RC25152 (W1104-113) April 25, 2011
Computer Science

IBM Research Report

A Convex-Hull Approach to Sparse Representations for Exemplar-Based Speech Recognition

Tara N. Sainath, David Nahamoo, Dimitri Kanevsky, Bhuvana Ramabhadran
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

Parikshit Shah
MIT
32 Vassar Street
Cambridge, MA 02139

A Convex Hull Approach to Sparse Representations for Exemplar-Based Speech Recognition

Tara N. Sainath TSAINATH@US.IBM.COM
1101 Kitachawan Road, Yorktown Heights, NY, 10598, USA

David Nahamoo NAHAMOO@US.IBM.COM
1101 Kitachawan Road, Yorktown Heights, NY, 10598, USA

Dimitri Kanevsky KANEVSKY@US.IBM.COM
1101 Kitachawan Road, Yorktown Heights, NY, 10598, USA

Bhuvana Ramabhadran BHUVANA@US.IBM.COM
1101 Kitachawan Road, Yorktown Heights, NY, 10598, USA

Parikshit Shah PARI@MIT.EDU
32 Vassar Street, Cambridge, MA 02139, USA

Abstract

In this paper, we propose a novel exemplar based technique for classification problems where for every new test sample the classification model is re-estimated from a subset of relevant samples of the training data. We formulate the exemplar-based classification paradigm as a sparse representation (SR) problem, and explore the use of convex hull constraints to enforce both regularization and sparsity. Finally, we utilize the Extended Baum-Welch (EBW) optimization technique to solve the SR problem. We explore our proposed methodology on the TIMIT phonetic classification task, showing that our proposed method offers statistically significant improvement over common classification methods, and provides an accuracy of 82.9%, the best number reported to date.

1. Introduction

In order to solve real-world machine learning classification and recognition problems, we need a principled way of modeling the phenomenon generating the observed data and the uncertainty in it. Typically, many aspects of the phenomenon generating the data are imperfectly known. Furthermore, observed data can be corrupted by environmental and channel noise. A good classification model is one that best represents these variations and inherent properties of the classes as captured in the observed data.

Approaches for modeling the observed data can be divided into two broad areas, parametric and non-parametric. In parametric models, the data is completely modeled by a finite set of parameters learnt from the observed data, while non-parametric models do not assume an underlying functional distribution for the model. All parametric and non-parametric methods use some amount of training data when learning the parameters or forming the non-parametric representation of the model. In some cases, such as Gaussian Mixture Model (GMM) estimation in speech recognition, all of the training data is used. For brevity purposes, we refer to methods that use all the training data as all-data methods. In other cases, exemplars of the observed data need to be judiciously selected such that they will generalize and produce robust models (Seeger, 2003). Exemplars can be in the form of instances from the training data or representations derived from the training data, such as prototypes or hyperplanes or basis functions (Wilson & Martinez, 2000). Approaches such as k-nearest neighbors (kNNs), support vector machines (SVMs) and a variety of sparse representation (SR) methods commonly used in pattern recognition tasks, fall under the category of exemplar-based methods. Exemplar-based methods use individual training examples in different ways. Techniques such as SVMs use training examples to generate a model, and fix this model when classifying a test sample. However, kNNs and SRs re-estimate a model from training points for each test sample.

While exemplar-based methods are popular in the machine learning community, all-data methods continue to be widely used in the speech recognition community. The most widely used all-data method is the GMM where all training samples are pooled together for estimating the parameters of the GMM. Exemplar based methods that re-estimate a new model for each test sample and use SRs for robust estimation of the model parameters are slowly being adopted as a new paradigm in the speech recognition community. These methods have been shown to offer improvements in accuracy over GMMs for classification tasks (Sainath et al., 2010), (Gemmeke & Virtanen, 2010). In this work, we introduce a novel extension of the exemplar-based methods by using Convex Hull (CH) constraints for the SR problem. We refer to this new technique as SR-CH. We show that SR-CH formulation lends itself to an elegant use of the Extended Baum-Welch (EBW) algorithm (Gopalakrishnan et al., 1991) for its optimization. Finally, we introduce a methodology to combine the all-data GMM method with the SR-CH exemplar-based technique to improve the robustness.

