SEISMIC TOMOGRAPHY TUTORIAL

CIDER 2018: VED LEKIC (BASED ON TUTORIAL DEVELOPED BY GUY MASTERS AND MODIFIED BY MARK PANNING)

BACKGROUND INFORMATION

As seismic waves – both body (P,S) and surface waves – travel across the Earth from source to receiver, they are affected by variations of density and seismic wavespeeds in the Earth's interior. For example, their **travel times** can become advanced or delayed, their **amplitudes** increased or reduced, or they may become dispersed as different frequency components travel at different velocities. If we can calculate how a particular travel-time, amplitude, or phase dispersion measurement depends on Earth structure (i.e. what its sensitivity kernel is), then we can use these measurements to image the Earth's interior. **Seismic tomography** is one way in which measurements from many source-receiver paths can be self-consistently combined to construct a three-dimensional image of structure in the Earth.

Love and Rayleigh waves, collectively referred to as **surface waves,** are particularly useful for constraining structure in the upper mantle. This is because they are readily observed and because the speed of propagation of their different frequency components has a different sensitivity to depth profiles of velocity. In this tutorial, you will be using tens of thousands of measurements of **phase velocity** – the distance a peak or trough at a specific frequency travels in one second – to infer the geographic variations of phase velocity at a particular frequency. **Phase velocity maps** at different frequencies can be combined to make a three-dimensional model of velocity variations in the upper mantle; however, we will not be performing this task in this tutorial.

MATHEMATICAL FORMULATION

How should a phase velocity measurement made on a particular source-receiver path be related to geographic variations in phase velocity? A simple but very useful answer to this question is provided by the **path average approximation** (PAVA, Woodhouse & Dziewonski, 1984), which states that the deviation $\delta c_k(\omega)$ of the phase velocity measured on a particular path *k* from a reference value $c^0(\omega)$ is the average of the variations in geographic phase velocities $\delta V(r,\theta,\phi)$ encountered along the great circle path joining source and receiver. Written as an equation, therefore, we have the following relationship:

$$\frac{\delta c_k(\omega)}{c^0(\omega)} = \frac{1}{X}\int_{SRC}^{RCV} \frac{\delta V(r,\theta,\phi)}{c^0(\omega)} ds$$

Where *X* is the distance from source to receiver, ω is the frequency at which the measurements is made, the integration is carried out over the path from source (SRC) to receiver (RCV), and velocity varies with location specified by radius (r), latitude (θ) and longitude (ϕ).

We can proceed by discretizing the space into a finite number (M) of blocks, which allows us to re-write the integration as a summation:

$$\frac{\delta c_k(\omega)}{c^0(\omega)} = \frac{1}{X}\sum_{j=1}^{M} \frac{\delta V_j}{c^0(\omega)} \Delta s_{kj}$$

where *k* is the index identifying the path in question, *j* is the index identifying the block of interest, and the infinitesimal path length element *ds* has been replaced by a finite step length Δs_{kj}. Subscripts *k* and *j* must be introduced because the length of a great circle within a block depends both on which block *(j)* and which great circle *(k)* we are considering.

If we measure the phase velocity on N paths, we can write a matrix equation:

$$\begin{bmatrix} \delta c_{k=1}/c^0 \\ \vdots \\ \delta c_{k=N}/c^0 \end{bmatrix} = \begin{bmatrix} \Delta s_{11}/X & \cdots & \Delta s_{1M}/X \\ \vdots & \ddots & \vdots \\ \Delta s_{N1}/X & \cdots & \Delta s_{NM}/X \end{bmatrix} \begin{bmatrix} \delta V_{j=1}/c^0 \\ \vdots \\ \delta V_{j=M}/c^0 \end{bmatrix}$$

We usually refer to the 1 by M vector of **slownesses** (reciprocals of velocity) as the the **model vector**, **m**, and the 1 by N vector of measured phase velocities as the **data vector, d**. Using this nomenclature, and written compactly, we have:

$\frac{\delta c_k}{c^0} = \ d_k = \left(\Delta s_{jk}/X\right) \delta V_j/c^0 = G_{kj} m_j$ or **d** = **Gm**

In order to incorporate information on measurement reliability (data uncertainty), we usually specify a variance σ_k^2 associated with each measurement. Because we want to assign less importance to data with high uncertainty, we incorporate measurement reliability information using a diagonal matrix, C_D^{-1}, whose entries are $1/\sigma_k^2$.

