

Subject Section

CoRAE: Concrete Relaxation Autoencoder for Differentiable Gene Selection and Pan-Cancer Classification

Abdullah Al Mamun, Ananda Mondal, and Sanjeev Kaushik Ramani

School of Computing and Information Sciences, Miami, US

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Selecting relevant features from a high-dimensional dataset is a critical study. It aims to select a small subset of features that will increase accuracy and decrease the cost of data classification or clustering. Due to high-dimension with a low number of samples in omics data, classification models encounter over-fitting problem. Thus, there is an urgent need for efficient feature selection methods that will be capable of selecting relevant features. In recent years, standard autoencoder and its variations have been used to select latent features to increase the classification performance. However, these methods are unable to provide which original features are contributing to these latent features. In this paper, we introduced a novel global feature selection method based on concrete relaxation discrete random variable selection, which can efficiently identify a subset of most significant features that have an effective contribution in data reconstruction and classification. The proposed method is a variation of standard autoencoder where a concrete feature selection layer is added in the encoder and a standard neural network is used as a decoder. During training, a predefined temperature of the feature selection layer steadily decreased which allows the model to learn a user-specified number of discrete features. Also, during testing, only selected features can be used to reconstruct the input in the decoder.

Results: We evaluated the proposed feature selection method on coding and non-coding gene expression of 33 cancer samples from TCGA where it significantly outperforms state-of-the-art methods in identifying top 100 coding and non-coding genes. Later, expression of selected genes is used to train a linear classifier to distinguish 33 cancer types where features selected by CoRAE shows highest performance up to 99%. The proposed method can be implemented by adding a few lines of code to the standard autoencoder.

Availability: Source code and sample dataset can be found in <https://github.com/pwaabdullah/MyPhD.git>

Contact: amondal@fiu.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

A major issue while working with recent omics data is the availability of greater number of features in comparison to the available number of samples leading to a highly imbalanced feature-sample ratio. It may be great to argue that the larger the feature set, the better the classification that is possible. However, in the contrary, in a general setting, not all of these features will be necessary for optimal classification. Only a selected number of significant features when used with the classifier can lead to optimal classification leading to the distinguishing samples

that belong to different classes. A large part of the remaining features are not too significant and could be either noise, irrelevant to the study or even redundant (Pirgazi *et al.*, 2019). The use of such insignificant features can lead to unwanted computational complexities and hamper the performance of the system. This is more pronounced when working with data having high dimensionality. Thus, it is essential to identify the set of significant features that can provide us with the optimal classification and clustering. For this to be accomplished, we need a robust method that can eliminate the redundant features and noise that do not have any information about the labels leaving us with only relevant features (Liu and Motoda, 2012).

Any dataset with N number of features has 2^N possible subset of features. In the presence of such large number of possible combinations, finding the best subset of N features is computationally challenging and expensive (Liang *et al.*, 2018). An optimally selected set of features optimizes the performance of the models and also helps in alleviating the effect of *overfitting* and *high-dimensionality*. Along with the above benefits, selecting the appropriate features helps in easier interpretation of the model and thus its predictions. Also, the use of the gratuitous features can significantly impact the training speeds and the accuracy of the learning models. Overall, appropriate feature selection can provide the following advantages: (a) reduced cost for computation and storage, (b) adequate use of the available sample set for improved performance, (c) improved timing for classification and thus predictions, (d) easier interpretation of the the data and thus the final predictions.

Filter, wrapper, embedded methods are the three general classes / types of feature selection algorithms. Numerous algorithms have been proposed for each of these types of feature selection methods. The filtering method works by a ranking the features using a statistical score that is assigned to each of them depending on their relevance to the class type. In both the univariate and multivariate filter methods, the interactions among features are disregarded in the selection process. Studies like the ones in Pearson correlation coefficient(PC), t-statistics(TS) (Speed, 2003), F-Test (Ding and Peng, 2005), and ANOVA (Ding and Li, 2015) are examples where the filtering method is used. It is observed that these methods are effective for selecting features in high-dimensional data because of the reduced computation expenses. However, they fail to provide good accuracy as discussed in (Sun *et al.*, 2018).

As an enhancement, the wrapper method is proposed with a learning algorithm and a classifier to find a suitable subset of features. Initially, a random solution is generated following which, an objective function is maximized using black-box optimization methods (Rau *et al.*, 2019) like simulated annealing (Jeong *et al.*, 2018), particle swarm optimization (Xue *et al.*, 2012), genetic algorithm (Wu *et al.*, 2011) and, ant colony optimization (Kabir *et al.*, 2012). The evaluation of every candidate subset of the feature iteratively by the method leads to identification of a strong relationship between features however with an increase in the computational expense. Embedded methods are very efficient as they are a part of the learning phase. This helps reduce the computation costs. Well-known example of the embedded method are LASSO (Tibshirani, 1996), recursive feature elimination with state vector machine estimator (SVM-RFE) (Abdullah, 2019; Guyon *et al.*, 2002; Fang, 2019), random forest (Pouyan and Kostka, 2018; Ram *et al.*, 2017), Adabost (Wang, 2012), KNN (Le *et al.*, 2019), and autoencoder (Lu *et al.*, 2019).

