**Ministry of Science and Higher education**
**of the Russian Federation**
**ITMO University**

Faculty of Digital Transformations

Educational program Big Data and Machine Learning

Subject area (major) 01.04.02. Applied mathematics and informatics

REPORT

# Feature selection algorithms

Student: Oleg Taratukhin C42111c

Supervisor: Sergey Muravyov, ITMO University

Date 10.01.2021

St. Petersburg

2021

# Table of Contents

# Introduction

Clustering is classical and fundamental task in machine learning. However, it is relatively less studied, there are more things to explore then in some other well studied areas such as supervised learning. Importance of unsupervised and semi-supervised learning is on the rise as datasets are getting larger and more complex. At the same time modeling data also reflects this trend in datasets.

Complex datasets contain a lot of features, some of them are noisy, redundant, weak or contain essentially the same information as in other features combined that can be extracted by the model. Large number of features also hurts interpretability of the model thus limiting its use in practice. Clustering is especially sensitive to interpretability of the model, as clustering results are often used as a foundation to other applications. Feature selection for machine learning in general can increase model interpretability, decrease model size and in some cases even improve model performance in terms of quality.

Quality for clustering however is a vague concept, there are a number of quality measures, they produce different results and no single algorithm usually archived optimal scores in terms of multiple different measures. This means that choice of exact quality measure is an important aspect in autoML pipelines.

Feature selection process can eliminate redundant features to make more useful models of studied process, make models smaller, faster, more robust and more interpretable. Feature selection for clustering is more difficult since target variable information is not known from value of specific feature in data but can only be obtained from overall dataset structure, which can be costly to evaluate. Special consideration regarding algorithm complexity should be taken into account to be able to work with large datasets.

## Formalization

Up to this moment commonly agreed mathematically correct formalization of clustering in general is not known[1]. Not formally speaking, data clustering aims to perform grouping of similar objects, so that similar objects belong to the same group and different objects belong to different groups.

Here I will formally define clustering as follows. Let X be data sample with N objects, each object is represented as vector in F-dimensional space:

$$X = \{x_1, x_2, \dots, x_N\} \in \mathbb{R}^F$$

Clustering result is than a set of non-intersecting groups C:

$$C = \{c_1, c_2, \dots, c_K\}, \qquad \bigcup_{c_k \in C} c_k = X, c_k \cap c_l = \emptyset, k \neq l$$

There exists a number of internal and external clustering scores. Some of them are not measures in mathematical point of view, but rather scores. We will focus on the following ones: Dunn measure, Davies-Bouldin measure, Silhouette, Calinski-Harabasz, CS measure, modified Davies-Bouldin (DB*) measure, Symmetrical index measure, COP index, SV index, OS index and generalized Dunn measures. The exact formulas are not given for conciseness.

Apart from internal clustering measures there exists a number of external measures, some of them are purity, Rand index, F-measure, Jaccard index, Dice index, Fowlkes-Mallows index, mutual information score,

Due to its unsupervised nature good solution cannot be easily found, the process can be slow and costly. Clustering algorithms can be hard or soft. Hard clustering assigns every object to one specific group, on the other hand soft clustering assigns each data point to each group with some likelihood.

Feature selection can be formally defined as follows. Let d represent the desired number of features in the selected subset X from original set of Y features. Let J(X) represent quality of the selected subset. Then the

[1] Henning, C. What are true clusters? // Pattern Recognition Letters, 2015. Vol. 64. P. 53-62

problem of feature selection is to find a subset $X \subseteq Y$ such that $|X| = d$ and:

$$J(X) = \max_{Z \subseteq Y, |Z| = d} J(Z)$$

The number of possibilities grows exponentially, making exhaustive search impractical. Furthermore, it has been shown that any not exhaustive approach is not guaranteed to produce optimal subset and the ordering of error probabilities of each feature subset is possible (Cover & Compenhout, 1977).

