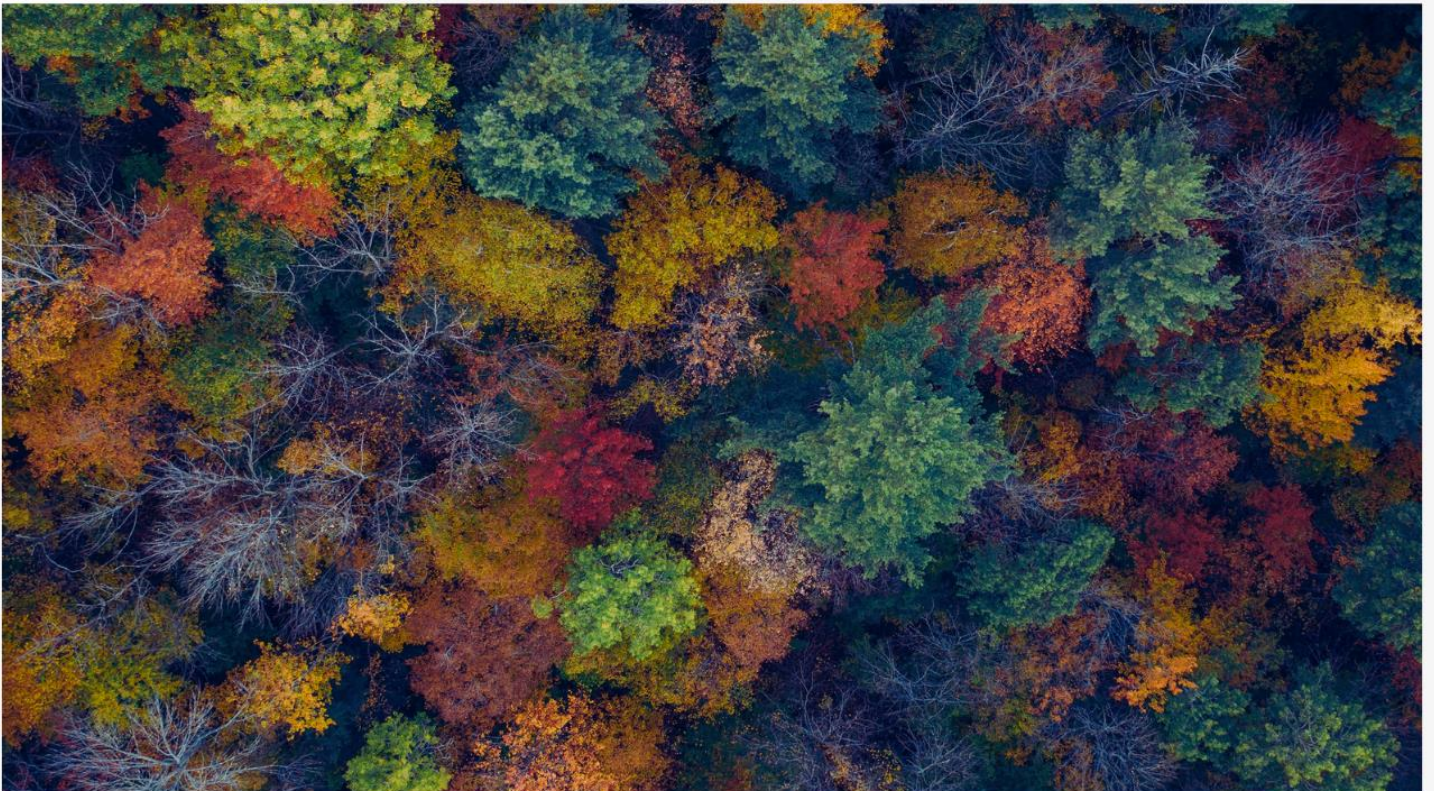# PRIORITY AREAS FOR REFORESTATION

**ANALYZING VULNERABILITY
AS A FUNCTION OF IMPACT AND ADAPTATIVE CAPACITY
IN ORDER TO CLASSIFY PRIORITY AREAS FOR REFORESTATION**

PREPARED AND PRESENTED BY

MARK ANTHONY CABANLIT

PRESENTED TO

THINKING MACHINES

Vulnerability

The Intergovernmental Panel on Climate Change defines Vulnerability as "the degree to which a system is susceptible to, or unable to cope with, adverse effects of climate change, including climate variability and extremes." It is a function of (1) the degree of the system's Exposure to climate hazards; (2) its Sensitivity to such hazards; and (3) its Adaptive Capacity (IPCC, 2001)
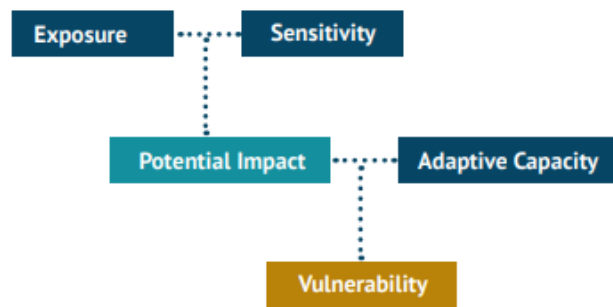
Adaptive Capacity

These are the layers used for calculating the adaptive capacity.

Political Category

This layer divides the data into the following categories. The shapefile was derived from the Phil-LiDAR Project of the Department of Science and Technology. Most cities would have more money to provide aid, thus higher adaptive capacity.

