Dependencies between transcription factor binding sites: comparison between ICA, NMF, PLSA and frequent sets

Heli Hiisilä and Ella Bingham
Neural Networks Research Centre, Laboratory of Computer and Information Science
Helsinki University of Technology, P.O. Box 5400, FIN-02015 HUT, Finland
heli.hiisila@hut.fi, ella@iki.fi

Abstract

Gene expression of eucaryotes is regulated through transcription factors, which are molecules able to attach to the binding sites in the DNA sequence. These binding sites are small pieces of DNA usually found upstream from the gene they regulate. As the binding sites play an important role in the gene expression, it is of interest to find out their characteristics.

In this paper we look for dependencies and independencies between these binding sites using independent component analysis (ICA), non-negative matrix factorization (NMF), probabilistic latent semantic analysis (PLSA) and the method of frequent sets. The data used are human gene upstream regions and possible binding sites listed in a biological database. Also, results on the baker's yeast (S.Cerevisiae) upstream regions are briefly discussed for comparison.

ICA, NMF and PLSA are latent variable methods that decompose the observed data into smaller components. Of these, ICA and NMF were originally aimed for continuous data. We show that these methods can be successfully used on discrete DNA data as well. PLSA and the method of frequent sets were created for discrete data sets.

The above methods reveal partially overlapping sets of possible binding sites such that the binding sites within a set are dependent of each other. The methods of frequent sets and NMF give a good overview of the most common data structures, whereas using ICA and PLSA we find large sets that are surprisingly frequent. That is, sets of very frequently occurring possible binding sites can be found near hundreds or thousands of genes; also interesting but less frequent ones co-occur surprisingly often.

1 Introduction

In the eucaryotic cells such as yeast or human, certain proteins, which are called *transcription factors* (TF), influence the transcription of a gene and thus participate in the gene expression. Factors bind to short pieces of DNA strand found mostly at the promoter regions upstream from the gene. These pieces of DNA are called as transcription factor binding sites, or in short, motifs.

The detection of these relatively small pieces of DNA that may regulate the gene expression has been under study in the past few years. Computational, probabilistic and statistical approaches [1][2][7][20] have been used, as well as methods utilizing gene expression data [3][14] to find possible DNA binding sites from the DNA sequence. The TRANSFAC [19] database contains a large amount of known transcription factor binding sites. In this study, we aim at characterizing the co-occurrences of these binding sites.

We are not able to detect online how the transcription happens with different genes, but as the DNA sequence includes the information of how this process is done, we may use it to get new knowledge of these processes. Two biologically interesting questions are: Are there common motifs, possible binding sites, near many genes? If there are, are there groups of these motifs co-occurring near many genes? These dependencies are looked for using four different kinds of algorithms: frequent sets, ICA, NMF and PLSA.

The idea of frequent sets is intuitive because we get the most frequently occurring patterns of the data as an output of the algorithm. The latent variable methods ICA and NMF have originally been developed to decompose a continuous data set into smaller parts, components. Here we use these methods to find components from a discrete data matrix. PLSA is a latent variable model that has a probabilistic point of view.

The aim in this paper is to find interesting patterns that characterize the dependencies of the motifs in the data set well or patterns that are surprising, and to provide a comparison between the methods used. We will introduce the methods briefly in section 2. Experimental results are given in section 3. Also baker's yeast upstream regions were used as a comparison, and the results are briefly discussed in section 4. Section 5 gives a conclusion.

2 Methods

The starting point of our study is an observation matrix X of attributes and observations. Each column of X is an upstream region of a gene (that is, an observation). The rows of X correspond to different motifs (attributes). If an upstream region contains a motif, the corresponding entry of X has a value 1; otherwise the value is 0. To find dependencies between the attributes, the method of frequent sets [18], independent component analysis (ICA) [12], non-negative matrix factorization (NMF) [16] and probabilistic latent semantic analysis (PLSA) [10] are used.

