Computational Modelling for Biomedical Imaging - Coursework 1

Razvan Valentin Marinescu
Student Number: 14060166
razvan.marinescu.14@ucl.ac.uk

February 7, 2015

Q1.1.1

From eyeballing the data in fig. 1 we notice that the fit is very poor, especially for the 3 entries where the b-value is 0, having a $\Delta S > 7 * 10^5$. The other entries don't fit the data well either.

The final value of RESNORM (1.2242e+12) is clearly above the kind of value we would normally expect. The fit is poor probably because the search strategy got stuck in a local minima.

The expected value of RESNORM would be given by the formula:

$$E\left[\sum_{i=1}^{33}(X_i - \bar{X}_i)^2\right] = E\left[\sum_{i=1}^{33}\sigma_i^2\right] = 33 * \sigma^2$$

Since sigma is between 5000-6000 this means that $8.25 * 10^8 < RESNORM < 1.188 * 10^9$.

Some of the parameter values we obtain are not sensible. For example, the value of f is negative, meaning that we have negative intra-axonal diffusion, which is not physically possible. The value of S0 is also really high, when in fact it should be around $1.1 * 10^5$.

Q1.1.2

We did the following transformations:

- $S0 \to S0^2$
- $d \to d^2$
- $f \to (1 + e^{-f})^{-1}$ (sigmoid)

This time the algorithm converges to the following parameters: $S0 = 1.132129e + 05, d = 1.534e - 03 f = 0.575, \theta = -1.03, \phi = -0.11$ with a RESNORM of 1.33e+09. These parameter values are realistic and give us a RESNORM that is around 1000 times smaller. Note that we this time the starting parameters were changed to more realistic values ($S0 = 1.5e + 05, d = 3e - 03 f = 0.5, \theta = 0, \phi = 0$). The reason we gett better parameters and RESNORM is because we introduced the transformations that keep the parameters within reasonable limits and because we used a better starting position.

Q1.1.3

Running the same procedure for gaussian-distributed starting points and for three different pixels gives us the following parameters:

Voxel	S0	d	f	θ	ϕ	GlobalMinCounter(out of 100)
(52,62,25)	1.132e+05	1.534e+03	0.575	2.10	6.17	89
(63,40,18)	1.121e+05	1.829e+03	0.5125	1.19	5.43	83
(50,64,23)	1.214e-05	7.208e-04	0.1955	5.05	5.02	87

GlobalMinCounter represents the number of times the global minimum was found was found. We are quite confident that those are the global minimums, since we made sure that the covariance of the gaussian distribution from which we sample the starting position is large enough to cover a wide area of the space. If we consider the probability p of finding the global minimum in one run, then after k runs the probability of **not** having found the global minimum would be $(1-p)^k$ where k is the number of runs. Then, the minimum number of runs in order to be 95% confident that we will find the global minimum is $\arg\min_k 0.05 > (1-p)^k$. Since $p \approx 0.86$ then $k = 2$, meaning that 2 runs are enough to be 95% confident.

Q1.1.4

The maps for $S0$, d, f, $RESNORM$ and the fibre direction n are shown in figures 2a, 2b, 3a, 3b and 4 respectively.

Q1.1.5

I implemented the DTI model and used the following mappings to get to the parameters of the BallStick model:

1. $d = mean(D)$, $f = 0.5(|\lambda'_1 - \lambda'_2| + |\lambda'_1 - \lambda'_3| + |\lambda'_2 - \lambda'_3|)$ where $\lambda'_i = \lambda_i/(\lambda_1 + \lambda_2 + \lambda_3)$ for $i \in 1..3$
2. $d = tr(D)/3$, f as above
3. $d = max(D)$, $f = 0.5((\lambda'_1 - \lambda'_2)^2 + (\lambda'_1 - \lambda'_3)^2 + (\lambda'_2 - \lambda'_3)^2)$, for λ'_i as above

The most efficient was method 1, for which the global minimum was found in approximatively 93/100 runs.

