LA-UR-17-28229
Approved for public release; distribution is unlimited.

Title: Seismic classification through sparse filter dictionaries

Author(s): Hickmann, Kyle Scott
Srinivasan, Gowri

Intended for: Report

Issued: 2017-09-13 (rev.1)

Seismic classification through sparse filter dictionaries

Kyle S. Hickmann[*], Gowri Srinivasan[†]

Abstract

We tackle a multi-label classification problem involving the relation between acoustic-profile features and the measured seismogram. To isolate components of the seismograms unique to each class of acoustic profile we build dictionaries of convolutional filters. The convolutional-filter dictionaries for the individual classes are then combined into a large dictionary for the entire seismogram set. A given seismogram is classified by computing its representation in the large dictionary and then comparing reconstruction accuracy with this representation using each of the sub-dictionaries. The sub-dictionary with the minimal reconstruction error identifies the seismogram class.

Keywords:

Introduction

The goal of modern seismology is to reconstruct estimates of the Earth's subsurface from acoustic and seismic observations. These observations are collected over a large region on the Earth's surface. Data collection usually consists of producing some large explosive pulse near the surface and collecting wave propagation data at an array of seismic or acoustic receivers. Here, seismic receivers collect three-axis velocity data while acoustic receivers collect pressure data.

State of the art seismological methods form subsurface estimates by assuming some class of model for the subsurface beneath the region that the seismic survey is conducted over. These *a priori* models range from simplistic, one dimensional layered models of the density and elastic properties of the subsurface, to complex heterogeneous models based on anisotropic elastic properties with a specified correlation structure. The seismic data is then used to estimate parameters in these models. However, selecting a class of *a priori* models is usually done ad hoc, either by attempting reconstruction using a few different model classes or by relying on expert knowledge of the substructure. The work presented here offers an algorithmic method, based in state of the art machine learning methods, to use seismic data to inform the model selection process over a wide range of subsurface models.

[*]XCP-8, Mathematician, Los Alamos National Laboratory, hickmank@lanl.gov
[†]T-5, Mathematician, Los Alamos National Laboratory, gowri@lanl.gov
LA-UR-17-28229 (ver.2)

Inevitably the task of model selection relies on solving a classification problem, assigning a *best* model type to an observed set of seismic data. Our method for developing a classifier is to artificially generate synthetic seismic data, through numerical simulation, for a large set of potentially applicable subsurface model types. Our examples presented here construct a classifier for eight subsurface models. Briefly, our method of classification is to construct a sparse dictionary for the range of seismic observations that can be generated from each model type. This dictionary construction relies on solving a convex optimization problem for each model type. The optimal dictionaries that are found form a set of convolutional filters that can be combined with sparse representations of features in the seismic data to reconstruct a given seismic data set. Once a sparse filter dictionary is learned for each model type a seismic observation can be classified by finding the dictionary which allows the most accurate, and sparse, reconstruction.

This research has grown out of attempts to develop methods to reduce the dimensionality of the seismic inverse problem. As seismic data collection has become more advanced the size and scope of seismic inversion has reached the point of relying on large supercomputers to handle the computational loads involved. A great deal of this computational burden is due to the fact that subsurface models often start with hundreds of parameters even though the seismic observations may only inform a lower dimensional manifold in that parameter space. Moreover, the seismic observations themselves live in a very high-dimensional space even though many *a priori* subsurface models can only produce seismic observations in a lower-dimensional submanifold. Accomplishing model selection through classification allows the specification of an optimal low-dimensional submanifold of subsurface model space, parameterized by the model resulting from the classification algorithm, that can be searched during seismic inversion. This is not directly dimension reduction but accomplishes many of the same goals and is a step in the correct direction.

1 Seismic Data Classification Problem

Given a set of seismic measurements the *seismic inversion* problem seeks to reconstruct an image of the elastic properties of the Earth's interior, or at least the elastic properties in some region of the Earth's subsurface. If acoustic measurements are given instead of directional seismic measurements then the goal is to reconstruct just the acoustic properties. Though a great deal of success has been found using seismic inversion methods the computational demand is very great. This computational demand can be reduced if macroscopic questions are asked about the subsurface structure, such as:

- "Does there exist a strong acoustic reflector within a certain depth?"
- "Is there a large jump in acoustic speed within a given depth?"
- "Is there a compact, strong reflector within a given depth?"

Such questions lend themselves nicely to machine-learning classification problems and result in actionable information, at a fraction of the data requirements and computational cost.

The research described here involves the construction of a sparse, convolutional-filter-based classifier able to classify seismograms by the presence or absence of macroscopic

acoustic features in the subsurface. We first explain the construction of the training set of seismograms consisting of representatives from 8 different classes. Next we detail the process of learning dictionaries of filters for each of the eight classes. The class specific dictionaries are then combined into a large dictionary used for classification.

2 Generating Synthetic Seismic Data

We generate seismograms from acoustic profiles consisting of a combination of four features:

1. Heterogeneous, Gaussian field, background
2. Horizontal jump in acoustic speed
3. Strong horizontal reflector
4. Strong, circular, compact reflector.

All acoustic profiles have a heterogeneous background. The class of an acoustic profile, and therefore that of the seismogram generated, is then determined by the presence or absence of the remaining three features. Classes are labeled by integers determined by their binary relations to the presence or absence of each feature. The binary representation of a class is determined by the boolean triple, $\{\textit{Compact reflector}, \textit{Horizontal reflector}, \textit{Acoustic jump}\}$. A seismogram generated from an acoustic profile containing a compact reflector, without a horizontal reflector, containing an acoustic jump would have binary class label $(1, 0, 1)$ and therefore would be in class $5 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$. This multi-label classification results in 8 distinct classes with labels $0, 1, 2, 3, 4, 5, 6, 7$. Each of the features present in the acoustic profiles are shown in figures 1-5.

