Quantitative and Functional Imaging

BME 4420/7450

Project #1

**Mapping T1 and T2 relaxation times in MRI**

This project deals with mapping NMR properties of tissues using image data. As in all the projects, **you are free to get your results in some other way—these procedures are just one (not necessarily optimal) method**. Matlab commands are given in *italics* for easy reference. Use *help <command>* (for example, *help squeeze*) or the Matlab Help pages for more details on any Matlab function.

# Part I: Mapping T2 relaxation times

1. Load the data file proj1aData\_QFI.mat into your Matlab workspace with

*load(‘proj1aData\_QFI’)*

(assuming the directory holding this file is on your Matlab search path). There are two arrays in the file:

image\_3d A 3D array of (**magnitude) image values**. The indices of the array are (image row, image col, time). The images are axial slices through the brain.

te\_v A 1D array **of echo (measurement) time TEs**, in seconds.

The value of te\_v(n) is the time (after the tipping B1 pulse) that the data for the nth image were measured. The image data were acquired on a 3 Tesla scanner. You can get the number of elements in **each dimension of an array** using the *size* function:

*[nRows, nCols, nTe] = size(image\_3d);*

1. Display **all five images** (in one figure), and label each with the time it was acquired. Use the same intensity scaling for all images so you can easily see how image intensity changes with time. Get the **intensity scale from the first image**:

*figure*

*subplot(2, 3, 1)*

*imagesc(squeeze(image\_3d(:, :, 1)))*

***intLimits\_v = get(gca, ‘CLim’);***

*axis image*

*axis off*

*colormap(gray)*

*title([‘Echo time = ‘, num2str(te\_v(1)\*1000), ‘ ms’])*

and use the same scale for the subsequent images:

*for index = 2:nTE*

*subplot(2, 3, index)*

*imagesc(squeeze(image\_3d(:, :, index)), intLimits\_v)*

*axis image*

*axis off*

*title([‘Echo time = ‘, num2str(te\_v(index)\*1000), ‘ ms’])*

*end*

The second argument to *imagesc* **fixes the intensity scale**. The *gca* (get current axes) command gets the handle (identifier) of the current axes. The *get* command returns **properties of whatever is drawn in the axes** (the minimum and maximum image intensities in this case). Note that the argument of the *title* command is a string, so the numbers held in te\_v must be converted to strings (with *num2str*) before being passed to *title*.

1. Use **the magnitude of the first image to form a ‘binary mask’** matrix that has 1’s in pixels that are inside the head and 0’s outside the head.
2. For **each pixel** in the head, calculate the **time constant of the decaying signal**. Since the signal decays exponentially with time, the logarithm of the **signal is linear in *TE*:**



Fitting a line to the ***ln(S)*** vs. ***TE*** data, the slope of the line is ***–1/T2***. We can use Matlab’s polynomial fitting function, *polyfit (time, ln(S), 1)*. This finds the coefficients of the nth order polynomial that best fit the data (by minimizing the squared error in the fit). For our case, the order **is 1** (a line). For the pixel at each image position (row, col), calculate

*signal\_v = squeeze(image\_3d(row, col, :));*

*% Insert your code here to fit the natural log of the signal to a first order*

*% polynomial in TE:*

*coeff\_v = …;*

*slope = coeff\_v(1);*

*logS0 = coeff\_v(2); % Intercept.*

*t2 = -1 / slope;*

Store the results for *T2* and S0 in t2\_m and s0\_m, respectively. Use

*imagesc(t2\_m, [0, 0.2])*

to display the *T2* map with **values up to 0.2 seconds** (larger values are set to 0.2 s to show the smaller *T2* values in the brain more clearly). You may need to discard some data that violate assumptions of the model.

**Calculate the residuals of the fit: for each pixel in the mask**,

calculate the norm (square root of the sum of squares) of the difference between the measured signal and the model, based on your estimates of ***logS0*** and ***slope*** above.

