Feature Selection for High-Dimensional Data: A Kolmogorov-Smirnov Correlation-Based Filter

Jacek Biesiada[1] and Włodzisław Duch[2,3]

[1] Division of Computer Methods, Dept. of Electrotechnology, The Silesian University of Technology, Katowice, Poland
[2] Dept. of Informatics, Nicolaus Copernicus University, Toruń, Poland
[3] School of Computer Engineering, Nanyang Technological University, Singapore
Google: Duch

Summary. An algorithm for filtering information based on the Kolmogorov-Smirnov correlation-based approach has been implemented and tested on feature selection. The only parameter of this algorithm is statistical confidence level that two distributions are identical. Empirical comparisons with 4 other state-of-the-art features selection algorithms (FCBF, CorrSF, ReliefF and ConnSF) are very encouraging.

1 Introduction

For large highly dimensional datasets feature ranking and feature selection algorithms are usually of the filter type. In the simplest case feature filter is a function returning a relevance index $J(\mathcal{S}|\mathcal{D}, C)$ that estimates, given the data $\mathcal{D}$, how relevant a given feature subset $\mathcal{S}$ is for the task C (usually classification or approximation of the data). The relevance index $J(\mathcal{S}|\mathcal{D}, C)$ is calculated directly from data, without any reference to the results of programs that are used on data with reduced dimensionality. Since the data $\mathcal{D}$ and the task C are usually fixed and only the subsets $\mathcal{S}$ varies an abbreviated form $J(\mathcal{S})$ is used. Instead of a simple function (such as correlation or information content) an algorithmic procedure, such as building a decision tree or finding nearest neighbors, may be used to estimate this index.

Relevance indices computed for individual features $X_i, i = 1 \dots N$ provide indices that establish a ranking order $J(X_{i_1}) \leq J(X_{i_2}) \cdots \leq J(X_{i_N})$. Those features which have the lowest ranks are filtered out. For independent features this may be sufficient, but if features are correlated many of them may be redundant. For some data distributions the best pair of features may not even include a single best feature [14, 2]! Thus ranking does not guarantee that the largest subset of important features will be found. Methods that search for the best subset of features may also use filters to evaluate the usefulness of subsets of features.

Although in the case of filter methods there is no direct dependence of the relevance index on the adaptive algorithms obviously the thresholds for feature rejection may be set either for relevance indices, or by evaluation of the feature contributions by the final system. Features are ranked by the filter, but how many are finally taken may be determined using adaptive system as a wrapper. Evaluation of the adaptive system performance (frequently crossvalidation tests) are done only for a few pre-selected feature sets, but still this “filtrapper” approach may be rather costly. What is needed is a simple filter method that may be applied to large datasets ranking and removing redundant features, parameterized in statistically well-established way. Such an approaches is described in this paper.

In next section a new relevance index based on the Kolmogorov-Smirnov (K-S) test to estimate correlation between the distribution of feature values and the class labels is introduced. Correlation-based filters are very fast and may be competitive to filters based on information theory. Therefore in section 3 empirical comparisons between K-S filter, Pearson’s correlation based filter and popular filters based on information gain is made on a number of datasets.

2 Theoretical framework

2.1 Correlation-Based Measures

For feature X with values x and classes C with values c, where X, C are treated as random variables, Pearson’s linear correlation coefficient is defined as [11]:

$$\varrho(X,C) = \frac{E(XC) - E(X)E(C)}{\sqrt{\sigma^2(X)\sigma^2(C)}} = \frac{\sum_i (x_i - \bar{x}_i)(c_i - \bar{c}_i)}{\sqrt{\sum_i (x_i - \bar{x}_i)^2 \sum_j (c_i - \bar{c}_i)^2}}. \quad (1)$$

$\varrho(X,C)$ is equal to ± 1 if X and C are linearly dependent and zero if they are completely uncorrelated. The simplest test estimating significance of the differences in $\varrho(X,C)$ values is based on the probability that two variables are correlated [11]:

$$\mathcal{P}(X \sim C) = \text{erf}\left(|\varrho(X,C)|\sqrt{N/2}\right), \quad (2)$$

where erf is the error function. The feature list ordered by decreasing values of the $\mathcal{P}(X \sim C)$ may serve as feature ranking. Non-parametric, or Spearman’s rank correlation coefficients may be useful for ordinal data types.

