Robust Feature Selection using Sparse Centroid-Encoder

Anonymous authors
Paper under double-blind review

Abstract

We develop a sparse optimization problem for the determination of the total set of features that discriminate two or more classes. This is a sparse implementation of the centroid-encoder for nonlinear data reduction and visualization called Sparse Centroid-Encoder (SCE). We also provide an iterative feature selection algorithm that first ranks each feature by its occurrence, and the optimal number of features is chosen using a validation set. The algorithm is applied to a wide variety of data sets including, single-cell biological data, high dimensional infectious disease data, hyperspectral data, image data, and GIS data. We compared our method to various state-of-the-art feature selection techniques, including three neural network-based models (DFS, SG-L1-NN, G-L1-NN), Sparse SVM, and Random Forest. We empirically showed that SCE features produced better classification accuracy on the unseen test data, often with fewer features.

1 Introduction

Technological advancement has made high-dimensional data readily available. For example, in bioinformatics, the researchers seek to understand the gene expression level with microarray or next-generation sequencing techniques where each point consists of over 50,000 measurements (Pease et al. (1994); Shalon et al. (1996); Metzker (2010); Reuter et al. (2015)). The abundance of features demands the development of feature selection algorithms to improve a Machine Learning task, e.g., classification. Another important aspect of feature selection is knowledge discovery from data. Which biomarkers are important to characterize a biological process, e.g., the immune response to infection by respiratory viruses such as influenza (O'Hara et al. (2013))? Additional benefits of feature selection include improved visualization and understanding of data, reducing storage requirements, and faster algorithm training times.

Feature selection can be accomplished in various ways that can be broadly categorized into the filter, wrapper, and embedded methods. In a filter method, each variable is ordered based on a score. After that, a threshold is used to select the relevant features (Lazar et al. (2012)). Variables are usually ranked using correlation (Guyon & Elisseeff (2003); Yu & Liu (2003)), and mutual information (Vergara & Estévez (2014); Fleuret (2004)). In contrast, a wrapper method uses a model and determines the importance of a feature or a group of features by the generalization performance of the predetermined model (El Aboudi & Benhlima (2016); Hsu et al. (2002)). Since evaluating every possible combination of features becomes an NP-hard problem, heuristics are used to find a subset of features. Wrapper methods are computationally intensive for larger data sets, in which case search techniques like Genetic Algorithm (GA) (Goldberg & Holland (1988)) or Particle Swarm Optimization (PSO) (Kennedy & Eberhart (1995)) are used. In embedded methods, feature selection criteria are incorporated within the model, i.e., the variables are picked during the training process (Lal et al. (2006)). Iterative Feature Removal (IFR) uses the absolute weight of a Sparse SVM model as a criterion to extract features from the high dimensional biological data set (O'Hara et al. (2013)).

This paper proposes a new embedded variable selection approach called Sparse Centroid-Encoder (SCE) to extract features when class labels are available. Our method extends the Centroid-Encoder model (Ghosh et al. (2018); Ghosh & Kirby (2020)), where we applied a l_1 penalty to a sparsity promoting layer between the input and the first hidden layer. We evaluate this Sparse Centroid-Encoder on diverse data sets and show that the selected features produce better generalization than other state-of-the-art techniques. Our results showed that SCE picked fewer features to obtain high classification accuracy. As a feature selection tool, SCE uses a single model for the multi-class problem without the need to create multiple one-against-one binary models typical of linear methods, e.g., Lasso (Tibshirani (1996)), or Sparse SVM (Chepushtanova et al. (2014)). Although SCE can be used both in binary and multi-class problems, we focused on the multi-class feature selection problem in this paper. The work of Li et al. (2016) also uses a similar sparse layer between the input and the first hidden with an Elastic net penalty while minimizing the classification error with a softmax layer. The authors used Theano's symbolic differentiation (Bergstra et al. (2010)) to impose sparsity. In contrast, our approach minimizes the centroid-encoder loss with an explicit differentiation of the l_1 function using the sub-gradient.

The article is organized as follows: In Section 2 we present the Sparse Centroid-Encoder algorithm. In Section 3 we apply SCE to a range of bench-marking data sets taken from the literature. In Section 4, we review related work, for both linear and non-linear feature selection. In Section 5, we present our discussion and conclusion.

2 SPARSE CENTROID-ENCODER

Centroid-encoder (CE) neural networks are the starting point of our approach (Ghosh & Kirby (2020); Ghosh et al. (2018); Aminian et al. (2021)). We present a brief overview of CEs and demonstrate how they can be extended to perform non-linear feature selection.

2.1 CENTROID-ENCODER

The CE neural network is a variation of an autoencoder and can be used for both visualization and classification tasks. Consider a data set with N samples and M classes. The classes denoted $C_j, j = 1, \dots, M$ where the indices of the data associated with class C_j are denoted I_j. We define centroid of each class as $c_j = \frac{1}{|C_j|}\sum_{i \in I_j} x^i$ where $|C_j|$ is the cardinality of class C_j. Unlike autoencoder, which maps each point x^i to itself, the CE maps each point x^i to its class centroid c_j by minimizing the following cost function over the parameter set θ:

$$\mathcal{L}_{ce}(\theta) = \frac{1}{2N}\sum_{j=1}^{M}\sum_{i \in I_j} \|c_j - f(x^i;\theta))\|_2^2 \tag{1}$$

The mapping f is composed of a dimension reducing mapping g (encoder) followed by a dimension increasing reconstruction mapping h (decoder). The output of the encoder is used as a supervised visualization tool (Ghosh & Kirby (2020); Ghosh et al. (2018)), and attaching another layer to map to the one-hot encoded labels performs robust classification (Aminian et al. (2021)).

