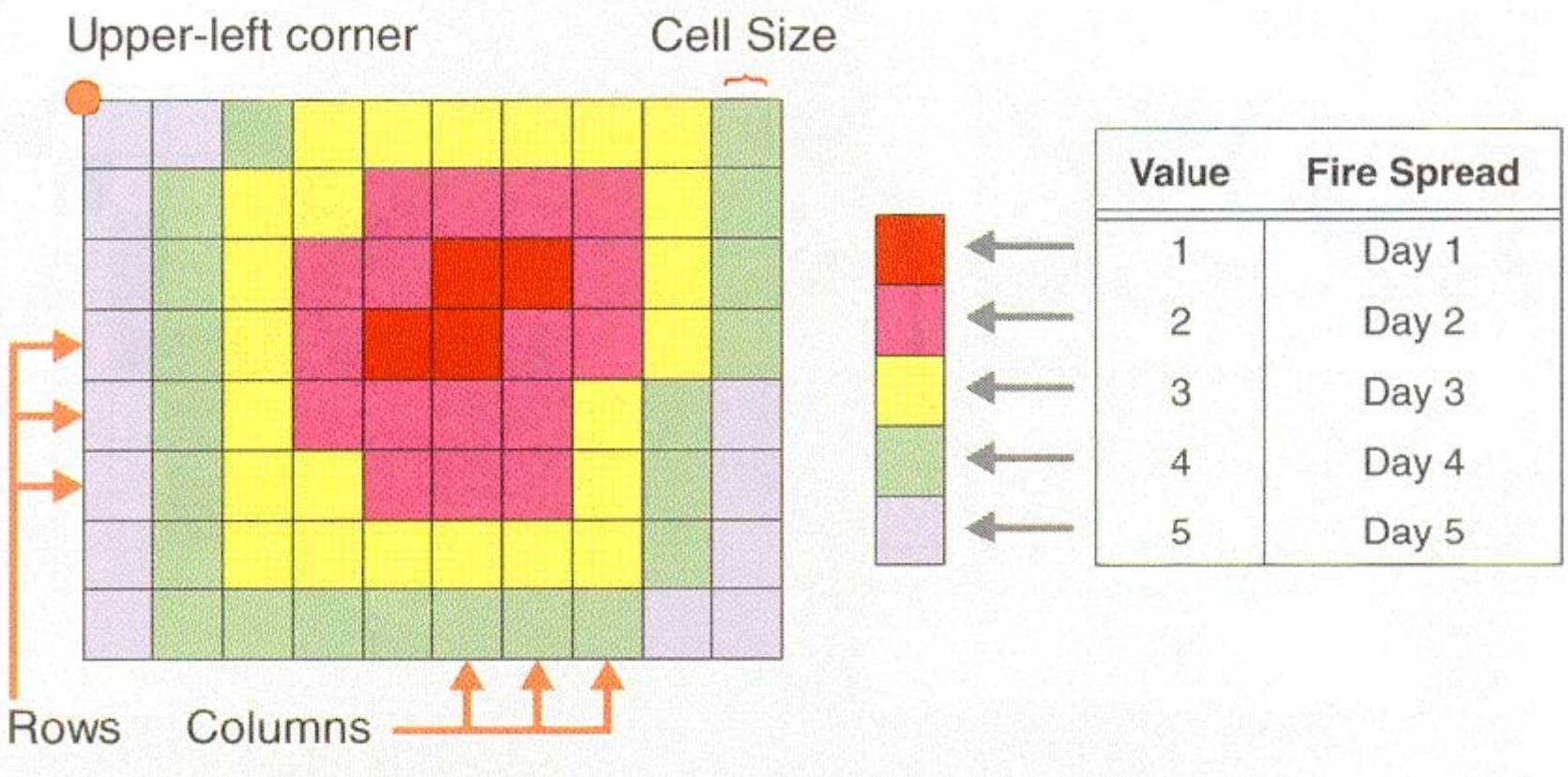
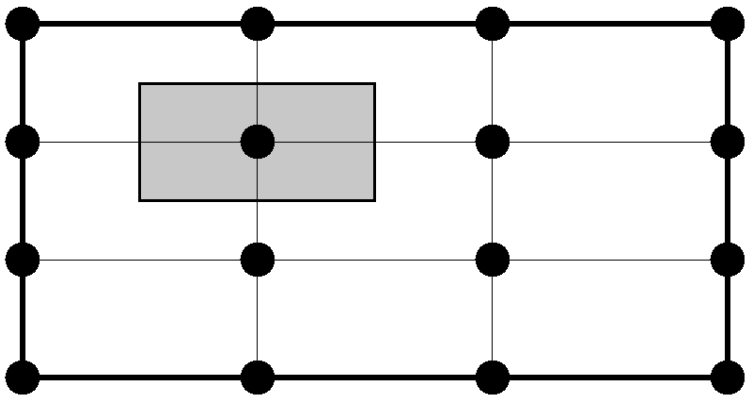
Gridded data sets (*.grd files)

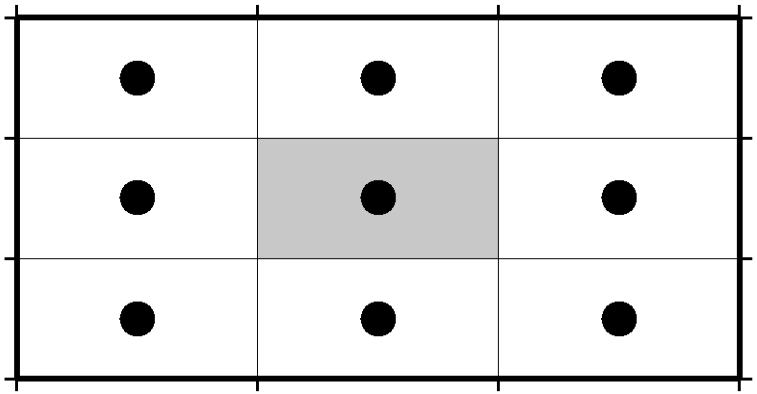
- Stored as binary files
  - written in netCDF
  - Do not attempt to open in a texteditor!
- Grids are equidistant
  - Grid spacing is fixed (Δx, Δy are constants)
- Header section contains all information
  - w/e/s/n region
  - grid spacing
  - various text strings describing the data

Raster Data


Upper-left corner
Cell Size
Rows
Columns
Value
Fire Spread
1
Day 1
2
Day 2
3
Day 3
4
Day 4
5
Day 5

Grid file registrations





- Gridline registration has 1 row/column more than pixel registration
- Gridline registration has nodes at gridline intersections whereas pixel registration has nodes centered on the grid boxes

Grid use in this course

- Contouring of grids
  - Assumes we have grids ready to use
- Gridding of arbitrarily spaced data
  - Have (x,y,z) data but need a regular grid
- Imaging of grids
  - Assign colors and illumination
- Mathematical manipulation of grids
  - Filtering, masking, whatever

