# A Computational Method to Identify Splicing Elements

Kian Huat (Eric) Lim

July 2007

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Clustering Analysis

## 1.1 Introduction

Previous studies have demonstrated that enhancer motifs have signature distributions relative to splice signals (need reference). These distributions are informative in several regions across an exon. Exonic Splicing Enhancers (ESEs) tend to be over represented in exons and under represented in introns and both of these properties are useful in their identification. The non-random distribution of short functional DNA sequence elements in genomes is both a way to identify *cis*-elements, and a way to detect synergistic and antagonistic relationship between individual *cis*-elements when they co-occur. Our proposed method assumes that sequences important to splicing will have a signature distribution around splice sites. One could imagine that enhancers will occur with increasing frequency and silencers with decreasing frequency as the distance to the splice site decreases. Some enhancers may be specific to the three prime splice sites (3'ss) or five prime splice sites (5'ss) whereas others may be specific to exons or introns. We propose a method of detecting functional elements based on their skewed distributions relative to splice sites.

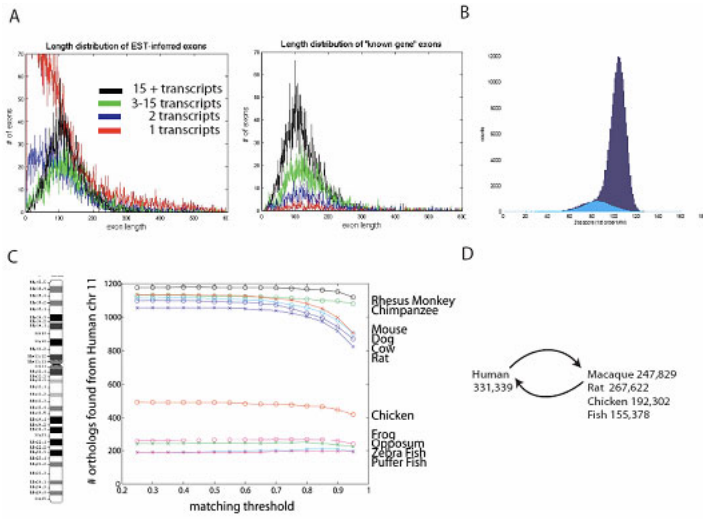



Figure 1.1: Evaluating Annotation Options for Orthologous Exon Dataset A) Length distribution of exons inferred from EST/genomic alignments (left panel) or known genes (right panel) at various confidence levels (red = 1 transcript evidence, blue = 2, green = 3-15, black - 15+). B) 235,500 3' splice scores for EST exons (grey histogram) and 23,500 randomly chosen AG dinucleotides within 200 nucleotides of annotated splice sites (teal histogram). C) Success rate of recovering human exon regions in chromosome 11 using reciprocal best blast hit strategy in eleven different vertebrates. D) Summary of resultant orthologous exon datasets.

## 1.2 Materials and Methods

### 1.2.1 Orthologous Exon Database Identification

Critical to the identification of these motifs and determination of conservationis is the possession of a reliable set of exon annotations and with orthologs in multiple species. This type of annotation is often performed by some combination of *ab initio* gene prediction and transcript support (Hsu, Kent *et. al* 2006). Gene prediction programs use splice sites and sequence composition to determine exon-intron junctions. However, this approach selectively filters splicing substrates with poor splice sites or unusual sequence composition from the database, thereby creating an undesirable bias.

An exon/intron dataset was made from dbEST Hg17 alignments stored at the UCSC *ftp* site (Karolchik, Hinrichs *et. al.* 2004). Relative to normal length distribution of human-curated genes (Figure 1.1A, right) EST defined genes are

enriched for short aligned blocks (Figure 1.1A, left). As these short blocks are greatly reduced with increased transcript evidence, they are likely alignment artifacts. To test this 235493 EST genomic alignment blocks bounded by an AG at the 5' end (i.e. the most conserved part of the 3'ss) were score for agreement to the 3'ss motif using a first order Markov model. Figure 1.1B shows two distinct populations in the slate blue histogram: a high scoring peak (roughly 90 percents of all counts) and a broad low scoring tail (10 percents). Sampling random AG positions within these alignment regions and scoring them for splice site strength result in a similar broad distribution. This suggests that 10 percents of these EST genomic alignments describe exon models within splice sites so weak as to be indistinguishable from background - probably due to the result of misalignment. To counter this, we required three ESTs to define an exon and also included the alignment coordinates of known gene transcripts in the annotation.

These genomic coordinates have been translated across species with the UCSC program *liftover* which parses the output of the multiple alignment tool *blastz* and returns orthologous coordinates. Orthologs are identified by the reciprocal best hit strategy where the highest scoring match to a human exon in a second species is deemed the ortholog if its best match in the reverse direction is the original human exon. Preliminary analysis was performed matching exon from chromosome 11 using a range of identity thresholds to the genomes of eleven vertebrates. This analysis demonstrates that the probability of identifying a vertebrate ortholog is roughly proportional to its phylogenetic distance from human. Noticeable exceptions occur with scaffold based assemblies which are more likely to result in punctuated assemblies that perform poorly in alignments optimized for large syntenic blocks.

### 1.2.2 Clustering Procedure

Any distribution pattern (profile) from a collection of hexamers is represented by a feature vector. Therefore this collection of patterns can be represented by a finite subset $X = \{\vec{x}^1, \vec{x}^2, \ldots, \vec{x}^n\}$ of the feature space. For comparing two profiles $\vec{x}^1$ and $\vec{x}^2$, we use the *euclidean* distance, $d(\vec{x}^1, \vec{x}^2)$, which is simply defined as

$$d(\vec{x}^1, \vec{x}^2) = ||\vec{x}^1 - \vec{x}^2|| \equiv \sqrt{\sum_{i=1}^{d} (x_i^1 - x_i^2)^2}, \tag{1.1}$$

where $d$ is the dimension of the feature space.

To group profiles with similar distribution pattern, we employ DIvisive ANAlysis (*diana*) (Kaufman and Rousseeuw 1990), a top-down divisive method that has performed well in other contexts (Datta and Datta 2003; Ding and Lawrence 2005). To determine the correct number of clusters, we use the CH index (Calinski and Harabasz 1974).

#### Counting Profiles

Each hexamer (word) has a feature vector corresponding to its occurences in positions relative to splice sites. Each of the vector has precisely 602 values which describes the occurences of the word being observed at a particular location. Furthermore, we divive these 602 positions into three distinct regions (Figure 1.2).

If a word is observed, its corresponding position in the feature vector will be updated. This procedure is straightforward and is applied to every sequence in the exon database describe above. To avoid over counting words with simple repeat, a simple forwarding strategy is implemented such that if a word is observed in position $x$, the same word cannot be observed again in any of the

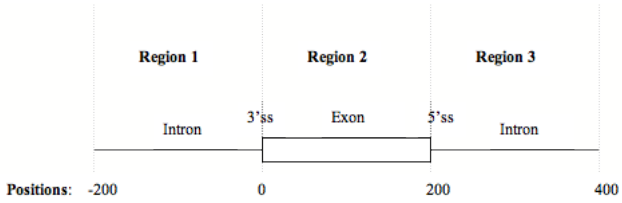



Figure 1.2: Three separate regions of the feature vector.

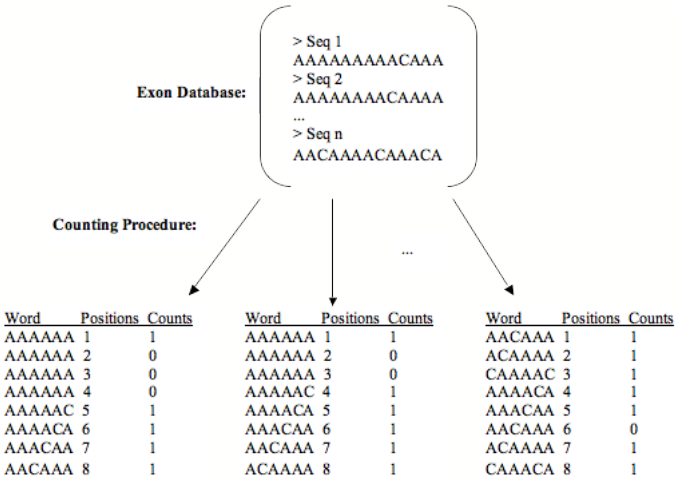



Figure 1.3: A simple example of how the counting and forwarding procedures are performed. Given an exon database, the counting procedure is performed on each sequence in the database. The results are shown for each word observed in the sequence and their positions are recorded. If the same word within the forwarding window is observed, a count of zero will be assigned. After scanning through every sequence in the database, each word will receive a feature vector highlights the number of occurences of each position.

$x+5$ positions. This process is illustrated in Figure 1.3. After scanning through the entire database, each feature vector highlights the number of occurences of each word relative to splice sites.

### Normalizing Profiles

Given a feature vector $\vec{x}$, we normalize $\vec{x}$ such that $\forall x \in \vec{x}$, $x' = \frac{x-\mu}{\sigma}$, where $\mu$ and $\sigma$ are the mean and standard deviation of $\vec{x}$.

