

Spring 2012

Promoter Prediction Based on E. coli Characteristics

Li Wen

San Jose State University

Follow this and additional works at: https://scholarworks.sjsu.edu/etd_projects



Part of the [Computer Sciences Commons](#)

Recommended Citation

Wen, Li, "Promoter Prediction Based on E. coli Characteristics" (2012). *Master's Projects*. 216.

DOI: <https://doi.org/10.31979/etd.tcpk-tx2n>

https://scholarworks.sjsu.edu/etd_projects/216

This Master's Project is brought to you for free and open access by the Master's Theses and Graduate Research at SJSU ScholarWorks. It has been accepted for inclusion in Master's Projects by an authorized administrator of SJSU ScholarWorks. For more information, please contact scholarworks@sjsu.edu.



2011

Promoter Prediction Based on E. coli Characteristics

This project uses the characteristic in TATA-less regions on E. coli sequences to predict the promoter region before TSS, which indicate that the real gene has been located. It uses several well-known algorithms and methods such as the sliding window algorithm, and a clustering method to predict promoters. It also contains D2K algorithm and method to compare predicted result with other online promoter package result.

Li Wen
Dr. Sami Khuri
12/16/2011



Table of Contents

Introduction	2
Theory Testing (Approach)	4
The D2K Algorithm	18
Implementation	21
Result	22
Conclusion.....	31
Reference	31
Appendix 1—Training Data	34
Appendix 2—List of NCBI Accession Number of Testing Data.....	35
Appendix 3—D2K Algorithm Coding in EGF	36
Appendix 4—Result Comparison	41
Appendix 5—Web Application	61

Introduction

Recent promoter predictions

A promoter is a piece of DNA sequence which frequently appears before its associated gene in an E. coli sequence. Researchers predict the location of a promoter when trying to locate a gene in a given sequence. These researchers use a variety of algorithms to predict locations of promoters. These algorithms include machine learning, artificial neural network, the Markov model, the weight matrix, etc. Abeel et al apply machine learning [10][13] and Bland et al use artificial neural networks [8] to classify and output the predicted promoter location of any given unknown sequence. Burden et al apply the weight matrix algorithm [5] to identify motifs in the promoter region. Burden et al also use Markov model [5] to find the shortest path to the promoter location. Most of the researchers use the combination of artificial neural network, machine learning, and Markov model to increase the true positive prediction results [8].

Some researchers provide additional information when predict promoter location. For example, Gan et al introduced the idea of using non-CpG [6] region information as CpG region for prediction. Gan et al found the equal significance of non-CpG region in their experiment. Burden et al use the distance between TLS and TSS for promoter prediction [5]. The distance between TSS and TLS provide additional clues of promoter location which can increase the true positive of prediction. Davuluri et al are interested in finding the first exon [4], since it is the most difficult promoter location to find. First Exon Finder—FirstEF [4] uses CpG information to find the first donor site for first gene exon predict.

Data used for promoter prediction varies. Wang et al use the comparison of human and mouse genome for prediction [14] [4]. Laser et al use mammal and plant genome for prediction [26]. Most of the data used in promoter prediction is E. coli [8], especially E. coli K12 for its promoter richness [30]. In my research, I used E. coli from NCBI [30] database and plant data that I retrieved from plantDB [3]. Plant data are only used for testing and training purpose.

The researchers discussed above using their algorithms and data were able to reliably predict the location of a promoter.

The lack of using TATA-less regions in researches

TATA box is a piece of DNA sequence that usually appears in promoter region. Originally, it has been used to locate promoter locations [2]. However some researchers prefer not use TATA box as a signal of the promoter region. Burden et al state that the TATA box is not an effective resource for promoter prediction [5]. The data sequences they used in their experiment are mixed with TATA-rich, and

TATA-less promoters; their analysis shows no strong relation connects TATA-less regions with promoter regions. For example, in the paper “Improving promoter prediction”, Burden et al said that the characteristic of TATA box is limited and cannot be used to recognize when indel happens [5]. Also, the definition of TATA box says that less than 20% of human promoters have TATA box, which leads the rest of the promoter regions unsearchable by using TATA box as searching factor [21]. Burge et al show in their experiment only 70% of core promoter contains TATA box [23]. Therefore, most researchers ignored a large amount of available data—TATA-less region during promoter prediction process.

Motive of using non-TATA region

The idea of using non-TATA region for promoter prediction comes from one of the research papers—“A pattern-based nearest neighbor search approach for promoter prediction using DNA structural profiles”. In their research, Gan et al discovered that non-CpG-island region contains similar characteristics with CpG-island region [6]. Gan et al researched CpG-island region by compute the gravity of CpG in every promoter region in given sequences. Researchers did the same calculation for non-CpG-island region. The results show that non-CpG-island region provide equally important prediction information. For any given unknown sequence, the promoter can be predicted by using both CpG-island and non-CpG island classes to increase prediction result.

In my research I will use information on non-TATA regions for promoter prediction. TBP (TATA binding protein) is used to bind sequences with TATA box, and I am looking for the regions in TATA-less sequences that will be bind with other binding proteins in TFIID (transcription factor II D) [22]. Promoter regions usually contain TFIIB recognition element (BRE), TATA box, Inr, and downstream promoter element (DPE). Most of promoters miss one of these elements, and for promoters that do not have TATA box will have a high probability of having DPE [23].

State of theory

Thus far, no research has focused on using non-TATA regions for promoter prediction. TATA box is used for promoter prediction in many papers for sequences contain TATA box. For sequences without TATA box information, existing algorithms cannot do much analysis in the data mining step of the prediction. Sequences without strong TATA box information are basically ignored during data mining step.

In this paper, I studied the characteristic of non-TATA or less-TATA region. I found the characteristic in TATA location of non-TATA regions, to get clues as to predict promoter region based on both TATA and non-TATA information. The model of this project was trained and acts like a polymerase. Polymerase does not use TATA box regions, but it was based on the structure or chemical statement of TSS in order to open the double helix, and start coping gene. This initial location indicates that a real gene is many nucleuses away, and waiting to be copied. Finding the initial location

will be used as characteristic of the TATA box in this paper. These characteristics will be coded in the model to detect any given E. coli sequence from NCBI.

I hope to show that the predicted gene location will be close to real gene location in NCBI database, or get more true positive values than several popular online prediction tools. These tools include EasyGene, GenScan, Virtual Footprint, and Glimmer (the original prediction that used by NCBI).

Theory Testing (Approach)

Software and algorithms used during theory testing step

For this research I used a combination of commercial and custom software. The commercial software is SAS—a well-known business analytical software. I'll first try to find the significant of non-TATA region information by using SAS. For this research, I also created a custom Perl program—called E. coli Gene Finder (EGF), which encodes machine learning, data mining, five folds, and the artificial neural network algorithm. I used this software combination to find the most important characteristics of the non-TATA region in my training data. I also use these algorithms to find the threshold which define the decision tree. The EGF is adjusted in the theory testing step to filter out a promoter sequence out of any given E. coli sequences. The steps of using SAS for sequences are listed in the steps section below. During model training step, fivefold method will be used.

Data set used in theory testing step

The data set I used for training my model purpose is plant promoters that are obtained from PlantProm DB [7]. 170 of them are TATA rich plant sequences and 130 of them are TATA-less plant sequences. Once the significance of non-TATA information is discovered by using SAS and my program, I'll use E. coli data sets for both my training and testing data in the machine learning step. The reason to choose promoter data set from PlantProm DB are: first, they provide a clear plant data for both TATA rich promoter region, and TATA-less promoter region. I choose the plant data with the intention that the theory could be applied to other species.

Pretest on TATA region

The basic characteristic of TATA box is it matches the expression 5'-T A T A (T/A) A (T/A) ---3'. Data in TATA rich promoter are input in an online promoter prediction service—BDGP [3]. It is a neural network promoter prediction web service that developed by Berkeley Drosophila Genome Project. It uses the combination of a neural network and weight pruning to search for consensus elements, such as TATA box, CpG Island, CAT box, etc. When testing 170 TATA rich sequences, TATA rich sequence's characteristic is clearly displayed in the similar location of each of

submitted sequence—25 base pair upstream of transcription start site. None TATA rich sequences—130 of them, also have been tested by using the same service, but no characteristic are identified. The target of using SAS is to find some valuable information around the location where TATA usually appears; since in reality, when polymerase is walking along on the DNA double helix, it will find the initial point not the letter TATA inside promoter region.

Steps of finding characteristic

Data of plant promoter are obtained from PlantProm DB [7] to test my hypotheses.

Steps of finding TATA and non-TATA data:

1. Get both TATA promoter and TATA-less promoter data from PlantProm DB
 - 1.1 Base URL: <http://mendel.cs.rhul.ac.uk/mendel.php?topic=plantprom>
 - 1.2 175 TATA rich location: http://mendel.cs.rhul.ac.uk/pprom/PLPR_TATA.seq
 - 1.3 130 TATA less location: http://mendel.cs.rhul.ac.uk/pprom/PLPR_TATA-less.seq
2. Filter out the known promoter parts from both data sets
 - 2.1 Write EGF (Perl program) to filter the two original data sets
 - 2.1.1 Change the input and output file names of both data files
 - 2.1.2 Delete the unwanted original data file
 - 2.2 Extract the promoter parts
 - 2.3 Output two generated promoter data sets
 - 2.3.1 Match all the word character lines in the file
 - 2.3.2 Go to a new line for a new sequence
 - 2.3.3 Both TATA rich and TATA less sequences were processed
3. Generate statistics on both promoter sets
 - 3.1 Perform a relation test on TATA promoter data set
 - 3.1.1 Find the average location of TATA box for both “TATA” and “TATAA”
 - 3.1.1.1 Get TATA location in each sequence
 - 3.1.1.2 Get average location by using below formula:

$$\text{average} = \frac{\sum \text{locations}}{\text{total number of sequence}}$$
 where locations are cutting off from 160 since less than 160 will be out of promoter region. Average = 171. This formula was coded and tested inside the main Perl program in data testing section. The results are given in Figure 1 and Figure 2.

```

C:\ Command Prompt
163 --> 173
164 --> 195
165 --> 174
166 --> 46
167 --> 169
168 --> 18
169 --> 89
170 --> 88
172 --> 46
173 --> 172
174 --> 172
175 --> 164

TATA promoter sequences average TATA location is: 171.083333333333

```

Figure 1. Test result for TATA location for TATA rich promoters in file TATApromoter175.txt. First column is the sequence number from 1 to 175, followed by the TATA location that is detected by the program in that sequence. Location number less than 160 is ignored since it is out of promoter region. The average of remaining sequence that matches promoter region definition is calculated as 171.08333.

```

accactatataaaatcagggctcattttctcgctcctcacaggctcaicicg
acaattctagattttgttataaaattcacatattgtatgagtataATACAT
atgggagctataaaaagccttgtatgatcatcatccttcctcACCCAT
ataaacactataaaaaccactgcaacaaccttgtatcaacgcattGAAAGG
ctctggtatataatagataaccaaaagcgatagacaaacaagtaagtTAAGAG
atctacatttcactatataataaccaacttagcttgccttctcatcATAATC
tgccgtgctgcacctataaaattcacatgcaccggcatgccactccACACAA
tcgggttcctctataaaatacatttcctacatcttcttctcctcACATCC
accttgctttctaaatataacatacatataacttgccctattacgcaaAGTCAC
gcctttatctcactataaatgcacgatgatttctcattgtttctcACAAAA
cacacccctccctataaaataccaggcaccttagtacacttgtaacCATCAG
tggcatcgacttctctataaaataccaagcacgtagaactcttgtaaCCATCA
tggcatccactgcctataaaataccaagcacgaggacacttgtagCCATCA
tggcatcgactgcctataaaataccaagcacgaggacacttcttagCCATCA
tggcatcgactgcctataaaataccaagcacgtagcgaacacttgtaaCCATCA
tcgagtcctcctgcctataaaataccaagcacgtaggtacgcttgtagCCATCA
tcgagtcctcctgcctataaaataccaagcacgtagtacccttgtagCCATCA
ctgctcaccttactataaaatctctctctctctctctctctctctctgAACAG
ctctcctccggccaatataaaacaccaattctcactctcacttttTATACT
atgctgcagcacactataaaatacctggccagacacacaagctgaATGCAT
cgagtttgtagctataaaacctctccacttggttcttctcactctcACTGTT
ctcattaatcccttatataaaaggactccatagcctcaccattcACTCAT
cagtaaggccccaactaatataaaaaccagttattgggtgtgttACTCAT
tcctaatacacaggcattataaatggcacagggaattagcctcatcTACACA
taaccactctataaaaatcacctgatccttcctatgaaatccaCGTCCC
ccagctataaaatacgtctcctccttctccttctcctcctcatcgCCCTCA
ctgtaaagccatttatatacacttagtgcaaagcccatgaaactCAAGCC

```

Figure 2. The graph shows that the TATA location (in green) is almost at the same location for each sequence. The capitalized sequence is the start of TSS.

3.1.2 Find average location of TATAAA box

3.1.2.1 Get TATAAA location in each sequence

3.1.2.2 Get average location by using below formula:

average = $\sum \text{locations} / \text{total number of sequence}$

where locations are cutting off from 160 since less than 160 will be out of promoter region. Average = 174. Results are given by Figure 3 and Figure 4.

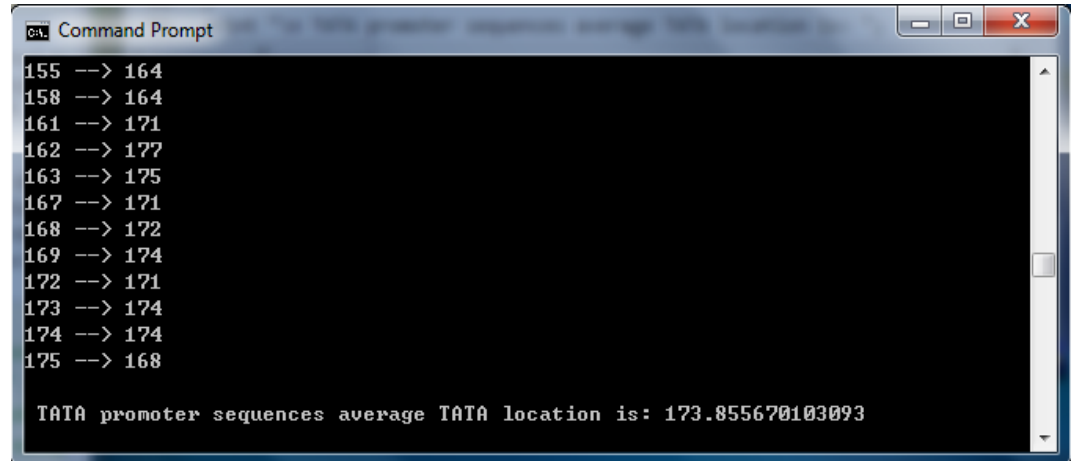


Figure 3. Test result for TATAAA location for TATA rich promoters in file TATApromoter175.txt. First column is the sequence number from 1 to 175, followed by the starting point of TATA location in that sequence. Location number less than 160 is ignored since it is out of promoter region as before. The average TATA start point is calculated as 173.8556

```

tctcactataaaatgcacgatgatttctcattgtttctcACAAA
ctccctataaaataccaggcacctagtagtacctgttaacCATCA
gacttctataaaataccaagcacgtagaactcttgtaaCCATC
actgcctataaaataccaagcacgaggacacttggtggCCATC
gactgcctataaaatacgaagccggtacacttcttagCCATC
gactgcctataaaataccaagcacgtcgaacacttgtaaCCATC
ccctgcctataaaataccaagcacgtggtacgcttgtagCCATC
ccctgcctataaaataccaagcacgtagtagtaccttgtagCCATC
cttactataaaatctctctctctctctctctctctctgAACCA
tccggccaatataaaacaccaattctcactctcacttttTATAC
agcacactataaaatacctggccagacacacaagctgaATGCA
gtgactataaaacacctctccacttggttcttctcactctcACTGT
aatccctataaaaggactccatgatgcctcaccattcACTCA
gcccccaactaatataaaaaccagttattgggtgtgttACTCA
acaggcattataaaatggcacaggcaattagcctcatcTACAC
cttataaaaatcacctgatccttcctatgaaatccaCGTCC
ataaaatagctctccctcccttctccttctcctcatcgCCCTC

```

Figure 4. The graph—from program notepad, shows that the TATAAA location (in green) is almost at the same location for each sequence with several nucleotides length difference. The capitalized sequence is the start of TSS.

- 3.1.3 Result: the location of TATA box in these plant sequences is around 170, since the data starts from -200 of TSS, the relative location of TATA to TSS will be around -30 (-30=200-170). This confirms that promoters containing TATA box are right before TSS, around -30

location [1][2].

3.2 Perform the same test on TATA-less promoter data set (TATAlessPromoter130.txt)

3.2.1 Theory: Get sub-sequence from -35 (get from -30-5 above) with length 20 out of each sequence. The idea of getting the information out of TATA-less sequences on the same location where TATA box appear in TATA-rich sequence is the motivation of polymerases. The polymerase has to be initialed before face the real gene; and that initialization must happen on the similar location of each sequence, and it doesn't matter whether it contains TATA or not. Therefore, this section will get other characteristics in addition to TATA box out of TATA-less sequence. The finding will be used to predict gene in TATA-less sequences.

