Data field for mining big data

Shuliang Wanga,b, Ying Li and Dakui Wangc

Abstract
Big data is a highlighted challenge for many fields with the rapid expansion of large-volume, complex, and fast-growing sources of data. Mining from big data is required for exploring the essence of data and providing meaningful information. To this end, we have previously introduced the theory of physical field to explore relations between objects in data space and proposed a framework of data field to discover the underlying distribution of big data. This paper concerns an overview of big data mining by the use of data field. It mainly discusses the theory of data field and different aspects of applications including feature selection for high-dimensional data, clustering, and the recognition of facial expression in human–computer interaction. In these applications, data field is employed to capture the intrinsic distribution of data objects for selecting meaningful features, fast clustering, and describing variation of facial expression. It is expected that our contributions would help overcome the problems in accordance with big data.

1. Introduction
Big data has become an indispensable part of different fields with rapid growth of large-scale and multiple sources of data-sets. Massive data-sets have far exceeded human ability for complete storage, management, interpretation, and utility. The rapid advance in massive digital data leads to the growth of vast computerized data-sets at unprecedented rates (Li, Wang, and Li 2006). Remote sensing is playing an important role in gaining spatio-temporal information of dynamic resources and environment within a wide area of human activity. The remotely sensed techniques become multi-purpose, multi-sensor, multi-band, multi-resolution, and multi-frequency. They are further being marked with high space, high dynamic, high spectral, and high data capacity. National Aeronautics and Space Administration (NASA) Earth Observing System (EOS) is projected to generate some 50 gigabit of remotely sensed data per hour. Only the sensors of EOS-AM1 and -PM1 satellites have the ability to daily collect the remotely sensed data up to Terabyte level. Landsat satellites are able to obtain a global coverage of satellite imaging data every two weeks. Trillions of bytes of information are generated involving customers, suppliers, and operations; millions of networked sensors are being embedded in the physical world in devices such as mobile phones, smart energy meters, automobiles, and industrial machines that sense, create, and communicate data in the Internet of Things. It was estimated that enterprises globally stored more than 7 exabytes (1 exabyte of data is the equivalent of more than 4000 times the information stored in the US Library of Congress) of new data on disk drives in 2010, while consumers stored more than 6 exabytes of new data on devices (McKinsey Global Institute 2011). Furthermore, an exponential growth is continuously increasing and amassed in the velocity, variety, volume, and value of digital data every time. The huge amount of spatial data and their complex relations provide one of the crucial challenges to extract previously unknown, potentially useful and ultimately understandable knowledge (Li, Wang, and Li 2006).

In order to take full advantage of these data repositories, it is necessary to develop techniques for uncovering the physical nature of such huge data-sets on interacted objects. It is acknowledged that objects are mutually interacted in space. Physical nature was "a portion of everything in everything." Scientists have even proposed an idea of correlation for the twenty-first century in 2011. Visible or invisible particles are mutually interacted. Sometimes, the interaction is untouched between physical particles. Physical field describes the interaction among physical particles. Data space is an exact branch of physical space. From a generic physical space to an exact data space, the physical nature of the relationship between the objects is identical to that of the interaction...
between the particles. In the data space, all objects are mutually interacted via data that refer to object attributes as determined from observation. The rule of the attribute interaction may be the interactional mapping of objects distributing in the universe of discourse. If an object described with data is taken as a particle with mass, then data field can be derived from physical field (Wang et al. 2011). Inspired by physical field, data field is proposed to describe the interaction among data objects.

2. Principles of data field

Inspired by the physical field to portray the interaction between particles, data field is proposed to model the object interaction (Wang et al. 2011). In the light of data field, the object’s contribution to a given task diffuses from the local object to the global space. When the objects are observed on its specific attribute, the contribution actually diffuses from the universe of sample to the universe of population. In the case of clustering, a strength function of short-range field can enlarge the interactional strength between data objects nearby and reduce the strength for objects far away.

2.1. Definition

Data field is to virtually portray the interaction by diffusing the contributions from the sample space to the population space (Li, Wang, and Li 2006). Objects are depicted with attribute data in the computerized data space. The dimensions of data space correspond to the attributes of data objects. An object with \( n \) attributes becomes a data point in the \( n \)-dimensional data space. With the data field, the data point diffuses its contributions to a clustering task throughout the data space. The exact data field may be derived from the corresponding physical field, and more than one data field will be superposed on the same position in the data space.

Assume that there is a data space \( \Omega \subseteq \mathbb{R}^d \). In \( \Omega \), data-set \( D = \{X_1, X_2, \ldots, X_n\} \) denotes a \( d \)-dimensional independent random sample, where data object \( X_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \) with \( i = 1, 2, \ldots, n \). Similar to physics, each data object \( X_i \) is treated as a physical particle with a certain mass \( m \), that depicits the characteristics of \( X_i \). In the context of a task, \( X_i \) is completely diffuses its data contribution from its position to the whole data space \( \Omega \), that is, from the universe of sample to the universe of population. The diffusion on data contribution makes a virtual field encircling \( X_i \). The virtual field is called data field of \( X_i \).

To clearly present data field, Table 1 interprets the terms in this monograph.

### Table 1. Terms and interpretations.

| Terms                        | Interpretation                                                                 |
|------------------------------|-------------------------------------------------------------------------------|
| Data space \( \Omega \subseteq \mathbb{R}^d \) | that is \( d \)-dimension.                                                     |
| Data-set \( D = \{X_1, X_2, \ldots, X_n\} \) | that is an independent random sample in \( \Omega \).                           |
| Data object \( X_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \) | with \( i = 1, 2, \ldots, n \) in \( D \).                                     |
| Arbitrary object (point) \( x = [x_1, x_2, \ldots, x_n] \) in \( \Omega \) | to measure the distance between the given \( x \) and an arbitrary \( x \). |
| Physical field \( \forall x \in D \), if a certain physical parameter or mathematical function correspondingly has a determined value, there exists a field of the parameter or function in \( \Omega \). |
| Physical potential \( \varphi \) | The work to move a unit particle from a point to a referenced point in the given field. It is also the energy difference between the point and its referenced point in the given field. |
| Physical potential function \( \varphi \) | (a) a continuous, smooth and finite function; \( \varphi \) is a decreasing single-valued function of the distance. When the distance from the field source increases, the function value decreases. Simplified as \( \varphi \). |
| Data field \( X_i \) | Given a task, data object \( X_i \) diffuses its data contribution from its position to \( \Omega \) under physical fields. The diffusing data function makes a virtual field encircling \( X_i \), which is called data field. All data fields are superposed in \( \Omega \). |
| Potential of data field \( \varphi \) | Given a task, the contribution to move a unit object from a position to a referenced position in the given field. It is also the contribution difference between the object position and its referenced position in the given field. |
| Potential function of data field \( \varphi \) | Under physical field, the function may mathematically describe the potential of data field if it satisfies three conditions: \( \varphi \) is a continuous, smooth and finite function; \( \varphi \) is isotropic; \( \varphi \) is a decreasing single-valued function of the distance. When the distance from the field source increases, the function value decreases. Simplified as \( \varphi \). |
2.1.2. Superposed field