2.2 SPARSE CENTROID-ENCODER FOR FEATURE SELECTION

The Sparse Centroid-encoder (SCE) is a modification to the centroid-encoder architecture as shown in Figure 1. Unlike centroid-encoder, we haven't used a bottleneck architecture as visualization is not our aim here. The input layer is connected to the first hidden layer via the sparsity promoting layer (SPL). Each node of the input layer has a weighted one-to-one connection to each node of the SPL. The number of nodes in these two layer are the same. The nodes in SPL don't have any bias or non-linearity. The SPL is fully connected to the first hidden layer, therefore the weighted input from the SPL will be passed to the hidden layer in the same way that of a standard feed forward network. During training, a l_1 penalty will be applied to the weights connecting the input layer and SPL layer. The sparsity promoting l_1 penalty will drive most of the weights to near zero and the corresponding input nodes/features can be discarded. Therefore, the purpose of the SPL is to select important features from the original input. Note we only apply the l_1 penalty to the parameters of the SPL.

Denote θ_{spl} to be the parameters (weights) of the SPL and θ to be the parameters of the rest of the network. The cost function of sparse centroid-encoder is given by

$$\mathcal{L}_{sce}(\theta) = \frac{1}{2N}\sum_{j=1}^{M}\sum_{i \in I_j} \|c_j - f(x^i;\theta))\|_2^2 + \lambda\|\theta_{spl}\|_1 \tag{2}$$

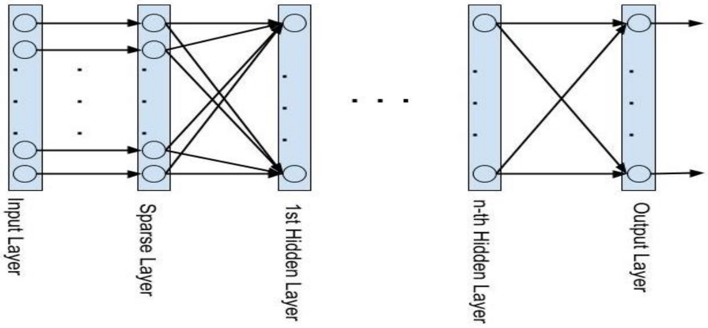


Figure 1: The architecture of sparse centroid-encoder. This is similar to that of a centroid-encoder except the sparse layer in-between the input layer and first hidden layer. Unlike centroid-encoder, the SCE doesn't use a bottleneck architecture.

where λ is the hyper-parameter which controls the sparsity. A larger value of λ will promote higher sparsity resulting more near-zero weights in SPL. In other words, λ is a knob that controls the number of features selected from the input data.

Like centroid-encoder, we trained sparse centroid-encoder using error backpropagation, which requires the gradient of the cost function of Equation 2. As l_1 function is not differentiable at 0, we implement this term using the sub-gradient.

2.3 Iterative Feature Selection Using Sparse Centroid-encoder

By design, sparse methods identify a small number of features that accomplish a classification task. If one is interested in *all* the discriminatory features that can be used to separate multiple classes, then one can repeat the process of removing good features. This section describes how sparse centroid-encoder (SCE) can be used iteratively to extract all discriminatory features from a data set; see O'Hara et al. (2013) for an application of this approach to sparse support vector machines.

SCE is a model based on neural network architecture; hence, it's a non-convex optimization. As a result, multiple runs will produce different solutions, i.e., different feature sets on the same training set. These features may not be optimal given an unseen test set. To find out the robust features from a training set, we resort to frequency-based feature pruning. In this strategy, first, we divide the entire training set into k folds. On each of these folds, we ran the SCE and picked the top N (user select) number of features. We repeat the process T times to get $k \times T$ feature sets. Then we count the number of occurrences of each feature and call this number the frequency of a feature. We ordered the features based on the frequency and picked the optimum number from a validation set. We present the feature selection steps in Algorithm 1. In Figure 2, we plotted the magnitude of the feature weights for MNIST and GSE73072 in descending order to show the ability to promote sparsity of SCE. In both cases, the model ignored many features by setting their weight to near zero.

3 Experimental Results

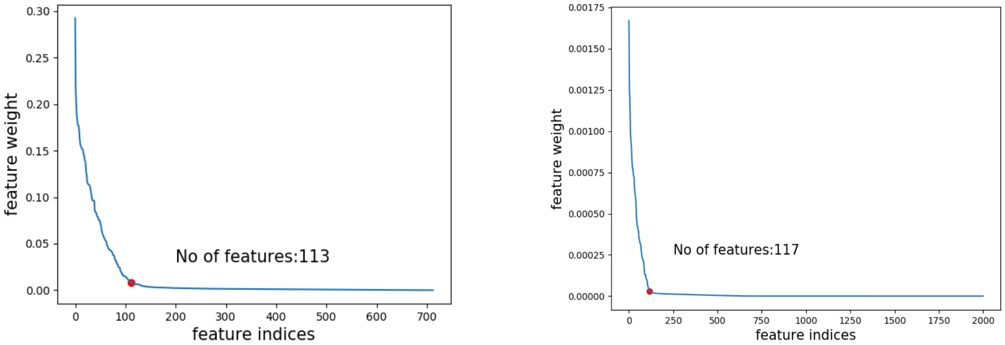
Here we present a range of comparative benchmarking experiments on a variety of data sets and feature selection models. We used five disparate data sets including single-cell data (GM12878), high dimensional infectious disease data set (GSE73072), vision letter data (MNIST), hyperspectral imagery (Indian Pines), and GIS data (Forest Cover).