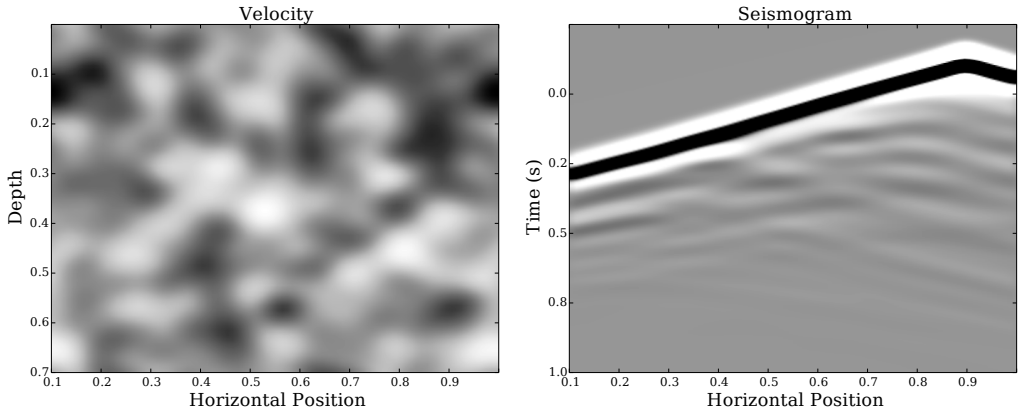

Figure 1: Typical acoustic velocity structure (LEFT) and corresponding seismogram (RIGHT) for class 0. Class 0 consists of a heterogeneous background created by a Gaussian random field. The correlation length is set to $\sigma = 0.0125$ and the velocity field is scaled so that the acoustic velocity ranges over $[1.75, 2.25]$.

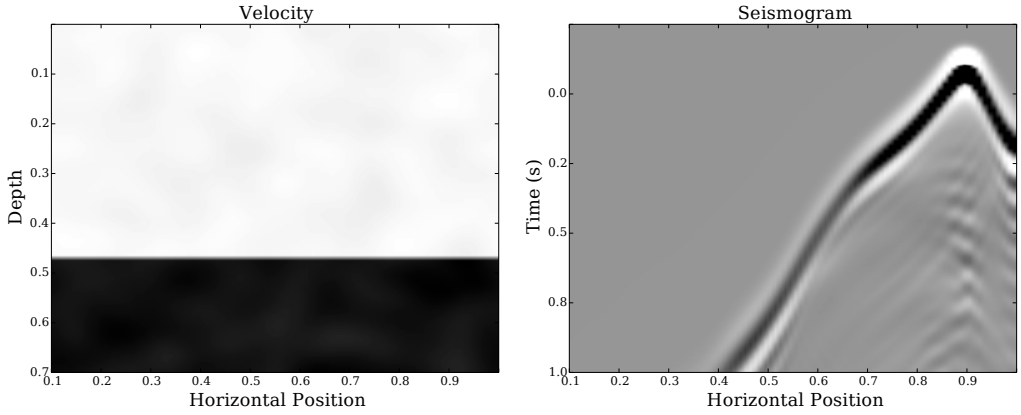

Figure 2: Class 1 has a heterogeneous background generated as in class 0. However, class 1 has an acoustic jump at a random depth, ranging over $[0.3, 0.6]$, and random background acoustic speed. The range of the surface acoustic speed is originally over $[0.5, 1.0]$ and the bottom speed is over $[1.5, 3.0]$. The acoustic profile is then scaled to range over $[-1.5, 1.75]$ and a realization of a Gaussian random field as in class 0 is added to the scaled acoustic profile. This gives an acoustic profile with speeds over $[0.25, 4.0]$ with a heterogeneous background. A typical acoustic profile from class 1 is shown (LEFT) alongside the corresponding seismogram (RIGHT).

3 Learning Dictionaries for Each Class

For each class of seismogram, described above, we learn a dictionary of convolutional filters. We will denote the filter dictionary for class α by $\{d_m^\alpha\}_{m=1}^{M_\alpha}$. For each class, this filter dictionary is learned using a set of training seismograms generated from the class, α, $\{s_k^\alpha\}_{k=1}^{K_\alpha}$. A dictionary for class α is learned through solving the optimization problem

$$\begin{aligned} \underset{\{d_m^\alpha\},\{x_m^\alpha\}}{\arg\min} \frac{1}{2} \sum_k \left\| \sum_m d_m^\alpha * x_{k,m}^\alpha - s_k^\alpha \right\|_2^2 + \lambda \sum_k \sum_m \|x_{k,m}^\alpha\|_1, \\ \text{such that } \|d_m^\alpha\|_2 = 1 \text{ for all } m = 1, 2, \ldots, M_\alpha. \end{aligned} \tag{3.1}$$

In problem (3.1) the representations, x_m, are the same size as the seismograms, s_k. We have also chosen the seismograms to be the same size for each class of acoustic profile. The filters, d_m, must have size and number specified before solving the optimization problem but are chosen to have equal sizes and numbers across classes. In our research we have chosen filters in $\mathbb{R}^{25\times 10}$ and dictionaries consisting, initially, of 20 filters. Our training seismograms were images in $\mathbb{R}^{500\times 90}$ so the filters are less than a 10% the size of the original images. Filters are randomly initialized using a standard normal distribution for pixel values and then an iterative *Alternating Direction Method of Multipliers* (ADMM) algorithm is used to solve the optimization problem (3.1).

Since the number of initial dictionary elements was chosen arbitrarily, a solution of problem (3.1) usually only adjusts a limited number of dictionary elements for each class of seismogram. Due to the L^1-constraint on the representations a minimal set of convolutional filters is learned for each class. Dictionary elements that are not informed by a solution to

