Dissertation Methodology

Notes:

* Mention that this method is pretty much the same as Uieda only difference is the cross validation.
* Brief overview on parametrization, forward problem, inverse problem, regularisation, and Bott’s method.
* Cross validation – split up have training size ranging between 2/3, 3/4, and 4/5 to see the mean error between gravity and seismic estimates. Want to see how many points are needed to still get relatively accurate values (small MSE’s) so we know that when seismic point estimates are not present the gravity moho calculated is still accurate (at around 2.3-2.5km error).
* Statement about using python software.

Methodology:

To accurately obtain a model of the Moho depth one must first remove all other effects that contribute overall gravitational values recorded over an area. This is achieved through correcting for things such as removing the scalar gravity of an ellipsoidal reference Earth (the Normal Earth) and then the removal of all other effects via a Bouguer correction otherwise known as the correction for topography. Initially though the effect of the Normal Earth needs to be removed from the same point as where the gravity observation was made and is calculated from the closed-form solution in Li & Götze (2001). The value obtained here is called the gravity disturbance and can be seen in the equation below (insert gravity disturbance equation once on .tex file). (Also include Figure 1 in Uieda & Barbosa paper, stages of gravity correction).

The gravity disturbance is still not a direct result of the change in density associated with the Moho discontinuity but also an amalgamation of topography with reference to the normal ellipsoid, variations of density in the crust (e.g. sedimentary basins and igneous intrusions), anomalies below the upper mantle, and mass deficiency due to the oceans. These effects are removed through a topography correction, (insert topography correction equation). This is the method used in Uieda & Barbosa (2017) and assumes that the effects of other crustal and mantle sources are negligible, and after this correction the gravitational values attained are purely a result of the density variations either side of the Moho discontinuity. Seeing as the data is obtained for a sufficiently large area (South America) this correction for topography amongst other things is calculated using tesseroids as part of a spherical Earth approximation. Effects of the tesseroids are calculated using a GLQ integration presented in Asgharzadeh et al. 2007) and improved upon in Uieda et al. 2016 through the adaptive discretization scheme. (insert Figure 2 in Uieda & Barbosa, tesseroids).

Overview of the methodology of calculating a gravity derived Moho model from Uieda (2017)

Note: for each section don’t write more than 200 words, around 1000 overall for this bit.

Parametrization, the forward problem, inverse problem, regularization, Bott’s method, estimating the parameters:

As the implementation of cross validation to estimate uncertainty in the model is implemented as part of the code used in the Uieda & Barbosa (2017) paper the method in calculating the model is largely similar. For context, an overview of this method will be given but for more detail see Uieda & Barbosa (2017).

Upon the calculation of the Bouguer disturbance from the removal of topography, sediments etc. the forward model is parameterized by discretizing the anomalous Moho onto tesseroids. The forward model aims to calculate the difference between the Normal Earth Moho and the true Moho depth, and depending on which is shallower will result in either a positive (red) or negative (blue) density contrast displayed by the colour of the tesseroids. The overall absolute value of the density contrast is predetermined, this produces a nonlinear problem with equation, (insert equation 3 from Uieda, don’t include i values just generic equation). Where d is the data vector, p is the parameter vector containing Moho depths, and f is the non-linear function.

Leading on to the inverse problem the parameter vector is estimated using least squares that reduces the misfit to the data, (insert least-squares equation). Where d0 is the observed gravity data, the equation means that this is a non-linear inverse problem, but we can calculate the minimum error using optimization, where a perturbation vector Δp0 is iterated until a minimum is reach which leads to the minimum value of φ(p).

The optimization of the least squares estimate however is not enough for estimating the relief associated with the Moho and needs regularization in the form of a first-order Tikhonov regularization (Tikhonov & Arsenin 1977) to ensure smoothness in the model and no sharp jumps in Moho depth, to provide a realistic model. (inset regularization equation). R a matrix composed of first order differences between tesseroid depths. This along with the least squares estimate leads to an inverse problem that is solved by minimising the goal function, (insert goal function), µ is the regularization parameter that helps control the fit to the observed data and the smoothness.

After the rearrangement and substitution of equations we arrive at a linear equation system which can calculate the real Earth Moho depths with reference to the Normal Earth Moho. (insert linear equation system equation), where Ak is the Jacobian matrix, and Δpk is the parameter perturbation vector.

Bott’s method from Bott, 1960, is a method that calculates the thickness of a sedimentary basin based of gravitational data, the method is iterative so recalculates a new vector of basement depths from the previous calculation until a value where the residuals (equation numerator) fall below the noise level. (insert Bott’s method equation) Where Δp is the density contrast between the sediment and the reference density, and G is the gravitational constant (6.67x10-11 m-3 kg-1 s-2). However, in 2014 Silva et al showed that Bott’s method can be written as (insert special case Bott’s method) and the main advantage to this is that this method does not need the solution of equation system. This is through using a diagonal matrix of values that scale the model depths to fit the gravitational data.

For calculating the depth to the Moho Uieda & Barbosa, 2017 uses Bott’s method in the inversion process and adapts it onto a spherical coordinate system using tesseroids. Stating in the paper that this method retains the efficiency of Bott’s method while accounting for the stability problem previously present in the method. Following this step, the final part of the method involves calculating the hyperparameters (regularization parameter µ, Moho density-contrast Δp, and depth of the Normal Earth Moho zref) which will be used in the inversion process. The calculation of the regularization parameter is through a method of hold out cross validation from Hansen, 1992 and from this optimal regularization value the other two hyperparameters can be calculated as these depend on the value of the regularization but not vice versa. The main way these parameters are calculated is by finding the smallest Mean Square Error (MSE) through a cross validation method which compares known Moho depths to calculated ones from the hyperparameters and picks the values with the smallest associated MSE.

