Finite Sample Properties of the EM Algorithm Applied to Transmission Tomography

Jimmy Hickey, Alvin Sheng and Eric Yanchenko

Abstract

We successfully implement an EM algorithm on simulated CT scans. After showing the mathematical derivations and framework of the EM algorithm, we demonstrate its effectiveness on images of different patterns and sizes. We show that as the number of projection angles increases, the scan greatly improves in quality, whereas larger images are harder to capture accurately.

Keywords: EM Algorithm, Transmission Tomography, Monte Carlo Simulation. Dennis Boos

1 Introduction

Transmission tomography refers to imaging cross sections of an object via a penetrating wave. By observing how the wave is attenuated as it travels through the object, one can infer the density of the object along the path of the projection. One way to construct the image of the cross section is through Fourier analysis, which is fast but do not account for the variability of the attenuation process. Another method to construct the image is through statistical methods, which can give better reconstructions with fewer projections (Lange, 2010). For our project, we focus on the Expectation-Maximization algorithm.

One example of transmission tomography is computerized tomography (CT) scans, which will serve as the basis of our simulation studies. In a CT scan, projections are streams of X-ray photons. Many projections are made at different angles around a patient, which can then be combined to reconstruct a detailed cross section of the patient. This medical imaging technique is often used to examine internal injuries, diagnose disease, or plan medical, surgical, or radiation treatment (Cli, 2020).

2 Methodology

2.1 CT Scans

We consider 3 different sections of a scan, shown in Figure 2.

1. The space outside of the scan, shown in gray. This will not be estimated.
2. The space outside the patient but inside the scan, shown in purple. The true θ value for these pixels is 0, but we still need to estimate them.
3. The space inside the patient, shown in yellow. We need to estimate this.

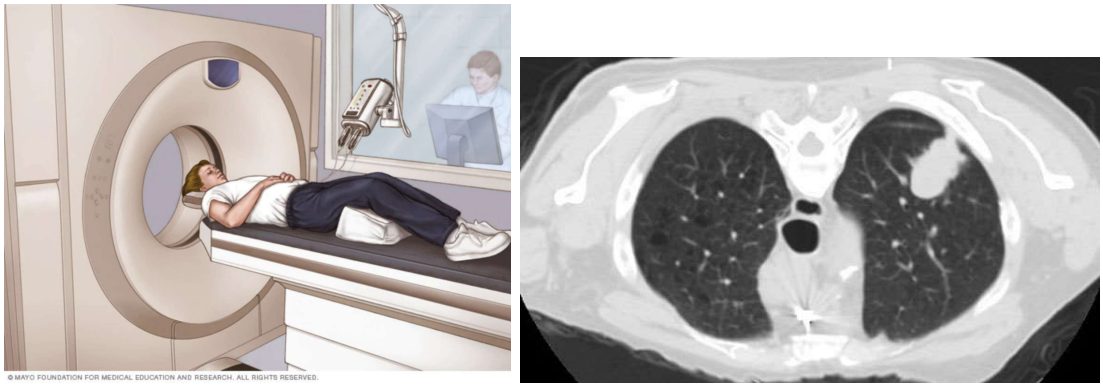


Figure 1: CT machine and scan. As the patient enters the machine, an X-ray source rotates around the patient in order to image his or her cross section (Cli, 2020). The right image is a CT scan of a patient's lungs (https://scienceblog.cancerresearchuk.org/2018/12/04/lung-cancer-screening-part-1-the-benefits-and-harms-according-to-clinical-trials/).

In Figure 3, X-ray projections are represented as red, blue, and green lines. For each projection, we track which of the purple and yellow pixels it intersects, and the final number of photons detected at the end of the projection.

To increase the sample size for the Monte Carlo simulation studies, we initially repeated horizontal or vertical projections or reproduced them backwards. However, there was no difference between doing a projection once or multiple times in terms of accuracy. Additionally, repeating the projections greatly increased computation time. We needed a new solution to vary the sample size. Instead of increasing the number of horizontal and vertical projections, we instead increased the number of possible angles at which the projections intersected the cross section. For convenience, we call the scenario with only horizontal and vertical projections "0 Angles" (Figure 3a), the scenario with additional diagonal projections "2 Angles" (Figure 3b), and the scenario with additional projections going at slopes of -2, -0.5, 0.5, and 2 "6 Angles" (Figure 3c).