An alternative approach is to use χ^2 statistics, but in both cases for large number of samples probability $\mathcal{P}(X \sim C)$ is so close to 1 that ranking becomes impossible due to the finite numerical accuracy of computations. For example, with $N = 1000$ samples small coefficients $\varrho(X,C) \approx 0.02$ lead to probabilities of correlation around 0.5. The $\varrho(X,C)$ or χ^2 thresholds for the significance of a given feature may therefore be taken from a large interval corresponding to almost the same probabilities of correlation.

Information theory is frequently used to define relevance indices. The Shannon information, or feature and class entropy, is:

$$H(X) = -\sum_i \mathcal{P}(x_i) \log \mathcal{P}(x_i); \qquad H(C) = -\sum_i \mathcal{P}(c_i) \log \mathcal{P}(c_i) \quad (3)$$

and the joint Shannon entropy is:

$$H(X, C) = -\sum_{i,j} \mathcal{P}(x_i, c_j) \log \mathcal{P}(x_i, c_j) \quad (4)$$

Mutual Information (MI) is the basic quantity used for information filtering:

$$MI(X, C) = H(X) + H(C) - H(X, C) \quad (5)$$

Symmetrical Uncertainty Coefficient (SU) has similar properties to mutual information:

$$SU(X, C) = 2 \left[\frac{MI(X, C)}{H(X) + H(C)} \right] \quad (6)$$

If a group of k features $\mathbf{X}_k$ has already been selected, correlation coefficient may be used to estimate correlation between this group and the class, including intercorrelations between the features. Denoting the average correlation coefficient between these features and classes as $r_{kc} = \bar{\varrho}(\mathbf{X}_k, C)$ and the average between different features as $r_{kk} = \bar{\varrho}(\mathbf{X}_k, \mathbf{X}_k)$ the relevance of the feature subset is defined as:

$$J(\mathbf{X}_k, C) = \frac{k r_{kc}}{\sqrt{k + (k-1) r_{kk}}}. \quad (7)$$

This formula has been used in the Correlation-based Feature Selection (CFS) algorithm [6] adding (forward selection) or deleting (backward selection) one feature at a time.

A definition of predominant correlation proposed by Yu and Liu [16] for Fast Correlation-Based Filter (FCBF) includes correlations beetwen feature and classes and between pairs of features. The FCBF algorithm does a typical ranking using SU coefficient (eq. 6) to determine class-feature relevance, setting some threshold value $SU \geq \delta$ to decide how many features should be taken. In the second part redundant features are removed by defining the “predominant features”.

A different type of selection method called ConnSF, based on inconsistency measure, has been proposed by Dash *et al.* [3] and will be used for comparison in Sec. 3. Two identical input vectors are inconsistent if they have identical class labels (a similar concept is used in rough set theory). Intuitively it is clear that inconsistency grows when the number of features is reduced and that feature subsets that lead to high inconsistency are not useful. If there are n samples in the dataset with identical feature values x_i, and n_k among them belong to class k then the inconsistency count is defined as $n - \max_k c_k$. The total inconsistency count for a feature subset is the sum of all inconsistency counts for all data vectors.

A different way to find feature subsets is used in the Relief algorithm ([8] and [13]). This algorithm (see Fig. 1) estimates weights of features according to how well

```
set all weights $W_{xi} = 0$
for j=1 to m do begin
  randomly select instance X;
  find nearest hit $H$ and nearest miss $M$;
    for i:=1 to k do begin
    $W_{xi} \leftarrow W_{xi} - D(x_i, X, H)/m + D(x_i, X, M)/m$
  end;
end;
```

Fig. 1. Sketch of the Relief algorithm.

their values distinguish between data vectors that are near to each other. For a randomly selected vector X from a data set $\mathcal{S}$ with k features Relief searches the dataset for its two nearest neighbors: the nearest hit H from the same class and the nearest miss M from another class. For feature x and two input vectors X, X' the contribution to the weight W_x is proportional to the $D(x, X, X') = 1 - \delta(X(x), X'(x))$ for binary or nominal features, and $D(x, X, X') = |X(x) - X'(x)|$ for continuous features. The process is repeated m times, where m is a user defined parameter [8]. Normalization with m in calculation of W_x guarantees that all weights are in the $[-1, 1]$ interval. In our empirical studies (Sec. 3) we have used an extension of this agorithm for multiclass problems, called ReliefF [13].