> 5 – City
> 3 - Municipality

Regional GDP

The regional gross domestic product was accessed from PSA. The values were then divided into five classes as shown in the layer properties below.
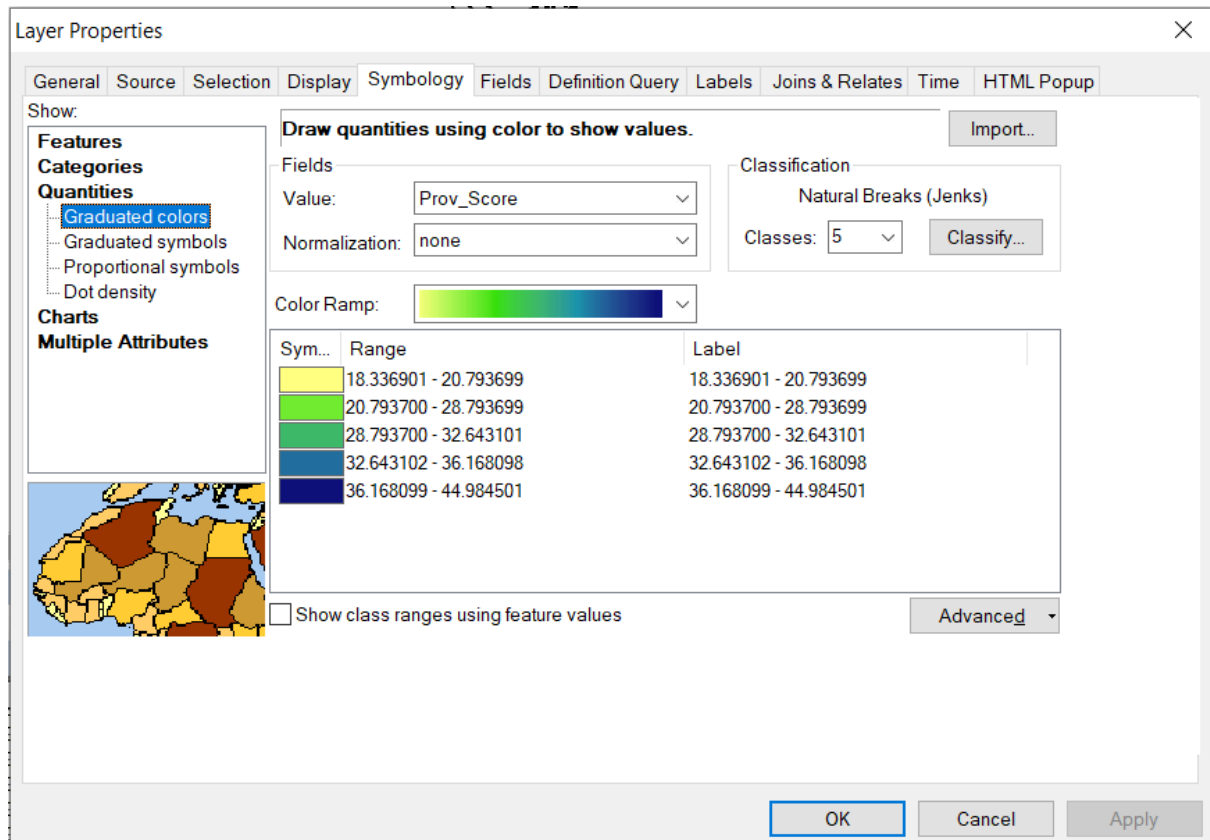
- https://psa.gov.ph/grdp/grdp-id/167338
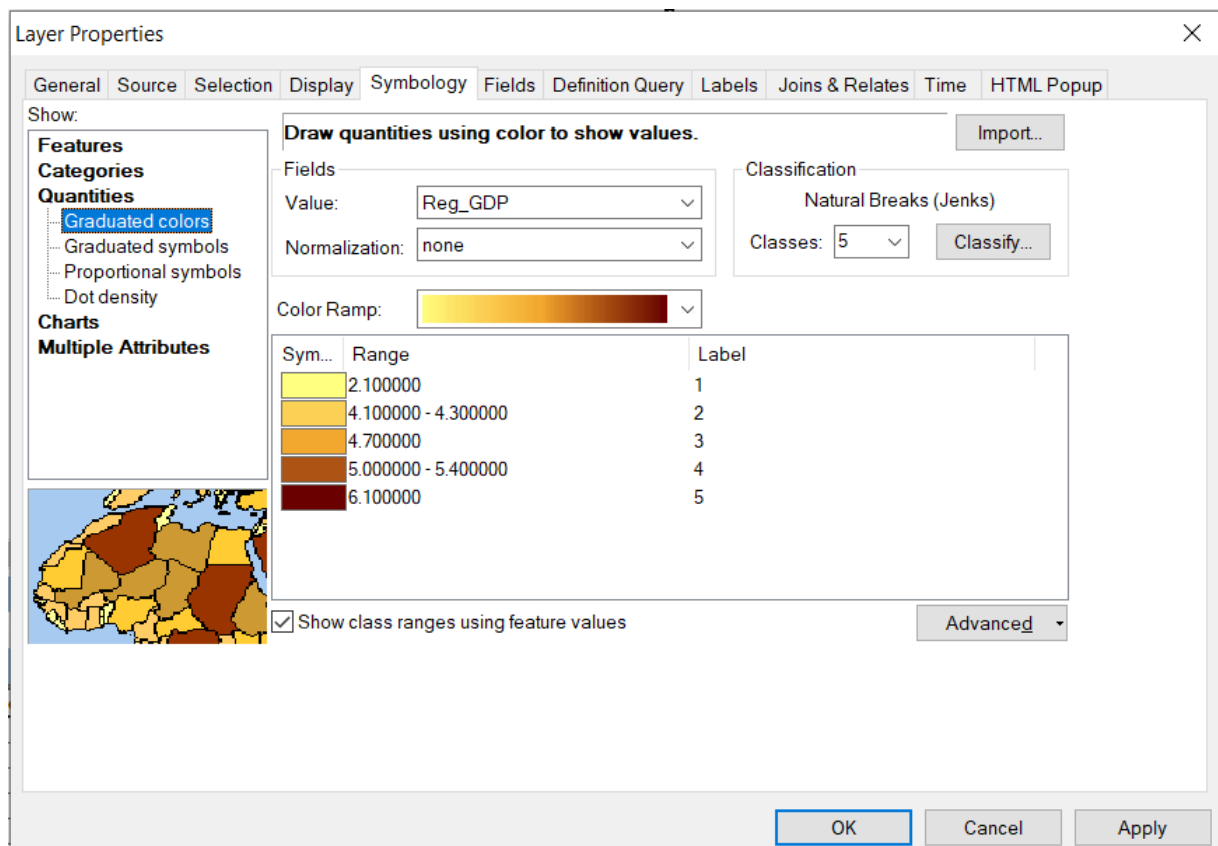
Provincial Score

This is a score derived from the Department of Trade and Industry. Rankings of Cities and Municipalities are based on the sum of their scores on 4 Pillars: Economic Dynamism, Government Efficiency, Infrastructure, and Resiliency. Provincial rankings are based on population and income weighted average of the Overall scores of cities and municipalities under a province.

