UNSUPERVISED FEATURE RANKING AND SELECTION BASED ON AUTOENCODERS

Sasan Sharifipour[1], *Hossein Fayyazi*[1], *Mohammad Sabokrou*[2], *Ehsan Adeli*[3]

[1]AI & ML Center of Part [2]Institute for Research in Fundamental Sciences (IPM) [3]Stanford University

ABSTRACT

Feature selection is one of the most important and widely-used dimension reduction techniques due to its efficiency and intractability of the results. In this paper, we propose a simple but efficient unsupervised feature ranking and selection method by exploiting the geometry of the original feature space using AutoEncoders. Average reconstruction error of training samples by ignoring features, one at time, and the contribution of feature in the latent space (bottleneck of the auto-encoder) are proposed as two useful measures for ranking the features. The proposed method is evaluated for three different tasks: (1) feature selection, (2) discovering image interest points, and (3) extracting important blocks of an images Result on standard benchmarks confirm that the performance of our method is better than state-of-the-art methods.

Index Terms— Feature selection, ranking, auto-encoder.

1. INTRODUCTION

Performance of machine learning algorithms, especially for classification and clustering tasks, is highly depended on how they are equipped or provided with a low dimensional representation of raw data. To simplify models, facilitate easier interpretations, reduce training time, avoid the curse of dimensionallity, and refrain from over-fitting, the input raw data must be described by a discriminative feature set. Often this procedure is referred to as feature (or variable) selection [1]. With regards to the availability of labels, feature selection techniques can be divided into two main categories: *supervised* and *unsupervised* methods. Supervised algorithms [2, 3, 4, 5] benefit from available labels to select discriminative features that classify the samples according to the class labels. However, with the increasing number of data samples generated with many different types, labeling them all is a cumbersome and an expensive task. Consequently, unsupervised methods have been heavily researched in different fields [6, 7, 8, 9]. Without label information to define feature relevance, a number of alternative criteria have been proposed for unsupervised feature selection. One commonly used criterion is to select features that can preserve the data similarity or manifold structure constructed from the whole feature space [7, 4]. In recent years, applying sparse learning for unsupervised feature selection has attracted increasing attention. These methods usually generate cluster labels via clustering algorithms and then transform unsupervised feature selection to sparse learning based supervised feature selection with these generated cluster labels. Some

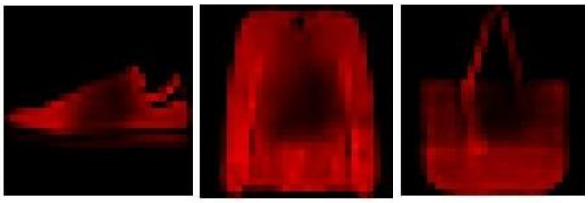

Fig. 1. Examples of three images with ranked pixels using our approach. The (red) color indicates the importance and the tone relates to its importance.

of these methods are Multi-cluster feature selection (MCFS) [10], Nonnegative Discriminative Feature Selection (NDFS) [11], and Robust Unsupervised Feature Selection (RUFS) [12]. Unlike supervised methods, all these approaches assume there is no class label associated with the training samples. In this paper, we explore the ability to reconstruct samples and the correlation between features, as these are two important criteria to select a discriminiative non-redundant subset of features. Principal Component Analysis (PCA) and Auto-Encoder (AE) [13], which are good tools for exploiting the former two criteria, are widely exploited for learning discriminative features (in unsupervised settings) from the raw data (e.g., in [14]). These methods provide a low dimension representation of the raw data. Albeit extracted features by PCA and AE efficiently contribute to building different machine learning models, but these methods often lose the relevant information about the original feature space and importance and relevance of features may not be identified by these methods [15]. Furthermore, in many applications (such as many image processing or medical use cases), a set of predetermined features are at hand and identifying a compact set of important and relevant ones is crucial for understanding the underlying reasons and causation of the results. Although previous feature selection methods (including the ones described above based on sparse learning) have interpertable results, they suffer from several weaknesses: (1) Their performance results are often not comparable to those of PCA or AE, i.e., transforming the feature space to a lower dimensional space works better than solely selecting feature subsets; (2) These methods are often unable to rank features based on their importance, and often. In this paper, we propose a feature ranking and selection strategy based on AEs. As mentioned earlier, AEs are popular tools for feature learning by providing an informative representation of samples (in a

