



## PYALLFFS: AN OPEN-SOURCE LIBRARY FOR ALL FILTER FEATURE SELECTION METHODS

Tohid YOUSEFI<sup>1\*</sup>, Özlem VARLIKLAR<sup>1</sup>

<sup>1</sup>Dokuz Eylul University, Faculty of Engineering, Department of Computer Engineering, 35160, Izmir, Türkiye


**Abstract:** Feature selection is a significant data mining and machine learning technique that enhances model performance by identifying important features within a dataset, reducing the risk of overfitting while aiding the model in making faster and more accurate predictions. Pyallffs is a Python library developed to optimize the feature selection process, offering rich content and low dependency requirements. With 19 different filtering methods, pyallffs assists in analyzing dataset features to determine the most relevant ones. Users can apply custom filtering methods to their datasets using pyallffs, thereby achieving faster and more effective results in data analytics and machine learning projects. The source codes, supplementary materials, and guidance is publicly available on GitHub: <https://github.com/tohid-yousefi/pyallffs>.

**Keywords:** Feature selection, Filter methods, Feature selection library, Open-source library, Python software

\*Sorumlu yazar (Corresponding author): Dokuz Eylul University, Faculty of Engineering, Department of Computer Engineering, 35160, Izmir, Türkiye

E mail: [tohid.yousefi@ogr.deu.edu.tr](mailto:tohid.yousefi@ogr.deu.edu.tr) (T. YOUSEFI)

Tohid YOUSEFI  <https://orcid.org/0000-0003-4288-8194>

Özlem VARLIKLAR  <https://orcid.org/0000-0001-6415-0698>

Received: April 09, 2024

Accepted: September 03, 2024

Published: September 15, 2024

Cite as: Yousefi T, Varlıklar Ö. 2024. Pyallffs: An open-source library for all filter feature selection methods. BSJ Eng Sci, 7(5): 971-981.

### 1. Introduction

Optimizing model performance and efficiency in machine learning relies heavily on feature selection, where the most relevant attributes are chosen for model building. This process streamlines the model, reducing computational complexity and enhancing its ability to generalize to new data by focusing solely on essential features. However, in high-dimensional datasets, the growing number of features can pose challenges, such as the curse of dimensionality (Miao and Niu, 2016; Shardlow, 2016). This phenomenon can lead to overly complex models, increasing the risk of overfitting and diminishing their generalization capabilities. Thus, effective feature selection techniques are vital to address these challenges and ensure the reliability and robustness of machine learning models, particularly in the face of increasing data dimensionality (Kalousis et al., 2007).

Feature selection plays a fundamental role in the realms of machine learning and data analytics, aiming to enhance model performance and diminish unnecessary noise within datasets. This process is carried out through various methods, including filter (Chandrashekar and Sahin, 2014), wrapper (Kohavi and John, 1997), embedded (Zheng and Casari, 2018), ensemble (Opitz and Maclin, 1999), and hybrid methods (Kabir et al., 2010). In this study, we will specifically delve into filter methods. Filter methods assist in identifying the most significant features within a dataset by analyzing their relationships and impact on the target variable. These methods examine the independence between features and select the most suitable ones, thereby improving model

performance while mitigating the risk of overfitting. Consequently, filter methods play a pivotal role in the feature selection process (Yousefi and Varlıklar, 2024).

Filter methods are integral to feature selection in machine learning, focusing on identifying the most relevant attributes within a dataset based on intrinsic characteristics (Yousefi and Aktaş, 2024). The pyallffs library, developed for this purpose, offers a comprehensive array of filtering techniques, facilitating seamless integration and exploration of various methods. With pyallffs, researchers and practitioners can efficiently pinpoint the most influential features within datasets, enhancing the accuracy and robustness of machine learning models.

This paper makes the following key contributions:

1. Novel Integration of Filter Methods: We introduce the *pyallffs* library, which consolidates multiple filtering techniques into a single, accessible platform, enabling researchers to efficiently apply and compare different methods.
2. Empirical Evaluation: Through rigorous testing on diverse datasets, we demonstrate the effectiveness of filter methods in improving model accuracy and robustness, particularly in high-dimensional data environments.
3. Comprehensive Analysis: The paper provides a detailed examination of the impact of various filter methods on model performance, offering valuable insights into their strengths and limitations.

The structure of this paper is organized as follows: The *Materials and Methods* section details the feature



selection process, with a particular focus on filter feature selection methods. It also describes the datasets used in the study and provides information on the implementation and usage of the *pyallfs* library. Following this, the *Results* section presents the outcomes of our experiments, highlighting the effectiveness of the selected methods. Finally, the *Conclusion and Future Work* section summarizes the key findings and discusses potential directions for further research.

**2.1. Feature Selection**

Feature selection is an important part of machine learning and data analysis. It helps to pick out the most useful features while leaving out the ones that don't matter much. This process, shown in Figure 1, makes models better by focusing on the most important features and making them easier to understand. It also helps to prevent problems like overfitting and makes models more accurate. By choosing features carefully, people who work with data can make better models and understand their data better. So, feature selection is a big help in analyzing data and making good decisions (Yousefi and Varliklar, 2024).

**2. Materials and Methods**



Figure 1. Feature selection process.

**2.2. Filter Methods**

The filter feature selection method (Chandrashekar and Sahin, 2014) is a technique used to identify the most important features in a dataset. This method aims to select the most suitable features by analyzing relationships between features and how they affect the target variable. As depicted in Figure 2, filter methods help improve model performance by reducing the dataset's dimensionality. These methods typically use metrics that evaluate the importance of features, such as statistical measures. Filter methods reduce model complexity, prevent overfitting, and enhance the model's generalization ability. Therefore, the filter feature selection method plays a significant role in data analytics and machine learning projects (Yousefi and Varliklar, 2024).

Filter methods are paramount in feature selection, serving to enhance model performance by identifying relevant attributes within datasets. However, given the plethora of techniques utilized within filter methods and the absence of a unified tool for their simultaneous application, a library addressing this need has been developed in this study. This library, namely *pyallfs*, is a versatile toolkit boasting 19 distinct filter feature selection methods. With *pyallfs*, researchers and practitioners can effortlessly navigate through various filtering techniques, as depicted in Figure 3, facilitating efficient and informed feature selection processes tailored to their specific dataset needs.

