Satellite data at the Australian Antarctic Division & the Antarctic Climate and Ecosystems CRC

MICHAEL SUMNER AND BEN RAYMOND

SATELLITE OCEANOGRAPHY USERS WORKSHOP, 9-11 NOV 2015

Australian Government
Department of the Environment
Australian Antarctic Division

Australian Antarctic Division and Antarctic Climate and Ecosystems CRC

- broad use of remotely-sensed oceanography products for the Southern Ocean
- species distribution, ecosystem modelling, biodiversity research, fisheries
- central repository of remote sensing data sets used across multiple groups
- analytical toolbox in R built on remote sensing data library

1. description of data collection (for administrators)
2. toolbox description and examples (for users)

R data library tools

- data repository mirrored from a range of online sources
- updated with regular, automatic downloads of new data
- data sources are user-configurable - straightforward to add new sources
- system is extensible, can be installed anywhere (in complete or partial form)
- leverages R and normal operating system utilities (free and open software)

AAD / ACE data library

- collection includes > 200,000 files of various formats (HDF, NetCDF, GeoTIFF, raw binary, vector formats, imagery)
- master copy is at ACE CRC, we mirror it at AAD for local use
- delivered as standard metadata-rich R Spatial objects, via “raadtools” package

Example data source

name	NSIDC SMMR-SSM/I Nasateam sea ice concentration
description	Passive-microwave estimates of sea ice concentration at 25km spatial resolution. Daily and monthly resolution, available from 1-Oct-1978 to present.
source_urls	ftp://sidads.colorado.edu/pub/DATASETS/nsidc0051_gsfc_nasateam_seaice
reference	http://nsidc.org/data/nsidc-0051.html
do_sync	TRUE
method	wget
method_flags	\=--exclude-directories=pub/DATASETS/nsidc0051_gsfc_nasateam_seaice/final-gsfc/browse --recursive --level=inf
postprocess	
access_function	readice
data_group	Sea ice
user	
password	
wget_flags	
http_proxy	
ftp_proxy	
local_file_root	/raad/data
clobber	1
skip_downloads	FALSE
wait	0

The raadtools package

- integrated with standard R tools
- leverages NetCDF, HDF, GDAL, GEOS, ...
- catalogue of available files in the repository
- read[x] functions for product [x] with consistent structure
- files hidden by default, functions work by date-time query
  - other arguments for spatial subset or specifics (U / V vs. mag / dir)
- functions augment incomplete/missing metadata, orientation, etc.
- flexible extraction methods, read slabs or extract point-in-time values

raadtools functions

- **readcurr**: ocean surface current vectors
- **readice**: sea ice concentration
- **readssh**: sea surface height / anomaly
- **readsst**: sea surface temperature
- **readwind**: surface wind vectors
- **readtopo**: topographic data, bathymetry and elevation
- readmodissst: monthly MODIST Mapped
- readoc: aggregator / gridding for L3BIN ocean colour
- model outputs: ROMS, GEM,

- custom and new read functions are easily created
- delivered via AAD desktops, AAD Linux server, and RStudio Server in the Nectar cloud
- working system can be cloned and carried (with minor configuration)

Structure of read functions (1)

```
##' Read Polar model data.
##'
##' Polar biology model, on Stereographic grid.
##' @title Polar data
##' @param date date or dates of data to read
##' @param returnfiles if TRUE return just the files from \code{polarfiles}
##' @param time.resolution choice of temporal resolution, weekly only
##' @param xylim crop or not
##' @param ... ignored
##' @return RasterLayer or RasterBrick
##' @export
readpolar <- function(date,  time.resolution = "weekly", xylim = NULL, returnfiles = FALSE, ...) {
      ## private function to read a single file/time slice specific to this product
      read0 <- function(x) {
        proj <- "+proj=stere +lat_0=-90 +lon_0=180 +ellps=sphere"
        offset <- c(5946335, 5946335)
        dims <- c(1280L, 1280L)
        pdata <- readBin(x, numeric(), prod(dims), size = 4, endian = "little")
        pdata[pdata < 0] <- NA
        x <- list(x = seq(-offset[1L], offset[1L], length = dims[1L]),
                  y =  seq(-offset[2L], offset[2L], length = dims[2L]),
                  z = matrix(pdata, dims[1L])[rev(seq_len(dims[2L])),])
        raster(x, crs = proj)

    }
    ## process input options
    time.resolution <- match.arg(time.resolution)
    files <- polarfiles()
    if (returnfiles) return(files)

    ## provide a default for readpolar() with no input
    if (missing(date)) date <- min(files$date)
    date <- timedateFrom(date)


   ## continued . . .
```