Data field superposes on the same position if many objects exist in the data space. Each data point has its data field spreading all over the data space. If many objects are observed, there will be a large amount of data points in the data space. The mutual interactions among lots of data points result in the fact that a data point not only eradicates its own data field, but also receives the data fields from others. In other words, all data fields will superpose on the same position in the data space.

All data fields are superposed in \( \Omega \). An object transforms its field to other objects, and it also receives all the fields from those. In data space, the resulting field from all objects is a virtually superposed field. Given a task in \( \Omega \), data object \( X_i (i = 1, 2, \ldots, n) \) diffuses its data contribution, and \( X_i (i = 1, 2, \ldots, n) \) is also diffused by the data contribution of \( X_j (j = 1, 2, \ldots, n, j \neq i) \) simultaneously. Sequentially, \( X_i \) not only creates its own data field, but also is in the data fields from other data objects \( X_j (j \neq i) \).

The superposed strength of all data field may reflect the distribution of data objects in data space. With the same parameters, data field strength is high in densely populated regions but low in sparse areas. Distribution of data items could be explored by analyzing the changing regularity of data field strength. For example, given appropriate parameters, nuclear field can describe the distribution of data objects inside data-sets in a way that maximizes the similarity between two nearby data objects, whereas minimizes the similarity for remote objects.

2.1.3. Isotropy and anisotropy

Given a task in multi-dimensional data space, anisotropy is generic for the contribution of sample diffusing throughout the universe of population. Isotropy is a special case of anisotropy when the property is the same in every directional dimensions. For example, it is isotropy when data field is in one-dimensional data space. Data field further becomes a generalized data field in multi-dimensional data space. The distribution of potential value is mainly determined via the impact factor, and the unit potential function has much smaller influence on the result of estimation.

An arbitrary task virtually feels a contribution with strength when it is put into such data field without touching the objects. According to their contributions to the given task, the field strength of each object is different, which uncovers different interaction between the task and the referenced objects. By using such the mutual interaction, objects may be self-organized under the given data-set and task. Some patterns that are previously unknown, potentially useful, and ultimately understood may be further extracted from data-sets. For example, in the data field, data objects are grouped into clusters by simulating their opposite movements under an internal force without any external forces.

2.2. Potential function

Data field is mathematically described with a potential function under the umbrella of physical fields. The function portrays how data contribution is diffused from the universe of sample to the universe of population. Assume that there is an arbitrary object \( x = [x_1, x_2, \ldots, x_n] \) in the data field generated by \( X_i (i = 1, 2, \ldots, n) \). Its potential value is not only proportional to the parameters, but also it is negatively correlated with the data field spreading all over the data space. If many data fields result in the fact that a data point not only eradicates its own data field, but also receives the data fields from others. In other words, all data fields will superpose on the same position in the data space.

In \( \Omega \subseteq \mathbb{R}^n \), the potential value of \( x \) in the data field of object \( X_i (i = 1, 2, \ldots, n) \) is defined as follows:

\[
\hat{\phi}(x) = m_i \times K \left( \frac{x - X_i}{\sigma} \right)
\]

where \( m_i \) is the mass of \( X_i \), \( \sigma (\sigma \in (0, +\infty)) \) is an impact factor, \( ||x - X_i|| \) is the distance between \( X_i \) and \( x \), \( K(x) \) is the unit potential function.

2.2.1. Impact factor

The impact factors \( \sigma \) control the interaction distance between data objects. \( \sigma \) are various, for example, radiation brightness, radiation gene, data amount, distance between the neighbor equipotential lines, grid density of Descartes coordinate, and so on. They all make their contributions to the data field.

Assume that the potential function is fixed. When the value of \( \sigma \) is small, the interaction between data objects is weak. Sequentially, the object is mainly impacted by its nearby neighbor objects. That is, the data field of object \( X_i \) is short ranged. For each data object, potential function \( \varphi_X (x) \) is a peak function that centers around \( X_i \) (\( i = 1, 2, \ldots, n \)). In extreme cases, the potential value of each position in the field is personal different. So small \( \sigma \) may sharpen the individual features of data objects to discover distinct personality patterns in the given tasks. Contrarily, when the value of \( \sigma \) is big, the interaction between data objects is strong. Sequentially, the object \( X_i \) is almost impacted by all other objects \( X_j \) (\( i, j = 1, 2, \ldots, n; i \neq j \)). That is, the data field of object \( X_i \) is long ranged. For each data object, potential function \( \varphi_X (x) \) is a basic function that changes slowly in a wide range. The potential value around \( X_i \) (\( i = 1, 2, \ldots, n \)) is big. In extreme cases, the potential value of each position in the field is almost the same. So, big \( \sigma \) may smooth the individual features of data objects to discover public common rules for the given task.

Therefore, the selection of the impact factor \( \sigma \) should make the potential distribution of data field perfectly demonstrate the underlying distribution of original
data-sets. Because it is stable and short range, data field should decrease much more rapidly as the distance $||x - X||$ increases.

### 2.2.2. Unit potential function

$K(x)$ expresses the law that data object $X_i$ always diffuses its data contribution in the same way with data field. In order to overcome some difficulties from proving the feasibility of the data field, the unit potential function $K(x)$ should satisfy the mathematical properties of $\int K(x)dx = 1$, $\int xK(x)dx = 0$, and $0 < R(K) = \int (x)^2dx < \infty$. Also the mass $m_i$ satisfies $\sum_{i=1}^{n} m_i = 1, m_i \geq 0$ and $\lim_{n \to \infty} n \sup_{1 \leq i \leq n} \{m_i\} = 1$.

