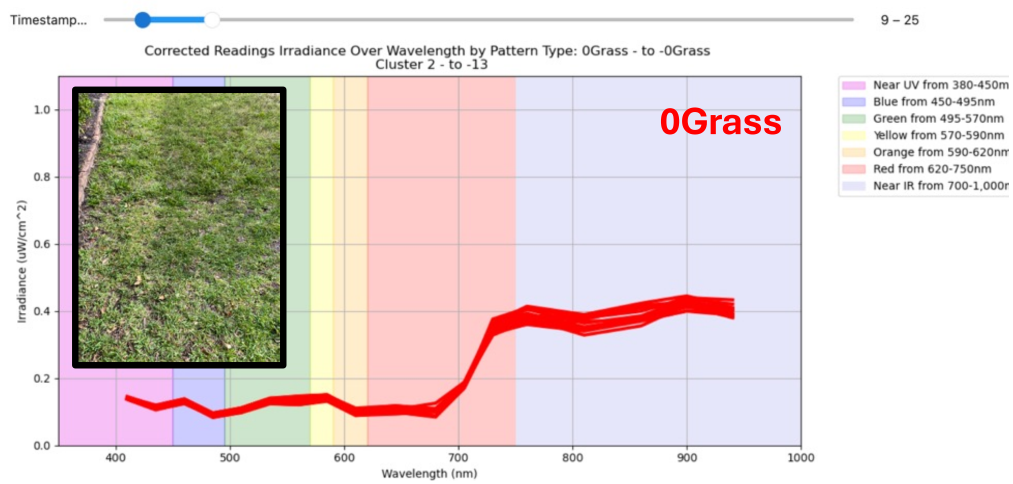
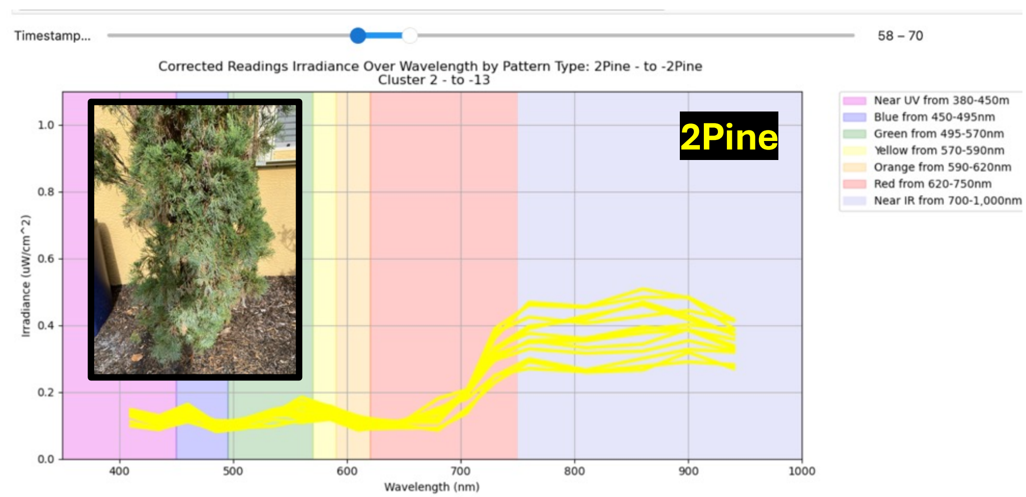
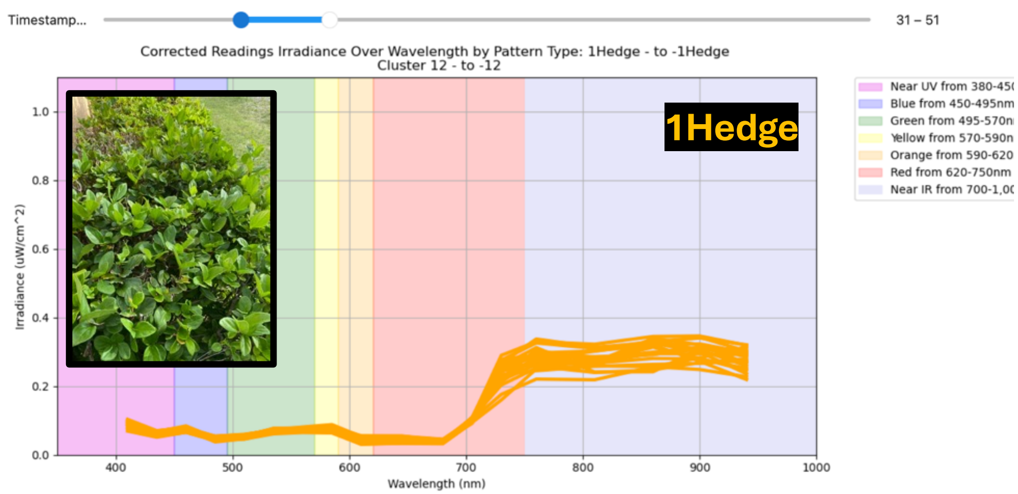
Initial Vegetative Species Library Development and Applications

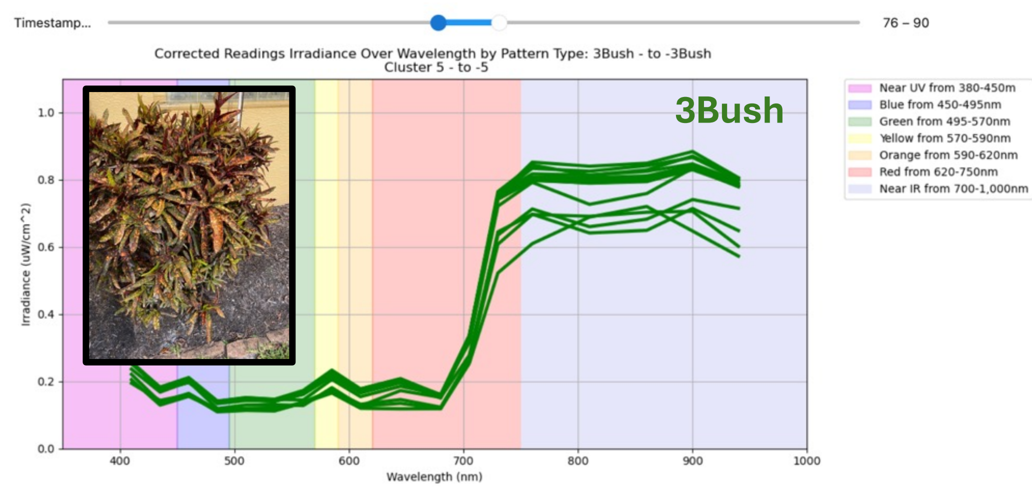
All White-Card Corrected Data using White-Card knee-high readings from Tim Childress, April 27 STELLA Data

