Fuzzy Empirical Copula for Estimating Data Dependence Structure

Zhaojie Ju [a] Youlun Xiong [b] and Honghai Liu [a]

[a] *Intelligent Systems & Robotics Group, School of Creative Technologies, University of Portsmouth, UK*

[b] *School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan China*

Fuzzy Empirical Copula, Data Aggregation, Dependence Structure.

1. Introduction

The information in the world is becoming more and more electronic. Due to the improvements in data collection and storage during the past decades, huge amounts of data can lead to the problem of information overload [1] to many researchers in domains such as engineering, economics and astronomy. The increase of the number of dimensions associated with each observation and growth of the sampling time points are the main reasons of information overload. In many cases, datasets contain not only useful messages but also considerable trivial and redundant information both in the dimensions (attributes) and samples. How to remove the redundant information and maintain the important information is crucial in many applications. Two important methods are normally employed to solve this problem: dimensionality reduction and clustering. The formal reduces trivial attributes, maintaining the number of samples, while the later eliminates the redundant samples without changing the number of attributes.

There are various traditional and current state of the art dimensionality reduction methods to solve the above problem. Principal Component Analysis (PCA) was invented in 1901 by Karl Pearson [2] and is mostly used for dimensionality reduction in a dataset by retaining the characteristics of the dataset that contribute most to its variance. It keeps lower-order principal components and ignores higher-order ones. Such lower-order components often contain the "most important" aspects. Like PCA, Factor analysis (FA) is another second-order method [3]. FA becomes essentially equivalent to PCA if the "errors" in the FA model are all assumed to have the same variance. These second-order methods require classical matrix manipulations and assumption that datasets are realizations from Gaussian distributions. For non-Gaussian datasets, higher-order dimension reduction methods such as Projection Pursuit (PP) [4] and Independent Component Analysis (ICA) [5] are introduced. Additionally, non-linear PCA can also deal with non-Gaussian datasets using non-linear objective functions to determine the optimal weights in principal [6]. Its resulting components are still linear combinations of the original variables, so it can be regarded as a special case of ICA. Other non-linear methods such as Principal Curves (PC) [7] and Self Organizing Maps (SOM) [8] can be thought to be non-linear ICA [9] in that they replace the linear transformation of ICA with a real-valued non-linear vector function. Curvilinear Component Analysis (CCA) is a relatively new non-linear mapping method, being improved from Sammon's mapping by Jeanny Heault and Pierre Demartines [10]. It uses a new cost function able to unfold strongly non-linear or even closed structures, which significantly speeds up the calculation and interactively helps users control the minimized function. However, more parameters should be considered for most of these high-order and non-linear dimensionality reduction methods and their performances strongly depend on complex adjustments of these parameters, for instance there are three parameters in CCA: the projection space dimension and the two time decreasing parameters.

However, dimensionality reduction methods can not be used during estimating data dependence structure, because dependence structure includes all the interrelations of the attributes and high-order attributes are not supposed to be ignored. Clustering is the classification of objects into clusters so that objects from the same cluster are more similar to each other than objects from different clusters. It can effectively reduce the number of data samples, so it is suitable for reducing the redundant information when estimating data dependence structure. The most common algorithms include K-means [11], fuzzy C-means [12], and fuzzy C-means-derived clustering approaches such as fuzzy J-means [13] and fuzzy SOM [14], which construct clusters on the basis of pairwise distance between objects, so that they are incapable of capturing non-linear relationships and thereby fail to represent a dataset with non-linear structure. Hierarchical clustering is another important approach but suffers from lack of robustness, non-uniqueness, and inversion problems [15]. Gaussian Mixture Model (GMM) is based on the assumption that datasets are generated by a mixture of Gaussian distributions with certain probability. But this assumption is not always satisfied for general datasets even after various transformations aimed at improving the normality of the data distribution [16, 17].

Copula is a general way of formulating a multivariate distribution with uniform marginal distributions in such a way that various general types of dependence can be presented. The copula of a multivariate distribution can be considered as the part describing its dependence structure as opposed to the behaviour of each of its margins [18]. It is a good way of studying scale-free measures of dependences among variables and also a good starting point for constructing families of bivariate distributions [19]. Sklar's theorem [20] elucidates that a multivariate distribution function can be represented by a copula function which binds its univariate margins. Further, empirical copulas were introduced and first studied by Deheuvels in 1979 [21, 22], which can be used to study the interrelations of marginal variables with unknown underlying distributions. The copula approach has many advantages [23] and has been used widely in finance [24, 25] and econometrics [26, 27]. Kolesarova *et al.* [28] defined a new copula called discrete copulas on a grid of the unit square and showed that each discrete copula is naturally associated with a bistochastic matrix. Baets and Meyer [29] also presented a general framework for constructing copulas, which extended the diagonal construction to the orthogonal grid construction. Simultaneously, empirical copula has gained an increasing amount of attention recently. Dempster *et al.* [30] constructed an empirical copula for Collateralized debt obligation tranche pricing and achieved a better performance than the dominant base correlation approach in pricing non-standard tranches. Ma and Sun [31] proposed a Chow-Liu like method based on a dependence measure via empirical copulas to estimate maximum spanning product copula with only bivariate dependence relations, while Morettin *et al.* [32] proposed wavelet estimators based on empirical copulas which can be used for independent, identically distributed time series data.

It is evident, however, that the efficiency of empirical copula is outstandingly poor though it provides effective performance on data dependence structure estimation. It is common that natural datasets are represented by tremendous storage size, and it is impossible to process them using empirical copula in most cases. In order to overcome this problem, we propose an algorithm named fuzzy empirical copula which integrates fuzzy clustering with empirical copula. Fuzzy Clustering by Local Approximation of Memberships (FLAME) [17] is firstly extended into multi-dimensional space, then the FLAME$^+$ is utilized to reduce the number of sampling data and maintaining the interrelations at the same time before data dependence structure estimation takes over. The remainder of the paper is organized as follows. Section 2 presents copula theory with a focus on dependence structure estimation using empirical copula. Section 3 proposes the Fuzzy empirical copula algorithm. Section 4 presents the experiments whose results demonstrate the effectiveness of the proposed fuzzy empirical copula. Concluding remarks and future work are found in Section 5.