**[Theorem 1]** If $\sum_{i=1}^{n} m_i = 1$ and $\lim_{n \to \infty} n \sup_{1 \leq i \leq n} \{m_i\} = 1$, then $\lim_{n \to \infty} \sum_{i=1}^{n} m_i = 1$.

All data fields are superposed in $\Omega$. On an arbitrary point in $\Omega$, the strength of data field is a superposed sum of all potential functions $\phi(x)$ from $D = \{X_1, X_2, ..., X_n\}$.

At the position of $x$, $\phi(x)$ is equivalent to the superposed sum of all potential functions $\phi(X_i)$ from $D = \{X_1, X_2, ..., X_n\}$.

$$\phi(x) = \frac{1}{n\sigma} \sum_{i=1}^{n} m_i \times K\left(\frac{x - X_i}{\sigma}\right) \quad (2)$$

If each data object is regarded as the same in terms of mass, the potential function is simplified in Equation (3) as a result.

$$\hat{\phi}(x) = \frac{1}{n\sigma} \sum_{i=1}^{n} K\left(\frac{x - X_i}{\sigma}\right) \quad (3)$$

It can be seen that Equation (3) is similar with the equation of kernel estimation.

### 2.2.3. Mean integrated squared error

In order to get a better understanding of potential value estimation, integrated squared error (ISE) is given to measure the estimation error between the underlying function $\varphi$ and its estimation $\hat{\varphi}$ in multi-dimensional data field.

$$\text{ISE}(\sigma) = \int_{-\infty}^{\infty} (\hat{\varphi}(x) - \varphi(x))^2dx \quad (4)$$

Ignoring the special samples of observation, it is reasonable to average $\text{ISE}(\sigma)$ of all possible observations for getting a mean integrated squared error (MISE). That is, $\text{MISE}(\sigma)$ is the expectation of $\text{ISE}(\sigma)$, ie $\text{MISE}(\sigma) = E(\text{ISE}(\sigma))$.

$$\text{MISE}(\sigma) = E(\text{ISE}(\sigma)) = \int_{-\infty}^{\infty} E((\hat{\varphi}(x) - \varphi(x))^2)dx \quad (5)$$

**[Theorem 2]** If Equation (2) is absolutely continuous and $\int \varphi''(x)^2dx < \infty$, the impact factor $\sigma$ is optimized as

$$\sigma_{opt} = \left(\frac{\sum_{i=1}^{n} m_i^2 R(K)}{\sigma_{opt}^2 R(\varphi'')}\right)^{1/5}.$$

### 2.3. Potential function estimation

Bandwidth selection methods of kernel estimation can be used to approximate the optimal impact factor $\sigma_{opt}$. A large body of literature exists on bandwidth selection for univariate kernel density estimation (Faraway and Jhun 1990; Hall and Marron 1987; Ruppert, Sheather, and Wand 1995). Cross-validation method, plug-in method, and maximal smoothing criterion are commonly used for bandwidth selection. Assumed that $f(x)$ is the kernel density estimation of density function $f$. Then we can see from Theorem (2) that the optimal bandwidth of kernel estimation $\sigma_f$ is

$$\sigma_f = \left(\frac{R(K)}{n\sigma_{opt}^4 R(\varphi'')}\right)^{1/5} \quad (6)$$

Then we can obtain $\sigma_{opt} = \left(\sum_{i=1}^{n} m_i^2\right)^{1/5} \sigma_f$.

As there is a constant difference between $\sigma_{opt}$ and $\sigma_f$ in Equation (6), bandwidth selection methods of kernel estimation can be used to approximate the optimal impact factor $\sigma_{opt}$. We used four different methods for approximating the impact factor, including likelihood cross-validation (Duong and Hazelton 2003; Roberts 1996), rule-of-thumb method (Silverman 1986), Sheather-Jones method (Jones, Marron, and Sheather 1996; Sheather and Jones 1991), and maximal smoothing principle (Terrell 1990).

The results of the above four different kernel estimation are shown in Figure 1, where the red line represents underlying function and the blue represents kernel estimation. Seen from Figure 1, Sheather-Jones' plug-in method and likelihood cross-validation method perform well, despite that Sheather-Jones' plug-in method is tendency to smooth insufficiently. Conversely, Silverman's rule-of-thumb method and maximal smoothing principle is tendency to excessive smoothing.

### 2.4. Generalized data field

For the multi-dimensional data field, the generalized potential function estimation is proposed in order to get a better estimation of the potential function.

#### 2.4.1. Definition of generalized data field

For the existent multi-dimensional data, if Equation (2) is used for estimating the potential value, it indicates that the impact factor $\sigma$ is isotropic. Namely, the impact factor $\sigma$ is supposed to have the same value when measured in different directions. Consequently, the energy of each observed data is evenly arranged in all directions. In fact, data objects usually have different properties along different axes. The impact factor $\sigma$ should be anisotropic. That is, when measured in different directions, different value on $\sigma$ should be obtained. Furthermore, when the data object has different variations corresponding to
different directions or all the data objects are almost in a low-dimensional manifold, to some extent, the result of estimation with Equation (2) is not appropriate. So, in order to get a better estimation of the potential value in field, we should construct the data field again which is called generalized data field. Considering that the impact is anisotropic, replace impact factor \( \sigma \) by a matrix \( H \). In this case, for any given point \( x \) in multi-dimensional space, the potential value estimation is defined as follows:

\[
\hat{\phi}(x) = \frac{1}{|H|} \sum_{i=1}^{n} m_i \times K(H^{-1}(x - X_i))
\]  

(7)

where \( K(x) \) is a multivariate potential function and \( H \) is a positive definite \( p \times p \) matrix that is a non-singular constant matrix. Let \( H = \sigma B \), where \( \sigma > 0, |B| = 1 \). That is,

\[
H = \sigma B = \sigma \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1d} \\ b_{21} & b_{22} & \cdots & b_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ b_{d1} & b_{d2} & \cdots & b_{dd} \end{pmatrix} = \begin{pmatrix} \sigma b_{11} & \sigma b_{12} & \cdots & \sigma b_{1d} \\ \sigma b_{21} & \sigma b_{22} & \cdots & \sigma b_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma b_{d1} & \sigma b_{d2} & \cdots & \sigma b_{dd} \end{pmatrix}
\]

To simplify the calculation, \( H \) can be restricted to a class of positive definite diagonal:

\[
H = \sigma B = \sigma \begin{pmatrix} b_{11} & 0 & \cdots & 0 \\ 0 & b_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_{dd} \end{pmatrix} = \begin{pmatrix} \sigma b_{11} & 0 & \cdots & 0 \\ 0 & \sigma b_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma b_{dd} \end{pmatrix}
\]

Therefore, the Equation (7) can also be written as

\[
\hat{\phi}(x) = \frac{1}{\sigma^p|A|} \sum_{i=1}^{n} m_i \times K\left( B^{-1}(x - X_i) / \sigma \right)
\]  

(8)

To calculate simply, \( H \) can also be restricted to a class of positive definite diagonal, and then the corresponding potential equation becomes

\[
\hat{\phi}(x) = \frac{1}{\sigma^p|A|} \sum_{i=1}^{n} m_i \times K\left( x - X_i \right)
\]  

(9)

where \( \sigma_j \) is the bandwidth in the \( j \)-th dimensional attributes. For example, if the data objects are of two attributes, then \( j = 1, 2 \).