3.1 Experimental Details

We did bench-marking experiments to compare the sparse centroid-encoder with other state-of-the-art feature selection methods. To make the evaluation objective, we compared the classification accuracy on the unseen data using the selected features of different models. All experiments share the following workflow:

Algorithm 1: Iterative Feature Selection using Sparse Centroid-Encoder.

```
Input: Labeled data (D) = $\{x^i\}_{i=1}^N$, $x^i \in \mathcal{R}^p$. User defined parameters: penalty term $\lambda$, no. of
       partitions $k$ of training set, max. no. of features from each training fold $N$, repetition $T$.
Initialization: Partition D into training (Tr) and validation set (V). Feature set $F := \emptyset$
Output: $f$ discriminatory features from $D$ where $f < p$.
Step 1: Partition Tr into stratified $k-$fold $Tr_1, Tr_2, ..., Tr_k$.
Step 2: For each $Tr_i$ do
Step 3:     Initialize a temporary list $F_{temp} := \emptyset$
Step 4:     Run SCE to get $W_{SPL}$ := SCE($Tr_i, \lambda$)
Step 5:     Sort $W_{SPL}$ on the magnitude of each $w_i$ in descending order
Step 6:     Pick the top $n$ features corresponding to the largest $w_i$ from ordered $W_{SPL}$ using
        the Elbow Method
Step 7:     Insert the $n$ features in $F_{temp}$
Step 8:     If cardinality of $F_{temp} >$ N
Step 9:         Insert the features in $F_{temp}$ to $F$. Go to Step 2
Step 10:    else
Step 11:        Remove the $n$ features from $Tr_i$ and go to Step 4
Step 12: Repeat Step 1 to 11 for $T$ times.
Step 13: Calculate the no. of occurrences ($r$) of each feature in $F$. Rank the features in
         descending order of $r$.
Step 14: Using a classifier pick the top $f$ features from ordered set $F$ which gave the highest
         accuracy on the validation set.
```



(a) Sparsity Plot for MNIST.

(b) Sparsity Plot of GSE73072

Figure 2: Sparsity plot of the weight of W_{SPL} shown for MNIST (a) and GSE73072 (b). The l_1 penalty sets the weight of most of the features to near zero and those features were ignored. The Elbow method picked 113 and 117 features from the two data sets. The red dot indicates the location of the elbow.

- SCE is used to select an optimal number of features on the training samples. The l_1 penalty parameter $\lambda = 0.01$ for all experiments save for MNIST where $\lambda = 0.001$.
- Build K classification models with these features on the training set. We used centroid-encoder as the classification model (Aminian et al. (2021)).
- Compute the accuracy on the sequestered test set using the K trained models and report the mean accuracy with standard deviation.

3.2 Quantitative and Qualitative Analysis

Now we present the results from a comprehensive analysis across five data sets.

3.2.1 Single Cell Data

GM12878 is a single cell data set that has been previously used to test multiclass feature selection algorithms (Li et al. (2016)). The samples were collected from the annotated DNA region of lymphoblastoid cell line. Each sample is represented by a 93 dimensional features sampled from three classes: active enhancer (AE), active promoter (AP) and background (BG) where each class contains $2,156$ number of samples. The data set is split equally into a separate training, validation and test sets. We used the validation set to tune hyper-parameters and to pick the optimal number of features. After the feature selection step, we merged the training and validation set and trained $K = 10$ centroid-encoder classifiers with the selected features, and reported classification accuracy on the test set.

We use the published results for deep feature selection (DFS), shallow feature selection, LASSO, and Random Forest (RF) from the work of Li et al. to evaluate SCE as shown in Table 1. To compare with Li et al., we used the top 16 features to report the mean accuracy of the test samples. We also report the test accuracy using the top 48 features picked from the validation set, this was the best result in our experiment. When restricted to the top 16, we see that the SCE features still outperform all the other models.

Feature Selection Method	No. of Features	Accuracy
Sparse Centroid-encoder	**48**	**89.40 ± 0.24**
Sparse Centroid-encoder	**16**	**88.33 ± 0.13**
Deep DFS	16	85.67
Shallow DFS	16	85.34
LASSO	16	81.86
Random Forest	16	88.21

Table 1: Classification accuracies using the top 16 features by various techniques. Results of Deep DFS, Shallow DFS, LASSO, and Random Forest are reported from Li et al. (2016). We present accuracy with the top 48 features which were selected from a validation set.

Among all the models, LASSO features exhibit the worst performance with an accuracy of 81.86%. This relatively low accuracy is not surprising, given LASSO is a linear model.

The classification performance gives a quantitative measure that doesn't reveal the biological significance of the selected genes. We did a literature survey of the top genes selected by sparse centroid-encoder and provided the detailed description in the appendix. Some of these genes play an essential role in transcriptional activation, e.g., H4K20ME1 (Barski et al. (2007)), TAF1 (Wang et al. (2014)), H3K27ME3 (Cai et al. (2021)), etc. Gene H3K27AC (Creyghton et al. (2010)) plays a vital role in separating active enhances from inactive ones. Besides that, many of these genes are related to the proliferation of the lymphoblastoid cancer cells, e.g., POL2 (Yamada et al. (2013)), NRSF/REST (Kreisler et al. (2010)), GCN5 (Yin et al. (2015)), PML (Salomoni & Pandolfi (2002)), etc. This survey confirms the biological significance of the selected genes.

