Sparse Representation and Possible Application for Neural Data from a Utah Array

Pablo Gutierrez, Alexandros Kontogiorgos-Heintz, Peter LaFontaine, Vidhu Nath
Department of Computer Science
Boston University
Boston, MA 02215
{gutpab18, akontih, lafonta, vnath}@bu.edu

Abstract

In this paper, L1-minimization was used to reconstruct an accurate representation of the dataset of neuronal action potentials, using sample points from this dataset, with the peaks of the values in both the original data, and the reconstructed data being relatively equal. We found that with a higher percentage of sampling, we managed to get lower error rates. This can be extended to different applications such as determining which neurons fired a certain set of action potentials, using the interspike intervals (ISI) through K-means clustering.

1 Introduction

Analyzing neuronal data is a fundamental problem today. Not only are we limited in knowledge about the brain and its functions, actually collecting data and making reasonable conclusions can be difficult. There are so many processes occurring at any one time that it can be complicated to sample specific data without much noise. And even if noise could be eradicated, the data can vary. For example, in just a few seconds of sampling many action potentials from different neurons are recorded. Inherently, analysis of neuronal data is dependent on computer science to collect, measure and organize.

This paper develops three parts involved in neuronal data analysis: the formulation of a mathematical constraint problem derived from the collected data, proving and implementing the theoretical background that makes this constraint problem solvable, and discussing the results of the approach.

Formulation of the problem stems from making certain assumptions and observations about the data so that it mirrors the l_1*-minimization problem*. Transforming neuronal data into this minimization allows sampling of a small percentage of this data. By reducing the problem size, preforming operations on and distribution of data becomes simpler. Not only is sampling percentage reduced, but also analysis of the approach is discussed with respect to clustering action potentials to identify which neuron fired the action potentials in each cluster.

2 Neuroscience Background

The goal of this project is to build a system that will take a compressive sensing approach to sampling neuronal data collected from a single probe of a micro-electrode array (Utah Array), recreate it with enough accuracy to identify action potentials and inter-spike intervals (ISIs), and cluster the resulting ISIs. Utah arrays are compact 10x10 matrices of electrical recording spikes

that can be surgically implanted in the brain. The property of the signal that allows a compressive sensing approach is the scarcity of the occurrence of an action potential in the time domain. Since the recorded neuron is more likely not firing than firing at any given instance, the data can be considered sparse.

The probes on the Utah array work by determining the electrical potential created by neurons. Since the arrays cover an area of cortex larger than a single neuron, the resulting data collected can be assumed to cover multiple neurons. The benefit of this is that it allows for analysis of the firing in regions of the brain, instead of just along a single neuron. Neuronal data is transmitted by the frequency at which neurons spike, therefor the important component of the data is the ISI. A potential source of a problem with these probes is that they will record from all neurons that they touch. It has recently been shown though that it is the firing frequency in a small, localized region that is important, not the individual neuron.

Signals are transported along the lengths of the dendrites and the axon of a neuron in discreet signals known as action potentials. These signals are caused by the neuron actively pumping sodium, potassium, and calcium ions in and out of the cell membrane. These signals all follow a general pattern of an initial depolarization of the membrane, followed by a sharp hyper polarization, and then depolarizing back to the initial starting potential. Because of this regularity of spiking patterns, the firing of different neurons can be analyzed together. This inflow/outflow of ions can be modeled by the Hodgkin-Huxley equation; an extension of the Nerst equation.

3 Mathematical and Computational Approach

Given this neuronal dataset, the goal is to fit it into the *l_1-minimization problem*. This section explains firstly what the *l_1-minimization problem* is and how it is solved algorithmically; then it shows how the data is transformed into a minimization problem; and lastly, it reviews the computational approach that was used to arrive at our results.

