A Dimensionality-Reduction Method for Test Data

Fault Diagnosis

Mohamed Denguir, Sebastian M. Sattler
Friedrich-Alexander-University Erlangen-Nuremberg (FAU)
Chair of Reliable Circuits and Systems (LZS)
Erlangen, Germany
{mohamed.denguir, sebastian.sattler}@fau.de

***Abstract*—When performing a separation of test results, coping with enormous high-dimensional data sets is necessary but problematic. The input of high-dimensional data, in which not a few elements are irrelevant or less relevant than others, usually lead to inadequate results. It is therefore useful to consult methods, which classify the individual dimensions of the data volumes according to their relevance. In this paper, we present the Principal Component Analysis (PCA) and a Self-developed non-linear Data Analysis (SEDA), used on a complete data collection, as classification methods. Both analyzes are clarified using the same example.**

Keywords— Dimensional data reduction; Analysis of test results; Fault diagnosis.

I. Introduction

In many areas of research, the analysis of test results regarding circuits or systems is necessary. Huge data sets of information and signals of countless redundant sensors of a system are characterized by criteria such as their amount, their complexity and their speed. Globally, companies and research institutes strive to discover valuable information and correlation from the vast amounts of data that have so far been difficult or impossible to determine [1]. Very often, enormous, high-dimensional data sets from experiments must be collected and analyzed. However, the input of high-dimensional data usually results in insufficient results in this manner [2]. It is therefore useful to use classification methods, which classify the individual dimensions of the data volume according to their relevance. In this paper, we consult and analyze the Principal Component Analysis (PCA) and the self-developed method named SEDA.

II. Principal Component Analysis

A. Definition

The PCA is a variable-orientated, linear classification method for data reduction. The method uses linear structures enabling the reduction and interpretation of large multivariate data sets. This method allows the user to replace a number of original variables by a smaller number and it extracts relevant information from a given data set by reducing the dimension. By means of an orthogonal transformation, a new set of uncorrelated variables, the so-called Principal Components (PCs), is generated as a transformed database [3]. The newly determined PCs are linear combinations of the original variables. The first PC is so designed to be responsible for most of the variation in the original data and thus causing the reduction of the data size [4]. If the first PC describes the majority of the data variation, than this can also reduce the dimension of the problem. Through the transformation into PCs, the data sets can be graphically visualized and interpreted better.

B. Mathematical derivation

The PCA allows to obtain PCs or a transformed database (same abbreviation: PC) after entering the original database (D), with n-rows and m-columns (n x m-matrix) and performing five steps. First (step 1), a standardized database (S) is generated, which is column-wise mean-free, has column-wise value one as mean variation and occupies the same dimension as D. The target of this step (standardization) is to transform the various variables in the database so that they accept similar values and are directly comparable. Then (step 2), a correlated database (C) is generated, from which a correlation matrix (m x m-matrix) emerges, giving information about the relationships of variables. Further (step 3), the eigenvalues λ_j for j = 1 to m of the calculated correlation matrix are determined. The eigenvalues λ_j, characterizing general properties of linear images, are ordered accordingly to their size from large to small. Next (step 4), the eigenvectors V_j are determined with the help of the calculated and ordered eigenvalues λ_{sj} of correlation matrix C. Last (step 5), the subsequent multiplication of the standardized data S with the eigenvector matrix $V = (V_j)$ results in the transformed database PC. Thus, we have converted the original database D into PC, which has the same dimension of an n x m-matrix [5]. Here one speaks of an orthogonal transformation or a projection of the standardized database S onto the eigenvectors V_j, which are therefore called the coefficients of the PCs. Fig. 1 describes the PCA algorithm by means of a block diagram. In it, the derivation of the PCs is represented by the mathematical formulas required.

However, the corresponding vectors PC_j (n x 1-matrix) to the columns of PC are not all equivalent. They can be arranged depending on the size of the ordered eigenvalues λ_{sj} of the correlation matrix C. The information value of the variables decreases from PC_1 to PC_m. The following considerations are used to determine the variances of each PC. This should give us an idea of how the variances are related to the eigenvalues λ_{sj}. In general, the variance of PC_j can be represented by means of (1).