3.2.2 MNIST and Forest Cover

In this section we compare SCE with the feature selection results presented in Scardapane et al. (2017) on MNIST and Forest Cover data. **Forest Cover** is a data set of seven forest cover types (e.g. ponderosa pine) where each sample is represented by a vector of 54 elements which were extracted from cartographic data. There are about half million samples. The data is available in UCI repository. **MNIST** This is a widely used collection of digital images of handwritten digits (0..9)[1] with separate training (60,000 samples) and test set (10,000 samples). Each sample is a grey level image consisting of 1-byte pixels normalized to fit into a 28 x 28 bounding box resulting in *vecced* points in $\mathbb{R}^{784}$. Following Scardapane et al. (2017), each data set was randomly partitioned into a training and test set with a ratio of 75:25. We used 20% of the training sample as a validation set to select the optimum number of features and hyper-parameters. After the feature selection using SCE,

[1] The data set is available at `http://yann.lecun.com/exdb/mnist/index.html`.

Feature Selection Method	No. of Average Features	Accuracy
Sparse Centroid-encoder	**355.32**	**98.44 ± 0.08**
SG-L1-NN	581.8	97.00
L1-NN	658.2	97.00
L2-NN	676.4	98.00

Table 2: Classification result using the top features by various models on the MNIST data set. Results of SG-L1-NN, L1-NN, and L2-NN are reported from Scardapane et al. (2017).

a CE is trained to predict the class label of the unseen test cases. We repeated the process $K = 25$ times and reported the mean accuracy with standard deviation.

Table 2 shows the classification accuracy using the features selected by four methods. The features of SCE produce the best accuracy beating the other sparse models SG-L1-NN and L1-NN by a margin of 1.44%. More significantly, our classification accuracy is achieved using 355 number of features on average, which is approximately 45% of the total number of pixels of an MNIST image. On the other hand, SG-L1-NN, L1-NN, and L2-NN used 74%, 84%, and 86% of the total number of features, respectively. We further analyze the features from a qualitative point of view. We present a visual representation of the selected variables of the SCE model in Figure 3 where we show the spatial location of the selected pixels in a 28 x 28 grid. Observe that most of the selected pixels are located in the middle of the grid, making sense as most MNIST digits are placed in the middle of the 28 x 28 bounding box. This fact establishes the robustness of the features of the Sparse Centroid-encoder.

The classification accuracy of the four models on the Forest Cover data is presented in Table 3. The average test accuracy is better using the SCE features. At the same time, the number of SCE features is considerably less than the other sparse models of Scardapane et al. (2017). Note that the mean accuracy of our model is higher by a margin of $3\% - 4\%$. It's noteworthy that the models of Scardapane et al. required most of the features for classification. The data set lives in $\mathbb{R}^{54}$ in the ambient space. Out of these 54 variables, SG-L1-NN, L1-NN, and L2-NN utilized $52.7, 53$, and 54 features, respectively. In contrast, our model only used 38 features on average. The high test accuracy with a relatively small feature set establishes the value of SCE as a robust variable selection technique.

3.2.3 Indian Pines Hyperspectral Imagery

Here we compare SCE with sparse support vector machines on the well-known Indian Pines hyperspectral imagery of a variety of crops following the experiments in (Chepushtanova et al. (2014)). The task is to identify the frequency bands which are essential to assign a test sample correctly to one of the sixteen classes. We took all the 220 bands in the feature selection step, including the twenty water absorption bands. Note that in literature these noisy water absorption bands are often excluded before the experiments (Reshma et al. (2016); Cao et al. (2017)). We wanted to check

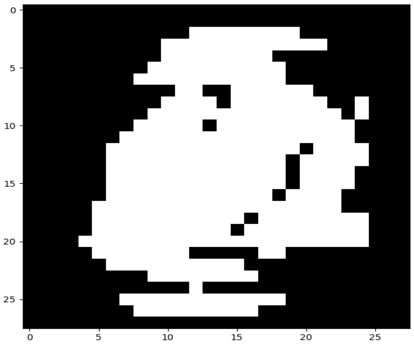


Figure 3: Locations of selected features of MNIST image shown in a 28 x 28 grid. The selected pixels are marked in white, and the ignored pixels are marked in black.

Feature Selection Method	No. of Average Features	Accuracy
Sparse Centroid-encoder	**38.1**	**87.37 ± 0.36**
SG-L1-NN	52.7	83.00
L1-NN	53.0	83.00
L2-NN	54.0	84.00

Table 3: Classification results using the top features by various models on the Forest Cover data set. Results of SG-L1-NN, L1-NN, and L2-NN are reported from Scardapane et al. (2017).

whether our model was able to reject them. In fact, our model did discard them as no water absorption bands were in the top 100 features. It appeared that SSVM included some of these noisy bands as described in(Chepushtanova et al. (2014)).

We followed the experimental protocol in (Chepushtanova et al. (2014)). The entire data set is split in half into a training and test set. Because of the small size of the training set, we did a 5-fold cross-validation on the training samples to tune hyper-parameters. After the feature selection on the training set, we took top $n = 1, 2, 3, 4, 5, 10, 20, 40, 60, 80$ features to build a CE classifier on the training set to predict the class labels of the test samples. For each n we repeat the classification task $K = 10$ times. Note that we compared the performance of SCE and SSVM features without spatial smoothing for a more direct comparison of the classification rates. Figure 4 presents the accuracy on the test data using the top n bands ($n = 1, 2, 3, 4, 5, 10, 20, 40, 60, 80$) which were calculated on the training set. Classification using SCE features generally produces better accuracy. Notice that SCE features yield better classification performance using fewer bands. In particular, the accuracy of the top SCE feature (band 13) is at least 15% higher than the top SSVM feature (band 1). See the Appendix for additional details on the feature sets selected.

