Pattern Recognition 66 (2017) 425–436

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Granger Causality Driven AHP for Feature Weighted kNN

Gautam Bhattacharya[a], Koushik Ghosh[b], Ananda S. Chowdhury[c,*]

[a] Department of Physics, University Institute of Technology, University of Burdwan, Golapbag(North), Burdwan 713104, India
[b] Department of Mathematics, University Institute of Technology, University of Burdwan, Golapbag(North), Burdwan 713104, India
[c] Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700032, India

ARTICLE INFO

Keywords:
KNN Classification
Feature Weighting
Analytic Hierarchy Process
Granger causality

ABSTRACT

The kNN algorithm remains a popular choice for pattern classification till date due to its non-parametric nature, easy implementation and the fact that its classification error is bounded by twice the Bayes error. In this paper, we show that the performance of the kNN classifier improves significantly from the use of (training) class-wise group-statistics based two criteria during pairwise comparison of features in a given dataset. Granger causality is employed to assign preferences to each criteria. Analytic Hierarchy Process (AHP) is applied to obtain weights for different features from the two criteria and their preferences. Finally, these weights are used to build a weighted distance function for the kNN classification. Comprehensive experimentation on fifteen benchmark datasets of the UCI Machine Learning Repository clearly reveals the supremacy of the proposed Granger causality driven AHP induced kNN algorithm over the kNN method with many different distance metrics, and, with various feature selection strategies. In addition, the proposed method is also shown to perform well on high-dimensional face and hand-writing recognition datasets.

1. Introduction

The kNN algorithm [1,2] remains a popular choice for pattern classification [3], as it is non-parametric in nature, is easy to implement and has its classification error bounded by twice the Bayes error. Very recent applications of the kNN algorithm and its variants can be found in diverse fields like automated web usage data mining [4], classification of big data [5] and hyperspectral image classification [6,7]. The key factors which influence the accuracy of the kNN classifier are the distance and similarity function [8,9] it uses to find the nearest neighbors of a query point, the selection of the optimal number of nearest neighbors [10], i.e., k [11], and data pre-processing like feature selection [12]. According to the concept of generalization power, one distance measure cannot be strictly better than any other, when considering all possible problems with equal probability [13,14]. Some common distance metrics which are used for classification without any form of learning are Euclidean distance, $L1$-norm distance, χ^2 distance and Mahalanobis distance [15]. For the better generalization capability, several learning based distances like Xing distance [16], Information Theoretic Metric Learning (ITML)-based distance [17], Kernel Relevant Component Analysis (KRCA) [18], Large Margin Nearest Neighbor (LMNN) [19], Information Geometric Metric Learning (IGML)-based linear metric and Kernel Information Geometric Metric Learning (KIGML)-based nonlinear metric [20] have also been used. The distance function in kNN should amplify informative dimensions (features) of the sample space to provide high generalization power. This requires assignment of relevance weights to the informative dimensions as compared to the others carrying irrelevant or redundant information. Feature selection for optimal classification is always a quite challenging task [21]. Weights to different features in a dataset can be mainly assigned in two different ways. The first approach is to provide weights to all the features according to their priority and effectiveness in classification. One example of the above strategy is RELIEF [22–24]. A second strategy is to turn off the weights of the irrelevant and redundant dimensions [25,26]. Most of the feature selection methods follow this type of approach, where a subset of features from the original datasets is selected before any classification algorithm is applied [27]. These sparse feature selection methods, though can efficiently handle the problem of curse of dimensionality, may suffer from the problem of information loss. Random Subset Feature Selection (RSFS) [28] is an example of sparse feature selection method.

Other classical supervised feature weighting algorithms include mutual information based Minimum Redundancy and Maximum Relevance (mRMR) algorithm [29]; Local Fisher Discriminant Analysis (LFDA) [30], an extension of Fisher discriminant analysis [31,32]; RELIEFF [23], Iterative Relief(I-RELIEF) [33] and other well-known extensions of the RELIEF [22] family. LFDA is an efficient algorithm to handle the problem of multimodality. RELIEFF [23], the extended version of RELIEF [22] is robust and efficient to deal with incomplete and noisy data [34]. The principle of I-RELIEF is to treat the nearest neighbors and the identity of a pattern as hidden random variables [33]. In I-RELIEF, the feature weights are adjusted by multiple iterations using the Expectation-Maximization (EM) algorithm [35]. Iterative RELIEF-1 and Iterative RELIEF-2 [33,36,37] are the two state-of-the-art versions of I-RELIEF. These two algorithms can handle the problem of outlier, mislabeling and irrelevant features in a better way.

* Corresponding author.
E-mail address: aschowdhury@etce.jdvu.ac.in (A.S. Chowdhury).

http://dx.doi.org/10.1016/j.patcog.2017.01.018
Received 31 May 2016; Received in revised form 15 December 2016; Accepted 10 January 2017
Available online 17 January 2017

In contrast to most of the well-known batch feature selection and online feature selection algorithms, some recent online feature selection (OFS) methods use only a small and fixed number of attributes/features of training instances, which is very appropriate for expensive high dimensional datasets [38] as well as for the sequentially streaming datasets like online spam email detection system. To handle the challenge of accurate prediction using limited number of fixed active features, these online algorithms take the help of exhaustive learning. To avoid the over and under fitting they use different regularization processes. These online learning algorithms are divided into two major categories: (i) first-order learning, [39], and (ii) second-order learning [40]. Irrespective of first-order learning algorithms, the second-order online learning exploit the underlying structures between features in far more better way.[1] Some well-known second order learning algorithms are binary or multiclass Confidence-Weighted (CW) learning algorithm [42], Adaptive Regularization of Weight Vectors (AROW) [40], Soft Confidence Weighted algorithm (SCW) [43]. These methods are efficient only for online systems, where space-time-budget is a more important issue than the requirement of accurate prediction. Inspite of all of these advantages of the online learning algorithms, high accuracy with reduced time-complexity is much more desirable than any one of these individual requirements.

The major contribution of the work is to improve the performance of the kNN algorithm based on Analytic Hierarchy Process (AHP) [44], a well-known multi-criteria based decision making tool. AHP has been previously applied for uncertainty modeling [45], cancer classification [46], and, classifier combination [47]. To the best of our knowledge, there is no reported work where the performance of the kNN algorithm is shown to be boosted from the use of AHP. We apply AHP for obtaining the importance of the individual features in form of a set of weights. Consequently, a weighted distance is used to derive the set of k neighbors for classification. AHP considers i) a set of criteria to evaluate several alternatives and ii) a set of preferential weights for these different criteria to rank the alternatives. It has been observed that sometimes the judgment from AHP can be inconsistent due to manual weight selection for the alternatives corresponding to individual criterion as well as manual selection of the criteria preferences. In this paper, we map the alternatives to features in a dataset. To get rid of the manual weight selection for the alternatives corresponding to individual criterion, we automatically set them by employing (training) class-wise group-statistics. In particular, we design two criteria, one based on the group-mean and the other based on the group-standard deviation. To avoid the problem of manual preferential weight selection for the (two) criteria, we assign the weights by checking their interdependence through Granger causality [48]. Granger causality was initially introduced to identify causal relation between two time series based on temporal precedence. In [49–51], the authors have extended the theory to reveal the causal relationship between pattern-based information. This motivates us to provide a concept of causation in the present work. Here, the idea of causation to find the interdependence between different criteria eventually reveals that the criterion considered as cause is expected to provide additional information about the criterion considered as effect. So, the Granger causality in the present context is governed by meaningful criterion-based interaction.

The rest of the paper is organized in the following manner: In Section 2, we provide the theoretical foundations. In Section 3, we describe the proposed algorithm in details and in Section 4, we have analyzed its time-complexity. In Section 5, we present experimental results with comprehensive comparisons. Finally, in Section 6, we conclude our work with directions for future research.

2. Theoretical foundations

We start this section with discussing the theoretical aspects of the AHP [44]. We next provide the theoretical basis of the Granger causality test [48].

[1] Full details of the online learning algorithms can be found at URL http://libol.stevenhoi.org/LIBOL_manual.pdf in [41]

2.1. Analytic hierarchy process

Analytic Hierarchy Process (AHP) receives a set of inputs or alternatives for choosing the best option. These alternatives are compared on the basis of different criteria following the Saaty-scale as shown in Table 1. Multiple sub-criteria may also be introduced under the initial set of criteria for improvement of the judgment. A pairwise comparison among the different criteria are also made according to some relative weights proposed by the same Saaty-scale. In final stage, a logical ranking of the alternatives are obtained as the output of AHP. AHP is based on the following four axioms [44]:

Axiom 1. The decision-maker can provide pairwise comparisons $A(i, j)$ of two alternatives i and j corresponding to a criterion/sub-criterion on a reciprocal ratio scale, i.e., $A(j, i) = 1/A(i, j)$.

Axiom 2. The decision-maker never judges one alternative to be infinitely better than another corresponding to a criterion, i.e., $A(i, j) \neq \infty$.