Querying Raster Data

- Raster data cannot be opened with a text editor
- grdinfo gives information about a grid
  - min, max
  - grid spacing
  - nx and ny
  - statistics
  - grid registration
  - Some options exist to modify output format; see the grdinfo man page

Example 23 – Bermuda

- In the Data directory there are two files for Bermuda covering an area -R-66/-60/30/35
  - age of oceanic crust
    bermuda_age.grd
  - bathymetry from ETOPO5
    bermuda_bath.grd

Example 23 – Bermuda

*GMT5
x_min: 294 x_max: 300

- Type the following in the terminal:

grdinfo bermuda_age.grd

```
bermuda_age.grd: Title: Sea floor age from Muller et al., 1998
   [v1.6]
bermuda_age.grd: Command: grdraster 1 -R-66/-60/30/35 -
   Gbermuda_age.grd -V
bermuda_age.grd: Remark: /geo/data/grid/i2grids/age_1.6.i2
bermuda_age.grd: Gridline node registration used
bermuda_age.grd: Grid file format: nf (# 18) GMT netCDF format
   (float)  (COARDS-compliant) [DEFAULT]
bermuda_age.grd: x_min: -66 x_max: -60 x_inc: 0.1 name: Longitude
   in degrees nx: 61
bermuda_age.grd: y_min: 30 y_max: 35 y_inc: 0.1 name: Latitude in
   degrees ny: 51
bermuda_age.grd: z_min: 96.47 z_max: 140.92 name: Ma
bermuda_age.grd: scale_factor: 1 add_offset: 0
```

Example 23 – Bermuda

- What is the min and max age of oceanic lithosphere around Bermuda?

```
grdinfo bermuda_age.grd
```

- What about the bathymetry?

```
grdinfo bermuda_bath.grd
```

- Answers: age 96 and 141 m.yrs, bath -5475 to -89 meters

Contouring

- grdcontour will draw contours
- Takes **–J** and optionally **–R** (default region is assumed to be the grid region)
- Several options determine how the contouring will take place
  - contour interval
  - annotation interval
  - contour limits
  - various embellishments

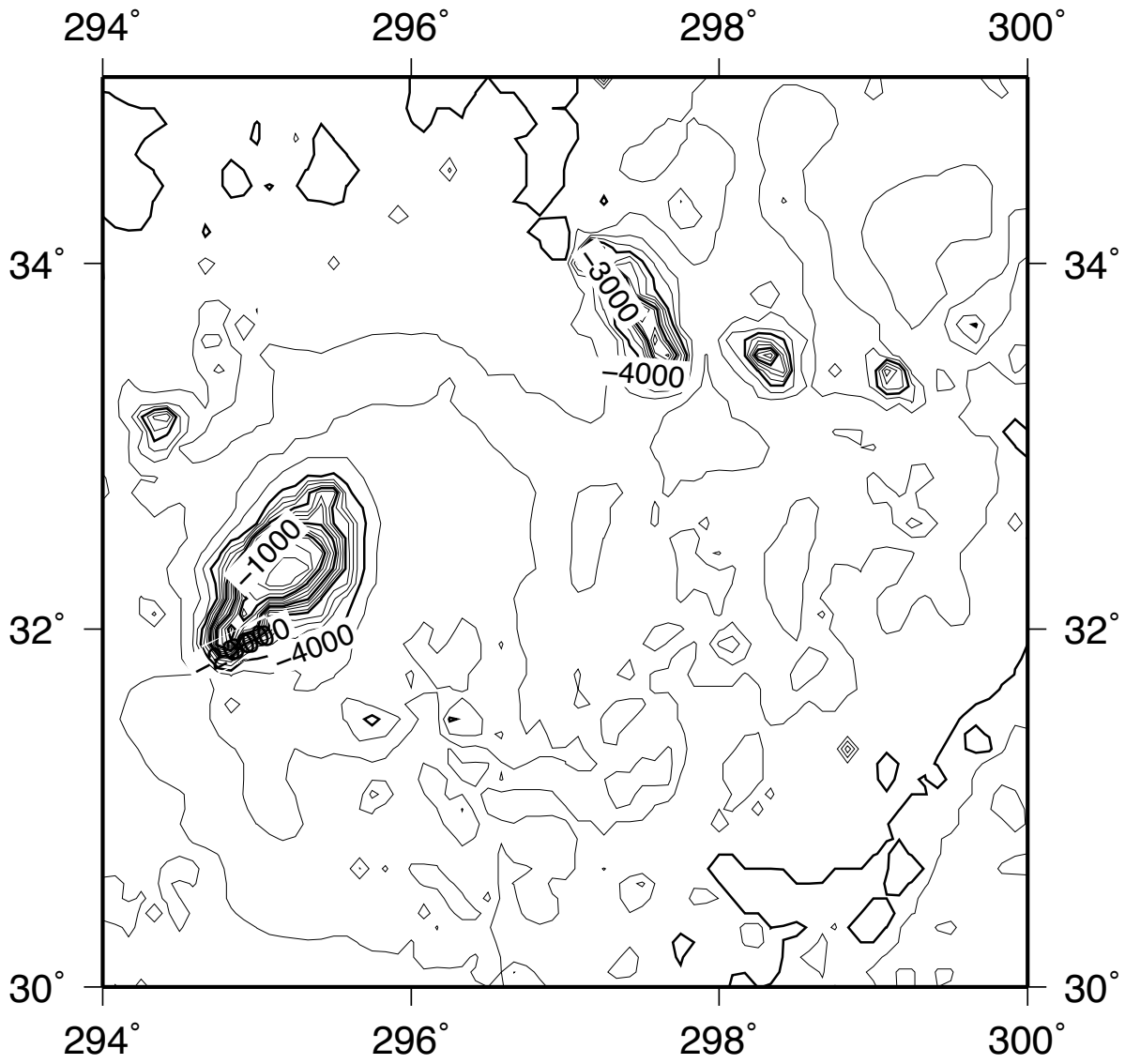
grdcontour options

Option	Purpose
–A	Set annotation interval
–C	Set contour interval or name of cpt file
–G	Choose where annotations occur (see App O)
–L	Limit the range of contours to draw
–N	Append unit to contour labels
–Q	Skip contours with very few points
–S	Resample contours to make them smoother
–T	Tick and annotate innermost closed contours
–W	Set pens for contour lines
–Z	Scale/offset values before processing

Example 23 – Bermuda contouring

- Make Mercator map with 250m contour interval and 1 km annotation interval of the Bermuda bathymetry

```
grdcontour bermuda_bath.grd –JM10 –
C250 –A1000 –P –B2 > ex16.ps
```


