university of groningen

university college groningen

Pattern recognition for the analysis of asthma

Using multiple sources of measurements

Master Thesis

Kyriaki Nektaria Pantelidou

Faculty of Science and Engineering
University of Groningen

31 January 2019

Supervisor:
Kerstin Bunte
Grigorios Tsoumakas

Abstract

Motivation: The large amount of genomic data makes necessary the use of machine learning and data analysis techniques in order to extract useful information and discover hidden patterns. In this project, we apply biclustering for biomarker detection on real asthma datasets.
Data: The datasets that we work with are 4. The first consists of clinical information for subjects that can be organized in 5 categories: general, blood, lung function, sputum, biopsy. The second and third contain gene expression and DNA methylation data, respectively. The last one includes micro RNA expression data.
Methods: We use a biclustering method, called 'FABIA: Factor Analysis for Bicluster Acquisition'. FABIA is a generative multiplicative model which is based on linear dependencies between gene expression and conditions to form biclusters. t-SNE algorithm is also used for the visualization of the data in 2 dimensions. We run FABIA algorithm multiple times and we rank the resulting biclusters. For the evaluation of the biclusters, 4 quality measures are being used. The first, information content, shows the amount of information each bicluster contains about the data. The next ones are the variance and mean squared residue (MSR). The fourth quality measure is the virtual error, which shows the tendency that genes follow under a set of conditions.
Results: After applying FABIA multiple times on the datasets and getting the biclusters, we choose to examine only the robust ones. We consider a bicluster as robust if it has average overlap percentage more than 80% over the runs. These biclusters gave us combinations of particular features from the first dataset that may indicate the existence of asthma.

Contents

1 Introduction

Asthma is a common long-term inflammatory disease of the airways of the lungs. Symptoms include episodes of wheezing, coughing, chest tightness, and shortness of breath. Asthma is thought to be caused by a combination of genetic and environmental factors and there is no cure for it. However, there are several kinds of medication that can deal with the symptoms.

Asthma can spontaneously remit or begin in adulthood, and the factors that lead to this are not widely understood. In some cases, patients may outgrow symptoms and reach clinical remission, but still use treatment, or others can be led to complete remission with absence of airway obstruction and bronchial hyperresponsiveness and do not have the need to use medication anymore [1]. The identification of asthma causes is rising the need for more personalized approaches for the prevention and treatment of it, creating also the possibility of total prevention and cure for the disease [2].

As stated before, asthma can be triggered by environmental or genetic conditions. Some environmental factors that may cause it are airborne substances, such as pollen, dust mites, mold spores or pet dander. Even cold air and air pollutants, such as smoke, can trigger symptoms of asthma. From genetic's side, there could be specific biomarkers that cause the development of the disease. The great number of gene expression data increases the challenge of how to identify biomarkers and critical genes associated with asthma and the challenge of being able to detect the disease's class [3].

New technologies, such as microarray, allow to measure the expression level of thousands of genes, under different circumstances, which results to a huge amount of data. This data is organized in a numerical matrix called expression matrix. With technologies like this, we can process DNA methylation data, as well as micro RNA

expession data. Both of them are organized in an expression matrix too. Apart from these data, we can work much easier with other medical measurements of patients and have them saved in matrices.

However, without appropriate methodologies and tools, significant information and knowledge hidden in these data may not be discovered. Therefore, there is a need for methods capable of handling and exploring large datasets. The field of data mining and machine learning provides a variety of methodologies and tools for analyzing these kind of datasets [4]. We use and review the results of an unsupervised learning technique and more specifically clustering. The main goal of clustering is to find groups of genes that present a similar variation of expression level under all the experimental conditions. Genes are grouped together according to their expression patterns across the whole set of conditions [5]. This is an important restriction in the use of clustering algorithms, since genes might be co-regulated and co-expressed only under certain experimental conditions, but behave almost independently under other conditions.

Nevertheless, relevant genes are not certainly related to every condition. That means that genes might be relevant only for a subset of experimental conditions [6]. For this reason, clustering should be performed not only on one dimension (genes) but on two dimensions (genes and conditions) at the same time. This technique is known as biclustering [7] and is becoming popular due to the ability of simultaneously grouping both genes and conditions. Biclustering application on gene expression data was first introduced by Cheng and Church [8]. In general, it is a much more complex problem than clustering, since it has been proven to be an NP-hard problem [9]. There are several biclustering methods, but we are going to use a generative model called FABIA [10].

t-SNE First of all, we will use the t-SNE algorithm on our data, for dimensionality reduction and visualization of the data points in 2 dimensions [11]. This way, we will have a first look on how the points are distributed and check if some clusters exist.

FABIA model FABIA [10] is a generative multiplicative model tailored to the special characteristics of gene expression data. After performing t-SNE on every dataset, FABIA is used to define the biclusters.

Structure The structure of the remainder of this document is the following:

In Chapter 2 (Related Work) the general research field is described together with details of how others tried to solve the problem of pattern recognition on biomedical data, with biclustering techniques.

In Chapter 3 (Approach) the description of the problem is stated together with the approach to solve it, as well as some basic preprocessing that has been done on the datasets.

In Chapter 4 (Methods) t-SNE method for the visualization of the data is described in detail, along with FABIA model that is used for the definition of the biclusters.

In Chapter 5 (Experiments) the above methods are applied on the datasets.

In Chapter 6 (Results) the results of the used techniques are evaluated and explained thoroughly.

In Chapter 7 (Conclusion and Future Work) the summarization of the thesis is done together with the listing of some points for future work and examination.

2
Related Work

Analysis of large amount of biomedical data, specifically gene expression, has focused on clustering methods in order to reveal unique patterns [12]. Biclustering algorithms seem to perform quite well in these kind of problems. We are going to present briefly some of them below.

Cheng and Church's algorithm Cheng and Church were the first to introduce biclustering for gene expression analysis [8]. The goal of their algorithm is to find biclusters with a small mean squared residue, which is a measure of bicluster homogeneity. Mean squared residue (MSR) is defined as:

$$MSR(B) = \frac{1}{|I|\,|J|}\sum_{i=1}^{|I|}\sum_{j=1}^{|J|}(b_{ij}-b_{iJ}-b_{Ij}+b_{IJ})^2 \tag{2.1}$$

where B is a bicluster that consists of a set I of $|I|$ genes and a set of J of $|J|$ conditions. Also, b_{ij} refers to the expression level of gene i uder sample j. Gene and sample means are represented as b_{iJ} and b_{Ij}, respectively. The mean of all values is referred to as b_{IJ} [13].The residue measures how an element differs from the row mean, column mean, and overall mean of the bicluster. If all the elements of the bicluster have small residue, then the MSR will be small.

Cheng and Church algorithm tries to find biclusters that are as large as possible, with the restriction that their MSR score must be less than some threshold δ. Like most biclustering problems, this one is NP-hard. Therefore, the method proceeds with a greedy approach: it starts with the largest possible bicluster, then removes rows and columns

that most reduce its MSR score. The deletion of nodes stops when MSR(B) $\leq \delta$. At this point, the algorithm tries to add nodes that do not make MSR(B) worse. After this step, the bicluster is added to the list of results and the algorithm starts again from the beginning.

Coupled two-way clustering This algorithm has been introduced by Getz, Levine and Domany [14] and uses a one-dimensional clustering algorithm to form biclusters. The goal of the clustering algorithm is to discover significant (stable) clusters. Using such an algorithm, coupled two-way clustering (CTWC) method will recursively apply it to submatrices of the initial gene expression matrix, aiming to find subsets of genes that form stable condition clusters and subsets of conditions that form stable gene clusters. The submatrices of such pairings are called stable submatrices and correspond to biclusters. The results of CTWC method is based on the one-dimensional clustering algorithm as well. Getz et al. are using the SPC hierarchical clustering algorithm [15] which performed quite well on gene expression data.

SAMBA algorithm The SAMBA algorithm [16] [17] uses probabilistic models and graph theory techniques to find subsets of genes that respond altogether across a subset of conditions. We say that a gene is responding in some condition if its expression level changes significantly at that condition. The expression data is represented as a bipartite graph, where one part corresponds to genes and the other to conditions. The edges that connect the nodes between the two parts represent significant expression changes. Every edge in the graph is assigned a weight according to a probabilistic model, so that heavy subgraphs correspond to biclusters with high likelihood. Now, in order to find the most significant biclusters, we just have to look for the heaviest subgraphs in the model bipartite graph.