Vegetative Test Pattern Library – STELLA-Q2 Data White-Card Corrected

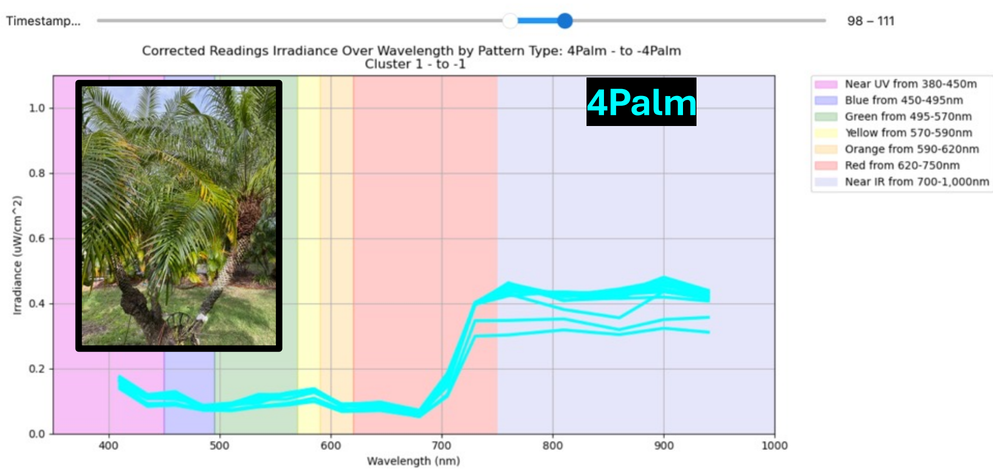


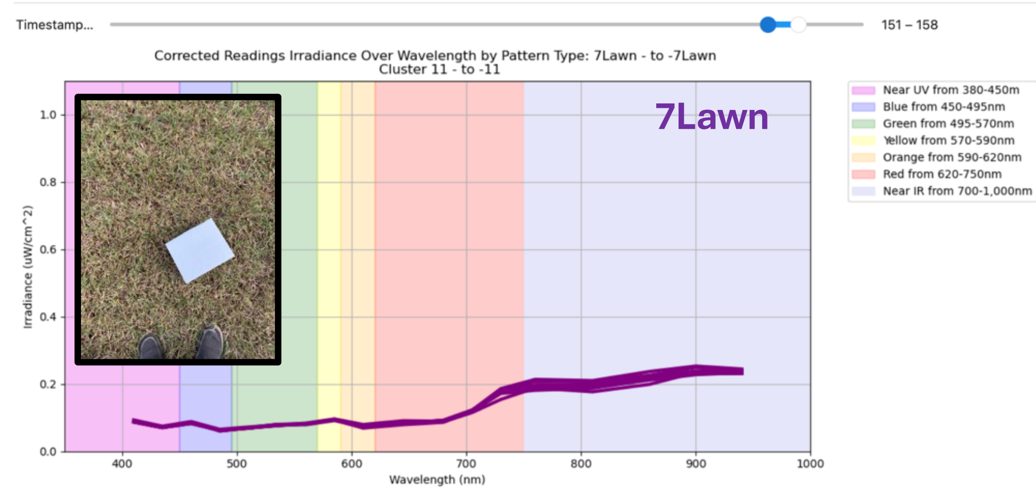


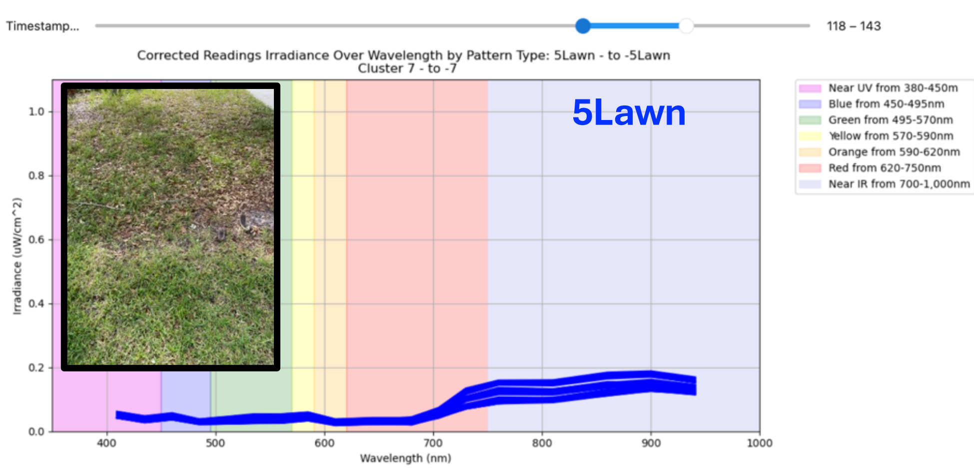


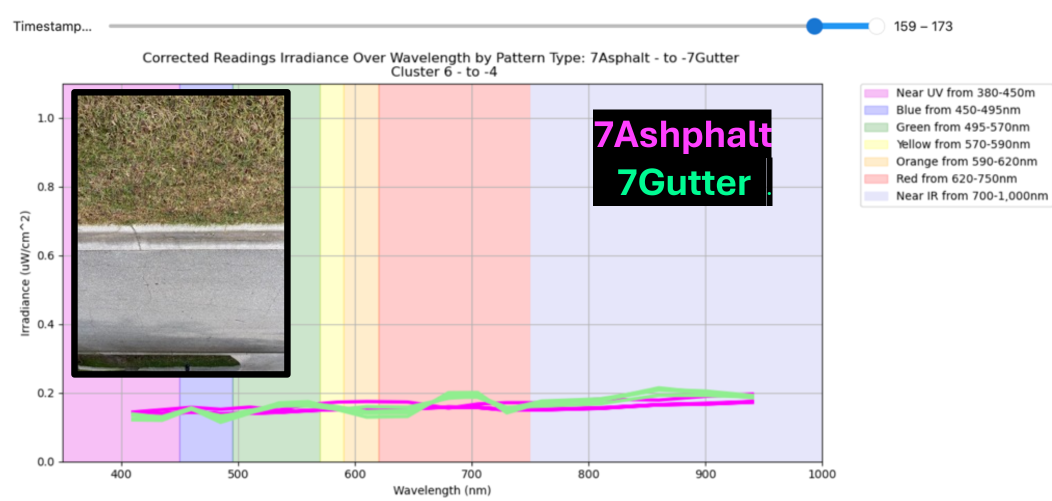


Vegetative Test Pattern Library – STELLA-Q2 Data White-Card Corrected

