Notes on using `surfacevel2strain.m` (and `sphereinterp.m`): A Matlab program to convert surface velocity fields into strain-rate maps

Carl Tape, Pablo Musé, Mark Simons

Last compiled: March 4, 2014

```
surfacevel2strain/USER_INFO/surfacevel2strain_manual.pdf -- this document
surfacevel2strain/USER_INFO/Tape2009gps.pdf              -- GJI2009 paper
surfacevel2strain/USER_INFO/Tape2009gps_supplement.pdf   -- supplemental notes
surfacevel2strain/matlab/                                -- matlab scripts
```

Please email Carl Tape (carltape@gi.alaska.edu) with suggestions or corrections for the code or these notes.

Contents

1 Introduction

The purpose of this program is to take an input set of discrete velocity vectors on the surface of a sphere, and then *estimate* a continuous velocity vector field on the surface. The estimated field is based on weighted and damped least squares, where the weights are determined from the errors associated with the measurements, but it does not yet use off-axis covariance values. The program considers 2- or 3-component velocity fields. Several real and synthetic examples were demonstrated in *Tape et al.* (2009).

We suggest testing and understanding `sphereinterp.m` (Section 2), a simpler version for 1D data, prior to using `surfacevel2strain.m` (Section 3). This version was used to make the estimated Moho map published in *Tape et al.* (2012).

2 Example with 1D data: `sphereinterp.m`

In order to demonstrate the basic features of `surfacevel2strain.m`, we will demonstrate the problem of estimating a continuous field on the sphere from a set of discrete 1D observations. The example data are Moho depths derived from receiver functions in southern California (*Yan and Clayton*, 2007), embedded within Moho depths from Crust2.0 (*Bassin et al.*, 2000). The data sets from these studies are available on-line.

The default example for `sphereinterp.m` should produce Figures 1–7, in addition to the text in Appendix A. We will review some of the key steps of the estimation below.

1. Obtain a set of observations and associated uncertainties (Figure 1).
2. Obtain the center-points for all basis functions (spherical wavelets) to be used in estimating the continuous field (Figure 3). A basis function is kept if a specified number of observation locations is within its spatial support.
3. Obtain a regularization parameter for the inverse problem (Figure 4).
4. Solve least-squares inverse problem for the model vector, which contains coefficients of basis functions. Then compute the estimated field, then analyze the residuals (Figures 5 and 6).
5. Using the same model vector, plot the estimated field on a uniform mesh (Figure 7). A fancier rendition of Figure 7 is shown in Figure 8.

3 Example with 2D or 3D data: `surfacevel2strain.m`

The general procedure for using `surfacevel2strain.m` is this:

1. Obtain a set of 2- or 3-component velocity field data: $\mathbf{v}(r_i, \theta_i, \phi_i)$, where i denotes the index of the observation. Save velocity field observations in a "standard" format.
2. Specify a latitude-longitude box for your region of interest. For example, for southern California, we use: `lonmin0 = -122.0 ; lonmax0 = -113.0 ; latmin0 = 30.0 ; latmax0 = 38.0` Observations outside this box will be excluded. Wavelet center-points outside this region may be used, depending on their scale q.
3. Specify user parameters in `surfacevel2strain.m` and compute the estimated continuous velocity field.
4. Plot output files in GMT.

3.1 Preparing the velocity field data set (`get_gps_dataset.m`)

The most general velocity field data set is one that contains the data covariances and standard errors. For a three-component field, the data covariance matrix will have six values per observation. Thus, for each new data set, I save a version in a "standard" format. In `get_gps_dataset.m`, the command

```
[dlon,dlat,ve,vn,vu,se,sn,su,ren,reu,rnu,start_date,finish_date,name] = read_gps_3D(filename);
```

will load the pre-saved GPS data set. Each user may have several different data sets, which in turn may be analyzed in a set of different regions, each described by a latitude-longitude box. The function `get_gps_dataset.m` is a template that should be changed for each user. The input and output variables are given by:

```
[dlon,dlat,vu,vs,ve,su,ss,se,ax1,slabel,stref] = get_gps_dataset(dir_data,ropt,dopt)
```

A few points regarding the input data files:

1. The data files can be synthetic or real (the estimation problem does not distinguish between the two).
2. The variables `start_date`, `finish_date`, and `name` are not used in `surfacevel2strain.m`.
3. For now, the program does not use the covariance terms `ren`, `reu`, `rnu`.
4. The data files should be as spatially complete as possible, because the user is given the option (`ropt`) to consider a subset region as a latitude-longitude "rectangular" patch.

3.2 Running `surfacevel2strain.m`

The default example for `surfacevel2strain.m` should produce Figures 9–14 (among many others), in addition to the text in Appendix B. The user is prompted to enter various parameters in the Matlab command window (Appendix B). The first run will threshold the basis functions, compute the design matrix for the inverse problem, and prompt the user for a regularization parameter. The user is prompted for the minimum scale wavelets, q_{min}, and maximum scale wavelets, q_{max}, for the estimation problem, in addition to the scale for the "secular field." The secular scale, $q_{\mathrm{min}} \leq q_{\mathrm{secular}} \leq q_{\mathrm{max}}$, represents the estimated secular field using scales $q = q_{\mathrm{min}}, \ldots, q_{\mathrm{secular}}$. It corresponds to a smooth "background" pattern that can be easily removed to analyze the details of the estimated field. The user can also estimate the velocity field using a single scale by entering $q_{\mathrm{min}} = q_{\mathrm{secular}} = q_{\mathrm{max}}$ (nor recommended). The user is also prompted to enter a minimum value for rotation magnitudes that will be used to threshold Euler poles for plotting purposes; a histogram is displayed to assist in this choice, but a value of zero can be entered as a default. In practical examples, the higher rotation magnitudes correspond to local apparent rotations from nearby poles, while lower rotation magnitudes may be artifacts associated with distant rotation poles.

The second run is primarily for plotting purposes and for analyzing different representations of the estimated field, such as

1. dilatation rate, strain rate, or rotation rate, computed directly from the estimated velocity field;
2. any quantity at cumulative or individual scales (q).

It is also used to adjust the mask for plotting purposes (*Tape et al.*, 2009).

Plotting scalar and vector fields in GMT

There are several `perl` scripts for plotting using GMT (*Wessel and Smith*, 1991) here:

`surfacevel2strain/gmt/util/`.

These scripts will require several modifications for your own purposes. The primary scripts to run are `plot_strain.pl` and `foursub.pl`. Example output figures are shown in Figures 15 and 16.

References

Bassin, C., G. Laske, and G. Masters (2000), The current limits of resolution for surface wave tomography in North America, in *Eos Trans. Am. Geophys. Un.*, vol. 81, p. F897.

Dong, D., P. Fang, Y. Bock, L. Prawirodirdjo, F. Webb, S. Kedar, and P. Lundgren (2011), Secular vertical crustal deformation field in California and Nevada regions from GPS data analysis: 1. Hydrological and anthropogenic signals, *J. Geophys. Res.* (in preparation).

Tape, C., P. Musé, M. Simons, D. Dong, and F. Webb (2009), Multiscale estimation of GPS velocity fields, *Geophys. J. Int.*, *179*, 945–971.

Tape, C., A. Plesch, J. H. Shaw, and H. Gilbert (2012), Estimating a continuous Moho surface for the California Unified Velocity Model, *Seis. Res. Lett.*, *83*(4), 728–735.

Wessel, P., and W. H. F. Smith (1991), Free software helps map and display data, *Eos Trans. Am. Geophys. Un.*, *72*(41), 441 ff.

Yan, Z., and R. W. Clayton (2007), Regional mapping of the crustal structure in southern California from receiver functions, *J. Geophys. Res.*, *112*, B05311, doi:10.1029/2006JB004622.

