Nonlinear Optimization Problem: Minimizing the Mean Square Error Objective Function

Alan Perez

May 20, 2021

Abstract

Newton's method is one of the most popular iterative methods for solving unconstrained optimization problems. It is a classical method that has some features that are found in several other iterative methods and other features which should be avoided in practical applications. The reasons for this being its sensitivity to both (1) the initial guess and (2) the "niceness" of the graph which we define as how wobbly the function is. We will compare this method to two other methods that are closely related to one another: the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method and the Davidson-Fletcher-Powell (DFP) method. We will use all three of these methods to solve a data-fitting problem and present analytical and numerical results to show which method was more successful in approximating the underlying function.

1 Introduction

This report will look at a data-fitting problem and discuss the methodology and results of five experiments utilizing Newton's method (NM), the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method, and, exclusively for experiment 5, the Davidson-Fletcher-Powell (DFP) method. It will also present the results of the five experiments in the forms of tables and plots.

2 Problem Description

It is common for an experiment in a laboratory setting to produce data that roughly takes the shape of an already known function, such as a sine wave or an exponential function. However, more often than not, this data contains noise that differentiates it from the function it resembles. The base problem is, given a set of noisy data that closely resembles a sine wave, to find the parameters so that the function

$$h(\boldsymbol{a}, \boldsymbol{x}) = a_0 + a_1 \sin(a_3 x + a_4)$$

best approximates the underlying true function producing the data. Here we define the parameters of interest as the DC component (offset), amplitude, frequency, and phase shift for $a_0, a_1, a_2,$ and a_3, respectively.

To find these parameters, we consider the general unconstrained problem:

$$\text{minimize} \quad f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$$
$$\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$$

where $f: \mathbb{R}^n \to \mathbb{R}$ is a smooth function and represents the mean square error function where $\boldsymbol{a}$ is the vector of parameters described above and the $\boldsymbol{x}$ and $\boldsymbol{y}$ vectors make up the data points (x_i, y_i). By finding a vector of parameters $\boldsymbol{a}$ that minimizes the function that describes the error between the actual data and the values produced by the function, we are able to determine a function that best fits the data.

The base problem was to solve for the parameters utilizing both Newton's method and the BFGS method and provide complementary analytical results in the form of plots and numerical results in the form of tables. As for the extension of the base problem, I also solved for the same parameters using the Davidson-Fletcher-Powell (DFP) method to determine if the DFP method is more than, less than, or equally accurate to the closely related BFGS method.

3 Mathematical Formulation

Base Problem

We are given to length k vectors $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix}$ and $\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix}$ forming the observed data points (x_i, y_i), with $i = 1, 2, \ldots, k$. As briefly mentioned before, we are interested in fitting a sinusoidal function to these data points by calculating a set of parameters $\boldsymbol{a} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$ where $a_0, a_1, a_2,$ and a_3 are the DC component, the amplitude, the frequency, and the phase shift, respectively. We find these values by minimizing the mean square error between the given data points and the values produced by the function, where the mean square error is given by

$$\frac{1}{K}\sum_{i=1}^{K}\big(y_i - h(\boldsymbol{a}, x_i)\big)^2$$

That is, we solve the nonlinear optimization problem described in the *Problem Description* section by finding $\boldsymbol{a}$ that minimizes the objective function

$$f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y}) = \frac{1}{K}\sum_{i=1}^{K}\big(y_i - h(\boldsymbol{a}, x_i)\big)^2.$$

To solve for the parameters that minimize the above objective function, we used slightly modified versions of NM and the BFGS method that allow for their application on a given set of data points. It should be noted that the following equations are written in general form and are not specific to this problem. For both methods, we implement a line search algorithm which allows us to find a t_k^* that minimizes the function

$$\phi(t) = f(\boldsymbol{x}^{(k)} - t_k \boldsymbol{r}).$$

The BFGS method already requires us to find the parameter t_k^* so we only implement the line search algorithm, however this parameter is new in Newton's method so we must implement it accordingly.

We introduce this step size parameter t_k^* in the update equation of Newton's method, yielding (again, in a general notation)

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + t_k^* \boldsymbol{q}^{(k)}$$

where $\boldsymbol{q}^{(k)}$ is the solution to the system

$$Hf\big(\boldsymbol{x}^{(k)}\big)\boldsymbol{q}^{(k)} = -\nabla f\big(\boldsymbol{x}^{(k)}\big).$$

That is,

$$\boldsymbol{q}^{(k)} = -\big[Hf\big(\boldsymbol{x}^{(k)}\big)\big]^{-1}\nabla f(\boldsymbol{x}^{(k)})$$

so that the final update equation in Newton's method becomes

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - t_k^*\big[Hf\big(\boldsymbol{x}^{(k)}\big)\big]^{-1}\nabla f(\boldsymbol{x}^{(k)})$$

Based on what was just covered, the following is the updated Newton's method algorithm.

Algorithm 1 Newton's method with Step Size Parameter t_k^*

Step 0 Let $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$ be an initial estimate of a critical point of $f(\boldsymbol{x})$. Let $\epsilon \in \mathbb{R}$ be a threshold for convergence. Initialize error estimate $e^{(0)} = \infty$. Let the number of iterations k = 0.

Step 1 If some stopping criteria are satisfied, then terminate otherwise go to **Step 2**. That is, while the error estimate is greater than 0 and k < max iterations go to **Step 2**.

Step 2

Step 2a Calculate the gradient and hessian of the function $f(\boldsymbol{x}^{(k)})$ for the current iteration. Also calculate the value $\boldsymbol{q}^{(k)} = -\left[Hf\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\nabla f\left(\boldsymbol{x}^{(k)}\right)$

Step 2b Set $\gamma = 0.95$, $t_k =$ initial guess, $t_k^* = t_k$, and $b = \frac{1}{2}\left(\nabla f\left(\boldsymbol{x}^{(k)}\right)\right)^T \boldsymbol{r}$ where the symbol T is the transpose and $\boldsymbol{r} = -\boldsymbol{q}^{(k)}$

Step 2c While $f\left(\boldsymbol{x}^{(k)}\right) - \boldsymbol{f}\left(x^{(k)} - t_k\boldsymbol{r}\right) < t_k b$ and $t_k > 10^{-10}$ proceed to **Step 2c1**, otherwise terminate and take value of t_k^* to **Step 2d**

Step 2c1 Set new t_k to be 0.95 times the previous t_k.

Step 2c2 if $f\left(\boldsymbol{x}^{(k)} - t_k\boldsymbol{r}\right) < f\left(\boldsymbol{x}^{(k)} - t_k^*\boldsymbol{r}\right)$, update t_k^* to current t_k

Step 2d Calculate the next vector in the sequence $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + t_K^*\boldsymbol{q}^{(k)}$

Step 2e Calculate the error approximation $e^{(k)} = \left\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right\|$

Step 2f Update the iteration number, *k*, and go to **Step 2**

As mentioned previously, the BFGS method already utilizes a step size parameter t_k^* in its algorithm and the only modification that was made to the algorithm was the implementation of the line search algorithm for said t_k^*.

