CARE: Finding Local Linear Correlations in High Dimensional Data

Xiang Zhang, Feng Pan, and Wei Wang
Department of Computer Science
University of North Carolina at Chapel Hill
Chapel Hill, NC 27599, USA
{xiang, panfeng, weiwang}@cs.unc.edu

***Abstract*— Finding latent patterns in high dimensional data is an important research problem with numerous applications. Existing approaches can be summarized into 3 categories: feature selection, feature transformation (or feature projection) and projected clustering. Being widely used in many applications, these methods aim to capture global patterns and are typically performed in the full feature space. In many emerging bio-medical applications, however, scientists are interested in the local latent patterns held by feature subsets, which may be invisible via any global transformation. In this paper, we investigate the problem of finding local linear correlations in high dimensional data. Our goal is to find the latent pattern structures that may exist only in some subspaces. We formalize this problem as finding strongly correlated feature subsets which are supported by a large portion of the data points. Due to the combinatorial nature of the problem and lack of monotonicity of the correlation measurement, it is prohibitively expensive to exhaustively explore the whole search space. In our algorithm, CARE, we utilize spectrum properties and effective heuristic to prune the search space. Extensive experimental results show that our approach is effective in finding local linear correlations that may not be identified by existing methods.**

I. Introduction

Many real life applications involve the analysis of high dimensional data. For example, in bio-medical domains, advanced microarray techniques [1], [2] allow to monitor the expression levels of hundreds to thousands of genes simultaneously. By mapping each gene to a feature, gene expression data can be represented by points in a high dimensional feature space. To make sense of such high dimensional data, extensive research has been done in finding the latent structure among the large number of features. In general, existing approaches in analyzing high dimensional data can be summarized into 3 categories [3]: feature selection, feature transformation (or feature projection) and projected clustering.

The goal of feature selection methods [4], [5], [6], [7] is to find a single representative subset of features that are most relevant for the task at hand, such as classification. The selected features generally have low correlation with each other but have strong correlation with the target feature.

Feature transformation methods summarize the dataset by creating linear combinations of features in order to uncover the latent structure. The commonly used feature transformation methods include principal component analysis (PCA) [8], linear discriminant analysis (LDA), and their variants (see [9] for an overview). PCA is one of the most widely used feature transformation methods. It seeks an optimal linear transformation of the original feature space such that most variance in the data is represented by a small number of orthogonal derived features in the transformed space. PCA performs one and the same feature transformation on the entire dataset. It aims to model the global latent structure of the data and hence does not separate the impact of any original features nor identify local latent patterns in some feature subspaces.

Recently proposed projected clustering methods, such as [10], [11], [12], can be viewed as combinations of clustering algorithms and PCA. These methods can be applied to find clusters of data points that may not exist in the axis parallel subspaces but only exist in the projected subspaces. The projected subspaces are usually found by applying the standard PCA in the full dimensional space. Like other clustering methods, projected clustering algorithms find the clusters of points that are spatially close to each other in the projected space. However, a subset of features can be strongly correlated even though the data points do not form any clustering structure.

A. Motivation

PCA is an effective way to determine whether a set of features, $F = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\}$, show strong correlation [8]. The general idea is as follows. If the features in F are indeed strongly correlated, then a few eigenvectors of the covariance matrix with the largest variance will describe most variance in the whole dataset. Only a small amount of variance is represented by the remaining eigenvectors. The variance on each eigenvector is its corresponding eigenvalue of the covariance matrix C_F of F. Therefore, if the sum of the smallest eigenvalues (i.e., the variance on the last few eigenvectors) is a small fraction of the sum of all eigenvalues (i.e., the variance in the original data), then the features in F are strongly correlated.

In many real life applications, however, it is desirable to find the *subsets of features* having strong linear correlations. For example, in gene expression data analysis, a group of genes having strong linear correlation is of high interests to biologists since it helps to infer unknown functions of genes and gives rise to hypotheses regarding the mechanism of the

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
p_1	0.7988	**3.8905**	0.6548	−0.6646	0.5536	1.3242	**−0.5532**	0.3158	**−1.1613**
p_2	0.8968	**1.3365**	−1.2484	0.5582	−1.5564	−0.1265	**0.2983**	1.3437	**−1.1098**
p_3	0.1379	**3.1503**	−0.5975	−1.1885	−0.2067	−0.7372	**−1.2266**	−2.2378	**0.2907**
p_4	−1.6191	**3.9939**	−0.4818	−0.7755	−0.4256	0.2137	**−0.1897**	1.2929	**−1.9102**
p_5	−1.6466	**−1.1069**	0.9834	0.271	0.4938	−0.4005	**−0.3017**	−0.3785	**1.3148**
p_6	0.4287	**−4.0447**	1.7621	1.535	−0.8709	0.0649	**0.957**	0.0025	**0.6653**
p_7	**−0.982**	**2.1536**	1.4274	−1.0523	**0.0798**	**−1.758**	**−0.5334**	**0.8846**	**−0.2751**
p_8	**−5.1084**	**2.7252**	0.9118	0.6256	**−0.5216**	**1.6867**	**−0.9011**	**0.5825**	**−0.023**
p_9	**10.9007**	**4.3305**	0.3268	−0.7976	**−1.4139**	**0.3274**	**−0.8926**	**−1.6142**	**−0.908**
p_{10}	**7.3744**	−0.8533	0.0696	−0.3135	**−0.3843**	**0.716**	0.2787	**−1.5037**	−1.0437
p_{11}	**−4.9437**	0.3456	−1.4998	−0.6022	**−0.4579**	**1.5986**	−0.7458	**0.5736**	0.3735
p_{12}	**9.7836**	0.1098	−0.4182	1.2591	**−0.2915**	**−2.0647**	1.6035	**−0.9105**	0.9015
p_{13}	**10.9429**	−1.133	−0.021	0.8585	**−0.3012**	**−0.7436**	0.5743	**−1.6313**	1.2785
p_{14}	**5.9643**	−0.6831	0.2284	−2.1053	**−1.5886**	**0.1762**	0.3207	**−0.3591**	−0.1285
p_{15}	**−2.0985**	−0.2779	−1.0082	−0.3609	**1.0943**	**0.5278**	−0.1514	**−0.3976**	0.6128

Fig. 1. An example dataset

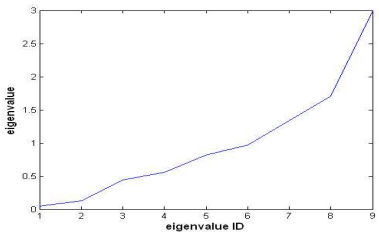


Fig. 2. Eigenvalues of the example dataset

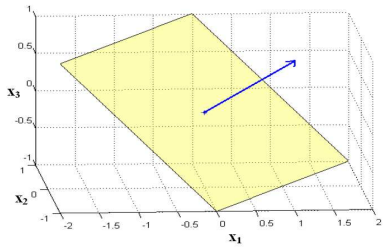


Fig. 3. Hyperplane determined by vector $[1, -1, 1]^T$

transcriptional regulatory network [1], [2]. We refer to such correlation among a subset of features in the dataset as a ***local linear correlation*** in contrast to the global correlation found by the full dimensional feature transformation methods.