Axiom 3. The decision problem can be formulated as a hierarchy.

Axiom 4. All criteria/sub-criteria which have some impact on the given problem, and all the relevant alternatives, are represented in the hierarchy in one go.

AHP is implemented through the following three major steps:

Step 1. Computation of the feature criteria matrix.

Step 2. Computation of the criteria preferential weights.

Step 3. Ranking of the alternatives.

We now discuss the above three steps. Let us assume that, d choices or alternatives are to be ranked using AHP on the basis of N_C criteria.

2.1.1. Computation of the feature criteria matrix

A criteria matrix C is created, which contains the weights for pairwise comparison of the alternatives corresponding to a particular criteria. The dimension of C is $d \times d$. An element $C(i, j)$ of this matrix represents the importance of the i^{th} alternative relative to the j^{th} alternative based on the given criterion. Following can be said about $C(i, j)$:

i) $C(i, j) > 1$, means the i^{th} alternative is more important than the j^{th} alternative.
ii) $C(i, j) < 1$, means the i^{th} alternative is less important than the j^{th} alternative.
iii) $C(i, j) = 1$ signifies both the alternatives have same preference.

The entries $C(i, j)$ and $C(j, i)$ satisfy the following constraint:

$$C(i, j).\ C(j, i) = 1 \tag{1}$$

and $C(i, i) = 1 \ \ \forall\, i$.

Once the matrix C is formed, the normalized pairwise comparison matrix C_n is constructed by making equal the sum of the entries of each column to 1. So, each element $C_n(i, j)$ of the matrix C_n is given by:

$$C_n(i, j) = \frac{C(i, j)}{\sum_{i=1}^{d} C(i, j)} \tag{2}$$

Finally, the criteria weight vector CV (an d-dimensional column vector) is built by averaging the entries on each row of C_n. So, the i^{th} element $CV(i)$ of this vector can be obtained using

$$CV(i) = \frac{\sum_{j=1}^{d} C_n(i, j)}{d} \tag{3}$$

In this manner, one can generate N_C criteria vectors CV_t from N_C criteria matrices C_t where $t = 1, \ldots, N_C$. These CV_t vectors are used to form a single feature criteria matrix F_C with dimension $d \times N_C$.

2.1.2. Computation of the preferential criteria weight vector

The dimension of the criteria preferential matrix P is $N_C \times N_C$. Each

Table 1
Definition and Explanation of Preference Weights according to Saaty.

Sl. No.	Preference Weights / level of importance	Definition	Explanation
1	1	Equally preferable	Two factors contribute equally to the objective
2	3	Moderate preferred	Experience and judgement slightly favour one over other.
3	5	Strongly preferred	Experience and judgement strongly favour one over the other.
4	7	Very strongly preferred	Experience and judgement very strongly favour one over the other.
5	9	Extremely preferred	The evidence favour one over the other is of the highest possible validity.
6	2,4,6,8	Intermediates values	Used to represent compromise between the preferences listed
7	Reciprocals	Reciprocals for inverse comparison	

entry $P(i, j)$ represents the preference of the i^{th} criterion with respect to the j^{th} criterion. The properties of $P(i, j)$ are similar to $C(i, j)$ in the previous subsection. We next generate the corresponding normalized matrix P_n by making equal the sum of the entries of each column to 1. So, each element $P_n(i, j)$ of the matrix P_n is given by:

$$P_n(i,j) = \frac{P(i,j)}{\sum_{i=1}^{N_C} P(i,j)} \tag{4}$$

Finally, the preferential criteria weight vector P_V (an N_C-dimensional column vector) is built by averaging the entries on each row of P_n. So, the i^{th} element $P_V(i)$ of this vector can be obtained using

$$P_V(i) = \frac{\sum_{j=1}^{N_C} P_n(i,j)}{N_C} \tag{5}$$

2.1.3. Ranking of the alternatives

Once the feature criteria matrix F_C and the preference vector P_V are obtained, AHP ranks the d alternatives by constructing a d-dimensional vector W using:

$$W = F_C \cdot P_V \tag{6}$$

The i^{th} element of W represents the score assigned by the AHP to the i^{th} alternative. The elements of the vector W can be sorted in decreasing order to rank the d alternatives.

2.2. Granger causality

Granger causality [48] is based on linear regression modeling of stochastic processes and is normally used to check whether one economic variable can help to forecast the other economic variable. It involves F-tests to check whether lagged information on a variable Y provides any significant statistical information about a variable X in the presence of lagged X. If that is not the case, then the null hypothesis "Y Granger-causes X" is rejected. Let X_t, Y_t be the two stationary time-series with zero means. The simple causal model with autoregressive lag length p, may be estimated by the following unrestricted equations of ordinary least squares:

$$X_t = c_t + \sum_{i=1}^{p} \alpha_i X_{t-i} + \sum_{i=1}^{p} \beta_i Y_{t-i} + u_t \tag{7}$$

According to null hypothesis H_0, Y does not cause X i.e., $\beta_1, \beta_2, \ldots, \beta_i = 0$. F-test is conducted to test null hypothesis by estimating the following restricted equation of ordinary least squares

$$X_t = c_t + \sum_{i=1}^{p} \gamma_i X_{t-i} + e_t \tag{8}$$

This model leads to two well-known alternative test statistics, the GrangerSargent and the GrangerWald test [52]. For the GrangerSargent test it needs to compare their respective sum of squared residuals

$$\left.\begin{aligned} RSS_u &= \sum_{t=1}^{T} \hat{u}_t^2 \\ RSS_e &= \sum_{t=1}^{T} \hat{e}_t^2 \end{aligned}\right\} \tag{9}$$

According to the GrangerSargent test [52] if the test statistics

$$GS = \frac{(RSS_e - RSS_u)/p}{RSS_u/(T - 2p - 1)} \sim F_{T-2p-1} \tag{10}$$

is greater than the specified critical value, then the null hypothesis that Y does not Granger-cause X is rejected. An asymptotically equivalent test is the GrangerWald [52] test, which is defined as

$$GW = \frac{T(RSS_e - RSS_u)}{RSS_u} \sim \chi^2(p) \tag{11}$$

The lag length p for the Granger causality is chosen from the Bayes information criteria(BIC), which is given by

$$BIC(p) = \ln\left(\frac{RSS_u}{T}\right) + (p+1)\frac{\ln T}{T} \tag{12}$$

The result is

$$\hat{p}^{BIC} \underset{p}{\longrightarrow} p! \tag{13}$$

i.e. the value of p that minimizes the BIC is a consistent estimator of the true lag length. In the present context, the instances of time domain X_t and Y_t used in Eqs. (7 to 13) may be assumed to be the criteria instances in spatial domain.

3. Proposed method

Let us assume that n number of well-defined and identically distributed sample points having d mutually independent features are chosen for classification. Out of n number of samples, a total of N number of training points are grouped into M classes $CL_1, CL_2, \ldots, CL_M$. The class CL_j contains N_j number of points, where, $j = 1, 2, \ldots, M$ and $\sum_{j=1}^{M} N_j = N$. These training points are expressed by a $N \times d$ matrix X. The goal is to classify a set of d-dimensional N_t number of test points, represented by a $N_t \times d$ matrix x.

Basically AHP accounts for deriving priority vectors which combines the eigenvalue concept with a constrained optimization based approach [53]. But the eigenvalue approaches of AHP [54] i.e., the right (REM) and left (LEM) eigenvalue methods have higher ranking contradictions with increasing dimensions and inconsistencies in comparisons [54]. Mean normalized value (MNV) have less ranking contradictions with respect to the right eigen method (REM) [54]. As a better approximate model of eigenvalue methods we have used MNV approach [54] to facilitate the incorporation of Granger Causality [48] for automatic prioritization of different criteria with respect to one another in AHP. The null hypothesis set in the Granger causality works as an additional constraint in the optimization algorithm. This ensures an enhancement in terms of robustness, interpretability and scalability in the drawn inference under a given set of conditions and frameworks [55].

We present two algorithms to describe our AHP-kNN method. Algorithm 1 shows the main steps of AHP-kNN and invokes algorithm 2 for the computation of the feature weights W. In the first few steps of the algorithm 1, data normalization is performed. The training data and their class information is fed to algorithm 2. In algorithm 2, class-wise group-statistics are obtained for each individual feature to construct two $d \times d$ criteria

Table 2
Brief characteristics of the datasets.

Dataset	No. of instances (n)	No. of attributes (d) without class	No. of classes (M)
Iris	150	4	3
Wine	178	13	3
Glass	214	9	6
Pima-Diabetes	768	8	2
Breast	683	10	2
Sonar	208	60	2
Ionosphere[a]	351	33	2
Vehicle	846	18	4
Wdbc	569	31	2
Spectf	267	44	2
Musk1	476	166	2
Breast-Tissue[a]	106	9	6
Parkinson[a]	195	22	2
Segmentation	210	18	7
Ecoli[a]	336	7	8
Balance-scale	625	4	3

[a] The 1^{st} column of the dataset contain serial numbers which has not been considered as an attribute in present work.