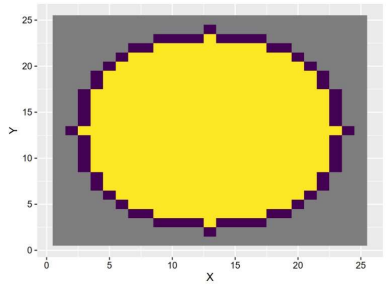


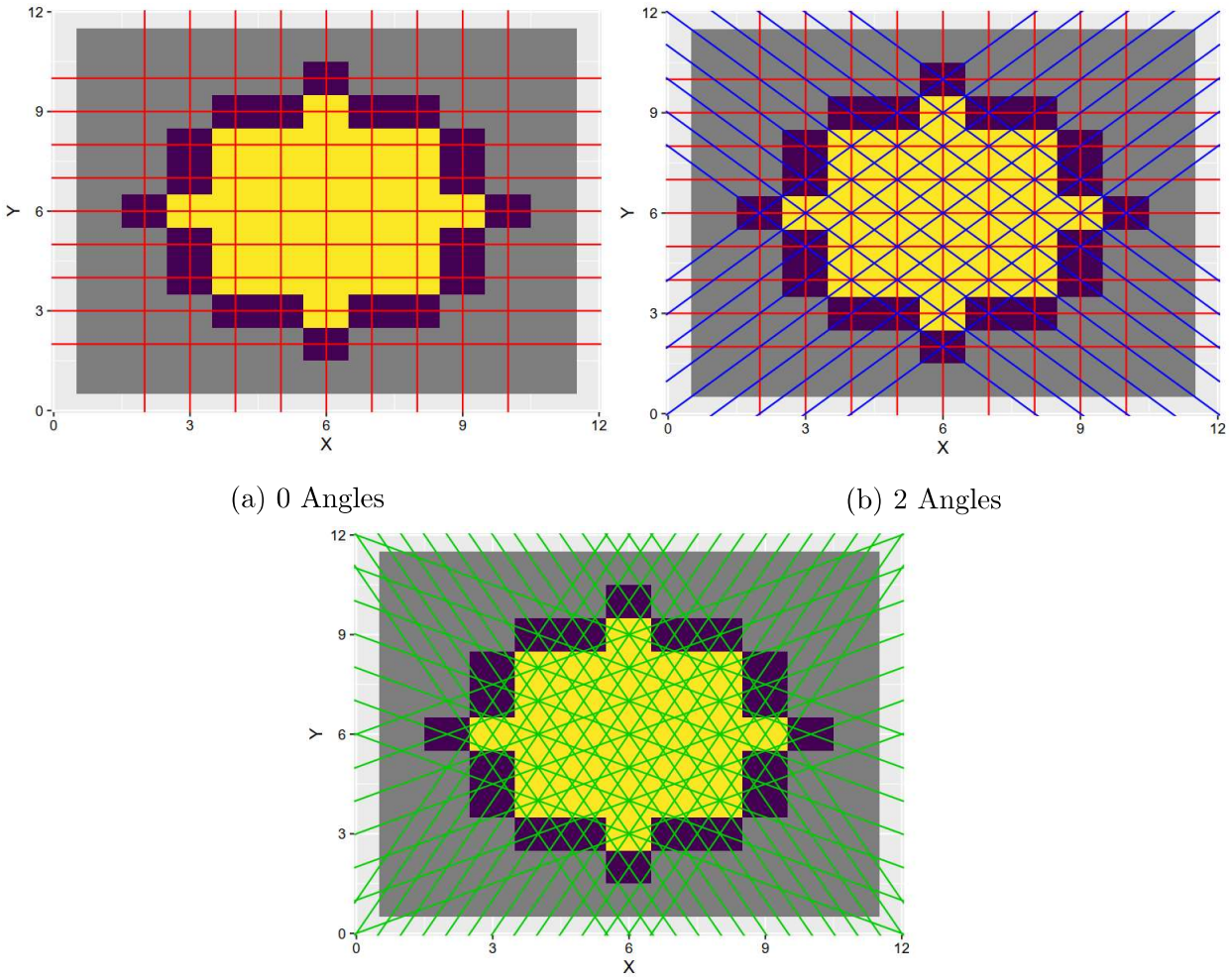
Figure 2: Scan Sections

Adding non-horizontal or vertical angles introduced a complication that the length of a projection within a pixel would depend on the angle. For instance, if a pixel has dimensions 1 by 1, then the length of a horizontal projection within the pixel would be 1, whereas the length of a diagonal projection within the pixel would be $\sqrt{2}$. The length of the projection within a pixel would affect how many photons would get blocked by the pixel, with longer lengths corresponding to more photons blocked. Thus, we also tracked the lengths of the projection within the pixels for each projection.

It is important to note that while the projections are independent, they are not identically distributed. Also notice that some projections may pass through more pixels than others.

2.2 EM Algorithm

The Expectation Maximization (EM) Algorithm is a popular approach to solving likelihood equations. The crux of the EM Algorithm is to consider some missing data $\boldsymbol{Z}$ along with the observed data $\boldsymbol{Y}$ (Boos et al., 2013). Combining these two we have the "complete" likelihood $L_C(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{Z})$. Without knowing $\boldsymbol{Z}$, we cannot compute $L_C(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{Z})$. Herein lies the "E" step of the EM Algorithm. We define a surrogate function



(a) 0 Angles

(b) 2 Angles

(c) 6 Angles

Figure 3: Three scenarios that change the number of projections by varying the number of possible non-horizontal or vertical angles. Horizontal, vertical and diagonal lines were omitted in the third graphic.

$$
\begin{aligned}
Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(\nu)}, \boldsymbol{Y}) &= E_{\boldsymbol{\theta}^{(\nu)}}\{\log L_C(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{Z})|\boldsymbol{Y}\} \\
