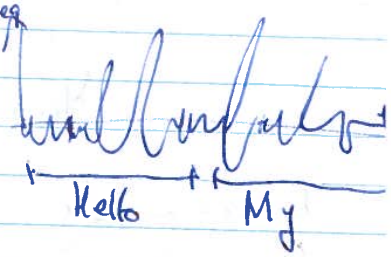
Hidden Markov Models (HMM)

Goal: "Decode" a sequence of observations

Speech recognition : Observation:



Biological sequence annotation : Observation: ACTACTAGGTAC

Genes

Protein: VLIYVRRCY

α-helix β-sh

Toy example: You visit a city with different neighborhoods.

set of states = $S = \{F, E, C\} = S = \{S_1, S_2, \ldots S_n\}$

F: French, E: English, C: Chinese

You walk randomly in city.

Every minute, someone greets you.

set of symbols = $\Sigma = \{b, h, n, a\} = \{\sigma_1, \sigma_2, \ldots \sigma_k\}$

b: bonjour, h: hello, n: chinese, a: namaste

You record seq. of greetings: $X = x_1, x_2, \ldots x_L$

where $x_i \in \Sigma$ $\quad X = b\,b\,a\,a\,h\,h\,h\,a\,h\,b \ldots$

Problem: Given: $X = x_1 \ldots x_n$

Find: Path $P = p_1 \ldots p_n$ that is most likely given X

We need to know:

① Emission Probabilities: $Pr_e[x_i = \alpha \mid p_i = \beta]$ ($\alpha \in \Sigma$, $\beta \in S$)

States \ Symbols	b	h	n	a
F	0.9	0.05	0.05	0
E	0.1	0.5	0.1	0.3
C	0.3	0.3	0.3	0.1

→ sum to 1

② Transition probabilities

$Pr_\varepsilon[P_{i+1} = \gamma \mid P_i = \beta]$

ε S

Destination

Source	E	C	F
E	0.9	0	0.1
C	0.1	0.4	0.5
F	0.3	0.1	0.6

Path: E EEEEEEEE FFFFEE FFFCCC

③ Initial State probability

$Pr[P_1 = \gamma]$

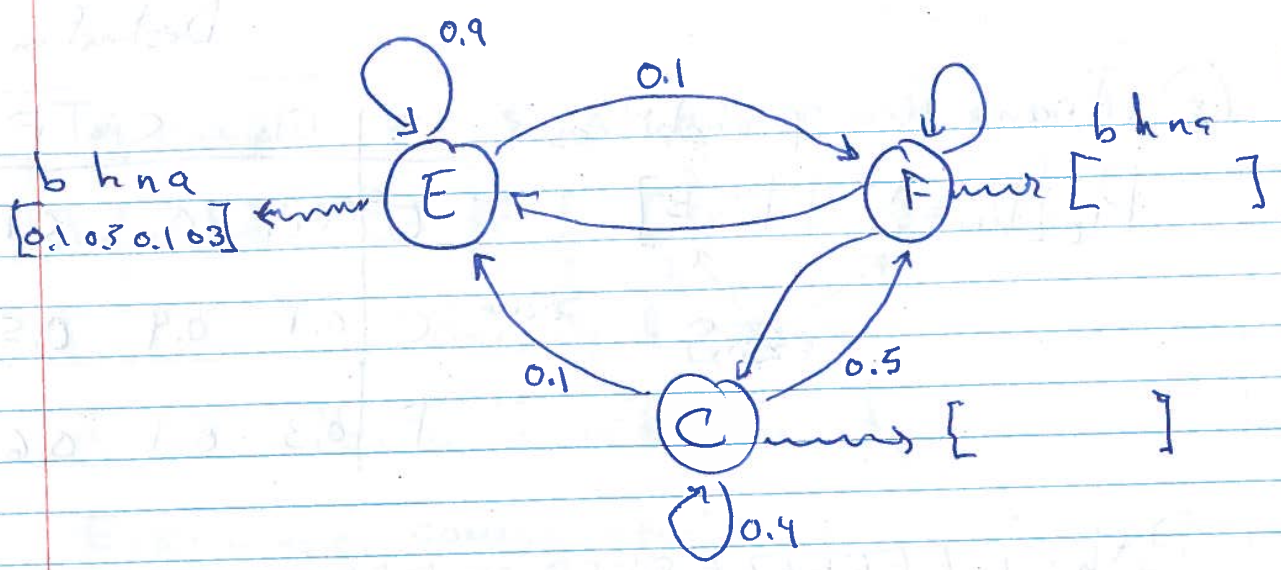
ε S

E	C	F
0.5	0	0.5

Assumptions: - Markovian assumption (1st order)

Prob of going to state γ only depends on where you are now, not on how you got there

- Observations are independent from each other (given the states)



HMM as generative model: Generate a random of sequence

① Pick randomly P_1 from the initial state prob. dist

Repeat: 2.1 Emit an observation from current state

2.2 Transition to new state acc. to transition prob. matrix

Questions (Assume we know S, Σ, emission prob, transition prob, initial prob)

① Maximum likelihood path:

Given: $X = x_1 \dots x_L$

Find: Path $P = P_1 \dots P_L$ that is most likely to have generated X, ie. $Pr[P_1 \dots P_L \mid x_1 \dots x_L]$ is maximized

Answer: Viterbi algo.