For example, Figure 1 shows a dataset consisting of 9 features and 15 data points. Among the 9 features, $\{\mathbf{x}_2, \mathbf{x}_7, \mathbf{x}_9\}$ have local linear correlation $2\mathbf{x}_2 + 6\mathbf{x}_7 + 3\mathbf{x}_9 = 0$ on point set $\{\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_9\}$, and $\{\mathbf{x}_1, \mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_8\}$ have local linear correlation $\mathbf{x}_1 + 3\mathbf{x}_5 + 2\mathbf{x}_6 + 5\mathbf{x}_8 = 0$ on point set $\{\mathbf{p}_7, \mathbf{p}_8, \cdots, \mathbf{p}_{15}\}$ with i.i.d. gaussian noise of mean 0 and variance 0.01. The eigenvalues of the example dataset is shown in Figure 2.

Figure 2 tells us that the features in the example dataset are somehow correlated, since the smallest eigenvalues are much smaller than the largest ones.

Eigenvectors	Linear correlations reestablished
$\mathbf{v}_1$	$-0.4775\mathbf{x}_1 + 0.4311\mathbf{x}_2 + 0.1018\mathbf{x}_3 - 0.1516\mathbf{x}_4$ $-0.1185\mathbf{x}_5 + 0.1318\mathbf{x}_6 + 0.6215\mathbf{x}_7 - 0.3437\mathbf{x}_8$ $-0.1312\mathbf{x}_9 = 0$
$\mathbf{v}_2$	$-0.4503\mathbf{x}_1 - 0.3533\mathbf{x}_2 - 0.0432\mathbf{x}_3 + 0.1931\mathbf{x}_4$ $-0.0460\mathbf{x}_5 - 0.2823\mathbf{x}_6 - 0.1219\mathbf{x}_7 - 0.4577\mathbf{x}_8$ $-0.5703\mathbf{x}_9 = 0$
$\mathbf{v}_3$	$-0.2072\mathbf{x}_1 + 0.3259\mathbf{x}_2 - 0.0742\mathbf{x}_3 + 0.4307\mathbf{x}_4$ $-0.5181\mathbf{x}_5 - 0.2438\mathbf{x}_6 - 0.4166\mathbf{x}_7 - 0.0333\mathbf{x}_8$ $+0.3966\mathbf{x}_9 = 0$

TABLE I

LINEAR CORRELATIONS REESTABLISHED BY FULL DIMENSIONAL PCA

To get the linear correlation identified by PCA, we can apply the following approach. Note that this approach has been adopted in [12] to derive the quantitative descriptions for projected clusters. As a basic concept of linear algebra, a hyperplane is a subspace of co-dimension 1 [8]. Each vector $\mathbf{a} = [a_1, a_2, \cdots, a_n]^T$ in an n-dimensional linear space uniquely determines a hyperplane $a_1\mathbf{x}_1 + a_2\mathbf{x}_2 + \cdots + a_n\mathbf{x}_n = 0$ through the origin and orthogonal to $\mathbf{a}$. For example, Figure 3 shows the hyperplane $\mathbf{x}_1 - \mathbf{x}_2 + \mathbf{x}_3 = 0$ that is orthogonal to vector $[1, -1, 1]^T$. Therefore, a straightforward way to discover the correlations by full dimensional PCA is to compute the hyperplanes that are orthogonal to the eigenvectors with smallest eigenvalues (variances).

Using the example dataset, Table I shows the hyperplanes (linear correlations) determined by the 3 eigenvectors with the smallest eigenvalues. Clearly, none of them captures the embedded correlations. This is because PCA does not separate the impact of different feature subsets that are correlated on different subsets of points.

Recently, many methods [1], [13] have been proposed for finding clusters of features that are pair-wisely correlated. However, a set of features may have strong correlation but each pair of features only weakly correlated.

For example, Figure 4 shows 4 genes that are strongly correlated in the mouse gene expression data collected by the biologists in the School of Public Health at UNC. All of these 4 genes have same Gene Ontology (GO) [14] annotation *cell part*, and three of which, Myh7, Hist1h2bk, and Arntl, share the same GO annotation *intracelluar part*. The linear relationship identified by our algorithm is $-0.4(Nrg4) + 0.1(Myh7) + 0.7(Hist1h2bk) - 0.5(Arntl) = 0$. As we can see from the figure, all data points almost perfectly lay on the same hyperplane, which shows that the 4 genes are strong correlated. (In order to visualize this 3-dimensional hyperplane, we combine two features, Nrg4 and Myh7, into a single axis as $-0.4(Nrg4) + 0.1(Myh7)$ to reduce it to a 2-dimensional hyperplane.) If we project the hyperplane onto 2 dimensional spaces formed by each pair of genes, we find none of them show strong correlation, as depicted in Figures 5(a) to 5(c).

Projected clustering algorithms [10], [11] have been proposed to find the clusters of data points in projected feature spaces. This is driven by the observation that clusters may

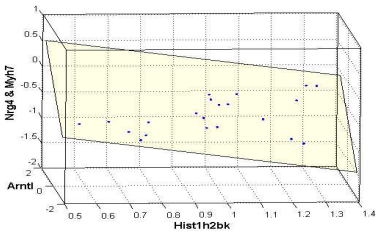
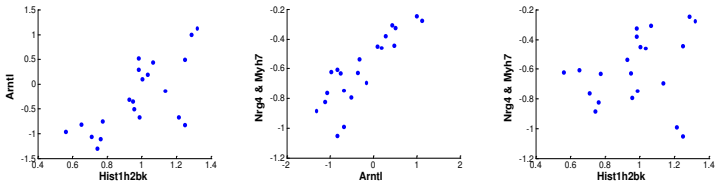


Fig. 4. A strongly correlated gene subset



(a) (Hist1h2bk, Arntl) (b) (Arntl, Nrg4&Myh7) (c) (Hist1h2bk,Nrg4&Myh7)

Fig. 5. Pair-wise correlations of a strongly correlated gene subset

exist in arbitrarily oriented subspaces. Like other clustering methods, these methods tend to find the clusters of points that are spatially close to each other in the feature space. However, as shown in Figure 4, a subset of features (genes in this example) can still be strongly correlated even if the data points are far away from each other. This property makes such strong correlations invisible to the projected clustering methods. Moreover, to find the projections of original features, projected clustering methods apply PCA in the full dimensional space. Therefore they cannot decouple the local correlations hidden in the high dimensional data.

B. Challenges and Contributions

In order to find the local linear correlations, a straightforward approach is to apply PCA to all possible subsets of features to see if they are strongly correlated. This approach is infeasible due to the large number of possible feature combinations. For example, given a 100-dimensional dataset, the number of feature subsets need to be checked is 2^{100}.

Real life datasets often contain noises and outliers. Therefore, a feature subset may be correlated only on a subset of the data points. In order to handle this situation, it is reasonable to allow the algorithm to find the local linear correlations on a large portion of the data points. This makes the problem even harder since for a fixed subset of features, adding (or deleting) data points can either increase or decrease the correlation among them. More details about the computational challenges of finding local linear correlations can be found in Section III.

In this paper, we investigate the problem of finding local linear correlations in high dimensional data. This problem is formalized as finding ***strongly correlated feature subsets***. Such feature subsets show strong linear correlations on a large portion of the data points. We examine the computational challenges of the problem and develop an efficient algorithm, CARE[1], for finding local linear correlations. CARE utilizes spectrum properties about the eigenvalues of the covariance matrix, and incorporates effective heuristic to improve the efficiency. Extensive experimental results show that CARE can effectively identify local linear correlations in high dimensional data, which cannot be found by applying existing methods.