294°
296°
298°
300°
34°
32°
30°
−3000
−4000
−1000
−4000

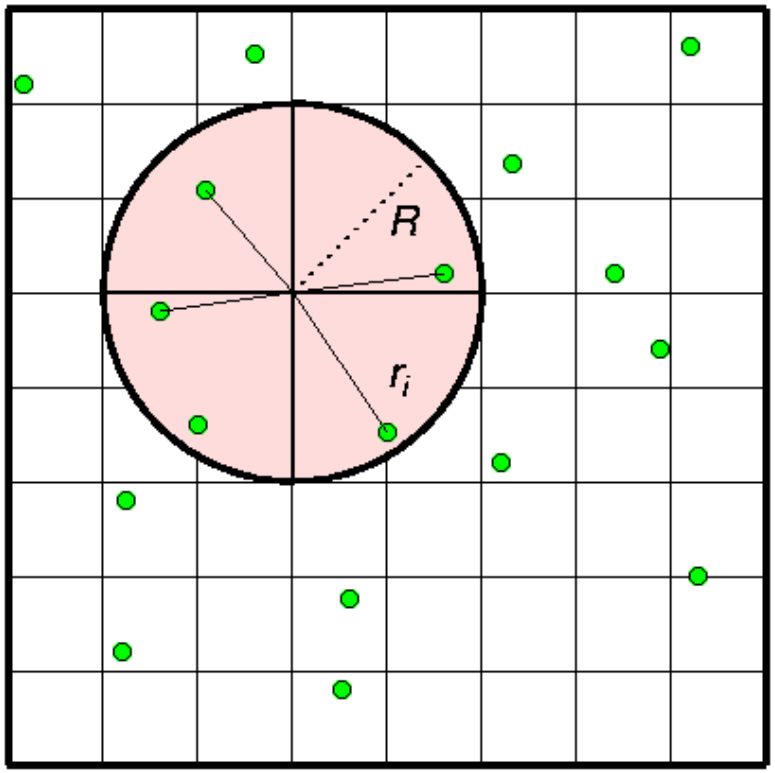
grdcontour theme variations

- Add smoothing with **–S4**
- Skip small features with **–Q**
- Override region using **–R**-70/-60/25/35
- Scale data to km and use km in the annotations (**–A+um**)

Gridding of data

- We distinguish between two scenarios:
  - The (x,y,z) data are already on a regular lattice
    - Simply reformat with xyz2grd
  - The (x,y,z) data are unevenly distributed
    - Grid data using local procedures
      - nearest neighbor (nearneighbor)
      - triangulation (triangulate)
    - Grid data using global procedures
      - surface splines in tension (surface)
    - All need -Rw/e/s/n, -Idx/dy, and -Ggridfile

Nearest Neighbour Gridding



$$w_i^r = \left(1 + \frac{9r_i^2}{R^2}\right)^{-1}$$

- Assigns an average value to each node that has one or more points within a radius centered on the node
- Average value is a weighted mean of the nearest point from each sector inside the search radius i.e. points have radial weight
- R = search radius
- r = distance from node

nearneighbor options

- Takes **–R**w/e/s/n, **–I**dx/dy, and **–G**gridfile
- Four other options are relevant:

Option	Purpose
–Sradius[**k**]	Search radius. Append **k** for km
–Eempty	Sets empty nodes to this value [NaN]
–Nsectors	Sets the number of sectors [4]
–W	Read point weights as well (x,y,z,w)

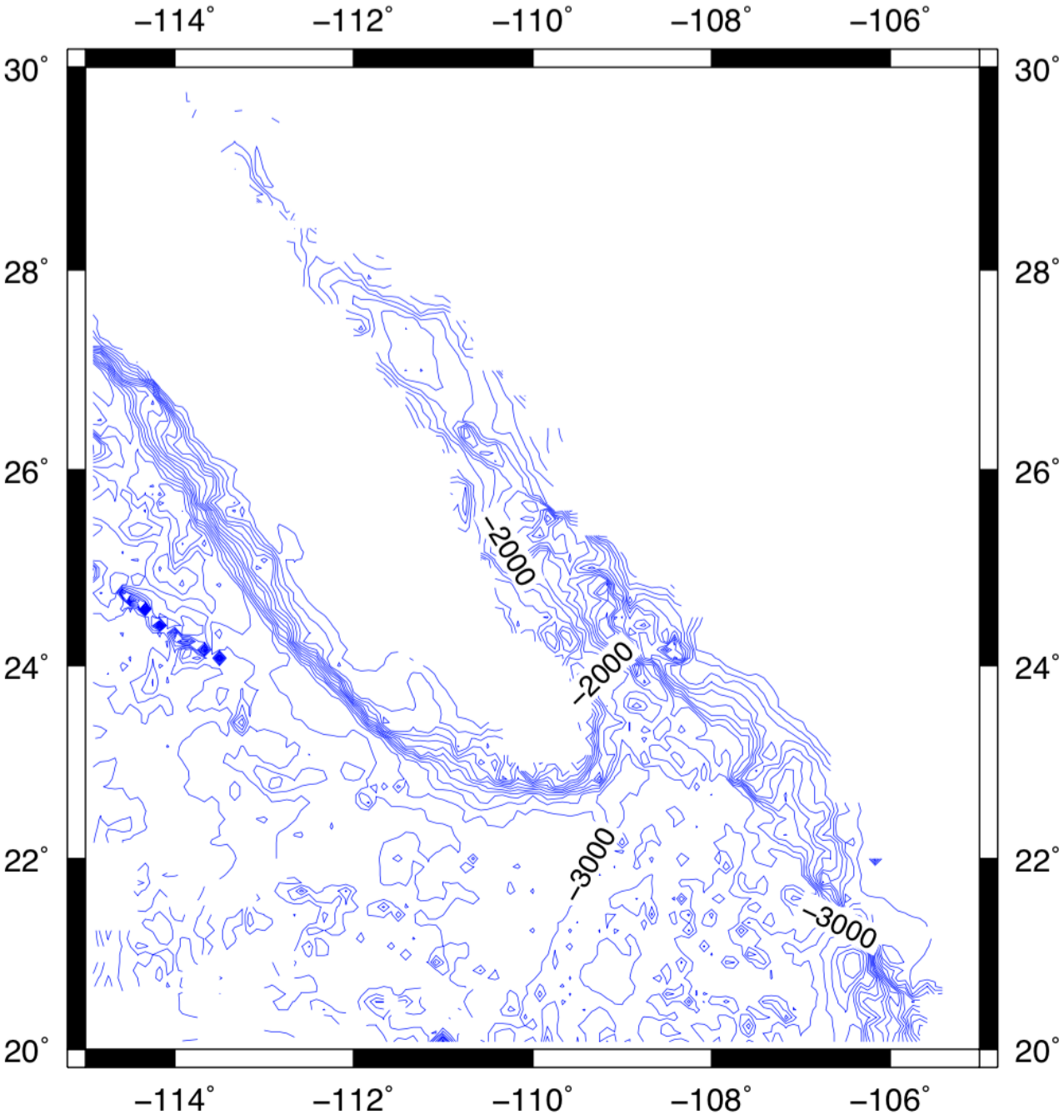
Exercise 24 – Nearneighbor

- Navigate to your tutorial directory
- Run minmax on ship.xyz to get region to nearest 5°
- Grid using nearneighbor
  - Select a 5 arc minute grid spacing
  - Specify 40 km search radius R
  - Use default sectors [4]

```
nearneighbor $region -I5m -Gship.nc
-S40k -V ship.xyz
```

