CHAPTER 3

FEATURE SELECTION PROCEDURES

3.1 Feature Selection Methods

Variable and feature selection have become the focus of much research in areas of application for which datasets with tens or hundreds of thousands of variables are available. These areas include text processing of internet documents, gene expression array analysis, and combinatorial chemistry. The objective of variable selection is three-fold: improving the prediction performance of the predictors, providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data. Dimensionality Reduction is a standard technique used to reduce the dimensionality without information loss. Various dimensionality reduction techniques are offered to reduce the dimensionality. Among them, Feature Selection has become one of the most popular techniques. Feature selection is different from dimensionality reduction. Both methods seek to reduce the number of attributes in the dataset, but a dimensionality reduction method does so by creating new combinations of attributes, whereas feature selection methods include and exclude attributes present in the data without changing them. Examples of dimensionality reduction methods include principal component analysis, singular value decomposition and Sammon's mapping. Features selection, a pre-processing technique, comes to the rescue of obtaining the relevant features from the huge volume of feature space. It focuses mainly on selecting the relevant features using some predefined criterion and thereby increasing the efficiency of data mining algorithms [Shu-Chuan Lo, 2010]. The task of feature selection improves the classification accuracy and understands the ability of the learning process. In the process of learning, a large number of features requires huge memory space and consumes longer time for its

processing. At this juncture, Feature Selection reduces the cost of data acquisition and cost of computation. The reduction of features regarding selecting highly relevant features results in authentic conclusions. However, feature selection should not victimize the highly informative features. Classification is a key data mining technique whereby database tuples, acting as training samples, are analyzed to produce a model of the given data. Classification is an important data mining technique with broad applications. It is used to classify each item in a set of data into one of a predefined set of classes or groups. Each tuple is assumed to belong to a predefined class as determined by one of the attributes, called the classifying attribute. Once derived, the classification model can be used to categorize future data samples, as well as provide a better understanding of the database contents.

Filter-based feature selection provides a selection of widely used statistical tests for determining the subset of input columns that have the greatest predictive power.

(i) Pearson Correlation

Pearson’s correlation statistics or Pearson’s correlation coefficient is also known in statistical models as the R-value. For any two variables, it returns a value that indicates the strength of the correlation. Pearson's correlation coefficient is computed by taking the covariance of two variables and dividing by the product of their standard deviations. The coefficient is not affected by changes of scale in the two variables.

(ii) Mutual Information

The Mutual Information Score method measures the contribution of a variable towards reducing uncertainty about the value of another variable, in this case, the label. Many variations of the mutual information score have been devised to suit different distributions. The mutual information score is particularly

useful in feature selection because it maximizes the mutual information between the joint distribution and target variables in data sets with many dimensions.

(iii) Kendall Correlation

Kendall's rank correlation is one of the several statistics that measures the relationship between rankings of different ordinal variables or different rankings of the same variable. In other words, it measures the similarity of orderings when ranked by the quantities. Both this coefficient and Spearman's correlation coefficient are designed for use with non-parametric and non-normally distributed data.

(iv) Spearman Correlation

Spearman's coefficient is a nonparametric measure of statistical dependence between two variables and is sometimes denoted by the Greek letter rho. The Spearman's coefficient expresses the degree to which two variables are monotonically related. It is also called Spearman rank correlation because it can be used with ordinal variables.

(v) Chi-Squared

The two-way Chi-squared test is a statistical method that measures how close expected values are to actual results. The method assumes that variables are random and drawn from an adequate sample of independent variables. The resulting chi-squared statistic indicates how far results are from the expected (random) result.

(vi) Fisher Score

The Fisher score (also called the Fisher method, or Fisher combined probability score) is sometimes termed the information score because it represents the amount of information that one variable provides about some unknown

parameter on which it depends. The score is computed by measuring the variance between the expected value of the information and the observed value. When the variance is minimized, information is maximized. Since the expectation of the score is zero, the Fisher information is also the variance of the score.

(vii) Count-Based

Count-based feature selection is a simple yet relatively powerful way of finding information about predictors. It is a non-supervised method of feature selection, meaning that there is no need of a label column. This method counts the frequencies of all values and then assigns a score to the column based on frequency count. It can be used to find the weight of information in a particular feature and reduce the dimensionality of the data without losing information.

3.2 General Feature Selection Procedures

In the process of feature selection, irrelevant and redundant features or noise in the data may be a hindrance in many situations, because they are not relevant and important on the class concept such as microarray data analysis [Dash and Liu, 1997]. When the number of samples is much less than the features, then machine learning gets particularly difficult, because the search space will be sparsely populated. Therefore, the model will not be able to differentiate accurately between noise and relevant data. There are two major approaches to feature selection. The first is an individual evaluation, and the second is subset evaluation. Ranking of the features is known as individual evaluation [Guyon and Elissee, 2003]. In Individual Evaluation, the weight of an individual feature is assigned according to its degree of relevance. In Subset Evaluation, candidate feature subsets are constructed using search strategy.

