# SPARSE HIDDEN MARKOV MODELS FOR PURER CLUSTERS

*Sujeeth Bharadwaj*[⋆]*, Mark Hasegawa-Johnson*[⋆]*, Jitendra Ajmera*[†]*, Om Deshmukh*[†]*, Ashish Verma*[†]

[⋆] Department of Electrical Engineering, University of Illinois, Urbana, IL, USA
[†] IBM India Resarch Lab (IRL), New Delhi, India
{sbhara3, jhasegaw}@ illinois.edu, {jajmera1, odeshmuk, vashish}@in.ibm.com

## ABSTRACT

The hidden Markov model (HMM) is widely popular as the *de facto* tool for representing temporal data; in this paper, we add to its utility in the sequence clustering domain – we describe a novel approach that allows us to directly control purity in HMM-based clustering algorithms. We show that encouraging sparsity in the observation probabilities increases cluster purity and derive an algorithm based on $l_p$ regularization; as a corollary, we also provide a different and useful interpretation of the value of $p$ in Renyi $p$-entropy. We test our method on the problem of clustering non-speech audio events from the BBC sound effects corpus. Experimental results confirm that our approach does learn purer clusters, with (unweighted) average purity as high as **0.88** – a considerable improvement over both the baseline HMM (**0.72**) and $k$-means clustering (**0.69**).

***Index Terms***— hidden Markov model, sequence clustering, sparsity, cluster purity, Renyi entropy

## 1. INTRODUCTION

Unsupervised clustering – grouping the data into clusters – is often a first step in the organization of unlabeled data, with important speech applications such as speaker diarization [1, 2] and speaker adaptation [3], to name a few. Clustering algorithms are useful if the resulting clusters predict the labels that will eventually be assigned. Performance metrics such as cluster purity, entropy, and accuracy attempt to quantify the usefulness of a given algorithm [4]. While many methods (e.g. $k$-means and spectral clustering) are known to be effective for producing good clusters, they generally only work well when the datapoints lie in some fixed length vector space [4]. Clustering sequences is much more challenging.

Most popular sequence clustering methods tend to be either model-based or distance-based [5, 6, 7, 8]. Model-based approaches make the assumption that the sequences are generated from $K$ different models, each of which represents a cluster [3, 5]. Distance-based methods rely on computing a similarity/distance metric between the sequences [8, 9]; a closely related approach is to extract relevant features and reduce the problem to that of clustering fixed length vectors [7]. There is, however, significant overlap between the two types of sequence clustering algorithms – a large subset of distance-based methods use generative models for obtaining better proximity measures [7, 8, 9]. In this paper, we focus on generative models and in particular, the HMM.

The popularity of HMMs, especially for describing time-varying signals, is unquestionable. Within the domain of sequence clustering, HMMs have been successfully used in both model-based and distance-based approaches [5, 6, 7, 8, 9, 10], and are quite natural for speech and audio data [10, 11]. Although they allow us to recover structure from sequences and represent observations of varying lengths, they typically do not favor parsimony. We argue that especially for the problem of clustering, parsimony or sparsity in the observation probabilities is essential.

In this paper, we show that sparsity in the observation probabilities leads to purer clusters and present an algorithm based on $l_p$ regularization for achieving it. The regularization parameter ($\eta$) that determines the tradeoff between model likelihood and $l_p$ prior, along with the value of $p$, allows us to directly control cluster purity. We discuss how our approach is equivalent to minimizing the Renyi $p$-entropy for HMMs and provide an intuitive and useful interpretation of $p$ in our framework. We present experimental results on clustering non-speech audio events from the BBC sound effects corpus.

### 1.1. Relation to Prior Work

Previous approaches to HMM-based clustering [3, 5, 6, 7, 8, 9, 10] do not explicitly consider cluster purity; we describe a general approach that directly maximizes purity in most of these methods, and present experimental results on an approach similar to [3, 5]. The authors in [12] use the Dirichlet prior to achieve sparsity in HMMs for image classification. We use the $l_p$ prior, which has a clean and intuitive relationship to Renyi $p$-entropy. Our work is also an extension of minimum entropy clustering [13, 14] to HMMs, and allows us to go beyond the quadratic and Shannon measures

Part of this research was supported by NSF CDI Program Grant Number BCS 0941268. The opinions expressed in this work are those of the authors and do not necessarily reflect the views of the funding agency

of entropy to more intuitive ones such as Renyi $p$-entropy with $0 < p < 1$.