978-1-5386-1813-4/17/$31.00 ©2017 IEEE

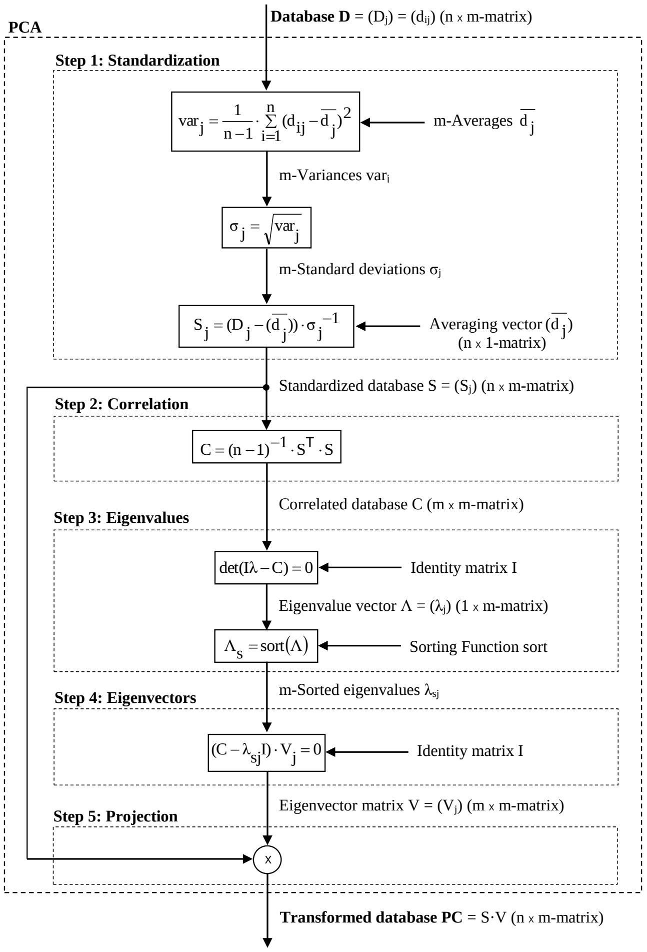
PCA
Database D = $(D_j) = (d_{ij})$ (n x m-matrix)
Step 1: Standardization
$\mathrm{var}_j = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (d_{ij} - \bar{d}_j)^2$
m-Averages $\bar{d}_j$
m-Variances var_j
$\sigma_j = \sqrt{\mathrm{var}_j}$
m-Standard deviations σ_j
$S_j = (D_j - (\bar{d}_j)) \cdot \sigma_j^{-1}$
Averaging vector $(\bar{d}_j)$ (n x 1-matrix)
Standardized database S = (S_j) (n x m-matrix)
Step 2: Correlation
$C = (n-1)^{-1} \cdot S^T \cdot S$
Correlated database C (m x m-matrix)
Step 3: Eigenvalues
$\det(I\lambda - C) = 0$
Identity matrix I
Eigenvalue vector $\Lambda = (\lambda_j)$ (1 x m-matrix)
$\Lambda_s = \mathrm{sort}(\Lambda)$
Sorting Function sort
m-Sorted eigenvalues λ_{sj}
Step 4: Eigenvectors
$(C - \lambda_{sj} I) \cdot V_j = 0$
Identity matrix I
Eigenvector matrix V = (V_j) (m x m-matrix)
Step 5: Projection
x
Transformed database PC = S·V (n x m-matrix)

Fig. 1. Block diagram of the PCA.

$$\mathrm{var}(PC_j) = \frac{1}{n-1}(PC_j - (\overline{PC_j}))^T (PC_j - (\overline{PC_j})) \qquad (1)$$

T stands for transposed. The standardized matrix S is mean-free, i.e. the averaging vector $\bar{S} = (0)$. It follows that $\overline{PC_j} = \bar{S} \cdot (\bar{V}_j)$ equals zero and thus the variance of PC_j can further be calculated as defined in (2).

$$\mathrm{var}(PC_j) = \frac{1}{n-1} \; (PC_j^T \cdot PC_j) = \frac{1}{n-1} (S \cdot V_j)^T (S \cdot V_j) \qquad (2)$$

According to general mathematical matrix rules follows $(S \cdot V_j)^T = V_j^T \cdot S^T$ and thus (3).

$$\mathrm{var}(PC_j) = \frac{1}{n-1} (V_j^T \cdot S^T \cdot S \cdot V_j) \qquad (3)$$

After conversion of the formula from step 2 (Correlation) from the block diagram (Fig. 1) one obtains (4).

$$S^T \cdot S = (n-1) \cdot C \qquad (4)$$

Substituting (4) into (3) following (5) results.