In general, the use of feature selection is worthwhile when using the whole set of features is difficult or the cost of execution is high. As an example, the gene expression dataset contains more than 60 thousand features with a very low number of samples. It thus is important to answer the impending questions: *Is it possible to identify important genes those expressions can classify available disease or cancer type?*. Feature selection works differently compared to the standard dimension reduction techniques such as principal component analysis (PCA) (Hotelling, 1933), and autoencoders (Hinton and Salakhutdinov, 2006). The above mentioned methods can preserve maximum variance with a highly reduced number of features. However, these methods do not provide the original features of the dataset making it difficult to improbable to eliminate redundant or irrelevant features from the dataset.

In this paper, we propose a novel feature subset selection method that increases the power of a standard deep autoencoder for differentiable feature selection. Our major contributions in this paper is two fold: (a) we create a new variant of the standard autoencoder by introducing the use of a concrete relaxation discrete random variable selection layer for

encoding allowing selection of user-defined number of original features, (b) evaluation of the performance of the proposed approach on coding and non-coding gene expression datasets, (c) modifying the codebase of a standard autoencoder to realize the proposed approach. Our initial evaluations show the improved performance of the proposed approach in comparison to the existing state-of-the-art methods in identifying the top 100 coding and non-coding genes that can distinguish 33 different types of cancers. The results also show a significant increase in the classification performance up to 99%. Idea of concrete distribution is adapted from (Maddison *et al.*, 2016; Kingma and Welling, 2013), and reparameterization technique to minimize the loss and reconstruction error from (Abid *et al.*, 2019).

2 Materials and Methods

2.1 Coding and Non-coding Gene Expression

To validated the proposed idea, TCGA RNAseq cancer (n=9566) and clinical samples for 33 cancers were downloaded from UCSC Xena database (<https://xenabrowser.net>). TCGA processed raw RNAseq data uses Illumina HiSeq 2000 RNA sequencing platform where per-gene normalized abundance estimation is calculated using the FPKM method. RNASeq normalized counts are then log transformed after adding a constant of 1. Subsequently, UCSC re-processed this using GENCODE v23 transcript annotation to quantify protein coding and non-coding transcripts expression (Harrow *et al.*, 2006). Coding genes refer to mRNA whereas non-coding genes refer to long non-coding RNA (lncRNA) in this experiment. To improve the focus on individual feature selection, we separated the mRNA and lncRNA expression from the combined database using the standard list of lncRNAs provided by TANRIC (Li *et al.*, 2015).

Another important reason for performing the experiments on individual RNA types is because of their varying expression levels. The number of mRNA and lncRNA are 18731, 12309 respectively. We merged all the cancer samples for individual RNA types to aid further experimentation. Each row is mapped to a unique Ensemble ID, and each column mapped to a patient ID. Normal patients or RNA with missing data were removed from the original dataset. Each RNA expression was further processed using min-max normalization method to achieve good training performance.

2.2 Concrete Relaxation Autoencoder

The concrete relaxation autoencoder (CoRAE) is a variation of the original autoencoder (AE) (Hinton and Salakhutdinov, 2006) which is used for dimension reduction. It is a neural network that consists of two parts: (a) an encoder that selects latent features and (b) a decoder that uses selected features to reconstruct an output that matches the input. Instead of using a sequence of fully connected layers in the encoder, we propose a concrete relaxation based feature selection layer where the user can define the number of nodes (features), k . This layer selects probabilistic linear arrangement of input features while training, which converges to a discrete set of k features by the end of training and during the testing.

The original features are selected based on the temperature of this layer which is tuned using an *annealing schedule*. More specifically, the concrete selector layer identifies k important features as the temperature decreases to zero. For reconstructing the input, a simple decoder similar to the ones associated with a standard AE is used. This simple neural network can be updated based on the characteristics of the data and its complexity.

The layer that selects the features and highlighted in Figure 1 is called concrete variable selector layer adopted from the concrete distribution (Maddison *et al.*, 2016) and categorical representation (Jang *et al.*, 2016).

Table 1. Sample Distribution for 33 cancers along with 75-25 split for training and testing.