2.1 Frequent sets

Advanced algorithms for finding frequent collections of events from databases are introduced by Mannila et al. [18]. The basic idea is to find sets of events that co-occur frequently. In the case of matrices built of the occurrences of motifs in the gene upstream regions, frequent sets are groups of DNA motifs occurring together in many upstream regions.

The algorithm used for finding frequent sets uses the Apriori candidate generation. The search for frequent sets begins from the sets of only one attribute, and frequent sets of size n are used when creating the candidates of sets of size $n+1$. The iterations end when no more frequent sets can be found.

The frequency threshold percentage parameter gives the limit of how many times must a set occur in the data set so that it is called frequent. With low values of the parameter, the amount of frequent sets tends to grow very large, and the results are not easy to analyze. With large parameter values only very common patterns in the data are revealed.

2.2 Independent component analysis (ICA)

Independent Component Analysis (ICA) [5][12][13] is a sophisticated method for finding latent variables s, that are statistically independent of each other, from the observed data. It is assumed that the observed data can be represented as a linear mixture: $x_j = a_{j1}s_1 + a_{j2}s_2 + \ldots + a_{jn}s_n$ for all j, where a_{ji} are mixing coefficients, all s_i are latent variables and x_j is an entry of a vector x.

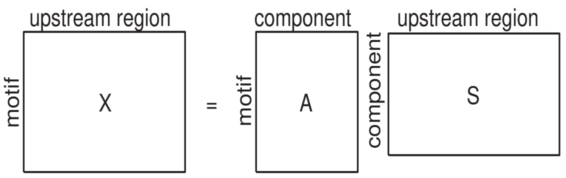

Figure 1. ICA mixing model. X matrix is the data matrix, A matrix explains which motifs (attributes) belong to which components and the S matrix lists the upstream regions (observations) where each component is found.

This mixing model may be written using vector-matrix notation:

$$x = As \tag{1}$$

where A is a matrix that has elements a_{ji}. The matrix A is unknown and constant for all observations x. Figure 1 illustrates the matrices of the mixing model of equation 1. The matrix X has observation vectors x as its columns. The columns of the A matrix explain to which degree each attribute is present in each component. The matrix S contains the vectors s as its rows.

Independent components can be estimated from the observed data with an assumption of the component independence and non-Gaussian distribution and assuming A is of full column rank [5].

The equation 1 may be written in a form $s = A^{-1}x = Wx$ when A^{-1} is a denoted by W. The ICA problem is solved by finding W such that the resulting s are statistically independent. Algorithms for ICA are given e.g. in [12]. In this paper we use the FastICA [13] [9] and we choose to maximize the skewness of s.

ICA was originally developed for continuous data in signal processing. Lately it has been discovered that it is a promising method for discrete data as well, for example text documents [4].

2.3 Non-negative matrix factorization (NMF)

The basic idea of Non-negative matrix factorization (NMF) is to search for matrices A and S such that $X = AS$, using iterative update rules given in [16] [17].

The setting is similar to Figure 1. But in contrast to ICA, in NMF one does not assume anything about the nature of A and S, except for nonnegativity. Again a column of the A matrix gives the attributes belonging to a latent component. In this paper we use an algorithm that minimizes the Euclidean distance $\|X - AS\|$, presented in [17].

2.4 Probabilistic Latent Semantic Analysis (PLSA)

Thomas Hofmann [10] [11] has created a method called probabilistic latent semantic analysis (PLSA) for factor analysis of count data. PLSA is influenced by latent semantic analysis (LSA), but has a probabilistic point of view.

The idea of PLSA is as follows. The observations (d) consist of occurrences of attributes (w). The prior probability for observation d is $P(d)$. $P(z|d)$ is the probability to pick a latent variable z in observation d and $P(w|z)$ is the probability to generate an attribute w, given latent variable z. Observations d and attributes w are assumed to be conditionally independent given z. An occurrence of attribute w in observation d is represented as a pair (d, w).

