# Spectral

## **Awesome Spectral Indices** for the **Google Earth Engine** JavaScript API

**David Montero, César Aybar, Miguel D. Mahecha,** and **Sebastian Wieneke**

@dmlmont @davemlz

UNIVERSITÄT LEIPZIG

RSC4Earth

Experts

FOSS4G FIRENZE 2022

“How do we usually compute **Spectral Indices** in **Google Earth Engine**?”

$$\mathrm{DVI} = \rho_{\mathrm{NIR}} - \rho_{\mathrm{red}}$$

```
DVI = my_image.select("NIR").subtract(my_image.select("RED"))
```

Source: https://doi.org/10.1016/0034-4257(94)00114-3

$$\mathrm{NDVI} = \frac{\rho_{\mathrm{NIR}} - \rho_{\mathrm{red}}}{\rho_{\mathrm{NIR}} + \rho_{\mathrm{red}}}$$

```
NDVI = my_image.normalizedDifference(["NIR", "RED"])
```

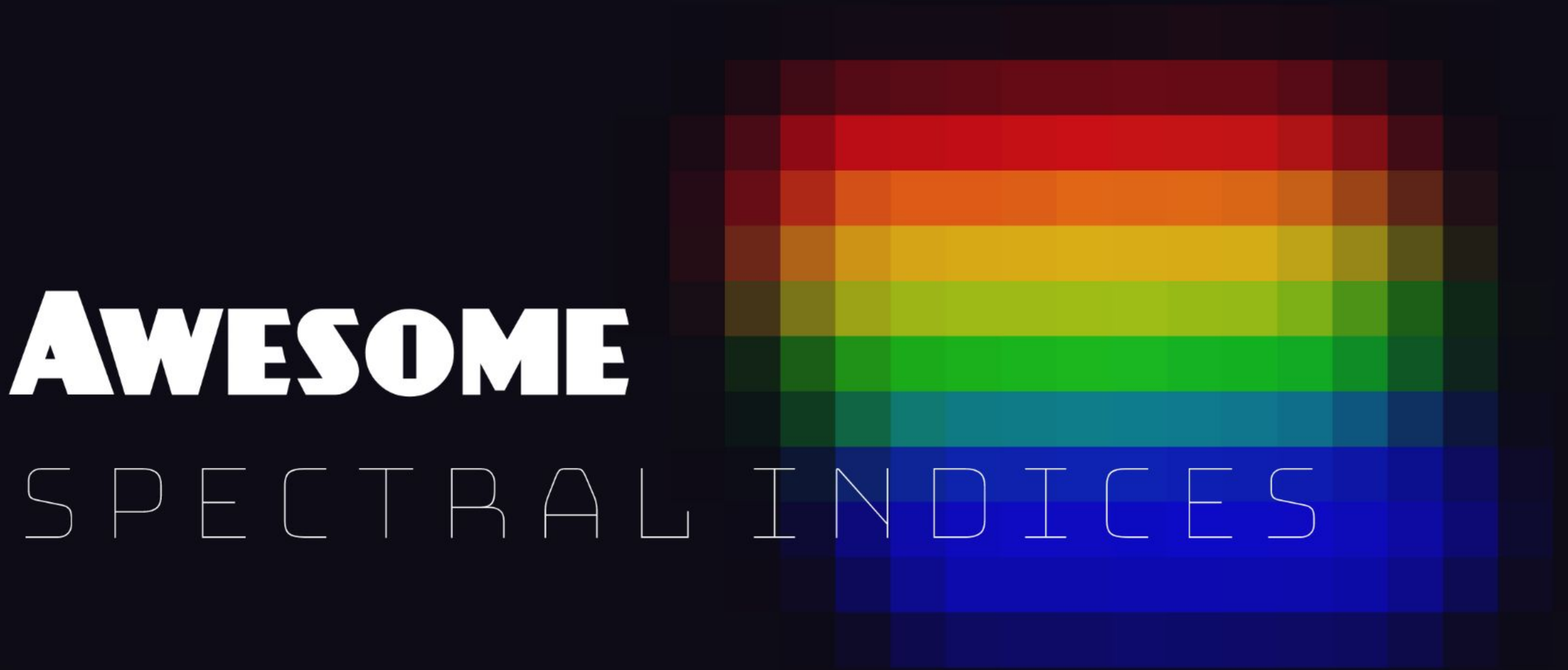
Source: https://ntrs.nasa.gov/citations/19740022614

$$\mathrm{EVI} = \frac{2.5(\rho_{\mathrm{NIR}} - \rho_{\mathrm{red}})}{\rho_{\mathrm{NIR}} + 6\rho_{\mathrm{red}} - 7.5\rho_{\mathrm{blue}} + 1}$$

```
EVI = my_image.expression(
      “2.5 * (N - R)/(N + 6.0 * R - 7.5 * B + 1.0)”,
      {
            “N”: my_image.select(“NIR”),
            “R”: my_image.select(“RED”),
            “B”: my_image.select(“BLUE”),
      })
```