Sl	Cancer site name	Acronym	#Sample	#Train	#Test	Sl	Cancer site name	Acronym	#Sample	#Train	#Test
1	Adrenocortical Cancer	ACC	77	57	20	18	Lung Squamous Cell Carcinoma	LUSC	498	373	125
2	Bladder Cancer	BLCA	407	305	102	19	Mesothelioma	MESO	86	64	22
3	Breast Cancer	BRCA	1089	816	273	20	Ovarian Cancer	OV	375	281	94
4	Cervical Cancer	CESC	304	228	76	21	Pancreatic Cancer	PAAD	177	132	45
5	Bile Duct Cancer	CHOL	36	27	9	22	Pheochromocytoma & Paraganglioma	PCPG	177	132	45
6	Colon Cancer	COAD	301	225	76	23	Prostate Cancer	PRAD	493	369	124
7	Large B-cell Lymphoma	DLBC	47	35	12	24	Rectal Cancer	READ	95	71	24
8	Esophageal Cancer	ESCA	161	120	41	25	Sarcoma	SARC	258	193	65
9	Glioblastoma	GBM	158	118	40	26	Melanoma	SKCM	465	348	117
10	Head and Neck Cancer	HNSC	499	374	125	27	Stomach Cancer	STAD	378	283	95
11	Kidney Chromophobe	KICH	66	49	17	28	Testicular Cancer	TGCT	132	99	33
12	Kidney Clear Cell Carcinoma	KIRC	527	395	132	29	Thyroid Cancer	THCA	501	375	126
13	Kidney Papillary Cell Carcinoma	KIRP	287	215	72	30	Thymoma	THYM	118	88	30
14	Acute Myeloid Leukemia	LAML	147	110	37	31	Endometrioid Cancer	UCEC	184	138	46
15	Lower Grade Glioma	LGG	507	380	127	32	Uterine Carcinosarcoma	UCS	56	42	14
16	Liver Cancer	LIHC	369	276	93	33	Ocular melanomas	UVM	79	59	20
17	Lung Adenocarcinoma	LUAD	512	384	128		Total		9566	7161	2405

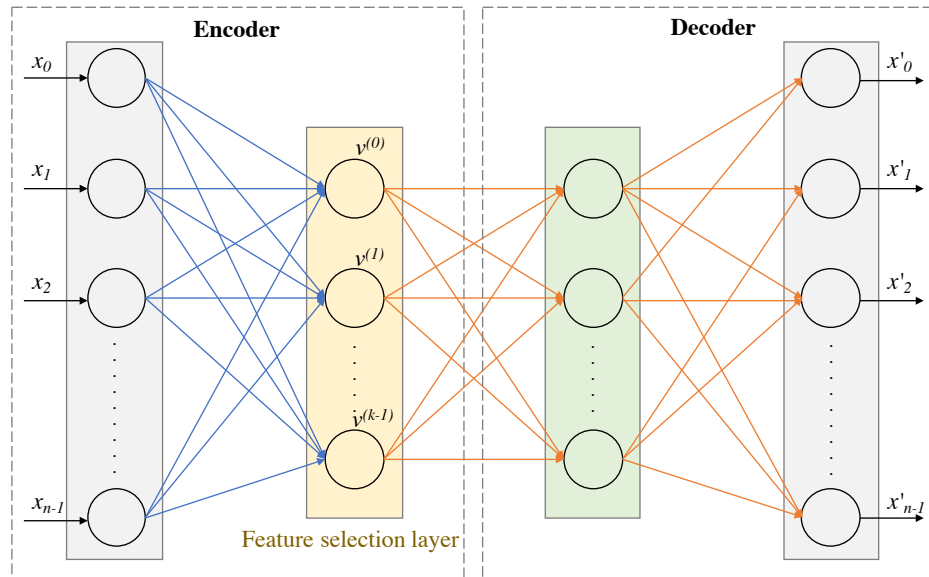


Fig. 1. Architecture of Concrete Relaxation Autoencoder. Proposed feature selection architecture consists of an encoder and a decoder. The layer after input layer in encoder is called concrete feature selection layer shown in yellow. This layer has k number of node where each node is for each feature to be selected. During the training stage, the i^{th} node $v^{(i)}$ takes the value $x^T f^{(i)}$. During testing stage, these weights are fixed and the element with the highest value is selected by the corresponding i^{th} hidden node. The architecture of the decoder remains the same during train and test stage.

Since backpropagation does not allow computation of the parameters' gradient through stochastic nodes of a standard autoencoder, gumbel *softmax* distribution g (Gumbel, 1954) is the right choice to pick samples z from categorical distribution with class probabilities α_k .