2.2 Kolmogorov-Smirnov Correlation-Based Filter Approach.

Equivalence of two random variables may be evaluated using the Kolmogorov-Smirnov (K-S) test [7]. The K-S test measures the maximum difference between cummulative distribution of two random variables. If a feature is redundant than the hypothesis that its distribution is equal to already selected feature should have high probability. n independent observations of two random variables X, X' are given in the training data, where for the K-S test to be valid n should be more than 40. The test for X, X' feature redundancy proceeds as follows:

- Discretization of feature values x into k bins $[x_i, x_{i+1}], i = 1 \dots k$ is performed.
- Frequency f_i, f'_k of occurrences of feature values in each bin are recorded.
- Based on the frequency counts cumulative distribution functions F_i and F'_i are constructed.
- λ (K-S statistics) is the largest absoulte difference between F_i and F'_i, i.e,

$$\lambda = \sqrt{n/2} \max_i |F_i - F'_i| \qquad \text{for } i = 1, 2, \dots k. \tag{8}$$

Probability that the maximum K-S distance λ_α is larger than observed may be calculated using K-S statistics for each parameter α [9] that has the meaning of statistical significance level. When $\lambda < \lambda_\alpha$ then the two distributions are equivalent with α significance level, and thus one of the features is redundant. Using typical significance values of 0.95 solves the problem of the threshold values for redundancy.

The Kolmogorov-Smirnov Correlation-Based Filter (K-S CBF) algorithm is presented below. First, the relevance is determined using the symmetrical uncertainty (other relevance criteria may also be used), and then K-S test applied to remove redundancy.

```
Algorithm K-S RBF:
Relevance analysis
1. Calculate the $SU(X, C)$ relevance indices and create an ordered list $\mathcal{S}$ of features according to the decreasing value of their relevance.
Redundancy analysis
2. Take as the feature $X$ the first feature from the $\mathcal{S}$ list
3. Find and remove all features for which $X$ is approximately equivalent according to the K-S test
4. Set the next remaining feature in the list as $X$ and repeat step 3 for all features that follow it in the $\mathcal{S}$ list.
```

Fig. 2. A two-step Kolmogorov-Smirnov Correlation Based Fiter (K-S CBF) algorithm.

3 Empirical Studies.

To evaluate the performance of the K-S CBF algorithm both artificial and real datasets have been used with a number of classification methods. Two artificial datasets, Gauss4, and Gauss8, have been used in our previous study [4]. Gauss4 is based on sampling from 4 Gaussian functions with unit dispersion in 4 dimensions, each cluster representing a separate class. The first function is centered at $(0, 0, 0, 0)$, the next at $(1, 1/2, 1/3, 1/4)$, $(2, 1, 2/3, 1/2)$, and $(3, 3/2, 3, 3/4)$, respectively. The dataset contains 4000 vectors, 1000 per each class. In this case the ideal ranking should give the following order: $X_1 > X_2 > X_3 > X_4$.

Gauss8 used here is an extension of Gauss4, adding 4 additional features that are approximately linearly dependent $X_{i+4} = 2X_i + \epsilon$, where ϵ is a uniform noise with a unit variance. In this case the ideal ranking should give the following order: $X_1 > X_5 > X_2 > X_6 > X_3 > X_7 > X_4 > X_8$ and the selection should reject all 4 linearly dependent features as redundant. K-S CBF algorithm and ConnSF [3] algorithm had no problem with this task, but FCBF [16] selected only 3 features, CorrSF [6] selected only first two and ReliefF [13] left only feature 1 and 5, giving them weight 0.154 (for features 2 and 6 the weight was 0.060, dropping to 0.024 for feature 3, 6 and to 0.017 for features 4, 8.

In Table 3 results of Naive Bayes Classifier (NBC) (Weka implementation, [15]), the nearest neighbor algorithm (1NN) with Euclidean distance function, C4.5 tree [12] and the Support Vector Machine with a linear kernel are given (Weka and SVM, Ghostminer 3.0 implementation[4]).