&= \int \log L_C(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{z}) f_{\boldsymbol{Z}|\boldsymbol{Y}}\left(\boldsymbol{z}|\boldsymbol{Y}, \boldsymbol{\theta}^{(\nu)}\right) d\boldsymbol{z}.
\end{aligned}
$$

Here, $\boldsymbol{\theta}^{(\nu)}$ is the current estimate of $\boldsymbol{\theta}$ for iteration ν of the algorithm. Once we have calculated $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(\nu)}, \boldsymbol{Y})$, we can move on to the "M" step. Here we calculate $\boldsymbol{\theta}^{(\nu+1)}$ that maximizes $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(\nu)}, \boldsymbol{Y})$ with respect to $\boldsymbol{\theta}$. In the next section, we will go through the derivation of the surrogate function for this problem as well as outline what data is observed, what data is missing, and what parameters we are trying to estimate.

2.3 Mathematical Derivations

We will present the problem along with an illustrative example. We will then provide the surrogate function for the EM algorithm. See Lange (2010) and Lange et al. (1984) for more details.

As previously mentioned, CT scans fire projections of X-ray photons through a patient's cross section and measure the number of photons that pass through. Repeating the process at different angles, the density within a patient is estimated. We assume that the number of photons emitted by the source follows a Poisson distribution with rate parameter d. After that, each pixel that the projection passes through will absorb or deflect a certain amount of photons. The number of photons that pass through each pixel is modeled by a Binomial distribution related to the θ value of each pixel and the length of the projection within each pixel. Finally, the number of photons that pass through the entire cross section is observed at the end.

Let us look at a small example. Here we have only 3 pixels that the stream of photons will intercept. The source is being fired horizontally through the cross section, so we do not

need to worry about any additional complications from firing at an angle for this example.