Q1.1.6

The computation time for `fmincon` is around 9x slower when compared to `fminunc` (averaged over 100 trials, `fmincon`=6.68sec, `fminunc`=56.28sec). Nevertheless, the global minimum is almost always found by both functions (87/100 runs)

Q1.1.7

Q1.1.8

Q1.2.1

The histogram of $p(x|A)$ along with the 2σ and 95% ranges are shown for parameters $S0$, d and f in figures 5, 6 and 11 respectively. These have been computed for the voxel (52,62,25). A table with the parameter estimates for various other voxels is given in figure 8. A better visualisation of the parameter intervals (along with MCMC and Laplace) is given is figure

Q1.2.2

Q1.3.1

Q1.3.2

Q1.3.3

Q1.4.1

Q1.4.2

Images

Q1.1.1

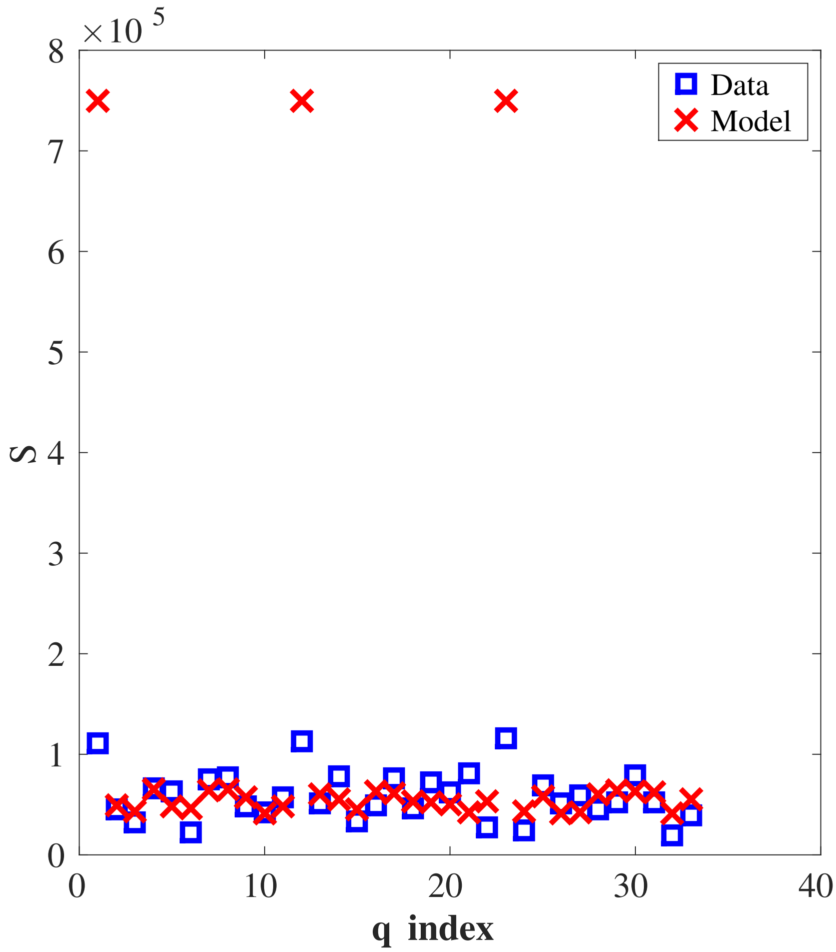
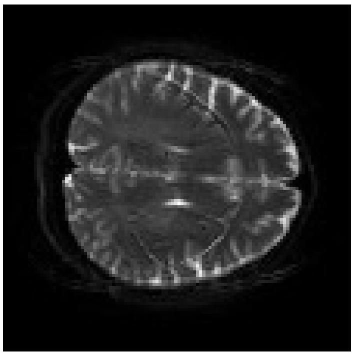
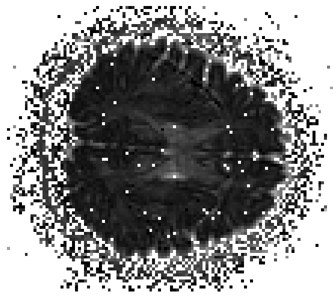


