Unravel motifs in UTRs and introns

Applied Bioinformatics, 2015

Tobias Frick	930410-2438	tfrick@kth.se
Lisa Berg	930118-0726	lisb@kth.se

Abstract

With facilitation of sequencing of big amounts of biological data in the recent years, there is an increasing demand of analysis techniques to interpret results from this data. The usage of sequence logos is one way to observe interesting patterns from sequence data. In this project, three different locations in the human genome have been studied. The positions analysed were the translation start site, the start of the first intron and the end of the first intron for every gene. This was made by constructing a Python script that extracted the wanted sequences from transcript data from the Ensembl database. The online version of WebLogo was used to construct the sequence logos. The results show clear similarities with the theoretical patterns in these positions. In conclusion, it is possible to use the constructed script for analysis of the sequences in translation start site, first intron start site, and first intron end site.

Introduction

Recent progress in sequencing technologies have made sequencing very cheap today and has resulted in vast amounts of data along with big databases to organise this data in order to make it accessible for researchers. From these databases, information can be easily accessed even though it might be hard to interpret the meaning of it which have resulted in the field of bioinformatics.

A very simple but yet effective way to filter information and study similar sequences is to create sequence logos for specified sites, e.g. translation start sites. A logo is created by taking sequences centered around one specific point, align them on top of each other and see if they share the same bases at any nearby positions [1].

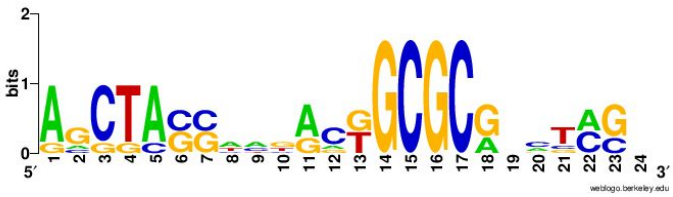

Figure 1: Sequence logo example. Above is a logo created from 6 arbitrary sequences, all sharing the sequence GCGC at positions 14-17. It can be imagined that the alignment was created by using e.g. the transcription start site as position 1 for every sequence.

This project has been centered around studying similarities at three sites in the human genome; translation start site, the first intron start site and the first intron end site. To do this, a python script called seqlogo was developed, which takes transcript sequences along with positional and identificational data in fasta format and returns one sequence for every gene around every given site. These sequences were then put into WebLogo in order to create the logos and investigate any similarities for the aligned sequences [2]. For every site, the 10 bases before and after was used to create the logo.

Lastly this report also had the aim to reflect the paper on computational biology structure by William Stafford Noble in a real computational biology project [3]. Therefore, this report will also compare and reflect on what has been done in this project in according to Stafford Noble and what has not.

Methods

After writing seqlogo it was tested with a number of test files to ensure the proper function of the script. This included files which sequences were not long enough to have bases 10 positions before or after the given site, and files which contained transcripts with only one exon. Some error handling was also tested to validate that seqlogo does not work on insufficient data. This was done by running seqlogo on testfiles which were empty, files which did not contain data for the position of the translation start position, and files with sequences shorter than 20 bases. For all test files, see appendix 1.

When sequences were not long enough around the site in focus, it was interpreted as there was no available data and gaps were inserted to indicate to the logo that no information had been found on these positions for that particular sequence. When there was only one exon, no sequences were retrieved for the intron positions, since there cannot be any introns in a transcript with only one exon.

The sites were studied by downloading transcripts from Ensembl, filtering and extracting the targeted site sequences with seqlogo and running these sequences at WebLogo to create the logos (figure 2).

Figure 2: Method overview. The data was first downloaded from ensembl and a file with transcriptional data was acquired. This was filtered by the use of a seqlogo and put into Weblogo to create a logo for the target sites.

When downloading the transcripts from Ensembl, the dataset Homo sapiens genes (GRCh38.p5) for protein coding genes was used. The unspliced transcripts were downloaded and some positional data, gene ID and strand information was also included (5'UTR End, 5'UTR Start, Transcript Start (bp), Transcript End (bp), Ensembl Gene ID, strand) to later be able to identify the target sites. The order of the positional data downloaded is important for seqlogo to function properly. The transcriptional data was entered into the python script (see appendix 2) with the following command:

```
./seqlogo.py human_transcriptome.fa translstart.aln intronstart.aln intronend.aln
```

The three output files (the last three statements) were then uploaded to WebLogo one by one and a logo was created with the default setting, except for the first position number which was set to -10 so the target site would be at position 0 in the logo.