- https://cmci.dti.gov.ph/rankings-data.php?unit=Provinces

Layer Properties

General Source Selection Display **Symbology** Fields Definition Query Labels Joins & Relates Time HTML Popup

Show:

Features
Categories
Quantities
   Graduated colors
   Graduated symbols
   Proportional symbols
   Dot density
Charts
Multiple Attributes

Draw quantities using color to show values.   Import...

Fields
Value: Prov_Score
Normalization: none

Classification
Natural Breaks (Jenks)
Classes: 5   Classify...

Color Ramp:

Sym...	Range	Label
	18.336901 - 20.793699	18.336901 - 20.793699
	20.793700 - 28.793699	20.793700 - 28.793699
	28.793700 - 32.643101	28.793700 - 32.643101
	32.643102 - 36.168098	32.643102 - 36.168098
	36.168099 - 44.984501	36.168099 - 44.984501

☐ Show class ranges using feature values   Advanced ▾

OK   Cancel   Apply

```
def reclass(x):
  if x <= 20.793699 :
    return 1
  elif x >= 20.793700 and x <= 28.793699 :
    return 2
  elif x >= 28.793700 and x <= 32.643101 :
    return 3
  elif x >= 32.643102 and x <= 36.168098 :
    return 4
  else:
    return 5

reclass( !Prov_Score! )
```

Land Cover

The land cover data was retrieved from below:

-

<all other values>
DESCRIPT

- Arable land, crops mainly cereals and sugar
- Built-up Area
- Closed canopy, mature trees covering > 50 percent
- Coconut plantations
- Coral Reef
- Crop land mixed with coconut plantation
- Crop land mixed with other plantation
- Cultivated Area mixed with brushland/grassland
- Eroded area
- Fishponds derived from mangrove
- Grassland, grass covering > 70 percent
- Lake
- Mangrove vegetation
- Marshy area and swamp
- Mossy forest
- Open canopy, mature trees covering < 50 percent
- Other barren land
- Other fishponds
- Other plantations
- Pine forest
- Quarry
- Riverbeds
- Siltation pattern in lake
- Unclassified

-

```
"DESCRIPT" in ( 'Arable land, crops mainly cereals and sugar' , 'Closed canopy,
mature trees covering > 50 percent', 'Coconut plantations', 'Crop land mixed with
coconut plantation', 'Crop land mixed with other plantation', 'Cultivated Area
mixed with brushland/grassland', 'Grassland, grass covering > 70 percent',
'Mangrove vegetation', 'Marshy area and swamp', 'Mossy forest', 'Open canopy,
mature trees covering < 50 percent', 'Other plantations', 'Pine forest')
```

Mean Green

The value was extracted from ESRI RGB green band.


```
def reclass(x):
  if x <= 58.453701 :
    return 1
  elif x >= 58.453702 and x <= 69.981697 :
    return 2
  elif x >= 69.981698 and x <= 80.967003:
    return 3
  elif x >= 80.967004 and x <= 95.461304 :
    return 4
  else:
    return 5

reclass( !Green_Val!)
```

Change in Global Tree Cover on Agricultural Land – 2000-2010

This data was retrieved from: http://apps.worldagroforestry.org/global-tree-cover/data-download.html

```
def reclass(x):
  if x <= -2.793330 :
    return 1
  elif x >= -2.793329 and x <= 0.785714 :
    return 2
  elif x >= 0.785715 and x <= 3.898620 :
    return 3
  elif x >= 3.898621 and x <= 7.75:
    return 4
  else:
    return 5

reclass( !TreeCover! )
```
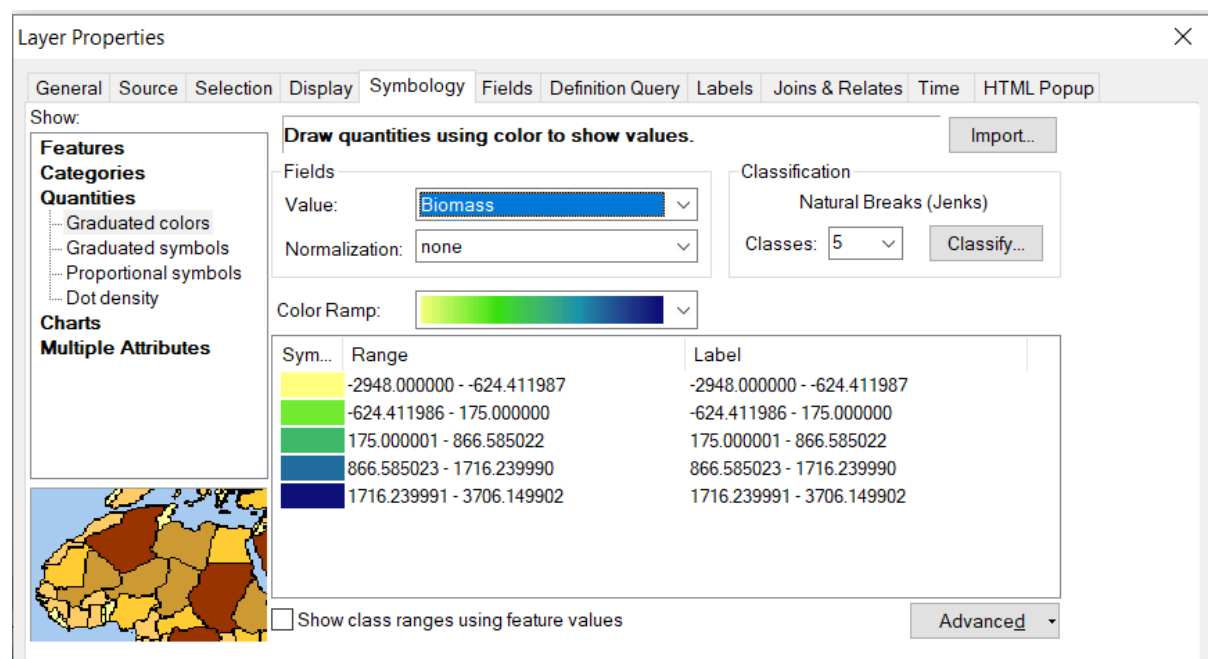
Change in Global Biomass Carbon on Agricultural Land – 2000-2010