Figure 1: Data points for the voxel at position $(52, 62, 25)$

Q1.1.4

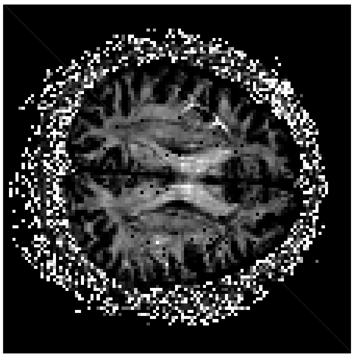


(a) $S0$ map

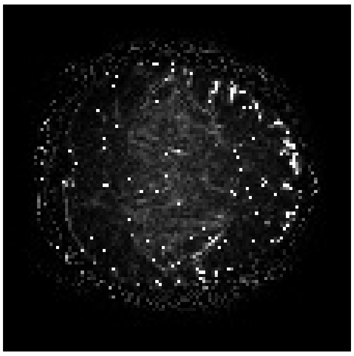


(b) D map

Figure 2



(a) F map



(b) $RESNORM$ map

Figure 3

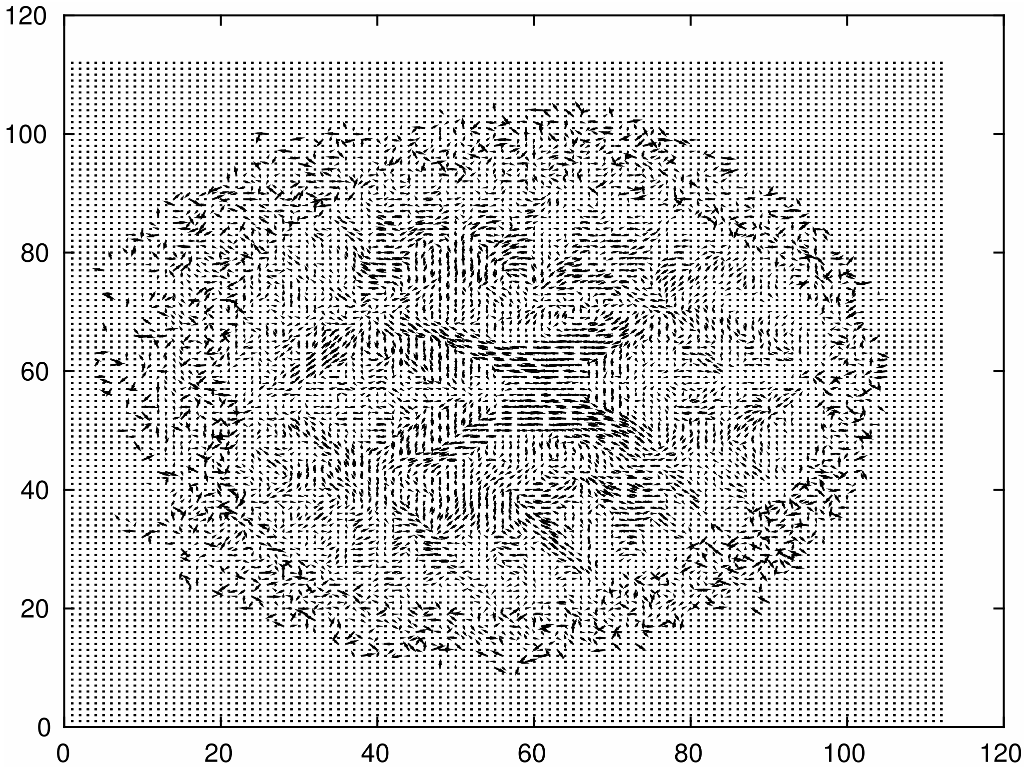


