HW 4

Mathematics 127
Mathematical and Computational Methods in Molecular Biology

Fall 2002
UC Berkeley, CA

Nasser M. Abbasi

Fall 2002 Compiled on August 2, 2022 at 8:10am [public]

Contents

1 Problems 3

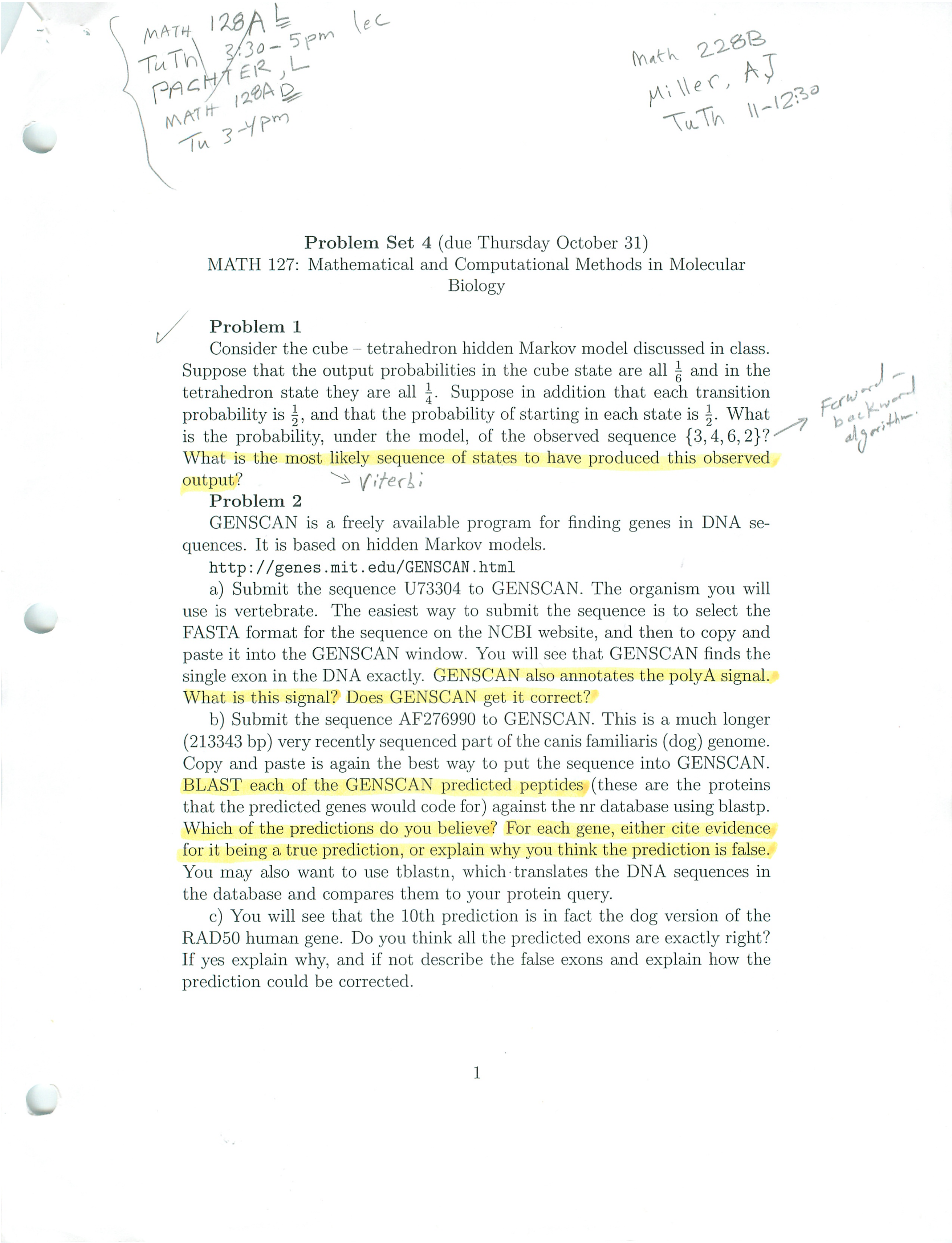
2 Problem 1 6

3 Problem 2 10

4 Problem 3 21

5 Problem 4 23

1 Problems



MATH 128A L
TuTh 3:30-5pm lec
PACHTER, L
MATH 128A D
Tu 3-4pm

Math 228B
Miller, AJ
TuTh 11-12:30

Problem Set 4 (due Thursday October 31)
MATH 127: Mathematical and Computational Methods in Molecular Biology

✓ **Problem 1**

Consider the cube – tetrahedron hidden Markov model discussed in class. Suppose that the output probabilities in the cube state are all $\frac{1}{6}$ and in the tetrahedron state they are all $\frac{1}{4}$. Suppose in addition that each transition probability is $\frac{1}{2}$, and that the probability of starting in each state is $\frac{1}{2}$. What is the probability, under the model, of the observed sequence $\{3, 4, 6, 2\}$? → Forward-backward algorithm. What is the most likely sequence of states to have produced this observed output? → Viterbi

Problem 2

GENSCAN is a freely available program for finding genes in DNA sequences. It is based on hidden Markov models.

`http://genes.mit.edu/GENSCAN.html`

a) Submit the sequence U73304 to GENSCAN. The organism you will use is vertebrate. The easiest way to submit the sequence is to select the FASTA format for the sequence on the NCBI website, and then to copy and paste it into the GENSCAN window. You will see that GENSCAN finds the single exon in the DNA exactly. GENSCAN also annotates the polyA signal. What is this signal? Does GENSCAN get it correct?

b) Submit the sequence AF276990 to GENSCAN. This is a much longer (213343 bp) very recently sequenced part of the canis familiaris (dog) genome. Copy and paste is again the best way to put the sequence into GENSCAN. BLAST each of the GENSCAN predicted peptides (these are the proteins that the predicted genes would code for) against the nr database using blastp. Which of the predictions do you believe? For each gene, either cite evidence for it being a true prediction, or explain why you think the prediction is false. You may also want to use tblastn, which translates the DNA sequences in the database and compares them to your protein query.

c) You will see that the 10th prediction is in fact the dog version of the RAD50 human gene. Do you think all the predicted exons are exactly right? If yes explain why, and if not describe the false exons and explain how the prediction could be corrected.

Problem 3*

Consider a state with a self-transition probability of p in a hidden Markov model. Clearly the probability of outputting d symbols consecutively from the state and then leave the state is $f(d) = p^{d-1}(1-p)$. What is the **expected** length of output from the state (i.e. calculate $\sum_d df(d)$.

Problem 4

Construct a **constant** space Viterbi algorithm for the four state gene finding HMM (analogous to Hirschberg's algorithm for alignment). What is its running time complexity?

Optional Problems

1. Draw the state space diagram for a gene finding HMM that will not allow stop codons to span introns.

2. Derive an algorithm for **sampling** a state path through an HMM with probability proportional to the probability (weight) of the path. What is the complexity of the algorithm.

3. (Possible final project for the class). Download the latest set of RefSeq genes from `genome.ucsc.edu` and BLAST the genes to construct a non-redundant parameter training set for a human or mouse HMM gene finder.