$$z = \text{one-hot}(\arg\max_k [g_k + \log \alpha_k]) \quad (1)$$

Because $\arg\max$ is not differentiable, simple *softmax* function can be used as a continuous approximation of $\arg\max$. The aim of using Concrete random variables is to relax the state of a discrete variable and the relaxation degree is controlled by a temperature parameter $\tau \in (0, \infty)$. To sample a concrete random variable in z dimension with parameter $\alpha \in \mathbb{R}^z > 0$ and τ , one must sample a z -dimensional vector of *i.i.d.* (independent and identically distributed) samples from a Gumbel

distribution, g . Then each element of the sample f from the Concrete distribution can be defined as:

$$f_k = \frac{\exp((\log \alpha_k + g_k)/\tau)}{\sum_{i=1}^z \exp((\log \alpha_i + g_i)/\tau)} \text{ for } k = 1, \dots, z \quad (2)$$

where f_k refers to the k_{th} element in a particular sample vector. With the limit $\tau \rightarrow 0$, the concrete variable uniformly progresses the discrete distribution, producing one-hot vector with $f_k = 1$ with a probabilistic chance of $\alpha_k / \sum_p \alpha_p$. The advantage of using a concrete random discrete variable is that it is differentiable *w.r.t* α using reparameterization technique as mentioned by (Kingma and Welling, 2013).

More concisely, the way original feature is selected using the concrete random variable as follows: a z -dimensional concrete random variable $f^{(i)}$ is sampled for each node of the selector layer with k nodes where i refers to the index of the node, $i \in \{1 \dots k\}$. The output of the i^{th} node is $\mathbf{x} \cdot f^{(i)}$. Although it is a combination of the input feature's weight, every node of the selector layer produces exactly one of the original input features in the limit $\tau \rightarrow 0$. After training the network, a discrete argmax layer is replaced with the concrete selector layer by which $x_{\arg\max_j \alpha_j^{(i)}}$ is produced as an output of i^{th} node during the testing phase. The value of α_i initially starts with a small positive random number so that it can explore various combinations of input features. As the model is being trained, the value of α_i , in other words the probability of class i becomes more stable. As a result, the model reduces its stochasticity rather increases the confidence in drawing a particular subset of features.

The temperature of the random variable in the selector layer has a significant impact in forming the output of each node. Initially, when τ is high, search space is large since it considers a linear combination of all features. In contrast, the selector layer will not be able to search all possible combinations of features in low τ and thus, model converges to a bad local minima. Instead of using a fixed temperature, a simple annealing scheduling scheme is used for every concrete variable. It starts with an user-defined high temperature (τ_s) and steadily lowers the temperature until it touches the ending bound (τ_e) by every epoch as follows:

$$\tau(e) = \tau_s (\tau_e / \tau_s)^{e/N} \quad (3)$$

where $T_{(e)}$ is the temperature at epoch e , N refers to the total number of epochs. The proposed annealing schedule is good enough to explore the feature combinations during the training phase and finally lowered temperature enables the model to strict to the best set of features which is shown in Figure 3.

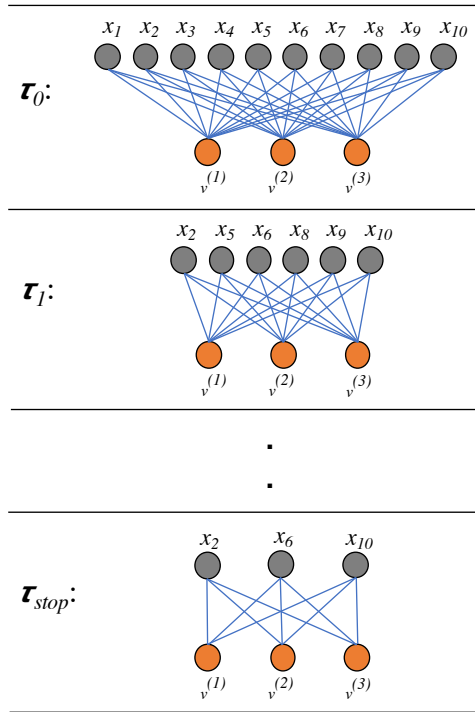


Fig. 2. Temperature Effect.

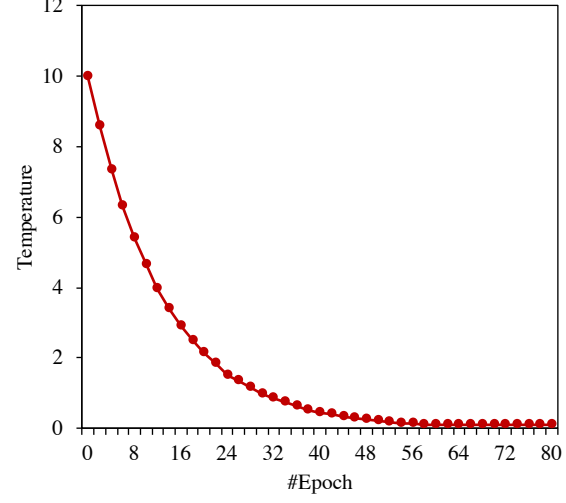


Fig. 3. Annealing schedules for the CoRAE. Effect of different annealing schedules on a concrete autoencoder trained on the mRNA dataset with $k = 100$ selected features. If the temperature is exponentially decayed (the annealing schedule), the feature selected layer (model) converges to informative features.