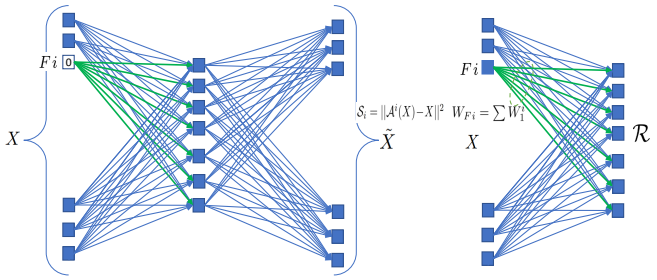

Fig. 2. Let X be the input feature vector, in which i^{th} feature (i.e., F_i) is set to zero. (Left): The reconstruction error S_i on a learned AE can measure the importance of F_i. (Righ): Summation of connected weights (from input to hidden layer) to i^{th} features can also measure the importance of F_i.

latent space) with respect to minimizing the reconstruction error. Two main criteria is used to rank the features and identify their importance. First, we look at the contribution of each input feature in forming the latent space (defined by the AE network weights initiate from that feature). Second, we propose a procedure to identify how ignoring each feature may affect the reconstruction loss of the AE. We iteratively zero out each feature, one at a time, and if a high loss is imposed for the reconstruction, that feature is deemed as informative and important. Fig. 1 shows the output of our method for ranking of image pixels. Based on these two intuitions, the main contribution of this paper are two folds:(1)Two new measures are proposed for feature selection. We show that their combination leads to the state-of-the-art result (2) Our method can efficiently find image interest-point (and blocks).

2. PROPOSED METHOD

Auto-encoders (AEs) are simple but efficient means of representing samples in lower dimensions [13]. They are optimized to learn an efficient representation $\mathcal{R}$ in their bottleneck with minimum loss of information, i.e., with a good representation learned; the original input can be reconstructed from this latent representation with minimum error. Using this characteristic of auto-encoders, we propose a method for feature selection with two intuitions: (1) $\mathcal{R}$ is an informative representation of its input, therefore, the features contributing the most on reconstruction of the original input from $\mathcal{R}$ are more important; and (2) The overall reconstruction error of the AE in absence of any specific feature is a very good indication of feature importance. If this error is low, we can conclude that specific feature is not of great importance for representing the sample or may be highly correlated with other present features.

Based on these intuitions, our proposed method uses the structure of AEs to rank features reflecting on their importance. Fig. 2 shows a sketch of our method. Building on top of this method, we generalize our method for detecting informative (important) pixels and blocks in images.

2.1. Ranking based on AE

Let $X^{1\times n} = \{F_1, \cdots, F_n\}$ be the raw data, represented by n features. The goal is to find a procedure $\mathcal{P}$ to select informative

Algorithm 1: Unsupervised Feature Selection.

Input: Feature set $\{F_1, \cdots, F_n\}$, $\{X_j \in \mathcal{R}^{1\times n}\}_{j=1}^{j=N}$
Output: K most discriminative feature Learn W_1 and W_2 by optimizing $\mathcal{A}$ on $\{X_j \in \mathcal{R}^{1\times n}\}_{j=1}^{j=N}$
while $i \leq< N$ **do**
  i++
  $\mathcal{S}_i = \frac{1}{N}\sum ||\mathcal{A}^i(X_j) - X||^2{}_{j=1}^{j=N}$
end
$Z = \arg\min_z(1 - \frac{\sum_{j=1}^{j=z}\mathcal{S}_j}{\sum_{i=1}^{i=N}\mathcal{S}_i} > 0.9)$
F :=Sort features based on $\mathcal{S}_i$
F' := Z first elements of F.
for $F_i \in F'$ **do**
  $W_{F_i} = \sum_{j=1}^{j=H} W_2^{(i,j)}$
end
F" =Sort features based on W_F
$\mathcal{K}$ =Select K first features F"

subset of X such as $X' = \{F'_1, \cdots, F'_m\}\&(m \leq n-1)$ that can be used for any subsequent procedure such as clustering or classification. For this purpose, we learn an AE $\mathcal{A}$ on the available data, $\mathcal{F}$. As a result, by feeding the sample X to $\mathcal{A}$, the AE will reconstruct $X \in \mathcal{R}^{1\times n}$ with a minimum loss, i.e., $\mathcal{A}(X) = \tilde{X}$ where $||X - \tilde{X}||^2$ is close to 0.