II. Related work

The goal of feature transformation (or projection) methods, such as PCA, is to find linear combinations of original features in order to uncover the latent structures hidden in the data. Feature transformation methods can be further divided into supervised methods and unsupervised methods. Principal component analysis (PCA) is a representative unsupervised projection method. PCA finds the eigenvectors which represent the directions with maximal variances of the data by performing singular value decomposition (SVD) to the data matrix [8]. Supervised methods take the target feature into consideration. Existing supervised methods include linear regression analysis (LRA) [15], linear discriminant analysis (LDA) [16], principal component regression (PCR) [17], supervise probabilistic PCA (SPPCA) [18] and many others [9]. LRA and LDA find the linear combinations of the input (predictor) features which best explain the target (dependent) feature. In these methods, the input features are generally assumed to be non-redundant, i.e., they are linearly independent. If there are correlations in the input features, PCR first applies PCA to the input features. The principal components are then used as predictors in standard LRA. SPPCA extends PCA to incorporate label information. These feature transformation methods perform one and the same feature transformation for the entire dataset. It does not separate the impact of any original features nor identify local correlations in feature subspaces.

Feature selection methods [4], [5], [6], [7] try to find a subset of features that are most relevant for certain data mining task, such as classification. The selected feature subset usually contains the features that have low correlation with each other but have strong correlation with the target feature. In order to find the relevant feature subset, these methods search through various subsets of features and evaluate these subsets according to certain criteria. Feature selection methods can be further divided into two groups based on their evaluation criteria: wrapper and filter. Wrapper models evaluate feature subsets by their predictive accuracy using statistical re-sampling or cross-validation. In filter techniques, the feature subsets are evaluated by their information content, typically statistical dependence or information-theoretic measures. Similar to feature transformation, feature selection finds one feature subset for the entire dataset.

Subspace clustering is based on the observation that clusters of points may exist in different subspaces. Many methods [19], [20], [21] have been developed to find clusters in axes paralleling subspaces. Recently, the projected clustering was

[1]CARE stands for finding loCAl lineaR corrElations.

studied in [10], [11], inspired by the observation that clusters may exist in arbitrarily oriented subspaces. These methods can be treated as combinations of clustering algorithms and PCA. Similar to other clustering methods, these methods tend to find the clusters of points that are close to each other in the projected space. However, as shown in Figure 4, a subset of features still can be strongly correlated even if the data points are far away from each other. Pattern based bi-clustering algorithms have been studied in [1], [13]. These algorithms find the clusters in which the data points share pair-wise linear correlations, which is only a special case of linear correlation.

III. Strongly Correlated Feature Subset

In this section, we formalize the problem and study its computational challenges.

A. Problem Definition

Let $D = A \times B$ be a data matrix consisting of M N-dimensional data points, where the feature set $A = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ and the point set $B = \{\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_M\}$. Figure 1 shows an example dataset with 15 points and 9 features.

A strongly correlated feature subset is a subset of features that show strong linear correlation in a large portion of data points.

Definition 1: Let $F = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\} \times \{\mathbf{p}_{j_1}, \cdots, \mathbf{p}_{j_m}\}$ be a submatrix of D, where $1 \leq i_1 < i_2 < \cdots < i_n \leq N$ and $1 \leq j_1 < j_2 < \cdots < j_m \leq M$. C_F is the covariance matrix of F. Let $\{\lambda_l\}$ $(1 \leq l \leq n)$ be the eigenvalues of C_F and arranged in increasing order[2], i.e., $\lambda_1 \leq \lambda_2, \cdots, \leq \lambda_n$. The features $\{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\}$ is a **strongly correlated feature subset** if the value of the objective function $f(F,k) = \frac{\Sigma_{t=1}^{k}\lambda_t}{\Sigma_{t=1}^{n}\lambda_t} \leq \epsilon$ and $m/M \geq \delta$, where k, ϵ and δ are user specified parameters.

Eigenvalue λ_l is the variance on eigenvector $\mathbf{v}_l$ [8]. The set of eigenvalues $\{\lambda_l\}$ of matrix C_F is also called the *spectrum* of C_F [22].

Geometrically, each $m \times n$ submatrix of D represents an n-dimensional space with m points in it. This n-dimensional space can be partitioned into two subspaces, S_1 and S_2, which are orthogonal to each other. S_1 is spanned by the k eigenvectors with smallest eigenvalues and S_2 is spanned by the remaining $n-k$ eigenvectors. Intuitively, if the variance in subspace S_1 is small (equivalently the variance in S_2 is large), then the feature subset is strongly correlated. The input parameters k and threshold ϵ for the objective function $f(F,k) = \frac{\Sigma_{t=1}^{k}\lambda_t}{\Sigma_{t=1}^{n}\lambda_t}$ are used to control the strength of the correlation among the feature subset. The default value of k is 1. The larger the value of k, the stronger the linear correlation.

The reason for requiring $m/M \geq \delta$ is because a feature subset can be strongly correlated only in a subset of data points. In our definition, we allow the strongly correlated feature subsets to exist in a large portion of the data points in order to handle this situation. Note that it is possible that a data point may participate in multiple local correlations held by different feature subsets. This makes the local correlations more difficult to detect. Please also note that for a given strongly correlated feature subset, it is possible that there exist multiple linear correlations on different subsets of points. In this paper, we focus on the scenario where there exists only one linear correlation for a strongly correlated feature subset.

[2]In this paper, we assume that the eigenvalues are always arranged in increasing order. Their corresponding eigenvectors are $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$.

Feature subset	$\{\mathbf{x}_2, \mathbf{x}_7, \mathbf{x}_9\}$
Eigenvalues of C_F	$\lambda_1 = 0.001, \lambda_2 = 0.931, \lambda_3 = 2.067$
Input parameters	$k = 1$, $\epsilon = 0.004$ and $\delta = 60\%$
Objective function value	$f(F,k) = 0.0003$

TABLE II

An example of strongly correlated feature subset

For example, in the dataset shown in Figure 1, the features in submatrix $F = \{\mathbf{x}_2, \mathbf{x}_7, \mathbf{x}_9\} \times \{\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_9\}$ is a strongly correlated feature subset when $k = 1$, $\epsilon = 0.004$ and $\delta = 60\%$. The eigenvalues of the covariance matrix, C_F, the input parameters and the value of the objective function are shown in Table II.

In real world applications, it is typical that many local correlations co-exist, each of which involves a small number of features. Thus, it is reasonable to set the maximum size, max_s, of the feature subsets to be considered for each local correlation[3]. The co-occurrence of multiple local correlations poses serious challenge, since neither the feature subsets nor the supporting data points of these correlations are independent. It is crucial to decouple the compound effects of different local correlations.

Our goal is to find all strongly correlated feature subsets in the database D. This problem is computationally challenging. In the following subsection, we study the properties concerning the monotonicity of the objective function with respect to the feature subsets and point subsets separately.

B. Monotonicity of the Objective Function

1) Monotonicity with respect to feature subsets: The following theorem concerning the spectrum of covariance matrix developed in the matrix theory community is often called the *interlacing eigenvalues theorem*[4] [22].