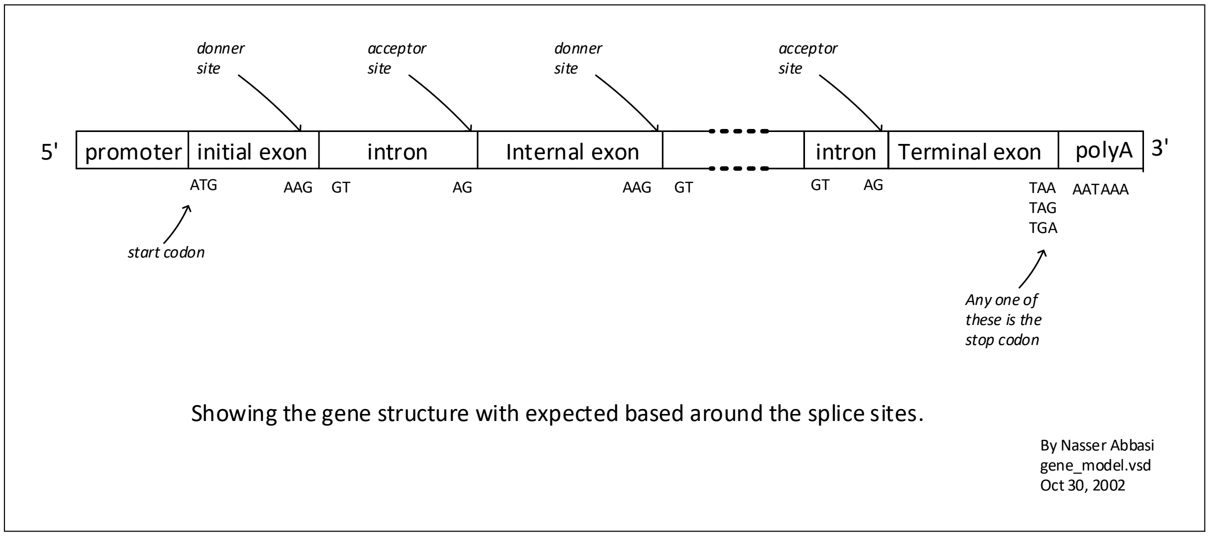


Figure 1: Gene model

2 Problem 1

MATH 127
HW 4
Problem 1

Nasser Abbasi

$\frac{+35}{40}$

problem 1.

let S be state when in cube. let T be state when in tetrahedron.

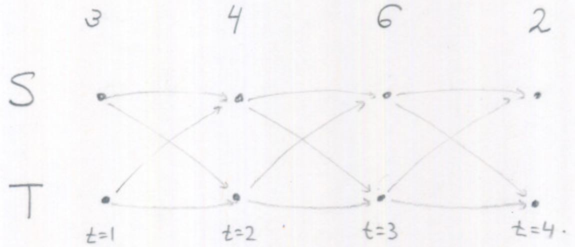
$\pi(S) = \frac{1}{2}$; $\pi(T) = \frac{1}{2}$

$P_{SS} = \frac{1}{2}$, $P_{TT} = \frac{1}{2}$, $P_{ST} = \frac{1}{2}$, $P_{TS} = \frac{1}{2}$

$b_S('k') = \frac{1}{6}$ which is prob. to emit symbol 'k' when in state S

$b_T('k') = \frac{1}{4}$ 〃 T

+7



To find the prob. of emitting {3, 4, 6, 2} use the Forward algorithm $\alpha(i,t)$.

$\alpha(i,t)$ means the prob of being in state 'i' at time 't'

$$P(3,4,6,2) = \alpha(S, t=4) + \alpha(T, t=4)$$

to calculate α for state S at t=4 start at t=1 and goforword

$$\alpha(S,1) = \pi_S \; b_S('3')$$
$$\alpha(S,2) = [\alpha(S,1) P_{SS} + \alpha(T,1) P_{TS}] \; b_S('4')$$
$$\alpha(S,3) = [\alpha(S,2) P_{SS} + \alpha(T,2) P_{TS}] \; b_S('6')$$
$$\alpha(S,4) = [\alpha(S,3) P_{SS} + \alpha(T,3) P_{TS}] \; b_S('2')$$

$b_S('x')$ means the emission prob. of emitting 'x' when in state 'S'

⟶

To calculate α for state T up to $t=4$

$$\alpha(T,1) = \pi_T \, b_T('3')$$

$$\alpha(T,2) = \left[\alpha(T,1) P_{TT} + \alpha(S,1) P_{ST}\right] b_T('4')$$

$$\alpha(T,3) = \left[\alpha(T,2) P_{TT} + \alpha(S,2) P_{ST}\right] b_T('6')$$

$$\alpha(T,4) = \left[\alpha(T,3) P_{TT} + \alpha(S,3) P_{ST}\right] b_T('2')$$

now, $\alpha(T,1) = \frac{1}{2} \cdot \frac{1}{4} = \frac{1}{8}$ $\boxed{0.1250}$

$\alpha(S,1) = \frac{1}{2} \cdot \frac{1}{6} = \frac{1}{12}$ $\boxed{0.0833}$

so $\alpha(S,2) = \left[\frac{1}{12} \cdot \frac{1}{2} + \frac{1}{8} \cdot \frac{1}{2}\right] \frac{1}{6} = \left(\frac{1}{24} + \frac{1}{16}\right) \frac{1}{6} = \frac{5}{288}$ $\boxed{0.0174}$

$\alpha(T,2) = \left[\frac{1}{8} \frac{1}{2} + \frac{1}{12} \frac{1}{2}\right] \frac{1}{4} = \left(\frac{1}{16} + \frac{1}{24}\right) \frac{1}{4} = \frac{5}{192}$ $\boxed{0.0260}$

$\alpha(S,3) = \left[\frac{5}{288} \frac{1}{2} + \frac{5}{192} \frac{1}{2}\right] \frac{1}{6} = \frac{25}{6912}$ $\boxed{0.0036}$

$\alpha(T,3) = \left[\frac{5}{192} \frac{1}{2}{}^{\cdot 0} + \frac{5}{288} \frac{1}{2}{}^{\cdot 0}\right] \frac{1}{4} = \cancel{\frac{25}{4608}} \approx 0$

$\alpha(S,4) = \left[\frac{25}{6912} \frac{1}{2} + \frac{25}{4608} \frac{1}{2}\right] \frac{1}{6} = \frac{\cancel{1}25}{165888} = \frac{25}{576 \cdot 144}$

-3

$\alpha(T,4) = \left[\frac{25}{4608} \frac{1}{2} + \frac{25}{6912} \frac{1}{2}\right] \frac{1}{4} = \frac{\cancel{1}25}{110592} \approx \frac{25}{576 \cdot 96}$

$\approx 7.5 \times 10^{-4}$

$$\text{so final Prob} = \frac{125}{165888} + \frac{125}{110592}$$

$$= \frac{625}{331776} = \boxed{\cancel{0.001883801119}}$$

Dr, I did not use $\log_2$ here, but will in the next problem.

