Intro2R-spatial

Analysing spatial patterns of the landscape

Song, Xiao Ping
xp.song@u.nus.edu

Course materials: https://github.com/xp-song/Intro2R-spatial
updated 2020-10-09

Intro2R-spatial

1. Navigate to course webpage for instructions and background information
   https://github.com/xp-song/Intro2R-spatial

2. Download workshop materials
   (green button on webpage)

Outline

Spatial pattern of a landscape

A function of the ***composition*** and *configuration* of the physical landscape

Photo: Philipp Urech (2017)

Spatial pattern of a landscape

Spatial composition

- Variety - What types/classes?
- Abundance - How much?

Photo: Philipp Urech (2017)

Spatial pattern of a landscape

A function of the *composition* and ***configuration*** of the physical landscape

Photo: Philipp Urech (2017)

Spatial pattern of a landscape

Spatial configuration

- Position
- Orientation
- Shape/arrangement

Photo: Philipp Urech (2017)

Why is spatial pattern important?

The study of landscape ecology

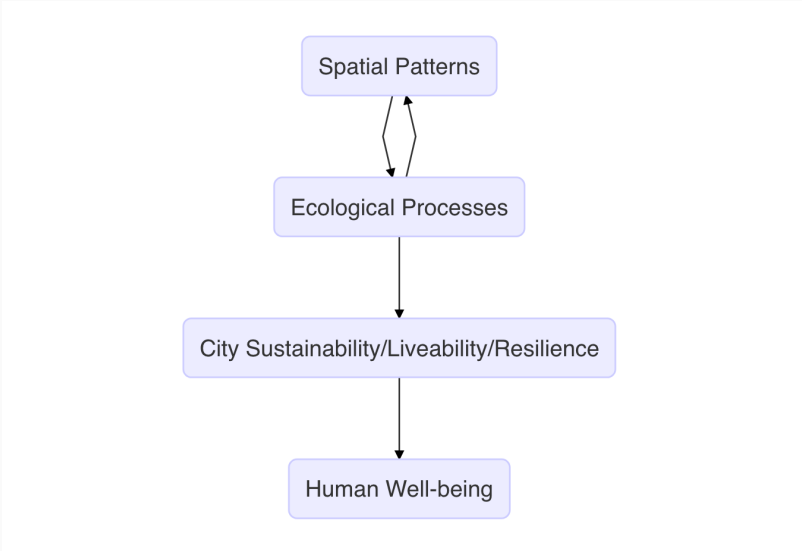


Figure adapted from Wu et al. (2015)

Why is spatial pattern important?

The study of landscape ecology

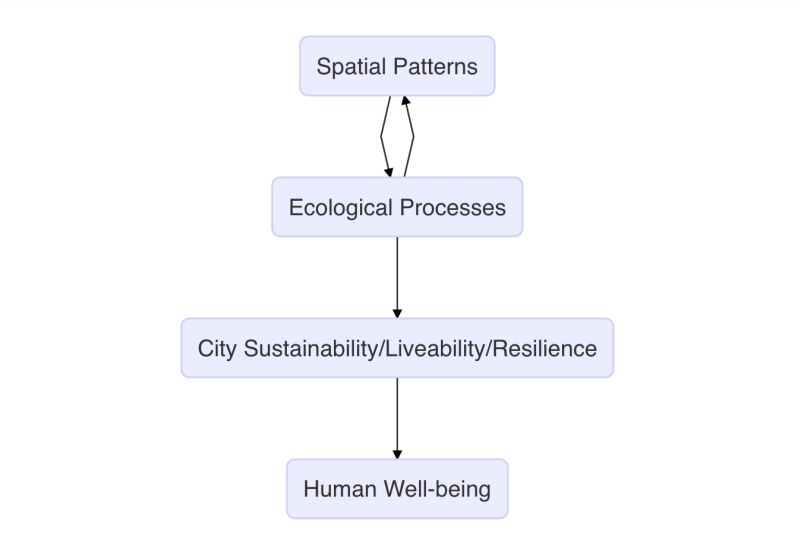


Figure adapted from Wu et al. (2015)

Pattern-process relationships

- How do we optimise spatial *patterns* to achieve desired ecological *processes*?
- How can we incorporate such thinking into policy & design?

Outline

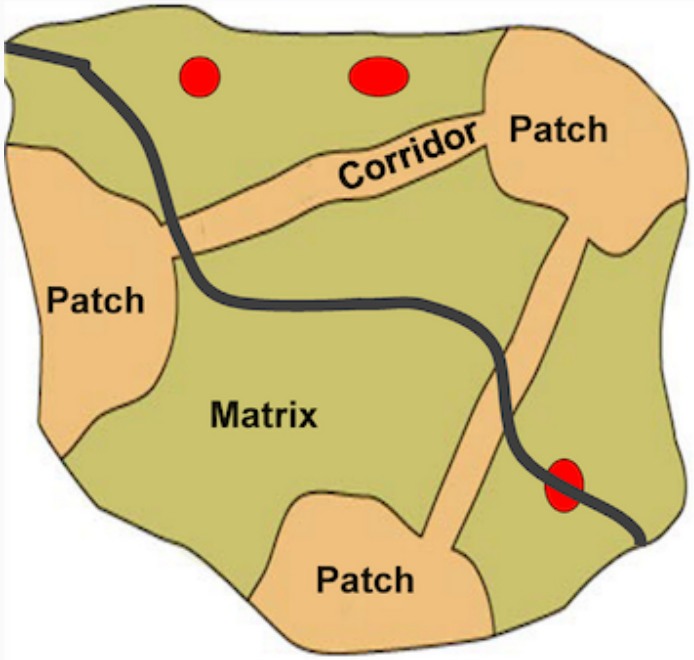
Landscape ecology: Conceptual models

Mosaic model vs. Gradient model

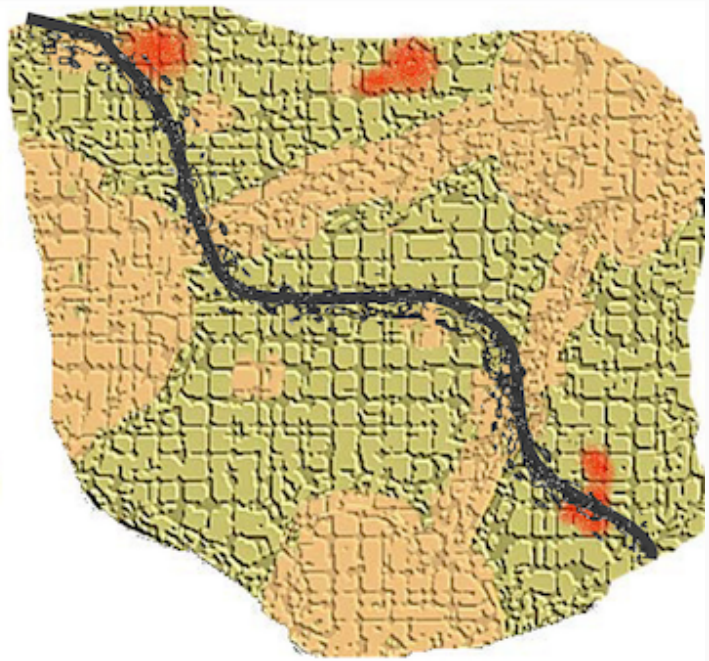
Photo: Philipp Urech (2017)

Landscape ecology: Conceptual models

Figure from Lausch et al. (2015)



PCM (mosaic) model



Gradient model

Landscape ecology: Conceptual models

Do you see the patches & corridors in this landscape?

Photo: Philipp Urech (2017)

Outline

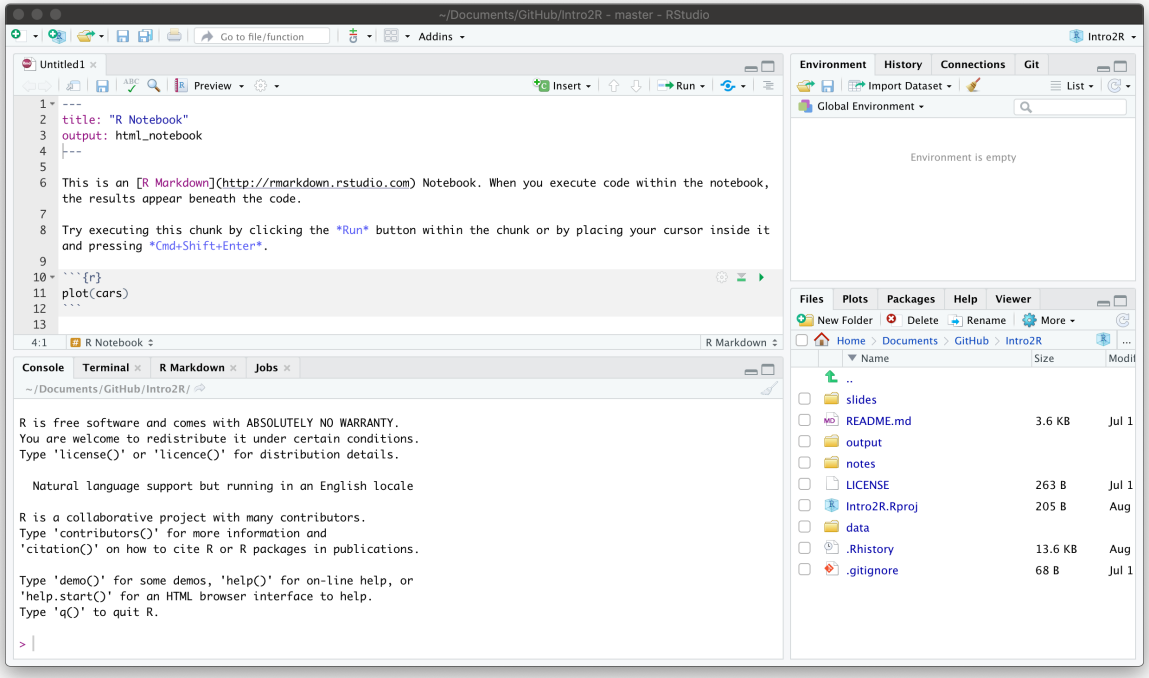
Course materials

On your computer, navigate to folder downloaded from
https://github.com/xp-song/Intro2R-spatial