$$\mathrm{var}(PC_j) = \frac{1}{n-1} (V_j^T \cdot (n-1) \cdot C \cdot V_j) \qquad (5)$$

Since correlation matrices are in general symmetrically and square, the eigenvectors V_j of correlation matrix C are orthogonal [5], i.e. $V^T = V^{-1}$, which is why (6) results the following way.

$$\mathrm{var}(PC_j) = \frac{1}{n-1} (V_j^{-1} \cdot (n-1) \cdot C \cdot V_j) \qquad (6)$$

After reducing the constant (n-1) and using (6) finally (7) results for the variance of a j-th PC.

$$\mathrm{var}(PC_j) = (V_j^{-1} \cdot C \cdot V_j) = \lambda_{sj} \qquad (7)$$

So mathematically, it can be shown that the variance of a j-th PC equals the j-th eigenvalue of the correlated database. Essentially, the PCA corresponds to a rotation of the coordinate system in the direction of maximum variance [5]. The first PC shows the greatest variance, since within the analysis the eigenvalues were arranged according to their size. Equation (8) follows accordingly, which reproduces the proportion of shared variance of the data.

$$\frac{1}{m} \cdot \lambda_{sj} = \frac{1}{m} \cdot \mathrm{var}(PC_j) \qquad (8)$$

Thus PCs with great variance represent interesting dynamics while PCs with low variance represent low noise and therefore not much of information of the original database gets lost, when PCs with low variance are ignored [6]. The following application example eases the understanding of the theory and mathematics of the PCA discussed so far.

C. Example

In many areas of research, error detection and prediction of the causes of early failure is necessary. For that, PCA is a very useful analysis tool. Experiments very often result in enormous, high-dimensional data sets, which need to be collected and analyzed in a proper way.

Let us say a certain company produces and sells an electronic product, which consists of many digital and analog subsystems. Often their product "breaks" before the warranty period. The reasons for the early failure must be identified in order to achieve improvements in product production. Meanwhile, in many products an integrated chip stores important information about user- and product-behavior. Engineers can use these information as a database and filter the most important user variables, which are responsible for the early failure, by using the PCA. For this, the knowledge about user variables of functional, not early failed products is necessary to enable a separation of the variables. In this case, the user variables e.g. voltage, current, temperature, etc. are the eigenvectors and the products are the PCs.

We demonstrate the PCA analysis through the following study: We consider a database of 900 data sets or objects and 68 characteristics (user variables) that should represent 900 different devices of the same product of a company. They are sorted according to their lifespan, so that the first 450 represent early failed products and the last 450 represent late failed products. The PCA-algorithms (Fig. 1) is then executed. Since the method bases on matrices, we used a self-written program in MATLAB. As result, the first four PCs have the largest eigenvalues and cover over 75% of the variance (data not shown). Next, it is useful to display the object distribution in a plot with respect to the PCs. For the graphic representation, the coordinates of the 900 objects, sorted according to lifespan, are plotted with respect to PC_1 and PC_2 in a coordinate system with PC_1 as x-axis and PC_2 as y-axis. We use a clear 2D-graphic and obtain Fig. 2. This figure shows something interesting: From this simple representation, a first separation of the data between early and late failed can be observed. Although not all objects can be separated to a hundred percent and overlapping being avoided, a large part of the object distribution is specific.

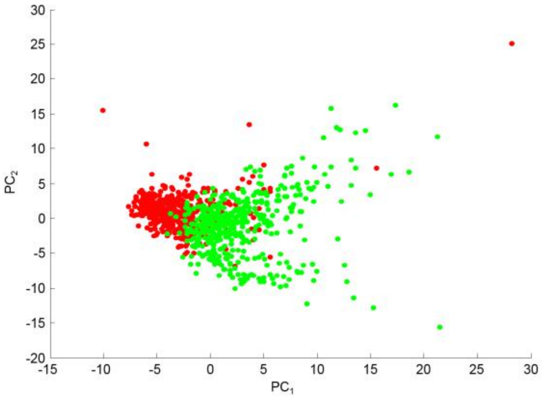

Fig. 2. Representation of the object data distribution for the first two PCs.

In order to display the characteristics graphically, their coordinates are plotted. These are listed in eigenvectors, which are defined as coefficients of the PCs. In addition, the individual points in the plot are linked to the origin in order to obtain vectors of characteristics and thus better represent their location in the coordinate system (see Fig. 3). It is obvious, that some characteristics cannot be seen clearly, since overlapping occurs, which arise by the same coordinates in the eigenvectors.