This data was retrieved from: http://apps.worldagroforestry.org/global-tree-cover/data-download.html


```
def reclass(x):
  if x <= -624.411987 :
    return 1
  elif x >= -624.411986 and x <= 175 :
    return 2
  elif x >= 175.000001 and x <= 866.585022 :
    return 3
  elif x >= 866.585023 and x <= 1716.239990:
    return 4
  else:
    return 5

reclass( !Biomass! )
```
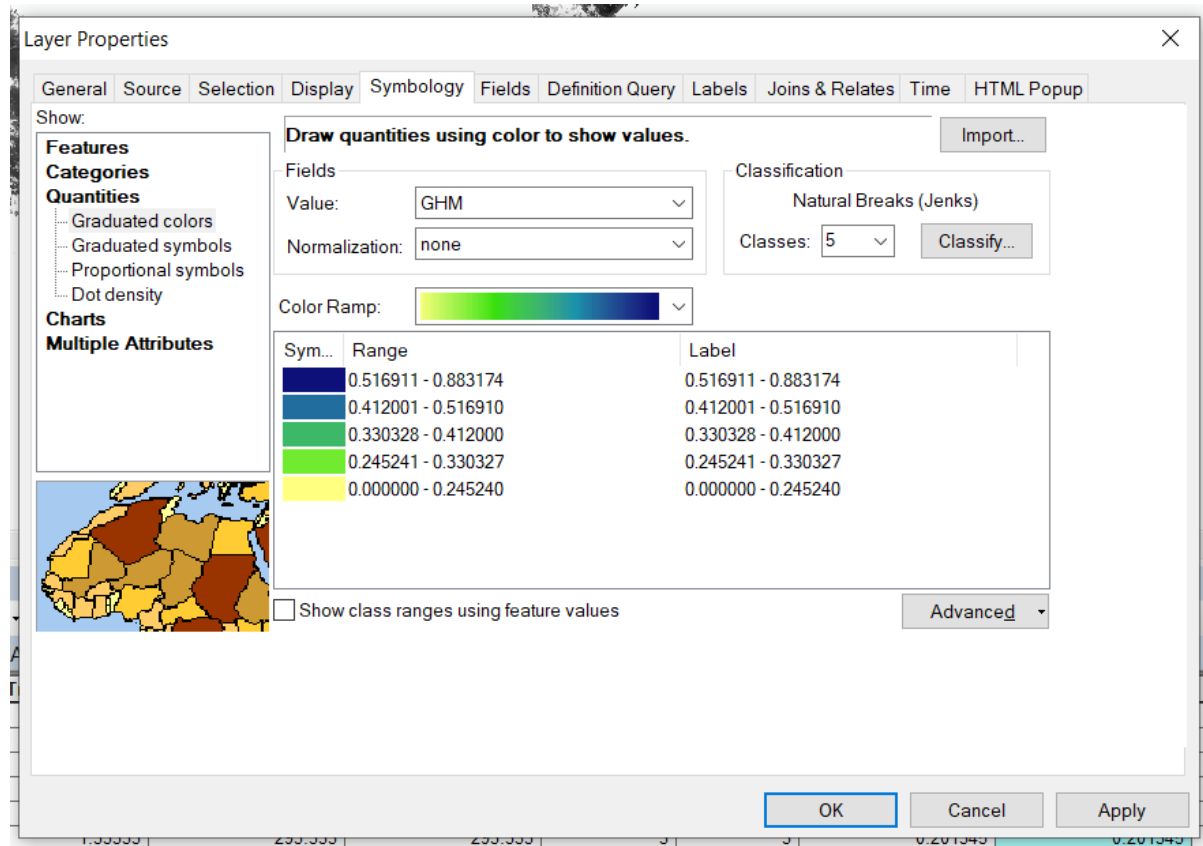
Global Human Modification

The data was retrieved from:


```python
def reclass(x):
  if x <= 0.245240 :
    return 5
  elif x >= -0.245241 and x <= 0.330327 :
    return 4
  elif x >= 0.330328 and x <= 0.412 :
    return 3
  elif x >= 0.4120001 and x <= 0.516910:
    return 2
  else:
    return 1

reclass( !GHM!)
```

Normalized DifferenceVegetation Index

Using google earth engine, calculate for NDVI before and after Odette to estimate the change in cover.

```javascript
function addNDVI(image) {
  var ndvi = image.normalizedDifference(['B5', 'B4']);
  return image.addBands(ndvi)
```

```
}
var visPaaramsTrue = {bands: ['B4', 'B3', 'B2'], min: 0, max: 0.3, gamma: 1.4};

var before_odette = ee.ImageCollection("LANDSAT/LC08/C02/T1_TOA")
.filterDate('2021-01-01', '2021-12-15')
.filterBounds(roi)
.min();

var after_odette = ee.ImageCollection("LANDSAT/LC08/C02/T1_TOA")
.filterDate('2021-12-17', '2022-03-30')
.filterBounds(roi)
.min();

var ndvi_before_odette = addNDVI(before_odette)
var ndvi_after_odette = addNDVI(after_odette)
Map.addLayer(ndvi_before_odette, {min:0, max:0.5}, 'NDVI Before Odette')
Map.addLayer(ndvi_after_odette, {min:0, max:0.5}, 'NDVI After Odette')
Map.addLayer(before_odette, visPaaramsTrue, 'Before Odette')
Map.addLayer(after_odette, visPaaramsTrue, 'After Odette')
Map.centerObject(roi, 8);

var area =
ee.FeatureCollection('users/kalyejenner/odette_affected_municipalities');
Map.addLayer(area, {}, 'Affected Areas')


Export.image.toDrive({
  image: before_odette.float(),
  description: 'BeforeOdette',
  scale: 30,
  region: roi
  });
Export.image.toDrive({
  image: after_odette.float(),
  description: 'AfterOdette',
  scale: 30,
  region: roi
  });
```

Impact

These are the layers used for calculating the impact. Most of the values on this layer was retrieved from the National Disaster Risk Reduction's monitoring dashboard.