First, it is important to note that l_1-minimization has a subset of problems. That is, there are different types of l_1-minimizations; for example, there is one subset of l_1-minimization with equality constraints while there is another with quadratic constraints[1]. The neuronal problem that is being solved is simply a matter of equality constraints. The *l_1-minimization problem*, which can be modeled using the following program,

$$\min\|x\|_1 \ subject\ to\ Ax = b \qquad (1),$$

finds the vector with the smallest *l_1-norm*,

$$\|x\|_1 := \ \sum_i |x_i| \qquad (2),$$

that satisfies the results in b, where A is a $m \times n$ matrix, $x \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$. Given that x is sufficiently sparse and that all values in A are mutually perpendicular and unit vectors, there exists a solution x_0 that satisfies the conditions[5]. There are a few possible algorithms used to solve this problem, including the *Primal-Dual Interior Point method*, which is the method used in this paper[2].

To even begin this method, equation (1) should be cast as a linear program as follows:

$$minimize\ c^T x\ subject\ to\ Ax + s = b, s \geq 0 \qquad (3)$$

$$maximize\ -b^T z\ subject\ to\ A^T z + c = 0, z \geq 0 \qquad (4).$$

Equations (3) and (4) are the inputs for the *Primal-Dual Interior Point method*, equation (3) being the primal minimization while equation (4) refers to the dual[2]. Using this linear program, we can apply the algorithm as follows[4]:

1. Initialize $s, x\ and\ z$, where s and z > 0
2. Compute residuals and evaluate stopping criteria (algorithm will end here)
3. Compute the scaling direction
4. Select the barrier parameter
5. Compute the search direction
6. Update iteration variables and return to step 1

The basic idea lies in the residuals, which can be defined as follows:

$$r_p = Ax + s - b, r_d = A^T z + c,\ r_g = s^T z \qquad (5)$$

Once these residuals are calculated in step 2, steps 3-5 define the linearization of the three residuals within the predefined equations in (3) and (4)[4]. That is, the solutions for the equations are found. At each iteration a new solution is found until satisfy the stopping criteria. The stopping criteria for the algorithm is defined below:

$$\|r_p\| \leq \epsilon_{feas} \max\{1, \|b\|\}\ and\ \|r_d\| \leq \epsilon_{feas} \max\{1, \|c\|\} \qquad (6)$$

If both of these are satisfied then we must assure that the duality gap is at the proper level:

$$c^T x < 0\ and\ \frac{s^T z}{-c^T x}\ or\ -b^T z > 0\ and\ \frac{s^T z}{-b^T z} \qquad (7)$$

So at each step s, x and c must satisfy conditions (6) and (7); if this is not the case, then the algorithms terminates and we have found the vector x_0 which represents the value of the minimization[4]. This is the *Primal-Dual Interior Point method* used to calculate the minimized vector for the *l_1-minimization problem*.

Since it is shown previously that the neuronal dataset is sparse, it can be used with the *l_1-minimization problem*. In equation (1), x can be defined as a vector of n samples from the dataset, A is a $m \times n$ orthonormal matrix with unit vectors and b is a vector of m points from the original neuronal dataset[1][3][5]. To apply the compressive sensing technique of data recovery, the algorithm described above can be applied to this same data. The matrix A is computed, n samples are taken from dataset and then the equation and algorithm is applied. The result should be extremely similar to the original dataset.

Having the mathematical background and values necessary for this problem, the algorithm can now be applied. This was done using the *l_1-magic* Matlab package, which makes it quite simple to perform these type of minimization[1]. More on the implementation of this

algorithm can be seen in the code provided.

4 Results and Discussion

The results of the project indicate several trends about the reconstructed data. First, the more sampling that is used – in the case above, 40% of the original data was sampled – the higher the accuracy of the reconstructed data when compared to the original data. The reconstructed data was calculated to be 8.2616% different from the original data that was used, which indicates that while there are a few deviant points which are drastically different from the original data, most of the reconstructed data was relatively accurate to the original. In another experiment, the sampling percentage was set to 0.15, so that 15% of the values from the original data set were used to develop the reconstructed data. When this value was used, the percent error changed to -14.3735%. The negative percentage indicates that on average, the reconstructed values were smaller than the original values, and as it can be noted, the percentage magnitude is greater than that of the data sampled at 40%.