In some cases, each data object is considered to be the same in terms of mass, Equation (7) is simplified as

\[
\hat{\phi}(x) = \frac{1}{n|H|} \sum_{i=1}^{n} K(H^{-1}(x - X_i))
\]  

(10)

2.4.2. Feasibility of generalized data field theory

For generalized data field, the potential function \( K(x) \) and the mass \( m_i \) \((i = 1, 2, \ldots, n)\) are also be restricted with some conditions as in one-dimensional data field.

\[
\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1d} \\ b_{21} & b_{22} & \cdots & b_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ b_{d1} & b_{d2} & \cdots & b_{dd} \end{pmatrix}, \sigma > 0, \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_d \end{pmatrix} = 1
\]

\[
\begin{pmatrix} \sigma b_{11} & 0 & \cdots & 0 \\ 0 & \sigma b_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma b_{dd} \end{pmatrix}
\]

[Theorem 3] For Equation (7), the potential function \( K(x) \) must satisfy the following conditions: \( \int K(x)dx = 1, \int xK(x)dx = 0, \int xx'^T K(x)dx = C_k I_p \) Also, the mass \( m_i \) \((i = 1, 2, \ldots, n)\) satisfies \( \sum_{i=1}^{n} m_i = 1, m_i \geq 0 \) and \( \lim_{n \to \infty, I_2 \leq I_4} \{m_i\} = 1 \). As
The optimal impact factor that minimizes AMISE ($\sigma$) about $k$ and $A$ is obtained as

$$
\sigma_{\text{opt}} = \left( \frac{p \int K(x)^{2}dx}{nC_{k} \min_{A} \int tr^{2}(A^{T}S_{c}A)dx} \right)^{1/(p+4)} \tag{11}
$$

Also, AMISE_{min} = n^{-2/(p+4)} \times (C_{k})^{2p/(p+4)} \times \left( \frac{1}{4} + \frac{1}{p} \right)^{4/(p+4)} \times \left[ \min_{A} \int tr^{2}(A^{T}S_{c}A)dx \right]^{p/(p+4)}.

where $S_{c}$ is the Hessian matrix of $\varphi$.

### 2.4.3. Potential function estimation of generalized data field

For the generalized data field, the methods of rule-of-thumb, Sheather-Jones’s plug-in, multi-dimensional maximal smoothing principle, and likelihood cross-validation are based on MCMC for estimating the impact factor. Figure 2 shows the result of kernel estimation, respectively, with rule-of-thumb method, cross-validation method, Sheather-Jones method, and maximal smoothing principle for a set of two-dimensional data. From Figure 2, it may be found that methods of Sheather-Jones’ plug-in and likelihood cross-validation are tendency to smooth insufficiently in generalized data field. In contrast, Silverman’s rule-of-thumb method and maximal smoothing principle are tendency to excessive smoothing. When insufficient smoothing is done, the resulting estimation is too rough and contains spurious features that are artifacts of the sampling process. When excessive smoothing is done, the important features of the underlying structure are smoothed away. In practice, some different values for impact factor can be tried to select an appropriate one.

#### 2.5. Choice of unity potential function

The unit potential function $K(x)$ is to express the law that $X_{i}$ always diffuses its data contribution in the same way in its data field. The selecting of the function format may have to consider such conditions as the feature of data object, the characteristic of data radiation, the standard of function determination, the application of data field, and so on (Li, Wang, and Li 2006).

In order to maintain the same minimum for different unit potential function $K_{1}$ and $K_{2}$, the following condition must be satisfied.

$$
\frac{\sum_{i=1}^{n_{1}} m_{1i}^{2}}{\sum_{i=1}^{n_{2}} m_{2i}^{2}} = \frac{\sigma_{K_{1}} R(K_{2})}{\sigma_{K_{2}} R(K_{1})}, \tag{12}
$$

In generalized data field, if the function $K$ satisfies the conditions that $\int K(x)dx = 1$, $\int xK(x)dx = 0$, and $\int xK(x)dx = C_{3} I_{3}$, the same conclusion can be obtained. Therefore, the asymptotic relative efficiency is measured by Equation (12).

By calculating $\sigma_{K_{1}} R(K_{2}) / \sigma_{K_{2}} R(K_{1})$ it can be acquired as $\sigma_{K_{1}} R(K_{2}) / \sigma_{K_{2}} R(K_{1}) \approx 1$. Also is $\sum_{i=1}^{n_{1}} m_{1i}^{2} / \sum_{i=1}^{n_{2}} m_{2i}^{2} \approx 1$.

From known as $\lim n \sum_{i=1}^{n} m_{1i}^{2} = 1$, if $n_{1}$ and $n_{2}$ are sufficiently large, we can get $n_{1}/n_{2} = 1$.

It can be seen that the choice of the specific potential function does not significantly influence the potential estimation as Figure 3. It may draw that the performance of potential function estimation is primarily determined by the choice of bandwidths and only in a minor way by the choice of potential function.

For the convergence of algorithms in practice, the corresponding potential value should rapidly decrease to 0 as the distance increases. For example, a potential function
of data field may simulate nuclear field. Referring to the nuclear field, \( K(x) \) is defined in the exponent form as
\[
K(x) = m_i \times e^{-(\frac{\|x_i-x\|}{u1D70E})^4},
\]
where \( m_i \geq 0 \) is the mass of data object \( X_i, k \in N \) is a distance index.

3. Applications with data field

In this section, a few applications based on data field are proposed for the data mining of big data. They are feature selection with data field, clustering with data field, and facial data field for expression face recognition.