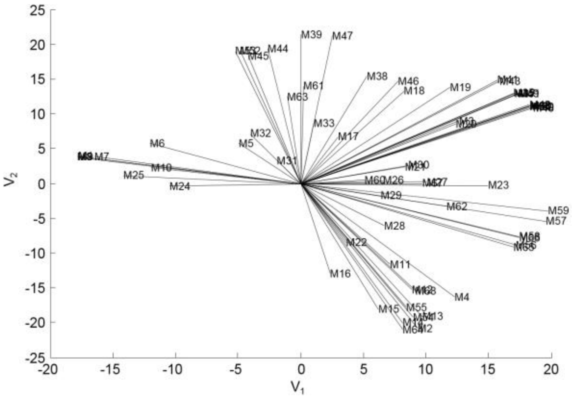

Fig. 3. Representation of the vectors of characteristics for the first two PCs.

We now want to investigate what the causes of early and late failure are. To answer this question, we overlay the representations of object data distribution and vectors of characteristics (Fig. 2 and Fig. 3) in Fig. 4. This allows seeing, which characteristics are in which areas of the objects and thus possibly influencing the behavior of the devices. It is clear, that the characteristics M6, M7, M8, M9, M10, M24 and M25 clearly correlate with red objects and e.g. characteristics M4, M55, M57, M59 and M65 along the green object cloud (see Fig. 4).

In order to confirm and investigate more precisely the above-observed correlation of the mentioned characteristics with the object clouds, the entire database is reduced to some of these characteristics. This is followed by a re-execution of the PCA at the reduced database. As result, the statements made in Fig. 4 are confirmed in Fig. 5. Here the overlap of some vectors of characteristics is also shown.

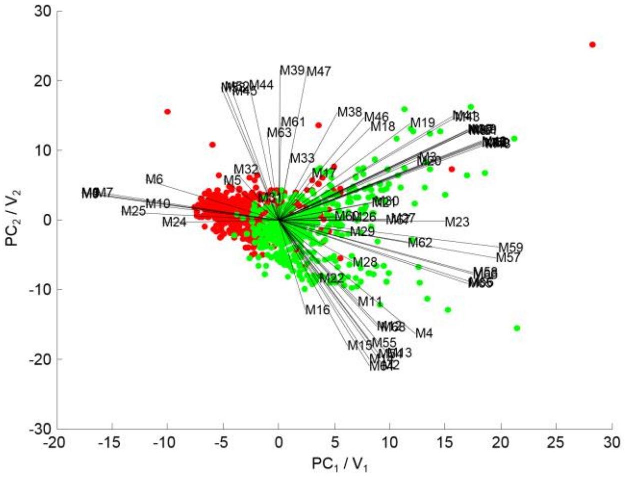

Fig. 4. Representation of the object data distribution and vectors of characteristics for the first two PCs.

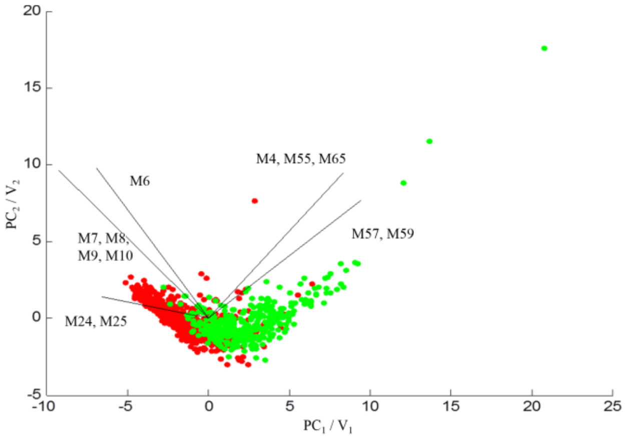

Fig. 5. Representation of the object data distribution and vectors of characteristics of reduced database.

III. SELF-DEVELOPED METHOD SEDA

A. *Definition*

The Self-developed Data Analysis (SEDA) is a self-defined, multi-dimensional analysis, which serves like PCA for data separation. SEDA executes its analysis in four steps, which are iteratively repeated in sequence, until complete data separation occurs. The following sections explain in detail the mathematical considerations of all four steps. In addition, a graphical representation (Fig. 7) is shown in analogy to PCA, and finally SEDA will be carried out on an example for a better understanding (see section IIIC).