Theorem 3.1: Let $F = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\} \times \{\mathbf{p}_{j_1}, \cdots, \mathbf{p}_{j_m}\}$ and $F' = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}, \mathbf{x}_{i_{(n+1)}}\} \times \{\mathbf{p}_{j_1}, \cdots, \mathbf{p}_{j_m}\}$ be two submatrices of D. C_F and $C_{F'}$ are their covariance matrices with eigenvalues $\{\lambda_l\}$ and $\{\lambda_l^{'}\}$. We have

$$\lambda_1^{'} \leq \lambda_1 \leq \lambda_2^{'} \leq \lambda_2 \leq \cdots \leq \lambda_{n-1} \leq \lambda_n^{'} \leq \lambda_n \leq \lambda_{n+1}^{'}.$$

Theorem 3.1 tells us that the spectra of C_F and $C_{F'}$ interleave each other, with the eigenvalues of the larger matrix bracketing those of the smaller one.

By applying the interlacing eigenvalues theorem, we have the following property for the strongly correlated feature subsets.

[3]Setting the maximum size of the feature subsets is also used in many other feature selection and feature transformation methods [5], [8].

[4]This theorem also applies to Hermitian matrix [22]. Here we focus on the covariance matrix, which is semi-positive definite and symmetric.

Point subset $P_1 = \{\mathbf{p}_1, \cdots, \mathbf{p}_{15}\}$	
Feature subset X_1	$f(F_1, k) = 0.1698$
Feature subset $X_1 \cup \{\mathbf{x}_9\}$	$f(F_1', k) = 0.0707$
Feature subset $X_1 \cup \{\mathbf{x}_4, \mathbf{x}_9\}$	$f(F_1'', k) = 0.0463$

TABLE III

MONOTONICITY WITH RESPECT TO FEATURE SUBSETS

Feature subset $X_2 = \{\mathbf{x}_2, \mathbf{x}_7, \mathbf{x}_9\}$	
Point subset P_2	$f(F_2, k) = 0.0041$
Point subset $P_2 \cup \{\mathbf{p}_{10}\}$	$f(F_2', k) = 0.0111$
Point subset $P_2 \cup \{\mathbf{p}_{14}\}$	$f(F_2'', k) = 0.0038$

TABLE IV

NO MONOTONICITY WITH RESPECT TO POINT SUBSETS

Property 3.2: (Upward closure property of strongly correlated feature subsets) Let $F = X \times P$ and $F' = X' \times P$ be two submatrices of D with $X \subseteq X'$. If X is a strongly correlated feature subset, then X' is also a strongly correlated feature subset.

Proof: We show the proof for the case where $|X'| = |X| + 1$, i.e., X is a subset of X' by deleting one feature from X'. Let C_F and $C_{F'}$ be the covariance matrices of F and F' with eigenvalues $\{\lambda_l\}$ and $\{\lambda_l^{'}\}$. Since X is a strongly correlated feature subset, we have $f(F,k) = \frac{\Sigma_{t=1}^{k}\lambda_t}{\Sigma_{t=1}^{n}\lambda_t} \leq \epsilon$. By applying the interlacing eigenvalues theorem, we have $\Sigma_{t=1}^{k}\lambda_t \geq \Sigma_{t=1}^{k}\lambda_t^{'}$ and $\Sigma_{t=1}^{n}\lambda_t \leq \Sigma_{t=1}^{n+1}\lambda_t^{'}$. Thus $f(F',k) = \frac{\Sigma_{t=1}^{k}\lambda_t^{'}}{\Sigma_{t=1}^{n+1}\lambda_t^{'}} \leq \epsilon$. Therefore, X' is also a strongly correlated feature subset. By induction we can prove for the cases where X is a subset of X' by deleting more than one feature. ■

The following example shows the monotonicity of the objective function with respect to the feature subsets. Using the dataset shown in Figure 1, let $F_1 = X_1 \times P_1 = \{\mathbf{x}_2, \mathbf{x}_7\} \times \{\mathbf{p}_1, \cdots, \mathbf{p}_{15}\}$, $F_1' = (X_1 \cup \{\mathbf{x}_9\}) \times P_1$, and $F_1'' = (X_1 \cup \{\mathbf{x}_4, \mathbf{x}_9\}) \times P_1$. The values of the objective function, when $k = 1$, are shown in Table III. It can be seen from the table that the value of the objective function monotonically decreases when adding new features.

Property 3.2 shows that for a fixed set of points, if a subset of features are strongly correlated, then all of its supersets are also strongly correlated. Therefore, in our algorithm, we can focus on finding ***the minimum strongly correlated feature subsets***, of which no subset is strongly correlated.

2) Lack of monotonicity with respect to point subsets: For a fixed feature subset, adding (or deleting) data points may cause the correlation of the features to either increase or decrease. That is, the objective function is non-monotonic with respect to the point subsets. The following property states this fact.

Property 3.3: Let $F = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\} \times \{\mathbf{p}_{j_1}, \cdots, \mathbf{p}_{j_m}\}$ and $F^{'} = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\} \times \{\mathbf{p}_{j_1}, \cdots, \mathbf{p}_{j_m}, \mathbf{p}_{j_{(m+1)}}\}$ be two submatrices of D. $f(F,k)$ can be equal to, or less than, or greater than $f(F',k)$.

We use the following example to show the non-monotonicity of the objective function with respect to the point subsets. Using the dataset shown in Figure 1, let $F_2 = X_2 \times P_2 = \{\mathbf{x}_2, \mathbf{x}_7, \mathbf{x}_9\} \times \{\mathbf{p}_1, \cdots, \mathbf{p}_9, \mathbf{p}_{11}\}$, $F_2' = X_2 \times (P_2 \cup \{\mathbf{p}_{10}\})$, and $F_2'' = X_2 \times (P_2 \cup \{\mathbf{p}_{14}\})$. The values of their objective functions, when $k = 1$, are shown in Table IV. It can be seen from the table that the value of the objective function f can either increase or decrease when adding more points.

In summary, the value of the objective function will monotonically decrease when adding new features. On the other hand, adding new points can either increase or decrease the value of the objective function.

IV. CARE

In this section, we present the algorithm CARE for finding the minimum strongly correlated feature subsets. CARE enumerates the combinations of features to generate candidate feature subsets. To examine if a candidate is a strongly correlated feature subset, CARE adopts a 2-step approach. It first checks if the feature subset is strongly correlated on all data points. If not, CARE then apply point deletion heuristic to find the appropriate subset of points on which the current feature subset may become strongly correlated. In Section IV-A, we first discuss the overall procedure of enumerating candidate feature subsets. In Section IV-B, we present the heuristics for choosing the point subsets for the candidates that are not strongly correlated on all data points.

A. Feature Subsets Selection

For any submatrix $F = X \times \{\mathbf{p}_1, \cdots, \mathbf{p}_M\}$ of D, in order to check whether feature subset X is strongly correlated, we can perform PCA on F to see if its objective function value is lower than the threshold, i.e., if $f(F,k) = \frac{\Sigma_{t=1}^{k}\lambda_t}{\Sigma_{t=1}^{n}\lambda_t} \leq \epsilon$.

Starting from feature subsets containing a single feature, CARE adopts depth first search to enumerate combinations of features to generate candidate feature subsets. In the enumeration process, if we find that a candidate feature subset is strongly correlated by evaluating its objective function value, then all its supersets can be pruned according to Property 3.2.

Next, we present an upper bound on the value of the objective function, which can help to speed up the evaluation process. The following theorem shows the relationship between the diagonal entries of a covariance matrix and its eigenvalues [22].

Property 4.1: Let F be a submatrix of D and C_F be the $n \times n$ covariance matrix of F. Let $\{a_i\}$ be the diagonal entries of C_F arranged in increasing order, and $\{\lambda_i\}$ be the eigenvalues of C_F arranged in increasing order. Then $\Sigma_{t=1}^{s}a_t \geq \Sigma_{t=1}^{s}\lambda_t$ for all $s = 1, 2, \cdots, n$, with equality held for $s = n$.