The general procedure for feature selection has four key steps as shown in Figure 3.1.

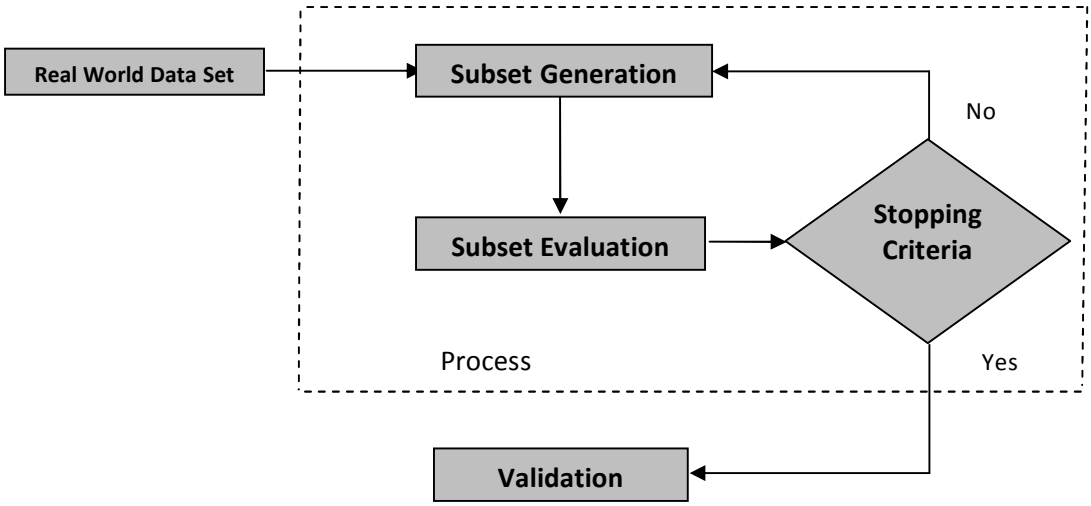

Figure 3.1 General Feature Selection Procedure

Figure 3.1 explains the flow of common methodology adopted in the feature selection. Feature space is reduced to a subset of features which are evaluated based on the criterion. Finally, these features are validated by the validating measures. The steps followed for general feature selection are elucidated in the following sub sections.

(i) Subset Generation

Subset Generation is a search process that generates the candidate feature subset using certain search strategy. The process has two basic issues, namely, search direction and search strategy. Firstly, a starting point must be selected which in turn influences the search direction. The search directions are divided into forwarding search, backward search, and bi-directional search. The search process starts with an empty set and adds the features progressively one by one (forward search) or starts with full sets and removes the features one by one (backward search) or starts with both ends and adds and removes the

features simultaneously (bi-directional search). Secondly, a search strategy must be decided. The search strategies are broadly categorized into three namely, complete search, sequential search and random search (*Ladha et al.*, 2011).

For *n* features in a dataset, there are 2^n possible subsets. In the case of this development, a thorough or complete (exhaustive) search of this space is practically not possible. The most realistic strategy involves some search strategy to reduce the search space. Different approaches applied to generate subsets in feature selection algorithms are briefly described here.

(a) Complete Search

In complete search approach, all 2^n subsets of features are taken into consideration. The best possible or optimal subset ought to be found. The clear disadvantage of this method is in the computational complexity of the search, O (2^n). Though the search is exhaustive, there is no guarantee of an optimal subset. This means that not all 2^n have to be evaluated to guarantee an optimal subset.

(b) Heuristic Search

The search through the space of subsets is guided by a heuristic algorithm in a way to avoid a complete search. However, since only a fraction of the search space is considered in the search, there are no guarantees that the optimal subset will be found. Several heuristics have been used for this purpose.

(c) Random Search

Algorithms that apply on random search approach generate a new subset randomly at each iteration. Even though the search space remains 2^n, the exact number of subsets that are considered by the algorithm is controlled by the number of iterations. As a result, the performance of the search process will depend on the resources available.

(ii) Subset Evaluation

In subset evaluation, an evaluation criterion is used to evaluate each newly generated subset. The evaluation criterion is used to determine the goodness of the subset (i.e., an optimal subset selected using one criterion may not be optimal according to another criterion). The evaluation criteria are divided into Independent, Dependent and Hybrid criteria (*Ladha et al*., 2011).

Selecting a final subset of features involves picking the best subset according to some evaluation measure. The evaluation method will set a value to each subset based on its ability to distinguish the different target classes. As a result, the subset with best evaluation measure should be able to generate highly accurate classification models. Different evaluation methods have been used for feature selection. These methods can be broadly grouped into four categories.