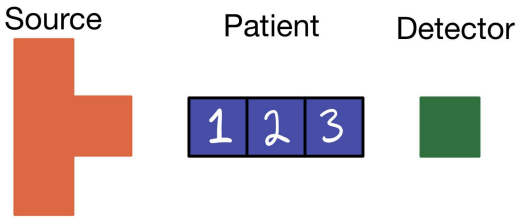


Figure 4: An example of a scan.

1. The source emits $X_1 \sim \text{Poisson}(d)$ photons.
2. Pixel 1 has a value of θ_1. When the photons from the emitter intercept, $X_2|X_1 \sim \text{Binomial}(n = X_1, p = e^{-\theta_1})$ photons pass through.
3. Pixel 2 has a value of θ_2 and $X_3|X_2 \sim \text{Binomial}(n = X_2, p = e^{-\theta_2})$ pass.
4. Pixel 3 has a value of θ_3 and $X_4|X_3 \sim \text{Binomial}(n = X_3, p = e^{-\theta_3})$ pass.
5. The detector then receives X_4 photons.

In the context of the EM Algorithm, the observed data is only the number of photons that make it all the way through the patient, $Y = X_4$. The unobserved data is how many photons are initially emitted by the source as well as how many photons pass through each pixel $\boldsymbol{Z} = (X_1, X_2, X_3)$. We fix the mean number of photons emitted from the source, d. We are trying to estimate the density of the pixels that projection i passes through from the source to the detector, $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$.

Now we can assemble our surrogate function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(\nu)}, \boldsymbol{Y})$. Here the index i represents the projections while the index j represents the pixels within that projection. Incorporating the distributional assumptions above, the "E" step can be shown to be

$$
\begin{aligned}
M_{ij} &= E\left(X_{ij}|Y_i = y_i, \boldsymbol{\theta}_n\right) \\
&= d\left(\exp\left(-\sum_{k\in S_{ij}} l_{ik}\theta_{nk}\right) - \exp\left(-\sum_{k} l_{ik}\theta_{nk}\right)\right) + y_i
\end{aligned}
$$

$$
\begin{aligned}
N_{ij} &= E\left(X_{ij'}|Y_i = y_i, \boldsymbol{\theta}_n\right) \\
&= d\left(\exp\left(-\sum_{k\in S_{ij}\cup\{j\}} l_{ik}\theta_{nk}\right) - \exp\left(-\sum_{k} l_{ik}\theta_{nk}\right)\right) + y_i,
\end{aligned}
$$

where j' is the pixel after pixel j along projection i, S_{ij} is the set of pixels between the source and pixel j (noninclusive), and ℓ_{ij} is the length of projection i within pixel j. Here M_{ij} is the number of incident photons entering pixel j along projection i and N_{ij} is the number of photons that pass through pixel j. Then our surrogate function is

$$
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{\nu}) = \sum_i \sum_j \Big[- N_{ij}\ell_{ij} + (M_{ij} - N_{ij}) \log(1 - e^{-\ell_{ij}\theta_j}) \Big].
$$

For the "M" step, we can maximize this function by taking partial derivatives with respect to θ_i and setting them equal to 0, as shown below. However, this results in transcendental equations that must be solved numerically. To solve the equations, we use Brent's method as implemented in the R function `uniroot`.

$$
\frac{\partial Q}{\partial \theta_j} = -\sum_i N_{ij}\ell_{ij} + \sum_i \frac{(M_{ij} - N_{ij})\ell_{ij}}{e^{\ell_{ij}\theta_j} - 1} \stackrel{\text{set}}{=} 0
$$

3 Monte Carlo Experiment

Since we don't have access to a real CT machine and patient, we generate the data ourselves using a simulation in `R`. There were several degrees of freedoms and metrics that we investigated in this experiment. First, we chose the number of Monte Carlo replicates to be $N = 10$ for the sake of computational time and arbitrarily set $d = 10^9$. We considered several true $\boldsymbol{\theta}$'s, including designs with three circles, checkered patterns and some non-physical but fun patterns. We varied the size of $\boldsymbol{\theta}$ (and therefore the image) by adjusting the radius of the cross section from 3 to 5 to 10. We treated the number of projections as the "sample size" to be varied by using zero, two, or six non-horizontal or vertical angles. Our metrics of interest were the Spectral Norm and Frobenius Norm (RMSE) of $|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}|$, as well as the number of iterations before the EM algorithm converged. We used the change in iterate values as our stopping condition. Specifically, we stopped if $\sum_{j=1}^{n} \left(\boldsymbol{\theta}_j^{(\nu+1)} - \boldsymbol{\theta}_j^{(\nu)}\right)^2 < 0.0001$.