For the initial comparison on real data several datasets from the UCI Machine Learning Repository [10] and the UCI KDD Archive [1] were used. A summary of all datasets is presented in Table 3. For each data set all five feature selection

[4] http://www.fqspl.com.pl/ghostminer/

Title	Selected features					
	Full set	FCBF	CorrSF	ReliefF	ConnSF	K-S CBF
Features	1 to 8	1+2+3	1+2+5	1+5	1 to 4	1to 4
NBC	82.13	81.57	80.25	76.95	82.13	82.13
1NN	73.42	73.90	71.10	68.12	73.42	73.42
C4.5	78.30	79.12	78.95	76.15	78.70	78.70
SVM	81.88	81.70	80.90	76.95	81.73	81.73
Average	79.91	79.09	78.83	75.34	80.40	80.40

Table 1. Accuracy of 4 classifiers on selected subsets of features for the Gauss8 dataset.

Title	Features	Instances	Classes
Hypothyroid	21	3772	3
Lung-cancer	58	32	3
Promoters	59	106	2
Splice	62	3190	3

Table 2. Summary of the datasets used in empirical studies.

algorithms are compared (FCBF [16], CorrSF [6], ReliefF [13], ConnSF [3], and K-S CBF) and the number of features selected by each algorithm is given. For data sets containing features with continuous values the MDLP discretization algorithm has been applied[5] [5]. 5 neighbors and 30 instances were used for ReliefF, as suggested by Robnik-Sikonia and Kononenko [13]. For CorrSF and ConnSF forward search strategy has been used, and for FCBF, ReliefF, and the K-S CBF forward search strategy based on ranking.

Dataset	Selected features					
	Full set	FCBF	CorrSF	ReliefF	ConnSF	K-S CBF
Hypothyroid	21	5	**1**	*11*	6	6
Lung-cancer	58	6	*11*	8	4	**3**
Splice	62	*22*	**6**	19	10	14
Promoters	59	*6*	**4**	**4**	**4**	5
Average	50	*9.8*	**5.5**	10.5	6	7

Table 3. The number of selected features for each algorithm; bold face – lowest number, italics – highest number.

The overall balanced accuracy (accuracy for each class, averaged over all classes) obtained from 10-fold cross-validation calculations is reported. For datasets with significant differences between samples from different classes balanced accuracy is a more sensitive measure than the overall accuracy. Results of these calculations are collected in Table 3.

[5] available from www.public.asu.edu/∼huanliu/

Method	C 4.5 tree					
Dataset	Full set	FCBF	CorrSF	ReliefF	ConnSF	K-S CBF
Hypothyroid	**99.91**	*66.94*	85.92	98.94	93.45	98.68
Lung-cancer	*49.43*	63.87	**67.21**	63.87	63.22	64.76
Splice	94.35	94.05	*93.39*	**94.80**	93.63	94.05
Promoters	81.13	**85.84**	83.96	*65.09*	84.90	81.13
Average	81.21	*77.68*	82.62	80.68	83.80	**84.66**
Method	Naive Bayes					
Dataset	Full set	FCBF	CorrSF	ReliefF	ConnSF	K-S CBF
Hypothyroid	83.20	66.94	*58.48*	70.28	67.14	**85.83**
Lung-cancer	47.92	50.57	**73.48**	50.57	65.68	*41.74*
Splice	94.88	**96.03**	*93.31*	95.54	94.17	94.75
Promoters	90.56	**95.28**	93.39	*61.32*	93.39	87.35
Average	79.14	73.21	79.67	*69.43*	**80.10**	77.42
Method	1 Nearest Neighbor					
Dataset	Full set	FCBF	CorrSF	ReliefF	ConnSF	K-S CBF
Hypothyroid	*61.60*	83.89	71.72	78.92	**86.94**	83.93
Lung-cancer	39.94	*36.01*	**66.84**	53.13	65.41	45.70
Splice	*80.08*	84.45	**87.68**	84.04	86.77	83.82
Promoters	85.85	**92.45**	86.79	*59.43*	81.13	85.85
Average	*66.49*	74.28	78.26	69.01	**80.06**	74.83
Method	SVM					
Dataset	Full set	FCBF	CorrSF	ReliefF	ConnSF	K-S CBF
Hypothyroid	52.65	45.49	*44.07*	51.24	45.13	**84.31**
Lung-cancer	*41.37*	55.41	**66.07**	61.60	59.37	47.35
Splice	92.81	95.73	93.75	**95.75**	*90.08*	95.11
Promoters	**93.40**	91.50	77.36	*58.49*	87.33	93.11
Average	70.06	72.03	70.31	*66.77*	70.48	**80.04**

Table 4. Balanced accuracy for the 4 classification methods on features selected by each algorithm; bold face – best results, italics – worst.