Source: https://doi.org/10.1016/S0034-4257(96)00112-5

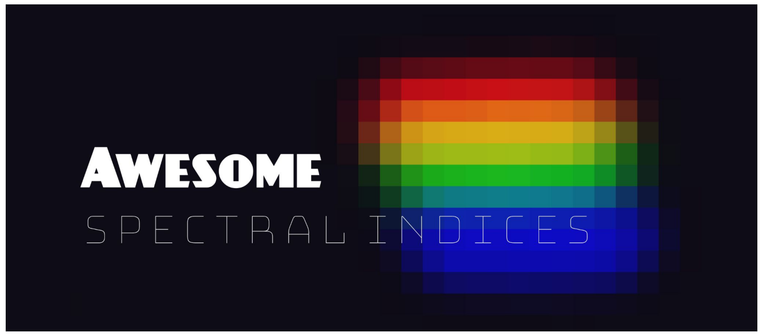
“What if we don’t need to **hard-code** every spectral index we want to compute?”




Source: https://github.com/awesome-spectral-indices/awesome-spectral-indices

# 201

Spectral Indices (as of version 0.1.0)




**Catalogue**
Awesome Spectral Indices


**Python Package**
spyndex


**Streamlit App**
Espectro

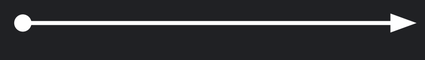
Source: https://github.com/awesome-spectral-indices