We define the basic exemplar-based SR problem using Equation 1 below.

$$y = H\beta \;\; \texttt{s.t.} \quad \parallel \beta \parallel_q^a < \epsilon \tag{1}$$

Given a test sample, y, the goal of the sparse representation scheme is to solve Equation 1 for β, where $\parallel \beta \parallel_q^a < \epsilon$ imposes a different regularization (constraint) on the vector β for various choices of q and a, thus selecting a small number of examples from H. Many techniques in the literature exist to solve this problem, including Lasso, Approximate Bayesian Compressive Sensing (ABCS) (Carmi et al., 2009) and Nonnegative Matrix Factorization (NMF) (Hoyer, 2004). The SR-CH method proposed in this work imposes additional constraints on the values of β to control the flexibility of $H\beta$ transformation in scaling as well as polar and axial reflections. This paper focuses on the performance of the SR-CH method on a well-studied classification task in speech, namely the TIMIT phone classification task (Lamel et al., 1986). We will show that the proposed method offers improvements in classification accuracy over other SR methods.

There have been several variations of kNNs, GMMs, and SVMs proposed in the literature. An extensive survey of these is beyond the scope of this paper. However, we present comparisons with kNNs, SVMs, GMMs and other exemplar-based SR methods. We find that the SR-CH method offers statistically significant improvements over these other classification techniques and in fact achieves a classification accuracy of 82.87% and 85.14% on TIMIT using two different representations of sample vectors, both of which are the best reported numbers to date.

The rest of this paper details the following key contributions:

- Improved performance via the use of a convex hull to constrain the values of the sparseness defining coefficients
- A novel solution to the SR-CH optimization problem based on the well-known Extended Baum-Welch (EBW) method used to estimate parameters in speech recognition
- Impose additional regularization on the SR-CH method through the all-data GMM for improved robustness

2. SRs for Classification

Before we describe our convex hull sparse representation methodology, we first begin by reviewing the use of SRs for classification, as first presented in (Sainath et al., 2010). To populate the matrix H in Equation 1, let us consider training examples from k different classes as columns in matrix H, i.e., $H = [h_{1,1}, h_{2,1}, \dots, h_{n,k}] \in \Re^{m \times n}$. Here $h_{i,j} \in \Re^m$ represents feature vector i from class j with dimension m, and n is the total number of training examples from all classes. H is assumed to be an overcomplete dictionary where $m < n$.

(Wright et al., 2009) shows that a test sample $y \in \Re^m$ can be represented as a linear combination of the entries in H weighted by β. However, if sparsity is enforced on β such that only a few elements in H are chosen, then ideally the optimal β should be non-zero for the elements in H which belong to the same class as y. This motivates us to solve for β using a SR technique, which solves for $y = H\beta$ subject to a sparsity constraint on β.

After solving for β, the β entries are used to make a classification decision. In this work, we explore a classification decision rule linked to the objective function of our convex hull SR method. As we introduce the convex hull method in Section 3, we will reserve our discussion of this classification rule to Section 3.4.

3. SRs using a Convex Hull Framework

In this section, we provide motivation behind using a convex hull approach for SRs, and then present our formulation and solution using the convex hull.

3.1. Motivation

The goal of classification methods are twofold. First, they look to capture salient differences between different classes. Second, these methods look to account for variability in the data due to channel effects, transmission effects, system noise, etc. In many machine learning applications, features are transformed to accentuate differences between classes and also reduce the variability in a class due to noise and channel effect. For examples, in speech recognition applications, techniques such as Linear Discriminant Analysis (LDA) are used to accentuate class differences, and cepstral mean normalization and subtraction to reduce data variability (Sainath et al., 2009).

A typical SR formulation in Equation 1 does not constrain β to be positive and normalized, which can result in the projected training points $h_i\beta_i \in H\beta$ being reflected and scaled. While this can be desirable when data variability exists, allowing for too much flexibility when data variability is minimized can reduce the discrimination between classes. Driven by this intuition, below we present two examples where data variability is minimized, and demonstrate how SRs manipulate the feature space in this instance, thus leading to classification errors.

First, consider two clusters in a 2-dim space as shown in Figure 1 with sample points $\{a_1, a_2, \ldots, a_6\}$ belonging to Class 1 and $\{b_1, b_2, \ldots, b_6\}$ belonging to Class 2. Assume that points a_i and b_i are concatenated into a matrix $H = [h_1, h_2, \ldots, h_{12}] = [a_1, \ldots, a_6, b_1, \ldots b_6]$, with a specific entry being denoted by $h_i \in H$. In a typical SR problem, given a new point y indicted in Figure 1, we project y into the linear span of training examples in H by trying to solve the following:

$$\arg\min \parallel \beta \parallel_0 \quad \texttt{s.t.} \quad y = H\beta = \sum_{i=1}^{12} h_i\beta_i \qquad (2)$$

As shown in Figure 1, the best solution will be obtained by setting all $\beta_i = 0$ except for $\beta_8 = -1$, corresponding to the weight on point b_2. At this point $|\beta|_0$ takes the lowest value of 1 and $y = -b_2$, meaning it is assigned to Class 2. The SR method misclassifies point y, as it is clearly in Class 1, because it puts no constraints on the β values. Specifically, in this case, the issue arises from the possibility of β entries taking negative values.