Structure of read functions (2)

```
  ## readpolar <- function(date,  time.resolution = "weekly", xylim = NULL, returnfiles = FALSE, ...) {

  ## cont.

    ## normalize input to file dates
    files <- .processFiles(date, files, time.resolution)

    ## crop if requested
    cropit <- FALSE
    if (!is.null(xylim)) {
        cropit <- TRUE
        cropext <- extent(xylim)
    }

    ## collect individual slices and bundle as a RasterBrick
    nfiles <- nrow(files)
    r <- vector("list", nfiles)
    for (ifile in seq_len(nfiles)) {
        r0 <- read0(files$fullname[ifile])
        if (cropit)
            r0 <- crop(r0, cropext)
        r[[ifile]] <- r0
    }
    r <- brick(stack(r))

    ## tidy up the names and the 3rd axis metadata
    names(r) <- sprintf("polar_%s", format(files$date, "%Y%m%d"))
    setZ(r, files$date)
}
```

R Tools - time-series data

Raster can read from a variety of file types, and deal with 3D space-time grids, or assemble them from individual files

```
library(raster)
dp <- "//aad.gov.au/files/AADC/Scientific_Data/Data/gridded/data"
sp <- "seaice/ssmi/ifremer/antarctic/daily/2013"
files <- c("20130719.nc", "20130726.nc", "20130802.nc", "20130809.nc", "20130816.nc", "20130823.nc",
"20130830.nc", "20130906.nc")
fs <- file.path(dp, sp, files)
icedata <- stack(fs)
```

R AAD tools adds simplified helpers for rogue file collections:

```
library(raadtools)
dates <- seq(as.Date("2013-07-19"), by = "1 week", length = 8)
icedata <- readice(dates, product = "ssmi")
```

```
> icedata
class       : RasterBrick
dimensions  : 664, 632, 419648, 8  (nrow, ncol, ncell, nlayers)
resolution  : 12500, 12500  (x, y)
extent      : -3943750, 3956250, -3943750, 4356250  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=stere +lat_0=-90 +lat_ts=-71 +lon_0=0 +k=1 +x_0=0 +y_0=0 +datum=WGS84 +units=m +no_defs +ellps=WGS84 +towgs84=0,0,0
data source : in memory
names       : X20130719.nc, X20130726.nc, X20130802.nc, X20130809.nc, X20130816.nc, X20130823.nc, X20130830.nc, X20130906.nc
min values  :           1,           1,           1,           1,           1,           1,           1,           1
max values  :         100,         100,         100,         100,         100,         100,         100,         100
time        : 2013-07-19, 2013-07-26, 2013-08-02, 2013-08-09, 2013-08-16, 2013-08-23, 2013-08-30, 2013-09-06
```

Raster tools for 2D and 3D grids

- cellFromXY(x, pts)
- xyFromCell(x, cellnum)
- getValues(x, format = “matrix”)
- adjacent(x, 10)
- writeRaster(x, filename, …)