Spectral biclustering Here we present spectral biclustering technique as it is described by Kluger et al. [18]. This method clusters genes and conditions by finding checkerboard patterns in matrices of gene expression data, if they exist. Spectral biclustering is based on the fact that checkerboard structures can be found in eigenvectors corresponding to characteristic expression patterns of genes or conditions. These eigenvectors can be identified by linear algebra approaches, in particular the singular value decomposition (SVD).

3 Approach

3.1 The problem statement

The problem that is to be solved has to do with hidden pattern and biomarker detection in gene expression, DNA methylation and microRNA expression data or in other patients' medical measurements, such us blood, lung function, sputum and biopsy. The samples (patients) can be assigned to a label of 2 or 4 classes. The 2-classes approach contains the labels asthma and healthy, while the 4-classes are asthma, complete remission, clinical remission and healthy. It is obvious that the 2-class approach problem is easier to handle, so we will deal with this one.

Using the FABIA model, we want to acquire biclusters of each of the above datasets. We take under consideration only the robust biclusters, meaning the ones that have quite high percentage of overlap over multiple runs. When we come to examine them, we keep count of asthma versus healthy subjects. If there are considerable more asthma over healthy in a bicluster, we regard it as an asthma bicluster. Same with healthy over asthma subjects. If the distinction is not clear, then we look at extra labels, such as smoking status or medicine use. Both smoking and medicines can affect clinical measurements and this could explain the reason why some patients are grouped together.

3.2 Datasets

The biomedical data has been provided by the UMCG (University Medical Center Groningen). It consists of four datasets:

1) Database_biopten_v2_6.csv
2) GE.txt
3) METH.txt
4) microrna4_default_aggr_normalized_log.txt

The first dataset contains clinical information of 232 subjects. This information can be organized in 5 categories: general, blood, lung function, sputum and biopsy. Some of the attributes in general category are sample id, gender, age, weight, height, asthma/smoking status and more. Blood category consists mainly of information about cholesterol, creatinin, triglycerids and lymfocytes, monocytes, eosinophilic and basophilic cell count. Lung function attributes are about lungs' capacity, while sputum contains macrophages, neutrophils, basophils, bronchial epithelium or squamous cells etc. Some of biopsy's attributes are percentage of mucus, mast cells and T cells. Several rows of this whole dataset have missing values on some features and the data is not normalized.

The second one contains gene expression data that has been normalized by TMM method of the edgeR R-package. There are 22918 expressed features in 184 subjects. Each subject can be linked to the rnaseq.id in the Database_biopten_v2_6.csv dataset. METH.txt file has DNA methylation data that has been normalized by the DASEN method from the wateRmelon package in R. There are 427427 CpG-sites interrogated in 178 subjects. Individual subjects can be linked to the rnaseq.id in the first dataset as well. The fourth and last dataset, microrna4_default_aggr_normalized_log.txt contains normalized miRNA expression data, and more specifically, 518 microRNAs from 205 subjects. In this one, the subjects can be matched in rnaseq.id of the first dataset, too. In Figure 3.1 the details of each dataset are gathered together.

Datasets	Number of subjects before preprocessing	Number of subjects after preprocessing	Asthma subjects after preprocessing	Healthy subjects after preprocessing
Blood	232	226	133	93
Lung function	232	154	70	84
Sputum	232	152	94	58
Biopsy	232	185	119	66
Gene expression	184	184	107	77
DNA methylation	179	178	109	69
miRNA expression	206	205	124	81

Figure 3.1: The table shows in detail the number of subjects before and after preprocessing, as well as the number of asthma and healthy subjects after preprocessing too, for every dataset.

3.3 Preprocessing

The second, third and fourth datasets do not need any preprocessing, since they do not contain any missing values and are already normalized. So, it is only the Database_biopten_v2_6 dataset that needs to be preprocessed. The preprocessing has been done in Python 2.7. First of all, we select and keep in different variables the columns of each category. So, there is one structure for blood data, one for lung function, another for sputum and one last for biopsy. In these structures, the columns 'asthma', 'currentsmoking' and 'ics.use' are included as well, since they are going to be the labels in the problem.

After this separation, the following preprocessing is made for every individual category. First, we decided to keep only the numeric columns. Then, each column of the category is checked for missing values. If the missing values of one column are more than 35% of its whole values, then this column is striked out. The next step is to check the rows. If a row has at least one missing value, then it is deleted as well. Then, the numeric columns are normalized with z-score transformation (also known as standardization), so every column has zero mean and standard deviation equal to 1. After this preprocessing, the indices of the rows and columns of the matrix are written in output files, in order to reconstruct the preprocessed datasets at any time needed.

3.4 Visualization

Since the datasets are preprocessed, the t-SNE method is fitted in every category and the results are plotted for each label (asthma, currentsmoking). That way, there is a first visualization of the data points in 2 dimensions and some clusters are already visible. That way, we can decide which datasets have more distinct clusters and then work with them first.

3.5 Biclustering method

Now, the FABIA method is applied on each dataset. The best number to set for the biclusters' amount is unknown, so we run the algorithm with different numbers. To evaluate the biclusters of every run, we use some quality measures, which are:
1) Information content, which shows the amount of information that every bicluster holds.
2) Variance, where the goal is to minimize the variance of the biclusters.
3) MSR (Mean Squared Residue), the lower this measure is, the stronger the coherence exhibited by the bicluster.
4) Virtual Error (VE), where biclusters with lower VE are preferable than those with higher.

After the evaluation, we are able to decide the number of biclusters to be used. We run FABIA 10 times for every dataset and check which biclusters appear robustly, based on a threshold. For these ones, we examine the labels of the samples that are clustered together and decide which are the most interesting.

4 Methods

4.1 t-SNE

t-Stochastic Neighbor Embedding (t-SNE) is a technique for dimensionality reduction that is particularly well suited for the visualization of high-dimensional datasets [11]. It can find a 2-dimensional embedding of clusterable data that correctly separates individual clusters to make them visually identifiable. In the high-dimensional space, t-SNE creates a probability distribution that dictates the relationships between various neighboring points. It then tries to recreate a low dimensional space that follows that probability distribution as best as possible. We will use this algorithm to get a first and general view of our data. It will help with the visualization and also give us a very good sense of hidden patterns. As a result, we can pay more attention and work more thoroughly with the categories and areas which t-SNE indicates that some patterns and defined clusters exist.

Our approach is to deal with the categories blood, sputum, lung function, biopsy, gene expression, methylation and micro RNA individually. For the labels, we have three different options. The first one is to use only two, healthy and asthma. The second and more advanced is to use four: current asthma, clinical remission, complete remission, healthy. The third option is a bit different, as the labels that we use are smoker and non smoker.

Another important factor, that needs to be mentioned, is that t-SNE cannot handle NaN values. Since one of the datasets that we have to work with has a lot of missing values, we need to find a solution to this issue first. This is why the preprocessing which

is mentioned in Chapter 3 is being done. As a result, the categories consist of fewer rows and attributes, but without NaN values.

For the implementation of t-SNE we use Python 2.7 and the package sklearn.manifold in which TSNE method is included. For the TSNE method we use the value 15 for the perplexity and pca for the initialization. The rest of the arguments keep their default values.