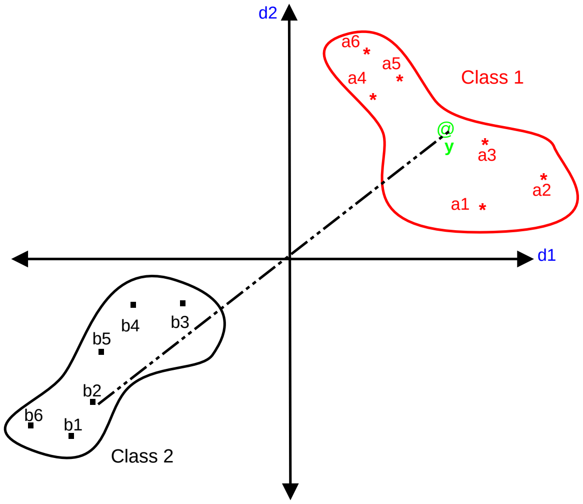

Figure 1. Reflective Issue with Negative β

Second, consider two clusters in a 2-dimensional space as shown in Figure 2 with sample points belonging to Class 1 and 2. Again, we try to find the best representation for test point y by solving Equation 2. The best solution will be obtained by setting all $\beta_i = 0$ except for $\beta_5 = 0.5$. At this value, $|\beta|_0$ will take the lowest possible value of 1 and $y = 0.5 \times a_5$.

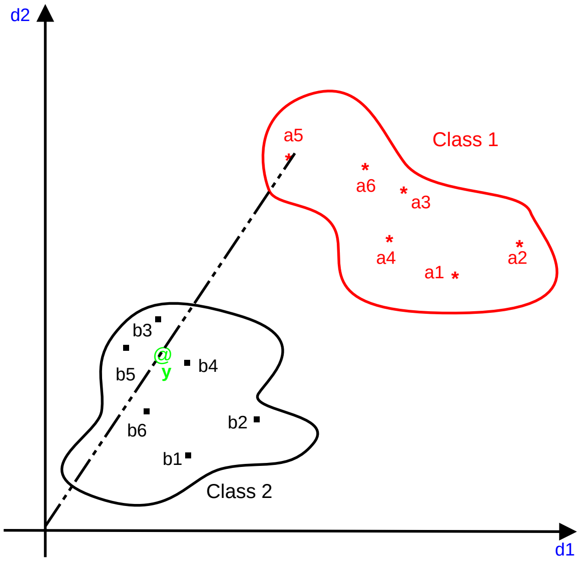

Figure 2. Scaling Issue with Unnormalized β

This leads to a wrong classification decision as y clearly is a point in Class 2. The misclassification is due to having no constraint on the β elements. Specifically, in this case, the issue arises from total independence between the β values and no normalization criteria as a way to enforce dependency between the β elements.

While negative matrix factorization (NMF) (Hoyer,

2004) is a popular technique for SRs to ensure that β entries are sparse and positive, it does not address the normalization issue. If we enforce β to be positive and normalized, then training points $h_i \in H$ form a convex hull. Mathematically speaking, a convex hull of training points H is defined by the set of all convex combinations of finite subsets of points from H, in other words a set of points that satisfy the following: $\sum_{i=1}^{n} h_i \beta_i$. Here n is any arbitrary number and the β_i components are positive and sum to 1.

Since many classification techniques can be sensitive to outliers, we examine, through an example, the sensitivity of our convex hull SR method. Consider two clusters shown in Figure 3 with sample points in Classes 1 and 2. Again, given point y, we try to find the best representation for y by solving Equation 2, where now we will use a convex hull approach to solve, putting extra positivity and normalization constraints on β.