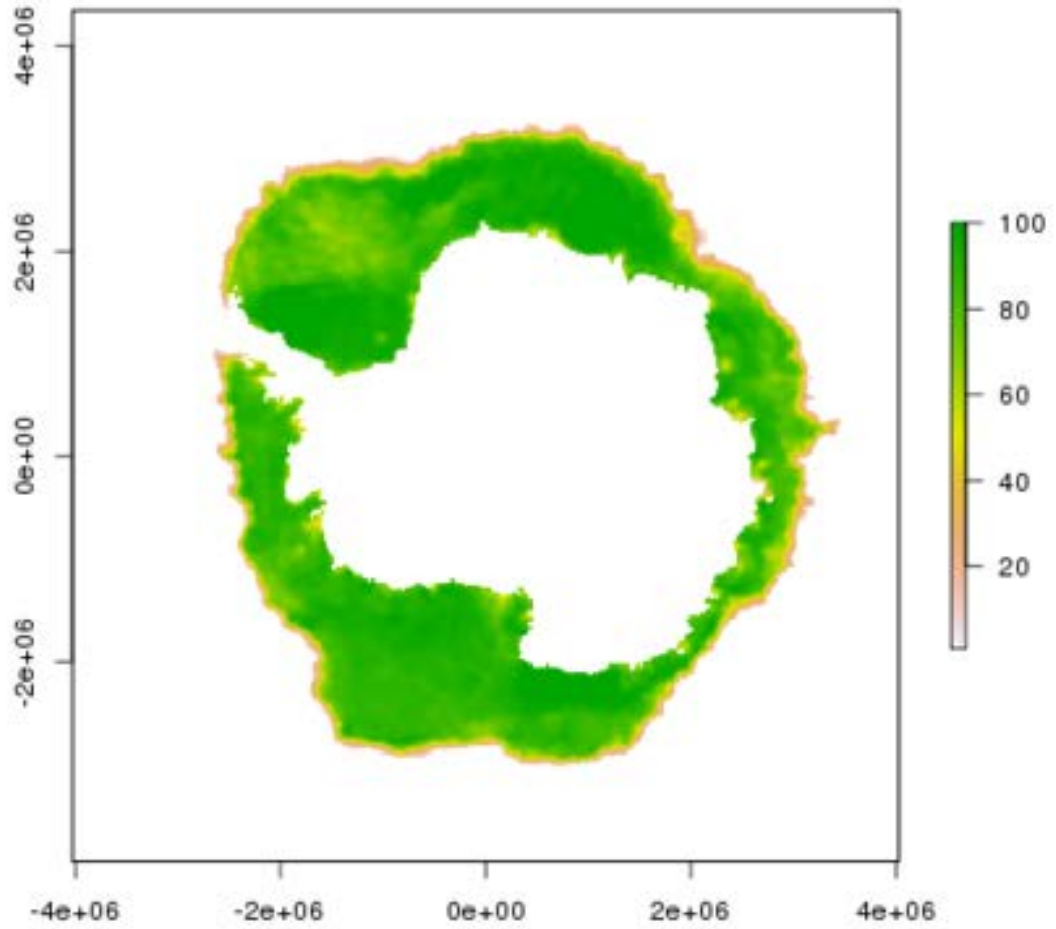
High level

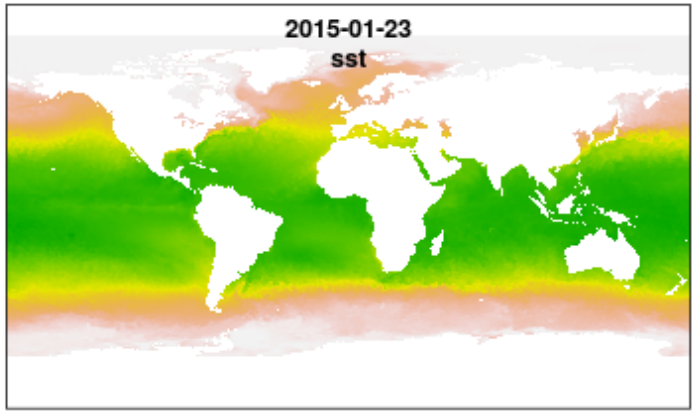
- extent(x); dim(x); projection(x)
- merge, crop, projectRaster, aggregate, reclass, rasterize, distance, focal, terrain …
- Raster algebra: x + y; sqrt(y); log(x) etc.
- Temporal or 'Z' axis for 3D