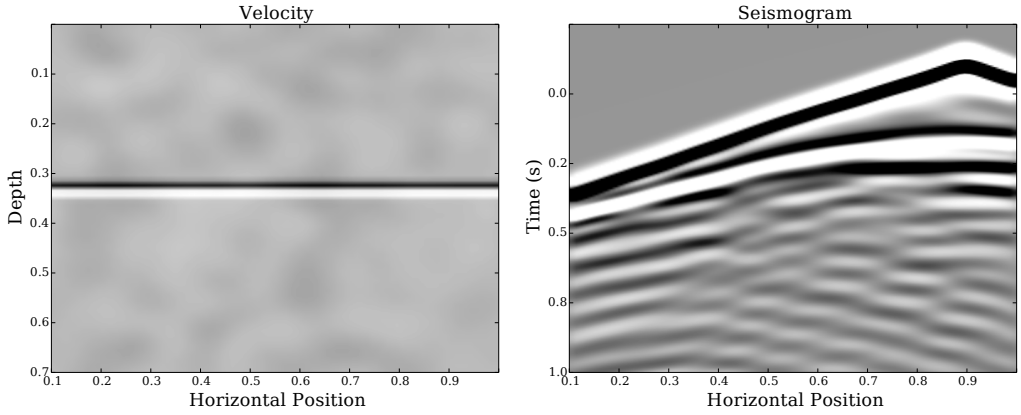

Figure 3: Class 2 consists of a heterogeneous background profile along with a single randomly generated horizontal reflector. The amplitude of the reflector varies over $[1.0, 5.0]$ while the depth varies over $[0.45, 0.95]$. As in the case of class 1 the single reflector profile is scaled to vary over $[-1.5, 1.75]$ and then a Gaussian background field is added. This yields acoustic speeds varying over $[0.25, 4.0]$. A typical acoustic profile from class 2 is shown (LEFT) alongside the corresponding seismogram (RIGHT).

the minimization problem then do not play a role in reconstruction for that class and can, therefore, be discarded. We show the 8 sets of trained filter dictionaries in figures 8-15. We can see that the size of learned dictionaries can vary greatly between classes. However, the error in reconstruction for each class-specific dictionary is comparable, meaning, regardless of size, each dictionary can faithfully represent its class.

3.1 "Mini-batch" Training Sets for DL

In learning the dictionaries through problem (3.1), the computational burden grows at least quadratically with the number of training seismograms, $\{s_k^\alpha\}$, included in the problem. First, each step of the optimization procedure is more expensive as more samples are included. Second, the optimization problem is more restrictive, and therefore more iterations are required when more samples are included. This makes solving the optimization problem, even with a very efficient algorithm, computationally intractable for more than tens of training samples. Our solution is to randomly sample training seismograms from a much larger bank of training seismograms. For our work here we used a set of 1000 seismograms generated from each class of acoustic profile, 8000 seismograms total. A given classes filter dictionary was then learned by randomly initializing the filter dictionary and approximately solving the optimization problem for a set of 20 training seismograms. The learned dictionary was then used as the initialization in problem (3.1) with a new set of 20 randomly selected training seismograms from the same class and the filter dictionary was further refined. This process was then repeated 20 times to arrive at a dictionary that has been informed by a large subset of the available training seismograms. Procedures like the one just described are known as *mini-batch* training methods and have been used successfully to train deep neural networks and convolutional neural networks in supervised learning problems with backpropagation algorithms.

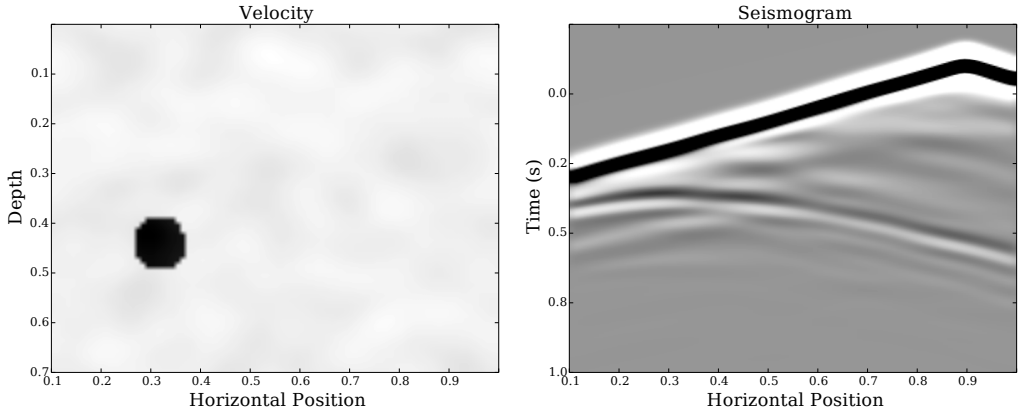

Figure 4: The fourth and final type of feature in our set of synthetic acoustic profiles is a compact strong reflecting inclusion. The inclusion is randomly placed with horizontal position between $[0.2, 0.9]$ and vertical position between $[0.5, 0.6]$. Inclusions are circular with a random radius between $[0.05, 0.1]$. Within the inclusion the acoustic speed jumps to 2.0 which, when added to our Gaussian heterogeneous background makes the range of class 4 acoustic speeds between $[1.75, 4.25]$. A typical acoustic profile from class 4 is shown (LEFT) alongside the corresponding seismogram (RIGHT).

3.2 Resulting Learned Filter Dictionaries

The mini-batch dictionary-learning algorithm performed well. The optimization problem (3.1) had a regular decrease in error until a basin of minimum error was reached. The learned dictionary retained a similar reconstruction error for multiple samples from the training set, indicating a robust dictionary was learned for each class. For each class, the filter dictionaries learned yielded sparse reconstructions of the seismograms within the class for which the dictionary was trained, see Figure 7. Moreover, the within-class reconstructions using the learned dictionary were indistinguishable by eye from the actual seismograms, Figure 6. Though the *eyeball norm* is a very crude metric solutions to classification problems like ours would hopefully not be sensitive to finer evaluation metrics, if they were the classification solution would be very sensitive to measurement noise. This is further quantified by examining the diagonal terms in table 1. For each classes filter-dictionary, table 1 shows the value of the objective function,

$$g(s;\lambda) = \min_{\{x_m\}} \frac{1}{2} \left\| \sum_m d_m * x_m - s \right\|_2^2 + \lambda \sum_m \|x_m\|_1 \tag{3.2}$$

averaged over 20 randomly sampled seismograms from another class. For example, the entry in table 1 corresponding to *Dictionary 1* and *Class 4* shows the objective function using the dictionary trained on class 1 seismograms averaged over 20 seismograms from class 4. The values from the diagonal of the table are all low, indicating a good dictionary for reconstructing seismograms. An example reconstruction is seen in figure 6. Unfortunately for the classification problem the off diagonal elements also yield small objective functions, indicating inter-class reconstruction error is low. We will discuss this topic further in the classification section. The filter-dictionary reconstructions are sparse as well as accurate by