When the number of phase velocity measurements is equal to the number of unknowns $(N = M)$, the matrix **G** is square, and the model vector can be calculated by multiplying d by the inverse of **G**, i.e. $(m = G^{-1} C_D^{-1} d)$, provided that the inverse exists (i.e. that the matrix G is not singular). When N ≠ M, we can multiply through by the transpose of G, in which case the solution will be given by: $m = (G^T C_D^{-1} G)^{-1} G^T C_D^{-1} d$. However, if the number of measurements is smaller than the number of unknowns, i.e. $N < M$, the problem is said to be

underdetermined, and $(G^T C_D^{-1} G)$ will always be singular. Even when the number of measurements is greater than the number of unknowns, i.e. $N > M$ and the problem may be **overdetermined,** the invertibility of $(G^T C_D^{-1} G)$ will depend on how complementary the path coverage of the various source-receiver pairs is. This makes sense, since even a million measurements of phase velocity between Japan and California will not tell you what the phase velocity in the South Atlantic is.

TASK #1: SETTING UP DATA VECTOR AND COVARIANCE MATRIX, AND BLOCK SIZE

- Run BLOCK-1 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER.
- Specify the frequency of observation (ω). You only have one choice: 100 s.
- Specify the appropriate reference phase velocity $c^0(\omega)$ in km/sec. For 100 s, it is 4.088.
- Specify the approximate size (in degrees, 1° ≈ 111 km) of the blocks. Note that choosing values smaller than 2.0 might cause the inversion to run extremely slowly.
- Specify the multiplier on the data error estimates. If you want to use the actual reported measurement uncertainties, set the multiplier to 1. Values higher than 1 will act to "degrade" the data, while values less than 1 will force the model that is obtained to predict the measures phase velocities more precisely.

TASK #2: COMPUTE THE DERIVATIVE MATRIX – G – AND PLOT UP DATA COVERAGE

- Run BLOCK-2 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER → This computes the G matrix one row at a time so it takes a while.
- Run BLOCK-3 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER → This plots the data coverage, colored by $\log_{10}$ of the number of paths that cross within each block.

REGULARIZATION

Tomographers can use a number of procedures for **regularizing** the problem, or, in other words, for obtaining estimates of **m** even when the matrix $(G^T C_D^{-1} G)$ is not invertible. In this tutorial, we will explore the effects of two regularization procedures:

1. Introduction of prior information on model parameters using the model covariance matric C_M.
2. Singular value decomposition and elimination of eigenvalues smaller than some threshold value.

1. MODEL COVARIANCE MATRICES

Usually, tomographers have other sources of information about phase velocities in the Earth besides those coming from the particular dataset of phase velocity measurements. For example, we might know that $\delta V/c^0$ should be normally distributed with a variance σ^2. Or, we might have reason to believe that the geographic variations of $\delta V/c^0$ are smooth, in the sense that the velocities at locations closer than some distance L will tend to be similar. Mathematically, an *a priori* model covariance matrix that contains both types of information would be: $C_{ij} = \sigma^2 \exp\left[-\left({X_{ij}}/{L}\right)^2\right]$, where $X_{ij}$is the distance between blocks *i* and *j*.

Taking account of this prior information, the solution is given by (Tarantola & Valette, 1982): $m = \widetilde{C_M} G^T C_D^{-1} d$, where $\widetilde{C_M} = (G^T C_D^{-1} G + C_M^{-1})^{-1}$ is the posterior model covariance matrix. The advantage of incorporating this prior information is that it will ensure that $\widetilde{C_M}$ exists and that a solution can be obtained. The posterior model covariance matrix is an extremely useful object, because it not only contains information on the uncertainty of the block estimates obtained with the phase velocity measurement data, but also contains information on how much velocities in two different blocks may trade off with each other. We will return to this later in the tutorial.

Choosing a small value of variance σ^2 is equivalent to strongly **damping** the model vector toward the starting model (in essence, doing our best to force $\delta V_j/c^0 \to 0$). When L is set to 0, the model covariance matrix becomes a diagonal matrix, and its application is called **norm damping**. On the other hand, setting L to be a large distance will result in a very laterally smooth model, and force $\delta V_i - \delta V_j \to 0$ across nearby blocks.