To find most likely sequence of states to have produced the sequence $\{3, 4, 6, 2\}$, I use the Viterbi algorithm. (γ)

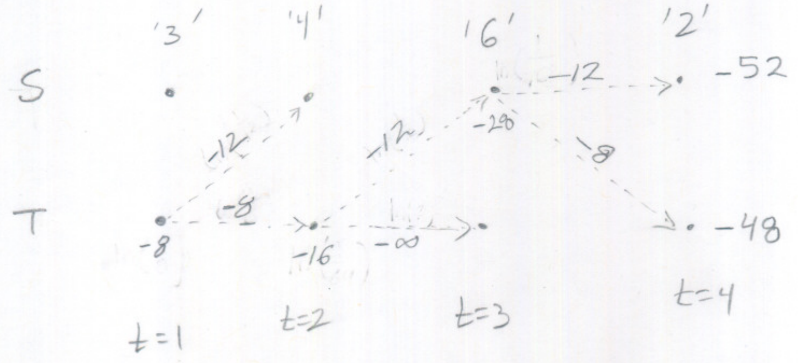
in this algorithm, find the weight on the edge from one state to another, and take the maximum weight at time t. at time t+1, use the max. weight from t in addition to weight to transition.

weight on an edge is $\ln\left(P_{ij}\ b_j(x_t)\right)$

so, at $t = 1$, $\gamma_1(S) = \ln\left(\pi_S\ b_S('3')\right) = \ln\left(\frac{1}{2} \cdot \frac{1}{6}\right) = \ln\left(\frac{1}{12}\right) = -12$

and $\gamma_1(T) = \ln\left(\pi_T\ b_T('3')\right) = \ln\left(\frac{1}{2} \cdot \frac{1}{4}\right) = \ln\left(\frac{1}{8}\right) = -8$

So first state is T since that is the larges one.



at $t = 2$, weight of edge from $T_{t=1}$ to $T_{t=2}$ is $\ln\left(P_{TT}\ b_T('4')\right) = \ln\left(\frac{1}{2} \cdot \frac{1}{4}\right) = \ln\left(\frac{1}{8}\right) = -8$, while weight of edge from $T_{t=1}$ to $S_{t=2}$ is $\ln\left(P_{TS}\ b_S('4')\right) = \ln\left(\frac{1}{2} \cdot \frac{1}{6}\right) = \ln\left(\frac{1}{12}\right) = -12$

So at $t=2$, take max $\begin{cases} -8 \ -8 \\ -8 \ -12 \end{cases} = \max \begin{cases} -16 \\ -20 \end{cases} = -16$

So at $t=2$, state is 'T' since this gives max. →

now at $t=3$, look at edge $T_{t=2} \to T_{t=3}$ and edge $T_{t=2} \to S_{t=3}$.

weight on $T_{t=2} \to T_{t=3} = \ln\left(P_{TT}\ b_T('6')\right) = \ln\left(\frac{1}{2}\cdot \emptyset\right) = \ln(\emptyset) \cong -\infty$

weight on $T_{t=2} \to S_{t=3} = \ln\left(P_{TS}\ b_S('6')\right) = \ln\left(\frac{1}{2}\cdot\frac{1}{6}\right) = \ln\left(\frac{1}{12}\right) = -12$

$$\text{so max} \begin{cases} -28 \quad -\infty \\ -28 \quad -12 \end{cases} = \max \begin{cases} -\infty \\ -40 \end{cases} = -40$$

so state at $t=3$ is 'S' since this is the max.

at $t=4$, look at edges $S_{t=3} \to S_{t=4}$ and $S_{t=3} \to T_{t=4}$.

for weight on $S_{t=3} \to S_{t=4} = \ln\left(P_{SS}\ b_S('2')\right) = \ln\left(\frac{1}{2}\cdot\frac{1}{6}\right) = \ln\left(\frac{1}{12}\right) = -12$

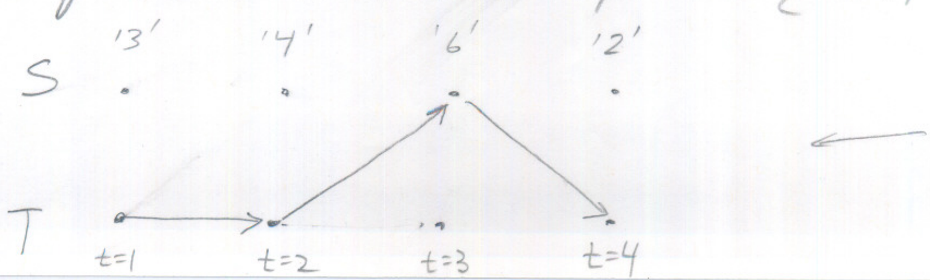
for weight on $S_{t=3} \to T_{t=4} = \ln\left(P_{ST}\ b_T('2')\right) = \ln\left(\frac{1}{2}\cdot\frac{1}{4}\right) = \ln\left(\frac{1}{8}\right) = -8$

$$\text{So max} \begin{cases} -40 \quad -12 \\ -40 \quad -8 \end{cases} = \max \begin{cases} -52 \\ -48 \end{cases} = -48$$

So state at $t=4$ is 'T'