As a final experiment, the sampling amount was set to 65% of the original data. The result was that the average error was -3.494%. The magnitude of this error is less than both of the previous experiments, which demonstrates the understanding that more sampling means higher accuracy, simply because more of the original data is being used in the reconstruction process. For these calculations, the following percentage difference formula was used:

$$\Delta x(percentage) = \frac{x - x_0}{x_0}$$

for x being the reconstructed data and x_0 being the original data value.

The second major observation taken from this experiment is that the time values of the peaks are exactly the same. Several experiments were run, and the peaks of each signal – where peak is defined to be the maximum positive value of a signal – from the original data set were in the exact same location as in the original data set. This implies that the sampling method that was used, which was the primal-dual interior point method, was able to accurately reconstruct the locations of peaks. This was assumed to be due to the extreme values the peaks take with regards to the other values in the data set. The max value of the peaks in the original data was over 238.7 times higher than the average value of the data; for the reconstructed data, the max value was over 315.8 times higher per neuron spike. Because of this nature, the algorithm that was reconstructed could readily identify and reconstruct this outlier value.

5 Conclusion

While our goal of producing an algorithm that identifies which neuron fired a given action potential was not met, we are satisfied with our results. Dealing with data of such size can be complicated. We made the correct analysis on our data, applied the mathematical foundations that went with our analysis, and our results indicate that it worked. We were able to sample at a rate of 15% and still have a solid recovery, as indicated by our error results.

The most important result, though, is always reproducing the action potentials in the data. Plenty of noise is observed, but the action potentials were always recovered, and that is the most important aspect of this project. A very important next step would be to deliver on our goal. This could be done by calculating the interspike intervals (ISIs) of the action potentials, applying the k-means algorithm to this set of ISIs, and use the result, a set of clusters, to map a neuron to each cluster. This approach seems as it would work because neurons fire at different frequencies. By applying k-means to the ISIs, we have clustered with respect to firing frequency. This surely would result in positive identification.