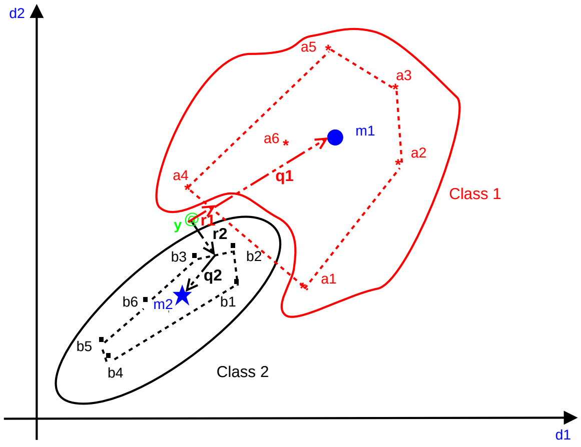

Figure 3. Outliers Effect

As shown in Figure 3, if we project y onto the convex hulls of Class 1 and Class 2, the distance from y to the convex hull of Class 1 (indicated by r_1) is less than the distance from y to the convex hull of Class 2 (i.e. r_2). This leads to a wrong classification decision as y clearly is a point in Class 2. The misclassification is due to the effect of outliers a_1 and a_4, which create an inappropriate convex hull for Class 1.

However, all-data methods, such as GMMs, are much less susceptible to outliers, as a model for a class is built by estimating the mean and variance of training examples belonging to this class. Thus, if we include the the distance between the projection of y onto the two convex hulls of Class 1 and Class 2, as well as the distance between this projection and the means m_i of Class 1 and 2 (distance indicated by q_1 and q_2) respectively, then test point y is classified correctly. Thus combining purely exemplar-based distances (r_i) with GMM-based distances (q_i), which are less susceptible to outliers, provides a more robust measure.

3.2. Convex Hull Formulation

In our SR-CH formulation, first we seek to project test point y into the convex hull of H. After y is projected into the convex hull of H, we compute how far this projection (which we call $H\beta$) is from the Gaussian means[1] of all classes in H. The full convex hull formulation, which tries to find the optimal β to minimize both the exemplar and GMM-based distances, is given by Equation 3. Here $N_{classes}$ represents the number of unique classes in H, and $\parallel H\beta - \mu_t \parallel_2^2$ is the distance from $H\beta$ to the mean μ_t of class t.

$$\arg\min_{\beta} \parallel y - H\beta \parallel_2^2 + \sum_{t=1}^{N_{classes}} \parallel H\beta - \mu_t \parallel_2^2$$

$$\texttt{s.t.} \quad \sum_i \beta_i = 1 \ \texttt{and} \ \beta_i \geq 0 \qquad (3)$$

In our work, we associate these distance measures with probabilities. Specifically, we assume that y satisfies a linear model as $y = H\beta + \zeta$ with observation noise $\zeta \sim N(0, R)$. This allows us to represent the distance between y and $H\beta$ using the term $p(y|\beta)$, which we will refer to as the exemplar-based term as given in Equation 4.

$$p(y|\beta) \propto \exp(-1/2(y - H\beta)^T R^{-1}(y - H\beta)) \qquad (4)$$

We also explore a probabilistic representation for the $\sum_{t=1}^{N_{classes}} \parallel H\beta - \mu_t \parallel_2^2$ term. Specifically, we define the GMM-based term $p_M(\beta)$, by seeing how well our projection of y onto the convex hull of H, as represented by $H\beta$, is explained by each of the $N_{classes}$ GMM models. Specifically, we score $H\beta$ against the GMM from each of the classes and sum the scores (in log-space) from all classes. This is given more formally in Equation 5 (log-space), where $p(H\beta|GMM_t)$ indicates the score from GMM t.

$$\log p_M(\beta) = \sum_{t=1}^{N_{classes}} \log p(H\beta|GMM_t) \qquad (5)$$

Given the exemplar-based term $p(y|\beta)$ and GMM-based term $p_M(\beta)$, the total objective function we would like to maximize is given in the log-space by Equation 6.

$$max_{\beta} F(\beta) = \{\log p(y|\beta) + \log p_M(\beta)\}$$

[1] Note that the Gaussian means we refer to in this work are built from the original training data, not the projected $H\beta$ features.

$$\texttt{s.t.} \quad \sum_i \beta_i = 1 \texttt{ and } \beta_i \geq 0 \tag{6}$$

Equation 6 can be solved using a variety of optimization methods. We use a technique widely used in the speech recognition community, namely the Extended Baum-Welch transformations (EBW) (Gopalakrishnan et al., 1991), to solve this problem. In (Kanevsky, 2004), the author shows that the EBW optimization technique can be used to maximize objective functions which are differentiable and satisfy constraints given in Equation 6. In the next section, we describe the EBW algorithm in more detail.

3.3. Solution using EBW Transformations

Given an initial parameter $\beta = \beta^0$ and an objective function, many optimization techniques typically involve taking a step along the gradient of the objective function, to estimate an updated parameter β^1, and then continue this parameter estimation process in an iterative fashion. The EBW algorithm is one example of a gradient-based technique which provides an iterative closed form solution to estimate β.