3.2.2 Find characteristic

3.2.2.1 It cannot be done by using the same approach since it is TATA less sequence. TATA are all over the sequences. It is shown in Figure 5.

```
atatatcttttcattttgtataattgagattattaacaaagcttaatttCTAGTA  
ccagctccagtcggcaccgataaaagcggcaggcacttggattgctgACGAGA  
taaaagatactgattaatccagagggtttatatctacgccgtctccattgATTATT  
tgcccagagcatcccacgactacaaaacacggctggcggaggattataaCACGATC  
caatatgccaatccacgtgtatttaaggcgtcacatagctcggcctctatACTTTG  
gtgtccacaatcagctccatccattccataataacaagcagctcgagacATAAAAC  
aggccacccacccgcctatttaagccgcctccccctccattccccctcAAGAAG  
cagagcactgggggtttgcaactatttatttggtccttctggatctcggagAAACCT  
cgtatgcatctctttacttataattcggacagcgcgaatccctatgcgccAGCTCA  
tccacactggccactccctgcacttctataaacctttgtagcatatcttACTCTT  
ttaatatattttttattttattttattcatttatgtccaacaaattcatTTGATA  
aacaaaactacagtacttttctacaaatcaaatgtaaatcaattccatttCATTAC  
tctacgtgcgccttgggcctacatatgccctgctgtgggagtaccgctgACAACA  
ctttctctctcctccttcataaaaaaaccttctcactgatcccatccAGAAAA  
gtatcatcaaaccaacctctctctcctcactctactcatcccttatctgCGTATC  
tgatcatagagcatattataagagagtgaactaatggaggttgccctcAAGAAG  
ctattaattttataaatttcattttatpapaatpctaatttatttppacGCCAAT
```

Figure 5. In TATA less file, the TATA characteristic cannot be used since their location is not fixed, and some sequences don't have any TATA sequences.

3.2.2.2 Get the promoter region (-35, -1) with TSS region (+1, 40), and find characteristics from the same data by using SAS, where all characters are converted to numbers, [a,c,t,g,A,C,T,G]/[1,2,3,4,5,6,7,8]. SAS is the data analysis software, it can tell information out of given data. Next section will try to get characteristic by using SAS.

3.2.3 Relational test by using SAS to get characteristics

3.2.3.1 Pearson correlation coefficient is tested between promoter region (X1-X35) and TSS region (X36-X86). Pearson tests are done between every pair of column. Some of the columns are related with correlation value bigger than 0.9, such as the correlation between X17 and X37 is 0.9783.

3.2.3.2 Given X17, what is the value for X37? In other words, how to predict X37 of TSS region by using X17 of promoter region? We need to find a line that best fits the regression, so that the regression testing is done by using dependent variable X37 and independent variable X17 to minimize the sum of squared vertical distance from each data point to the line (residual). The formula we used is:

$$X37 = a + b * X17$$

where "a" is intercept and b is slope. The dependent variable—b is not significant enough to predict the independent variable X37.

3.2.3.3 Similar test for codon—3 nucleotides a pair, instead of single nucleotide are also finished. No significant results are found between codons.

3.2.3.4 Similar test for first TSS codon of each single nucleotide contains a percentage of the promoter regions which means testing a nucleotide in the first TSS codon that appears in the promoter region. Half of the total length of the promoter region ($33/2=17$) is used to prune noisy data. Noisy data is defined as outstanding data—the first 3 nucleotides of TSS is less than half of the total promoter nucleotide. Steps to calculate first 3 TSS single nucleotides are list as below.

- a) Get number of each of the first 3 nucleotides of TSS in promoter region. Such as \$aa = number of first nucleotide used in promoter region.
- b) Sum them for each sequence. \$dd = sum of all three nucleotides usage.
- c) Get percentage of above sum = \$dd / promoter region length
- d) Sum the percentage for all sequences with total usage bigger or equal to half of the promoter length. \$total = \$total + \$dd for every \$dd ≥ 17 .
- e) Take the average = \$total / number of sequences with \$dd ≥ 17 .

3.2.3.5 Test Result.

3.2.3.5.1 The result shows that most of the promoter region contains more nucleotides that belong to the first 3 TSS. In this case 130 TATA-less sequences are tested, 97 of them contains high usage of the first 3 TSS with average usage equal to 0.7085.

```

C:\> Command Prompt
121 --> tctatgtataatgaagccagaataacttcagtt==cat==0==c:5 ==a:12 ==t:11==>28
122 --> ttctcattataatctcttgcctctccaccaa==ctc==3==c:11 ==t:14 ==c:11==>25
123 --> ccactttaataatgatatattctattcagtggt==gtg==1==g:3 ==t:15 ==g:3==>18
124 --> ggtacaaaatgttgccgcgcagtcagtgctggt==atc==0==a:7 ==t:8 ==c:7==>22
125 --> actacttaatgcaagctgcggggcagagaaatt==ata==0==a:11 ==t:7 ==a:11==>18
126 --> atataagcatgatccgaggaacacacttaaac==aat==0==a:15 ==a:15 ==t:6==>21
127 --> ataaagggaacactacctctcctaatggcagt==acc==1==a:11 ==c:9 ==c:9==>20
128 --> acaataatttaatttgactatttagtgaatgaa==tga==2==t:12 ==g:5 ==a:14==>31
129 --> aagcgacagttctgatttctgcccctcccaatc==caa==1==c:12 ==a:7 ==a:7==>19
130 --> agaggtgagagctaagattttcccatcaactcc==att==1==a:10 ==t:8 ==t:8==>18
first3n==>64
higer==>97==total==>0.708528584817245

C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>

```

Figure 6. Test result for TATA less promoters

- 3.2.4 Conclusion: From this experiment we can see that even though the sequences don't have TATA box as characteristic to signal promoter region, the nucleotide usage in promoter region can be used as a clue of TSS region. From Figure 6, sequence 128, it clearly shows that the 'tga' usage is 31 out of 33, there is only one 'c' used before TSS. This finding can be used as a secondary characteristic to indicate a real gene in TATA-rich sequence. And for the same reason, it can be used as the main characteristic to predict real gene from TATA-less sequence.
4. Analyze the results and get characteristic for both data set
 - 4.1 For TATA rich sequences, use TATA as consensus sequence to identify the promoter region.
 - 4.2 For TATA less sequences, use first 3 TSS nucleotides to signal the TSS position. Since the finding depends on individual nucleotides, it is not very significant, therefore it cannot be used as a characteristic to locate promoter region in general.
5. Working with the data on protein level
 - 5.1 Translate both TATA rich and TATA-less sequences into amino acids equivalents and see if any significance appears up in the next attempt. The idea behind this transition is when polymerase bind to the TATA boxes, it is not attracted by the nucleotides of TATA or TATAAA, and instead it is attracted by the product of the nucleotides. Therefore, if polymerase can bind to the promoter region without TATA box as common notation, then it needs to find similar protein to bind. The TATA-less region needs to have such a protein to attract polymerase for initiation of TSS.
 - 5.2 Test the TATA rich sequences without using TATA as characteristic
 - 5.2.1 Translate each sequence to its amino acid equivalent by using hash table [1]. In order to find the relationship between the first amino acid with a promoter region, use the number to represent amino acids instead of the real protein name[15][16].
 - 5.2.2 Clean the data by filtering all sequences containing at least a quarter

of the first amino acids of the TSS region; and the resulting data after this step looks like Figure 7 below. Only 142 out of 175 met the requirement.

```

132 2211122411221222111220422110411122121411221122212204111011214421122
133 42424214221141121212122121121211241122402212041124212210422412212
134 101221112221221122421121222211212140112221112412111212221222211112
135 211142124410221212211401202224114122224222211202112141121211212222
136 212011020021012111121241221221241122224222211200142121101411224222
137 2211222114020241211222112122212141240222211221411112121121422111442
138 1412222122242421112214121121122422121042001441220112112101120241211
139 1121120211111400222122221121210111400222111122121121211212221442121
140 2222221212111242222111422421222214112121421222122122242104122142112
141 222112112114414224112122122220422122224411210001114111101222121212
142 22241421412221221111221141121122222112112112221212210121411114211
143

```

Figure 7. TATA rich file numerical protein product representation

- 5.2.3 Find the location that is most closely associated with the first amino acid in TSS by using the sliding window algorithm [1]. For each 10 amino acid (because the length of TATA box) in each sequence, the maximum appearance of the last number in this sequence will be counted; and the middle location of that maximum number will be calculated by using below formula:

Peak of the first aa in TSS in promoter region = Max (appearance in each 10 amino acid)

Location of the max = the peak location + 5; 5 means set the location to the middle of the window, since the sliding window size is 10.

- 5.2.4 Thresholds: data will be cut if it does not meet the thresholds in two conditions. If the maximum total appearance is less than 6 (half of the sliding window) and the location is not close to TSS region. In this case the threshold for location is 33, which is 32 amino acids long to TSS; otherwise, the location of such peak cannot be characterized. Below is the running result with above thresholds.

```

131 ==> 7 ==> 62
132 ==> 7 ==> 45
133 ==> 6 ==> 22
134 ==> 8 ==> 58
135 ==> 9 ==> 39
136 ==> 8 ==> 40
137 ==> 7 ==> 24
138 ==> 7 ==> 54
139 ==> 7 ==> 45
140 ==> 8 ==> 48
141 ==> 8 ==> 25
142 ==> 7 ==> 62
Average max number of first aa in the sequence is: 7.21428571428571
And the average location of that window is: 50.2380952380952
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>

```

Figure 8: Test result of the max appearance and the average location in TATA rich file. The average peak in one sequence contains about 72% of the first amino acid of TSS. The peak appears around the 50th amino acid, which is very close to the TATA box location exams before.

5.3 Search for similar characteristic in TATA-less file, so this can be used to detect promoter region for any given unknown sequence.

- 5.3.1 Translate the TATA-less file into its protein equivalent [15] [16].
- 5.3.2 Clean the data by filtering all sequences containing at least a quarter of the first amino acid of TSS region; and the result data after this step is represented in Figure 9 below. Only 107 out of 130 meet the requirement.

```

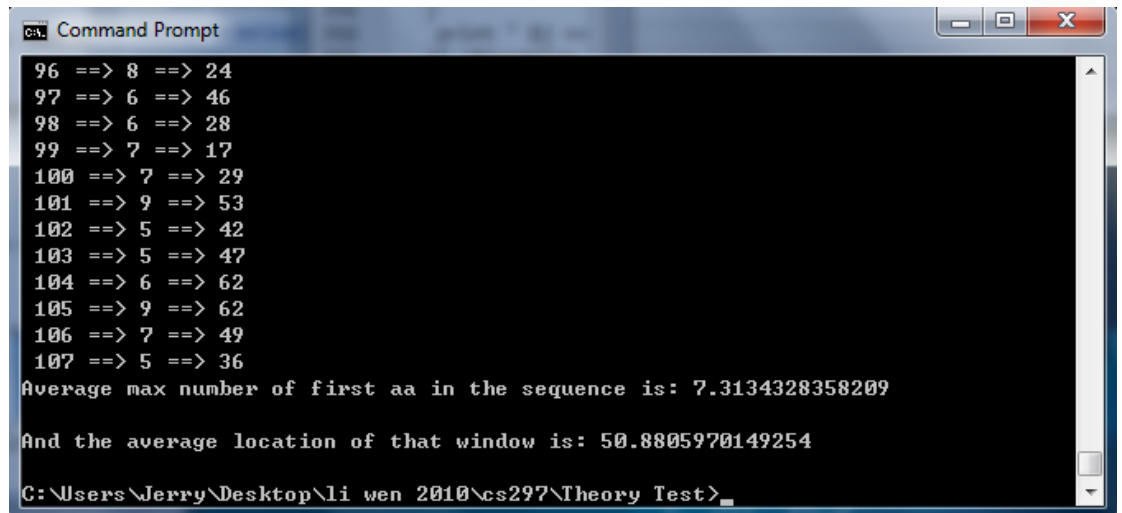
97 1410101112124122220141221202242241242421411101412212002122224204241
98 11112210101222110241242221241221211121112121122112111120111222
99 1214112121211121104111021212102411212211010122210014121214442022221
100 22042212012212121421121111201112112222211222212121221121211221121
101 2220102202121212201244122441222221221111112144111111110002111211
102 211214142222112220222211212122421124112410121222120101224224112121
103 24222421244121124210222114422114211242212122112142122220241422211
104 2122121220212110421141221122111201212412012444110222421101022242022
105 211214112121244121211411111124211224101141120221122042214222212222
106 212022112120211212111121041211422211122124422212024222044210111112
107 2121002112211424222142240120022111222112202241012211224222202111221
108

```

Figure 9. TATA-less file numerical protein product representation

- 5.3.3 Find the location that is most associated with the first amino acid in

TSS by using the same algorithm and formula as in 5.2.3. Test result on TATA-less file is showed in figure 10 below.



```
C:\> Command Prompt
96 ==> 8 ==> 24
97 ==> 6 ==> 46
98 ==> 6 ==> 28
99 ==> 7 ==> 17
100 ==> 7 ==> 29
101 ==> 9 ==> 53
102 ==> 5 ==> 42
103 ==> 5 ==> 47
104 ==> 6 ==> 62
105 ==> 9 ==> 62
106 ==> 7 ==> 49
107 ==> 5 ==> 36
Average max number of first aa in the sequence is: 7.3134328358209
And the average location of that window is: 50.8805970149254
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

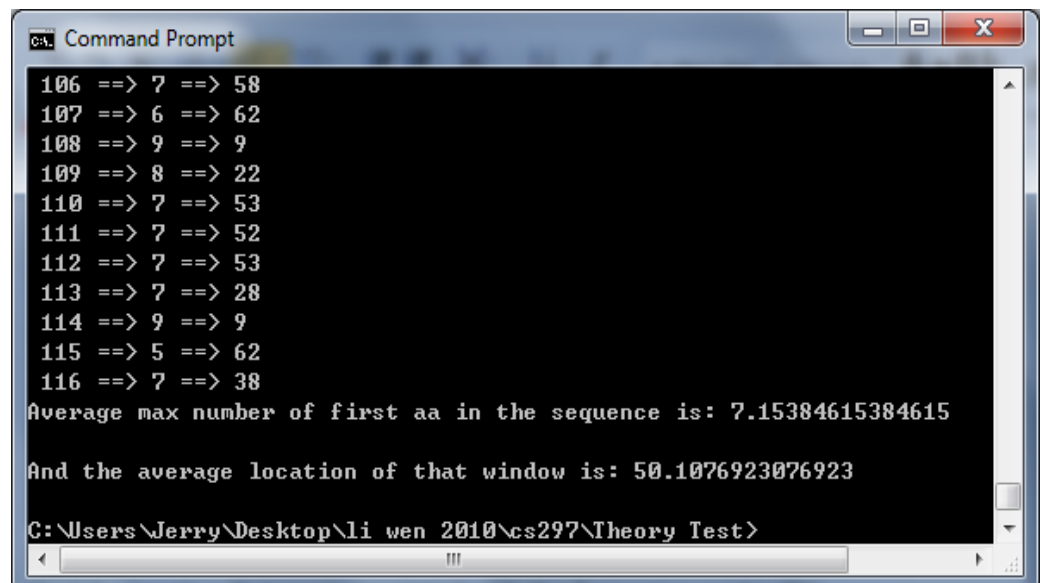
Figure 10. Test result of max appearance and average location in TATA-less file. The average peak in one sequence contains about 73% of the first amino acid of TSS. The peak appears around the 51th amino acid, which is very close to the results in TATA rich file.

5.4 Discussion

Further testing needs to be done by using the five folds method [2]. We randomly select 20% of sequences from TATA-less file, and test if the finding will give the right TSS position.

5.4.1 Test TATA rich sequence by using five folds

5.4.1.1 Five Fold run results



```
C:\> Command Prompt
106 ==> 7 ==> 58
107 ==> 6 ==> 62
108 ==> 9 ==> 9
109 ==> 8 ==> 22
110 ==> 7 ==> 53
111 ==> 7 ==> 52
112 ==> 7 ==> 53
113 ==> 7 ==> 28
114 ==> 9 ==> 9
115 ==> 5 ==> 62
116 ==> 7 ==> 38
Average max number of first aa in the sequence is: 7.15384615384615
And the average location of that window is: 50.1076923076923
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

Figure 11. Randomly choose 80% of data out of 175 TATA rich sequences, and run the same program. The maximum of first aa is about 7.15, and average location is around 50.

```
Command Prompt
105 ==> 7 ==> 45
106 ==> 6 ==> 22
107 ==> 8 ==> 58
108 ==> 9 ==> 39
109 ==> 8 ==> 40
110 ==> 7 ==> 24
111 ==> 7 ==> 54
112 ==> 7 ==> 45
113 ==> 8 ==> 48
114 ==> 8 ==> 25
115 ==> 7 ==> 62
Average max number of first aa in the sequence is: 7.21739130434783
And the average location of that window is: 50.7391304347826
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

Figure 12. Random choose 80% data out of TATA rich sequences again, and run it again. The maximum of first aa is about 7.22, and average location is around 51.

5.4.1.2 Discussion

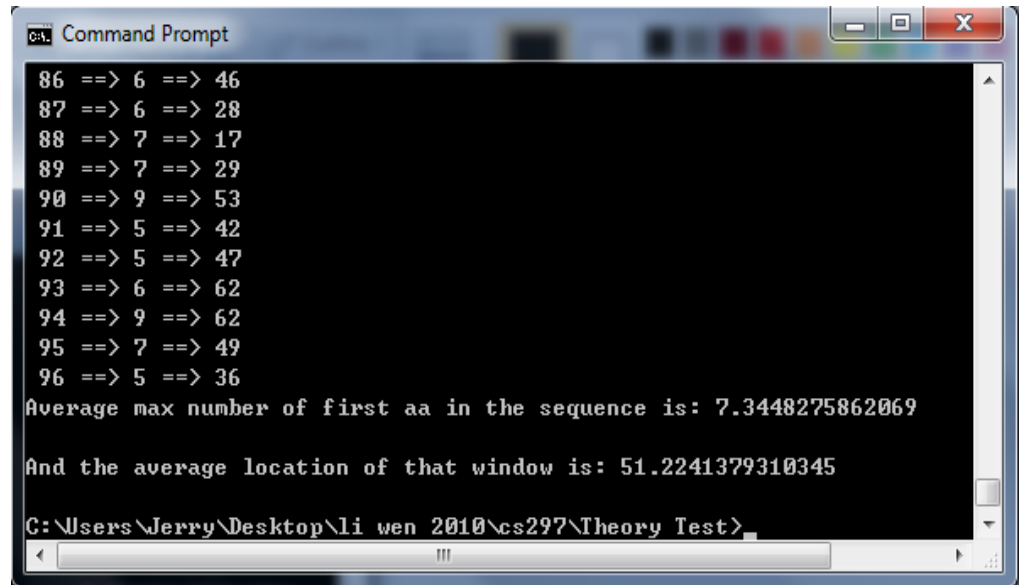
Based on the above two observations by using the five fold method, we can see that the results are similar to what we get by testing all 175 sequences. It tells that the maximum of first aa is about 7, and the peak location is around 50 in TATA rich sequences.

5.4.2 Test TATA less sequence by using five folds

5.4.2.1 Five Fold run results