2. Dependence Structure Estimation via Empirical Copula

As a general way of formulating a multivariate distribution, copula can be used to study various general types of dependence between variables. Other ways of formulating multivariate distributions include conceptually-based approaches in which the real-world meaning of the variables is used to imply what types of relationships might occur. In contrast, the approach via copulas might be considered as being more raw, but it does allow much more general types of dependencies to be included than would usually be invoked by a conceptual approach. Nelsen [19] has proven that these measures, such as Kendall's tau, Spearman's rho and Gini's gamma, can be re-expressed only in terms of copula. Though their direct calculation may have much less computational cost than when using copulas, copula summarizes all the dependence relations and provides a natural way to study and measure dependence between variables in statistics. It is a very important approach since copula properties are invariant under strictly increasing transformations of the underlying random variables. Spearman's rho and Gini's gamma are considered in this paper. In this section, we firstly revisit the theoretical foundation of copula and empirical copula, then introduce the theorem of calculating Spearman's rho and Gini's gamma using bivariate empirical copula, finally analyse the time complexity of the computation.

2.1. Copula

A n-dimensional copula is defined as a multivariate joint distribution on the n-dimensional unit cube $[0,1]^n$ such that every marginal distribution is uniform on the interval $[0,1]$.

Definition 2.1.1. *A n-dimensional copula is a function C from I^n to I with the following properties [19]:*

1. *C is grounded, i.e., for every* ***u*** *in I^n, $C(\boldsymbol{u}) = 0$ if at least one coordinate $u_j = 0$, $j = 1, \cdots, n$.*
2. *If all coordinates of* ***u*** *are 1 except for some u_j, $j = 1, \cdots, n$, then $C(\boldsymbol{u}) = C(1, \cdots, 1, u_j, 1, \cdots, 1) = u_j$.*
3. *C n-increasing, i.e., for each hyperrectangle $B = \times_{i=1}^{n}[x_i, y_i] \subseteq [0,1]^n$*

$$V_c(B) = \sum_{z \in \times_{i=1}^{n}\{x_i, y_i\}} (-1)^{N(z)} C(z) \geq 0 \quad (1)$$

where the $N(z) = card\{k \,|z_k = x_k\}$. $V_c(B)$ is the so called C-volume of B.

Sklar's Theorem [20] is central to the theory of copula and underlies most applications of the copula. It elucidates the role that copula plays in the relationship between multivariate distribution functions and their univariate margins.

Sklar's Theorem 2.1.1. *Let H be a joint distribution function with margins $F_i (i = 1, 2, \cdots, n)$. Then there exists a copula C such that for all x_i in $\bar{R}$,*

$$H(x_1, \cdots, x_n) = C(F_1(x_1), \cdots, F_n(x_n)) \quad (2)$$

where C is a n-dimensional copula, F_i are marginal distribution function of x_i.

If $F_i (i = 1, \cdots, n)$ are continuous, C is unique. If C is a n-dimensional copula and $F_i (i = 1, \cdots, n)$ are distribution functions, then the function H defined by equation 2 is a joint distribution function with margins $F_i (i = 1, \cdots, n)$. More details can be seen in [19,23].

2.2. Empirical Copula and Dependence Estimation

The empirical copula is a characterization of the dependence function between variables based on observational data using order statistics theory and it can reproduce any pattern found in the observed data. If the marginal distributions are normalized, the empirical copula is the empirical distribution function for the joint distribution. Priority has been given to bivariate empirical copula due to computational cost. The reason is twofold: one is that the interrelation between every two attributes is the basic relationship in most attributes, and it is practical to use bivariate empirical copula to construct the whole structure of every two attributes' dependence; the second is that the dependence structure of dataset X including r attributes would have $\begin{pmatrix} r \\ 2 \end{pmatrix} = \frac{1}{2} r(r-1)$ bivariate interrelations. Bivariate empirical copula is given as follows.

Definition 2.2.1. *Let $\{(x_k, y_k)\}_{k=1}^{n}$ denote a sample of size n from a continuous bivariate distribution. The empirical copula is the function C_n given by*

$$C_n(\tfrac{i}{n}, \tfrac{j}{n}) = \tfrac{card\{(x,y): x \leq x_{(i)}, y \leq y_{(j)}\}}{n} \quad (3)$$

where $x_{(i)}$ and $y_{(j)}$, $1 \leq i, j \leq n$, denote order statistics from the sample [19].
The empirical copula frequency c_n is given by

$$c_n(\tfrac{i}{n}, \tfrac{j}{n}) = \begin{cases} \frac{1}{n}, \text{ if } (x_{(i)}, y_{(j)}) \text{ is an element of the sample} \\ 0, \qquad \text{otherwise} \end{cases} \quad (4)$$

Note that C_n and c_n are related via

$$C_n(\frac{i}{n}, \frac{j}{n}) = \sum_{p=1}^{i} \sum_{q=1}^{j} c_n(\frac{p}{n}, \frac{q}{n}) \quad (5)$$

Theorem 2.2.1. *Let C_n and c_n denote, respectively, the empirical copula and the empirical copula frequency function for the sample $\{(x_k, y_k)\}_{k=1}^{n}$. If ρ and*

γ denote, respectively, the sample versions of Spearman's rho, and Gini's gamma [33, 34], then

$$\rho = \frac{12}{n^2-1} \sum_{i=1}^{n} \sum_{j=1}^{n} \left[C_n\left(\frac{i}{n} \cdot \frac{j}{n}\right) - \frac{i}{n} \cdot \frac{j}{n}\right] \tag{6}$$

and

$$\gamma = \frac{2n}{\lfloor n^2/2 \rfloor} \left\{ \sum_{i=1}^{n-1} C_n\left(\frac{i}{n}, 1 - \frac{i}{n}\right) - \sum_{i=1}^{n} \left[\frac{i}{n} - C_n\left(\frac{i}{n}, \frac{i}{n}\right)\right] \right\} \tag{7}$$

Spearman's rho and Gini's gamma are two ways of measuring two variables' association [19]. According to the definition and theorem, we can estimate correlations between variables using empirical copula and Spearman's rho & Gini's gamma. Suppose the number of objects is n and number of attributes is r. For $r << n$, according to the equations 3, 6 and 7, the time complexity of Spearman's rho or Gini's gamma is $O(n^3)$.