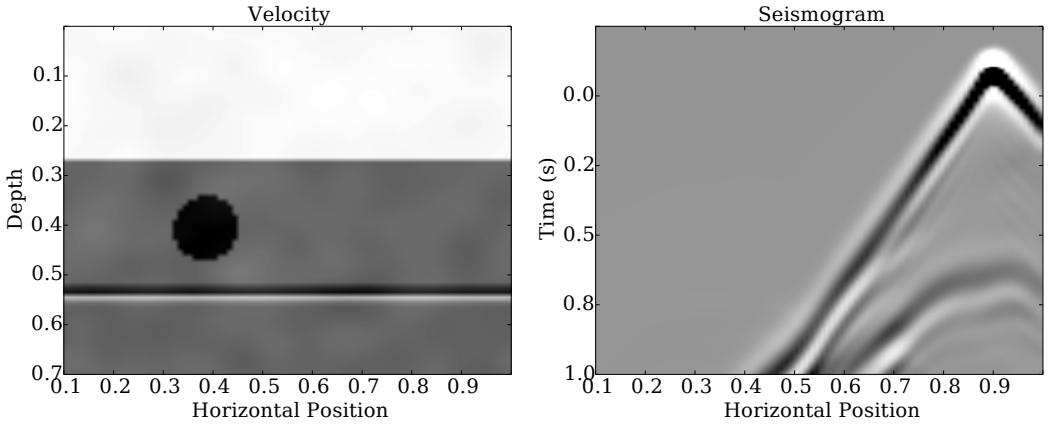

Figure 5: In general each acoustic profile class is defined by the types of features it possesses from class 0, 1, 2, and 4. All acoustic profiles have a Gaussian heterogeneous background. Then each class can contain any permutation of the three features, acoustic ridge, acoustic reflector, and compact inclusion. For example, class 7, consists of profiles having all three features present. A typical acoustic profile from class 7 is shown (LEFT) alongside the corresponding seismogram (RIGHT).

Table 1: Objective function value, averaged over 20 separate reconstructions, during inter-class reconstruction.

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
Dictionary 0	0.008497	0.005549	0.009940	0.011247	0.008930	0.003466	0.011664	0.012125
Dictionary 1	0.010132	0.002175	0.013390	0.002172	0.010496	0.001920	0.014288	0.001973
Dictionary 2	0.008385	0.003607	0.011136	0.009190	0.008902	0.003722	0.011961	0.003249
Dictionary 3	0.008386	0.002121	0.011851	0.003242	0.008786	0.002556	0.012339	0.002551
Dictionary 4	0.008185	0.004378	0.011975	0.004105	0.008143	0.004257	0.012062	0.002957
Dictionary 5	0.008754	0.003901	0.013953	0.001892	0.010446	0.002444	0.011816	0.002946
Dictionary 6	0.008185	0.004367	0.013445	0.004016	0.008711	0.003967	0.009819	0.003666
Dictionary 7	0.008519	0.002975	0.012538	0.001883	0.008958	0.002467	0.012452	0.011757

design. In figure 7 we show the sum of all $\{x_m\}$ for one particular reconstruction. We can see that weight is only put on pixels in the seismogram image where an important feature, such as a wavefront, exists. By examining the individual dictionary elements in figures 8-15 we see that the majority of dictionary elements consist of a type of wavefront. This makes intuitive sense because the seismograms consist almost entirely of superpositions of wavefronts. If we examine the individual x_m corresponding to a single dictionary element in a reconstruction we can see where the filter-dictionary reconstruction is placing the features represented by each dictionary element. We show this for common seismogram features in figures 16-18. Figure 16 shows the reconstruction of a common incoming wave feature while figures 17 and 18 correspond to features present due to interaction of the acoustic wave with the boundary of the computational domain or the source.

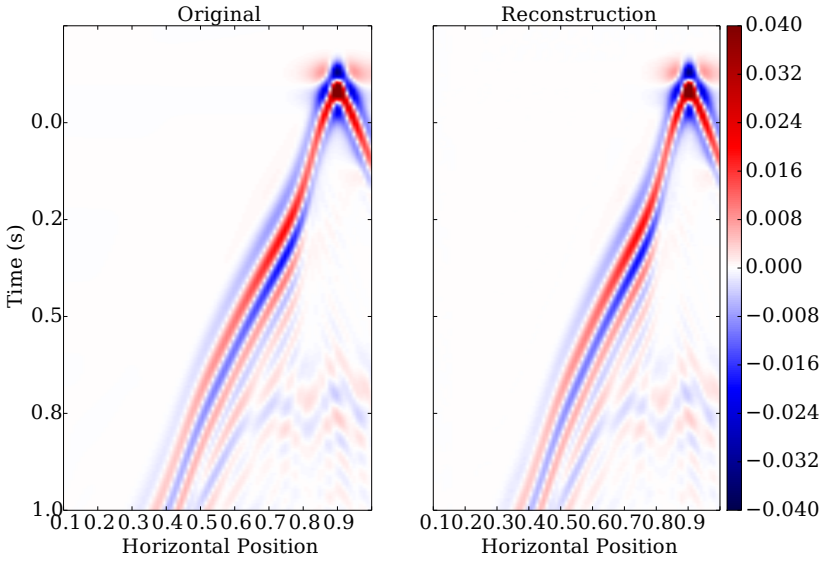

Figure 6: Here we show the reconstruction of a typical, class 1, seismogram using the 9 filters learned through the sparse coding algorithm. The reconstruction is visually very good when compared, by eye, to the original seismogram.

4 Classification Using a Super-set Dictionary

Though the filter-dictionary learning algorithm worked well for finding dictionaries capable of giving very sparse and accurate reconstructions of seismograms for a wide variety of acoustic subsurface models the method did not work well for classifying seismograms. We will detail the two classification methods that were tested and then discuss possible reasons for the poor classification accuracy that was observed.