Data generation process may be written as [10] $P(d, w) = P(d)P(w|d)$, where $P(w|d) = \sum_z P(w|z)P(z|d)$, giving $P(d, w) = P(d) \sum_z P(w|z)P(z|d)$.

The goal is to identify the conditional probabilities $P(w|z)$ and $P(z|d)$ by maximizing the log-likelihood function using the EM algorithm. In the expectation (E) step, posterior probabilities will be computed for the latent variables and in the M-step the expected complete data log-likelihood should be maximized. Each component or latent variable z can be characterized by listing the attributes w which give the largest values for $P(w|z)$. This is analogous to listing the attributes having a high value in a column of the of the A matrix in ICA and NMF.

2.5 Selecting the relevant sets

In the sequel, we will use the term "component" to refer to a set of motifs (transcription factor binding sites). Such a set is identified either as a frequent set, or as attributes having a large value in a column of the A matrix in ICA or NMF or as attributes w having a large value of $P(w|z)$ in PLSA.

One problem with all the methods described in this section is that it is not easy to select the parameters defining the amount of components to be looked for. A fair selection of these parameters is especially important when comparing the results produced by different methods.

When the data matrix is large, as in the case of human gene upstream regions, it may also contain a large amount of components. The solution selected here is to first look for a relatively large amount of components, and then select the best components based on certain criteria. With the smaller yeast data of section 4 this is unnecessary as we are able to analyze all the components found.

A good method finds components explaining the observations well. Also the components should not be trivial, but rather reveal some underlying structure in the data set, and be surprisingly frequent.

With each method described above in this section, the component selection used here goes as follows. First we find a large set of components. Then we choose, one by one, the components that cover the observations best. At each step the best component explains a larger amount of observations than any other component. When the best component is found, the 1's explained by this component are removed from the data matrix, and the remaining ones may still be explained by other components. A certain amount of the best components are then reported. This procedure is used with each method described in this section.

Another desired feature of a component is that it is surprising. Surprisingness here is defined based on the amount of component occurrences $F(C)$ in the data and the expected frequency $E(C)$ of the component C. The expected frequency of the component is calculated based on the frequencies of the attributes of the component:

$$E(C) = N \prod_{i \text{ s.t. } w_i \in C} f(w_i) \tag{2}$$

where C is the component, N is the total number of upstream regions in the matrix X, w_i is a transcription factor binding site belonging to the component C, and $f(w_i)$ is the frequency of the factor w_i in the binary matrix X. $F(C)$ is the amount of upstream regions where an occurrence of component C can be found. The ratio $F(C)/E(C)$ describes how surprising the component C is based on the frequencies of the attribute w_i belonging to the component C.

3 Experimental results

3.1 Data used

The upstream regions of 18406 human genes were downloaded from [6], and the areas of 1000 bp upstream from the genes were selected. Also a large collection of transcription factor (TF) binding site sequences (motifs) was downloaded from the TRANSFAC database [19]. Very long and very short motifs were left out as well as ones including an other motif as a subsequence. After removing these uninteresting motifs, a total of 216 motifs were selected.

When starting to analyze this problem, we have only little knowledge of the TF binding sites in the upstream regions. The aim is to discover both common and more rare but surprising structures from the data using different methods. A binary matrix, like the X matrix of figure 1, was created by searching for motif occurrences in each upstream region. The X matrix of size 216×18406 contains approximately 6.8% ones and 93.2% zeros, and X is thus quite sparse. There are many motifs, binding sites, that occur very frequently, and the 10 most common ones are introduced in Table 1. Each of these motifs may be found in as many as 37-50% of all upstream regions.

Table 1. Most common motifs and their frequencies in the observation matrix.