## 2. HMMS FOR CLUSTERING SEQUENCES

Efficient implementation, impressive results, and a natural and intuitive interpretation justify the extensive use of the hidden Markov model (HMM) within speech, natural language processing, and several other communities [11]. HMMs are parameterized by $\lambda = (\pi, A, B)$, where $\pi$ is a distribution over the states, $A$ is the state transition matrix, and $B$ is a matrix (when the observation space is finite) where $B_{ij}$ represents the probability of emitting observation $j$ given state $i$. We would like to group a set of $N$ sequences, $O = \{O^j\}_{j=1}^N$, into $K$ clusters. In this paper, we assume that $K$ is known; if $K$ is unknown, we can draw from a rich set of model selection methods to estimate it [15].

HMM-based clustering algorithms make some assumptions about the relationship across the $K$ HMMs, each of which generates the samples that belong to its respective cluster. For example, Smyth makes the following mixture model assumption: $f_K(O^i) = \sum_{j=1}^K f_j(O^i|\lambda_j)p_j$, where $O^i$ is the $i^{th}$ sequence, and $\lambda_j$ is the set of model parameters for the $j^{th}$ HMM $f_j(.)$ [5]. The idea in [5] is to construct a similarity matrix, $S_{ij}^N = P(O^i|\lambda_j)_{i,j=1\ldots N}$, by first training a separate HMM on each of the $N$ sequences. Given any such matrix $S$, it is easy to group the sequences into $K$ clusters using some standard method such as spectral clustering [16]. Smyth then proposes to train $K$ new HMMs (one for each cluster) with its corresponding set of sequences [5]. The mixture model assumption allows us to fuse the $K$ HMMs into one big HMM and train on all $N$ sequences [5]. Mixed approaches do not necessarily focus on learning an overall generative model; instead, they use $S^N$ or more discriminative estimates of it to directly partition the data into $K$ clusters [7, 8, 9].

In applications such as ours, the data is not naturally segmented into $N$ different sequences – initialization as described in [5, 7, 8, 9] is difficult. We therefore allow transitions across the HMMs and train a single super-HMM that automatically segments and clusters the data. A similar model was used effectively in [3] for the problem of speaker adaptation. In all of the approaches outlined and cited here, HMMs are trained using the maximum likelihood criterion; in the next section, we discuss why sparsity is essential and how we can incorporate it into HMM-based clustering algorithms.

## 3. ENCOURAGING SPARSITY IN HMMS

Let us consider just one cluster and take purity to be the measure of its goodness. The purity of a cluster $C$ is given by $purity(C) = \frac{1}{|C|}\max_i(|C|_{class=i})$, where $|C|_{class=i}$ denotes the number of items of class $i$ in the cluster, and $|C|$ is the total size of the cluster. This definition requires us to have access to ground truth labels. In some applications, however, it is difficult to predefine a fixed number of classes and even more difficult to assign labels to all of the datapoints. In such cases, the majority class associated with any given cluster can be reasonably defined to be the most frequently produced sequence of symbols, after deleting repetitions [1]. Cluster purity can then be defined as the fraction of tokens assigned to a cluster that share the same symbol sequence.

We can maximize purity, as defined above, by minimizing the total number of *different* sequences that belong to a particular cluster – a quantity that depends on the HMM parameters. We simplify our argument by making the assumption that the state transition matrix is left-to-right; this structure allows us to view the observation sequence as a set of symbols emitted by the first state, followed by a set of symbols emitted by the second state, and so on. It is then clear that minimizing the number of symbols emitted by each state reduces the total number of possible observations generated by the HMM. Encouraging sparsity in the observation probabilities allows us to directly minimize the number of symbols emitted by each state.

In the following subsection, we summarize maximum a posteriori (MAP) estimation for HMMs. In subsection 3.2, we present the $l_p$ norm, $0 < p < 1$, as a suitable prior for encouraging sparsity. In subsection 3.3, we relate the $l_p$ prior to Renyi $p$-entropy and provide an interpretation of our work in the minimum entropy clustering framework.