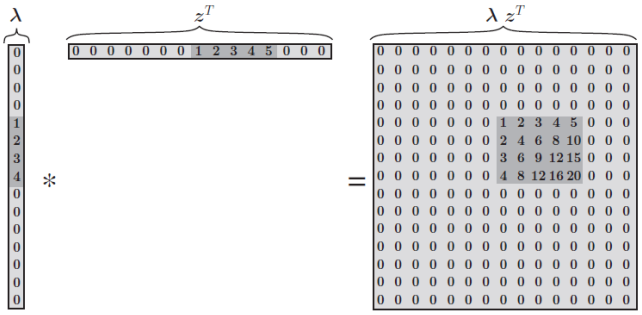
4.2 FABIA

FABIA (Factor Analysis for Bicluster Acquisition) is a multiplicative model class for analyzing gene expression datasets. A multiplicative model allows to model heavy tailed data, as observed in gene expression. Furthermore, it can relate the strength of gene expression patterns to characteristics of the conditions. The FABIA model is used to define biclusters on the data. A bicluster is defined as a pair of a row (gene) set and a column (sample) set for which the rows are similar to each other on the columns and vice versa. In FABIA model, two vectors are similar if one is a multiple of the other (the angle between them is zero). A dependency like this can be represented as an outer product λz^T of two vectors λ and z. The vector λ corresponds to a column vector that contains zeros for genes not participating in the bicluster and z is a vector of factors, that also contains zeros for samples not participating in the bicluster. Vectors containing many zeros or values close to zero are called sparse vectors [10]. The math type that describes the model, with p biclusters and noise is:

$$X = \sum_{i=1}^{p} \lambda_i z_i^T + Y \ = \Lambda Z + Y \tag{4.1}$$

where Y is additive noise and λ_i and z_i are sparse column and row vectors of the i-th bicluster, respectively. Figure 4.1 visualizes this representation by sparse vectors.

Figure 4.1: Two sparse vectors that their outer product results in a matrix with a bicluster [10]



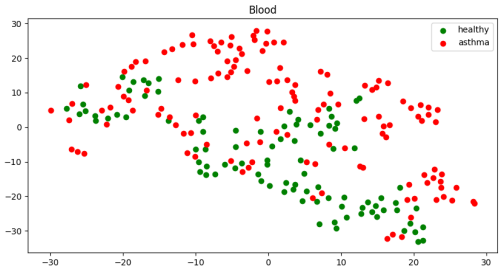
5 Experiments

5.1 Application of t-SNE

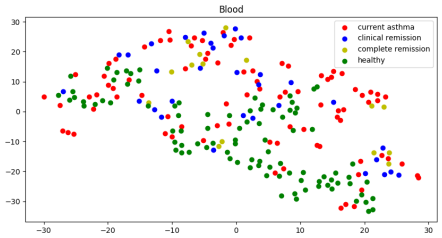
First, we work with the dataset Database_biopten_v2_6. For the blood measurements, we choose the attributes that correspond to this category. With the process described above, we keep only the non missing values and then perform normalization. After that, we fit the TSNE model and plot the results three times, with different labels. The results can be seen in Fig.5.1. Asthma subjects are represented by red points, clinical remission with blue, complete remission with yellow and healthy with green. In the smoking status figures, smokers are represented by red dots and non smokers with green.

In figure 5.1 (a) we can see some areas on the highest right part, where patients with asthma are gathered. The rest data points are mixed. The area with asthma patients, though, can be used later for further analysis. For figures 5.1 (b) and (c) there is no clear distinction between groups of patients that belong to different categories. So, we cannot extract useful information out of them.

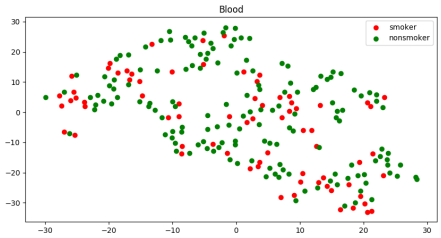
The next category that we examine is the sputum measurements. With the same procedure, we choose the numeric attributes, keep only the non missing values and normalize the data. After fitting TSNE and plotting the output, we get the results shown in Fig.5.2. In figure 5.2 (a) with two labels, healthy and asthma, we observe that the data points are really close together. Specifically asthma subjects form a group in the middle of all the points, with the healthy people being in small groups around them. So, we keep in mind that this case needs further examination. In the next figure, 5.2 (b), the data points look messy and with no structure, same as the last figure, 5.2 (c), where we



(a) healthy, asthma



(b) asthma, clinical remission, complete remission, healthy



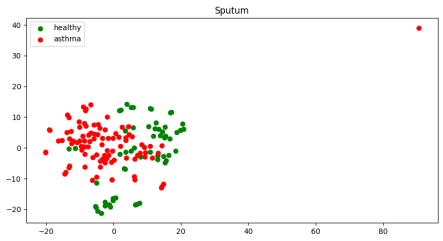
(c) smoker, non smoker

Figure 5.1: Blood measurements, categorized by healthy or asthma in (a), grouped by asthma, clinical remission, complete remission or asthma in (b) and grouped by smoker or non smoker in (c).

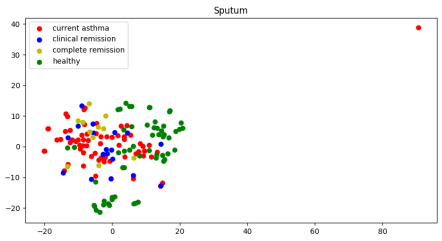
cannot extract much information just by looking at it.

Up next, we deal with the lung function measurements, picking only the numeric attributes, eliminating the missing values and implementing normalization on the remaining data. We fit TSNE model once again and plot the results. In the first figure, 5.3 (a) we can see clearly one cluster on the upper right side which consists of patients with asthma. In the next figure, 5.3 (b), the same cluster contains mainly data points of patients with current asthma, one with clinical remission and one with complete remission. We cannot come to a certain conclusion based on the figure with four labels. That is why we examine the plot with the labels smoker and non smoker, as well. In that one (figure 5.3 (c)), we observe that this group consists of non smokers. This outcome is a bit confusing, since there is no obvious reason for these patients to cluster together. However it could be interesting to analyze it further and it may give us some unexpected results.

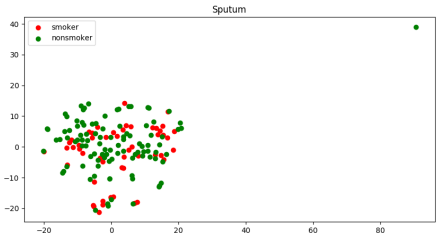
The last category of Database_biopten_v2_6 dataset is the biopsy measurements. In the same way, we process the data of this category, we fit the TSNE model and plot the results that are shown below. In the first figure 5.4(a) we can see a lot of asthma data points gathered together in the right part of the graph. It will be quite useful to examine biopsy category with these two labels, healthy and asthma. In the rest of the figures, we



(a) healthy, asthma



(b) asthma, clinical remission, complete remission, healthy



(c) smoker, non smoker

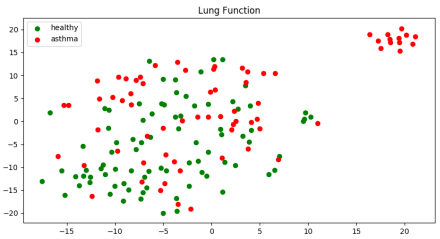
Figure 5.2: Sputum measurements categorized by healthy/asthma in (a), by asthma/clinical remission/complete remission/healthy in (b) and by smoker/non smoker in (c).

cannot distinguish any clear group that corresponds to the labels that we have.

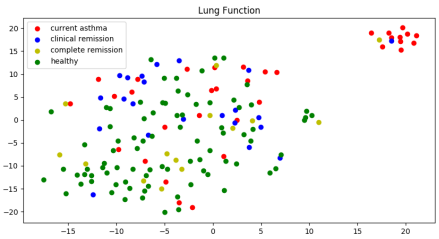
After that, we continue with the examination of GE.txt dataset that contains the gene expression of some patients. The first step is to match the id of each patient from GE.txt file to his id from Database_biopten_v2_6 dataset. Then, it is easy to assign the asthma label to the patient as well. The data is already normalized so we do not need to do any extra procedure. We are able now to fit TSNE and plot the results. Figures 5.5 (a) and (b) are a bit confusing and do not indicate any group clearly. Gene data will be more difficult to analyze due to this fact.

When we come to examine METH.txt and microrna4_aggr_normalized_log.txt files, which are already normalized too, we get similar plots where the data points are not in separate categories. So we get the view that these datasets will be a bit more difficult to process and extract patterns out of them.