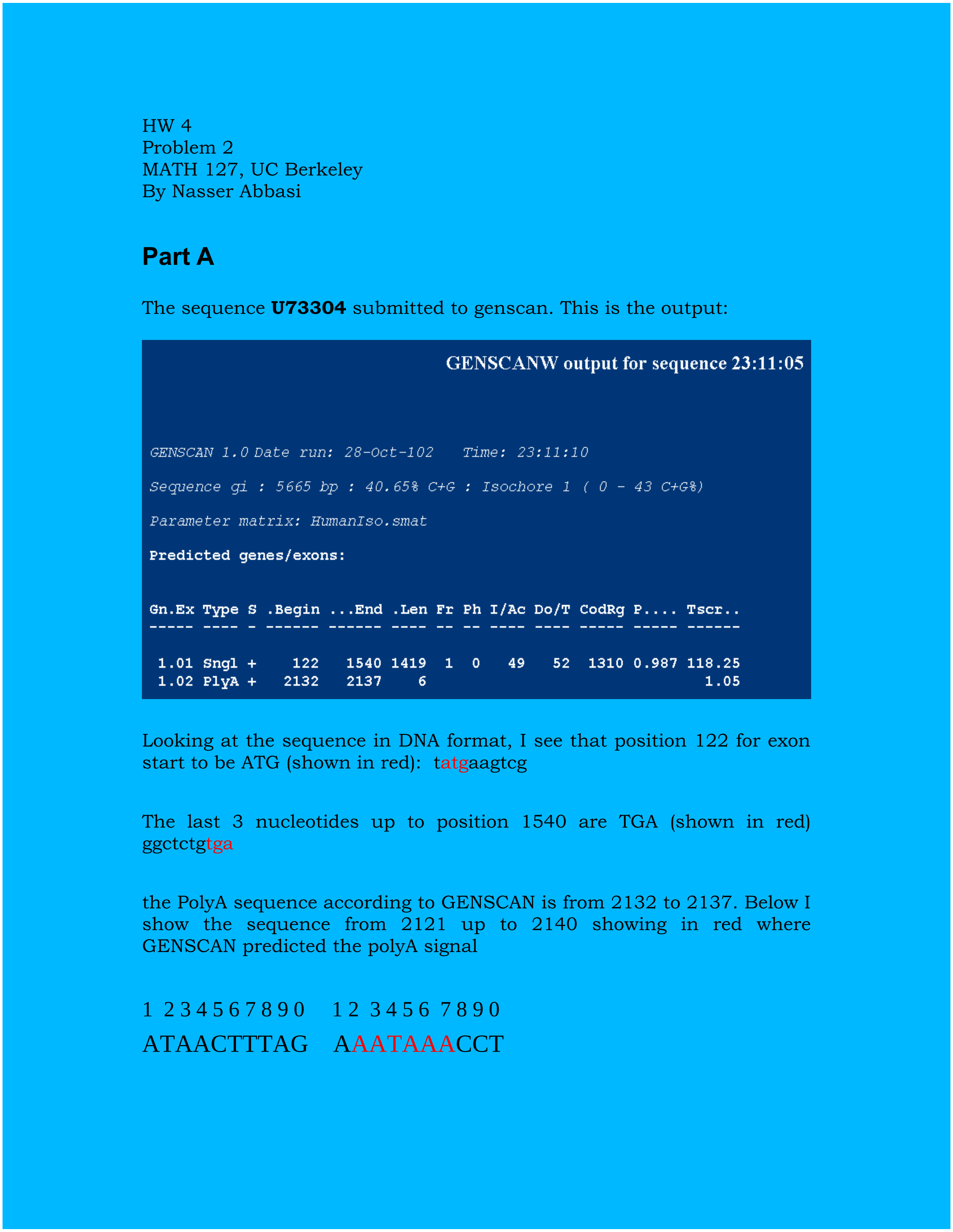
OK

So state sequence is $\boxed{\{T, T, S, T\}}$ the most likely state transitions to have produced $\{3, 4, 6, 2\}$



Final state path.

3 Problem 2



HW 4
Problem 2
MATH 127, UC Berkeley
By Nasser Abbasi

Part A

The sequence **U73304** submitted to genscan. This is the output:

GENSCANW output for sequence 23:11:05

```
GENSCAN 1.0 Date run: 28-Oct-102    Time: 23:11:10

Sequence gi : 5665 bp : 40.65% C+G : Isochore 1 ( 0 - 43 C+G%)

Parameter matrix: HumanIso.smat

Predicted genes/exons:


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

 1.01 Sngl +    122   1540 1419  1  0   49   52  1310 0.987 118.25
 1.02 PlyA +   2132   2137    6                                1.05
```

Looking at the sequence in DNA format, I see that position 122 for exon start to be ATG (shown in red): tatgaagtcg

The last 3 nucleotides up to position 1540 are TGA (shown in red) ggctctgtga

the PolyA sequence according to GENSCAN is from 2132 to 2137. Below I show the sequence from 2121 up to 2140 showing in red where GENSCAN predicted the polyA signal

1 2 3 4 5 6 7 8 9 0   1 2 3 4 5 6 7 8 9 0

ATAACTTTAG   AAATAAACCT

GENSCAN did correctly find the polyA (polyadenylic acid) site. This special consensus signal (AATAAA, which becomes AAUAAA in mRNA) is a special site that is recognized in the pre mRNA during *the splicing process* as to where to cleavage the pre mRNA at to produce the final mRNA.

So this signal is used to know where to cut (cleavage) the pre mRNA at during the splicing process.

Also, during the splicing process, a particular polymerase will recognize this signal and then add about 60-200 Adenylic Acid (A nueclitieds) which as called the A-tail, to the end of this site during a process called polyadenulation. From the reference paper we were given to read (Active Alu Element A-tails: size does matter), it says: “the length of the Alu A-tail is one of the principle factors in determining the retropositional of an Alu element” peptide_1|150_aa

Part b

The sequence **AF276990** submitted to GENSCAN. The result shows that 10 Genes found.

The I went to http://www.ncbi.nlm.nih.gov/BLAST/ and clicked on the link to blastp. Then in the new screen, made sure the data base is set to 'nr' (non-repeats). And copied/paste each of the GENSCAN predicted peptides to the blastp window and run BLASTb. Then when the result is obtained, I clicked on the 'FORMAT' button. Then a new screen comes up which shows all the protein sequences that were matched to the query sequences ordered by decreasing blast score. In the table below I show the score for the top sequence hit from each run.

This is the description of blastp from the NCBI web page (the one I used is the standard protein-protein blast):

Protein BLAST allows one to input protein sequences and compare these against other protein sequences.
Standard protein-protein BLAST - Takes protein sequences in FASTA format, GenBank Accession numbers or GI numbers and compares them against the NCBI protein databases.

The database used to search against is 'nr' which is defined as:

nr
All non-redundant GenBank CDS translations+PDB+SwissProt+PIR+PRF

Matrix used for similarity by blastp is BLOSUM62.

This is the final result shown in the table below.

BLASTP result

GENSCAN peptide number	**Blastp highest score**	**E value**	**LOCUS of the highest matching protein sequence**	**Authors**	**Definition of the highest matching protein sequence**
Peptide_1|150_aa	34	0.43	LOC224881	NCBI Annotation Project	similar to Retrovirus-related POL polyprotein (Endonuclease) [Mus musculus].
Peptide_2|442_aa	660	0.0	KIAA0202	NCBI Annotation Project.	similar to Septin-like protein KIAA0202 [Homo sapiens].
Peptide_3|405_aa	175	9e-43	Ccni	Jensen	cyclin I [Mus musculus].
peptide_4|968_aa	708	0.0	LOC192762	NCBI Annotation Project.	similar to KINESIN-LIKE PROTEIN KIF3A (MICROTUBULE PLUS END-DIRECTED KINESIN MOTOR 3A) [Mus musculus].
peptide_5|131_aa	244	2e-64	AF244915_1	Yang,S.	interleukin-13 [Canis familiaris].
peptide_6|303_aa	N/A	N/A	N/A	N/A	No significant similarity found
peptide_7|64_aa	N/A	N/A	N/A	N/A	No significant similarity found
peptide_8|271_aa	213	1e-54	CAA99729	Offenberg,H.H.	RAD50 homologue hsRAD50 [Homo sapiens].
peptide_9|408_aa	615	e-175	BAA90817	Kitamura	glyceraldehyde-3-phosphate dehydrogenase [Canis familiaris].
peptide_10|847_aa	941	0.0	RAD50	Dolganov	RAD50 homolog isoform 1 [Homo sapiens].

Next, I used tblastn against each peptide to seach for all 6 reading frames. From NCBI web page, this is the definition of tblastn:

Protein query - Translated db [tblastn] - Takes a protein query sequence and compares it against an NCBI nucleotide database which has been translated in all six reading frames.

tblastn result

GENSCAN peptide number	**tBlastn highest score**	**E value**	**LOCUS of the matching Nucleotide sequence**	**Authors**	**Definition of the highest matching Nucleotide sequence**
Peptide_1|150_aa	77	3e-13	AC004775	Kimmerly	Homo sapiens chromosome 5, P1 clone 1308e5 (LBNL H13), complete sequence.
Peptide_2|442_aa	665	0.0	AK057797	Nishi,T.,	Homo sapiens cDNA FLJ25068 fis, clone CBL05137, highly similar to Mus musculus Sep2 mRNA.
Peptide_3|405_aa	191	2e-46	AC004775	Kimmerly	Homo sapiens chromosome 5, P1 clone 1308e5 (LBNL H13), complete sequence.
peptide_4|968_aa	723	0.0	BC032599	Strausberg, R.	Homo sapiens, Similar to kinesin family member 3A, clone IMAGE:5533541, mRNA.
peptide_5|131_aa	259	3e-68	AF244915	Yang,S.,	Canis familiaris interleukin-13 mRNA, complete cds.
peptide_6|303_aa	49	6e-04	AY079157S1	Zangerl	Canis familiaris glucocorticoid receptor DNA binding factor 1 (GRLF1) gene
peptide_7|64_aa	36	0.78	AL845433	Whitehead, S.	Human DNA sequence from clone RP11-674N8 on chromosome X, complete sequence.
peptide_8|271_aa	231	9e-59	HSRAD50	Offenberg, H.H.	H.sapiens mRNA for RAD50.
peptide_9|408_aa	605	e-173	RABGLY3PHO	Applequist	Oryctolagus cuniculus glyceraldehyde-3-phosphate dehydrogenase mRNA, complete cds.
peptide_10|847_aa	1196	0.0	RAD50	Dolganov	Homo sapiens RAD50 homolog (S.cerevisiae) (RAD50), transcript variant 2, mRNA.