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```


vegetation
water
burn
snow
urban
kernel
radar

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```

Contributor's github

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```


When was the spectral index added to ASI

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```

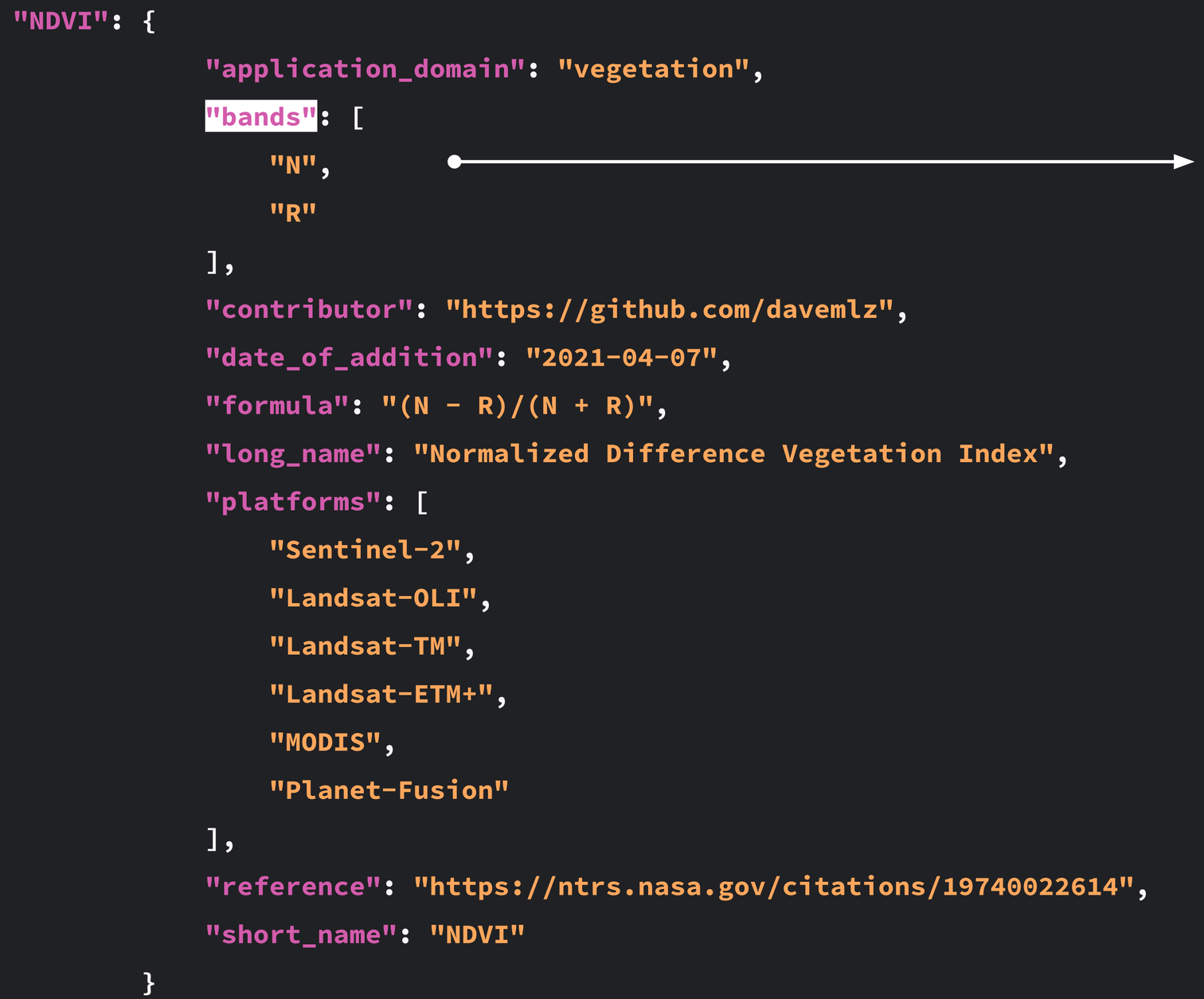

Long name of the spectral index

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```


Some satellite platforms that can be used for computing the spectral index

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```


Paper, DOI, link, or reference to the original spectral index




| | |
|---|---|
| A | Aerosols |
| B | Blue |
| G | Green |
| R | Red |
| RE1 | Red Edge 1 |
| RE2 | Red Edge 2 |
| RE3 | Red Edge 3 |
| N | NIR |
| N2 | NIR 2 |
| S1 | SWIR1 |
| S2 | SWIR2 |
| T1 | TIR1 |
| T2 | TIR2 |

\+ additional parameters, e.g.

| | |
|---|---|
| L | Canopy Background Adjustment |


"NDVI": {
"application_domain": "vegetation",
"bands": [
"N",
"R"
],
"contributor": "https://github.com/davemlz",
"date_of_addition": "2021-04-07",
"formula": "(N - R)/(N + R)",
"long_name": "Normalized Difference Vegetation Index",
"platforms": [
"Sentinel-2",
"Landsat-OLI",
"Landsat-TM",
"Landsat-ETM+",
"MODIS",
"Planet-Fusion"
],
"reference": "https://ntrs.nasa.gov/citations/19740022614",
"short_name": "NDVI"
}
Standardized expression

“How can we use the **Awesome Spectral Indices** catalogue to compute spectral indices in **Google Earth Engine**?”

# ee.Image.expression



Evaluates an arithmetic expression on an image, possibly involving additional images.

The bands of the primary input image are available using the built-in function b(), as b(0) or b('band_name').

Variables in the expression are interpreted as additional image parameters which must be supplied in opt_map. The bands of each such image can be accessed like image.band_name or image[0].

Both b() and image[] allow multiple arguments, to specify multiple bands, such as b(1, 'name', 3). Calling b() with no arguments, or using a variable by itself, returns all bands of the image.

If the result of an expression is a single band, it can be assigned a name using the '=' operator (e.g.: x = a + b).

Returns the image computed by the provided expression.

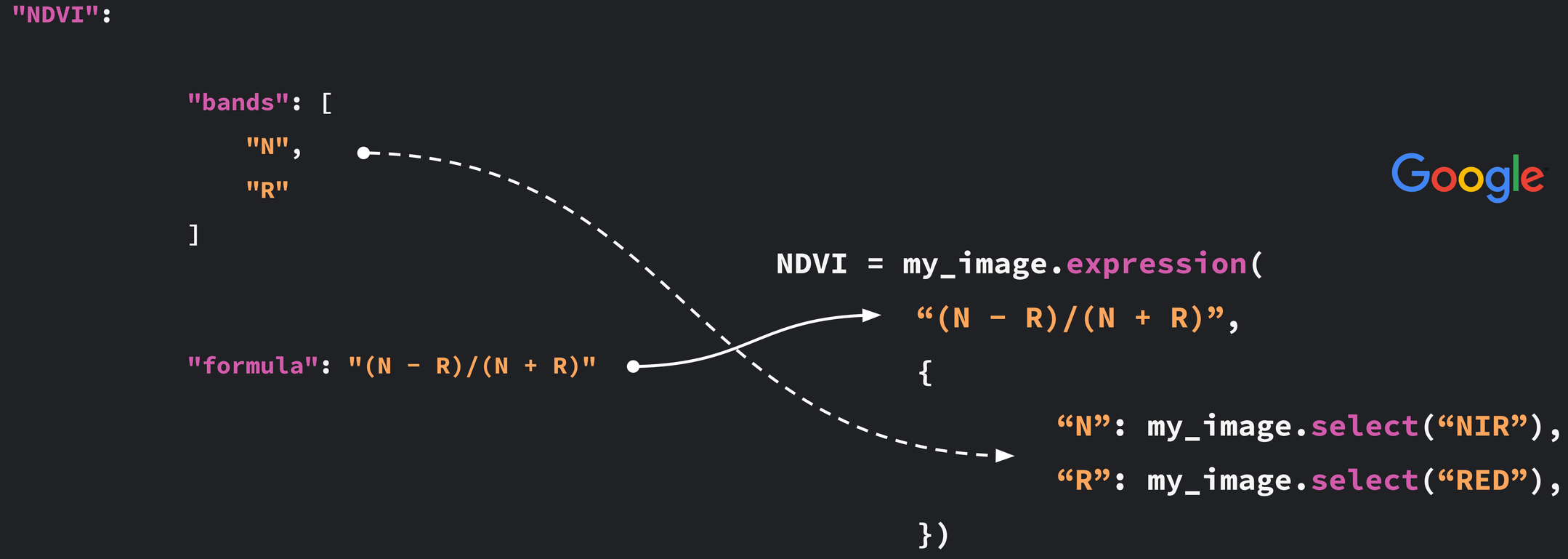
| Usage | Returns |
|---|---|
| `Image.expression(expression, map)` | Image |

| Argument | Type | Details |
|---|---|---|
| this: `image` | Image | The Image instance. |
| `expression` | String | The expression to evaluate. |
| `map` | *Dictionary, optional* | *A map of input images available by name.* |