Feature selection procedure can be defined as reducing input features when developing a predictive model. This process can be performed for both supervised and unsupervised cases. In supervised case it can be further specified as selection of the most relevant features that contribute most to the output value. However, in our case of unsupervised application no target variable data is available hence some supervised approaches will not be applicable.

Feature selection can be seen as a search of optimal initial feature subset with evaluation measure which does scoring of given subsets. The exact solution can be obtained by testing every single feature subset and selecting the best one according to the given quality measure. This approach is impractical for any real-world dataset as search space grows exponentially with increase of dimensionality. Feature selection can be viewed as binary optimization problem, then search space is shaped as a hypercube.

## General overview

Feature selection for unsupervised learning is gaining more attention in the literature lately. Also, feature selection for supervised learning is well studied and some approaches can be reused to create analogs for unsupervised case.

Feature selection methods from a taxonomic point of view can be classified into filter methods, wrapper methods, embedded methods and hybrid approaches.

### Wrapper feature selection methods

Wrapper methods are algorithm specific and use model to score feature subsets. Each new subset is used to train the model, which is used to evaluate out-of-sample error of that model with the feature subset, which gives score to the subset itself. Wrapper methods are very computationally intensive, certain budget and early stopping criteria should be used to use models in real-world applications. Apart from evaluating certain heuristic of feature values is it possible to optimize subset search space instead. The most known approaches are recursive feature adding, recursive feature elimination and their combinations.

## Filter feature selection methods

Filter methods are completely model agnostic and rely solely on different proxy measures of variable values to determine which features are most useful. Most of the known filter methods are not really applicable here as they require target variable, the most known and used methods are information gain, chi-square test for categorical variables and Fisher's score. Filter methods are model agnostic and usually fast to compute. The lack of tuning means that feature sets produced by filter methods are more general than the set from a wrapper, usually giving lower prediction performance than a wrapper. Some filter methods provide feature ranking instead of fixed subset, so cut-off point can be chosen by cross-validation or other means. Filter methods are often used as preprocessing step in other algorithms, for instance with wrapper methods, allowing wrapper methods to use used on bigger datasets.

## Embedded feature selection methods

Embedded methods of feature selection perform selection during the model building. For example, Lasso regression uses L1 regularization that forces weights of less relevant features for linear model to be set to zero. One can say that any non-zero weight is assigned to feature that is 'selected' by embedded selection algorithm. More complicated algorithms such as FeaLect (Zare, 2013) use combinatorial analysis of regression coefficients of Lasso model. AEFS algorithm extends this approach to nonlinear scenarios with autoencoders (Kai, Yunhe, Chao, Chao, & Chao, 2018). These methods tend to be modestly computationally intensive, falling in between filter and wrapper methods.

## Subset search optimization

For subset selection exhaustive search is impractical, but the best result. Many algorithms use greedy hill climbing approach, which iteratively evaluates feature subset and then modifies the subset and repeats the process. Alternatively, algorithms may be based on targeted projection pursuit technique, which searches for low-dimensional data projection which gives high scores according to given quality measure.

Optimized search strategies include but are not limited to best first search, simulated annealing, genetic algorithm, greedy forward selection and greedy backward elimination combinations, particle swarm optimization, targeted projection pursuit, scatter search and variable neighborhood search.

# Algorithms

Apart from general approaches to clustering and general feature selection algorithms more specific algorithms were developed targeting specifically feature selection for clustering.

Interesting solution is proposed in (Deep Feature Selection using Teacher-Student Network). The idea is to train larger teacher model to learn dimensionality reduction function (or use existing manifold-learning approach) and then use smaller student network to distill larger network and perform feature selection in low dimensional space.

Another interesting solution is described in (I Abdelaziz Hammouri; Majdi Mafarja; Mohammed Azmi Al-Betar; Mohammed Awadallah; Iyad Abu-Doush , 2020). In this article authors propose to use improved Dragonfly algorithm for feature selection.