Table 3
Brief characteristics of the large datasets.

Dataset	No. of instances (n)	No. of attributes (d) without class	No. of classes (M)
svmguide3	1243	21	2
Spambase	4601	57	2
Magic04	19020	10	2
Segment	2310	19	7
Waveform	5000	21	3
USPS	9298	256	10
Yale	165	1024	15
ORL	400	1024	40

matrices C_1 and C_2. The elements of the criteria matrices represent pairwise comparison of any two features for the respective criterion. Two criteria are designed to achieve good classification. The first criterion is based on group-means. Let X_l be the l^{th} training point in the i^{th} class, $l = 1, \ldots, N_i$. X_{lj} denotes the j^{th} feature of X_l. The group mean of the i^{th} class for the j^{th} feature can then be defined as

$$\mu_{ij} = \frac{1}{N_i}\sum_{l=1}^{N_i} X_{lj} \tag{14}$$

A particular feature should be given high importance if the the group-means of different classes for that feature are well separated. So, in order to assign weights according to the first criteria, we sum the class-wise group-mean differences for each feature. Higher the value of this sum, the more should be the preference given to that feature. Let μ_{ij} be the group-mean of the i^{th} class and μ_{lj} be the group-mean of the l^{th} class for the j^{th} feature. Then, the absolute group-mean difference of all M classes for the j^{th} feature can be expressed as

$$D_j = \sum_{i=1,l=1,i\neq l}^{M} \left|\mu_{ij} - \mu_{lj}\right| \tag{15}$$

The elements of a criterion matrix denote the pairwise comparison of features (based on that criterion). So, the first criteria matrix C_1 based on group-means is given by

$$C_1 \Rightarrow \begin{bmatrix} 1 & D_1/D_2 & \cdots & D_1/D_d \\ D_2/D_1 & 1 & \cdots & D_1/D_d \\ \vdots & \vdots & \ddots & \vdots \\ D_d/D_1 & D_d/D_2 & \cdots & 1 \end{bmatrix} \tag{16}$$

The second criterion is developed from group-standard deviation. The group standard deviation of the i^{th} class for the j^{th} feature can be obtained in the following manner

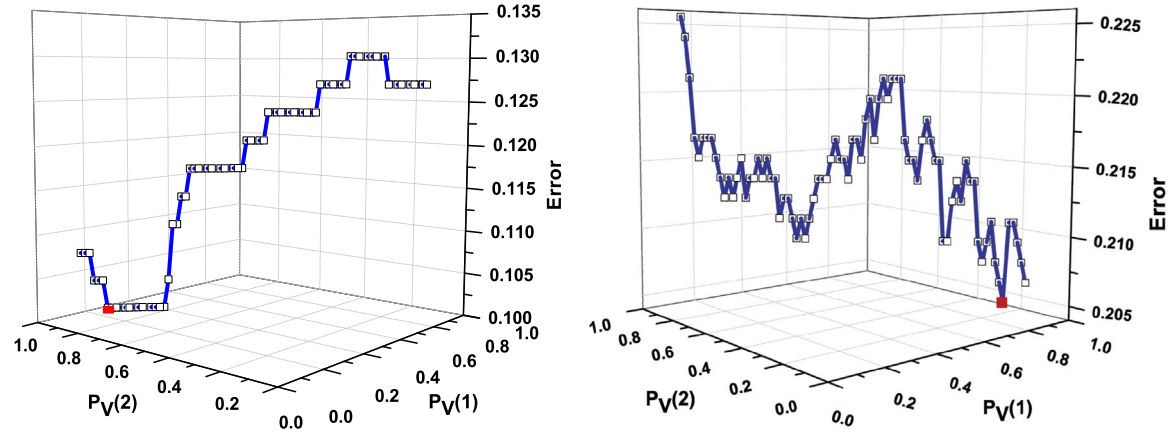
$$\sigma_{ij} = \left(\frac{1}{N_i}\sum_{l=1}^{N_i} X_{lj}^2 - \left(\frac{1}{N_i}\sum_{l=1}^{N_i} X_{lj}\right)^2\right)^{1/2} \tag{17}$$

Preference for a particular feature will be high when (i) the group-standard deviations and hence their mean (over all classes) is low and (ii) the standard deviation of the group-standard deviations (over all classes) is high. Let the standard deviation of the group-standard deviations for the j^{th} feature is denoted by σ_j. Then, the standard deviation based weight for the j^{th} feature is given by

$$S_j = \frac{\sigma_j}{\frac{\sum_{i=1}^{M}\sigma_{ij}}{M}} \tag{18}$$

Thus, the second criteria matrix C_2 based on group-standard deviations is written as

$$C_2 \Rightarrow \begin{bmatrix} 1 & S_1/S_2 & \cdots & S_1/S_d \\ S_2/S_1 & 1 & \cdots & S_2/S_d \\ \vdots & \vdots & \ddots & \vdots \\ S_d/S_1 & S_d/S_2 & \cdots & 1 \end{bmatrix} \tag{19}$$



(a) Ionosphere (b) Pima-Diabetes

Fig. 1. Variation of classification error with preferential weights for the two criteria vectors. The red dots mark the minimum error and the corresponding weights are the same as obtained from the Granger causality test, (a) Ionosphere, (b) Pima-Diabetes.

Algorithm 1. AHP-kNN algorithm main block.

procedure AHP-kNN(X,x,Y)
  $X, x \leftarrow$ *Normalized training & test samples*
  $Y \leftarrow$ *Classes of all training samples*
  $[N, d] \leftarrow size(X)$ ▷ N training samples in d-dimension
  $U \leftarrow unique(Y)$ ▷ Unique(distinct) training classes
  $M \leftarrow count(U_c)$ ▷ No. of unique classes
  $W \leftarrow$ MODAHP(X, Y, U_c, M)
  $[N_t, d] \leftarrow size(x)$ ▷ N_t test samples in d-dimension
  for $l \leftarrow 1, N_t$ **do**
    $D_w(t) \leftarrow dist(x(l), X(t), W) \forall t \in [1, N]$
    $[D_s, I_s] \leftarrow sort(D_w)$ ▷ Sorted Distances & Indices
    $Y_s \leftarrow Y(I_s)$ ▷ Sorted Classes
  ▷ **Similarity for k-nearest neighbors of l^{th} test point**
    $Sim_s(K) \leftarrow 1/D_s(K) \forall K \in [1, k]$
    for $i \leftarrow 1, M$ **do**
      $Index \leftarrow find(U_c(i) = Y_s)$
      $n_c \leftarrow count(Index)$
      $Sim \leftarrow Sim + Sim_s(Index(c)) \forall c \in [1, n_c]$
      $Score(i) \leftarrow Sim$
    end for
    $[Sc_{max}, I_l] \leftarrow max(Score)$ ▷ Max. score & Index
    $y_{pred}(l) \leftarrow U_c(I_l)$ ▷ Predicted test class
  end for
  return y_{pred} ▷ Predicted test classes
end procedure

After determination of the criteria matrices, normalization and row-wise averaging are performed to obtain the criteria vectors (CV_1 and CV_2). The two criteria vectors are combined to form the feature criteria matrix F_C. Preferential weights to the individual criterion are assigned on the basis of the Granger-causality and variances of the criteria vectors (s_1 and s_2). The Granger causality test captures mutual significance of the two criteria. If the test indicates that CV_1 causes CV_2 more than CV_2 causes CV_1, then CV_2 should be under-emphasized with respect to CV_1 by the ratio s_1/s_2 and vice-versa. The criteria preference matrix P is constructed from the relative weights/preferences of the two criteria vectors. Then, a preference vector P_V is obtained from the normalized P matrix, denoted by P_n. F_C and P_V are used to obtain the feature weight vector W. These weights, as outputted by algorithm 2, are used in the distance function in algorithm 1 for the final classification. Note that any standard distance function like Euclidean, Cityblock can be used. We choose the Cityblock distance.

4. Time-complexity analysis

For n number of samples B-fold partitioning have been used for the purpose of cross-validation. Let N be the number of training samples. Let k and M represents the number of nearest neighborhood points and the number of classes assigned for classification of patterns respectively. We now present the detailed worst-case time-complexity analysis of our proposed algorithm.

