possible across the samples. According to [9], the algorithm consists of the following steps:

1. Start with the entire expression matrix *X*, each row centered to have zero mean
2. Compute the leading principal component of the rows of *X*
3. Shave off the proportion (typically 10 %) of the genes having smallest absolute inner-product with the leading principal component
4. Repeat steps 2 and 3 until only one gene remains
5. This produces a nested sequence of gene clusters $S_N \supset S_k \supset S_{k_1} \supset S_{k_2} \supset \cdots \supset S_1$, where S_k denotes a cluster of *k* genes. Estimate the optimal cluster size $\hat{k}$ using the gap statistic [10].
6. Orthogonalize each row of *X* with respect to $\bar{x}_{S_k}$, the average gene in $S_{\hat{k}}$
7. Repeat steps 1-5 above with the orthogonalized data, to find the second optimal cluster. This process is continued until a maximum of *M* clusters are found, where *M* is chose a priori.

When implementing this method we made some changings compared with the steps form [9]. But I have made some changes in the algorithm. First of all, we shaved off not *α %* genes, but 1 gene each time. This is because we will lose the precision of the algorithm if using *α%*. For example, supposing we remain with S_k clusters with *k* being 135, 122, ..., 53, 47, etc. genes, and according to the gap statistic step, the algorithm decides to have a cluster with 52 or 47 genes, which is not correct. This is the reason why we have decided to have clusters with ..., 53, 52, 51, 50, 49, 48,... genes, in this way hoping to obtain a maximum *Gap(k)* closer to ideal 50. Also when computing the gap statistics we have made some changes. If we consider all possible permutations and after that finding each *D(k)*, then, in case of 150 genes with 4 characteristics of each gene, it is required to have all possible permutations of the matrix by permuting the elements within each row. This means that each row has from 4 parameters a number of 24 possible permutations, this implies that we have another sets of 24150 matrices, which is too much (we cannot take for example 3rd permutation from gene 1 with 3rd permutation for gene 2, 3rd permutation for gene 3, and so on, because we obtain the same *D(k)*). We have tried to consider random matrices, a number of 5000, but the problem in this case is that the result is varying at every other analysis and they are also different at each simulation. Finally we have decided to use just the first input matrix and get its *D(k)* and *Gap(k)*, without any permutations.

III. Data Clustering

A. Unsupervised clustering

Three unsupervised algorithms were used to cluster microarray data. We used for our study two different datasets from public Affymetrix databases. The first set was the Chowdary database [11] the authors compared pairs of snap-frozen and RNA later preservative-suspended tissue from 62 lymph node-negative breast tumors and 42 colon tumors, with purpose of separating them. The second set [12] contains 24 acute lymphoblastic leukemia (ALL), 28 acute myelogenous leukemia (AML) and 20 mixed-lineage leukemia (MLL) samples.

For the microarray data the clustering was done by a two-way clustering or bi-clustering [13] in which both the samples and the genes are clustered in the same time using the portioning method.

Regarding the conclusions from [3] a useful classification was obtained for microarray data when EM clustered the genes and k-means the samples. In this study we will use the k-means algorithm to cluster the samples and the PAM and respectively the EM algorithm to cluster the genes. Before combining the algorithms we applied the clustering validation methods, both external and internal indexes.

Also, based on these results we combined the unsupervised method with a supervised one. So we clustered the samples by k-means algorithm and we classify the genes using the gene shaving method.

In Table I, the numbers of clusters obtained after using the internal and external indexes are indicated. For the k-means algorithm the number refers to sample partitioning, while for the PAM and EM algorithms the numbers refers to gene clustering. In the EM clustering validation only the external indexes were computed.

TABLE I
THE NUMBER OF CLUSTERS OBTAINED WITH THE CLUSTERING VALIDATION METHODS

Index	Chowdary database			Leukemia database		
	k-means	PAM	EM	k-means	PAM	EM
Rand	2	3	3	3	3	4
Jaccard	2	2	2	3	3	3
Fowlkes-Mallows	2	2	2	3	3	3
Silhouette	2	2	-	2	3	-
Calinski-Harabasz	2	2	-	3	3	-
Krzanowski-Lai	2	2	-	3	3	-
Davies-Bouldin	2	2	-	2	3	-

After the validation was done we applied the combined clustering for the datasets. Thereby, for the Chowdary dataset the genes were clustered into 2 groups, with the PAM method, Fig 2.a, and then with the EM algorithm, Fig.2.b. The samples were clustered with the k-means algorithm.

The obtained values were compared with the given values from the microarray databases and a similarity between these values was observed.

Fig. 1.a and 1.b shows all the computed indexes for the Chowdary database, internal and external, in the case of PAM algorithm. For the external indexes the highest values obtained gave the optimal number of clusters. In the case of internal indexes the optimal value was marked by a square in Fig. 1.b.

According with the optimal number of clusters indicated by the validation indexes, in the case of the leukemia dataset the clustering was done in 3 clusters. Thus the samples were group into three sets by the k-means algorithm, while the genes formed also three groups once with the PAM method