4 Results

The next example will illustrate similar scan patterns over different radii. In the images, pixels with higher θ values are represented by lighter blue and yellow. This symbolizes an area of a patient with a higher density that is less likely to allow photons to transfer through.

We will examine the results for the three circle $\boldsymbol{\theta}$ patterns in depth. Our results for other patterns are similar, so we have placed them in Appendix A.

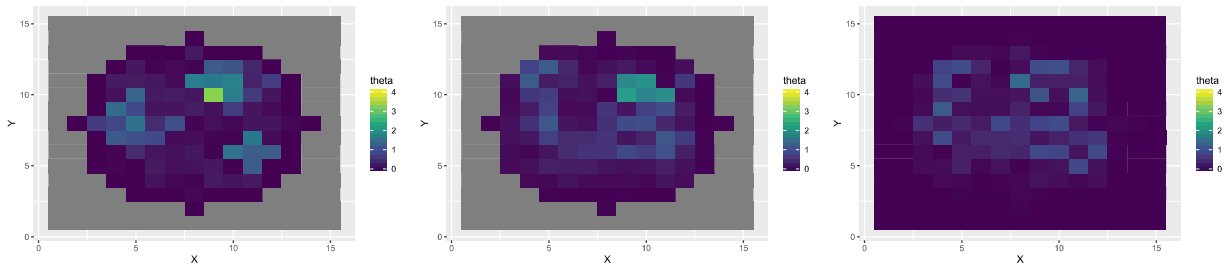


Figure 5: True theta, estimated theta, absolute difference for three circles. Radius 5 and 0 angles.

4.1 Three Circles

There are several clear trends that we can see from this example. First, looking at the plot with zero extra angles, we can hardly make out the three circles. As the number of angles increases, however, from zero to two to six, the clarity of the image greatly increase (Figures 5, 6, 7). This is also reflected in Table 1 as both the Spectral and Frobenius norm decrease as the number of angles increases. Recall that increasing the number of angles is our way of increasing sample size, so it makes sense that as the sample size increases, the quality of the images also increases.

Another trend that we see from Table 1 is that as the radius of the image increases, so too does the error. This makes sense because as the image size increases, we are working with more parameters which makes the inference more difficult. A large image also requires more iterations to converge which again makes intuitive sense.

Both of these trends continue in the other images which we include in the Appendix.

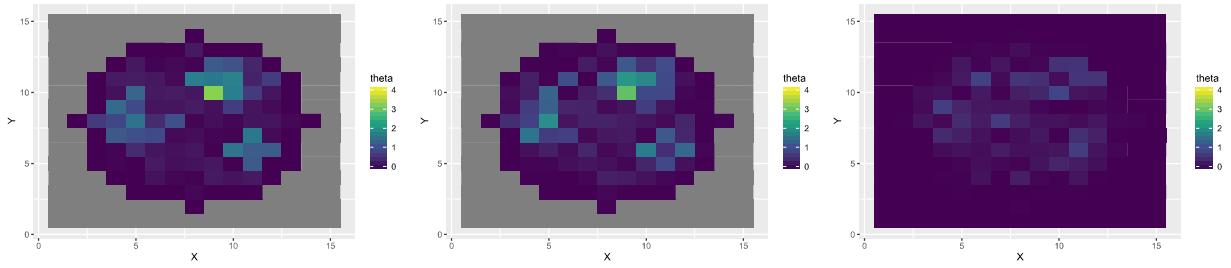


Figure 6: True theta, estimated theta, absolute difference for three circles. Radius 5 and 2 angles.