Source: https://developers.google.com/earth-engine/apidocs/ee-image-expression

```
"NDVI": {
        "application_domain": "vegetation",
        "bands": [
            "N",
            "R"
        ],
        "contributor": "https://github.com/davemlz",
        "date_of_addition": "2021-04-07",
        "formula": "(N - R)/(N + R)",
        "long_name": "Normalized Difference Vegetation Index",
        "platforms": [
            "Sentinel-2",
            "Landsat-OLI",
            "Landsat-TM",
            "Landsat-ETM+",
            "MODIS",
            "Planet-Fusion"
        ],
        "reference": "https://ntrs.nasa.gov/citations/19740022614",
        "short_name": "NDVI"
    }
```

```
"NDVI":

            "bands": [
                "N",
                "R"
            ]


            "formula": "(N - R)/(N + R)"
```


"NDVI":
"bands": [
"N",
"R"
]
"formula": "(N - R)/(N + R)"
NDVI = my_image.expression(
“(N - R)/(N + R)”,
{
“N”: my_image.select(“NIR”),
“R”: my_image.select(“RED”),
})
Google

```
"SAVI": {
            "application_domain": "vegetation",
            "bands": [
                "L",
                "N",
                "R"
            ],
            "contributor": "https://github.com/davemlz",
            "date_of_addition": "2021-04-07",
            "formula": "(1.0 + L) * (N - R) / (N + R + L)",
            "long_name": "Soil-Adjusted Vegetation Index",
            "platforms": [
                "Sentinel-2",
                "Landsat-OLI",
                "Landsat-TM",
                "Landsat-ETM+",
                "MODIS",
                "Planet-Fusion"
            ],
            "reference": "https://doi.org/10.1016/0034-4257(88)90106-X",
            "short_name": "SAVI"
        }
```

```
"SAVI":

            "bands": [
                "L",
                "N",
                "R"
            ]


            "formula": "(1.0 + L) * (N - R) / (N + R + L)"
```


"SAVI":
"bands": [
"L",
"N",
"R"
]
"formula": "(1.0 + L) * (N - R) / (N + R + L)"
Google
SAVI = my_image.expression(
“(1.0 + L) * (N - R) / (N + R + L)”,
{
“N”: my_image.select(“NIR”),
“R”: my_image.select(“RED”),
“L”: 0.5,
})

“Can we make this even **easier**?”


Source: https://github.com/awesome-spectral-indices/spectral

```
var spectral = require(“users/dmlmont/spectral:spectral”)
```

```
print(spectral.indices)
```

```
▸ NDPolI: Object (9 properties)
▸ NDREI: Object (9 properties)
▸ NDSI: Object (9 properties)
▸ NDSII: Object (9 properties)
▸ NDSInw: Object (9 properties)
▸ NDSaII: Object (9 properties)
▸ NDSoI: Object (9 properties)
▾ NDVI: Object (9 properties)
  ▾ bands: ["N","R"]
      0: N
      1: R
    application_domain: vegetation
    contributor: https://github.com/davemlz
    date_of_addition: 2021-04-07
    formula: (N - R)/(N + R)
    long_name: Normalized Difference Vegetation Index
  ▾ platforms: List (6 elements)
      0: Sentinel-2
      1: Landsat-OLI
      2: Landsat-TM
      3: Landsat-ETM+
      4: MODIS
      5: Planet-Fusion
    reference: https://ntrs.nasa.gov/citations/19740022614
    short_name: NDVI
▸ NDVI705: Object (9 properties)
▸ NDVIMNDWI: Object (9 properties)
▸ NDVIT: Object (9 properties)
▸ NDWI: Object (9 properties)
▸ NDWIns: Object (9 properties)
▸ NDYI: Object (9 properties)
```

```
print(spectral.indices.NDVI)
```

```
▼ Object (9 properties)
  ▶ bands: ["N","R"]
    application_domain: vegetation
    contributor: https://github.com/davemlz
    date_of_addition: 2021-04-07
    formula: (N - R)/(N + R)
    long_name: Normalized Difference Vegetation Index
  ▶ platforms: List (6 elements)
    reference: https://ntrs.nasa.gov/citations/19740022614
    short_name: NDVI
```

```
print(spectral.indices.NDVI.formula)

print(spectral.indices.NDVI.reference)
```

```
(N - R)/(N + R)
```

https://ntrs.nasa.gov/citations/19740022614

```
spectral.computeIndex(img, index, params)
```

Image to use for an spectral index computation

```
spectral.computeIndex(img, index, params)
```