Motif	$F(w)$
GGCGGG	9304
CCGCCC	8947
GGGCGG	8872
GGTGGG	8544
GGGAAG	8266
GGGGCC	7869
AGGCGG	7178
TTCCTT	6925
AAGGAA	6849
GAGGCG	6737

3.2 Results

The amount of components looked for with ICA, NMF and PLSA methods was 200, and the frequency threshold percentage for finding about 200 frequent sets was 10%. Components with only one motif were left out, as they do not include information about the relationships of the motifs. For each method ten components giving the best coverage of the data are reported here.

The results given by different methods are shown in Tables 2, 3, 4, 5 and 6. Some comparison between the methods can be found in the section 3.3 and discussion about the biological relevance of the results in the section 3.4.

3.2.1 Frequent sets

First the frequent sets were searched for and the sets giving the best coverage of the data are reported in Table 2. For comparison, maximal frequent sets were also extracted, and the sets covering the observations are available in Table 3. The frequency threshold parameter in both experiments was 10%, that is, each frequent set can be found in at least 10% of gene upstream regions.

The frequent sets of Table 2 include groups of motifs that appear very frequently in the gene upstream regions. These 10 components cover 25% of all observations, and thus give a good overview of the most common structures in the data matrix. The components are short, as they only include two motifs. Three of the first components include overlapping motifs and are not thus very interesting, even though consensus motifs GGGCGGG, CCCCGCCC, GAGGCGG must be very important motifs, as they can be found so frequently in the data. The other sets 4 to 10 are nontrivial, although not very surprising, and found in more than 1200 gene upstream regions.

Table 2. Frequent sets giving the best coverage of the data.

	Frequent set C	$F(C)$	$\frac{F(C)}{E(C)}$
1	GGGCGG GGCGGG	7437	1.7
2	CCGCCC CCCCGCC	5215	1.8
3	GAGGCG AGGCGG	5123	1.9
4	GGGGCC GGTGGG	4442	1.2
5	GGGAAG AAGGAA	3208	1.0
6	TTCCTT TTTCCA	2708	1.1
7	CACCTG TGTGGG	2125	1.0
8	GGGAAG CCCCTCC	1763	1.2
9	AAGGAA GGAAAT	1280	1.3
10	TTCCTT AAGTGA	1237	1.1

Table 3. Maximal frequent sets giving the best coverage of the data.

	Maximal frequent set C	$F(C)$	$\frac{F(C)}{E(C)}$
1	GGGCGG GGCGGG CCGCCC GAGGCG AGGCGG	2194	7.0
2	GGGCGG GGCGGG GGGGCC GGGAAG	1628	3.3
3	AAGGAA GGTGGG	3001	0.94
4	TTCCTT TTTCCA	2708	1.1
5	CCCCTCC CCGCCC CCCCGCC	1758	2.7
6	GGGAAG CACCTG	2418	1.0
7	GGGGCC TGTGGG	2092	1.1
8	GGTGGG GACCCT	1537	1.1
9	CCGCCC ACGCCC	1446	1.3
10	TGTCCT GGCGGG	1413	0.97

The maximal frequent sets with best coverage of observations shown in Table 3 explain 19% of all 1's of the data matrix. Component 1 includes components 1 and 2 of Table 2, and can still be found in many upstream regions. Also component 2 includes component 1 of Table 2 and component 5 includes component 2 of Table 2. Other components of Tables 2 and 3 seems to also have some common motifs, but with other pairs.

3.2.2 ICA components

The FastICA algorithm was run and 200 independent components were estimated. For each component i we picked those motifs j for which the entry a_{ij} in A was significantly larger than zero. Components with only one motif were left out as was done in the case of frequent sets, too. Ten components giving the best coverage of the data were selected, and represented in Table 4.

Components of Table 4 cover only about 7% of the observations, but there are some longer and more surprising components also included. For example, component 4 of Table 4 can be found in 211 gene upstream regions, though

Table 4. ICA giving the best coverage of the data.