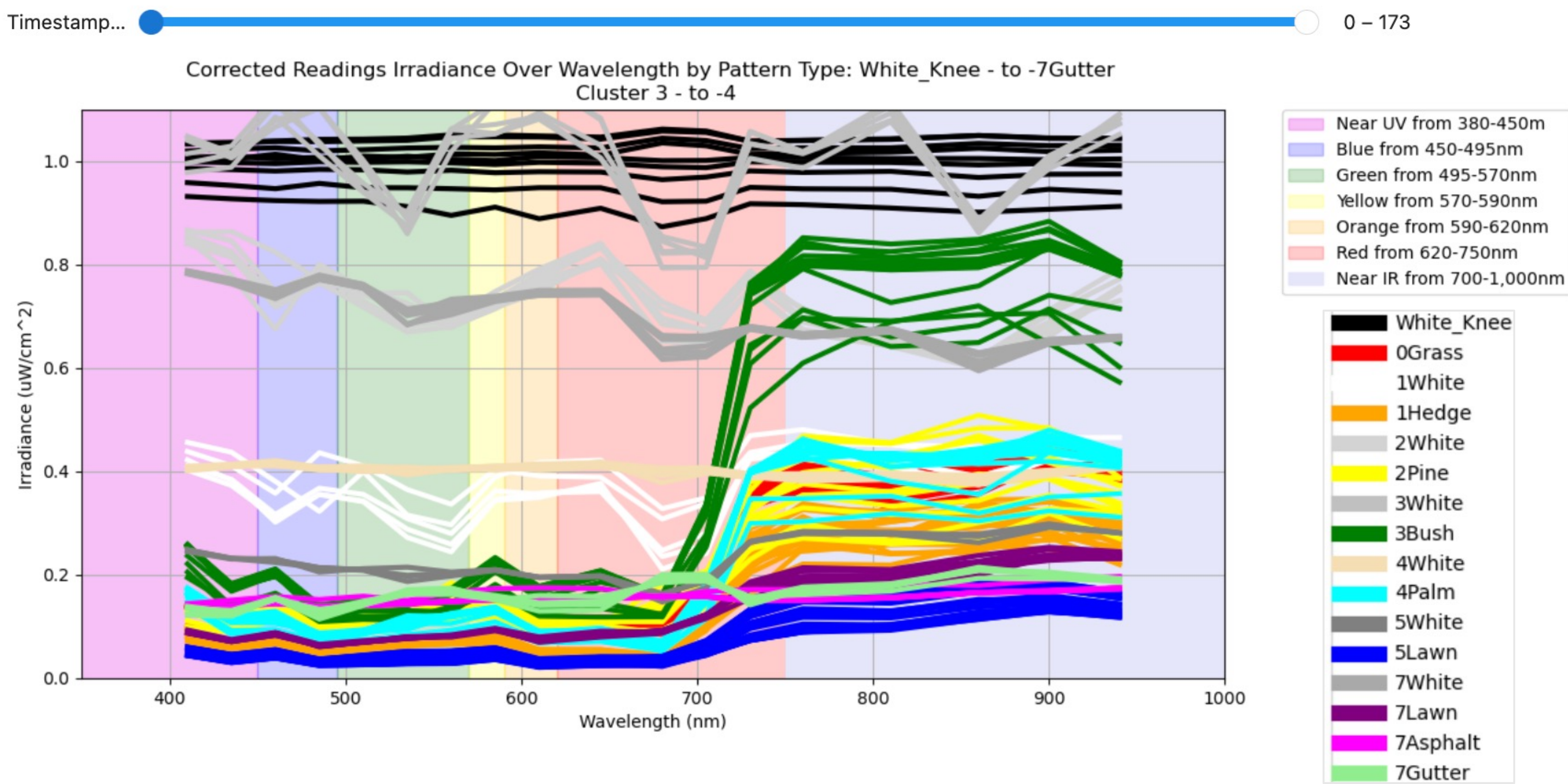








Test Pattern Library – STELLA-Q2 Data White-Card Corrected

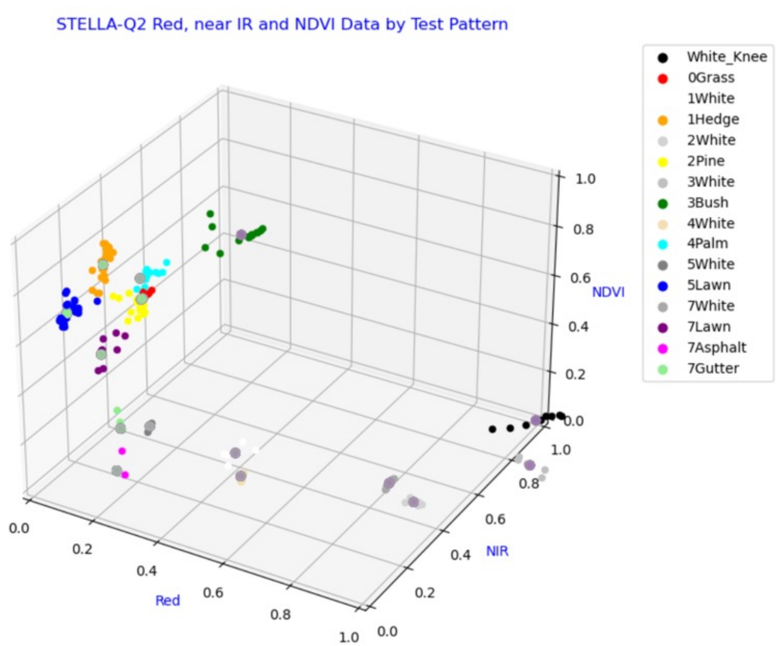


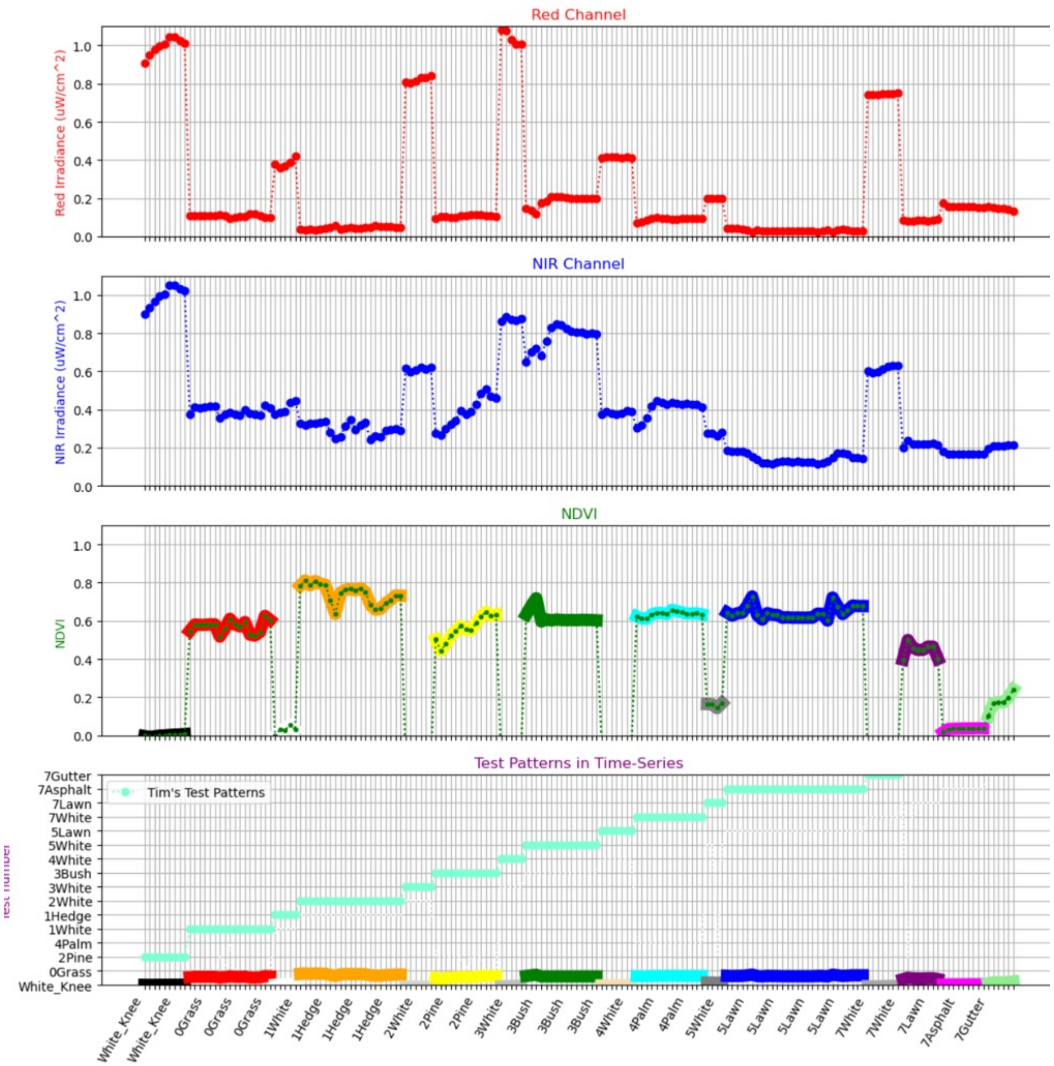
How Do we use these Vegetative Test Pattern Libraries as Calibration Data?