In the paper written by Stafford Noble [3], it is highlighted that it is of importance to keep the documentation of the work updated throughout the project to easier make changes and come back to parts of the work at a later point in time. This was considered important in this project since it at many points was needed to change big parts of the script. The script was continuously updated with documentation. Also, when creating seqlogo, a main project directory was used with one subfolder for temporary data or scripts. Furthermore, google drive was used to upload current script versions along with test files. A coding diary was also kept on google drive for every day where the project was being worked on. No README file, nor any result summary file were created, but the argparse module of Python was used to indicate the usage and the demanded structure of the input file for the user. This was made in line with the requirements from Stafford Noble that all scripts or programs should be able to show the usage statement [3].

Results

The results of the test runs on the created test files containing simple cases or corner cases can be seen in appendix 1.

13,549 sequences were aligned for the translation start site, and 13,094 sequences were aligned for the intron start and end sites. WebLogo accepted only a maximum of 10,000 rows, and the sequence logos are therefore constructed using the 10,000 first rows of the aligned sequences.

The logo acquired after running WebLogo showed what was to be expected by the respective logos. The translation start site logo showed a strong bias to ATG at the start of the sequence, as well as some bias to the presence of G and A at position 0 and 2 respectively, see figure 3.

Figure 3: Translation start site logo. A strong bias to the sequence ATG can be seen at positions 0-2, and a minor bias toward G and A at position 0 and 2, respectively.

The logo for the intron start site showed that every extracted intron sequence started with GT, and some similar regions after the intron start site with a tendency to A/G, see figure 4.

Figure 4: Intron start site logo. At positions 0-1 the bases GT has been found for seemingly every intron studied. Before and after the intron some bias toward G/A can also be seen.

Lastly there was the intron end site, which showed that every intron studied ended with AG just before the last base of the intron. Furthermore, they seemed to have a higher amount of T/C leading up to this position, see figure 5.

Figure 5: Intron end site logo. The dominating sequence at -3 to -2 is AG and furthermore, there is a bias towards T/C leading up to the end of the intron.

There were no obvious problems as a result from lack of organization in this project. Some minor difficulties were faced when trying to run the analysis on different sets of data and a lack of organisation made it necessary to rerun some data sets.

Discussion

After running the testfiles it seemed like seqlogo was fairly robust to any eventual faulty data or incomplete data. Some improvements that could be done would be to make sure it is actually a fasta file that is being read, however since this was not the primary focus of this project it was not regarded as important.

seqlogo incorporates gaps if the sequence before and/or after the target site is not available. This is because the sequence data is not included in the unspliced transcript data if the position of the translation start site is located less than 10 bases into the first exon, or if the last exon is less than 10 bases long, which is very unlikely. It is still wanted to include the available bases, and therefore hyphens are included to be compatible with WebLogo. It can be argued that it would have been better to use gene data instead of transcript data in order to avoid this problem, but since it was assumed only in rare cases sequences did not contain all 20 bases needed for creating the logo it was considered an accepted loss of information.

Overall the sequence logos seemed to make sense according to what was to be expected from the different sites and what is known about the targeted sites [4,5]. One at first unexpected thing was the fairly strong bias to G/A at the translational start site, since the most common start codon is ATG [4]. As discussed by Ivanov et al, it is possible for genes to start with other start codons than ATG.

The workflow of this project has only to some extent reflected the recommendations of Stafford Noble. A general theme is that almost every main point made by Stafford Noble has some influence of the work performed but almost never in such detail or in such extent. As this project was deemed as a very small one and the need to revisit the code later was also deemed as quite unlikely it was often seen as unnecessary work to put more effort into making directories with specific dates or keep track of older versions in a more ordered manner. However, it should also be taken into account that Stafford Nobles primary audience probably is not two biotechnology students when working on a programming project over only a few weeks, so it should probably be expected that everything is to be scaled down in order to fit the circumstances.

Conclusion

The analysis of testfiles as well as real human transcript data from ensembl has shown that seqlogo can perform some basic sequence retrieval for logo creations and also handles some of the most common errors that users might infer whilst running the script.

Intron start and intron end sites have been shown to have the motifs they are supposed to have according to common knowledge whilst the translation start site has shown to have have other bases then what is expected according (the ATG start codon). Since no reason has been found for this, there are still some minor doubts about the proper function of seqlogo. On the other hand, it seems more reasonable to believe either the human transcriptome contains a larger amount of non-standard start codons or the logo investigation method is flawed since two of the three investigated sites give results which are to be expected from this kind of investigation.