### 3.1. MAP Estimation to Encourage Sparsity

A popular approach for learning the HMM parameters is Baum Welch estimation based on the expectation maximization (EM) algorithm [17]. The idea is to iterate between computing the expectation (the $Q$ function) and maximizing it. $Q(\lambda, \lambda')$ is given by

$$Q(\lambda, \lambda') = \sum_{q \epsilon S} \log P(O, q|\lambda) P(O, q|\lambda') \qquad (1)$$

where $S$ is the space of all state sequences, $O = \{O_t\}_{t=1}^T$ is the observation sequence, and $\lambda'$ is the previous estimate of the parameters. It is easy to see that $Q(\lambda, \lambda')$ can be written as a sum of functions of the three types of parameters: the initial distribution of states ($\pi$), the state transition matrix ($A$), and the matrix of observation probabilities ($B$) [17]. We can independently optimize over each of the three sets of parameters (at a given iteration). To incorporate prior knowledge/constraints (sparsity or otherwise), we use maximum a posteriori (MAP) estimation. Here, we present the update equations for $B$ with some general prior, $g(B)$. Extension to other sets of parameters ($\pi$ and $A$) is straightforward, but not necessary for our problem. We maximize

$$\sum_{i=1}^N \sum_{t=1}^T \log b_i(O_t) P(O, q_t = i|\lambda') - \eta g(B) \qquad (2)$$

where $b_i(O_t)$ is the probability that the $i^{th}$ state emits the $t^{th}$ observation in the sequence $\{O_t\}_{t=1}^T$. By setting the gradient to zero and satisfying the usual constraints that for each $i$, $\sum_j B_{ij} = 1$ and $B_{ij} \geq 0$, we get

$$B_{ij} = \frac{(\sum_{t=1}^T P(O, q_t = i|\lambda')\mathbf{1}\{O_t = j\} - \eta S_{ij})^+}{\sum_{t=1}^T P(O, q_t = i|\lambda') - \eta S_{ij})^+} \quad (3)$$

where $(x)^+ = \max(x, 0)$, $S_{ij} = B_{ij}\nabla_{B_{ij}} g(B)$, and $\mathbf{1}\{arg\}$ is an indicator function that is 1 if $arg$ is true and 0 otherwise.

Equation (3) is a fixed point equation which can be shown to converge to a local optimum whenever $g(B)$ is convex (making $-g(B)$ concave). The overall function (likelihood + prior) can be shown to increase irrespective of how many additional terms we introduce to the likelihood function, as long as they satisfy Jensen's inequality [18].

### 3.2. The Appropriate $g(B)$

Given a vector $x$ in the $N$-dimensional euclidean space, $\|x\|_q$, the $l_q$ norm of $x$ is given by $\|x\|_q = (\sum_{i=1}^N x_i^q)^{\frac{1}{q}}$. The $l_2$ norm is the most commonly used metric for regularization [15]. The intractable $l_0$ norm and its relaxation, the $l_1$ norm (lasso) encourage sparsity [15]. We, however, cannot directly use the $l_1$ norm since $B$ is a stochastic matrix – the entries are non-negative and each row sums to 1; the $l_1$ norm of each row is also 1, thus $l_1$regularization is meaningless. A few approaches to sparsifying probability vectors (or simplex) exist [19, 20]; for example, the authors in [19] present a new convex optimization problem that does not use $l_1$. We, however, use the $l_p$ norm because it can be easily integrated into the Baum Welch algorithm and because of its relation to Renyi entropy.

Intuitively, the $l_p$ norm for $0 < p < 1$ also encourages sparsity and its use is theoretically justified in [21]. We minimize the sum of the $l_p$ norm of each row of $B$. In this work, $g(B) = \|B\|_{1,p} = \sum_i (\sum_j B_{ij}^p)^{\frac{1}{p}}$. When $p < 1$, $g(B)$ is not convex and convergence of Equation (3) is not guaranteed; however, our experiments demonstrate good convergence properties in practice.

### 3.3. Entropy

Entropy is a measure very similar to cluster (im)purity. A low entropy suggests that all samples within a cluster are "close" to each other. Given a random variable $X$ with probability mass vector $Q$, Renyi $p$-entropy is defined to be

$$H_p(X) = \frac{p}{1-p}\log(\|Q\|_p)$$

It can be shown that $\lim_{p\to 1} H_p(X)$ is Shannon entropy [22]; owing to computational benefits, quadratic entropy ($p = 2$) is also commonly used [13, 14, 22].