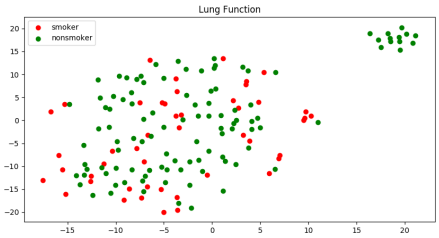
After performing t-SNE technique on our datasets and based on the results, we can say that it would be more efficient to examine further the blood category data first, as well as sputum and biopsy measurements. Following up, we will work with the lung function category that gave some interesting cluster formations. We will examine gene expression data, too, but keeping in mind that it is a more difficult dataset for robust results. We are not going to examine the datasets DNA methylation and microRNA expression data



(a) healthy, asthma



(b) asthma, clinical remission, complete remission, healthy



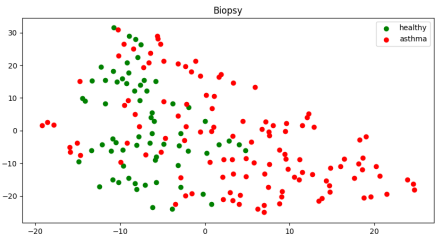
(c) smoker, non smoker

Figure 5.3: Lung function measurements grouped by 3 different label types.

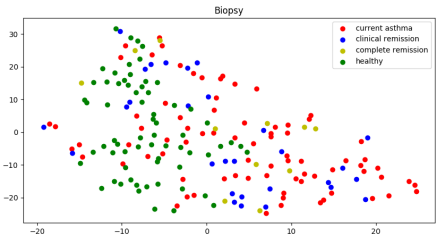
because they are quite complicated and we might get no useful information.

5.2 R implementation

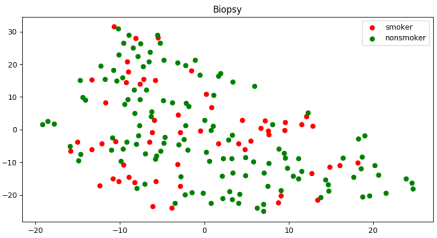
After the application of t-SNE on the data, FABIA algorithm is used for the definition of biclusters. FABIA model is implemented in the R package 'fabia' [19]. The code has been run in desktop with 2.20 GHz CPU and 8 GB RAM. First, Database_biopten_v2_6.csv file is loaded, since we will only work with this dataset. Using the output files from the preprocessing of the data in t-SNE, we get the row and column indices to reconstruct the datasets with no missing values. After that, the dataset needs to be normalized, the same way as in t-SNE, with z-score transformation. The rows of the dataset are the conditions (clinical variables) and the columns are the samples (subjects/patients). FABIA method is called with the dataset, the number of biclusters and the value 0.1 for sparseness factor, as parameters.



(a) healthy, asthma



(b) asthma, clinical remission, complete remission, healthy



(c) smoker, non smoker

Figure 5.4: Biopsy measurements visualized by t-SNE with healthy/asthma labels in (a), asthma/clinical remission/complete remission/healthy in (b) and smoker/non smoker in (c).

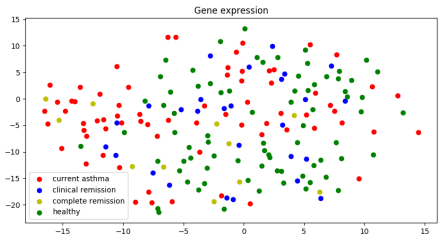
5.3 Biclusters' features

In order to get a first look at the formation of the biclusters and at the features that structure them, we create one counter matrix for each dataset. The matrix has the features of the dataset as rows and the samples as columns. It is initialized to zero. After that, we run FABIA 10 times for each dataset and if one element is included in a bicluster, then its value is increased by 1. Since the biclusters may overlap, some elements can join in more than one of them in each run. After this process, the counter matrices are plotted with heatmaps to get a better understanding. The heatmaps can be seen in Fig. 5.6. White color indicates that the element does not take part in any bicluster. Yellow means that it is included in some biclusters, orange in more and red in even more. With this visualization we know which features and samples to expect in the upcoming biclusters.

5.3.1 Quality Measures

Of course the number of the biclusters cannot be known beforehand. That is why we run the algorithm more than once with different values for the biclusters' number. More specifically, we run it 5 times for 2 biclusters, 5 times for 4 biclusters, 5 times for 6 etc.

(a) healthy, asthma



(b) asthma, clinical remission, complete remission, healthy

Figure 5.5: Gene expression data categorized by healthy/asthma labels in (a) and grouped with asthma/clinical remission/complete remission/healthy labels in (b).

until there are no significant differences between the number of biclusters. For each number, we get an average of the 5 runs of four quality measures. Quality measures are used in every run, in order to evaluate and rank the extracted biclusters. The quality measures that are being used are four.

Let us say that a bicluster B consists of a set I of $|I|$ genes and a set J of $|J|$ conditions, in which b_{ij} refers to the expression level of gene i under sample j. Then B can be represented as seen in Fig. 5.7, where the gene g_i is the i^{th} row and condition c_j is the j^{th} column. Gene and sample means in biclusters are frequently used in several evaluation measure definitions. These values are represented here as b_{iJ} and b_{Ij} referring to the i row (gene) and j column (sample) means, respectively. Furthermore, the mean of all the expression values in B is referred to as b_{IJ} [13].

1) Information Content

This measure shows how much information each bicluster contains about the data and is based on the entropy. Note that the information content grows with the size of biclusters.

2) Variance (VAR)

The variance of a bicluster can be computed with the following math type:

$$VAR(B) = \sum_{i=1}^{|I|} \sum_{j=1}^{|J|} (b_{ij} - b_{IJ})^2 \tag{5.1}$$

Lower values of variance are preferable for each bicluster.

3) Mean Squared Residue (MSR)

The next measure that is used for the assessment of the quality of biclusters is based on

the Mean Squared Residue (MSR). It aims at evaluating the coherence of the genes and conditions of a bicluster. MSR can be calculated with the math type shown below:

$$MSR(B) = \frac{1}{|I|\,|J|}\sum_{i=1}^{|I|}\sum_{j=1}^{|J|}(b_{ij}-b_{iJ}-b_{Ij}+b_{IJ})^2 \tag{5.2}$$

The lower the mean squared residue, the stronger the coherence and better the quality of the bicluster. If a bicluster has zero MSR, it can be considered as a perfect bicluster [13].

4) Virtual Error (VE)

The last quality measure for the evaluation of the biclusters is called Virtual Error (VE). The goal of VE is to measure how features follow the tendency in a bicluster. If all the features of a bicluster follow the same tendency under a set of conditions, then it means that they are activated/deactivated under the same conditions [20]. This kind of biclusters might be interesting for further investigation. In order to catch the tendency of the features, we first calculate a new column, called virtual pattern. Given a bicluster B, we define its virtual pattern p as the set of elements $p = \{p_1, p_2, ..., p_J\}$ where p_i is defined as the mean of i_{th} row:

$$p_i = \frac{1}{|J|}\sum_{j=1}^{J} b_{ij} \tag{5.3}$$

Each point of the virtual pattern represents the average value for all features under a specific condition. After this calculation, we perform standardization in virtual pattern column as well, in order to have all the columns scaled to a common range. We now define VE as the average value of all the differences between the standardized feature values and the standardized virtual pattern [20]:

$$VE(B) = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}\left|\hat{b_{ij}}-\hat{p_i}\right| \tag{5.4}$$

where $\hat{b_{ij}}$ is the standardized feature value of the element $b_{ij} \in$ B, and $\hat{p_i}$ is the standardized value of the element p_i in the virtual pattern p. VE computes the differences between the real features and the virtual pattern, once they have been standardized [20]. As a result, the more similar the features are, the lower VE value they have. Lower VE indicates better bicluster.

5.4 Application of FABIA

After running FABIA 5 times for different number of biclusters, for all four categories, we get the figures shown in Fig. 5.8 of the quality measures. Based on these figures,

we are choosing the number of biclusters for each category, having in my mind that we want a combination of high information content but at the same time low variance, mean squared residue and virtual error. For the first one, blood, we choose 10 biclusters, since the information content is the higher possible and the values of the rest quality measures are getting lower. About lung function, 8 biclusters seem to have a good combination of quality measures' values, same for sputum category as well. For biopsy, 6 biclusters have quite high information content, while keeping lower values for the rest of the measures.

About the gene expression data, the maximum number of biclusters that we could get, due to the power of the machine that is being used, was 10. Gene expression dataset contains 22918 genes and 184 samples, meaning that it is much bigger than the clinical information dataset. So we cannot expect that just 10 biclusters will give useful results. However, we are going to examine FABIA's output with 10 as bicluster number.