4 Conclusion

A new algorithm, K-S CBF, for finding non-redundant feature subsets based on the Kolmogorov-Smirnov test has been introduced. It has only one parameter, statistical significance or the probability that the hypothesis that distributions of two features is equivalent is true. Our initial tests are encouraging: on the artificial data perfect ranking has been recreated and redundant features rejected, while on the real data, with rather modest number of features selected results are frequently the best, or close to the best, comparing with four state-of-the-art feature selection algorithms. The new algorithm seems to work especially well with the linear SVM classifier. Computational demands of K-S CBF algorithm are similar to other correlation-based filters and much lower than ReliefF.

It is obvious that sometimes statistical significance at 0.05 level selected for our tests is not optimal and for the lung cancer data too few features have been selected, leading to a large decrease of accuracy. This parameter may be optimized in crossvalidation tests on the training set, but the method guarantees that each time only non-redundant subset of features will be selected. Various variants of the Kolmogorov-Smirnov test exist [11] and the algorithm may be used with other indices for relevance indication. These possibilities remain to be explored. Further tests on much larger bioinformatics data will be reported soon.

Acknowledgement. This work was financed by the Polish Committee for Scientific Research (KBN) grant (2005-2007).

References

1. S.D. Bay. *The UCI KDD archive*. Univ. of California, Irvine, 1999. http://kdd.ics.uci.edu.
2. T.M. Cover. The best two independent measurements are not the two best. *IEEE Transactions on Systems, Man, and Cybernetics*, 4:116–117, 1974.
3. M. Dash and H. Liu. Consistency-based search in feature selection. *Artificial Intelligence*, 151:155–176, 2003.
4. W. Duch, T. Winiarski, J. Biesiada, and A. Kachel. Feature ranking, selection and discretization. In *Proceedings of Int. Conf. on Artificial Neural Networks (ICANN)*, pages 251–254, Istanbul, 2003. Bogazici University Press.
5. U.M. Fayyad and K.B. Irani. Multi-interval discretization of continous-valued attributes for classification learning. In R. Bajcsy, editor, *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, Chambery, France*, pages 1022–1027, San Francisco, CA, 1993. Morgan Kaufmann.
6. M.A. Hall. *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, Department of Computer Science, University of Waikato, Waikato, N.Z., 1999.
7. R. Laha I. Chakravarti and J. Roy. *Handbook of Methods of Applied Statistics*. John Wiley and Sons, Chichester, 1967.
8. K. Kira and L.A. Rendell. A practical approach to feature selection. In *Proceedings of the Ninth International Conference on Machine Learning (ICML-92)*, pages 249–256, San Francisco, CA, 1992. Morgan Kaufmann.
9. N. Hastings M. Evans and B. Peacock. *Statistical Distributions, 3rd. ed.* John Wiley and Sons, Chichester, 2000.
10. C.J. Mertz and P.M. Murphy. *The UCI repository of machine learning databases*. Univ. of California, Irvine, 1998. http://www.ics.uci.edu.pl/ mlearn/MLRespository.html.
11. W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical recipes in C. The art of scientific computing*. Cambridge University Press, Cambridge, UK, 1988.
12. J.R. Quinlan. *C4.5: Programs for Machine Learning*. M. Kaufman, San Mateo, CA, 1993.
13. M. Robnik-Sikonja and I. Kononenko. Theoretical and empirical analysis of relieff and rrelieff. *Machine Learning*, 53:23–69, 2003.
14. G.T. Toussaint. Note on optimal selection of independent binary-valued features for pattern recognition. *IEEE Transactions on Information Theory*, 17:618–618, 1971.
15. I. Witten and E. Frank. *Data minig – practical machine learning tools and techniques with JAVA implementations*. Morgan Kaufmann, San Francisco, CA, 2000.
16. L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proc 12th Int Conf on Machine Learning (ICML-03), Washington, D.C.*, pages 856–863, San Francisco, CA, 2003. Morgan Kaufmann.