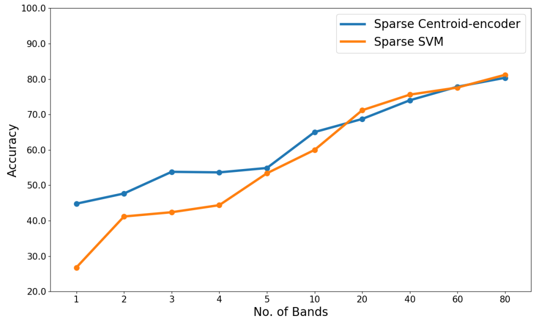


Figure 4: Comparison of classification accuracy using SCE and SSVM features.

3.2.4 Respiratory Infections in Humans

GSE73072 This microarray data set is a collection of gene expressions taken from human blood samples as part of multiple clinical challenge studies (Liu et al. (2016)) where individuals were infected with the following respiratory viruses HRV, RSV, H1N1, and H3N2. In our experiment we excluded the RSV study. Blood samples were taken from the individuals before and after the inoculation. RMA normalization (Irizarry et al. (2003)) is applied to the entire data set, and the LIMMA (Ritchie et al. (2015)) is used to remove the subject-specific batch effect. Each sample is represented by 22,277 probes associated with gene expression. The data is publically available on the NCBI GeneExpression Omnibus (GEO) with identifier GSE73072.

We conducted our last experiment on the GSE73072 human respiratory infection data where the goal is to predict the classes control, shedders, and non-shedders at the very early phase of the infection, i.e., at time bin spanning hours 1-8. Controls are the pre-infection samples, whereas shedders and non-shedders are post-infection samples picked from the time bin 1-8 hr. Shedders actually disseminate virus while non-shedders do not. We considered six studies, including two H1N1 (DEE3, DEE4), two H3N2 (DEE2, DEE5), and two HRV (Duke, UVA) studies. We used 10% training samples as a validation set—the training set comprised all the studies except for the

DEE5, which was kept out for testing. We did a leave-one-subject-out (LOSO) cross-validation on the test set using the selected features from the training set. In this experiment we compared SCE with Random Forest (RF).

The results on this data set in shown in Table 4. The top 35 features of SCE produce the best Balanced Success Rate (BSR) of 90.61% on the test study DEE5. For the Random Forest model, the best result is achieved with 30 features. We also included the results with 35 biomarkers, but the BSR didn't improve. Note both the models picked a relatively small number of features, 30 and 35 out of the 22,277 genes, but SCE features outperform RF by a margin of 7%. Although RF selects features with multiple classes using a single model, it weighs a single feature by measuring the decrease of out-of-bag error. In contrast, SCE looks for a group of features while minimizing its cost. We think the multivariate approach of SCE makes it a better features detector than RF.

Time Bin	Model	No. of Features	BSR
$1-8$	SCE	35	90.61 ± 2.38
	RF	30	83.05 ± 2.42
	RF	35	82.65 ± 2.51

Table 4: Balanced success rate (BSR) of LOSO cross-validation on the DEE5 test set. The selected features from training set is used to predict the classes of control, shedder, and non-shedder.

4 Related Work

Feature selection has a long history spread across many fields, including bioinformatics, document classification, data mining, hyperspectral band selection, computer vision, etc. It's an active research area, and numerous techniques exist to accomplish the task. We describe the literature related to the embedded methods where the selection criteria are part of a model. The model can be either linear or non-linear.

4.1 Feature Selection using Linear Models

Adding an l_1 penalty to classification and regression methods naturally produce feature selectors. For example, least absolute shrinkage and selection operator or Lasso (Tibshirani (1996)) has been used extensively for feature selection on various data sets (Fonti & Belitser (2017); Muthukrishnan & Rohini (2016); Kim & Kim (2004)). Elastic net, proposed by Zou et al. (Zou & Hastie (2005)), combined the Lasso penalty with the Ridge Regression penalty (Hoerl & Kennard (1970)) to overcome some limitations of Lasso. Elastic net has been widely applied, e.g., (Marafino et al. (2015); Shen et al. (2011); Sokolov et al. (2016)). Note both Lasso and Elastic net are convex in the parameter space. Support Vector Machines (SVM) (Cortes & Vapnik (1995)) is a state-of-the-art model for classification, regression and feature selection. SVM-RFE is a linear feature selection model which iteratively removes the least discriminative features until a parsimonious set of predictive features are selected (Guyon et al. (2002)). IFR (O'Hara et al. (2013)), on the other hand, selects a group of discriminatory features at each iteration and eliminates them from the data set. The process repeats until the accuracy of the model starts to drop significantly. Note IFR uses Sparse SVM (SSVM), which minimizes the l_1 norm of the model parameters. Lasso, Elastic Net, and SVM-based techniques are mainly applied to binary problems. These models are extended to the multi-class problem by combining multiple binary one-against-one (OAO) or one-against-all (OAA) models. Chepushtanova et al. (2014) used 120 Sparse SVM models to select discriminative bands from the Indian Pine data set, which has 16 classes. On the other hand, Random forest Breiman (2001), a decision tree-based technique, finds features from multi-class data using a single model. The model doesn't use Lasso or Elastic net penalty for feature selection. Instead, the model weighs the importance of each feature by measuring the out-of-bag error.

4.2 Feature Selection using Deep Neural Networks

While the linear models are fast and convex, they don't capture the non-linear relationship among the input features (unless a kernel trick is applied). Because of the shallow architecture, these models don't learn a high-level representation of input features. Moreover, there is no natural way to incorporate multi-class data in a single model. Non-linear models based on deep neural networks overcome these limitations. In this section, we will briefly discuss a handful of such models.