3. Fuzzy Empirical Copula

In this section Fuzzy clustering by Local Approximation of Memberships (FLAME) is extended first in terms of dimension and distance functions, then is integrated into empirical copula to enhance its computational efficiency. FLAME was proposed to cluster DNA microarraydata [17]. It defines clusters in the relatively dense regions of a dataset and performs cluster assignment solely based on the neighbourhood relationships among objects. One of the FLAME algorithm features is that the memberships of neighbouring objects in the fuzzy membership space are set according to the neighbourhood relationships among neighbouring objects in the feature space. FLAME has been extended in terms of dimension and distance function (*i.e.*, FLAME$^+$), which still consists of three main steps of FLAME algorithm: initialization, approximation and assignment.

3.1. Initialization

The first step, initialization, is to classify three types of objects: Cluster Supporting Object (CSO), cluster outliers and the rest which are named Normal Points (NPs).

Let X be a r-dimensional dataset with n objects. The r-dimensional distance between two instances is

$$d_p(x, y) = (\sum_{i=1}^{r} |x_i - y_i|^p)^{(1/p)} \tag{8}$$

where $x, y \in X$; $1 \leq p \leq \infty$; d_1 is the Manhattan distance, d_2 is the familiar Euclidean distance, and d_∞ corresponds to the maximum distance in any dimension. Then the similarity of these two objects is calculated as:

$$s_{xy} = \frac{1}{d_p(x,y)} \tag{9}$$

Similarity is the degree of resemblance between two or more objects. There are different ways to calculate the similarity. Since "the density of each object is calculated as one over the average distance to the k-nearest neighbors" in the FLAME clustering algorithm [17], to make the relation between similarity and density more direct and simple, we choose Eq. 9 to calculate the similarity in the paper.

The K-Nearest Neighbours (KNNs) for each object are defined as the k objects ($k \leq n$) with the k highest similarity. The density of object x with KNNs can be obtained

$$Den_p(x) = \frac{k}{\sum\limits_{y \in knn(x)} d_p(x,y)} \tag{10}$$

where $knn(x)$ stands for the set of KNNs of the object x.

Subsequently, the set of CSOs is defined as the set of objects with local maximum density, *i.e.*, with a density higher than that of every object in their KNNs. The higher k is, the less CSOs will be identified, then less clusters will be generated. A density threshold needs defining to find possible cluster outliers, so objects with densities below the threshold are defined as possible outliers.

Each object x is associated with a membership vector $p(x)$, in which each element $p_i(x)$ indicates the membership degree of x in cluster i

$$p(x) = (p_1(x), ..., p_m(x)), \tag{11}$$

where $0 \leq p_i(x) \leq 1$; $\sum_{i=1}^{m} p_i(x) = 1$; m is the total number of CSOs and the outlier cluster, *i.e.*, $m = c+1$ where c is the number of CSOs; Each element of membership vector takes value between 0 and 1, indicating how much percentage an object belonging to a cluster, or being an outlier.

Based on the density estimation, each CSO is assigned with fixed and full membership to itself to represent one cluster, for example $p(x) = (0, 1, ...0)$ indicates that object x is the second CSO . Each outlier is assigned with fixed and full membership to the outlier group, $p(x) = (0, \cdots, 0, 1)$, and the NP is assigned with equal memberships to all clusters and the outlier group, $p(x) = (1/m, \cdots, 1/m)$.

3.2. Approximation

The second step is named local/neighbourhood approximation of fuzzy memberships, in which each NP's fuzzy membership is updated by a linear combination of the fuzzy memberships of its KNNs, while CSOs and outliers maintain the fixed and full memberships to themselves respectively.

The weights defining how much each neighbour will contribute to approximation of the fuzzy membership of that neighbour are estimated in equation 12, based on the fact that the neighbours that have higher similarities must have higher weights.

$$w_{xy} = \frac{s_{xy}}{\sum\limits_{z \in knn(x)} s_{xz}} \tag{12}$$

where $y \in knn(x)$. The membership vector of each NP is approximated according to equation 13, minimizing the overall difference between membership vectors and their approximations.

$$p^{t+1}(x) = \sum_{y \in knn(x)} w_{xy} p^t(y) \tag{13}$$

The overall local/neighbourhood approximation error is calculated by:

$$E(\{p\}) = \sum_{x \in X} \left\| p(x) - \sum_{y \in \mathrm{knn}(x)} w_{xy} p(y) \right\|^2 \tag{14}$$

The iteration of equation 13 breaks under the condition that $E(\{p\})$ is less than a predetermined threshold.

3.3. Assignment

Finally, it is to assign each object to the cluster based on its fuzzy membership. Usually, one cluster contains the objects that have higher membership degrees in this cluster than other clusters.

An example of FLAME$^+$ is provided in Fig. 1, where a dataset with 600 objects is randomly generated from a 3 dimensional distribution. FLAME$^+$ is applied to this dataset and three groups of objects are clustered as outliers, CSOs and NPs.

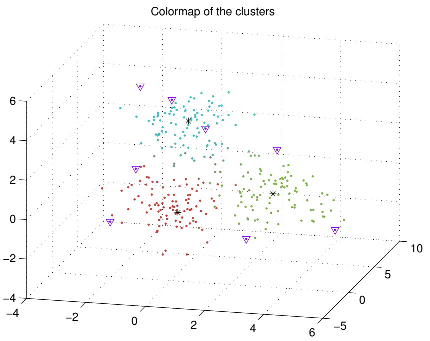

Fig. 1. Clustering random 3D Euclidean positions using FLAME$^+$. The star points in black are the centres of the clusters (CSO); points labeled with triangles are the outliers; the colour range of NPs represents their membership degrees.

Accurately calculating the entire time complexity of FLAME$^+$ is very challenging in that each iteration of local/neighbourhood approximation depends on the error threshold. However, it is necessary to analyse the complexity of the first step of the algorithm. Suppose the number of objects is n, number of attributes is r, its CSOs' number is c and number of nearest neighbours is k. For $r << n$ and $k << n$, the time complexity of the initialization is $O(n^2)$. An empirical study of the time complexity of FLAME$^+$ compared with other algorithms is performed to illustrate that FLAME$^+$ has significant computational advantage over hierarchical clustering, fuzzy C-means and fuzzy SOM, an exception is K-means [17]. In section 4, FLAME+ and K-means are compared in the context of fuzzy empirical copula.

We aim at developing an algorithm which can efficiently reduce the computational cost of empirical copula by filtering out redundant information in the sample. In addition, this algorithm should also be capable of dealing with arbitrary-distributed datasets in order to inherit the main advantage of empirical copula for data structure estimation. FLAME algorithm is selected for this purpose in that it not only fulfills the above requirements but also possesses the merit of few parameters, *i.e.*, the number of nearest neighbours and the value of the outlier's threshold.