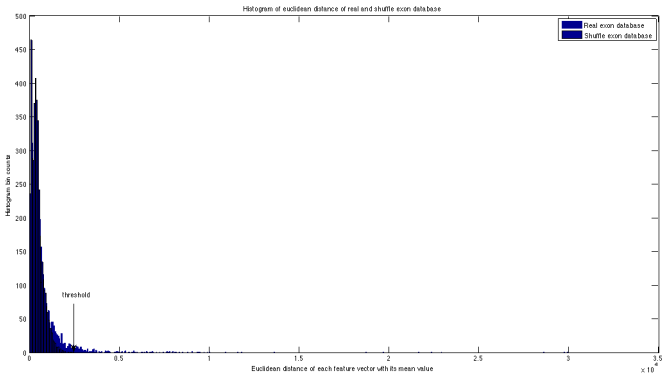



Figure 1.4: Euclidean distance distributions of two sets of feature vectors: the blue historgram is for the real database and the blue histogram with black border is for the shuffle database. A cut-off threshold is chosen by using the largest euclidean distance from the shuffle database.

### Filtering Profiles

To improve the resolution of the clustering results, we remove feature vectors which are likely to be obtained by chance alone. In the previous section, all 4096 feature vectors are created by applying the counting procedure on the exon database (real database) described in Section 1.2.1. Here, the same counting procedure is performed but on a randomly shuffled exon database (random database) generated by the Altschul-Erikson dinucleotide shuffle program (Altschul and Erikson 1985). This results in exactly 8192 feature vectors; 4096 from the real database and 4096 from the random database. Then the *euclidean* distance calculation (Eq 1.1) is performed on each of the feature vectors with its mean value to determine how much the distribution pattern deviates from uniformity. Two euclidean distance distributions are plotted in histogram format (Figure 1.4). To determine words of interests (words that exhibit the significant deviation from uniformity), a threshold $t$ is chosen by using the largest euclidean distance from the shuffle database. This threshold is very stringent and only leaves us with 194 words. Therefore we relax the threshold by 10 percents (by allowing 10 percents of potential false positive) so that $t_{relax} = 914$ which leaves us with 905 words.

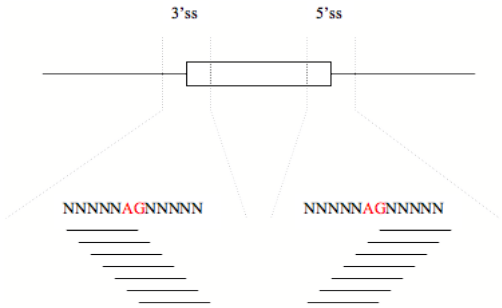



Figure 1.5: A masking and collapsing strategy around splice sites regions. 7 positions around each the splice site regions are either masked (removing) or collapsed (taking the maximum count within the 7 positions).

#### Masking and Collapsing Splice Sites

The exon database is prepared by aligning alignment blocks bounded by an AG around the splice sites (ie. the most conserved part of the splice sites) (Figure 1.5). Due to this reason, hexamers which contain AG at various starting points will have peaks around the splice sites regions in their distribution patterns. This can potentially cause our clustering method to put these slight variations into multiple clusters. To resolve this, we tried two approaches: 1) masking by removing a window of 7 positions around each of the the splice site regions and 2) collapsing each of the splice site regions into one position by taking the position with the maximum count within the window.

### 1.2.3 DIvisive ANAlysis Clustering

DIvisive ANAlysis (*diana*) is a hierarchical clustering technique in which it constructs the hierarchy in the inverse order. Initially, there is one large cluster consisting $n$ profiles, where $n$ is the number of words and SELEX profiles. At each subsequent step, the largest available cluster is split into two clusters until finally all clusters, comprise of single profile.

If considered all possible fusions of two profiles in the agglomerative method, there are $\frac{n(n-1)}{2}$ combinations. In the divisive method there are $2^{n-1}-1$ pos-

siblities. This number is considerably larger than that in the case of the agglomerative method. To avoid considering all possible fusions, the algorithm proceeds as follow:

1. Find a profile that has the highest average dissimilarity to all other profiles and initiate it as a new cluster - called *splitter group*
2. For each profile $i$ outside the *splitter group*
3. Calculate $D_i = [average\, d(i,j)\ for\ j \notin R_{splitter\ group}] - [average\, d(i,j)\ for\ j \in R_{splitter\ group}]$
4. Find a profile $h$ for which the difference $D_h$ is the largest. If $D_h$ is positive, then $h$ is, over average close to the splitter group.
5. Repeat *Step 2* and *3* until all differences $D_h$ are negative. The dataset is then split into two clusters
6. Select the cluster with the largest diameter. The diameter of a cluster is the largest dissimilarity between any two of its profiles. Then divide the cluster, repeat *Step 1* to *4*
7. Repeat *Step 5* untill clusters contain only a single profile

### 1.2.4 Validity Indices

There are no completely satisfactory methods for determining the number of classifications for any type of cluster analysis (Everitt 1979, 1980; Hartigan 1985; Bock 1985). We have investigated multiple validity indices proposed in the literature: Calisnki-Harabasz (CH) index (Calinski and Harabasz 1974), Simply Structure index (SSI), Dunn index (DI), and the Davies-Bouldin (DB) index. From multiple trials of evaluations and human observations, CH index returns the most promising clustering structure produced by *Diana* algorithm. CH index has also been assessed as the best in a comprehensive study (Milligan and Cooper 1985). Therefore we employ CH index to be the validity index used in

this study. CH index is a quantity describing the degree of inter-cluster separation and intra-cluster homogeneity. Clearly a good clustering result should minimize the dispersion within clusters and maximize separation between clusters. For a given divisive level on the clustering tree from *Diana*, CH index is calculated as

$$CH(K) = [B(K)/(K-1)]/[W(K)/(N-K)], \tag{1.2}$$

where $K$ is the number of clusters, $N$ is the total number of feature vectors, $W(K)$ is the within-cluster sum of squares, and $B(K)$ is the between-cluster sum of squares. The within- and between-cluster sum of squares can be written, respetively, as

$$W(K) = \sum_{j=1}^{K} \sum_{\vec{x}^i \in C_j} ||\vec{x}^i - \vec{m}^j||^2$$

$$B(K) = \sum_{j=1}^{K} ||\vec{m}^j - \vec{m}||^2$$

where $\vec{m}$ is the cluster center of the entire dataset. The cluster centers required for the sum of squares are computed by simply calculating the mean values of all the profiles in a cluster.

## 1.3 Results

From previous section, CH index is a quantity describing the degree of inter-cluster separation and intra-cluster homogeneity. CH index calculation is performed over a range of different numbers of clusters ($2 < k < 100$) and the results are shown in Figure 1.6.

Since there isn't yet a satisfactory way to determine the optimal $k$, we chose an arbitrary $k$ by manually judge the best local maxima of the CH index scores.

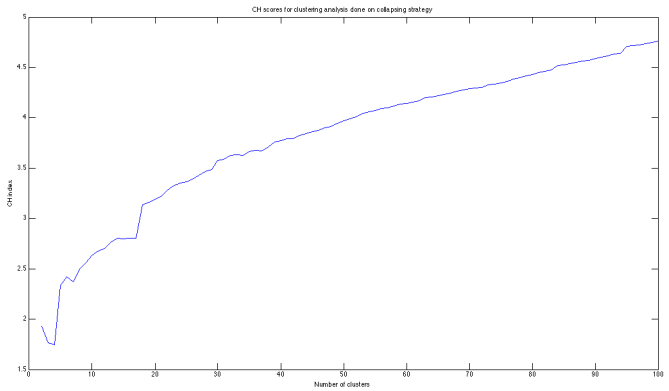



Figure 1.6: Over different number of clusters $k$ CH index is computed by calculating the ratio between inter-cluster separation and intra-cluster homogeneity explained in Eq 1.2. We reason the optimal $k$ is at which the CH index is maximized. Clearly from the CH index distribution shown in this figure it is an increasing function as the number of clusters increases. Other approaches will be introduced such that the CH index is more informative.

For clustering results done using the collapsing strategy, we found that $k = 18$ might be one of the most representative clusters (Figure 1.6). The clustering results are shown in circular dendrogram along with the corresponding frequency distribution patterns of each cluster and the pictograms that represent all words in the same cluster (Figure 1.7.

Beside from collapsing the splice site regions, we have also attempted to cluster words by masking the splice site regions. The CH scores and clustering results of the masking version is shown in Figure??.