To better compare blastp result with tblastn, I show the result in this table.

Blastp and tblastn score comparison

GENSCAN peptide number	**blastp highest score**	**blastP E value**	**tblastn highest score**	**tblastn E value**
Peptide_1|150_aa	34	0.43	77	3e-13
Peptide_2|442_aa	660	0.0	665	0.0
Peptide_3|405_aa	175	9e-43	191	2e-46
peptide_4|968_aa	708	0.0	723	0.0
peptide_5|131_aa	244	2e-64	259	3e-68
peptide_6|303_aa	N/A	N/A	49	6e-4
peptide_7|64_aa	N/A	N/A	36	0.78
peptide_8|271_aa	213	1e-54	231	9e-59
peptide_9|408_aa	615	e-175	605	e-173
peptide_10|847_aa	941	0.0	1196	0.0

Conclusion. To answer the question on which prediction I should believe, I use the blast score and the expect value E as the main criteria. The expect value is defined in NCBI web page. Here is the text

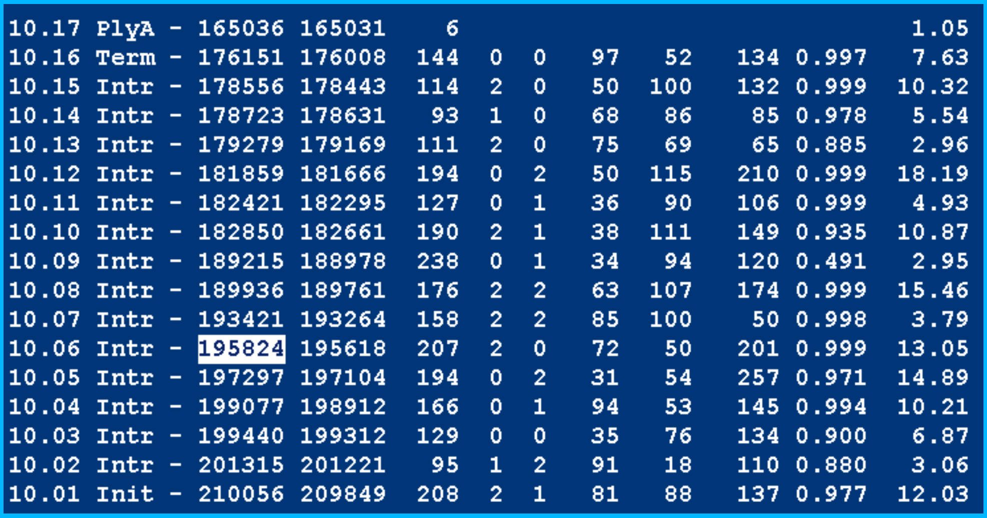
> Q: What is the Expect (E) value?
> The Expect value (E) is a parameter that describes the number of hits one can "expect" to see just by chance when searching a database of a particular size. It decreases exponentially with the Score (S) that is assigned to a match between two sequences. Essentially, the E value describes the random background noise that exists for matches between sequences. For example, an E value of 1 assigned to a hit can be interpreted as meaning that in a database of the current size one might expect to see 1 match with a similar score simply by chance. **This means that the lower the E-value, or the closer it is to "0" the more "significant" the match is.** However, keep in mind that searches with short sequences, can be virtually indentical and have relatively high EValue. This is because the calculation of the E-value also takes into account the length of the Query sequence. This is because shorter sequences have a high probability of occuring in the database purely by chance.

The higher the scores and the lower the E values, the more belivalble the prediction will be. Since this means GENSCAN did produce a sequence which actually exist in the database and documented to high similarity score.

This table below is my final result of the prediction by GENSCAN.

GENSCAN peptide number	**True or false prediction?**	**WHY?**
Peptide_1|150_aa	FALSE	Low score and E values from both blastp and tblastn
Peptide_2|442_aa	TRUE	High score and E value from both blastp and tblastn. Documented protein sequence.
Peptide_3|405_aa	TRUE	As above
peptide_4|968_aa	TRUE	As above
peptide_5|131_aa	TRUE	As above
peptide_6|303_aa	FALSE	Blastp failed to find a significant match, tblastn low score.
peptide_7|64_aa	FALSE	As above
peptide_8|271_aa	TRUE	High score and E value from both blastp and tblastn. Documented protein sequence.
peptide_9|408_aa	TRUE	As above.
peptide_10|847_aa	TRUE	As above.

Part C

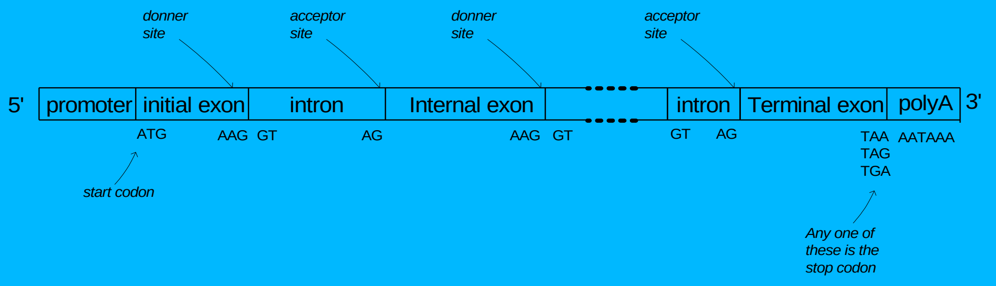


```
10.17 PlyA - 165036 165031    6                                 1.05
10.16 Term - 176151 176008  144  0  0   97   52   134 0.997    7.63
10.15 Intr - 178556 178443  114  2  0   50  100   132 0.999   10.32
10.14 Intr - 178723 178631   93  1  0   68   86    85 0.978    5.54
10.13 Intr - 179279 179169  111  2  0   75   69    65 0.885    2.96
10.12 Intr - 181859 181666  194  0  2   50  115   210 0.999   18.19
10.11 Intr - 182421 182295  127  0  1   36   90   106 0.999    4.93
10.10 Intr - 182850 182661  190  2  1   38  111   149 0.935   10.87
10.09 Intr - 189215 188978  238  0  1   34   94   120 0.491    2.95
10.08 Intr - 189936 189761  176  2  2   63  107   174 0.999   15.46
10.07 Intr - 193421 193264  158  2  2   85  100    50 0.998    3.79
10.06 Intr - 195824 195618  207  2  0   72   50   201 0.999   13.05
10.05 Intr - 197297 197104  194  0  2   31   54   257 0.971   14.89
10.04 Intr - 199077 198912  166  0  1   94   53   145 0.994   10.21
10.03 Intr - 199440 199312  129  0  0   35   76   134 0.900    6.87
10.02 Intr - 201315 201221   95  1  2   91   18   110 0.880    3.06
10.01 Init - 210056 209849  208  2  1   81   88   137 0.977   12.03
```

For 10th prediction. To help me solve this I needed to find what happens around the splice sites. This is the exon/intron boundaries. Looking at a gene from 5' to 3', the following is typically found

1. Initial exon starts with ATG
2. Exon ends with AAG. (Donor site)
3. Intron starts with GT.
4. Intron ends with AG (Acceptor site).
5. Terminal exon (last exon in a gene) ends with TAA or TAG or TGA.
6. PolyA is AATAAA.