AEs include two important components: (1) Encoder (W_1); and (2) Decoder (W_2), where $\tilde{X} = X \times W_1 \times W_2$ (Here, for simplicity activation function are ignored). Let $W_k^{(ij)}$ be the weight between the i^{th} neuron at layer $(k-1)^{th}$ to j^{th} neuron of the k^{th} layer. $\mathcal{X} = \{X_i \in \mathcal{R}^n\}^{i=1:N}$ define available set of unlabeled training data, on which $\mathcal{A}$ is trained.

Reconstruction Error (RE) measure: To find the importance of the i^{th} feature (i.e., F_i), $\mathcal{S}_i = ||\mathcal{A}^i(X) - X||^2$ is computed. Hence, a low value of $\mathcal{S}_i$ would mean ignoring this feature will not harm the recostruction of the original data. $\mathcal{A}^i$ is a modified version of $\mathcal{A}$, in which $W_1^{(i,:)}$ is set to zero.

For robustness against noisy samples, $\mathcal{S}_i$ is calculated based on averaging on all training data:

$$\mathcal{S}_i = \frac{1}{N}\sum ||\mathcal{A}^i(X_j) - X||^2{}_{j=1}^{j=N}, \qquad (1)$$

where N is the number of training samples. Briefly, it can be said that F_i is a more important feature compared to F_j if $\mathcal{S}_i > \mathcal{S}_j$. As a result, if we sort $\{\mathcal{S}_1, \cdots, \mathcal{S}_n\}$ in a descending order, and replace $\mathcal{S}_i$ with its equivalent feature (i.e., F_i), we will get the ranked features. Now, based on application, the first Z features can be selected and the remaining features may be ignored. A simple heuristic way to automatically select Z is to set it large enough so that the model leads to the least error.

If we consider the summation of all REs with the absence one feature at a time (i.e., $\sum_{i=1}^{i=N}\mathcal{S}_i$), we can compute the importance of each single feature, F_j, by looking at how much from that total RE is decreased in that feature is selected, which

is $\frac{\mathcal{S}_j}{\sum_{i=1}^{i=N}\mathcal{S}_i}$ percent. The value is denoted as the Importance Factor (IF) of the j^{th} features. As a result, Z can be set based on a threshold on this total RE criteria, i.e.,

$$\arg\min_{Z}(1-\frac{\sum_{j=1}^{j=Z}\mathcal{S}_j}{\sum_{i=1}^{i=N}\mathcal{S}_i} > \tau). \quad (2)$$

For example, to preserve 90% of information, τ must be set to 0.9. Note, Principal Component Analysis (PCA) also shares the same concept of maximizing the variance, but it involves feature transformation and obtains a set of transformed features rather than a subset of the original features.

Feature Contribution (FC) measure: The RE measure leads to remarkable performance for feature selection and comparable with the state-of-the-art. However, this measure may fail in selecting good features when there are many features with high correlations with each other. Supposes we have two features, F_i and F_j, that are highly correlated, but they are both also very informative features. By ignoring the feature F_i in the reconstruction process of X, F_j will efficiently retrieve the missed information (as they are highly correlated), and vice verse. As a result, both $\mathcal{S}_i$ and $\mathcal{S}_j$ will have low values, and consequently low ranks for being selected. To overcome this weakness, we introduce a second measure that re-orders the Z selected features in previous step. To introduce this measure, we use the latent representation of the AE (the bottleneck). Intuitively, features that contribute more in constructing this representation can simply be considered as important features.

To this end, a new AE, $\mathcal{A}'$, is trained on Z selected features from the previous step. Let $W_{F_i} = \sum_{j=1}^{j=H} W_2^{(i,j)}$ define the sum of output weights of the input neurons of F_i in the trained $\mathcal{A}'$; with H as the size of the hidden layer. This value can be a good indication for determining the importance of the selected features. Sorting the features based on W_{Fi} in a descending order can identify the contribution of the features in building the latent representation.

In summary, the RE measure ranks features based on how bad the model will operate in absence of that feature, whereas FC measure ranks them based on their contribution in building the latent representation. Therefore, the final set of selected features by our method, first incorporates RE to select Z features, and from these Z ones, the top K features based on FC are selected. The pseudo-code of the proposed method is presented in Algorithm 1.