In this work, we sparsify $B$ on a row-by-row basis by minimizing the $l_p$ norm, which is equivalent to minimizing the Renyi $p$-entropy up to changes in the regularization parameter $\eta$. This allows us to interpret the value of $p$ in new light – minimizing Renyi $p$-entropy for $0 < p < 1$ directly increases cluster purity. It is easy to show that for $p \leq r$, $H_p(Q) \geq H_r(Q)$ [22] and by minimizing the Renyi entropy for $p < 1$ (encouraging cluster purity), we also minimize an upper bound on the Shannon and quadratic measures of entropy. The converse, unfortunately, is not true. Methods based on quadratic entropy do not necessarily learn purer clusters. Entropy measures with $p < 1$ are therefore attractive and should be favored whenever it is feasible to incorporate them; in the case of HMMs, a simple modification to the EM algorithm suffices.

## 4. EXPERIMENTS

We test our method on clustering non-speech audio events from the BBC sound effects corpus [23]. The dataset contains 48 files ranging from 15 seconds to 5 minutes in length. The files consist of common events such as rain, waterfall, gunshot, birds, dog, baby crying, etc. We assume that the events can vary drastically in length; for example, a typical gunshot is much shorter than a baby crying. We hypothesize that there are 35 clusters uniformly distributed across 7 event lengths, ranging from 3 states per HMM to 9 states per HMM. In order to detect multiple events per file, we allow transitions from the last state of one HMM to the first state of another and we refer to the resulting HMM as super-HMM. Viterbi decoding is used to segment each audio file into sequences, and to assign each sequence to one of the 35 cluster HMMs. We discretize the observation space by computing 13 mel-frequency cepstral coefficients (MFCCs) with a window of 250 ms and an overlap of 100 ms over all 48 files, and group them into 70 clusters using the $k$-means algorithm. Each event can then be approximated by a sequence of integers.

Figure 1 shows the observation probability matrices of the super-HMM for two cases: no sparsity (top) and some sparsity encouraged (bottom). The exact choice of the parameters ($p = 0.4, \eta = 0.09$) is arbitrary and simply illustrates that our proposed algorithm indeed sparsifies the observation probabilities.

Although we previously defined majority class to be the most frequently produced sequence of symbols, we report results on the more realistic and practical situation in which there are exactly 48 sound classes, each corresponding to a particular file in the dataset. We report results on frame-wise clustering of the data since it allows for a much easier comparison with $k$-means clustering. We use two measures of average purity: unweighted and weighted.

If we partition the dataset $D$ into $K$ clusters, $\{C_j\}_{j=1}^K$, unweighted purity is $P_{unweighted} = \frac{1}{K}\sum_{j=1}^K purity(C_j)$ and

**Fig. 1**. Observation matrices $B_{ij}$ (displayed as images with $i$ = row index and $j$ = column index) for $\eta = 0$ (top) and $p = 0.4, \eta = 0.09$ (bottom)