② Posterior decoding

Given: $X = x_1 \dots x_L$, time i, state β

Calculate: $Pr[P_i = \beta \mid X]$

Answer: Forward-Backward algo

③ Estimation Problem: Assume we know S, Σ but not emission prob, not transition prob, not initial

Given: $X = x_1 \ldots x_k$

Find: Emission prob, Transit prob, Initial prob } such that $Pr[X \mid E, T, I]$ is max

Answer: Baum-Welch algorithm

Viterbi Algorithm:

Given: $X = x_1 \ldots x_k$ (seq. of k observations)

(S, Σ, E, T, I) = HMM

- S: states
- Σ: alphabet
- E: emission prob matrix
- T: transition prob matrix
- I: initial state prob. vector

Find: $P = P_1 \ldots P_k$, where $P_i \in S$

such that $Pr[P_1 \ldots P_k \mid X = x_1 \ldots x_n]$ is maximized

Question 1: How to calculate $Pr[P_1 \dots P_L \mid X = x_1 \dots x_L]$?
for a given path $P = P_1 \dots P_L$

$$Pr[P_1 \dots P_L \mid X = x_1 \dots x_L] = \frac{Pr[P_1 \dots P_L \wedge X = x_1 \dots x_L]}{Pr[X = x_1 \dots x_L]}$$

Independent of Path P

$$Pr[P_1 \dots P_L \wedge X = x_1 \dots x_L] = Pr[X = x_1 \dots x_L \mid P_1 \dots P_L] \cdot Pr[P_1 \dots P_L]$$

$$= \left(\prod_{i=1}^{L} Pr_E[x_i \mid P_i] \right) \cdot Pr_I(P_1) \cdot \prod_{i=1}^{L-1} Pr_T[P_{i+1} \mid P_i]$$

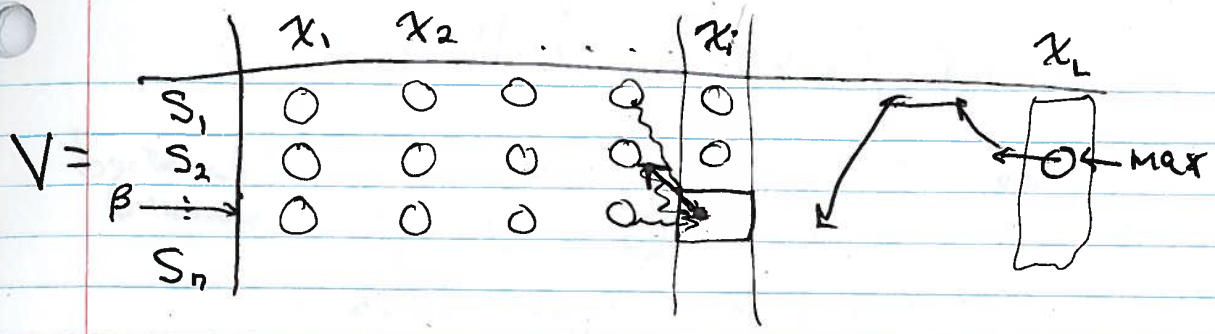
Question 2: How to find P s.t. $Pr[P_1 \dots P_L \mid x_1 \dots x_L]$ is max?

$\Rightarrow$ Viterbi algorithm

Define $V(\beta, i)$ = Prob. of the most likely path of length i, given observation $x_1 \dots x_i$, assuming path ends in state β

$\beta \in S$, $i \in \{1 \dots L\}$

$$= \max_{P_1 \dots P_i \text{ where } P_i = \beta} \{ Pr[P_1 \dots P_i, x_1 \dots x_i] \}$$



$$V(\beta, i) = \max_{\gamma \in S} \{ V(\gamma, i-1) \cdot Pr_t[\beta | \gamma] \} \cdot Pr_e[x_i | \beta]$$

Fill V table column by column, left-to-right

$V(\beta, 1) = Pr_I(\beta) \cdot Pr_e[x_1 | \beta]$ (Initialization)

$$\max_{P_1 \ldots P_L} \{ Pr[P_1 \ldots P_L, x_1 \ldots x_L] \} = \max_{\beta \in S} \{ V(\beta, L) \}$$

To recover path P that maximized $Pr[P_1 \ldots P_L, x_1 \ldots x_L]$

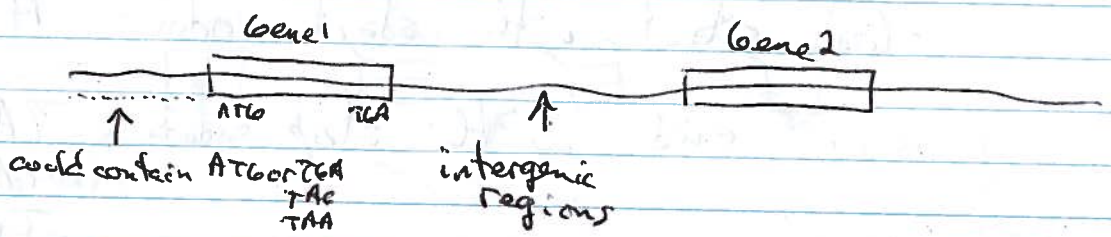
- Trace back dyn. prog algo from $\max \{ V(\beta, L) \}$

