Essentially; Pland has percentages for a value of coverage. To best pick a single digit and use that as the “greater percentage” would be to select a percentage cover greater than 60-70% although, a statistical test may be required to resolve this.

For example, if I have this dataset where I’m taking per year pland values so the coverage of a habitat per 5x5 modis cell – And for a specific ID, where ID represent each unique 5x5 modis cell (the geometry contained in this) I find that the pland for any specific ID does not equal to one when both year 2010 and 2019 are adjacent to one another. Its fragmented, however, the two years have different habitat types during this fragmentation from 2010 to 2019. My concern is that it should total 100% percentage, or one, however I may find that a particular habitat type is taking the majority of land type so 80% and the 20% is distributed, there’s much uncertainty with this because it may be a problem with the algorithm used to collect the landsat data, as you’ll find many differences between 2010 and 2019 that are less than 0.1%.

My question is a statistical one and most likely a prediction based one. Are there any tests for my situation where given that one habitat type is far greater than the other for example 80%, this is the greater accumulation of the pland coverage, should I just select the highest percentage when there’s a fragmentation? Or is there a statistical test, that synthesises the plandcover.

# A tibble: 15 x 4

id y2010 y2019 pland

<int> <int> <int> <dbl>

1 1 12 12 1

2 2 12 12 1

3 3 12 12 1

4 4 12 12 1

5 5 12 12 1

6 6 3 5 0.125

7 6 9 9 0.125

8 6 9 10 0.125

9 6 10 12 0.125

10 6 10 14 0.125

11 6 12 12 0.25

12 6 14 9 0.125

13 7 12 12 1

14 8 12 12 1

15 9 12 12 1

So essentially; given that id 6 has multiple splits in coverage for habitat types between the two years, picking pland that equals to 0.25 won’t be good enough. What can be done however, is combining the values that are similar in 2012; so 12 and 12 = 0.375.

There’s also the potential to split the data based on ‘most’ significant and least significant of habitat coverage.

For example, id 6 will have to rows; one where 12 = 0.375; the other is left as ‘uncertainty’. Or some significant name to represent the ‘lower’ aspect of the pland coverage suggesting fragmentation. Although, the highest value is always picked. The uncertainty can be named ‘fragmentation’.

There’s also the option of calculating this coverage against those fragmented habitats – which can be named ‘patches’.

The theoretical decision behind this is likeso:

An aggregation of these habitat values are taken; For example – The highest value is kept (when additioning with other pland values relative to habitat type), the lower value is also taken as an aggregate, although this is named ‘fragmentation’, or ‘patch’, because of it representing a lower percentage of habitat values in that modis cell – assumption is that it’s an uncertainty – although, further mathematical equation can be used to determine the patch-area of these values not equal to one, to see how impacted bird-encounter rates are towards fragmented areas.

Another perspective for calculating pland values;

Essentially taking the mean or sum of the landcover values for each ID and taking the sum of pland; although, taking the mean values may work better – for example, taking the mean values across the if the mean of y2010 is less than mean y2019 then its positive change; if its greater than y2019 then its negative change. The problem with this is there’s no deterministic way of representing change by this method, because these are habitat types and these values are merely codes for habitat type.

So, taking the overall pland value from any one habitat type, that with the highest will denote positive or negative change.

Reasons why I cannot do what is mentioned previously:

* The lc\_extract has different row numbers from 2010 to 2019; 2019 has more than 10x as many values because of more recent investigations in bird sightings. Given this, I lose out on recent data when comparing if I only use those values in 2010 from 2019; however, if I want to understand the changes, it would only make sense to check those changes in places already observed. So this could be an extra investigation to understand changes based on coordinates that are the same, to investigate what’s mentioned however, you can also mention the lack of observations because of computing disadvantages from 2010 i.e. popularised awareness.

I have tried taking the intersection of those values in 2010 from the other years for lc\_extract; However, most values are less than 1 suggesting fragmentation in the buffer zone. Somehow, to account for the changes using randomForest and prediction, the columns names must remain the same.

**KEY 🡪** *{So If I taking the intersection values, and take the changes of values from 2010 – the problem is that some years have more rows for a particular locality\_id than 2010; How would I calculate those values from this?*

*Unless I can find a close approximation of taking these habitat values and concentrating it onto a single habitat metric, and splitting the rest of this metric as ‘fragmentation’, then a more concise approximation between complete pland values within a modis cell, the majority of a habitat type and the fragmentation. }*

So essentially, another interpretation is to collect the coordinates from ebird-buff by extract them from *r* and using those values to calculate where ebird-checklists have been made. By doing so, the fragmentation problem can be corrected? How so, the id values should now represent only those values taken from ebird-buff, essentially, if 2010 < 2019, then using this to represent fragmentation?