To describe the EBW algorithm in more detail, first define $F(\beta)$ be a differentiable function in β satisfying constraints given in Equation 6. Given a value for component i in β at iteration $k-1$, denoted by β_i^{k-1}, an updated estimate for β_i^k is given by Equation 7.[2] In Appendix A, we provide a closed-form solution for β_k^i given the exemplar-based term in Equation 4 and a GMM-based term in Equation 5.

$$\beta_i^k = \frac{\beta_i^{k-1}\left\{\partial F(\beta^{k-1})/\partial \beta_i^{k-1} + D\right\}}{\sum_j \beta_j^{k-1}\left\{\partial F(\beta^{k-1})/\partial \beta_j^{k-1}\right\} + D} \tag{7}$$

The parameter D controls the growth of the objective function. In (Kanevsky, 2004) it was shown that the EBW transformations can guarantee growth in the objective function at each iteration. Specifically, for a function F which is differentiable at β, if we define β^k by Equation 7, then for sufficiently large positive D, while growth of the objective function is slow, $F(\beta^k) \geq F(\beta^{k-1})$ and the objective function increases on each iteration. While a smaller value of D can lead to a larger jump along the objective function, it cannot guarantee growth in the objective function.

We explore setting D to a small value to ensure a large jump in the objective function. However, for a specific choice of D if we see that the objective function value has decreased when estimating β^k, i.e. $F(\beta^k) < F(\beta^{k-1})$, or one of the β_i^k components is negative, then we double the value of D and use this to estimate a new value of β^k in Equation 7. We continue to increase the value of D until we guarantee a growth in the objective function, and all β_i components are positive. This strategy of setting D is similar to other applications in speech where the EBW transformations are used (Povey et al., 2008). The process of iteratively estimating β continues until there is very little change in the objective function value.

3.4. Convex Hull Classification Rule

In this paper, since we are trying to solve for the β which maximizes the objective function in Equation 6, it seems natural to also explore a classification rule which defines the best class as that which maximizes this objective function. Using Equation 6, with the exemplar-based term from Equation 4 and the GMM-based term from Equation 5, then the objective-function linked classification rule for the best class t^* is given in Equation 8. Here $\delta_t(\beta)$ is a vector which is only non-zero for entries of β corresponding to class t.

$$t^* = \max_t \{\log p(y|\delta_t(\beta)) + \log p(H\delta_t(\beta)|GMM_t)\} \tag{8}$$

4. Experiments

Classification experiments are conducted on TIMIT (Lamel et al., 1986) acoustic phonetic corpus. The training set consists of over $140,000$ phoneme samples, whilet he development set is composed of over $15,000$ phoneme samples. Results are reported on the core test set, which contains 7,215 phoneme samples. In accordance with standard experimentation on TIMIT (Lee & Hon, 1989), the 61 phonetic labels are collapsed into a set of 48 for acoustic model training, ignoring the glottal stop [q]. For testing purposes, the standard practice is to collapse the 48 trained labels into a smaller set of 39 labels.

Two different feature representations are explored. First, we use the recognition setup described in (Sainath et al., 2009) to create a set of discriminatively-trained feature-Space Boosted Maximum Mutual Information (fBMMI) features (Povey et al., 2008). Given the frame-level features, we split each phonetic segment into thirds, taking the average of these 40 dimensional frame-level features around 3rds, and splice them together to form a 120 dimensional vector. This allows us to capture time dynamics into each segment. Second, we explore using the recognition setup in (Sainath et al., 2009) to

[2] The EBW update formula requires an initial value for β at iteration 0. We set the initial value of each β_i component be 1/N, where N is the total number of β components.

create a set of speaker-adapted (SA), discriminatively trained features per frame. These frame-level features are again averaged and spliced to form a feature vector per phone segment.

First, we analyze the performance of our SR-CH method, including the convex hull constraints on β and the addition of the GMM-based term. Next, we compare the performance of our SR-CH method to other standard classifiers used on the TIMIT task, including the GMM, SVM, kNN and ABCS (Carmi et al., 2009) SR methods. For the GMM, we explored training it via a maximum likelihood objective function, and a discriminative BMMI objective function (Povey et al., 2008). The parameters of each classifier were optimized for each feature set on the development set. The ABCS method is another SR technique that combines exemplar and GMM-based terms (Sainath et al., 2010), and since this method has provided the best classification accuracy among many SR techniques (Kanevsky et al., 2010), we compare SR-CH to this method. Note that for the ABCS classification rule, the best class is defined as that which has the maximum l_2 norm of β entries (Sainath et al., 2010).