Complexity Time = $O(n^2 \cdot L)$

Space = $O(n \cdot L)$

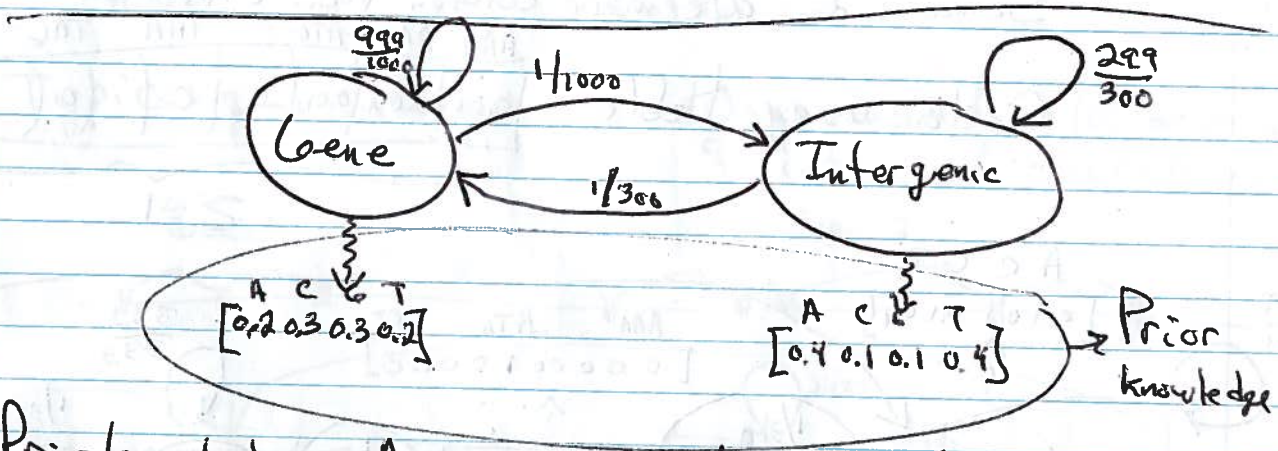
HMMs for Gene Finding

Bacterial genome:



Problem: Given: Genome sequence X

Find: Start/End position of each gene in X



Prior knowledge: Avg. gene length ~ 1000bp

Avg. intergenic length ~ 300bp

Genome X = ATAGATAACAGGCTCGTGGTC | ATAT

Viterbi Path = IIIIIIII | 66666666666666 | IIII

Integ | Gene | Integ

Gene Properties:

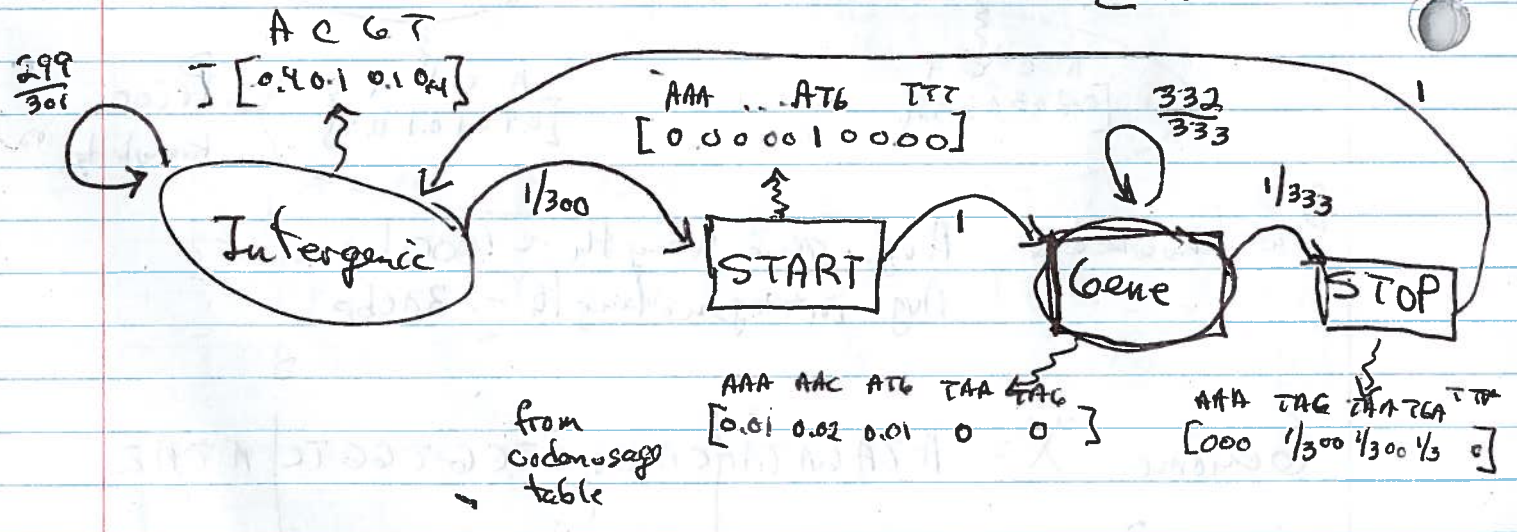
- Gene start with start codon: ATG
- . end with stop codons: TAG
  TAA
  TGA

- Gene is made of codons (triplets of nuc.)
- Some codons are more common than others