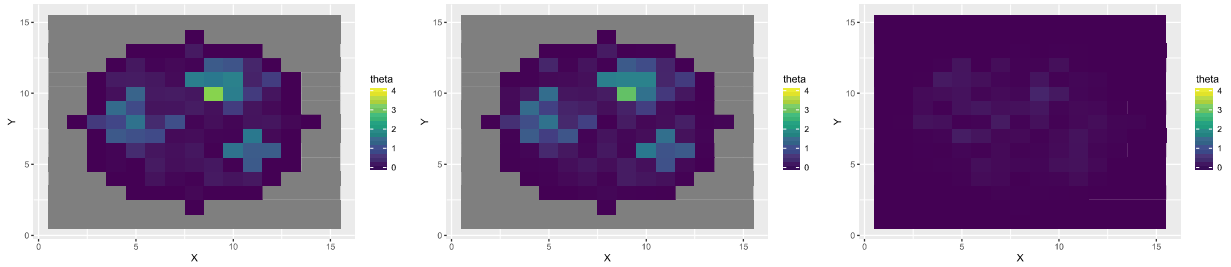


Figure 7: True theta, estimated theta, absolute difference for three circles. Radius 5 and 6 angles.

Radius	Angles	RMSE (0.09)	Spec. Norm (0.7)	Iterations
3	0	0.43	2.7	34 (9)
	2	0.09	0.5	30 (5)
	6	0.06	0.3	30 (5)
5	0	0.51	3.9	50 (10)
	2	0.31	2.1	90 (40)
	6	0.13	0.9	100 (20)
10	0	0.63	7.5	300 (100)
	2	0.48	5.2	210 (40)
	6	0.38	3.9	250 (20)

Table 1: Monte Carlo Simulation Results for Three Circles with $N = 10$. Maximum standard error in () at top of each column.

5 Conclusion

There are a variety of areas to explore that could improve effectiveness and efficiency of this algorithm. The most drastic performance improvement that we made was adding angles to projections. We consider a very limited number of angles and leave the effect of further increasing the number of angles as future work.

There are a lot of elements that can be varied when generating data for this problem. It would be useful to determine the effects on error and computation time for changing parameters such as N, the radius, and the number of angles. Further it would be interesting to explore interactions between these. For example, a scan with a smaller radius may require fewer angles to provide accurate estimates whereas a scan with a larger radius may require many angles to produce useful results.

Another avenue of future work is increasing the resolution of the scan. When we changed the radius in our simulation studies, we were changing the physical size of the scan. However, if we scaled the pixels down at the same time, i.e. made the dimensions of the pixels smaller, then the increase in the radius could be re-interpreted as the increase in the resolution of the scan. That is, an increase in radius would not necessarily mean that the actual scan cross section is increasing in physical area but rather that the same area is being represented by more pixels.

Further, Lange argues that the MM Algorithm is more computationally efficient and has a simpler mathematical derivation than the EM algorithm for this problem (Lange, 2010). A comparison study between the theoretical and empirical results of these methods could help formulate future study in the area.

References

(2020, February). *CT scan*. Mayo Clinic.

Boos, D. D., L. A. Stefanski, et al. (2013). *Essential statistical inference*. Springer.

Lange, K. (2010). *Numerical analysis for statisticians*. Springer Science & Business Media.

Lange, K., R. Carson, et al. (1984). Em reconstruction algorithms for emission and transmission tomography. *J. Comput. Assist. Tomogr 8*(2), 306–316.