T2\_m(row,col)

signal\_v = squeeze(image\_3d(row, col, :));

ln(S) – ln(Ssignal)

1. Make a new figure with **two subplots**. In the first show the T2 map:

*% View fit:*

*figure*

*subplot(1, 2, 1)*

*t2Max = max(t2\_m(:));*

*red0\_m = t2\_m/t2Max;*

*green0\_m = t2\_m/t2Max;*

*blue0\_m = t2\_m/t2Max;*

*color\_3d = cat(3, red0\_m, green0\_m, blue0\_m);*

*image(color\_3d)*

*axis image*

*title('T2 map: click on a point of interest')*

Construct a **loop that allows the user of your program to select an arbitrary number of pixels to examine**. For each selected pixel, plot the signal intensity versus echo time (TE) along with the best-fit curve - like in lecture slides (i.e., an exponential decay with time constant equal to **your estimate of T2** and value at TE = 0 equal to your estimated S0): **signal** vs **TE**

*nPoints = 50;*

*contFlag = 1;*

*while (contFlag == 1)*

*% Get position of mouse-click on image:*

*[x, y] = ginput(1);*

*row = round(y);*

*col = round(x);*

*% Exit loop if the mouse click is outside the image:*

*if (row < 1 | row > nRows | col < 1 | col > nCols)*

*contFlag = 0;*

*continue*

*end*

*red\_m = red0\_m;*

*green\_m = green0\_m;*

*blue\_m = blue0\_m;*

*% Show the position of the pixel in red:*

*red\_m(row, col) = 1;*

*green\_m(row, col) = 0;*

*blue\_m(row, col) = 0;*

*color\_3d = cat(3, red\_m, green\_m, blue\_m);*

*image(color\_3d)*

*axis image*

*title('T2 map: click on a point of interest')*

*% Show fit:*

*t2 = t2\_m(row, col);*

*s0 = s0\_m(row, col);*

*s\_v = squeeze(image\_3d(row, col, :));*

*% Insert your code here to create an array of nPoints TE values from*

*% the minimum to maximum te:*

*teFit\_v = …;*

*% Insert your code here to find the corresponding signal at each TE,*

*% using your estimates of T2 and S0:*

*sFit\_v = …;*

*subplot(1, 2, 2)*

*plot(teFit\_v, sFit\_v, ':', te\_v, s\_v, '+')*

*title(['At row = ', num2str(row), ', col = ', num2str(col), ': T2 = ', ...*

*num2str(t2\*1000), 'ms'])*

*end*

Use this to **compare your model (shown by the smooth curve)** to the **data.** How well do they agree? Compare *T2* values in the gray matter, white matter and cerebrospinal fluid (CSF)—**how different are these?**

1. Display the *T2* data as **a gray scale map in a new figure**. Use the Data Cursor to show the (x, y) coordinate of the cursor on the image along with the***T2* value** for that pixel.
2. Identify the **gray matter and white matter** in the first (i.e., shortest TE) image (you can find examples on web-based resources such as Google Images and Wikipedia).

1. Use the *roipoly* command to draw a region of interest (ROI) in each tissue (one ROI for , another for **white matter**). Make the **regions as large as you can without including the other tissue type**. Make a histogram of *T2* values in each region and draw the ROI that the histogram represents. Find the mean and standard deviation of *T2* for gray and white matter.

# Part II: Mapping T1 relaxation times

1. Load the data file proj1bData\_QFI.mat into your Matlab workspace with

*load(‘proj1bData\_QFI’)*

(assuming the directory holding this file is on your Matlab search path). There are three arrays in the file:

irImage\_3d A 3D array of Mz values from an inversion recovery experiment. The indices of the array are (image row, image col, time). The images are coronal slices through the brain.

m0\_m A 2D array of equilibrium magnetization values.

ti\_v A 1D array of inversion times, in seconds.

The image data were acquired on a 3 Tesla scanner. The time ti\_v(n) is the time (after the inversion B1 pulse) at which Mz is measured for the image on the nth page of irImage\_3d. Ideally, the value of Mz depends on inversion time according to the relation



A simple (although not the most robust) way to find *T*1 is to linearize the dependence on the time ratio:

 --- find 1/T1

Plotting the expression on the right-hand side of the equation versus TI should **produce a straight line**. The slope of the line is **1/T1**. Image noise produces errors in the measurements of Mz (and M0), so in practice the fit to a straight line will not be perfect. In addition, the relations above neglect some factors that could be important, for example *imperfect inversion (rotation of the magnetization by an angle not equal to exactly 180 degrees).*