2.3 Gene Selection, Classification, Reconstruction, and Evaluation

The encoder of the Concrete relaxation autoencoder (CoRAE) architecture is constructed with a hidden layer of k nodes where k is the number of genes selected. The decoder, on the other hand, is consisting of one hidden layer with $3k/2$ nodes. The number of nodes in this layer is tuned in a range of $[4k/7, 2k/5, 3k/2]$. Adam optimizer with a learning rate of 10^{-3} is used for all the experiments. The starting temperature of the CoRAE was set to 10 and it ends at 0.01. To avoid overfitting, the dataset is split into the train and test set according to 75/25 ratio. The training set is used to estimate the learning parameters and the test set is used for performance evaluation. To control the performance, the model is trained for the same number of epoch 100.

Performance of CoRAE has been compared with state-of-the-art feature selection techniques such as LASSO and SVM-RFE on both mRNA and lncRNA expression datasets. In LASSO, a regularization parameter α decides the number of most important features. More precisely, the higher the α , the more feature's coefficient shrinks to zero, and fewer features will be selected. Recursive feature elimination is a recursive method in which less important features are eliminated in every iteration. In the recursive feature elimination technique, SVM is used as an estimator. Linear kernel with a regularization parameter $C = 0.05$ is used. C controls the tradeoff between the error and norm of the learning weights. GridSearch algorithm is used to estimate the best set of parameters for SVM. In every iteration of RFE, the number of dropped features is set to 100.

We extract a subset of features by varying k from 10 to 500. For the comparison to be fair and along the same grounds with CoRAE, the same number of genes are selected using LASSO and SVM-RFE. The dataset with reduced number of features (expression of selected genes) to the SVM for classifying 33 cancer types on both mRNA and lncRNA expression. Similarly, to reconstruct all the input features, we trained a linear regressor with no regularization and measure the reconstruction mean square error. LASSO and SVM-RFE are developed using scikit learn framework (Pedregosa et al., 2011) whereas CoRAE is built using Google developed Tensorflow (Abadi et al., 2015) based

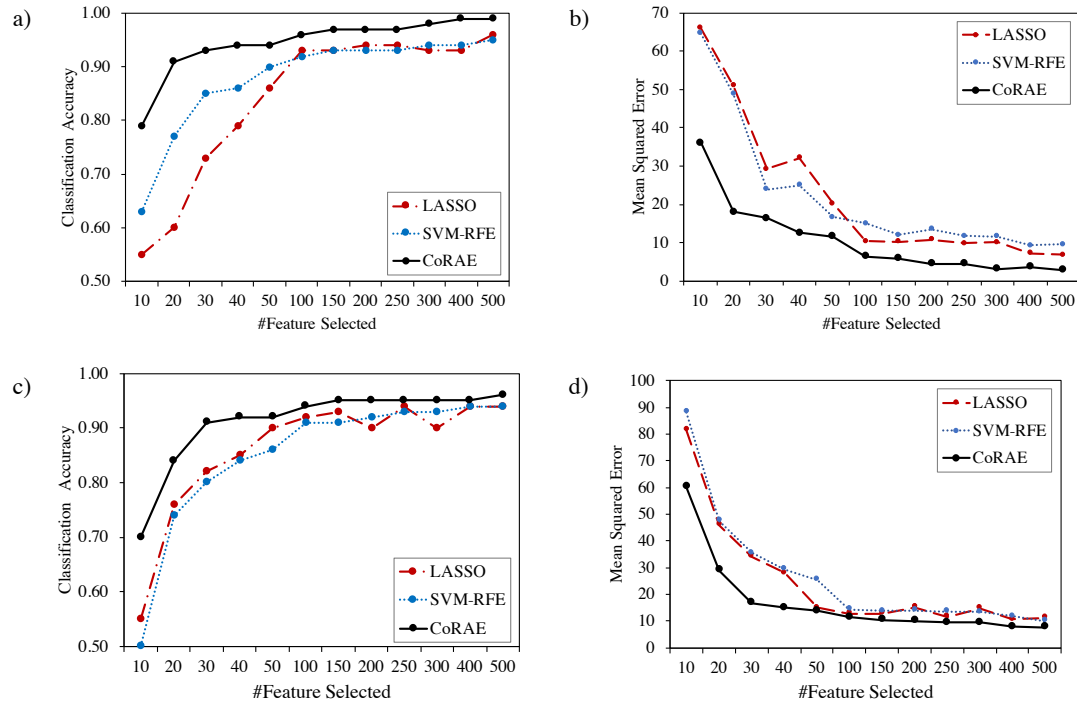


Fig. 4. Classification performance using selected RNA features. Comparison of CoRAE with other feature selection methods. Throughout the all values of k tested on both mRNA(a) and lncRNA(c) CoRAE have highest classification accuracy. Similarly, it shows lowest reconstruction mean squared error on both mRNA(b) and lncRNA(d)

deep learning framework Keras (Chollet *et al.*, 2015). Experiments are parallelized on NVIDIA Quadro K620 GPU with 384 cores and 2GB memory devices. Five different evaluation metrics have been used to record the classification and reconstruction performance such as accuracy, precision, recall, f1 score, and mean squared error (MSE).