3.1. Feature selection with data field

Feature selection is one of the important and frequently used techniques in data processing for data mining (Liu and Motoda 1998; Pyle 1999). It chooses a subset of relevant features from the set of the original attributes so as to improve the algorithmic performance. The selected features keep most of the useful information, while the abundant attributes are eliminated. Feature selection has been a fertile field of research and development since the 1970s in statistical pattern recognition (Ben-Bassat 1982; Mitra, Murthy, and Pal 2002), machine learning (Blum and Langley 1997; Kohavi and John 1997), and data mining (Dash et al. 2002) and widely applied to many fields such as text categorization, image retrieval customer relationship management, intrusion detection, and genomic analysis. Feature selection algorithms designed with different evaluation criteria mainly fall into two categories: Wrapper method (Caruana and Freitag 1994) and Filter method (Dash et al. 2002; Guyon and Elisseeff 2003; Hall 2000; Yu and Liu 2003). Wrapper method is common for clustering via a classification engine. But its evaluation function is required to be able to distinguish clustering results from different feature subspaces. And there is no generic criterion to estimate the clustering accuracy. The result depends on the specific algorithm. Filter method scores the selected features, ignoring classifier algorithm. It is independent of the specific algorithm, without selecting criterion or setting parameter.

3.1.1. Fundamentals on feature selection

The important attributes are selected as the features to depict the nature of objects by using generalized data field. Then potential entropy from generalized data field is proposed to measure the importance of feature subsets for finding out the optimal feature subset. The optimal feature subset is uncovered via implementing hierarchical search for important feature subsets. Figure 4 shows the flow chart of this algorithm.

3.1.2. Potential entropy to measure the importance of feature subsets

From all kinds of feature subsets, the important features with well-formed clusters can degrade entropy. Conversely, unimportant or noisy features can confuse the data distribution and make the entropy increase. The important feature subset can be extracted by assessing the entropy of feature subsets

The entropy may measure the uncertainty of objects distributing in data space \( \Omega_{fd} \). In face of observed data-sets, the distribution of objects is unknown, the number of clusters is undiscerned, and the structure of clustering is undiscovered at the beginning of clustering. From the perspective of information theory, it is hard to determine the distribution of objects. The determination is uncertain, which results in big entropy. The uncertainty of the determination is small, which also leads to small entropy.

Potential entropy (Pe) is presented to measure the importance of feature subsets for distinguishing feature subsets with clusters from feature subsets without clusters. Potential entropy is constructed via generalized data field. The potential value of \( X_{\phi}^F \) \((p=1, 2, \ldots, n)\) is \( \phi \) in the generalized data field. From \( \Omega_{fd} \) to \( \Omega_{\phi} \), \( X_{\phi}^A = [X_1^A, X_2^A, \ldots, X_m^A] \) is projected as \( X_{\phi} = [X_{\phi 1}^F, X_{\phi 2}^F, \ldots, X_{\phi n}^F] \). As to \( F \), there is a series of potentials, \( \phi_1^F, \phi_2^F, \ldots, \phi_n^F \). Then the potential entropy \( Pe(F) \) may measure the uncertainty of objects distribution in the context of \( F \), as

\[
Pe(F) = -\sum_{p=1}^{n} \frac{\phi_p^F}{Z} \times \log \left( \frac{\phi_p^F}{Z} \right).
\]

Because \( 0 < Pe(F) \leq \log(n) \), the importance of feature subset \( F \) can be defined as

\[
\text{Im}(F) = \log(n) - Pe(F) = \log(n) + \sum_{p=1}^{n} \frac{\phi_p^F}{Z} \times \log \left( \frac{\phi_p^F}{Z} \right),
\]

where

\[
\phi_p^F = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{1}{d} \sum_{j=1}^{d} K \left( \frac{X_{ij}^A - X_{ij}^F}{a_j} \right) \right\}, \quad \forall X_{ij}^A, X_{ij}^F \in X_{\phi}^F
\]

and

\[ Z = \sum_{i=1}^{n} \phi_p^F \]. Because the importance of feature...
subset will decrease when deleting any feature of the subset. $\text{Im}(F)$ is anti-monotonous, or it has the character of Apriori.

When the distribution of projected objects obeys uniform distribution, every potential value of the position that each object locates is approximately equal. Sequentially, the value to measure the corresponding feature's importance tends to 0. On the other hand, if the distribution of projected objects is very asymmetric, every potential value of the position that each object locates is also very asymmetric. The data field on such asymmetric objects will have smaller potential entropy, which results in larger value to measure the corresponding feature's importance. When calculating the potential entropy of data field, Sheather-Jones method (Jones, Marron, and Sheather 1996; Sheather and Jones 1991) is used to get the optimal impact factor $\sigma_F^*$. In the context of $\sigma_F^*$, the importance of the optimized feature subset $F$ can be obtained with Equation (13).

\[
\text{Im}(F)_{\text{opt}} = \text{Im}(F)_{\sigma=\sigma_F^*} = \left( \log(n) + \sum_{p=1}^{m} \phi_p^F(x) \times \log \left( \frac{\phi_p^F(x)}{Z} \right) \right)_{\sigma=\sigma_F^*}
\]

Before calculating the potential entropy, it is unnecessary to standardize the data because the feature variables $F = (f_1, f_2, \ldots, f_d)$ have been standardized by Sheather-Jones method when searching for $\sigma_F^*$. In practice, in order to increase the difference of importance among feature subsets, the impact factor $\sigma_F$ is usually selected to be less than the optimal value of impact factor, ie $\sigma_F = c \times \sigma_F^*$, where $c < 1$ (Wang et al. 2011).

### 3.1.3. Hierarchical search for important feature subset

The basic idea of clustering requires that the similarity between objects inside the same cluster is much bigger than that between objects belonging to different clusters. When the similarity between objects is close to each other or exactly equal, it is very difficult to decide to which cluster an object belongs. On the other hand, when the similarity between objects is decentralized or very asymmetric, it is easy to determine the distribution of objects. In order to find out the global optimal feature subset, it is necessary to evaluate and contrast all the feature subsets. However, it is unrealistic to exhaustively search for all the feature combinations for the optimal solution. The existed search algorithms may converge to the local optimal solution, for example, sequential forwards or backwards methods. Hierarchical search for important feature subsets is presented to improve the efficiency to seek the global optimal solution.