(a) Based on Distance

These measures are based on the assumption that instances of a different class are distant in the instance space. The Euclidian Distance is used by some algorithms to compute the distance between instances.

(b) Based on Information Gain

First introduced by Quinlan in his classification algorithm ID3 (Quinlan, 1986), information gain refers to the measure of how well a given feature separates instances according to their target classification. Such a statistical measure can be used to compare and consequently select features. Entropy, which measures the purity of a collection of instances, is often used to characterize information gain.

Based on Dependency

These methods are based on the rationale that good subsets contain features highly correlated with (predictive of) the class, yet uncorrelated with

(not predictive of) each other. Pearson's correlation coefficient is an example of a measure used to determine the degree of correlation between a subset and the target class, while the uncertainty coefficient and symmetrical uncertainty coefficient (Press, 1988) and the information gain ratio (Quinlan, 1986) can be used to determine the feature-feature and feature-class dependencies.

(c) Based on Consistency

To evaluate a given subset of features, its inconsistency rate is calculated by considering only the features of this subset. This rate refers to the number of instance pairs with same feature values but belonging to different classes. This evaluation scheme uses the Min-Features bias when selecting the best subset of features. Consequently, these measures find out the minimally sized subset that satisfies the acceptable inconsistency rate.

(d) Based on Classifier Accuracy

The classifier created from a given subset of features is used as an evaluation function.

(iii) Stopping Criterion

It is used to stop the feature selection process. The feature selection process may stop under one of the following criteria (*Ladha et al.,* 2011).

1. A predefined number of features is selected
2. Predefined number of iterations is reached
3. In case, addition (or deletion) of a feature fails to produce a better subset
4. An optimal subset according to the evaluation criterion is obtained

(iv) Validation

The validation process is used to measure the resultant subset using the prior knowledge about the data. In some applications, the relevant features are known beforehand; a comparison is made between the known set of features

with the selected features (*Ladha et al.,* 2011). However, in most real-world applications, the prior knowledge about the data is not available. In such case, the validation task is performed by an indirect method. For example, the classifier error rate test is used as an indirect method to validate the selected features. The error rate on the full set of features and the same on the selected set of features are compared to find the goodness of the feature subsets.

3.3 Search Strategies

Forward selection start with no variables and add them one by one, at each step adding the one that decreases the error the most, until any further addition does not significantly decrease the error. Backward Selection starts with all the variables and removes them one by one, at each step removing the one that decreases the error the most (or increases it only slightly), until any further removal increases the error significantly. To reduce overfitting, the error referred to above is the error on a validation set that is distinct from the training set. The bidirectional search starts from both sides - from an empty set and the whole set, simultaneously considering larger and smaller feature subsets. Heuristic selection generates a starting subset based on a heuristic (e.g. a genetic algorithm), and then explores it further. The most common search strategies that can be used with multivariate filters can be categorized into exponential algorithms, sequential algorithms and randomized algorithms. Exponential algorithms evaluate a number of subsets that grows exponentially with the feature space size. Sequential algorithms add or remove features sequentially (one or few), which may lead to local minima. Random algorithms incorporate randomness into their search procedure, which avoids local minima.

3.4 Types and Different Approaches of Feature Selection

A large number of algorithms have already been proposed for the feature selection issues. These algorithms are varied from its functionality based on the following:

1) The search strategy they use to determine the right subset of features
2) The evaluation of each subset.

There are two types of feature selection algorithms namely supervised and unsupervised. Supervised feature selection algorithms rely on measures that take into account the class information. A well-known measure is information gain, which is widely used in feature selection (*Dash et al.,* 1997). For feature selection in unsupervised learning, learning algorithms are designed to find a natural grouping of the examples in the feature space. Thus feature selection in unsupervised learning aims to find a good subset of features that forms the high quality of clusters for a given number of clusters (*Dy et al.,* 2004), (*Liu et al.,* 2008). Feature selection techniques can be divided into three main categories. They are,

1) Filter Approach (*Yu et al.,* 2003)
2) Wrapper Approach (*Kohavi et al.,* 1997)
3) Hybrid Approach (*Veerabhadrappa et al.,* 2010)

(i) Filter Approach

In the filter approach, the feature selection is performed as a pre-processing step in classifying the data. Here, the selection process is continued independently to improve the classification accuracy of the machine learning algorithm. In filter approach, to evaluate a feature or a subset of features, it applies an evaluation function that measures the discriminating ability of the feature or the subset to differentiate class labels. In practice, different evaluation functions are used by different algorithms. Filters are much less computationally expensive than the wrapper and hybrid algorithms. However, they may suffer from low performance if the evaluation criterion does not match the classifier well.