Scardapane et al. (2017) used group Lasso (Tibshirani (1996)) to impose the sparsity on a group of variables instead of a single variable. They applied the group sparsity simultaneously on the input and the hidden layers to remove features from the input data and the hidden activation. On MNIST, their algorithm discarded more than 200 features from the input vector with an accuracy of 97% on the test data. Although on the Forest Cover data set, the algorithm used most of the input variables 52.7 out of 54. Li et al. proposed deep feature selection (DFS), which is a multilayer neural network-based feature selection technique (Li et al. (2016)). DFS uses a one-to-one linear layer between the input and the first hidden layer. As a sparse regularization, the authors used elastic-net (Zou & Hastie (2005)) on the variables of the one-to-one layer to induce sparsity. The standard soft-max function is used in the output layer for classification. With this setup, the network is trained in an end-to-end fashion by error backpropagation. Despite the deep architecture, its accuracy is not competitive, and experimental results have shown that the method did not outperform the random forest (RF) method. Kim et al. (2016) proposed a heuristics based technique to assign importance to each feature. Using the ReLU activation, Roy et al. (2015) provided a way to measure the contribution of an input feature towards hidden activation of next layer. Han et al. (2018) developed an unsupervised feature selection technique based on the autoencoder architecture. Using a $l_{2,1}$-norm to the weights emanating from each input node, they measure the contribution of each feature while reconstructing the input. The model excavates the input features, which have a minimum contribution. Taherkhani et al. (2018) proposed a RBM (Hinton et al. (2006); Hinton & Salakhutdinov (2006)) based feature selection model. This algorithm runs the risk of combinatorial explosion for data set with $50K-60K$ features (e.g., microarray gene expression data set).

5 Discussion and Conclusion

In this paper, we presented Sparse Centroid-encoder as an effective feature selection tool for multi-class classification problems. The benchmarking results span 5 data sets and 7 methods providing evidence that the features of SCE produce better generalization performance compared to other models on a diverse set of data sets. We compared SCE with deep feature selection (deep DFS), a deep neural network-based model, and found that the features of SCE significantly improved the classification rate on test data. The survey of the known functionality of these genes indicates the plausible biological significance. We have also demonstrated that our feature selection algorithm often selected fewer features than other models—the MNIST, Forest Cover experiments establish this fact. These experiments include the comparison to some neural network-based models where group-sparsity is applied for variable selection. In addition to extracting the most robust features, the model shows the ability to discard noisy features. On the Indian Pine data set, our model didn't pick any of the water absorption bands considered noisy.

Our model has the advantage over the class of linear techniques, where a binary feature selection method is used as a multi-class method by one-against-one(OAO) or one-against-all(OAA) class pairs. For example, Chepushtanova et al. used 120 binary class SSVM models on the Indian Pine data set. Similarly, Lasso needed three models for the GM12878 data. These models will suffer a combinatorial explosion when the number of classes increases. In contrast, SCE uses a single model to extract features from a multi-class data set.

SCE has a potential advantage over the neural network-based models (deep DFS, shallow DFS, SG-L1-NN, G-L1-NN), where a sample is mapped to its corresponding class label. Mapping samples to their class label ignores the intra-class variation. Sparse Centroid-encoder can be modified to handle multi-modality by assigning multiple centers per class. In the future, we will extend our model to investigate this aspect further.

6 Reproducibility Statement

We have uploaded our code package in a separate file as supplementary material. Please read the ”README.txt” before running the code.

REFERENCES

M Aminian, T Ghosh, A Peterson, AL Rasmussen, S Stiverson, K Sharma, and M Kirby. Early prognosis of respiratory virus shedding in humans. *Scientific reports*, 11(1):1–15, 2021.

Artem Barski, Suresh Cuddapah, Kairong Cui, Tae-Young Roh, Dustin E Schones, Zhibin Wang, Gang Wei, Iouri Chepelev, and Keji Zhao. High-resolution profiling of histone methylations in the human genome. *Cell*, 129(4):823–837, 2007.

James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. Theano: a cpu and gpu math expression compiler. In *Proceedings of the Python for scientific computing conference (SciPy)*, volume 4, pp. 1–7. Austin, TX, 2010.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Yichao Cai, Ying Zhang, Yan Ping Loh, Jia Qi Tng, Mei Chee Lim, Zhendong Cao, Anandhkumar Raju, Erez Lieberman Aiden, Shang Li, Lakshmanan Manikandan, et al. H3k27me3-rich genomic regions can function as silencers to repress gene expression via chromatin interactions. *Nature communications*, 12(1):1–22, 2021.

Xianghai Cao, Cuicui Wei, Jungong Han, and Licheng Jiao. Hyperspectral band selection using improved classification map. *IEEE geoscience and remote sensing letters*, 14(11):2147–2151, 2017.

Sofya Chepushtanova, Christopher Gittins, and Michael Kirby. Band selection in hyperspectral imagery using sparse support vector machines. In *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XX*, volume 9088, pp. 90881F. International Society for Optics and Photonics, 2014.

Corinna Cortes and Vladimir Vapnik. Support vector machine. *Machine learning*, 20(3):273–297, 1995.

Menno P Creyghton, Albert W Cheng, G Grant Welstead, Tristan Kooistra, Bryce W Carey, Eveline J Steine, Jacob Hanna, Michael A Lodato, Garrett M Frampton, Phillip A Sharp, et al. Histone h3k27ac separates active from poised enhancers and predicts developmental state. *Proceedings of the National Academy of Sciences*, 107(50):21931–21936, 2010.

Naoual El Aboudi and Laila Benhlima. Review on wrapper feature selection approaches. In *2016 International Conference on Engineering & MIS (ICEMIS)*, pp. 1–5. IEEE, 2016.

François Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine learning research*, 5(9), 2004.