First Calculate NDVI:

***NDVI* = (*NIR* – *Red*)/(*NIR* + *Red*)**

We calculate NDVI from STELLA-Q2 Red channel at 645nm and the near IR channel at 860nm using white-Card Corrected STELLA-Q2 data.

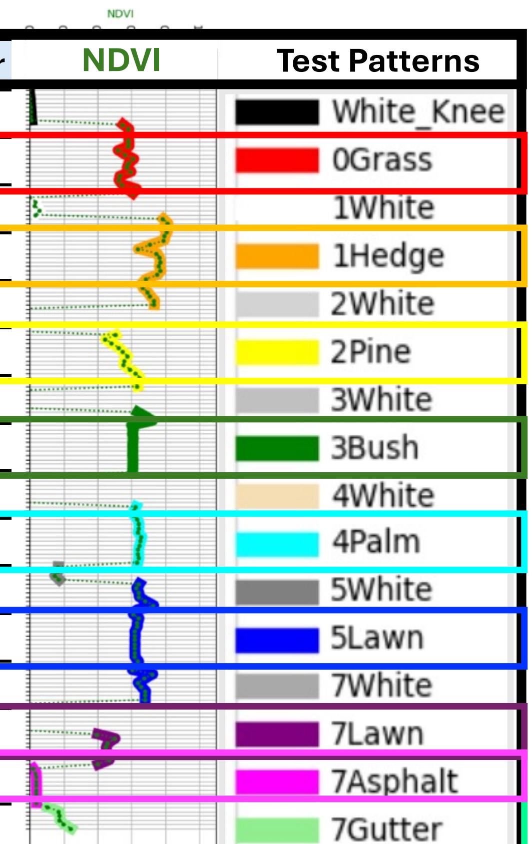




How Do we use these Vegetative Test Pattern Libraries as Calibration Data?

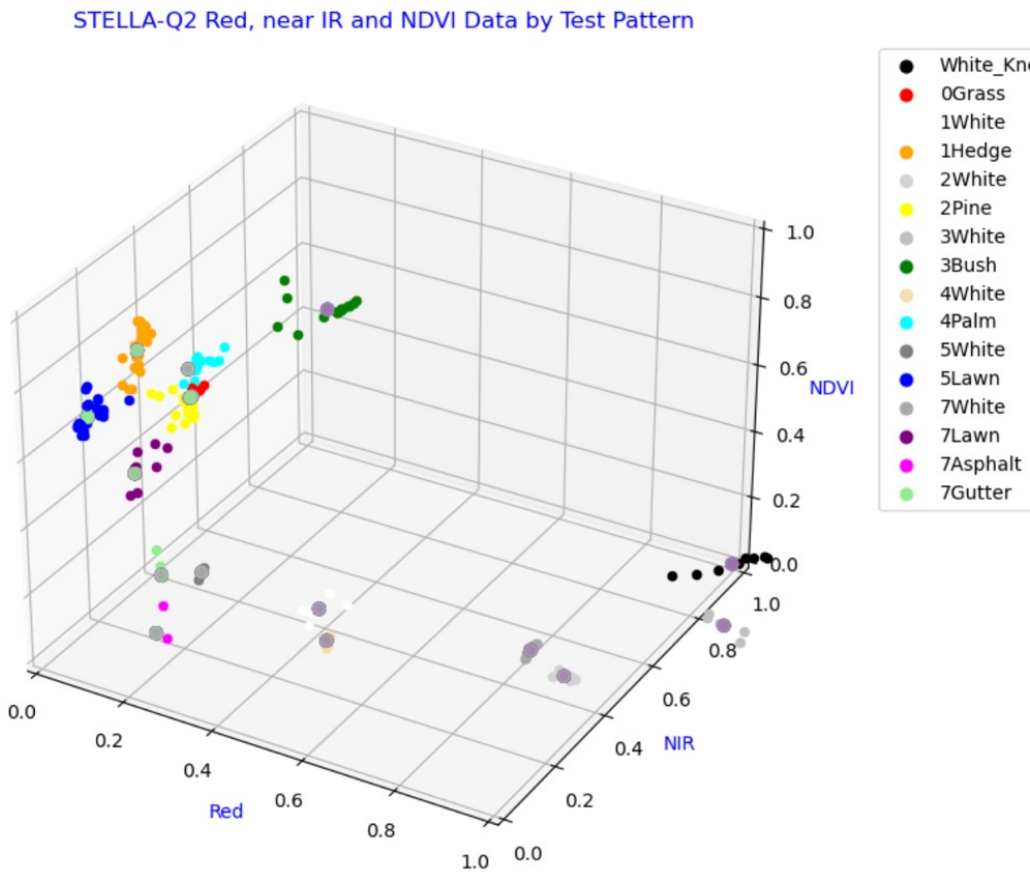
Mean Red, NIR and NDVI Data for each Test Pattern:

Red_Endmember	NIR_Endmember	NDVI_Endmember	Test_Pattern	Test_Number
0.996	0.995	0.000	White_Knee	2
0.107	0.392	0.571	0Grass	4
0.383	0.406	0.029	1White	5
0.044	0.300	0.743	1Hedge	6
0.821	0.612	-0.146	2White	7
0.106	0.386	0.571	2Pine	8
1.041	0.874	-0.087	3White	9
0.185	0.778	0.616	3Bush	10
0.414	0.382	-0.040	4White	11
0.090	0.407	0.637	4Palm	12
0.197	0.272	0.160	5White	13
0.031	0.143	0.645	5Lawn	14
0.745	0.613	-0.098	7White	15
0.083	0.219	0.448	7Lawn	16
0.156	0.167	0.033	7Asphalt	17
0.146	0.208	0.176	7Gutter	18

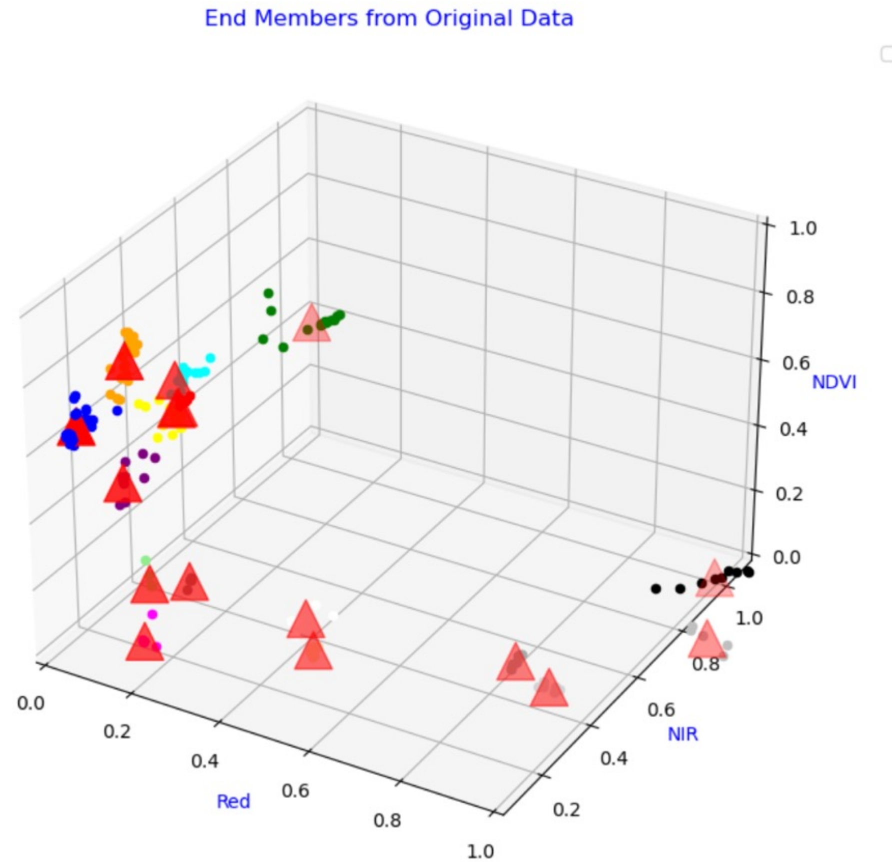


How Do we use these Vegetative Test Pattern Libraries as Calibration Data?