Algorithm 2 The Broyden-Fletcher-Goldfarb-Shanno method

Step 0 Let $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$ be an initial estimate of a minimizer of $f(\boldsymbol{x})$. Let D_0 be an initial positive definite matrix. Let $\epsilon \in \mathbb{R}$ be a threshold for convergence. Initialize the error estimate $e^{(0)} = \infty$. Initialize the iteration number $k = 0$.

Step 1 While $e^{(k)} > \epsilon$ and $k <$ max iterations go to step 2, otherwise terminate

Step 2

Step 2a Calculate the gradient and hessian of $f\left(\boldsymbol{x}^{(k)}\right)$ for the current iteration. Also calculate $\boldsymbol{p}^{(k)} = D_k^{-1}\nabla f\left(\boldsymbol{x}^{(k)}\right)$

Step 2b Set $\gamma = 0.95$, $t_k =$ initial guess, $t_k^* = t_k$, and $b = \frac{1}{2}\left(\nabla f\left(\boldsymbol{x}^{(k)}\right)\right)^T \boldsymbol{r}$ where the symbol T is the transpose and $\boldsymbol{r} = \boldsymbol{p}^{(k)}$

Step 2c While $f\left(\boldsymbol{x}^{(k)}\right) - \boldsymbol{f}\left(x^{(k)} - t_k \boldsymbol{r}\right) < t_k b$ and $t_k > 10^{-10}$ proceed to **Step 2c1**, otherwise terminate and take value of t_k^* to **Step 2d**

Step 2c1 Set new t_k to be 0.95 times the previous t_k.

Step 2c2 if $f\left(\boldsymbol{x}^{(k)} - t_k \boldsymbol{r}\right) < f\left(\boldsymbol{x}^{(k)} - t_k^* \boldsymbol{r}\right)$, update t_k^* to current t_k

Step 2d Calculate the next vector in the sequence $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - t_K^* \boldsymbol{p}^{(k)}$

Step 2e Calculate the error approximation $e^{(k)} = \left\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right\|$

Step 2f Calculate $\boldsymbol{d}^{(k)} = \boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}$

Step 2g Calculate $\boldsymbol{y}^{(k)} = \nabla f\left(\boldsymbol{x}^{(k+1)}\right) - \nabla f\left(\boldsymbol{x}^{(k)}\right)$

Step 2h Calculate

$$D_{k+1} = D_k + \frac{\boldsymbol{y}^{(k)}\left(\boldsymbol{y}^{(k)}\right)^T}{\left(\boldsymbol{y}^{(k)}\right)^T \boldsymbol{d}^{(k)}} - \frac{\left(D_k \boldsymbol{d}^{(k)}\right)\left(D_k \boldsymbol{d}^{(k)}\right)^T}{\left(D_k \boldsymbol{d}^{(k)}\right)^T \boldsymbol{d}^{(k)}}$$

Step 2i Update the iteration number, *k*, and go to **Step 2**

It should be noted that in our calculations for this problem we set D_0 to be the identity matrix I_4 and also set our initial guesses for t_k for Newton's method and the BFGS method as $t_k = 1$ and $t_k = 5$, respectively. The final algorithm that will be explored in this report is the Davidson-Fletcher-Powell (DFP) method.

The DFP method is an algorithm that is closely related to the BFGS method and will be the focus of the extension portion of this report. The method will be implemented in its pure form with the only modification being the implementation of the line search algorithm.

Algorithm 3 The Davidson-Fletcher-Powell method

Step 0 Let $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$ be an initial estimate of a minimizer of $f\left(\boldsymbol{x}^{(k)}\right)$. Let E_0 be an initial positive definite matrix. Let $\epsilon \in \mathbb{R}$ be a threshold for convergence.

Step 1 While $e^{(k)} > \epsilon$ and $k <$ max iterations go to **Step 2**, otherwise terminate

Step 2

Step 2a Calculate the gradient and hessian of $f\left(\boldsymbol{x}^{(k)}\right)$ for the current iteration. Also calculate $\boldsymbol{p}^{(k)} = E_k \nabla f\left(\boldsymbol{x}^{(k)}\right)$

Step 2b Set $\gamma = 0.95$, $t_k =$ initial guess, $t_k^* = t_k$, and $b = \frac{1}{2}\left(\nabla f\left(\boldsymbol{x}^{(k)}\right)\right)^T \boldsymbol{r}$ where the symbol T is the transpose and $\boldsymbol{r} = \boldsymbol{p}^{(k)}$

Step 2c While $f\left(\boldsymbol{x}^{(k)}\right) - \boldsymbol{f}\left(x^{(k)} - t_k \boldsymbol{r}\right) < t_k b$ and $t_k > 10^{-10}$ proceed to **Step 2c1**, otherwise terminate and take value of t_k^* to **Step 2d**

Step 2c1 Set new t_k to be 0.95 times the previous t_k.

Step 2c2 if $f\left(\boldsymbol{x}^{(k)} - t_k \boldsymbol{r}\right) < f\left(\boldsymbol{x}^{(k)} - t_k^* \boldsymbol{r}\right)$, update t_k^* to current t_k

Step 2d Calculate the next vector in the sequence $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - t_K^* \boldsymbol{p}^{(k)}$

Step 2e Calculate the error approximation $e^{(k)} = \left\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right\|$

Step 2f Calculate $\boldsymbol{d}^{(k)} = \boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}$

Step 2g Calculate $\boldsymbol{y}^{(k)} = \nabla f\left(\boldsymbol{x}^{(k+1)}\right) - \nabla f\left(\boldsymbol{x}^{(k)}\right)$

Step 2h Calculate

$$E_{k+1} = E_k + \frac{\boldsymbol{d}^{(k)}\left(\boldsymbol{d}^{(k)}\right)^T}{\left(\boldsymbol{y}^{(k)}\right)^T \boldsymbol{d}^{(k)}} - \frac{\left(E_k \boldsymbol{y}^{(k)}\right)\left(E_k \boldsymbol{y}^{(k)}\right)^T}{\left(E_k \boldsymbol{y}^{(k)}\right)^T \boldsymbol{y}^{(k)}}$$

Step 2i Update the iteration number, *k*, and go to **Step 2d**

Similar to our application of the BFGS method, we will set our initial positive definite matrix E_0 to be the identity matrix I_4. For all three methods we will set the threshold of convergence $\epsilon = 10^{-6}$.

4 Experimental Results and Discussion

We consider a total of five experiments relating to the provided data, four of which are asked for directly in the prompt and the last one is an extension of the problem. For a brief overview, experiments 1, 2, and 3 explore how different initial guesses of the parameter vector affect the results of both Newton's method and the BFGS method. Experiment 4 considers several randomized initial guesses for the vector $\boldsymbol{a}$ and provides results supporting which method is more accurate and consistent between NM and BFGS. The last experiment, my personal extension, is similar to experiment 4, but instead uses the DFP method. We will now cover the procedures and findings in depth.