Many proposed novel approaches introduced in the literature do not have publicly available code with experiments, making results from articles not easily reproducible. For benchmarks I will use well known datasets: Breastcancer, BreastEW, WineEW, HeartEW.

Although the research topic is focused on unsupervised learning, for evaluation we will use supervised learning datasets and evaluate average selected features and accuracy of the selected subset based on results of multiple evaluation with random initialization of several clustering algorithms. The reason is that different feature sets can produce different clustering which are reasonable given feature set. To compare results the following algorithms will be compared: QBDA, SBDA, BSO, QBSO-FS (Bee Swarm Optimization for Feature Selection) (Sadeg, et al., 2019).

| Dataset | Metric | QBSO-FS | BSO-FS | QBDA | SBDA |
|---|---|---|---|---|---|
| Iris | Avg. accuracy | 0.973 | 0.973 | 0.971 | 0.972 |
| | Avg. # features | 2.27 | 3.87 | 2.57 | 3.27 |
| Breastcancer | Avg. accuracy | 0.962 | 0.962 | 0.983 | 0.993 |
| | Avg. # features | 3.43 | 3.73 | 3.43 | 3.00 |
| Wine | Avg. accuracy | 0.955 | 0.956 | 0.955 | 0.956 |
| | Avg. # features | 6.83 | 6.73 | 4.63 | 3.33 |
| Ionoshpere | Avg. accuracy | 0.958 | 0.956 | 0.923 | 0.984 |
| | Avg. # features | 11.27 | 11.50 | 11.87 | 12.03 |

# Conclusion

It is difficult to assess performance of unsupervised learning algorithms with feature selection, as benefits of using feature selection comes in different forms for different purposes. Interpretability is a complex characteristic that can hardly be measured and compared. The performance of the novel approaches is not easy to reproduce due to known reproducibility crisis in modern research. I used only results of novel approaches that can be verified by running publicly available official implementations.

From collected results it seems that SBDA (Sinusoidal Binary Dragonfly Algorithm) select fewer on average features in datasets under evaluation. However, on other datasets the results could be different, so this conclusion only holds on abovementioned datasets. Many novel approaches couldn’t be included in the report, as they do not have reproducible results.

# Bibliography

Zare, H. (2013). Scoring relevancy of features based on combinatorial analysis of Lasso with application to lymphoma diagnosis. *BMC Genomics*, 14.

Meiri, R., & Zahavi, J. (2006). Using simulated annealing to optimize the feature selection problem in marketing applications. *uropean Journal of Operational Research*, 842–858.

Kai, H., Yunhe, W., Chao, Z., Chao, L., & Chao, X. (2018). Autoencoder inspired unsupervised feature selection. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Ali, M., Vahid, P., Mehran, S., & Hamid, S. (2019). Deep Feature Selection using Teacher-Student Network. *arXiv (preprint)*.

Lei, Y., & Liu, H. (2003). Feature selection for high dimensional data: a fast correlation-based filter solution. *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, 856–863.

Roffo, G., Melzi, S., & Cristani, M. (2015). Infinite Feature Selection. *EEE International Conference on Computer Vision (ICCV)*, 4202–4210.

Roffo, G., Melzi, S., & Cristani, M. (2016). Feature selection via Eigenvector Centrality. *NFMCP*.

Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 273-324.

I Abdelaziz Hammouri; Majdi Mafarja; Mohammed Azmi Al-Betar; Mohammed Awadallah; Iyad Abu-Doush . (2020). An improved Dragonfly Algorithm for feature selection. *Knowledge-Based Systems*.

Cover, T., & Compenhout, M. (1977). On possible orderings in the measurement selection problem. *IEEE Transactions on Systems, Man, and Cybernetics*, 657-661.

Sadeg, S., Hamdad, L., Remache, A. R., Karech, M. N., Benatchba, K., & Habbas, Z. (2019). QBSO-FS: A reinforcement Learning Based Bee Swarm Optimization Metaheuristic for Feature Selection. *IWANN: Advances in Computational Intelligence*, 785-796.