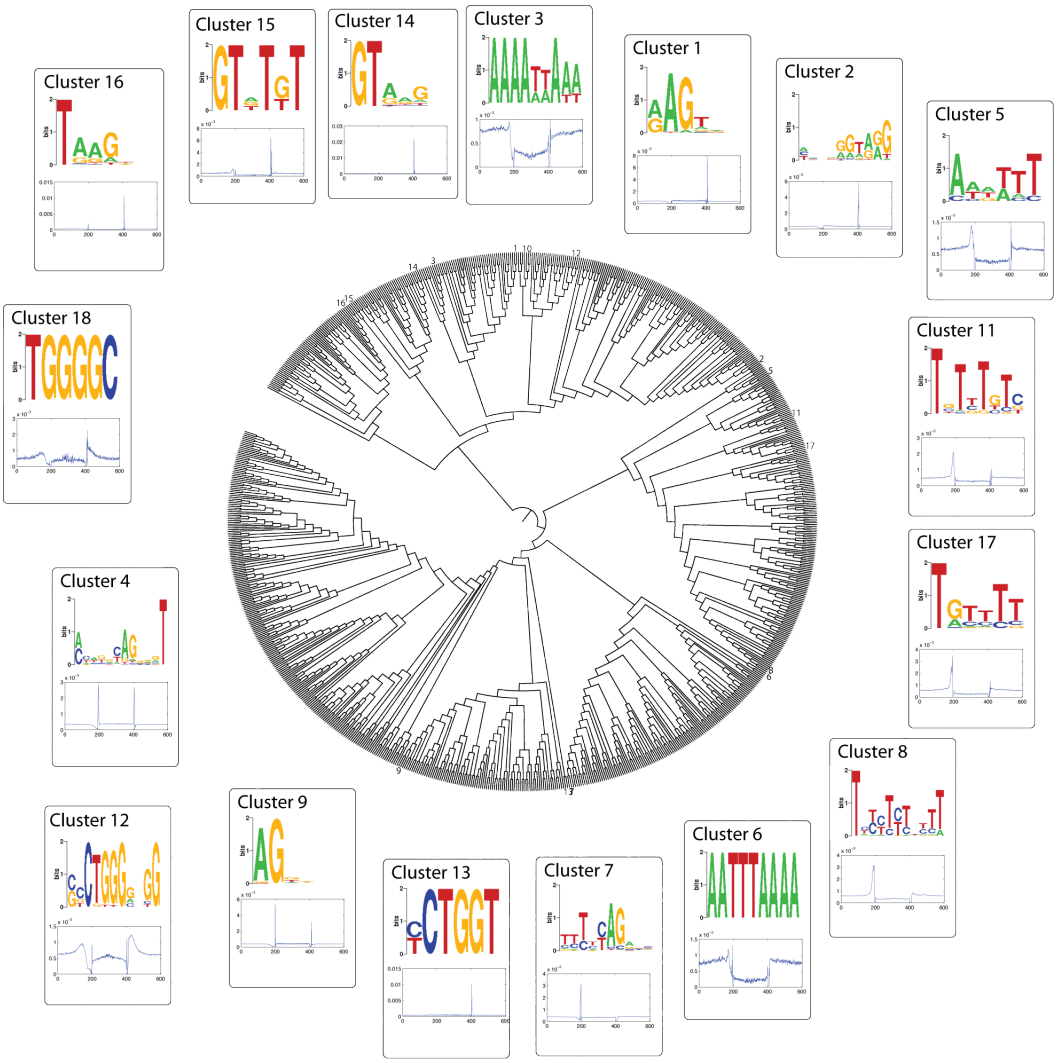



Figure 1.7: Circular dendogram display of the clustering results. Feature vectors are normalized and collapsed as described in the clustering procedure section. For each cluster box, the top figure is the pictogram that represents all the words in the same cluster. These words are first aligned by using *clustalw* with standard option and not allowing gap. The aligned sequences are then used to generate the pictogram logo. The bottom figure is the average frequency distribution pattern of all the words in the same cluster. NOTE: The image is missing connectors that connect each cluster box to their corresponding leaves in the dendogram. Because generating the image requires huge effort of manual work, I've decided to include a simpler image for illustrations and suggestions before getting the final version completed. For a full version of the clustering results, see Appendix A.

# Chapter 2

# SELEX

## 2.1 Introduction

The underlying assumption of our computational approach to finding signals is that signals that splicing factors recognize will have a non-uniform distribution relative to splice sites. Computationally predicted enhancers certainly have non-uniform distribution relative to splice sites but as they are frequently identified by their enrichment in exons. Despite the fact that many of the RESCUE-ESE were validated experimentally, it is important to demonstrate the importance of distribution via an independent method.

## 2.2 Materials and Methods

The binding specificity can be determined by the Systematic Evolution of Ligands by Exponential Enrichment (SELEX) experiments for several RNA binding proteins (Table 2.1). The output of a SELEX experiment is typically 10 to 40 sequences (SELEX sequences) that have been identified via iterative selection protocol as high affinity binding sites for the factor. From these SELEX sequences, motifs are derived using a Gibbs sampling strategy (Thompson and Lawrence 2003). To determine the length of these motifs for different RNA

binding proteins, we examine the maximum *a posteriori* probability (MAP) of the alignment given by Gibbs sampling. The MAP value is measured relative to an empty of *null* alignment, by taking the difference between the log of the probability of the alignment and the log of the probability of an empty alignment. We reason the motif is the most informative with length at which the MAP value is the highest.

Table 2.1: Binding specifications of known splicing factors

| Name | Function | Reference |
|---|---|---|
| SRp40 | ESE | (Tacke, Chen *et. al.* 1997) |
| Tra2 | ESE | (Tacke, Tohyama *et. al.* 1998) |
| ASF/SF2 | ESE,ISS,5'ss | (Tacke and Manley 1995) |
| SC35 | ESE | (Tacke and Manley 1995) |
| SRp20 | ESE | (Cavaloc, Bourgeois *et. al.* 1999) |
| 9g8 | ESE | (Cavaloc, Bourgeois *et. al.* 1999) |
| hnRNP L | ISE,ISS | (Hui, Hung *et. al.* 1994) |
| hnRNP C | ISE,PPT | (Gorlach, Burd *et. al.* 1994) |
| hnRNP A1 | ESS,ISE | (Singh, Valcarcel *et. al.* 1995) |

These motifs are then searched for matches in the exon database using the *patser* program (Hertz and Stormo 1999). From *patser*, each window of the sequences in the exon database will receive a matching score. We select a minimum matching score (threshold) for each motif so that for those windows that receive less than the threshold will not be reported by *patser*. Then we perform the counting procedure on these final matching windows according to their positions relative to splice sites. Applying *patser* with various threshold constraints leads to different distribution patterns. Because we are interested in the genomic distribution of these RNA binding proteins, we reason the optimal threshold to be the threshold in which it results the most skewed distribution pattern (most deviated from uniformity). Therefore, from a range of thresholds, we calculate the *manhattan* distance (Eq 2.1) region by region (1.2) of the distribution pattern, $\vec{x}$ with its mean value, $\mu$.

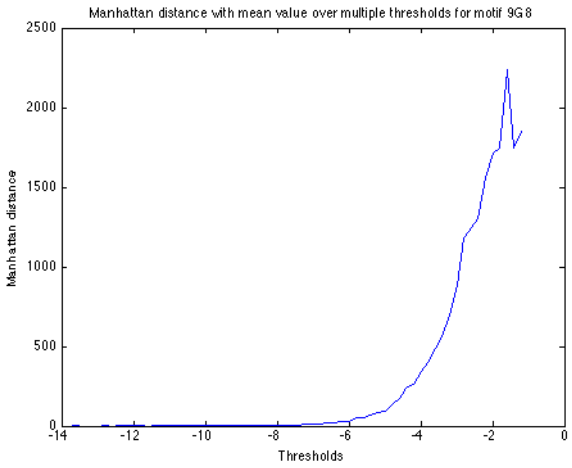



Figure 2.1: Manhattan distance is calculation from the 9G8 feature vector and its mean value.

$$d(\vec{x}, \mu) = \sum_{x \in \vec{x}} ||x - \mu||, \tag{2.1}$$

Our first intuition was that the threshold in which $d(\vec{x}, \mu)$ is the highest should result the most skewed distribution pattern. However, decreasing threshold leads to the increase of the skew and the magnitude of the frequency of the distribution patterns, which also results in greater *manhattan* distance from the mean. Therefore, calculating Eq 2.1 is not informative as it is always an increasing function as the threshold decreases (ie. larger threshold is always preffered) (Figure 2.1). We have also tried to normalized $\vec{x}$; however, the calculation tends to prefer lower threshold (Figure 2.2. Note that because of all SELEX motifs exhibit similar results from the Manhattan distance calculation, we only use the 9G8 SELEX motif for illustration.

Another approach to finding the optimal threshold is by sampling equal amounts of counts on the distribution patterns resulted from different thresholds. This is a sampling with replacement strategy to ensure that each distribution pattern of interests has the exact same mean value. Given a fix number of counts $c$, a random number generator picks a position according to the proba-

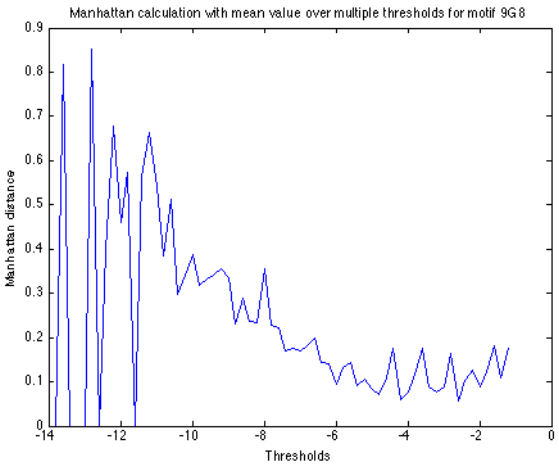



Figure 2.2: Manhattan distance is calculation from the normalized feature vector and its mean value.

bility of observing a match of the motif in that position, then the feature vector for the sampled distribution is updated accordingly. The same *manhattan* distance described above is calculated. Again, we are interested in the highest *manhattan* distance as it is an indication of a most skewed distribution pattern. However, the results turn out to be an decreasing function. In other words, this approach tends to favor the distribution patterns resulted from low thresholds (See Figure 2.3 blue plotted line).