**2.2.1. Fisher score**

The Fisher Score (Gu et al., 2012), employed in binary classification scenarios, assesses the discriminative capability of a feature across two classes. It quantifies this by computing the ratio of squared mean differences

between feature values for each class to the sum of variances within each class. A higher Fisher Score suggests a more pronounced discrimination between classes, rendering the feature more significant. The Fisher Score formula can be expressed as in equation 1:

$$Fisher\_score(f_i) = \frac{\sum_{j=1}^c n_j (\mu_{i,j} - \mu_i)^2}{\sum_{j=1}^c n_j \sigma(i,j)^2} \tag{1}$$

Here,  $n_j$ ,  $\mu_i$ ,  $\mu_{i,j}$ , and  $\sigma(i,j)^2$  respectively denote the number of samples in class  $j$ , the mean value of feature  $f_i$ , the mean value of feature  $f_i$  for samples in class  $j$ , and the variance value of feature  $f_i$  for samples in class  $j$ . This feature selection method is commonly employed for binary classification purposes (Gu et al., 2012).

**2.2.2. T-score**

The fundamental concept behind the T-score (Carey and Delaney, 2010) is to assess whether a feature can statistically differentiate between the means of two classes by calculating the ratio between the mean difference and the variance of the two classes. Generally, the higher the t-score, the more significant the feature (Faulkner, 2005). The T-score relies on the t-value, which is among the most commonly used filter methods. As mentioned earlier, a relationship score is computed for each class using the sample size, mean, and standard deviation values of features, and features with high scores are added to the subset in the t-score method (Budak and Taşabat, 2016). The T-Score formula can be expressed as in equation 2 (Chandrashekar and Sahin, 2014):

$$TScore(f_i) = \frac{|\mu_1 - \mu_2|}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \tag{2}$$

Here,  $\mu_1$  and  $\mu_2$  are the mean feature values for samples

from the first and second classes, respectively, while  $\sigma_1^2$  and  $\sigma_2^2$  represent the corresponding standard deviation values, and  $n_1$  and  $n_2$  denote the number of samples from these two classes (Chandrashekar and Sahin, 2014).

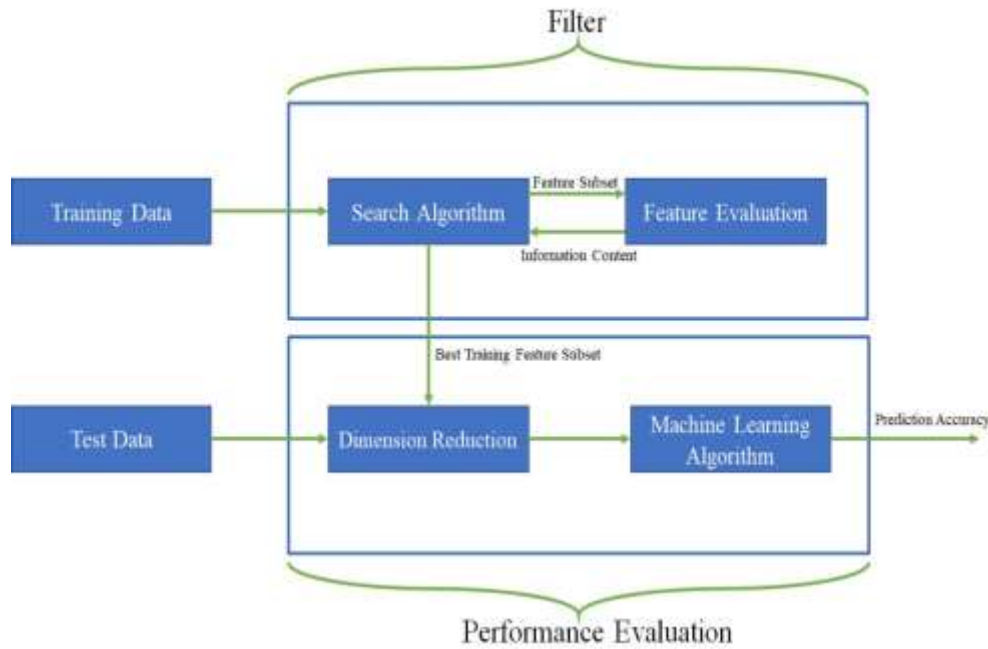


Figure 2. The general framework of filter method

Fisher Score	T-Score	Laplacian Feature Selection Score	
Information Gain	Gain Ratio	Symmetric Uncertainty Coefficient	
Relief Score	Relief-F Score	Absolute Pearson Correlation Coefficients	
Mutual Information	Euclidean Distance	Maximum Likelihood Feature Selection	
Welch's t-statistic	Chi-Squared	Least Squares Feature Selection	
Cramer's V test	Markov Blanket Filter	Kruskal-Wallis test	Laplacian

Figure 3. All filter feature selection methods.

2.2.3. Welch's T-statistic

Welch's t-statistic (Welch, 1947) is a technique used in feature selection to spot differences between groups of features. It calculates variations in group means, within-group variances, and sample sizes. Unlike the standard t-test, Welch's method is more reliable when group sizes and variances differ. It offers a flexible approach to evaluating how features stand out across groups by normalizing mean differences. The Welch's t-statistic formula can be expressed as in equation 3:

$$Welch'sTstatistic(f_i) = \frac{|\mu_1 - \mu_2|}{\sqrt{\frac{(\sigma_1^2)^2}{n_1} + \frac{(\sigma_2^2)^2}{n_2}}} \tag{3}$$

Here,  $\mu_1$  and  $\mu_2$  are the mean feature values for samples from the first and second classes, respectively, while  $\sigma_1^2$  and  $\sigma_2^2$  represent the corresponding variances values, and  $n_1$  and  $n_2$  denote the number of samples from these two classes (Delacre et al., 2017).

2.2.4. Chi-squared

The chi-square (Kass, 1980) statistic serves as a

technique for feature selection, primarily assessing the association between features and the target variable, particularly in classification tasks involving categorical features. It determines whether a feature is independent of the target variable by measuring the strength of their relationship. This test evaluates the independence hypothesis between two categorical variables through the ratio of squared differences between observed and expected frequencies, thereby gauging the existence of a relationship between them. The mathematical formula of chi-square statistic is as in equation 4 (Witten and Frank, 2002):

$$ChiSquared(f_i) = \sum \frac{(O_i - E_i)^2}{E_i} \quad (4)$$

Here,  $O_i$  stands for observed frequency, indicating the actual occurrence frequencies of category combinations within the dataset, while  $E_i$  represents expected frequency, depicting the frequencies expected under the assumption of independence between the two variables (Ugoni and Walker, 1995). The chi-square statistic calculates the sum of the squared differences between observed and expected frequencies for category combinations, serving as a measure of the relationship between the feature and the target variable. A high chi-square value indicates a robust relationship, whereas a low value suggests a weaker association between the two variables (Bryant and Satorra, 2012).