All STELLA Data Colored by Test Patterns



End Member Cluster Centers by Test Patterns

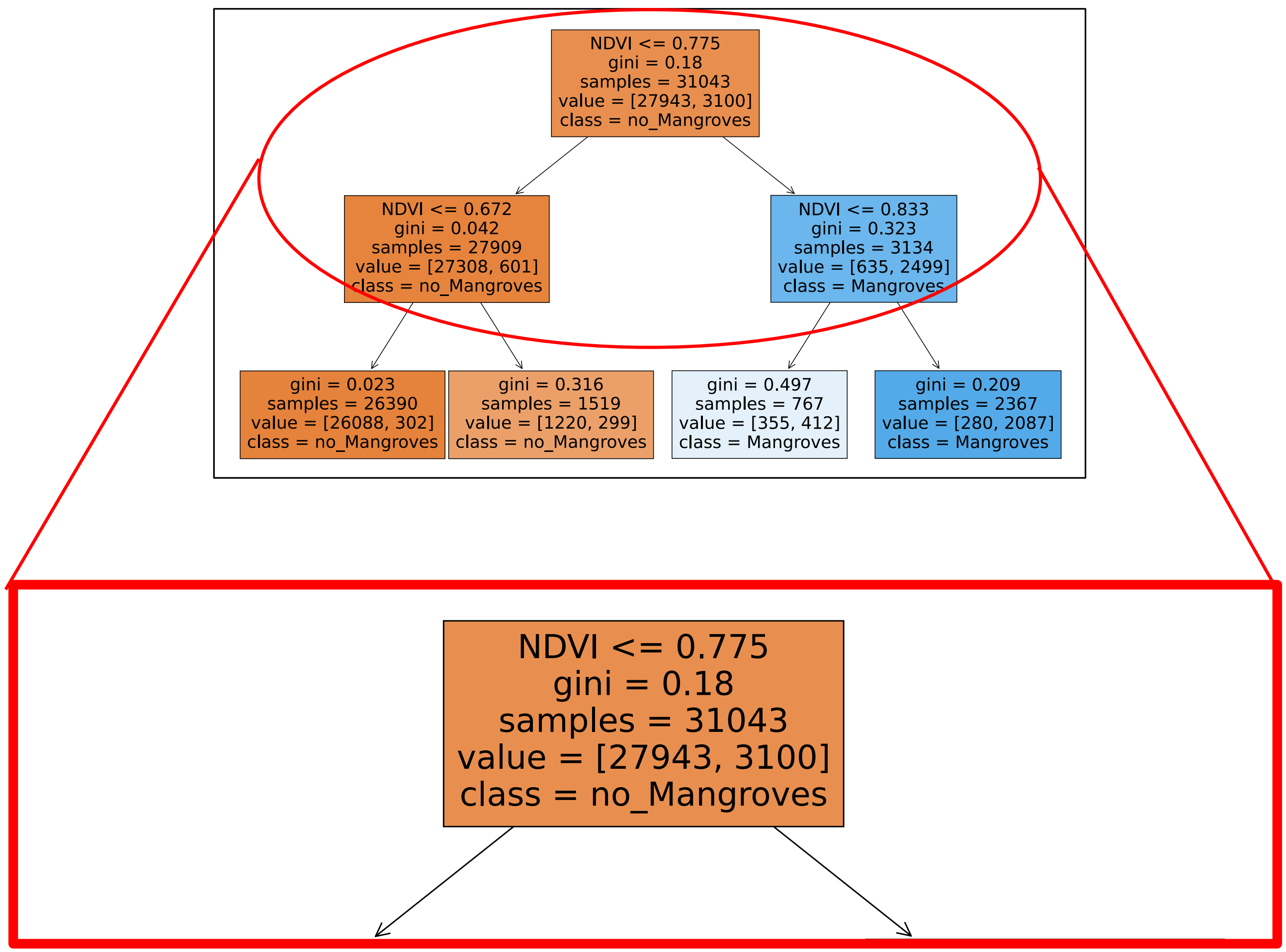


How Do we use these Vegetative Test Pattern Libraries as Calibration Data?

Methods Used to Predict Test Patterns from STELLA-Q2 Channel Data:

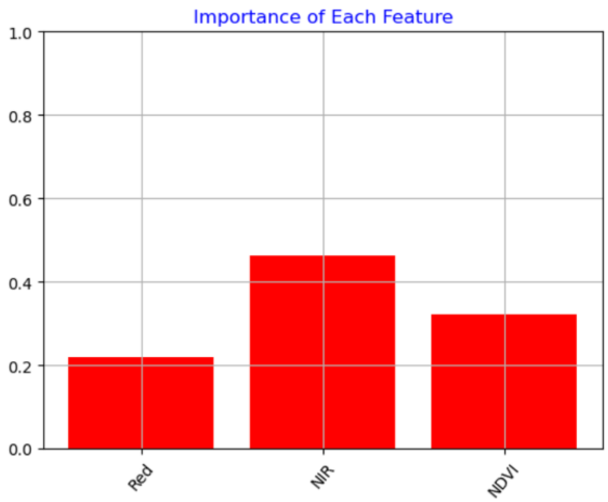
- **Decision Tree Analysis** – useful in understanding the logic in selecting threshold values.
  - Good for Landsat Mangrove Habitats
  - Not so good for predicting 14 different species.
- **Spectral Angle** - good
- **Nearest Neighbor (Knn)** – dead simple and works well.
  - Calculate mean values for each wavelength used for each species: End Members for each species
  - Compare each spectra to End Members
  - Calculate Euclidian Distance for each End Member to spectra being studied
  - Take the Inverse of the square for each
  - Sort by largest to smallest values in K of Knn
  - Calculate weighted average values to estimate Species.
- **Spectral Clustering -** Black Box and not even good
- **Spectral Unmixing** – still working on this in python. Earth Engine has this but working on developing our own code in python