A Output from `sphereinterp.m`

Below is the output generated to the Matlab command window in the process of generating Figures 1–7 (Section 2).

```
 Type an index corresponding to a region (1=socal): 1
 Type an index corresponding to a dataset (1=moho): 1
getsubset.m: 124 points in the subset out of 124
------------------------------------------------------------
entering sphereinterp_grid.m to obtain spherical wavelet basis functions
Support of the spherical wavelets:
   q      deg          km
   0     82.442     9167.1
   1     47.310     5260.7
   2     24.707     2747.3
   3     12.500     1389.9
   4      6.268      697.0
   5      3.136      348.8
   6      1.569      174.4
   7      0.784       87.2
   8      0.392       43.6
   9      0.196       21.8
  10      0.103       11.4
  11      0.052        5.8
  12      0.026        2.9

minimum allowable grid order is 3
   1.39e+06 meters (support of q = 3 wavelet) < 1.94e+06 meters (2*Lscale)
getspheregrid.m: lon-lat subregion of sphere

GRID ORDER q = 3
  dbase = 7.929e+00 deg
```

```
  lonmin, lonmax, latmin, latmax:
  -145.79, -89.21, 6.21, 61.79
  phmin, phmax, thmin, thmax:
  -2.5445, -1.5570, 0.4924, 1.4624
Patch occupies this fraction of the sphere: 6.074e-02
This corresponds to a square patch with side length 50.057 deg
getting the gridpoints for q = 3
  number of total possible gridpoints : 642
  maximum number of subset gridpoints : 39
q = 3, nf = 8, dbase = 7.929e+00 deg
  actual number of subset gridpoints : 42

GRID ORDER q = 4
  dbase = 3.965e+00 deg
  lonmin, lonmax, latmin, latmax:
  -133.89, -101.11, 18.11, 49.89
  phmin, phmax, thmin, thmax:
  -2.3369, -1.7646, 0.7000, 1.2548
Patch occupies this fraction of the sphere: 2.068e-02
This corresponds to a square patch with side length 29.207 deg
getting the gridpoints for q = 4
  number of total possible gridpoints : 2562
  maximum number of subset gridpoints : 53
q = 4, nf = 16, dbase = 3.965e+00 deg
  actual number of subset gridpoints : 53

GRID ORDER q = 5
  dbase = 1.982e+00 deg
  lonmin, lonmax, latmin, latmax:
  -127.95, -107.05, 24.05, 43.95
  phmin, phmax, thmin, thmax:
  -2.2331, -1.8684, 0.8038, 1.1510
Patch occupies this fraction of the sphere: 8.312e-03
This corresponds to a square patch with side length 18.517 deg
getting the gridpoints for q = 5
  number of total possible gridpoints : 10242
  maximum number of subset gridpoints : 85
q = 5, nf = 32, dbase = 1.982e+00 deg
  actual number of subset gridpoints : 78

GRID ORDER q = 6
  dbase = 9.912e-01 deg
  lonmin, lonmax, latmin, latmax:
  -124.97, -110.03, 27.03, 40.97
  phmin, phmax, thmin, thmax:
  -2.1812, -1.9203, 0.8557, 1.0991
Patch occupies this fraction of the sphere: 4.179e-03
This corresponds to a square patch with side length 13.130 deg
getting the gridpoints for q = 6
  number of total possible gridpoints : 40962
  maximum number of subset gridpoints : 171
q = 6, nf = 64, dbase = 9.912e-01 deg
  actual number of subset gridpoints : 158

GRID ORDER q = 7
  dbase = 4.956e-01 deg
  lonmin, lonmax, latmin, latmax:
```

```
  -123.49, -111.51, 28.51, 39.49
  phmin, phmax, thmin, thmax:
  -2.1553, -1.9463, 0.8816, 1.0731
Patch occupies this fraction of the sphere: 2.636e-03
This corresponds to a square patch with side length 10.429 deg
getting the gridpoints for q = 7
  number of total possible gridpoints : 163842
  maximum number of subset gridpoints : 432
q = 7, nf = 128, dbase = 4.956e-01 deg
  actual number of subset gridpoints : 401

GRID ORDER q = 8
  dbase = 2.478e-01 deg
  lonmin, lonmax, latmin, latmax:
  -122.74, -112.26, 29.26, 38.74
  phmin, phmax, thmin, thmax:
  -2.1423, -1.9592, 0.8946, 1.0602
Patch occupies this fraction of the sphere: 1.997e-03
This corresponds to a square patch with side length 9.077 deg
getting the gridpoints for q = 8
  number of total possible gridpoints : 655362
  maximum number of subset gridpoints : 1309
q = 8, nf = 256, dbase = 2.478e-01 deg
  actual number of subset gridpoints : 1216

 threshold the gridpoints

    q   num    id    i1    i2
    3    12    12     1    12
    4    14    26    13    26
    5    24    50    27    50
    6    45    95    51    95
    7    77   172    96   172
    8   102   274   173   274

GRIDPOINTS q = 3 to 8 (274):
274 wavelets / 1948 total with >= 3 stations inside their corresponding spatial supports

 q = 3 to 8, j = 1 to 274 (274)
 q = 3 to 6, j = 1 to 95 (95)
 q = 7 to 7, j = 96 to 172 (77)
 q = 8 to 8, j = 173 to 274 (102)
------------------------------------------------------------
entering sphereinterp_est.m to estimate smooth scalar field on the sphere
no input polygon provided --> full plotting grid will be used
choice of regularization parameter:
ordinary cross validation
input uncertainties will be used within inversion

Constructing the design matrix...
 creating the L-curve...
 ii = 1/40, lam = 0.001
 ii = 2/40, lam = 0.0017013
   :
 ii = 37/40, lam = 203091.7621
 ii = 38/40, lam = 345510.7295
 ii = 39/40, lam = 587801.6072
```

```
 ii = 40/40, lam = 1000000

Pick the regularization parameter:
L-curve lambda = 7.017e-02 (index 9)
    OCV lambda = 3.455e-01 (index 12)
    GCV lambda = 2.031e-01 (index 11)
your pick lam0 = 3.455e-01 (index 12)
 computing the model vector...

 computing values at the plotting points...
100 out of 2200
  :
2200 out of 2200
Elapsed time is 15.812080 seconds.

Number of observations, ndata = 124
Number of basis functions, ngrid = 274
For testing purposes, try decreasing one of these:
  qmax = 8, the densest grid for basis functions
  nx = 50, the grid density for plotting
  ndata = 124, the number of observations (or ax0)
>>
```

B Output from `surfacevel2strain.m`

Part 1: Estimating the model vector of spherical wavelet coefficients

The full output (and input) is shown below. This will also generate Figures 9–11, among many more (Section 3). For basic testing, specify 0 for "write output files for GMT plotting".

```
 Type 1 for new inversion or 0 otherwise: 1
 Type 1 to use spherical wavelets, 2 for spherical splines: 1
 Type 1 to use the DIAGONAL covariance matrix for weighting (0 otherwise): 1
 Type the number of components of the v-field for the inversion (2 or 3) : 2
 Type 1 to plot with the mask (0 otherwise) : 1
 Type 1 to write output to files for GMT plotting (0 otherwise) : 0
 Type an index corresponding to a region (1=us, 2=cal, 3=socal, ..., 8=parkfield): 3
 Type an index corresponding to a v-field dataset (1=REASON): 1
read_gps_3D.m: /home/carltape/compearth/surfacevel2strain/data/examples/reason_subset_3D.dat
getsubset.m: 408 points in the subset out of 510
tp2xyz.m: uniform radial value
 Type 1 to remove a uniform rotation, 0 otherwise: 1
tp2xyz.m: uniform radial value
Support of the spherical wavelets:
   q      deg          km
   0     82.442     9167.1
   1     47.310     5260.7
   2     24.707     2747.3
   3     12.500     1389.9
   4      6.268      697.0
   5      3.136      348.8
   6      1.569      174.4
   7      0.784       87.2
   8      0.392       43.6
```

```
   9      0.196      21.8
  10      0.103      11.4
  11      0.052       5.8
  12      0.026       2.9

minimum allowable grid order is 3
   1.39e+06 meters (support of q = 3 wavelet) < 1.94e+06 meters (2*Lscale)
 Type min allowable grid order, qmin >= 0 (try 3): 3
 Type max allowable grid order, qmax: 7
getspheregrid.m: lon-lat subregion of sphere

GRID ORDER q = 3
  dbase = 7.929e+00 deg
  lonmin, lonmax, latmin, latmax:
  -145.79, -89.21, 6.21, 61.79
  phmin, phmax, thmin, thmax:
  -2.5445, -1.5570, 0.4924, 1.4624
Patch occupies this fraction of the sphere: 6.074e-02
This corresponds to a square patch with side length 50.057 deg
getting the gridpoints for q = 3
  number of total possible gridpoints : 642
  maximum number of subset gridpoints : 39
q = 3, nf = 8, dbase = 7.929e+00 deg
  actual number of subset gridpoints : 42

GRID ORDER q = 4
  dbase = 3.965e+00 deg
  lonmin, lonmax, latmin, latmax:
  -133.89, -101.11, 18.11, 49.89
  phmin, phmax, thmin, thmax:
  -2.3369, -1.7646, 0.7000, 1.2548
Patch occupies this fraction of the sphere: 2.068e-02
This corresponds to a square patch with side length 29.207 deg
getting the gridpoints for q = 4
  number of total possible gridpoints : 2562
  maximum number of subset gridpoints : 53
q = 4, nf = 16, dbase = 3.965e+00 deg
  actual number of subset gridpoints : 53

GRID ORDER q = 5
  dbase = 1.982e+00 deg
  lonmin, lonmax, latmin, latmax:
  -127.95, -107.05, 24.05, 43.95
  phmin, phmax, thmin, thmax:
  -2.2331, -1.8684, 0.8038, 1.1510
Patch occupies this fraction of the sphere: 8.312e-03
This corresponds to a square patch with side length 18.517 deg
getting the gridpoints for q = 5
  number of total possible gridpoints : 10242
  maximum number of subset gridpoints : 85
q = 5, nf = 32, dbase = 1.982e+00 deg
  actual number of subset gridpoints : 78

GRID ORDER q = 6
  dbase = 9.912e-01 deg
  lonmin, lonmax, latmin, latmax:
  -124.97, -110.03, 27.03, 40.97
```

```
  phmin, phmax, thmin, thmax:
  -2.1812, -1.9203, 0.8557, 1.0991
Patch occupies this fraction of the sphere: 4.179e-03
This corresponds to a square patch with side length 13.130 deg
getting the gridpoints for q = 6
  number of total possible gridpoints : 40962
  maximum number of subset gridpoints : 171
q = 6, nf = 64, dbase = 9.912e-01 deg
  actual number of subset gridpoints : 158

GRID ORDER q = 7
  dbase = 4.956e-01 deg
  lonmin, lonmax, latmin, latmax:
  -123.49, -111.51, 28.51, 39.49
  phmin, phmax, thmin, thmax:
  -2.1553, -1.9463, 0.8816, 1.0731
Patch occupies this fraction of the sphere: 2.636e-03
This corresponds to a square patch with side length 10.429 deg
getting the gridpoints for q = 7
  number of total possible gridpoints : 163842
  maximum number of subset gridpoints : 432
q = 7, nf = 128, dbase = 4.956e-01 deg
  actual number of subset gridpoints : 401

 threshold the gridpoints

     q   num    id    i1    i2
     3    11    11     1    11
     4    15    26    12    26
     5    25    51    27    51
     6    52   103    52   103
     7   124   227   104   227
 Enter max q grid for secular field (3 <= qsec <= 7): 5

Thresholding GRIDPOINTS q = 3 to 7 (227):
227 wavelets / 732 total with >= 3 stations inside their corresponding spatial supports

 q = 3 to 7, j = 1 to 227 (227)
 q = 3 to 5, j = 1 to 51 (51)
 q = 6 to 6, j = 52 to 103 (52)
 q = 7 to 7, j = 104 to 227 (124)

Constructing the design matrix...
regularization curves for scalar field vsouth
 ii = 1/40, lam = 0.001
 ii = 2/40, lam = 0.0017013
   :
   :
 ii = 39/40, lam = 587801.6072
 ii = 40/40, lam = 1000000
L-curve lambda = 4.924e+03 (index 30)
    OCV lambda = 1.701e+03 (index 28)
    GCV lambda = 2.424e-02 (index 7)
Type an index for lambda (try iOCV = 28): 28
regularization curves for scalar field veast
 ii = 1/40, lam = 0.001
 ii = 2/40, lam = 0.0017013
```

```
   :
   :
 ii = 38/40, lam = 345510.7295
 ii = 39/40, lam = 587801.6072
 ii = 40/40, lam = 1000000
L-curve lambda = 4.924e+03 (index 30)
    OCV lambda = 1.701e+03 (index 28)
    GCV lambda = 7.017e+01 (index 22)
Type an index for lambda (try iOCV = 28): 28

 got the regularization parameters (vr, vth, vphi):
    lam0 = NaN 1.70e+03 1.70e+03
>>
```

Part 2: Plotting and analyzing results

At this point, the design matrix has been constructed and the regularization parameter has been selected. To avoid recomputing these quantities, the user is given the option to plot and analyze results while changing additional parameters. So the next stage is to *re-run* `surfacevel2strain.m`. The input choices are now

```
 Type 1 for new inversion or 0 otherwise: 0
 Enter minimum value of omega for euler vectors (default = 0 rad/yr): 1.5e-7
```