- https://monitoring-dashboard.ndrrmc.gov.ph/page/situation/situational-report-for-tc-odette-2021

Casualties

REGION	CONFIRMED			FOR VALIDATION			TOTAL REPORTED		
	dead	injured	missing	dead	injured	missing	dead	injured	missing
TOTAL	101	318	11	304	1053	41	405	1371	52
MIMAROPA	18	0	0	7	2	4	25	2	4
Region 6	54	170	7	0	3	0	54	173	7
Region 7	2	2	0	218	544	32	220	546	32
Region 8	18	112	4	8	0	0	26	112	4
Region 9	1	2	0	0	0	0	1	2	0
Region 10	8	28	0	0	0	0	8	28	0
CARAGA	0	4	0	71	504	5	71	508	5

Table title: CASUALTIES

Number of Deaths


```
def reclass(x):
  if x >= 72:
    return 5
  elif x >= 27 and x <= 71:
    return 4
  elif x >= 9 and x <= 26:
    return 3
  elif x >= 2 and x <= 8:
    return 2
  else:
    return 1

reclass( !Death! )
```

Number of Injured

General | Source | Selection | Display | **Symbology** | Fields | Definition Query | Labels | Joins & Relates | Time | HTML Popup

Show:

Features
Categories
Quantities
 — Graduated colors
 — Graduated symbols
 — Proportional symbols
 — Dot density
Charts
Multiple Attributes

Draw quantities using color to show values.   Import...

Fields

Value: Injured ⌄

Normalization: none ⌄

Classification

Natural Breaks (Jenks)

Classes: 5 ⌄   Classify...

Color Ramp: ⌄

Sym...	Range	Label
	174 - 546	174 - 546
	113 - 173	113 - 173
	29 - 112	29 - 112
	3 - 28	3 - 28
	0 - 2	0 - 2

☐ Show class ranges using feature values   Advanced ▾
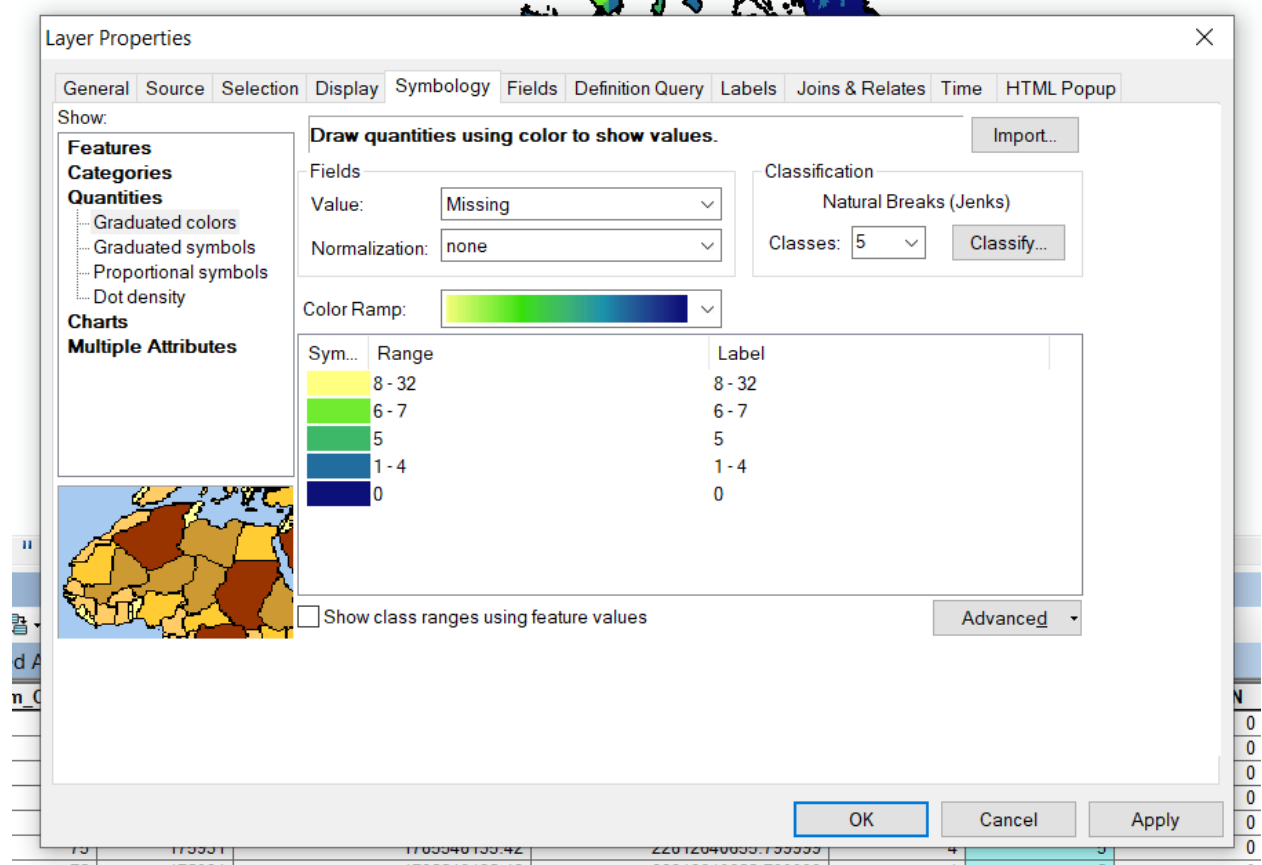
OK   Cancel   Apply

```
def reclass(x):
  if x >= 174:
    return 5
  elif x >= 113 and x <= 173:
    return 4
  elif x >= 29 and x <= 112:
    return 3
  elif x >= 3 and x <= 28:
    return 2
  else:
    return 1
reclass( !Injured! )
```