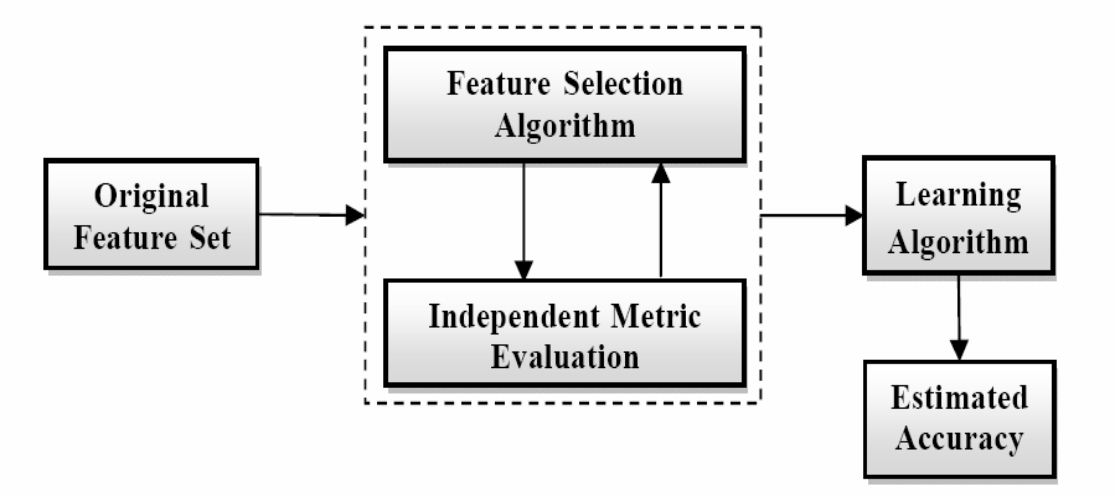

Figure 3.2 Filter Approach in Feature Selection

Figure 3.2 depicts the functionality of the filter approach in the feature selection. It encompasses the feature selection and its evaluation in one component. In the filter approach, the individual features are analyzed and evaluated. Based on this evaluation, the features are selected for the data mining task such as classification.

(ii) Wrapper Approach

In contrast to Filter approach, a Wrapper approach algorithm uses the learning algorithm as an integral part of the selection process. (John *et al.*, 1994) observed that the idea behind Wrappers come from the fact that the optimal subset of features depends on the specific bias of the learning system. Therefore, the selection of features should consider the characteristics of the classifier. Then, to evaluate subsets, wrappers use the classifier error rate induced by the learning algorithms as its evaluation function. This aspect of wrappers results in higher accuracy performance for subset selection than simple filters. However, since wrappers have to train a classifier for each subset evaluation, they are often much more time-consuming.

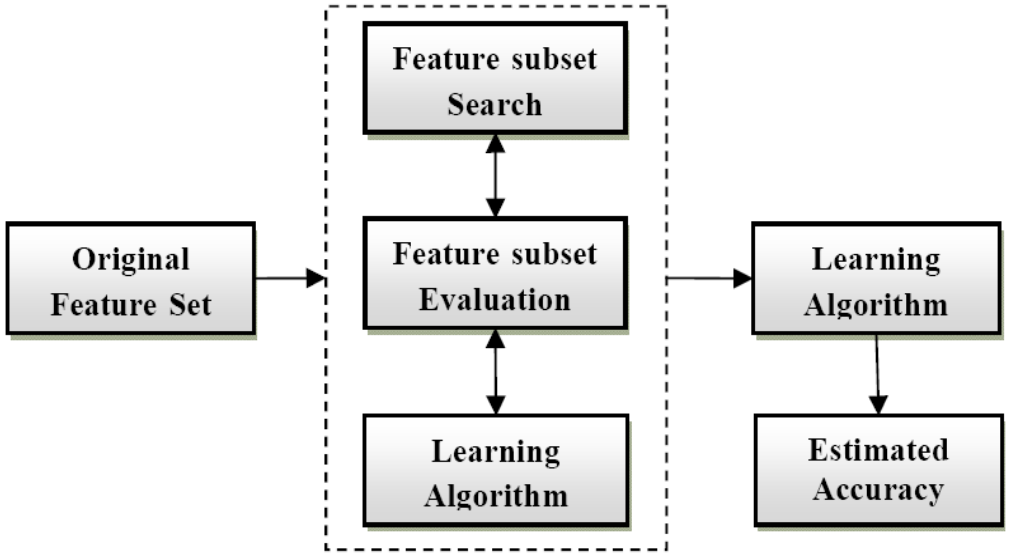

Figure 3.3 Wrapper Approach in Feature Selection

Figure 3.3 shows the functionality of wrapper feature selection process that encompasses three in one single component. Feature subset search, evaluation and learning algorithm are performed by a single unit.