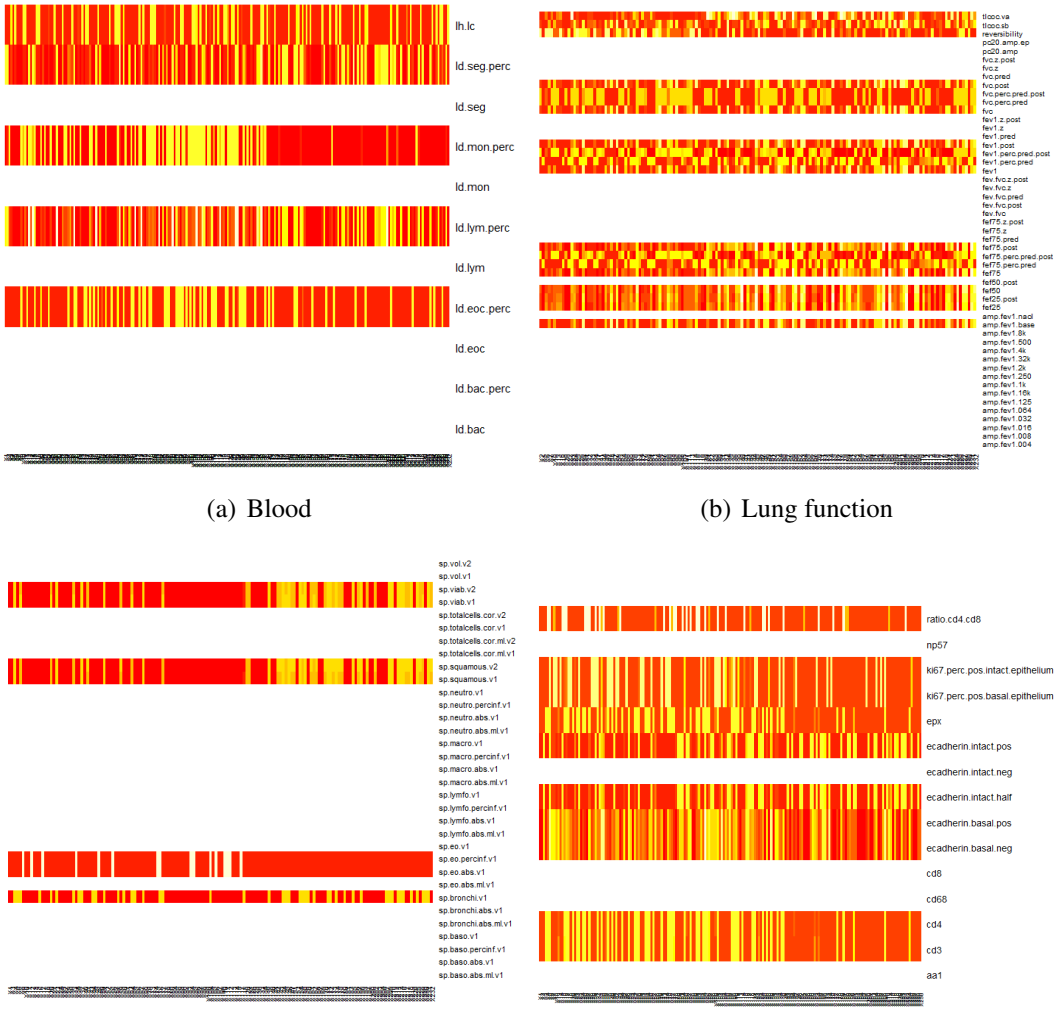
5.5 Robust biclusters

After choosing the number of biclusters for each catogory, we run FABIA algorithm one time in order to get the first biclusters. Next, we run it 10 more times. In each run, we compare all new biclusters with each one of the first run. The goal is to find for every old bicluster the one that is more similar to it from the next run. We have to do n*m comparisons, if the first run contains n biclusters and the next m. To calculate the similarity, we find the amount of common elements between the two biclusters and then divide by the size of the first one. This way, we get the percentage of overlap between these two. After doing all comparisons, we keep the highest percentage of overlap for each bicluster of the first run.

Doing this process 10 times and getting the highest possible percentage of overlap for each bicluster, we can conclude which ones are more robust (those that appear more over the runs). We consider robust the biclusters with average overlap percentage over 80% . In Fig. 5.9 one can see 4 barplots, for each category, where every bar represents the higher percentage for a particular bicluster in one run.

For blood category, we get 3 biclusters with 87.9, 93 and 96.3 percent of overlap, respectively. All of them are over 80% so they will be examined further. Lung function has only 3 robust biclusters, more specifically the first, fourth and fifth one. Sputum has only one robust bicluster that appears with 100% overlap over all runs. About biopsy, all biclusters, except for one, have average overlap percentage over 80% and are going to be examined as well.

Running FABIA for gene expression dataset with 10 biclusters, 10 times, gives the barplot that is shown in Figure 5.10. There is no robust bicluster, meaning that no one has average percentage of overlap over 80% . The bicluster with the maximum average overlap is b8, with 55.9% .



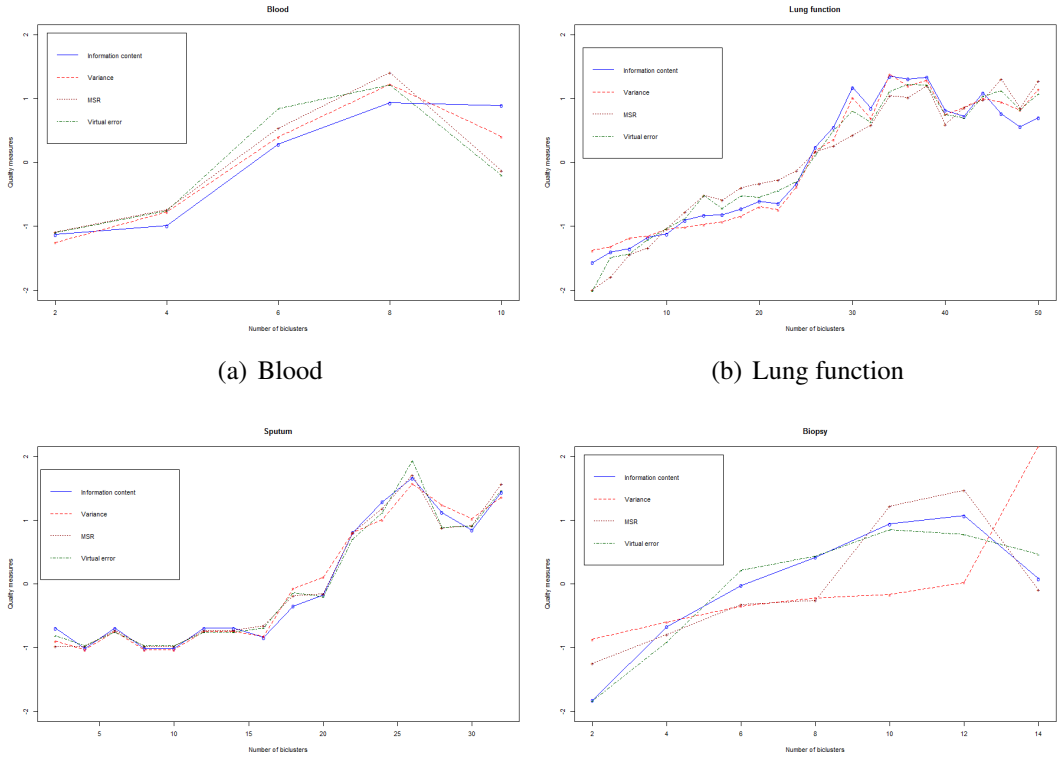
(a) Blood (b) Lung function

(c) Sputum (d) Biopsy

Figure 5.6: Heatmaps of counter matrices for the four categories that represent how many times each feature participates in biclusters.

$$\mathcal{B} = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1|J|} \\ b_{21} & b_{22} & \dots & b_{2|J|} \\ \vdots & \vdots & \ddots & \vdots \\ b_{|I|1} & b_{|I|2} & \dots & b_{|I||J|} \end{pmatrix}$$