- Complexity of normalization of n samples: $O(n)$.
  1. Complexity for weight adjustment using MOD-AHP:
     Let N_i is the number of points within i^{th} group and M is the number of classes of the dataset.
  2. Complexity for group mean calculation: $O(MN_i) \approx O(N)$.
  3. Complexity for group standard deviation calculation: $O(MN_i \log N_i)$
  4. Complexity for standard deviation of group standard deviation calculation: $O(M \log M)$
  5. Complexity for overall standard deviation:$O(N \log N)$
  6. Complexity for mean-difference calculation: $O(M^2)$
  7. Complexity of each criteria matrix formation by pair-wise comparison of features:$O(d^2)$
     Hence for each N_C criteria matrices time complexity is $O(d^2)$. So, the total complexity is $O(d^2)$
  8. For normalization and criteria vector formation from the criteria matrices the time-complexity is $O(d)+O(d) \approx O(d)$

Algorithm 2. Modified AHP algorithm.

procedure MODAHP(X,Y,U_c,M)

▷ **Feature attributes for Criteria matrices**

$\mu_{ij} \leftarrow GroupMean(X)\ \forall\ U_c \in Y$ ▷ Eq. 14

$D_j \leftarrow \sum_{i,l} \left|\mu_{ij} - \mu_{lj}\right|\ \forall\ i,l \in M\ \wedge\ i \neq l$ ▷ Eq. 15

$\sigma_{ij} \leftarrow GroupStd(X_j)\ \forall\ U_c \in Y$ ▷ Eq. 17

▷ **Standard deviation of group standard deviations**

$\sigma_j \leftarrow std(\sigma_{ij})$

$S_j \leftarrow \sigma_j / (\sum_i \sigma_{ij}/M)\ \forall\ i \in M$ ▷ Eq.18

$N_C \leftarrow 2$ ▷ No. of criterion

▷ **Formation of N_C No. of $(d \times d)$ criteria matrices**

$C_1 \leftarrow \mathbb{1}, C_2 \leftarrow \mathbb{1}$ ▷ Initialize criteria matrices

for $i \leftarrow 1, d-1$ **do**

  for $j \leftarrow i+1, d$ **do**

    $C_1(i,j) \leftarrow D(i)/D(j)$ ▷ Eq. 16

    $C_1(j,i) \leftarrow 1/C_1(i,j)$ ▷ Eq. 16

    $C_2(i,j) \leftarrow S(i)/S(j)$ ▷ Eq. 19

    $C_2(j,i) \leftarrow 1/C_2(i,j)$ ▷ Eq. 19

  end for

end for

▷ **Normalization of C_1, C_2 matrices of Eq. 2**

$C_{tN}(i,j) \leftarrow C_t(i,j) / \sum_{i=1}^{d} C_t(i,j) \forall j \in [1,d] \forall t \in [1, N_C]$

▷ **N_C no. of $(d \times 1)$ criteria vectors CV_t of Eq. 3**

$CV_t(i) \leftarrow \sum_{j=1}^{d} C_{tN}(i,j)/d\ \forall i \in [1,d] \forall t \in [1, N_C]$

▷ **Formation of single Feature-criteria matrix**

$F_C \leftarrow [CV_1\ CV_2]$ ▷ F_C has dimensions $d \times N_C$

▷ **Criteria preferences by Granger Causality** G_C

$s_t \leftarrow (std(CV_t))^2\ \forall\ t \in [1, N_C]$ ▷ Criteria variances

$\alpha \leftarrow 0.01$ ▷ significance level of F-statistics

$p_{max} \leftarrow 1$ ▷ maximum lag for G_C

▷ F_i**: F-Statistics (GS in Eq. 10),** cv_i**: Critical Value**

$[F_i, cv_i] \leftarrow G_C(CV_i, CV_j, \alpha, p_{max}) \forall i, j \in [1, N_C] \wedge i \neq j$

▷ $(N_C \times N_C)$ **criteria-preference matrix formation**

$P \leftarrow \mathbb{1}$ ▷ Initialization of criteria preference matrix

for $i \leftarrow 1, N_C - 1$ **do**

  for $j \leftarrow i + 1, N_C$ **do**

    if $s_j > s_i$ **then**

      $W_P \leftarrow s_i / s_j$

    else

      $W_P \leftarrow s_j / s_i$

    end if

    if $F_i > F_j$ & $F_i > cv_i$ **then**

      $P(i, j) \leftarrow W_P, P(j, i) \leftarrow 1/W_P$

    else

      $P(j, i) \leftarrow W_P, P(i, j) \leftarrow 1/W_P$

    end if

  end for

end for

▷ **Normalization of** P **to get** P_n **in Eq. 4**

$P_n(i, j) \leftarrow P(i, j) / \sum_{i=1}^{N_C} P(i, j)\ \forall\ j \in [1, N_C]$

▷ $(N_C \times 1)$ **Preference Vector** P_V **for Eq. 5**

$P_V(i) \leftarrow \sum_{j=1}^{N_C} P_n(i, j) / N_C\ \forall\ i \in [1, N_C]$

▷ **Weight for different alternatives**

$W \leftarrow F_C \cdot P_V$ ▷ $d \times 1$ Weight Vector W for Eq. 6

return W

end procedure

Table 4
Comparison of average accuracies of AHP-kNN with kNN using different distance metrics (wins in **bold**). For AHP-kNN Standard deviations have been given within bracket. We have performed 10-Fold Cross-validation with $k = [\sqrt{N}]$.

Dataset	Mahalanobis [58]	Xing [16]	LMNN [19]	ITML [17]	KRCA [18]	IGML [20]	KIGML [20]	AHP-kNN
Wine	92.5	89.2	95.9	92.3	95.4	95.0	93.9	**96.92(3.88)**
Glass	65.1	58.3	65.1	63.8	63.1	64.2	66.7	**72.94(7.26)**
Pima-Diabetes	72.2	72.1	72.9	72.2	72.2	72.4	72.2	**76.72(4.12)**
Sonar	71.1	71.1	79.7	71.7	73.5	71.9	85.4	80.36(7.88)
Ionosphere	81.6	89.7	85.0	88.9	82.8	83.4	85.8	**88.11(5.02)**
Average	76.5	76.1	79.7	77.8	77.4	77.4	80.8	**83.01(5.63)**

Table 5
Comparison of average accuracies and average standard deviations of AHP-kNN with non-sparse feature selection methods (wins in **bold**).

Dataset	10-Fold Cross-validation with $k = [\sqrt{N}]$				
	RELIEFF [24]	MLMI [59]	LSMI [60]	kNN [9]	AHP-kNN
Iris	**95.16**(5.27)	94.89(5.38)	94.89(5.38)	94.63(5.71)	95.10(**5.24**)
Wine	97.08(3.78)	97.53(3.53)	96.62(4.02)	**97.70(3.52)**	96.92(3.88)
Glass	71.16(8.89)	67.91(8.21)	69.11(8.67)	71.90(8.20)	**72.94(7.26)**
Pima-Diabetes	75.99(4.10)	75.09(**3.89**)	76.29(4.54)	75.57(4.02)	**76.72**(4.12)
Breast	96.12(2.26)	96.29(**2.14**)	96.18(2.34)	96.34(2.15)	**96.41**(2.24)
Sonar	80.09(8.69)	76.70(8.24)	77.48(8.85)	79.88(**7.45**)	**80.36**(7.88)
Ionosphere[a]	85.31(5.65)	84.00(5.66)	85.06(5.74)	84.43(5.67)	**88.11(5.02)**
Vehicle	69.57(**3.33**)	69.46(4.20)	69.70(3.78)	70.14(3.84)	**70.32**(3.59)
Wdbc	**96.03(2.56)**	94.99(2.69)	95.10(2.97)	95.36(2.74)	95.08(2.78)
Spectf	**79.14**(6.66)	77.62(6.41)	78.17(7.20)	77.42(**5.75**)	78.51(7.41)
Musk1	82.86(5.82)	79.40(6.69)	78.42(6.78)	**83.16**(5.80)	82.44(**5.62**)
Breast-Tissue[a]	68.80(11.96)	60.87(11.69)	65.98(**11.28**)	66.73(11.59)	**68.82**(11.30)
Parkinson[a]	**92.15(6.27)**	87.64(**6.27**)	87.94(6.83)	91.04(6.79)	91.70(6.35)
Segmentation	86.57(7.24)	84.00(7.11)	83.38(6.92)	86.90(6.82)	**87.48(6.75)**
Ecoli[a]	77.58(27.82)	84.06(5.90)	85.19(5.72)	**86.23**(5.61)	85.93(**5.64**)
Average	83.57(7.35)	82.03(5.87)	82.63(6.07)	83.83(5.71)	**84.46(5.64)**

[a] The 1^{st} column of the dataset contain serial numbers which has not been considered as an attribute in present work.