Accuracy is the number of correct predictions made by the model over all kinds of predictions made. True positives(TP) and True Negatives(TN) are the correct predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

Precision is the number of correct positive results divided by the number of positive results predicted by the classifier. It indicates the predicted positive portion of the samples.

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

Recall is the number of correct positive results divided by the number of all relevant samples.

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

F1 score is a measure of a test's accuracy. It considers both the precision and the recall of the test to compute the score.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

Mean squared error (MSE) is the average of $(\frac{1}{n} \sum_{i=1}^n)$ of the square of the errors $(Y_i - Y'_i)$ where Y_i is a true label and Y'_i is a predicted label. All performance metrics is measured on the predicted labels and true labels of independent test samples.

Top genes can be selected based on two criteria - a) classification accuracy needs to be higher, and b) the number of genes should be as less as possible so that biologists can conduct a wet lab experiment easily. The capabilities of selected genes in pan-cancer classification is visually validated using unsupervised visualization technique t-SNE (Maaten and Hinton, 2008).

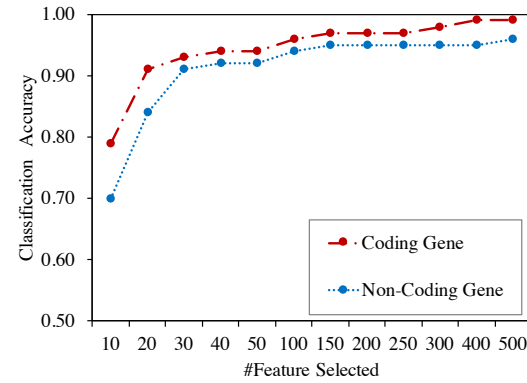


Fig. 5. Classification accuracy comparison between coding and non-coding genes expression. Across all the k , mRNA expression shows slightly better classification accuracy over lncRNA expression. Here, these features has been selected using proposed method only.

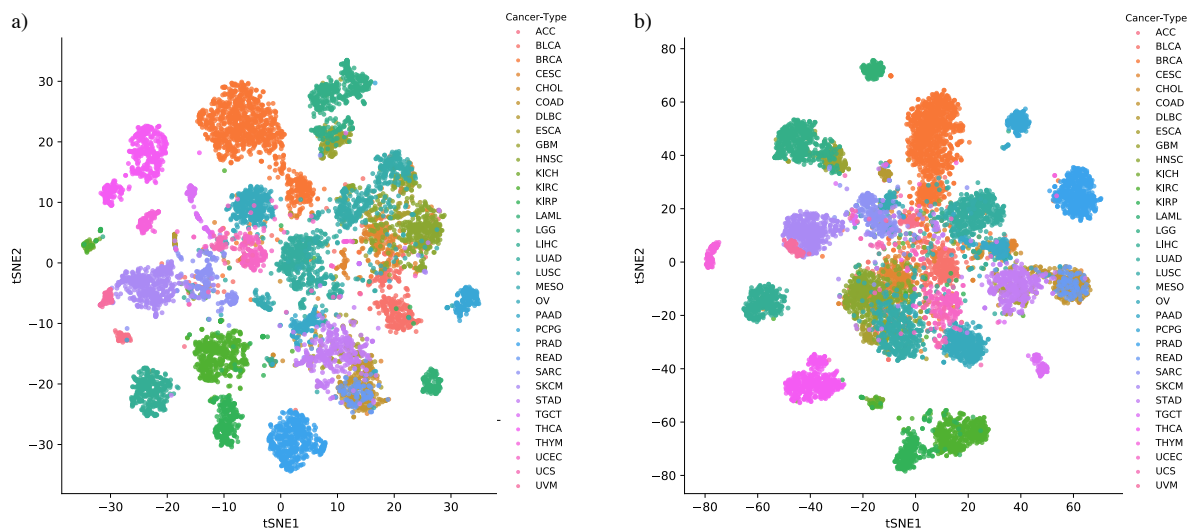


Fig. 6. Visualization of 33 different cancer types using top-100 CoRAE features. Here, we show the t-SNE representation of 33 cancer samples using selected features. Each dot represents a cancer sample and each color represents a cancer type. a) t-SNE using top 100 mRNA, b) t-SNE using top 100 lncRNA

3 Results

A series of experiments were conducted to compare the performance of CoRAE with other state-of-the-art feature selection methods such as LASSO and SVM-RFE. Each of these three methods were used to select features in the range of 10 to 500. These features are then used to train a linear classifier (SVM) using selected coding and non-coding gene expression of 33 cancer patients. Figure 4 showcases the selected features. It can be observed that the scale used in the x-axis to depict the number of features does not increase with a consistent step size. The initial stages of the experiments were performed with a smaller subset of the selected features as we wanted to understand the performance of the models being compared. The optimal classification performance was observed when 100 features were used. Beyond this the trend continues as shown in the figure. We continued to monitor the performance of the models till 500 features to check for the best possible performance.