(iii) Hybrid Approach

The term "hybrid" refers to the fact that two different evaluation methods are used, a filter-type of evaluation and hybrid-type evaluation. In this approach, the feature set is evaluated using both independent measure and a data mining algorithm. The independent measure is used to choose the best subset for a given cardinality, and the data mining algorithm selects the finest subset among the best subsets across diverse cardinalities (*Veerabhadrappa et al.*, 2010).

Figure 3.4 depicts the Hybrid approach of feature selection.

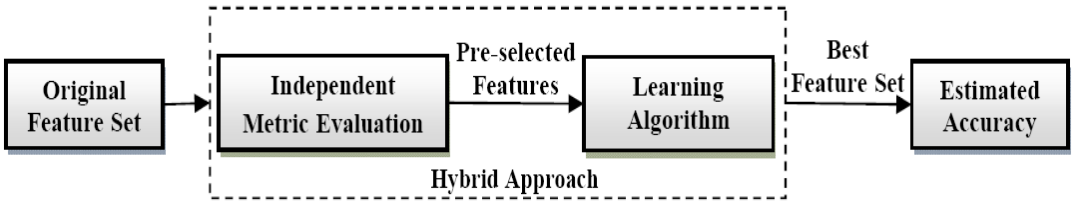

Figure 3.4 Hybrid Approach in Feature Selection

General Algorithm for Feature Selection

Input

S - data sample with features X, |X| = n

J - evaluation measure to be maximized

GS – successor generation operator

Output

Solution – (weighted) feature subset

L: = Start_Point(X);

Solution: = { best of L according to J };

repeat

L: = Search_Strategy (L, GS(J), X);

X':= {best of L according to J };

If J(X') $\geq$ J (Solution) or (J(X') = J(Solution) and |X'| < |Solution|)

Then Solution: =X';

Until Stop (J, L).

3.5 Dimensionality Reduction

Dimensionality reduction [Barak Chizi and Oded Maimon, 2005] is an important technique in various fields such as Data Mining, Machine Learning, Pattern Recognition, Image Retrieval and Text Mining and the like. Various real world applications in data mining usually have a high dimensional data. To handle the data adequately, its dimensionality needs to be reduced. The main objective of dimensionality reduction is to transform the high dimensional data samples into the low dimensional space such that the intrinsic information contained in the data is preserved. Once the dimensionality gets reduced, it helps to improve the robustness of the classifier, and it reduces the computational complexity.

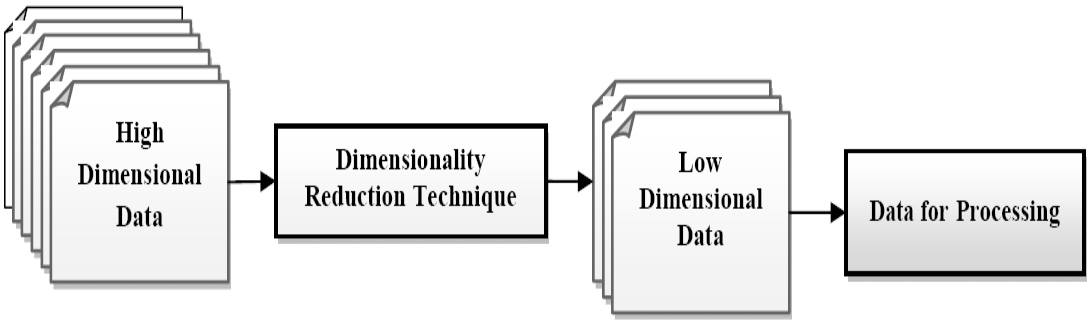

Figure 3.5 Dimensionality Reduction Process

Figure 3.5 shows the dimensionality reduction process. Generally, dimensionality reduction is performed using various techniques such as Principal Component Analysis, Principal Feature Analysis, Fisher Criterion, Factor Analysis and Classical Scaling.

3.5.1 Principal Component Analysis

Principal Component Analysis (PCA) (*Fengxi Song et al*., 2010) is a classical statistical technique which is widely used to reduce the dimensionality of a dataset consisting of the enormous amount of interrelated variables. Principal Component Analysis reduces the dimensionality by transforming the original dataset into a new set of variables, called principal components, where the largest variance present in the original dataset is captured by the highest component to extract the most important information. To show the workings of Principal Component Analysis, consider two-dimensional dataset (p, q) with fifty observations. Figure 3.6 shows the plot of fifty observations on the two variables *p*, *q* that are highly correlated and Figure 3.7 shows the transformed dataset using these principal components.