4.1 Filter-Dictionary Classification

Once a set of dictionary filters is learned for each of the 8 acoustic profile classes the dictionaries are combined into a super-set dictionary that is used to construct our seismogram classifier. For each class, α, a set of convolutional filters, $D_\alpha = \{d_m^\alpha\}_{m=1}^{M_\alpha}$ is learned, exhibited in Figures 8-15. By taking the union of all filter dictionaries over the eight classes we arrive at a dictionary capable of representing any seismogram generated from the eight classes,

$$\mathcal{D} = \bigcup_{\alpha=1}^{8} D_\alpha. \tag{4.1}$$

A seismogram s^* from one of the eight classes, $\mathcal{S} = \bigcup_{\alpha=1}^{8} S_\alpha$, is then classified using the reconstruction error obtained from each of the filter-dictionary classes. However, we have some choice in exactly how this is done.

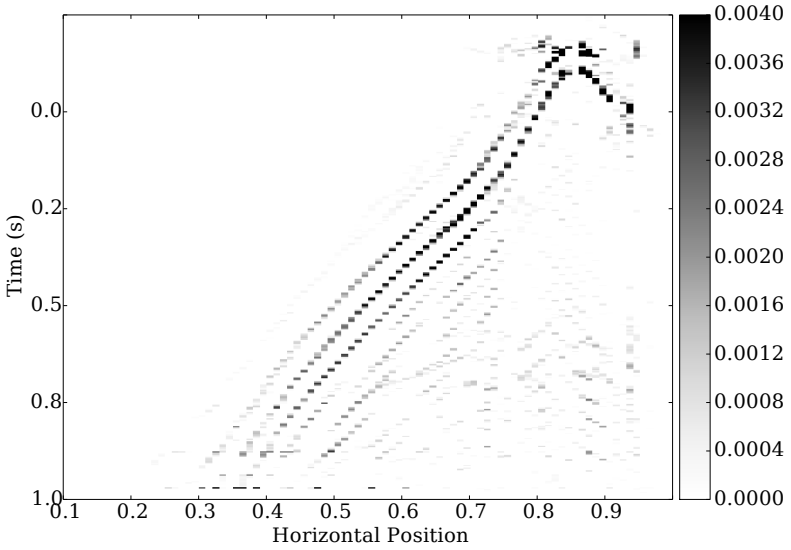

Figure 7: By summing the sparse representations, x_m, corresponding to each dictionary element we can see where filter weight is placed in the seismogram reconstruction. Here we can see that the majority of weight in the reconstruction goes into the arrival of the large amplitude waves at the receivers.

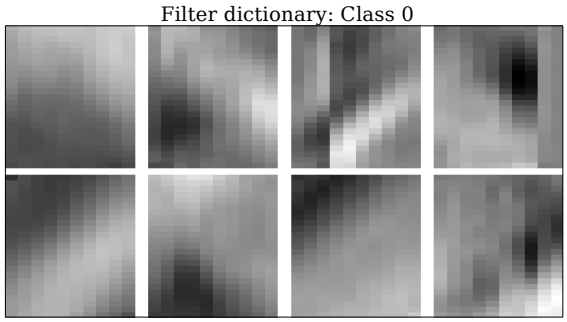

Figure 8: Dictionary filters learned from class 0 seismograms. Each filter contains 25×10 pixels.

4.1.1 Method 1

In our first method a reconstruction is performed, for the seismogram s^*, using the entire super-set filter-dictionary, $\mathcal{D}$. The sparse representation $\{x_m^\alpha\}$ is computed by solving

$$\underset{\{x_m^\alpha\}}{\arg\min} \frac{1}{2} \left\| \sum_{\alpha,m} d_m^\alpha * x_m^\alpha - s^* \right\|_2^2 + \lambda \sum_{\alpha,m} \|x_m^\alpha\|_1. \tag{4.2}$$

The class of s^* is then given by

$$\alpha^* = \underset{\alpha}{\arg\min} \frac{1}{2} \left\| \sum_{m} d_m^\alpha * x_m^\alpha - s^* \right\|_2^2, \tag{4.3}$$

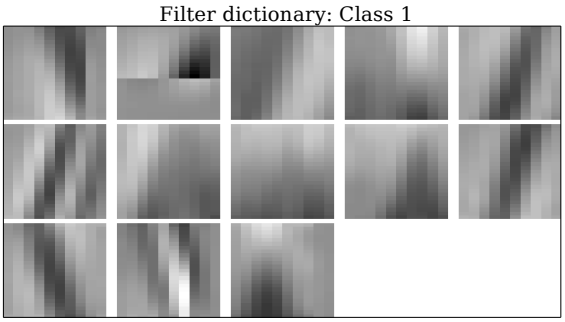

Figure 9: Dictionary filters learned from class 1 seismograms. Each filter contains 25×10 pixels.

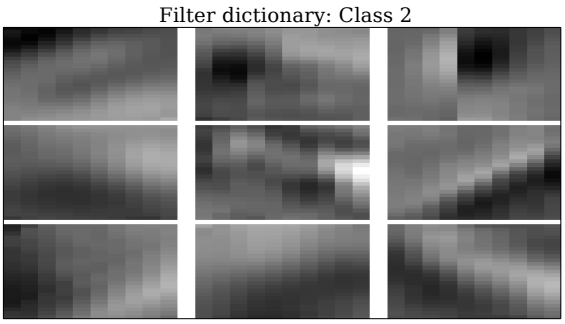

Figure 10: Dictionary filters learned from class 2 seismograms. Each filter contains 25×10 pixels.

with the $\{x_m^\alpha\}$ having been determined by solving (4.2). In summary, s^* is assigned the single class yielding the lowest data fidelity *after* a reconstuction has been learned with the super-set filter-dictionary.

4.1.2 Method 2

In method two, reconstructions are performed for the seismogram s^* using each class' filter-dictionary, $\{d_m^\alpha\}$. Sparse representations $\{x_m^\alpha\}$ are computed for each class by solving

$$\underset{\{x_m\}}{\arg\min} \frac{1}{2} \left\| \sum_m d_m^\alpha * x_m - s^* \right\|_2^2 + \lambda \sum_m \|x_m\|_1. \tag{4.4}$$

The class of s^* is then given by the class yielding the most accurate reconstruction individually. In summary, s^* is assigned the single class yielding the lowest data fidelity obtained through independent reconstructions from each class.