Our last approach is by shuffling the motif column-by-column. Given the position weight matrix (PWM) of the motif, we shuffle the nucleotides correspond to each of the column of the PWM. We then ran *patser* on the shuffled PWM and follow by the *manhattan* distance calculation. The reason of trying this approach is that we expect more uniform distribution from the shuffled PWM, for which we can combine it with the sampled distribution to derive the optimal threshold. However, the result tends to also favor low thresholds (See Figure 2.3 red plotted line). Also, one problem can be quickly noticed is that it's very likely to randomly assign a duplet of AG in the motif. Due to the way how we prepare our exon database (Section 1.2.1) by aligning the most conserved part

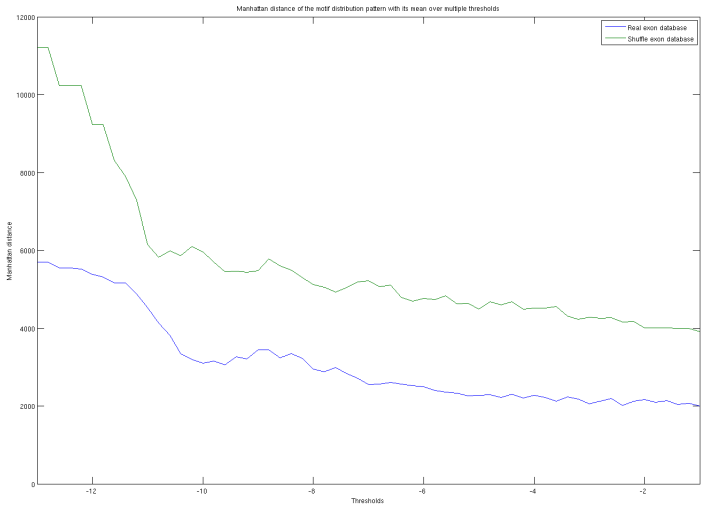



Figure 2.3: Manhattan distance is calculated on two sets of analysis. One by sampling equal amounts of counts on the distribution pattern results from different threshold (Blue). The other is by shuffling the motif column-by-column. Notice that the motif shuffling strategy produces a higher Manhattan distance across all thresholds represent that it is likely to randomly shuffle the motif and replace a dinucleotides of AG side-by-side. Due to the way how we align our exon database, the dinucleotides of AG will produce a peak around the splice sites region, which leads to higher Manhattan distance.

of the splicing juntions, any motif with a duplet of AG will most likely produce a spike around the splice juntions. This will, in fact, increase the *manhattan* distance of the overall distribution pattern (See Figure 2.3).

Because the problem of picking an optimal threshold is still an ongoing work, we have decided to use -6 as the threshold for all the distribution patterns showed in the Results section. This threshold is selected mainly from human observations and prior knowledge. However, as it can be easily noticed from Table 2.2 and Table 2.3 in the Results section, some of the SELEX motifs and distribution patterns are not optimized at the given fixed threshold. For example, hnRNP L should be further relaxed such that the frequency of the distribution pattern is higher. Therefore, it is crucial that we can use prior knowledge in the binding affinity of these proteins to determine the optimal threshold for each of them.

## 2.3 Results

### 2.3.1 U2AF and hnRNP C recognize the splice sites

U2AF is composed of a 65kd and 35kd subunit that plays an early role in 3'ss recognition (Ruskin, Zamore 1988). As depicted in Figure ?? U2AD65 recognizes the polyprymidine tract (PPT) - a short stretch of C's and U's located just upstream of the 3'ss. Although less is known about hnRNP C, this factor has been implicated as an intronic enhancer that also recognizes the 3'ss end of the intron (Swanson and Dreyfuss 1988). The distribution of the binding sites for these factors is consistent with their known function. Both U2AF and hnRNP C binding sites are heavily biased towards the 3' end of the intron (See Figure ??).

### 2.3.2 SR proteins predominantly binds in the exon

SR proteins are splicing activators that often function at exonic locations. Examining the genomic distribution of these sequences confirm this notion that splicing elements that are bound by SR proteins are typically located in exons (Figure ??). In addition to an exonic bias the distribution tends to be greater when it is closer to the splice site than in the internal portion of the exon.

### 2.3.3 hnRNP proteins predominantly bind in the intron

hnRNPs are generally regarded as non-specific RNA binding proteins. The genomic distribution of hnRNPs is slightly biased towards introns in most cases. PTB has been cited in a wide variety of inhibitory roles in both introns and exons in splicing. Also, hnRNP A1 has been found to modulate splicing in both exons and introns. hnRNP L has been demonstrated to function only in introns by binding CA repeats (Hui, Hung 2005).

Table 2.2: Distribution patterns for known splicing elements. The most left columns (splicing element) is the known SELEX results in interests. The middle column (Motif) is the motif generated by *Gibbs* sampler using published SELEX data. The most right column (Genomic Distribution) is the *patser* matches against the exon database using a fixed threshold described above.

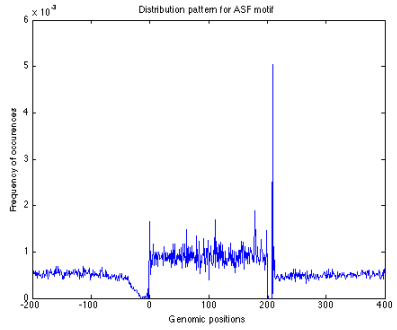
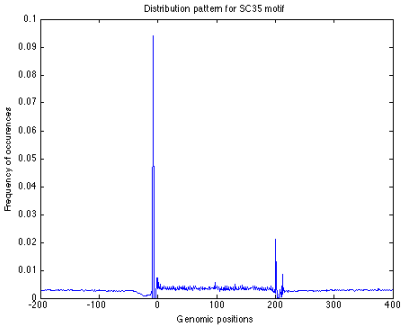
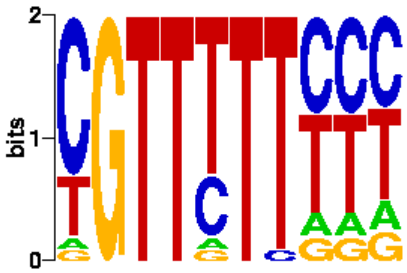
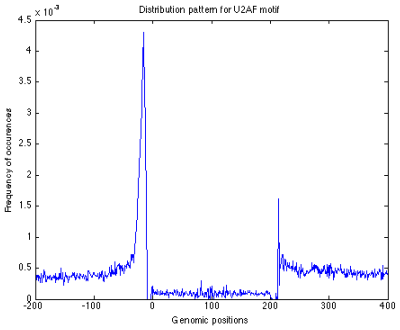
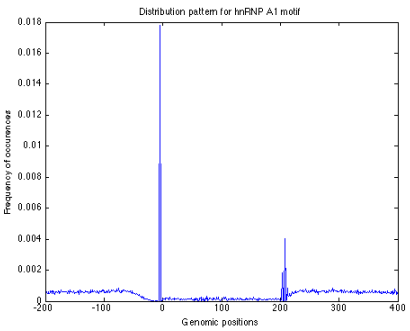
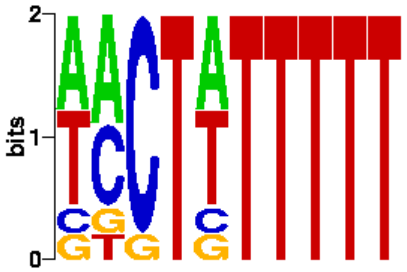
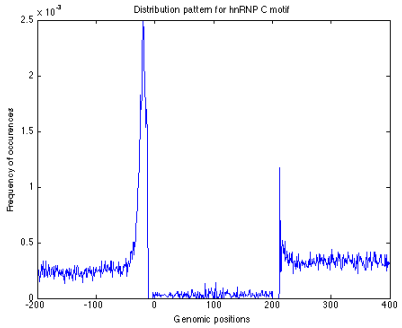
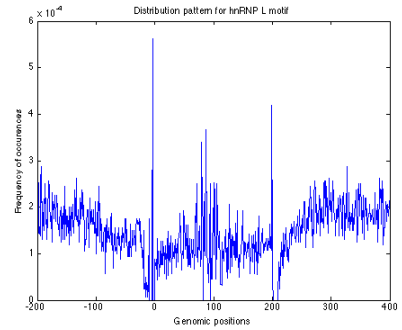
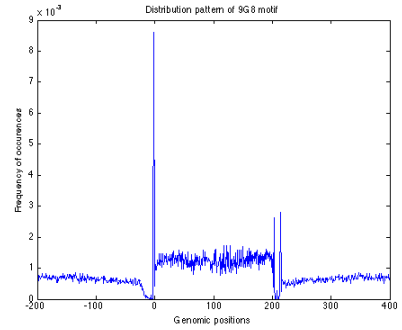
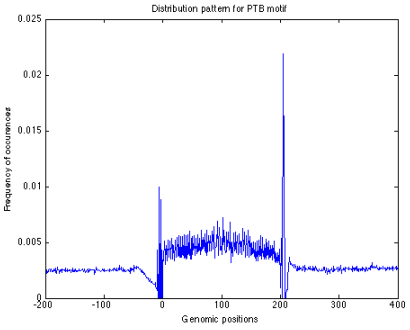
| Splicing Elements | Motif | Genomic Distribution |
|---|---|---|
| ASF |  |   |
| SC35 |  |   |
| U2AF |  |   |
| hnRNP A1 |  |   |

Table 2.3: Distribution patterns for known splicing elements. The most left columns (splicing element) is the known SELEX results in interests. The middle column (Motif) is the motif generated by *Gibbs* sampler using published SELEX data. The most right column (Genomic Distribution) is the *patser* matches against the exon database using a fixed threshold described above.

| Splicing Elements | Motif | Genomic Distribution |
|---|---|---|
| hnRNP C |  |   |
| hnRNP L |  |   |
| 9G8 |  |   |
| PTB |  |   |

# Chapter 3

# Cooperativity

To explore the repetitive nature of local RNA structure, we extended our approach by capturing the double occurences of each hexamer in the exon database described above. From the same exon sequence, a word is *doubly occured* if and only if an aleady observed word $w_i$ at position $p_k$ is observed again at another position $p_l$ such that $p_l \neq p_k$ and $p_l$ is outside of the forwarding window from $p_k$. Let $c(w_i)$ be the total number of occurences of word $w_i$ in the exon database, $t$ be the total number of all possible windows of hexamers from the exom database, and $k$ be the number of exon sequences that contain at least one $w_i$, the probability of observing word $w_i$ in the database is $Pr(w_i) = c(w_i)/t$ and the conditional probability of observing the second or more $w_i$ given the first occurence of $w_i$ is $Pr(w_i|w_i) = (c(w_i) - k)/t_{w_i}$ where $t_{w_i}$ is the total number of all possible windows of hexamers from the every exon database sequence that contains at least one $w_i$.