Valeria Fonti and Eduard Belitser. Feature selection using lasso. *VU Amsterdam Research Paper in Business Analytics*, 30:1–25, 2017.

Tomojit Ghosh and Michael Kirby. Supervised dimensionality reduction and visualization using centroid-encoder, 2020.

Tomojit Ghosh, Xiaofeng Ma, and Michael Kirby. New tools for the visualization of biological pathways. *Methods*, 132:26 – 33, 2018. ISSN 1046-2023. doi: https://doi.org/10.1016/j.ymeth.2017.09.006. URL http://www.sciencedirect.com/science/article/pii/S1046202317300439. Comparison and Visualization Methods for High-Dimensional Biological Data.

David E Goldberg and John Henry Holland. Genetic algorithms and machine learning. 1988.

Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422, 2002.

Kai Han, Yunhe Wang, Chao Zhang, Chao Li, and Chao Xu. Autoencoder inspired unsupervised feature selection. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2941–2945. IEEE, 2018.

Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural Comput.*, 18(7):1527–1554, July 2006. ISSN 0899-7667. doi: 10.1162/neco.2006.18.7.1527. URL `http://dx.doi.org/10.1162/neco.2006.18.7.1527`.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Chun-Nan Hsu, Hung-Ju Huang, and Stefan Dietrich. The annigma-wrapper approach to fast feature selection for neural nets. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 32(2):207–212, 2002.

Rafael A Irizarry, Bridget Hobbs, Francois Collin, Yasmin D Beazer-Barclay, Kristen J Antonellis, Uwe Scherf, and Terence P Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2):249–264, 2003.

James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pp. 1942–1948. IEEE, 1995.

Seong Gon Kim, Nawanol Theera-Ampornpunt, Chih-Hao Fang, Mrudul Harwani, Ananth Grama, and Somali Chaterji. Opening up the blackbox: an interpretable deep neural network-based classifier for cell-type specific enhancer predictions. *BMC systems biology*, 10(2):243–258, 2016.

Yongdai Kim and Jinseog Kim. Gradient lasso for feature selection. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 60, 2004.

A Kreisler, PL Strissel, R Strick, SB Neumann, U Schumacher, and CM Becker. Regulation of the nrsf/rest gene by methylation and creb affects the cellular phenotype of small-cell lung cancer. *Oncogene*, 29(43):5828–5838, 2010.

Thomas Navin Lal, Olivier Chapelle, Jason Weston, and André Elisseeff. Embedded methods. In *Feature extraction*, pp. 137–165. Springer, 2006.

Cosmin Lazar, Jonatan Taminau, Stijn Meganck, David Steenhoff, Alain Coletta, Colin Molter, Virginie de Schaetzen, Robin Duque, Hugues Bersini, and Ann Nowe. A survey on filter techniques for feature selection in gene expression microarray analysis. *IEEE/ACM transactions on computational biology and bioinformatics*, 9(4):1106–1119, 2012.

Yifeng Li, Chih-Yu Chen, and Wyeth W Wasserman. Deep feature selection: theory and application to identify enhancers and promoters. *Journal of Computational Biology*, 23(5):322–336, 2016.

Tzu-Yu Liu, Thomas Burke, Lawrence P Park, Christopher W Woods, Aimee K Zaas, Geoffrey S Ginsburg, and Alfred O Hero. An individualized predictor of health and disease using paired reference and target samples. *BMC bioinformatics*, 17(1):1–15, 2016.

Ben J Marafino, W John Boscardin, and R Adams Dudley. Efficient and sparse feature selection for biomedical text classification via the elastic net: Application to icu risk stratification from nursing notes. *Journal of biomedical informatics*, 54:114–120, 2015.

Michael L Metzker. Sequencing technologies—the next generation. *Nature reviews genetics*, 11(1):31, 2010.

Wenyi Mi, Yi Zhang, Jie Lyu, Xiaolu Wang, Qiong Tong, Danni Peng, Yongming Xue, Adam H Tencer, Hong Wen, Wei Li, et al. The zz-type zinc finger of zzz3 modulates the atac complex-mediated histone acetylation and gene activation. *Nature communications*, 9(1):1–9, 2018.

R Muthukrishnan and R Rohini. Lasso: A feature selection technique in predictive modeling for machine learning. In *2016 IEEE international conference on advances in computer applications (ICACA)*, pp. 18–20. IEEE, 2016.

Stephen O'Hara, Kun Wang, Richard A Slayden, Alan R Schenkel, Greg Huber, Corey S O'Hern, Mark D Shattuck, and Michael Kirby. Iterative feature removal yields highly discriminative pathways. *BMC genomics*, 14(1):1–15, 2013.

A C Pease, D Solas, E J Sullivan, M T Cronin, C P Holmes, and S P Fodor. Light-generated oligonucleotide arrays for rapid dna sequence analysis. *Proceedings of the National Academy of Sciences*, 91(11):5022–5026, 1994. ISSN 0027-8424. doi: 10.1073/pnas.91.11.5022. URL `https://www.pnas.org/content/91/11/5022`.

R Reshma, V Sowmya, and KP Soman. Dimensionality reduction using band selection technique for kernel based hyperspectral image classification. *Procedia Computer Science*, 93:396–402, 2016.

Jason A Reuter, Damek V Spacek, and Michael P Snyder. High-throughput sequencing technologies. *Molecular cell*, 58(4):586–597, 2015.

Matthew E Ritchie, Belinda Phipson, DI Wu, Yifang Hu, Charity W Law, Wei Shi, and Gordon K Smyth. limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic acids research*, 43(7):e47–e47, 2015.

Debaditya Roy, K Sri Rama Murty, and C Krishna Mohan. Feature selection using deep neural networks. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6. IEEE, 2015.