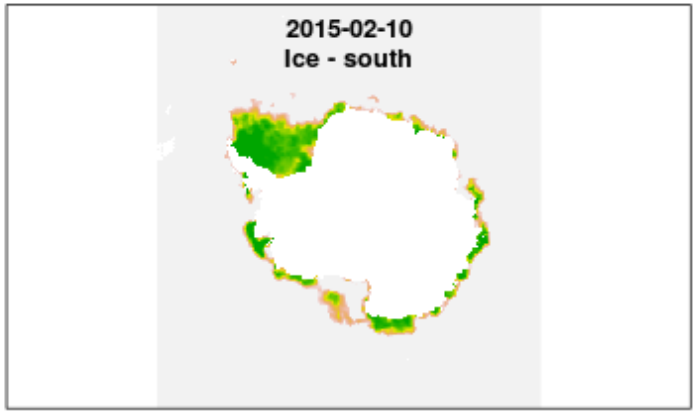
Regridding and extraction

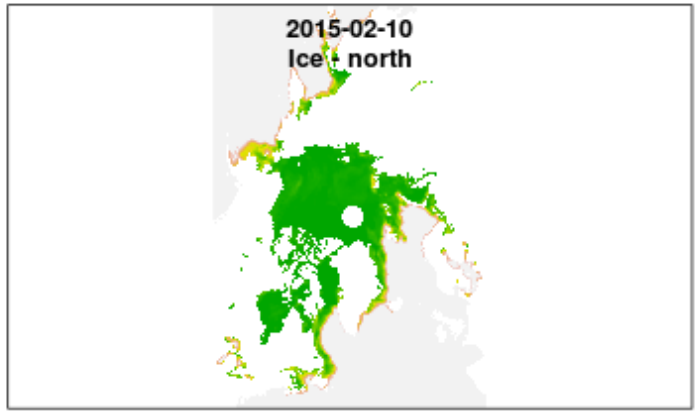
- nearest-neighbour / bilinear-interpolation resampling
- aggregation/disaggregation to coarser/lower resolution
- warping (resampling grid points via projection transformation)

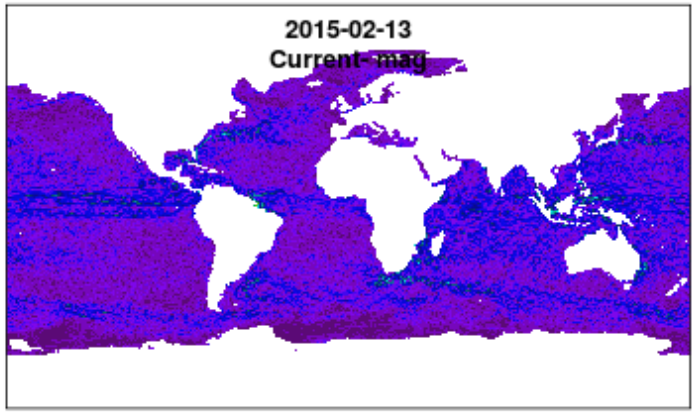
```
x <- readice("2013-06-15")
plot(x)
```