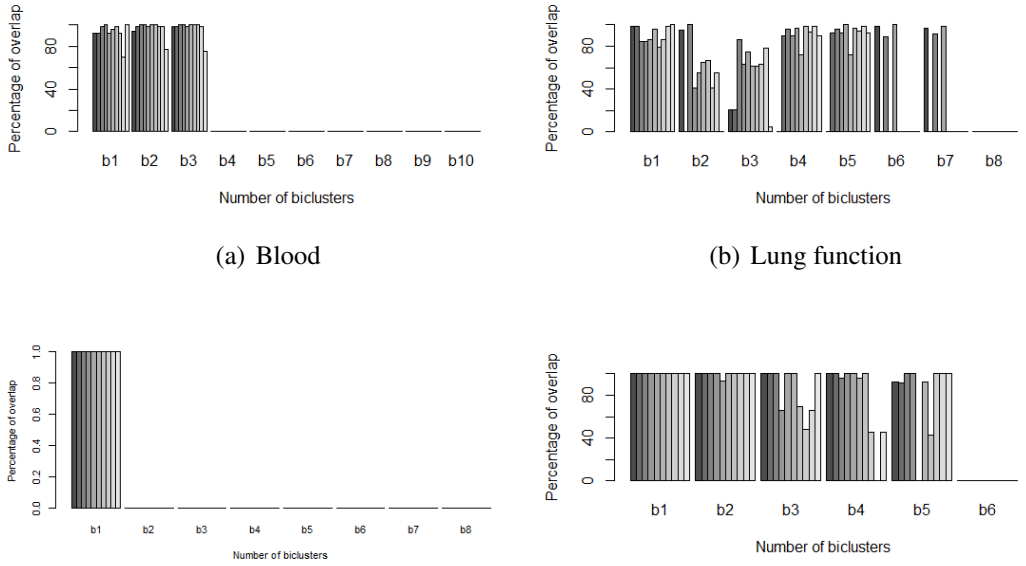
Figure 5.7: Bicluster representation as a 2-dimensional matrix.



(a) Blood (b) Lung function

(c) Sputum (d) Biopsy

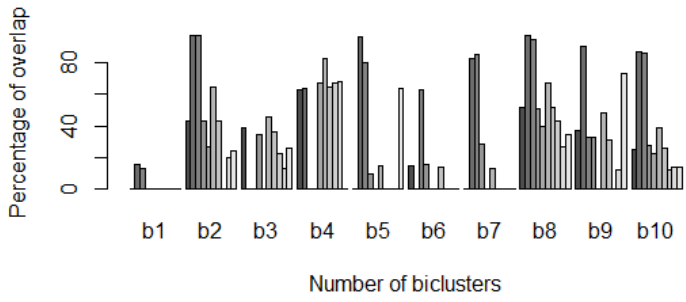
Figure 5.8: Quality measures of 4 categories, where blue curve represents the information content, red curve the variance, dark red curve the MSR and the green one the virtual error.



(a) Blood (b) Lung function

(c) Sputum (d) Biopsy

Figure 5.9: Percentage of overlap for 10 runs for the 4 categories. The robust biclusters are the ones that have average overlap percentage over 80% .



(a) Gene expression

Figure 5.10: Percentage of overlap for 10 runs for gene expression data. No bicluster is robust for this category.

6 Results

6.1 Overview of robust biclusters

Looking through the labels of every robust bicluster, we can come to a conclusion about its composition. The labels that are being used are healthy and asthma.

6.1.1 Blood

This category has 3 robust biclusters. The first one contains 2 features: ld.lym.perc, ld.seg.perc and 66 samples of which 35 are asthma and 31 healthy subjects. Just by these numbers, there is no clear conclusion about the category of this bicluster (healthy or asthma). It will need further examination in order to know if this bicluster is helpful on its own. The second one contains just one feature, ld.mon.perc, with 69 subjects of asthma class and 3 of healthy. The last bicluster has one feature as well, ld.eoc.perc, with 41 asthma subjects and 14 healthy. About the last two biclusters, we can say that they include mainly asthma samples and do not need to be examined more. Regarding the first one, we create its heatmap, in order to check if its features will give further information about the structure of the bicluster.

The two features are anticorrelated, as seen in Fig. 6.1, since ld.seg.perc has negative and ld.lym.perc positive values for all of the samples. There is no clear conclusion based only on the heatmap, but other techniques may give useful results.

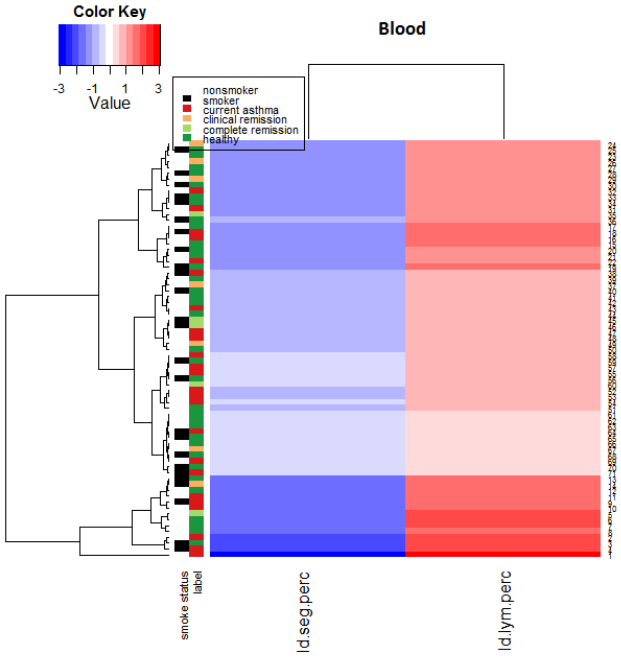
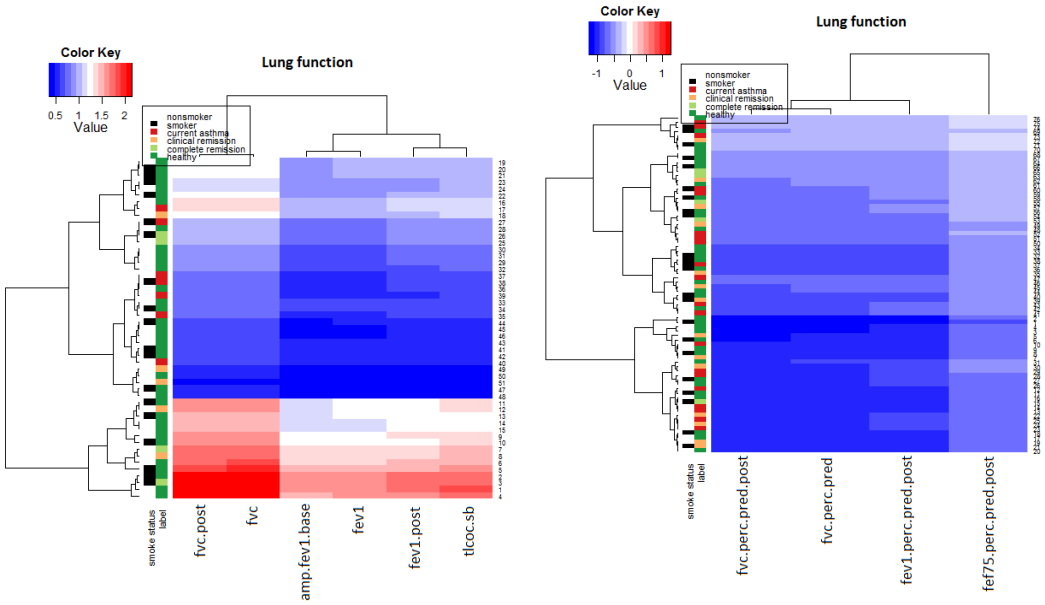


Figure 6.1: Blood robust bicluster's heatmap, that contain ld.seg.perc and ld.lym.perc features. Based on the figure, these two are anticorrelated.

6.1.2 Lung function

In lung function category, there are 3 robust biclusters. The first one consists of 6 features: fvc, fvc.post, tlcoc.sb, fev1.post, fev1, amp.fev1.base. The samples that are contained in this bicluster are 19 of class asthma and 32 of class healthy. The second bicluster contains 4 features: fvc.perc.pred.post, fvc.perc.pred, fev1.perc.pred.post, fef75.perc.pred.post. About the samples, 40 of them are asthma subjects and 37 healthy. The last bicluster contains the same features as the second one and its subjects are 40 asthma and 37 healthy, as well. The heatmaps of the biclusters are shown in Fig. 6.2.