It is evident that samples with higher densities are more reliable when used to represent whole samples in such a way that the main feature of the whole sample is maintained. The FLAME algorithms have the capability of identifying those “special" sampling points based on the objects’ density analysis. The “special" points are represented by the CSOs with the highest densities in all clusters. Therefore, the fuzzy empirical copula algorithm is proposed for achieving the following: high dimension FLAME algorithm is employed to identify characteristic feature points, and dependence structure, on the other hand, it is estimated via empirical copula.

Let X be r-dimensional dataset with n objects:

$$X = \begin{pmatrix} x_{11} \cdots x_{1n} \\ \vdots \ \ \ddots \ \ \vdots \\ x_{r1} \cdots x_{rn} \end{pmatrix}$$

so the i_{th} object is represented by the i_{th} column in matrix X: $x_i = [x_{1i}, x_{2i}, \cdots, x_{ri}]^T$ and the j_{th} attribute of X is defined as the j_{th} row: $x(j) = [x_{j1}, x_{j2}, \cdots, x_{jn}]$. The dependence structure in this paper is defined as the whole structure of every two attributes’ dependence which can be calculated by bivariate empirical copula, because the interrelation between every two attributes is the most basic relationship in several attributes. Given interrelations between every two attributes, relations of three or more attributes would be derived from their dependence structure which would have $\begin{pmatrix} r \\ 2 \end{pmatrix} = \frac{1}{2}r(r-1)$ interrelations (r is the number of attributes).

In fuzzy empirical copula, firstly FLAME$^+$ reduces the samples from n objects to c CSOs, and then empirical copula analyses the dependence of every two attributes in the derived CSO matrix. The first step can be considered as the operation on the column and the later on the row. For ideal performance of the proposed algorithm, one of outputs of FLAME$^+$, CSOs, is computed leading to efficient computation. That is to say we only have to implement the first step in the FLAME algorithm which has less time complexity than empirical copula and only one parameter, the number of neighbours, is required since the threshold works only for outliers. The Spearman’s rho and Gini’s gamma of the CSOs would be

$$\rho(u,v) = \tfrac{12}{c^2-1} \sum_{i=1}^{c} \sum_{j=1}^{c} \left[C_c^{(uv)} \left(\tfrac{i}{c}, \tfrac{j}{c} \right) - \tfrac{i}{c} \cdot \tfrac{j}{c} \right] \quad (15)$$

and

$$\gamma(u,v) = \frac{2n}{\lfloor n^2/2 \rfloor} \left\{ \sum_{i=1}^{c-1} C_c^{(uv)} \left(\frac{i}{c}, 1 - \frac{i}{c} \right) - \sum_{i=1}^{c} \left[\frac{i}{c} - C_c^{(uv)} \left(\frac{i}{c}, \frac{i}{c} \right) \right] \right\} \quad (16)$$

where $u \in [1, \cdots, r)$ and $v \in (u, \cdots, r]$; $C_c^{(uv)}$ is the bivariate empirical copula of the u_{th} and v_{th} attributes with c objects, and $C_c^{(uv)} = C_c^{(vu)}$.

The optimization is designed to automatically identify the optimized number of neighbours with acceptable errors. The number of nearest neighbours is increased by one at every step during the optimization until proper number of neighbours is identified. The optimization stops when the overall error of Spearman’s rho or Ginis gamma in equation is under the preset error threshold. The pseudo-code of fuzzy empirical copula is presented in algorithmic form 1 and its ‘EmpSG’ function is in algorithm form 2.

4. Experiment and Discussions

Experiments are conducted in this section, and results and discussions are provided for evaluating the effectiveness and efficiency of fuzzy empirical copula. After a brief explanation of the datasets, empirical copula and fuzzy empirical copula are employed respectively to estimate the dependence structures of the datasets. The section is concluded with the roles that clustering algorithms play in fuzzy empirical copula.

4.1. Data

Abalone [35] and yeast [36] datasets from UCI machine learning repository [37] were selected to evaluate the proposed algorithm in this paper. The abalone dataset was used to predict the age of abalone from the physical measurements such as weight and length, and it is not a trivial task to get their ages by counting the number of rings in their bodies through a microscope. 4177 abalone are sampled with 9 attributes in this dataset. Fig. 2 shows interrelations of length, diameter, whole weight and shell weight. This dataset could be regarded as 9 dimensional data with 4177 samples in which some measurements are intrinsically interrelated. The yeast dataset was constructed for predicting the cellular localization sites of proteins. It

Algorithm 1 Fuzzy Empirical Copula algorithm

```
Require: $X = \{x_1, x_2, ..., x_n\}$ {$X$ is a $r$ dimensional dataset with $n$ objects and $r << n$}
Require: $[\rho_{all}, \gamma_{all}]$ {the Spearman's rho and Gini's gamma of the original data $X$}
for all $i$ such that $1 \leq i \leq n$ do
    for all $j$ such that $1 \leq j \leq n$ do
        $d_2(x_i, x_j) \leftarrow eqution8$ {calculate the Euclidean distance between two objects using equation 8}
    end for
end for
$K = 0$ {$K$ is the number of nearest neighbours under consideration}
repeat
    $K = K + 1$
    for all $i$ such that $1 \leq i \leq n$ do
        $Den_2(x_i) \leftarrow equation10$ {get the density of every object}
    end for
    $c = 0$ {the number of CSOs}
    for all $i$ such that $1 \leq i \leq n$ do
        if $Den_2(x_i) \geq max(Den_2(y_i))$ where $y_i \in knn(x_i)$ then
            $c = c + 1$
            $CSO_c \leftarrow x_i$ {get CSOs which have the local maximum densities}
        end if
    end for
    for all $u$ such that $1 \leq u \leq r$ do
        for all $v$ such that $(u + 1) \leq v \leq r$ do
            $[\rho(u, v), \gamma(u, v)] \leftarrow EmpSG(CSO(u), CSO(v))$ {$CSO(i)$ is the $i^{th}$ attribution of CSO, $EmpSG$ is a function to calculate the data's Spearman's rho and Gini's gamma as showed in algorithm 2, $\rho_{r\times r}$ is the matrix of Spearman's rho and $\gamma_{r\times r}$ is the matrix of Gini's gamma}
        end for
    end for
    $error = \|[\rho_{all}, \gamma_{all}] - [\rho_{r\times r}, \gamma_{r\times r}]\|$ {take the Euclidean distance of the $\rho$ and $\gamma$ as the overall error}
until $error \geq threshold$ {$threshold$ is the threshold of overall error decided according to the original dataset}
```