4.2 Accuracy of Classification Methods

Unfortunately, both classification methods yielded poor classification accuracy and classification accuracy rates were no better than random chance. The reason for this is most

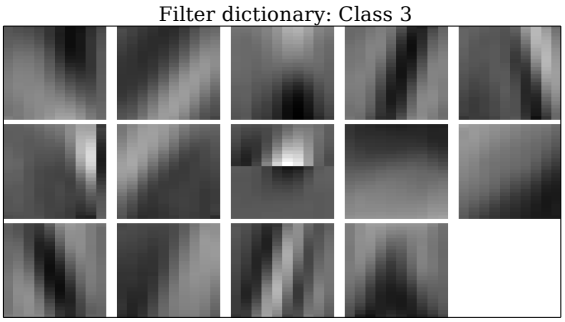

Figure 11: Dictionary filters learned from class 3 seismograms. Each filter contains 25×10 pixels.

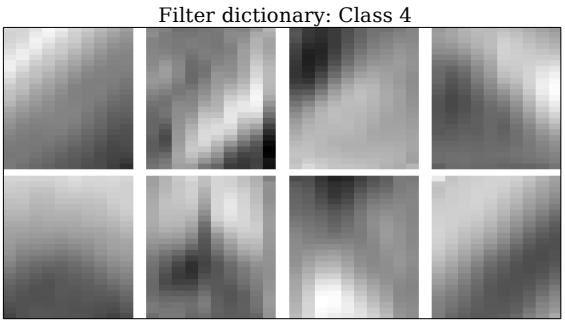

Figure 12: Dictionary filters learned from class 4 seismograms. Each filter contains 25×10 pixels.

probably due to the results in table 1, which shows that every class' dictionary yields an accurate reconstruction of every other class. The hope would be that the distinguishing features for seismograms from different classes are captured in the filters yielding more sparse reconstructions even though the overall reconstruction accuracy is similar. However, looking at the data fidelity term, $\frac{1}{2}\left\|\sum_m d_m * x_m - s\right\|_2^2$, and the term controling sparsity, $\sum_m \|x_m\|_1$, in the objective function (3.2) separately yields no discernible pattern between sparsity and inter-class reconstruction. The inter-class data fidelity values after reconstruction are given in table 2 and the sparsity values are given in table 3. In both table 2 and table 3 values are obtained by averaging over 20 reconstructions.

5 Increasing Inter-Dictionary Incoherence

It was suspected that there may be a few dictionary elements that yielded accurate reconstructions for every class of seismogram but were not particular to any one class. This is a reasonable assumption since all classes of seismograms are dominated by superpositions of large magnitude, single-reflection waves. In other words, it is possible that the distinguishing features of each class may be obscured by inter-class commonalities in the data. One metric for judging commonality between filter-dictionaries is to look at the *incoherence* between dictionary elements. Given two filter dictionaries we can look at the absolute value of the inner product between individual filters, $|\langle d_i, d_j \rangle|$. This quantity is referred to

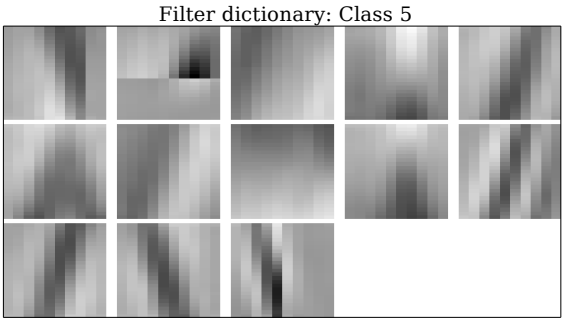

Figure 13: Dictionary filters learned from class 5 seismograms. Each filter contains 25×10 pixels.

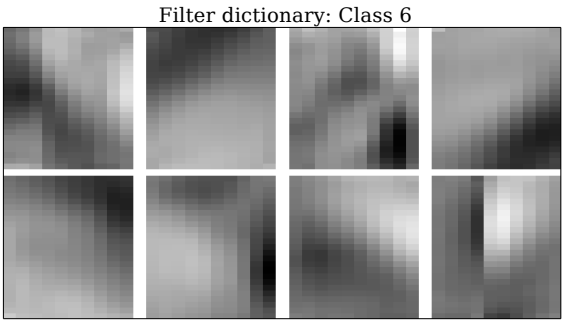

Figure 14: Dictionary filters learned from class 6 seismograms. Each filter contains 25×10 pixels.

as the incoherence, $I(d_i, d_j)$, between filters and has been used to asses, and increase, the discriminatory power of dictionary classification routines in [4]. This measures similarity by looking at the high-dimensional angle, θ_{ij}, between the filters,

$$I(d_i, d_j) = |\langle d_i, d_j \rangle| = |\cos(\theta_{ij})|. \tag{5.1}$$

Thus, the incoherence between two filters is always in the interval $[0, 1]$, with higher values of $I(d_i, d_j)$ indicating more similarity between d_i and d_j.

To increase the discriminatory power in our dictionary classification problem we pruned each class' dictionary. This was accomplished by comparing a filter, d^*, from one class with every other filter in the super set dictionary, $\mathcal{D}$, from a different class and removing d^* if $I(d^*, \hat{d})$ was above a threshold $h > 0$ for some $\hat{d}$. This pruned the dictionaries, discarding up to 80% of some class dictionaries, but did not seem to increase the discriminatory power of the classification algorithm. Thresholds of $h = 0.5$, 0.75, 0.85, and 0.9 were tried and classification accuracy was still little better than random chance in each of these cases. By $h = 0.9$ many class dictionaries only consisted of one or two elements so even the reconstruction of the pruned dictionaries began to suffer.

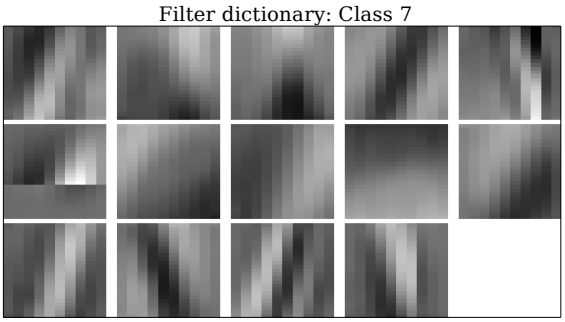

Figure 15: Dictionary filters learned from class 7 seismograms. Each filter contains 25×10 pixels.

