* What packages are actually being used?
  + rgdal (and required package sp) - for reading in the spatial layers -- CHANGE TO SF
  + rgeos (for gIntersect function, gArea)
  + maptools (for unionSpatialPolygons function) - nb. also loads rgeos as a dependency.

Thoughts:

* if we do rasterise the PA layers, we need to make sure we remove any tiny overlaps which are where the boundaries of two sites overlap very slightly along the boundary.

INPUT FILES:

|  |  |  |  |
| --- | --- | --- | --- |
| **tabmf** | tabular | table of SitRecId; type (marine, terrestrial etc); ISO3; % montane; % marine; country. | this classification will need to be re-run with new techniques |
| **isos** | tabular | country categories: country names; ISO3; ISO\_BL; world back categories etc. | needs to be checked in advance. |
| **kbas** | spatial | KBA layer: SitRecId; region; country; ISO code etc.   * remove extraneous columns * ensure ISO3 codes match   + ISO3 = '---' -> assign to ABNJ or RUS based on country='High Seas' | ideally cleaned - without overlaps. |
| **pas** | spatial | modified PA layer from UNEP: WDPAID; year; parent\_ISO3; ISO3; shape length and area (from Arc) etc.  \*\* why pas$ISO3? WHERE IS IT USED? | conditions of loading - wkbPolygon |

FILES MADE IN CODE:

* code section 1 - still data cleaning

|  |  |  |  |
| --- | --- | --- | --- |
| cnpa | tabular | number of characters in the ISO3 field of WDPA   * + find where nchar > 4   + -> transboundary PA (in more than one country) | TODO: ignore transboundary PAs - don't use them to allocate random years to PAs missing years. |

* code section 2 - the actual analysis

|  |  |  |  |
| --- | --- | --- | --- |
| areasov | table | SitRecId; kba [area of KBA in dd]; ovl [area of overlap with PAs in dd]; year; nPAs; percPA; ISO; COUNTRY |  |
| pac and kbac | spatial | PAs and KBAs subset to the country we're currently dealing with |  |
| ovkba | tabular | matrix of PAs and KBAs in the country and whether or not the overlap (TRUE / FALSE, produced by gIntersects()) |  |
| kbaz | spatial | the KBA that we're currently dealing with. |  |
| pacz | spatial | loop over the PAs that overlap with kbaz - pacz is all the PAs that the KBA that we're currently dealing with overlap with. |  |
| ovf | spatial polygons (no data frame) | polygons which are the overlap of the current kba (kbaz) with the PAs with which it overlaps (pacz) - produced by gIntersection(). |  |
| ovfpol | spatial polygons data frame | add data back in to ovf (data including year) |  |
| ovf1 and ovf11 | spdf | overlap in year 1 - then unionised if more than one PA in year 1 to make ovf11 |  |
| ovlz | value | gArea of ovf11. |  |
| areasov1 | data frame | SitRecId; kba; ovl; 1981; year; nPAs |  |

Parts of the analysis:

* 1. All data available - both year of PA and polygons for both KBA and PA
  2. No year available - give PA a random year
  3. No % overlap available - give it a random based on the range for that country.

ANALYSIS

* 1. Read in the files:
     1. TODO: ability to run in batches of ISO codes - to run one third of countries at a time.
     2. NB. need to repair geometry of all the data frames.
     3. TODO: Sort out character encodings - UTF8 etc.
  2. Convert dataframe columns to the right classes - numerics, characters etc.
     1. TODO: Convert all factors to characters, check that number columns are numerics.
  3. Change ISO3 codes in the PA to match KBA layer where they are wrong
     1. this has to be done ad-hoc to fix wrong ISO codes
     2. ISO in wdpa matches ISO3 in isos, need to match to ISO\_BL in isos.
     3. use the PARENT\_ISO3 column in wdpa to match.
  4. Transboundary PAs:
     1. current code picks the first ISO3 code and uses that (using strsplit() on ';')
     2. TODO: \*\*\* why does it matter what ISO codes the wdpa has? we're just overlapping them…
        1. possible reason: to allocate years to PAs in a country which doesn't have a year
        2. -> just remove this.
        3. OR duplicated each transboundary PA into each country.
     3. The KBA/PA overlap runs country-by-country, so starts with subsetting both the PA and KBA datasets to a particular country.

TODO: add in to data cleaning:

* 1. To Do Transboundary PAs - duplicate in all the countries that they span - strsplit the ISO code, find the individual ISO3 codes, duplicate the site record into all of those countries.
  2. To Do Sites where ISO3 = '---' -> assign to High Seas (ABNJ) or Russia (RUS) - make sure all the attributes are reassigned, not just the ISO codes
  3. To Do PAs with missing years - assign a random year within range of years for PAs of that country.
  4. To Do Add in area for each KBA.

### end of data cleaning ###

OVERLAP ANALYSIS

Preliminary analyses

* 1. Calculate areas (in decimal degrees) of all KBAs
     1. -> add into a summary data frame.

Preliminary checks

* 1. Remove countries which have no PA polygons in the dataset
     1. -> Send the KBAs straight to output with overlap = 0
  2. ~~Use~~ **~~gIntersect~~** ~~(or similar) to find KBAs which don't overlap with any PAs~~
     1. ~~-> Send the KBAs straight to output with overlap = 0~~

=> reduced our input to the loop to just KBAs that do have intersection with PAs.

* 1. LOOP BY COUNTRY:
     1. Subset PAs and KBAs by country
        1. [some other data cleaning stuff that we can move above]
     2. Plot KBAs and PAs for country
     3. IF length(PAs of country) == 0{ #i.e. if no PAs in country

THEN make areasov dataframe with 0s and kba=9999 to show error. }

* 1. IF (there are PAs){

ovkba <- NULL

ovkba <- gIntersect(pac, kbac, by.id=T) ## TESTING if the polygons overlap.

IF no overlap, THEN fill in areasov with 9999s in all columns.

IF there are overlaps, then:

Assign random year if year is missing

LOOP BY KBA IN COUNTRY:

* + set to 9999 in case gArea doesn't work, then calculate area of KBA.
  + Look for how many PAs overlap with this KBA, put them in pacz
  + Order PAs by year
  + Calculate area of overlap in year 1 (after unionising year 1 as year 11)
  + for year 2:no. years:
    - calculate remaining area of KBA unprotected (i.e. if the KBA is now fully covered, we can stop calculating) - using the area of intersect with PA in year 1 and the overall area of the kba (akba). If all the area is covered in year 1, we don't need to cycle through the rest of the years. If there is more than 2% of the KBA area remaining, carry on.
    - for each next year (e.g. year 2, year3):
      * unionise all the PAs in this year to make 1 polygon - as yearnext
      * get all the PAs from previous years as yearprev
      * **gDifference**(yearnext, yearprev) - finds what of yearnext is not in yearprev. This is IMPORTANT to avoid counting twice any area that is in two PAs of different years.

##########################################

GRAPHS:

* + ERROR BARS are only where we have had to insert random years for PAs - error is in the percentage coverage of KBAs by PAs.