Algorithm 2 Function of $EmpSG$ for Spearman's rho and Gini's gamma

```
Require: $CSO(u), CSO(v)$ {two attributes in CSO}
Ensure: $SP, GI$ {Spearman's rho and Gini's gamma of above two attributes}
$x = CSO(u); y = CSO(v)$ {$x$ and $y$ are two vectors with $c$ elememts}
$x' = sort(x); y' = sort(y)$ {$x'$ and $y'$ are the order statistics of $x$ and $y$}
for all $i$ such that $1 \leq i \leq c$ do
    for all $j$ such that $1 \leq j \leq c$ do
        $num \leftarrow 0$ {initialization}
        for all $t$ such that $1 \leq t \leq c$ do
            if $x(t) \leq x'(i)$ and $y(t) \leq y'(j)$ then
                $num \leftarrow num + 1$
            end if
        end for
        $EC(i, j) \leftarrow num/c$
    end for
end for
$SP \leftarrow 0$ {the return value of Spearman's rho}
$GI \leftarrow 0$ {the return value of Gini's gamma}
for all $i$ such that $1 \leq i \leq c$ do
    for all $j$ such that $1 \leq j \leq c$ do
        $SP \leftarrow SP + EC(i, j) - (j * i)/(c * c)$
    end for
    if $i \neq b$ then
        $GI \leftarrow GI + EC(i, c - i) - i/c + EC(i, i)$
    end if
end for
$SP \leftarrow SP * 12/(c * c - 1)$
$GI \leftarrow 2 * c/(c * c/2) * (GI - 1 + EC(c, c))$
return $SP$ and $GI$
```

contains 1484 instances with 8 attributes for each instance. Both of these two datasets contain strong nonlinear dependences between attributes. Priority herein is given to the sampling density and the interrelations among attributes. Supposing one dataset contains s attributes, its dependence structure includes $\binom{s}{2}$ interrelations of every two attributes among this dataset, which means the abalone and yeast datasets have two dependence structures of 36 and 28 interrelations to be analysed respectively.

4.2. Dependence Structure Estimation via Empirical Copula

Spearman's rhos and Gini's gammas are calculated according to equations 6 and 7 via empirical copula using

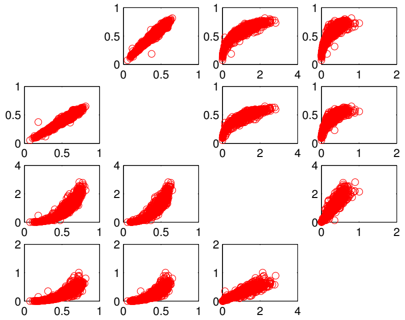

Fig. 2. The interrelations of length, diameter, whole weight and shell weight of abalone dataset

the two above datasets. The results of abalone's dependences of 36 correlations for 9 attributes are listed in the Fig. 3, and yeast dataset of 28 correlations for 8 attributes in the Fig. 4. The whole computation time for abalone is 27226 seconds, which is unrealistic for related applications. On the other hand, though the yeast dataset has fewer instances its computational time is still high, at 2813 seconds. It should be noted that it has to carefully handle the tradeoff between efficiency and accuracy of fuzzy empirical copula. The more nearest neighbours are considered (*e.g.*, Fig.3), the fewer CSOs will appear, the more efficient the algorithm is. It also, however, leads to larger error.

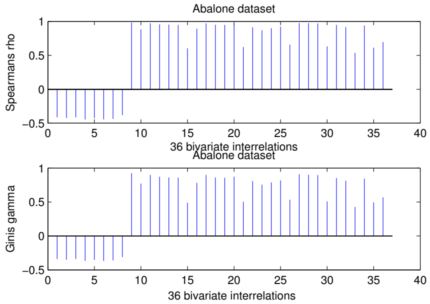

Fig. 3. The result of dependences of 36 correlations of the original abalone dataset

4.3. Dependence Structure Estimation via Fuzzy Empirical Copula

The proposed fuzzy empirical copula is employed in this section to reduce computation time for dependence structure analysis of these two datasets. The more nearest neighbours are considered, the fewer CSOs will appear. It indicates that the calculation of fuzzy empirical copula will be more efficient at the cost of lower accuracy. The proposed fuzzy empirical copula has the ability to identify the proper number of nearest neighbours which guarantees the fast computation with the overall error under the preset threshold. The threshold for Spearman's rho from equation 15 is predefined as,

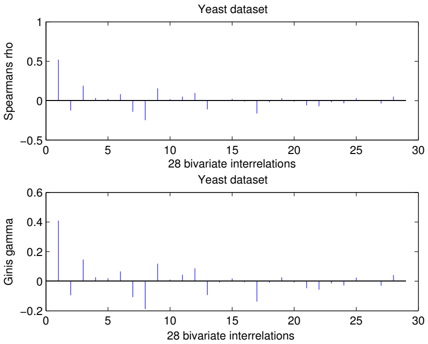

Fig. 4. The result of dependences of 28 correlations of the original yeast dataset

$$threshold = p \times \binom{s}{2} = \frac{ps(s-1)}{2} \tag{17}$$

where $\binom{s}{2}$ is the number of interrelations, combining 2 attributes out of s attributes; p is the average error percentage for each interrelation. Different threshold results in different computational time, and p is defined to take a value in the range from 0.5% to 1% according to the different features of datasets. From the above results of the two datasets using empirical copula, p is predefined as 0.6% and 1% for abalone and yeast datasets respectively, which indicates that abalone and yeast datasets have the thresholds of 0.228 and 0.28.

4.3.1. Abalone dataset

Under the overall error threshold of 0.228 for Spearman's rho, fuzzy empirical copula with 12 nearest neighbours has the lowest computation time. Thanks to the FLAME^{+}'s density sampling, when the number of nearest neighbours is 12, the number of data instances is reduced to 100 from 4177 depicted in Fig. 5. The 36 interrelations of the 9 attributes of these 100 CSOs are estimated as shown in red line in Fig. 6, where the blue lines are results generated from empirical copula. It illustrates that fuzzy empirical copula with 12 nearest neighbours does not cause unacceptable error to Spearman's rho and Gini's gamma compared to empirical copula. However, the computation time of fuzzy empirical copula algorithm is 68 seconds, which is only 0.25 percent of the computational time conducted by empirical copula algorithm.