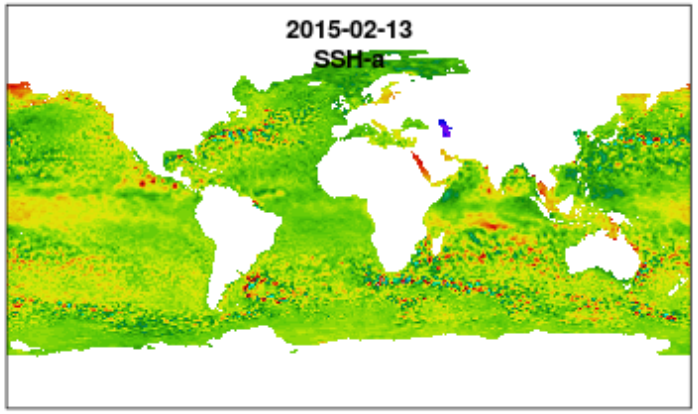


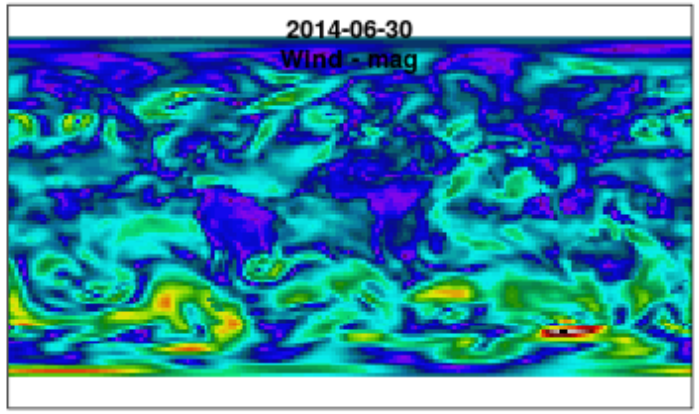

2015-01-23
sst

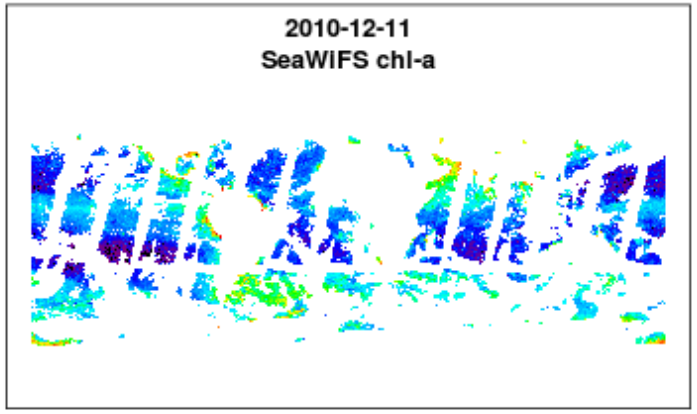

2015-02-10
Ice - south

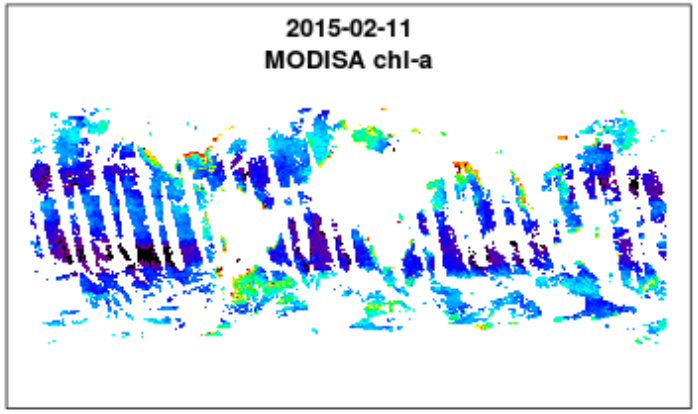

2015-02-10
Ice - north


2015-02-13
Current - mag

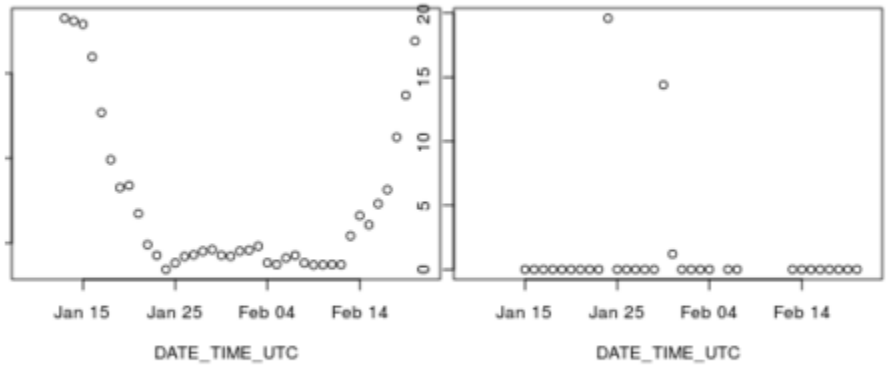

2015-02-13
SSH-a


2014-06-30
Wind - mag


2010-12-11
SeaWIFS chl-a

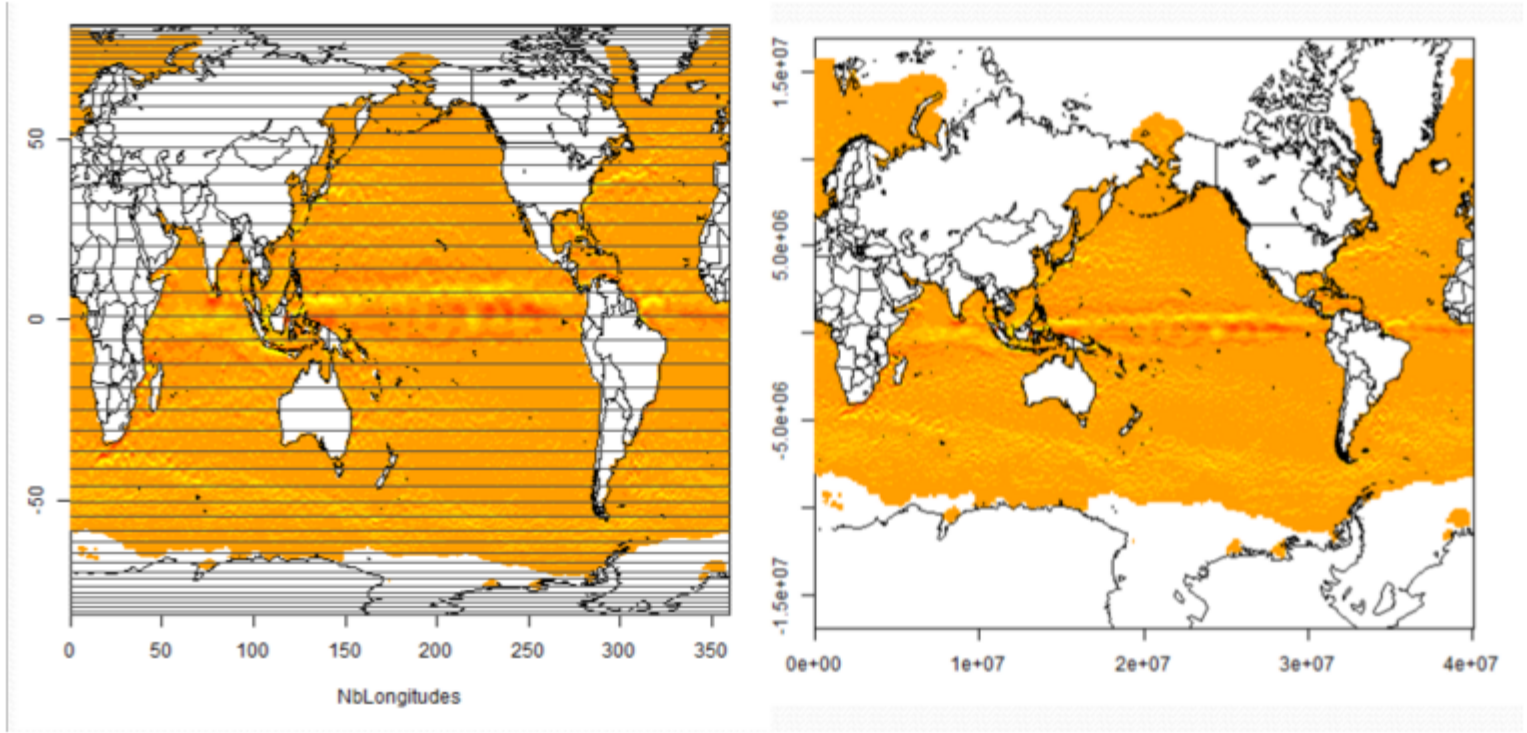

2015-02-11
MODISA chl-a

Flexible extraction - space-time overlay




```
## extraction methods for "longitude", "latitude", "date-time"
voyage$sst <- extract(readsst, voyage[,1:3])
voyage$ice <- extract(readice, voyage[,1:3])
```

Mercator-grid, longlat vs native



PROJ.4 +proj=merc +lon_0=0 +k=1 +x_0=0 +y_0=0 +ellps=WGS84 +over +units=m +no_defs

- Left panel is *rectilinear*, latitude lines are denser towards the poles.
- Right panel is *regular in native projection*, and this allows straightforward usage (and fast extraction methods).

Open software

- R tools for data library:

https://github.com/AustralianAntarcticDataCentre/raadsync

https://github.com/AustralianAntarcticDivision/raadtools

- R tools for L3BIN ocean colour:

https://github.com/mdsumner/roc

- R tools for coupling physical ocean and ecosystem models

Examples

- R AAD tools server, in Nectar cloud (Tasmania node)
- Ubuntu Linux + GDAL + NetCDF + HDF + GEOS + R + Rstudio Server
- AAD/ACE data library mounted from RDSI / TPAC
- http://144.6.224.186:8787/

- Thanks!

Michael Sumner michael.sumner@aad.gov.au

Ben Raymond ben.raymond@aad.gov.au

Satellite Oceanography Users Workshop, 9-11 Nov 2015

Australian Government
Department of the Environment
Australian Antarctic Division