```
Command Prompt
86 ==> 7 ==> 62
87 ==> 6 ==> 56
88 ==> 8 ==> 29
89 ==> 6 ==> 7
90 ==> 7 ==> 10
91 ==> 8 ==> 50
92 ==> 8 ==> 56
93 ==> 8 ==> 10
94 ==> 8 ==> 22
95 ==> 8 ==> 48
96 ==> 8 ==> 24
Average max number of first aa in the sequence is: 7.30645161290323
And the average location of that window is: 50.5967741935484
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

Figure 13. First run against TATA less sequences by using five fold methods. The maximum number of first aa is about 7.3, and the average location is about 50.6.



```
c:\ Command Prompt
86 ==> 6 ==> 46
87 ==> 6 ==> 28
88 ==> 7 ==> 17
89 ==> 7 ==> 29
90 ==> 9 ==> 53
91 ==> 5 ==> 42
92 ==> 5 ==> 47
93 ==> 6 ==> 62
94 ==> 9 ==> 62
95 ==> 7 ==> 49
96 ==> 5 ==> 36
Average max number of first aa in the sequence is: 7.3448275862069
And the average location of that window is: 51.2241379310345
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

Figure 14. Second run against TATA less sequences by using five fold methods. The maximum number of first aa is about 7.3, and the average location is about 51.2.

5.4.2.2 Discussion

Based on above two tests against TATA less sequences by using Five Fold method, we can see that it shows the similar maximum number of first aa and average location with the results getting from all 130 sequences.

5.4.3 Conclusion

If we use these findings to detect the promoter region in either TATA rich or TATA-less sequences, then the first step is to partition the data in two classes [2] [17]. Sequences with the clear TATA box characteristic will go to the TATA rich class. And sequences without such characteristics will go to the TATA less class. In TATA rich class, use TATA box to find the location for promoter and TSS. In TATA less class, use the finding to detect the promoter region.

Since the finding in both TATA rich and TATA-less file is very close, then no classification will be needed, which means use the finding directly.

Which method will provide the most accurate result? Compare the two results; also compare the results with online tools. For E. coli data that needs to be used later, ORF needs to be found first.

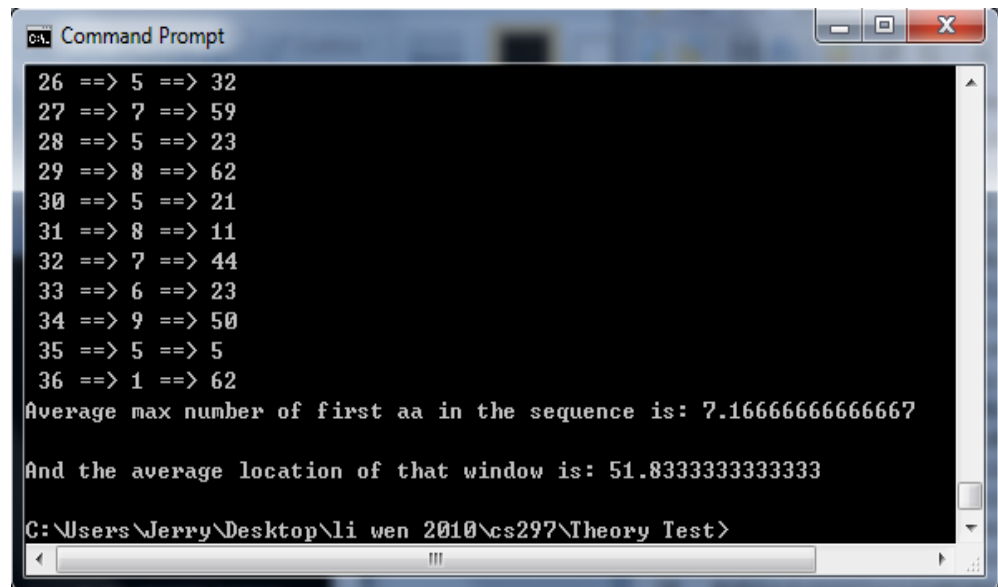
6. Use the finding in section 5 to test the prediction accuracy on both TATA rich and TATA-less files by using five-fold method [2]. Using 4/5 as training data, and the rest 1/5 as test data to test the prediction accuracy.

6.1 Test the finding in TATA rich file

- 6.1.1 Random select 1/5 to be testing data, and 4/5 as training data among

TATA rich file.

- 6.1.2 Input 4/5 data to get the location of the peak value of the first TSS codon. This was covered in section 5.4.1.
- 6.1.3 Use the location to predict the 1/5 data's TSS and compare it to the real TSS of each and get the accuracy. See the test result below by using 20% of given TATA rich data.

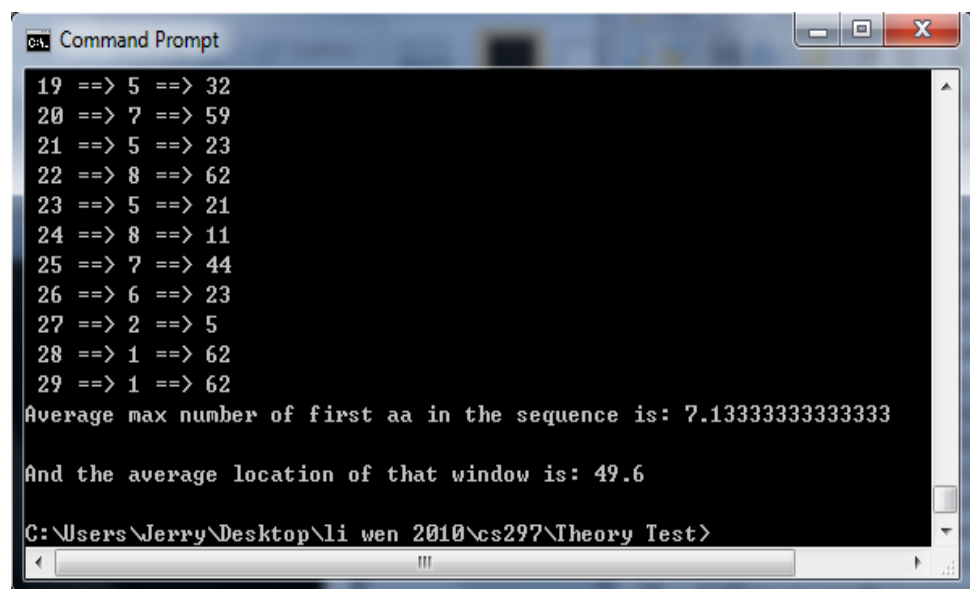


```
Command Prompt
26 ==> 5 ==> 32
27 ==> 7 ==> 59
28 ==> 5 ==> 23
29 ==> 8 ==> 62
30 ==> 5 ==> 21
31 ==> 8 ==> 11
32 ==> 7 ==> 44
33 ==> 6 ==> 23
34 ==> 9 ==> 50
35 ==> 5 ==> 5
36 ==> 1 ==> 62
Average max number of first aa in the sequence is: 7.166666666666667
And the average location of that window is: 51.83333333333333
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

Figure 15. Test result of 1/5 of 175 TATA rich data. It is similar with 4/5 and all TATA rich data. Therefore the maximum number and the average location can be decided as 7 and 50.

6.2 Test the finding in TATA-less file

- 6.2.1 Random select 4/5 of TATA less sequences as training data. Please see the results 5.4.2.
- 6.2.2 Test 1/5 out of 130 TATA less data.



```
Command Prompt
19 ==> 5 ==> 32
20 ==> 7 ==> 59
21 ==> 5 ==> 23
22 ==> 8 ==> 62
23 ==> 5 ==> 21
24 ==> 8 ==> 11
25 ==> 7 ==> 44
26 ==> 6 ==> 23
27 ==> 2 ==> 5
28 ==> 1 ==> 62
29 ==> 1 ==> 62
Average max number of first aa in the sequence is: 7.133333333333333
And the average location of that window is: 49.6
C:\Users\Jerry\Desktop\li wen 2010\cs297\Theory Test>
```

6.2.3 Discussion

The results are close to what we expected, which means we can use the characteristic in real E. coli data from NCBI.

6.3 Conclusion

The characteristic found in both TATA-rich and TATA-less sequence provide similar results as show in this section. The results indicate that gene can be predicted by using first “aa” to detect the start of a gene in about 50 nuclides away from the peak. The algorithm will be discussed in the Algorithm section.

7. Signals or characteristics that can be used for promoter detection.

7.1 Ideas that been using in this project

7.1.1 Inr: finding Inr without DPE will provide the same result with finding Inr with DPE, since TBP will bind Inr with DPE for the lack of TATA box. The consensus sequence of Inr is PyPyAN(T/A)PyPy, where Py is pyrimidine (C or T), N is any base (A, C, G, T), the underline A is TSS [23].

7.1.2 DPE: downstream promoter elements located about 30bp downstream of TSS with consensus sequence G(A/T)CG in Drosophila when there are no TATA box.

7.1.3 Start Codon: in 5-10% of cases, the initiator will pass the first start codon, and use the next one [23 page 539]. A hair-pin before AUG will make this AUG a start codon. This information can be used to justify the start codon location in the late of the process. When AUG at the beginning of mRNA, it is start codon; if AUG in the middle of mRNA, it codes for methionine.

7.1.4 T here are only 15 TFIIBs, each will bind to different sequence in promoter region. If I can find what they bind in TATA less region, then I'll be able to locate promoter region. The chemical reaction of TBP with TATA box is explained in [25].

7.1.5 C C box: upstream of TATA box are GC box with GGGCGG and CCGCCC in -47 to -61 and -80 to -105 region[23].

7.2 Ideas that can be used in for other researchers

7.2.1 Use RNA secondary structure to find correlations in sequences, need more test to support this idea

7.2.2 TBP (TATA binding protein) binds to the minor groove of TATA box, and then other element of TFIID may bind to region without TATA box [23]. Compares to other steps like A-U, or G-C, T-A is much easier to distort to initial the transcription. TBP works on both TATA rich and TATA less promoters; TBP is not TATA sensitive, but temperature sensitive [23]. Substituting C for T and I for A in the sequence will get the same result since the minor groove of C and I is the same as with T and A. “What about the promoter that lack a TATA box?” [23]. TBP will bind to initiators, DPEs, or GC box to secure TFIID's functionality with the help from TAF (TBP associated factors) 150 and TAF 250. According to figure 11.13[23], TBP will find either TATA box, Inr with DPE, or GC box to bind on the sequence; therefore, there will be three clusters with

each.

- 7.2.3 Polymerase melting DNA based on Figure 11.5 can also be used to detect promoter region. Polymerase trying to find the weak connection to melt, for any T-A pair it passed, it will provide 25% damage (from TATA, and ATAT it hits four same base pair, and it melts the double strand). Without TATA box, in any location, if the damage adds up to 1, then it will melt the DNA. For example, in Inr, if the sequence is TTAGTT, then the pairs for Inr will be AATCAA, so the calculation will be $25\%+25\%+25\%-25\%+25\%+25\% = 1$; then the Inr is melt by polymerase. AT rich region that located before TSS in the promoter region is important as CG contents; it acts as enhancer [26]. So AT-rich can be used to locate the promoter without TATA box; or TATA box is just part of AT-rich. In progress, program is partially done.
- 7.2.4 Shine-Dalgarno sequence AGGAGGU [23]: after TSS and before start codon, it will attract ribosomes to the nearby AUG to start translation. Eukaryotes do not have a SD sequence, but use a cap called eIF4E at the 5' end, that help attract ribosomes [23].
- 7.2.5 Testing *E. albertii*: genes are overlapping, for example ealbertii1.txt with gi number 169405087 contains many genes with end location mixed with the start location of next gene. 687-2534, 2518-2724, 2721-4313, etc. And genes predicted are in different reading frames. Try to use start codon + 3n+ end codon, and the next start codon is not +3 but any.
- 7.2.6 Use AT rich as melting point, and CG as looking location. For example, for any given sequence, first search for the melting point by using sliding window algorithm find rich AT region; and then combine the result with CpG island profile in the same sequence to decide if the region is most likely to be the promoter region before TSS.

The D2K Algorithm

Software and algorithms used

I use EGF (*E. coli* Gene Finder, Perl program) as detection tool in this step. The purpose of this research is to explore the use of TATA-less regions for promoter prediction and based on existing algorithms and methods to find an algorithm which performs as well as or better than the existing approaches. As a result of my research and testing of existing algorithms and methods, I discovered an algorithm D2K (double k-mean with $k=2$) which performs better than most of online promoter predictors.

D2K Algorithm depends on the findings of TATA-less regions in Theory Testing section, and implemented in clustering step in this section. Improved promoter prediction

will use the characteristic from both TATA-rich regions and TATA-less regions to increase true positive prediction result. During data mining step, k-mean with k=2 will be used since in this case there are only two classes during data classification. K-mean is a well known clustering algorithm, it partitioning all observation into r clusters where each observation belongs to the nearest mean. D2K indicates that k-mean will be use twice as explained below.

1. First clustering will partition the given sequences into 2 clusters, based on the characteristic of TATA box. Sequences with clear TATA box information are grouped as resolved sequence. It will be discussed in detail in section 3 below.
2. Second clustering process the reminder unresolved sequence in order to detect more promoter regions that do not contain TATA box. By combine both results, EGF use k-mean one more time to get the probabilities of six reading frames. The probability of the highest reading frame will be recorded as detected promoter region. It will be discussed in detail in section 5 below.

Data set used in algorithm step

For Eukaryotes, since promoter contains core promoter (TATA -35, TFIIB—upstream of TATA, Inr, and downstream promoter element—DPE) and upstream promoter element [23], all four elements will be considered. For TATA less promoters, GC box or DPE will appear. Combine both CG content and TATA box information to locate the promoter region in Eukaryotes is the next step of this project.

Based on the clustering algorithm introduced in chapter 16 [2], a sub solution—E. coli Gene Finder (EGF) of promoter prediction is finished on E. coli sequences. For E. coli uses E. coli Gene Finder from cs123b, where -10 (TATAAT) and -35(TTGAC) can be easily found. For any short E. coli sequence from NCBI, the program will predict the location of possible genes.

Steps of detecting promoter region

1. Data preparation. One method that can make the DNA data independent in the prepare data level is PCA (principal component analysis). There are dependencies in DNA analysis, such as the properties of some data may not be truly independent. Another example is some genes are co-expressed. Samples of such data are against the principle of data mining, in which each pieces of data must be independent [17]. One way to minimize the dependency effect is to use PCA—principal component analysis. PCA will transform those data into components, which are independent of each other. New variables will become linear combination of its raw data [18]. PCA usually reduces the raw data principle to two or three components, which contain the most of the variations and ignore others, so those components can be used to classify sample experiments. For example, M sequences with N genes in each will create a matrix $X = N * M$. After passing through PCA, the formula will be changed to

$$X = U \varepsilon V^T$$

Where U is the expression level of every gene, ε is the Ath eigengene that is

expressed in the Ath eigensample, and V^T is the expression level in each sample. Then the data can be plot for each gene/protein pair. The PCA step used in this project will test only one sequence on six frames each run, so $M = 6$ each time a new sequence entered in the program. N is the number of genes that will be predicted in each frame, and N varies in each frame. The matrix will become $X = 6N$. A group of results will be chosen out of six in the clustering step.

2. Distance definitions. There are three distances calculated in our program: the length of gene, the distance between TATA box and TSS, and the distance between -35 element and TSS. In this step we do not use any of Euclidean, Pearson, or Mahalanobis; instead we use the direct distance by finding the difference between two locations.
3. Clustering. Since there are only TATA rich and TATA less two classes in the data set, I am using k-means to classify our data with $K=2$. We partition the data set into two clusters with the number of TATA boxes as classifier. K is the number of clusters and it is fixed when use k-means clustering method. The centroids of clusters are random assigned and then relocated during each cycle, and then finalized when the centroids stop change. To make k-means more accurate, it must be run several times. A similar way to cluster TATA rich and TATA less data will use SVM (supervised clustering with support vector machines). SVM can be used to classify data in one of two classes [17]. After SVM has all the training data, the unknown data will be classified into one group among the training data. Therefore, that unknown data will have the characteristic of the group to which it was assigned. In our program we treat k-mean and SVM in the same way since there are only two final classes.
4. Significance of differential expression. We use this step to evaluate our test results. I compare my test results with several online gene predictors, and provide statistic comparison between the findings. Statistical testing measures to measure true positive (TP) [18], and false discovery rate (FDR) [18]. It will be explained in detail in the discussion section.
5. Improvement after getting first result—the second K-mean clustering process
 - 5.1 The original design can only get less than 80% of gene compare to real gene in NCBI data base. In order to detect more genes from given sequence, the cluster algorithm is refreshed with a second K-mean clustering process to classify the remaining TATA-less group.
 - 5.2 The original cluster decision was made by using 175 TATA-rich plan sequences, and 135 TATA-less plan sequences as training data.
 - 5.3 Both k-mean and DBSCAN [17] are used to predict more location of gene during this step, since only testing can tell which algorithm will give a better result. However, they both do not predict more gene in this step since the number of object is too small for cluster.
 - 5.4 There are six reading frames need to be tested for each given sequence. Using k-mean will get the probability of each reading frame. The reading frame with the highest probability is recorded as final predict result.