Below is the output generated to the Matlab command window in the process of generating Figures 12–14, among many more (Section 3).

```
 Type 1 for new inversion or 0 otherwise: 0

 computing the model vector...
surfacevel2strain_figs.m: plotting with ifigs1==1

Constructing the base design matrix for plotting...
 computing a mask for plotting...
3918 out of 8800 plotting points are unmasked
 compute L : assume isotropic linear elasticity, Poisson solid, and surface condition
surfacevel2strain_evec.m: computing euler vectors...
 Enter minimum value of omega for euler vectors (default = 0 rad/yr): 1.5e-7
tp2xyz.m: uniform radial value
tp2xyz.m: uniform radial value
 computing the max eigenvalue at each point...
surfacevel2strain_figs.m: plotting with ifigs2==1
 plotting velocities at different scales ...
 plotting strain maps...
    1.0000    0.0246
    2.0000    0.0269
    3.0000    1.6246
    4.0000    1.9438
    5.0000    1.3046
    6.0000    1.8954
 plotting components of the scalar quantity STRAIN...
    1.0000    0.0000
    2.0000    0.0000
    3.0000    0.0000
```

```
    5.0000    0.0000
    6.0000    0.0000
1  2.9750e-07
2  3.1625e-07
3  1.2570e-07
4  1.3862e-07
5  2.4027e-07
6  4.2046e-07
1  -1.2570e-07
2  4.2046e-07
3  2.4027e-07
4  3.2916e-07
exiting surfacevel2strain_figs.m
DONE with surfacevel2strain.m
>>
```

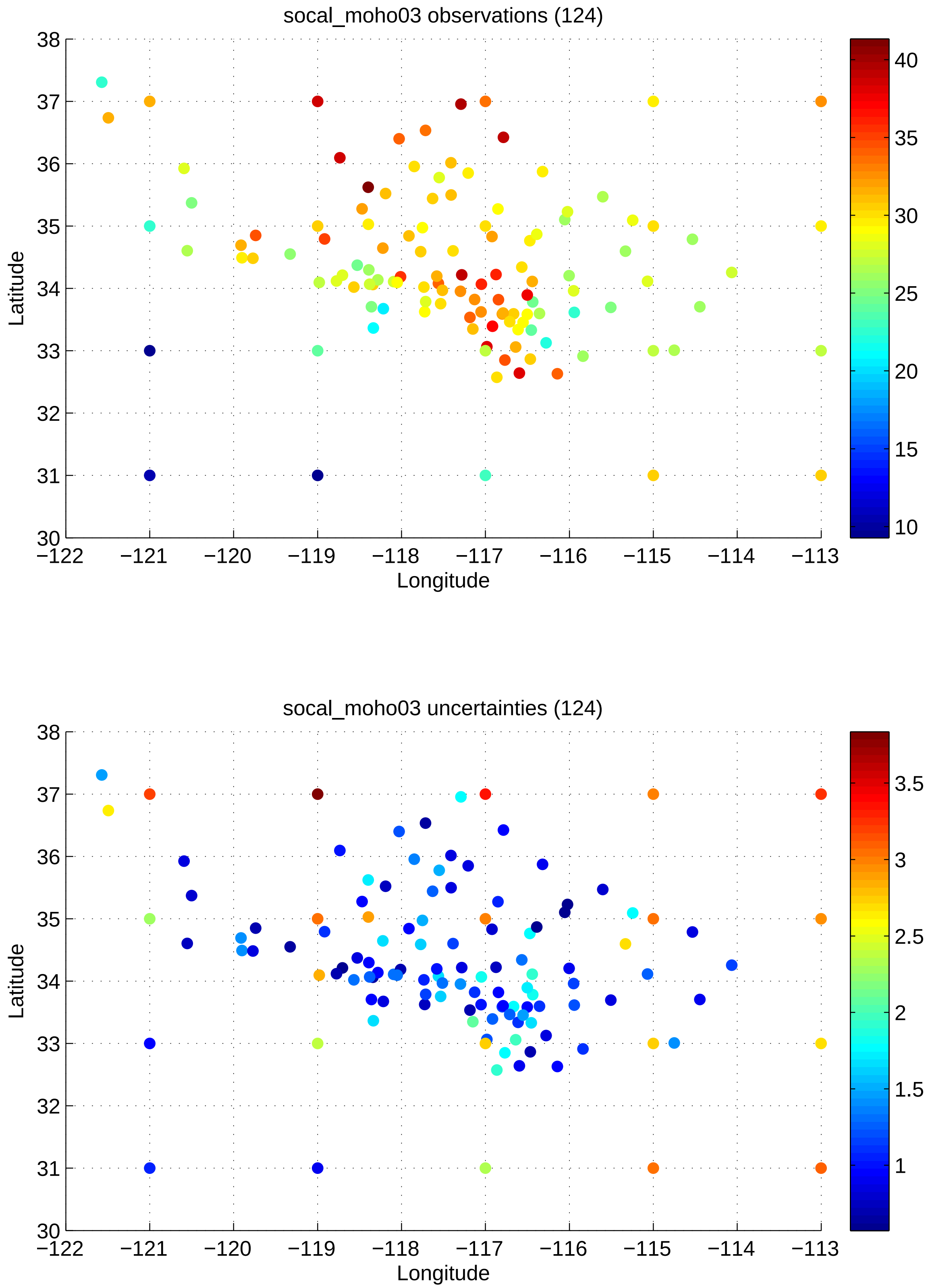


Figure 1: Example 1D data set for estimating a continuous function on the sphere (Section 2). The data set is comprised of Moho depth estimates at discrete locations. The top plot is for the Moho depth, and the bottom plot is for the estimated uncertainty associated with each depth. Values are from *Yan and Clayton* (2007) and *Bassin et al.* (2000).

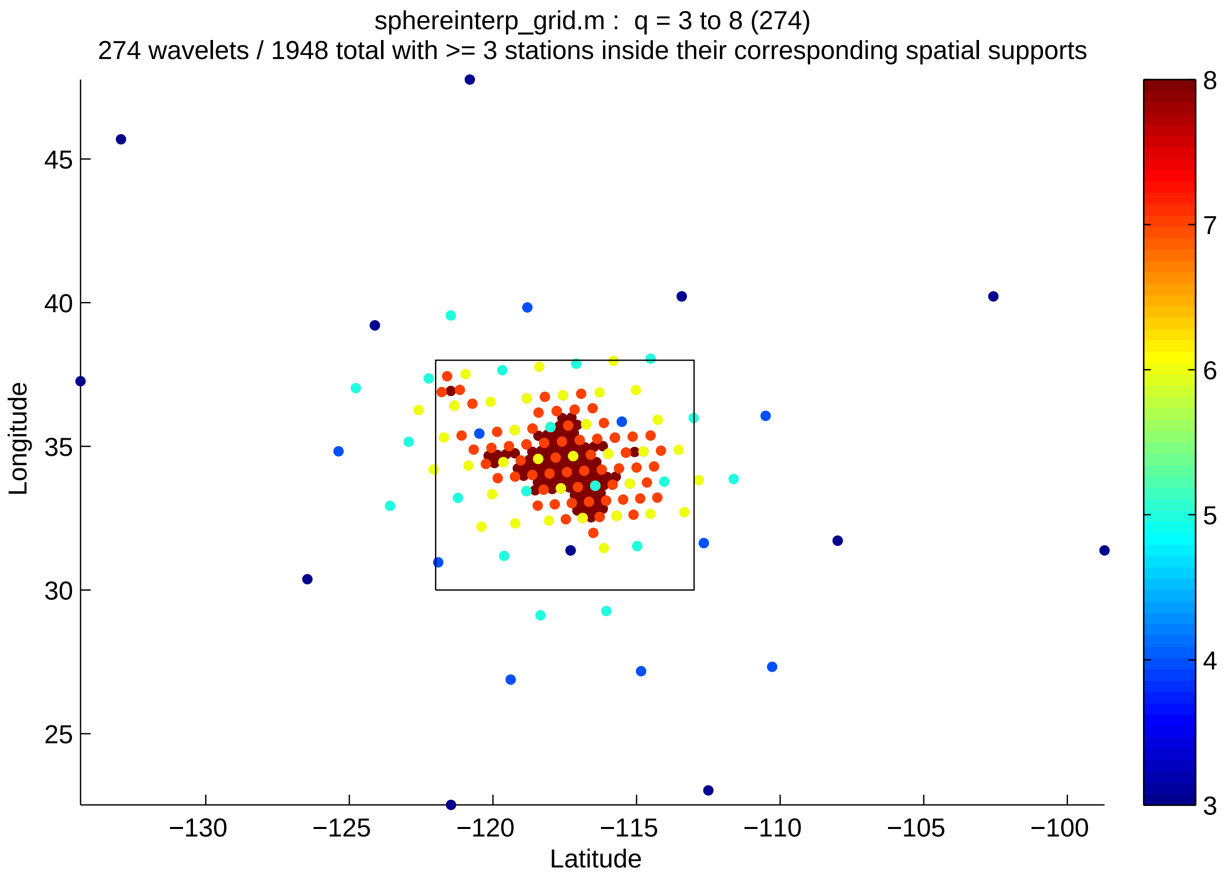


Figure 2: Full set of center-points for basis functions used in the inversion (Section 2). Color corresponds to the scale q for each basis function. Note that some dots correspond to multiple basis functions with different scales q, but only the highest-q is visible. See partition by scale in Figure 3.