### 2.2.5. Information gain

Information gain, also known as Kullback-Leibler (Kullback and Leibler, 1951) divergence, is a measure of the entropy gained through operations performed on a dataset or random variable (Cover, 1999). Entropy represents the level of variation in data; lower entropy indicates less variation and stronger relationships. Higher information gain signifies greater importance of a feature. It operates independently across all features and is based on their information gain (Hall and Smith, 1998). The information gain of a feature denotes the difference between previous uncertainty and expected subsequent uncertainty. Information gain is highest for classes with equal probability, indicating lowest uncertainty. Shannon entropy is commonly used as a measure of uncertainty (Shannon, 1948). The mathematical formula of information gain is as in equation 5:

$$H(X) = - \sum_{x_i \in X} P(x_i) \log(P(x_i))$$

$$H(X|Y) = \sum_{y_i \in Y} P(y_i) \sum_{x_i \in X} P(x_i|y_i) \log(P(x_i|y_i)) \quad (5)$$

$$Information\ Gain(X, Y) = H(X) - H(X|Y)$$

Here, Information Gain ( $X, Y$ ) denotes the information gain between dataset  $X$  and feature  $Y$ .  $H(X)$  signifies the entropy of dataset  $X$ , which equals zero when the dataset is entirely homogeneous.  $H(X|Y)$  represents the conditional entropy of dataset  $X$  given feature  $Y$ . It assesses the homogeneity of the dataset following its division based on feature values.

### 2.2.6. Gain ratio

Gain Ratio is a feature selection criterion utilized to gauge the importance of features in the feature selection process. As a variant of Information Gain, it offers a normalized perspective, considering the intrinsic information content of a split. It assesses the homogeneity of subsets formed by feature splitting while also factoring in the number of values the feature can assume. This metric aims to address the bias towards features with a large number of unique values, which often exhibit higher Information Gain, by normalizing it with the split information (Witten et al., 2005). The Gain Ratio is typically calculated as in equation 6 (Novaković, 2016):

$$GainRatio(X) = \frac{Information\ Gain(X)}{Split\ Information(X)} \quad (6)$$

Here, Information Gain ( $X$ ) denotes the information gain attributed to feature  $X$ , while Split Information ( $X$ ) quantifies the information generated by splitting based on feature  $X$ . Split Information is determined by the cardinality of feature values; higher unique values lead to diminished Split Information, consequently elevating the Gain Ratio, thus ensuring a more equitable metric for feature selection across diverse feature sets (Novaković, 2016; Priyadarsini et al., 2011).

### 2.2.7. Symmetric uncertainty coefficient

The Symmetric Uncertainty Coefficient is a method devised to overcome the drawbacks of information gain by dividing the entropies of  $X$  and  $Y$  (Dash and Liu, 2003). It assesses the suitability of features in relation to the target class, with higher values indicating greater importance. Symmetric Uncertainty Coefficient can be calculated using the formula in equation 7 (Hernández-Torruco et al., 2014):

$$SUC(X, Y) = \frac{2 * Information\ Gain(X, Y)}{H(X) + H(Y)} \quad (7)$$

Here,  $SUC(X, Y)$  denotes the Symmetric Uncertainty Coefficient between variables  $X$  and  $Y$ , while  $Information\ Gain(X, Y)$  measures the information gain between them, with  $H(X)$  and  $H(Y)$  representing their respective entropies (Ali and Shahzad, 2012). Similar to gain ratio, this method also ranges between 0 and 1. When the symmetric uncertainty coefficient equals 1, it indicates that one feature completely predicts the other, whereas a value of 0 signifies no relationship between  $X$  and  $Y$  (Mani and Kalpana, 2016).

### 2.2.8. Relief score

Relief, proposed by Kira and Rendell (Kira and Rendell, 1992), is a widely used classic filter method for classification problems, functioning as a multivariate feature selection technique. It operates by randomly selecting instances from the data and then finding the nearest neighbors from the same and opposite classes, updating relevance scores for each feature based on these comparisons (Hall and Holmes, 2000). In other words, Relief measures the relevance of features by comparing the value of the current feature for instances classified as the same and different classes (Esmael et al., 2012). Its strengths lie in its independence from intuitive scans,

efficient operation in low-degree polynomial time, and applicability to binary and continuous data, and resilience to noisy data and feature interactions. However, it fails to distinguish among redundant features, thus potentially misleading the algorithm with a limited number of training instances (Lun Gaoa et al., 2013). Relief score can be calculated using the formula in equation 8 (Miao and Niu, 2016; Nilsson, 2007):

$$Relief\_Score = \frac{1}{2} \sum_{j=1}^l d(X(j, i) - X(NM(j), i)) - d(X(j, i) - X(NH(j), i)) \quad (8)$$

Here, NM(j) and NH(j) indicate the nearest data instances to  $x_j$  with the same class label and a different class label, respectively. Typically,  $d$  is set as the Euclidean distance metric (Miao and Niu, 2016; Nilsson, 2007).

### 2.2.9. Relief-F score

Relief-F (Kononenko, 1994) is an extension of the original Relief algorithm. Whereas the original Relief operates by randomly selecting an instance from the data and then finding the nearest neighbors from the same and opposite classes, Relief-F extends its capabilities to handle multi-class problems and offers increased robustness against missing and noisy data (Arauzo-Azofra et al., 2004; Urbanowicz et al., 2018). This method is universally applicable, has low error rates, accounts for feature interactions, and can capture local dependencies overlooked by other methods. The core idea of this approach is to select features capable of distinguishing examples originating from different classes (Kononenko, 1994; Vora and Yang, 2017). Relief-F calculates its score using the formula in equation 9 (Kononenko, 1994):

$$ReliefF(f_i) = \frac{1}{c} \sum_{j=1}^l \left( -\frac{1}{m_j} \sum_{x_r \in NH(j)} d(X(j, i) - X(r, i)) + \sum_{y \neq y_j} \frac{1}{h_{jy}} \frac{P(y)}{1 - P(y)} \sum_{x_r \in NM(j, y)} d(X(j, i) - X(r, i)) \right) \quad (9)$$

Here, NH(j) and NM(j, y) denote the nearest data instances to  $x_j$  with the same class and a different class, respectively, with sizes  $h_{jy}$  and  $m_j$ .  $P(y)$  represents the proportion of examples with class label  $y$  (Vora and Yang, 2017).