TASK #3: CARRY OUT TARANTOLA & VALETTE INVERSION

- Run BLOCK-4 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER.
- Specify a value for *a priori* model variance (we will use the same value for all blocks). Smaller numbers will yield more highly damped models.
- Specify a value for smoothing distance L (in degrees, remembering that 1° ≈ 111 km). Setting L = 0 will result in pure norm damping, whereas any positive value will incorporate smoothing in the solution.
- Experiment with different values of L and σ^2, and see how the resulting phase velocity maps change. Some combinations of L and σ^2 will yield crazy phase velocity maps (this is equivalent to not introducing a sufficient amount of prior information); generally, you will need to decrease the σ^2 as you increase L.

QUANTIFYING UNCERTAINTY

In science, measurements and inferences are only meaningful if they are associated with an uncertainty. There are three main sources of error that can affect our estimated phase velocity maps:

1. *Errors due to data uncertainty* – Measurements of average phase velocity on a particular source-receiver path are inherently uncertain. In this tutorial – and commonly in global seismology – we assume that this measurement uncertainty is normally distributed and uncorrelated from path to path.
2. *Errors due to regularization* – The introduction of smoothing/damping (a priori model covariance) and SVD thresholding both affect our estimated phase velocity maps.
3. *Errors due to theoretical approximations* – The phase velocity measured on a particular path $(\delta c_k/c^0)$ depends on geographic variations in local wavespeed $(\delta V_j/c^0)$ in a more complicated way than described by PAVA. Therefore, this theoretical approximation will introduce additional error into our phase velocity maps. Mathematically, these errors can be partially modeled by introducing additional terms in the data covariance matrix, and are equivalent to adding noise to our measurements. **Finite frequency kernels** can partially alleviate errors due to theoretical approximations. **Standard uncertainty analysis techniques – such as those we will perform in this tutorial – neglect this source of error.**

Here, we will estimate phase velocity map uncertainty in two ways.

POSTERIOR MODEL COVARIANCE MATRIX

The diagonal entries of the posterior covariance matrix $\widetilde{C_M} = (G^T C_D^{-1} G + C_M^{-1})^{-1}$ are the variances of the block velocity estimates $(\delta V_j/c^0)$. Taking the square root of the diagonal entries and plotting them at the geographic locations of the blocks to which they are associated, we obtain a map of uncertainty on our phase velocity estimates. The off-diagonal entries of $\widetilde{C_M}$ contain information on trade-offs between parameters. In other words, they provide a quantitative answer to the question: if I change the velocity in block A by some amount, by how much does velocity in block B have to change? If there are M blocks given our chosen block size, then there are M*(M-1)/2 pairs of blocks to analyze for tradeoffs. Visualizing the information contained in these off-diagonal elements is not easy.

TASK #4: ESTIMATE UNCERTAINTIES

- Run BLOCK-5 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER → this plots the square root of the diagonal entries of $\widetilde{C_M}$ which is related to the standard deviation of the posterior uncertainty on the velocities in each block.

RESOLUTION MATRIX

A standard way of visualizing how well constrained our phase velocity maps are is to use a resolution matrix approach. In this approach, we start with an exact, arbitrary, phase velocity map. Commonly, a checkerboard pattern **m** is chosen, in which $\delta V_j/c^0$ alternates between $\pm\gamma$ with some characteristic wavelength. (NB: this choice is in many ways suboptimal, due to the fact that it only probes the retrieval of structure at the chosen wavelength.) Then, artificial data are calculated using this checkerboard pattern $d'_k = G_{kj}m_j$.

Now, we can Using this simulated and noisy data **d'**, we can obtain a model estimate **m'** in the same way we did using the actual phase velocity measurements: $m' = \widetilde{C_M} G^T C_D^{-1} d' = \widetilde{C_M} G^T C_D^{-1} Gm$. Equivalently, we can write **m = Rm'**, where **R** is called the **resolution matrix,** and is given by:

$$R = \widetilde{C_M} G^T C_D^{-1} Gm$$

By comparing the estimate **m'** with the exact input pattern **m**, we can get a sense of how well our input exact model is resolved. To quantify how well any possible structure is resolved, we simply have to multiply its model vector by the resolution matrix.

TASK #5: RUN RESOLUTION ANALYSIS WITH CHECKERBOARD TESTS

- For Tarantola and Valette inversion, run BLOCK-6 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER.
- Choose a spherical harmonic angular order *l* for generating a global checkerboard pattern. The length scale (in degrees) corresponding to a particular *l* is 180/*l*. If you set the angular order that is too high for your chosen block size, the input checkerboard will not have the desired appearance.
- BLOCK-6 will plot up the posterior model uncertainties as well as the input and output checkerboard patterns for your choice of inversion parameters.