Implementation of cross validation in error estimation

Cross validation (CV) often used for large data sets in order to see how well the model produced from said data set performs independently (i.e. when there is no data to base the model off). The result of cross validation is often an MSE result or mean square error which is the accuracy of the new predicted independent model and is often the goal of the cross validation in the first place, and how this can be minimised. For many cases of CV the data set is split up into a training and testing set with the training set being used to find the best solution or model attained from the smallest cross validation value while the testing (validating) set is kept separate. It is then compared to the model created from the training set and compared against to get a form of prediction of errors, and how well a model will perform for a completely independent set which in this case the testing set is. The cross-validation procedure also helps with overfitting of the model or selection bias where some points tend to skew the overall model more than others. And in order to minimise these problems the most the process is repeated multiple times with different training and testing sets along with the variation in the size of these subsets.

There are many types of CV that are relevant for different case specific things, although mostly the methods are split into two main types: exhaustive and non-exhaustive. Exhaustive CV is where all possible combinations of separating the full data into training and testing sets are used leading to a limited number of iterations that can be run. This method often works best for small volumes of data as with larger sets the computational time becomes uneconomical and an overall waste of time. Non-exhaustive CV does not use all possible combinations but rather a large enough number of iterations to be considered representative of the full data set.

For this method in particular a procedure of non-exhaustive repeated random sub-sampling validation also known as the Monte Carlo method works on as the name states repeatedly selecting a random selection of the data into a training and testing set, demonstrated in Figure [X], with the sizes of each set being determined by the user. With the training data being used to find the best model or solution with the associated lowest cross validation score and this then being compared to the testing or validating set to find the associated errors on the model. The procedure used here is similar but not to be confused with the exhaustive counterpart leave-p-out cross validation which is the exact same process except it uses all combinations of the data, which hasn’t been used here as the data is too large and would computationally be a waste of time. The data used here is seismic point data that is compared to a gravitationally derived moho model from selected hyperparameters. All the different models from different hyperparameter combinations are weighed up against a training set of seismic point estimates to find the model with the smallest variance or best match to these point estimates. It is then compared to the rest of the seismic data “held back” to find the Mean Square Error (MSE) and subsequently the Root Mean Square Error (RMSE), which is the average uncertainty of the model in kilometres. With 200 iterations per size and there being 3 sizes of training sets each being the closest integer value to fractions 2/3, 3/4, and 4/5 of the full data this would lead to a large enough proportion of all possible combinations to attain a representative insight to how well the model performs for an independent set. The full data consists of 937 seismic point estimates from Assumpção et al. ([2013](javascript:;)), of which the locations can be seen in Figure [X], meaning that the different training subset sizes are 625, 703 and 750, respectively. The training size always has to make up a larger proportion of the full data than the validating set as the model initially attained is representative of the overall data and hence the results of the MSE calculation is significant. This is supported by Berrar, 2018 who stated that 70-90% of the full data should be part of the training set to be considered useful.

A single iteration of this procedure works through randomly selecting a select number of elements in an array of 937 points, these element positions are the data points in the latitude, longitude and seismic estimates selected to be part of the training set and the leftover elements not used are placed in separate latitude, longitude and seismic estimates array and are held back. The training set is used to find the best model through the score\_all function which gets the cross-validation scores for all solutions from which the best solution is selected that has the smallest CV score. This solution is then compared to the testing set arrays with the score\_seismic\_constraints function returning the MSE value between the best solution and the point estimates. This value is then stored in an array with all the other iterations and is then plotted as a histogram to find the mean and standard deviation of the MSE to obtain an estimate of the uncertainty of the model in predicting the Moho depth where there is no seismic data available.

However, with all methods comes the disadvantages, repeated random sub-sample validation (RRSSV) suffers from some randomly generated selection bias, where some datum may not be selected for any iteration as a part of the validating or testing subset but on the other hand some datum may have been selected multiple times, possibly skewing the MSE result. Additionally albeit unlikely testing sets selected for separate iterations may be identical, but this should not be a problem given the sufficiently large data set so the chance of exact same subsets in different iterations is very small.

Software Implementation

This inversion and error estimation method put forward in the methodology is executed in the Python programming language. Software is available under the BSD 3 clause open-source software license. The code in this project depend on open-source libraries scipy and numpy (Harris et al., 2020) for computational number exploitation, matplotlib (Hunter, 2007, <http://matplotlib.org>) and seaborn (Waskom et al. 2015, <http://stanford.edu/∼mwaskom/software/seaborn>) for plots and maps, Fatiando a Terra (Uieda et al. 2013, <http://www.fatiando.org>) for geophysics tasks. scipy.sparse package is implemented for use on sparse matrix arithmetic and linear algebra and solves the linear equation system equation.

The use of Jupyter notebooks (Perez & Granger 2007, <http://jupyter.org/>), which merge the source code, results, and figures of the project.

All source code, Jupyter notebooks, data, and error estimate results are available through an online repository (<https://github.com/compgeolab/moho-uncertainty>).