The problem is that I can collect those values with fragmentations, and name it “fragmentation”, however, the locality ID does not have this problem. Unless, there is a way to connect the locality id with coordinates to those values with fragmentations relative to the buffer-zone, and using that 5x5 modis cell to represent the difference percentages of those fragmentated areas i.e. split ID values, and those whole pland metrics at 100%.

Calculating the pland values relative to each region, as opposed to any modis cell and framing this as – percentages, and potentially using landcover values to suggest which have the greatest frequency i.e. water appears 200 times, whilst deciduous forest 1000 times.

So, taking an overall percentage understanding of this. Although, this would require identifying which ID represents an entire region. In this case, then we can have 17 unique ID values to represent these regions, whilst duplicating these relative to each landcover and pland value. Then aggregating these values to determine an overall percentage of landcover type for each region from 2010 – 2019.

* What’s been mentioned above has been achieved. Although mapping this would be a challenge. Somehow, mapping these onto a raster, where each region should be divided into 15 fragments relative to habitat.

Mapping them based on percentage values for each coordinate? Such that, where there are no values = Pland is equal to 0, then replace that with a habitat where values are found. If habitats intersect than overlay the greater value.

I have a dataframe that looks like this:

Columns: 18

$ id <dbl> 1

$ year <dbl> 2010

$ pland\_00\_water <dbl> 0.721

$ pland\_01\_evergreen\_needleleaf <dbl> 0.0612

$ pland\_02\_evergreen\_broadleaf <dbl> 0

$ pland\_03\_deciduous\_needleleaf <dbl> 0.0934

$ pland\_04\_deciduous\_broadleaf <dbl> 1.1118

$ pland\_05\_mixed\_forest <dbl> 0.1242

$ pland\_06\_closed\_shrubland <dbl> 0

$ pland\_07\_open\_shrubland <dbl> 0

$ pland\_08\_woody\_savanna <dbl> 1.9831

$ pland\_09\_savanna <dbl> 2.1979

$ pland\_10\_grassland <dbl> 33.5914

$ pland\_11\_wetland <dbl> 1.6612

$ pland\_12\_cropland <dbl> 56.1875

$ pland\_13\_urban <dbl> 0.2884

$ pland\_14\_mosiac <dbl> 1.5861

$ pland\_15\_barren <dbl> 0.3927

There are 17 ID's, and above is just one out of 17 dataframes. I wish to overlay these values relative to the geometry belonging to IDonto a plot through a raster. Here's what I have tried:

#Polygon with `layer` that's named `ID` afterwards, and it's geometry points.

Simple feature collection with 1064723 features and 1 field

geometry type: POINT

dimension: XY

bbox: xmin: -9783543 ymin: 2731460 xmax: -5053120 ymax: 6018664

CRS: +proj=sinu +lon\_0=0 +x\_0=0 +y\_0=0 +R=6371007.181 +units=m +no\_defs

First 10 features:

layer geometry

1 1 POINT (-7399799 6018664)

2 1 POINT (-7409065 6016347)

3 1 POINT (-7406749 6016347)

4 1 POINT (-7404432 6016347)

5 1 POINT (-7402116 6016347)

6 1 POINT (-7399799 6016347)

7 1 POINT (-7418332 6014031)

8 1 POINT (-7416015 6014031)

9 1 POINT (-7413698 6014031)

10 1 POINT (-7411382 6014031)

#Take the geometry points from the polygon and bind them with the dataframe above by id

pland\_coords <- st\_transform(polygon\_data, crs = 4326) %>%

st\_coordinates() %>%

as.data.frame() %>%

cbind(id = polygon\_data$layer, .) %>%

rename(longitude = X, latitude = Y) %>%

inner\_join(y2010, by = "id")

#rasterise with the map detailing the area

forest\_cover <- pland\_coords %>%

# convert to spatial features

st\_as\_sf(coords = c("longitude", "latitude"), crs = 4326) %>%

st\_transform(crs = projection(r)) %>%

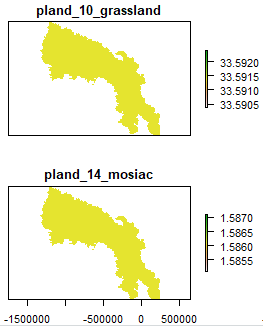
# rasterize points

rasterize(r) %>%

# project to albers equal-area for mapping

projectRaster(crs = st\_crs("ESRI:102003")$proj4string, method = "ngb") %>%

# trim off empty edges of raster

[](https://i.stack.imgur.com/VaX9O.png) trim()

However, when I plot the map I get separate plots and essentially I would like them to overlay one another relative to their percentage. Theoretically, I imaged this can be done by mapping habitat based on percentage values for each coordinate? Such that, where there are no values; Pland = 0, then replace that with a habitat where values are found. If habitats intersect than overlay the greater value.

This is what the plot looks like, although how would I get it to plot one map? whilst showing the values on the map

Essentially, somehow creating 0 as a value in areas that a habitat is not present, then replacing them by areas that are present. This may be possible with the raw dataframe?