5. Results

5.1. Analysis of SR-CH Algorithm

5.1.1. Algorithmic Behavior

In this section, we analyze the behavior of the SR-CH method. As discussed in Section 3.3, for an appropriate choice of D, the objective function of the SR-CH method is guaranteed to increase on each iteration. To observe this behavior experimentally on TIMIT, we chose a random test phone segment y, and solve $y = H\beta$ using the SR-CH algorithm. Figure 4 plots the value of the objective function at each iteration. Notice that the objective function increases rapidly until about iteration 30 and then increases slower, experimentally confirming growth in the objective function.

We also analyze the sparsity behavior for the SR-CH method. For a randomly chosen test segment y, Figure 5 plots the sparsity level (defined as the number of non-zero β coefficients), for each iteration of the SR-CH algorithm. Notice that as the number of iterations increases, the sparsity level continues to decrease and eventually approaches 20. Our intuitive feeling is that the normalization and positive constraints on β in the convex hull formulation allow for this sparse solution. Recall that all β coefficients are positive and the sum of the β coefficients is small (i.e., $\sum_i \beta_i = 1$). Given that the initial β values are chosen to be uniform, and the fact we seek to find a β to maximize Equation 6, then naturally only a few β elements will dominate and most β values would evolve to be close to zero.

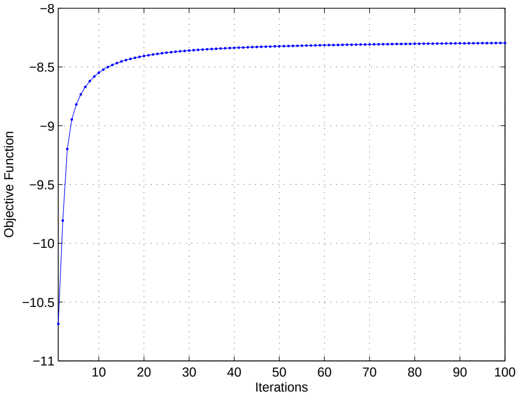

Figure 4. Number of Iterations vs. Objective Function

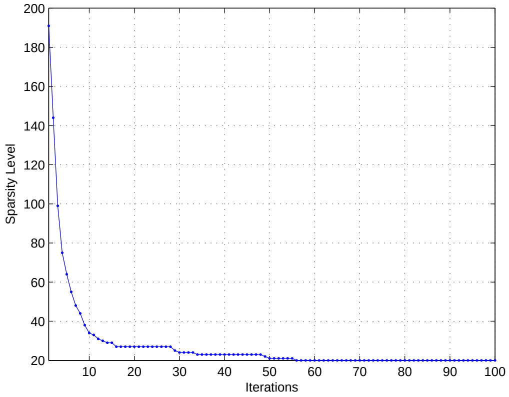

Figure 5. Number of Iterations vs. Sparsity Level

5.2. Comparison to ABCS

To explore the constraints on β in the CH framework, we compare SR-CH to ABCS, a SR method which puts no positive and normalization constraints on β. To fairly analyze the different β constraints in the SR-CH and ABCS methods, we compare both methods only using the exemplar terms, since the GMM-based terms for the two are different. Table 1 shows that SR-CH method offers improvements over ABCS on the fBMMI feature set, experimentally demonstrating that constraining β values to be positive and normalized, and not allowing data in H to be reflected and shifted, allows for improved classification accuracy.

5.2.1. GMM-Based Term

In this section, we analyze the behavior of using the exemplar-term only, given in Equation 4, versus

Table 1. Accuracy of SR Methods, TIMIT Dev. Set

Method	Accuracy
SR-CH (Exemplar-Only)	**83.86**
ABCS (Exemplar-Only)	78.16

including the additional model-based term given in Equation 5. Table 2 shows the classification accuracy on the development set with the fBMMI features. Notice that including the additional $H\beta$ GMM modeling term over the exemplar-based term offers a slight improvement in classification accuracy, demonstrating that including the GMM term allows for a slightly better classifier.