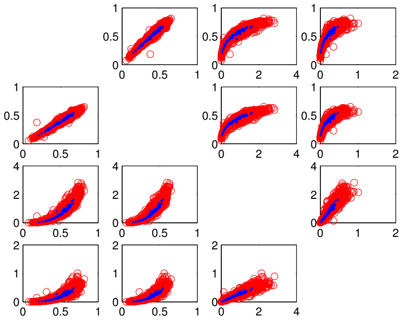

Fig. 5. Result of sampled abalone data using 12-neighbour-FLAME^{+}

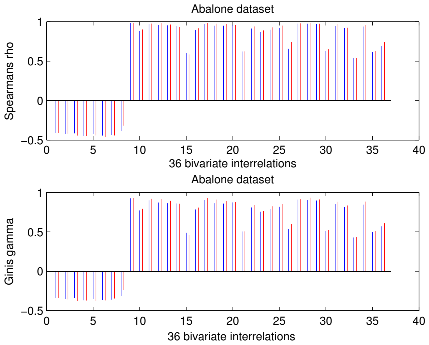

Fig. 6. Comparison of Spearman's rhos and Gini's gammas. Blue lines are the correlations of Empirical Copula algorithm while red lines are of Fuzzy Empirical Copula algorithm

In order to have a better understanding of the performance of fuzzy empirical copula, Fig. 7 displays the change of CSO's number with the growing number of nearest neighbours from 1 to 20. It shows that the numbers of abalone's CSOs drops exponentially with the growth of the number of nearest neighbours. With the growing nearest neighbours, Fig. 8 shows the overall error changes of Spearmans rho and Ginis gamma and Fig. 9 presents the time change. The error threshold locates the place between 12 and 13 nearest neighbours. Given a threshold, the error and the computational time can easily be decided from Figs. 8 and 9.

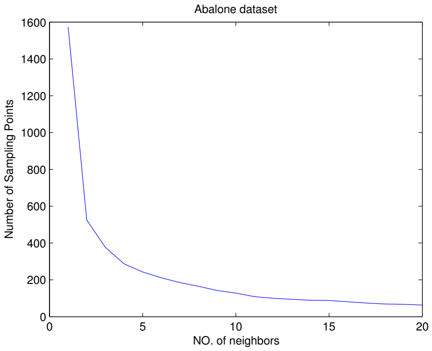

Fig. 7. The relationship between number of nearest neighbours and number of abalone's CSOs

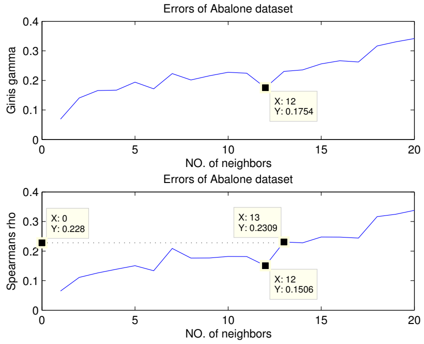

Fig. 8. Change of overall errors of Spearman's rhos and Gini's gammas with the growth of number of nearest neighbours in FLAME^{+}

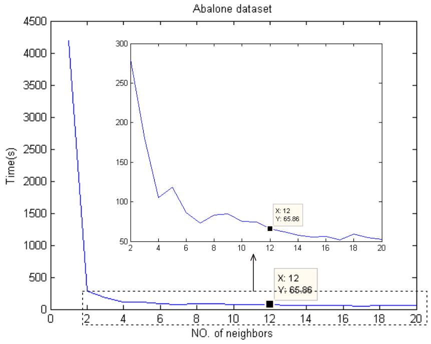

Fig. 9. Change of time costed by Fuzzy Empirical Copula with the growth of number of nearest neighbours in FLAME^{+}

4.3.2. Yeast dataset

Similar data processing in above section was employed on yeast dataset. If the threshold of Spearman's rho overall error is set to be 0.28, 2 nearest neighbours would make the estimation perform well because of the relatively small error which is under the threshold and fast computation. The result with 2 nearest neighbours is shown in red in Fig. 10, compared with the result in blue from Empirical Copula, which demonstrates that the dependence structure of yeast dataset is maintained. The errors are only 0.28 for Spearman's rho and 0.24 for Gini's gamma in Fig. 10, and computation time is 26.3 seconds which is only 0.93 percent of the time cost by Empirical Copula.

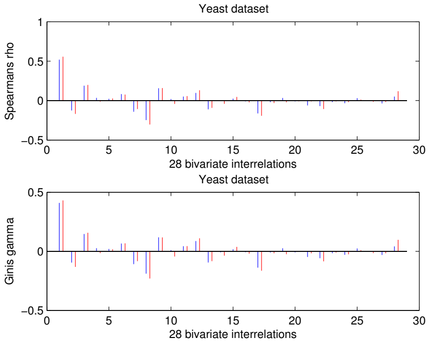

Fig. 10. Comparison of Spearman's rhos and Gini's gammas. Blue lines are the correlations of Empirical Copula algorithm while red lines are of Fuzzy Empirical Copula algorithm

The number of yeast's CSOs also drops exponentially with the growth of the number of nearest neighbours from 1 to 20 shown in Fig. 11. Fig. 12 shows the changes of errors of Spearman's rho and Gini's gamma grows with the number of nearest neighbours, while Fig. 13 displays the changes of cost time.

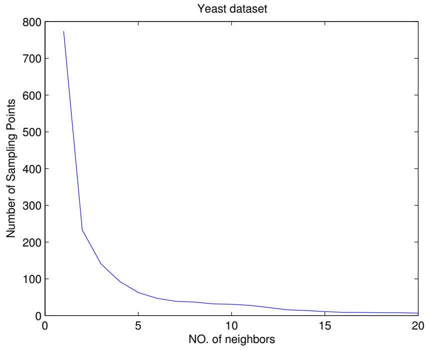

Fig. 11. The relationship between number of nearest neighbours and number of yeast's CSOs

However, for 5 or more nearest neighbours, the number of sampling data is reduced to less than 70 instances in Fig. 11, which are so few that the result becomes unacceptable with huge errors since the sampling data can not cover the main feature area of the whole dataset.