Applying Property 4.1, we can get the following proposition.

Proposition 4.2: Let F be a submatrix of D and C_F be the $n \times n$ covariance matrix of F. Let $\{a_i\}$ be the diagonal entries of C_F and arranged in increasing order. If $\frac{\Sigma_{t=1}^{k}a_t}{\Sigma_{t=1}^{n}a_t} \leq \epsilon$, then we have $f(F,k) \leq \epsilon$, i.e., the feature subset of F is a strongly correlated feature subset.

The proof of Proposition 4.2 is straightforward and omitted here. This proposition gives us an upper bound of the objective function value for a given submatrix of D. For any submatrix $F = X \times \{\mathbf{p}_1, \cdots, \mathbf{p}_M\}$ of D, we can examine the diagonal entries of the covariance matrix C_F of F to get the upper bound of the objective function. The computational cost of calculating of this upper bound is much less than that of evaluating the objective function value directly by PCA. Therefore, before evaluating the objective function value of a candidate feature subset, we can check the upper bound in Proposition 4.2. If the upper bound is no greater than the threshold ϵ, then we know that the candidate is a strongly correlated feature subset without performing PCA on its covariance matrix.

B. Choosing the Subsets of Points

In the previous subsection, we discussed the procedure of generating candidate feature subsets. A feature subset may be strongly correlated only on a subset of the data points. As discussed in Section III-B.2, the monotonicity property does not hold for the point subsets. Therefore, some heuristic must be used in order to avoid performing PCA on all possible subsets of points for each candidate feature subset. In this subsection, we discuss the heuristics that can be used for choosing the subset of points.

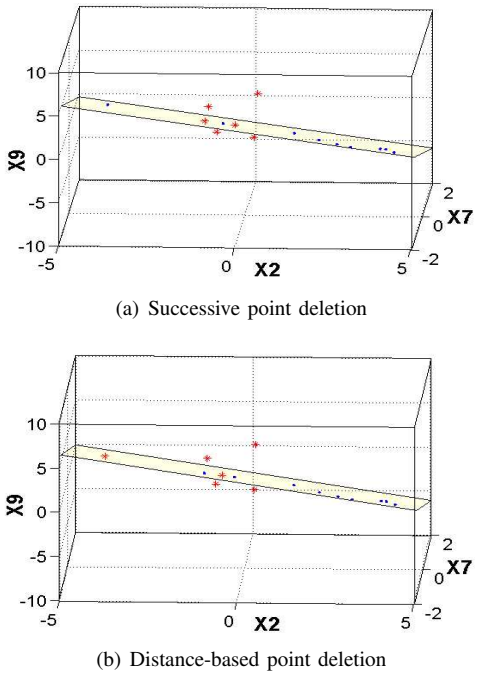
1) A successive point deletion heuristic: For a given candidate feature subset, if it is not strongly correlated on all data points, we can delete the points successively in the following way.

Suppose that $F = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\} \times \{\mathbf{p}_1, \cdots, \mathbf{p}_M\}$ is a submatrix of D and $f(F, k) > \epsilon$, i.e., the features of F is not strongly correlated on all data points. Let $F_{\setminus \mathbf{p}_a}$ be the submatrix of F by deleting point $\mathbf{p}_a$ ($\mathbf{p}_a \in \{\mathbf{p}_1, \cdots, \mathbf{p}_M\}$) from F. This heuristic deletes the point $\mathbf{p}_a$ from F such that $f(F_{\setminus \mathbf{p}_a}, k)$ has the smallest value comparing to deleting any other point. We keep deleting points until the number of points in the submatrix reaches the ratio $m/M = \delta$ or the feature subset of F turns out to be strongly correlated on the current point subset.

This is a successive greedy point deletion heuristic. In each iteration, it deletes the point that leads to the most reduction in the objective function value. This heuristic is time consuming, since in order to delete one point from a submatrix containing m points, we need to calculate the objective function value m times in order to find the smallest value.

2) A distance-based point deletion heuristic: In this subsection, we discuss the heuristic used by CARE. It avoids calculating objective function value m times for deleting a single point from a submatrix containing m points.

Suppose that $F = \{\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_n}\} \times \{\mathbf{p}_1, \cdots, \mathbf{p}_M\}$ is a submatrix of D and $f(F, k) > \epsilon$, i.e., the features of F is not strongly correlated on all data points. As discussed in Section III-A, let S_1 be the subspace spanned by the k eigenvectors with the smallest eigenvalues and the S_2 be the subspace spanned by the remaining $n - k$ eigenvectors. For each point $\mathbf{p}_a$ ($\mathbf{p}_a \in \{\mathbf{p}_1, \cdots, \mathbf{p}_M\}$), we calculate two distances: d_{a_1} and d_{a_2}. d_{a_1} is the distance between $\mathbf{p}_a$ and the origin in sub-eigenspace S_1 and d_{a_2} is the distance between $\mathbf{p}_a$ and the origin in sub-eigenspace S_2. Let the distance ratio $r_{\mathbf{p}_a} = d_{a_1}/d_{a_2}$. We sort the points according to their distance ratios and delete $(1-\delta)M$ points that have the largest distance ratios.



(a) Successive point deletion

(b) Distance-based point deletion

Fig. 6. Points deleted using different heuristics

The intuition behind this heuristic is that we try to reduce the variance in subspace S_1 as much as possible while retaining the variance in S_2.

Using the running dataset shown in Figure 1, for feature subset $\{\mathbf{x}_2, \mathbf{x}_7, \mathbf{x}_9\}$, the deleted points are shown as red stars in Figures 6(a) and 6(b) using the two different heuristics described above. The reestablished linear correlations are $2\mathbf{x}_2 + 5.9\mathbf{x}_7 + 3.8\mathbf{x}_9 = 0$ (successive), and $2\mathbf{x}_2 + 6.5\mathbf{x}_7 + 2.9\mathbf{x}_9 = 0$ (distance-based). Note that the embedded linear correlation is $2\mathbf{x}_2 + 6\mathbf{x}_7 + 3\mathbf{x}_9 = 0$. As we can see from the figures, both methods choose almost the same point subsets and correctly reestablish the embedded linear correlation.

The distance-based heuristic is more efficient than the successive approach since it does not have to evaluate the value of the objective function many times for each deleted point.

As a summary of Section IV, CARE adopts the depth-first search strategy to enumerate the candidate feature subsets. If a candidate feature subset is not strongly correlated on all data points, then CARE applies the distance-based point deletion heuristic to find the subset of points on which the candidate feature subset may have stronger correlation. If a candidate turns out to be a strongly correlated feature subset, then all its supersets can be pruned.