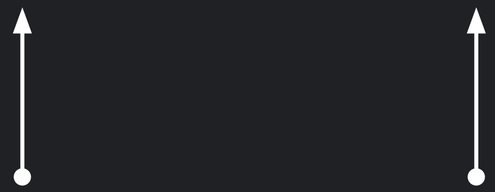
Image to use for an spectral index computation

```
spectral.computeIndex(img, index, params)
```

Short name of the index to compute

Image to use for an spectral index computation

Dictionary of parameters



```
spectral.computeIndex(img, index, params)
```

Short name of the index to compute

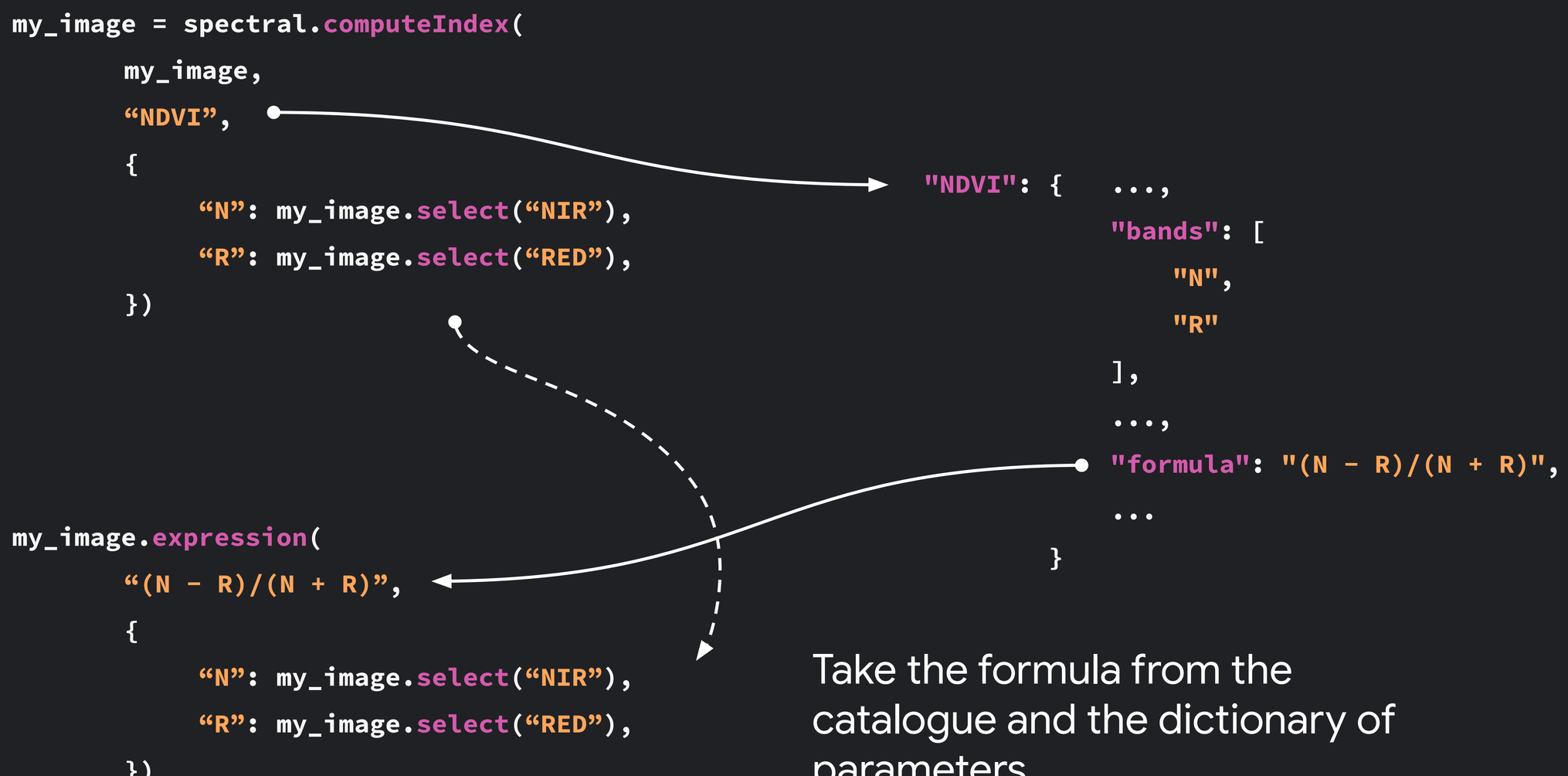
```
my_image = spectral.computeIndex(
        my_image,
        “NDVI”,
        {
                “N”: my_image.select(“NIR”),
                “R”: my_image.select(“RED”),
        })
```

```
my_image = spectral.computeIndex(
        my_image,
        “NDVI”,
        {
             “N”: my_image.select(“NIR”),
             “R”: my_image.select(“RED”),
        })
```