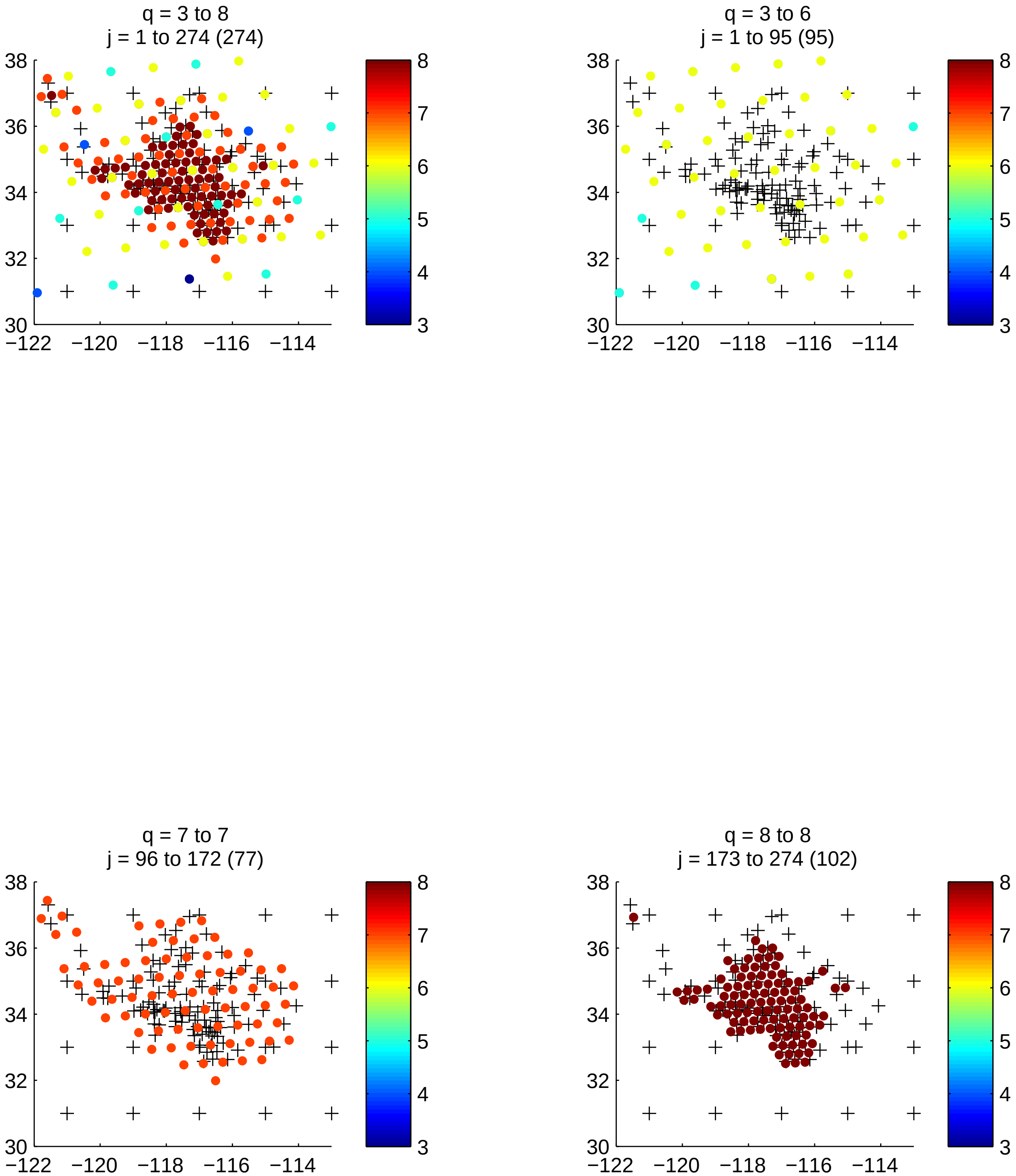


Figure 3: Basis function gridpoint centers for grids q=3-8 (Figure 2), q=3-6, q=7, and q=8. Color corresponds to the scale q for each basis function. Observation locations are plotted at '+' markers (Figure 1).

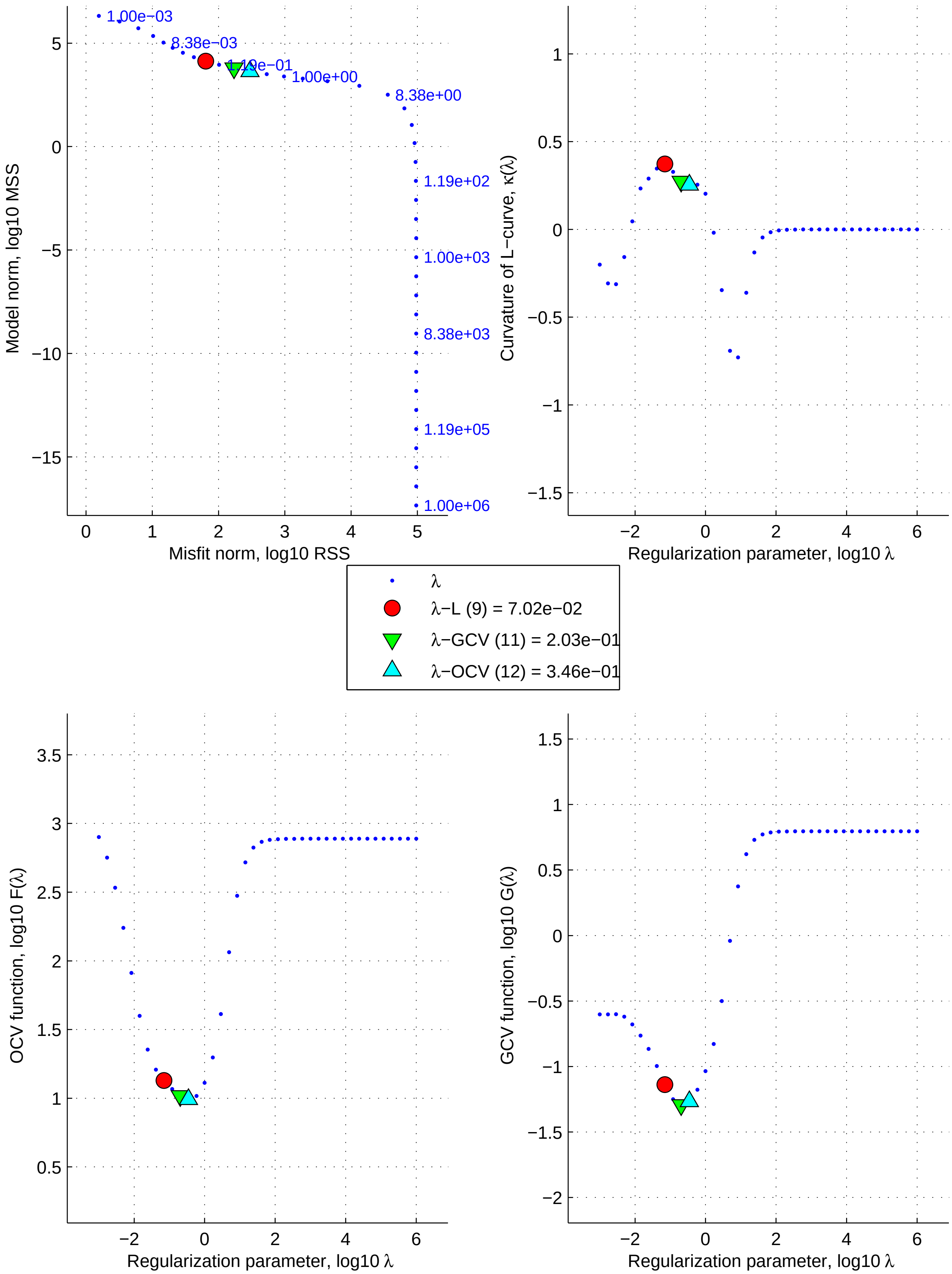


Figure 4: Selection of regularization parameter, λ, considering three different approaches: L-curve, ordinary cross-validation, general cross-validation. See *Tape et al.* (2009) for details and references. For this example, all three techniques select a similar regularization parameter.

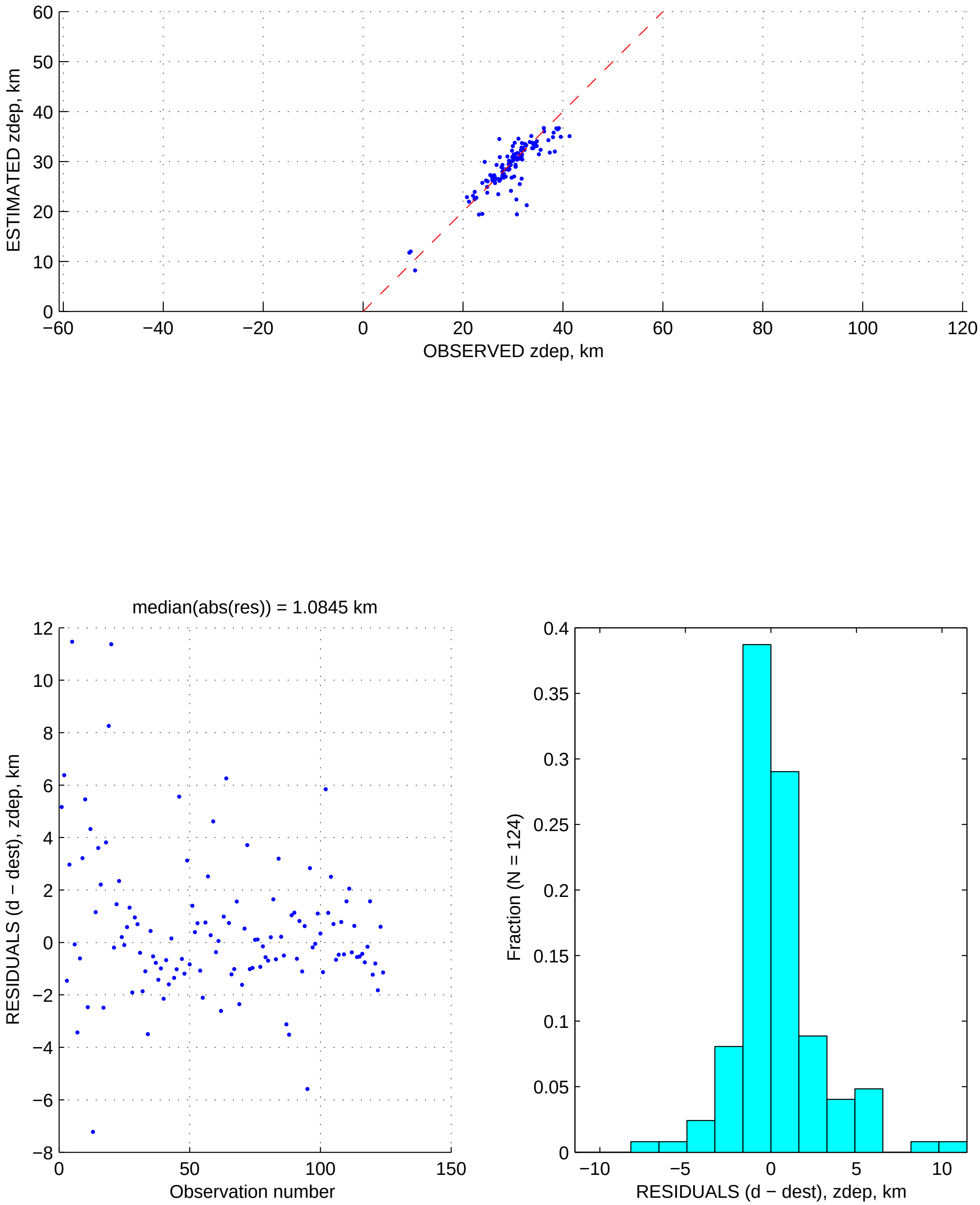


Figure 5: Comparison between observations and predictions for the 124 Moho depths.