The first three experiments use columns 1, 2, and 3 for the initial guess $\boldsymbol{a}^0$:

$$\begin{bmatrix} 1.23249720 & 1.90064870 & 1.82428450 \\ 0.14844971 & 2.79581380 & 2.36446250 \\ 3.01837270 & 2.80402290 & 2.86998480 \\ 0.51600794 & 2.92119350 & 0.93243968 \end{bmatrix}$$

where column 1 is the initial guess $\boldsymbol{a}^{(0)}$ for the vector $\boldsymbol{a}$ of experiment 1, column 2 is the initial guess $\boldsymbol{a}^{(0)}$ for the vector $\boldsymbol{a}$ of experiment 2, and so on for experiment 3. As mentioned before, these first three experiments are more focused on showing the differences that can perpetuate from variations in the initial guess $\boldsymbol{a}^{(0)}$. In each experiment we take its corresponding column (initial guess) for the vector $\boldsymbol{a}$ and run both Newton's method and the BFGS method on it. After applying Algorithm 1 and Algorithm 2 (NM and BFGS) to our initial guess and receiving a sequence of parameters from each, we created an iterative fit plot. For each method, we plotted the data points provided using open red circle markers and then, for each iteration of the sequence, plotted the function $h(\boldsymbol{a}, x)$ using a solid green line and no markers using the $\boldsymbol{a}^{(k)}$ calculated that iteration and vector of x values from -5 to -10 with a sufficiently small increment, which I took to be 0.01 Lastly, we plotted the curve generated by evaluating $h(\boldsymbol{a}, x)$ using the final $\boldsymbol{a}^{(k)}$ calculated in the sequence (which we take to be the optimal solution for **a** that minimizes objective function) and the same x values used for plotting the lines in green.

Since we were also interested in the error estimations of both methods for experiments 1, 2 and 3, we created an error estimate vs. iteration plot which utilized a linear scale on the x-axis and a base-10 logarithmic scale on the y-axis. We used these scales to make the changes in error with respect to iteration more obvious.

Table 1: Final $\boldsymbol{a}^{(k)}$, corresponding k, and corresponding $f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$ of each Experiment

	Experiment 1		**Experiment 2**		**Experiment 3**	
	NM	BFGS	NM	BFGS	NM	BFGS
$\boldsymbol{a}^{(k)}$	$\begin{bmatrix} 1.9969 \\ -1.0061 \\ 2.9994 \\ -8.4193 \end{bmatrix}$	$\begin{bmatrix} 1.9969 \\ 1.0061 \\ 2.9994 \\ 1.0055 \end{bmatrix}$	$\begin{bmatrix} 1.9689 \\ 0.0001 \\ 2.8048 \\ 3.0067 \end{bmatrix}$	$\begin{bmatrix} 1.9969 \\ 1.0061 \\ -2.9994 \\ 2.1361 \end{bmatrix}$	$\begin{bmatrix} 1.9969 \\ 1.0061 \\ 2.9994 \\ 1.0055 \end{bmatrix}$	$\begin{bmatrix} 1.9709 \\ -0.1385 \\ 12.7569 \\ 1.4845 \end{bmatrix}$
k	7	9	4	14	5	16
$f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$	0.01016	0.01016	0.50009	0.01016	0.01016	0.49013

From Table 1, it is clear that the number of iterations it takes for the BFGS to terminate is higher than the number of iterations for Newton's method in all three experiments. Also, four out of the six values for the final objective function value were approximately 0.01016 (we say approximately to account for roundoff error). Lastly, the final $\boldsymbol{a}^{(k)}$ vectors show that the amplitude, frequency, and phase shift parameters vary regularly while the DC component is consistently around the value 1.9969.

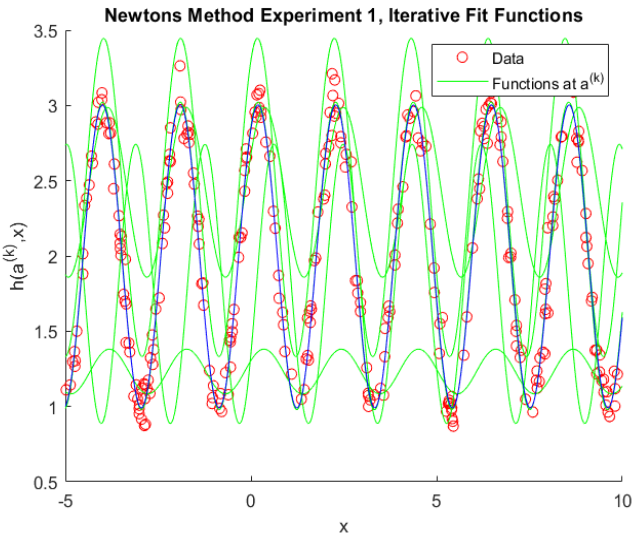

Figure 1. Plot of data values in red and curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using $\boldsymbol{a}^{(k)}$ calculated using NM that iteration with initial guess $\boldsymbol{a}^{(0)}$ corresponding with Experiment 1; Best $h(\boldsymbol{a}, \boldsymbol{x})$ using final $\boldsymbol{a}^{(k)}$ is in blue

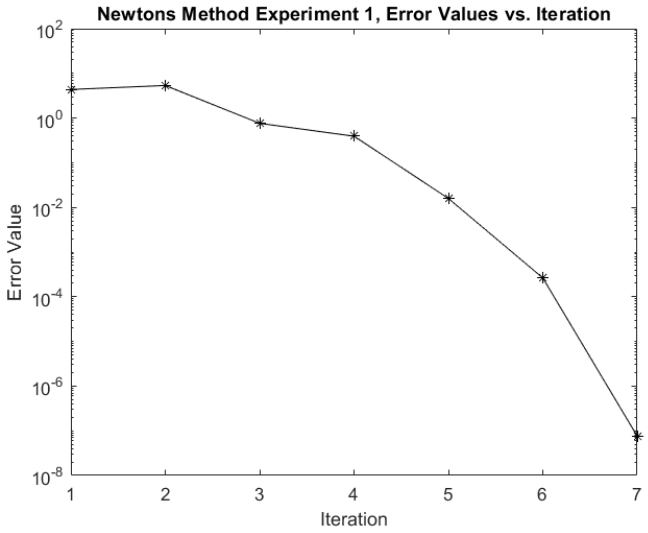

Figure 2. Plot of error estimate values vs. iteration for Newton's method

Figure 1 shows that after finding 7 terms of the Newton's method sequence—resulting in 7 unique $\boldsymbol{a}^{(k)}$ vectors—with the corresponding initial guess for Experiment 1, we achieve a vector of parameters $\boldsymbol{a}^{(k)}$, namely the 7th term of the sequence, for the function $h(\boldsymbol{a}, \boldsymbol{x}) = a_0 + a_1 \sin(a_2 x + a_3)$ that provides a good fit for the data (in blue). Interestingly, even though our initial guess had an amplitude that was far different from the final amplitude, the final set of parameters minimized the objective function $f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$ and created a function that fit the data well. Figure 2 shows a very consistent decrease in error value as the iterations increase which is an indication of our parameters getting closer in value each iteration