notes.Rmd
notes.html [1]
/slides [1]
/data
/clean_data
Intro2R-spatial.Rproj

[1] View in your web browser by opening the '*.html*' files

R Studio Client



- **Console:** Command line input/output
- **Script editor:** View/edit files that contain code
- **Environment/History**
- **Files/Plots/Packages/Help/Viewer**

RStudio Projects

- RStudio Projects help organise your work into separate 'R sessions'.
- Each project has it's own workspace a.k.a. 'working directory' (separate configuration, history, etc.)
- The location of the '*.RProj*' file defines the 'working directory'

🌟 Best Practice

- Use *relative* paths in your script, based on *.RProj* file location
- Keep all project items in the working directory

Set up

Use a new R notebook for our analysis [1]

Open the R notebook *notes.Rmd* file for reference

[1] Ensure that it is in the working directory (i.e. same folder as the *Intro2R-spatial.Rproj* file)

Set up

Install packages

- tidyverse: A collection of packages commonly used for data analyses
- raster: Analyse raster data
- sf: Analyse spatial data as "simple features" (i.e. tabular 2D data)
- landscapetools: Collection of functions for landscape analysis
- landscapemetrics: Analyse spatial patterns of discrete landscape classes

```
install.packages("tidyverse", dependencies = TRUE)
install.packages("raster", dependencies = TRUE)
install.packages("sf", dependencies = TRUE)
install.packages("landscapetools", dependencies = TRUE)
install.packages("landscapemetrics", dependencies = TRUE)
```

Note:

- Enter `n` if you get the following prompt: `Do you want to install from sources the package which needs compilation?`
- Click 'Yes' if you are asked to restart R

Outline

Land cover classification

Images

Possible data sources:

- Remotely sensed data (i.e. satellites, drones, planes)
- Historical maps
- Field surveys

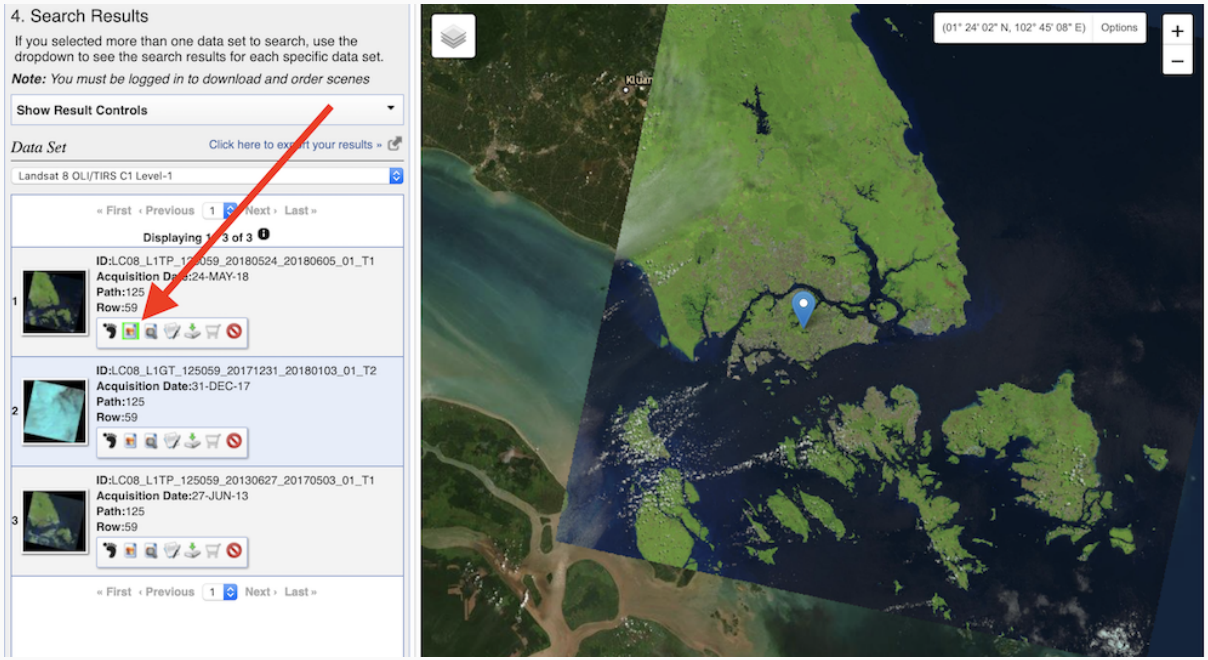
We will be using freely available Landsat satellite images from the public database https://earthexplorer.usgs.gov/

- Sign up for an account
- Specify search criteria (i.e. location & date range)
- Specify dataset (Landsat > Landsat Collection 1 Level-1 > Landsat 8 OLI/TIRS C1 Level-1)
- Specify additional criteria (e.g. < 10% cloud cover; Day)
- Order scene and checkout
- Submit Order to download scenes when status is "Processing Complete"
- Install Bulk Download Application (BDA) on your computer
- Login & download ordered scenes

Land cover classification

Images

Previewing images on USGS Earth Explorer



Images

Remotely-sensed data: A brief introduction

- Satellites have multiple sensors that measure surface *reflectance*

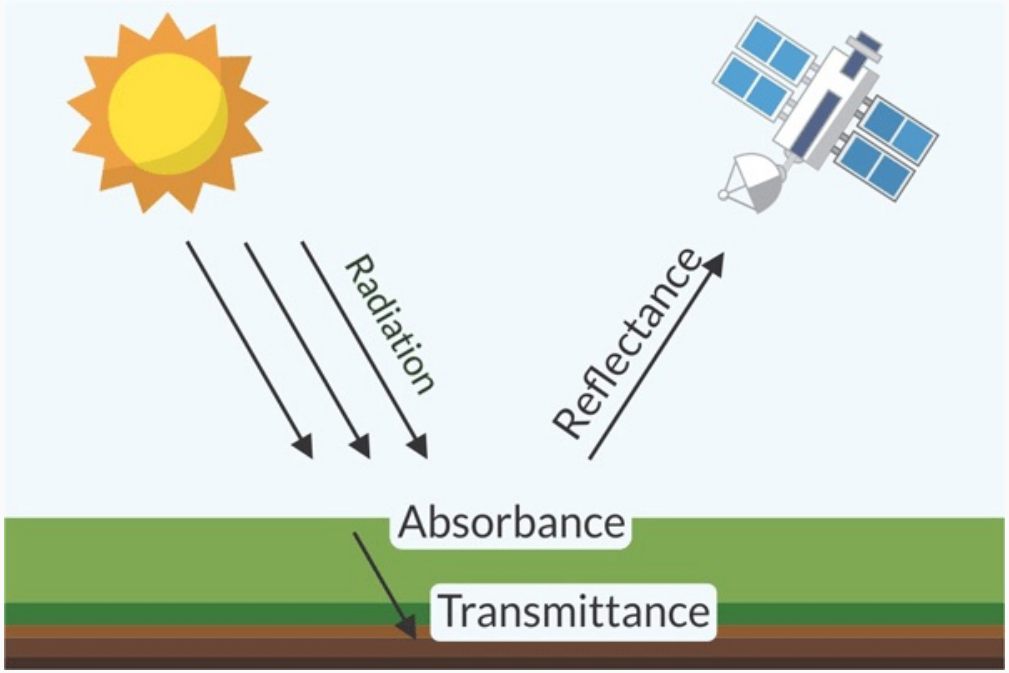


Figure by Luciana Nieto and Ignacio Ciampitti, K-State Research and Extension

Land cover classification

Images

Remotely-sensed data: A brief introduction

- Satellites have multiple sensors that measure surface *reflectance*
- Specific bands of wavelengths measured along the electromagnetic spectrum (spatial resolution varies)