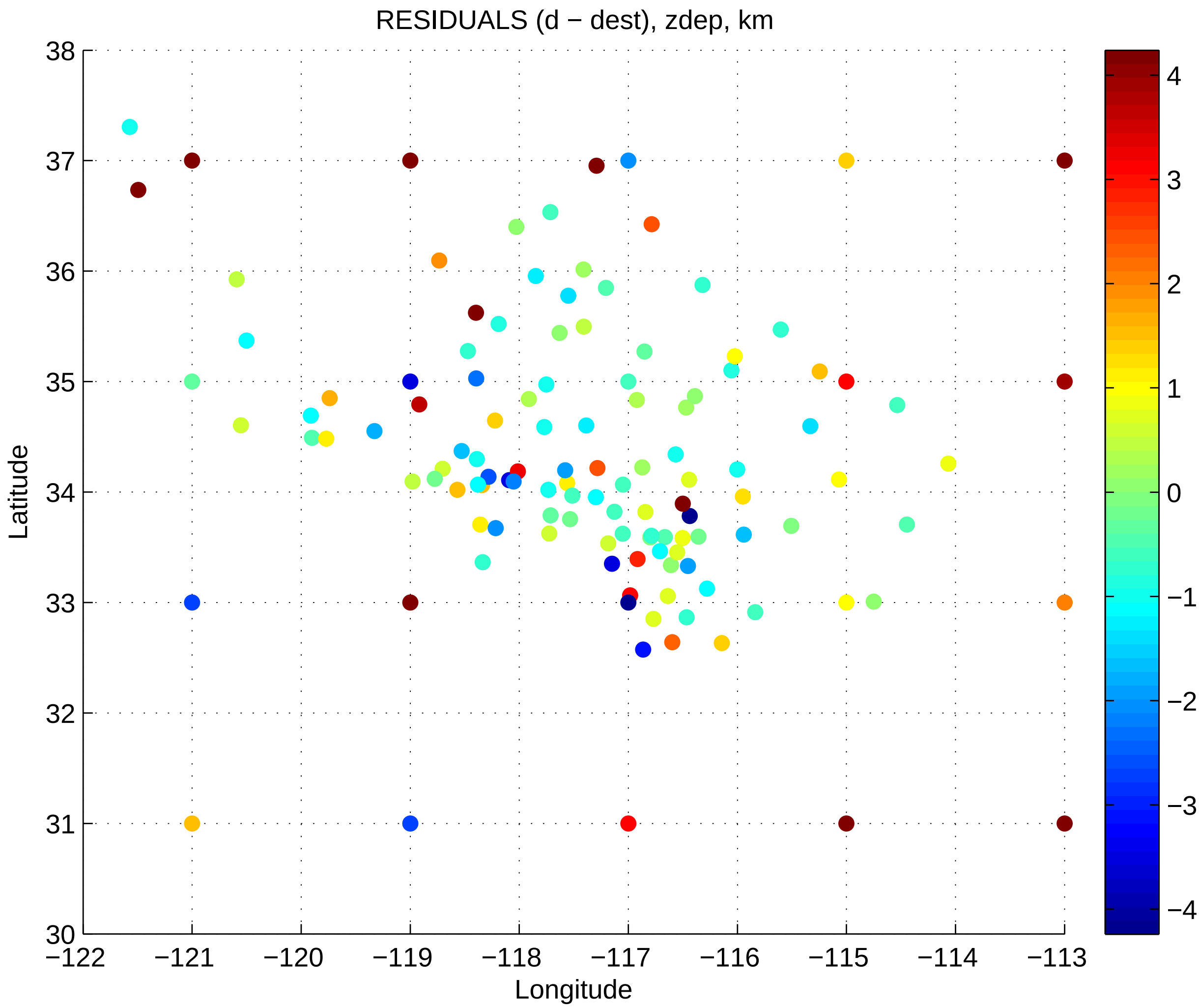


Figure 6: Spatial plot of residuals between observed and estimated Moho depths.

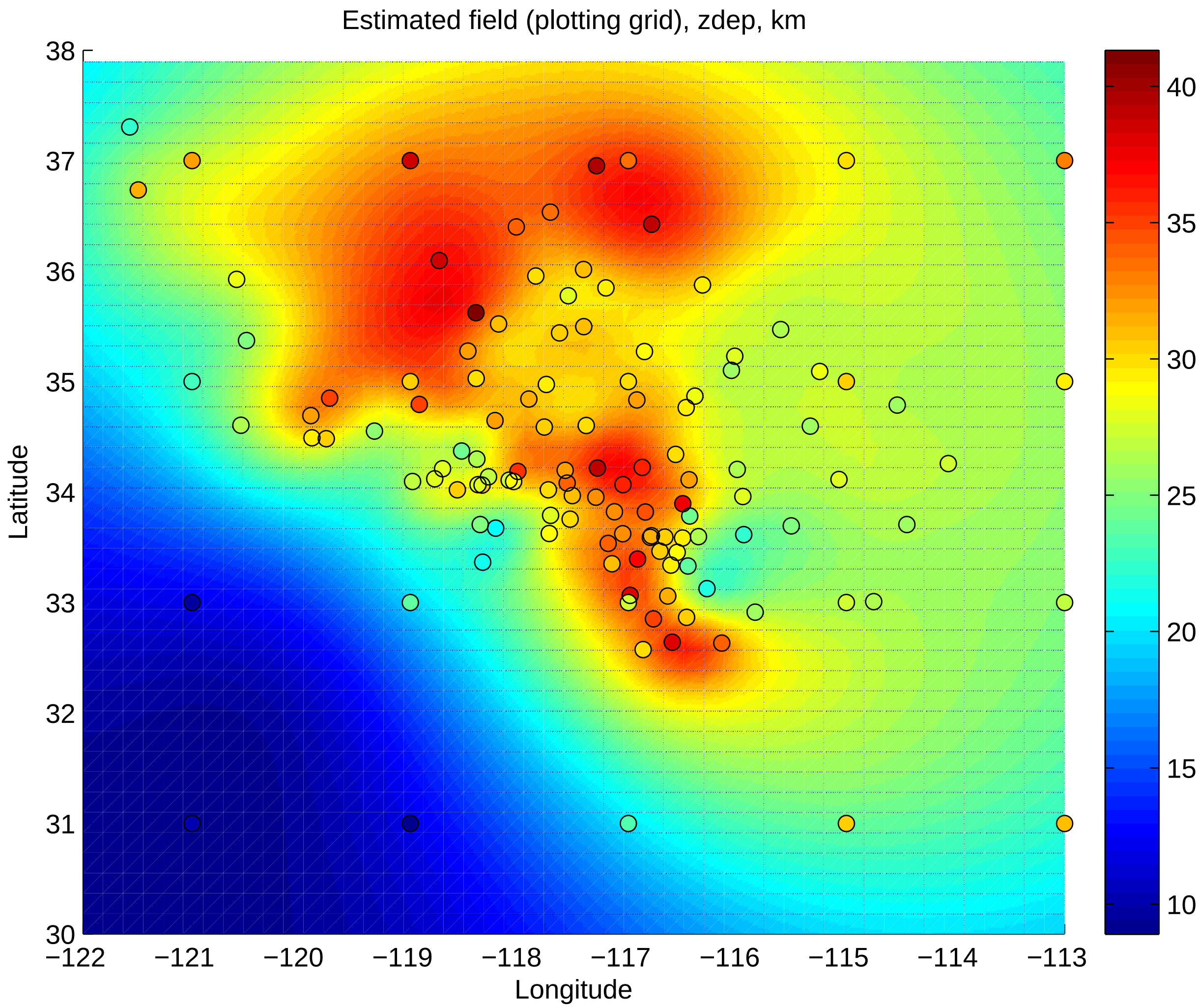


Figure 7: Estimated field, with observed values plotted as circles.

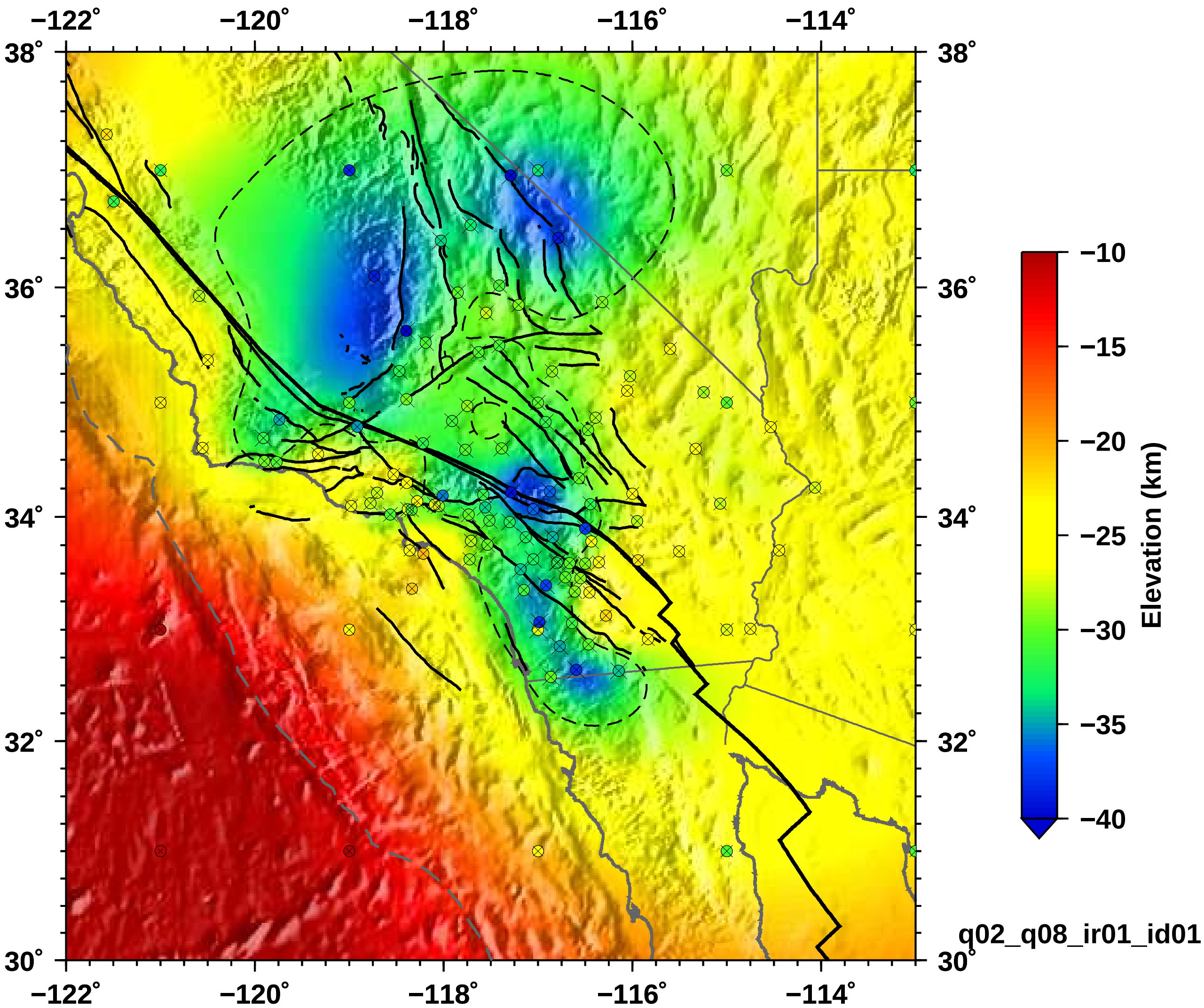


Figure 8: A fancier rendition of Figure 7. The dashed line shows the −30 km contour. Each filled circle is an observed values, with the 'x' marker proportional to the uncertainty.