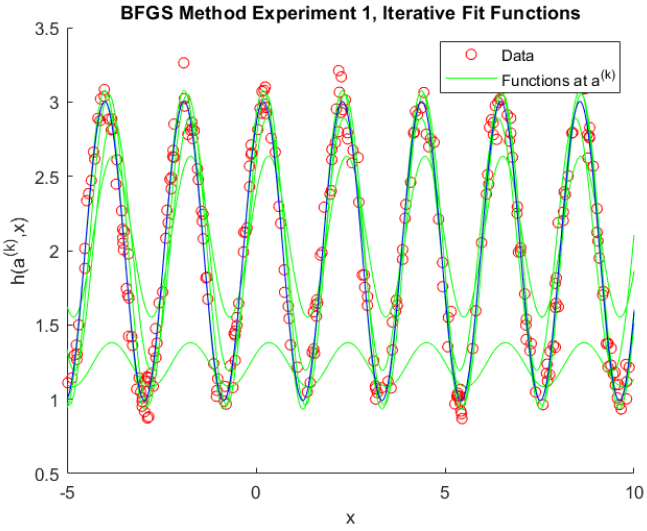

Figure 3. Plot of data values in red and curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using $\boldsymbol{a}^{(k)}$ calculated using BFGS that iteration with initial guess $\boldsymbol{a}^{(0)}$ corresponding with Experiment 1; Best $h(\boldsymbol{a}, \boldsymbol{x})$ using final $\boldsymbol{a}^{(k)}$ is in blue

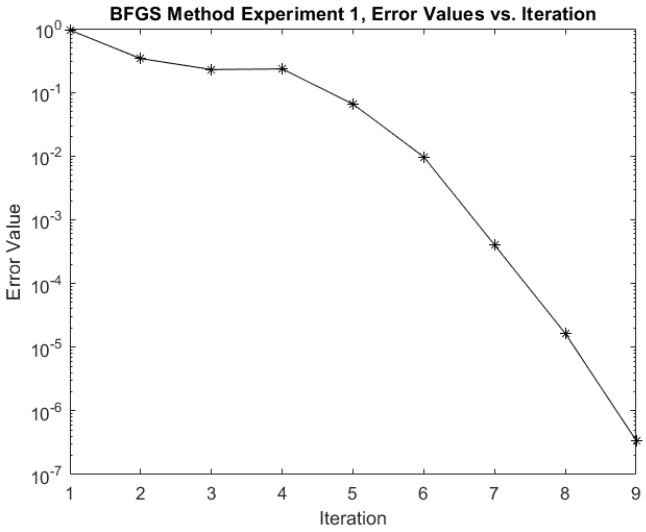

Figure 4. Plot of error estimate values vs. Iteration for the BFGS method

Figure 3 also demonstrates that after finding 9 terms of the BFGS sequence with the corresponding initial guess for Experiment 1, we achieve a vector $\boldsymbol{a}^{(k)}$ for $h(\boldsymbol{a}, \boldsymbol{x})$ that provides a good fit for the data. As seen in Figure 4, our error values remain about the same for the first 4 points and then begin to drop rapidly with the following iterations. In all, it seems that both the Newton's method and the BFGS method handled the initial guess vector $\boldsymbol{a}^{(0)}$ corresponding to Experiment 1 very well and resulted in some excellent fits for the data.

In Experiment 2 with its corresponding initial guess $\boldsymbol{a}^{(0)}$, we get 8 terms from Newton's method. Figure 5 shows the function $h(\boldsymbol{a}, \boldsymbol{x}) = a_0 + a_1 \, sin \, (a_2 x + a_3)$ plotted for each of those terms and, more importantly, shows that the final parameters $\boldsymbol{a}^{(k)}$ do not product a plot $h(\boldsymbol{a}, \boldsymbol{x})$ the data well. A bad quality of Newton's method is that in order for our sequence to converge to a solution $\boldsymbol{a}^*$, the initial guess $\boldsymbol{a}^{(0)}$ must be sufficiently close to $\boldsymbol{a}^*$ and the graph must not be too "wobbly." Although not labeled on the figure, the sinusoidal wave with the largest amplitude is our function $h(\boldsymbol{a}, \boldsymbol{x})$ plotted with initial guess $\boldsymbol{a}^{(0)}$ corresponding to Experiment 2. It is hard to judge the quality of our initial guesses directly, but by plotting it we can see that our initial guess has an amplitude, frequency, and phase shift that does not line up well with our data. As for

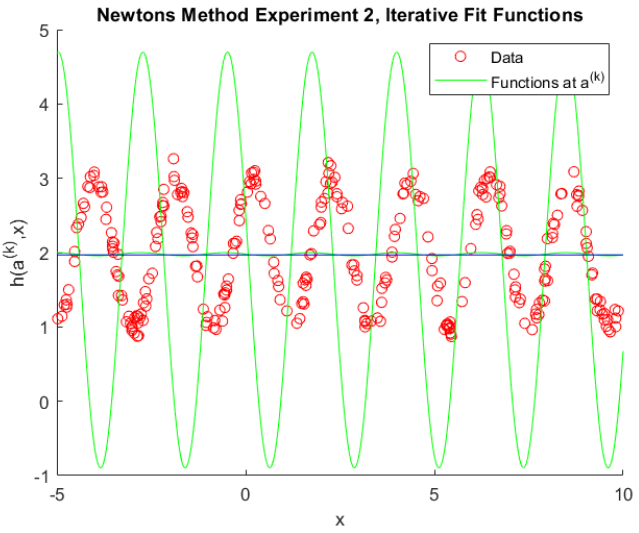

Figure 5. Plot of data values in red and curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using $\boldsymbol{a}^{(k)}$ calculated using NM that iteration with initial guess $\boldsymbol{a}^{(0)}$ corresponding with Experiment 2; Best $h(\boldsymbol{a}, \boldsymbol{x})$ using final $\boldsymbol{a}^{(k)}$ is in blue

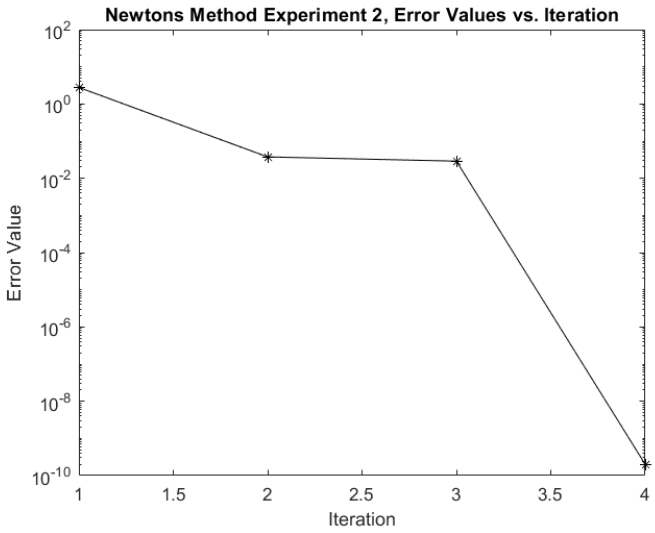

Figure 6. Plot of error estimate values vs. Iteration for the NM method

Figure 6, we still see the error value decreasing each iteration, but more rapidly than the error values in Figure 5. I argue, though, that this is not necessarily a good thing because it shows our values getting extremely close to each other very quickly. This could be detrimental to our final results if the term following our initial guess oversteps and the vectors in the sequence converge to a solution that does not produce a good fit. I believe this happened here because, after our initial guess $\boldsymbol{a}^{(0)}$, the terms of the sequence converge but not to a vector $\boldsymbol{a}^{(k)}$ creates a good fit which produced a minimum function value of 0.50009.