I drew a diagram to help illustrate the above:



Showing the gene structure with expected based around the splice sites.

By Nasser Abbasi
gene_model.vsd
Oct 30, 2002

So, for each exon/intron boundaries as predicted by GENSCAN, I verified if the above is correct or not. I put the result in this table. In this table, I show for each exon the 2 bases at the end of the codon before, the codon at the start and end of the exon, and the 2 bases at the start of the next intron. If those value meet the diagram above, then I call the prediction correct. Note that GENSCAN found this 10th gene on the reverse strand, so in this table below I show the on both strands, then in the final table I show it from 5’ to 3’ sense to make it easier to compare with the above diagram.

exon	Exon position	2 bases at end of previous intron	Codon at start of this exon	Codon at end of this exon	2 bases at start of next codon
Init	210056: 209849	N/A	5’ CAT 3’ 3’ GTA 5’	5’ CTC 3’ 3’ GAG 5’	5’ AC 3’ 3’ TG 5’
Internal	201315: 201221	5’ CT 3’ 3’ GA 5’	5’ AAT 3’ 3’ TTA 5’	5’ CTG 3’ 3’ GAC 5’	5’ CT 3’ 3’ GA 5’
Internal	199440: 199312	5’ CT 3’ 3’ GA 5’	5’ ATT 3’ 3’ TAA 5’	5’ CTT 3’ 3’ GAA 5’	5’ AC 3’ 3’ TG 5’
Internal	199077: 198912	5’ CT 3’ 3’ GA 5’	5’ AAC 3’ 3’ TTC 5’	5’ CCT 3’ 3’ GGA 5’	5’ AC 3’ 3’ TG 5’
internal	197297: 197104	5’ CT 3’ 5’ GA 5’	5’ GAC 3’ 3’ CTG 5’	5’ CAT 3’ 3’ GTA 5’	5’ AC 3’ 3’ TG 5’
Internal	195824: 195618	5’ CT 3’ 3’ GA 5’	5’ ATT 3’ 3’ GAA 5’	5’ AGC 3’ 3’ TCG 5’	5’ AC 3’ 3’ TG 5’
Internal	193421: 193264	5’ CT 3’ 3’ GA 5’	5’ AGC 3’ 3’ TCG 3’	5’ TTC 3’ 3’ AAG 5’	5’ AC 3’ 3’ TG 5’
Internal	189936: 189761	5’ CT 3’ 3’ GA 5’	5’ TTG 3’ 3’ AAC 5’	5’ CTC 3’ 3’ GAG 5’	5’ AC 3’ 3’ TG 5’
Internal	189215: 188978	5’ GA 3’ 3’ CT 5’	5’ TAA 3’ 3’ GTT 5’	5’ CTC 3’ 3’ GAG 3’	5’ AC 3’ 3’ TG 5’
Internal	182850: 182661	5’ CT 3’ 3’ GA 5’	5’ TGC 3’ 3’ ACG 5’	5’ CTG 3’ 3’ GAG 5’	5’ AC 3’ 3’ TG 5’
Internal	182421: 182295	5’ TC 3’ 3’ AG 5’	5’ CAT 3’ 3’ GTA 3’	5’ ACC 3’ 3’ TGG 5’	5’ TT 3’ 3’ AA 5’
Internal	181859: 181666	5’ CT 3’ 3’ GA 5’	5’ AAA 3’ 3’ TTT 5’	5’ CTT 3’ 3’ GAA 5’	5’ AC 3’ 3’ TG 5’
Internal	179279: 179169	5’ CT 3’ 3’ GA 5’	5’ ATC 3’ 3’ TAG 5’	5’ TTT 3’ 3’ AAA 5’	5’ AC 3’ 3’ TG 5’
Internal	178723: 178631	5’ CT 3’ 3’ GA 5’	5’ CAT 3’ 3’ GTA 5’	5’ CTT 3’ 3’ GAA 5’	5’ AC 3’ 3’ TG 5’
Internal	178556: 178443	5’ CT 3’ 3’ GA 5’	5’ TTG 3’ 3’ AAC 5’	5’ CTT 3’ 3’ GAA 5’	5’ AC 3’ 3’ TG 5’
Terminal	176151: 176008	5’ CT 3’ 3’ GA 5’	5’ AAT 3’ 3’ TTA 5’	5’ TCA 3’ 3’ AGT 5’	5’ GC 3’ 3’ CG 5’
PolyA	165036: 165031	5’ AG 3’ 3’ TC 5’	5’ TTTATT 3’ 3’ AAATAA 5’	N/A	N/A

Now I show the above table, but list everything from 5’ to 3’ sense. I.e. when looking at 3’ ATG 5’, I list it now as 5’ GTA 3’