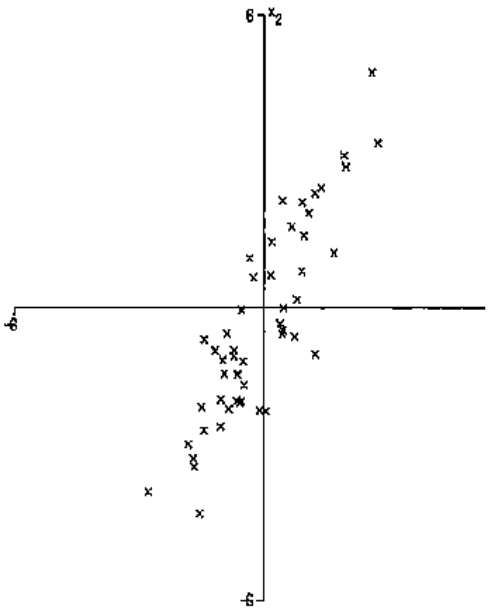

Figure 3.6 Plot of 50 Observations on Two Variables p ,q

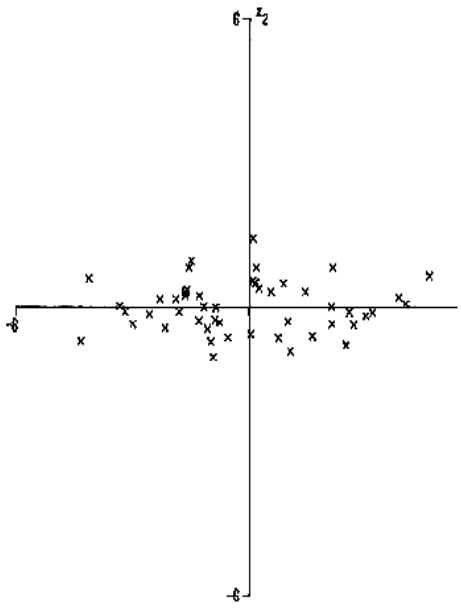

Figure 3.7 Transformed Data sets of Figure 3.6

The computational steps of the PCA algorithm are given below:

Step 1: Calculate the Mean:

$$\bar{y} = \frac{1}{M}\sum_{i=1}^{M} y_i$$

Step 2: Subtract the Mean from variables: F*i* = *yi* - *y*

Step 3: Form the Matrix A = [*Φ*1, *Φ*2 … *ΦM*] (N x M matrix), then compute:

$$\mathrm{C} = \frac{1}{M}\sum_{i=1}^{M} \Phi_p \Phi_p^T = AA^T$$

(Sample covariance matrix, *NxN*, characterizes the scatter of the data)

Step 4: Calculate the Eigen values of C: $\lambda_1 > \lambda_2 > ... > \lambda_N$

Step 5: Calculate the Eigenvectors *C*: $u_1, u_2 ... u_N$

Since C is symmetric, $u_1, u_2 \ldots u_N$ form a basis, (i.e., any vector *x* or *y* - $\bar{y}$, can be written as a linear combination of the Eigenvectors):

$$y - \bar{y} = b_1u_1 + b_2u_2 + ... bN\ uN = \sum_{i=1}^{N} b_i u_i$$

Step 6: (*dimensionality reduction step)* keep only the terms corresponding to the K largest Eigen values:

$$y - \bar{y} = \sum_{i=1}^{K} b_i u_i \text{ where } K \ll N$$

3.5.2 Principal Feature Analysis

The Principal Feature Analysis (PFA) (*Yijuan Lu et al.,* 2007) technique is derived from the Principal Component Analysis. It is an unsupervised technique.

Let *P* be a zero mean *m*-dimensional random feature vector and consider *X* be the covariance matrix. Let *D* be a matrix whose columns are the orthonormal Eigenvectors of the matrix *X*.

$$X = D\Lambda D^{\mathrm{T}}$$

$$\Lambda = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_m \end{bmatrix}$$

Where $\lambda_1, \lambda_2 \ldots \lambda_m$, be the Eigen values of *X*. Let *Dq* be the first *q* column of *D*. Let $V_1, V_2, V_3 \ldots V_n \in R_m$ be the rows of *Dm*. Each vector V_i represents the projection of the i^{th} feature of the vector *P*. To find the best subset, row vector V_i is used to cluster the features which have a high correlated measure. Finally, relevant features are obtained from each cluster to form a feature subset. The algorithm is summarized in the following five steps:

Step 1: Calculate the sample covariance matrix or true covariance matrix.

In few cases, the correlation matrix is preferred to use instead of the covariance matrix. The correlation matrixes is computed by

$$\mathrm{N_{xy}} = \frac{\left\lfloor j_x j_y \right\rfloor}{M\left[j_x^2 \right] M\left[j_y^2 \right]}$$

Step 2: Calculate the Principal components and Eigen values of the covariance / Correlation matrix by $X = D\Lambda D^T$.