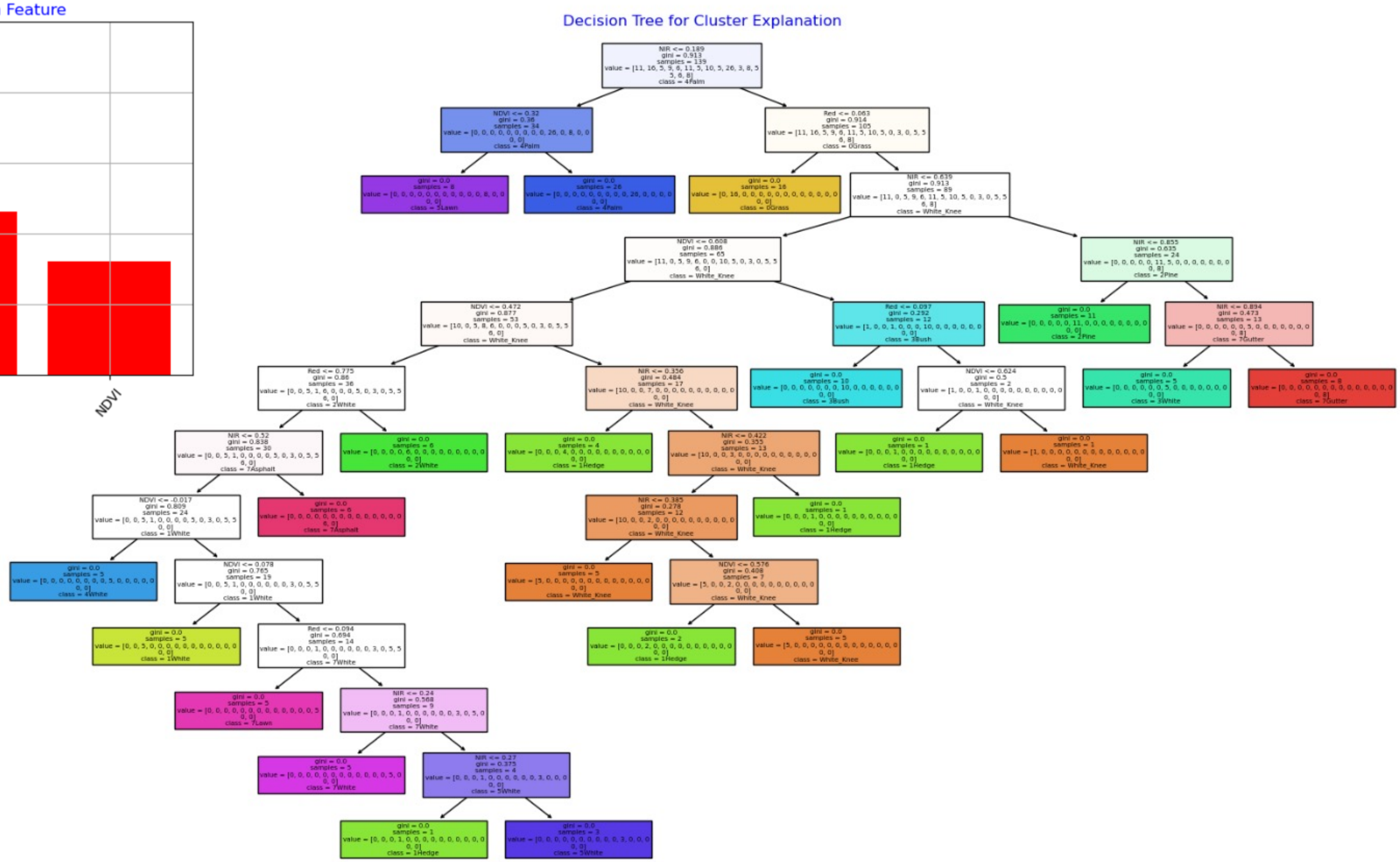
Decision Tree from Landsat NDVI to Predict Mangroves