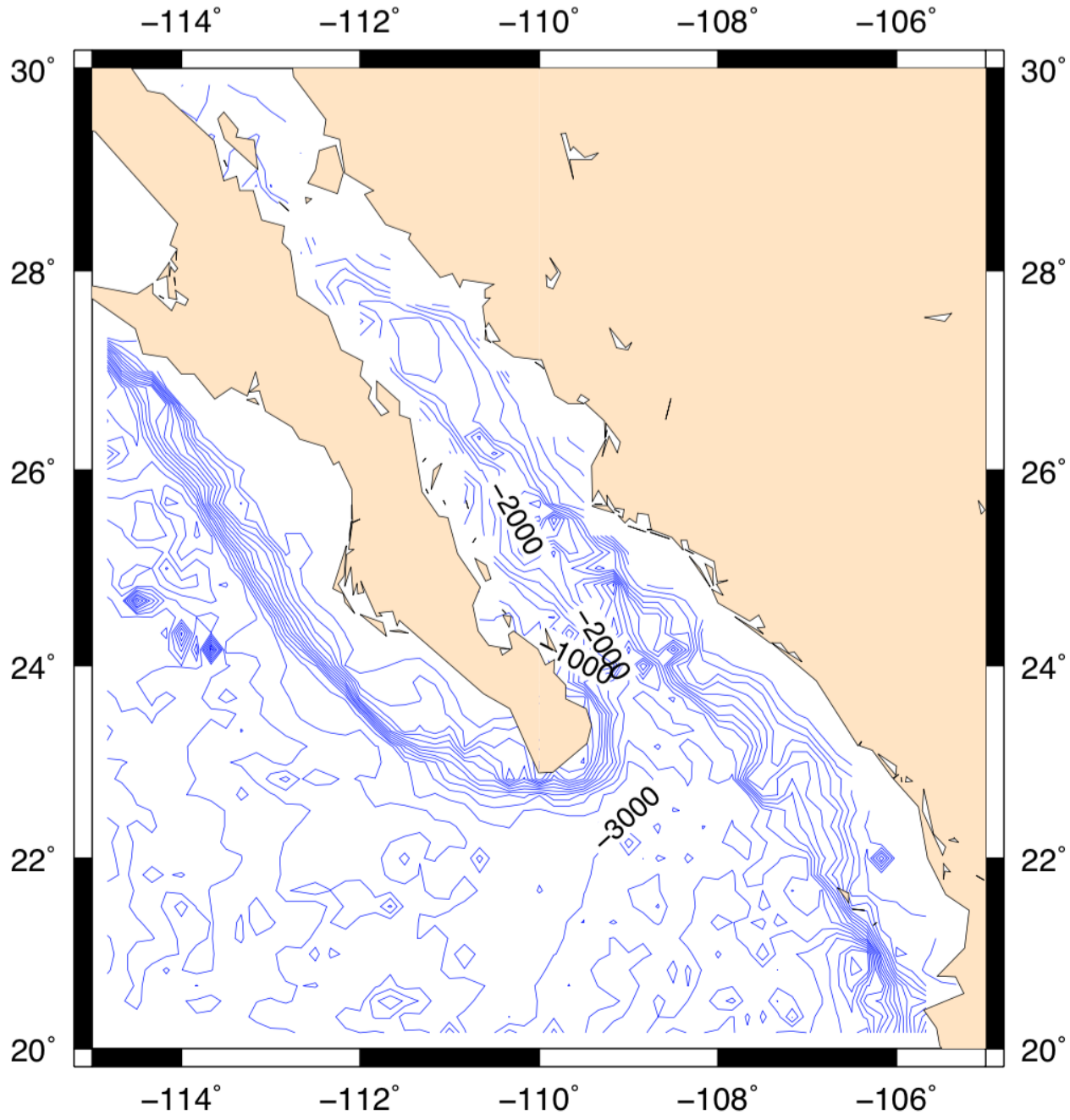
Nearneighbor Exercise 24 cont ..

- Make a Mercator contour map using the new data set:
  - contours every 250 m
  - annotations every 1 km
  - contours in blue
  - don't do any smoothing or filtering


−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
−2000
−2000
−3000
−3000

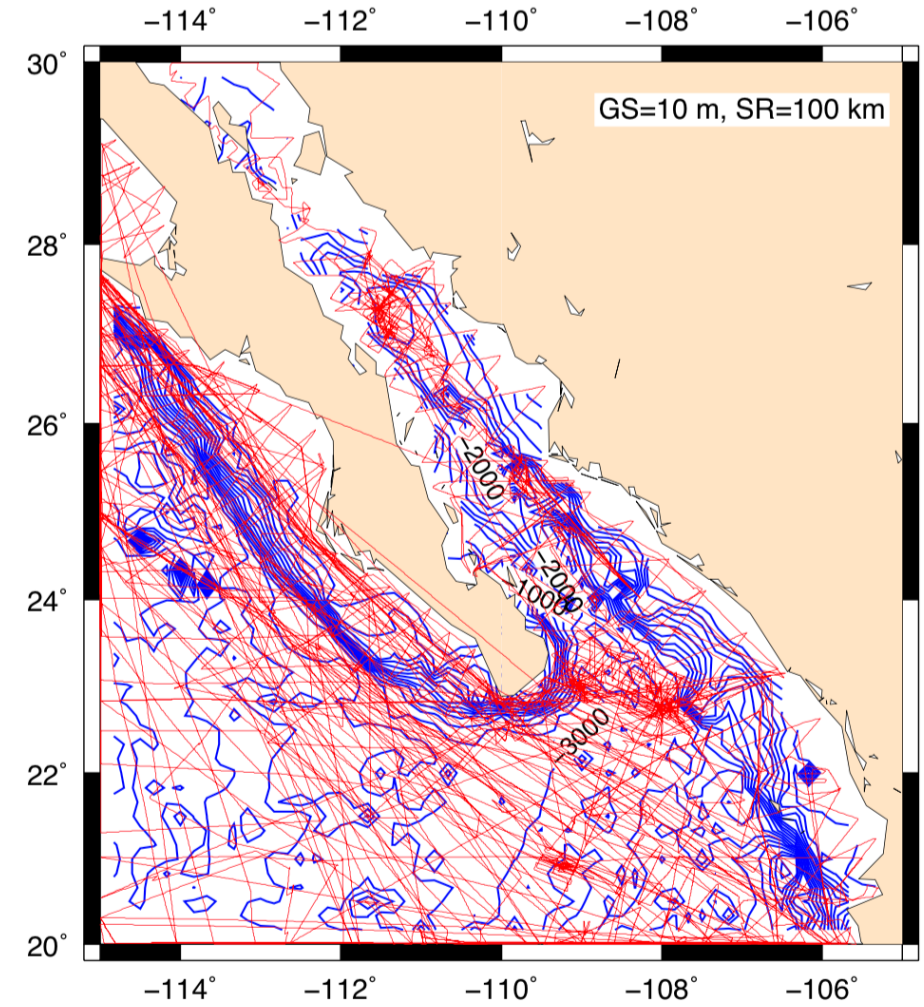
Nearneighbor Exercise cont ..