Step 3: Matrix D_q is constructed from D by choosing the subspace dimension q. This can be chosen by deciding how much of the variability of the data is desired to be retained.

Step 4: Cluster the vector $| V_1 |, | V_2 |... | V_n | \in R^q$ into p clusters using the K-Means algorithm. Euclidean distance is used as a distance measure in K-Means algorithm.

Step 5: Find the corresponding vector V_i from each cluster which is closest to the mean of the cluster. Choose the corresponding features, S_i, as a principal feature. This step yields the choice of p features.

3.5.3 Fisher Criterion

Fisher Criterion plays an important role in dimensionality reduction technique. This criterion is used to select the features by minimizing within the class distance and maximizing the between-class distance. Based on the Fisher criterion, two methods are developed namely Fisher score and Linear Discriminant Analysis (LDA). Fisher score is a feature selection method, and Linear Discriminant Analysis (LDA) is a subspace learning method.

(i) Fisher score

Fisher score is one of the simplest criterion used for feature selection algorithms (*Jian-Bo Yang et al.*, 2009). In this criterion, the features having the similar values in the same class and the dissimilar values in different classes are selected. The Fisher score is calculated using the formula in equation (3.1)

$$\text{FS} = \frac{\sum_{k=1}^{m} S_k (\mu_{i,k} - \mu_i)^2}{\sum_{k=1}^{m} S_k \sigma_{i,k}^2} \quad \text{... (3.1)}$$

where,

μ_i is the mean of the features,

S_k is the number of samples in the k^{th} class,

$\mu_{i,k}$ is the mean of the features in the k^{th} class,

$\sigma_{i,k}^2$ is the variance of the features in the k^{th} class.

(ii) Linear Discriminant Analysis (LDA)

In many domains like Data Mining, Machine Learning, and Pattern Recognition, high dimensional data are commonly available. Acquiring valuable knowledge in high dimensional spaces is a challenging task. Under this condition, the data points are far apart from each other, and the similarities between data points are difficult to compare and analyze (Ye, 2009). Linear Discriminant Analysis is an important dimensionality reduction method to handle the high dimensional data. This technique mainly projects the high-dimensional data into lower dimensional space. Linear Discriminant Analysis aims to maximize the between-class distance and minimize the within-class distance in the dimensionality-reduced space. The Linear Discriminant Analysis is computed by the following equation (3.2)

$$f(X) = \text{trace}\left((X^T S_W X)^{-1} (X^T S_b X)\right) \quad \text{... (3.2)}$$

where,

S_b is the between – class matrix

S_w is the within – class matrix

$$S_b = \frac{1}{n}\sum_{i=1}^{n} K_i (c_i - c)(c_i - c)^T \quad \text{...(3.3)}$$

$$S_w = \frac{1}{n}\sum_{i=1}^{m}\sum_{x \in X_1} (x - c_i)(x - c_i)^T \quad \text{... (3.4)}$$

where,

X_i is the index set of i^{th} class

; c_i is the mean vector of i^{th} class.

K_i is the number of samples in the i^{th} class

3.6 Feature Transformation

Feature transformation is a process through which a new set of features is created. The variants of feature transformation are feature construction and feature extraction. Both are sometimes called feature discovery. Assuming the original set consists of A_1, A_2... Features, these variants are defined below. Feature construction is a process that discovers missing information about the relationships between features and augments the space of features by inferring or creating additional features (Matheus, 1991) (*Wnek et al.*, 1994) (Thornton, 1992). After feature construction, the set may have additional m features A_{n+1}, A_{n+2}, ..., A_{n+m}. For example, a new feature A_k ($n < k \leq n + m$) could be constructed by performing a logical operation on A_i and A_j from the original set.

Feature extraction is a process that extracts a set of new features from the original features through some functional mapping (Wyse *et al.*, 1980). After feature extraction, the set may have B_1, B_2, ..., B_m ($m < n$), $B_i = F_i(A_1, A_2, ..., A_n)$, and F_i functions. For instance, $B_1 = c_1A_1 + c_2A_2$ where c_1 and c_2 are coefficients. Subset selection is different from feature transformation wherein no new features will be generated instead only a subset of original features is selected and thus the feature space is reduced (*Dash et al.*, 1997) (Langley, 1994). As to feature transformation, feature construction often expands the feature space, whereas feature extraction usually reduces the feature space.

Feature transformation and subset selection are not two totally independent issues. For example, feature construction and subset selection can be viewed as two sides of the representation problem. Features can be considered as a representation language. In some cases where this language contains more features than necessary, subset selection helps to simplify the language; in other cases where this language is not sufficient to describe the problem, feature construction helps enrich the language. It is common that some constructed features are not useful at all. Subset selection can then remove these useless

features. It is also common to see the combined use of feature extraction and subset selection.