Number of Missing People

Layer Properties

General | Source | Selection | Display | **Symbology** | Fields | Definition Query | Labels | Joins & Relates | Time | HTML Popup

Show:

Features
Categories
Quantities
   Graduated colors
   Graduated symbols
   Proportional symbols
   Dot density
Charts
Multiple Attributes

Draw quantities using color to show values.　　Import...

Fields

Value: Missing

Normalization: none

Classification

Natural Breaks (Jenks)

Classes: 5　Classify...

Color Ramp:

Sym...	Range	Label
	8 - 32	8 - 32
	6 - 7	6 - 7
	5	5
	1 - 4	1 - 4
	0	0

☐ Show class ranges using feature values

Advanced ▾

OK　Cancel　Apply

```python
def reclass(x):
  if x >= 8:
    return 5
  elif x >= 6 and x <= 7:
    return 4
  elif x >= 5 and x <= 5:
    return 3
  elif x >= 1 and x <= 4:
    return 2
  else:
    return 1

reclass( !Missing! )
```

Roads and Bridges

ROADS AND BRIDGES

	NOT PASSABLE		PASSABLE (previously reported as not passable)	
	ROADS	BRIDGES	ROADS	BRIDGES
GRAND TOTAL	8	1	233	26
MIMAROPA	0	0	23	8
Region 6	0	0	43	1
Region 7	8	1	64	5
Region 8	0	0	55	8
Region 10	0	0	27	3
CARAGA	0	0	21	1

Damaged Roads


```
def reclass(x):
  if x >= 56:
    return 5
  elif x >= 28 and x <= 55:
    return 4
  elif x >= 24 and x <= 27:
    return 3
  elif x >= 1 and x <= 23:
    return 2
  else:
```

```
        return 1

reclass( !Blk_Roads! )
```
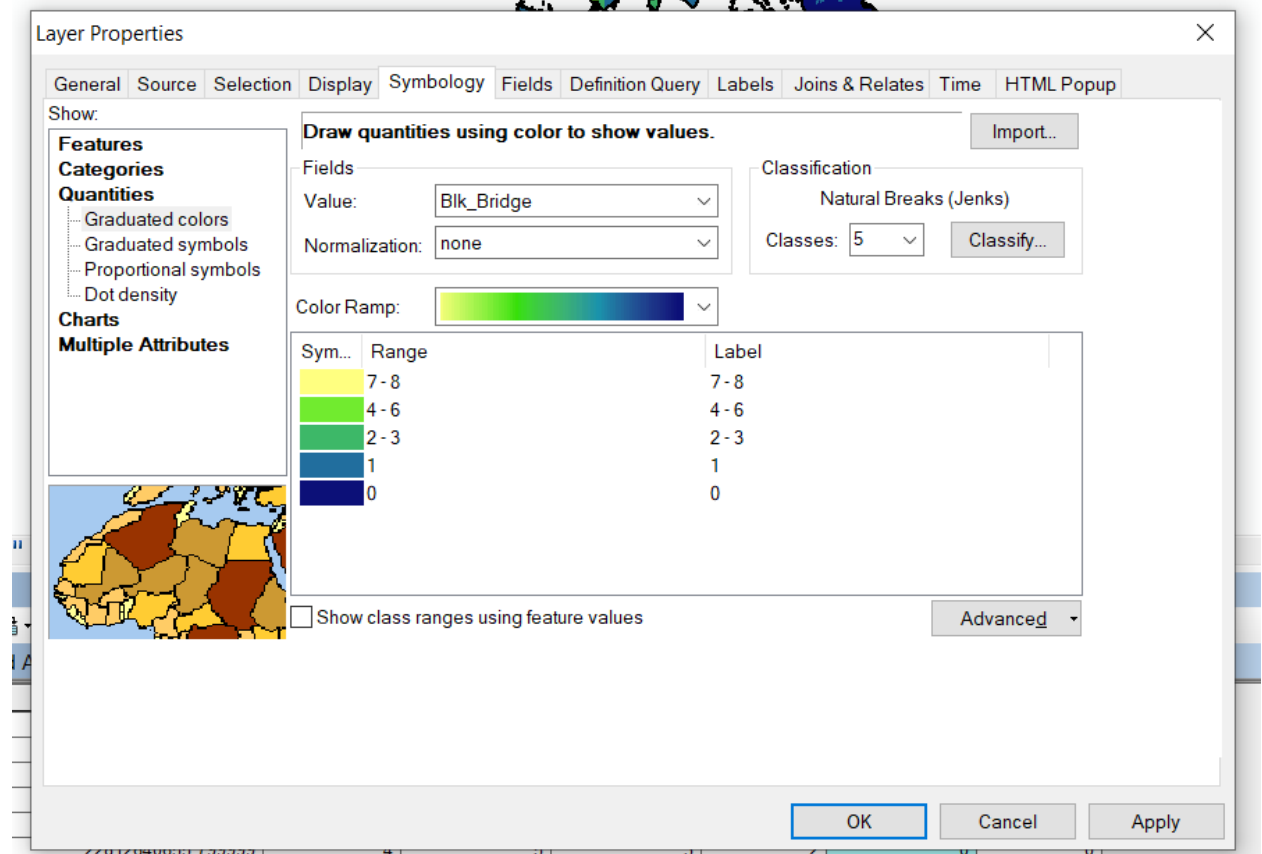
Damaged Bridges


```
def reclass(x):
  if x >= 7:
    return 5
  elif x >= 4 and x <= 6:
    return 4
  elif x >= 2 and x <= 3:
    return 3
  elif x >= 1 and x <= 1:
    return 2
  else:
    return 1

reclass( !Blk_Bridge! )
```
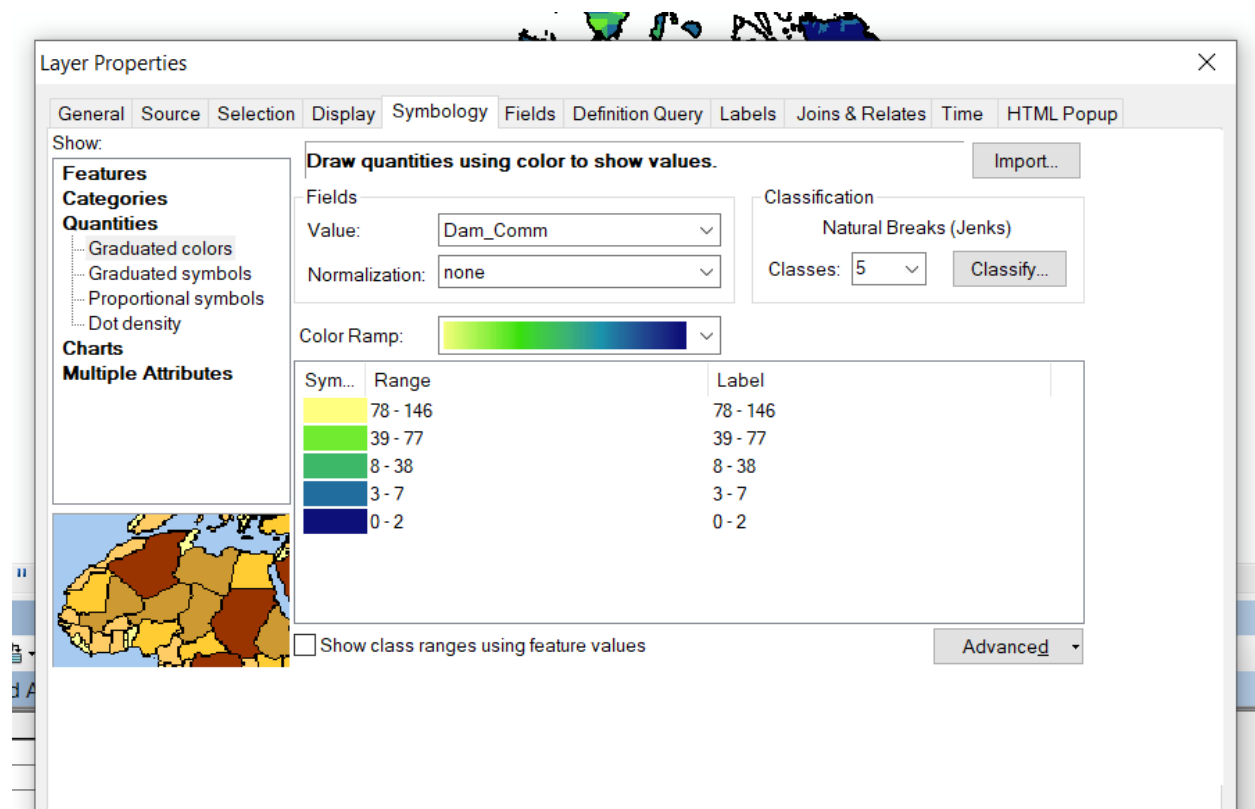
Communication Lines