As $\text{Im}(F)$ has the character of Apriori, iterative hierarchical search method is used to search for the original attribute set's highest dimensional important feature subset on the basis of Apriori algorithm. Specifically, candidate feature subsets are used to search for important feature subset. Firstly, the algorithm needs to determine the one-dimensional important feature subsets as $L_1$. Then, $L_1$ is used to generate two-dimensional candidate feature subsets and find out all the two-dimensional important feature subsets as $L_2$. The algorithm iterates hierarchically until new important feature subset cannot be found. An iteration is divided into merging step and pruning step. For example, if $fd = k$, $k$-dimensional important feature subsets $L_k$ connect with itself to generate $(k+1)$-dimensional candidate feature subset. Specifically, if two important feature subsets $F_i^{(k)}$ and $F_j^{(k)}$ that belong to $L_k$ are not identical, but share $(k+1)$ identical feature variables, then they are merged to be a $(k+1)$-dimensional candidate feature subset. All these $(k+1)$-dimensional candidate feature subsets are denoted as $C_{k+1}$. Obviously, $C_{k+1}$ is the superset of $L_{k+1}$, which means that the member of $C_{k+1}$ is likely to be an important feature subset and all $(k+1)$-dimensional important feature subsets are definitely contained in $C_{k+1}$. After calculating every member's importance of $C_{k+1}$, if one member's importance is larger than its parents' importance, this member belongs to $L_{k+1}$. So the algorithm takes advantage of the Apriori’s property of the feature subset’s importance measure to prune unimportant feature subsets in $C_{k+1}$ and generates $L_{k+1}$ to search for higher dimensional important feature subsets. Finally,
the algorithm will get all dimensional important feature subsets and ensure the optimal subset by sorting them.

### 3.2. Clustering with data field

Based on data field, an object with attributes becomes a data point in the data space. Each sample point in the spatial database radiates its data contribution from the sample space to the popular space. For a data mining task, the data point spreads its contributions throughout the whole data space.

In the data field that describes the interactions among the objects, data objects are grouped into clusters by simulating their mutual interactions and opposite movements. For a clustering task in the data space, a data point is a \( n \)-dimensional independent random sample. Without external force, the data points will move in opposite direction with respect to the interaction on the basis of physical field, for which final aggregate to clusters. The data points gathered around the potential center of the data field have greater data field force. The further data points leave from the center of the data field, the smaller the data field force is. On the average, the distributed density of the data points reflects these changes in the data field force.

Given a data-set, the process of this clustering method is as follows:

**Input: data-set**

**Output: the clustering result**

**Process:**

1. The data field is generated according to the data-set, then the whole data field is divided into grid units (see Section 2.1.). The grid units can be viewed as the candidate set for cluster centers.
2. According to the nuclear radiation field-strength function, the potential value for every grid unit can be computed.
3. Traverse each grid unit, we can find all of the cluster centers by using spatial neighborhood discriminate algorithm.
4. Due to the cluster centers already recorded, the cluster results can be acquired.

Here, more details about spatial neighborhood discriminate algorithm will be given. The spatial neighborhood discriminate algorithm is the most important part in the spatial neighborhood clustering method.

After data field divided into grid units, they can be viewed as the candidate set for cluster centers. In the data field, potential center is a special point which has the maximum potential value in multi-dimensional space. The potential value of the points in the domain of neighborhood window of the maximum point is less than the maximum point’s. Using this property, the problem for finding the potential centers is transformed into finding the local maximum value of the grid units. It proposes the spatial neighborhood discriminate algorithm as follows:

**Input: data-set**

**Output: the cluster centers**

**Process:**

1. Establish the data field and divide it into grid units, then the grid units is the candidate set of the cluster centers.
2. Select any grid unit as the inspection point.
3. Specify the inspection point’s neighborhood window.
4. Compute the potential value of the points in this neighborhood window:
   1. If the potential value of this point is larger than the value of any other units in the domain of its neighborhood window, then the inspection point is the cluster center and records this value and its position. Otherwise, the inspection point is not the cluster center.
5. Select any other grid unit in the multi-dimensional grid units as the inspection point, and go back to Step 3 until all the grid units have been already detected.

In the multi-dimensional grid units, arbitrarily selecting a unit as the inspection unit, the neighborhood window of this unit will be analyzed. We specify the neighborhood window. One way to construct the window is to find the adjacent units of the selected unit. If its potential value is larger than any other units in the domain of its neighborhood window, then the inspection point is the cluster center and records this value and its position. Otherwise, the inspection point is not the cluster center.

6. Select any other grid unit in the multi-dimensional grid units as the inspection point, and go back to Step 3 until all the grid units have been already detected.

Seen from the experiment (Fang, Wang, and Jin 2010), it is a distinguishing feature that both cutting potential centers and spatial neighborhood determinant have to compute the potentials by scanning every data points throughout the whole space. However, the feature may become a common shortcoming to dramatically decrease the algorithmic performance when the amount of spatial data is too large, or the data distribute very unevenly. These scanning procedures cost a lot and cause...
is partitioned to form a number of grids at next lower layer. On the constructed hierarchical grids, the candidate area can be found out. Data field are generated for only the data points in the candidate area. From the potential centers of the data field, the clustering centers are discovered. With these clustering centers, the data points are finally clustered. The output is the clusters on the objects of the data points. Following is the pseudo-code of this algorithm.

**Algorithm: HGCUDF (Hierarchical Grid Clustering Using Data Field)**

**Input:** data-sets, start point, bottom  
**Output:** data clusters

1. for each layer from bottom to start do
2.    compute n, d, x of each grid
3.    end for
4. for i-layer from start point to bottom do
5.    compute the candidate area using heuristic approach
6.    if there is no candidate area then
7.        continue
8.    end if
9.    compute the candidate area for (i+1)-layer based on the candidate area of i-layer
10.   if the candidate area of next layer is smaller and the density d is bigger then
11.      continue
12.    else
13.      break
14.    end if
15.   end for
16. for each grid in candidate area do
17.    generate the data field
18.   end for
19. discover clustering centers by the neighborhood determinant method
20. output data clusters

A performance comparison is made between the proposed HGCUDF and the related clustering methods, i.e., cutting potential center with data field, spatial neighborhood determinant with data field, and CURE that is a typical hierarchical algorithm using representatives. In the four methods, cutting potential centers method, spatial neighborhood determinant method and HGCUDF are all based on the data field. When generating the data field, it takes a substantial portion of the total processing time. Cutting potential centers and spatial neighborhood determinant method is inefficient in data mining, especially when the data-set is very large. Thus, it is necessary to discover novel algorithms for clustering large data-sets. To improve the effectiveness and efficiency of clustering in large spatial databases, a novel clustering method called Hierarchical Grid Clustering Using Data Field (HGCUDF) is proposed in this paper, lightened by the subspace clustering algorithms (Parsons, Haque, and Liu 2004).