For Experiment 2 and its associated $\boldsymbol{a}^{(0)}$ vector, the BFGS method produces 15 terms and Figure 7 holds the plots of $h(\boldsymbol{a}, \boldsymbol{x})$ for each of the $\boldsymbol{a}^{(k)}$ terms in the sequence. And even though the initial guess produced a function $h(\boldsymbol{a}, \boldsymbol{x})$ that did not fit the actual data well, the BFGS method eventually produced the final vector $\boldsymbol{a}^{(k)}$ in Table 1 that fit the data. The error estimations for the sequence produced were interesting. As seen in Figure 8, the first three iterations show a sharp decrease in error value similar to what we saw in Figure 6 for Newton's method. However, this time the error values go back up almost as if the method is trying to "undo" that term in the sequence. Then, after creating a vector that more closely approximates the actual data, the error values decrease accordingly. These corrected $\boldsymbol{a}^{(k)}$ vectors produce the

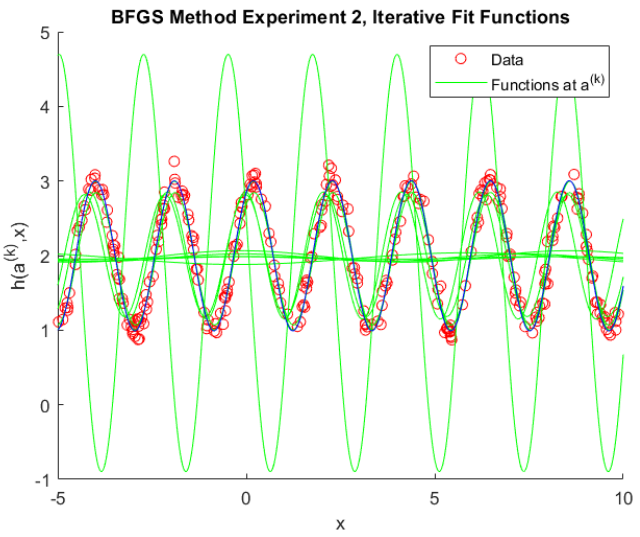

Figure 7. Plot of data values in red and curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using $\boldsymbol{a}^{(k)}$ calculated using BFGS that iteration with initial guess $\boldsymbol{a}^{(0)}$ corresponding with Experiment 2; Best $h(\boldsymbol{a}, \boldsymbol{x})$ using final $\boldsymbol{a}^{(k)}$ is in blue

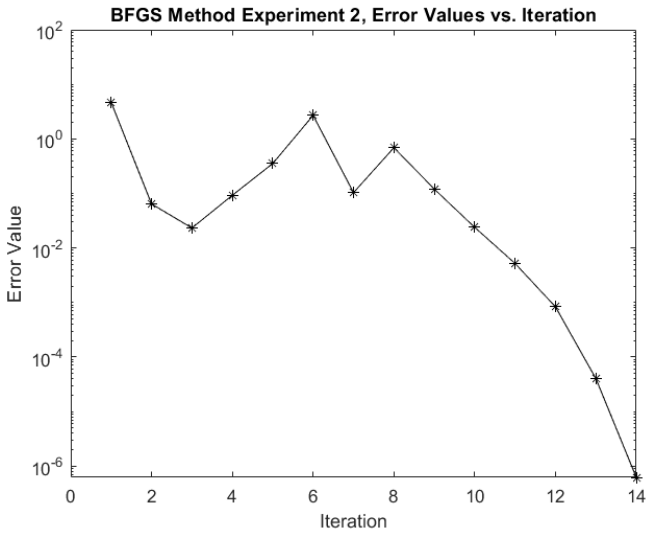

Figure 8. Plot of error estimate values vs. Iteration for the BFGS method

curves $h(\boldsymbol{a}, \boldsymbol{x})$ that are closer to the actual data and the first few $\boldsymbol{a}^{(k)}$ terms of the sequence produce the sinusoidal curves with the low amplitude in the middle of the plot.

The Newton's method section of Experiment 3 produced some very interesting results. We first note that the initial guesses for Experiment 2 and Experiment 3 are very similar with the only big difference being the a_3 parameter, or the phase shift. As discussed previously, NM in Experiment 2 produced a vector of parameters that did not produce a good fit for the data. However, as can be seen in Figure 9, Newton's method now produces a sequence of 5 parameters such that when each is plugged into $h(\boldsymbol{a}, \boldsymbol{x}) = a_0 + a_1 \sin(a_2 x + a_3)$ and plotted, the functions converge to a nice fit for the actual data. This leads me to believe that, since almost everything else stayed the same between the two experiments, the parameter in the initial guess that affects Newton's method the most is the phase shift. Figure 9 replicates what we've seen in both Experiments 1 and 2 of a consistent decrease in error value as the iterations increase.

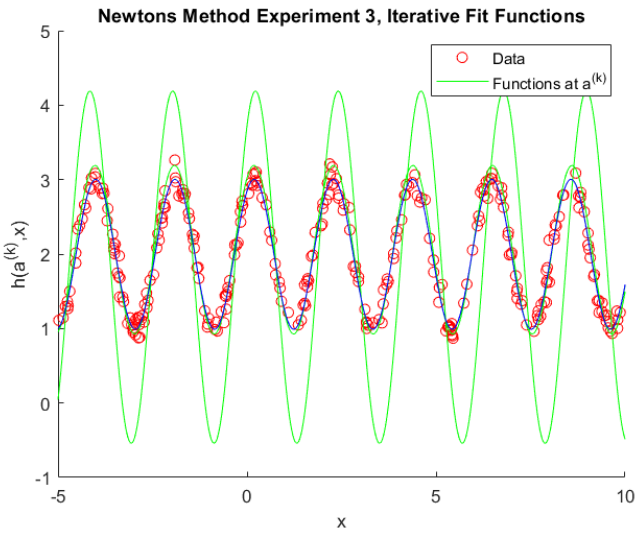

Figure 9. Plot of data values in red and curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using $\boldsymbol{a}^{(k)}$ calculated using NM that iteration with initial guess $\boldsymbol{a}^{(0)}$ corresponding with Experiment 2; Best $h(\boldsymbol{a}, \boldsymbol{x})$ using final $\boldsymbol{a}^{(k)}$ is in blue