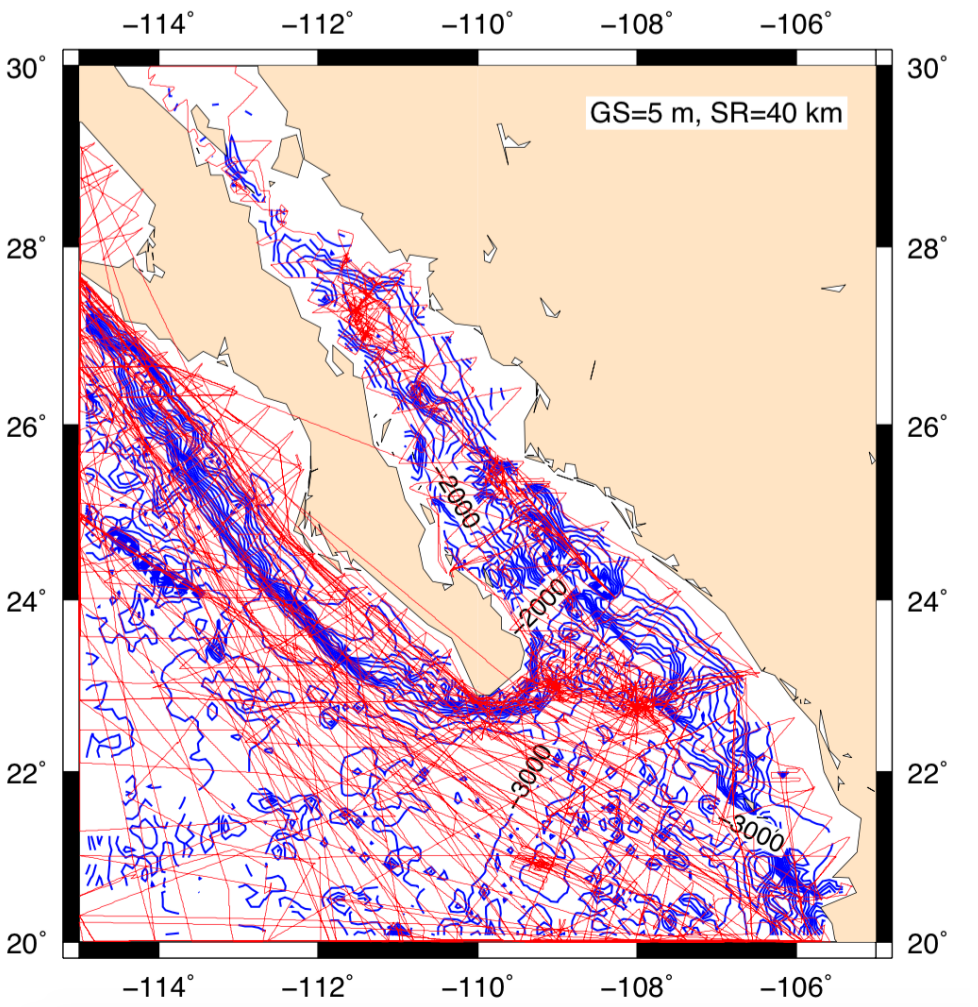
- Try a search radius of 100 km and a 10 minute grid spacing
- How do the plots differ?
- Use pscoast to plot coastlines.
- Where in the world are we?


−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
−2000
−2000
−1000
−3000

Interpreting Results

- In order to find out which gridding method works best, we have to know the spacing of our original data
- Use psxy to plot the data points on top of each of your interpolation grids
- Input file is ship.xyz


GS=10 m, SR=100 km
−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
−1000
−2000
−3000


GS=5 m, SR=40 km
−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
−2000
−3000

Gridding with Splines in Tension

- Physically, we force a thin elastic plate to go through all data points while pulling at the edges (tension).
- Takes **–R**w/e/s/n, **–I**dx/dy, and **–G**gridfile
- Three other options are relevant:

Option	Purpose
–Aaspect	Sets aspect ratio [1]
–Climit	Sets convergence limit [1/1000 of the actual data range
–Ttension	Sets the tension [0]

Preprocessing

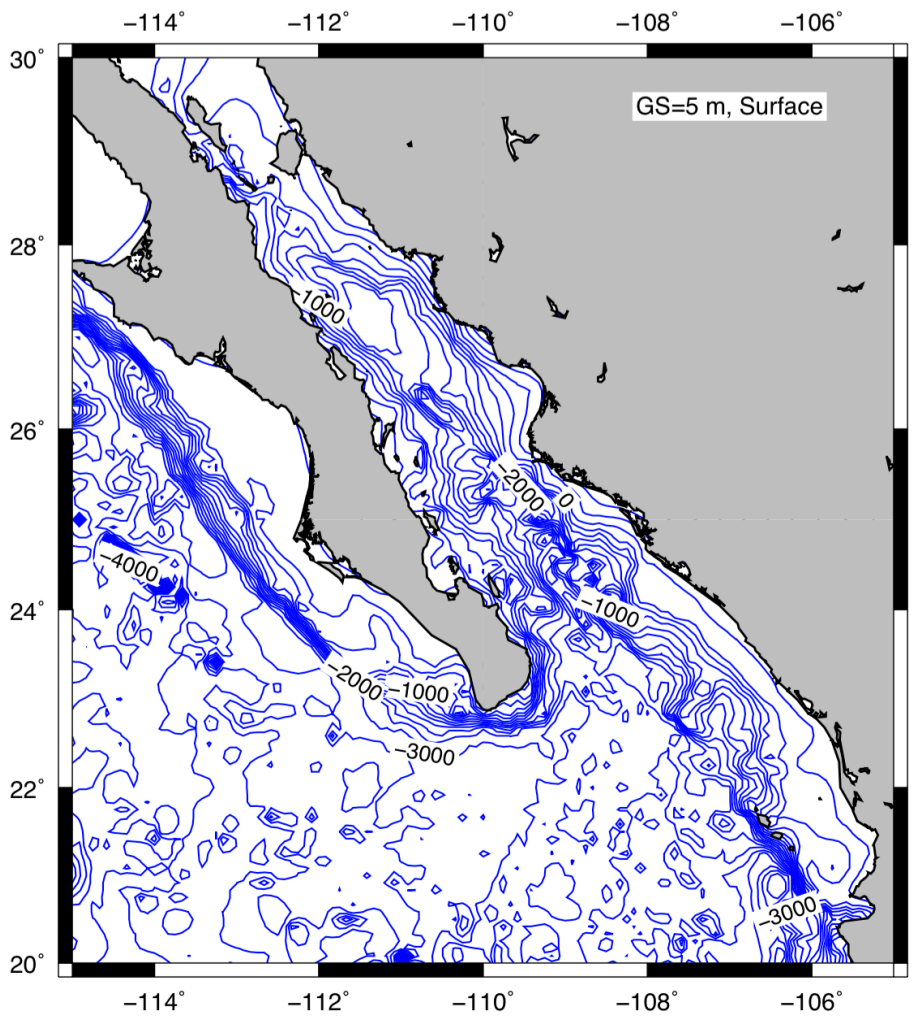
- surface needs either one or no data points per node; more will introduce aliasing
- preprocessing depends on data properties; we usually average using
  - means (blockmean)
  - medians (blockmedian)
  - modes (blockmode)
- Each program takes **–R**, **–I**
  - Use **–W** if there are data point weights
- Output has one or no data point per node