Point subsets	Local linear correlations
$\{\mathbf{p}_1, \cdots, \mathbf{p}_{60}\}$	$\mathbf{x}_{50} - \mathbf{x}_{20} + 0.5\mathbf{x}_{60} = 0$
$\{\mathbf{p}_{30}, \cdots, \mathbf{p}_{90}\}$	$\mathbf{x}_{40} - \mathbf{x}_{30} + 0.8\mathbf{x}_{80} - 0.5\mathbf{x}_{10} = 0$
$\{\mathbf{p}_{50}, \cdots, \mathbf{p}_{110}\}$	$\mathbf{x}_{15} - \mathbf{x}_{25} + 1.5\mathbf{x}_{45} - 0.3\mathbf{x}_{95} = 0$

TABLE V

LOCAL LINEAR CORRELATIONS EMBEDDED IN THE DATASET

Eigenvectors	Linear correlations reestablished
$\mathbf{v}_1$ ($\lambda_1 = 0.0077$)	$0.23\mathbf{x}_{22} - 0.25\mathbf{x}_{32} - 0.26\mathbf{x}_{59} \approx 0$
$\mathbf{v}_2$ ($\lambda_2 = 0.0116$)	$0.21\mathbf{x}_{34} - 0.26\mathbf{x}_{52} \approx 0$
$\mathbf{v}_3$ ($\lambda_3 = 0.0174$)	$-0.22\mathbf{x}_6 - 0.29\mathbf{x}_8 + 0.22\mathbf{x}_{39}$ $-0.23\mathbf{x}_{72} + 0.26\mathbf{x}_{93} \approx 0$

TABLE VI

LINEAR CORRELATIONS IDENTIFIED BY FULL DIMENSIONAL PCA

$\mathbf{x}_{50} - 0.99\mathbf{x}_{20} + 0.42\mathbf{x}_{60} = 0$
$\mathbf{x}_{40} - 0.97\mathbf{x}_{30} + 0.83\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$
$\mathbf{x}_{15} - 0.9\mathbf{x}_{25} + 1.49\mathbf{x}_{45} - 0.33\mathbf{x}_{95} = 0$

TABLE VII

LOCAL LINEAR CORRELATIONS IDENTIFIED BY CARE

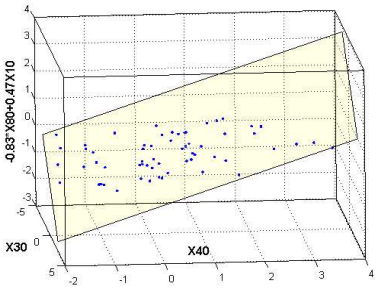
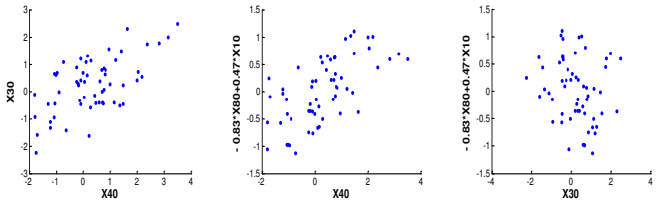


Fig. 7. The hyperplane representation of a local linear correlation reestablished by CARE



(a) (b) (c)

Fig. 8. Pair-wise correlations of a local linear correlation

V. EXPERIMENTS

To evaluate CARE, we apply it on both synthetic datasets and real life datasets. CARE is implemented using Matlab 7.0.4. The experiments are performed on a 2.4 GHz PC with 1G memory running WindowsXP system.

A. Synthetic Datasets

1) Effectiveness evaluation: To evaluate the effectiveness of the CARE, we generate a synthetic dataset of 100 features and 120 points in the following way. The dataset is first populated with randomly generated points for each one of the 100 features. Then we embedded three local linear correlations into the dataset as described in Table V. For example, on points $\{\mathbf{p}_1, \cdots, \mathbf{p}_{60}\}$ we create local linear correlation $\mathbf{x}_{50} - \mathbf{x}_{20} + 0.5\mathbf{x}_{60} = 0$. Gaussian noise with mean 0 and variance 0.01 is added into the dataset.

a) Comparison with full dimensional PCA: We first show the comparison of CARE and full dimensional PCA. We perform PCA on the synthetic dataset described above. To present the linear correlation discovered by PCA, we show the resulting hyperplanes determined by the three eigenvectors with the smallest eigenvalues. Each such hyperplane represents a linear correlation of all the features in the dataset. Due to the large number of features, we only show the features with coefficients with absolute values greater than 0.2. The linear correlations reestablished by full dimensional PCA are shown in Table VI. Clearly, these are not the local linear correlations embedded in the dataset.

Table VII shows the local linear correlations reestablished by CARE, with $k = 1$, $\epsilon = 0.006$, $\delta = 50\%$, and $max_s = 4$. As can be seen from the table, CARE correctly identifies the correlations embedded in the dataset.

b) Comparison with projected clustering methods: Figure 7 shows the hyperplane representation of the local linear correlation, $\mathbf{x}_{40} - 0.97\mathbf{x}_{30} + 0.83\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$, reestablished by CARE. Since this is a 3-dimensional hyperplane in 4-dimensional space, we visualize it as a 2-dimensional hyperplane in 3-dimensional space by creating a new feature $(-0.83\mathbf{x}_{80} + 0.47\mathbf{x}_{10})$. As we can see from the figure, the data points are not clustered on the hyperplane even though the feature subsets are strongly correlated. The existing projected clustering algorithms [10], [11], [12] try to find the points that are close to each other in the projected space. Therefore, they can not find the strongly correlated feature subset as shown in this figure. In Section V-B.3, we further compare CARE with projected clustering method on real dataset.

c) Pair-wise correlations of strongly correlated feature subsets : In Figures 8(a) to 8(c), we show the pair-wise correlations between the features of the local linear correlation $\mathbf{x}_{40} - 0.97\mathbf{x}_{30} + 0.83\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$. These figures demonstrate that although the feature subset is strongly correlated, the pair-wise correlations of the features may still be very weak. The clustering methods [1], [13] focusing on pair-wise correlations cannot find such local linear correlations.

d) Sensitivity with respect to parameters: We run CARE under different parameter settings. Table VIII shows the local linear correlations reestablished by CARE for the embedded correlation $\mathbf{x}_{40} - \mathbf{x}_{30} + 0.8\mathbf{x}_{80} - 0.5\mathbf{x}_{10} = 0$. As we can see from the table, CARE is not very sensitive to the parameters. Similar results have also been observed for other embedded correlations.

2) Efficiency evaluation: To evaluate the efficiency of CARE, we generate synthetic datasets as follows. Each synthetic dataset has up to 500K points and 60 features, in which

k	ϵ	δ	Linear correlations reestablished
1	0.006	50%	$\mathbf{x}_{40} - 0.97\mathbf{x}_{30} + 0.83\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$
1	0.006	40%	$\mathbf{x}_{40} - 0.98\mathbf{x}_{30} + 0.78\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$
1	0.006	30%	$\mathbf{x}_{40} - 0.98\mathbf{x}_{30} + 0.78\mathbf{x}_{80} - 0.48\mathbf{x}_{10} = 0$
1	0.009	50%	$\mathbf{x}_{40} - 0.96\mathbf{x}_{30} + 0.82\mathbf{x}_{80} - 0.53\mathbf{x}_{10} = 0$
1	0.012	50%	$\mathbf{x}_{40} - 1.06\mathbf{x}_{30} + 0.85\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$
1	0.03	55%	$\mathbf{x}_{40} - 0.79\mathbf{x}_{30} + 1.05\mathbf{x}_{80} - 0.33\mathbf{x}_{10} = 0$
2	0.006	50%	$\mathbf{x}_{40} - 0.97\mathbf{x}_{30} + 0.85\mathbf{x}_{80} - 0.47\mathbf{x}_{10} = 0$
3	0.02	50%	$\mathbf{x}_{40} - 0.95\mathbf{x}_{30} + 0.86\mathbf{x}_{80} - 0.55\mathbf{x}_{10} = 0$

TABLE VIII

LOCAL LINEAR CORRELATIONS REESTABLISHED UNDER DIFFERENT PARAMETER SETTINGS

40 linear correlations are embedded. Gaussian noise with mean 0 and variance 0.01 is added into the dataset. The default dataset for efficiency evaluation contains 5000 points and 60 features if not specified otherwise. The default values for the parameters are: $k = 1$, $\epsilon = 0.006$, $\delta = 50\%$, and $max_s = 4$.