Table 2: Data fidelity term value, averaged over 20 separate reconstructions, during inter-class reconstruction.

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
Dictionary 0	0.000160	0.000228	0.000210	0.000227	0.000138	0.000220	0.000217	0.000233
Dictionary 1	0.000232	0.000105	0.000290	0.000127	0.000213	0.000111	0.000278	0.000115
Dictionary 2	0.000172	0.000177	0.000213	0.000185	0.000147	0.000173	0.000218	0.000206
Dictionary 3	0.000167	0.000118	0.000219	0.000119	0.000160	0.000116	0.000214	0.000123
Dictionary 4	0.000152	0.000261	0.000218	0.000262	0.000140	0.000277	0.000208	0.000266
Dictionary 5	0.000245	0.000123	0.000289	0.000134	0.000225	0.000113	0.000294	0.000122
Dictionary 6	0.000178	0.000209	0.000226	0.000228	0.000165	0.000207	0.000224	0.000216
Dictionary 7	0.000187	0.000102	0.000251	0.000124	0.000178	0.000108	0.000245	0.000140

6 Conclusion & Future Directions

In this work the convolutional filter-dictionary based classifier did not work well to classify seismograms for the eight classes tried. However, the filter-dictionaries were learned well, the mini-batch algorithm is highly parellizable and performed well, and the dictionaries learned offered very accurate reconstructions within classes. We have detailed two methods of using learned filter-dictionaries for classification. Moreover, we have discussed one method of improving the discriminatory power of the dictionaries.

However, the fact remains that the classifiers performed poorly. So what could possibly be done to improve the classification algorithm. The most obvious problem is that the learned dictionaries for each class provide good reconstructions for every other class. This is not ideal for classification based on assigning class by most accurate and sparse reconstruction. The main cause of our dictionaries behavior is their similarity. Indeed if one looks at the filter-dictionaries in figures 8-15 we can notice many similarities. It is possible that the distinguishing features between the seismograms from different classes occur at different scales than our single-scale filters capture. One approach to understand if this is the case would be to learn multi-scale filter-dictionaries as in [8]. However, learning a multi-scale dictionary takes considerably more computational expense than a single scale dictionary and there still exists the problem of how to choose each scale in a multi-scale dictionary.

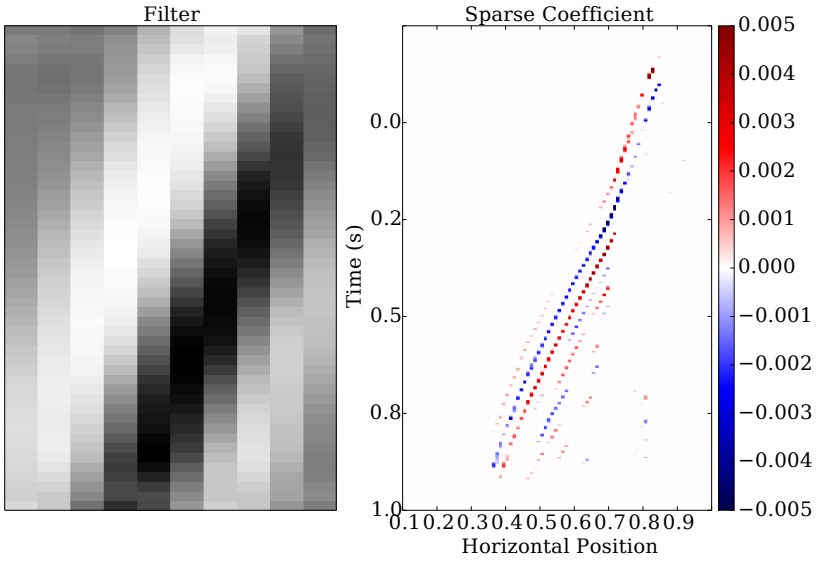

Figure 16: Each filter corresponds to a specific type of feature in the reconstruction. For instance, this filter type corresponds to distinct waves arriving at receivers to the left of the source. (LEFT) A single dictionary filter, (RIGHT) the sparse representation corresponding to this filter. We can see how weight in this representation is placed on the arriving, large-amplitude waves.

Another possible improvement would be to build the incoherence measure into our dictionary-learning objective function as was done in [4]. In this work the authors add a *incoherence* penalty term to their objective function so that the dictionary learning problem would look like

$$\begin{gathered}\underset{\{d_m^\alpha\},\{x_{k,m}^\alpha\}}{\arg\min} \frac{1}{2}\sum_k \left\| \sum_m d_m^\alpha * x_{k,m}^\alpha - s_k^\alpha \right\|_2^2 + \lambda \sum_k \sum_m \|x_{k,m}^\alpha\|_1 + \mu \|\mathcal{D}^T\mathcal{D}\|_{\mathcal{F}}, \\ \text{such that } \|d_m^\alpha\|_2 = 1 \text{ for all } m = 1, 2, \ldots, M_\alpha.\end{gathered} \tag{6.1}$$

Here the term $\|\mathcal{D}^T\mathcal{D}\|_{\mathcal{F}}$ represents the Frobenius norm of the matrix of inter-class incoherence values. The problem is, (6.1) is a much more difficult optimization problem to solve and we have introduced a second regularization parameter that must be tuned, μ. These two facts makes this approach infeasible for the authors of this manuscript currently.