Figure 4: Fibre direction map

Q1.2.1

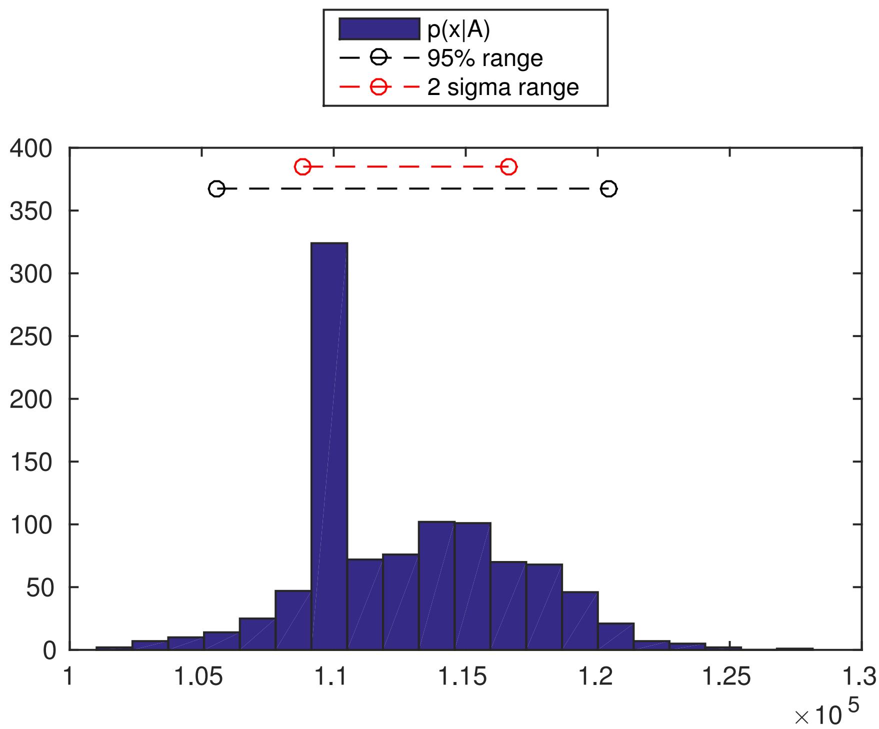


Figure 5: Parametric bootstrap for $S0$ using voxel (52,62,25)

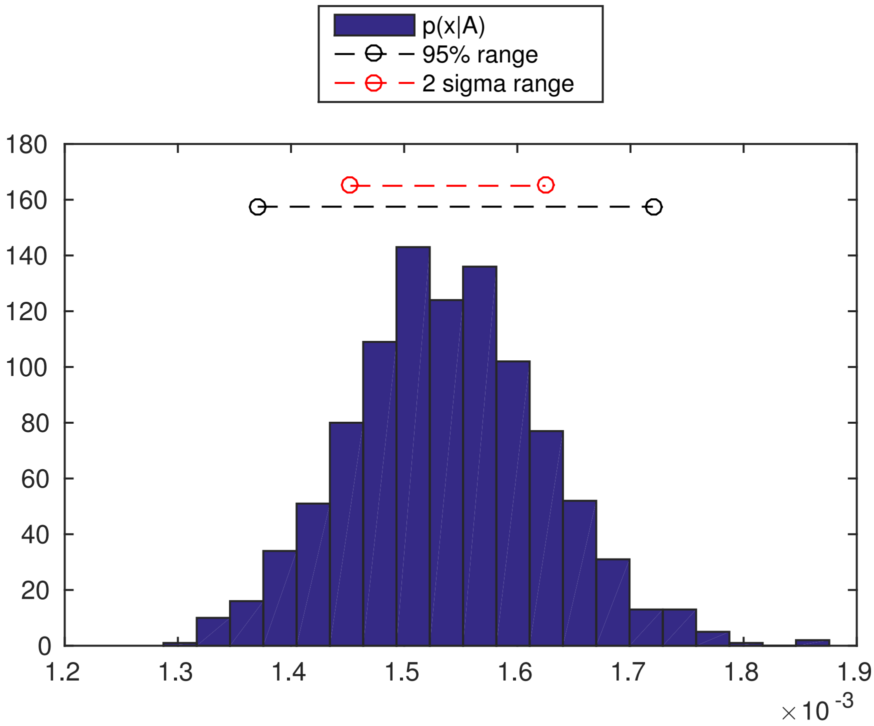


Figure 6: Parametric bootstrap for d using voxel (52,62,25)