Use the short name of the index to look inside the catalogue keys

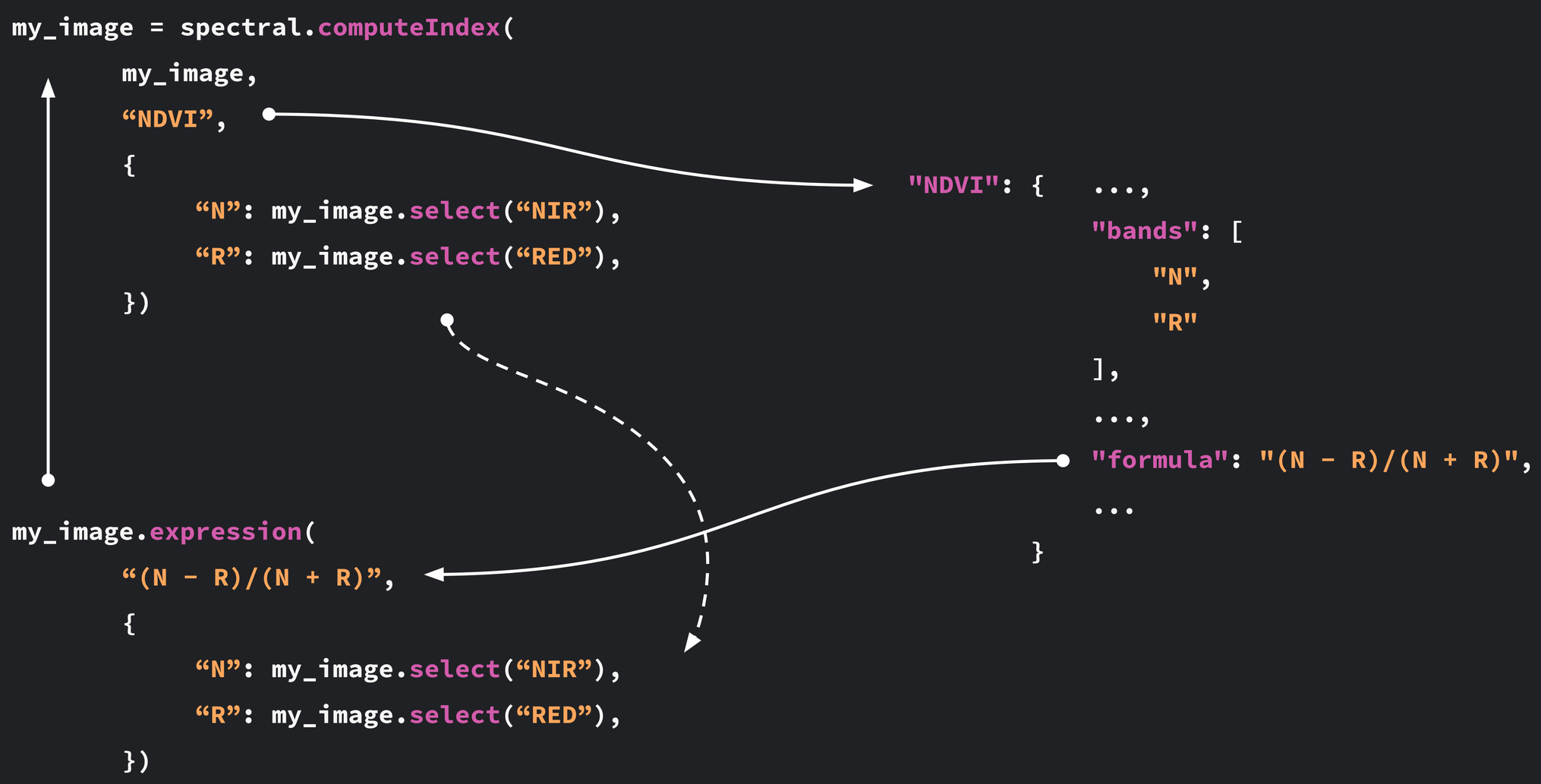
```
my_image = spectral.computeIndex(
        my_image,
        “NDVI”,
        {
             “N”: my_image.select(“NIR”),
             “R”: my_image.select(“RED”),
        })
```

```
"NDVI": {   ...,
            "bands": [
                "N",
                "R"
            ],
            ...,
            "formula": "(N - R)/(N + R)",
            ...
        }
```