B. *Representation and graphical vizualisation*

In the first step of SEDA, a given database with m-attributes and n-samples (objects) will be orthogonally transformed into a new set of uncorrelated variables. This can be done using the

PCA. As a result, one obtains m-new variables, the PCs (see Chapter II) to be seen as the replacement of the original database. According to an additional feature (criteria e.g. lifespan), the original database is a compilation of two object groups each with m-characteristics. For a better understanding, these two groups are distinguished through their color: The first n_1-objects are presented in red and the remaining n_2-objects in green with a total number $n = n_1 + n_2$. Due to this separation, the m-PCs are displayed colored too. The goal is now to determine the PC with the best separation of the data. For this purpose, step two and step three of SEDA are used.

In the second step of SEDA, the frequency distributions of the individual PCs are determined in the form of histograms. The principle of a histogram representation is the same principle of an analog-to-digital converter (ADC). For this purpose, the maximum and minimum value (pc_{jmax} and pc_{jmin} for j = 1 to m) is determined for each PC. In addition, a value for a number of bins (#Bins) is entered as input. This value defines an interval width ΔB (see (9)) in which the frequency of the individual values of the PC_j (pc_{ij} for i = 1 to n) are classified with respect to the intervals Bin_k (see (10)) and under consideration of the colors (see Fig. 6).

$$\Delta B = \frac{pc_{jmax} - pc_{jmin}}{\#Bins} \tag{9}$$

$$Bin_k \in [(k-1)\Delta B, k\Delta B] \text{ with } k = 1, 2, \ldots, \frac{\#Bins}{\Delta B} \tag{10}$$

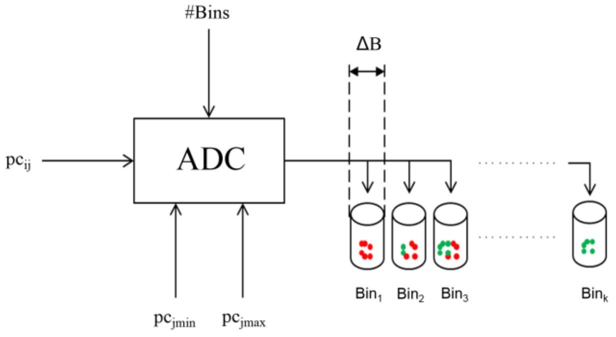

Fig. 6. Graphical representation of the frequency distribution of the objects.

The size of the #Bins should be below the dimensions n_1 und n_2 of the objects for the sake of clarity. The larger the value, the smaller the interval width ΔB and the more intervals Bin_k are necessary to divide the objects. This decreases the number of hits per Bin_k.

In the third step of SEDA, the number of separated objects is determined based on the frequency distributions. This is determined for the red and green objects individually for each PC. This means, the number of red separated objects (# red-separated) separated by the green objects is determined by summing the frequencies of the red objects under the condition of the exclusion of the green objects. The same applies to the determination of the separated green objects (# green-separated). Subsequently, the number of red and green separate objects can be summed and compared in tabular form for each PC. The PC with the largest value of separated objects has the best separation potential. After the determination of the PC with the best separation potential, the fourth step of SEDA now has to search out for this determined PC through all separated objects in the database. SEDA removes then these objects from the original database, resulting in a new database with $n_{new} = n_{1new} + n_{2new}$ lines or objects (database-new). The new number of red and green objects is now described as in (11) and (12).

$$n_{1new} = n_1 - \#\text{red-separated} \tag{11}$$
$$n_{2new} = n_2 - \#\text{green-separated} \tag{12}$$

The four steps of SEDA are iteratively repeated until a complete data partitioning of all objects is achieved, i.e. mathematically as long as n_{1new} and n_{2new} are greater than zero. The following Fig. 7 shows a schematic representation of SEDA.