2. SINGULAR VALUE DECOMPOSITION

An alternative approach toward obtaining a model estimate m even when $(G^T C_D^{-1} G)$ is not invertible is to perform a singular value decomposition, in the matrix is represented in the coordinate system defined by its eigenvectors $\vec{e_j}$ corresponding to eigenvalues λ_j:

$$G^T C_D^{-1} G = \begin{pmatrix} \uparrow & \uparrow & \uparrow & \uparrow \\ \vec{e_1} & \vec{e_2} & \dots & \vec{e_N} \\ \downarrow & \downarrow & \downarrow & \downarrow \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_N \end{pmatrix} \begin{pmatrix} \leftarrow & \vec{e_1} & \rightarrow \\ \leftarrow & \vec{e_2} & \rightarrow \\ \leftarrow & \vdots & \rightarrow \\ \leftarrow & \vec{e_N} & \rightarrow \end{pmatrix}$$

With this representation, we can write an expression for the desired inverse matrix:

$$(G^T C_D^{-1} G)^{-1} = \begin{pmatrix} \uparrow & \uparrow & \uparrow & \uparrow \\ \vec{e_1} & \vec{e_2} & \dots & \vec{e_N} \\ \downarrow & \downarrow & \downarrow & \downarrow \end{pmatrix} \begin{pmatrix} 1/\lambda_1 & 0 & \cdots & 0 \\ 0 & 1/\lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/\lambda_N \end{pmatrix} \begin{pmatrix} \leftarrow & \vec{e_1} & \rightarrow \\ \leftarrow & \vec{e_2} & \rightarrow \\ \leftarrow & \vdots & \rightarrow \\ \leftarrow & \vec{e_N} & \rightarrow \end{pmatrix}$$

From this expression, we can immediately see that the desired inverse will not exist if any of the eigenvalues $\lambda_j \rightarrow 0$, because this will result in an undefined value of $(G^T C_D^{-1} G)^{-1}$.

To correct for this problem, we will do something that may seem quite unusual. First, we will arrange the eigenvalues in decreasing order so that $\lambda_j = \lambda_{j+1}$. Then, for all eigenvalues smaller than some threshold value, i.e. $\lambda_j < \varepsilon\lambda_1$, we will set $1/\lambda_j \rightarrow 0$. Doing so allows us to approximate $(G^T C_D^{-1} G)^{-1}$, and obtain a matrix analogous to the $\widetilde{C_M}$ in the Tarantola & Valette formalism. What does this mathematical procedure do to the problem of using phase velocity measurements to estimate phase velocities in geographic blocks across the globe? Conceptually, you can think about it this way: We are unable to resolve velocity variations in the combination of blocks represented by the eigenvector $\vec{e_j}$ that corresponds to those $\lambda_j < \varepsilon\lambda_1$. By setting $1/\lambda_j \rightarrow 0$, we are telling the inversion to simply ignore these combinations of blocks, and estimate velocities in blocks where such an estimate is feasible given the phase measurement data. Which combinations of blocks get ignored is determined by data.

Choosing larger threshold values (e.g. $\varepsilon = 0.5$) will result in fewer effective model parameters being estimated (less variation in $\delta V_j/c^0$), but does not have a direct relationship to norm damping or smoothing the way that specifying the *a priori* model covariance matrix does.

TASK #6: CARRY OUT SVD-BASED INVERSION

- Run BLOCK-7 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER.
- Specify how many eigenvalues λ_j to keep (the others will be ignored, i.e. $1/\lambda_j \rightarrow 0$).

- Experiment with numbers that are small compared to the total number of blocks, ranging up to the total number of blocks (which may fail, depending on your block size).

TASK #7: RUN RESOLUTION ANALYSIS WITH CHECKERBOARD TESTS

- Run BLOCK-8 of *InverseTheoryTutorial.py* by clicking inside it and typing CTRL+ENTER.
- Choose a spherical harmonic angular order *l* for generating a global checkerboard pattern. The length scale (in degrees) corresponding to a particular *l* is 180/*l*. If you set the angular order that is too high for your chosen block size, the input checkerboard will not have the desired appearance.
- BLOCK-8 will plot up the posterior model uncertainties as well as the input and output checkerboard patterns for your choice of inversion parameters.