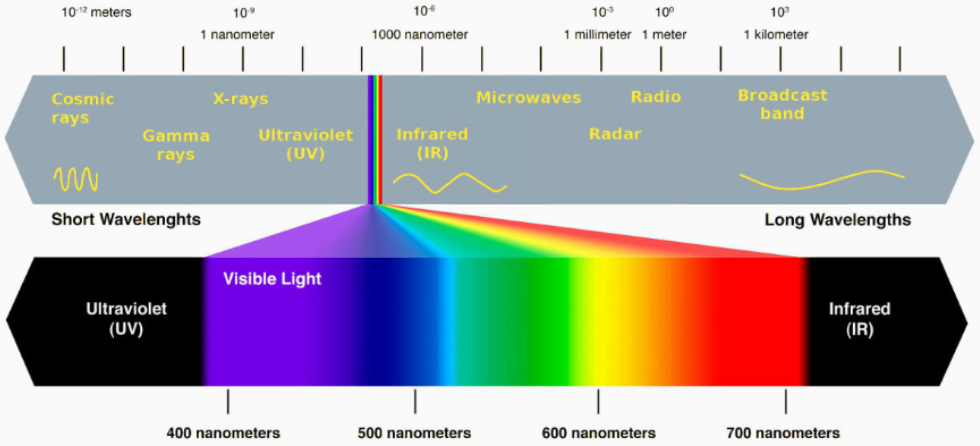


Figure from cleanpng.com

Land cover classification

Images

Remotely-sensed data: A brief introduction

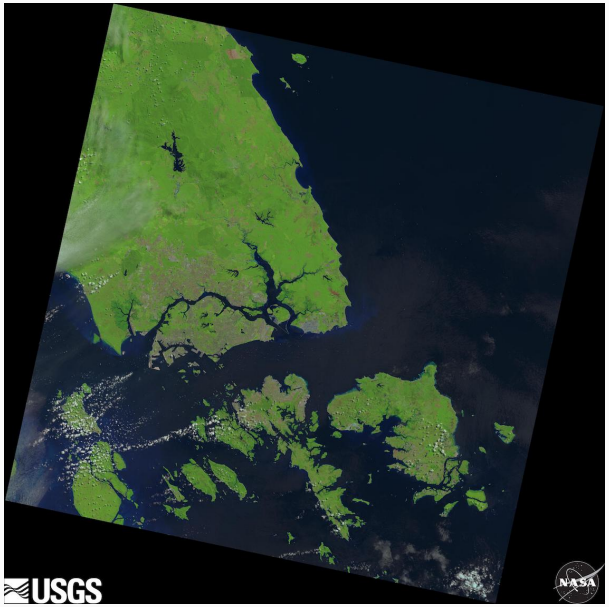
- Satellites have multiple sensors that measure surface *reflectance*
- Specific bands of wavelengths measured along the electromagnetic spectrum (spatial resolution varies)
- Data stored as multiple layers of raster images, a.k.a. "scenes" or "bands"

Land cover classification

Images

We're using the Landsat-8 scene of Singapore on 2018/05/24
(30m resolution for RGB, NIR & SWIR)

- Navigate to the *data* folder



Landsat-8 natural color image. Source: U.S. Geological Survey.

Land cover classification

Images

We're using the Landsat-8 scene of Singapore on 2018/05/24
(30m resolution for RGB, NIR & SWIR)

- Navigate to *data/LC08_L1TP_125059_20180524_20180605_01_T1* (named according to product identifier)
- Unzip each file into individual bands (*.TIF* rasters)
- Unzipped folder will contain 5 multispectral bands of the same region

Land cover classification

Images

Bands:

- **B1:** Ultra Blue
- **B2:** Blue
- **B3:** Green
- **B4:** Red
- **B5:** Near Infrared (NIR)
- **B6:** Shortwave Infrared (SWIR) 1
- **B7:** SWIR 2
- **B8:** Panchromatic
- **B9:** Cirrus
- **B10:** Thermal Infrared (TIRS) 1
- **B11:** TIRS 2
- **BQA:** Band quality assessment (based on cloud cover)

Land cover classification

Images

Import

Let's import the visible light bands into R!

```
library(raster)

#blue
b2 <- raster('data/Landsat 8 OLI_TIRS C1 Level-1/LC08_L1TP_125059_20180524_20

#green
b3 <- raster('data/Landsat 8 OLI_TIRS C1 Level-1/LC08_L1TP_125059_20180524_20

#red
b4 <- raster('data/Landsat 8 OLI_TIRS C1 Level-1/LC08_L1TP_125059_20180524_20
```

Land cover classification

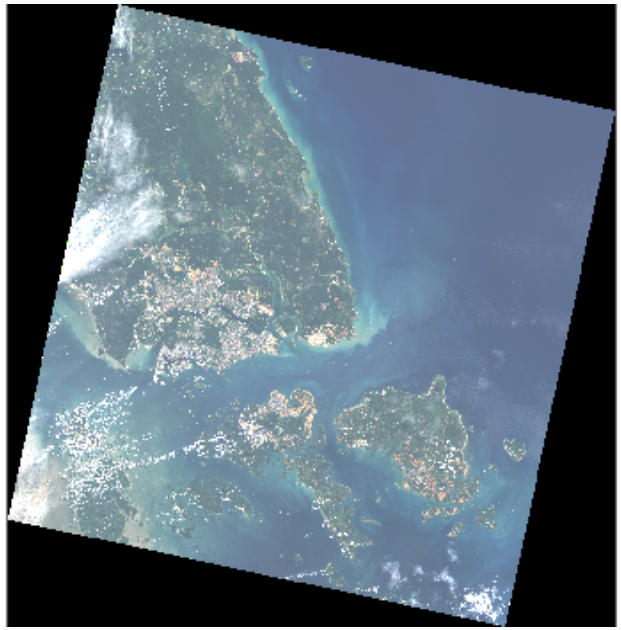
Images

Import

Combine into RGB RasterStack & plot

```
landsatRGB <- stack(b4, b3, b2) #order is impt

plotRGB(landsatRGB,
        stretch = "lin")
```



Landsat-8 true color composite (RGB). Source: U.S. Geological Survey.

Land cover classification

Images

Import

Import bands 1-5 as a RasterStack named `landsat`

```
filenames <- paste0('data/Landsat 8 OLI_TIRS C1 Level-1/LC08_L1TP_125059_2018
landsat <- stack(filenames)
```

Rename the individual bands

```
names(landsat) <- c('ultra-blue', 'blue', 'green', 'red', 'NIR')
```

Check coordinate reference system: WGS 1984 UTM Zone 48N

```
crs(landsat)
```

```
## CRS arguments:
##  +proj=utm +zone=48 +datum=WGS84 +units=m +no_defs
```

Land cover classification

Images

Import

Crop

Use city boundaries to crop out the landscape of interest

Polygon file in *data* folder, downloaded from data.gov.sg

Land cover classification

Images

Import

Crop

Import polygon of Singapore boundaries as `sgshp`

```
library(sf)
sgshp <- st_read("data/master-plan-2014-region-boundary-web-shp/MP14_REGION_W
```

Check CRS of `sgshp`

```
crs(sgshp)
```

```
## CRS arguments:
##  +proj=tmerc +lat_0=1.36666666666667 +lon_0=103.833333333333 +k=1
## +x_0=28001.642 +y_0=38744.572 +datum=WGS84 +units=m +no_defs
```

Transform CRS of `sgshp` to match `landsat`

```
sgshp <- st_transform(sgshp, crs = crs(landsat))
```

Land cover classification

Images

Import

Crop

Crop RasterStack `landsat` to the extent of `sgshp` [1]

```
landsat <- crop(landsat, sgshp) # crop to rectangle
```

Mask values in `landsat` based on shape of `sgshp`

```
landsat <- mask(landsat, sgshp)
```

[1] The polygon should be in 2D format. Use `st_zm()` to drop the Z-dimension if it is a 3D object.

Land cover classification

Images

Import

Crop

Plot the RGB bands of `landsat`

```
landsatRGB <- subset(landsat, c(4,3,2))
plotRGB(landsatRGB,
        stretch = "lin")
```

Landsat-8 true color composite (USGS, 2018) cropped to city boundaries (URA, 2014)

Land cover classification

Images

Import

Crop

Classify

In this workshop, we use a simplistic approach to classify land cover

- Use vegetation indices such as the Normalized Difference Vegetation Index (NDVI)
- Ranges from -1 to 1 (higher = greener)

- Vegetation reflects NIR & absorbs red light, captured in the formula:

$$NDVI = \frac{NIR - Red}{NIR + Red}$$

Land cover classification

Images

Import

Crop

Classify

Make a function to calculate the NDVI from RasterStack `landsat`

Note: Calculations are made per pixel

$$NDVI = \frac{NIR - Red}{NIR + Red}$$

```
ndvi <- function(x, y) {
    (x - y) / (x + y)
}
```

Land cover classification

Images

Import

Crop

Classify

Apply function to the NIR and Red bands of `landsat`

```
landsatNDVI <- overlay(landsat[[5]], landsat[[4]], fun = ndvi)
```

Limit range of values to be from -1 to 1

```
landsatNDVI <- reclassify(landsatNDVI, c(-Inf, -1, -1)) # <-1 becomes -1
landsatNDVI <- reclassify(landsatNDVI, c(1, Inf, 1)) # >1 becomes 1
```