	ICA component C	$F(C)$	$\frac{F(C)}{E(C)}$
1	GGGAAG GGGGCC GGTGGG TGTGGG	1002	2.0
2	TTTCCA TGTCCT	1817	1.0
3	TTGGCA GATTGG	856	1.3
4	GGGCGG GAGGCG GGGGCC GGCGGG CCCCTCC CCGCCC CCCCGCC ACGCCC	211	33
5	TTCCTT CTTTGCT	794	1.2
6	CTATCT TTTCCA	697	1.1
7	CAAAAGG AAGGAA	681	1.4
8	GGGAAG TGCAAAG	650	1.1
9	CACCTG CATCTG TGTCCT	372	1.4
10	GGGCGG GGGAAG CACCTG GGTGGG GACCCT	202	1.9

it has as many as 8 motifs. Component 8 shares common motifs with maximal frequent sets 2 and 5 of Table 3. One may consider these long independent components as a kind of summary components, as the subsets of these components are very frequent.

3.2.3 NMF components

The NMF algorithm was run and 200 components were searched. As in the case of ICA, we picked those motifs j whose entry a_{ij} in A was significantly larger than zero. Again, components with only one motif were left out and the components giving the best coverage can be found in Table 5.

Table 5. NMF components giving the best coverage of the data.

	NMF component C	$F(C)$	$\frac{F(C)}{E(C)}$
1	GGGCGG GGCGGG	7437	1.7
2	CCGCCC CCCCGCC	5215	1.8
3	GAGGCG AGGCGG	5123	1.9
4	GGGAAG GGGGCC	4122	1.2
5	AAGGAA GGTGGG	3001	0.94
6	TTCCTT TTTCCA	2708	1.1
7	CCCCTCC CTGCCCC	1678	1.5
8	TGGGTC GGTGGG	1455	1.2
9	GGTGGG CCGCCC	915	1.1
10	GGGGCC CCGCCC	621	1.1

The NMF components of Table 5 give a nice coverage of 24% of the data, but contain only two motifs each. The motifs in the NMF components are also very common. NMF algorithm has also found many components where the motifs are overlapping. This kind of components are 1 (with consensus sequence GGGCGGG), 2 (CCCCGCCC) and 3 (GAGGCGG). These components are a result of a longer consensus motif, not of two different motifs, like other NMF components found. There are several common components found by NMF (Table 5) and frequent sets (Table 2).

3.2.4 PLSA components

Instead of decomposing X into A and S, PLSA gives the probabilities of motifs in latent components. For each component z we pick the motifs w whose probability $P(w|z)$ is significantly larger than zero. The PLSA algorithm was used to estimate 200 components from the data set, and the components giving the best coverage of the data are shown in Table 6. Again, components consisting of one motif only were discarded.

Table 6. PLSA components giving the best coverage of the data.

	PLSA component C	$F(C)$	$\frac{F(C)}{E(C)}$
1	AAGGAA TCGCGG	2068	1.4
2	TAGCGGGTG GAGGCG	1617	100
3	TTCCGA CACCTG	1614	2.0
4	ACCTGTT TTCCTT	1613	3.2
5	CACAGGAA CCTTATATGG GGGGCC	760	3200
6	TTCCGA TCGCGG	235	1.8
7	GGTTAG GAGGATGT	200	4.5
8	TATGGTAAT TTCCTT	104	8.4
9	CACAGGAA CCTTATATGG ATTGAA	63	1300
10	CACAGGAA CCCCACCTC	65	3.5

Though PLSA components of Table 6 cover only 4% of the data, they are quite interesting. There are many longer and less frequent motifs in the components, which makes components like 5 and 9 quite surprising.

3.3 Comparison of methods

When analyzing biological data sets, one should always keep in mind to use more than one computational method. Also the selection of parameters may be tricky, as they may have a major impact to the results. When selecting the method to be used, one also needs to decide whether to concentrate to the big picture or to details.