In the first heatmap, there are some particular samples that have higher values for all features and all the rest have lower. When looking at the side label columns on the left, we cannot see any clear distinction of the classes on the higher or lower values. Although, we can say that in higher values are included only healthy and remission subjects. The second and third biclusters have quite similar heatmaps, as the one that is shown in Figure 6.2 (b). The features of these biclusters have low values for all subjects, so we cannot extract any useful information only by this representation.



(a) Bicluster 1 (b) Bicluster 2

Figure 6.2: Lung function robust biclusters, where the first one has distinction on its values for all the features, while the second one does not have.

6.1.3 Sputum

Sputum category gave only one robust bicluster, with 100% overlap over all runs. This bicluster contains 4 features: sp.viab.v2, sp.squamous.v2, sp.viab.v1, sp.squamous.v1. The subjects that it consists of are 15 asthma samples and 38 healthy. If we look at the heatmap of the bicluster, in Figure 6.3, we can see that two features have low values for all the samples and the other two have high values. More specifically, if we look closer, sp.viab.v1 and sp.squamous.v1 are two totally anticorrelated features. We could also say the same about the other two features, sp.viab.v2 and sp.squamous.v2. Therefore, using more advanced techniques, we can investigate if the usage of only one of the anticorrelated features is enough to characterize the bicluster.

6.1.4 Biopsy

The last category, biopsy, has 5 robust biclusters. The first contains 2 features: ecadherin.basal.pos, ecadherin.basal.neg and 46 samples of which 34 are asthma subjects and 12 healthy. The second one has 2 features as well: ecadherin.intact.half, ecadherin.intact.pos and, about the samples, 32 of class asthma and 27 of healthy. These two biclusters do not have clear distinction between the classes, so their heatmaps are going

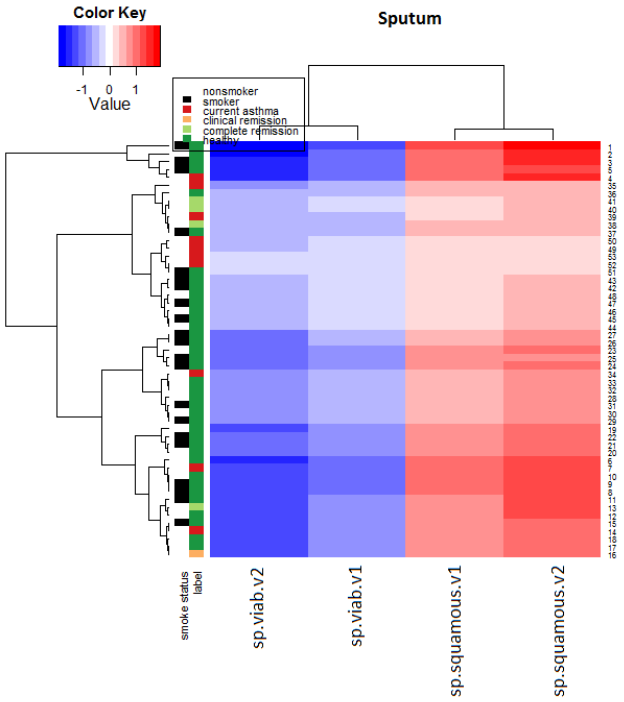


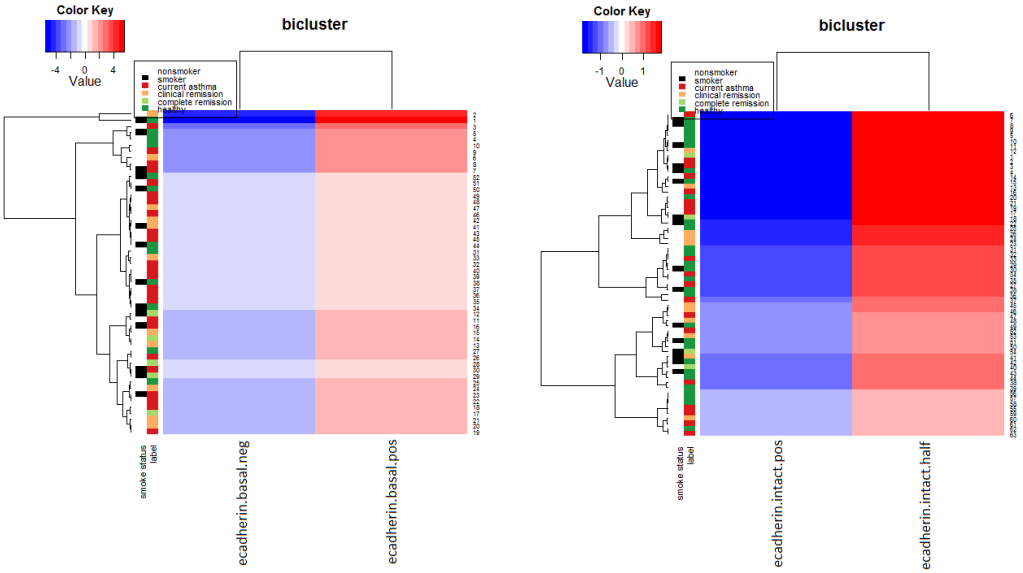
Figure 6.3: Sputum robust bicluster with anticorrelated features.

to be examined as well.

In both heatmaps of Figure 6.4, we can see two totally anticorrelated features, where one has a lower value, the other has a corresponding higher one. With further research, we can investigate if only one of these features is enough for the formation of the bicluster. Following up, the third bicluster contains 2 features: ki67.perc.pos.intact.epithelium, ki67.perc.pos.basal.epithelium with 32 asthma subjects and 9 healthy. The next one has only one feature, ratio.cd4.cd8, with 35 asthma subjects and 6 healthy. The fifth, and final one, consists of 2 features: cd3 and cd4. It has 57 samples in total, of which 51 are asthma subjects and 6 healthy. All of the above biclusters have way more asthma samples than healthy ones. That is why we can consider them as groups of asthma subjects.

6.1.5 Gene expression

For the gene expression dataset, we are going to examine further only the bicluster with the maximum percentage of overlap (55.9%). The heatmap for this bicluster is shown in Figure 6.5. As one can see in the side label annotation, the bicluster contains way more asthma subjects than healthy. More specifically, there are 57 asthma versus 20 healthy.



(a) Bicluster 1 (b) Bicluster 2

Figure 6.4: 2 out of 5 biopsy robust biclusters that both contain two anticorrelated features.

6.2 Cross-data

After examining each category of the clinical information dataset individually, we take combinations between two or three, or even all of the categories together, as one dataset. This way, we investigate if features of different categories are taking part in same biclusters. However, after running FABIA for the new combined datasets, no features from different categories were joining in biclusters together. Instead, the results were the same robust biclusters that FABIA was giving for each category individually, in the previous process.

6.3 Summary

After examining the robust biclusters of every category, we can come to some conclusions about which ones contain more useful information. The categories that seem to be more helpful are blood and biopsy, with 2 and 3 clear asthma biclusters, respectively. The rest of the biclusters, and the other categories, as well, may need more advanced methods in order to extract some asthma patterns out of them. About the combinations of the datasets, there were no new biclusters with features of different categories.

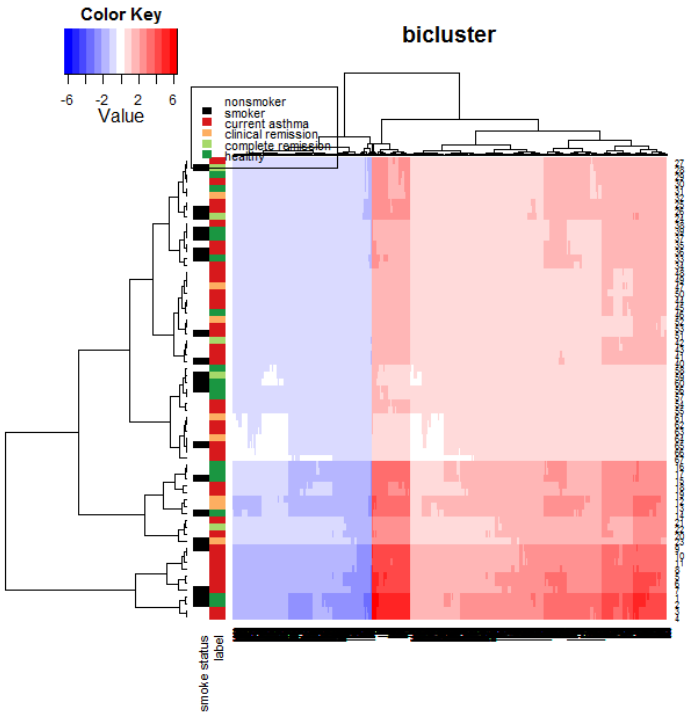


Figure 6.5: Gene expression bicluster which contains significantly more asthma than healthy subjects.