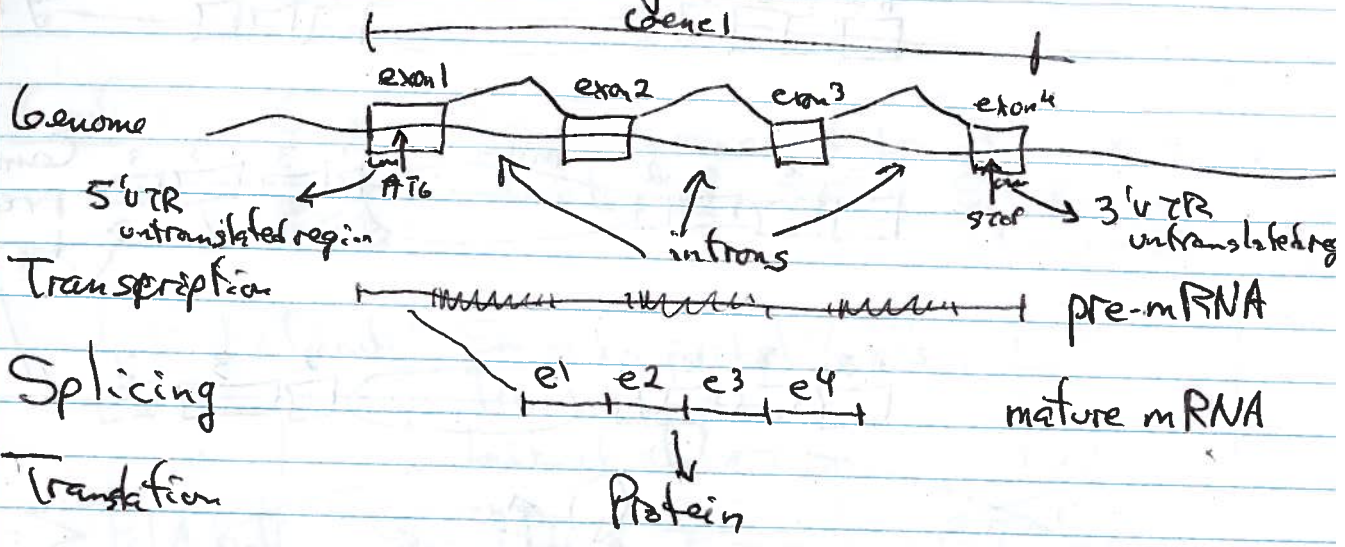
Codon usage table

AAA	AAC	ATG	...	TAA	TAG	TGA
0.01	0.02	0.01		0	0	0

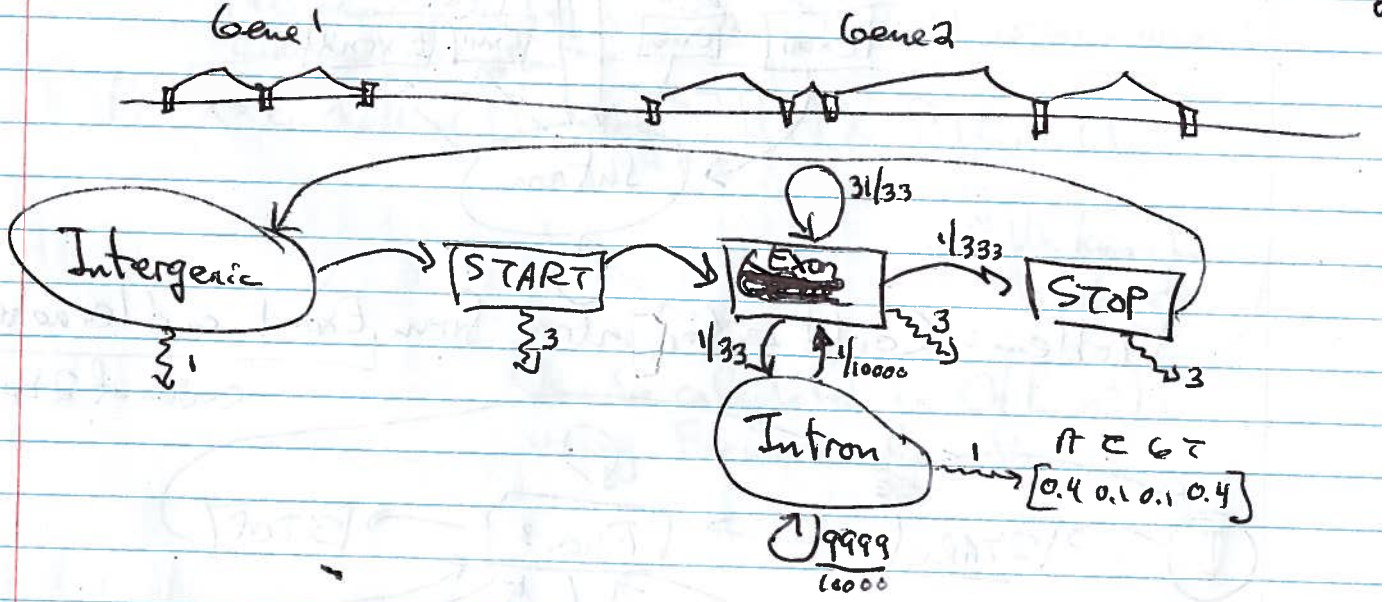
$\Sigma = 1$

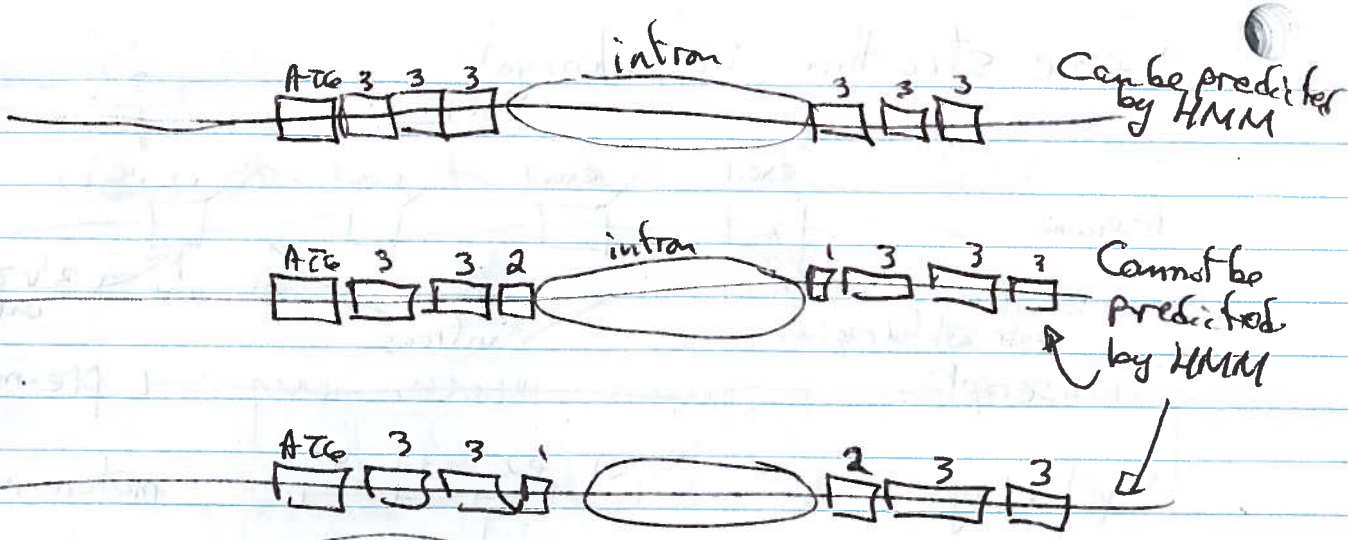


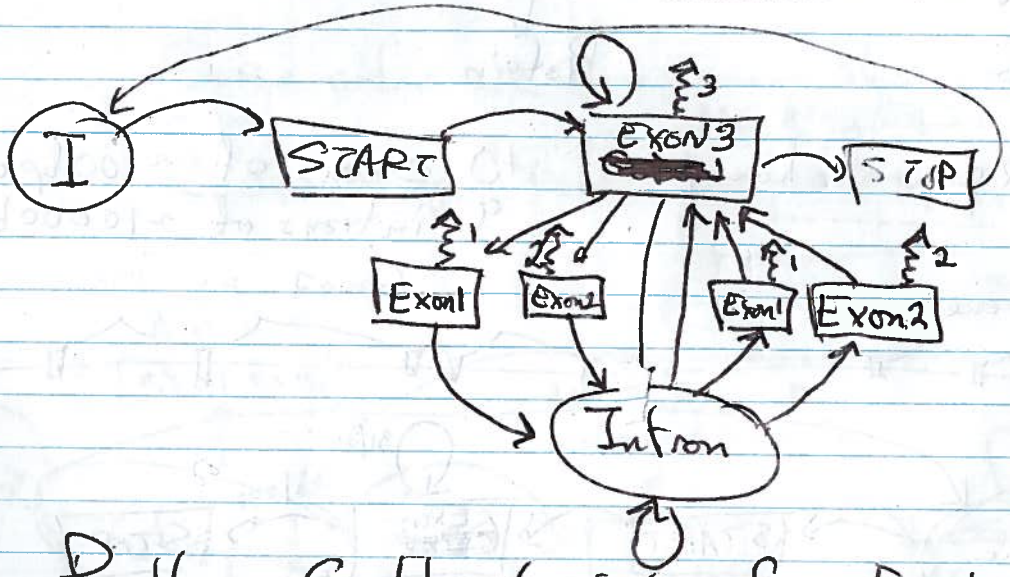
Gene structure in eukaryote



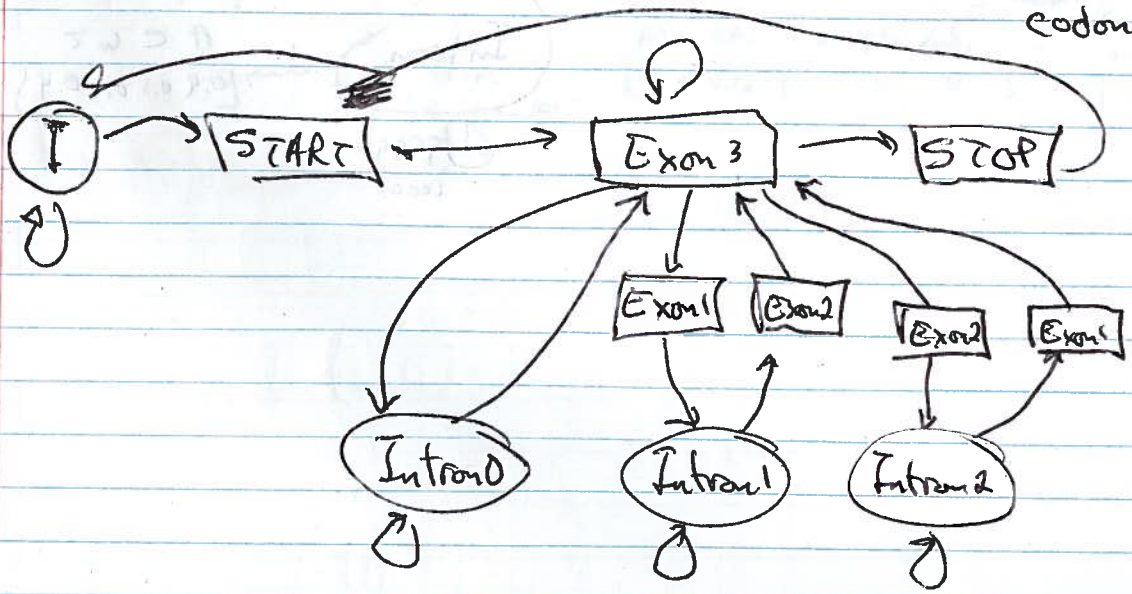
Typical gene in human : 10 exons of ~100 bp on aver
9 introns of ~10000 bp on av







Problem: Could enter intron from Exon1 and leave from Exon1 codon of 2 bp



Training HMMs

Goal: Given: • Long sequence of observations $X = x_1 \dots x_L$
• Structure of an HMM: Set of states S
Set of possible trans. T

Find: • Emission probabilities : E