Implementation

Steps to research non-TATA:

1. Get E. coli sequences from NCBI
2. Design: Structuring the data requires translating data to its RNA form. This gene predictor follows the tradition gene finding process with additional clustering and statistical methods from chapter 16 [1]. It first gets the RNA forms of a given data, finds all orfs from six reading frames by using PCA, then EGF decides which orfs are real genes, depending on some of the consensus sequence characters. The last step is to choose one frame out of six frames by using the cluster algorithm to pick one with the highest probability. To show how accurate the result is, we compare the detected genes with NCBI, EasyGene, and GenScan in two categories: true positive and false discovery rate. Result and discussion depends on the genes find by all four packages.
3. Bioinformatics Analysis: To choose the best frame out of six is a statistic process based on the characteristic of a real gene. For example, one way to say the finding is a real gene is to find a TATA box in its promoter region. The sum of the number of resulting genes with TATA box before TSS will be calculated to decide the probability of this frame is the best among others.

Produce for testing D2K

1. Construct the table of findings with the probability on the same data promoter prediction data.
2. Choose several findings with the highest value to build a profile.
3. Put the findings in step 2 in the program.
4. Test the same data in EGF and some online popular tools, such as EasyGene, Genscan, etc.
5. Compare the results with discussion.

Integrating D2K into a web application

1. Choose the right platform
As recommended by Dr. Tseng, this project will use LAMP (Linux, Apache, MySQL, PHP) platform to perform the result of detection. In addition, the Zend framework will be used to minimize the amount of code need to be implemented. I will process data using EGF and excel then extract the data using PHP to get real data from EGF, and partial data from excel.
2. Design UI: user can select several available accession numbers from left menu to see what has been predicted, and also have the chance to see the prediction from other online prediction, such as Virtual footprint, EasyGen, etc. It will show each gene's start and stop location in NCBI, Glimmer and other above online tool. It also displays the start and stop gene location that predicted by using EGF. User can easily compare the predicted results. TP and FDR will show the standing of EGF among other promoter predictors.
3. Coding: PHP, Perl, HTML, XHTML, JavaScript, JQuery, YahooSiteBuilder, etc.

3.1 Combine LAMP with YahooSiteBuilder

The original design is to try to get result from other online web gene finder and put the result on one web page for any given E. coli sequence. However, I have spent some time on each of the online gene predictor; I recognized that it is hard to accomplish this goal (within one semester). For example, Virtual footprint is PHP based software suite, it does not support automatic promoter analysis. The user has to go through several steps to get the result of a given sequence. For EasyGen, the predicted result may come from email instead of instance result. Therefore, it is not effective to implement an automatic web result compare tool. To fully support the main origin of this paper, it is redesigned as a web representation tool, to support the main goal of this research.

3.2 YahooSiteBuilder with Excel

Since most result were processed by using excel, displaying data from excel to a web browser made it more user friendly. To achieve this display, I insert an iframe inside YahooSiteBuilder page. This iframe can upload an excel and display it on a web browser.

4. Testing: using accession number from section “Result and Conclusion”

5. Improving:

5.1 Adding more data from NCBI to test

5.2 Choose a nice layout for each sequence

5.3 Make an index page with

5.3.1 Links to each sequence result page

5.3.2 Short summary of this paper

5.3.3 Purpose of this web implementation

5.3.4 How to use, etc.

Result

Twenty different E. coli Data are tested, and genes locations are listed. Start and Stop is the result of E. coli Gene Finder. The last row is the number of genes found in each. Please see appendix 4 for detail running result of each promoter predictor.

Result

1. Accession number 300901746

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
		38	4363								
4383	5129	4383	5129			86	4363	89	6890		

5188	6048	5188	6048	5188	6048	5614	6048			5470	5483
6151	6711	6151	6711	6151	6711	6151	6711				
6840	7052	7176	6922	6856	7098						
7285	7758	7222	7758	7222	7758			7271	8893	7439	7452
7804	8013	7804	8013	7783	8013					7592	7605
8051	8641	8051	8641	8212	8409	8153	8641				
8881	9141	8881	9141	8881	9141			8903	8942		
9429	10298	9429	10298	9574	9804						
		10450	10310								
		23	10717								
9		9		8		4		3		3	

Table 1: Result of gene search for 300901746

2. Accession number 403342

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
278	1372	278	1372	278	1372	431	1372	309	5512	66	79
1396	1785	1396	1785			1396	1785			975	988
1904	4777	1904	4777	1904	4777	1988	4777			4202	4215
3		3		2		3		1		3	

Table 2: Result of gene search for 403342

3. Accession number 325965637

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
138	1580	138	1580					254	2669	1197	1210
1592	2515	1592	2515	1592	2515					1600	1613
2564	3646	2624	3646	2564	3646			2672	4851		
3661	4644	3661	4644			3730	4644				
4823	5935	4751	5935	4751	5935			4868	6091	5390	5403
5		5		3		1		3		3	

Table 3: Result of gene search for 325965637

4. Accession number 346421495

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
1	546	88	210	81	323			69	516		
1		1		1		0		1		0	

Table 4: Result of gene search for 346421495

5. Accession number 260765442

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
1	1026	0	0	1	1026	232	1026	66	1079		
1		0		1		1		1		0	

Table 5: Result of gene search for 260765442

6. Accession number 167509193

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
1	519	128	498	174	392			50	1557	272	285
1		1		1		0		1		1	

Table 6: Result of gene search for 167509193

7. Accession number 354515243

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
1	253							214	92	17	26
1		0		0		0		1		1	

Table 7: Result of gene search for 354515243

8. Accession number 354515242

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
225	796			141	788	225	788	332	855	679	692
1		0		1		1		1		1	

Table 8: Result of gene search for 354515242

9. Accession number 354515240

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
104	808			35	277			204	874	257	266

				542	736					381	390
										418	427
1		0		2		0		1		3	

Table 9: Result of gene search for 354515240

10. Accession number 354515237

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
1	458			67	264	9	458	705	920	251	260
589	818	589	34	322	573					397	406
										451	460
2		1		2		1		1		3	

Table 10: Result of gene search for 354515237

11. Accession number 145467

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
				9	386			123	395	267	280
450	1688	489	1688	450	1688			564	1714	406	419
								2024	2191	1682	1695
1		1		2		0		3		3	

Table 11: Result of gene search for 145467

12. Accession number 342315677

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
										145	154
										224	233
1	674			322	621			400	685	351	360
1		0		1		0		1		3	

Table 12: Result of gene search for 342315677

13. Accession number 341941295

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop

1	1299			1	1299			132	1361	572	585
1		0		1		0		1		1	

Table 13: Result of gene search for 341941295

14. Accession number 41745

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
227	1621	227	1621	227	1621			280	1645	900	909
1		1		1		0		1		1	

Table 14: Result of gene search for 41745

15. Accession number 41727

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
1971	2459	1971	2459	1971	2459	2064	2459	1998	2037	1908	1971
1		1		1		1		1		1	

Table 15: Result of gene search for 41727

16. Accession number 41592

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
300	1505	300	1505	300	1505	432	1505	319	1524	1575	1588
1		1		1		1		1		1	

Table 16: Result of gene search for 41592

17. Accession number 41580

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint Stop
31	1107			31	1107	31	1107	132	1181	959	972
1		0		1		1		1		1	

Table 17: Result of gene search for 41580

18. Accession number 414745

NCBI start	NC BI stop	Glim mer Start	Glim mer Stop	Start	Stop	EasyGene start	EasyGen e stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtu al Footp rint
------------	------------	----------------	---------------	-------	------	----------------	----------------	---------------	--------------	-------------------------	---------------------

											Stop
334	561			334	561			605	2881	632	641
578	2899	821	2899	2890	3135	821	2899	2944	3146	2992	3001
2		1		2		1		2		2	

Table 18: Result of gene search for 414745

19. Accession number 312761

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
31	474			31	474	31	474			178	187
502	1884			502	1884	742	1884	502	1911	1354	1363
								1958	1963	2067	2076
2		0		2		2		2		3	

Table 19: Result of gene search for 312761

20. Accession number 297393

NCBI start	NCBI stop	Glimmer Start	Glimmer Stop	Start	Stop	EasyGene start	EasyGene stop	Genscan start	Genscan stop	Virtual Footprint Start	Virtual Footprint Stop
										27	36
721	1905	492	728	721	1905	721	1905	759	1997	984	993
										1120	1129
1		1		1		1		1		3	

Table 20: Result of gene search for 297393

Discussion:

TP is true positive which means the gene is predicted when there is a gene, the bigger value the better result. And the formula is

$$TP = \text{number of genes predicted} / \text{total genes in NCBI of this sequence}$$

FDR is false discovery rate which provides the rate of false location in prediction, the smaller the value the better. And the formula is

$$FDR = \text{total location shift predicted} / \text{number of genes predicted}$$

Accession Number	Glimmer TP	Glimmer FDR	EGF TP	EGF FDR	EasyGene TP	EasyGene FDR	Genscan TP	Genscan FDR	VFP TP	VFP FDR
300901746	1	58.77778	0.88889	147.25	0.444444	1397.75	0.333333	2785.667	0.333333	642.3333
403342	1	0	0.66667	0	1	79	0.333333	4171	1	1580.3333
325965637	1	0	0.6	24	0.2	69	0.6	906.333	0.6	1146

								3		
346421495	1	136	1	110 3	0	1103	1	38	0	1103
260765442	0	1027	1	0	1	231	1	118	0	1027
167509193	1	106	1	46	0	520	1	1087	1	37
354515243	0	254	0	254	1	254	1	52	1	211
354515242	0	1021	1	92	0	8	1	166	1	350
354515240	0	912	2	678	0	912	1	166	3	1227
354515237	0.5	1243	1	640	0.5	1399	0.5	241	1.5	359
145467	1	909	2	126 5	0	1268	3	5743	3	3481
342315677	0	675	1	268	0	675	1	410	3	792
341941295	0	1300	1	0	0	1300	1	193	1	143
41745	1	0	1	0	0	1848	1	77	1	39
41727	1	0	1	0	1	93	1	395	1	551
41592	1	0	1	0	1	132	1	38	1	1358
41580	0	1138	1	0	1	0	1	175	1	793
414745	1	5652	2	245 2	1	5652	2	204	2	2106
312761	0	2891	1	0	1	240	1	3443	1.5	4334
297393	1	1406	1	0	1	0	1	130	3	1663

Table 21: Result for TP and FDR comparison

From the above table we can see that EGF predicts most of the real genes with a lower false discovery rate among the other predictors.

Charts of above data

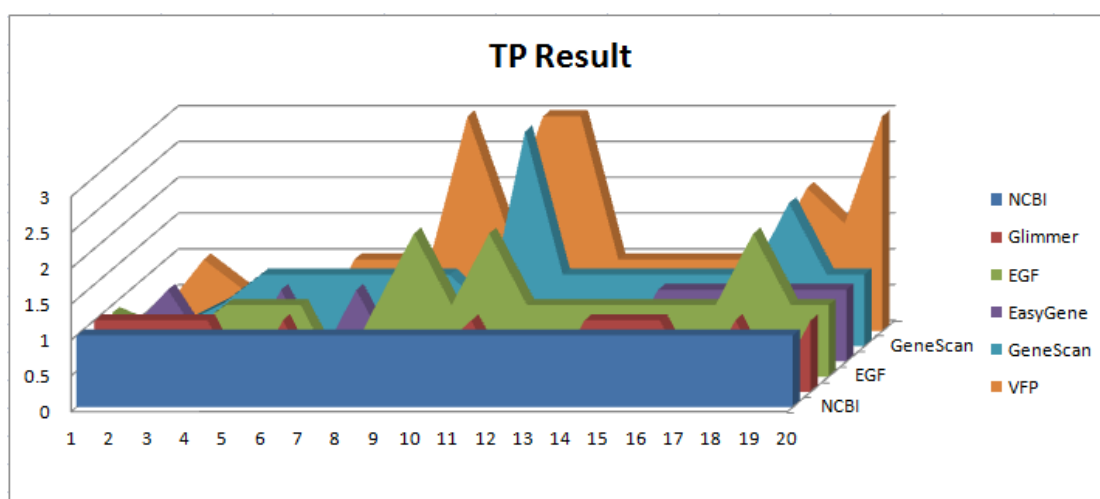


Figure 1: TP result. X axis is the 20 different genes, and y axis is the number of genes predicted/ gene in NCBI database. NCBI results are all 1s, since they are the number of genes in the database. For other predictors, some predicted more than expected, and some are less.

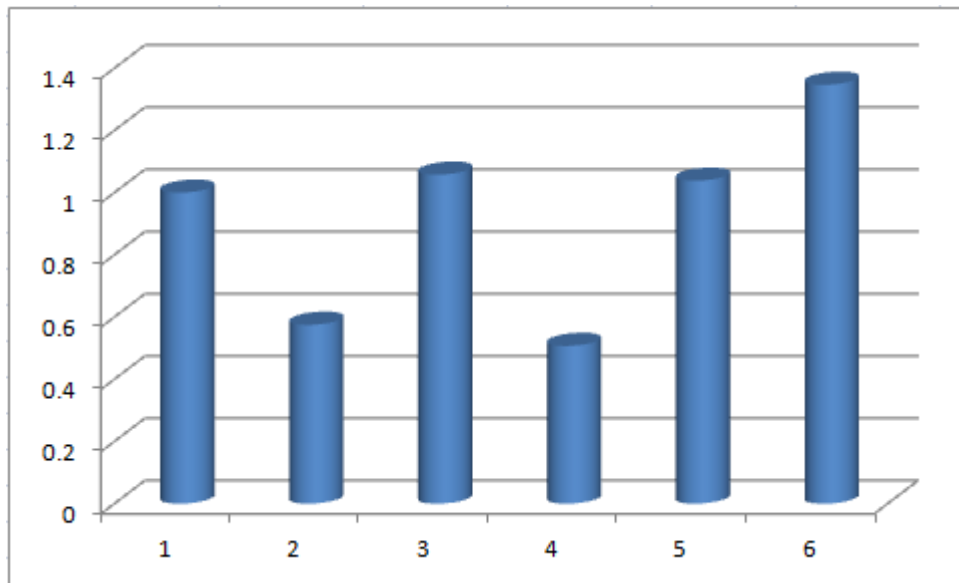


Figure 2: TP result average. $TP\ Average = TP\ result / \text{number of experiments (20)}$. To see the performance of each predictor, it is better to use an averaged data for comparison. Series 1 is the default genes existed in NCBI database, Series 2 is the percentage of genes found by Glimmer, Series 3 is the percentage of genes found by using E. coli Gene Finder, Series 4 is the percentage of genes found by EasyGene, and Series 5 is the percentage of genes found by GenScan, Series 6 is the percentage of genes found by Virtual Foot Print. In this case, the closer the better; therefore EGF and GenScan are more closer to NCBI database.

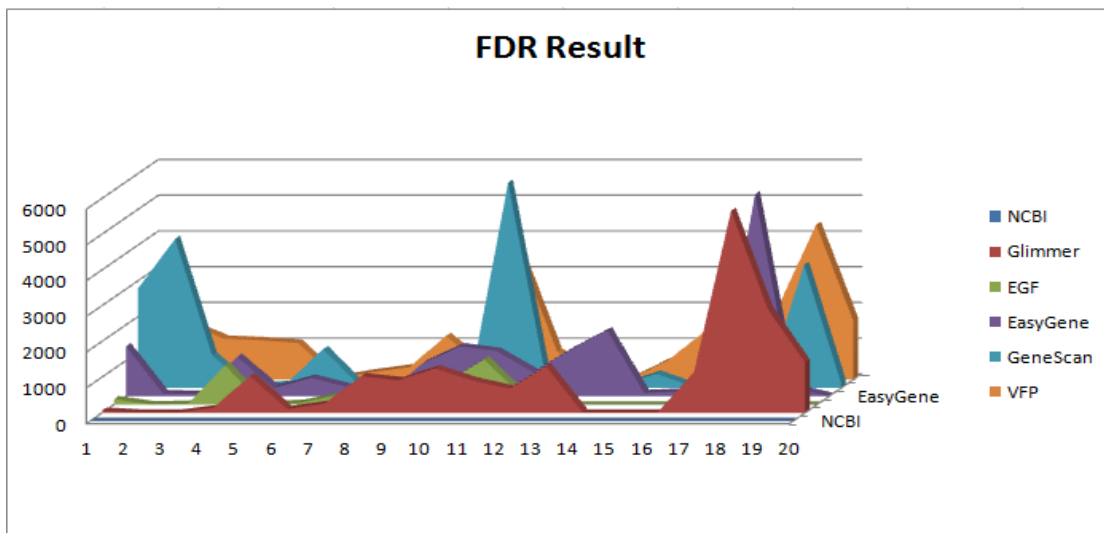


Figure 3: FDR Result. $FDR = \text{sum of number of position shift} / \text{number of gene predicted}$. X axis is the 20 genes used in this project; Y axis is the number of false gene locations. The data represent the difference between the predicted gene location with the gene location in the database, therefore the smaller the better.