Paolo Salomoni and Pier Paolo Pandolfi. The role of pml in tumor suppression. *Cell*, 108(2): 165–170, 2002.

Simone Scardapane, Danilo Comminiello, Amir Hussain, and Aurelio Uncini. Group sparse regularization for deep neural networks. *Neurocomputing*, 241:81–89, 2017.

Dari Shalon, Stephen J Smith, and Patrick O Brown. A dna microarray system for analyzing complex dna samples using two-color fluorescent probe hybridization. *Genome research*, 6(7):639–645, 1996.

Li Shen, Sungeun Kim, Yuan Qi, Mark Inlow, Shanker Swaminathan, Kwangsik Nho, Jing Wan, Shannon L Risacher, Leslie M Shaw, John Q Trojanowski, et al. Identifying neuroimaging and proteomic biomarkers for mci and ad via the elastic net. In *International Workshop on Multimodal Brain Image Analysis*, pp. 27–34. Springer, 2011.

Artem Sokolov, Daniel E Carlin, Evan O Paull, Robert Baertsch, and Joshua M Stuart. Pathway-based genomics prediction using generalized elastic net. *PLoS computational biology*, 12(3): e1004790, 2016.

Aboozar Taherkhani, Georgina Cosma, and T Martin McGinnity. Deep-fs: A feature selection algorithm for deep boltzmann machines. *Neurocomputing*, 322:22–37, 2018.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Jorge R Vergara and Pablo A Estévez. A review of feature selection methods based on mutual information. *Neural computing and applications*, 24(1):175–186, 2014.

Hui Wang, Elizabeth C Curran, Thomas R Hinds, Edith H Wang, and Ning Zheng. Crystal structure of a taf1-taf7 complex in human transcription factor iid reveals a promoter binding module. *Cell research*, 24(12):1433–1444, 2014.

Kohji Yamada, Mutsumi Hayashi, Hiroko Madokoro, Hiroko Nishida, Wenlin Du, Kei Ohnuma, Michiie Sakamoto, Chikao Morimoto, and Taketo Yamada. Nuclear localization of cd26 induced by a humanized monoclonal antibody inhibits tumor cell growth by modulating of polr2a transcription. *PloS one*, 8(4):e62304, 2013.

Yan-Wei Yin, Hong-Jian Jin, Wenjing Zhao, Beixue Gao, Jiangao Fang, Junmin Wei, Donna D Zhang, Jianing Zhang, and Deyu Fang. The histone acetyltransferase gcn5 expression is elevated and regulated by c-myc and e2f1 transcription factors in human colon cancer. *Gene expression*, 16(4):187, 2015.

Gui-Ping Yu, Yong Ji, Guo-Qiang Chen, Bin Huang, Kai Shen, Song Wu, and Zhen-Ya Shen. Application of runx3 gene promoter methylation in the diagnosis of non-small cell lung cancer. *Oncology letters*, 3(1):159–162, 2012.

Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pp. 856–863, 2003.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

A Appendix

A.1 Biological Significance of the Selected Genes of GM12878

- **POL2(POLR2A)**: It's a subunit of RNA polymerase II, which interacts with nuclear CD26 using a chromatin immunoprecipitation assay. This interaction led to transcriptional repression of the POLR2A gene, resulting in a proliferation of cancer cells Yamada et al. (2013).
- **H4K20ME1** This gene has been implicated in transcriptional activation. Recent studies showed a strong correlation between H4K20me1 and gene activation in the regions downstream of the transcription start site Barski et al. (2007).
- **NRSF(REST)** NRSF/REST is highly expressed in non-neuronal tissues like the lung. The findings of Kreisler et at. Kreisler et al. (2010) support that NRSF/REST may act as an essential modulator of malignant progression in small-cell lung cancer.
- **TAF1** It's the largest integral subunit of TFIID, initiates RNA polymerase II-mediated transcription. Wang et al. discovered a critical promoter-binding function of TAF1 in transcription regulation Wang et al. (2014).
- **H3K27AC** This gene distinguishes active enhancers from inactive/poised enhancer elements containing H3K4me1 alone Creyghton et al. (2010).
- **GCN5** GCN5 functions as a transcriptional coactivator of E2f1 target genes. In small-cell lung cancer, E2F1 recruits GCN5 to acetylate H3K9, facilitating transcription of E2F1, CYCLIN E, and CYCLIN D1 (39) all of which promote cellular proliferation and tumor growth Yin et al. (2015).
- **PML** The PML gene provides instructions for a protein that acts as a tumor suppressor, which means it prevents cells from growing and dividing too rapidly or in an uncontrolled way Salomoni & Pandolfi (2002).
- **RUNX3** This gene binds to the core DNA sequence 5'-PYGPYGGT-3' found in several enhancers and promoters. It also interacts with other transcription factors. It functions as a tumor suppressor, and the gene is frequently deleted or transcriptionally silenced in cancer Yu et al. (2012).
- **ZZZ3** It's portrein binding gene which oftens promotes gene activation Mi et al. (2018).
- **H3K27ME3** This gene can function as silencers to regulate gene expression Cai et al. (2021).

A.2 Indian Pines Feature Sets

Here we list the top ten features from each model. We have included the WaLuMI + SSVM model, where SSVM is applied to the WaLuMI features to prune the set further. There is no common feature among these three sets.

SCE	SSVM	WaLuMI + SSVM
13,148,51,47,49	1,34,2,3,29	5,25,100,55,183
43,45,17,134,44	32,41,39,28,42	129,79,52,68,88

Table 5: List of top ten bands from each model. Bands selected by SSVM and SSVM + WaLuMI are reported from Chepushtanova et al. (2014).