• Transition prob. : T
• Initial state prob : I

such that E, T, I best represent the observed seq. X

Maximum Likelihood estimation ← $\Pr[X = x_1 \dots x_L \mid E, T, I]$ is maximized

$$\Pr[A] = \sum_b \Pr[A, B = b]$$

$$\Pr[X = x_1 \dots x_L \mid E, T, I] = \sum_{\text{path } P = p_1 \dots p_L} \Pr[X, P \mid E, T, I]$$

We know how to calculate

Can be calculated in $O(L \cdot n^2)$ using Forward algorithm

Simplified version:

Given: $X = x_1 \dots x_L$
$P = P_1 \dots P_L$ ← annotation of X

Find: E, T, I s.t. $Pr[X, P \mid E, T, I]$ is max

Let $N_t(s, s')$ = # of times transition happened btw s and s'
= # of positions i s.t. $P_i = s \wedge P_{i+1} = s'$

Then, choose $T(s, s') = \dfrac{N_t(s, s')}{\sum_{x \in S} N_t(s, x)}$ → total # of times in state s

Example: X: A C A ' I A C I A T G A C T G
P: G G G I I I I I I G G G G I I

$T(G, I) = \frac{2}{7}$ $\quad T(G, G) = \frac{5}{7}$

For emissions: $N_e(s, a)$ = # times a was emitted from s
= # positions i s.t. $P_i = s \wedge X_i = a$

Choose $E(s, a) = \dfrac{N_e(s, a)}{\sum_b N_e(s, b)}$

Example: $E(G, A) = \frac{4}{7}$

Back to problem where P is not given

① Viterbi training

Ⓐ Choose "reasonable" E, T, I (either from prior knowledge or randomly)

Repeat until convergence

Ⓑ Use Viterbi algo to find best path for $X = x_1 \ldots x_k$ assuming E, T, I ⇒ Produces path P