Table 2. SR-CH Accuracy, TIMIT Development Set

SR-CH GMM-Based Term	Accuracy
Exemplar Term Only	83.86
Exemplar Term+ $H\beta$ GMM Term	**84.00**

5.3. Comparison to Other Techniques

Table 3 compares the classification accuracy of the SR-CH method on the TIMIT core test set to other common classification methods. Note that for ABCS, the best numbers for this method, which include the exemplar and GMM-based terms, are reported. Results are provided for the fBMMI and SA+fBMMI feature sets. Notice that SR-CH outperforms the GMM, kNN and SVM classifiers. In addition, enforcing β to be positive allows for improvements over ABCS. A McNemar's Significance Test indicates that the SR-CH result is statistically significant from other classifiers with a 95% confidence level. Our classification accuracy of 82.87% is the best number on the TIMIT phone classification task reported when discriminative features are used, beating the previous best number of 82.3% reported in (Chang & Glass, 2007). Finally, when using SA+fBMMI features, the SR-CH method achieves an accuracy of over 85%.

Table 3. Classification Accuracy, TIMIT Core Test Set

Method	Accuracy fBMMI	Accuracy SA+fBMMI
SR-CH (Ex.+GMM)	**82.87**	**85.14**
ABCS (Ex.+GMM)	81.37	83.22
kNN	81.30	83.56
GMM - BMMI Trained	80.82	82.84
SVM	80.79	82.62
GMM - ML Trained	79.75	82.02

5.4. Accuracy vs. Size of Dictionary

One disadvantage of many exemplar-based methods is that as the number of training exemplars used to make a classification decision increases, the accuracy deteriorates significantly. For example, in the kNN method, this implies that the number of training examples from each class used during voting increases. Similarly, for SR methods, this is equivalent to the size of H growing. Parametric-based classification approaches such as GMMs do not suffer from a degradation in performance for increased training data size.

Figure 6 shows the classification error versus number of training-exemplars (i.e. size of H) for different classification methods. Note that the GMM method is trained with all of the training data, and is just shown here as a reference. In addition, since the feature vectors in H have dimension 120, and for our SR methods we assume H is over-complete, we only report results on SR methods when the number of examples in H is larger than 120.

First, observe that the error rates for the two purely exemplar-based methods, namely kNN and ABCS with no model term, increase exponentially as the size of H grows. However, the SR-CH exemplar-only methodology is much more robust with respect to increased size of H, demonstrating the value of the convex hull regularization constraints. Including the extra GMM term into the SR-CH method improves the accuracy slightly. However, the SR-CH method still performs poorly compared to the ABCS technique which uses the GMM-based term. One explanation for this behavior is that GMM term for ABCS is capturing the probability of the data y given the GMM model, and thus the accuracy of the ABCS method eventually approaches the GMM accuracy. However, in SR-CH we capture the probability of $H\beta$ given the GMM. This is one drawback of SR-CH compared to ABCS for large H that we hope to address in the future.

6. Conclusions

In this paper, we proposed a novel exemplar based technique for classification problems where for every new test sample the classification model is re-estimated from a subset of relevant samples of the training data. We formulated this exemplar based paradigm as a SR problem in a convex hull framework, and explored using EBW to solve the SR-CH problem. We showed that the proposed SR-CH method offers improvements in classification accuracy over common classification methods, including other SR methods which place fewer constraints on β.

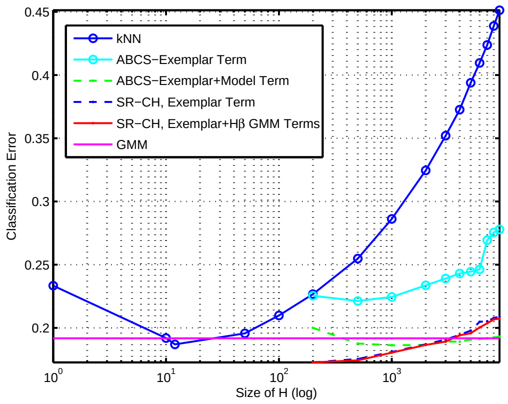

Figure 6. Classification Error vs. Size of H

Appendix A: EBW Update Solutions

In this section, we provide the detailed closed-form solution for β given the EBW solution in Equation 7 along with the exemplar-based term from Equation 4 and the model-based term from Equation 5. The objective function can be written more clearly in Equation 9. Here the unique GMM classes are denoted by t, C_t is the number of Gaussians belonging to class t, and w_{it}, μ_{it} and σ_{it} are the weight, mean and variance parameters for Gaussian i in GMM t.

$$F(\beta) = -1/2(y - H\beta)^T R^{-1}(y - H\beta) + \sum_{t=1}^{N_{classes}} \sum_{i=1}^{C_t} w_{it} \exp\left(-1/2 \times \frac{(\mu_{it} - H\beta)^T(\mu_{it} - H\beta)}{\sigma_{it}^2}\right) \tag{9}$$