Land cover classification

Images

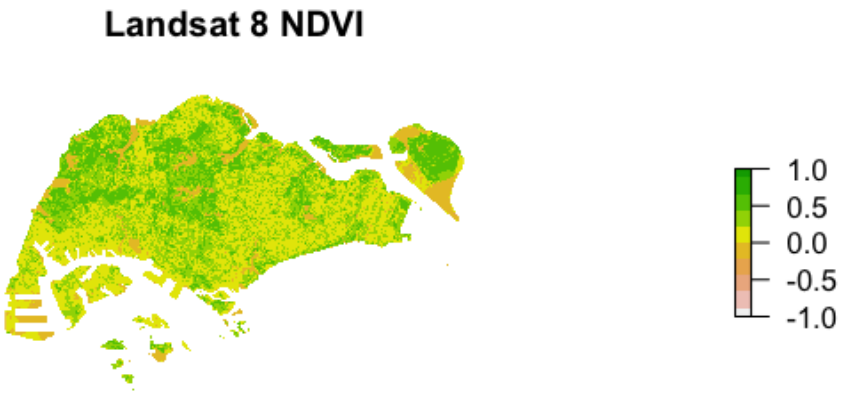
Import

Crop

Classify

Plot the NDVI

```
plot(landsatNDVI,
     col = rev(terrain.colors(10)),
     main = "Landsat 8 NDVI",
     axes = FALSE, box = FALSE)
```



Land cover classification

Images

Import

Crop

Classify

NDVI ranges for land cover classes derived from Landsat-8[1]

Class	NDVI
Water	−0.28–0.015
Built-up	0.015–0.14
Barren Land	0.14–0.18
Shrub and Grassland	0.18–0.27
Sparse Vegetation	0.27–0.36
Dense Vegetation	0.36–0.74

[1] Akbar et al. (2019). Remote Sensing, 11(2), 105.

Land cover classification

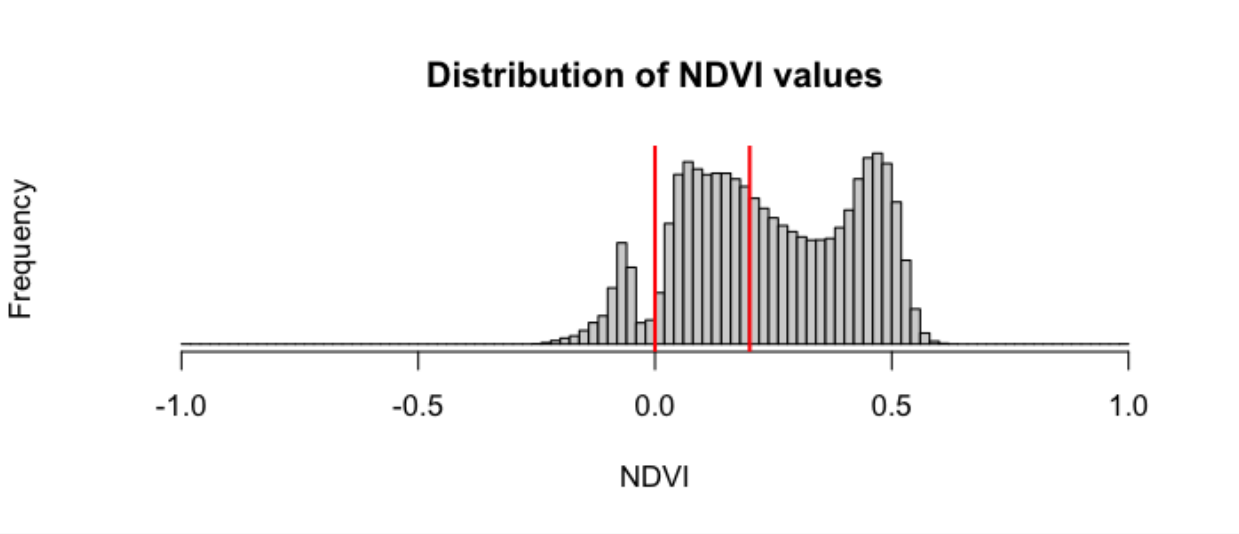
Images

Import

Crop

Classify

```
hist(landsatNDVI,
     main = "Distribution of NDVI values", xlab = "NDVI",
     xlim = c(-1, 1), breaks = 100, yaxt = 'n')
abline(v=0.2, col="red", lwd=2)
abline(v=0, col="red", lwd=2)
```



Let's define vegetation cover by threshold value 0.2

Land cover classification

Images

Import

Crop

Classify

Mask all pixels with NDVI < 0.2

```
landsatGreen <- reclassify(landsatNDVI, c(-1, 0.2, NA)) #-1 to 0.2 becomes NA
plot(landsatGreen, col = "darkgreen",
     axes = FALSE, box = FALSE, legend = FALSE)
```

Looks like vegetation cover

Land cover classification

Images

Import

Crop

Classify

Mask all pixels with NDVI > 0

```
landsatBlue <- reclassify(landsatNDVI, c(0, 1, NA)) # 0 to 1 becomes NA
plot(landsatBlue, col = "blue",
     axes = FALSE, box = FALSE, legend = FALSE,)
```

Looks like water cover

Land cover classification

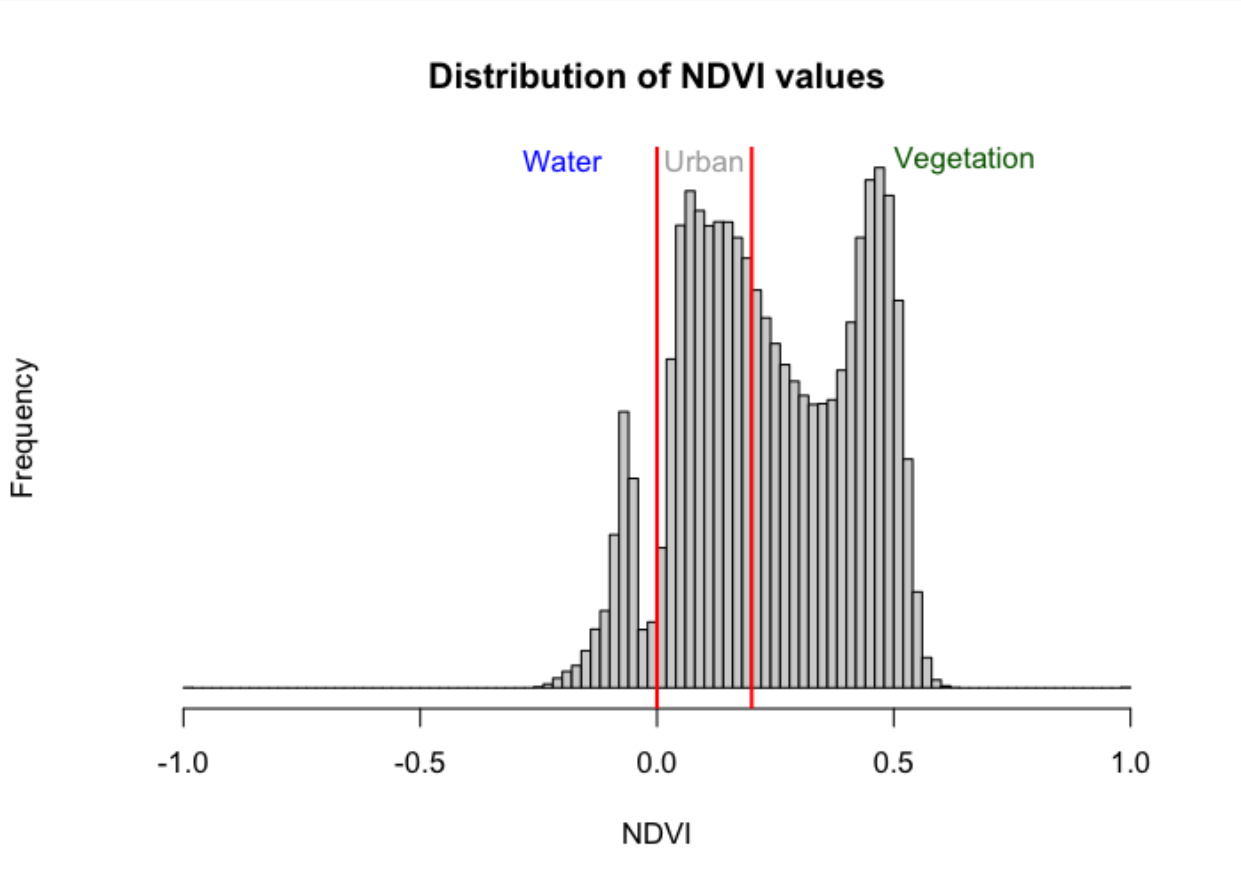
Images

Import

Crop

Classify

Let's create a new raster with pixels assigned to one of 3 classes



Land cover classification

Images

Import

Crop

Classify

Create a matrix to be used as an argument in the `reclassify()` function:

```
reclass_m <- matrix(c(-Inf, 0, 1, #water
                      0, 0.2, 2, #urban
                      0.2, Inf, 3), #veg
                    ncol = 3, byrow = TRUE)
reclass_m
```

```
##      [,1] [,2] [,3]
## [1,] -Inf  0.0    1
## [2,]  0.0  0.2    2
## [3,]  0.2  Inf    3
```