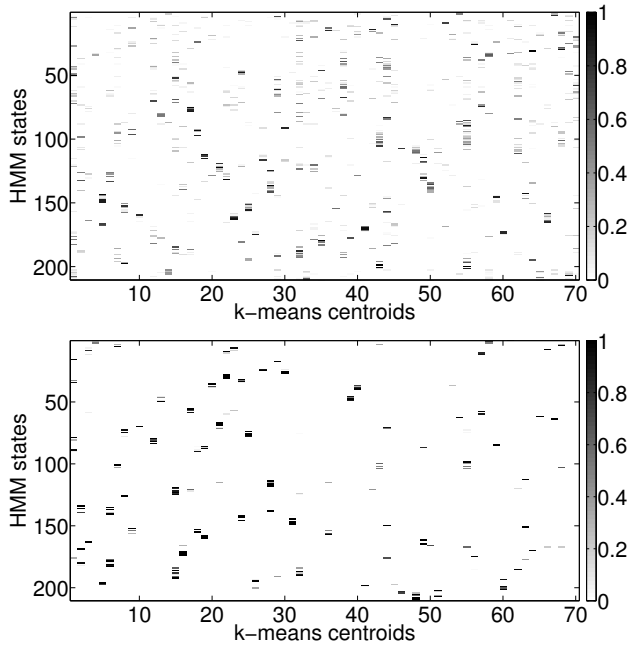
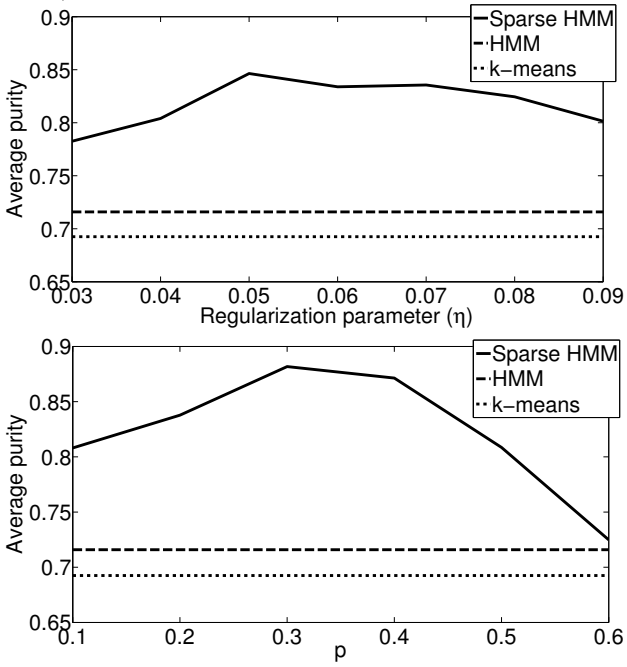



**Fig. 2**. Average (unweighted) purity as a function of $\eta$ with $p = 0.4$ (top) and as a function of $p$ with the best $\eta$ for each $p$ (bottom)




weighted purity is $P_{weighted} = \sum_{j=1}^{K} \frac{|C_j|}{|D|} purity(C_j)$. A high value of $P_{unweighted}$ implies that most of the individual clusters are very pure and only a few are impure; $P_{weighted}$, however, also takes into account the number of samples in each cluster – it acts as a check against trivial solutions such as one in which $K-1$ clusters contain one sample each and the $K^{th}$ cluster contains everything else in $D$.

Figure 2 shows the dependence of $P_{unweighted}$ on the regularization parameter $\eta$ (top) and on $p$ (bottom). It supports our claim (and intuition) that sparsifying the observation probabilities within each HMM purifies the cluster and on average, leads to many more pure clusters. The best values of $\eta$ (0.05) and $p$ (0.3) indicate that $B$ is neither too sparse nor too dense. It is intuitively clear that when the observation matrix is dense, clusters are bound to be less pure; but why does a little more sparsity lead to relatively less pure clusters? The parameters $\eta$ and $p$ explicitly control some tradeoff between likelihood and sparsity and in extreme situations, the model is heavily constrained and learning becomes no more than just randomly picking a few (sparse) observations for each state.

Table 1 contains the best results for all three methods and the two notions of average purity. The values of $(p, \eta)$ that maximize $P_{unweighted}$ and $P_{weighted}$ are $(0.3, 0.044)$ and $(0.3, 0.009)$, respectively, which is in line with our intuition – as discussed above, the observation matrix cannot be arbitrarily sparse when trying to maximize $P_{weighted}$. We see that in both cases, sparse HMMs do significantly better than the baseline HMM and $k$-means. A considerably higher value of $P_{weighted}$ (0.75) especially indicates that when the parameters are chosen appropriately, sparse HMMs do not just focus on a handful of samples and dump the rest into highly impure "garbage" clusters; sparsity is indeed an effective tool for learning purer clusters.

**Table 1**. Purity results

| Method | $P_{unweighted}$ | $P_{weighted}$ |
|---|---|---|
| $k$-means clustering | 0.69 | 0.66 |
| HMM | 0.72 | 0.57 |
| Sparse HMM | **0.88** | **0.75** |

## 5. CONCLUSIONS

We have shown that $l_p$-regularized Baum Welch algorithm can be used to learn clusters that are considerably more pure than those obtained by standard methods such as the baseline HMM or $k$-means clustering. Although we restrict our experiments to discrete HMMs in a generative framework, our approach can be extended to more general cases. Methods that use HMM as a tool for learning good distance metrics can also benefit from our algorithm; intuitively, sparse observation probabilities must lead to more discriminative (sparse) similarity matrices and naturally, to purer clusters. Our interpretation of Renyi $p$-entropy also provides for an extension to more general HMMs; to maximize purity, we can directly minimize the Renyi $p$-entropy, $0 < p < 1$, of each state.