## 3.1 Results

The histogram of $Pr(w_i)$ and $Pr(w_i|w_i)$ for all hexamers are shown in Figure 3.1 and 3.2 respectively. The histogram of the log ratio of the above-mentionned probabilities are also shown in Figure 3.3.

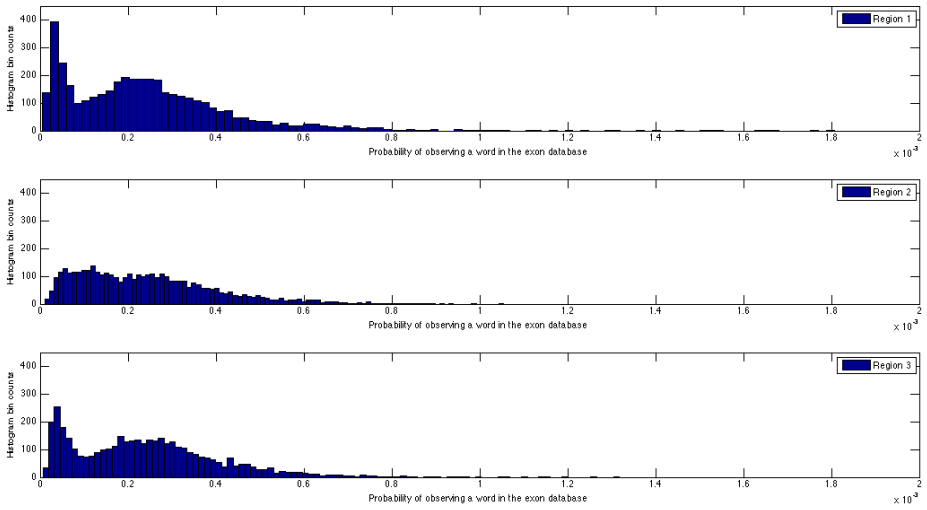



Figure 3.1: Histogram (100 bins) of the probability of observing at least one word in the exon database is shown. The calculation of the $Pr(w_i)$ is done region by region and each of the histogram represents an analysis from an unique region. The top figure represents the first region; the middle represents the second region; and the last figure represents the third region.

In addition to the analysis explained above, we compared the log ratio of the probabilities with the skewness (deviation from uniformity) of the distribution patterns of words. Similarily, the frequency distribution pattern profile is generated region by region using the counting procedure described in the Clustering section, and the *manhattan distance* is calculated from each of the patterns with its mean value. A scatter plot of the analysis is shown in Figure ??.

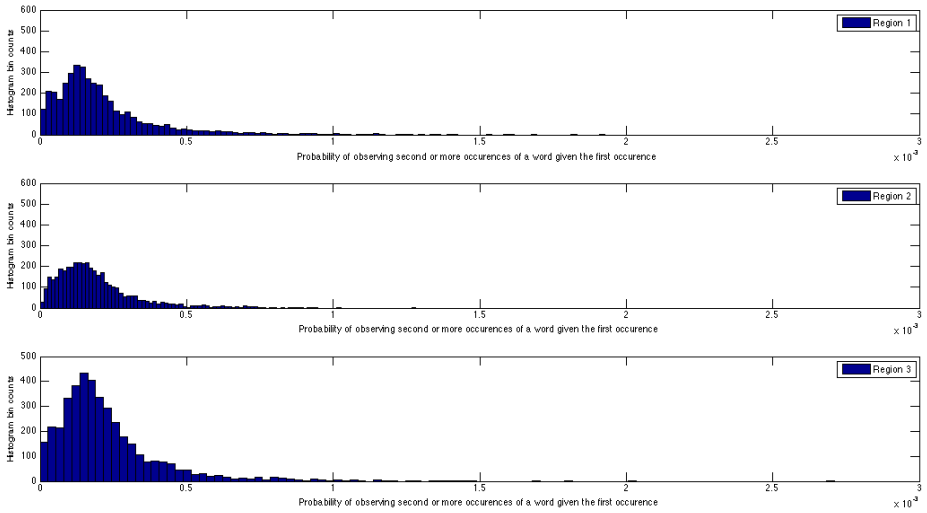



Figure 3.2: Histogram (100 bins) of the conditional probability of observing the second or more occurences of a word given the first occurence in the exon database is shown. The calculation of the $Pr(w_i|w_i)$ is done region by region and each of the histogram represents an analysis from an unique region.

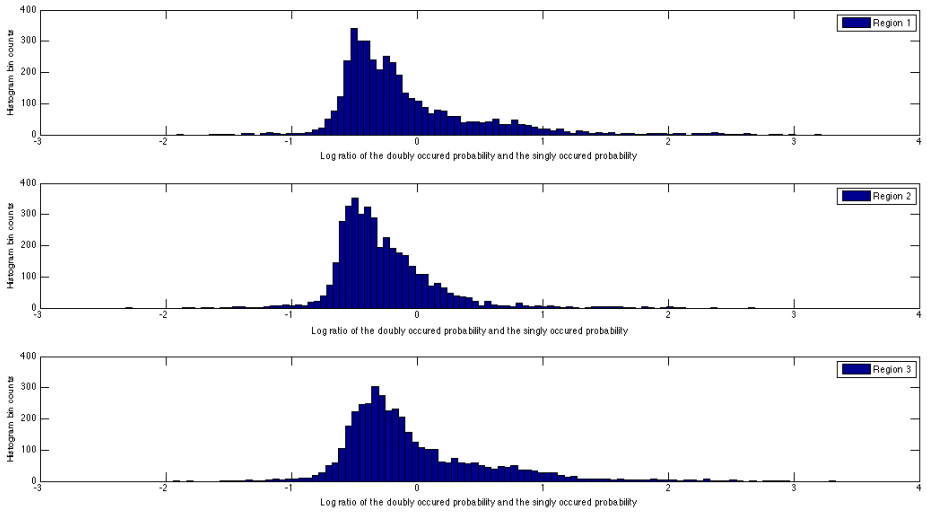



Figure 3.3: Histogram (100 bins) of the log ratio of $Pr(w_i)$ and $Pr(w_i|w_i)$ is shown.

# Chapter 4

# Cross-Species Analysis

In addition to clustering words between within a species, the distribution of each hexamer was compared in different species. This analysis was performed on species that have a clear ortholog to the human. Using these ortholog tables with each of the 4096 hexamers, the human distribution pattern was compared to the pattern observed in mouse, rat, fish, chicken, cow, and dog. To eliminate bias of various sample sizes, the exon database of each species are first sampled to the equal amount of sequences according to their orthologous coordinates to human. Then, the same counting procedures are performed on these databases region by region. To measure the difference in patterns from two species, we use *Manhattan* distance. That is, we calculate *Manhattan* distance from the feature vector of a word observed in human with the feature vector of the same word observed in another species. We are interested in the comparison in which it results the highest *Manhattan* distance. For full set of analysis between each species to human, see Appendix B.

Between fish and human there appears to be a huge chance in the importance of two simple repeats: GTGTGT and CACACA. The former peaks is enriched in the vicinity of the 3'ss and the later is enriched in the vicinity of the 5'ss. The sequence suggests a secondary structure element that spans either the exon or the intron in fish genes but not in human genes. An examination of the

distribution of these elements suggests that span the introns of fish genes. Folding the pre-mRNA suggests a mechanism that brings the 3'ss and 5'ss in close proximity - perhaps facilitating the correct splice site pairing across the intron. The relative distribution of the two words is consistent with the dependency that would be expected between the two arms of an RNA hairpin. Introns that contain a CACACA element in the 5ss intronic region (region 3) are 2.5 fold more likely to contain a GTGTGT element across the intron in region 1 of the next exon.

# Chapter 5

# Appendix A

## 5.1 Clustering results from collapsed feature vectors

This section shows the clustering results done using the collapsing strategy (explained in Clustering section). Because determining the optimal $k$ for *diana* algorithm is still an ongoing work, we have chosen $k = 18$ based on some local maxima of the CH index scores. Each figure in this section contains of four plots. The up right plot shows the frequency distribution patterns of all words belong to the cluster. The up left plot shows the frequency distribution patterns of all words belong to the cluster with the exception that the y-axis is scaled to $3x10^{-}3$. Scaling to y-axis enables us to have a better resolution of the pattern outside of the splicing regions. The bottom right plot shows the average frequency distribution patterns of all the words in the cluster. This is done by taking the average of all the frequency distribution patterns, position by positions. The bottom right plot shows the same average frequency distribution, but the y-axis is scaled to $3x10^{-}3$.