### 2.2.10. mRMR

The Minimum Redundancy Maximum Relevance (mRMR) score, proposed by Ding and Peng in 2005 (Ding and Peng, 2005), is a filter-based and supervised feature selection method. It selects features by minimizing redundancy among them while maximizing relevance, aiming to reduce feature redundancy and maximize feature relevance (Chandra and Gupta, 2011; Radovic et al., 2017). Additionally, this method utilizes the mutual information criterion as a fitness measure across different datasets (Ding and Peng, 2005; Bolón-Canedo et al., 2014). The mRMR score can be calculated using the formula in equation 10 (Ding and Peng, 2005):

$$w = \frac{1}{|S|^2} \sum_{i,j} c(i, j) \quad (10)$$

$$V_F = \frac{1}{|S|} \sum_{i \in S} F(i, h)$$

Here,  $S$  represents a set of features,  $|S|$  denotes the number of features in  $S$ ,  $c(i, j)$  represents the correlation between features  $i$  and  $j$ ,  $h$  is the target, and  $F(i, h)$  is the F-statistic. The Minimum Redundancy Maximum Relevance (mRMR) method is one of several feature selection techniques applicable in both classification and regression tasks. It has been observed to perform particularly well in high-dimensional datasets where the number of features is significantly larger than the number of samples (Ding and Peng, 2005; Peng and Fan, 2015).

### 2.2.11. Absolute Pearson correlation coefficients

Feature selection based on correlation involves assessing the connection between a feature and either the target variable or other features, indicating the strength of their relationship. The correlation coefficient, often measured using the Pearson correlation coefficient, quantifies this relationship between two variables. With values ranging from -1 to 1, a coefficient close to 1 signifies a positive linear relationship, while a coefficient near -1 suggests a negative linear relationship. Conversely, a coefficient close to 0 indicates no linear relationship between the variables. The Pearson correlation coefficient can be calculated using the formula in equation 11 (Sedgwick, 2012):

$$pearson\_correlation\_coefficients(r) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \quad (11)$$

Here,  $\bar{x}$  and  $\bar{y}$  represent the means, while  $\sigma_x$  and  $\sigma_y$  denote the standard deviations of  $x_i$  and  $y_i$ , respectively (Goswami and Chakrabarti, 2014).

### 2.2.12. Maximum likelihood feature selection

Maximum likelihood feature selection (Suzuki et al., 2008) is a multivariate and supervised feature selection method that prioritizes variables based on the measure of input-target dependency. Estimators utilize maximum likelihood mutual information to measure the dependency between input and target. This method is a density estimation-based mutual information estimator. The density ratio of this method is calculated using the formula in equation 12 (Ding and Peng, 2005; Suzuki et al., 2008; Suzuki Ding and Peng, 2009):

$$w(x, y) = \frac{P_{xy}(x, y)}{P_x(x)P_y(y)} \quad (12)$$

Here,  $P_{xy}(x, y)$  denotes the joint density of  $X$  and  $Y$ , while  $P_x(x)$  and  $P_y(y)$  represent the densities of  $X$  and  $Y$ , respectively. Maximum likelihood feature selection is a method employed for both classification and regression problems (Suzuki et al., 2008).

### 2.2.13. Least squares feature selection

Least Squares Feature Selection is a method utilized to enhance model accuracy or effectively explain the target variable by selecting features from a dataset. It employs a linear regression model to gauge feature importance,

utilizing the least squares method within this framework. Practically, it involves estimating coefficients in the linear regression model and evaluating their absolute magnitudes to determine feature importance. Features are prioritized or ranked based on the absolute values of their coefficients, providing insights into their significance in the model. Mathematically, this technique is represented by estimating coefficients in the linear regression model and examining their absolute values to ascertain feature importance. The least squares feature selection can be calculated using the formula in equation 13 (Xiang et al., 2012):

$$\hat{\beta} = (X^T X)^{-1} X^T y \quad (13)$$

Here,  $\hat{\beta}$  denotes the estimated parameter vector of the model,  $X$  represents the data matrix, and  $y$  is the vector of the target variable.

#### 2.2.14. Laplacian feature selection score

The Laplacian feature selection score (He et al., 2005) is fundamentally based on Laplacian eigenmaps (Belkin and Niyogi, 2001) and locality preservation forces (He and Niyogi, 2003); moreover, this method is a graph-based, unsupervised, and univariate feature selection algorithm that ranks features according to their locality preservation forces (Von Luxburg, 2007). In the Laplacian algorithm, features are evaluated independently; therefore, this algorithm cannot assess feature redundancy (Liu et al., 2010). The Laplacian score of a feature can be calculated using the formulas in equation 14 (He et al., 2005):

$$\begin{aligned} Lap(f_i) &= \frac{\tilde{f}_i' L f_i'}{\tilde{f}_i' D f_i'} \\ f_i &= f_i - \frac{f_i' D 1}{1' D 1} 1, \quad 1 = [1, 1, \dots, 1]' \\ D(i, j) &= \sum_{j=1}^n S(i, j) \\ S(i, j) &= \frac{e^{-\|x_i - x_j\|^2}}{t} \end{aligned} \quad (14)$$

$$Laplacian\_matrix(L) = D - S$$

It is well-known that constructing the Laplacian graph is computationally expensive, particularly when the number of features is high (He et al., 2005).

#### 2.2.15. Mutual information

Mutual Information was initially proposed by Shannon in 1948 (Shannon, 1948). This method is a univariate and supervised feature weighting technique. Moreover, it calculates the mutual information between each feature and the target class label, then ranks the features accordingly and selects the best ones. In other words, this method quantifies the amount of information that two random variables convey about each other. Additionally, it has a symmetric structure  $I(X; Y) = I(Y; X)$  and can detect nonlinear relationships between variables. Hence, it has become a very popular criterion (Battiti, 1994; François et al., 2007). The reason is that mutual information, unlike other methods, does not only handle linear dependencies (Doquire and Verleysen, 2011). The mutual information method has been successfully adopted in filter feature selection methods to assess both

the relevance of a subset of features in predicting the target variable and their redundancy with respect to other variables (Beraha et al., 2019). Mutual information can be calculated using the formula in equation 15 (Cover, 1999):

$$\begin{aligned} &Mutual\_Information(X, Y) \\ &= \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x) * p(y)} \end{aligned} \quad (15)$$

Here,  $X$  and  $Y$  represent two random features or variables, with  $p(x)$  and  $p(y)$  being the probability density functions and  $p(x, y)$  the joint probability density function (Kannan and Ramaraj, 2010; Vergara and Estévez, 2014). Mutual Information is a fundamental method for evaluating how much information is associated between two features. It is defined as the difference between the sum of marginal entropies and the joint entropy. For completely independent objects, mutual information is always zero (Singh et al., 2014). A prediction or classification model aims to reduce uncertainty in the output, the dependent variable. As mentioned above, it is a good criterion for assessing the relevance of a set of features, a simplified prediction model. Naturally, it measures the uncertainty of the output due to knowledge of the inputs (Rossi et al., 2006).