82% Solution in first branch

Decision Tree used to Segregate STELLA Test Patterns

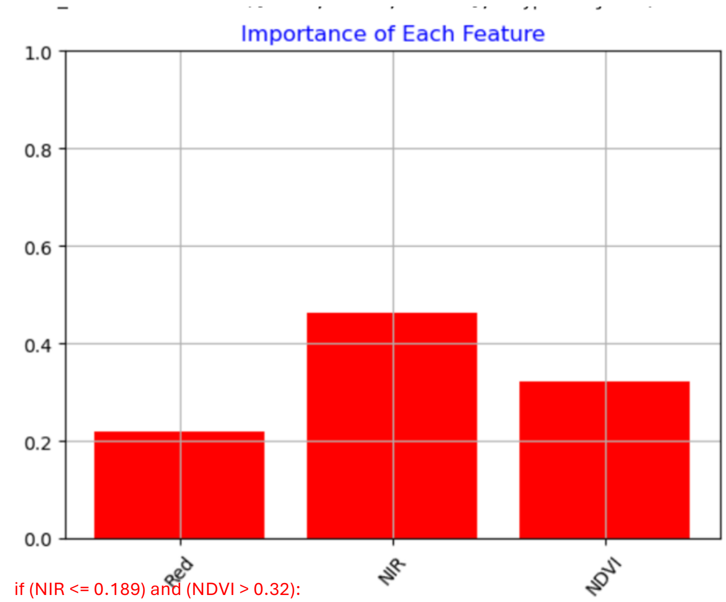




Decision Tree Logic used to Segregate STELLA Test Patterns

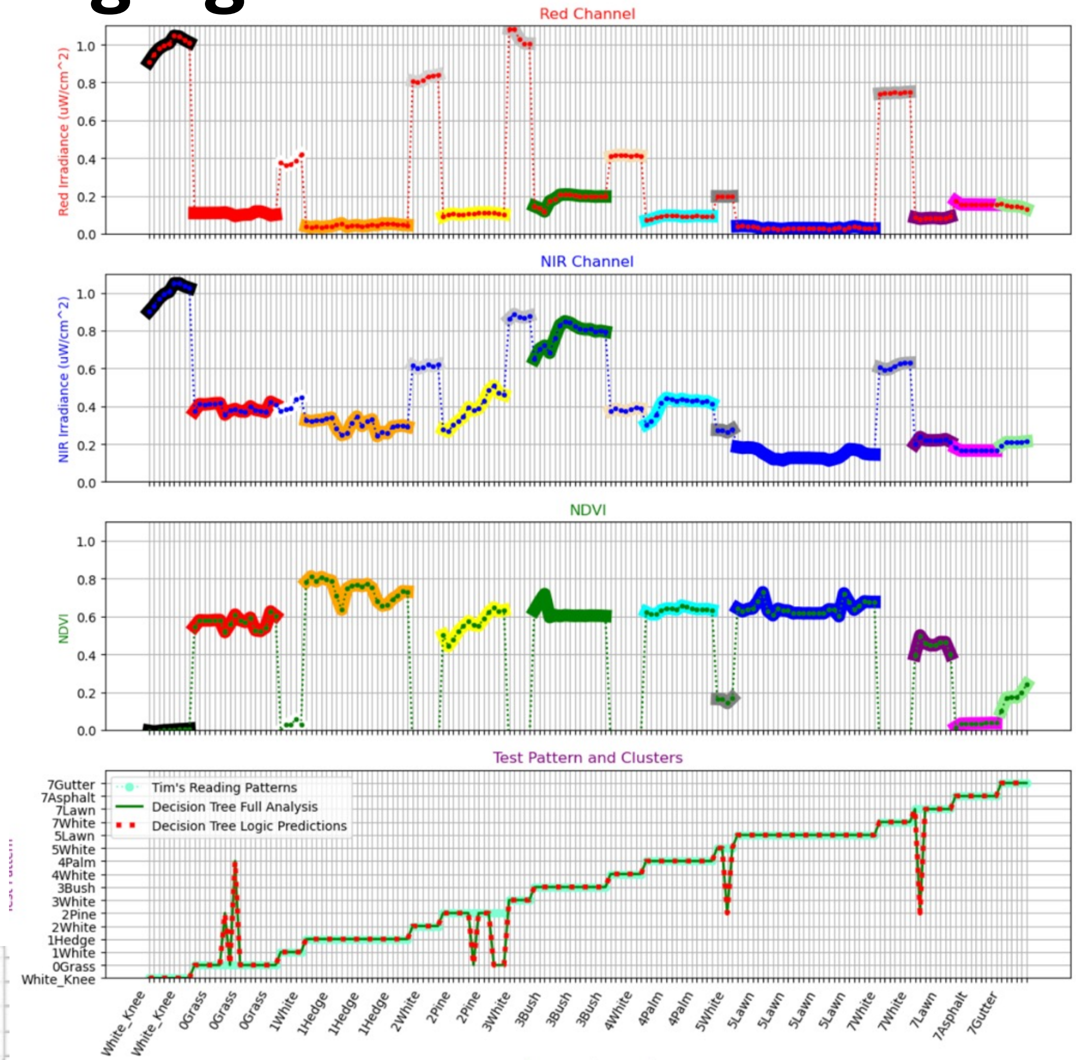
```
def classify_pixel(NDVI, NIR, Red):
   if (NIR <= 0.189) and (NDVI > 0.32):
      return '5Lawn' # probability of 100.0% , based on 26 samples
   if (NIR > 0.189) and (Red <= 0.063):
      return '1Hedge' # probability of 100.0% , based on 16 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR > 0.639) and (NIR <= 0.855):
      return '3Bush' # probability of 100.0% , based on 11 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI > 0.608) and (Red <= 0.097):
      return '4Palm' # probability of 100.0% , based on 10 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR > 0.639) and (NIR > 0.855) and (NIR > 0.894):
      return 'White_Knee' # probability of 100.0% , based on 8 samples
   if (NIR <= 0.189) and (NDVI <= 0.32):
      return '7Asphalt' # probability of 100.0% , based on 8 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI <= 0.472) and (Red <= 0.775) and (NIR > 0.52):
      return '7White' # probability of 100.0% , based on 6 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI <= 0.472) and (Red > 0.775):
      return '2White' # probability of 100.0% , based on 6 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI > 0.472) and (NIR > 0.356) and (NIR <= 0.422) and (NIR > 0.385) and (NDVI > 0.576):
      return '0Grass' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI > 0.472) and (NIR > 0.356) and (NIR <= 0.422) and (NIR <= 0.385):
      return '0Grass' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR > 0.639) and (NIR > 0.855) and (NIR <= 0.894):
      return '3White' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI <= 0.472) and (Red <= 0.775) and (NIR <= 0.52) and (NDVI > -0.017) and (NDVI > 0.078) and (Red > 0.0
      return '7Gutter' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI <= 0.472) and (Red <= 0.775) and (NIR <= 0.52) and (NDVI > -0.017) and (NDVI > 0.078) and (Red <= 0.
      return '7Lawn' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI <= 0.472) and (Red <= 0.775) and (NIR <= 0.52) and (NDVI > -0.017) and (NDVI <= 0.078):
      return '1White' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI <= 0.472) and (Red <= 0.775) and (NIR <= 0.52) and (NDVI <= -0.017):
      return '4White' # probability of 100.0% , based on 5 samples
   if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI <= 0.608) and (NDVI > 0.472) and (NIR <= 0.356):
      return '2Pine' # probability of 100.0% , based on 4 samples
```

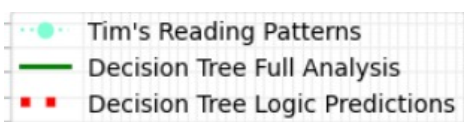
Decision Tree used to Segregate STELLA Test Patterns



if (NIR <= 0.189) and (NDVI > 0.32):
    return '5Lawn' # probability of 100.0% , based on 26 samples
  if (NIR > 0.189) and (Red <= 0.063):
    return '1Hedge' # probability of 100.0% , based on 16 samples
  if (NIR > 0.189) and (Red > 0.063) and (NIR > 0.639) and (NIR <= 0.855):
    return '3Bush' # probability of 100.0% , based on 11 samples
  if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI > 0.608) and (Red <= 0.097):

.
.
.
.

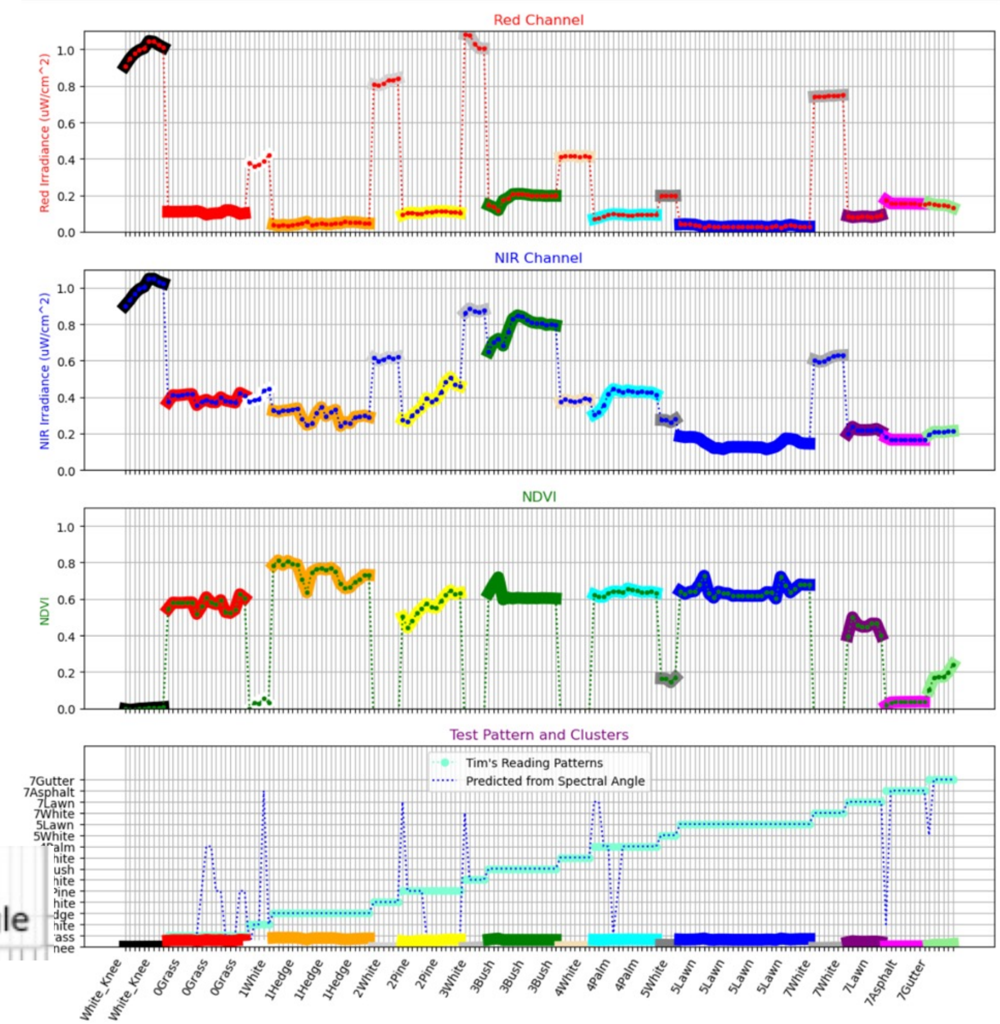


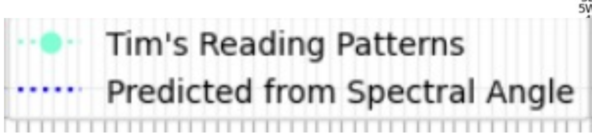


Spectral Angle used to Segregate STELLA Test Patterns

The **Spectral Angle** function takes three **spectral reflectance vectors** (`a`, 'b' and `c`) and calculates the angle between them in spectral space. This angle reflects how similar the spectral signatures of the three vectors are. **A smaller angle indicates higher spectral similarity**, while a larger angle suggests they have different spectral characteristics.