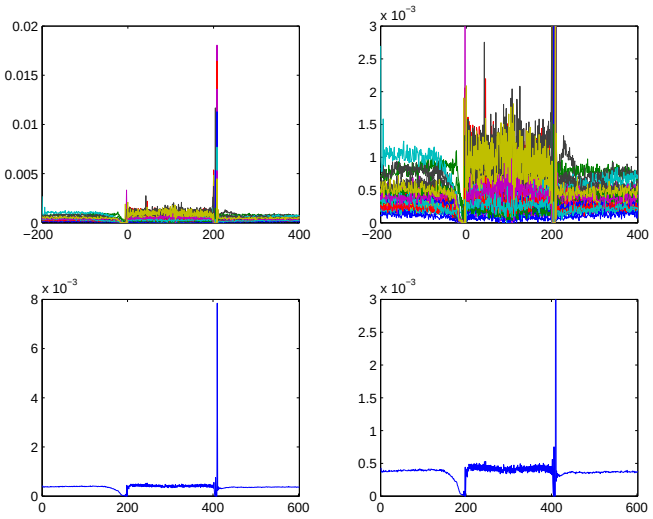



Figure 5.1: Collapsing Clustering Results: Cluster 1

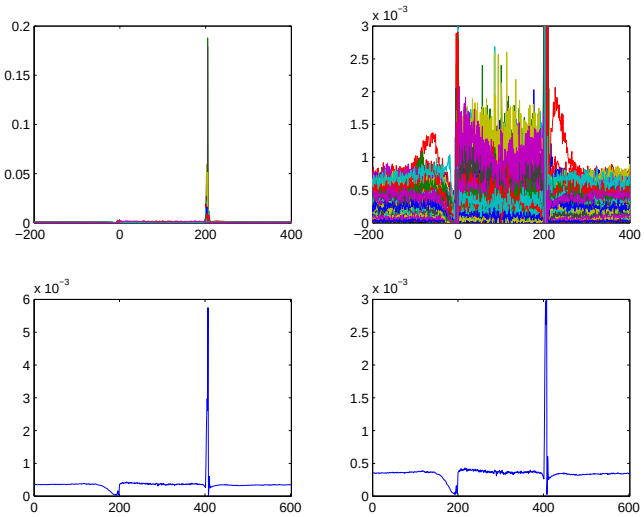



Figure 5.2: Collapsing Clustering Results: Cluster 2

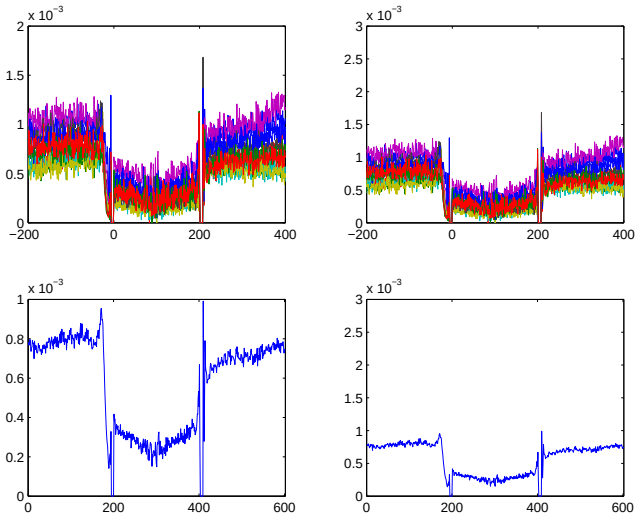



Figure 5.3: Collapsing Clustering Results: Cluster 3

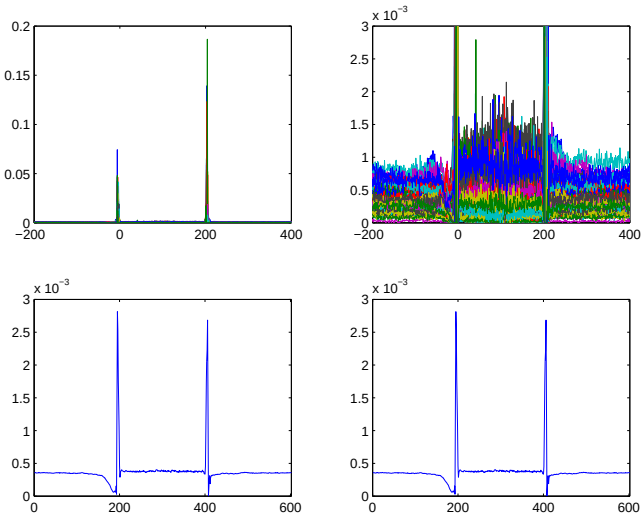



Figure 5.4: Collapsing Clustering Results: Cluster 4

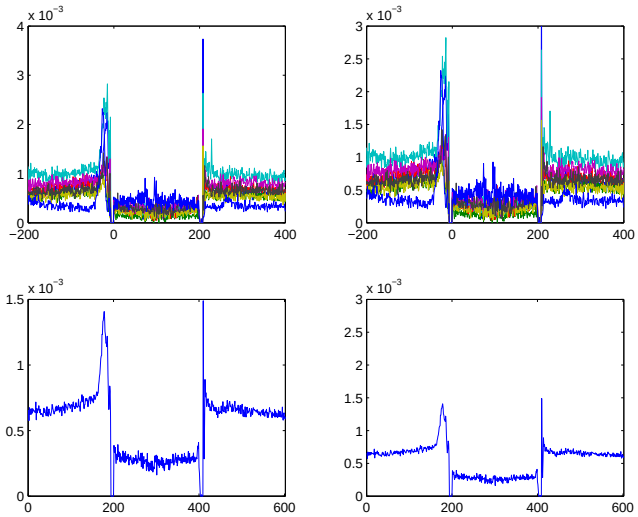



Figure 5.5: Collapsing Clustering Results: Cluster 5

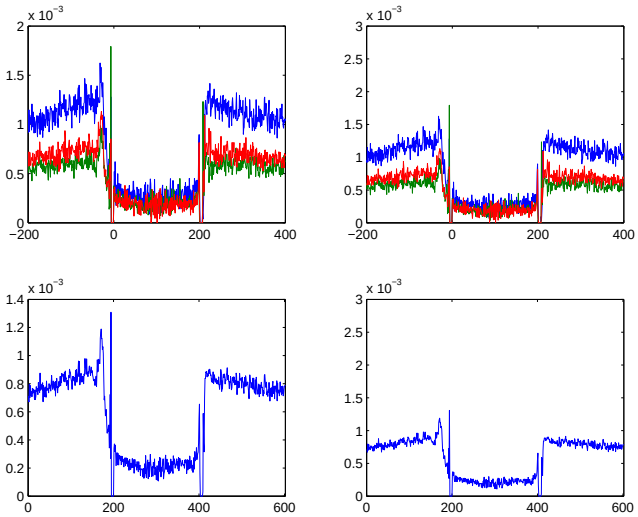



Figure 5.6: Collapsing Clustering Results: Cluster 6

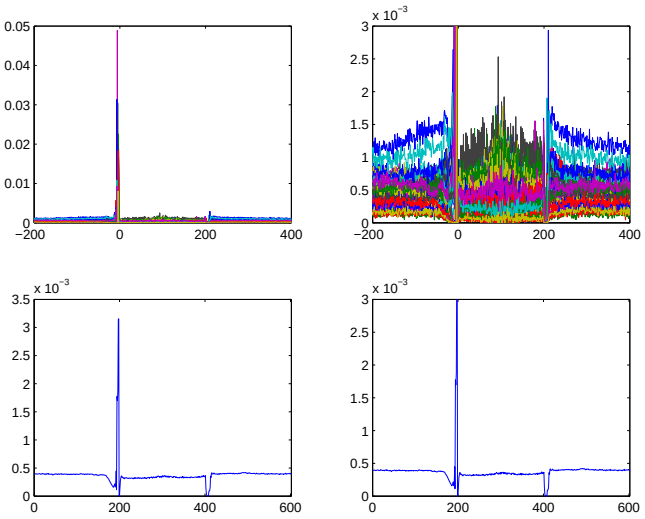



Figure 5.7: Collapsing Clustering Results: Cluster 7

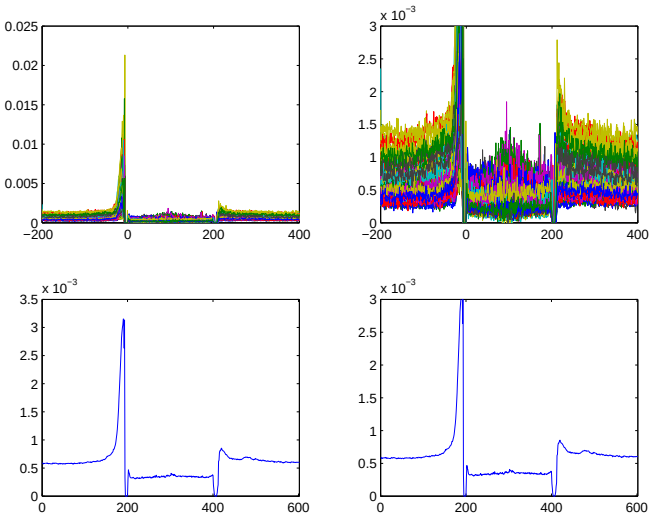



Figure 5.8: Collapsing Clustering Results: Cluster 8

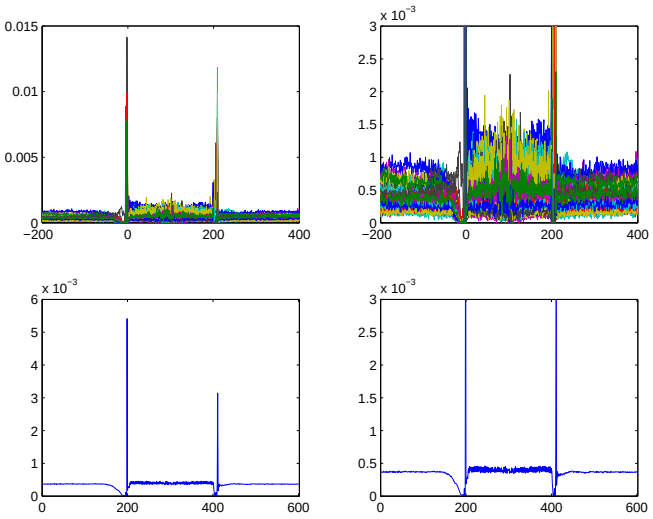



Figure 5.9: Collapsing Clustering Results: Cluster 9

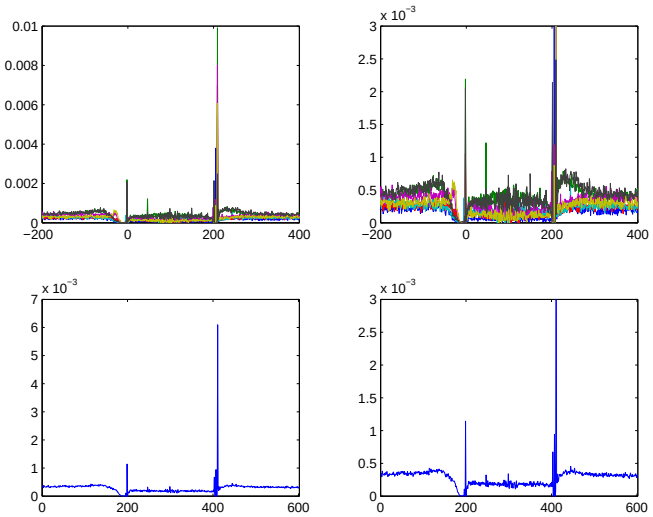



Figure 5.10: Collapsing Clustering Results: Cluster 10

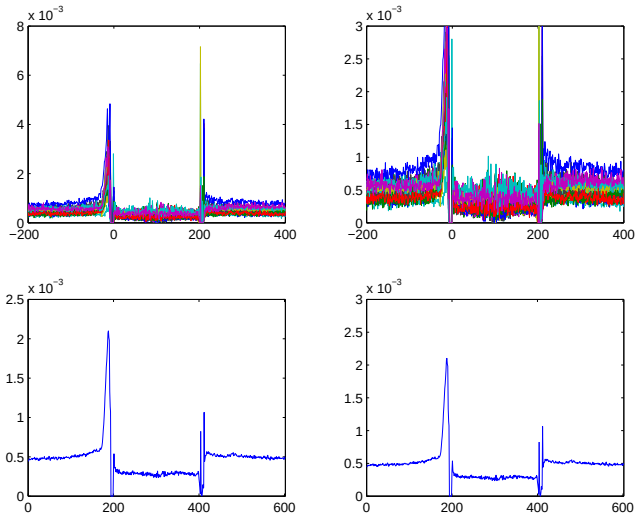



Figure 5.11: Collapsing Clustering Results: Cluster 11

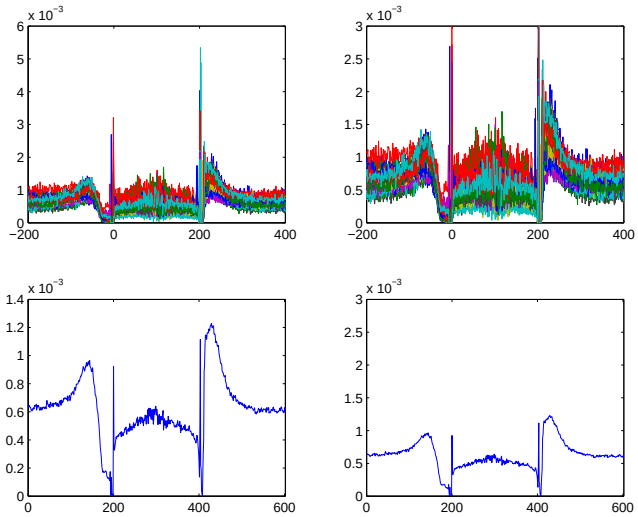



Figure 5.12: Collapsing Clustering Results: Cluster 12

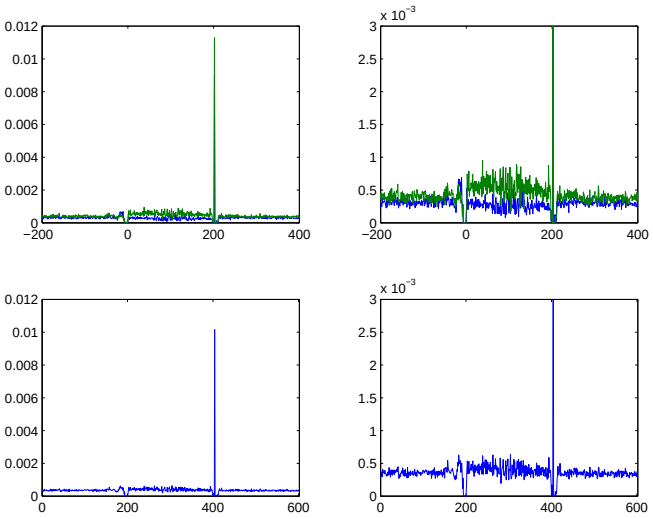



Figure 5.13: Collapsing Clustering Results: Cluster 13

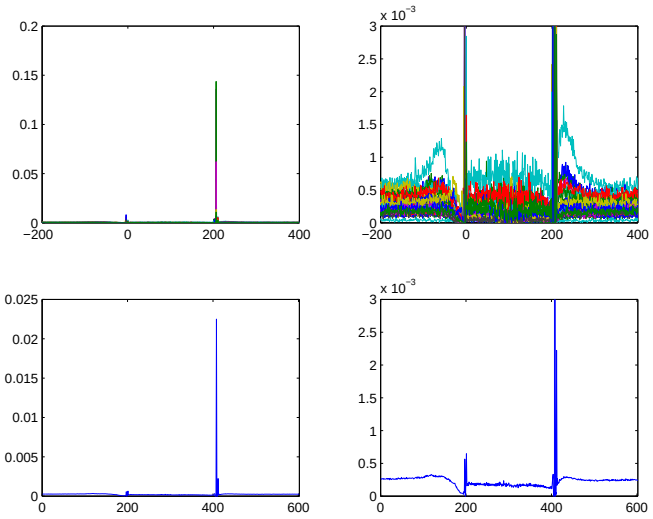



Figure 5.14: Collapsing Clustering Results: Cluster 14

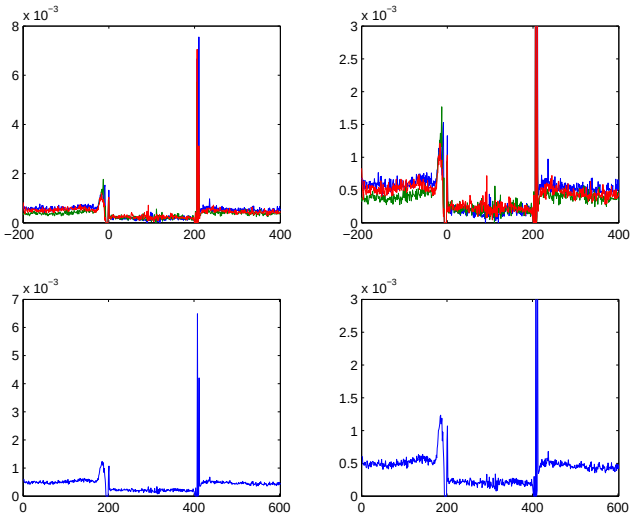



Figure 5.15: Collapsing Clustering Results: Cluster 15

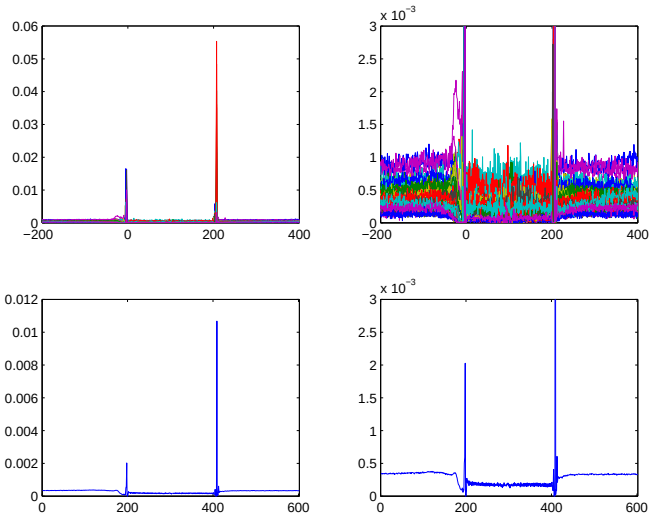



Figure 5.16: Collapsing Clustering Results: Cluster 16

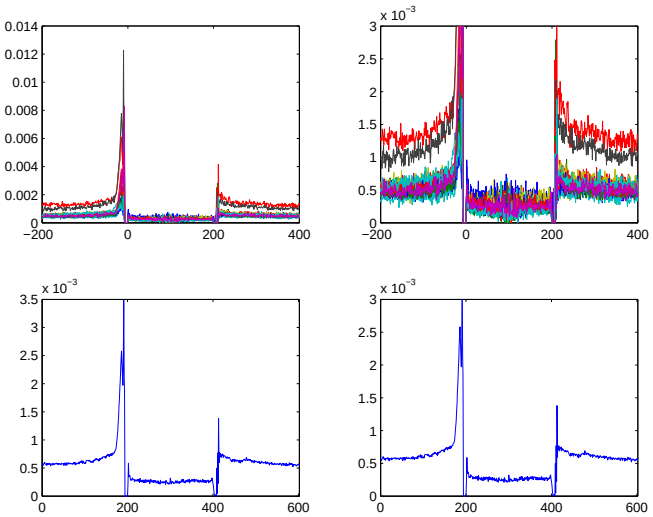



Figure 5.17: Collapsing Clustering Results: Cluster 17

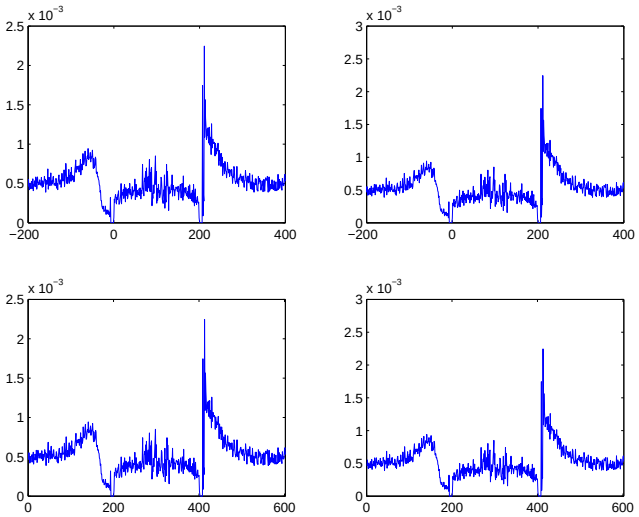



Figure 5.18: Collapsing Clustering Results: Cluster 18

# Chapter 6

# Appendix B

## 6.1 Human and fish cross species analysis

## 6.2 Human and cow cross species analysis

Table 6.1: Cross species analysis for region 1 (position -200 to 0) between human and fish. The most left column represents the top 5 words with highest Manhattan distance. The middle column shows the the actual Manhattan distance for the corresponding word between two species. The most right columns shows the genomic distribution plot. The x-axis represents the genomic positions and the y-axis represents the frequency of occurences. The green plot is the fish distribution and the blue plot is the human distribution.

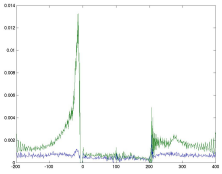
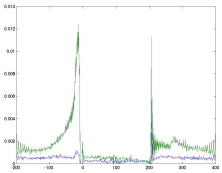