Take the formula from the catalogue and the dictionary of parameters

Add the index as a new band

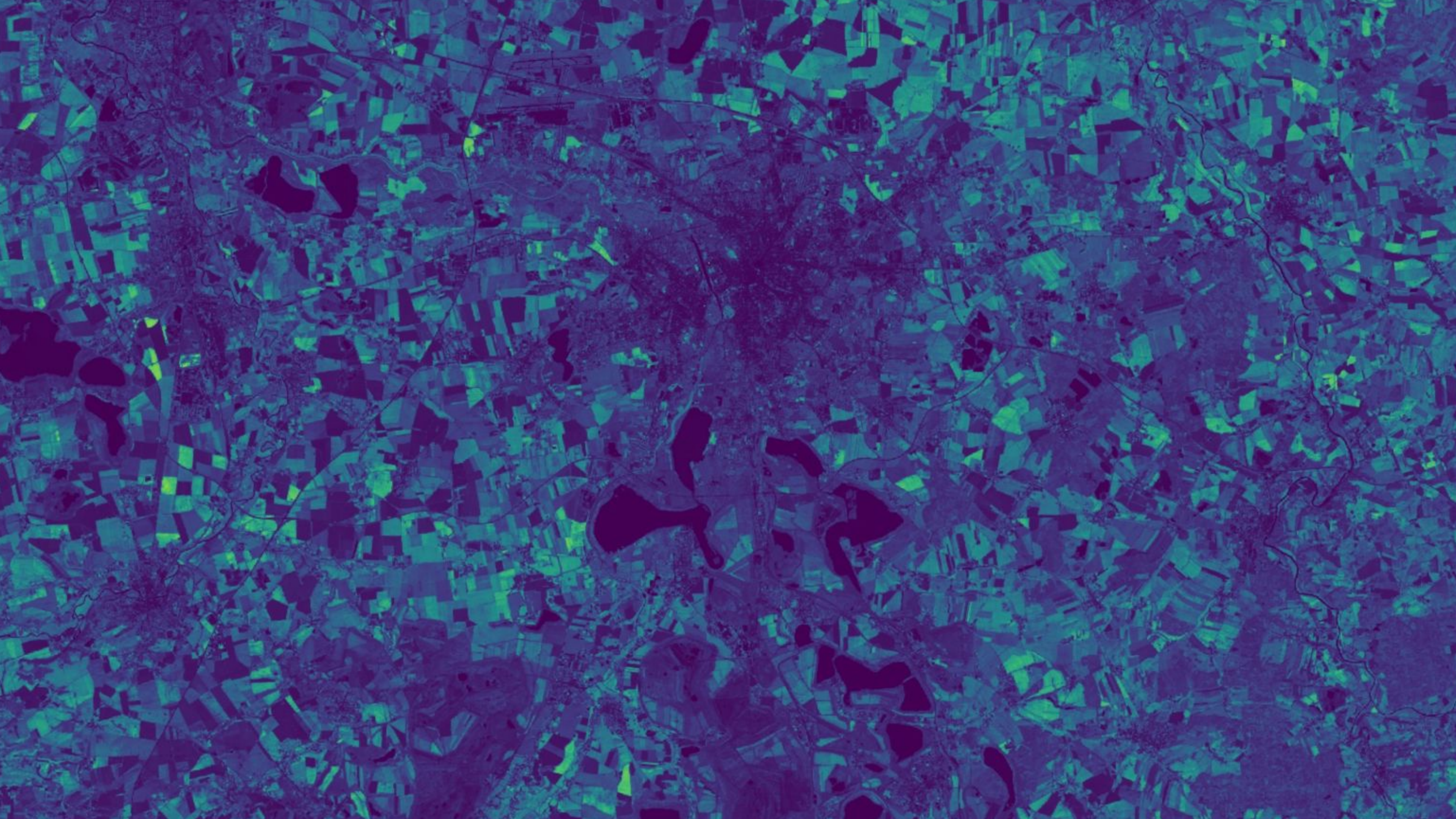


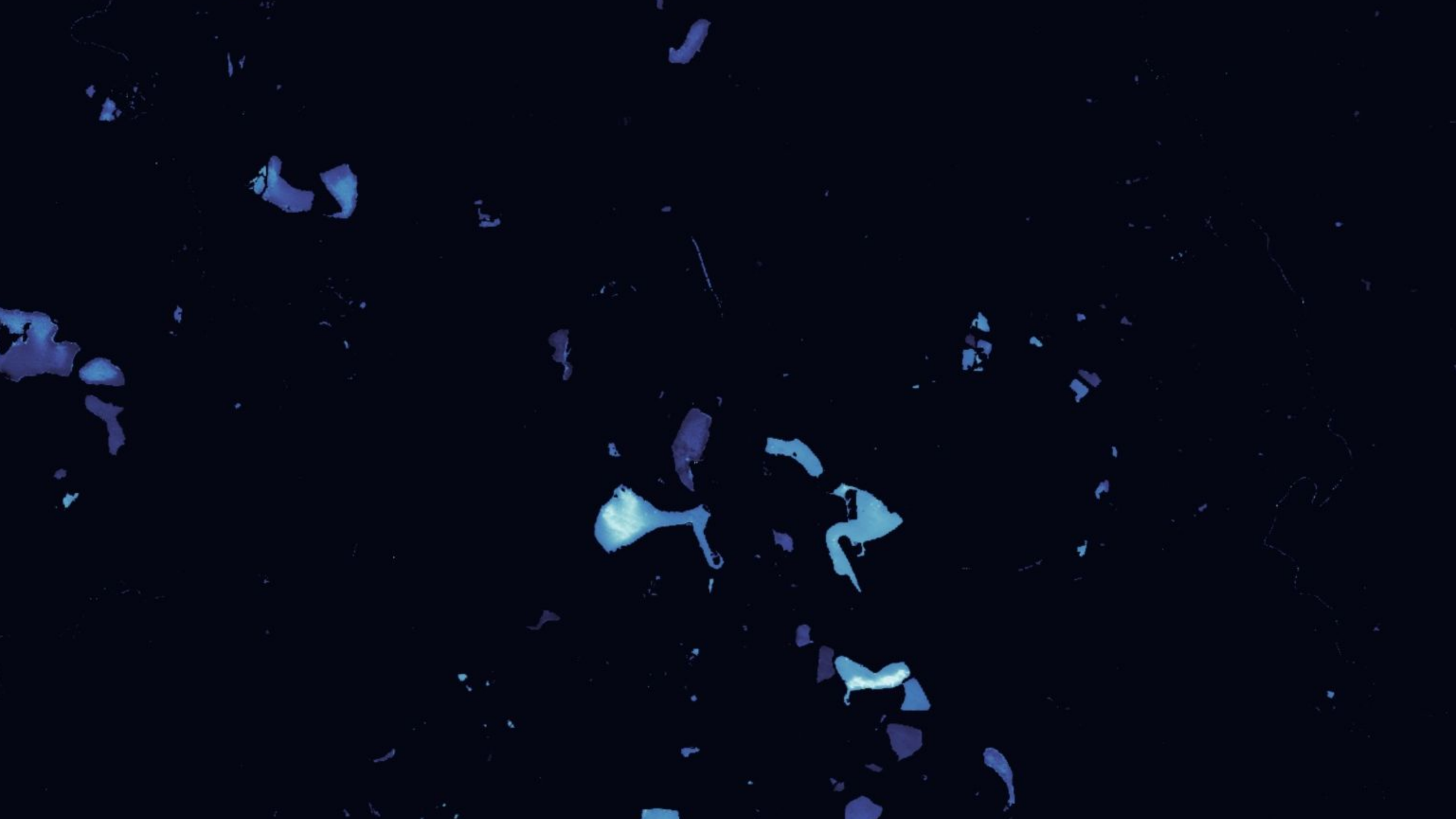
```
my_image = spectral.computeIndex(
        my_image,
        [“DVI”, “NIRv”, “NDVI”, “SAVI”],
        {
                “N”: my_image.select(“NIR”),
                “R”: my_image.select(“RED”),
                “L”: 0.5,
        })
```

```
my_image.bandNames()
# [“B1”, “B2”, ..., “DVI”, “NIRv”, “NDVI”, “SAVI”]
```

```
var spectral = require("users/dmlmont/spectral:spectral");

var geom = ee.Geometry.Point([-76.3486, 3.6448]);

var S2 = ee.ImageCollection("COPERNICUS/S2_SR")
  .filterBounds(geom)
  .filter(ee.Filter.lt("CLOUDY_PIXEL_PERCENTAGE",20))
  .sort("CLOUDY_PIXEL_PERCENTAGE",true)
  .first();

S2 = spectral.scale(S2,"COPERNICUS/S2_SR");