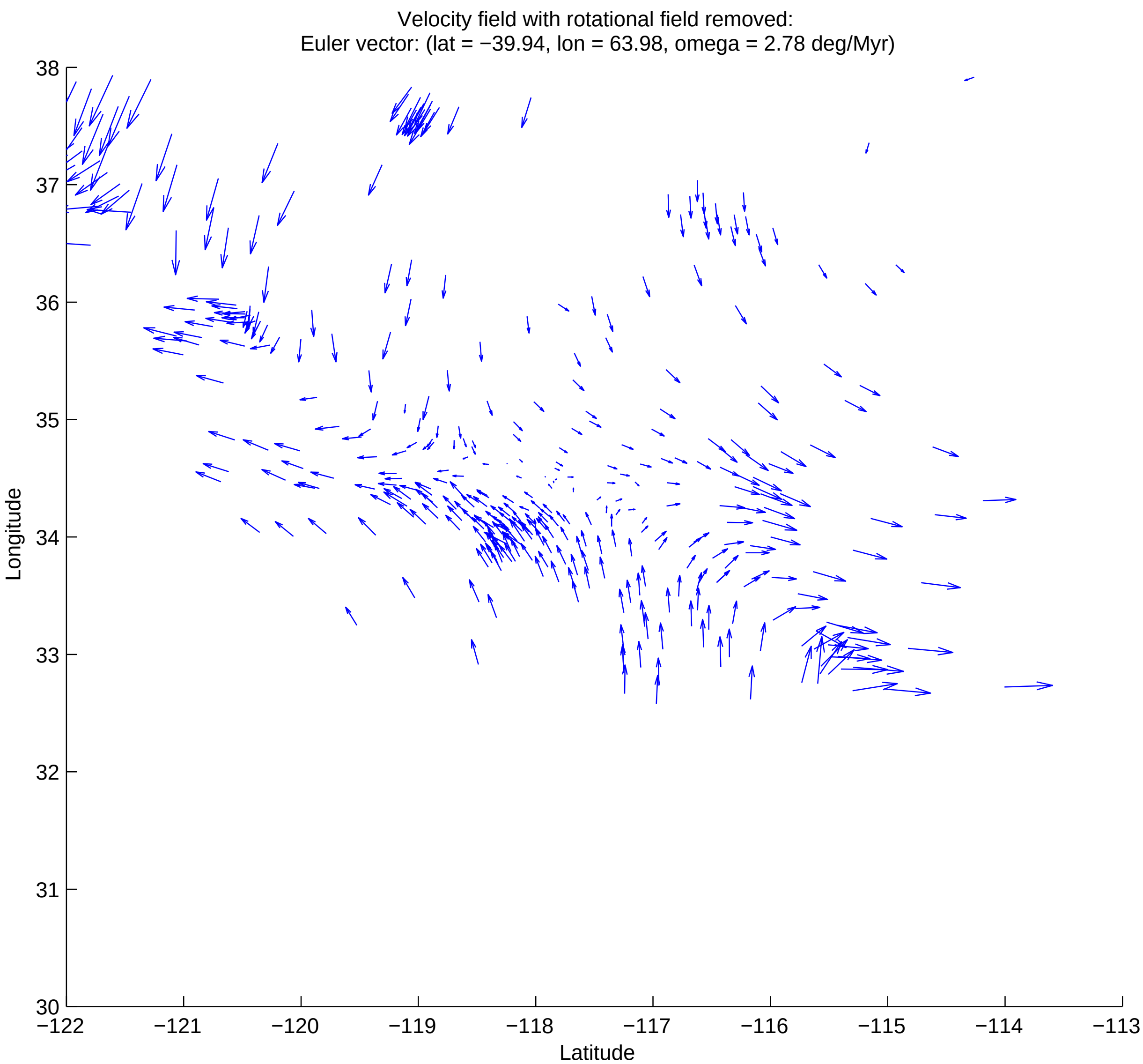


Figure 9: GPS velocity observations for southern California (*Dong et al.*, 2011). These observations are used as the example in Section 3. A uniform rotational field has been removed from the original velocity field (e.g., *Tape et al.*, 2009), though this is not necessary.

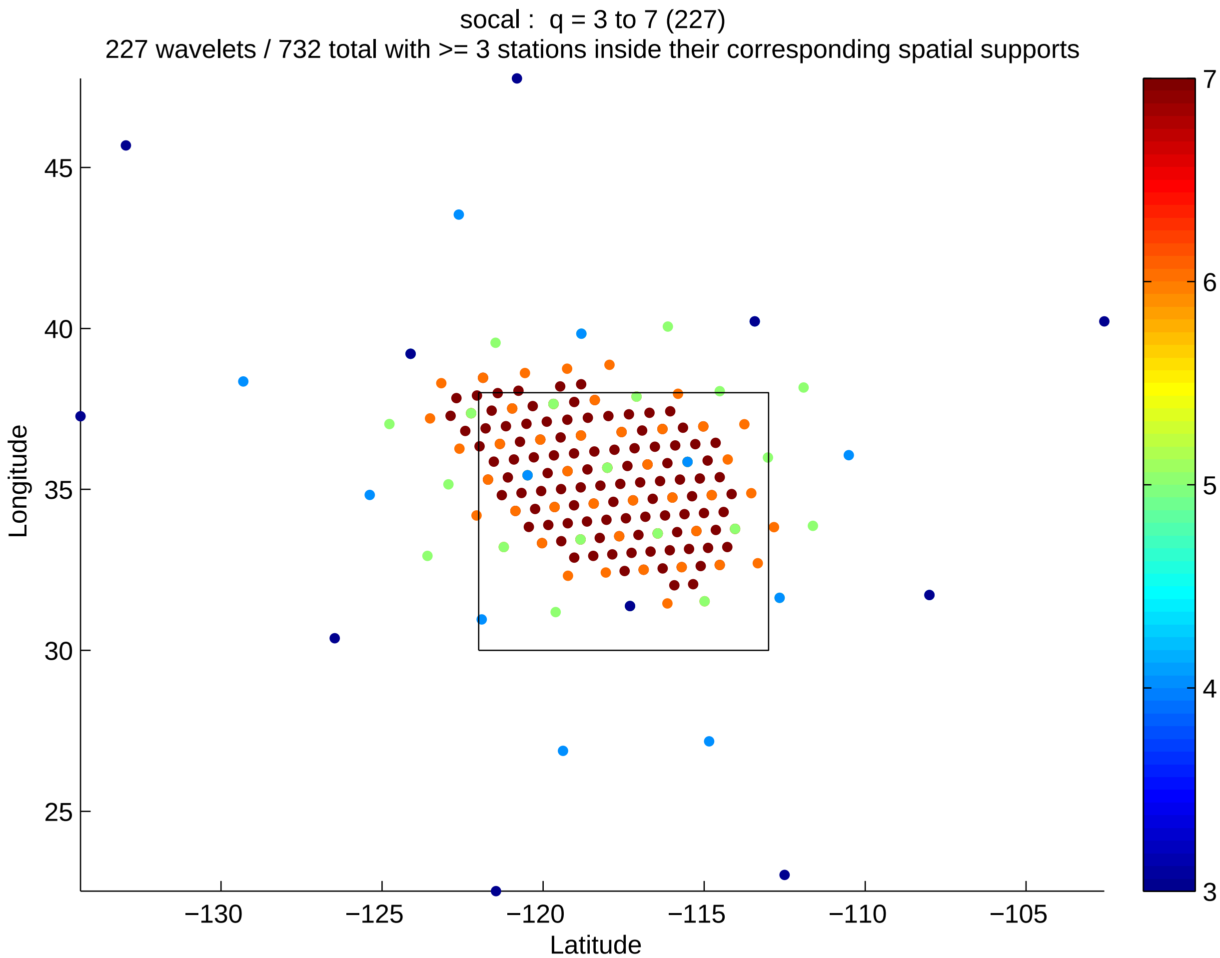


Figure 10: Full set of center-points for basis functions used in the inversion (Section 3). Color corresponds to the scale q for each basis function. Note that some dots correspond to multiple basis functions with different scales q, but only the highest-q is visible. See partition by scale in Figure 12.