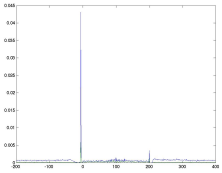
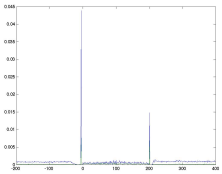
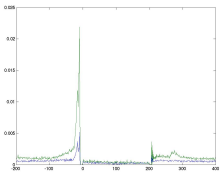
| Word | Manhattan Distance | Genomic Distribution |
|---|---|---|
| tgtgtg | 0.04228 |  |
| gtgtgt | 0.04147 |  |
| ccccag | 0.03802 |  |
| cccagg | 0.03735 |  |
| tgtgtt | 0.03571 |  |

Table 6.2: Cross species analysis for region 2 (position 0 to 200) between human and fish. The most left column represents the top 5 words with highest Manhattan distance. The middle column shows the the actual Manhattan distance for the corresponding word between two species. The most right columns shows the genomic distribution plot. The x-axis represents the genomic positions and the y-axis represents the frequency of occurences. The green plot is the fish distribution and the blue plot is the human distribution.

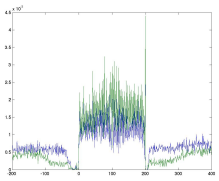
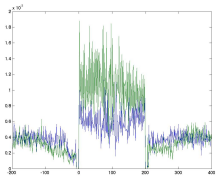
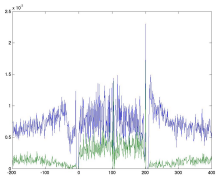
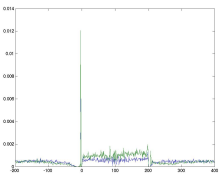
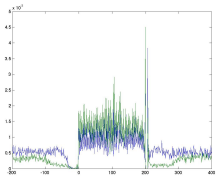
| Word | Manhattan Distance | Genomic Distribution |
|---|---|---|
| ctggag | 0.007448 |  |
| tctgga | 0.007309 |  |
| ccctgg | 0.006860 |  |
| cagaga | 0.006843 |  |
| gaggag | 0.006414 |  |

Table 6.3: Cross species analysis for region 3 (position 200 to 400) between human and fish. The most left column represents the top 5 words with highest Manhattan distance. The middle column shows the the actual Manhattan distance for the corresponding word between two species. The most right columns shows the genomic distribution plot. The x-axis represents the genomic positions and the y-axis represents the frequency of occurences. The green plot is the fish distribution and the blue plot is the human distribution.

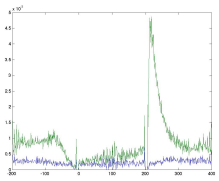
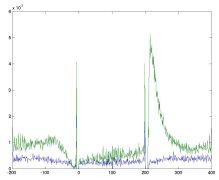
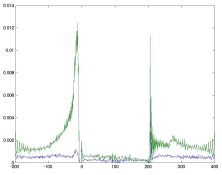
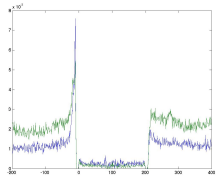
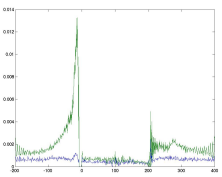
| Word | Manhattan Distance | Genomic Distribution |
|---|---|---|
| acacac | 0.2155 |  |
| cacaca | 0.2096 |  |
| gtgtgt | 0.2060 |  |
| tttatt | 0.2010 |  |
| tgtgtg | 0.1927 |  |

Table 6.4: Cross species analysis for region 1 (position -200 to 0) between human and cow. The most left column represents the top 5 words with highest Manhattan distance. The middle column shows the the actual Manhattan distance for the corresponding word between two species. The most right columns shows the genomic distribution plot. The x-axis represents the genomic positions and the y-axis represents the frequency of occurences. The green plot is the cow distribution and the blue plot is the human distribution.