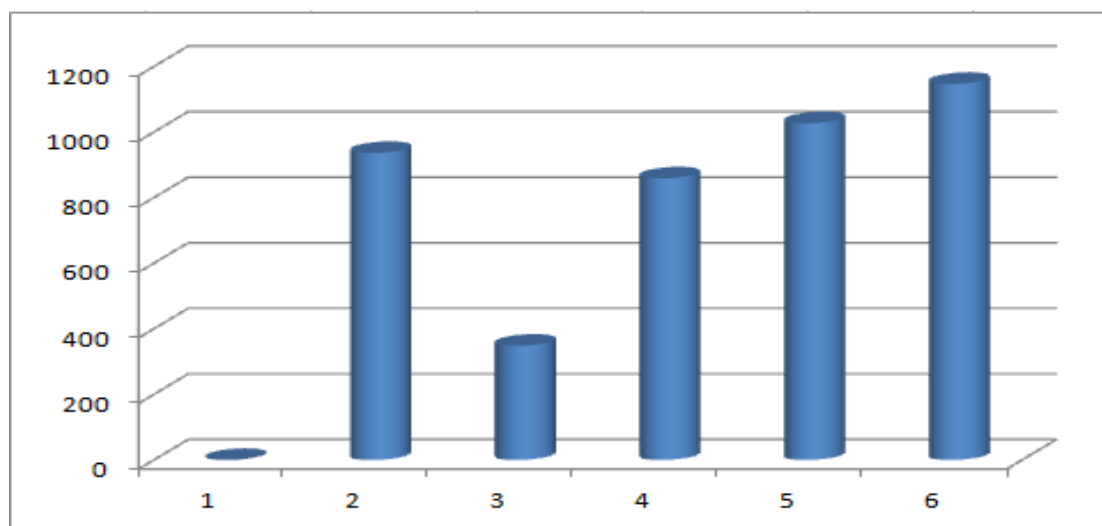


Figure 4. FDR average. FDR Average = sum of FDR/ total number of predicted gene (20). Series 1 is the default false gene location in NCBI which is 0, Series 2 is the miss calculated gene location by Glimmer, Series 3 is the false predicted gene location by EGF, Series 4 is the false gene location that was predicted by EasyGene, Series 5 shows the wrong gene locations as predicted by Genscan, and Series 6 is the false prediction by Virtual Foot Print. EGF predicts the less FDR among other predictors.

Final results can be improved by checking the start codon condition. For example, if there is a hairpin structure before AUG; then this AUG will be most likely the start of a gene. Or, if there is another AUG just few codon after a start codon; then the start codon will be passed and this AUG will become the start of a gene [24]. This step can be done to either locate promoter phase or improve the result phase.

Related topics:

Cancer: the cancer is caused not only by a single mutation, but several mutations on the chromosome. It can be explained by the exponential growth of some cancer cause death growth with age. One example is the colon cancer death raise in “One Renegade Cell” page 47 [27]. If colon cancer is caused by 3 mutations, then the formula will be 2^3 . If it takes about 2 years for one mutation to happen, then after 16 ($2^3 \times 2$) years, someone who has all the mutations will develop colon cancer. If it caused by 4 mutations, then the time needed are 32. Off cause, mutations that happened not related to colon cancer will result in no colon cancer even when more than 4 mutations are detected in one sequence. To prevent colon cancer or any other cancer, couples of analysis need to be done. First, all mutations required by colon cancer need to be defined. Second, use examples (someone who has colon cancer family history, does not have colon cancer yet), and find out the difference. Third, prevent the last (one or more) mutation from happening by providing some treatment or medicine.

Conclusion

The prediction made by this program works only for some E. coli sequences as we can see from the data from NCBI. Compare each predicted gene with NCBI, EGF predicts more real gene than other online predictor, and it also gives less wrong prediction than other predictor. Even though the results look better than some of the online packages prediction, it still has the limitation on predictions. For example, it can't predict short gene (length less than 60 nucleotides) and overlapping gene (gene inside gene). Those genes do exist in NCBI database which include the E. coli gene I used in this paper. It is just the start point of this project by using the basic characteristic that found in E. coli; further study on other related organisms needed to broaden the search power of the system. After all the ideas have being finalized in section "steps to find characteristics 7", then the prediction will be enhanced by adding more idea in the original program.

Reference

1. Caroline St. Clair and Jonathan Visick: Exploring Bioinformatics 2010
2. Marketa Zvelebil and Jeremy O. Baum: Understanding Bioinformatics 2008
3. Berkeley Drosophila Genome Project <http://www.fruitfly.org/about/index.html>
4. Ramana V. Davuluri, Ivo Grosse, Michael Q. Zhang: Computational identification of promoters and first exons in the human genome. Nature Genetics (2001) 29, 412--417. <http://rulai.cshl.edu/tools/FirstEF/Readme/README.html>
5. S. Burden, Y.-X. Lin and R. Zhang: Improving promoter prediction. Bioinformatics 2005 21(5) 601-607. <http://bioinformatics.oxfordjournals.org/content/21/5/601.full>
6. Yanglan Gan, Jihong Guan, and Shuigeng Zhou: A pattern-based nearest neighbor search approach for promoter prediction using DNA structural profiles. Bioinformatics (2009) 25 (16). <http://bioinformatics.oxfordjournals.org/content/25/16/2006>
7. PlantProm DB: Bioinformatics WEB server. A Database of Plant Promoter Sequences. <http://mendel.cs.rhul.ac.uk/mendel.php?topic=plantprom>
8. Charles Bland, Abigail S Newsome and Aleksandra A Markovets: Promoter prediction in E. coli based on SIDD profiles and Artificial Neural Networks. BMC Bioinformatics 2010, 11(Suppl 6):S17. <http://www.biomedcentral.com/1471-2105/11/S6/S17>
9. Thomas Abeel, Yvan Saeys, Pierre Rouz , and Yves Van de Peer: ProSOM: core promoter prediction based on unsupervised clustering of DNA physical profiles. Bioinformatics. 2008 July 1; 24(13): i24-i31. <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2718650/?tool=pubmed>
10. Thomas Abeel, Yvan Saeys, Eric Bonnet, Pierre Rouz , and Yves Van de Peer:

- Generic eukaryotic core promoter prediction using structural features of DNA. *Genome Res.* 2008. 18: 310-323. <http://genome.cshlp.org/content/18/2/310.full>
11. Wenjie Song, Paul J. Maiste, Daniel Q. Naiman, Mandy J. Ward: Sigma 28 promoter prediction in members of the Gammaproteobacteria. *FEMS Microbiology Letters* Volume 271, Issue 2, pages 222–229, June 2007. <http://onlinelibrary.wiley.com/doi/10.1111/j.1574-6968.2007.00720.x/full>
 12. Thomas Abeel, Yves Van de Peer, and Yvan Saeys: Toward a gold standard for promoter prediction evaluation. *Bioinformatics* (2009) 25 (12): i313-i320. <http://bioinformatics.oxfordjournals.org/content/25/12/i313.full>
 13. Rajeev Gangal and Pankaj Sharma: Human pol II promoter prediction: time series descriptors and machine learning. *Nucleic Acids Res.* 2005; 33(5): 1739. <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1069632/>
 14. Xiaowo Wang, Zhenyu Xuan, Xiaoyue Zhao, Yanda Li1 and Michael Q. Zhang: High-resolution human core-promoter prediction with CoreBoost_HM. *Genome Res.* 2009. 19: 266-275. <http://genome.cshlp.org/content/19/2/266.full>
 15. Amino Acids table
http://www.bioss.ac.uk/~dirk/genomeOdyssey/go_1955_to_66.html
 16. RNA codon chart
<http://www.ccs.k12.in.us/chsteachers/BYost/Biology%20Notes/translationnoteguide.htm>
 17. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, *Introduction to Data Mining*, 2006 by Pearson Education, Inc.
 18. Ronald P. Cody, and Jeffery K. Smith, *Applied Statistics and the SAS programming Language*, 2006 by Pearson Education, Inc.
 19. Frederick J. Gravetter, and Larry B. Wallnau, *Essential of Statistics for the Behavioral Sciences*, 2008 by Wadsworth Publishing.
 20. Basic MOLECULAR BIOLOGY
http://preuniversity.grkraj.org/html/10_MOLECULAR_BIOLOGY.htm
 21. TATA-binding protein http://en.wikipedia.org/wiki/TATA-binding_protein
 22. TFIID <http://en.wikipedia.org/wiki/TFIID>
 23. Christopher B Burge, and Samuel Karlin, Finding the Genes in Genomic DNA, *Current Opinion in Structural Biology* 1998, 8:346-354
 24. Robert F. Weaver, *Molecular Biology*, by McGraw-Hill Inc. 2008
 25. Wikipedia, <http://en.wikipedia.org/wiki/Escherichia>
 26. Kube D, Laser H, von Knethen A, Tesch H, The AT-rich region between -54 to -66 is important for the promoter activity of interleukin-10 in Epstein-Barr virus positive Burkitt's lymphoma cell. *PubMed* PMID: 11196656
<http://www.ncbi.nlm.nih.gov/pubmed/11196656>
 27. Robert A. Weinberg, *One Renegade Cell*
 28. O'Reilly, *HTML and XHTML—The Definitive Guide*, 2002 by RepKover
 29. Marc Wandschneider, *Core Web Application Development With PHP and MySQL*, 2006 by Pearson
 30. NCBI: National Center for Biotechnology Information
<http://www.ncbi.nlm.nih.gov/>

31. Glimmer: <http://www.cbcb.umd.edu/software/glimmer>
32. EsayGene: <http://www.cbs.dtu.dk/services/EasyGene/>
33. Genscan: <http://genes.mit.edu/GENSCAN.html>
34. Virtual Footprint: http://www.prodoric.de/vfp/vfp_promoter.php

Appendix 1—Training Data

1. 175 TATA rich sequences

http://mendel.cs.rhul.ac.uk/pprom/PLPR_TATA.seq

2. 130 TATA less sequences

http://mendel.cs.rhul.ac.uk/pprom/PLPR_TATA-less.seq

Appendix 2—List of NCBI Accession Number of Testing Data

1. `ecoli41580`
2. `ecoli41592`
3. `ecoli41727`
4. `ecoli41745`
5. `ecoli145467`
6. `ecoli297393`
7. `ecoli312761`
8. `ecoli403342`
9. `ecoli414745`
10. `ecoli167509193`
11. `ecoli260765442`
12. `ecoli300901746`
13. `ecoli325965637`
14. `ecoli341941295`
15. `ecoli342315677`
16. `ecoli346421495`
17. `ecoli354515237`
18. `ecoli354515240`
19. `ecoli354515242`
20. `ecoli354515243`

Appendix 3—D2K Algorithm Coding in EGF

This program take any E.coli sequence from NCBI and find genes from it

```
#!/bin/perl
if(!open(infile, 'ecoli356875267.txt')){
if(!open(infile, 'ecoli297393.txt')){
    print "error opening input file\n";
    exit;
}
if(!open(outfile, '>out.txt')){
    print "error opening output file\n";
    exit;
}
if(!open(outfile1, '>out1.txt')){
    print "error opening output file\n";
    exit;
}
}
$data = <infile>;    #ignore FASTA comment
while ($data = <infile>){
    chomp $data;
    $seq = $seq . $data;
}

# $seq is the nontemplate strand from the 5' end
# first three reading frames come from $seq

# other three reading frames come from the reverse complement
$complement = $seq;
$complement =~ tr/ACGTacgt/TGCAtgca/; # complement of strand
$reversecomplement = reverse($complement); # reverse of compement
$reversecomplement =~ s/T/U/g; # convert to RNA

$seq =~ s/T/U/g; # convert to RNA

# find ORF in original sequence
$foundorf = 0;

# find ORF in reversecomplement
$stop = 0;

print "seq strand\n";
$seq_result = findORF($seq);
```

```

print "\n\n reversecomplement strand\n";
$reversecomplement_result = findORF($reversecomplement);

#print "reverse complement result: $reversecomplement_result";
#print "seq result: $seq_result";

if (($reversecomplement_result < 0) && ($seq_result < 0)){
    print outfile "ORF not found\n";
}
#find orf
sub findORF{
    my($seq) = @_ ;
    $found = 0;
    # set the starting position of the reading frame
    for($frame = 0; $frame < 3; $frame++){
        $start = $frame;
        $ataCount = 0;
        #$missingStart = substr($seq, 4383, 3);
        #$missingStop = substr($seq, 5129-3, 3);
        #print "missing gene: $missingStart--$missingStop";
        print "\n frame:  ";
        print $frame;
        print "\n";
        $findPromoter = 0;#one promoter per sequence
        while ($start < length($seq)){

            # find start codon in reading frame
            $start = findStartStop($seq, $start,0);
            if ($start == -1) {last;}
            #print "start: $start ";
            #print "\n";
            # look for stop codon at least 60 codons out or 180nt
            if (($start != -1) && ($start+180<=length($seq)-3)){
                $stop = findStartStop($seq, $start,1);
                # print " stop: $stop";
                #print "\n";
                if ($stop >= $start+180){ # length of the gene > 60 codon

                    # $totalLength = length($seq);
                    $realStart = ($start +1);
                    $realStop = ($stop +3);

                    #$startString = substr ($seq, $realStart, 3);

```

```

#$stopString = substr ($seq, $realStop-3, 3);
print "start-stop $realStart--$realStop \n"; # $startString---$stopString \n";
print outfile1 "$frame start-stop $realStart---$realStop \n";
$tataLocation = findTata($seq, $realStart);
if ($tataLocation != -1) {$tataCount = $tataCount +1;}
    $found = 1;
    if ($foundorf == 1){
        print outfile "\n ---Next--\n";
    }
    $foundorf = 1;
    # printed assuming first position is 1
    print outfile "ORF found in reading frame ", $frame+1, " Start Loc: ",
        $start+1, " Stop Loc: ", $stop+1, " and Shine-Dalgarno is
found or not (-1): ", findShine($frame, $start, $seq), ". \n";
        print outfile substr($seq, $start, $stop+3-$start);
    #print promoter info
    if (findPromoters($seq, $start)==1){
        print outfile "\n ORF supported by promoters\n";
        $findPromoter = 1;
    }
    else{

        if ($findPromoter = 0){
            print outfile "\n ORF not supported by promoters\n";
        }
        else {
            print outfile "\n ORF is in the operon\n";
        }
    }

    if ($stop != -1) {$start = $stop;}
    }
    $start = $start + 3; # use $start + 3 in E. coli, and $start -15 in E.
albertii

    }
    $tataCount = $tataCount/6;
    print "\n Probability of using frame $frame is $tataCount.\n";
    print outfile1 "\n Probability of using frame $frame is $tataCount.\n";

}
if ($found == 0){return(-1);}
elseif ($found == 1){return(1);}

}

```



```

sub findStartStop{ #combine find start and stop of orf
  my($seq, $start, $choice) = @_ ;
  for ($i=$start; $i<=length($seq)-3; $i+=3){
    if ($choice){      #start
      if ((substr($seq, $i, 3) eq "UAA")
          || (substr($seq, $i, 3) eq "UAG")
          || (substr($seq, $i, 3) eq "UGA")){
        return($i);
      }
    }
    else {      #stop
      if ((substr($seq, $i, 3) eq "AUG")
          || (substr($seq, $i, 3) eq "GUG")
          # || (substr($seq, $i, 3) eq "UGA")
          || (substr($seq, $i, 3) eq "CUG")
          || (substr($seq, $i, 3) eq "UUG")){
        return($i);
      }
    }
  }
  return(-1);
}

```

```

sub findShine{ #find shine-dalgarno sequence
  my($frame, $start, $seq) = @_ ;
  #print "frame ", $frame+1, " \n";
  $position = ($start -5)*3 + 2;
  $string = substr($seq, $position, 7);
  #print "\n String is >>> $string <<< \n";
  #print "\n String is >>> $seq <<< \n";
  return index($string, "AGGAGG");
}

```

```

sub findPromoters{ #find promoter
  my($seq, $orfstart) = @_ ;

  # modified to work with exercise 1 program
  $element35 = "UUGACA";
  $element10 = "UAUAAU";

  # initialize the search position for the -35 element
  $search35 = 0;

```

```

# check distance from translational start site
while ($search35 < $orfstart-85){
    if (index(substr($seq, $search35, length($element35) +10) , $element35)> -1){
        # element -35 found, initialize search position for -10 element
        $elementdist = 15;
        $search10 = $search35 + length($element35) + $elementdist;
        while ($search10 < $orfstart-60 && $elementdist <= 19){
            if (index(substr($seq, $search10, length($element10)+10) ,
$element10)> -1){
                # valid -10 element found, return success
                return (1);
            }
            # continue searching for -10 element
            $search10++;
            $elementdist++;
        }
        # -35 element not found, continue searching
        $search35 = $search35 + length($element35);
    }
    else{
        # -35 element not found, continue searching
        $search35++;
    }
}
# valid promoters not found - return 0
return(0);
}

```

```

sub findTata{ #find TATA box
    my($seq, $start) = @_ ;
    $tataRegion = $start - 20;
    $string = substr($seq, $tataRegion, 20);
    #print " tata-->$string $tataRegion ";
    #print index($string, "UAUU"); #find TATA box location

    return index($string, "UAUU");

}