3.7 Application of Feature Selection in Real World

During data collection, many problems are often encountered such as a high dependency of features, too many features, or redundant and irrelevant features. To deal with the mentioned problem, feature selection provides a tool to select a feature subset or feature to learn algorithms effectively.

(i) Text Categorization

The massive volume of online text data on the internet such as e-mails, social sites, and libraries is increasing. Therefore, automatic text categorization and clustering are important tasks. A major problem with text classification or clustering is the high dimensionality of the document features. A moderate size text document may have hundreds of thousands of features. Therefore, feature selection (dimension reduction) is highly enviable for the efficient use of mining algorithms. Many applications of feature selection techniques are effectively used in the area of text mining. Information Gain Ratio is used for text classification (*Kumar et al.,* 2014). Many feature selection techniques are used for feature reduction, then evaluated and compared to the classification problem (*Kalousis et al.,* 2007) (*Luis et al.,* 2002) (*Das et al.,* 1997) (John *et al.,* 1994).

(ii) Intrusion Detection

In this modern age, information sharing, distribution, or communication is widely done by network-based computer systems. Therefore, the security of the system is an important issue protecting communication networks from intrusion by enemies and criminals. One of the ways to protect communication networks (computer systems) is intrusion detection. Feature selection plays an important role to classifying system activity as legitimate or an intrusion. Data

mining techniques and feature selection techniques are used for intrusion detection (*Lappas et al.,* 2007), (Bhavneet Kaur, 2017).

(iii) Genomic Analysis

A large quantity of genomic and proteomic data is produced by microarray and mass spectrometry technology for an understanding of the function of an organism, and the behavior, dynamics, and characteristics of diseases. Tens of thousands of genes are measured in a typical microarray assay and mass spectrometry proteomic profile. Special data analysis is demanded because of the high dimensionality of the microarray data. One of the common ways to handle high dimensionality is the identification of the most relevant features in the data. Filter, Wrapper, and Embedded methods have been used for feature selection and dimensionality reduction. Accuracy can be significantly boosted by a small number of genes by using a feature selection method.

(iv) Image Retrieval

Recently, the number of image collections from military and civilian equipment has increased. To access the images or make use of the information, images should be organized in a way that allows effective browsing, retrieving, and searching. Content-based image retrieval is scalable for the large size of images, but it is also cursed by high dimensionality (*Hastic et al.,* 2001). Therefore, feature selection is an important task for effective browsing, searching, and retrieval. Content-based image retrieval is proposed that annotate images by their colours, textures, and shape (*Rui et al*., 1999).

3.8 Advantages of Feature Selection

- It reduces the dimensionality of the feature space, to limit storage requirements and increase algorithm speed.
- It removes the redundant, irrelevant or noisy data.

- The immediate effects for data analysis tasks are speeding up the running time of the learning algorithms.
- It improves the data quality.
- It increases the accuracy of the resulting model.
- Feature set reduction is used to save resources in the next round of data collection or during utilization.
- It improves the performance to gain predictive accuracy.
- Data understanding gains knowledge about the process that generated the data or simply visualizes the data.

3.9 Performance Evaluation of Feature Selection Methods (POE)

- **The probability of error:** Features are ranked based on the expected probability of error, which is the fraction of patterns that are misclassified using a single feature. This technique belongs to pair-wise feature ranking using discriminative power.
- **The average correlation coefficient (ACC):** This is a feature subset configuration method. The first feature was chosen is the one with the smallest probability of error. The second feature was chosen is the one that has the smallest correlation coefficient with the first feature. The third feature is chosen such that its average correlation coefficient with the first two features is the smallest. Subsequent features are chosen based on the smallest Average correlation coefficient.
- **Sequential:** This is also a feature subset configuration method. Instead of using ACC, the feature to be added to a feature subset is the one that best discriminates the two most confused classes by the current feature subset.

- **Eigenvector analysis:** This is a not a feature selection but a feature transformation technique. Principal component analysis is used to create new features by linearly combining the original features. The new features are ranked based on their Eigen values.
- **Incomplete Eigenvectors:** Original features that make small contributions to the eigenvectors in the above mentioned principal component analyses are dropped from each eigenvector when computing new features. This is a feature transformation technique applied to a selected subset of features.
- **Property weighting by eigenvector component:** The average absolute weight of the original feature over the first 35 eigenvectors is used for feature ranking. This technique can be viewed as a simultaneous feature ranking method.
- **Weighted sum:** Features are ranked according to a weighted sum of their probability of error when used alone and their average correlation coefficient with the current feature subset. It is slightly more sophisticated than the second technique but is still a feature subset configuration method.