Create a new raster `landsatCover` using the threshold values defined in `reclass_m`

```
landsatCover <- reclassify(landsatNDVI, rcl = reclass_m)
```

Land cover classification

Images

Import

Crop

Classify

Plot `landsatCover` with custom colours

```
plot(landsatCover,
     col = c("blue", "grey", "darkgreen"),
     legend = FALSE,
     axes = FALSE,
     box = FALSE,
     main = "Land cover (mosaic) in Singapore")
legend("bottomright",
       legend = c("Water", "Urban", "Vegetation"),
       fill = c("blue", "grey", "darkgreen"),
       border = FALSE,
       bty = "n")
```

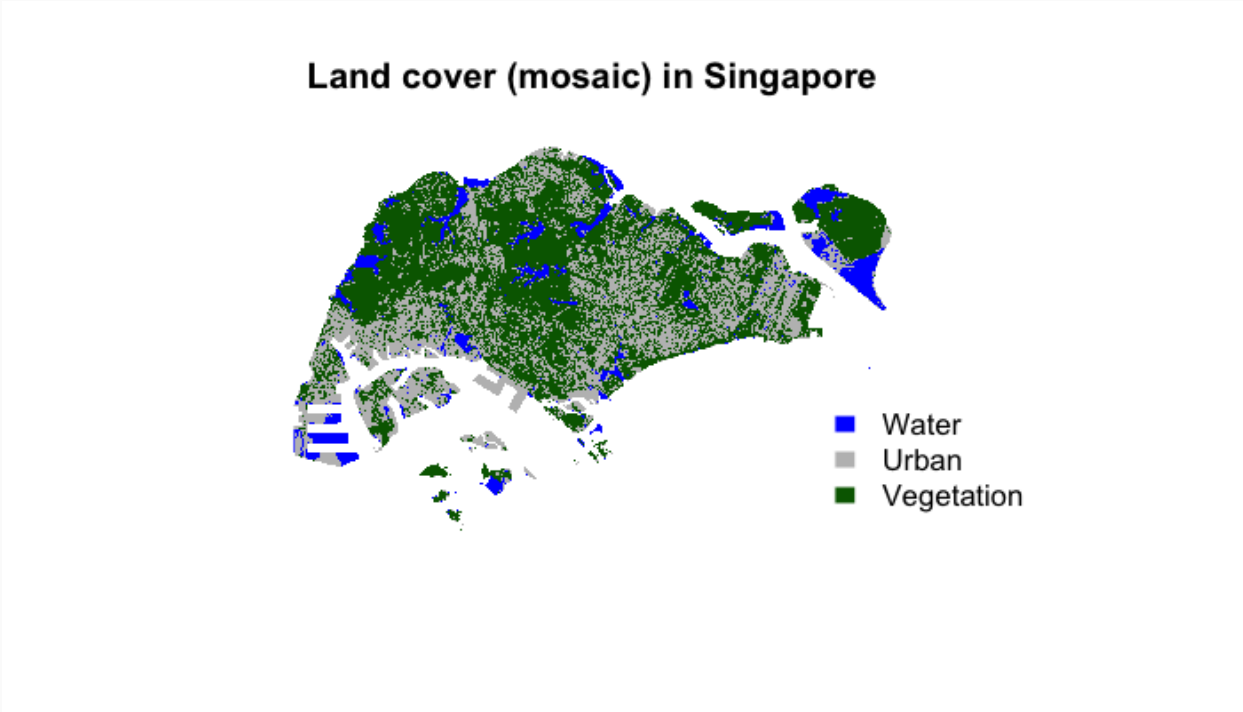
Land cover classification

Images

Import

Crop

Classify



Land cover classification

Images

Import

Crop

Classify

Save

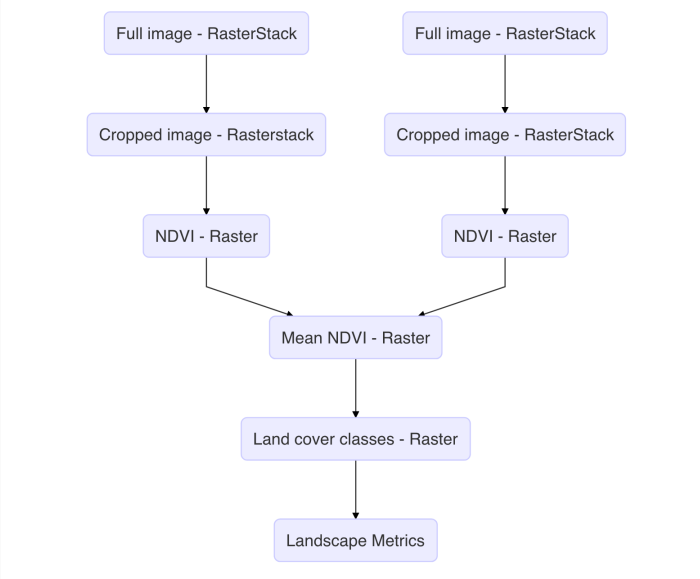
Save raster `landsatCover` to disk

```
writeRaster(landsatCover,
            filename = "clean_data/landsat_landcover.tif",
            overwrite = TRUE)

#save as GeoTiff format
```

Land cover classification

💭 What if the satellite image does not cover the whole city?



1) Calculate the NDVI for multiple images (each a *single* raster layer)

2) Combine the rasters by using the mean NDVI between images

3) Use combined image to derive land cover classes

Land cover classification

💭 What if the satellite image does not cover the whole city?

If CRS of images are different, re-project one to match the other

```
NDVIraster2 <- projectRaster(NDVIraster2, NDVIraster1) #overwrite NDVIraster2
```

Calculate the mean NDVI for overlapping sections between images

```
NDVIcombined <- mosaic(NDVIraster1, NDVIraster2, fun = mean) #mean() function applied
```

Outline

Landscape metrics

- Quantifies spatial patterns in a patch-corridor-matrix (mosaic) model
- We will be using the package `landscapemetrics`, which is based on the software FRAGSTATS