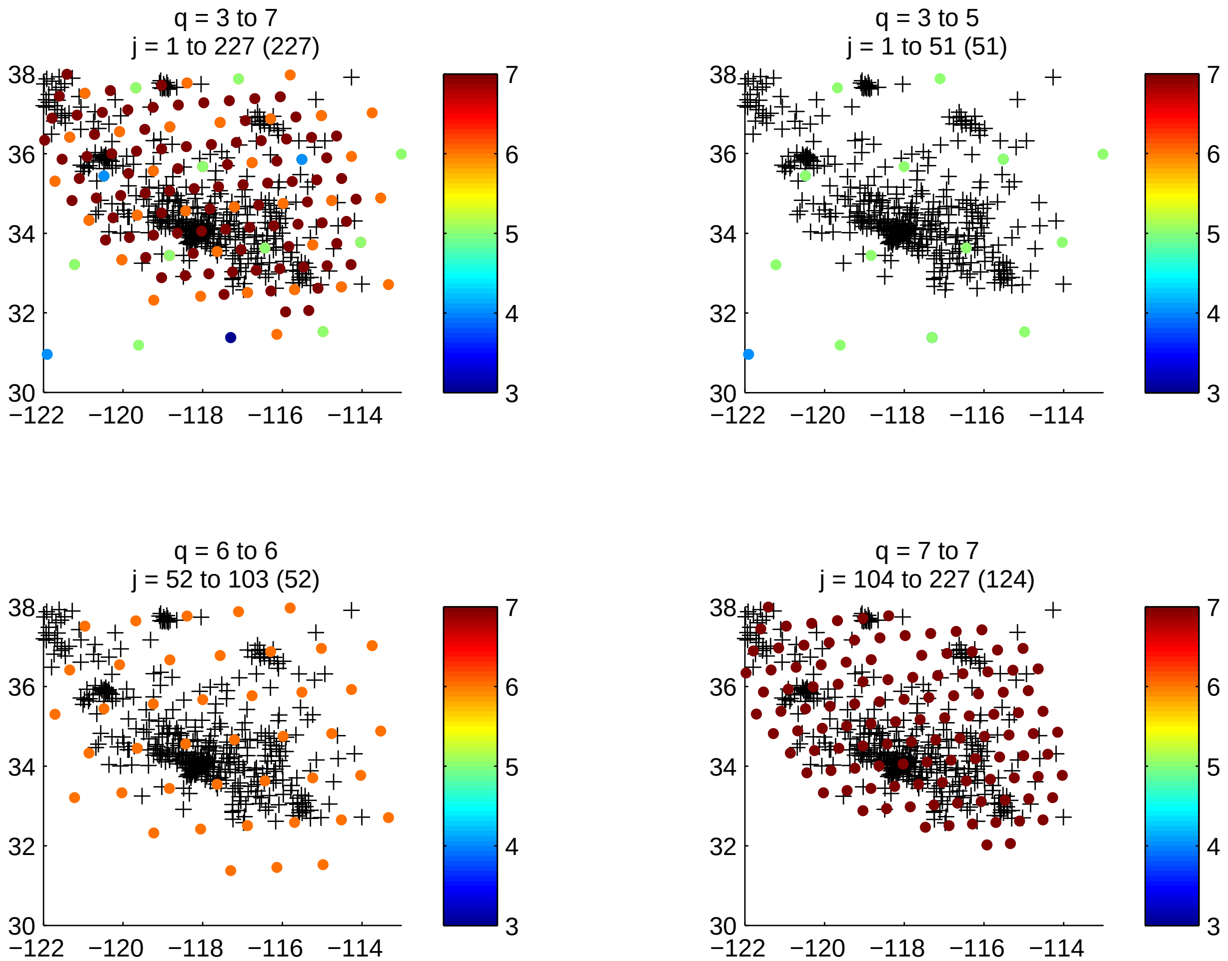
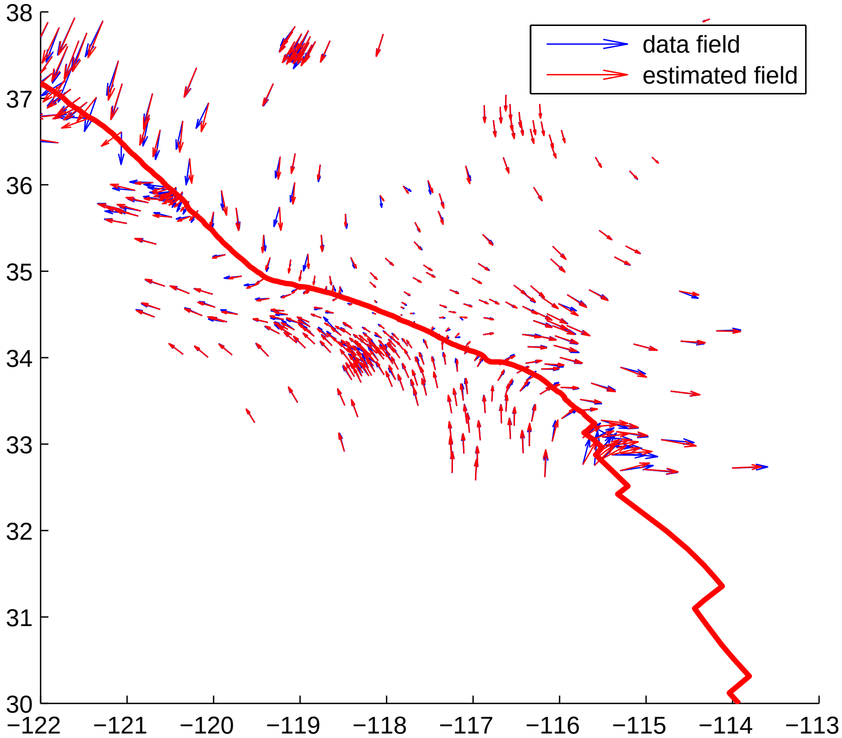


Figure 11: Basis function gridpoint centers for grids q=3-7 (Figure 12), q=3-5, q=6, and q=7. Color corresponds to the scale q for each basis function. Observation locations are plotted at '+' markers (Figure 10).



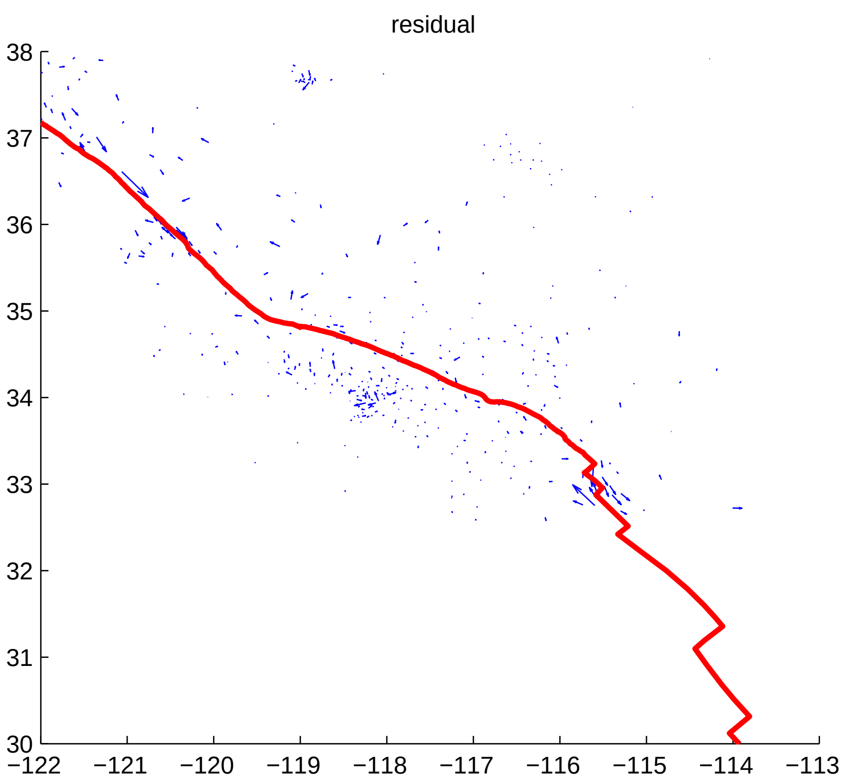


Figure 12: Observed velocity field and estimated velocity fields (top), as well as the residual field (bottom). The two components of the estimated velocity field are obtained from a least-squares inversion for coefficients of spherical wavelet basis functions.

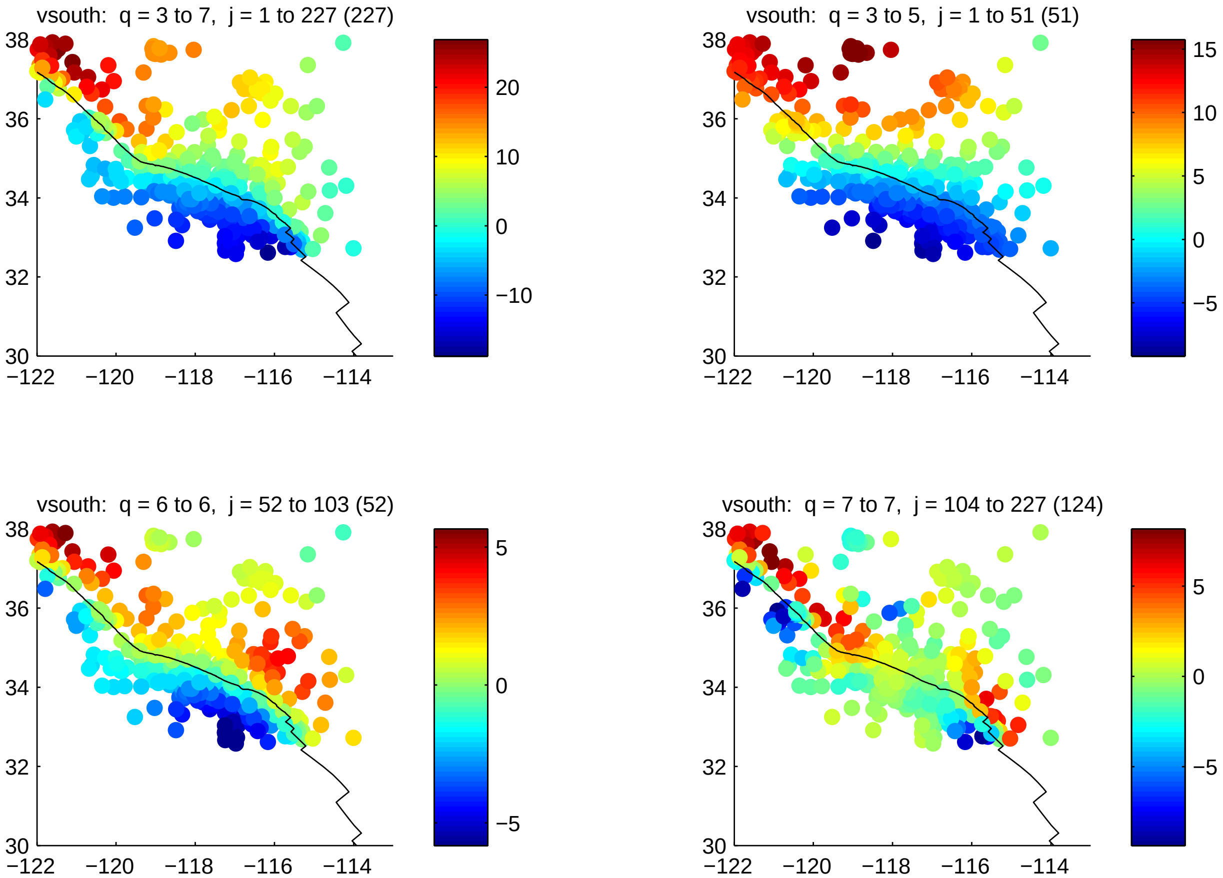


Figure 13: Estimated south (θ) component of the velocity field, plotted at multiple scales: q=3-7, q=3-5, q=6, and q=7.