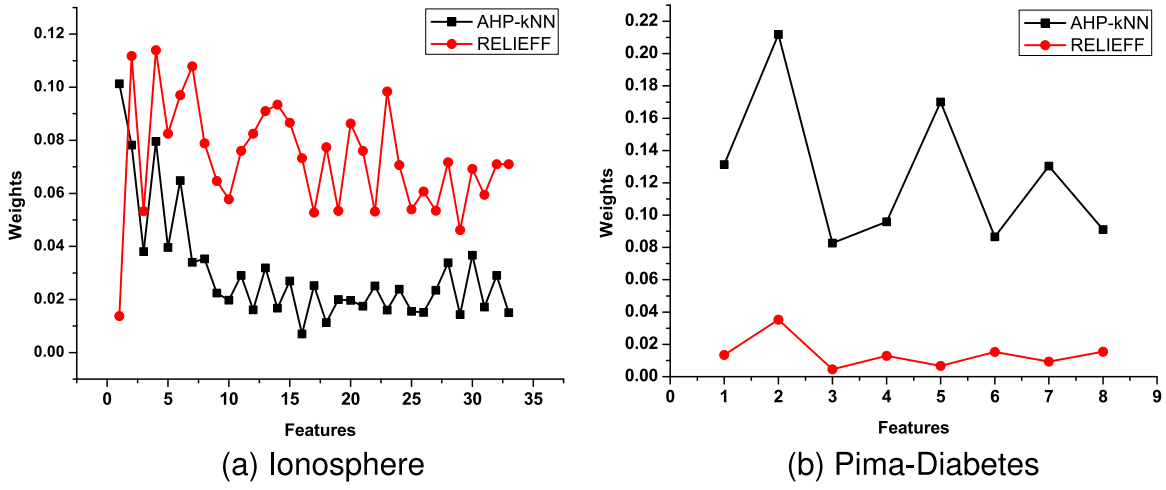


Fig. 2. Feature Weights comparison for RELIEFF [24] and AHP-kNN, (a) Ionosphere, (b) Pima-Diabetes.

9. Complexity for preference matrix formation from criteria vector variance and using Granger-causality: $O(d\log d) + O(d^2)$
10. Complexity for criteria preference vector formation from criteria preference matrix: $O(N_C) + O(N_C) \approx O(N_C)$
11. Now, the time complexity for assignment of weights:$O(1)$

So, the total time complexity for calculation of feature weights:
$O(N) + O(MN_i\log N_i) + O(M\log M) + O(N\log N) + O(M^2)$
$+O(d^2) + O(d) + O(d^2 + d\log d) + O(N_C) + O(1)$
$\approx O(N\log N) + O(d^2)$ since $N \gg N_i, k, N_C, M$

- Now, for each test sample,
  1. Complexity of calculating distance for N training samples: $O(N)$.
  2. Complexity of sorting N training samples and calculating the similarity of k samples: $O(NlogN) + O(k)$.
  3. Complexity of score calculation: $O(kN_C)$. Total complexity of kNN classification for each test sample: $O(N) + O(NlogN) + O(k) + O(kN_C)$.
- So, total complexity for the modified kNN algorithm taking the normalization (step 1) and AHP weight calculation (step 2) parts into account is:
$O(n) + (O(N\log N) + O(d^2)) + (O(N) + O(NlogN) + O(k) + O(kN_C))$
Since, $k, N_C \ll n, N$, the above complexity becomes:
$O(n) + O(N\log N) + O(d^2) \approx O(N\log N) + O(d^2)$. Thus, if $N \gg d$ then the time complexity is $O(N\log N)$, otherwise, if d≫N then the time-complexity

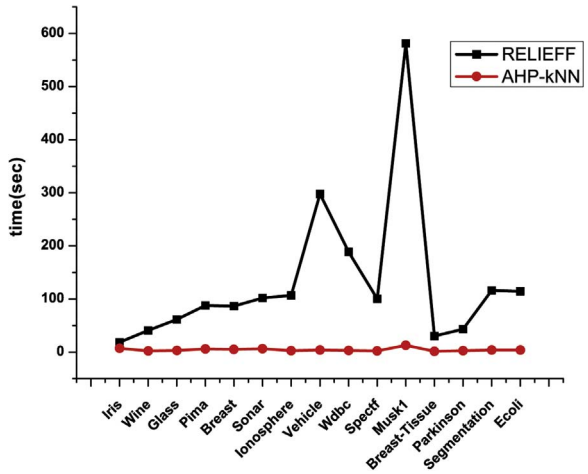


Fig. 3. Comparison of elapsed time for RELIEFF [24] & AHP-kNN for $k = [\sqrt{N}]$ and 10-fold cross validation.

is $O(d^2)$. The dependency on the dimension d is due to the calculation of weight by AHP. Otherwise, the time-complexity of our proposed kNN is as same as standard kNN i.e. $O(N \log N)$.

5. Experimental results

The proposed AHP induced kNN classification has been applied on the datasets of UCI machine learning repository [56] and on the datasets in [57]. We have also taken some data from LIBSVM website. For detailed information regarding total instances, dimension and the number of classes of the datasets, please see Table 2, 3 respectively. The classification performance of our method is compared with (i) kNN using different distance metrics, (ii) kNN using different feature selection strategies and with (iii) some state-of-the-art feature selection algorithms. We have also applied our method on some large datasets mentioned in 3. We also indicate that the proposed approach performs well on high dimensional face recognition and handwriting recognition datasets.

We have experimentally verified the effectiveness of the Granger causality based preferential weight selection for the two criteria. As a part of this experiment, we vary the two preferential weights with the classification error and the results for two datasets are shown in Fig. 1. It has been observed that the minimum error for both the datasets (marked by the red dots) are achieved for the same values of preferential weights as obtained from the Granger causality test.

In Table 4, we compare our results using a value of $k = [\sqrt{N}]$ with seven existing distance metrics, many of which also employ learning. The distance metrics include Mahalanobis distance [58], Xing distance [16], Large Margin Nearest Neighbor (LMNN)-based distance [19], Information Theoretic Metric Learning (ITML)-based distance [17], Kernel Relevant Component Analysis (KRCA) distance [18], Information Geometric Metric Learning (IGML)-based distance [20] and Kernel Information Geometric Metric Learning (KIGML)-based distance [20]. In four out of the five datasets shown (for which the results of the other distances are available), our method turns out to be the best. We also obtain the best average accuracy of 83.05.

In Table 5, we show the superiority of our method with three feature selection approaches, namely, RELIEFF [24,61], Maximum Likelihood Mutual Information(MLMI) [59] and Least Square Mutual Information(LSMI) [60] which do not apply any form of dimensionality reduction or are non-sparse in nature. We also include a previous work of ours on kNN for comparison where all the features in a dataset were used for classification [9]. For a fair comparison, we have used 10-fold cross validation with $k = [\sqrt{N}]$. The table indicates that AHP-kNN wins in 8 out of 15 datasets and produce the best average accuracy of 84.46 among all

Table 7
Comparison of AHP and SFS framework using same mean and standard deviation based criteria.(10-Fold Crossvalidation with $k=[\sqrt{N}]$).

Dataset	Criteria based on Mean and Standard Deviation	
	SFS-kNN Framework	AHP-kNN Framework
Iris	94.76(5.59)	95.10(5.24)
Wine	93.26(5.34)	96.92(3.88)
Glass	71.90(8.20)	72.94(7.26)
Pima-Diabetes	75.57(4.02)	76.72(4.12)
Breast	96.34(2.15)	96.41(2.24)
Sonar	68.79(9.56)	80.36(7.88)
Ionosphere[a]	88.71(4.67)	88.11(5.02)
Vehicle	68.83(3.52)	70.32(3.59)
Wdbc	93.13(3.42)	95.08(2.78)
Spectf	78.44(7.56)	78.51(7.41)
Musk1	79.60(5.89)	82.44(5.62)
Breast-Tissue[a]	66.73(11.59)	68.82(11.30)
Parkinson[a]	85.76(7.96)	91.70(6.35)
Segmentation	79.52(8.22)	87.48(6.75)
Ecoli[a]	86.23(5.61)	85.93(5.14)
Average	81.84(6.22)	**84.46 (5.64)**

[a] The 1^{st} column of the dataset contain serial numbers which has not been considered as an attribute in present work.

Table 6
Comparison of average accuracies of AHP-kNN with sparse feature selection methods (wins in **bold**).

Dataset	5-Fold Stratified Cross-validation with ten random seeds and k=5				
	RSFS [62]	SFS [62]	SFFS [62]	SFSW-MOEAD [63]	AHP kNN
Iris	96.39(3.02)	95.52(3.31)	95.53(3.29)	96.27(3.09)	94.88(3.19)
Wine	96.66(2.75)	95.47(4.75)	95.25(4.74)	96.53(2.99)	**97.38(2.41)**
Glass	63.62(7.40)	62.39(9.82)	63.03(6.27)	66.26(6.26)	**70.40(5.13)**
Pima-Diabetes	69.90(3.56)	70.63(**2.92**)	69.79(3.49)	71.30(3.16)	**73.53**(2.95)
Breast	**96.98(1.44)**	96.43(1.53)	96.51(1.67)	96.27(1.62)	96.75(1.51)
Sonar	81.99(**5.92**)	78.17(5.71)	76.57(6.02)	79.48(7.74)	**82.37**(6.36)
Ionosphere[a]	**90.01**(3.41)	89.43(2.76)	89.27(2.77)	87.18(3.36)	89.94(**2.44**)
Vehicle	70.55(3.34)	71.79(3.78)	70.90(4.98)	67.27(3.50)	**71.74(3.33)**
Wdbc	94.70(1.78)	94.20(2.23)	94.29(4.04)	93.48(2.06)	**96.26(1.77)**
Spectf	68.28(5.96)	69.88(6.19)	71.54(7.74)	**76.00**(6.14)	74.77(**4.48**)
Musk1	85.50(3.76)	80.55(5.54)	78.82(7.16)	81.28(4.55)	**86.01(3.48)**
Breast-Tissue[a]	61.50(9.21)	62.03(8.30)	61.20(**8.20**)	61.56(12.48)	**67.57**(9.62)
Parkinson[a]	88.23(4.97)	86.51(4.74)	86.80(5.15)	89.90(5.10)	**90.93(4.52)**
Segmentation	87.86(**3.92**)	86.09(4.59)	84.46(7.64)	84.32(4.84)	**89.10**(3.94)
Ecoli[a]	80.76(**3.89**)	79.87(4.38)	79.77(4.86)	71.73(5.01)	**85.37**(4.07)
Average	82.20(4.29)	81.26(4.70)	80.91(5.20)	81.26(4.79)	**84.47(3.95)**

[a] The 1^{st} column of the dataset contain serial numbers which has not been considered as an attribute in present work.