#### 2.2.16. Euclidean distance

The Euclidean Distance is a widely employed metric for gauging the similarity or dissimilarity between features, denoting the straight-line distance between two points within the feature space. This distance measure finds extensive application in various machine learning and data mining algorithms to quantify the distance or similarity between features (Suebsing and Hirasakolwong, 2009). The mathematical expression for the Euclidean Distance between two points is given by the formula in equation 16 (Ladha and Deepa, 2011):

$$Euclidean\_Distance(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (16)$$

Here,  $x$  and  $y$  are feature vectors,  $x_i$  and  $y_i$  represent the  $i$ -th components of vectors  $x$  and  $y$ , respectively, and  $n$  is the dimensionality of the feature vectors (i.e., the number of features). A smaller Euclidean Distance suggests similarity between two vectors, whereas a larger Euclidean Distance suggests dissimilarity. This metric is especially valuable for assessing the proximity or separation of clusters or data points within the feature space (Ladha and Deepa, 2011).

#### 2.2.17. Cramer's V test

The chi-square test is a widely recognized method employed to examine associations between variables, demonstrating efficacy in the domain of feature selection (Lu and Weng, 2007). However, its sensitivity to sample size is a well-documented limitation. To address this issue, researchers often turn to Cramer's V test, a prominent nominal technique used to quantify the strength of relationships between variables. Notably, Cramer's V is advantageous as it remains unaffected by sample size variations, making it particularly valuable in scenarios where statistical significance in chi-square

results may be attributed to large sample sizes rather than genuine associations between variables. As such, Cramer's V test serves as a reliable tool for assessing the degree of relationship between target and predictor variables. The calculation of Cramer's V value is determined by the formula in equation 17 (Martínez Casasnovas et al., 2008):

$$V = \sqrt{\frac{x^2}{N * (k - 1)}} \quad (17)$$

Here, Cramer's V value is derived from the chi-square ( $x^2$ ) statistic and is calculated based on the total number of observations ( $N$ ) and the number of categories ( $k$ ) in the features. This value ranges between 0 and 1, where a higher value signifies a stronger relationship between the categorical variables. In feature selection, features exhibiting higher Cramer's V values are generally deemed more significant and prioritized over others (Martínez Casasnovas et al., 2008).

### 2.2.18. Markov blanket filter

The concept of the Markov Blanket, originating from Pearl's seminal work in 1988 (Pearl, 1988), serves as a fundamental component in probabilistic graphical modeling. The Markov Blanket Filter, an essential tool for feature selection, identifies a subset of variables crucial for maintaining the conditional independence of a target variable within a probabilistic framework. For a variable  $X_i$ , its Markov blanket includes directly connected variables that influence or are influenced by  $X_i$ , encompassing both parents and children nodes within a graphical model. This blanket, defined as  $MB(X_i)$ , plays a pivotal role in Bayesian networks and graphical models by encapsulating the minimal set of variables needed to predict  $X_i$  given all others in the network (Tsamardinos et al., 2003; Tsamardinos et al., 2003). The mathematical definition of the Markov blanket for the variable  $X_i$ , is expressed as in equation 18:

$$MB(X_i) = Pa(X_i) \cup Ch(X_i) \cup Pa(Ch(X_i)) \quad (18)$$

Here, the Markov blanket  $MB(X_i)$  of  $X_i$  comprises its parents  $Pa(X_i)$ , its children  $Ch(X_i)$ , and the parents of its children  $Pa(Ch(X_i))$ . This concept is instrumental in depicting the independence relationships within Bayesian networks and serves as a tool in feature selection methodologies (Koller and Sahami, 1996; Shen et al., 2008).

### 2.2.19. Kruskal-Wallis test

The Kruskal-Wallis test is a supervised, univariate, non-parametric feature selection method that assesses whether two or more classes have equal medians and provides a corresponding value. In essence, this method is a cost-effective and straightforward feature selection technique with lower computational overhead. A value close to zero indicates discriminatory power of the feature, effectively selecting features containing discriminatory information while discarding others. Similar to other statistical tests, the Kruskal-Wallis test computes a test statistic and compares it with a critical value to determine significance (Saeyns et al., 2007; Ali Khan et al., 2014). The formula used to apply the Kruskal-Wallis test is as in equation 19 (Naik and Rangwala,

2016):

$$kruskal\_wallis = (N - 1) \frac{\sum_{i=1}^L n_i (\bar{r}_i - \bar{r})^2}{\sum_{i=1}^L \sum_{j=1}^{n_i} n_i (r_{ij} - \bar{r})^2} \quad (19)$$

Here,  $n_i$  represents the number of examples in class  $i$ ,  $r_{ij}$  denotes the ranking of example  $j$  in class  $i$ , and  $\bar{r}$  indicates the average ranking across all examples.

### 2.3. Dataset

In this study, we utilized the breast cancer dataset to evaluate the performance of the pyallffs library we developed. The dataset consists of 30 features associated with breast cancer, excluding identifiers and diagnosis labels. These features are employed to forecast the occurrence of breast cancer in individuals. You can freely access the dataset on Kaggle using the following link: <https://www.kaggle.com/datasets/yasserh/breast-cancer-dataset>.

### 2.4. How to Use pyallffs Library?

In this study, we introduce the pyallffs library, which provides a convenient way to apply 19 different feature selection methods to your datasets for rapid identification of the most important features. This library is designed to streamline the process of feature selection by offering a comprehensive suite of methods that can be easily adapted to various datasets. By leveraging pyallffs, researchers and data scientists can efficiently identify critical features for predictive modeling and analysis.

To access the codes and utilize the functionalities of the pyallffs library, you can install the library using the command "pip install pyallffs". After installation, you can import all the mentioned methods into your workflow using "from pyallffs import \*". Once imported, you can instantiate an object of interest from the available methods and feed your dataset into it. The library will then generate graphical outputs showcasing the most important features based on the chosen method.