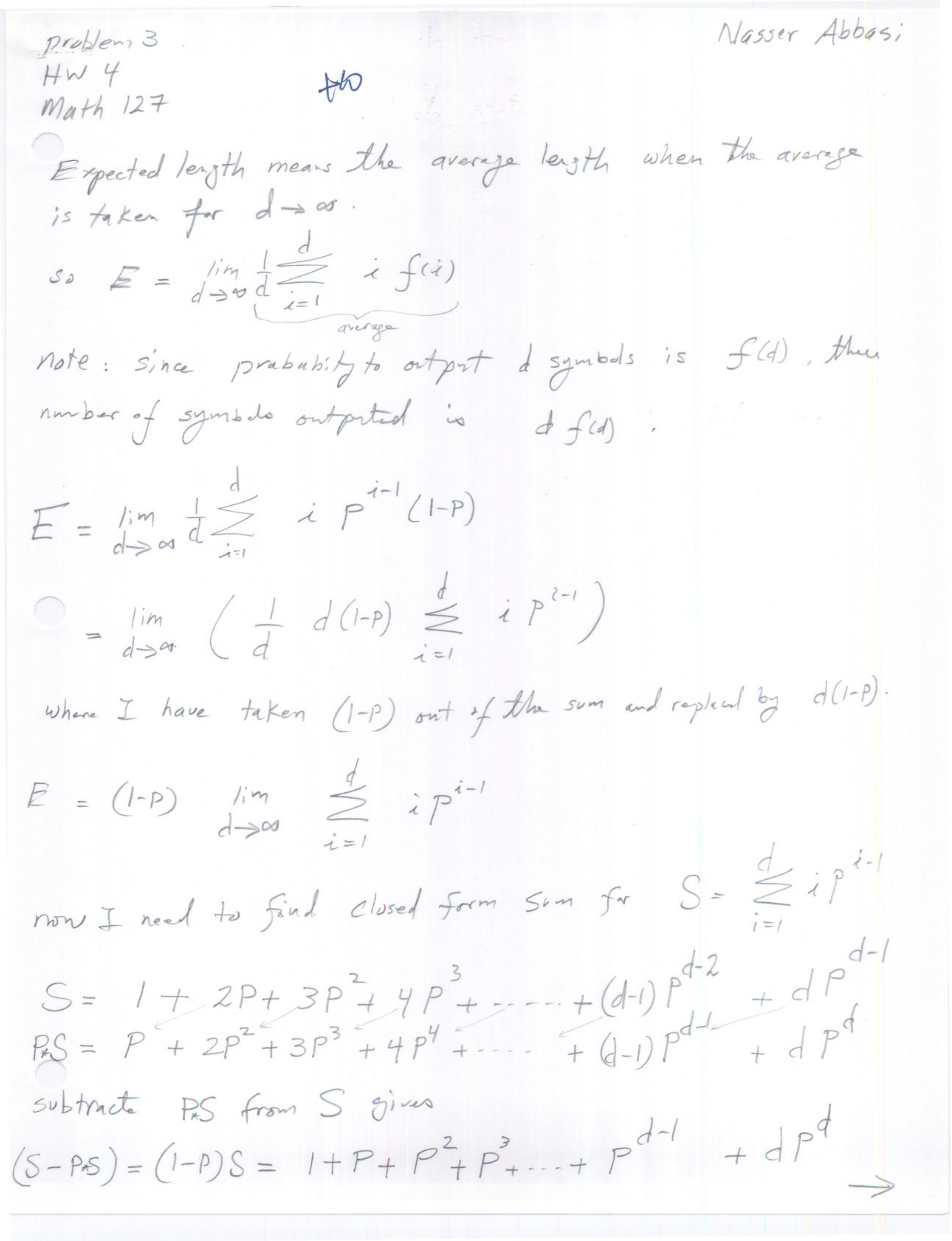
exon	2 bases at end of previous intron	Codon at start of this exon	Codon at end of this exon	2 bases at start of next codon	GENSCAN probability	Correct prediction?
Init	N/A	ATG	GAG	GT	12.03	YES
Internal	AG	ATT	CAG	AG (error)	3.06	NO
Internal	AG	AAT	AAG	GT	6.87	YES
Internal	AG	CTT	AGG	GT	10.21	YES
Internal	AG	GTC	ATG	GT	14.89	YES
Internal	AG	AAG	GCT	GT	13.05	YES
Internal	AG	GCT	GAA	GT	3.79	YES
Internal	AG	CAA	GAG	GT	15.46	YES
Internal	TC (error)	TTG	GAG	GT	2.95	NO
Internal	AG	GCA	GAG	GT	10.87	YES
Internal	GA(error)	ATG	GGT	AA (error)	4.93	NO
Internal	AG	TTT	AAG	GT	18.19	YES
Internal	AG	GAT	AAA	GT	2.96	YES
Internal	AG	ATG	AAG	GT	5.54	YES
Internal	AG	CAA	AAG	GT	10.32	YES
Terminal	AG	ATT	TGA	GC	7.63	YES
PolyA	N/A	AATAAA	N/A	N/A	1.05	YES

Some observation: From what I understood, the codon at the end of each internal exon should be AAG. However from the table above, this does not show to be the case, so since I am not sure if this rule is correct now, I will not use it.

I will only use the rules that says that the start of each intron after an exon should be GT and the end of each intron just before an exon start should be AG.

Based on these two rules, I see that GENSCAN did not correctly predict 3 exons. These are the ones where I wrote an ‘error’ next to them in the above table. Also notice that the ones that GENSCAN did not predict correctly has LOW probability of less than 5.

4 Problem 3



problem 3
HW 4
Math 127

Nasser Abbasi

+10

Expected length means the average length when the average is taken for $d \to \infty$.

so $$E = \lim_{d\to\infty} \underbrace{\frac{1}{d} \sum_{i=1}^{d} i\, f(i)}_{\text{average}}$$

note: since probability to output d symbols is $f(d)$, then number of symbols outputted is $d\, f(d)$.

$$E = \lim_{d\to\infty} \frac{1}{d} \sum_{i=1}^{d} i\, p^{i-1} (1-p)$$

$$= \lim_{d\to\infty} \left(\frac{1}{d}\, d(1-p) \sum_{i=1}^{d} i\, p^{i-1} \right)$$

where I have taken $(1-p)$ out of the sum and replaced by $d(1-p)$.

$$E = (1-p) \lim_{d\to\infty} \sum_{i=1}^{d} i\, p^{i-1}$$

now I need to find closed form sum for $S = \sum_{i=1}^{d} i\, p^{i-1}$

$$S = 1 + 2P + 3P^2 + 4P^3 + \cdots + (d-1)P^{d-2} + dP^{d-1}$$
$$P{*}S = P + 2P^2 + 3P^3 + 4P^4 + \cdots + (d-1)P^{d-1} + dP^d$$

subtract $P{*}S$ from S gives

$$(S - P{*}S) = (1-P)S = 1 + P + P^2 + P^3 + \cdots + P^{d-1} + dP^d$$

→

So

$$(1-P)\,S = 1 + P + P^2 + \cdots + dP^d$$

Now, since $0 \le P \le 1$, then the sum on the right is convergent and equalls $\frac{1}{1-P}$ in the limit $d \to \infty$.