7 Conclusion and Future Work

7.1 Conclusion

In this project we used data analysis and machine learning techniques in order to discover hidden patterns on asthma data. The datasets that we worked with were 4. Clinical information, gene expression, DNA methylation and micro RNA expression. We focused on the first two.

After preprocessing the data, we used t-SNE method to visualize it in 2 dimensions. That way, we had a first image of the groups that are being formed in each dataset. The easiest dataset to work with was the clinical information, that contained data of 4 categories. After this one, we analyzed the gene expression dataset as well. We used FABIA algorithm to get the biclusters of the datasets. In order to rank them, 4 quality measures were used: information content, variance, mean squared residue and virtual error. Based on these, we decided the number of biclusters for each category and ran FABIA.

For each category, we kept the biclusters of the first run and then ran the algorithm 10 more times. Each time, we compared the new biclusters with the initial ones and calculated the percentage of overlap. So, in the end, we considered as robust biclusters those that had average overlap percentage over 80% .When we came to examine the robust biclusters for each category, we found out that some of them have useful information and others need further investigation and more advanced techniques for their analysis. The categories that had more defined biclusters were blood and biopsy, with the first one having 2 and the second 3 mainly asthma biclusters. We tried to work with

combinations of the datasets as well, but no new bicluster popped up with features of different categories.

Contributions of this thesis

- Analysis of medical datasets that gives clues about the useful categories that need further investigation.
- Information to medical experts about which clinical categories indicate the existence of asthma.
- Information about features of each category that indicate asthma, as well.

7.2 Future Work

The next step, for the continuation of this project, is the analysis of DNA methylation and microRNA expression datasets, with FABIA algorithm as well. Since gene expression dataset didn't give robust biclusters with the power of the machine that we were using, it would be a good option to investigate this dataset further with a more powerful one. The same can be done for the two other datasets, since their size is quite big too. Other, more advanced, techniques for the analysis of asthma can be used also.

GFA GFA (Group Factor Analysis) [21], for example, would be a good option. This method is an extension of factor analysis, that finds factors that correlate datasets, instead of individual variables [22] [23]. From another perspective, GFA extends multi-battery factor analysis (MBFA), introduced by McDonald [24] and Browne [25] as a generalization of inter-battery factor analysis (IBFA) [26] [27] to more than two variable groups. Either way, with GFA one could work with the different datasets that we have and not only with the features included in each of them.

Supervised learning Supervised learning methods can be applied also. With models that are trained using the labels too, the biclusters may be better defined and have more clear distinction between asthma and healthy subjects. They can even perform quite well with all four labels: asthma, clinical remission, complete remission and healthy. Both supervised [28] and semi-supervised [29] biclustering can be used on microarray gene expression data.

References

[1] M. Broekema, W. Timens, J. M. Vonk, F. Volbeda, M. E. Lodewijk, M. N. Hylkema, N. H. T. ten Hacken, D. S. Postma, "Persisting remodeling and less airway wall eosinophil activation in complete remission of asthma," *American Journal of Respiratory and Critical Care Medicine*, 2010.

[2] S. T. Holgate, S. Wenzel, D. S. Postma, S. T. Weiss, H. Renz, P. D. Sly, "Asthma," *Nature Reviews Disease Primers*, 2015.

[3] M. Ram, A. Najafi, M. T. Shakeri, "Classification and biomarker genes selection for cancer gene expression data using random forest," *Iranian Journal of Pathology*, 2017.

[4] W. Dubitzky, M. Granzow, D. Berrar, "Data mining and machine learning methods for microarray analysis," *Methods of Microarray Data Analysis*, 2002.

[5] M. B. Eisen, P. T. Spellman, P. O. Brown, D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proceedings of the National Academy of Sciences of the United States of America*, 1998.

[6] H. Wang, W. Wang, J. Yang, P. S. Yu, "Clustering by pattern similarity in large data sets," *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, 2002.

[7] J. Hartigan, "Direct clustering of a data matrix," *Journal of the american statistical association*, 1972.

[8] Y. Cheng, G. M. Church, "Biclustering of expression data," *n Proceedings for the Eighth International Conference on Intelligent Systems for Molecular Biology*, 2000.

[9] B. Pontes, R. Girldez, J. S. Aguilar-Ruiz, "Biclustering on expression data: A review," *Journal of biomedical informatics*, 2015.

[10] S. Hochreiter, U. Bodenhofer, M. Heusel, A. Mayr, A. Mitterecker, A. Kasim, T. Khamiakova, S. Van Sanden, D. Lin, W. Talloen, L. Bijnens, H. W. H. Ghlmann, Z.

Shkedy, D. A. Clevert, "Fabia: factor analysis for bicluster acquisition," *Bioinformatics*, 2010.

[11] L. van der Maaten, G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, 2008.

[12] A. Tanay, R. Shamir, R. Sharan, "Biclustering algorithms: A survey," *Handbook of computational molecular biology*, 2004.

[13] B. Pontes, R. Girldez, J. S. Aguilar-Ruiz, "Quality measures for gene expression biclusters," *PloS ONE*, 2015.

[14] G. Getz, E. Levine, E. Domany, "Coupled two-way clustering analysis of gene microarray data," *Proceedings of the National Academy of Sciences of the United States of America*, 2000.

[15] G. Getz, E. Levine, E. Domany, M.Q. Zhang, "Super-paramagnetic clustering of yeast gene expression profiles," *Physica A: Statistical Mechanics and its Applications*, 2000.

[16] A. Tanay, R. Sharan, M. Kupiec, R. Shamir, "Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data," *Proceedings of the National Academy of Sciences of the United States of America*, 2004.

[17] R. Sharan, A. Maron-Katz, N. Arbili, R. Shamir, "Expander: Expression analyzer and displayer," *BMC Bioinformatics*, 2002.

[18] Y. Kluger, R. Barsi, JT. Cheng, M. Gerstein, "Spectral biclustering of microarray data: coclustering genes and conditions," *Genome research*, 2003.

[19] S. Hochreiter, "Fabia: factor analysis for bicluster acquisition - manual for the r package," *Bioconductor*, 2018.

[20] F. Divina, B. Pontes, R. Giraldez, J. S. Aguilar-Ruiz, "An effective measure for assessing the quality of biclusters," *Computers in biology and Medicine*, 2011.

[21] A. Klami, S. Virtanen, E. Leppaaho, S. Kaski, "Group factor analysis," *IEEE Transactions on Neural Networks and Learning Systems*, 2014.

[22] K. Bunte, E. Leppaaho, I. Saarinen, S. Kaski, "Sparse group factor analysis for biclustering of multiple data sources," *Bioinformatics*, 2016.

[23] S. Virtanen, A. Klami, S. A. Khan, S. Kaski, "Bayesian group factor analysis," *Artificial Intelligence and Statistics*, 2012.

[24] R. McDonald, "Three common factor models for groups of variables," *Psychometrika*, 1970.

[25] M. Browne, "Factor analysis of multiple batteries by maximum likelihood," *British Journal of Mathematical and Statistical Psychology*, 1980.

[26] L. R. Tucker, "An inter-battery method of factor analysis," *Psychometrika*, 1958.

[27] M.W. Browner, "The maximum-likelihood solution in inter-battery factor analysis," *British Journal of Mathematical and Statistical Psychology*, 1979.

[28] P. M. Pardalos, S. Busygin, O. A. Prokopyev, "On biclustering with feature selection for microarray data sets," *BIOMAT*, 2006.

[29] H. L. Turner, T. C. Bailey, W. J. Krzanowski, C. A. Hemingway, "Biclustering models for structured microarray data," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2005.