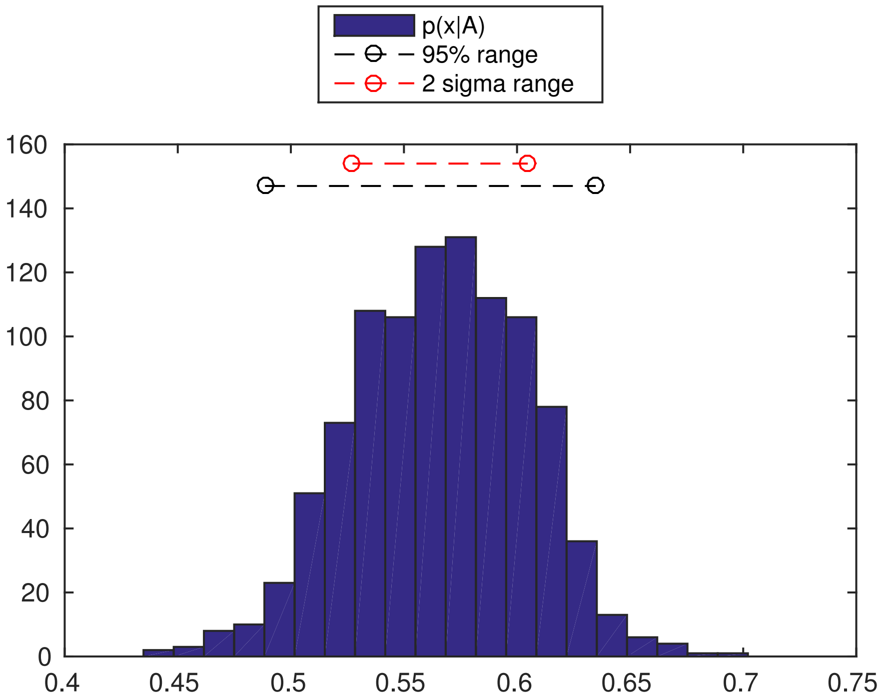


Figure 7: Parametric bootstrap for f using voxel (52,62,25)

2-sigma ranges

Voxel	$S0$		d		f	
(52,62,25)	1.088e+05	1.166e+05	1.452e-03	1.624e-03	0.527	0.604
(63,40,18)	1.017e+05	1.229e+05	1.026e-03	1.397e-03	0.136	0.326
(70,64,14)	1.022e+05	1.112e+05	7.584e-04	8.928e-04	0.078	0.185

95% confidence intervals

(52,62,25)	1.055e+05	1.204e+05	1.370e-03	1.719e-03	0.488	0.634
(63,40,18)	0.918e+05	1.330e+05	0.870e-03	1.608e-03	0.000	0.404
(70,64,14)	0.976e+05	1.152e+05	6.888e-04	9.507e-04	0.000	0.231

Figure 8: Table with 2-sigma and confidence intervals for voxel (52,62,25)

Q1.2.2

Q1.2.3

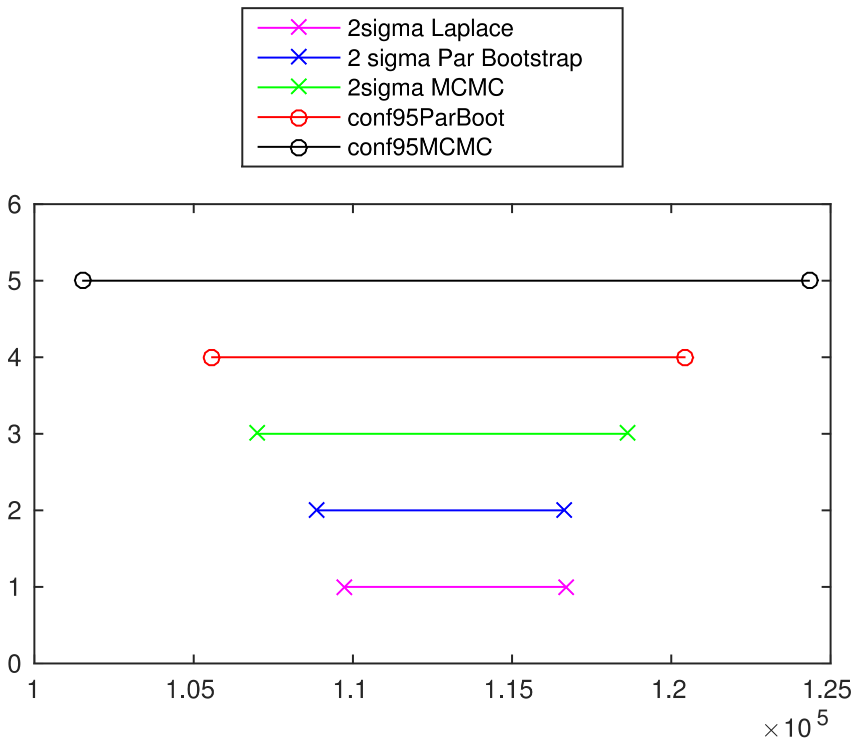


Figure 9: 2σ and 95% confidence intervals on parameter $S0$ using three different methods: parametric bootstrap, MCMC and Laplace. Voxel used was (52,62,25)