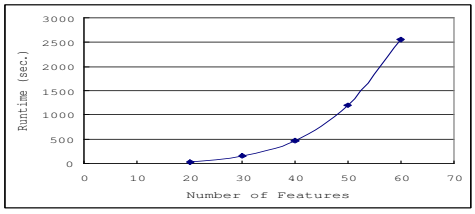
Figures 9(a) to 9(f) show the efficiency evaluation results. Figure 9(a) shows that the running time of CARE is roughly quadratic to the number of features in the dataset. Note that the theoretical worst case should be exponential when the algorithm has to check every subset of the features and data points. Figure 9(b) shows the scalability of CARE with respect to the number of points when the dataset contains 30 features. The running time of CARE is linear to the number of data points in the dataset as shown in the figure. This is due to the distance-based point deletion heuristic. As we can see from the figure, CARE finishes within reasonable amount of time for large datasets. However, since CARE scales roughly quadratically to the number of features, the actual runtime of CARE mostly depends on the number of features in the dataset.

Figure 9(c) shows that the runtime of CARE increases steadily until ϵ reaches certain threshold. This is because the higher the value of ϵ, the weaker the correlations identified. After certain point, too many weak correlations meet the criteria will be identified. Figure 9(d) demonstrates that CARE's runtime when varying δ. Figure 9(e) shows CARE's runtime with respect to different max_s when the datasets contain 20 features.

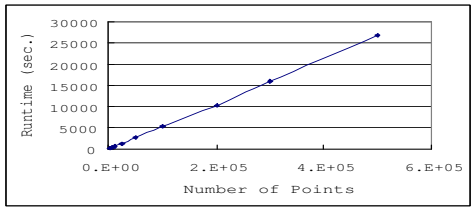
Figure 9(f) shows the number of patterns evaluated by CARE before and after applying the upper bound of the objective function value discussed in Section IV-A.

B. Real Life Datasets

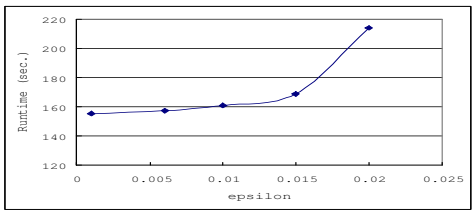
1) Gene expression data: We apply CARE on the mouse gene expression data provided by the School of Public Health at UNC. The dataset contains the expression values of 220 genes in 42 mouse strains. CARE find 8 strongly correlated gene subsets with parameter setting: $k = 1$, $\epsilon = 0.002$, $\delta = 50\%$, and $max_s = 4$. Due to the space limit, we show 4 of these 8 gene subsets in Table IX with their symbols and the corresponding GO annotations. As shown in the table, genes in each gene subset have consistent annotations. We also plot the hyperplanes of these strongly correlated gene subsets in



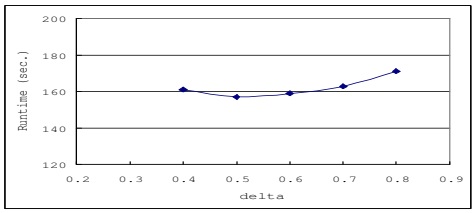
(a) Varying number of features



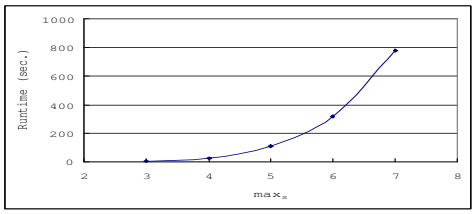
(b) Varying number of data points



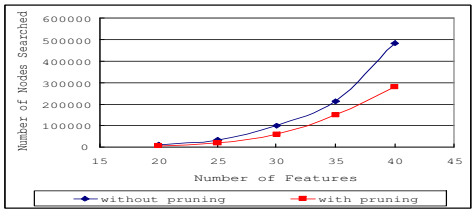
(c) Varying ϵ



(d) Varying δ

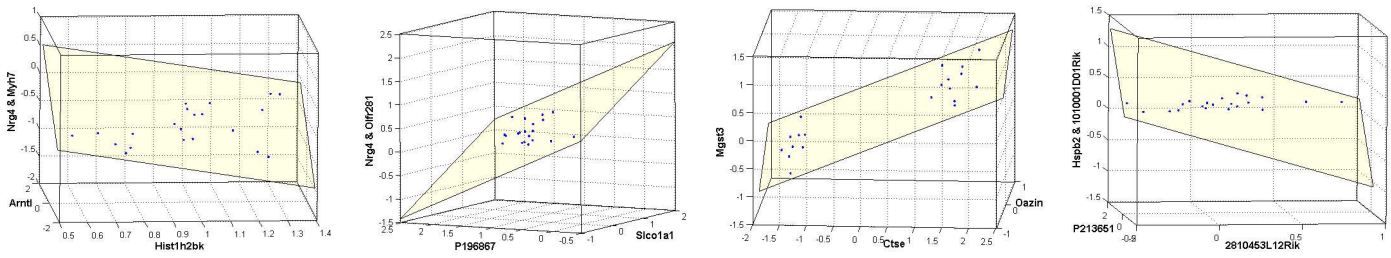


(e) Varying max_s



(f) Pruning effect of the upper bound of the objective function value

Fig. 9. Efficiency evaluation



(a) Gene subset 1 (b) Gene subset 2 (c) Gene subset 3 (d) Gene subset 4

Fig. 10. Hyperplane representations of strongly correlated gene subsets

Subsets	Gene IDs	GO annotations
1	Nrg4	cell part
	Myh7	cell part; intracelluar part
	Hist1h2bk	cell part; intracelluar part
	Arntl	cell part; intracelluar part
2	Nrg4	integral to membrane
	Olfr281	integral to membrane
	Slco1a1	integral to membrane
	P196867	N/A
3	Oazin	catalytic activity
	Ctse	catalytic activity
	Mgst3	catalytic activity
4	Hspb2	cellular physiological process
	2810453L12Rik	cellular physiological process
	1010001D01Rik	cellular physiological process
	P213651	N/A

TABLE IX
STRONGLY CORRELATED GENE SUBSETS

	Local linear correlations
1	TOT = 0.99OFF + DEF
2	GP = 0.21(2P%) + 0.86(FT%)
3	3PM = 0.99(3P%)
4	FGM = 0.17(2P%) + 0.89(3PA)
5	GS= 0.38OFF+0.74FTM

TABLE X
LOCAL LINEAR CORRELATIONS IDENTIFIED BY CARE IN THE NBA DATASET

3-dimensional space in Figures 10(a) to 10(d). As we can see from the figures, the data points are sparsely distributed in the hyperplanes, which again demonstrates CARE can find the groups of highly similar genes which cannot be identified by the existing projected clustering algorithms.

2) NBA dataset: We apply CARE on the NBA statistics dataset[5]. This dataset contains the statistics of 28 features for 200 players of season 2006-2007. Since the features have different value scales, we normalized each feature by its variance before applying CARE. The parameter setting is: $k = 2$, $\epsilon = 0.003$, $\delta = 50\%$ and $max_s = 4$. We report some interesting local linear correlations found by CARE in Table X.