Table 8
Comparison of average accuracies of AHP-kNN with some state-of-the-art feature selection methods (wins in **bold**).

Dataset	50 random runs with holdout 0.3 and k=[$\sqrt{N}$]			
	I-RELIEF-1 [33]	**I-RELIEF-2** [33]	**mRMR** [29]	**AHP-kNN**
Iris-corrected	94.22 (2.80)	94.40 (2.88)	**95.29**(2.72)	94.40 (**2.59**)
Wine	96.57 (2.04)	**97.13**(**1.95**)	96.15 (2.25)	96.49 (2.09)
Glass	67.56 (4.68)	67.88 (**4.33**)	65.41 (5.42)	**72.63**(4.92)
Pima-Diabetes	76.26 (2.17)	76.26 (2.17)	76.23 (2.20)	75.96(**1.94**)
Breast	95.97 (**1.28**)	95.96 (1.27)	96.15 (1.19)	**96.25**(1.32)
Sonar	73.65 (**3.86**)	73.65 (3.86)	76.32 (4.57)	**79.29**(4.95)
Ionosphere (excl. Serial No.)	85.68 (2.92)	85.64 (2.90)	87.41 (3.15)	**88.27**(**2.66**)
Vehicle	70.39 (2.49)	**70.88**(2.56)	63.08 (2.33)	70.43(**2.10**)
Wdbc	95.47 (1.47)	95.49 (1.49)	**96.20**(**1.29**)	94.74(1.33)
Spectf	**77.83**(3.77)	77.83(**3.74**)	77.53(4.11)	77.70 (4.17)
Musk1	79.87 (3.23)	79.87(3.23)	79.15 (3.44)	**81.00**(**2.70**)
Breast-Tissue (excl. Serial No.)	64.71 (6.56)	62.45(6.34)	61.48 (**6.06**)	**66.32** (7.52)
Parkinson (excl. Serial No.)	87.07 (3.85)	87.07(3.84)	87.24(4.47)	**90.03**(**3.57**)
Segmentation	82.51 (3.37)	84.44(3.13)	83.71(4.19)	**87.56**(**3.00**)
BalanceScale	88.84 (1.06)	88.92(1.10)	67.11(2.08)	**89.45**(**1.28**)
Average	82.44 (3.04)	82.52(**2.99**)	80.56(3.30)	**84.03**(**3.08**)

The 1st column of the dataset contain serial numbers which has not been considered as an attribute in present work.

Table 9
Comparison of average accuracies of AHP-kNN with some state-of-the-art feature selection methods along-with some online feature selection algorithm for large datasets with binary classes(wins in **bold**).

Datasets with	10 random runs with holdout 0.3 and k=5						
Binary Class	**CW** [42]	**AROW** [64]	**SCW** [43]	**SCW2** [43]	**mRMR** [29,37]	**LFDA** [30,37]	**AHP-kNN**
svmguide3	70.1 (1.1)	77.8 (**0.4**)	79.4 (**0.4**)	78.8 (0.9)	79.1 (1.3)	80 (1.8)	**80.8** (1.3)
Spambase	86.7 (0.4)	90.6 (0.4)	89.4 (**0.2**)	88.8 (0.7)	91.4 (0.7)	92.8 (0.7)	**93.3** (0.4)
Magic04	66.3 (0.4)	78.4 (**0.1**)	79.0 (0.2)	77.0 (1.3)	78.3 (0.6)	83.9 (0.4)	**84.3** (0.4)
Average	74.4 (0.6)	82.3 (**0.3**)	82.6 (**0.3**)	81.5 (1.0)	82.9 (0.8)	85.5 (1.0)	**86.1** (0.7)

competing methods. In Fig. 2, the weights predicted by RELIEFF [24] and our AHP-kNN have been sketched for the Ionosphere and the Pima-diabetes datasets. The two plots in this figure clearly reveal that the variation in feature weights is much more pronounced for AHP-kNN leading to more accurate classification as compared to RELIEFF [24]. In Fig. 3, total time taken by our method for each dataset is compared with RELIEFF [24] on an Intel(R) Core(TM) $i5-3230M$, processor with 2.60 GHz clock speed and 4 GB RAM. The average elapsed time taken by our method for 15 datasets is only 4.4068 s for hundred runs, which is much better than the elapsed time of 131.6 s of the RELIEFF [24] algorithm for the same number of runs.

In Table 6, we compare our method with four different feature selection strategies, namely, Random Subset Feature Selection(RSFS) [28,65], Sequential Forward Selection(SFS) [62,65], Sequential Floating Forward Selection (SFFS) [65,66], and, Simultaneous Feature Selection & Weighting decomposition based Multi-Objective Evolutionary Algorithm (SFSW-MOEA/D) [63], all of which employ dimensionality reduction or use sparse feature weights. To have the same basis for comparison, in this case, we use k=5 and 5-fold stratified cross validation for 10 random seeds. As can be seen from this table, our method once again wins in 8 out of the 15 datasets and outperforms all the 4 methods by yielding the mean classification accuracy value of 84.47 %.

In Table 7, we have particularly established the effectiveness of our AHP based feature weighting strategy keeping the set of criteria (i.e., mean and standard deviation) same. So, we compare AHP and SFS in this regard. The table shows that we obtain an accuracy of only 81.84% (s.d.:6.22) from the SFS framework. However, the proposed AHP-kNN algorithm with the same set of criteria yields a much better accuracy of 84.46% (s.d.:5.64).

In Table 8, our method has been compared with some state-of-the art algorithms using holdout cross-validation for 50 random runs. Our accuracy 84.03 % clearly outperforms the other state-of-the-art algorithms mentioned in the table. Application on the large datasets have also been compared in Table 9–11 using some online learning algorithms mentioned in [41]. The second order learning algorithms used for binary or multiclass classification are Confidence-Weighted (CW) learning algorithm [42], Adaptive Regularization of Weight Vectors (AROW) [40], Soft Confidence Weighted algorithms (SCW) found in [43]. We have also compared our results with two state-of-the-art algorithms mRMR [29] and LFDA [30]. Table 8 comprises the results for the datasets with binary classes, whereas, Table 10 comprises the results for the datasets with multiple classes. Same holdout value 0.3 as like as 9 has also been used in these tables with 10 random runs.

Throughout Table 8–11 our accuracies clearly ensure the effectiveness of our proposed method in case of large datasets also. In Table 11 we have compared our method with some online learning algorithm on the basis of two high dimensional datasets. The average result of our proposed method for these high dimensional datasets is 78.1 %, which, clearly outperforms the results of other online algorithms as evident from Table 11.

In case of CW, AROW, SCW, SCW1, SCW2 the trade off between a regularization term and a loss term is C. The default value of C=1. For the family of confidence-weighted learning algorithms, η is a parameter used in defining a key parameter ϕ of the loss function, i.e., $\phi = \phi^{-1}(\eta)$, where ϕ is the cumulative function of the normal distribution. The parameter a is typically used for initializing the covariance matrix in the second order algorithms, i.e, $\sum = a*I$, where I is an identity matrix. For most cases, parameter a is not too sensitive and typically fixed to 1 [57]. For CW the default value of η=0.70, for SCW and SCW2 the default values of η are respectively 0.75 and 0.90. For M-CW a=1, η=0.75, for AROW C=1 and a=1, for M-SCW and M-SCW2 both C=1, a=1 and η=0.75.

The proposed AHP-kNN method has been further applied on a face recognition dataset with 1209 features[2] and a handwriting recognition dataset with 400 features.[3] These two datasets are also used in [67]. In

[2] Full details on the data set can be found at URL http://www.uk.research.att.com/facedatabase.html

[3] Full details on the data set can be found at URL http://www.ics.uci.edu/mlearn/databases/letter-recognition/letter-recognition.names

Table 10
Comparison of average accuracies of AHP-kNN with some state-of-the-art feature selection methods along-with some online feature selection algorithm for large datasets with multiple classes(wins in **bold**).

