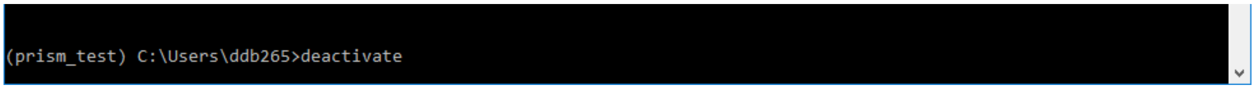
Patricia_bed_observations.cpg
Patricia_bed_observations.csv
Patricia_bed_observations.dbf
Patricia_bed_observations.shp
Patricia_bed_observations.shx
Patricia_CRF_confusion_matrix.png
Patricia_crf_map.tif
Patricia_CRF_map_image.png
Patricia_CRF_prob_image.png
Patricia_crf_prob_map.tif
Patricia_Dists_bs_per_sed.png
Patricia_GMM_CRF_map_image.png
Patricia_GMM_CRF_prob_image.png
Patricia_gmm_map.tif
Patricia_GMM_map_image.png
Patricia_GMM_prob_image.png
Patricia_gmm_prob_map.tif

Item type: PNG File
Dimensions: 1672 x 1278
Size: 776 KB

Patricia_gmm_map.tif
paint.net Image

Date taken:	Specify date taken
gs:	Add a tag
ting:	☆☆☆☆☆
mensions:	1280 x 1259
Size:	566 KB
Title:	Add a title
Authors:	Add an author
Comments:	Add comments
Availability:	Available offline
Camera maker:	Add text
Camera model:	Add a name
Subject:	Specify the subject
Date created:	2/19/2018 9:04 PM
Date modified:	2/19/2018 9:04 PM

When you are finished, remember to deactivate your conda environment, like so:

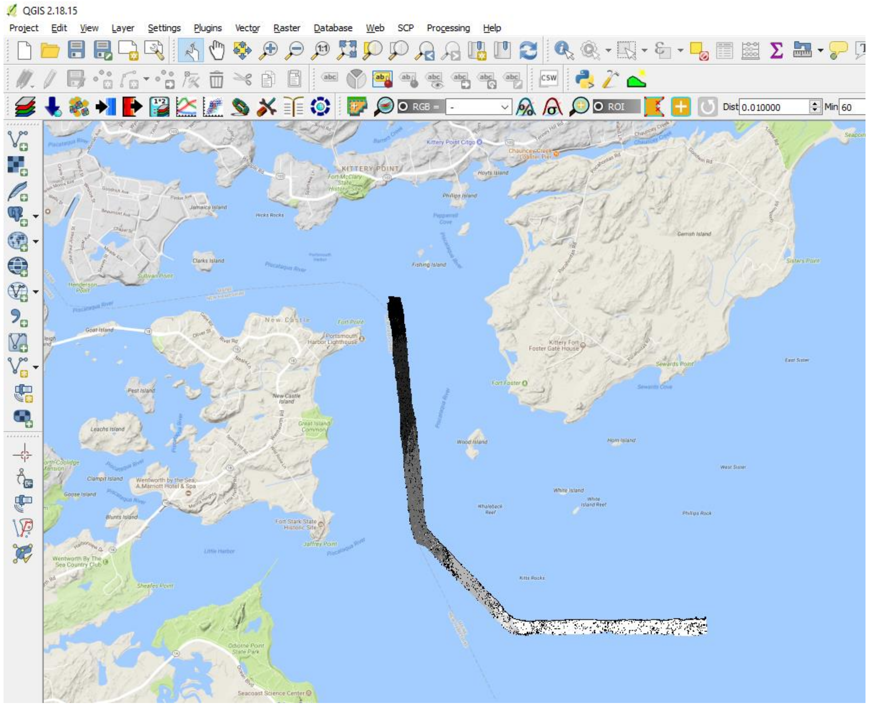


```
(prism_test) C:\Users\ddb265>deactivate
```

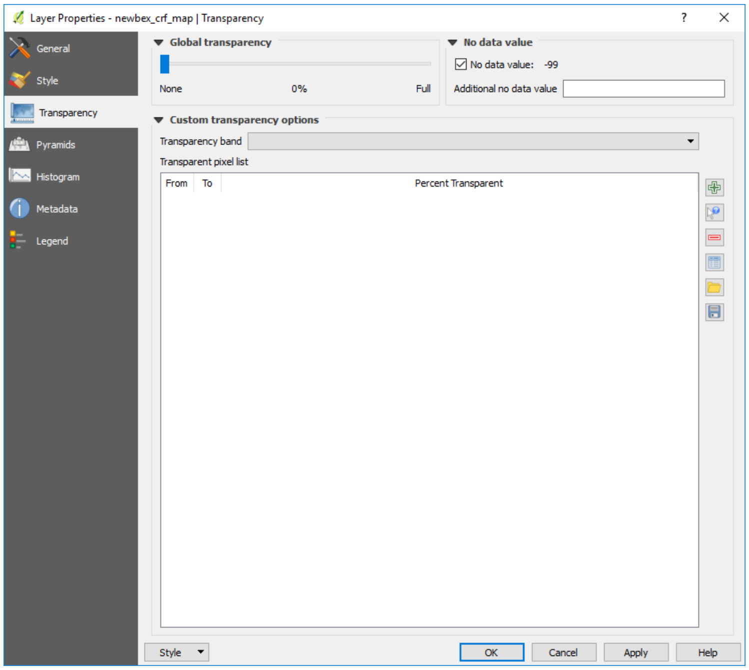
Using data exports in GIS

Quick Example Using QGIS

Open a new project and drag GeoTIFFs into the main window. This automatically creates a new layer