```

```

close (infile);
close (outfile);
close (outfile1);

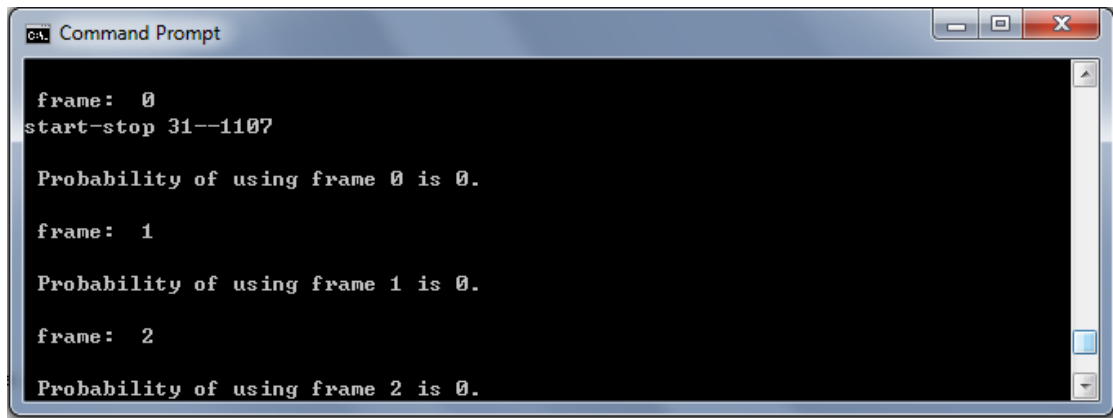
```

Appendix 4—Result Comparison

1. ecoli41580

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:
orfID start end frame score
----- -



```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname            model    feature start    end        score    +/-    ?    startc    odds
# -----
gi_41580_emb_X56907.1_ AP02    CDS       31       1107    2.51355e-10    +    0    #ATG    47.3481
#
```

```
Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Sngl +    132    1181 1050   2   0    74    35   1128 0.280 103.26

Suboptimal exons with probability > 1.000
```

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	282	291	+	6.36	TGTTACTGAT
ArcA Escherichia coli (strain K12)	789	798	+	6.23	GGTTAATTAC
ArcA Escherichia coli (strain K12)	1108	1117	+	6.08	TGTTACTGCA
ArgR Escherichia coli (strain K12)	959	972	+	9.00	TGCATCCTTATACC
ArgR Escherichia coli (strain K12)	260	273	+	8.45	TGCATGACCATTAC
ArgR Escherichia coli (strain K12)	117	130	-	8.35	TGACTATTTTTTCG

2. ecoli41592

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
>gi 41592 emb X17499.1 E. coli gltS gene				
orf00001	308	129	-3	0.65
orf00002	300	1505	+3	3.46
orf00005	44	1674	-3	2.21

```

Command Prompt
frame: 1
start-stop 1187--1384

Probability of using frame 1 is 0.

frame: 2
start-stop 300--1505

Probability of using frame 2 is 0.

reversecomplement strand
  
```

```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?    startc  odds
# -----
gi_41592_emb_X17499.1_ AP02  CDS      432     1505     1.24302e-09  +      0      #GTG    38.9893
#
  
```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Sngl + 319 1524 1206 0 0 41 41 1188 0.450 103.76

Suboptimal exons with probability > 1.000
  
```

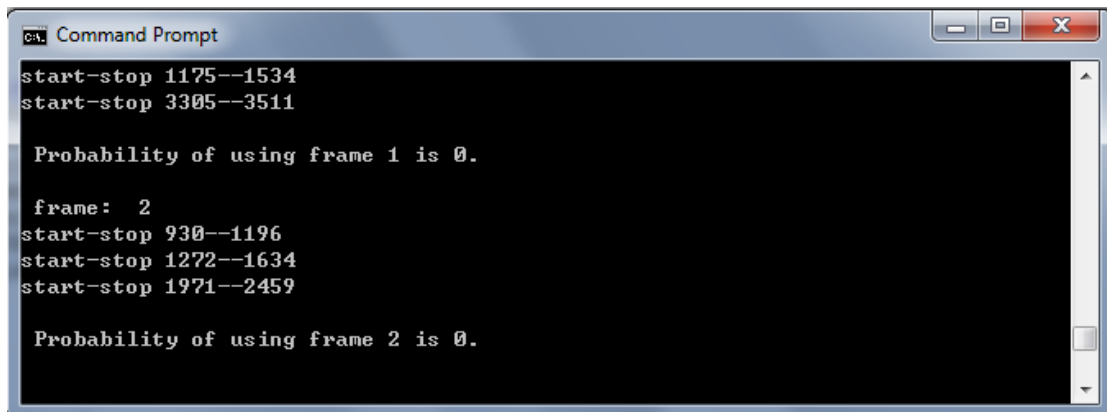
PWM (species)	Start Position	End Position	Strand	Score	Sequence
Arca Escherichia coli (strain K12)	174	183	-	7.33	TGTTAATGAA
Arca Escherichia coli (strain K12)	129	138	-	6.56	AGTTATATAA
Arca Escherichia coli (strain K12)	172	181	+	6.51	TGTTCAITAA
ArgR Escherichia coli (strain K12)	1575	1588	-	9.43	TGAATCAATACGCA
ArgR Escherichia coli (strain K12)	1575	1588	+	8.94	TGCGTATATGATTCA
ArgR Escherichia coli (strain K12)	1582	1595	+	8.88	TGATTCAATAATTAA

3. ecoli41727

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
>gi 41727 emb X65013.1 E.coli DNA sequence of hlyT (rfaH, sfrB) locus and ORF				
orf00001	546	313	-1	7.08
orf00002	1804	620	-2	9.37
orf00003	1971	2459	+3	6.04
orf00005	2966	2664	-3	9.07
orf00006	3252	2950	-1	5.11
orf00008	3572	3282	-3	11.32
orf00009	156	3550	-1	3.84



```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?      startc  odds
# -----
gi_41727_emb_X65013.1_ AP02  CDS      2064    2459    1.25157 +    0      #ATG    1.9585
gi_41727_emb_X65013.1_ AP02  CDS      620     1654    9.94302e-07 -    0      #TTG    -0.035
# -----
```

1.03	Intr	-	323	51	273	2	0	39	-2	373	0.810	21.67
1.02	Intr	-	732	567	166	2	1	-57	35	177	0.321	-1.62
1.01	Init	-	1851	734	1118	0	2	82	34	1207	0.298	108.65
1.00	Prom	-	1968	1929	40							-13.12
2.00	Prom	+	1998	2037	40							-11.87
2.01	Sngl	+	2111	2506	396	1	0	57	41	462	0.975	34.99
2.02	PlyA	+	2598	2603	6							-5.51
3.03	PlyA	-	2701	2696	6							-5.80
3.02	Term	-	3162	2711	452	1	2	13	37	440	0.682	27.13
3.01	Init	-	3631	3394	238	1	1	61	31	247	0.655	14.57

Suboptimal exons with probability > 1.000

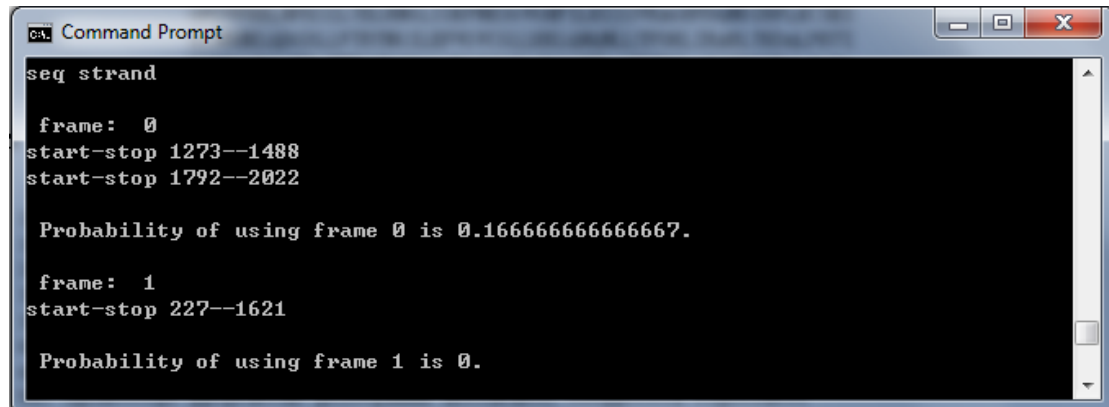
PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	1908	1917	-	7.29	AGTTAATTAT
ArcA Escherichia coli (strain K12)	330	339	+	7.29	TGTTAAAAAT
ArcA Escherichia coli (strain K12)	1862	1871	+	7.07	TGTTAAATTA

4. ecoli41745

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
>gi 41745 emb V00288.1 E. coli specificity gene of EcoK restriction enzyme (hsdS)				
orf00001	2351	230	+3	4.29
orf00002	227	1621	+2	3.50
orf00004	1914	1753	-1	2.58
orf00005	1983	2189	+3	1.26



```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?      startc  odds
# -----
#
  
```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Init + 280 1645 1366 0 1 49 50 911 0.162 74.31
1.02 Intr + 1897 2031 135 2 0 -39 86 221 0.349 9.74
1.03 Intr + 2076 2222 147 1 0 -17 54 165 0.294 2.81

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
  
```

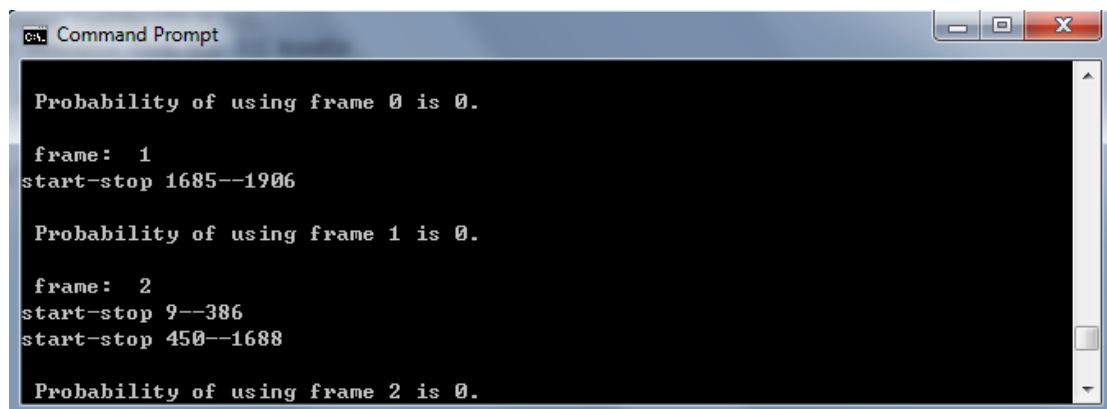
PWM (species)	Start Position	End Position	Strand	Score	Sequence
Arca Escherichia coli (strain K12)	424	433	+	6.59	TGTTAAAGAA
Arca Escherichia coli (strain K12)	838	847	+	6.22	AGTTAATGGA
Arca Escherichia coli (strain K12)	900	909	+	6.21	AGTTAAAITT
ArgR Escherichia coli (strain K12)	526	539	+	8.53	TGAATGTAGTTTCG
ArgR Escherichia coli (strain K12)	715	728	-	8.12	TGATTTTTTGTTCG
ArgR Escherichia coli (strain K12)	2418	2431	-	8.08	TGGATAATCTTTAG

5. ecoli145467

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
>gi 145467 gb M12788.1 ECOCCA E.coli cca gene encoding tRNA nucleotidyltransferase, complete cds				
orf00001	2068	386	+3	3.13
orf00006	489	1688	+3	2.91
orf00008	2015	1869	-3	1.63
orf00009	2236	2033	-2	4.26



```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname          model feature start end score +/- ? startc odds
# -----
#
  
```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Intr + 123 395 273 1 0 40 10 454 0.293 30.85
1.02 Intr + 564 1714 1151 1 2 -9 10 1305 0.060 102.55
1.03 Term + 2024 2191 168 1 0 60 43 283 0.209 19.19

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
  
```

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	1280	1289	+	7.04	TGTTAAGTGA
ArcA Escherichia coli (strain K12)	99	108	-	6.66	TGTTATCGAT
ArcA Escherichia coli (strain K12)	172	181	+	6.38	GGTTAAAAGA
ArgR Escherichia coli (strain K12)	1682	1695	+	9.29	TGAATGAGTATTGG
ArgR Escherichia coli (strain K12)	267	280	-	9.12	TGCATGATGATGGT
ArgR Escherichia coli (strain K12)	406	419	+	8.58	TGCTTATTATTAG

6. ecoli297393

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
>gi 297393 emb X57091.1 E.coli tufB gene for translation elongation factor EF-Tu				
orf00002	344	171	-3	1.15
orf00003	492	728	+3	4.99
orf00006	145	706	-1	2.95

```

seq strand

frame: 0
start-stop 721--1905

Probability of using frame 0 is 0.

frame: 1
start-stop 1547--1735

Probability of using frame 1 is 0.

frame: 2
  
```

```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?    startc  odds
# -----
gi_297393_emb_X57091.1_AP02  CDS      721     1905     1.37835e-10  +      0    #ATG    37.8136
# -----
  
```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----

1.02 PlyA - 696 691 6 1.05
1.01 Sngl - 1997 759 1239 2 0 60 47 1768 0.932 166.26

Suboptimal exons with probability > 1.000
  
```

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	984	993	+	7.01	TGTTAAAAAC
ArcA Escherichia coli (strain K12)	27	36	+	6.46	TGTTGAAAAA
ArcA Escherichia coli (strain K12)	1120	1129	-	6.15	TGTTCAGGAA

7. *ecoli312761*

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
-----	-----	-----	---	-----

```

Command Prompt
seq strand
frame: 0
start-stop 31--474
start-stop 502--1884
Probability of using frame 0 is 0.166666666666667.
frame: 1
start-stop 41--274
start-stop 971--1219
Probability of using frame 1 is 0.

```

```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname          model  feature start  end      score    +/-    ?    startc  odds
# -----
gi_312761_emb_X72677.1_AP02  CDS    31      474      0.728413    +      0    #ATG    -0.9831
gi_312761_emb_X72677.1_AP02  CDS    742     1884     7.06481e-09  +      0    #ATG    18.5812
# -----

```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Term + 520 1911 1392 0 0 90 42 1120 0.454 98.81
1.02 PlyA + 1958 1963 6 -1.75

```

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA <i>Escherichia coli</i> (strain K12)	2067	2076	+	6.81	TGTTAATT
ArcA <i>Escherichia coli</i> (strain K12)	1354	1363	-	6.55	AGTTAACA
ArcA <i>Escherichia coli</i> (strain K12)	178	187	-	6.50	TGTTAATG
ArgR <i>Escherichia coli</i> (strain K12)	1240	1253	-	9.11	TGAATAATCATGCG
ArgR <i>Escherichia coli</i> (strain K12)	387	400	+	8.86	TGAATTACTGTGCG
ArgR <i>Escherichia coli</i> (strain K12)	1217	1230	+	8.42	TGAGTAAACGTTCT

8. ecoli403342

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
-------	-------	-----	-------	-------

>gi 354515237 gb JF918432.1 Escherichia coli class I integron dihydrof				
orf00001	589	34	+2	4.50

```
Command Prompt

C:\Users\Jerry\Desktop\li wen 2010\cs123b\Term Project>perl PromoterFinder.perl
seq strand

frame: 0
start-stop 1393--1785
start-stop 4201--4476

Probability of using frame 0 is 0.

frame: 1
start-stop 278--1372
start-stop 1904--4777

Probability of using frame 1 is 0.166666666666667.

frame: 2
start-stop 2037--2303
start-stop 2670--2975

Probability of using frame 2 is 0.
```

```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?      startc  odds
# -----
gi_354515237_gb_JF918432.1_  AP02    CDS      9         458     0.328944  +      0      #GTG    3.2777
# -----
```

1.01	Init +	309	1399	1091	2	2	86	-21	1369	0.311	119.87
1.02	Intr +	1459	1645	187	1	1	-26	74	218	0.001	9.21
1.03	Intr +	1923	4804	2882	2	2	17	-13	4072	0.000	378.50
1.04	Intr +	5105	5512	408	2	0	-9	0	491	0.002	24.55

Promoter Sequence:											
1	TTCCTGAGT	GTGAGAAAT	GGGTAATCG	GTAGTGGTC	TGATATCCG	TGGTCAGTA	AAGGTGAGCA	ACTACTCTT	80		
81	AAAGGAGTCA	CATATAATCA	ATGGCTCTT	GTGGAAGAA	AAACATTTGA	CTCTATGGT	GTCTTCCAA	ATGCAAAAT	160		
161	TGACAGTAGT	TCAAGAGACG	GAATTTCCAG	CTCAATGAA	AACGTCTTAG	TTTTCTCTC	AAATAGAAAT	GCTTTGAAAG	240		
241	AGCTATCAAA	AGTATAGCA	CATGTATATG	TCCTGGGCG	GGGTCAATC	TATAATAGC	TTATTGAAA	AGCAGATATA	320		
321	ATTCATTTG	CTACTGTCA	CGTGAAGTC	GAAGGTGATA	TCAAATTCG	TATAATAGC	GAGAAATTC	ATTGCTG	400		
401	TCGACGCTT	TTTATGTCTA	ATATTAATTA	TACATACGAG	ATTGGAAGAA	AGGCTATAG	ATGCTTCCA	GCACAGTCC	480		
481	CTTCTCTCT	TGACAGCTT	TTAAGTGGG	CTTTGTGTG	TTTGTGGCG	AAAAGTATC	CACAAAGCG	CAACTTAAA	560		
561	GCTGCGGCT	AACCTAACGT	TAGGATCAT	GGGTGAATTT	TTGCTGCGC	AAGTTTTC	GCAGTGTCC	CAAGCTGCG	640		
641	CGGTGATCG	GCAGCATCT	GCTGCGAC	TGGACACAT	CCACTGTTC	GGATCTGCG	TGATGGAGG	GCTGAAGCG	720		
721	GACAGCGACA	TAGACTTCT	CGTGAAGCT	AGCGCGCAC	CTAACGATC	GCTCGGCG	GCCTAATGC	TGATTTGCT	800		
801	GAAGTCTCA	TCACCGCC									
PWM (species)											
Start Position				End Position				Strand	Score	Sequence	
Arca Escherichia coli (strain K12)				397				-	6.05	TGTCAAAAA	
Arca Escherichia coli (strain K12)				251				+	5.92	AGTTACAGAT	
Arca Escherichia coli (strain K12)				451				-	5.85	TGTATGCTT	

9. ecoli414745

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:
 orfID start end frame score

 >gi|414745|emb|X71063.1| E.coli feoA and feoB genes
 orf00002 821 2899 +2 3.00