## 6. REFERENCES


[1] J. Ajmera and C. Wooters, "A robust speaker clustering algorithm," in *Proc. IEEE Workshop Automatic Speech Recognition and Understanding (ASRU)*, 2003, pp. 411–416.

[2] J. Ajmera, H. Bourlard, I. Lapidot, and I. McCowan, "Unknown-multiple speaker clustering using HMM," in *Intl. Conf. Spoken Language Proc.*, 2002.

[3] M. Padmanabhan, L.R. Bahl, D. Nahamoo, and M. A. Picheny, "Speaker clustering and transformation for speaker adaptation in speech recognition systems," *IEEE Trans. on Speech and Audio Processing*, vol. 6, pp. 71–77, 1998.

[4] R. Xu, "Survey of clustering algorithms," *IEEE Trans. on Neural Networks*, vol. 16, pp. 645–678, 2005.

[5] P. Smyth, "Clustering sequences with HMM," in *Advances in Neural Information Processing (NIPS)*, 1997, vol. 9, pp. 648–654.

[6] C. Li and G. Biswas, "Clustering sequence data using hidden Markov model representation," in *Proc. of SPIE Conf. on Data Mining and Knowledge Discovery*, 1999, pp. 14–21.

[7] M. Bicego, V. Murino, and M.A.T Figueiredo, "Similarity-based clustering of sequences using hidden Markov models," in *Proc. of Intl. Conf. on Machine Learning and Data Mining in Pattern Recognition*, 2003, pp. 86–95.

[8] D. Garcia-Garcia, E. Parrado-Hernandez, and F. Diaz-de Maria, "A new distance measure for model-based sequence clustering," *IEEE Trans. on PAMI*, vol. 31, no. 7, pp. 1325–1331, 2009.

[9] A. Panuccio, M. Bicego, and V. Murino, "A hidden Markov model-based approach to sequential data clustering," in *Proc. of. Intl. Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, pp. 734–742.

[10] L.R. Rabiner, C.H. Lee, B.H. Juang, and J.G. Wilpon, "HMM clustering for connected word recognition," in *Proc. of ICASSP*, 1989, vol. 1, pp. 405–408.

[11] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257–286, 1989.

[12] M. Bicego, M. Cristani, and V. Murino, "Sparseness achievement in hidden Markov models," in *Proc. of the Intl. Conf. on Inage Analysis and Processing*, 2007, pp. 67–72.

[13] R. Jensen, II Hild, K.E., D. Erdogmus, J.C. Principe, and T. Eltoft, "Clustering using Renyi's entropy," in *Proc. of the Intl. Joint Conf. on Neural Networks*, 2003, vol. 1, pp. 523–528.

[14] E. Gokcay and J.C. Principe, "Information theoretic clustering," *IEEE Trans. on PAMI*, vol. 24, pp. 158–171, 2002.

[15] T. Hastie, R. Tibshirani, and J. Friedman, *Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, 2009.

[16] A.Y. Ng, M.I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing (NIPS)*, 2002, vol. 14.

[17] J.A. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," Tech. Rep. TR-97-021, International Computer Science Institute, Berkeley, California, 1998.

[18] G.J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, John Wiley & sons, Inc., 2008.

[19] M. Pilanci, L. El Ghaoui, and V. Chandrasekaran, "Recovery of sparse probability measures via convex programming," in *Advances in Neural Information Processing Systems (NIPS)*, 2012, vol. 25, pp. 2429–2437.

[20] A. Kyrillidis, S. Becker, and V. Cevher, "Sparse projections onto the simplex," *arXiv:1206.1529*, 2012.

[21] R. Chartrand and V. Staneva, "Restricted isometry properties and nonconvex compressive sensing," *Inverse Problems*, vol. 24, pp. 1–14, 2008.

[22] J.C. principe, *Information Theoretic Learning*, Springer, 2010.

[23] M. Slaney, "Semantic-audio retrieval," in *ICASSP*, 2002, pp. 1408–1411.