S2 = spectral.computeIndex(S2,"IRECI",{
  R: S2.select("B4"),
  RE1: S2.select("B5"),
  RE2: S2.select("B6"),
  RE3: S2.select("B7"),
});
```

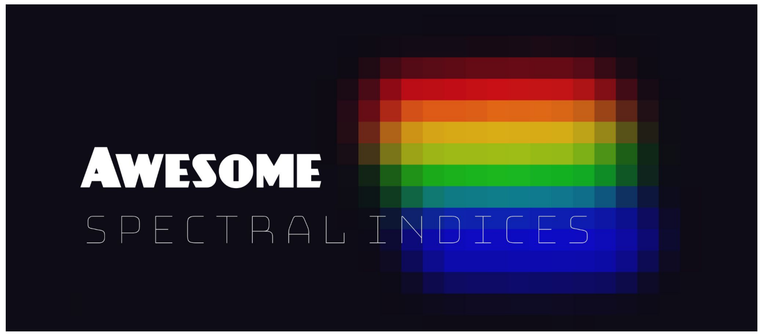
```
var spectral = require("users/dmlmont/spectral:spectral");

var geom = ee.Geometry.Point([12.6832, 51.6736]);
var dataset = "LANDSAT/LC09/C02/T1_L2"

var L9 = ee.ImageCollection(dataset)
  .filterBounds(geom)
  .filter(ee.Filter.lt("CLOUD_COVER",50))
  .sort("CLOUD_COVER",true)
  .first();

L9 = spectral.scale(L9,dataset);
L9 = spectral.offset(L9,dataset);

L9 = spectral.computeIndex(L9,["NIRv", "NDWI"],{
  R: L9.select("SR_B4"),
  N: L9.select("SR_B5"),
  G: L9.select("SR_B3"),
});
```





# Thank you!

# Contributions are welcome!




**Catalogue**
Awesome Spectral Indices


**GEE JS Module**
spectral


**Python Package**
spyndex


**Streamlit App**
Espectro