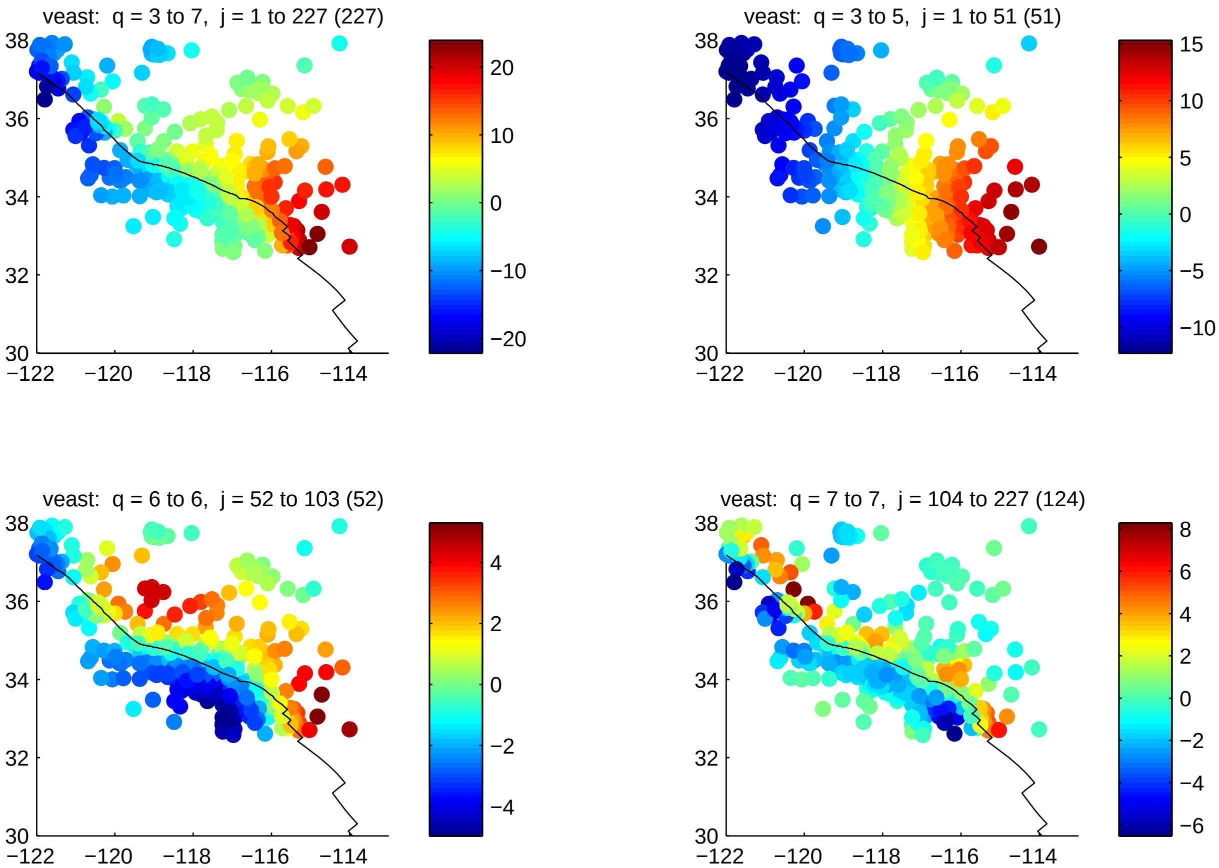


Figure 14: Estimated east (ϕ) component of the velocity field, plotted at multiple scales: q=3-7, q=3-5, q=6, and q=7.

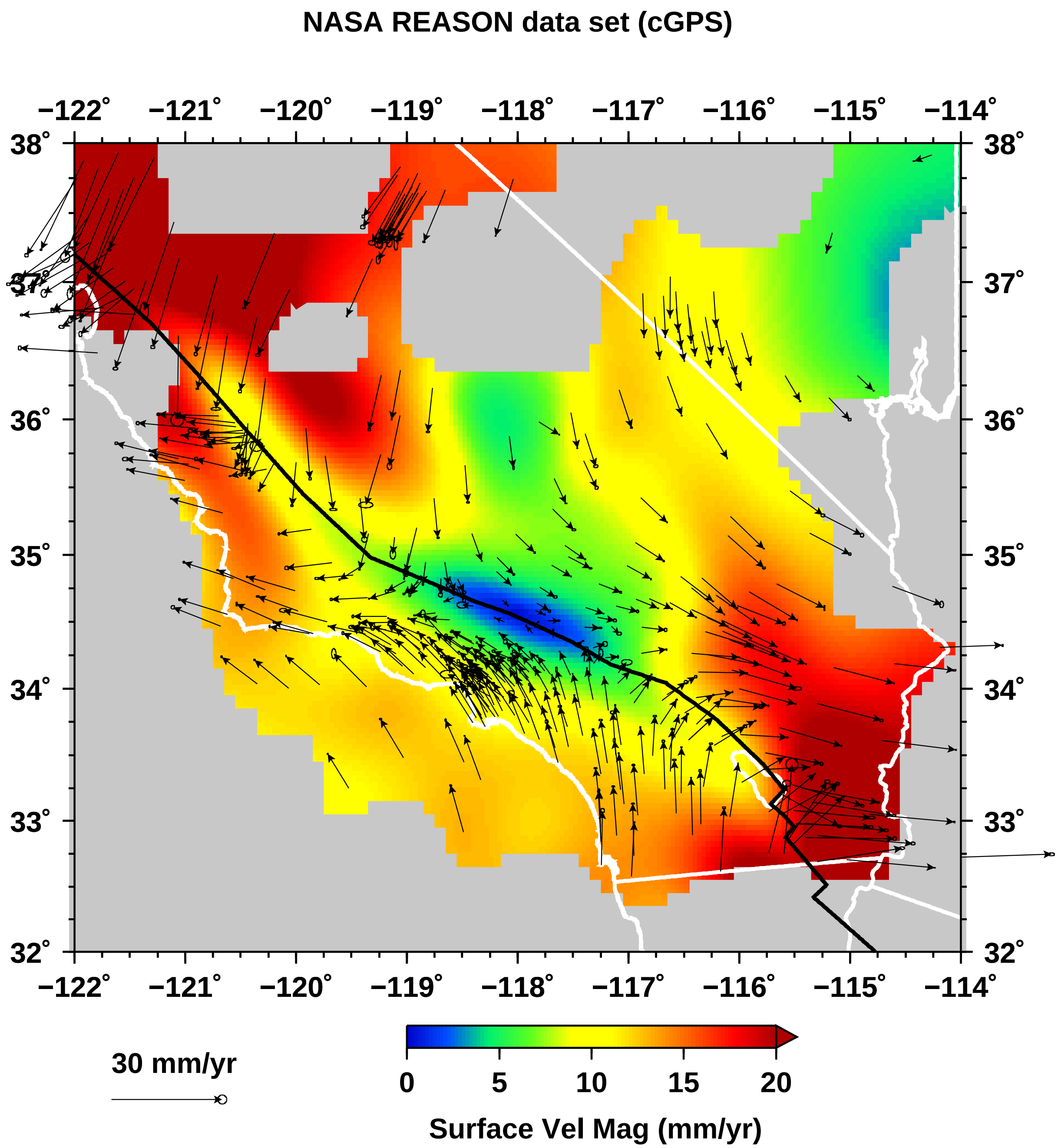


Figure 15: Observed velocity field (arrows) overlying the magnitude of the estimated velocity field (colors). The figure was plotted using the Perl/GMT script `plot_strain.pl`.

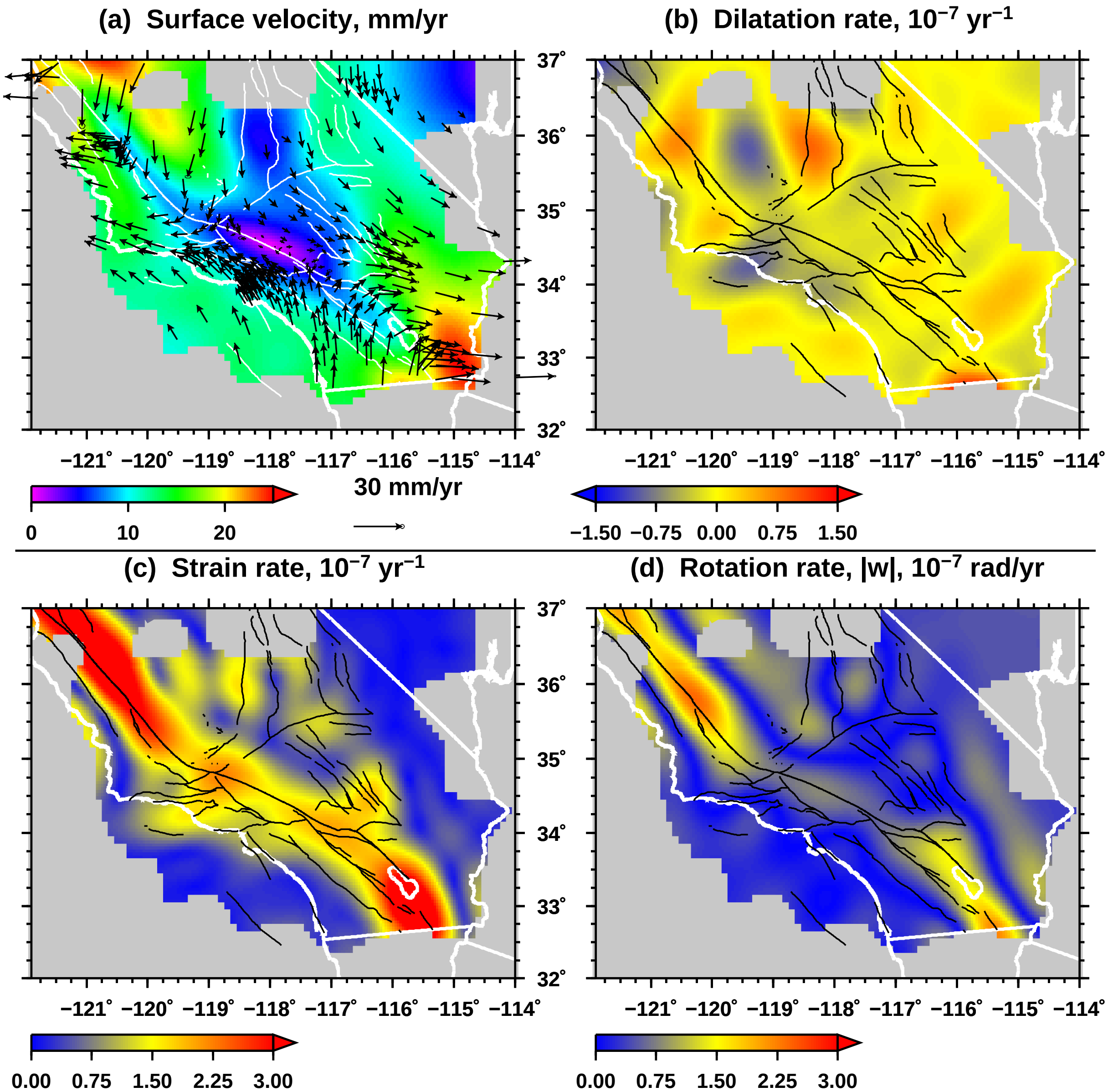


Figure 16: Four-subplot representation of the estimated velocity field for the southern California example. The figure was plotted using the Perl/GMT script `foursub.pl`. These should resemble Figure 11a-b-c-e of *Tape et al.* (2009); the difference is that this example considered only the horizontal components of the velocity (`ndim=2`), whereas in Figure 11 we considered the vertical component as well (`ndim=3`). (NOTE: To plot this file you need to have specified `iwrite=1` when running `surfacevel2strain.m`).