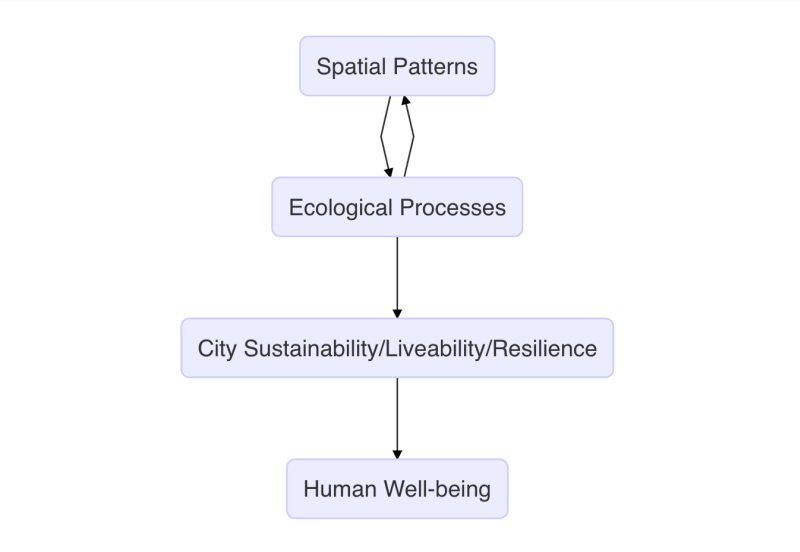


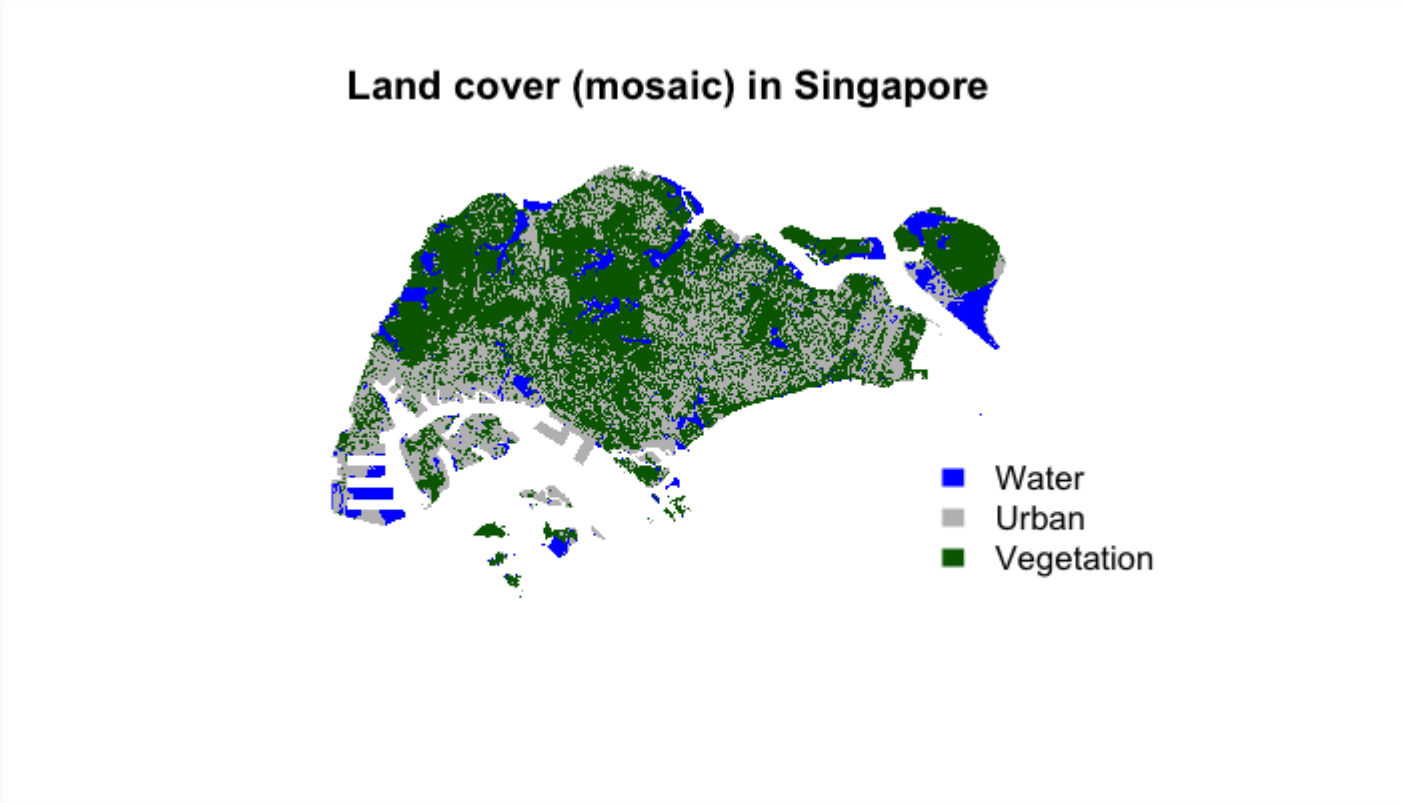
Figure adapted from Wu et al. (2015)

Key question: How does the metric link to an explicit ecological process?

Landscape metrics

3 levels of analyses:

- Patch `lsm_p_<metric>()`
- Class `lsm_c_<metric>()`
- Landscape `lsm_l_<metric>()`



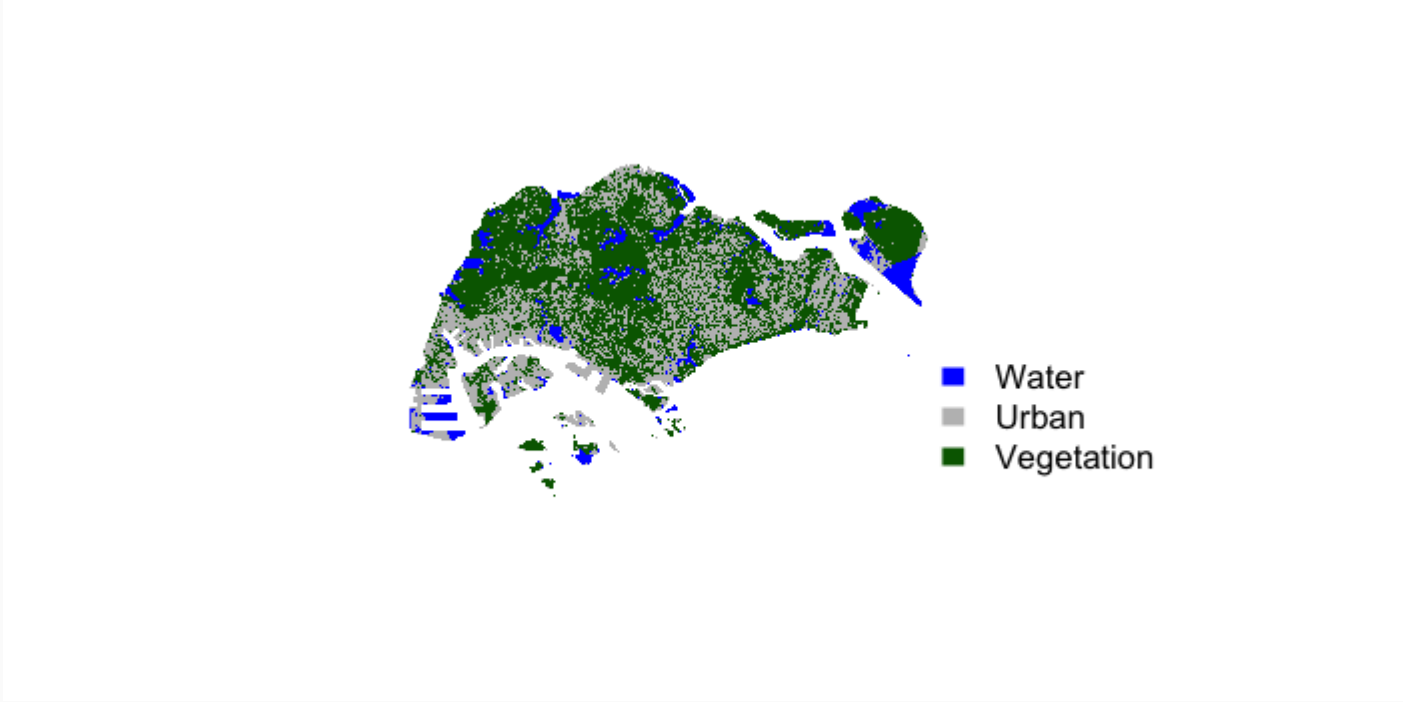
Landscape metrics

Subject Areas

Composition	**Configuration**
Proportional abundance	Patch area & edge
Richness	Patch shape complexity
Evenness	Core area
Diversity	Isolation/proximity
	Contrast
	Dispersion
	Contagion & interspersion
	Subdivision
	Connectivity

More info available in documentation by UMass Landscape Ecology Lab

Landscape metrics

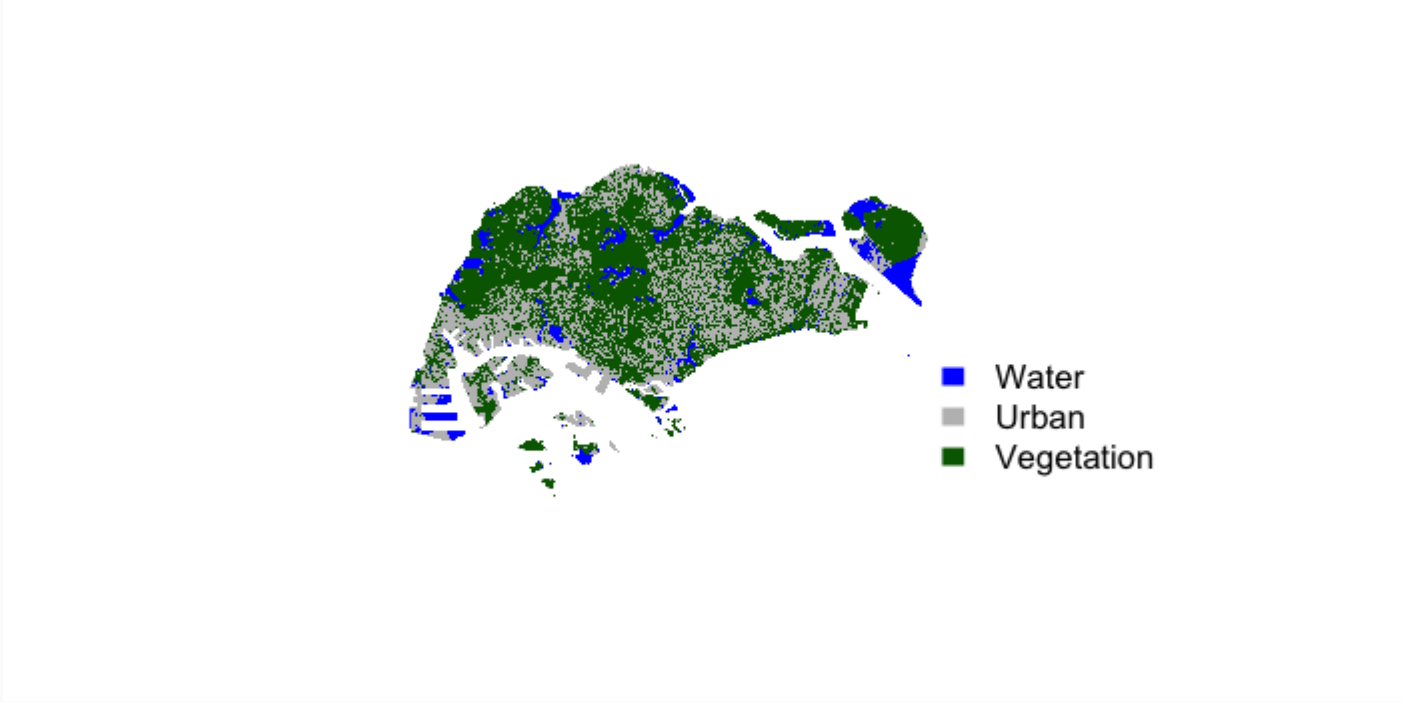


Patch

Area & edge: Area `area`, Perimeter `perim`, Perimeter-area-ratio `para`
Shape: Shape `shape`

Note: See R documentation for all functions. More egs grouped by *subject area* & *level of analysis*: Area/density/edge, Shape, Diversity, Contagion.