Components found using frequent sets naturally include most common and thus quite short motifs. Components with longer and more frequent motifs are not found using the method of frequent sets. If details are wanted, a very small frequency threshold parameter must be used, and the amount of frequent sets found grows very high, which makes finding interesting sets gets even more difficult. The strength of the frequent sets method is to get exact knowledge of the common structures of data. Other complimentary methods may be used when details is what we are looking for. Maximal frequent sets is a more compact way of presenting components and rules of the data structure as well. However, frequent sets did give a better coverage of data, as the found components were shorter and matched with more observations.

In general, ICA finds relatively long components that are not very frequent. However, the subsets of these long components are very frequent. It seems that ICA is able to create summary components of the data and thus find underlying structures. As ICA is able to extract these surprising long components, it is also a good method for extracting interesting details from the data set. Most of the independent components found in different runs are similar.

NMF components found are similar to frequent sets: they are short, and contain very common motifs. NMF gives a very good coverage of data set, but the components found are only pairs of motifs. The selection of parameter values seems to have more effect to NMF than to other methods, and longer components may be found with different amount of components to be estimated.

PLSA found components with rare and long motifs. These motifs co-occur together very often. PLSA is most suitable for count data instead of binary data, which may be one of the reasons why PLSA did not cover the data well. Also, in PLSA it is assumed that all attributes (motifs) belonging to a component might not appear in the same observation (upstream region). However, PLSA found most surprising components: components containing motifs that have strong dependencies.

The data coverage of the components found by each of the methods may seem poor, but one must remember that we have discarded components consisting of one motif only. The data could be nicely covered with these motifs that are very common, but in this study we aim at finding relationships between the motifs.

One of the advantages of latent variable methods such as ICA, NMF and PLSA is that they give a parsimonious representation of the data. We may present the data as a set of latent variables, and these latent variables can be described either as lists of representative attributes (here, motifs) or as lists of representative observations (here, upstream regions). This can be seen as multi-way clustering, where the clusters are allowed to overlap.

3.4 Biological relevance

There are still much unknown issues when it comes to transcription and gene expression in general. There are transcription factors and there are sites where they bind, but we are not able to detect how this all really happens. Because of this, we must use the data we have: DNA sequence, which includes the binding sites.

Based on the experiments with frequent sets and NMF, one may conclude that there is a group of very common motifs occurring in the upstream regions of genes. As all of these motifs have been listed in biological databases as transcription factor binding sites of certain genes, it is quite interesting that they can be found near thousands of other genes also. This common vocabulary might suggest that if certain genes have similar regulatory patterns, their gene expressions have common functionalities. ICA and PLSA algorithms found less frequent components, but the motifs of these components have strong dependencies. So there are also less frequent binding sites that co-occur together very often.

An interesting application of the structures found would be to cluster genes based on these binding site profiles. This kind of clustering would certainly give us new knowledge of the genes based on their regulatory regions. Also interesting topics are to compare transcription profiles of gene families or genes of different chromosomes.

The structures found and introduced in this study are characteristic to a large amount of gene upstream regions. This knowledge of the sequences before the genes could be used to improve gene prediction. As the known genes have groups of motifs shown in this paper, similar groups may probably be found in the upstream regions of new genes also. If some of the components introduced previously are found in the DNA, a gene may very well be near by.

4 Yeast data

As a comparison to human upstream regions, the same experiments were done with the upstream regions of baker's yeast, Saccharomyces Cerevisiae. The data set is much smaller, as there were 6349 gene upstream regions and 48 motifs. Similar binary matrix was created based on this data, and the methods introduced previously were used. A short summary of the results is given in section 4.1 and human and yeast results are briefly discussed in section 4.2.

The yeast data included 9 very common motifs: ATTTTT (4474 upstream regions include at least one occurrence), TTATC (4383), GATAA (4343), TGAAAA (3647), ATATAA (3235), GAGGA (3079), TATAAA (3030), TCTCC (3005), CATCC (2761). All of these motifs can be found in more than 43% of all upstream sequences and they all co-occur in 32 different upstream regions. When comparing to the most common human motifs of Table 1, there are no matches.