```

C:\> Command Prompt

frame: 0
start-stop 334--561
start-stop 2890--3135

Probability of using frame 0 is 0.

frame: 1
start-stop 566--2899
start-stop 3056--3247

Probability of using frame 1 is 0.

frame: 2
  
```

```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-06
##Type DNA
# model: AP02 Aeropyrum pernix
# segname        model    feature start    end        score    +/-    ?    startc    odds
# -----
gi_414745_emb_X71063.1_ AP02    CDS        821        2899    1.91998e-21    +    0    #TTG    69.001
# -----
  
```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Init +    605    2881 2277    1    0    64   111   2992 0.567 287.09
1.02 Intr +    2944   3146   203    0    2    44   40    209 0.510 10.31

Suboptimal exons with probability > 1.000
  
```

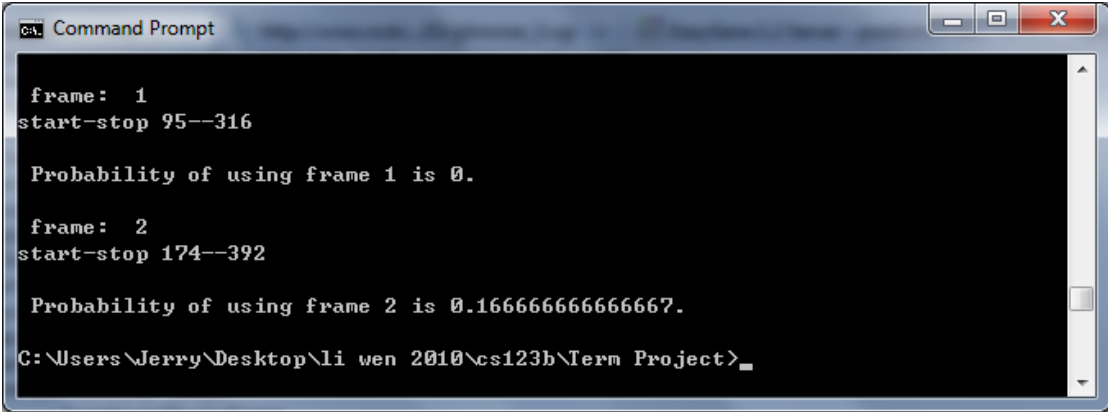
PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	47	56	-	7.41	TGTTAATGAA
ArcA Escherichia coli (strain K12)	632	641	-	6.96	GGTTAAATAA
ArcA Escherichia coli (strain K12)	2992	3001	-	6.93	CGTTAATCAT
ArgR Escherichia coli (strain K12)	153	166	-	9.73	TGAATAATAAICTG
ArgR Escherichia coli (strain K12)	3204	3217	-	9.28	TGCATTTTTAIGGA
ArgR Escherichia coli (strain K12)	2265	2278	+	9.04	TGCATGAAGATAAC

10. ecoli167509193

[Submit new Data](#)

GLIMMER (ver. 3.02; iterated) predictions:
orfID start end frame score

>gi|167509193|gb|EU447295.1| Escherichia coli strain 2622 SodC (sodC) gene, partial cds
orf00002 128 498 -3 6.38



```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?      startc  odds
# -----
# -----
# -----
Explain the output. Go back.
```

```
Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Init +      50    557  508  1  1   19  -11   604 0.578  36.77

Suboptimal exons with probability > 1.000
```

Promoter Sequence:					
1	ATGAACGTT	TTAGTCTGGC	TATCTGGCG	CTGGTGTGTC	CAACGGGCGC
81	CCTGCTCAGC	TCGCAAGGGG	TAGGGGCGTC	AAITGGTAGC	GTACCAATTA
161	CGATCTGAA	AGCAATACCC	CCGGTGAAC	ATGGCTTCCA	TATTCATGCC
241	GGCAAGCCA	GGCGCGGGA	ATCGCAGGC	GGCACTTG	ATCCAAAA
321	CGGCACTTA	GGCGATCTGC	CTGCACTGGT	GTCAATAAT	GACGGCAAG
401	AACTACTGA	TGAATCAAA	GACAAAGGC	TGATGGTCCA	CGTTGGCGGC
481	GGCGGTGGC	GTGAACGCTA	TGCTGTGGT	GTAATTAG	

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArgR Escherichia coli (strain K12)	272	285	-	9.21	TGGATCAAGATGCC

11. ecoli260765442

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
-----	-----	-----	--	-----

```
C:\Users\Jerry\Desktop\li wen 2010\cs123b\Term Project>perl PromoterFinder.perl
seq strand

frame: 0
start-stop 1--1026

Probability of using frame 0 is 0.

frame: 1
start-stop 110--310

Probability of using frame 1 is 0.
```

```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model      feature start  end      score    +/-    ?    startc    odds
# -----
gi_260765442_gb_GQ906590.1_    AP02    CDS      232     1026    0.000487696    +      0      #ITG    -5.3721
# -----
```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..

1.01 Sngl + 66 1079 1014 2 0 45 41 871 0.874 75.60

Suboptimal exons with probability > 1.000

Promoter Sequence:									
1	>GI GB GQ . ESCHERIC	HIACOLISTR	AINKCHITOB	IASEGENE,C	OMPLETECD5	GTGGGTCCAG	TAATGTTGGA	80	
81	TGTCGAAGGT	TACGAACCTGG	ACGCGGSAAGA	GGGTGAAATA	CTGGGCGAIC	CGCTGGTGGG	AGGGCTGATT	160	
161	GTAACATACA	TGATCTCTGCC	CAGTACAGTG	AACCTGGTGG	CCAGATCCGC	GCAGCTTCGC	GCAATGCTCT	240	
241	GTTGATCAGG	AAGGTGGAGC	CGTGCAGCGT	TTTCGTGAAG	GTITTAACCG	CTTGCCAGCG	GCSCAATCAT	320	
321	GTACGGAATG	GAAGAGGGTG	GCAAACTGGC	GCAGGAGGCA	GGTTGGTTGA	TGGCCAGCGA	AATGATCGCT	400	
401	ATATCAGCTT	TGCGCTGTGT	CTGGAITGCG	GCATATCAG	CGCGGCGATT	GGCGAGCGTT	CTTATCATGC	480	
481	AAAGCCCTGG	CAATTGCCAG	CGGTTTATT	GATGGTATGC	ATGAAGCCGG	AATGAAAACG	ACCGGGAAAC	560	
561	ACACGGTGCA	GTAACGGCAG	ACTCACACAA	AGAAACACCG	TGCGATCCAC	GTCCACAAGC	GGAGATTGCG	640	
641	TGTCGGTCTT	CAGTTCCCTA	ATCCGCGAAA	ATAAACTCGA	CGCCATTATG	CCTGCGCATG	TGATCTACAG	720	
721	CCGCGTCCGG	CGAGCGTTTC	TCCCTACTGG	CTGAAAACCG	TTTTCGCTCA	GGAACTGGGT	TTTGACGGCG	800	
801	TGACGATTTA	TCGATGGAAG	GTGCGCGGAT	TATGGGCAGT	TATGCCGAAC	GCGGGACAGC	TTCACTGGAT	880	
881	ATATGATCCT	GGTCTGCAAT	AATCGTAAAG	GGGCGGTCAG	CGTGTTAGAT	AATCTGTAC	CGATCAAGGC	960	
961	ACACGTTTGT	ATCATAAAGG	TTCAATTTTC	CGACAGGAAC	TGATGGACTC	GGCTCGCTGG	AAAGCGATCA	1040	
1041	GAATCAGTTA	CATGAACGCT	GGCAGGAAGA	GAAAGCAGGT	CACTAA				
PWM (species)		Start Position		End Position		Strand			

12. ecoli300901746

```

GLIMMER (ver. 3.02; iterated) predictions:
orfID      start      end      frame      score
-----
>gi|300901746|ref|NZ_ADTJ01000560.1| Escherichia coli MS 198-1 E_coli198-1-1.0_Cont914.2,
orf00002      38      4363      +2      6.04
orf00005     4383     5129      +3      4.91
orf00009     5188     6048      +1      4.49
orf00011     6151     6711      +1     11.15
orf00014     7176     6922      -1      5.61
orf00015     7222     7758      +1      7.70
orf00017     7804     8013      +1      5.71
orf00019     8051     8641      +2     10.51
orf00020     8881     9141      +1      8.17
orf00021     9429    10298      +3      3.12
orf00023    10450    10310      -2      3.56
orf00026      23     10717      -1      0.34

```

```

C:\Users\Jerry\Desktop\li wen 2010\cs123b\Term Project>perl PromoterFinder.perl
seq strand

frame: 0
start-stop 5188--6048
start-stop 6151--6711
start-stop 6856--7098
start-stop 7222--7758
start-stop 7783--8013
start-stop 8212--8409
start-stop 8881--9141
start-stop 9574--9804

Probability of using frame 0 is 0.3333333333333333.

frame: 1
start-stop 38--4363
start-stop 4442--4654
start-stop 5282--5557
start-stop 5864--6061

```



EasyGene 1.2 Server - prediction results

Technical University of Denmark

```

##gff-version 2
##source-version easygene-1.2b
##date 2011-04-30
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model      feature start      end      score      +/-      ?      startc      odds
# -----
gi_300901746_ref_NZ_ADTJ01000560.1_      AP02      CDS      86      4363      3.08386e-45      +      0      #GTG      96.097
gi_300901746_ref_NZ_ADTJ01000560.1_      AP02      CDS      5614     6048      0.218941      +      0      #ATG      7.4309
gi_300901746_ref_NZ_ADTJ01000560.1_      AP02      CDS      6151     6711      0.158231      +      0      #ATG      -9.0152
gi_300901746_ref_NZ_ADTJ01000560.1_      AP02      CDS      8183     8641      1.63256 +      0      #TTG      -15.2881
# -----

```

Explain the output. Go [back](#).

1. 01	Intr	+	89	4384	4296	1	0	44	22	5745	0.993	552.37
1. 02	Intr	+	4515	5090	576	2	0	38	3	566	0.323	36.62
1. 03	Intr	+	5104	5195	92	0	2	99	64	20	0.528	-0.01
1. 04	Intr	+	5364	6078	715	0	1	41	-27	546	0.054	30.48
1. 05	Term	+	6202	6774	573	0	0	16	42	774	0.085	60.46
1. 06	PlyA	+	6885	6890	6							-1.95
2. 00	Prom	+	7172	7211	40							-7.40
2. 01	Init	+	7360	7762	403	0	1	62	16	267	0.423	12.88
2. 02	Intr	+	7856	8072	217	0	1	48	85	125	0.472	6.29
2. 03	Term	+	8182	8704	523	1	1	35	48	436	0.539	28.45
2. 04	PlyA	+	8888	8893	6							-0.45
3. 00	Prom	+	8903	8942	40							-14.13
3. 01	Init	+	8944	9115	172	0	1	84	50	112	0.890	6.57
3. 02	Term	+	9493	10361	869	2	2	30	48	923	0.979	75.81

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	7596	7605	+	7.09	TGTTAATCAG
ArcA Escherichia coli (strain K12)	5886	5895	+	6.99	TGTTAATCCA
ArcA Escherichia coli (strain K12)	5160	5169	-	6.79	TGTTTATTAA
ArgR Escherichia coli (strain K12)	7439	7452	-	9.48	TGATCAATATTTA
ArgR Escherichia coli (strain K12)	5470	5483	-	9.19	TGATTATCAATACA
ArgR Escherichia coli (strain K12)	7592	7605	+	9.16	TGATGTTAATCAG

13. ecoli325965637

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:
orfID start end frame score
>gi|354515237|gb|JF918432.1| Escherichia coli class I integron dihydrofo
orf00001 589 34 +2 4.50

Command Prompt

start-stop 8173--8523

Probability of using frame 0 is 0.

frame: 1
start-stop 1592--2515
start-stop 2564--3646
start-stop 4751--5935
start-stop 6389--6640
start-stop 7430--7630
start-stop 7631--7813

Probability of using frame 1 is 0.166666666666667.

frame: 2
start-stop 129--1580
start-stop 1599--1793
start-stop 7866--8339
start-stop 8397--8639

Probability of using frame 2 is 0.

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
model: AP02 Aeropyrum pernix
seqname model feature start end score +/- ? startc odds

gi_354515237_gb_JF918432.1_ AP02 CDS 9 458 0.328944 + 0 #GTG 3.2777

1.01	Init	+	254	1337	1084	1	1	73	64	440	0.213	34.70
1.02	Term	+	1700	2631	932	0	2	63	38	508	0.444	34.61
1.03	PlyA	+	2664	2669	6							-4.04
2.00	Prom	+	2672	2711	40							-14.53
2.01	Init	+	2740	3745	1006	0	1	45	-29	846	0.887	62.26
2.02	Term	+	3763	4760	998	2	2	9	38	598	0.759	38.15
2.03	PlyA	+	4846	4851	6							-3.64
3.00	Prom	+	4868	4907	40							-8.05
3.01	Sngl	+	4939	6051	1113	0	0	49	40	425	0.915	30.35
3.02	PlyA	+	6086	6091	6							1.05
4.03	PlyA	-	6270	6265	6							1.05
4.02	Term	-	7614	7492	123	0	0	46	46	127	0.352	1.70
4.01	Init	-	8992	7877	1116	1	0	4	11	1520	0.504	130.22

Promoter Sequence:

1	TTCCTGAGT	GTGAGAAAT	GGGTAATCG	GTAGTGGTCC	TGATATCCG	TGGTCAGTAA	AAGGTGAGCA	ACTACTCITT	80
81	AAAGGCGTCA	CATATAATCA	ATGGCTCCTT	GTGGAAGAA	AAACATTTGA	CTCTATGGGT	GTCTTCCAA	ATGCAAAATA	160
161	TGCAGTAGTG	TCAAAGAACG	GAATTTCCAG	CTCAATGAA	AACTGCTAG	TTTTTCCTTC	AAATAGAAAT	GCTTTGAAAG	240
241	AGCTATCAAA	AGTATGCA	CATGTATATG	TCCTGCGCG	GGGTCAATC	TATAATAGCC	TTATTGAAAA	AGCAGATATA	320
321	ATTCATTTGT	CTACTGTGCA	CGTGAAGTTC	GAAGGTGATA	TCAAATTCCT	TATATAGGCT	GAGAAATTCA	ATTGCTTT	400
401	TCGACGCTTT	TTTATGTGTA	ATATAAATTA	TACATACGAG	ATTGGAAGAA	ATGGCTTCCG	GCACCACTGC		480
481	CTTGCCTCCT	TGACAGCTTT	TTAAGTGGCG	ICTTTGTGTG	TTTGTGCGGC	AAAAGTATTC	CACAAAGCGC	CAACTTAAAA	560
561	GCTGCGCGTG	AACCTTAAGCT	TAGGCAATCAI	GGGTGAATTT	TTCCCTGCAC	AAGTTTTCAA	GCAGCTGTCC	CAAGCTGCGC	640
641	CGGTGATGGA	GCGCCATCTG	GTGCGGACAC	TGGACACAAT	CCACTGTGTC	GGATCTGCGA	TGATGGAGG	GCTGAAGCGC	720
721	GACAGCGACA	TAGACTTGCT	CGTGACCGTC	AGCGCGCGAC	CTAACGATTC	GCTCGGCGAG	CGGCTAATGC	TGATTTGCT	800
801	GAAGTCTCA	TCACCGCC							

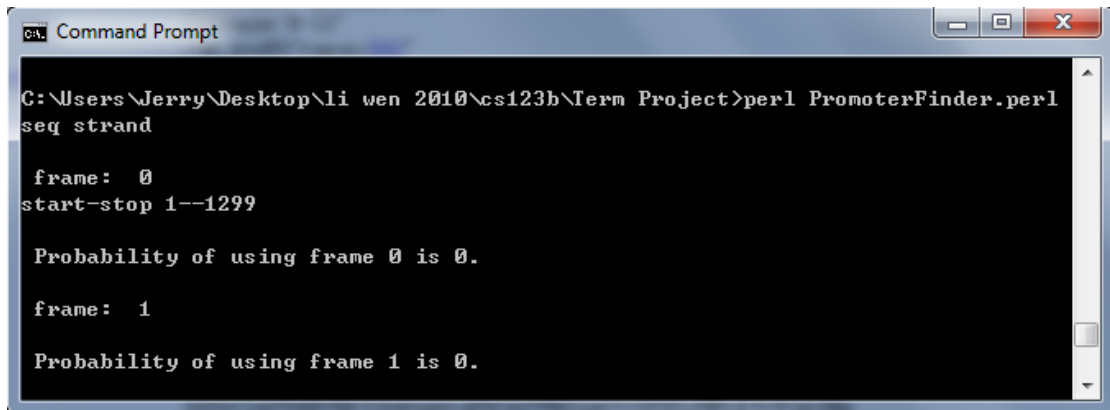
PWM (species)	Start Position	End Position	Strand	Score	Sequence
Arca Escherichia coli (strain K12)	397	406	-	6.05	TGTTCAAAAA
Arca Escherichia coli (strain K12)	251	260	+	5.92	AGTTACAGAT
Arca Escherichia coli (strain K12)	451	460	-	5.85	TGTTAGCCTT

14. ecoli341941295

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
-----	-----	-----	--	-----



```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?      startc  odds
# -----
# -----
```

```
Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----
1.01 Term +    132   1361 1230  2  0   56   41   652 0.787 49.47

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
```

PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	630	639	-	6.96	CGTTAATGAA
ArcA Escherichia coli (strain K12)	628	637	+	6.84	TGTTCAITTA
ArcA Escherichia coli (strain K12)	434	443	+	6.70	CGTTAATTTAA
ArgR Escherichia coli (strain K12)	572	585	+	8.66	TGCTTAATTTTCG

15. ecoli342315677

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:
 orfID start end frame score

```

C:\Users\Jerry\Desktop\li wen 2010\cs123b\Term Project>perl PromoterFinder.perl
seq strand

frame: 0
start-stop 322--621

Probability of using frame 0 is 0.

frame: 1

Probability of using frame 1 is 0.

frame: 2
  
```

```

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname            model    feature start    end        score    +/-    ?    startc    odds
# -----
# -----
  
```

```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----

1.01 Intr +    400    685   286   0   1    89   80   355 0.721   31.71

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
  
```

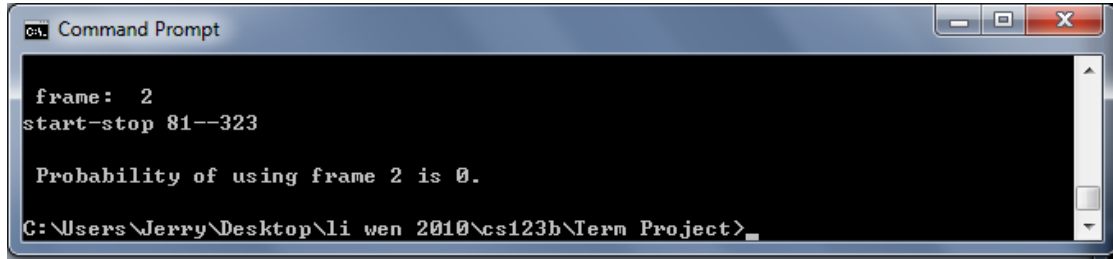
PWM (species)	Start Position	End Position	Strand	Score	Sequence
ArcA Escherichia coli (strain K12)	145	154	+	6.69	TGTTAICTAA
ArcA Escherichia coli (strain K12)	224	233	+	6.45	TGTTTATAAT
ArcA Escherichia coli (strain K12)	351	360	-	6.02	TGTTAAGTT

16. ecoli346421495

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
>gi 346421495 gb JN411912.1 Escherichia coli strain CQ beta-lactamase (blaCTX-M) gene, partial cds				
orf00001	454	27	+1	5.64
orf00002	88	210	+1	11.91



CENTERFOR
BIOLOGICAL
CALSEQUENCE
ANALYSIS
CBS

EasyGene 1.2 Server - prediction results

Technical University of Denmark

##gff-version 2
##source-version easygene-1.2b
##date 2011-11-29
##Type DNA
model: AP02 Aeropyrum pernix
seqname model feature start end score +/- ? startc odds

Explain the output. Go [back](#).

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..

1.01 Init + 69 516 448 2 1 51 18 388 0.403 24.27

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..

Promoter Sequence:

1	>gi 346421495 gb JN411912.1 Escherichia coli strain CQ beta-lactamase (blaCTX-M) gene, partial cds	80
81	TGTCGAGCAG CAGTAAAGTG ATGGCGCGGG CGCGGTGCT GAAGAAAGT GAAAGCGAAC CGAATCTGTT AAATCAGCGA	160
161	GTTCGATCA AAAATCTGA CTTGTTTAC TATAACCGA TTGCGAAAA GCACGTCAAT GGGAGATGT CACTGGCTGA	240
241	GCTTAGCGCG CGCGGCTAC AGTACAGCGA TACGTGGCG ATGAATAGC TGATTGCTCA GCTTGGCGCG CGGCTAGCG	320
321	TACCGGTTT CCGCGACAG CTGGAGAGCG AAGCTTCCG TCTGACGCT ACCGAGCGA GCTTAACAC CGCATTCGG	400
401	GGGATCGCG GTATACCGA TTGACCTCGG GCAATGGCG AACTCTGGG GAATCTGAG CTGGTTAAG CATTGGGCGA	480
481	CAGCCACGCG GCGAGCTGG TGACATGGAT GAAAGGCAAT ACCACGGTG CAGCGAGCAT TCAGGCTGGA CTGCTGCTT	560
561	CTGGGTTGT GGGGGATAA ACCGCGAGG GTGGCTATGG CACCACCAAC CATATCGCG	

PWM (species)	Start Position	End Position	Strand	Score	Sequence
---------------	----------------	--------------	--------	-------	----------

17. ecoli354515237

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

```
orfID      start      end      frame      score
-----
>gi|354515237|gb|JF918432.1| Escherichia coli class I integron dihydrofo
orf000001    589        34 +2        4.50
```

```
ca. Command Prompt

reversecomplement strand

frame: 0
start-stop 67--264
start-stop 322--573

Probability of using frame 0 is 0.

frame: 1

Probability of using frame 1 is 0.

frame: 2
```

```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model      feature start      end      score      +/-      ?      startc      odds
# -----
gi_354515237_gb_JF918432.1_ AP02      CDS      9          458      0.328944      +          0          #GTG      3.2777
# -----
```

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..

1.01 Init + 705 920 216 2 0 96 -47 229 0.588 8.97

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..

Promoter Sequence:									
1	TCTCGCAGT	GTGAGAAAT	GGGTAATCG	GTAGTGGTCC	TGATATCCCG	TGGTCAGTAA	AAGGTGAGCA	ACTACTCTTT	80
81	AAAGCGCTCA	CATATAATCA	ATGGCTCTTT	GTGGAAGAA	AAACATTTGA	CTCTATGGGT	GTCTCTCCAA	ATGCAAAATA	160
161	TGCAGTAGTG	TCAAGAACGC	GAATTTCCAG	CTCAATGAA	AACGTCTCTAG	TTTTCTCTTC	AAATAGAAAT	GCTTTGAAAG	240
241	AGCTATCAAA	AGTATGCA	CATGTATATG	TCCTGCGCG	GGGTCAATTC	TATAATAGCC	TTATTGAAAA	AGCAGATATA	320
321	ATTCATTTGT	CTACTGTTCA	CGTGAAGTTC	GAAGTGTGATA	TCAAATTCCT	TATATAGGCT	GAGAAATTTCA	ATTGCTGCT	400
401	TCGACGCTTT	TTTATGTGTA	ATATTAATTA	TACATACGAG	ATTGCGAAAA	ATGCTGTGCA	GCACCACTGC		480
481	CTTCCGCTCT	TGGACAGCTT	TTAAGTGGCG	CTTTGTGTGT	TTTGTGCGGC	AAAAGTATTC	CACAAAGCGC	CAACTTAATA	560
561	GCTGCGCGTG	AAGTTAAGCT	TAGGATCAT	GGGTGAATTT	TTCCCTGCAC	AAGTTTTCAA	CGAGCTGTCC	CAAGCTGCGC	640
641	CGGTGATGGA	GCSCCATCTG	GCTGGGACAC	TGGACACAAT	CCACTGTGTC	GGATCTGGGA	TGATGGAGG	GCTGAAGCGC	720
721	GACAGCGACA	TAGACTTGCT	CGTGACCGTC	AGCGCGGCAC	CTAACGATTC	GCTCGGCGAG	GCCTAATGCT	TGATTTGCT	800
801	GAAGTCTTCA	TCACCGCC							

PWM (species)	Start Position	End Position	Strand	Score	Sequence
Arca Escherichia coli (strain K12)	397	406	-	6.05	TGTTCAAAAA
Arca Escherichia coli (strain K12)	251	260	+	5.92	AGTTACAGAT
Arca Escherichia coli (strain K12)	451	460	-	5.85	TGTTAGCCTT

18. ecoli354515240

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID start end frame score

```
Command Prompt

frame: 1
start-stop 35--277
start-stop 542--736

Probability of using frame 1 is 0.166666666666667.

frame: 2
start-stop 42--254

Probability of using frame 2 is 0.

C:\Users\Jerry\Desktop\li wen 2010\cs123b\Term Project>
```

```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname model feature start end score +/- ? startc odds
# -----
#
```

```
Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----

1.01 Intr + 204 874 671 0 2 91 18 589 0.301 44.18

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
```

Promoter Sequence:									
1	TTCITCGTGA	AATAGTGATT	TTTGAAGCTA	ATAAAAAACA	CACGTGGAAT	TTAGGGACTA	TTCAITGTGT	TGTTAITTCG	80
81	TATCTTCAG	AATAAGGAAT	CCATGGTTA	AAAAATCACT	GGCCAGTTC	ACGCTGATGG	CGACGGCAAC	CGTCAACGCTG	160
161	TTGTTAGGAA	GTGTGCCCT	GTAIGGSCAA	ACGGCGGACG	TACAGCAAAA	ACTTGGCCAA	TTAGAGCGGC	AGTCGGGAGG	240
241	CAGACTGGGT	GTGGCTGSA	TTAACA	CAGC	AGATAATTGC	CAAACTACTT	ATGTGCTGTA	TGAGCGCTTT	320
321	GCACAGATA	AGTATGGCC	GGGGCGCGGG	TGCTGAAGRA	AAGTGAAGC	GAACCGAAT	TGTTAAATCA	CGGAGTTGAG	400
401	ATCAAAAAT	CTGACTCTG	TTAATGTA	AAT	CGGATTGGG	AAAGCAGCT	CAATGGGAGC	ATGTCACTGG	480
481	CGCGCGCGG	CTACAGTACA	GGGAACGT	GGCGATGAT	AAGCTGATTG	CTCACTTGG	CGGCCCGGCT	AGCTCACCG	560
561	CGTTGCGCG	ACAGCTGGGA	GAGGAACGT	TCGCTCTGGA	CGTACCGAG	CGAGGTAA	ACACCGGCAT	TCCGGGCGAT	640
641	CGCGTGATA	CCACTTCACC	TCGGGCAATG	GGCCTAACCT	TGCGGAATCT	GACGCTGGGT	AAAGCAITGG	GGCAGACGCA	720
721	ACGGCGCGAG	CTGTTGACAT	GGATGAAGG	CAATACACC	GGTCAGCGA	GCATTCAGCT	GGACTGCTTG	CTTCTGGGT	800
801	TGTGGGGG								
PWM (species)					Start Position	End Position	Strand	Score	Sequence
Arca Escherichia coli (strain K12)					257	266	-	7.29	TGTAAATCAA
Arca Escherichia coli (strain K12)					418	427	+	7.21	TGTAACTAT
Arca Escherichia coli (strain K12)					381	390	+	6.86	TGTAAATCA

19. ecoli354515242

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:

orfID	start	end	frame	score
-----	-----	-----	--	-----

```
ca. Command Prompt

Probability of using frame 0 is 0.

frame: 1

Probability of using frame 1 is 0.

frame: 2
start-stop 141--788

Probability of using frame 2 is 0.

##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname          model  feature start  end      score    +/-    ?      startc  odds
# -----
gi_354515242_gb_JF918434.1_  AP02    CDS      225      788      0.00194172  +      0      #ATG    18.9505
# -----
```

```
Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
-----

1.01 Term +      332      855  524  0  2   53   44   546 0.465  41.14

Suboptimal exons with probability > 1.000

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
```

Promoter Sequence:									
1	TGATTTGAAA	GGTGGTTGTA	AATGATGTTA	CAATGTGTGA	GAAGCAGTCT	AAATCTCTCG	TGAAATAGTG	ATTTTGAAG	80
81	CTAATAAAAA	ACACACGTGG	AAATTAGGAA	AACTTATAT	CTGCTGCTCA	ATTTAACCGT	TGTGCAACAC	GGTGCAATC	160
161	AAACACACTG	ATTGCGTCTG	ACGGGCCCGG	ACACCTTTTT	GCTTTTAAIT	ACGGAAGTGA	TTTCATGATG	AAAAAATCGT	240
241	TATGTGCGCG	CTGCTGCTG	ACAGCCTCTT	TCTCCACATT	TGCTGCGCGA	AAACACAGAC	ACAGAGTTGC	CGATATCGTT	320
321	AAITGACACA	TCACCCGTT	GAITCAGGAG	CAGCTATTTC	CGGTAATGCG	CGTTGCGGTT	ATCTACAGG	GAATACCCCTA	400
401	TTATTTGCGC	TGGGTAAAG	CGATATCGCG	CAATAACAC	CGATGACCG	AGCAACCGCT	GTTTGAGCTA	GGATCGGTTA	480
481	GTAAACGCTT	TACCGCGGTG	TTGGGCGCGG	ATGCTATCGC	CGCGGCGGAA	ATTAACTCTA	GGATCCCGTT	CAGGAATATC	560
561	TGGCCAGAAC	TGACAGGCAA	ACAGTGGCAG	GGTATCCGCG	TGCTGCACCT	AGCCACCTAT	ACGGCAGGCG	CGCTACCGCT	640
641	GCAGATCCCC	GATGACGTTA	GGGATAAGC	CGCATTACCG	CAATTTTATC	AACTTGGA	GGCGCAATGG	ACTCCGCGCG	720
721	CTAAGCGACT	TTACGCTAAC	TCCAGCAITG	GTCTGTTTGG	CGCGCTGGCG	GTGAACCTCT	CAGATGAGT	TACGAA	

PWM (species)	Start Position	End Position	Strand	Score	Sequence
Arca Escherichia coli (strain K12)	140	149	-	6.29	TGTTGACAAA
Arca Escherichia coli (strain K12)	129	138	-	6.26	GGTTAAATTT
Arca Escherichia coli (strain K12)	486	495	-	6.09	CGTTAAACGT
ArgR Escherichia coli (strain K12)	679	692	+	8.95	TGCAITTTTATCAA

20. ecoli354515243

Submit new Data

GLIMMER (ver. 3.02; iterated) predictions:
orfID start end frame score

```
ca: Command Prompt
frame: 0
Probability of using frame 0 is 0.
frame: 1
Probability of using frame 1 is 0.
frame: 2
Probability of using frame 2 is 0.
C:\Users\Jerry\Desktop\li wen 2010\cs123b\Term Project>
```

```
##gff-version 2
##source-version easygene-1.2b
##date 2011-12-05
##Type DNA
# model: AP02 Aeropyrum pernix
# seqname      model  feature start  end      score    +/-    ?      startc  odds
# -----
# -----
```

Gn.Ex	Type	S	.Begin ...End	.Len	Fr	Ph	I/Ac	Do/T	CodRg	P....	Tscr..

1.01	Intr	-	214 92	123	2	0	39	29	178	0.646	8.73
Suboptimal exons with probability > 1.000											
Exnum	Type	S	.Begin ...End	.Len	Fr	Ph	B/Ac	Do/T	CodRg	P....	Tscr..

Promoter Sequence:					
1	GTITGAGCTG	GGCTCTATAA	GTAAACCTT	CACCGGCGTA	CTGGGCGGCG
81	GCATCCGGT	AGCAAAATAC	TGGCCTGAGC	TCACGGGCAA	GCAGTGGCAG
161	ACCGCAGGCG	GTCTGCCGTT	ACAGGTGCGG	GATGAGGTCA	CGGATACGCG
241	GGCGCAGTGG	AGG			
80	ATGCGGCTGG				
160	GGCAACCTAT				
240	AAAATGGA				
PWM (species)					
Arca Escherichia coli (strain K12)					
Start Position		End Position		Strand	Score
17		26		-	6.10
					TTTTACTTAT

Appendix 5—Web Application

1. Home page

E. coli Gene Finder

TAACA CT TTCCCAAATGGTGCTTTGGATTGAAAAGG
TGGTCTCTTAATAAACCTAGAAAAACGGGTAGTT
GGCAGGAGGGTAAATATGGCATAAGTTAATAACA CTTTTCCCAAATGGTGCTTTGGATTGAAAAGG
TGAAGGAGAACGTATCATCTAGCTTGGCTGCTTTAATAAGAGCTGAAAAGGTTAGTTA

Home

E. coli1

E. coli2

E. coli3

TP

FDR

About E. coli Gene Finder

Summary:

This project uses the characteristic in TATA-less regions on E. coli sequences to predict the promoter region before TSS, which indicate that the real gene has been located. It uses several well-known algorithms and methods such as the sliding window algorithm, and a clustering method to predict promoters. It also contains D2K algorithm and method to compare predicted result with other online promoter package result.

Cover:

- > Start and Stop location
- > TP
- > FDR

Project Overview

The prediction made by this program is only for some E. coli sequences as we can see from the data from NCBI. Even though the results look better than two of the online packages prediction, it still has the limitation on predictions.

Algorithm D2K

Algorithm depends on the finding of TATA-less region, the steps during classification and statistics. Improved promoter prediction will use the characteristic from both TATA-rich regions and TATA-less regions to increase truth positive prediction result.

2. One example

E. coli 300901746

TAACA CT TTCCCAAATGGTGCTTTGGATTGAAAAGG
TGGTCTCTTAATAAACCTAGAAAAACGGGTAGTT
GGCAGGAGGGTAAATATGGCATAAGTTAATAACA CTTTTCCCAAATGGTGCTTTGGATTGAAAAGG
TGAAGGAGAACGTATCATCTAGCTTGGCTGCTTTAATAAGAGCTGAAAAGGTTAGTTA

NCBI start	NCBI stop	Glimmer start	Glimmer stop	Start	Stop	EasyG
4383	5129	4383	5129			
5188	6048	5188	6048	5188	6048	
6151	6711	6151	6711	6151	6711	
6840	7052	7176	6922	6856	7098	
7285	7758	7222	7758	7222	7758	
7804	8013	7804	8013	7783	8013	
8051	8641	8051	8641	8212	8409	
8881	9141	8881	9141	8881	9141	
9429	10298	9429	10298	9574	9804	
9		9		8		
0		58.77777778		147.25		