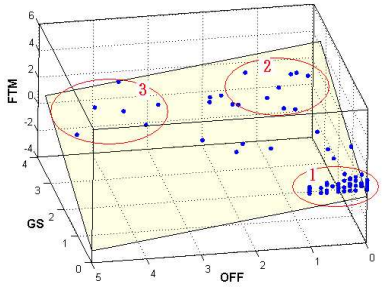


Fig. 11. A local linear correlation in NBA dataset

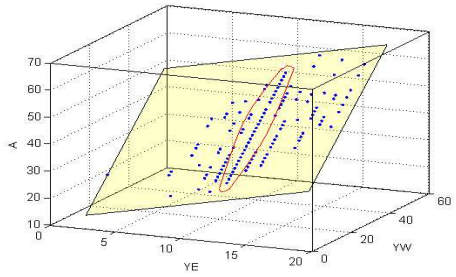
- Correlation 1 says that the total number of rebounds is equal to the sum of defensive and offensive rebounds. This is an obvious correlation that one would expect.
- The meaning of correlation 2 is that the number of games played is highly correlated with the 2-point shooting percentage and free throw percentage of the player.
- Correlation 3 says that players having high 3-point shooting percentage tend to get more 3-point field goals in the game.
- Correlation 4 tells us that the total number of field goals made by a player is correlated with his 2-point shooting percentage and the number of times he attempted to shoot 3-point.
- Correlation 5 shows the number of games started depends on how good the player is at offensive rebounds and free throws.

We plot Correlation 5 in Figure 11. This correlation holds on 3 different groups of players. The points in circle 1 show that the players not good at both offensive rebound and free throw get low game start. Circle 2 shows that players good at free throw get high game start and circle 3 show players good at offensive rebound get high game start. The points in circle 1 are close to each other but other points are far away from each other. Therefore this local linear correlation is invisible to the existing projected clustering algorithms.

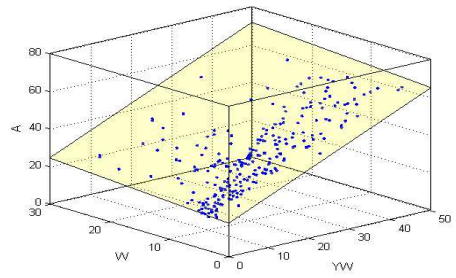
3) Wage dataset: We further compare CARE with the projected clustering method COPAC [12], which has been demonstrated to be more effective than ORCLUS [10] and 4C [11]. We apply CARE on the wage dataset[6], which also has

[5]http://sports.espn.go.com/nba/teams/stats?team=Bos&year=2007&season=2

[6]http://lib.stat.cmu.edu/datasets/CPS_85_Wages



(a) $YE+YW-A=-6$ (identified by both COPAC and CARE)



(b) $4.25YW+W-4.5A=-80$ (identified by CARE only)

Fig. 12. The hyperplane representations of local linear correlations in the wage dataset

been used in [12]. CARE successfully identifies both linear correlations reported in [12], i.e., $YE+YW-A=-6$ and $YW-1.03A=-17.4$. Further more, CARE identifies two new linear correlations, which are $4.25YW+W-4.5A=-80$ and $2.4YE+0.34YW-W=28.4$. These two linear correlations show the relationship among wage, working experience, age, and education, which are not discovered by COPAC. Figure 12(a) shows the hyperplane of linear correlation $YE+YW-A=-6$, which is found by both methods. In this figure, the points in the red circle form a density connected cluster. Therefore, the projected clustering method can find the correlation by first identifying this cluster. However, as shown in the figure, this correlation is also supported by other points outside the cluster. We also plot, in Figure 12(b), the hyperplane of correlation $4.25YW+W-4.5A=-80$, which is only found by CARE. Clearly, such correlation cannot be found by projected clustering methods because the points are sparsely distributed on the plane.

VI. Conclusion

In this paper, we investigate the problem of finding local linear correlations in high dimensional datasets. The local linear correlations may be invisible to the global feature transformation methods, such as PCA. We formalize this problem as finding the feature subsets that are strongly correlated on a large number of data points. We use spectrum theory to study the monotonicity properties of the problem. An efficient and effective algorithm, CARE, for finding such strongly correlated feature subsets is presented. The experimental results show that CARE can find these interesting local linear correlations that cannot be identified by the existing algorithms, such as full dimensional PCA, and projected clustering methods. The experimental results also demonstrate that CARE scales well to large datasets.

Our work reported in this paper focuses on the case where there is one linear correlation for a strongly correlated feature subset. For future work, one interesting direction is to extend current work to find multiple linear correlations in a feature subset. This is more challenging, since to find such correlations we have to decouple both features and points.

References

[1] M. Eisen, P. Spellman, P. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proc. Natl. Acad. Sci. USA*, vol. 95, pp. 14 863–68, 1998.

[2] V. Iyer and et. al., "The transcriptional program in the response of human fibroblasts to serum," *Science*, vol. 283, pp. 83–87, 1999.

[3] L. Parsons, E. Haque, and H. Liu, "Subspae clustering for high dimensional data: a review," *KDD Explorations*, 2004.

[4] A. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, vol. 97, pp. 245–271, 1997.

[5] H. Liu and H. Motoda, *Feature selection for knowledge discovery and data mining*. Boston: Kluwer Academic Publishers, 1998.

[6] L. Yu and H. Liu, "Feature selection for high-dimensional data: a fast correlation-based filter solution," *Proceedings of International Conference on Machine Learning*, 2003.

[7] Z. Zhao and H. Liu, "Searching for interacting features," *the 20th International Joint Conference on AI*, 2007.

[8] I. Jolliffe, *Principal component analysis*. New York: Springer, 1986.

[9] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning*. Springer Verlag, 1996.

[10] C. Aggarwal and P. Yu, "Finding generalized projected clusters in high dimensional spaces," in *SIGMOD*, 2000.

[11] C. Bohm, K. Kailing, P. Kroger, and A. Zimek, "Computing clusters of correlation connected objects," in *SIGMOD*, 2004.

[12] E. Achtert, C. Bohm, H.-P. Kriegel, P. Kroger, and A. Zimek, "Deriving quantitative models for correlation clusters," in *KDD*, 2006.

[13] H. Wang, W. Wang, J. Yang, and Y. Yu, "Clustering by pattern similarity in large data sets," *SIGMOD*, 2002.

[14] M. Ashburner and et al., "Gene ontology: tool for the unification of biology," *The gene ontology consortium, Nat. Genet.*, vol. 25, pp. 25–29, 2000.

[15] H. R. Lindman, *Analysis of variance in complex experimental designs*. Wiley-Interscience, 2001.

[16] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic Press, San Diego, California, 1990.

[17] W. Mendenhall and T. Sincich, *A Second Course in Statistics: Regression Analysis*. Prentice Hall, 2002.

[18] S. Yu, K. Yu, H.-P. Kriegel, and M. Wu, "Supervised probabilistic principal component analysis," *KDD*, 2006.

[19] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensinal data for data mining applications," *SIGMOD*, 1998.

[20] C. Aggarwal, J. Wolf, P. Yu, C. Procopiuc, and J. Park, "Fast algorithms for projected clustering," *SIGMOD*, 1999.

[21] C. Chen, A. Fu, and Y. Zhang, "Entropy-based subspace clustering for mining numerical data," *SIGKDD*, 1999.

[22] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge U. K.: Cambridge University Press, 1985.