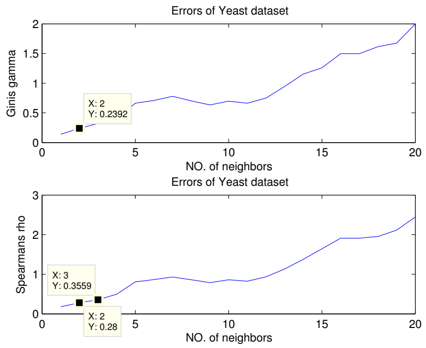

Fig. 12. Change of overall errors of Spearman's rhos and Gini's gammas with the growth of number of nearest neighbours in FLAME$^+$

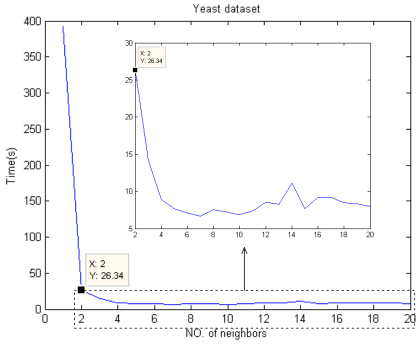

Fig. 13. Change of time cost by Fuzzy Empirical Copula with the growth of number of nearest neighbours in FLAME$^+$

4.4. *Comparison of FLAME$^+$ and K-means in Fuzzy Empirical Copula*

The reason to choose FLAME$^+$ as the fuzzy clustering algorithm instead of other algorithms (*e.g.*, K-menas) in Fuzzy Empirical Copula is on the basis of FLAME$^+$'s four main advantages. First it has the ability to capture non-linear relationships and non-globular clusters; secondly it can automatically define the number of clusters and identify cluster outliers; thirdly, compared with K-means and fuzzy C-means, the centres of FLAME$^+$ are real instances in the original dataset instead of the centroids of clusters with different traits which probably result in wrong dependence measures. Finally, FLAME$^+$ is also capable of dealing with a free-distributed dataset, which is not always true for algorithms like Gussian Mixture Models. In order to demonstrate the effectiveness of the proposed fuzzy empirical copula, we constructed K-means based Empirical Copula which employs the K-means algorithm to cluster the original dataset into numbers of subsets and uses centroids as the new dataset. Both of these two methods were applied to the abalone dataset. The comparison is based on the fact that cluster number of K-means is set to be the same as the abalone's CSOs number of FLAME$^+$ as listed in the table 1.

Fig. 14 demonstrates the comparison of computational cost by the proposed fuzzy empirical copula and K-means based empirical copula, where red curves are the changes of cost time by K-means based Empirical Copula while blue curves are by the proposed fuzzy empirical copula. It presents that both of the two algorithms achieve almost the same performance in saving the computational time.

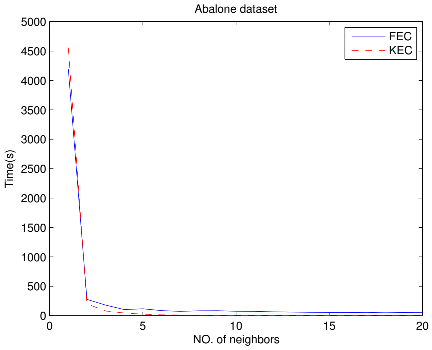

Fig. 14. Change of time cost when comparing the proposed Fuzzy Empirical Copula (FEC) and K-means based Empirical Copula (KEC)

In Fig. 15, with the decreasing number of clusters, the errors caused by K-means based Empirical Copula fluctuate violently and keep much higher than those by proposed fuzzy empirical copula. It illustrates that FLAME$^+$ outperforms K-means in maintaining the dependence structure though both of them have the almost same performance in reducing the cost time, and FLAME$^+$ is more suitable to be used in fuzzy empirical copula. One reason for the above results is that FLAME$^+$ is capable in dealing with non-linear relationships while K-means is not. Another reason is that FLAME$^+$ considers the real objects CSOs which are the samples in the datasets while K-means considers centroids which are virtual objects beyond the datasets. Centroids not belonging to the datasets may have different traits which probably result in wrong dependence measures.

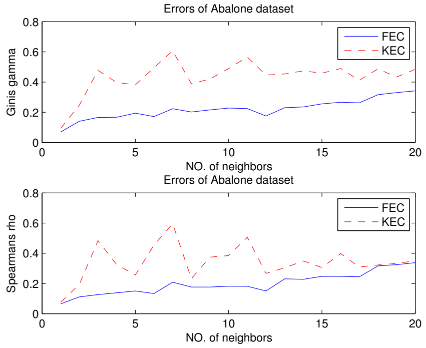

Fig. 15. Changes of overall errors of Spearman's rhos and Gini's gammas when comparing the proposed Fuzzy Empirical Copula (FEC) and K-means based Empirical Copula (KEC)

5. Concluding Remarks

Fuzzy empirical copula has been proposed to alleviate the computational burden of empirical copula. A high-dimensional FLAME$^+$ has been developed to identify the important objects containing the main features of the entire dataset, then empirical copula has been implemented to estimate the dependence structure of the objects. Abalone and yeast datasets from UCI machine learning repository are employed to evaluate the proposed method. The number of nearest neighbours is the tradeoff factor for handling accuracy and efficiency of data processing. With the preselected error threshold, fuzzy empirical copula has the capability of automatically identifying the optimized number of neighbours, which could be used to fast analyse similar datasets. Additionally nearest neighbours at the range of $0 - 20$ have been used to demonstrate the overall error changes of Spearman's rho and Gini's gamma, and the change of computational time. The experimental results have shown that fuzzy empirical copula can substantially reduce the computation cost while features of the data are maintained with the preselected error threshold. In addition, we compare FLAME$^+$ with K-means to evaluate the clustering role in fuzzy empirical copula and the result has illustrated that FLAME$^+$ outperforms K-means in maintaining the dependence structure of the datasets.

Further work will be concerned with releasing the limitation of calculating the true value of original dataset and making the method more applicable.Though Copula has been widely applied to finance problems in the past decades, some areas such as intelligent robotics, artificial intelligence and automation require empirical copula and its variants being both effective ap-

Table 1
Number of clusters corresponding to number of nearest neighbors

Neighbors	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Clusters	1574	525	377	287	243	211	185	165	142	128	109	100	94	89	88	81	74	69	67	63

proaches, and practical and efficient algorithms. Fuzzy Empirical Copula has succeeded in overcoming the problem of computation cost of dependence structure estimation via Empirical Copula. The algorithm will be evaluated in more real-time or near real-time applications, priority will be given to human hand gesture recognition and object manipulation using robotic hands [38–40].