For all selected k number of features, as depicted in Figure 4, CoRAE has highest accuracy and lowest error for both the mRNA and lncRNA expression. It can be observed that even with a smaller number of significant features (say 100), the accuracy of CoRAE is close to 80% whereas LASSO and SVM-RFE shows poor results for the same number of features. When the number of features is increased to a value of 50, CoRAE outperforms the other compared methods to provide an accuracy $> 90\%$.

The CoRAE method is resilient to errors that occur during reconstruction using a small feature set. In comparison, this error is more pronounced in the other competing methods. In specific, for the case considering mRNA, CoRAE starts with a MSE of 38 and quickly reduces to a value less than 10 within the use of the top 100 features. The behaviour in classification is highly comparable in both coding and non-coding genes. However, mRNA expression performs slightly better than lncRNA as shown in Figure 5.

3.1 Selected Features Interpretation

With the use of CoRAE, we are able to identify important features while allowing the user to examine the relevance of each feature by observing the corresponding estimated concrete parameter $\alpha^{(i)}$. In CoRAE, feature

selection is based on the value of vector $\alpha^{(i)}$ which gives the user the value of the importance score which in turn gives the power to identify features and based on their correlation with the other selected features. Figure 6, highlights how the top 100 mRNA or lncRNA selected using the CoRAE is capable of distinguishing the 33 cancer types. It can also be observed that the features selected using the CoRAE method carries more information than the ones selected by the other methods. The use of such features improves the prediction accuracy. The improved performance is highlighted in Figures ?? with the CoRAE method succeeding in picking more significant features in comparison to the other state-of-the-art methods.

4 Discussion

In this paper, a new differentiable feature selection method via backpropagation is proposed. The proposed concrete relaxation auto-encoder uses re-parameterization and concrete random variable technique to allow gradients to pass through a layer that stochastically selects discrete original input features of higher significance. The randomness of the proposed method enables it to effectively search and converge to a user-defined number of original features which maximizing the objective function and minimizing the loss as discussed in section 2.2. The estimated parameters learned by the models can be further examined by the biologists and other stakeholders to interpret biological relevance as discussed in section 3.1. The above mentioned characteristics of CoRAE provides it with a distinction from numerous other competing approaches which are based on regularization.

Since CoRAE is built on top of a standard autoencoder architecture, it is easily scalable to a higher number of samples or dimensions. It is observed that the features selected by the CoRAE outperformed the ones from the competing methods. This paper accounts for a generalized approach of CoRAE. However, there are other avenues that can be explored using the proposed method. As an example, CoRAE can be used to extract important genes during the molecular subtype classification of a single cancer dataset unlike the existing approaches which are based on the multiple cancer classification. The proposed method also provides

the privilege to users to integrate multi-omics data such as gene, protein, RNAseq expression, DNA methylation, copy number and so on.

CoRAE is easy to use and requires modifying a few lines for implementing it in the popular machine learning frameworks. Moreover, the runtime and space complexity is similar to that of the standard autoencoder. In addition, it enhances parallelization and hardware acceleration which is an obvious demand for deep learning. Starting and ending temperature are the only added hyperparameters used for annealing schedule. The default values used in the experiment are carefully identified and is found to work adequately for the various datasets.

5 Conclusion

In this paper, we propose a novel feature subset selection method that increases the power of a standard deep autoencoder for differentiable feature selection. The proposed CoRAE is a new variant of the standard autoencoder which uses a concrete relaxation discrete random variable selection layer for encoding allowing selection of user-defined number of original features. By modifying the codebase of a standard autoencoder, we are able to successfully realize the proposed CoRAE method. We evaluate the performance of the proposed approach on coding and non-coding gene expression datasets and compare the results with state-of-the-art methods like LASSO and SVM-RFE. Our experiments show that on publicly available gene expression cancer datasets that CoRAE efficiently maximizes the classification accuracy and minimizes the reconstruction error using a selected subset of genes. For both mRNA and lncRNA gene expression datasets, CoRAE outperformed several sophisticated feature selection techniques. Use of a single hidden layer in the decoder, minimizes the reconstruction error and allows for selection of features from large datasets.

As a part of the future work, we will conduct more biological validation such as survival analysis of 33 cancer patients using selected features to measure the prognostic capabilities. Similarly, pathway analysis of selected coding and non-coding genes will be analyzed in the future.