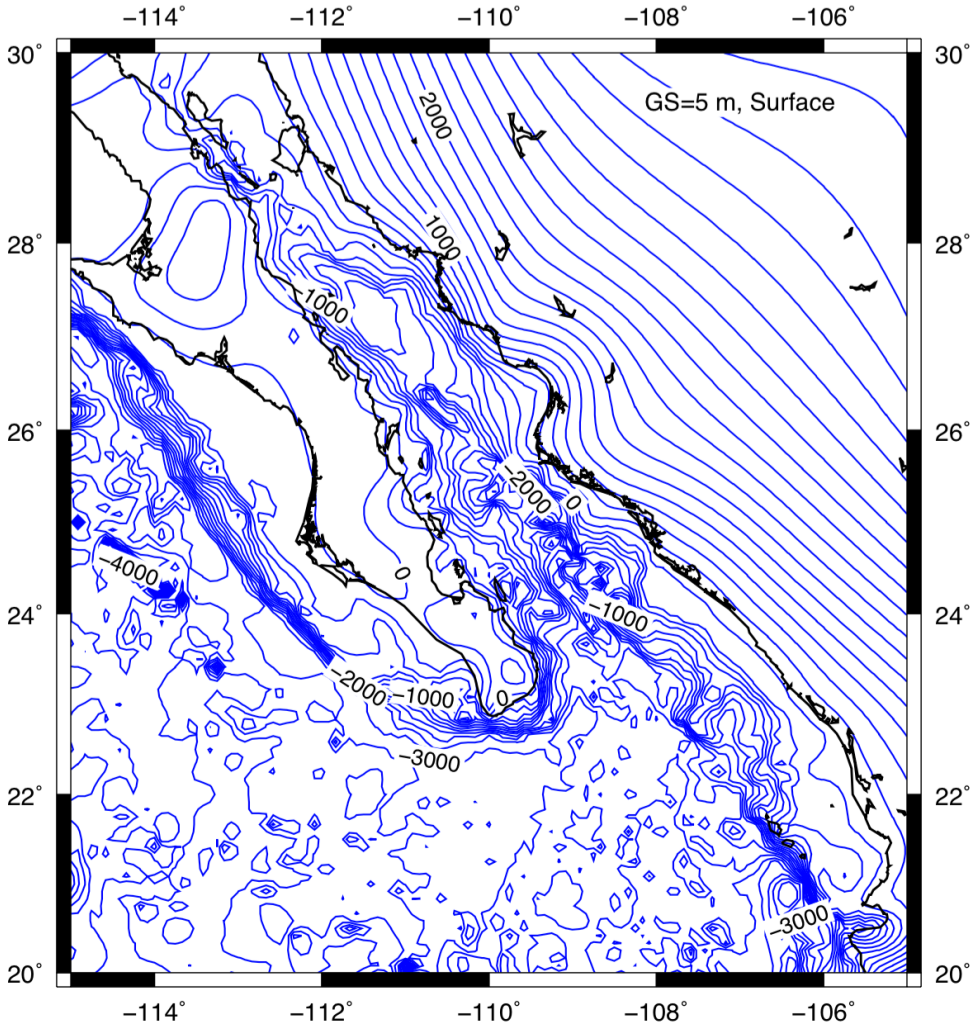
Map exercise 25

- Preprocess ship.xyz using medians, grid with surface, and repeat contouring exercise 24 but using the new dataset. Lay down light gray continents after contouring

```
blockmedian $region -I5m ship.xyz >
ship_5m.xyz
```

```
surface $region -I5m -Gship_s.nc -V
ship_5m.xyz -A0.9
```