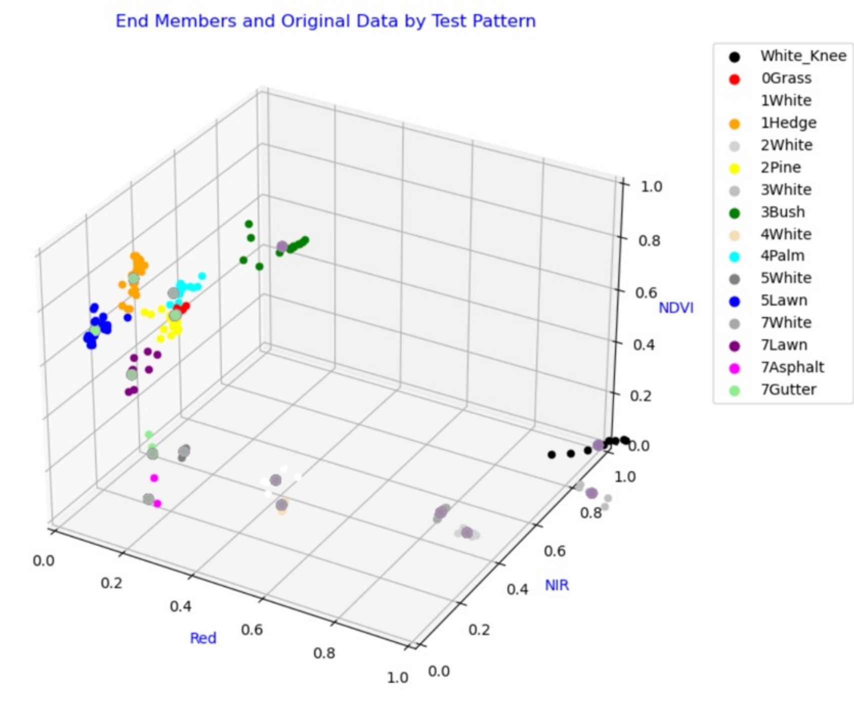
This function is commonly used in spectral unmixing and remote sensing applications where you want to compare the **spectrum of an unknown pixel** with the **spectra of known endmembers (representative materials)** to identify the dominant material present in that pixel.

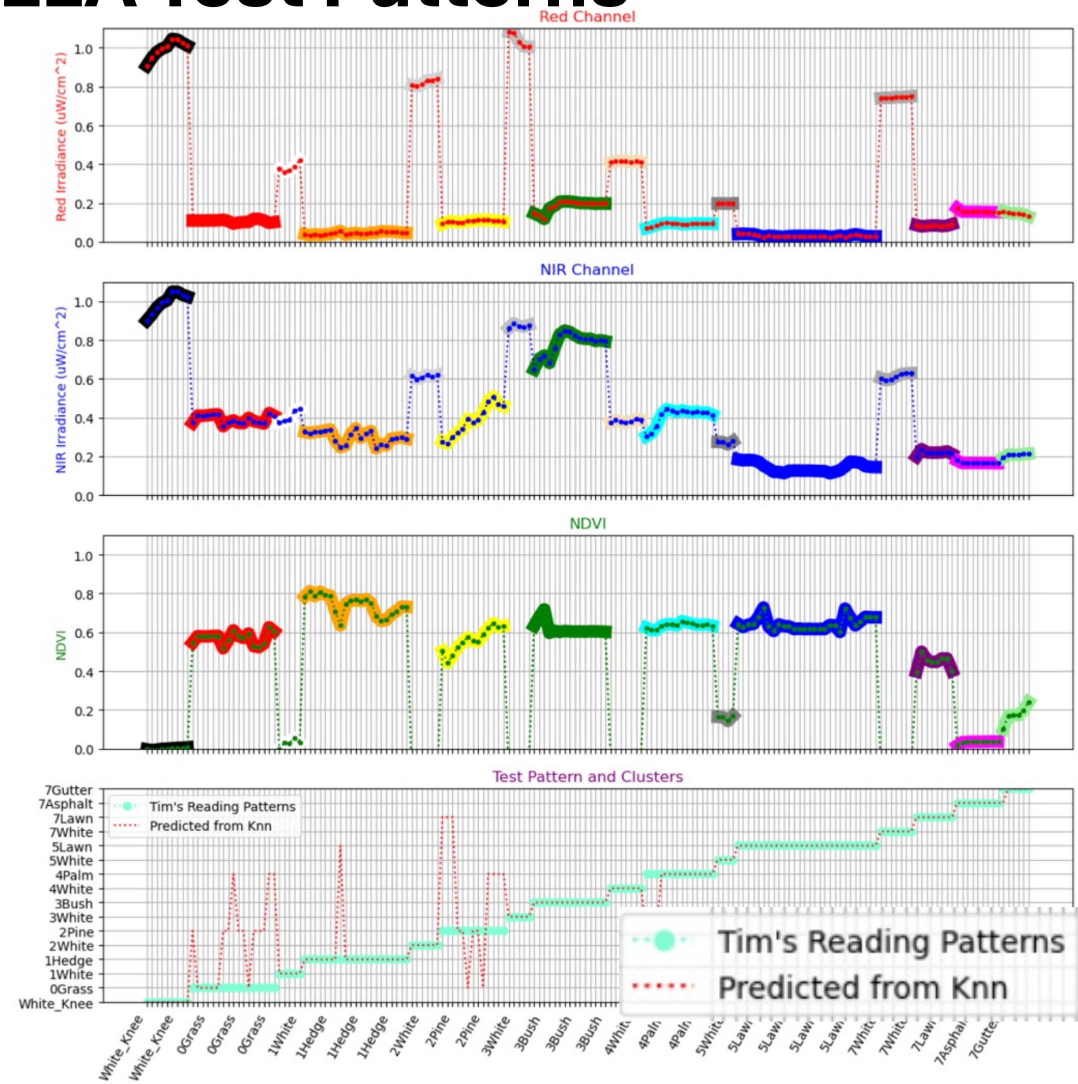




Knn used to Segregate STELLA Test Patterns

Knn is about as simple as it gets. We use **Knn** to relate spectral to our naturally clustering **End Member** data using the inverse square of the Euclidean Distance to weigh and sort our data to predict the **Most Likely** species from our STELLA Red, NIR and calculated NDVI data.





Summary:

Decision Tree, Spectral Angle and Knn Results

The Decision Tree method appears to be The best and this method provides the logic In making these decisions.

```
if (NIR <= 0.189) and (NDVI > 0.32):
    return '5Lawn' # probability of 100.0% , based on 26 samples
  if (NIR > 0.189) and (Red <= 0.063):
    return '1Hedge' # probability of 100.0% , based on 16 samples
  if (NIR > 0.189) and (Red > 0.063) and (NIR > 0.639) and (NIR <= 0.855):
    return '3Bush' # probability of 100.0% , based on 11 samples
  if (NIR > 0.189) and (Red > 0.063) and (NIR <= 0.639) and (NDVI > 0.608) and (Red <= 0.097):
```