Ranking Image Patches. In many vision-related tasks, finding informative parts of samples, especially images, is an important task. We generalize our proposed method for this task, by replacing single features with image patches. Specifically, for each $\mathcal{B} \in \mathcal{R}^{h1\times h2}$, at one step, all $h1 \times h2$ pixels of $\mathcal{B}$ are deactivated, and RE and FC measures are calculated. For RE, $\mathcal{S}'_{\mathcal{B}_i}$ is calculated based on Eq. (3). Let $\mathcal{A}^{\mathcal{B}_i}$ be a version of $\mathcal{A}$ whose weights connected to pixels of the image patch $\mathcal{B}_i$ are set to zero.

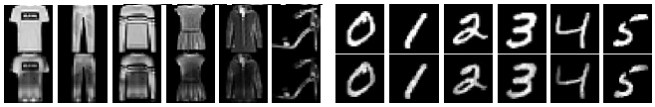

Fig. 3. Visualization of the feature importance in Fashion-MNIST (left) and MNIST (right) datasets.

Fig. 4. Selected important pixels (features) and patches on Fashion-MNIST dataset. *Two first rows*: From left to right, 50 to 750 pixels with a step-size of 50. *Two last rows*: From left to right, 5 to 75 patches of size 5×5 and step-size of 5. *Note*:Brighter pixels show higher ranks.

$$\mathcal{S}'_{\mathcal{B}_i} = \frac{1}{N}\sum_{j=1}^{j=N} ||\mathcal{A}^{\mathcal{B}_i}(X_j) - X||^2. \quad (3)$$

Hence, $\mathcal{B}_i$ is more important than $\mathcal{B}_j$, if $\mathcal{S}'_{\mathcal{B}_i} > \mathcal{S}'_{\mathcal{B}_j}$. Z top blocks are selected based on this measure. After that, the selected blocks are re-ordered based on their contribution on constructing the latent representation (of $\mathcal{A}'$ previously learned on these data). Contribution can be calculated as:

$$W_{\mathcal{B}_i} = \sum_{k\in\mathcal{B}_i}\sum_{j=1}^{j=H} W_2^{(k,j)}. \quad (4)$$

2.2. Training the AE

To learn the AE model, similar to previous works [13], the training process learns W_1 and W_2 using gradient descent. Then, $W_1 \times X$ can be used as the latent representation of the data. Suppose we have N training samples with n dimensions, i.e., $X_i \in \mathcal{R}^n, i \in \{1,\ldots,N\}$. The auto-encoder minimizes Eq. (5) by reconstructing the raw data:

$$L = \frac{1}{m}\sum_{i=1}^{m}|X_i - W_2\delta(W_1X_i + b_1) + b_2|^2 + \sum_{i=1}^{n}\sum_{j=1}^{s}(W_{ji}^2), \quad (5)$$

where s is the number of nodes in the hidden layer of auto-encoder, $W_1 \in \mathbb{R}^{s\cdot n}$ and $W_2 \in \mathbb{R}^{n\cdot s}$ are the weight matrices, which map the input layer nodes to hidden layer nodes and the hidden layer nodes to the output layer nodes, respectively. W_{ji} is the weight between the j^{th} hidden layer node and the i^{th} output layer node, and δ is equal to sigmoid function. Furthermore, b_1 and b_2 are the biases for the output layer and the hidden layer, respectively.

3. EXPERIMENTAL RESULT

We evaluate our proposed method on several standard benchmarks for classification and compare with the state-of-the-art feature selection methods. To showcase the performance of the method, we apply our method to find interest points (pixels) on MNIST [16] and Fashion MNIST [17] as two standard and

Table 1. Performance of different feature selection methods. Two methods with top performance in each of measure are colored, best method is blue and second best in red. If two top methods are equal, both are typeset in blue.