4.1 Yeast results

Due to the limited space, we cannot present all components found by different methods here. Let us just list a few interesting examples of the results on yeast.

About 20 components were looked for using the four algorithms, and because of the smaller size of the data set, they were all analyzed. Component selection such as what was described in Section 2.5 was not needed here.

Maximal frequent sets found give the most common pairs of motifs and cover 56% of the data, but the sets are not surprising. In that sense, there is no special intelligence in the method: the most common components are selected based on the frequencies and the chosen frequency threshold parameter. An interesting example of the frequent sets found is a component containing motifs ATTTTT, GATAA and TTATC. Motifs GATAA and TTATC are complements and reversed versions of each other and the component can be found in 2225 yeast gene upstream regions.

Independent components found were surprising, as the components occur in average 1.6 times more often than one could expect based on the motif frequencies. Also the components were larger and contain a larger amount of different motifs than components found with other methods. The data coverage of 32% is also fairly good. One of the most interesting ICA components was found in 73 gene upstream regions and had 6 motifs: ATATAA, ATTATCAA, GAGGA, GATAA, TATAAA and TTATC, and thus includes at least binding sites of well-known transcription factors TBP/TATA, MATalpha2 and GATA.

The NMF components found cover approximately 41% of the data. The components found were in average longer and more diverse than the ones found by the basic method of frequent sets. However, NMF components were not as surprising as the ICA components.

PLSA algorithm did a poor job in finding components of motifs from the yeast gene upstream regions. The components were quite rare and short and not really surprising. The PLSA components covered only 6.8% of the ones in the data matrix. Also the amount of components to be estimated has a major effect to the components found by the PLSA algorithm.

4.2 Comparison of human and yeast results

The analysis of yeast data is very different from the analysis of human data as the sizes of the data sets differ a lot. In the yeast data we were able to study all components exhaustively whereas in the case of human data we must select a subset of the components for display.

Human and yeast transcription factor binding sites are dissimilar, and thus it is not relevant to compare the exact components found. However, there is a certain group of very common motifs in the upstream regions of both species. Both organisms have also larger groups of motifs that are less common. Yeast motifs include more A's and T's whereas human motifs have much G's and C's.

The results of frequent sets on human and yeast data are quite similar in nature: simple components consisting of very common motifs are found. ICA may find several motifs, such that subsets of these motif sets are very common, meaning that an ICA component gives a summary of common patterns in the data. NMF found larger groups of yeast motifs than human motifs. PLSA did a poor job with the smaller yeast data, whereas PLSA results with human data are quite interesting.

5 Conclusions

What we have learned by doing this study is that there are many common structures of possible binding sites in the upstream regions of genes. Some are very frequent, occurring before thousands of genes; some other structures include many motifs which co-occur surprisingly often. Without going to details, one may conclude that the upstream regions have a common vocabulary, which may be a key to understanding the gene expression better. If certain genes have similar regulatory patterns, their gene expression may have common functionalities. Similar results were obtained with both human and yeast data sets.

Different kinds of approaches may be taken when decomposing a data matrix into smaller parts. In addition to methods discussed in this paper — frequent sets, ICA, NMF and PLSA — there are others suitable for binary observations. To mention a few, the Proximus method [15] decomposes the data matrix recursively by rank-one approximations consisting of binary feature vectors and their binary occurrence vectors. However, there is no latent variable interpretation in their model. Single-cause Bernoulli mixtures [8] are a common method but are restricted to assuming one latent cause per observation vector, whereas we wish to allow the appearance of several latent variables in an observation vector.

In our experiments, frequent sets and NMF give components with very common motifs, independent components are more rare but form kind of summary components and the PLSA algorithm finds surprising components with long and rare motifs. The 10 components giving the best coverage of motif occurrences in the human upstream regions found by each method have been presented here. With the smaller yeast data PLSA did not do very well, but ICA and NMF found interesting longer components and maximal frequent sets gave a good coverage of data.