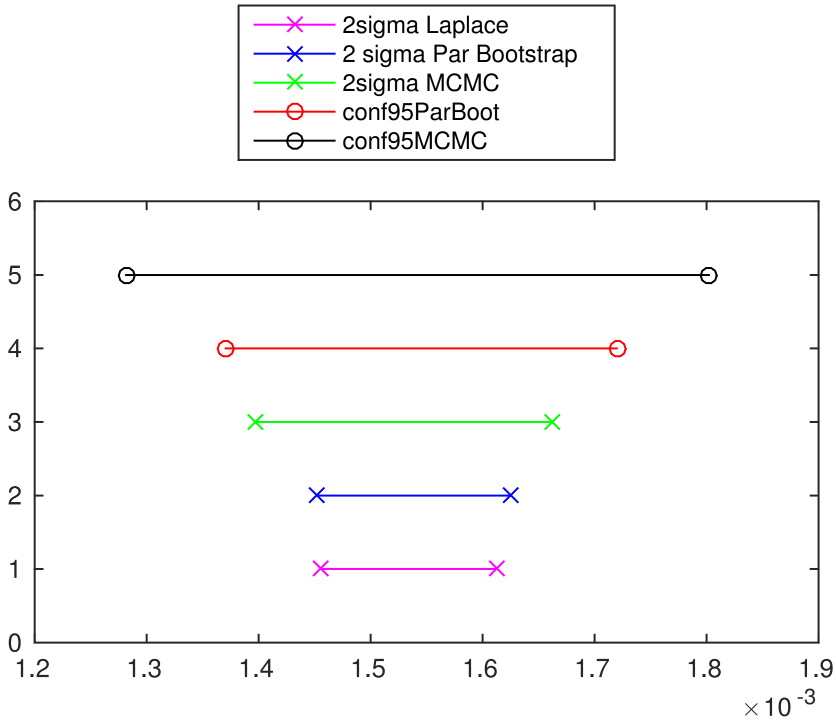


Figure 10: 2σ and 95% confidence intervals on parameter d using three different methods: parametric bootstrap, MCMC and Laplace. Voxel used was (52,62,25)

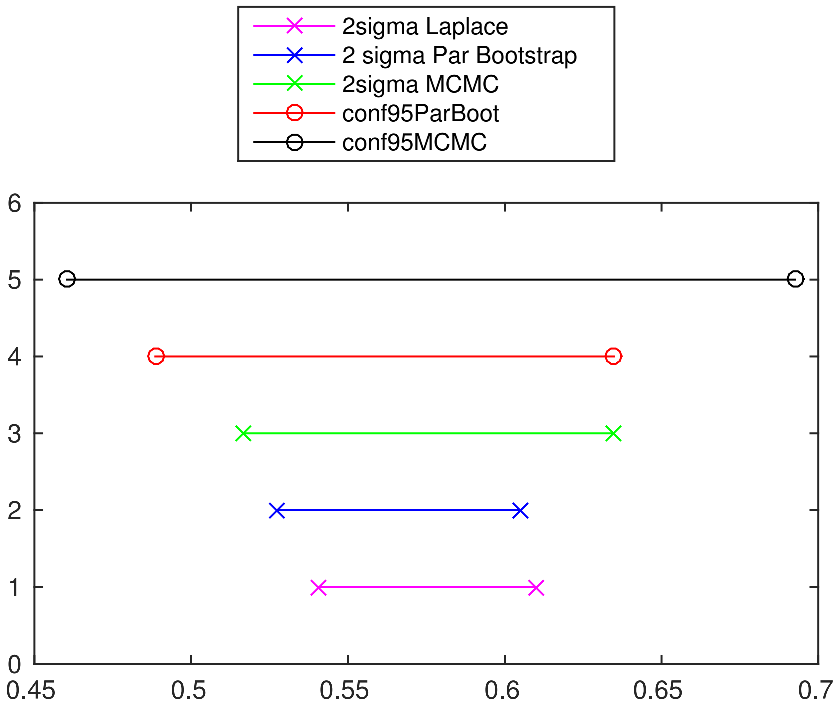


Figure 11: 2σ and 95% confidence intervals on parameter f using three different methods: parametric bootstrap, MCMC and Laplace. Voxel used was (52,62,25)