Landscape metrics

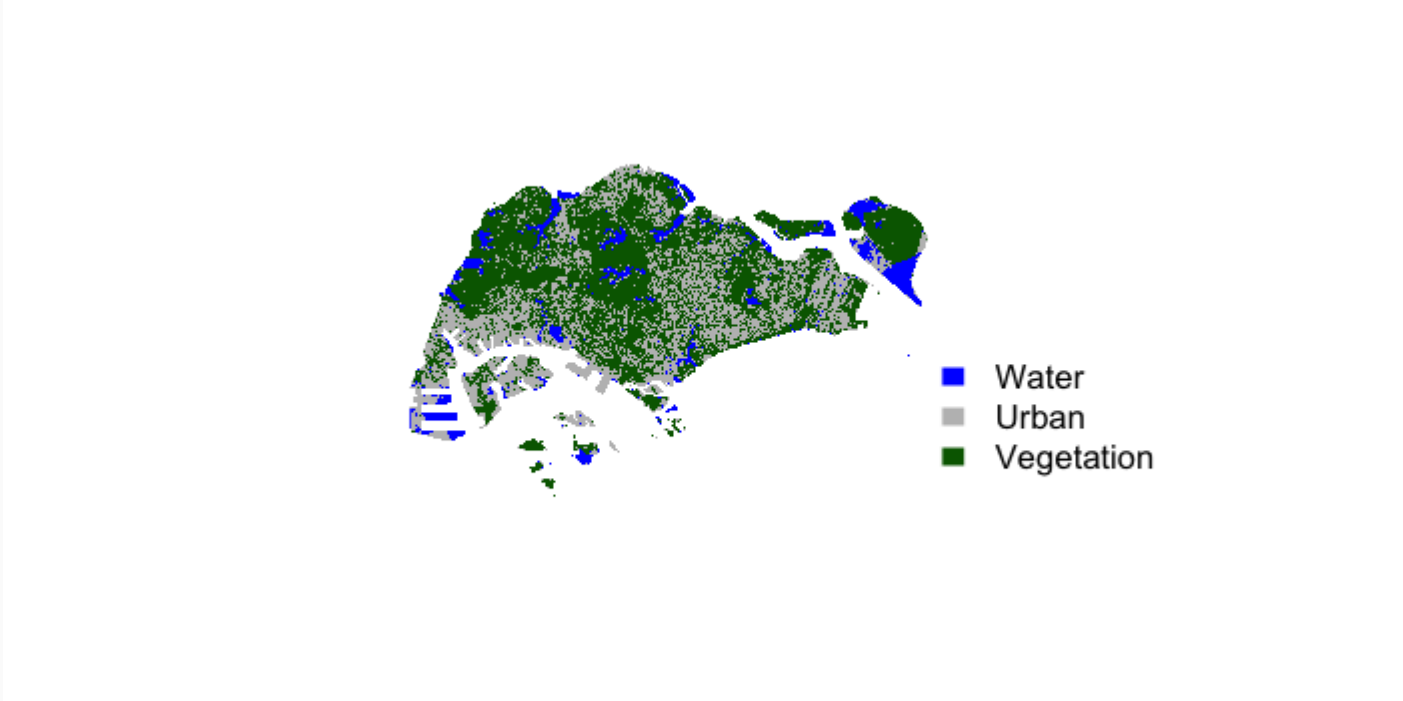


Class

Area & edge: Mean patch area `area_mn`, Edge density `ed`, Percentage of landscape `pland`, Largest patch index `lpi`
Shape: Mean perimeter-area ratio `para_mn`, Mean shape index `shape_mn`
Aggregation: No. of patches `np`, Patch density `pd`, Mean of euclidean nearest-neighbor distance `enn_mn`, Landscape shape index `lsi`

Note: See R documentation for all functions. More egs grouped by *subject area* & *level of analysis*: Area/density/edge, Shape, Diversity, Contagion.

Landscape metrics



Landscape

Area & edge: Total area `ta`, Mean patch area `area_mn`, Total edge `te`, Edge density `ed`
Shape: Mean perimeter-area ratio `para_mn`, Mean shape index `shape_mn`
Aggregation: No. of patches `np`, Patch density `pd`, Mean of euclidean nearest-neighbor distance `enn_mn`, Landscape shape index `lsi`
Diversity: Patch richness `pr`, Shannon's diversity index `shdi`, Simpson's diversity index `sidi`

Note: See R documentation for all functions. More egs grouped by *subject area* & *level of analysis*: Area/density/edge, Shape, Diversity, Contagion.

Landscape metrics

Load

Reload our saved landscape mosaic `landsatCover`

```
landsatCover <- raster('clean_data/landsat_landcover.tif')
```

Load required packages

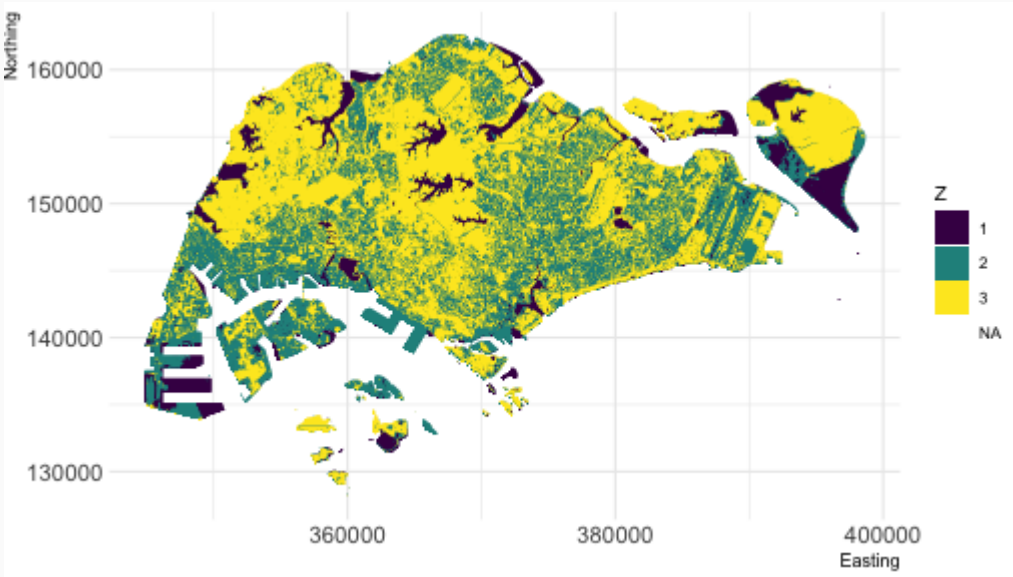
```
library(landscapemetrics)
library(landscapetools)
```

Landscape metrics

Load

Quick visualisation

```
show_landscape(landsatCover, discrete = TRUE)
```



Landscape metrics

Load

Check if `landsatCover` is in the right format

```
check_landscape(landsatCover)
```

Landscape metrics

Load

Analyse

List all available metrics

```
list_lsm()
```

Landscape metrics

Load

Analyse

List metrics by level and type

```
list_lsm(level = "patch", type = "area and edge metric")
```

Landscape metrics

Load

Analyse

Patch-level

```
lsm_p_area(landsatCover)
```

Note: Use `?lsm_p_area` for more info on the function

Landscape metrics

Load

Analyse

Class-level

```
lsm_c_ca(landsatCover) #total class area
lsm_c_area_mn(landsatCover) #mean area
```

Landscape metrics

Load

Analyse

Class-level

```
lsm_c_pland(landsatCover) #percentage of landscape
```

Vegetation cover of ~50% is relatively similar to results reported in Gaw et al. (2019), Data, 4(3), 116.

What are some reasons why our results may differ from actual land cover?