### 3.3. Hierarchical grid clustering

HGCUDF divides a data-set into clusters with hierarchical grids by using data field. The hierarchical grids are constructed under the distributed characteristics of data points. If the observed object has n attributes, the corresponding grids can be constructed with n dimensions and each dimension indicates different attributes of the object. The grid with much more intensive data points is selected into the candidate areas. For the refined candidate areas, the data field is generated in search of clustering centers. With the discovered clustering centers, all data points are clustered hierarchically (Figure 5).

#### 3.3.1. Algorithms and performance

The data space is divided into regular grids with hierarchies. Several different layers of such regular grids form a hierarchical structure. Each grid at a high layer
the candidate set of clustering centers. The abovementioned helps HGCUDF to save much more time again during the clustering process.

In addition, HGCUDF is relatively stable on time complexity and its computation complexity is an approximated linear in the size of the grids. CURE has quadratic time complexity in the volume of input for lower dimensional objects. Unlike CURE, HGCUDF does not deal with each data point but calculates some of them by the grid. For large amount of data, HGCUDF divides the data points into subsets which are calculated in a small set, instead of computing everyone in the large data-set. As the number of data points increase, the time consumed in HGCUDF is mainly cost to construct the hierarchical grids.

Therefore, the above results confirm that HGCUDF actually improves the efficiency. Its executed time is lower than others that may compute every data point in the whole data-set. And it is also implemented stably when the data points increase. It is practical for data mining in large amount of data points.

3.4. Data field for expressional face recognition

In the context of data field, a new facial feature abstraction and recognition method is proposed to express the influence at different levels of face’s different. During the process of automatic face recognition, face detection and orientation is to detect human face from complex background and segment it before processing. Feature abstraction and recognition is from unified human face image with geometry normalization and gray normalization. In order to care more about facial feature abstraction and recognition, the experimented face images are with simple background such as Japanese Female Facial Expression (JAFEE) database.

3.4.1. Facial data field

Let \( A_{r \times t} \) denote a gray image of human face based on \( r \times t \) pixel. Each pixel is considered as a data object in two-dimensional generalized data field and \( g_{pq} \) is the mass of data object, \( g_{pq} = A(p,q) \) from the gray level of facial image with \( i = 1, 2, \ldots, r; j = 1, 2, \ldots, t \). The data field of facial image is defined by the interaction of all the pixels in a two-dimensional image space. According to the equation of generalized potential function estimation, the potential function estimation of facial image can be calculated as

\[
\hat{\phi}(x) = \frac{1}{n} \sum_{p=1}^{r} \sum_{q=1}^{t} g_{pq} \exp \left( -\frac{||x - x_{pq}||^2}{2\sigma^2} \right)
\]  (14)

where \( \exp(\cdot) \) is \( e^{\cdot} \).

If the points with the same potential value are lined together, equipotential lines come into being. Furthermore, all the equipotential lines depict the interesting topological relationships among the objects,
Inefficient. In fact, only important features may help to show the underlying clustering structure of data distribution, improve the performance of clustering algorithms and reduce the misclassification.

In order to make full use of the data repositories, clustering becomes more and more indispensable. One of its bottlenecks is the performance of clustering algorithms. Preliminary method is cutting potential centers (Li, Wang, and Li 2006). When the data-set gets very large, its time cost to repeat the elimination will be unavailable. In order to overcome the shortcoming, the number of categories is naturally selected in the rule of actual data distribution, which generates the method of spatial neighborhood determinant (Fang, Wang, and Jin 2010). For further improving the performance, the hierarchical grids are constructed under the distributed characteristics of data points for clustering, ie HGCUDF. In order to rectify the improvement, the experiments are given from different viewpoints, as well as a complete comparison between the proposed HGCUDF and the existed methods.

Encouraged by the comparisons, HGCUDF will be studied further in the future. One is to keep the compressed data information in memory through the hierarchical grids. Both of these will speed up the processing of spatial data mining tremendously. All these advantages benefit from the hierarchical grid and parameters associated with them. Second, the parallelism framework is applied, for example, Hadoop, to speed up the processing of clustering. Third, some strategies can be applied when constructing the grid units to ensure that the number of grids is far less than that of data points.

As the use of face recognition systems expands towards less-restricted environments, the development of algorithms for view and illumination invariant face recognition becomes important. However, the performance of current algorithms degrades significantly when tested across pose and illumination as documented in a number of evaluations. The new recognition method

3.4.2. Fundamentals and algorithms

Face is recognized with data field hierarchy by hierarchy. Under the data field of data objects, data points can be divided into different classes according to the equipotential line. Considering a class as an entirety, this entirety represents all the data objects inside it, and each data object reflects the entirety in varying hierarchies. In $\Omega \subseteq \mathbb{R}^j$, the algorithms are implemented on different hierarchies. The procedures of face algorithm are described in Figure 7.

In the abovementioned flow of face recognition, the input is a set of human face images, the output is the recognized face, and the process is followed in such steps as normalizing faces, generating facial data field, extracting feature points in partitions, assigning weights, and recognizing faces.

4. Conclusions

Nowadays, the rapid advance in massive data-set results in the growth of vast computerized data-sets at unprecedented rates. It requires us to develop big data technologies for mining information from these massive data. Therefore, we propose a framework of data field for modeling and understanding hidden patterns of data. In addition, data field has been successfully used for selecting features, clustering and recognizing facial expressions, and so on.

Feature selection is one of the key questions for machine learning and pattern recognition, while high-dimensional feature selection is the difficult spot. The sparse and unsymmetrical data in high-dimensional space makes most popular clustering algorithms inefficient. In fact, only important features may help to show the underlying clustering structure of data distribution, improve the performance of clustering algorithm and reduce the misclassification.