For further information and to access the library's source code, please visit the following links:

<https://pypi.org/project/pyallffs/>

<https://github.com/tohid-yousefi/pyallffs>

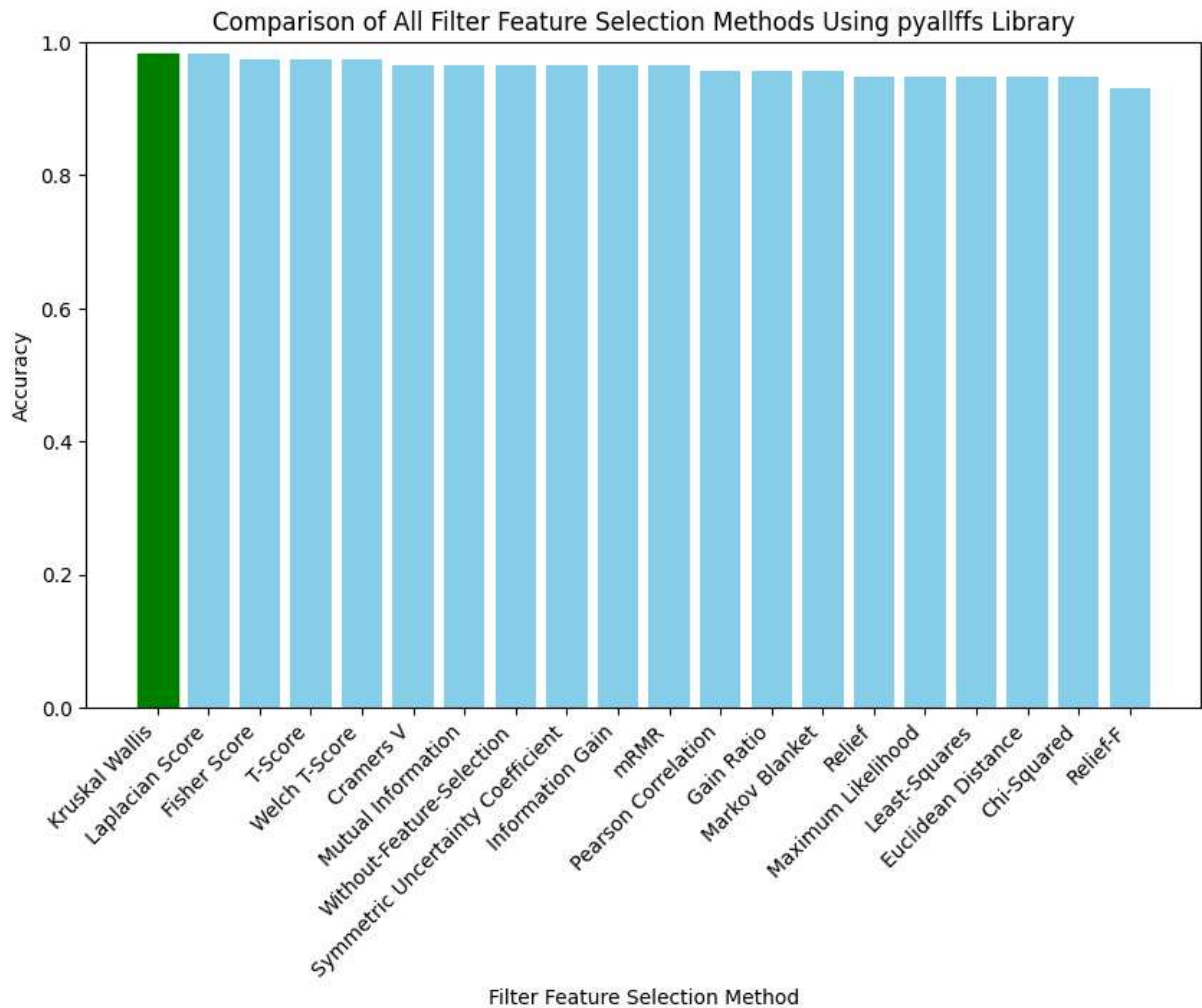
<https://www.kaggle.com/tohidyousefi/pyallffs>

## 3. Results and Discussion

In our study, we applied our developed pyallffs library to perform feature selection using all filter methods on the breast cancer dataset, reducing the feature set from 30 to 10 features. We conducted predictive modeling using the random forest algorithm based on this reduced feature set. Additionally, we compared this approach to using all features without feature selection. The results, as depicted in Table 1 and Figure 4, demonstrate that employing our library for feature selection led to improved predictive performance compared to using the entire set of features. This outcome underscores the effectiveness of our pyallffs library in enhancing predictive accuracy through efficient feature selection, contributing to more robust and reliable modeling outcomes in breast cancer prediction.

**Table 1.** Metrics of all filter feature selection methods using pyallffs library

Feature Selection Methods	Accuracy	Precision	Recall	F1 Score	ROC AUC
Kruskal Wallis	0.982	1.000	0.953	0.976	0.977
Laplacian Score	0.982	0.977	0.977	0.977	0.981
Fisher Score	0.974	0.976	0.953	0.965	0.970
T-Score	0.974	0.976	0.953	0.965	0.970
Welch T-Score	0.974	0.976	0.953	0.965	0.970
Cramers V	0.965	0.976	0.930	0.952	0.958
Mutual Information	0.965	0.976	0.930	0.952	0.958
Without-Feature-Selection	0.965	0.976	0.930	0.952	0.958
Symmetric Uncertainty Coefficient	0.965	1.000	0.907	0.951	0.953
Information Gain	0.965	0.976	0.930	0.952	0.958
mRMR	0.965	1.000	0.907	0.951	0.953
Pearson Correlation	0.956	0.952	0.930	0.941	0.951
Gain Ratio	0.956	0.975	0.907	0.940	0.946
Markov Blanket	0.956	0.952	0.930	0.941	0.951
Relief	0.947	0.930	0.930	0.930	0.944
Maximum Likelihood	0.947	0.951	0.907	0.929	0.939
Least-Squares	0.947	0.951	0.907	0.929	0.939
Euclidean Distance	0.947	0.951	0.907	0.929	0.939
Chi-Squared	0.947	0.951	0.907	0.929	0.939
Relief-F	0.930	0.889	0.930	0.909	0.930



**Figure 4.** Comparison of all filter feature selection methods using PYALLFFS library.



In conclusion, our developed pyallfs library facilitated feature selection using all filter methods on the breast cancer dataset, followed by predictive modeling using the random forest algorithm. Comparing the performance of this approach with predictive modeling using all features without feature selection, our results clearly demonstrate superior predictive accuracy when leveraging the pyallfs library for feature selection. Therefore, we believe that the pyallfs library offers significant advantages to developers by consolidating all feature selection methods under a single framework, providing enhanced performance, and simplifying the process of model development and optimization in breast cancer prediction tasks. This consolidation not only improves efficiency but also supports more informed decision-making in machine learning workflows.

In the future, we plan to enhance the pyallfs library by integrating metaheuristic algorithms to optimize the parameters of feature selection methods, aiming to create a more comprehensive tool. Additionally, we intend to design a user-friendly interface and offer this library as a free product, making it accessible to a broader audience and further supporting the machine learning community.

**Author Contributions**

The percentage of the authors contributions is presented below. All authors reviewed and approved the final version of the manuscript.