1. Display all inversion recovery images in one figure (using the same intensity scaling for each subplot to highlight how the images vary with inversion time). Make sure both **positive and negative values of Mz are displayed in the images**.
2. Estimate the T1 value for each pixel in the head and display the T1 map for the entire slice (you may need to discard some data that violate the assumption that , due to image noise). Also, calculate the residual of the fit (the norm of the difference between the **measured and modeled Mz**). Display your **T1** and **residual** maps.
3. **Add a loop to allow the user of your program to show the T1 fit for an arbitrary number of individual pixels.** For each selected pixel, plot the measured Mz versus TI along with the modeled curve using your estimate of T1. Examine several pixels. Do the estimated (modeled) recovery curves match the measured data well? Is the fit better for gray or white matter?
4. Display the **T1 map in a new figure.** Use *roipoly* to define regions of interest in the gray matter and white matter. **Plot histograms of the T1 distributions** in gray and white matter and show the ROIs used **to extract the histogram data. Find the mean and standard deviation of T1 for gray and white matter.**

# Questions

1. The large, bright “X” shaped structure in the center of the brain in part 1 is part of the ventricular system. The ventricles are cavities in the brain filled with cerebral spinal fluid (CSF). Based on the signal decay in the five T2–weighted images, does the CSF have longer or shorter T2 relaxation time than brain tissue? (**Explain using qualitative observations**, not your fitting results).
2. What is a typical T2 value for brain tissue? How **much variation** do you see over the brain?
3. Are the residuals of the fit uniform in the head? If not, where are they larger and where **are they smaller and what might cause the variation**?
4. How well can you distinguish between gray matter and white matter on the basis of T2 values in this map? What is the contrast in T2 values between gray and white matter (i.e., mean T2 for the gray matter ROI minus the mean T2 for the white matter ROI)? If the ‘noise’ in this measurement is the standard deviation of the difference between T2 values, , what is the contrast-to-noise ratio, CNR, between gray and white matter in the T2 map? Use the **propagation of errors t**o find an expression for in terms of and
5. Find the lateral ventricles in the inversion recovery images. Based on the Mz recovery as a function of TI, does CSF have longer or shorter T1 relaxation time than brain tissue? (Again, explain using your qualitative observations, not the fitting results).
6. **What is a typical T1 value for brain tissue? How much variation do you observe over the brain?**
7. Do the estimated (modeled) recovery **curves match the measured data well? Are the residuals similar for different tissue types**?
8. How well can you distinguish between gray matter and **white matter on the basis of T1? What is the contrast in T1 values between gray and white matter? If the ‘noise’ in this measurement is the standard deviation of T1 values around the means, what is the contrast-to-noise ratio, CNR, between gray and white matter in the T1 map?**
9. Which relaxation time, T1 or T2, **provides the highest CNR between gray and white matter? If you had to classify each pixel in the brain as either gray or white matter (ignoring CSF), which relaxation time map would you use**?
10. Ideally, **the residual maps reflect only random noise. Do you see evidence for non-random errors in the T1 or T2 fits?**

# Assignment

Create a Word document that includes

1. All five T2–weighted images in part 1 (labeled with the echo time).

Graphical user interface

Description automatically generated with low confidence

1. The T2 **map. Use the *colorbar* command to add a scale.**
2. A **residual map for the T2** fit.
3. An example of a **T2 relaxation curve (best fit and measurements) for a gray matter pixel and a white matter pixel.**
4. The **T2 histograms for gray matter and white matter. Remember to include units**. Also show an image with the regions of interest for each tissue type.
5. All inversion recovery **images (labeled with inversion time).**
6. The T1 map. Use **the *colorbar* command to add a scale.**
7. The **residual map for the T1 fits.**
8. An example of a T1 **relaxation curve (best fit and measurements) for a gray matter pixel and a white matter pixel.**
9. The T1 **histograms for gray matter and white matter (please** include units). Show the regions of **interest you defined for each tissue** type.
10. Your **answers to the questions above**.
11. Your Matlab code.

Please save your report as document named “Project1\_” followed by your name(s) and submit it via Brightspace. Reports are due on **Thursday, October 6**.