COMMUNICATION LINES

There are 76 areas in MIMAROPA, Region 7, Region 8, CARAGA, BARMM which are still experiencing communication outage/issues.

Region	No of Areas	
	Without Communication	Restored Communication Lines
GRAND TOTAL	76	473
MIMAROPA	3	35
Region 6	0	133
Region 7	24	122
Region 8	1	1
Region 9	0	71
Region 10	0	7
CARAGA	8	67
BARMM	40	37

Layer Properties   ✕

General | Source | Selection | Display | **Symbology** | Fields | Definition Query | Labels | Joins & Relates | Time | HTML Popup

Show:

Features
Categories
Quantities
   Graduated colors
   Graduated symbols
   Proportional symbols
   Dot density
Charts
Multiple Attributes

Draw quantities using color to show values.   Import...

Fields

Value: Dam_Comm

Normalization: none

Classification

Natural Breaks (Jenks)

Classes: 5   Classify...

Color Ramp:

Sym...	Range	Label
	78 - 146	78 - 146
	39 - 77	39 - 77
	8 - 38	8 - 38
	3 - 7	3 - 7
	0 - 2	0 - 2

☐ Show class ranges using feature values   Advanced ▾

```
def reclass(x):
  if x >= 78:
    return 5
  elif x >= 39 and x <= 77:
    return 4
  elif x >= 8 and x <= 38:
    return 3
  elif x >= 3 and x <= 7:
    return 2
  else:
    return 1

reclass( !Dam_Comm! )
```

Damaged Houses

DAMAGED HOUSES

A total of 2,031,824 damaged house/s are reported in MIMAROPA, Region 6, Region 7, Region 8, Region 9, Region 10, Region 11, Region 12, CARAGA, BARMM.

REGION	PARTIALLY	TOTALLY	TOTAL	AMOUNT (PHP)
GRAND TOTAL	1,585,252	446,572	2,031,824	62,675,818
MIMAROPA	74,319	15,465	89,784	0
Region 6	362,743	61,878	424,621	0
Region 7	900,558	245,659	1,146,217	0
Region 8	135,489	52,221	187,710	0
Region 9	103	136	239	0
Region 10	6,389	857	7,246	62,225,818
Region 11	0	2	2	0
Region 12	0	1	1	0
CARAGA	105,591	70,340	175,931	0
BARMM	60	13	73	450,000

Layer Properties

General | Source | Selection | Display | Symbology | Fields | Definition Query | Labels | Joins & Relates | Time | HTML Popup

Show:
- Features
- Categories
- **Quantities**
  - Graduated colors
  - Graduated symbols
  - Proportional symbols
  - Dot density
- Charts
- Multiple Attributes

Draw quantities using color to show values.   Import...