QGIS 2.18.15
Project Edit View Layer Settings Plugins Vector Raster Database Web SCP Processing Help
RGB =
ROI
Dist 0.010000
Min 60
KITTERY POINT
New Castle

Here is the CRF substrate map tiff for the NEWBEX site (lower Portsmouth Harbor, New Hampshire, USA)



Global transparency: None 0% Full

No data value

☑ No data value: -99

Additional no data value

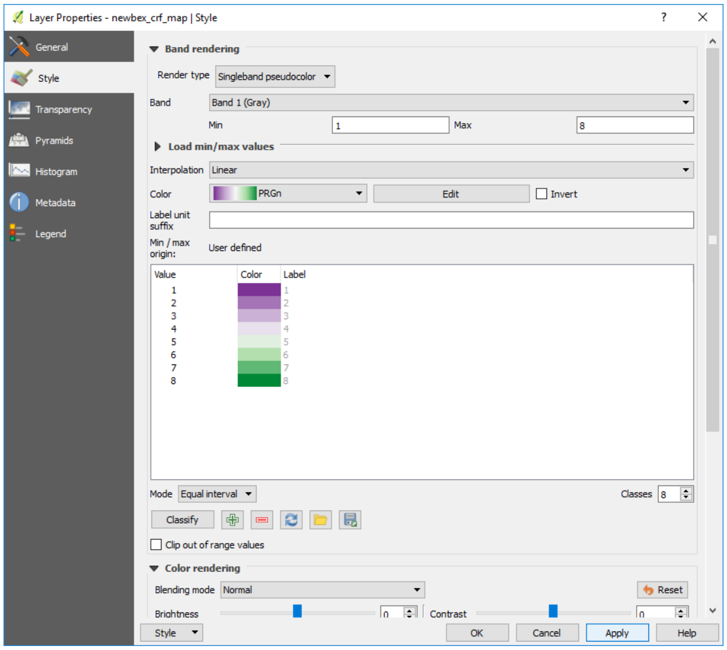
Custom transparency options

Transparency band

Transparent pixel list

From	To	Percent Transparent

Right click the layer in layer panel → Properties



Layer Properties - newbex_crf_map | Style

Band rendering

Render type: Singleband pseudocolor

Band: Band 1 (Gray)

Min: 1 Max: 8

Load min/max values

Interpolation: Linear

Color: PRGn Edit □ Invert

Label unit suffix:

Min / max origin: User defined

Value	Color	Label
1	🟪	1
2	🟪	2
3	🟪	3
4	⬜	4
5	⬜	5
6	🟩	6
7	🟩	7
8	🟩	8

Mode: Equal interval Classes: 8

Classify

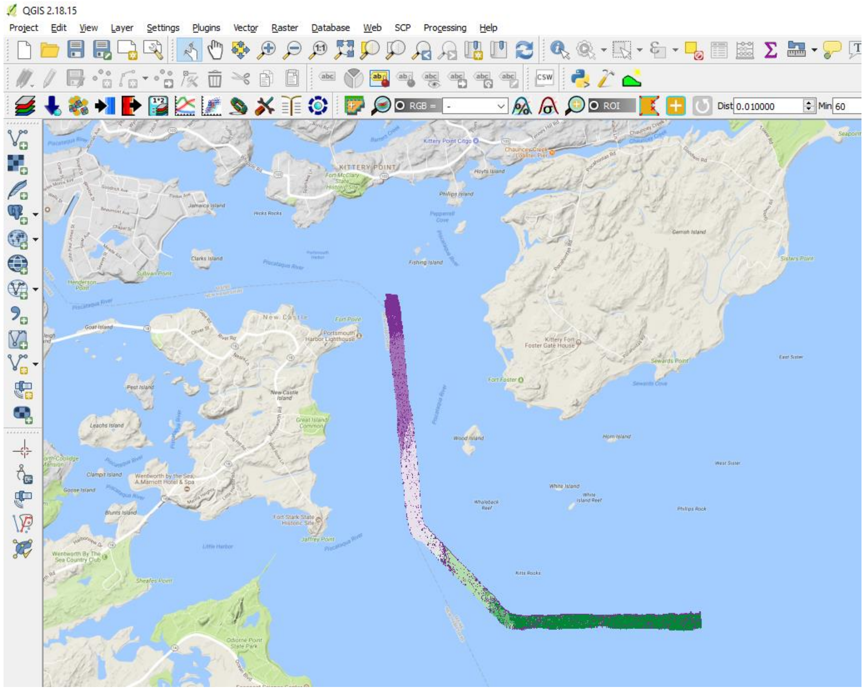
□ Clip out of range values

Color rendering

Blending mode: Normal Reset

Brightness: 0 Contrast: 0

In the Style tab, render type ‘singleband psuedocolor’, set min and max, choose a color scale, set mode to ‘equal interval’, choose the number of classes to match the number of integer values in the map, and hit Apply



QGIS 2.18.15
Project
Edit
View
Layer
Settings
Plugins
Vector
Raster
Database
Web
SCP
Processing
Help
RGB =
ROI
Dist 0.010000
Min 60
KITTERY POINT
New Castle
Gerrish Island