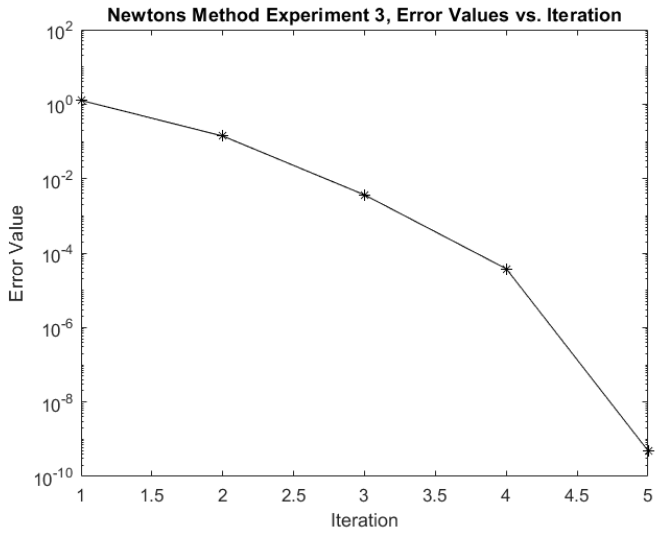

Figure 10. Plot of error estimate values vs. Iteration for the NM method

Lastly, the BFGS method of Experiment 3 with corresponding initial guess $\boldsymbol{a}^{(0)}$ was the only time we received a bad fit out of the three experiments while using the BFGS method. Looking back at Figures 3 and 7, we see that the BFGS method in both experiments produces a final $\boldsymbol{a}^{(k)}$ that made the function $h(\boldsymbol{a}^{(k)}, \boldsymbol{x})$ fit the data well. In this case, however, we start with an initial guess $\boldsymbol{a}^{(0)}$ and plot the corresponding function $h(\boldsymbol{a}^{(0)}, \boldsymbol{x})$ (this is the big sinusoidal wave on the graph) in Figure 11. The following term $\boldsymbol{a}^{(1)}$ of the BFGS sequence is drastically different from the first and results in a small sinusoidal wave in the middle of the graph. We saw in Experiment 2 that the BFGS method seemingly corrected itself and started producing vectors of parameters that better fit the data. This was not the case here and so, as a result, the sequence converged to the final $\boldsymbol{a}^{(k)}$ shown in Table 1. As Figure 11 shows using a solid blue line, the final function plotted $h(\boldsymbol{a}^{(16)}, \boldsymbol{x}) = a_0 + a_1 sin\,(a_2 x + a_3)$ is not a good fit for the data. Interestingly, we can still see the BFGS method trying to "correct" itself in the Error Values vs. Iteration plot in Figure 12. From iteration 2 to iteration 10, the error values go up and down occasionally, indicating that new parameters are being tried. However, the terms of the BFGS method begin converging and produce an error value that is smaller than our threshold for convergence. Thus, that final term of the sequence becomes the vector of parameters $\boldsymbol{a}$ that

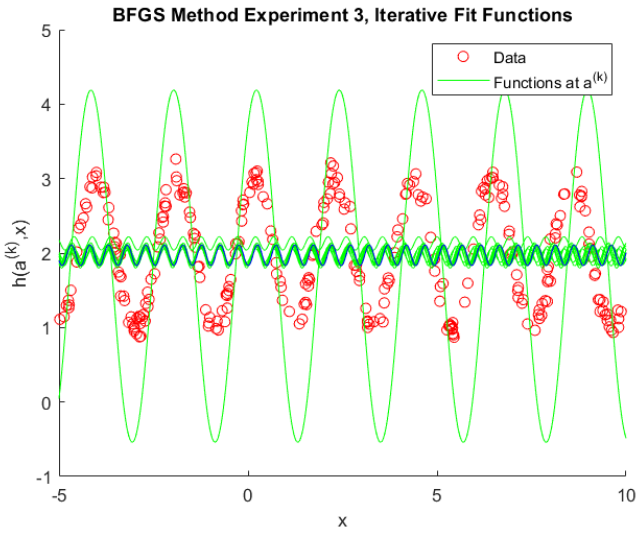

Figure 11. Plot of data values in red and curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using $\boldsymbol{a}^{(k)}$ calculated using BFGS that iteration with initial guess $\boldsymbol{a}^{(0)}$ corresponding with Experiment 2; Best $h(\boldsymbol{a}, \boldsymbol{x})$ using final $\boldsymbol{a}^{(k)}$ is in blue

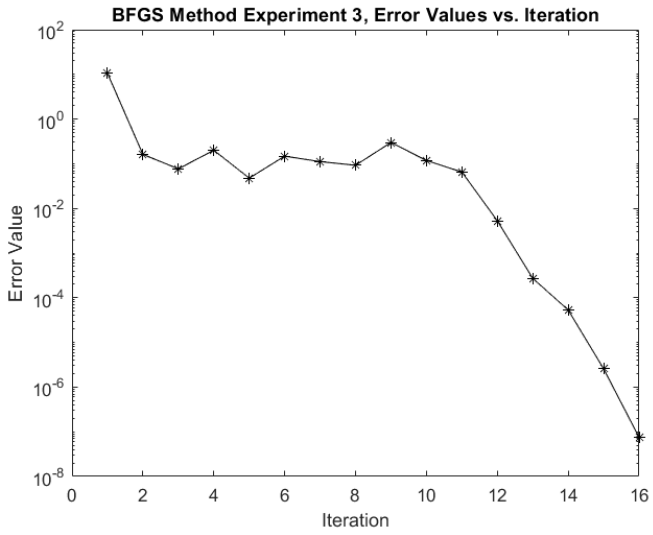

Figure 12. Plot of error estimate values vs. Iteration for the BFGS method

minimizes the objective function; even though it does not produce a function $h(\boldsymbol{a}, \boldsymbol{x})$ that fits the data at all.

In Experiment 4 we consider 100 random initial guesses (trials) for the vector $\boldsymbol{a}$, each with the elements of $\boldsymbol{a}$ drawn uniformly at random in the interval $(0, \pi)$. For both methods (NM and BFGS), we consider the objective function value $f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$ resulting from each trial and we determine the best objective function value over all trials and the associated vector of parameters $\boldsymbol{a}$. We are also interested in the percentage of trials which yield good fits, where a "good fit" is defined as any objective function value less than or equal to 0.1. We create two plots to present our data: The first is a best fit function plot and the second is an objective function values plot. The best fit function plot is created by first plotting the provided data points using open red circle markers and then, using a solid blue line, plotting the curve generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using the best $\boldsymbol{a}^{(k)}$ over all trials and a vector $\boldsymbol{x}$ with a sufficiently small increment. The objective function values plot is simply the final function value $f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$ vs. trial number. This graph is designed to show how often each method provides good fits.