We have shown that the observations can be decomposed into meaningful components using the frequent sets and latent variable methods. Also other similar kinds of biological data sets may be analyzed using the solutions given in this paper. If a quick overview of the most common patterns in the data matrix is needed, maximal frequent sets or NMF might be good methods to use. However, if interesting longer patterns should be looked for, ICA and PLSA might be a suitable choice. Or better still, to discover both frequent and surprising components, use all of the methods.

A possible future direction would be to find the most common and often simple data patterns by the method of frequent sets. After that, the "residual" (the part of data not covered by frequent sets) could be analyzed with ICA, PLSA or NMF, giving insight into the detailed structure of the data. Also the clustering of the genes based on their upstream region structures is something to consider. Finding groups of genes with similar patterns in their regulatory regions would give us new insights of the gene expression.

6 Acknowledgments

The authors would like to thank Professor Heikki Mannila for his insights into the topic. We would also like to thank Teemu Hirsimäki for the use of his PLSA algorithm implementation.

References

[1] A. Brazma, I. Jonassen, J. Vilo, and E. Ukkonen. Predicting gene regulatory elements in silico on a genomic scale. *Genome Research*, 8(11):1202–1215, November 1998.

[2] H. J. Bussemaker, H. Li, and E. D. Siggia. Building a dictionary for genomes: Identification of presumptive regulatory sites by statistical analysis. *PNAS*, 97(18):10096–10100, August 2000.

[3] H. J. Bussemaker, H. Li, and E. D. Siggia. Regulatory element detection using correlation with expression. *Nature Genetics*, 27:167–171, February 2001.

[4] J. Charles Lee Isbell and P. Viola. Restructuring sparse high dimensional data for effective retrieval. In *NIPS*, pages 480–486, 1998.

[5] P. Comon. Independent component analysis, a new concept? *Signal Processing*, 36:287–314, 1994.

[6] A. Derti. Upstream sequences for human and mouse Refseq mRNAs. http://arep.med.harvard.edu/labgc/adnan/hsmmupstream/.

[7] E. Eskin and P. A. Pevzner. Finding composite regulatory patterns in DNA sequences. *Bioinformatics*, 18(1):354–363, 2002.

[8] M. Gyllenberg, T. Koski, E. Reilink, and M. Verlaan. Nonuniqueness in probabilistic numerical identification of bacteria. *Journal of Applied Probability*, 31:542–548, 1994.

[9] Helsinki University of Technology: Neural Networks Research Centre. The FastICA Matlab package. http://www.cis.hut.fi/projects/ica/fastica/.

[10] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99*, pages 50–57, Berkeley, CA, 1999.

[11] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42:177–196, 2001.

[12] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.

[13] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9:1483–1492, 1997.

[14] S. Keles, M. van der Laan, and M. B. Eisen. Identification of regulatory elements using a feature selection method. *Bioinformatics*, 18(9):1167–1175, 2002.

[15] M. Koyutürk and A. Grama. PROXIMUS: A framework for analyzing very high-dimensional discrete attributed datasets. In *Proc. Ninth ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD 2003)*, pages 147–156, 2003.

[16] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, October 1999.

[17] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[18] H. Mannila, H. Toivonen, and I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.

[19] V. Matys, E. Fricke, R. Geffers, E. G. ling, M. Heubrock, R. Hehl, K. Hornischer, D. Karas, A. Kel, O. V. Kel-Margoulis, D.-U. Kloos, S. Land, B. Lewicki-Potapov, H. Michael, R. Münch, I. Reuter, S. Rotert, H. Saxel, M. Scheer, S. Thiele, and E. Wingender. TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Research*, 31(1):374–378, 2003.

[20] J. van Helden, B. André, and J. Collado-Vides. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *Journal of Molecular Biology*, 281:827–842, 1998.