Datasets with	**10 random runs with holdout 0.3 and k=5**						
Multiple Class	**M-CW** [42]	**M-AROW** [64]	**M-SCW1** [43]	**M-SCW2** [43]	**mRMR** [29,37]	**LFDA** [30,37]	**AHP-kNN**
Segment	87.8 (0.5)	89.1 (0.7)	91.2 (0.5)	90.7 (0.5)	95.6 (0.7)	95.0 (0.8)	**97.0** (**0.4**)
Waveform	79.9 (0.3)	**84.8** (**0.2**)	83.9 (0.3)	84.6 (0.3)	78.6 (1.2)	83.0 (0.9)	82.1 (0.8)
USPS	90.6 (**0.2**)	92.1 (**0.2**)	91.5 (**0.2**)	92.4 (**0.2**)	95.6 (0.4)	91.6 (0.3)	**95.8** (0.4)
Average	86.1 (**0.3**)	88.7 (0.4)	88.9 (**0.3**)	89.2 (**0.3**)	89.9 (0.6)	89.9 (0.8)	**91.6** (0.5)

Table 11
Comparison of average accuracies of AHP-kNN with some state-of-the-art feature selection methods along-with some online feature selection algorithm for high dimensional datasets with multiple classes(wins in **bold**).

Datasets with	**10 random runs with holdout 0.3 and k=5**					
Multiple Class	**M-CW** [42]	**M-AROW** [64]	**M-SCW1** [43]	**M-SCW2** [43]	**mRMR** [29,37]	**AHP-kNN**
Yale	57.0 (2.6)	56.7 (3.4)	53.3 (1.6)	56.6 (3.5)	65.9 (5.0)	68.4 (4.6)
ORL	69.8 (1.7)	73.8 (1.2)	69.8 (1.7)	71.1 (1.0)	81.1 (2.5)	87.9 (2.4)
Average	63.4 (2.2)	65.3 (2.3)	61.6 (**1.7**)	63.9 (2.3)	73.5 (3.8)	**78.1** (3.5)

Fig. 4. Face recognition by AHP-kNN(The first row contains test images and the second row contains k=1 images).

Fig. 5. Hand-writing Recognition by AHP induced kNN(The first row contains test images and the second row contains k=1 images).

Fig. 4, both the test images and the detected nearest training faces have been illustrated. Similarly, the nearest training patterns detected [7] in case of hand-writing recognition problem has been compared with the test patterns in Fig. 5. These two figures clearly suggest that our method performs well on both these high dimensional datasets.

6. Conclusion

In this paper, we use Granger causality and AHP to improve the performance of traditional kNN. Two criteria, based on training class-wise group-statistics, are used during pairwise comparison of features. Granger causality is employed to assign due preferences to the criteria matrices. AHP is applied to obtain weights for the different features. Finally, these weights are used to build a weighted distance function for the kNN algorithm. Comprehensive experimental comparisons on UCI datasets clearly indicate the potential of the proposed approach. In future, we plan to explore different regularizers to avoid overfitting and/or underfitting by proper adjustment of the bias and variance.

References

[1] T.M. Cover, P.E. Hart, Nearest neighbor pattern classification, IEEE Trans. Inf. Theory 13 (1967) 21–27.
[2] Y.-C. Liaw, M.-L. Leou, C.-M. Wu, Fast exact k nearest neighbors search using an orthogonal search tree, Pattern Recognit. 43 (6) (2010) 2351–2358.
[3] B. Yang, M. Xiang, Y. Zhang, Multi-manifold discriminant isomap for visualization and classification, Pattern Recognit. 55 (2016) 215–230.
[4] D. Adeniyi, Z. Wei, Y. Yongquan, Automated web usage data mining and recommendation system using k-nearest neighbor (knn) classification method, Appl. Comput. Inform. 12 (1) (2016) 90–108.
[5] J. Maillo, I. Triguero, F. Herrera, A mapreduce-based k-nearest neighbor approach for big data classification, in: IEEE TrustCom/BigDataSE/ISPA, Helsinki, Finland, Volume 2, 2015, pp. 167–172.
[6] J. Zou, W. Li, Q. Du, Sparse representation-based nearest neighbor classifiers for hyperspectral imagery, IEEE Geosci. Remote Sens. Lett. 12 (12) (2015) 2418–2422.
[7] W. Yang, C. Sun, L. Zhang, A multi-manifold discriminant analysis method for image feature extraction, Pattern Recognit. 44 (8) (2011) 1649–1657.
[8] P. Mitra, C.A. Murthy, S.K. Pal, Unsupervised feature selection using feature similarity, IEEE Trans. Pattern. Anal. Mach. Intell. 24 (3) (2002) 301–312.
[9] G. Bhattacharya, K. Ghosh, A.S. Chowdhury, An affinity-based new local distance function and similarity measure for knn algorithm, Pattern. Recog. Lett. 33 (3) (2012) 356–363.
[10] W. Zheng, L. Zhao, C. Zou, Locally nearest neighbor classifiers for pattern classification, Pattern Recognit. 37 (6) (2004) 1307–1309.
[11] G. Bhattacharya, K. Ghosh, A. Chowdhury, Test point specific k estimation for knn classifier, in: Patt. Recog. (ICPR), 2014 in: Proceedings of the 22nd International Conference on, 2014, pp. 1478–1483.
[12] M.A. Tahir, J.E. Smith, Creating diverse nearest-neighbour ensembles using simultaneous metaheuristic feature selection, Pattern. Recog. Lett. 31 (11) (2010) 1470–1480.
[13] C. Schaffer, A conservation law for generalization performance., in: W. W. Cohen, H. Hirsh (Eds.), ICML, Morgan Kaufmann, 1994, pp. 259–265.
[14] D.R. Wilson, T.R. Martinez, Improved heterogeneous distance functions, J. Artif. Intell. Res. 6 (1) (1997) 1–34.
[15] M. Liu, B.C. Vemuri, A robust and efficient doubly regularized metric learning approach, in: Proceedings of the 12th European Conference on Comput. Vis. - Volume Part IV, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 646–659.