Given this objective function, and the update for component β_j at iteration k in Equation 7, the gradient $\partial F(\beta^{k-1})/\partial\beta_j^{k-1}$ is given by Equation 10.

$$\partial F(\beta^{k-1})/\partial\beta_j^{k-1} = \frac{\partial(-0.5(y - H\beta^{k-1})^T R^{-1}(y - H\beta^{k-1}))}{\partial\beta_j^{k-1}} + \frac{\sum_{t=1}^{N_{classes}} \sum_i w_{it} \frac{\partial \exp\left(\frac{-0.5(\mu_{it}-H\beta)^T(\mu_{it}-H\beta)}{\sigma_{it}^2}\right)}{\partial\beta_j^{k-1}}}{\sum_{t=1}^{N_{classes}} \sum_i w_{it} \exp\left(\frac{-0.5(\mu_{it}-H\beta)^T(\mu_{it}-H\beta)}{\sigma_{it}^2}\right)} \tag{10}$$

The two components from Equation 10 are given as:

1.

$$\frac{\partial(-0.5(y - H\beta^{k-1})^T R^{-1}(y - H\beta^{k-1}))}{\partial\beta_j^{k-1}} = H_j^T R^{-1}(y - H\beta^{k-1}) \tag{11}$$

where H_j denotes the j-th column in matrix H.

2.

$$\frac{\partial \exp\left(\frac{-0.5(\mu_{it}-H\beta)^T(\mu_{it}-H\beta)}{\sigma_{it}^2}\right)}{\partial\beta_j^{k-1}} = \exp\left(\frac{-0.5(\mu_{it} - H\beta)^T(\mu_{it} - H\beta)}{\sigma_{it}^2}\right) \times \frac{H_j^T(\mu_{it} - H\beta^{k-1})}{\sigma_{it}^2} \tag{12}$$

References

Carmi, A., Gurfil, P., Kanevsky, D., and Ramabhadran, B. ABCS: Approximate Bayesian Compressed Sensing. Technical report, Human Language Technologies, IBM, 2009.

Chang, H. and Glass, J. Hierarchical Large-Margin Gaussian Mixture Models for Phonetic Classification. In *Proc. ASRU*, 2007.

Gemmeke, J. F. and Virtanen, T. Noise Robust Exemplar-Based Connected Digit Recognition. In *Proc. ICASSP*, 2010.

Gopalakrishnan, P.S., Kanevsky, D., Nahamoo, D., and Nadas, A. An Inequality for Rational Functions with Applications to Some Statistical Estimation Problems. *IEEE Trans. Information Theory*, 37(1), January 1991.

Hoyer, P.O. Non-negative Matrix Factorization with Sparseness Constraints. *JMLR*, 2004.

Kanevsky, D. Extended Baum-Welch Transformations for Genral Functions. In *Proc. ICASSP*, 2004.

Kanevsky, D., Sainath, T. N., Ramabhadran, B., and Nahamoo, D. An Analysis of Sparseness and Regularization in Exemplar-Based Methods for Speech Classification. In *Proc. Interspeech*, 2010.

Lamel, L., Kassel, R., and Seneff, S. Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus. In *Proc. of the DARPA Speech Recognition Workshop*, 1986.

Lee, K. F. and Hon, H. W. Speaker-independent Phone Recognition Using Hidden Markov Models. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37:1641–1648, 1989.

Povey, D., Kanevsky, D., Kingsbury, B., Ramabhadran, B., Saon, G., and Visweswariah, K. Boosted MMI for Model and Feature Space Discriminative Training. In *Proc. ICASSP*, 2008.

Sainath, T. N., Carmi, A., Kanevsky, D., and Ramabhadran, B. Bayesian Compressive Sensing for Phonetic Classification. In *Proc. ICASSP*, 2010.

Sainath, T.N., Ramabhadran, B., and Picheny, M. An Exploration of Large Vocabulary Tools for Small Vocabulary Phonetic Recognition. In *Proc. ASRU*, 2009.

Seeger, M. *Bayesian Gaussian Process Models: PAC-Bayesian Generalisation Error Bounds and Sparse Approximations*. PhD thesis, University of Edinburgh, 2003.

Wilson, D.R. and Martinez, T.R. Reduction Techniques for Exemplar-Based Learning Algorithms. *Machine Learning*, 38(3):257–286, 2000.

Wright, J., Yang, A.Y., Ganesh, A., Sastry, S. Shankar, and Ma, Y. Robust Face Recognition via Sparse Representation. *IEEE Trans. on PAMI*, 31:210–227, 2009.