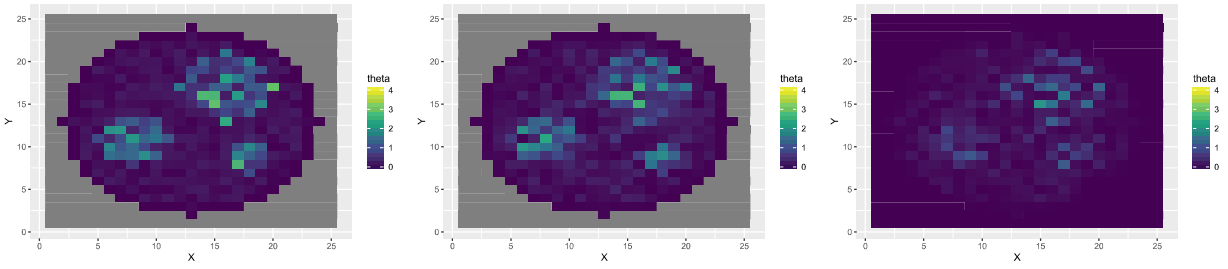


Figure 8: True theta, estimated theta, absolute difference for three circles. Radius 10 and 6 angles

Appendix A: Additional Results

Three Circle

Figure 8 shows additional three circle plots for radius 10 and 6 angles.

One Circle

A $\boldsymbol{\theta}$ pattern with one circle would represent a single area of high density within a patient. This is a fairly simple example. For this reason, we decided to explore the effect of increasing the Monte Carlo replicate size from $N = 10$ to $N = 100$. We only present the results for a radius of 3 since increasing the radius greatly increases computation time. See Figure 9 and Table 2.

Checkered

Here we used a checkered pattern. While this may not be a particularly realistic scan, we suspected that it would computationally challenging given that there are a lot of high θ

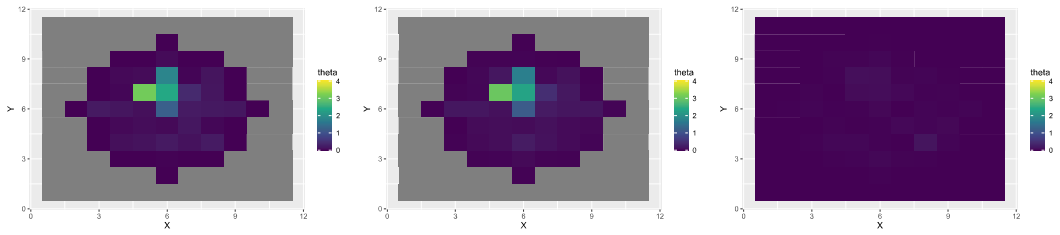


Figure 9: True theta, estimated theta, absolute difference for one circle. Radius 3 and 6 angles

Radius	Angles	RMSE (0.01)	Spec. Norm (0.10)	Iterations (1)
3	0	0.39	2.6	25
	2	0.09	0.48	30
	6	0.06	0.28	35

Table 2: Monte Carlo Simulation Results for One Circle with $N = 100$. Maximum standard error in () at top of each column.

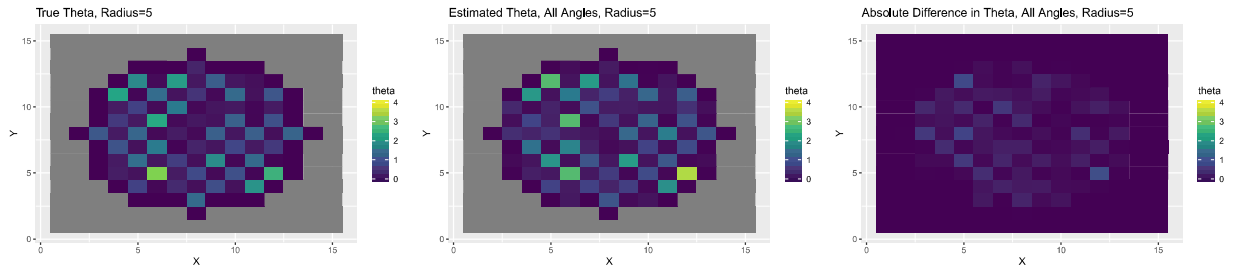


Figure 10: True theta, estimated theta, absolute difference for checker pattern. Radius 5 and 6 angles

values. Here we used $N = 10$. See Figure 10 and Table 3.

Fun

We were not able to secure funds to dissect cadavers to model what a real patient cross section may look like. As such, we took some artistic liberties with the next few true $\boldsymbol{\theta}$ matrices.