Dataset Algorithm	PCMAC	Madelon	USPS
CFS	(0.5000,0.68 ± 0.0492,0.7500)	(0.4500,0.52 ± 0.0246,0.5846)	(0.1920,0.93 ± 0.1116,0.9790)
LLCFS	(0.4974,0.65 ± 0.0470,0.7062)	(0.4558,0.52 ± 0.0173,0.5942)	(0.1699,0.94 ± 0.0972,0.9790)
UDFS	(0.4974,0.67 ± 0.0452,0.7139)	(0.5192,0.61 ± 0.0801,0.9115)	(0.1257,0.84 ± 0.1335,0.9752)
Relief	(0.5052,0.66 ± 0.0470,0.7294)	(0.5019,0.62 ± 0.0784,0.9096)	(0.1656,0.93 ± 0.1045,0.9795)
FSV	(0.5052,0.64 ± 0.0600,0.7268)	(0.5077,0.60 ± 0.0765,0.8865)	(0.1909,0.94 ± 0.0973,0.9806)
mRMR	(—-)	(0.4500,0.50 ± 0.0183,0.5442)	(0.2406,0.92 0.0821,0.9773)
InfFS	(0.5052,0.64 ± 0.0600,0.7268)	(0.4558,0.56 ± 0.0283,0.6288)	(0.1370,0.81 ± 0.1926,0.9768)
Ours	(0.5052,0.66 ± 0.0391,0.7662)	(0.5077,0.61 ± 0.772,0.9115)	(0.1931,0.94 ± 0.1024,0.9806)

popular datasets. Also, we show that the proposed method can efficiently rank the features on (PCMAC[1], madelon[2], USPS[3], and Semeion [18]).

Compared Methods. Performance of proposed method is compared with 7 state-of-the-art methods, including Correlation-based Feature Selection (CFS) [19], Local Learning based Clustering Feature Selection (LLCFS) [20], Unsupervised Discriminative Feature Selection (UDFS) [21], Relief [22], Feature selection via concave minimization and support vector machines (FSV) [23], minimal Redundancy Maximal Relevance (mRMR) [24], and Infinite Feature Selection (InfFS) [25].

Experiment Setup. Our implementations are done in MATLAB and we use the popular Feature Selection Library (FSLib)[4] for testing the compared methods. For all experiments, we divided dataset into two categories: Training data and test category. 80% of each class are randomly selected as the training data and the remainder for testing. Hidden layer size of is experimentally selected.

Results and Comparison. We compare the performance of our method in comparison of state-of-the-art feature selection methods, using K-Nearest neighbour as the classifier for all of them, for fair comparisons.

Table 1 compares the performance of feature selection algorithms for different datasets. The values presented in each cell are Minimum, Mean ± STD, and Maximum value of the accuracies obtained in all settings of the number of selected features (from 1 to #features). The proposed method is consistently among the best methods in all these cases, which shows the generality of the proposed method. In Table 1, the best method is typeset in blue color and the second best in red. If the two top model were equally good, both are colored blue. As can be seen in all three datasets, in most of cases, our method is the best or the second best method, while the performance of all methods are not stable.

To qualitatively evaluate the feasibility of our method, we rank the importance of the image pixels in the Fashion-MNIST dataset, by reshaping the pixels into feature vectors. Fig. 3 shows one sample from each class with its pixels ranked. More important pixels are brighter in the figure. As can be seen, the result are inline with the important parts of the objects and specific parts of the objects are selected with higher confidence. It can be observed that mainly general parts of the object (which are shared between objects of different classes) are dark, but those that enable distinguishing between objects are brighter. Furthermore, two first rows of Fig. 4 shows images from all classes gradually being completed. At first 50 important pixels are shown, after that in every step next 50 important pixels are added. As can be seen, images are completed in order of specific to general part of each of object in images. In a separate test, we ran the same experiments but this time on image patches of size 5×5 instead of pixels. The ranked images patches are visualized in two last rows of Fig. 4. Similar to two first rows, shows how images are gradually completed with respect to importance of their blocks. Fig. 4 confirm that proposed method can efficiently detect the informative pixels and image patches.

4. CONCLUSION

In this paper an efficient method based on AutoEncoders was introduced for feature selection (and also for discovering the informative image patches). Our method is presented based on (1) analyzing of reconstruction error of training samples in absence of a feature, i.e., introducing the RE measure, and (2) the amount of the contribution for each feature to reconstruct the original input, i.e., introducing the FC measure. Results confirm that our method can be used as a reliable feature selection method on different datasets.

5. ACKNOWLEDGEMENT

This research was in part supported by a grant from IPM (No. CS1396-5-01).

[1] http://qwone.com/ jason/20Newsgroups/
[2] http://clopinet.com/isabelle/Projects/NIPS2003/
[3] http://www.cad.zju.edu.cn/home/dengcai/Data/MLData.html
[4] https://it.mathworks.com/matlabcentral/fileexchange/56937-feature-selection-library

6. REFERENCES

[1] Ehsan Adeli, Guorong Wu, Behrouz Saghafi, Le An, Feng Shi, and Dinggang Shen, "Kernel-based joint feature selection and max-margin classification for early diagnosis of parkinsons disease," *Scientific reports*, vol. 7, pp. 41069, 2017.