	T.Y.	Ö.V.
C	100	100
D	100	
S		100
L	100	100
W	100	100
CR	100	100
SR	100	100
PM	100	100
FA	100	100

C=Concept, D= design, S= supervision, L= literature search, W= writing, CR= critical review, SR= submission and revision, PM= project management, FA= funding acquisition.

**Conflict of Interest**

The authors declared that there is no conflict of interest.

**Ethical Consideration**

Ethics committee approval was not required for this study because of there was no study on animals or humans.

**References**

Ali Khan S, Hussain A, Basit A, Akram S. 2014. Kruskal-Wallis-based computationally efficient feature selection for face recognition. *Sci World J*, 2014: 1-6.

Ali SI, Shahzad W. 2012. A feature subset selection method based on symmetric uncertainty and ant colony optimization. In: 2012 Inter Conference on Emerging Technologies, 8-9 October, 2012, Islamabad, Pakistan, pp: 1-6.

Arauzo-Azofra A, Benitez JM, Castro JL. 2004. A feature set measure based on relief. In: *Proceedings of the fifth Inter conference on Recent Advances in Soft Computing*, April 27-28, Copenhagen, Denmark pp: 104-109.

Battiti R. 1994. Using mutual information for selecting features in supervised neural net learning. *IEEE Transact Neural Networks*, 4: 537-550.

Belkin M, Niyogi P. 2001. Laplacian eigenmaps and spectral techniques for embedding and clustering. *Adv Neural Inform Proces Systems*, 2001: 14.

Beraha M, Metelli AM, Papini M, Tirinzoni A, Restelli M. 2019. Feature selection via mutual information: New theoretical insights. In: *2019 Inter Joint Conference on Neural Networks (IJCNN)*, 14-19 July 2019, Budapest, Hungary pp: 1-9.

Bolón-Canedo V, Sánchez-Marono N, Alonso-Betanzos A, Benítez JM, Herrera F. 2014. A review of microarray datasets and applied feature selection methods. *Inform Sci*, 282: 111-135.

Bryant FB, Satorra A. 2012. Principles and practice of scaled difference chi-square testing. *Struct Equation Model: A Multidisciplin J*, 3: 372-398.

Budak H, Taşabat SE. 2016. A modified t-score for feature selection. *Anadolu Univ J Sci Technol A-Applied Sci Engin*, 5: 845-852.

Carey JJ, Delaney MF. 2010. T-scores and Z-scores. *Clinical Rev Bone Mineral Metabol*, 8: 113-121.

Chandra B, Gupta M. 2011. An efficient statistical feature selection approach for classification of gene expression data. *J Biomed Inform*, 4: 529-535.

Chandrashekar G, Sahin F. 2014. A survey on feature selection methods. *Comput Elect Engin*, 1: 16-28.

Cover TM. 1999. *Elements of information theory*. John Wiley & Sons, London, UK, pp: 54.

Dash M, Liu H. 2003. Consistency-based search in feature selection. *Artificial Intel*, 1-2: 155-176.

Delacre M, Lakens D, Leys C. 2017. Why psychologists should by default use Welch's t-test instead of Student's t-test. *Inter Rev Soc Psychol*, 1: 92-101.

Ding C, Peng H. 2005. Minimum redundancy feature selection from microarray gene expression data. *J Bioinform Comput Biol*, 2: 185-205.

Doquire G, Verleysen M. 2011. Feature selection with mutual information for uncertain data. In: *Data Warehousing and Knowledge Discovery: 13th Inter Conference, DaWaK 2011*, Toulouse, France, August 29-September 2, pp: 330-341.

Esmael B, Arnaout A, Fruhwirth R, Thonhauser G. 2012. A statistical feature-based approach for operations recognition in drilling time series. *Inter J Comput Inform Systems Industrial Manage Applicat*, 4(6): 100-108.

Faulkner KG. 2005. The tale of the T-score: review and perspective. *Osteoporosis Inter*, 16, 347-352.

François D, Rossi F, Wertz V, Verleysen M. 2007. Resampling methods for parameter-free and robust feature selection with mutual information. *Neurocomput*, 70(7-9): 1276-1288.

Goswami S, Chakrabarti A. 2014. Feature selection: A practitioner view. *Inter J Inform Technol Comput Sci (IJITCS)*, 6(11): 66

Gu Q, Li Z, Han J. 2012. Generalized fisher score for feature selection. *arXiv preprint arXiv:1202.3725*.

Hall MA, Holmes G. 2000. Benchmarking attribute selection techniques for data mining. *IEEE Trans Knowl Data Eng*, 15 (2003): 1437-1447.

Hall MA, Smith LA. 1998. Practical feature subset selection for machine learning. In: *Computer Science Proceedings of the 21st Australasian Computer Science Conference ACSC'98*, Perth, 4-6 February, Berlin, Germany, pp: 181-191.

He X, Cai D, Niyogi P. 2005. Laplacian score for feature selection.