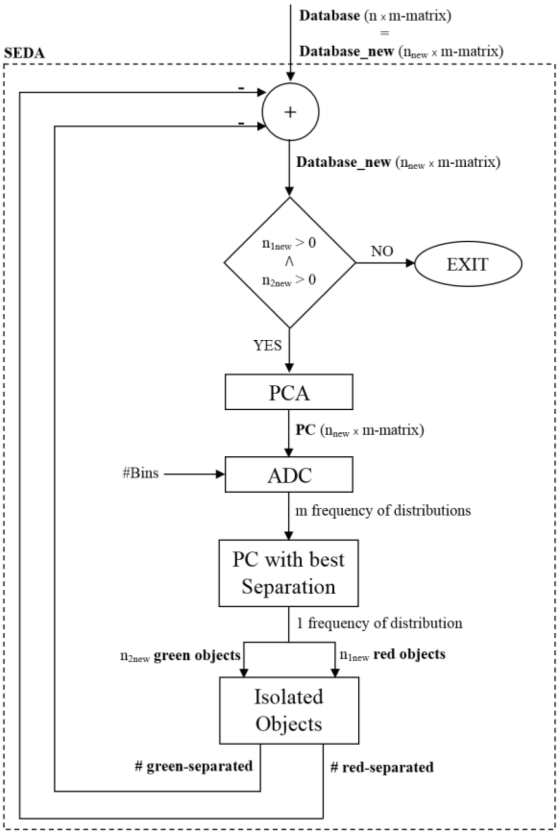

Fig. 7. Block diagram of SEDA.

C. *Example*

Analogously to the PCA, we present SEDA more precisely in this section by means of an example and through the analysis of the results. For this purpose, the same database as in the PCA (see section II.C) is used.

The database is analysed using the PCA. From this analysis 68 PCs were obtained. The frequency distributions of the individual PCs were determined. For test purposes, two different numbers of bins (#Bins = 200 and #Bins = 300) were investigated. This resulted in 136 (2 x 68) different plots of the frequency distributions for 68 PCs. The distribution of the red or green objects was displayed simultaneously for each plot. The frequency distributions for different #Bins are shown in Fig. 8 and Fig. 9. In these plots, the y-axis describes the frequencies, while the x-axis represents the bin number. A first visual view of the overlapping of the object points shows that the first PC (PC_1) has the best separation potential between early and late failures. PC_1 shows throughout the best separation of the object points in the different plots, so #Bins does not matter much in terms of separation potential.

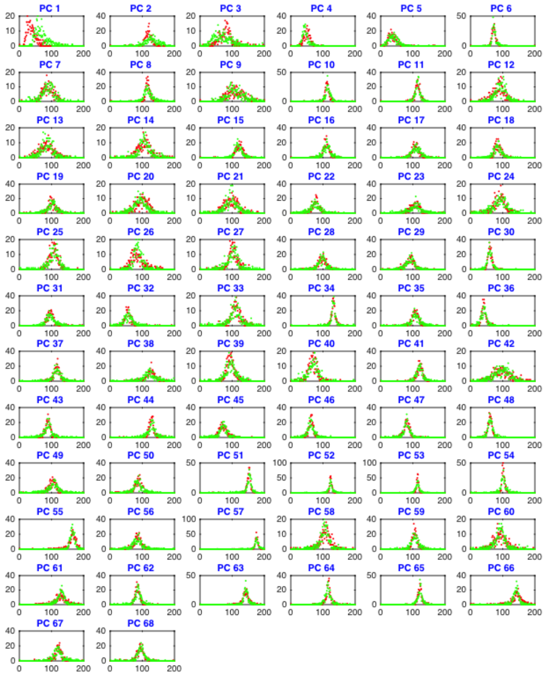

Fig. 8. Histograms of the frequency distributions for #Bins = 200.

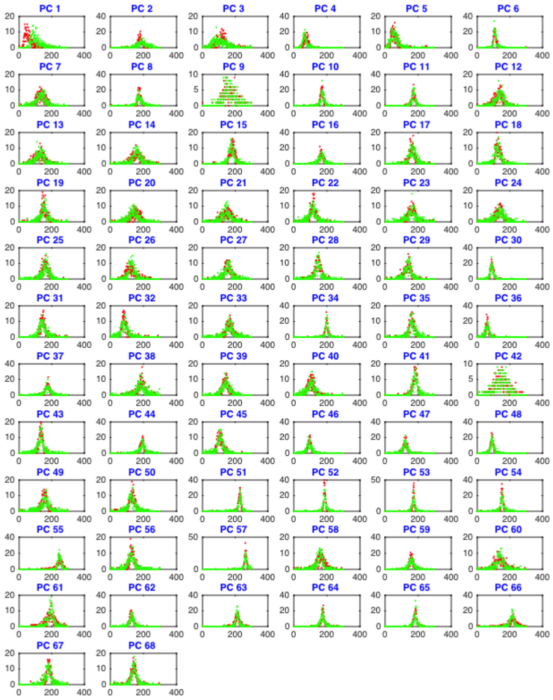

Fig. 9. Histograms of the frequency distributions for #Bins = 300.