References

[1] A. Edmunds and A. Morris. The problem of information overload in business organisations: a review of the literature. *International Journal of Information Management*, 20(1):17–28, 2000.

[2] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

[3] K.V. Mardia, J.T. Kent, J.M. Bibby, et al. Multivariate analysis. 1979.

[4] J.H. Friedman, S.L.A. Center, and J.W. Tukey. A PROJECTION PURSUIT ALGORITHM FOR EXPLORATORY DATA ANALYSIS. *The Collected Works of John W. Tukey: Graphics 1965-1985*, 1988.

[5] P. Comon et al. Independent component analysis, a new concept. *Signal Processing*, 36(3):287–314, 1994.

[6] J. Karhunen, P. Pajunen, and E. Oja. The nonlinear PCA criterion in blind source separation: Relations with other approaches. *Neurocomputing*, 22(1-3):5–20, 1998.

[7] T. Hastie and W. Stuetzle. Principal Curves. *Journal of the American Statistical Association*, 84(406):502–516, 1989.

[8] T. Kohonen. *Self-Organizing Maps*. Springer, 2001.

[9] J. Karhunen. Nonlinear Independent Component Analysis. *ICA: Principles and Practice*, pages 113–134, 2001.

[10] P. Demartines and J. Herault. Curvilinear component analysis: a self-organizing neural networkfor nonlinear mapping of data sets. *Neural Networks, IEEE Transactions on*, 8(1):148–154, 1997.

[11] JA Hartigan and MA Wong. A K-means clustering algorithm. *JR Stat. Soc. Ser. C-Appl. Stat*, 28:100–108, 1979.

[12] JC Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Cybernetics and Systems*, 3(3):32–57, 1973.

[13] N. Belacel, P. Hansen, and N. Mladenovic. Fuzzy J-Means: a new heuristic for fuzzy clustering. *Pattern Recognition*, 35(10):2193–2200, 2002.

[14] P. VUORIMAA. FUZZY SELF-ORGANIZING MAP. *Fuzzy sets and systems*, 66(2):223–231, 1994.

[15] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proc. Natl. Acad. Sci. USA*, 96:2907–2912, 1999.

[16] KY Yeung, C. Fraley, A. Murua, AE Raftery, and WL Ruzzo. Model-based clustering and data transformations for gene expression data. *Bioinformatics*, 17(10):977–987, 2001.

[17] L. Fu and E. Medico. FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC Bioinformatics*, 8:3, 2007.

[18] J.D. FERMANIAN and O. SCAILLET. Nonparametric Estimation of Copulas for Time Series. 2003.

[19] R.B. Nelsen. An Introduction to Copulas (Springer Series in Statistics). 2006.

[20] A. Sklar. Fonctions de répartition a n dimensions et leurs marges. *Publ Inst Statist Univ Paris*, 8:229–231, 1959.

[21] P. Deheuvels. La fonction de dépendance empirique et ses propriétés: Un test non paramétrique dŠindépendance. *Bulletin de lŠAcadémie royale de Belgique: Classe des sciences*, 65:274–292, 1979.

[22] P. Deheuvels. A non parametric test for independence. *Publ. Inst. Statist. Univ. Paris*, 26(2):29–50, 1981.

[23] N. Kolev, U. Anjos, and B.V.M. Mendes. Copulas: A Review and Recent Developments. *Stochastic Models*, 22(4):617–660, 2006.

[24] A. Dias and P. Embrechts. Dynamic copula models for multivariate high-frequency data in finance. *Manuscript, ETH Zurich*, 2004.

[25] P. Embrechts, F. Lindskog, and A. McNeil. Modelling dependence with copulas and applications to risk management. *Handbook of Heavy Tailed Distributions in Finance*, 8:329–384, 2003.

[26] P.K. Trivedi and D.M. Zimmer. Copula Modeling: An Introduction for Practitioners. *Foundations and Trends® in Econometrics*, 1(1):1–111, 2006.

[27] L. Hu. Dependence patterns across financial markets: a mixed copula approach. *Applied Financial Economics*, 16(10):717–729, 2006.

[28] A. Kolesarova, R. Mesiar, J. Mordelova, and C. Sempi. Discrete Copulas. *Fuzzy Systems, IEEE Transactions on*, 14(5):698–705, 2006.

[29] B. De Baets and H. De Meyer. Orthogonal Grid Constructions of Copulas. *Fuzzy Systems, IEEE Transactions on*, 15(6):1053–1062, 2007.

[30] M.A.H. Dempster, E.A. Medova, S.W. Yang, and Judge Business School (Cambridge University. Empirical Copulas for CDO Tranche Pricing Using Relative Entropy. *International Journal of Theoretical and Applied Finance*, 10(4):679–701, 2007.

[31] J. Ma and Z. Sun. Dependence Structure Estimation via Copula. *Arxiv preprint arXiv:0804.4451*, 2008.

[32] P.A. Morettin, C.M.C. Toloi, C. Chiann, and J.C.S. de Miranda. Wavelet Smoothed Empirical Copula Estimators. 2008.

[33] W.H. Kruskal. Ordinal measures of association. *Journal of the American Statistical Association*, 53(284):814–861, 1958.

[34] E.L. Lehmann. Some concepts of dependence. *Ann. Math. Statist*, 37(1):137–1, 1966.

[35] WJ Nash, TL Sellers, SR Talbot, AJ Cawthorn, and WB Ford.

The population biology of abalone (Haliotis species) in Tasmania. I. Blacklip abalone (H. rubra) from the north coast and the Furneaux group of islands. *Sea Fisheries Division Technical Report*, 48:1–69, 1994.

[36] P. Horton and K. Nakai. A Probabilistic Classification System for Predicting the Cellular Localization Sites of Proteins. pages 109–115, 1996.

[37] A. Asuncion and D.J. Newman. UCI machine learning repository. 2007.

[38] H. Liu. A Fuzzy Qualitative Framework for Connecting Robot Qualitative and Quantitative Representations. *IEEE Transactions on Fuzzy Systems*, 16(6):1522–1530, 2008.

[39] H. Liu, D. Brown, and G. Coghill. Fuzzy Qualitative Robot Kinematics. *IEEE Transactions on Fuzzy Systems*, 16(3):808–822, 2008.

[40] Z. Ju, H. Liu, X. Zhu, and Y. Xiong. Dynamic Grasp Recognition Using Time Clustering, Gaussian Mixture Models and Hidden Markov Models. *Advanced Robotics*, 23(10):1359–1371, 2009.