Acknowledgements

This research is partially funded by the US National Science Foundation CAREER award #1651917 (transferred to #1901628) to AMM.

References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- Abdullah, Al Mamun, A. M. (2019). Feature selection and classification reveal key lncrnas for multiple cancers.
- Abid, A., Balin, M. F., and Zou, J. (2019). Concrete autoencoders for differentiable feature selection and reconstruction. *arXiv preprint arXiv:1901.09346*.
- Chollet, F. et al. (2015). Keras. <https://github.com/fchollet/keras>.
- Ding, C. and Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology*, **3**(02), 185–205.
- Ding, H. and Li, D. (2015). Identification of mitochondrial proteins of malaria parasite using analysis of variance. *Amino acids*, **47**(2), 329–333.
- Fang, J. (2019). Tightly integrated genomic and epigenomic data mining using tensor decomposition. *Bioinformatics*, **35**(1), 112–118.
- Gumbel, E. J. (1954). Statistical theory of extreme values and some practical applications. *NBS Applied Mathematics Series*, **33**.
- Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, **46**(1-3), 389–422.
- Harrow, J., Denoeud, F., Frankish, A., Reymond, A., Chen, C.-K., Chrast, J., Lagarde, J., Gilbert, J. G., Storey, R., Swarbreck, D., et al. (2006). Gencode: producing a reference annotation for encode. *Genome biology*, **7**(1), S4.
- Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, **313**(5786), 504–507.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, **24**(6), 417.
- Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.
- Jeong, I.-S., Kim, H.-K., Kim, T.-H., Lee, D. H., Kim, K. J., and Kang, S.-H. (2018). A feature selection approach based on simulated annealing for detecting various denial of service attacks. *Software Networking*, **2018**(1), 173–190.
- Kabir, M. M., Shahjahan, M., and Murase, K. (2012). A new hybrid ant colony optimization algorithm for feature selection. *Expert Systems with Applications*, **39**(3), 3747–3763.
- Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Le, T. T., Urbanowicz, R. J., Moore, J. H., and McKinney, B. A. (2019). Statistical inference relief (stir) feature selection. *Bioinformatics*, **35**(8), 1358–1365.
- Li, J., Han, L., Roebuck, P., Diao, L., Liu, L., Yuan, Y., Weinstein, J. N., and Liang, H. (2015). Tanric: an interactive open platform to explore the function of lncrnas in cancer. *Cancer research*, **75**(18), 3728–3737.
- Liang, S., Ma, A., Yang, S., Wang, Y., and Ma, Q. (2018). A review of matched-pairs feature selection methods for gene expression data analysis. *Computational and structural biotechnology journal*, **16**, 88–97.
- Liu, H. and Motoda, H. (2012). *Feature selection for knowledge discovery and data mining*, volume 454. Springer Science & Business Media.
- Lu, X., Gu, H., Wang, Y., Wang, J., and Qin, P. (2019). Autoencoder based feature selection method for classification of anticancer drug response. *Frontiers in genetics*, **10**, 233.
- Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, **9**(Nov), 2579–2605.
- Maddison, C. J., Mnih, A., and Teh, Y. W. (2016). The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.
- Pirgazi, J., Alimoradi, M., Abharian, T. E., and Olyae, M. H. (2019). An efficient hybrid filter-wrapper metaheuristic-based gene selection method for high dimensional datasets. *Scientific Reports*, **9**(1), 1–15.
- Pouyan, M. B. and Kostka, D. (2018). Random forest based similarity learning for single cell rna sequencing data. *Bioinformatics*, **34**(13), i79–i88.
- Ram, M., Najafi, A., and Shakeri, M. T. (2017). Classification and biomarker genes selection for cancer gene expression data using random forest. *Iranian journal of pathology*, **12**(4), 339.
- Rau, A., Flister, M., Rui, H., and Auer, P. L. (2019). Exploring drivers of gene expression in the cancer genome atlas. *Bioinformatics*, **35**(1), 62–68.
- Speed, T. (2003). *Statistical analysis of gene expression microarray data*. Chapman and Hall/CRC.
- Sun, Y., Lu, C., and Li, X. (2018). The cross-entropy based multi-filter ensemble method for gene selection. *Genes*, **9**(5), 258.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, **58**(1), 267–288.
- Wang, R. (2012). Adaboost for feature selection, classification and its relation with svm, a review. *Physics Procedia*, **25**, 800–807.
- Wu, Y.-L., Tang, C.-Y., Hor, M.-K., and Wu, P.-F. (2011). Feature selection using genetic algorithm and cluster validation. *Expert Systems with Applications*, **38**(3), 2727–2732.
- Xue, B., Zhang, M., and Browne, W. N. (2012). Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE transactions on cybernetics*, **43**(6), 1656–1671.