By determining the number of separate red and green data sets (#red-separated and #green-separated) for each PC, as explained in section III.C, one can compute mathematically which PC actually provides the best separation result. Here e.g. #Bins was set to 300 and Fig. 10 resulted.

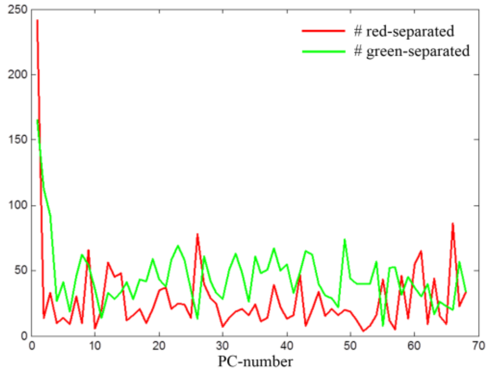

Fig. 10. Number of red- and green-separate objects for all PCs, #Bins = 300.

Here, too, it becomes clear that PC_1 has a distinctly higher separation number compared to the remaining PCs, which show an approximately similar separation potential from the fifth PC. Compared to the result of the PCA to the same database in Section II.C, the result recorded here is explainable, since the first four PCs in Table 1 had the largest eigenvalues and covered 75% of the variance. In view of the representation of the eigenvalues over the PC number in Fig. 10, the similarity with which the first four PC are responsible for a large portion of the data can be seen, whereas the remaining ones are rather standing for noise and thus do not cause a great loss of information in case of disregard.

After determining the PC, which is responsible for the best separation, the associated data of the separated objects are now removed from the database. The iterative process of SEDA takes place until all objects are completely separated. After each step, the frequency distributions are shown in Fig. 11 to Fig. 13. In this database used for SEDA, three iteration steps are sufficient to completely separate green and red objects as shown in Fig. 13. For each iterative procedure, PC1 was outputted for the best separation since the first PC is responsible for the largest portion of the variance in the PCA (see section IIII.B) and this variance is the information for the separation in this case.

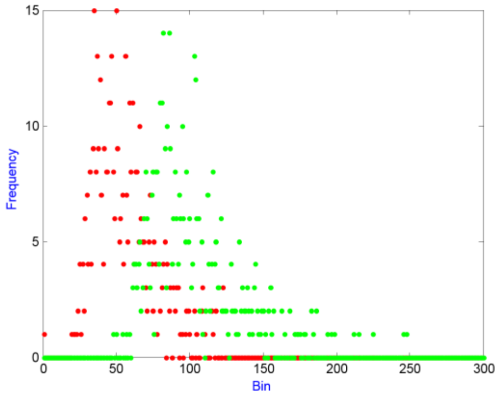

Fig. 11. First iteration: Frequency distribution for PC_1, #Bins = 300, 450 red objects and 450 green objects.

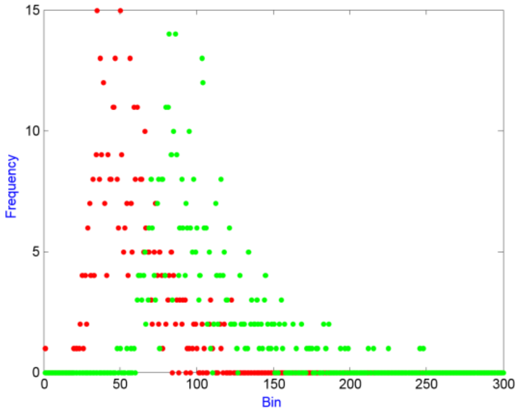

Fig. 12. Second iteration: Frequency distribution for PC$_1$, #Bins = 300, 266 red objects and 141 green objects.

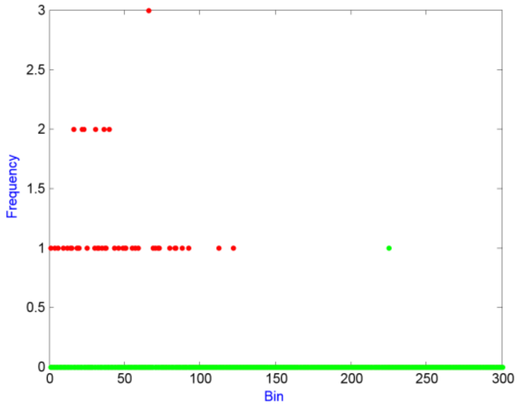

Fig. 13. Third iteration: Frequency distribution for PC$_1$, #Bins = 300, 52 red objects and one green object.