So $$(1-P)\,S = \frac{1}{(1-P)}$$

$$S = \frac{1}{(1-P)^2}$$

So $$E = (1-P)\frac{1}{(1-P)^2} = \boxed{\frac{1}{1-P}}$$ ok

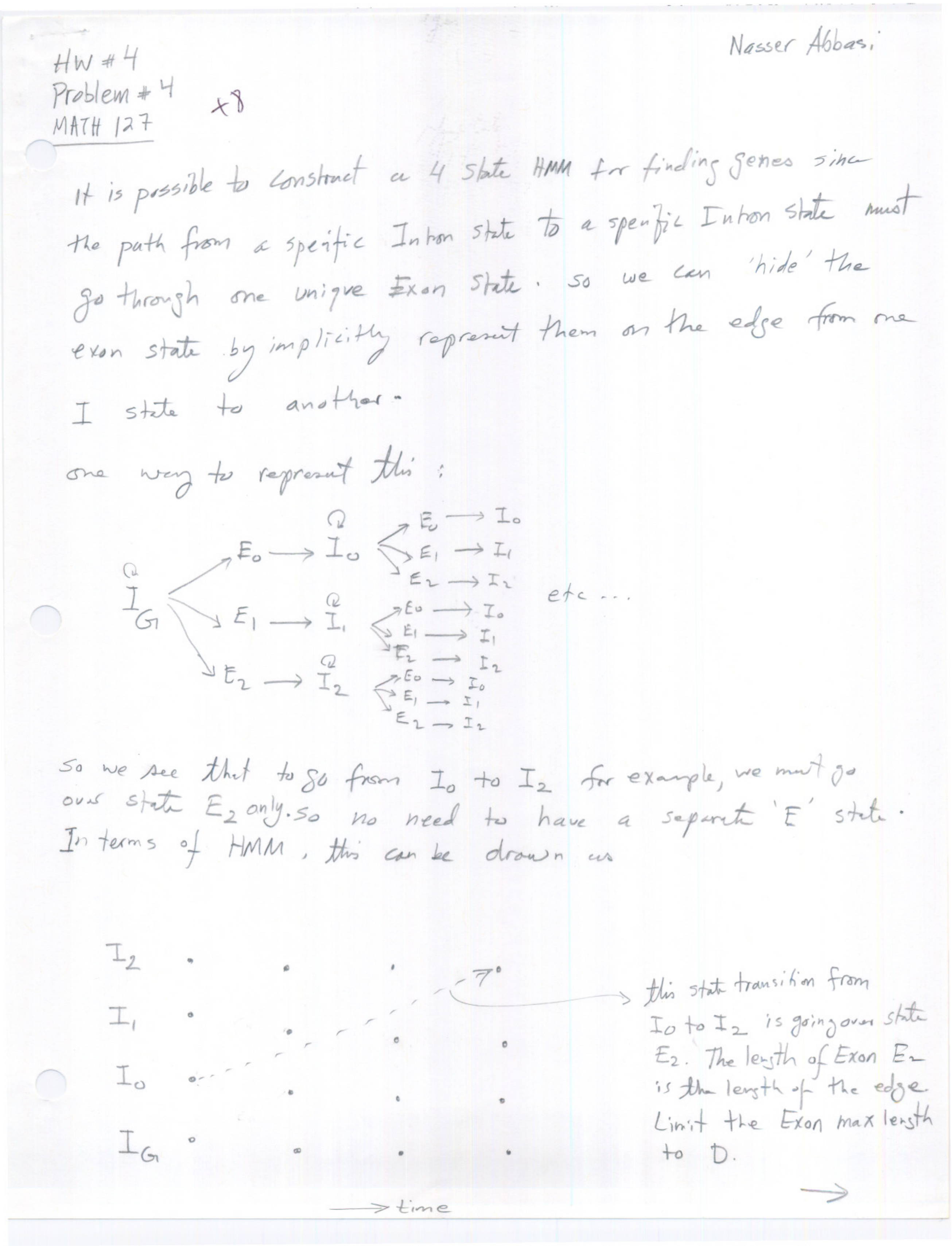
So expected length of output from state is $\frac{1}{1-P}$.

For example, for

$$\begin{aligned}
P = .1 &\Rightarrow E = 1.1111\\
P = .2 &\Rightarrow E = 1.25\\
P = .3 &\Rightarrow E = 1.428557\\
P = .4 &\Rightarrow E = 1.66\\
P = .5 &\Rightarrow E = 2\\
P = .6 &\Rightarrow E = 2.5\\
P = .7 &\Rightarrow E = 3.33\\
P = .8 &\Rightarrow E = 5\\
P = .9 &\Rightarrow E = 10\\
P = .99 &\Rightarrow E = 100
\end{aligned}$$

This makes sense, since it says the higher the prob to be in the state, the more symbols one will expect to output.

5 Problem 4



HW # 4
Problem # 4
MATH 127

+8

Nasser Abbasi

It is possible to construct a 4 State HMM for finding genes since the path from a speific Intron state to a specific Intron state must go through one unique Exon State. so we can 'hide' the exon state by implicitly represent them on the edge from one I state to another.

one way to represent this :

I_G
E_0
E_1
E_2
I_0
I_1
I_2
$E_0 \longrightarrow I_0$
$E_1 \longrightarrow I_1$
$E_2 \longrightarrow I_2$
etc ...

So we see that to go from I_0 to I_2 for example, we must go over state E_2 only. so no need to have a separate 'E' state. In terms of HMM, this can be drawn as

this state transition from I_0 to I_2 is going over state E_2. The length of Exon E_2 is the length of the edge. Limit the Exon max length to D.

→

at each intron state emit a sequence of bases, depending on the length of the edge.

since we set max length of exon to be D, we need to ony look D time steps (or bases) back when trying to find the max vertex weight in order to add to it the transition weight for the Viterbi algorithm.

recall That for the Viterbi algorithm.

example for i..k states

$$\delta(\underset{\text{time}}{t}, \underset{\text{state}}{i}) = \max \begin{cases} \delta(t-1, i) + P_{ii} \; b_i(\text{symbol at time } t) \\ \delta(t-1, j) + P_{ji} \; b_i(\text{symbol at time } t) \\ \vdots \\ \delta(t-1, k) + P_{ki} \; b_i(\qquad) \end{cases}$$

i.e. find the max at time t, by looking at the max at time $(t-1)$ plus the transition weight and emission weight to current state.

since now we want to account for Exon Of max length D, need to look D long ago. so Viterbi becomes

we look upto D time steps back

example for 2 states i, j only

$$\delta(t, i) = \max \begin{cases} \delta(t-1, i) + P_{ii} \; b_i(\text{symbol at time } t) \\ \delta(t-1, j) + P_{ji} \; b_i(\text{symbol at time } t) \\ \delta(t-2, i) + P_{ii} \; b_i(\qquad) \\ \delta(t-2, j) + P_{ji} \; b_i(\qquad) \\ \delta(t-3, i) + P_{ii} \; b_i(\qquad) \\ \delta(t-3, j) + P_{ji} \; b_i(\qquad) \\ \vdots \\ \delta(t-D, i) + P_{ii} \; b_i(\qquad) \\ \delta(t-D, j) + P_{ji} \; b_i(\qquad) \end{cases}$$

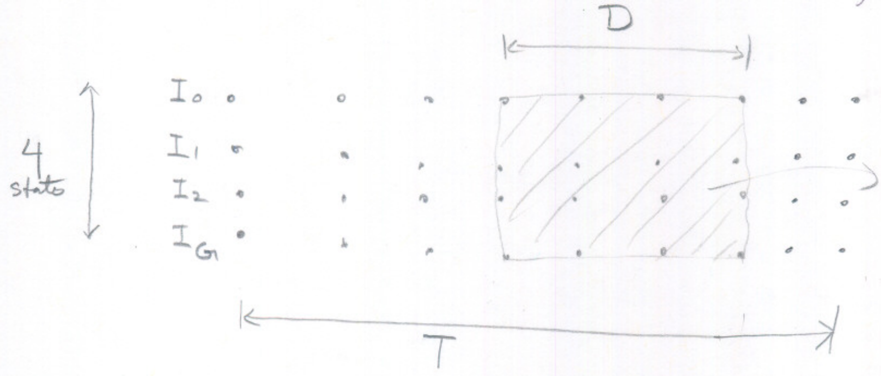
since here we have 4 states only, then for T long symbols this method will take time complexity $O(D\,4^2\,T) = O(16DT)$ →

16 DT because at each time step we have to look at (4 × D) states per each state. but we have 4 state, so we get 16×D. and since there are T positions or time steps, we get 16 DT ⇒ $O(DT)$

Now, I will show how to construct a constant space Viterbi algorithm for this HMM. using the idea from Hirschberg space improvemt where one column in the matrix is only needed and reused over and over, here I will have a space to store only $\underline{\underline{D \times 4}}$ states.

good.

looking at this diagram



only need this space

at each time step, the square is "shifted" one step to the right.

so for a given D (say 500 as max Exon length allowed) space complexity is 2000 or constant.

i.e I only need to keep storage to remember $\delta(t,i)$ for D steps back and for 4 states. so constant space.

-2

the running time complexity remains as shown befor $O(DT)$ where T is the total length of the sequence.

use divide and conquer strategy

QED