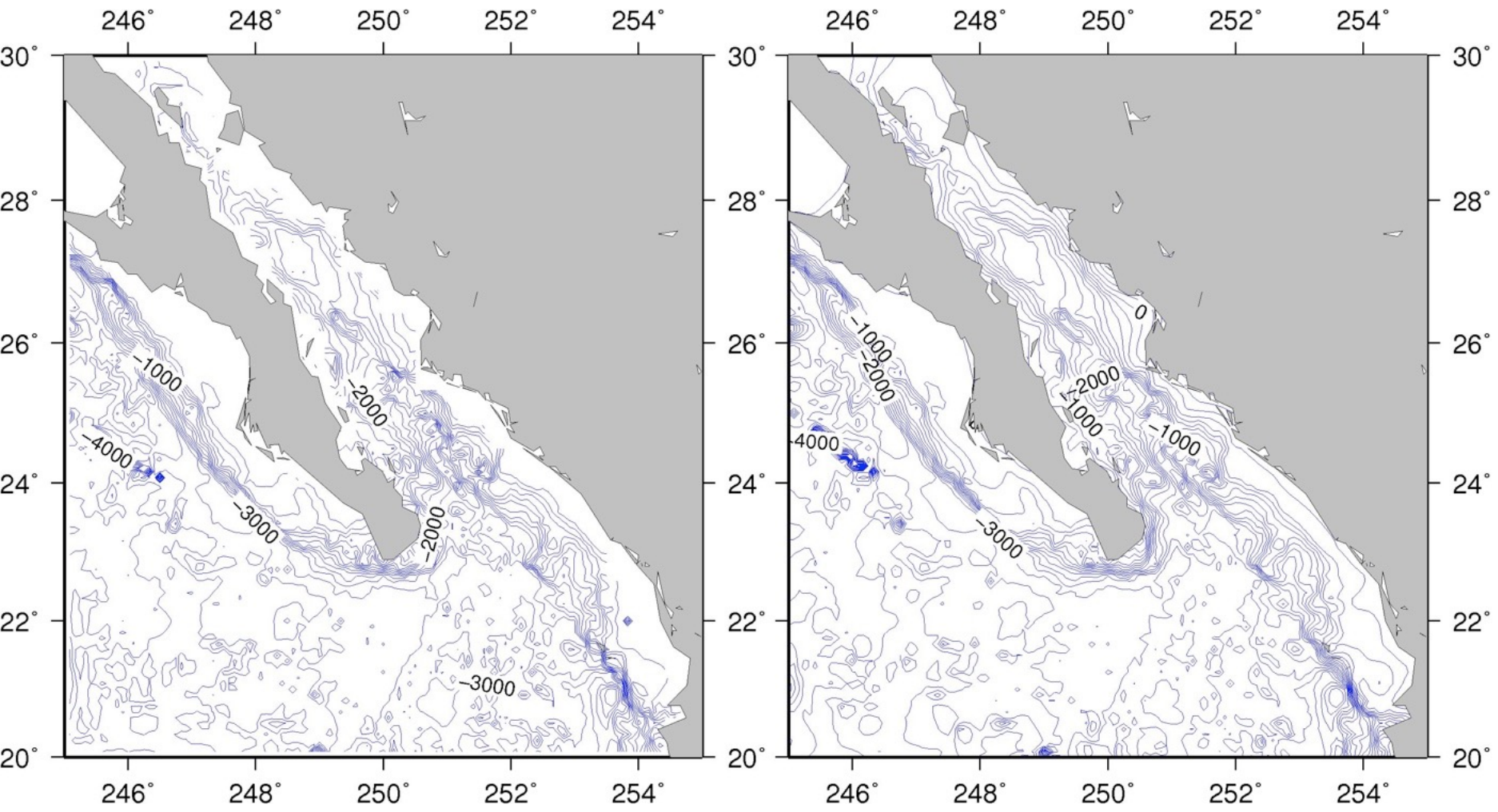

GS=5 m, Surface
−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
−1000
−2000
0
−4000
−1000
−2000
−1000
−3000
−3000


GS=5 m, Surface
−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
2000
1000
1000
−2000
0
−4000
0
−1000
−2000
−1000
0
−3000
−3000

Gridding comments

- Is there a difference between the grid made by nearneighbor and the one using surface?

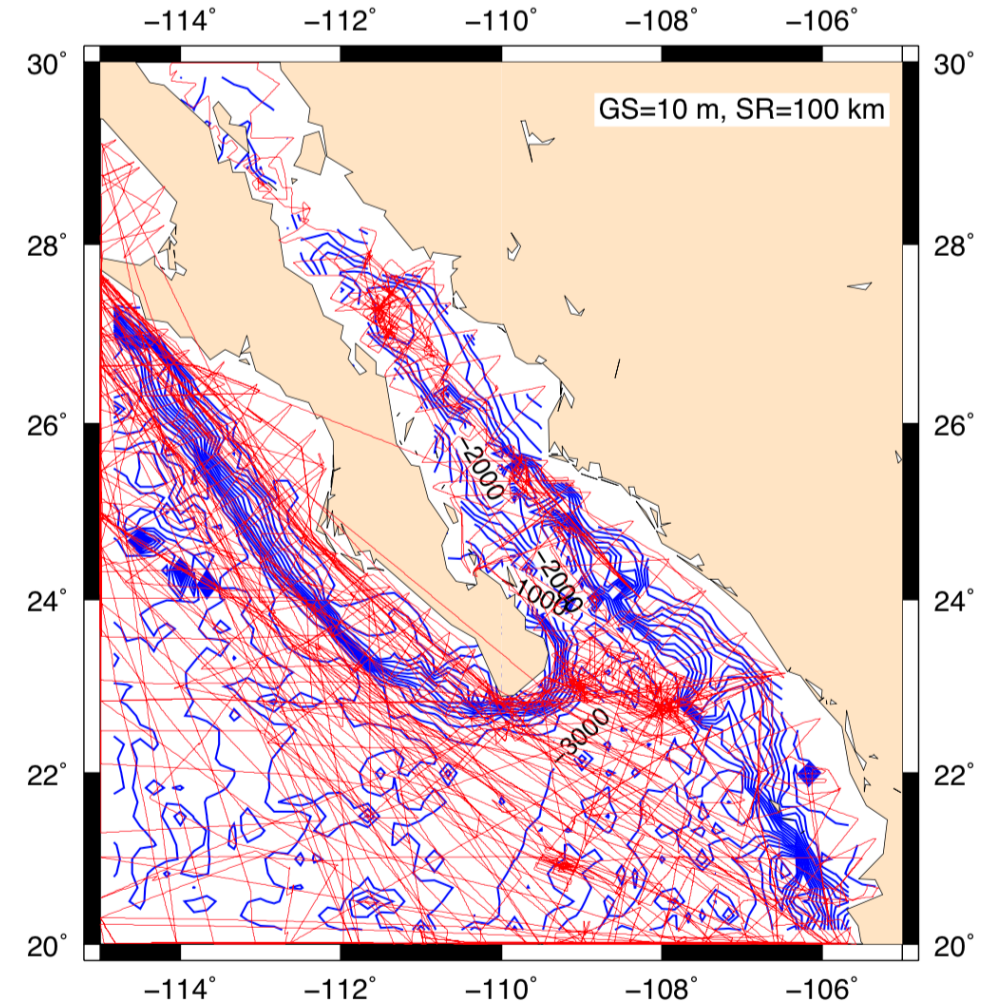
- Surface is a global gridding method and it will evaluate the solution at all nodes, even if there are no data constraints