After a complete data collection by SEDA has been reached, the cause of early failure can now be explicitly examined analogously to Fig. 4. This is done by reducing the database to the separate data after each iteration and thus performing a reliable error diagnosis without any loss of information (complete database).

IV. Summary and Outlook

The purpose of this paper is to present two test methods for the separation and classification of high-dimensional test data and thus reduction of data dimensions. The PCA and SEDA are multi-dimensional analysis methods and have the goal of classifying individual dimensions of the data sets according to their relevance, or according to the variance of the set in this dimension.

The PCA is an orthogonal transformation in the m-dimensional space of the original variables into a new set of variables, the PCs. At the beginning, the correlation matrix is calculated from the data set of variables. Subsequently, the eigenvalues and eigenvectors of the correlation matrix are determined followed by the determination of the number of the eigenvalues, which cover the percentage of the variances to the greatest extent. This number indicates the effective dimension of the data and determines the necessary PCs to be considered. These PCs are then analyzed more precisely in order to further reduce the data volume and to interpret the test result. It is advantageous if the data points initially have a correlation, hence are linearly related to each other and therefore containing decisive information for a separation. However, there is no complete data separation and the number of PC selected is not explicit and depends on the choice of the user [7].

SEDA as a self-designed, multi-dimensional analysis method serving as the PCA as data separation. With SEDA, PCA's partially incomplete data separation should be optimized and yet the advantages of the PCA still be exploited. Using the SEDA, a database is transformed orthogonally into a set of uncorrelated variables using the first PCA step. According to the principle of an analog-to-digital converter, the frequency distributions of the individual PCs detected are determined in the form of histograms. Based on the frequency distributions, the PC with the best separation potential can be determined and used to reduce the database. For this purpose, the entire separated objects are traced, selected from the original database and removed therefrom. Iterative steps therefore allow complete data separation, depending on the resolution. The complete separation of the data and the independence from a data distribution make SEDA a robust and effective method compared to the PCA.

SEDA, in comparison to PCA, allows a non-linear, complete separation of non-linear separable objects according to specific criteria. However, it would be important to determine the nonlinear multi-dimensional polynomial in the next step. In addition, it would be useful to find out how the number of bins are related to the number of iterations, and whether other variables, such as database dimension or complexity, play a role in iterating or, more generally, complete data reduction. In addition, it would be interesting and important to compare SEDA with other procedures for feature selection and general machine learning in the test field [8]

References

[1] L. Zhao, Z. Chen, Y.Hu, G. Min, Z. Jiang, “Distributed Feature Selection for Efficient Economic Big Data Analysis”, IEEE Transactions on Big Data. (2017)

[2] X. Bian, H. Krim, A. Bronstein, L. Dai, “Sparse null space basis pursuit and analysis dictionary learning for high-dimensional data analysis”, IEEE (ICASSP), pp. 3781 - 3785. (2015)

[3] T. Zhang, B. Yang, “Big Data Dimension Reduction Using PCA”, IEEE International Conference on Smart Cloud, pp. 152 -157. (2016)

[4] Y. Xie, T. Zhang, “A fault diagnosis approach using SVM with data dimension reduction by PCA and LDA method”, Chinese Automation Congress (CAC), pp. 869 - 874. (2015)

[5] Z. Alf: “Hauptkomponentenanalyse - Principal Component Analysis”, version B, FIM- psychology, Erlangen (Germany). (1983)

[6] University Magdeburg (Germany), “Hauptkomponentenanalyse - Principal Component Analysis (PCA)”, p. 23. (2009)

[7] D. Brauckhoff, K. Salamatian, M. May, “Applying PCA for Traffic Anomaly Detection: Problems and Solutions”, IEEE INFOCOM, pp. 2866 – 2870. (2009)

[8] H. Ayari, F. Azais, S. Bernard, M. Comte, M. Renovell, V. Kerzerho, O. Potin, C. Kelma, “Smart selection of indirect parameters for DC-based alternate RF IC testing”, IEEE 30th VLSI Test Symposium (VTS). (2012)