[16] P. Xie, E.P. Xing, Large scale distributed distance metric learning, CoRR abs/1412.5949.
[17] J.V. Davis, B. Kulis, P. Jain, S. Sra, I.S. Dhillon, Information-theoretic metric learning, in: Proceedings of the 24th International Conference on Mach. Learning (ICML), 2007, pp. 209–216.
[18] I.W. Tsang, P. ming Cheung, J.T. Kwok, Kernel relevant component analysis for distance metric learning, in: IEEE International Joint Conference on Neural Networks (IJCNN, 2005, pp. 954–959.
[19] K.Q. Weinberger, J. Blitzer, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, in: NIPS, MIT Press, 2006.
[20] S. Wang, R. Jin, An information geometry approach for distance metric learning, in: D. V. Dyk, M. Welling (Eds.), in: Proceedings of the Twelfth International Conference on Artif. Intell. and Statistics (AISTATS-09), Vol. 5, J. of Mach. Learning Research - Proceedings Track, 2009, pp. 591–598.
[21] J.M. Pena, R. Nilsson, On the complexity of discrete feature selection for optimal classification, IEEE Trans. Pattern. Anal. Mach. Intell. 32 (8) (2010) 1517–1522.
[22] K. Kira, L.A. Rendell, A practical approach to feature selection, Morgan Kaufmann Publishers Inc., pp. 249–256.
[23] I. Kononenko, Estimating attributes: Analysis and extensions of relief, in: L. De Raedt, and F. Bergadano (Eds.), Machine Learning: ECML-94, Springer-Verlag, 1994, pp. 171–182.
[24] I. Kononenko, E. Simec, M. Robnik-Sikonja, Overcoming the myopia of inductive learning algorithms with relieff, Appl. Intell. 7 (1997) 39–55.
[25] M. Scherf, W. Brauer, Feature selection by means of a feature weighting approach, Tech. rep. (1997).
[26] W. Yang, Z. Wang, C. Sun, A collaborative representation based projections method for feature extraction, Pattern Recognit. 48 (1) (2015) 20–27.
[27] M. Sewell, URL ⟨http://machine-learning.martinsewell.com/feature-selection/feature-selection.pdf⟩(2007).
[28] O. Räsänen, J. Pohjalainen, Random subset feature selection in automatic recognition of developmental disorders, affective states, and level of conflict from speech., in: INTERSPEECH, ISCA, 2013, pp. 210–214.
[29] H. Peng, F. Long, C. Ding, Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[30] M. Sugiyama, Local fisher discriminant analysis for supervised dimensionality reduction, in: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, ACM, 2006, pp. 905–912.
[31] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugen. 7 (7) (1936) 179–188.
[32] K. Fukunaga, Introduction to Statistical Pattern Recognition, Computer Science and Scientific Computing, Academic Press, Inc., 1990.
[33] Y. Sun, Iterative relief for feature weighting: algorithms, theories, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007) 1035–1051.
[34] M. Robnik-Šikonja, I. Kononenko, Theoretical and empirical analysis of relieff and rrelieff, Mach. Learn. 53 (1–2) (2003) 23–69.
[35] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the em algorithm, J. R. Stat. Soc. Ser. B (Methodol.) 39 (1) (1977) 1–38.
[36] Z. Deng, F. Chung, S. Wang, Robust relief-feature weighting, margin maximization, and fuzzy optimization, IEEE Trans. Fuzzy Syst. 18 (4) (2010) 726–744.
[37] C.-C. Chang, Generalized iterative relief for supervised distance metric learning, Pattern Recogn. 43 (8) (2010) 2971–2981.
[38] S.C. Hoi, J. Wang, P. Zhao, R. Jin, Online Feature Sel. Min. big data (2012).
[39] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, Y. Singer, Online passive-aggressive algorithms, J. Mach. Learn. Res. 7 (2006) 551–585.
[40] K. Crammer, A. Kulesza, M. Dredze, Adaptive regularization of weight vectors, in: Y. Bengio, D. Schuurmans, J.D. Lafferty, C.K.I. Williams, A. Culotta (Eds.), Advances in Neural Information Processing Systems 22, Curran Associates, Inc., 2009, pp. 414–422.
[41] S.C. Hoi, J. Wang, P. Zhao, Libol: a library for online learning algorithms, J. Mach. Learn. Res. 15 (2014) 495–499.
[42] K. Crammer, M. Dredze, F. Pereira, Exact convex confidence-weighted learning, in: Proceedings of the 21st International Conference on Neural Information Processing Systems, NIPS'08, Curran Associates Inc., 2008, pp. 345–352.
[43] S.C.H. Hoi, J. Wang, P. Zhao, Exact soft confidence-weighted learning., in: ICML, icml.cc / Omnipress.
[44] T.L. Saaty, The analytic hierarchy process : planning, priority setting, resource allocation, McGraw-Hill International book Co., 1980.
[45] N. Yaraghi, P. Tabesh, P. Guan, J. Zhuang, Comparison of AHP and monte carlo AHP under different levels of uncertainty, IEEE Trans. Eng. Manag. 62 (1) (2015) 122–132.
[46] T. Nguyen, S. Nahavandi, Modified ahp for gene selection and cancer classification using type-2 fuzzy logic, IEEE Trans. Fuzzy Syst. PP 24 (2) (2016) 273–287.
[47] L. Felföldi, A. Kocsor, Ahp-based classifier combination, in: Patt. Recog. in Inf. Syst., in: Proceedings of the 4th International Workshop on Patt. Recog. in Inf. Syst., PRIS 2004, in conjunction with ICEIS 2004, 2004, pp. 45–58.
[48] C.W.J. Granger, Investigating causal relations by econometric models and cross-spectral methods, Econometrica 37 (3) (1969) 424–438.
[49] P.A. Valdés-Sosa, J.M. Bornot-Sánchez, M. Vega-Hernández, L. Melie-García, A. Lage-Castellanos, E. Canales-Rodríguez, Granger Causality on Spatial Manifolds: Applications to Neuroimaging, Wiley-VCH Verlag GmbH & Co. KGaA, 2006, pp. 461–491.
[50] E. Kim, D.-S. Kim, F. Ahmad, H.W. Park, Pattern-based granger causality mapping in fmri, Brain Connect. 3 (6) (2013) 569–577.
[51] M.B. Schippers, R. Renken, C. Keysers, The effect of intra- and inter-subject variability of hemodynamic responses on group level granger causality analyses, NeuroImage 57 (1) (2011) 22–36.
[52] K. Hlaváková-Schindler, M. Paluš, M. Vejmelka, J. Bhattacharya, Causality detection based on information-theoretic approaches in time series analysis, Phys. Rep. 441 (1) (2007) 1–46.
[53] P. Kazibudzki, Comparison of analytic hierarchy process and some new optimization procedures for ratio scaling, Sci. Res. Inst. Math. Comput. Sci. 10 (1) (2011) 101–108.
[54] A. Ishizaka, M. Lusti, How to derive priorities in ahp: a comparative study, Cent. Eur. J. Oper. Res. 14 (4) (2006) 387–400.
[55] H. Qiu, Y. Liu, N.A. Subrahmanya, W. Li, Granger causality for time-series anomaly detection., in: M.J. Zaki, A. Siebes, J.X. Yu, B. Goethals, G.I. Webb, X. Wu (Eds.), ICDM, IEEE Computer Society, 2012, pp. 1074–1079.
[56] M. Lichman, UCI machine learning repository (2013). URL ⟨http://archive.ics.uci.edu/ml⟩
[57] J. Li, K. Cheng, S. Wang, F. Morstatter, T. Robert, J. Tang, H. Liu, Feature selection: A data perspective, arXiv:1601.07996
[58] P.C. Mahalanobis, On the generalized distance in statistics, Proceedings of the National Institute of Sciences (Calcutta) 2 (1936) 49–55.
[59] T. Suzuki, M. Sugiyama, T. Tanaka, Mutual information approximation via maximum likelihood estimation of density ratio, in: Inf. Theory, 2009. ISIT 2009. IEEE International Symposium on, 2009, pp. 463–467.
[60] T. Suzuki, M. Sugiyama, T. Kanamori, J. Sese, Mutual information estimation reveals global associations between stimuli and biological processes, BMC Bioinforma. 10 (1) (2009) 1–12.
[61] M. Robnik-Sikonja, I. Kononenko, An adaptation of relief for attribute estimation in regression (1997).
[62] D. Ververidis, C. Kotropoulos, Sequential forward feature selection with low computational cost, in: Signal Processing Conference, 2005 13th European, IEEE, 2005, pp. 1–4.
[63] S. Paul, S. Das, Simultaneous feature selection and weighting - an evolutionary multi-objective optimization approach, Pattern. Recog. Lett. 65 (2015) 51–59.
[64] K. Crammer, A. Kulesza, M. Dredze, Adaptive regularization of weight vectors, Mach. Learn. 91 (2) (2013) 155–187.
[65] J. Pohjalainen, O. Räsänen, S. Kadioglu, Feature Selection Methods and Their Combinations in High-Dimensional Classification of Speaker Likability, Intelligibility and Personality Traits, Comput. Speech & Language.
[66] D. Ververidis, C. Kotropoulos, Emotional speech classification using gaussian mixture models and the sequential floating forward selection algorithm, in: Proceedings of the 2005 IEEE International Conference on Multimedia and Expo, ICME, 2005, pp. 1500–1503.
[67] K.Q. Weinberger, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, J. Mach. Learn. Res. 10 (2009) 207–244.

Gautam Bhattacharya did his M. Sc. in Physics (Specialization: Electronics) from the University of Jadavpur, Kolkata, India in 1999 and is currently working as an Assistant Professor in Physics at UIT, Burdwan University, Burdwan, India since 2003. He obtained the Ph.D. degree from the University of Jadavpur, Kolkata, India in Engineering in 2016. He has 16 publications in pattern recognition, image processing and in solar physics.

Koushik Ghosh born on 25th June 1974 in Kolkata, India, is presently attached with the Engineering Section of the University of Burdwan, Burdwan, India (University Institute of Technology) as an Assistant Professor in Mathematics in the Department of General Science and Humanities. He did M.Sc. and M.Phil. in Applied Mathematics from the University of Calcutta, Kolkata, India in the year 1998 and 2000 respectively and obtained his Ph.D. in Astrophysics from the same university in the year 2004. In a nutshell his research areas are i) Stellar and Substellar Astrophysics, ii) Analysis of Time Series and Statistical Signal Processing (Applications in Solar Signals and Financial Time Series), iii) Nonlinear Systems and Dynamics, iv) Mathematical Modeling in Biological Systems, Social and Behavioral Sciences, v) Pattern Recognition. He has 122 publications in different reputed journals and conference proceedings and he has presented papers or delivered invited talks at 138 conferences/seminars/workshops inside or outside of India till mid of May 2016. Three candidates already got the Ph.D. award and two have submitted and waiting for the result under his supervision. In addition to this presently four research scholars are working under his guidance for their Ph.D. degree. He was awarded S. N. Bose Birth Centenary award for his research excellence by the Calcutta Mathematical Society, Kolkata, India in the year 2000. He was selected among the best six Young Scientists all over India in the years 2002, 2004, 2005 and 2006 in Indian Science Congress Association.

Ananda S. Chowdhury earned his Ph.D. in Computer Science from the University of Georgia, Athens, Georgia in July 2007. From August 2007 to December 2008, he worked as a postdoctoral fellow in the department of Radiology and Imaging Sciences at the National Institutes of Health, Bethesda, Maryland. At present, he is working as an Associate Professor in the department of Electronics and Telecommunication Engineering at Jadavpur University, Kolkata, India where he leads the Imaging Vision and Pattern Recognition group. He has authored or coauthored more than forty-five papers in leading international journals and conferences, in addition to a monograph in the Springer Advances in Computer Vision and Pattern Recognition Series. His research interests include computer vision, pattern recognition, biomedical image processing, and multimedia analysis. Dr. Chowdhury is a senior member of the IEEE and the IAPR TC member of Graph-Based Representations in Pattern Recognition. He currently serves as an Associate Editor of Pattern Recognition Letters and his Erdös number is 2.