Ⓒ Re-estimate E, T, I from P, X as done previously

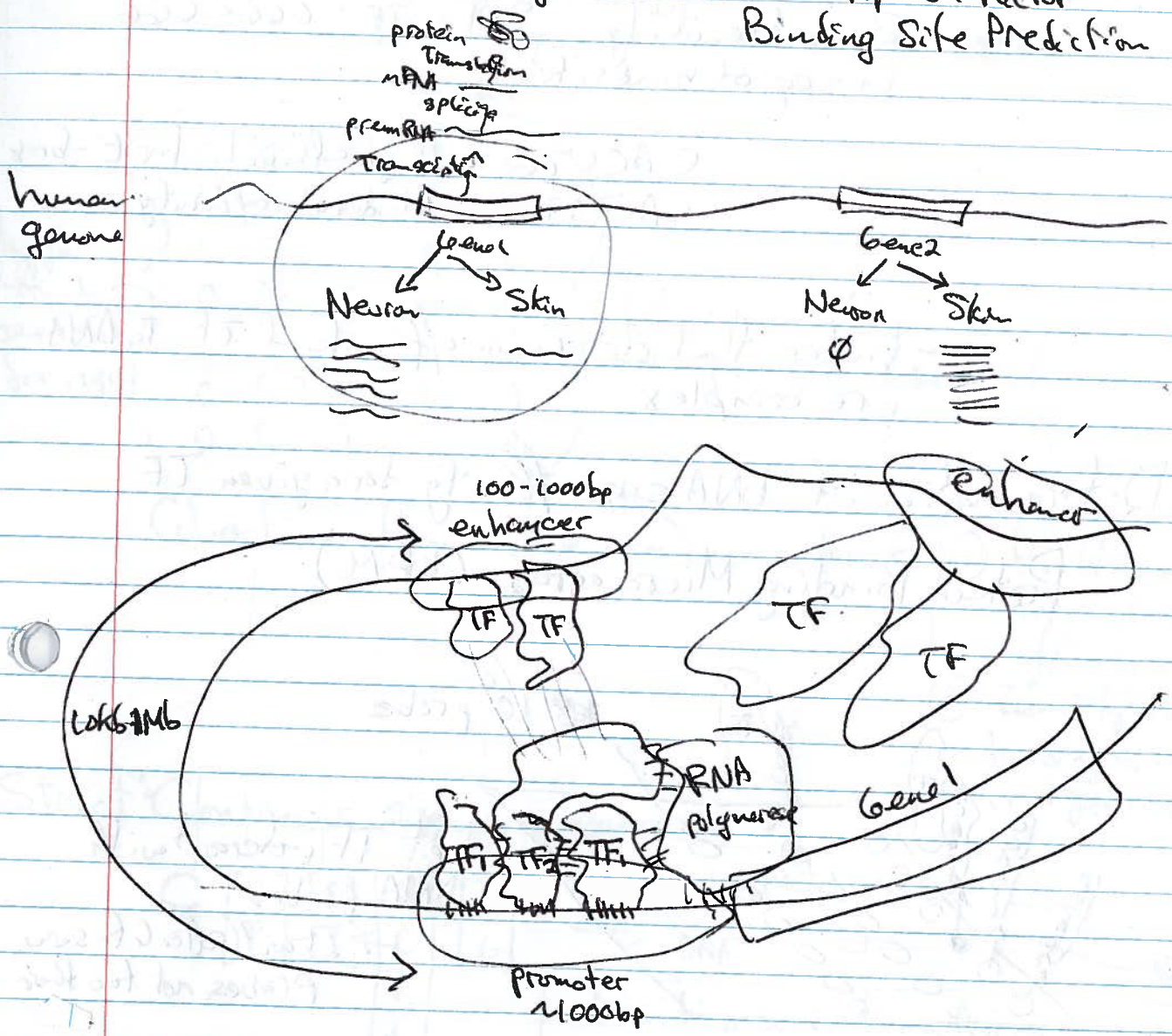
Problem: Algo. frequently gets stuck in local optima

② Baum-Welsch algorithm

Bases re-estimation of E, T, I on all paths (weighted by their probability)

⇒ Reduces the prob of getting stuck in local optima

Gene Regulation + Transcription Factor Binding Site Prediction



Transcription factors: - Proteins that - bind specific DNA seq.
- Alter expression of nearby gene(s)
- activation
- repression

- In humans: 2000 different TFs
↳ each binds to different DNA sequences
Transcription Factor Binding sites

TFBS: - 6-15 bp long
- Some flexibility in seq. of bind site

E-box TF bind: CACGTG
SP1 TF: GGGCGGG

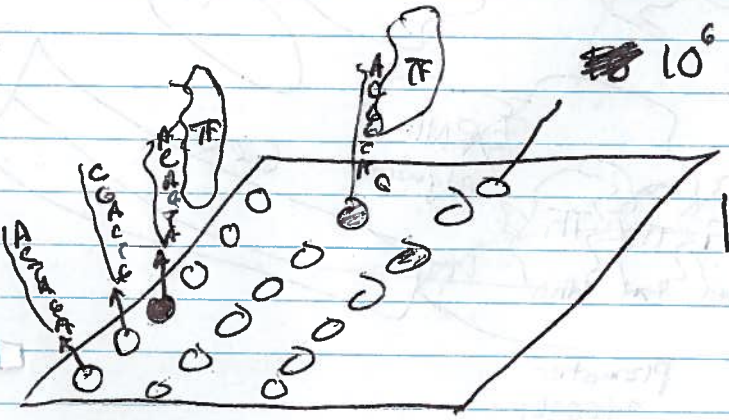
CACGTG: High affinity for E-box
CATGTG: Moderate affinity

⋮

- Rules that determine affinity of TF to DNA are complex

Determination of DNA seq. affinity for a given TF

Protein Binding Micro-array (PBM)



10^6 probes

Let TF interact with DNA probes.
→ It will attach to some probes, not to others

Take picture of PBM ⇒ infer affinity of TF to each probe

⇒

	intensity
ACAGTA	1000
ACGGTA	250
⋮	

Example: [redacted]

DNA sequences bound by TF [redacted] myc

not independent

C	C	T	A	A
C	T	A	C	G
C	T	T	A	G
C	T	T	G	G
C	C	T	T	A
C	C	T	C	A
C	C	T	T	A

high affinity sequences for myc

Representative sample of all possible sites TF

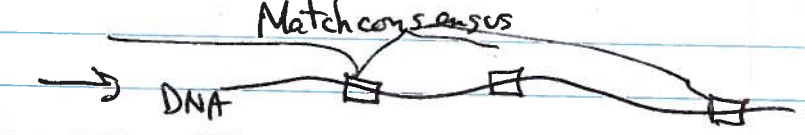
Question: How can we use in order to
① Build model for that TF's affinity
② Identify new binding sites in a genome