Figure 1

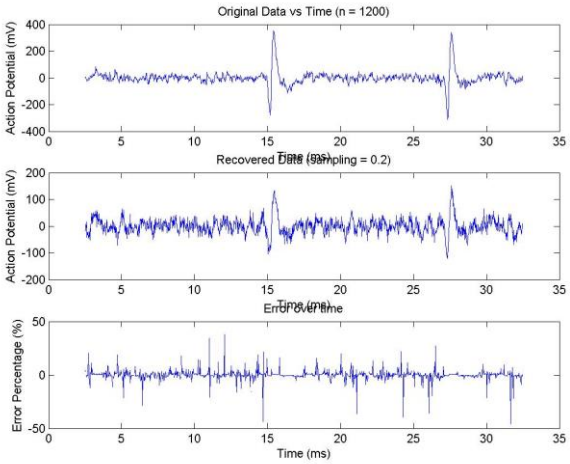


Figure 2

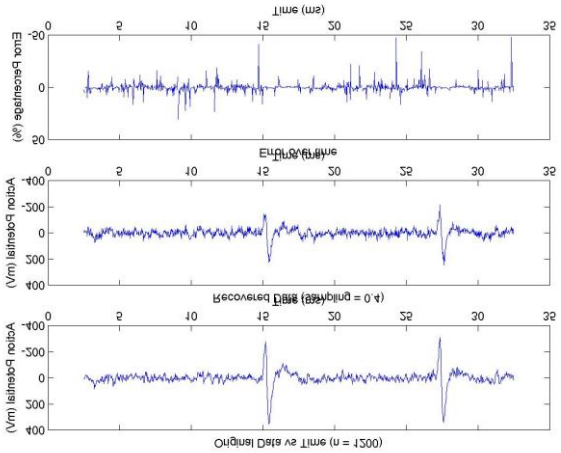


Figure 3

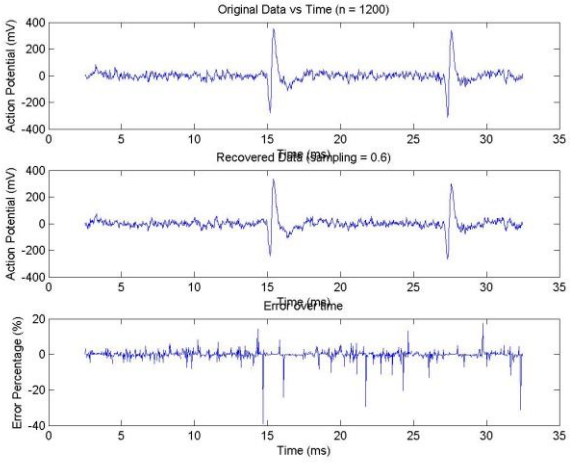


Figure 4

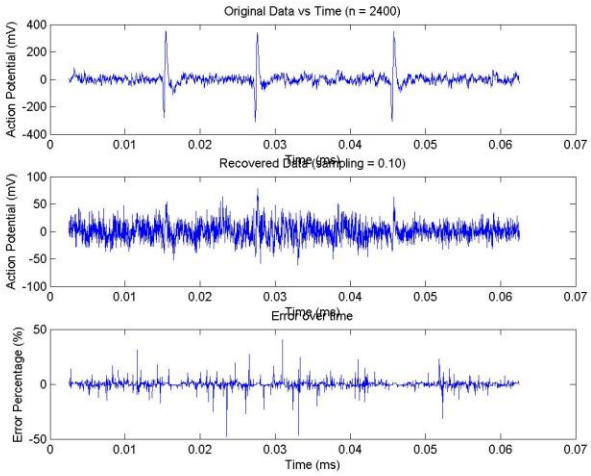


Figure 5

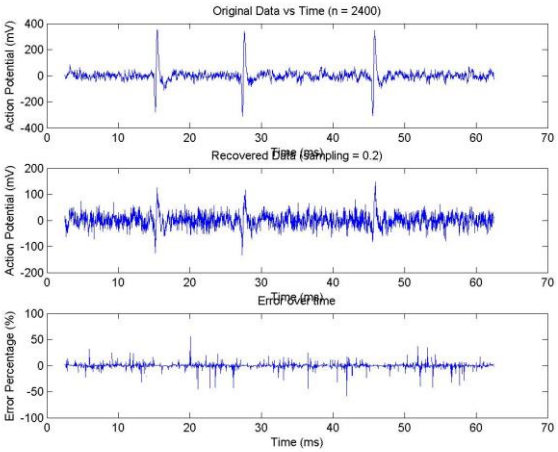


Figure 6

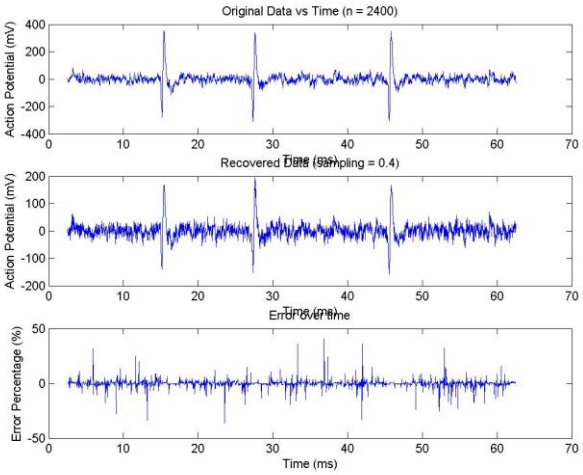


Figure 7

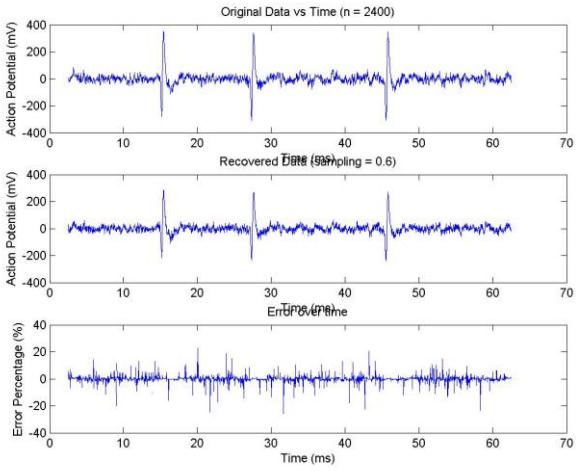


Figure 8

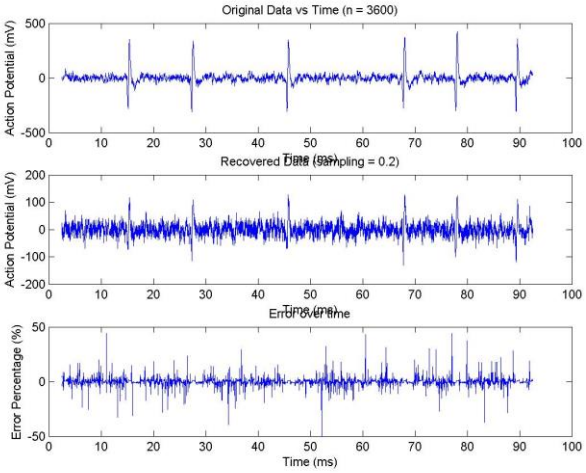
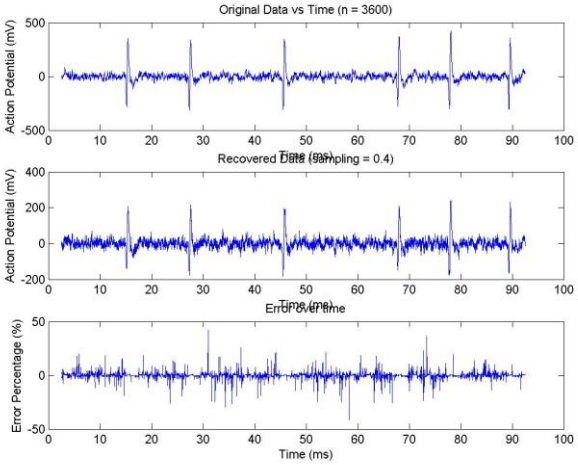


Figure 9

203



204

205 **References**

206 [1] Candes, Emmanuel, and Justin Romberg. "L1-magic: Recovery of Sparse Signals ViaConvex
207 Programming." (2005): n. pag. *Georgia Tech*. Georgia Tech, Oct. 2005. Web. 16 Dec. 2014.

208 [2] Carbonetto, Peter. *Intuition Behind Primal-Dual Interior Point Methods for Linear and*
209 *Quadratic Programming* (n.d.): n. pag. *UBC*. University of British Columbia. Web. 16 Dec. 2014.

210 [3] Rowland, Todd. "Orthonormal Basis." From MathWorld--A Wolfram Web Resource,
211 created by Eric W. Weisstein. http://mathworld.wolfram.com/OrthonormalBasis.html

212 [4] Vandenberghe, L. "Lecture 15 Primal-dual Interior-point Method." (n.d.): n. pag. *UCLA*.
213 University of California at Los Angeles, Sept. 2013. Web. 16 Dec. 2014.

214 [5] Zhang, Yin. "Theory of Compressive Sensing via l1-Minimization: A Non-RIP Analysis
215 and Extensions." (n.d.): n. pag. Rice University. Rice University. Web. 16 Dec. 2014.