Face

This pattern is comprised of two circles and a horizontal line across the bottom, making the apathetic face present in the images. Here we used $N = 10$ Monte Carlo simulations. See Figure 11 and Table 4.

Radius	Angles	RMSE (0.01)	Spec. Norm (0.7)	Iterations (5)
3	0	0.75	3.6	34
	2	0.35	1.8	74
	6	0.10	0.6	56
5	0	0.75	5.6	100
	2	0.61	4.2	160
	6	0.27	1.7	170

Table 3: Monte Carlo Simulation Results for Checkered Pattern with $N = 10$. Maximum standard error in () at top of each column.

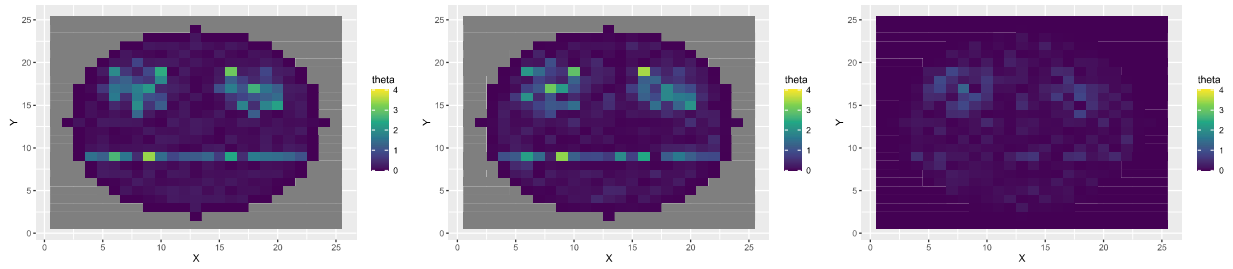


Figure 11: True theta, estimated theta, absolute difference for face. Radius 10 and 6 angles.

Radius	Angles	RMSE (< 0.01)	Spec. Norm (< 0.1)	Iterations (10)
3	0	0.62	3.7	80
	2	0.10	0.5	50
	6	0.05	0.3	53
5	0	0.61	4.6	150
	2	0.27	1.9	110
	6	0.12	0.9	97
10	0	0.65	9.5	250
	2	0.37	4.1	250
	6	0.24	2.9	220

Table 4: Monte Carlo Simulation Results for the Face Pattern with $N = 100$. Maximum standard error in () at top of each column.

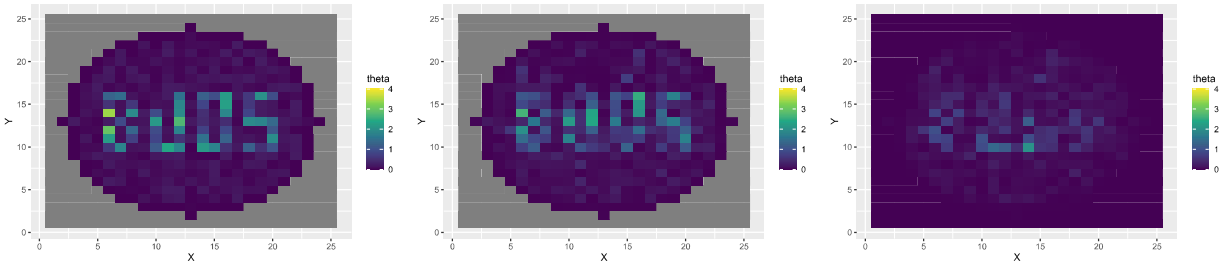


Figure 12: True theta, estimated theta, absolute difference for Boos. Radius 10 and 6 angles

Radius	Angles	RMSE (< 0.01)	Spec. Norm (0.1)	Iterations (3)
10	0	0.62	8.9	250
	2	0.40	4.8	270
	6	0.30	3.6	280

Table 5: Monte Carlo Simulation Results for Boos Pattern with $N = 10$. Maximum standard error in () at top of each column.

Boos

A tribute to one of the greats, Dennis Boos. Again with Monte Carlo replication $N = 10$. See Figure 12 and Table 5.