- Adv Neural Inform Proces Systems, 2005: 18.
- He X, Niyogi P. 2003. Locality preserving projections. *Adv Neural Inform Proces Systems*, 2003: 16.
- Hernández-Torruco J, Canul-Reich J, Frausto-Solís J, Méndez-Castillo JJ. 2014. Feature selection for better identification of subtypes of Guillain-Barré syndrome. *Comput Math Methods Med*, 2014: 432109.
- Kabir MM, Islam MM, Murase K. 2010. A new wrapper feature selection approach using neural network. *Neurocomput*, 73(16-18): 3273-3283.
- Kalousis A, Prados J, Hilario M. 2007. Stability of feature selection algorithms: a study on high-dimensional spaces. *Knowledge Inform Systems*, 12: 95-116.
- Kannan SS, Ramaraj N. 2010. A novel hybrid feature selection via symmetrical uncertainty ranking based local memetic search algorithm. *Knowledge-Based Systems*, 23(6): 580-585.
- Kass GV. 1980. An exploratory technique for investigating large quantities of categorical data. *J Royal Stat Soc: Series C (Applied Stat)*, 29(2): 119-127.
- Kira K, Rendell LA. 1992. The feature selection problem: Traditional methods and a new algorithm. In: *Proceedings of the Tenth National Conference on Artificial intelligence*, July 12-16, California, USA, pp: 129-134.
- Kohavi R, John GH. 1997. Wrappers for feature subset selection. *Artificial Intel*, 97(1-2): 273-324.
- Koller D, Sahami M. 1996. Toward optimal feature selection. In: *ICML*, 292.
- Kononenko I. 1994. Estimating attributes: Analysis and extensions of RELIEF. In: *European Conference on Machine Learning*, April 6-8, Catania, Italy, pp:71-182.
- Kraskov A, Stögbauer H, Grassberger P. 2004. Estimating mutual information. *Physical Rev E*, 69(6): 066138.
- Kullback S, Leibler RA. 1951. On information and sufficiency. *Annals Math Stat*, 22(1): 79-86.
- Ladha L, Deepa T. 2011. Feature selection methods and algorithms. *Inter J Comput Sci Engin*, 3(5): 1787-1797.
- Liu H, Motoda H, Setiono R, Zhao Z. 2010. Feature selection: An ever evolving frontier in data mining. *Feature Select Data Min*, 2010: 4-13.
- Lu D, Weng Q. 2007. A survey of image classification methods and techniques for improving classification performance. *Inter J Remote Sensing*, 28(5): 823-870.
- Lun Goa TL, Yaob L, Wenb F. 2013. Research and application of data mining feature selection based on relief algorithm. *Work*, 2013: 515.
- Mani K, Kalpana P. 2016. A review on filter based feature selection. *Inter J Innov Res Computer Communicat Engin (IJIRCC)*, pp: 2320-9801.
- Martínez Casasnovas JA, Klaasse A, Nogués Navarro J, Ramos Martín MC. 2008. Comparison between land suitability and actual crop distribution in an irrigation district of the Ebro valley (Spain). *Spanish J Agri Res*, 6(4): 700-713.
- Miao J, Niu L. 2016. A survey on feature selection. *Procedia Comput Sci*, 91: 919-926.
- Naik A, Rangwala H. 2016. Embedding feature selection for large-scale hierarchical classification. In: *2016 IEEE Inter Conference on Big Data (Big Data)*, December 5-8, Washington DC, USA, pp: 1212-1221.
- Nilsson R. 2007. Statistical feature selection: with applications in life science. *Institutionen för fysik, kemi och biologi*, Berlin, Germany, pp: 54.
- Novaković J. 2016. Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav J Operat Res*, 21: 1.
- Opitz D, Maclin R. 1999. Popular ensemble methods: An empirical study. *J Artific Intel Res*, 11: 169-198.
- Pearl J. 1988. Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan kaufmann.
- Peng H, Fan Y. 2015. Direct L<sub>2</sub>(p)-Norm learning for feature selection. *arXiv preprint arXiv: 1504.00430*.
- Priyadarsini RP, Valarmathi M, Sivakumari S. 2011. Gain ratio based feature selection method for privacy preservation. *ICTACT J Soft Comput*, 1(4): 201-205.
- Radovic M, Ghalwash M, Filipovic N, Obradovic Z. 2017. Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC Bioinform*, 18: 1-14.
- Rossi F, Lendasse A, François D, Wertz V, Verleysen M. 2006. Mutual information for the selection of relevant variables in spectrometric nonlinear modelling. *Chemometrics Intel Lab Systems*, 80(2): 215-226.
- Saeyns Y, Inza I, Larranaga P. 2007. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19): 2507-2517.
- Sedgwick P. 2012. Pearson's correlation coefficient. *BMJ*, 2012: 345.
- Shannon CE. 1948. A mathematical theory of communication. *Bell System Technic J*, 27(3): 379-423.
- Shardlow M. 2016. An analysis of feature selection techniques. *J Univ Manchester*, 2016: 1-7.
- Shen J, Li L, Wong W-K. 2008. Markov Blanket Feature Selection for Support Vector Machines. *AAAI*, 2008: 696-701.
- Singh B, Kushwaha N, Vyas OP. 2014. A feature subset selection technique for high dimensional data using symmetric uncertainty. *J Data Analysis Inform Proces*, 2(4): 95-105.
- Suebsing A, Hiransakolwong N. 2009. Feature selection using euclidean distance and cosine similarity for intrusion detection model. In: *2009 First Asian Conference on Intelligent Information and Database Systems*, April 1-3, Dong Hoi, Quang Binh, Vietnam, pp: 86-91.
- Suzuki T, Sugiyama M, Sese J, Kanamori T. 2008. Approximating mutual information by maximum likelihood density ratio estimation. *PMLR*, 2008: 5-20.
- Suzuki T, Sugiyama M, Tanaka T. 2009. Mutual information approximation via maximum likelihood estimation of density ratio. In: *2009 IEEE Inter Symposium on Information Theory*, 28 June - 3 July, Seoul, Korea, pp: 463-467.
- Tsamardinos I, Aliferis CF, Statnikov A. 2003. Time and sample efficient discovery of Markov blankets and direct causal relations. In: *Proceedings of the ninth ACM SIGKDD Inter Conference on Knowledge Discovery and Data Mining*, August 24-27, Washington, DC, USA, pp: 673-678.
- Tsamardinos I, Aliferis CF, Statnikov AR, Statnikov E. 2003. Algorithms for large scale Markov blanket discovery. *FLAIRS*, 2003: 376-381.
- Ugoni A, Walker BF. 1995. The Chi square test: an introduction. *COMSIG Rev*, 4(3): 61.
- Urbanowicz RJ, Olson RS, Schmitt P, Meeker M, Moore JH. 2018. Benchmarking relief-based feature selection methods for bioinformatics data mining. *J Biomed Inform*, 85: 168-188
- Vergara JR, Estévez PA. 2014. A review of feature selection methods based on mutual information. *Neural Comput Applicat*, 24, 175-186
- Von Luxburg U. 2007. A tutorial on spectral clustering. *Stat Comput*, 17: 395-416.
- Vora S, Yang H. 2017. A comprehensive study of eleven feature selection algorithms and their impact on text classification. In: *2017 Computing Conference*, 18-20 July, Kensington, London, UK, pp: 440-449.
- Welch BL. 1947. The generalization of 'STUDENT'S' problem when several different population variances are involved.

- Biometrika, 34(1-2): 28-35.
- Witten IH, Frank E. 2002. Data mining: practical machine learning tools and techniques with Java implementations. *Acm Sigmod Rec*, 31(1): 76-77.
- Witten IH, Frank E, Hall MA, Pal CJ, Data M. 2005. Practical machine learning tools and techniques. *Data Mining*, 2005: 403-413.
- Xiang S, Nie F, Meng G, Pan C, Zhang C. 2012. Discriminative least squares regression for multiclass classification and feature selection. *IEEE Transact Neural Networks Learn Systems*, 23(11): 1738-1754.
- Yousefi T, Aktaş ÖV. 2024. Predicting Customer Satisfaction with Hybrid Basic Filter-Based Feature Selection Method.
- Yousefi T, Varlıklar Ö. 2024. Breast cancer prediction with hybrid filter-wrapper feature selection. *Inter J Adv Nat Sci Engin Res*, 8: 411-419.
- Zheng A, Casari A. 2018. *Feature engineering for machine learning: principles and techniques for data scientists*. O'Reilly Media, London, UK, pp: 263.