To figure the ‘id’ column out. You notice that placing it from wide to long with X denoting every unique column. Then because there are 10 years as columns put into long format, then that same id will be assigned for each of those years. Suggesting that if you take these years separately, then you’ll only have a single year assigned. Then performing the calculation separately.

There’s also the outlook of attaching an ID column onto r\_centers, because this works naturally with the other dataset. Then counting them by ID when introducing the same unique ID onto Lc\_extract\_pred. However, the distribution of these should be dependent on ID and Layer, so the ID should match the layers, for example if ID goes up to 10,000 for Layer 1, then all of LC\_extract\_pred should have the same pattern.

Usually when counting by ID this works, however, test this with the original dataset to see if it can work this way, then figure out how to match ID with Layer, so the values correspond accordingly.

Currently, I have processed the data although, when understanding the percentage of pland for each habitat type, they’re different as to what is expected for the particular dataframe.

You have to reallocate those geometry points relative to the dataframe, especially those that are split into count data-form.

So finding which values of ID in the dataframe, when using count, is associated with the raster, then you can split or select those geometry relative to the ID, based on the total percentage of coverage.

Taking the count values, but with layer as group\_by(layer). Then performing the analysis individually, for each layer and year

However, there’s the possibility of getting a fragmented coverage by joining the dataframes in long format. Then stacking them when converted into a raster, so these maps join to complete the region. Somehow, the overlaying of this will be figured out.

Although, maybe overlaying it on an empty map, by overlay I mean all the different pland values onto this map. Which should provide a difference in coverage, then allocating the specific legend. A whole map could be created in this sense, then I could separate this by years, and discover the yearly changes relative to overall habitat percentage cover. I can also allocate a specific colour to each habitat type, and the percentage values on each habitat type for every region.

A regional schematic can be developed when keeping the layer values from the forest\_cover and the encounter rates. Then taking the ‘count’ of this, and developing a generalised linear model against these.

* Potentially, I could write the dataframes from encounter code for each individual year, then rbind all the years together, and develop a statistical model for encounter codes that include linear models.

Try taking the changes between years, and plotting a linear model against this. One way of determining a map likeso, is by minusing values from the pland-coords in 2019, from 2010, and using the changes as a prediction surface map.

This could be achieved by using lc\_extract\_pred of individual years, or the whole years.

A potential table to show the ranges of values for each layer. This should equate to 17 tables.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Range | Layer.1 | Layer.1\_count | Layer1.sum | Layer1.mean |
| 0 to 0.1 | c(0.002, 0.001, 0.004) | 3 | 0.007 | 0.0023 |
| 0.1 to 0.2 | c(0.102, 0.190) | 2 | 0.292 | 0.146 |

Include an ID with each list, unique to every list, hence these should be returned when rasterizing relative to landcover. Although, figuring out how to rasterize over many lists, such that they’re in the same raster is important.

The further improvement involve extracting the same amount of rows when rasterizing point data, because the attribute table de-selects some values.

To extract the values of ID belonging to each individual habitat type, from the encounter rate data, then joining those values with the ID of encounter rate, and extract the values relative to this and taking the mean.

Finally, this should compile a similar dataframe, with encounter rate relative to each habitat type for every layer.

Take the values from each individual year, so for example, take from lay.all year 2010 and from that year, request the values of encounter rate.

Because, the way I have done it, is that I’m taking the mean across all years, then storing that under 2010 encounter rates.

Collect the X Y variables from the pland-changes and encounter rates, then predict the estimated estimated encounter probability onto a surface map relative to the pland-changes in habitat, with a buffer-zone of 5-10km.

What does it mean to create predictions of estimated encounter-probabilities on relative habitat changes? Given that the estimated encounter-probabilities are predictions based on species presence – absence data, altogether, they’re likelihood estimations with a given error, of encountering a presence of species. Although, to predict the likelihood presence, as a further prediction of landcover changes, what does it mean to prediction, a predicted probability of encounter?

Pick every pixel or point within a buffer zone that is unique, and closest in distance to it. Calculate this distance (km); such that there is also no equivalence relation.

I.e. There is a set A with elements a, b, c

A = {a, b, c} Such that; The closest point to a is b, although a may be closest to be, however, equivalence relation’s are not allowed, hence c is the next nearest point. Therefore, c cannot be equivalent to b, so the next nearest point is a. Then calculate the distance of these points.

Calculate distance, then fit a poisson gam model, relative to latitude and distance for encounter rates (potentially fit these also for habitat, and year?) then create a prediction surface map relative to recent landcover values

Overlay points of yearly changes of encounter relative to differences in pland onto a map, to compare which areas have changed and others that have not. Make sure these points are distributed with different shape and sizes.

1. Take the counts of differences in habitat changes, to determine its significance relative to the linear models.
2. Use the encounter model code to fit the points of the raster onto a plat, which display the changes relative to a particular habitat since 2010.