Stafford Noble has a very well written paper on how to organise computational biology projects and even though some information might seem obvious to some readers, it is a very good foundation for what to document and how to document it by someone new to the field. Even though Stafford Noble has very good recommendations for a wide audience, there is one thing that could be improved - the need for adapting the amount of documentation and organisation according to the current situation. With that said however, adaptability seems to be Staffords Nobles intent, even though it is never explicitly said.

References

[1]	Schneider, T. D., & Stephens, R. M. (1990). Sequence logos: a new way to display consensus sequences. *Nucleic Acids Research*, *18*(20), 6097–6100.
[2]	Crooks, G. E., Hon, G., Chandonia, J.-M., & Brenner, S. E. (2004). WebLogo: A Sequence Logo Generator. *Genome Research*, *14*(6), 1188–1190.
[3]	Noble WS (2009) A Quick Guide to Organizing Computational Biology Projects. PLoS Comput Biol 5(7): e1000424.
[4]	Ivanov, I. P., Firth, A. E., Michel, A. M., Atkins, J. F., & Baranov, P. V. (2011). Identification of evolutionarily conserved non-AUG-initiated N-terminal extensions in human coding sequences. *Nucleic Acids Research*, *39*(10), 4220–4234.
[5]	Corden JL, Patturajan M. (1997). A CTD function linking transcription to splicing. *Trends Biochem Sci* 22, 413–416.

Appendix 1

Empty file

The simplest test file contains no information at all, e.g. an empty file. This should return an error message to Stderr (‘No data found.’).

Short sequences

This test file contains one sequence with more than 20 bases and one with only 15 bases. The sequence with only 15 bases (< 20 bases) should be excluded, and the result should contain only one sequence. Moreover, Gene1 in short_sequences.fa contains only one exon, and no sequences are extracted for the intron start and end sites.

```
short_sequences.fa
>15|3|3|40|Gene1|1
AAAAATTTTTGGGGGCCCCCAAAAATTTTTGGGGGCC
>10|2|2|17|Gene2|1
AAAAATTTTTGGGGG

./seqlogo.py short.fa out1 out2 out3

cat out1
AATTTTTGGGGGCCCCCAAA
```

Sites close to the edge

If the position of the translation start site is located less than 10 bases from the transcript start site, it is not possible to retain these bases from the transcript data. It is however wanted to still include the sequences in the sequence logo. Adding hyphens in these positions solves the problem. For example:

```
cat transl_start_close_to_edge.fa
>5|1|1;30|20;55|Gene1|1
AAAAATTTTTGGGGGCCCCCAAAAATTTTTGGGGGCCCCCAAAAATTTTTGGGGG
```

```
./seqlogo.py transl_start_close_to_edge.fa out1 out2 out3

cat out1
-----AAAAATTTTTGGGGG
cat out2
GGGGGCCCCCAAAAATTTTT
cat out3
AAAAATTTTTGGGGGCCCCC
```

Missing information

In some cases, the information about the start and end points of the UTR regions is unknown. This will be recognized as ‘’ and the transcript is not used for constructing the sequence logo.

```
cat missing_info.fa
>''|''|5:30|15:50|Gene1|1
AAAAATTTTTGGGGGCCCCCAAAAATTTTTGGGGGCCCCCAAAAA

./seqlogo.py missing_info.fa out1 out2 out3

No data found.
```

Transcripts from same gene

When transcripts origin from the same gene, only the transcript with the intron located closest to the translation start site is of interest, and the rest are disregarded. In from_same_gene.fa, the second entry has an intron located earlier than the first entry, thus only sequences containing Gs should be extracted.

```
cat from_same_gene.fa
>20|5|5;50|35;70|Gene1|1
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
>20|5|5;50|30;60|Gene1|1
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
```

Only one exon

When transcripts contain only one exon, there is no intron in the transcript and the sequences around intron start and intron end positions cannot be extracted. In these cases, only the bases around the translation start site are identified.

```
cat one_exon.fa
>15|5|5|30|Gene1|1
AAAAACCCCCGGGGGTTTTTAAAAA

./seqlogo.py one_exon.fa out1 out2 out3

cat out1
AAAACCCCCGGGGGTTTTTA
cat out2
cat out3
```

Appendix 2

https://github.com/lisaberg/appbio_project/blob/master/seqlogo.py