A factor that may improve the classification accuracy, but has less overhead, is to rigorously optimize all the hyper-parameters involved in the algorithm. From the start, the dictionary learning algorithm makes choices about the size and shape of the filter-dictionaries to be learned, the regularization parameter λ in the objective function, and the hyper-parameters in the *alternating direction method of multipliers* optimization routine used to solve the dictionary learning problem. We have not carried out a rigorous enough error analysis to be sure our choices for these parameters are optimal. Moreover, the regularization parameter used in the objective functions for classification do not necessarily have to be the same as those used to learn the dictionaries. In fact one may want to increase the value of the regularization parameter during classification to emphasize sparsity. A full

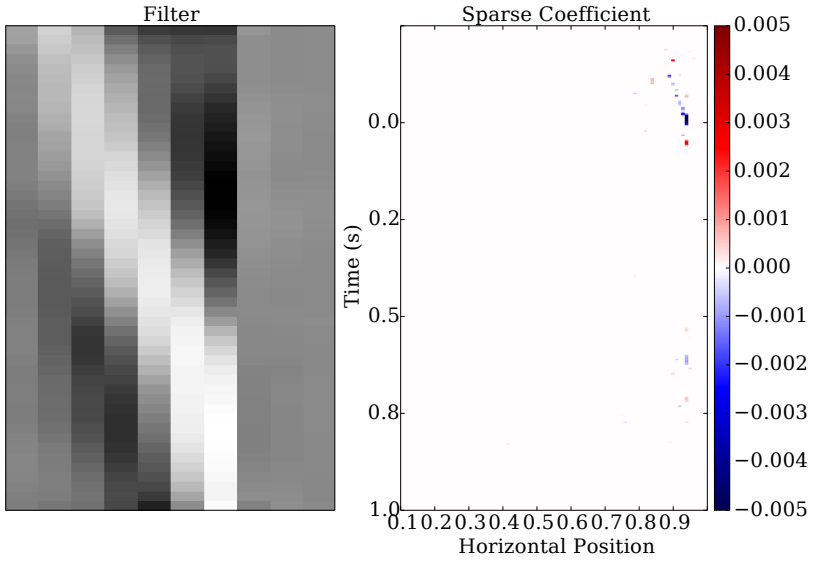

Figure 17: (LEFT) A single dictionary filter with a strong vertical discontinuity, (RIGHT) the sparse representation corresponding to this filter with most of the weight near the right boundary. Filters with vertical discontinuities assign weight, in the reconstruction, to boundary effects in the synthetic seismogram.

study of all these hyper-parameters would be needed to completely evaluate the filter-dictionary classification method for this class of seismograms.

Beyond improving the current classification work, this work has made the authors realize the importance of a development of *a priori* metrics for determining if dictionary based classification is likely to be successful for classification within the data set a researcher is interested in. Most of the success stories that the authors have read dealing with machine learning classification algorithms start with a classification problem in which the human eye can clearly pick out the important features for classification or the important features for classification are known ahead of time by experts in the field of study. This is a completely unsatisfying situation for someone attempting to use these machine learning tools on a completely new data set. If the field of machine learning classification is to truly advance it seems that one must be able to truly stand back and allow the algorithm to work. This can not be done until there are measures that can be calculated for a data set that would indicate the likelihood of success for a particular classification algorithm.

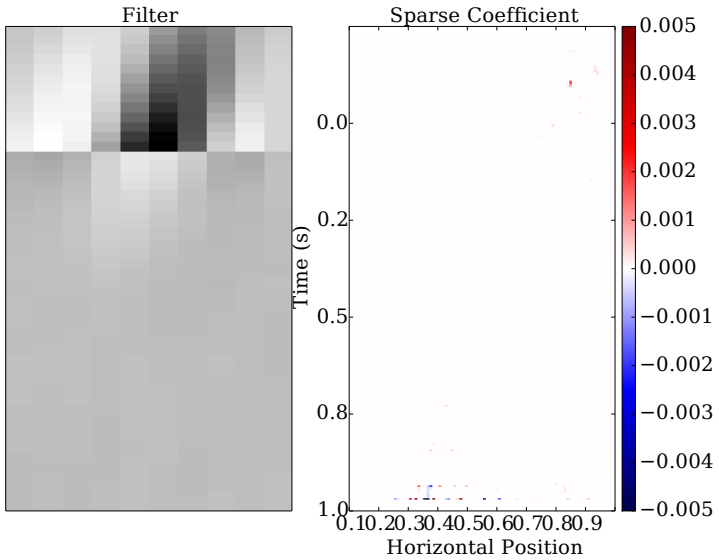

Figure 18: (LEFT) A single dictionary filter with a strong horizontal discontinuity, (RIGHT) the sparse representation corresponding to this filter with most of the weight near the bottom boundary. Filters with horizontal discontinuities assign weight, in the reconstruction, to lower boundary effects and finite source effects in the synthetic seismogram. These effects come from the initial start of the source and the temporal cutoff of the synthetic seismogram.

Table 3: Regularization penalty term value, averaged over 20 separate reconstructions, during inter-class reconstruction.

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
Dictionary 0	16.268611	8.704710	22.651495	10.785922	16.788609	8.063470	23.500759	10.039904
Dictionary 1	18.875309	4.582514	26.905470	6.811042	19.859427	5.076798	26.223256	5.020954
Dictionary 2	16.686846	7.074080	22.477713	8.702799	17.262481	7.367604	24.449256	11.169004
Dictionary 3	16.446867	4.671095	23.826724	6.123098	17.053028	4.727099	23.462045	5.622388
Dictionary 4	15.744254	8.530313	23.465365	10.002305	16.116717	8.453785	22.247540	9.662348
Dictionary 5	19.216164	4.671438	26.575604	7.889775	20.143596	4.651355	27.378496	5.879584
Dictionary 6	16.390634	7.097778	23.446981	9.181011	17.283340	7.478193	23.365966	8.707964
Dictionary 7	17.127285	4.369221	24.565003	6.460056	17.687462	5.009171	23.722547	7.224240

References

[1] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[2] Diego Carrera, Giacomo Boracchi, Alessandro Foi, and Brendt Wohlberg. Detecting anomalous structures by convolutional sparse models. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2015.

[3] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Discriminative learned dictionaries for local image analysis. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[4] Ignacio Ramirez, Pablo Sprechmann, and Guillermo Sapiro. Classification and clustering via dictionary learning with structured incoherence and shared features. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3501–3508. IEEE, 2010.

[5] Ron Rubinstein, Alfred M Bruckstein, and Michael Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6):1045–1057, 2010.

[6] Brendt Wohlberg. Efficient convolutional sparse coding. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7173–7177. IEEE, 2014.

[7] Brendt Wohlberg. Efficient algorithms for convolutional sparse representations. *IEEE Transactions on Image Processing*, 25(1):301–315, 2016.

[8] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2528–2535. IEEE, 2010.