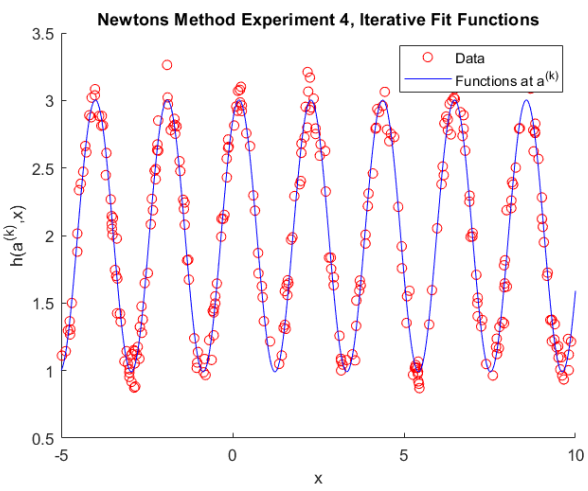

Figure 13. Plot of data values in red and best fit function in solid blue

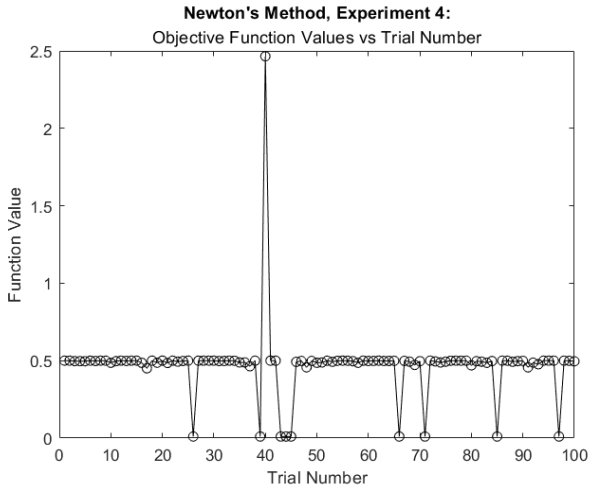

Figure 14. Plot of objective function values at every trial

Since each execution of Experiment 4 in our code produces a set of 100 randomized parameter vectors, the following results are exclusive to the trial I ran. For Newton's method, the percentage of trials yielding a good fit was 9%. That is, only 9 out of the 100 trials produced an objective function value les than or equal to 0.1. For the BFGS method, the percentage of trials yielding a good fit was considerably higher at 25%. These results, as well as their associated parameters are summarized below in Table 2.

Table 2: Results from Experiment 4

	Best Objective Function Value	Associated Parameters	Percentage of trials yielding good fits
Newton's method	$1.016278e-02$	$\begin{bmatrix}1.9969\\1.0061\\2.9994\\1.0055\end{bmatrix}$	9%
BFGS	$1.016278e-02$	$\begin{bmatrix}1.9969\\1.0061\\2.9994\\1.0055\end{bmatrix}$	25%

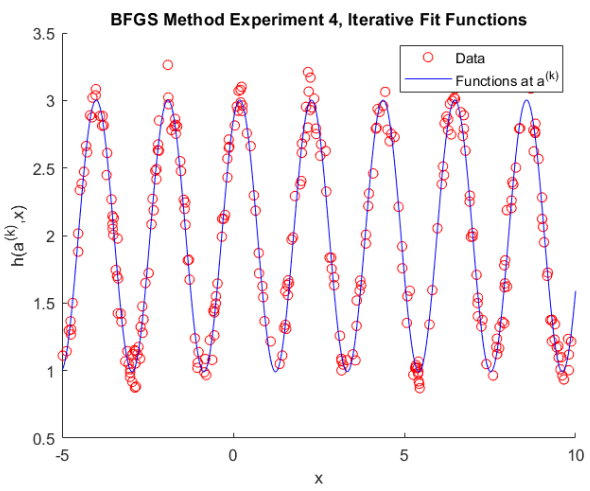

Figure 15. Plot of data values in red and best fit function in solid blue

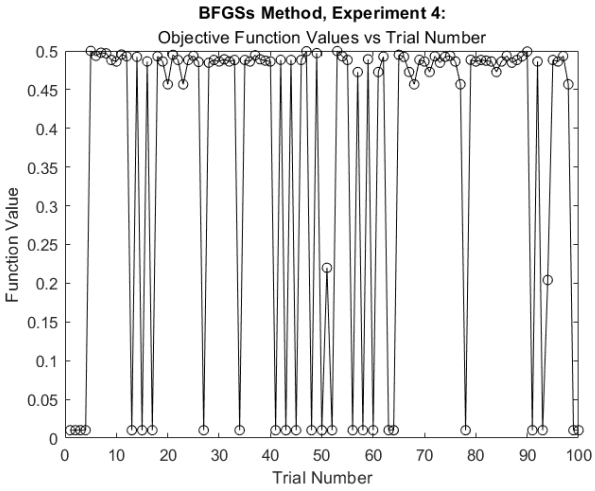

Figure 16. Plot of objective function values at every trial

Figure 13 shows how well the curve generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using the best $\boldsymbol{a}^{(k)}$, found using Newton's method, over all trials fits with the data. Figure 14 shows how many times over all 100 trials the objective function value dipped below 0.1. It also shows that for Newton's method, most of the trials produced an objective function value of about 0.5. Figure 14 shows how well the curve generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using the best $\boldsymbol{a}^{(k)}$, found using the BFGS method, over all trials fits with the data. Unsurprisingly, the curve fits very well. Figure 16 is of more interest to us because it depicts how frequently a function value below 0.1 is produced. It is clear from Figures 14 and 16 that, for this execution of Experiment 4, that the BFGS had more trials that yielded good fits.

Experiment 5 is the extension portion of this report and will consist of a process that is very similar to that of Experiment 4. Experiment 5 will consider 100 random initial guesses for the vector $\boldsymbol{a}$ where the parameters of $\boldsymbol{a}$ drawn uniformly at random in the interval $(0, \pi)$. We will use Algorithms 2 and 3 (the BFGS method and DFP method, respectively) and compare the percentage of trials which yield good fits. I will also provide the best objective function value and its associated parameters for both methods as well as complementary plots for the best fit and objective function values.

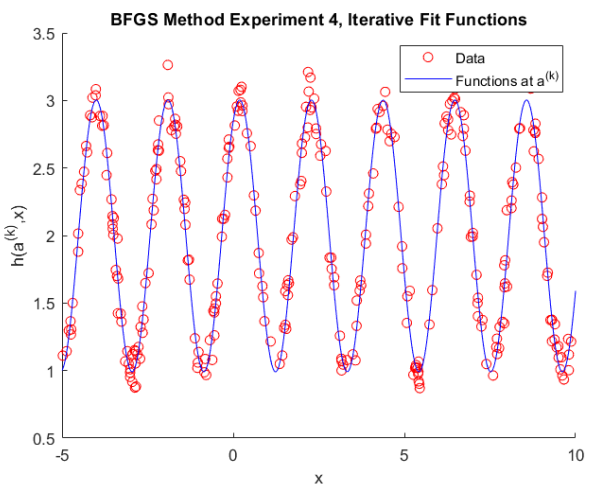

Figure 17. Plot of data values in red and best fit function in solid blue

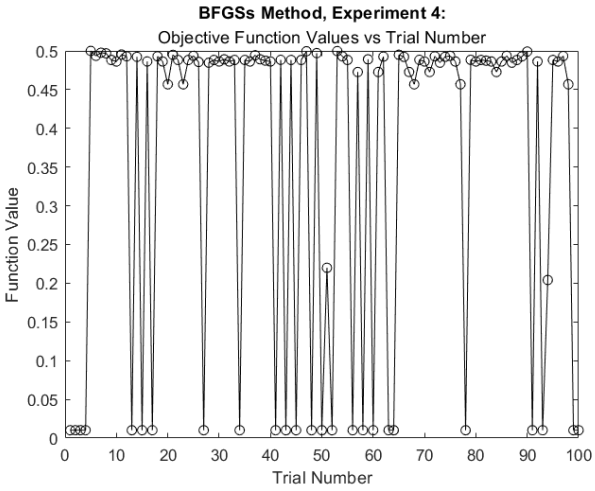

Figure 18. Plot of objective function values at every trial

Since this experiment produces 100 random initial guesses every time the program is executed, the results I present below are exclusive to my run. However, since the results are randomly generated within a closed range, it is fair to assume that these results can be replicated or approached. For the DFP method, the percentage of trials yielding a good fit was 24%. That is, out of the 100 trials, 24 of them produced an objective function value less than or equal to 0.1. For the BFGS method, the percentage of trials yielding a good fit was 31%. From the data in Table 3 alone, our experiment indicates that the BFGS method and the DFP method are both good approximators for the parameter vector $\boldsymbol{a}$. The results are summarized in Table 3 below.