Fields
Value: Dam_House
Normalization: none

Classification
Natural Breaks (Jenks)
Classes: 5   Classify...

Color Ramp:

Sym...	Range	Label
	424622 - 1146217	424622 - 1146217
	187711 - 424621	187711 - 424621
	89785 - 187710	89785 - 187710
	7247 - 89784	7247 - 89784
	2 - 7246	2 - 7246

☐ Show class ranges using feature values   Advanced ▾

OK   Cancel   Apply

```
def reclass(x):
  if x >= 424622 :
    return 5
  elif x >= 187711 and x <= 424621:
    return 4
  elif x >= 89785  and x <= 187710:
    return 3
  elif x >= 7247  and x <= 89784:
```

```
      return 2
  else:
      return 1

reclass( !Dam_House!)
```

Damage to Agriculture

Layer Properties

General | Source | Selection | Display | Symbology | Fields | Definition Query | Labels | Joins & Relates | Time | HTML Popup

Show:
Features
Categories
Quantities
  Graduated colors
  Graduated symbols
  Proportional symbols
  Dot density
Charts
Multiple Attributes

Draw quantities using color to show values.   Import...

Fields
Value: Dam_AgriP
Normalization: none

Classification
Natural Breaks (Jenks)
Classes: 5   Classify...

Color Ramp:

Sym...	Range	Label
	2529794285.000001 - 6182679072.060000	2529794285.000001 - 6182679072.060000
	1785548135.420001 - 2529794285.000000	1785548135.420001 - 2529794285.000000
	1284610039.000000 - 1785548135.420000	1284610039.000000 - 1785548135.420000
	54784894.790001 - 1284610039.000000	54784894.790001 - 1284610039.000000
	0.000000 - 54784894.790000	0.000000 - 54784894.790000

☐ Show class ranges using feature values   Advanced ▾

OK   Cancel   Apply

```
def reclass(x):
  if x >= 2529794285.000001 :
    return 5
  elif x >= 1785548135.420001 and x <= 2529794285.000000:
    return 4
  elif x >= 1284610039.000001 and x <= 1785548135.420000:
    return 3
  elif x >= 54784894.790001 and x <= 1284610039.000000:
    return 2
  else:
    return 1
reclass( !Dam_AgriP!)
```

Damage to Infrastructure

DAMAGE TO INFRASTRUCTURE

Estimated cost of damage to infrastructure of Php 29,338,185,355.94 was incurred in MIMAROPA, Region 7, Region 8, Region 10, Region 12, CARAGA, BARMM.

REGION	NUMBER OF DAMAGED INFRASTRUCTURE	COST OF DAMAGE (PHP)
GRAND TOTAL	1,733	29,338,185,355.94
MIMAROPA	1,181	1,437,594,142.18
Region 7	469	4,205,649,557.92
Region 8	20	173,400,000
Region 10	55	688,901,000
Region 12	1	20,000,000
CARAGA	0	22,812,640,655.84
BARMM	7	0

```
def reclass(x):
  if x >= 4205649557.920001 :
    return 5
  elif x >= 1437594142.180001 and x <= 4205649557.920000:
    return 4
  elif x >= 688901000.000001 and x <= 1437594142.1800000:
    return 3
  elif x >= 173400000.000001  and x <= 688901000.000000:
    return 2
  else:
    return 1

reclass( !Dam_InfrP!)
```

Calculating for Impact

```
IMPACT =
[IMP_DEATH]+ [IMP_INJURE]+ [IMP_MISSIN]+ [IMP_ROAD]+ [IMP_BRIDGE]+ [IMP_COMM]+
[IMP_HOUSE]+ [IMP_AGRI]+ [IMP_INFRA]+ [IMP_PATH]
```

If the no. of criteria = 10	
Maximum score	(10 x 5) = 50
Minimum score	(10 x 1) = 10
Total range	[max - min] = 40
Intervals	40 ÷ 3 = 13.3 or 13
Interval	40/3
	13.3

Rating	Range
Low	10-23
Medium	24-37
High	38-50

```
def reclass(x):
  if x >= 38:
    return 'H'
  elif x >= 24 and x <= 37:
    return 'M'
  else:
    return 'L'

reclass( !IMPACT!)
```

Calculating for Adaptive Capacity

```
ADCAP = [ADC_POLITI]+ [ADC_GDP_R]+ [ADC_PROV]+ [ADC_GREEN]+ [ADC_TreeC]+
[ADC_BioC]+ [ADC_GHM]+ [ADC_LCOVER]
```

If the no. of criteria = 8	
Maximum score	(8 x 5) = 40
Minimum score	(8 x 1) = 8
Total range	[max - min] = 32
Intervals	32 ÷ 3 = 10.7 or 11
Interval	32/3
	10.7

Rating	Range
Low	8-18
Medium	19-29
High	30-40

```python
def reclass(x):
  if x >= 30:
    return 'H'
  elif x >= 19 and x <= 29:
    return 'M'
  else:
    return 'L'

reclass( !ADCAP! )
```

Calculating Vulnerability

Vulnerability

		Adaptive Capacity		
		L	M	H
Potential Impact	L	M	L	L
	M	H	M	L
	H	H	H	M

```python
def reclass(im, ac):
  if im == 'L' and ac == 'L' :
    return 'M'
  elif im == 'L' and ac == 'M':
    return 'L'
  elif im == 'L' and ac == 'H':
    return 'L'
  elif im == 'M' and ac == 'L':
    return 'H'
  elif im == 'M' and ac == 'M':
    return 'M'
```

```
    elif im == 'M' and ac == 'H':
        return 'L'
    elif im == 'H' and ac == 'L':
        return 'H'
    elif im == 'H' and ac == 'M':
        return 'H'
    else:
        return 'M'

reclass( !ADCAP_LMH! , !IMPACT_LMH! )
```