Landscape metrics

Load

Analyse

Landscape-level

```
lsm_l_ta(landsatCover) #total area of landscape
```

Landscape metrics

Load

Analyse

Calculate multiple metrics by name

```
calculate_lsm(landsatCover,
              what = c("lsm_l_ta", #total area of landscape
                       "lsm_c_ca", #total area of each class
                       "lsm_c_pland"), #proportional area of each class
              full_name = TRUE) #include info about metrics in results
```

Note: use `options(scipen=999)` to disable scientific notation for reported values

Landscape metrics

Load

Analyse

Calculate multiple metrics by level and type

```
calculate_lsm(landsatCover,
              level = "class",
              type = "aggregation metric",
              full_name = TRUE) #include info about metrics in results
```

```
## # A tibble: 45 x 9
##    layer level class    id metric   value name         type        function_name
##    <int> <chr> <int> <int> <chr>    <dbl> <chr>        <chr>       <chr>
##  1     1 class     1    NA ai      88.4   aggregation… aggregatio… lsm_c_ai
##  2     1 class     2    NA ai      79.5   aggregation… aggregatio… lsm_c_ai
##  3     1 class     3    NA ai      87.8   aggregation… aggregatio… lsm_c_ai
##  4     1 class     1    NA clumpy   0.889 clumpiness … aggregatio… lsm_c_clumpy
##  5     1 class     2    NA clumpy   0.687 clumpiness … aggregatio… lsm_c_clumpy
##  6     1 class     3    NA clumpy   0.735 clumpiness … aggregatio… lsm_c_clumpy
##  7     1 class     1    NA cohesi… 95.5   patch cohes… aggregatio… lsm_c_cohesi…
##  8     1 class     2    NA cohesi… 99.2   patch cohes… aggregatio… lsm_c_cohesi…
##  9     1 class     3    NA cohesi… 99.8   patch cohes… aggregatio… lsm_c_cohesi…
## 10     1 class     1    NA divisi…  1.00  division in… aggregatio… lsm_c_divisi…
## # … with 35 more rows
```

Landscape metrics

Load

Analyse

Calculate multiple metrics by level and type

```
calculate_lsm(landsatCover,
              level = "landscape",
              type = "diversity metric",
              full_name = TRUE)
```

What do these "diversity metrics" actually mean?

Landscape metrics

Load

Analyse

Discuss

How can we use our understanding of spatial patterns to improve landscape planning and design?

Ecosystem services framework

Discuss how landscape metrics link to ecological processes that provide benefits or 'services' to humans

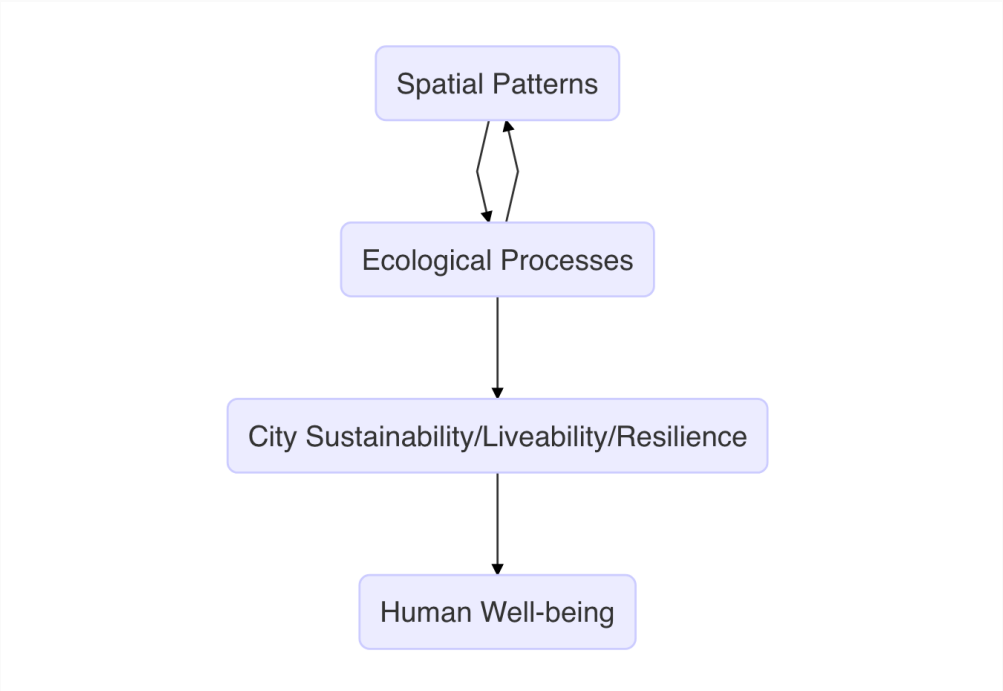
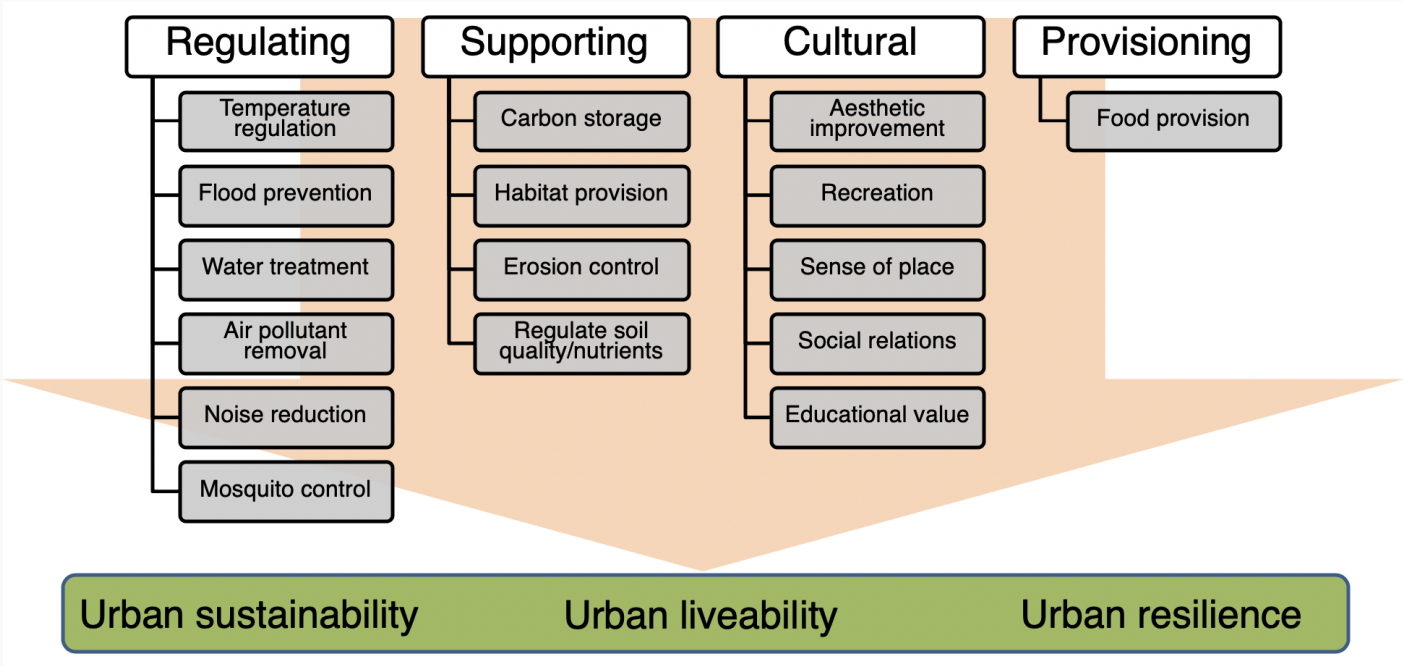


Figure adapted from Wu et al. (2015)

Ecosystem services framework

Discuss how landscape metrics link to ecological processes that provide benefits or 'services' to humans



Framework adapted from Tan et al. (2018)

Landscape metrics

E.g. Vegetation cover and temperature regulation

Correlation between metric & land surface temp (LST) [1]

Percentage of landscape `pland`: -0.7

- Larger proportional area of greenery associated with a lower LST
- Beijing should maintain existing amounts of greenery and adopt appropriate urban greening measures

[1] Li et al. (2013). Landscape and Urban Planning, 114, 1-8.

Landscape metrics

E.g. Vegetation cover and temperature regulation

Correlation between metric & land surface temp (LST) [1]

Mean patch area `area_mn`: -0.35

- Larger patches of greenery (less disturbed) could help lower temperatures
- Greening programs should reduce the fragmentation of green patches

[1] Li et al. (2013). Landscape and Urban Planning, 114, 1-8.

Landscape metrics

E.g. Vegetation cover and temperature regulation

Correlation between metric & land surface temp (LST) [1]

Patch density `pd`: 0.39
Edge density `ed`: 0.31

- Increased fragmentation/shape complexity of greenery is related to higher temperatures
- Policy measures should aim to decrease fragmentation

[1] Li et al. (2013). Landscape and Urban Planning, 114, 1-8.

Landscape metrics

💭 How can we compare metrics between different cities?

- Urban development: Sprawling vs. compact cities – Big difference in total area?
- City boundaries: Rural hinterlands included?

How can we account for such differences?

Questions?

Why analyse spatial patterns?

Landscape ecology: Conceptual models

Recap: R Environment

Land cover classification

Landscape metrics

Useful resources

Useful Resources

Free satellite imagery

- Overview
- http://earthexplorer.usgs.gov/
- https://lpdaacsvc.cr.usgs.gov/appeears/
- https://scihub.copernicus.eu/
- https://search.earthdata.nasa.gov/search

R Resources

- Spatial data science with R
- Remote Sensing Image Analysis in R
- R packages for spatial data analyses
- Interactive visualisations using `mapview`