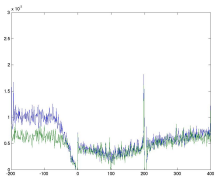
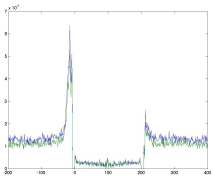
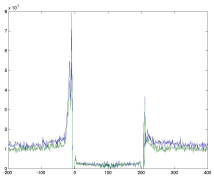
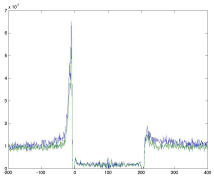
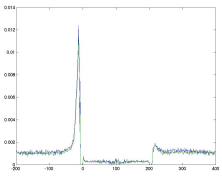
| Word | Manhattan Distance | Genomic Distribution |
|---|---|---|
| aaaaaa | 0.06025 |  |
| attttt | 0.04994 |  |
| tatttt | 0.04981 |  |
| ttattt | 0.04102 |  |
| ttttct | 0.03770 |  |

Table 6.5: Cross species analysis for region 2 (position 0 to 200) between human and cow. The most left column represents the top 5 words with highest Manhattan distance. The middle column shows the the actual Manhattan distance for the corresponding word between two species. The most right columns shows the genomic distribution plot. The x-axis represents the genomic positions and the y-axis represents the frequency of occurences. The green plot is the cow distribution and the blue plot is the human distribution.

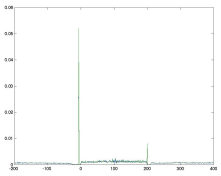
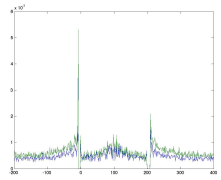
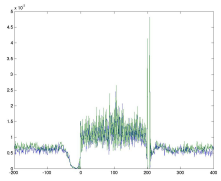
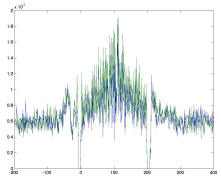
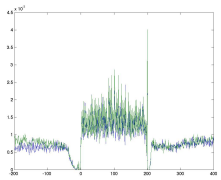
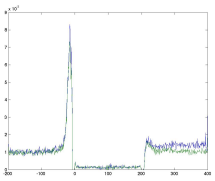
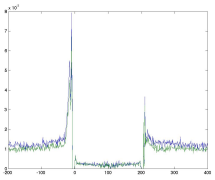
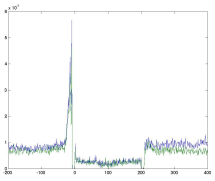
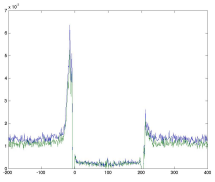
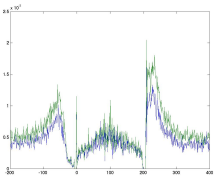
| Word | Manhattan Distance | Genomic Distribution |
|---|---|---|
| ctgcag | 0.026112410417473 |  |
| gccccc | 0.0257323505230477 |  |
| gaggag | 0.0254071069816399 |  |
| ctggcc | 0.0251540567525089 |  |
| ctggag | 0.0249829392841099 |  |

Table 6.6: Cross species analysis for region 3 (position to 200) between human and cow. The most left column represents the top 5 words with highest Manhattan distance. The middle column shows the the actual Manhattan distance for the corresponding word between two species. The most right columns shows the genomic distribution plot. The x-axis represents the genomic positions and the y-axis represents the frequency of occurences. The green plot is the cow distribution and the blue plot is the human distribution.

| Word | Manhattan Distance | Genomic Distribution |
|---|---|---|
| tttttt | 0.0609502772042235 |  |
| tatttt | 0.0458773112735974 |  |
| tttttg | 0.0453199185779619 |  |
| attttt | 0.0422775351635401 |  |
| gggggc | 0.0412430629486314 |  |