[2] Richard O Duda, Peter E Hart, David G Stork, et al., "Pattern classification. 2nd," *Edition. New York*, p. 55, 2001.

[3] Feiping Nie, Shiming Xiang, Yangqing Jia, Changshui Zhang, and Shuicheng Yan, "Trace ratio criterion for feature selection.," in *AAAI*, 2008, vol. 2, pp. 671–676.

[4] Zheng Zhao, Lei Wang, Huan Liu, et al., "Efficient spectral feature selection with minimum redundancy.," in *AAAI*, 2010, pp. 673–678.

[5] Jiliang Tang, Salem Alelyani, and Huan Liu, "Feature selection for classification: A review," *Data Classification: Algorithms and Applications*, p. 37, 2014.

[6] Lior Wolf and Amnon Shashua, "Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach," *JMLR*, vol. 6, no. Nov, pp. 1855–1887, 2005.

[7] Xiaofei He, Deng Cai, and Partha Niyogi, "Laplacian score for feature selection," in *NIPS*, 2006, pp. 507–514.

[8] Christos Boutsidis, Petros Drineas, and Michael W Mahoney, "Unsupervised feature selection for the k-means clustering problem," in *NIPS*, 2009, pp. 153–161.

[9] Ehsan Adeli, Kim-Han Thung, Le An, Guorong Wu, Feng Shi, Tao Wang, and Dinggang Shen, "Semi-supervised discriminative classification robust to sample-outliers and feature-noises," *IEEE TPAMI*, , no. 1, pp. 1–1, 2018.

[10] Deng Cai, Chiyuan Zhang, and Xiaofei He, "Unsupervised feature selection for multi-cluster data," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 333–342.

[11] Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, Hanqing Lu, et al., "Unsupervised feature selection using nonnegative spectral analysis.," in *AAAI*, 2012, vol. 2, pp. 1026–1032.

[12] Mingjie Qian and Chengxiang Zhai, "Robust unsupervised feature selection.," in *IJCAI*, 2013, pp. 1621–1627.

[13] Geoffrey E Hinton and Ruslan R Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] Arnaz Malhi and Robert X Gao, "Pca-based feature selection scheme for machine defect classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 6, pp. 1517–1525, 2004.

[15] Ehsan Adeli, Dongjin Kwon, Qingyu Zhao, Adolf Pfefferbaum, Natalie M Zahr, Edith V Sullivan, and Kilian M Pohl, "Chained regularization for identifying brain patterns specific to hiv infection," *NeuroImage*, vol. 183, pp. 425–437, 2018.

[16] Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne E Hubbard, and Lawrence D Jackel, "Handwritten digit recognition with a back-propagation network," in *NIPS*, 1990, pp. 396–404.

[17] Han Xiao, Kashif Rasul, and Roland Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[18] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu, "Feature selection: A data perspective," *ACM Computing Surveys (CSUR)*, vol. 50, no. 6, pp. 94, 2017.

[19] Mark A Hall and Lloyd A Smith, "Feature selection for machine learning: comparing a correlation-based filter approach to the wrapper.," in *FLAIRS conference*, 1999, vol. 1999, pp. 235–239.

[20] Hong Zeng and Yiu-ming Cheung, "Feature selection and kernel learning for local learning-based clustering," *TPAMI*, vol. 33, no. 8, pp. 1532–1547, 2011.

[21] Yi Yang, Heng Tao Shen, Zhigang Ma, Zi Huang, and Xiaofang Zhou, "l2, 1-norm regularized discriminative feature selection for unsupervised learning," in *IJCAI proceedings-international joint conference on artificial intelligence*, 2011, vol. 22, p. 1589.

[22] Kenji Kira and Larry A Rendell, "The feature selection problem: Traditional methods and a new algorithm," in *Aaai*, 1992, vol. 2, pp. 129–134.

[23] Paul S Bradley and Olvi L Mangasarian, "Feature selection via concave minimization and support vector machines.," in *ICML*, 1998, vol. 98, pp. 82–90.

[24] Hanchuan Peng, Fuhui Long, and Chris Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[25] Giorgio Roffo, Simone Melzi, and Marco Cristani, "Infinite feature selection," in *CVPR*, 2015, pp. 4202–4210.