In order to make full use of the data repositories, clustering becomes more and more indispensable. One of its bottlenecks is the performance of clustering algorithms. Preliminary method is cutting potential centers (Li, Wang, and Li 2006). When the data-set gets very large, its time cost to repeat the elimination will be unavailable. In order to overcome the shortcoming, the number of categories is naturally selected in the rule of actual data distribution, which generates the method of spatial neighborhood determinant (Fang, Wang, and Jin 2010). For further improving the performance, the hierarchical grids are constructed under the distributed characteristics of data points for clustering, ie HGCUDF. In order to rectify the improvement, the experiments are given from different viewpoints, as well as a complete comparison between the proposed HGCUDF and the existed methods.

Encouraged by the comparisons, HGCUDF will be studied further in the future. One is to keep the compressed data information in memory through the hierarchical grids. Both of these will speed up the processing of spatial data mining tremendously. All these advantages benefit from the hierarchical grid and parameters associated with them. Second, the parallelism framework is applied, for example, Hadoop, to speed up the processing of clustering. Third, some strategies can be applied when constructing the grid units to ensure that the number of grids is far less than that of data points.

As the use of face recognition systems expands towards less-restricted environments, the development of algorithms for view and illumination invariant face recognition becomes important. However, the performance of current algorithms degrades significantly when tested across pose and illumination as documented in a number of evaluations. The new recognition method...
in this paper may also offer a reference in expressionnal face recognition. For example, the complexity of facial expressions brings about high time complexity for neural network. If it is used together with the data field, the results may become better.

**Funding**

This study is supported by National Natural Science Fund of China [grant number 61472039].

**Notes on contributors**

**Shuliang Wang** is a Professor of the School of Software, Beijing Institute of Technology and the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University. His research interests include artificial intelligence, data mining and machine learning.

**Ying Li** is a PhD candidate of the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University. Her current research interests include big data mining, image analysis, and machine learning.

**Dakui Wang** is a PhD candidate in the International School of Software, Wuhan University. His current research interests include data mining and machine learning.

**References**

Ben-Bassat, M. 1982. "Pattern Recognition and Reduction of Dimensionality." In *Handbook of Statistics*, edited by P.R. Krishnaiah, 773–791. Oxford: Elsevier.

Blum, A. L., and P. Langley. 1997. "Selection of Relevant Features and Examples in Machine Learning." *Artificial Intelligence* 97 (1-2): 245–271.

Caruana, R., and D. Freitag. 1994. "Greedy Attribute Selection." In *Proceedings of the 11th International Conference on Machine Learning*, 28–36. New Brunswick, NJ: Morgan Kaufmann.

Dash, M., K. Choi, P. Scheuermann, and H. Liu. 2002. "Feature Selection for Clustering-a Filter Solution." In *Proceedings of the 2002 IEEE International Conference on Data Mining*, 115–122. Japan: Maebashi City.

Duong, T., and M. L. Hazelton. 2003. "Plug-in Bandwidth Matrices for Bivariate Kernel Density Estimation." *Journal of Nonparametric Statistics* 15 (1): 17–30.

Fang, M., S. L. Wang, and H. Jin. 2010. "Spatial Neighborhood Clustering Based on Data Field." In *Advanced Data Mining and Applications*, 262–269. Berlin Heidelberg: Springer.

Faraway, J. J., and M. Jhun. 1990. "Bootstrap Choice of Bandwidth for Density Estimation." *Journal of the American Statistical Association* 85 (412): 1119–1122.

Guyon, I., and A. Elisseeff. 2003. "An Introduction to Variable and Feature Selection." *The Journal of Machine Learning Research* 3: 1157–1182.

Hall, M. A. 2000. "Correlation-Based Feature Selection for Discrete and Numeric Class Machine Learning." In *Proceedings of 7th Intentional Conference on Machine Learning*, 359–366. San Diego, CA, December 11–13, 2008.

Hall, P., and J. S. Marron. 1987. "Estimation of Integrated Squared Density Derivatives." *Statistics & Probability Letters* 6 (2): 109–115.

Jones, M. C., J. S. Marron, and S. J. Sheather. 1996. "A Brief Survey of Bandwidth Selection for Density Estimation." *Journal of the American Statistical Association* 91 (433): 401–407.

Kohavi, R., and G. H. John. 1997. "Wrappers for Feature Subset Selection." *Artificial Intelligence 97* (1-2): 273–324.

Li, D. R., S. L. Wang, and D. Y. Li. 2006. *Spatial Data Mining Theories and Applications*. Beijing: Science Press. (in Chinese).

Liu, H., and H. Motoda. 1998. *Feature Selection for Knowledge Discovery and Data Mining*. Boston, MA: Kluwer Academic.

McKinsey Global Institute. 2011. "Big Data: The Next Frontier for Innovation, Competition, and Productivity." (Technical Report) May 2011, edited by J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers. http://www.mckinsey.com/business-functions/business-technology/our-insights/big-data-the-next-frontier-for-innovation

Mitra, P., C. A. Murthy, and S. K. Pal. 2002. "Unsupervised Feature Selection Using Feature Similarity." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (3): 301–312.

Parsons, L., E. Haque, and H. Liu. 2004. "Subspace Clustering for High Dimensional Data." *ACM SIGKDD Explorations Newsletter* 6 (1): 90–105.

Pyle, D. 1999. *Data Preparation for Data Mining*. Burlington, MA: Morgan Kaufmann.

Roberts, G. O. 1996. "Markov Chain Concepts Related to Sampling Algorithms." In *Markov Chain Monte Carlo in Practice*, edited by W. R. Gilks, S. Richardson, and D. J. E. Spiegelhalter, 45–57. London: Chapman and Hall.

Ruppert, D., S. J. Sheather, and M. P. Wand. 1995. "An Effective Bandwidth Selector for Local Least Squares Regression." *Journal of the American Statistical Association* 90 (432): 1257–1270.

Sheather, S. J., and M. C. Jones. 1991. "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation." *Journal of the Royal Statistical Society. Series B (Methodological)* 53: 683–690.

Silverman, B. W. 1986. *Density Estimation for Statistics and Data Analysis*. London: CRC Press.

Terrell, G. R. 1990. "The Maximal Smoothing Principle in Density Estimation." *Journal of the American Statistical Association* 85: 470–477.

Wang, S. L., W. Y. Gan, D. Y. Li, and D. R. Li. 2011. "Data Field for Hierarchical Clustering." *International Journal of Data Warehousing and Mining* 7 (4): 43–63.

Yu, L., and H. Liu. 2003. "Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution." In *Proceedings of the Twentieth International Conference on Machine Learning (ICML-03)*, 856–863. Washington, DC, August 21–24.