Table 3: Results from Experiment 5

	Best Objective Function Value	Associated Parameters	Percentage of trials yielding good fits
DFP method	$1.016278e-02$	$\begin{bmatrix}1.9969\\1.0061\\2.9994\\1.0055\end{bmatrix}$	24%
BFGS method	$1.016278e-02$	$\begin{bmatrix}1.9969\\1.0061\\2.9994\\1.0055\end{bmatrix}$	31%

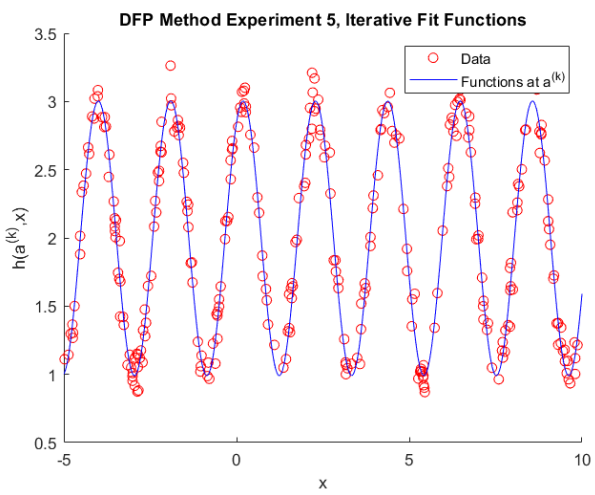

Figure 19. Plot of data values in red and best fit function in solid blue

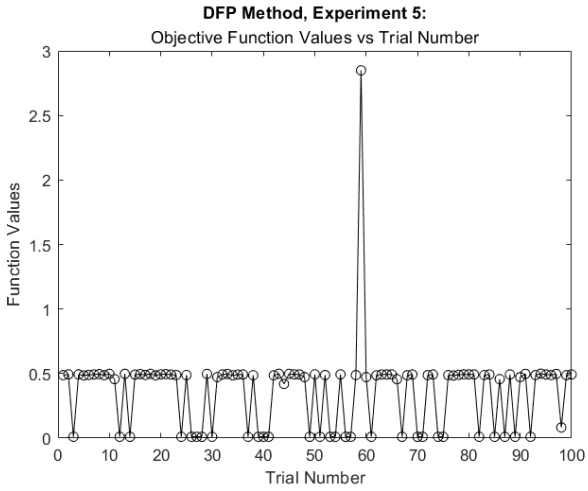

Figure 20. Plot of objective function values at every trial

Not surprisingly, we get the very similar objective function values, similar associated parameters, and similar percentages for the number of trials yielding good fits for both methods. Figures 19 and 21 show that the DFP and BFGS methods are capable of producing sequences of vectors $\boldsymbol{a}$ such that the curve generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ at the final term $\boldsymbol{a}^{(k)}$ of the sequence is a good fit for the data. Figures 20 and 22 do a great job at visualizing the number of trials which yield good fits for the DFP and BFGS methods, respectively. We can see from Figure 20 in particular that most of the trials for the DFP method produce an objective function value of about 0.5. Similarly, Figure 21 shows that most trials for the BFGS method produce an objective function value of about 0.5.

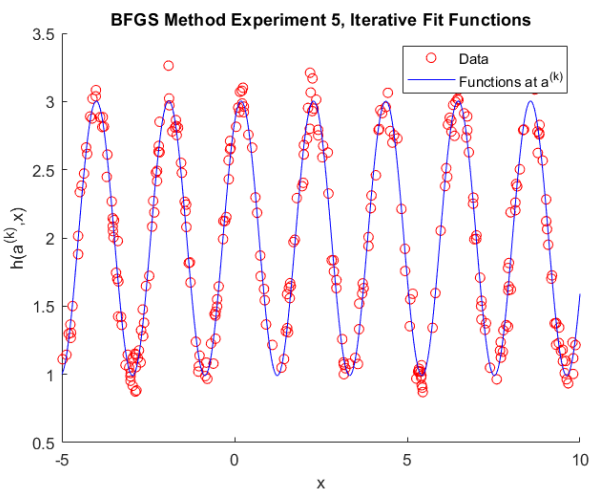

Figure 21. Plot of data values in red and best fit function in solid blue

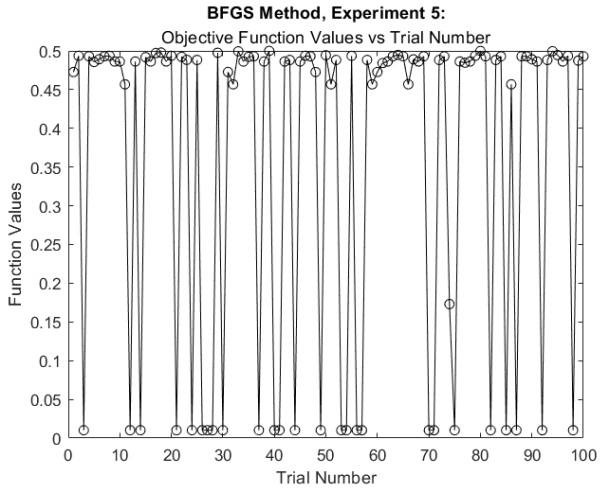

Figure 22. Plot of objective function values at every trial

5 Conclusion

From experiments 1, 2, and 3 we were able to collect that our initial guess for the solution has a very strong effect on the results produced by both Newton's method and the BFGS method. By plotting the curves generated by evaluating $h(\boldsymbol{a}, \boldsymbol{x})$ using the $\boldsymbol{a}^{(k)}$ vectors from the sequences for both methods, we are able to determine which parameters affect each method the most. For example, we saw in the experiments 2 and 3 that our initial parameters were almost identical except for the phase shift. This had a major effect on both methods—from experiment 2 to experiment 3, this improved the iterative fit for Newton's method but ruined the iterative fit for the BFGS method. We also saw that the error estimation plots for both methods had different qualities. Specifically, the error plots for Newton's method were consistently decreasing while the error estimation plots for the BFGS method would increase and decrease sporadically. As for the numerical results of the methods, Table 1 showed that NM would always take less iterations to terminate than the BFGS method. Also, the same final $\boldsymbol{a}^{(k)}$ vector and final objective function $f(\boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y})$ value appeared consistently throughout the experiments. Experiment 4 showed us that the BFGS method tends to produce a higher percentage of trials which yield good fits than Newton's method. Similarly, experiment 5 showed us that the BFGS and the DFP methods produce similar percentages of trials which yield good fits.