246°
248°
250°
252°
254°
30°
28°
26°
24°
22°
20°
−1000
−2000
−3000
−4000
0

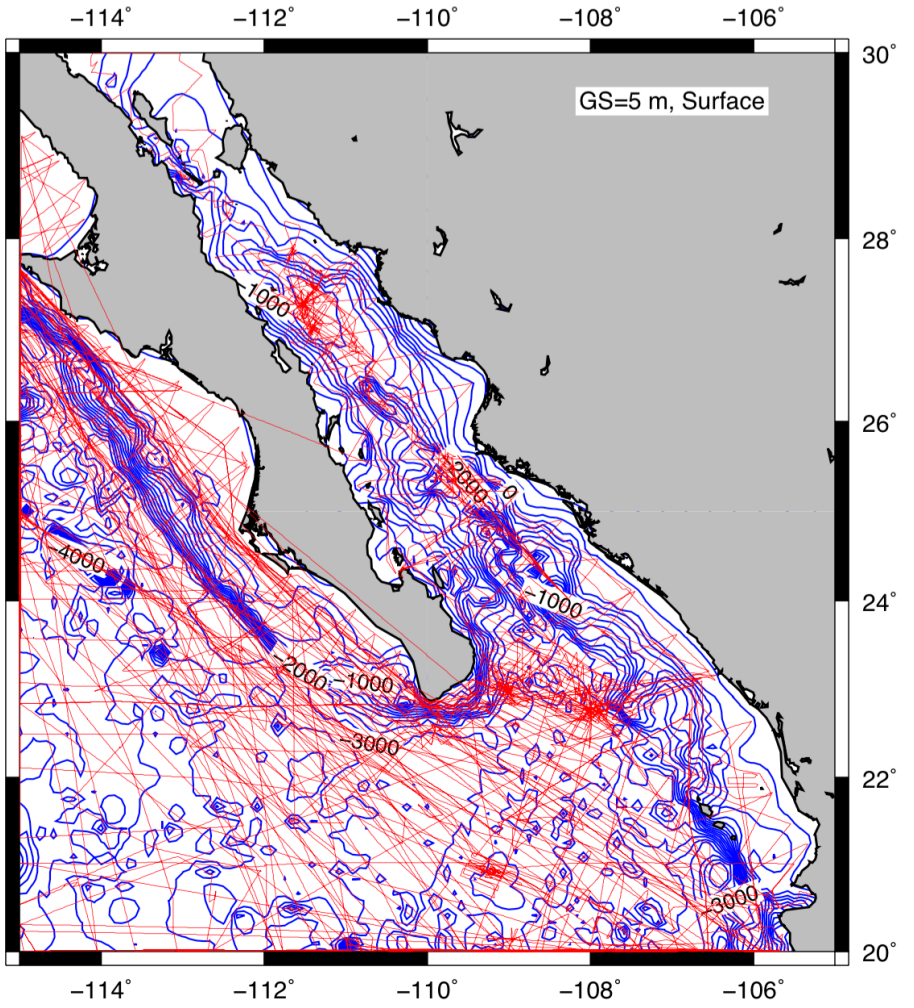
Interpreting Results

- In order to find out which gridding method works best, we have to know what the spacing of our original data was.
- Use psxy to plot the data points on top of each of your interpolation grids
- Input file is ship.xyz

NEARNEIGHBOR



SURFACE



Gridding comments

- To deal with unconstrained areas:
  - Reset nodes too far from data to NaN
    - grdmask – grdmath
  - Paint the unconstrained regions white
    - psmask
  - Plot land on top
    - pscoast
  - Use clip path so only constrained contours will appear.
    - psmask

Exercise 26

- Write a script based on the ship.xyz exercises that
  - Paints areas constrained by data light green (meaning areas within 50 km of data)
  - Overlay land as brown
  - Draw contours as before but this time label contours every 1000m and make them a shade of grey
  - Where is this area? Label notable features using pstext – label at least 2 things

Exercise 26 cont...

- Read the manual page for psmask, look at each option
- Set variables within your script for the following:
  - region
  - projection
  - width
  - gridspacing
  - searchradius
  - colour
  - infile
  - psfile

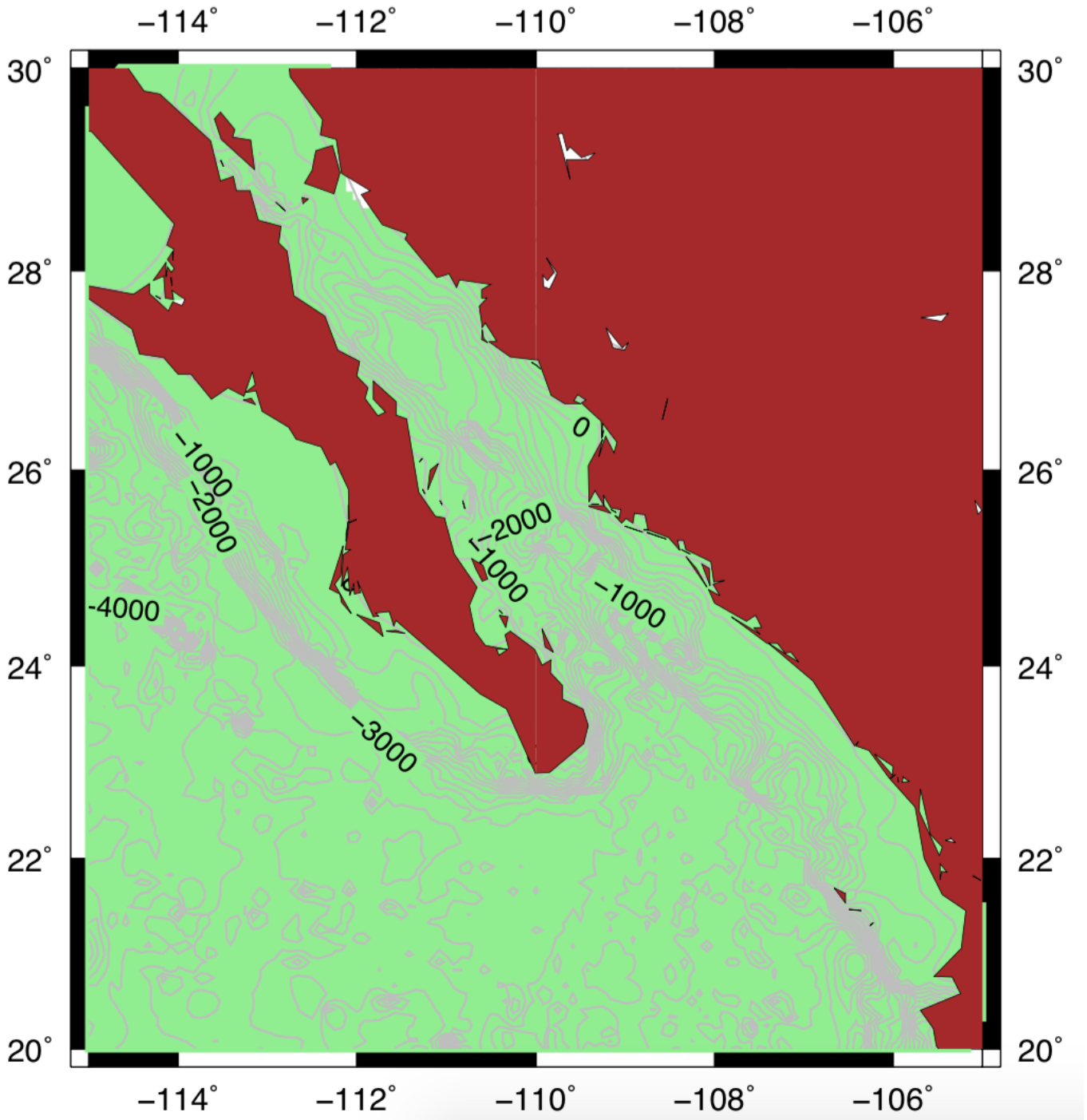
Exercise 26 cont …

- Use these as your first and second lines

```
psmask -R$region -I$gridspacing
$infile -J$projection$width -B2 -P -K
-V -S$searchradius -G$colour > $psfile

psmask -C -O >> $psfile
```

psmask -R245/255/20/30 ship.xyz -JM10 -B2 -P -V -S50k -Glightgreen -I5m > psfile.ps


−114°
−112°
−110°
−108°
−106°
30°
28°
26°
24°
22°
20°
0
−1000
−2000
−4000
−3000