Strict Consensus sequence approach

$$C\begin{bmatrix}C\\T\end{bmatrix}\begin{bmatrix}A\\T\end{bmatrix}\begin{bmatrix}A\\C\\G\\T\end{bmatrix}\begin{bmatrix}A\\G\end{bmatrix}$$

or

$$C\begin{bmatrix}C\\T\end{bmatrix}[T]\begin{bmatrix}A\\C\\G\\T\end{bmatrix}\begin{bmatrix}A\\G\end{bmatrix}$$



Position weight matrices

	1	2	3	4	5
A	0	0	4/7	2/7	4/7
C	1	4/7	0	2/7	0
G	0	0	0	4/7	3/7
T	0	3/7	6/7	2/7	0

Candidate sequence to score: C T A G G

$1 \times 3/7 \times 4/7 \times 1/7 \times 3/7 = 9/7^4 = 0.00\ldots$

How to identify matches for a given PWM in a given sequence

S= A C A G T G A C T T

For each starting position i in S

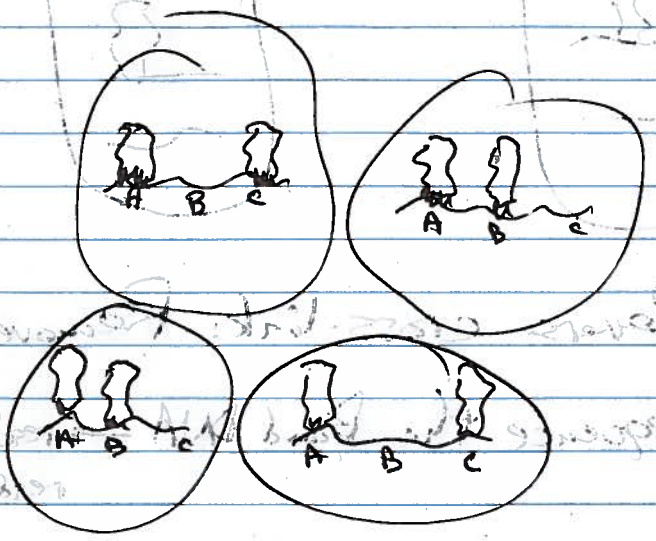
Calculate score of S[i, i+1, ... i+5-1] on PWM

If score > Threshold, then predict Binding
other no binding

ChIP-Seq + Motif discovery

Goal of ChIP-Seq: Identify regions of genome bound by a given TF in a given type of cells.

ChIP-Seq: Chromatin Immunoprecipitation followed by Seq.



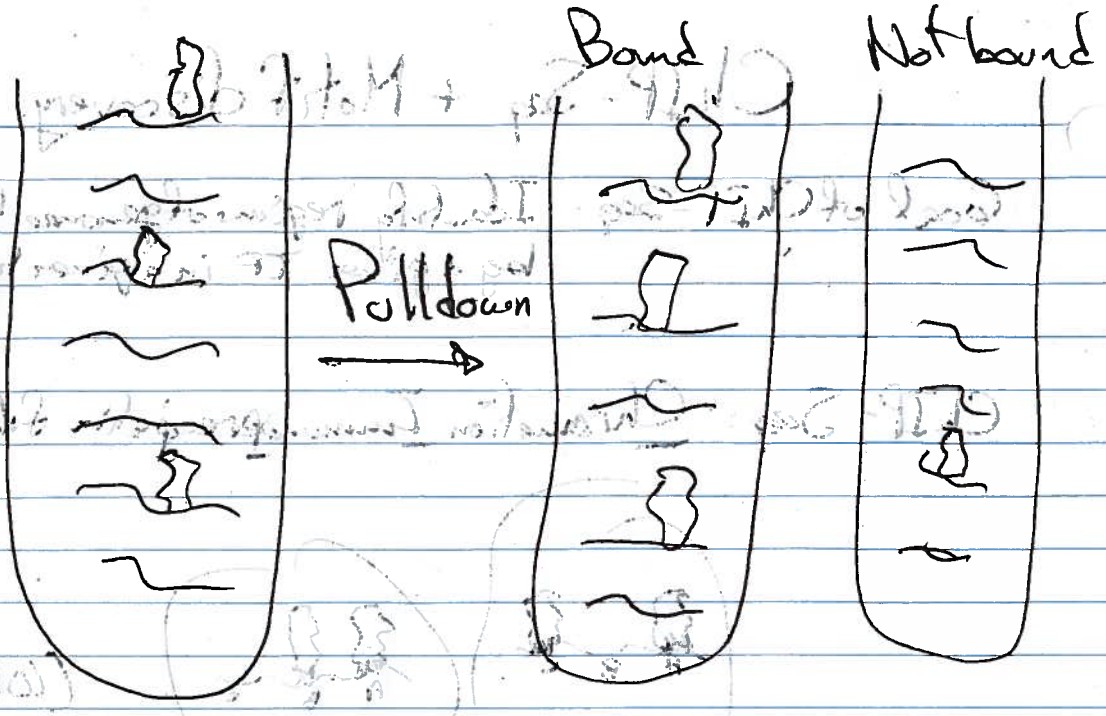
(10 Million cells)

(Typically: 100 - 100,000 sites occupied by TF)

① Cross-linking proteins to DNA: Strengthen bonds btw prot. and DNA

② Extract DNA with proteins attached to it

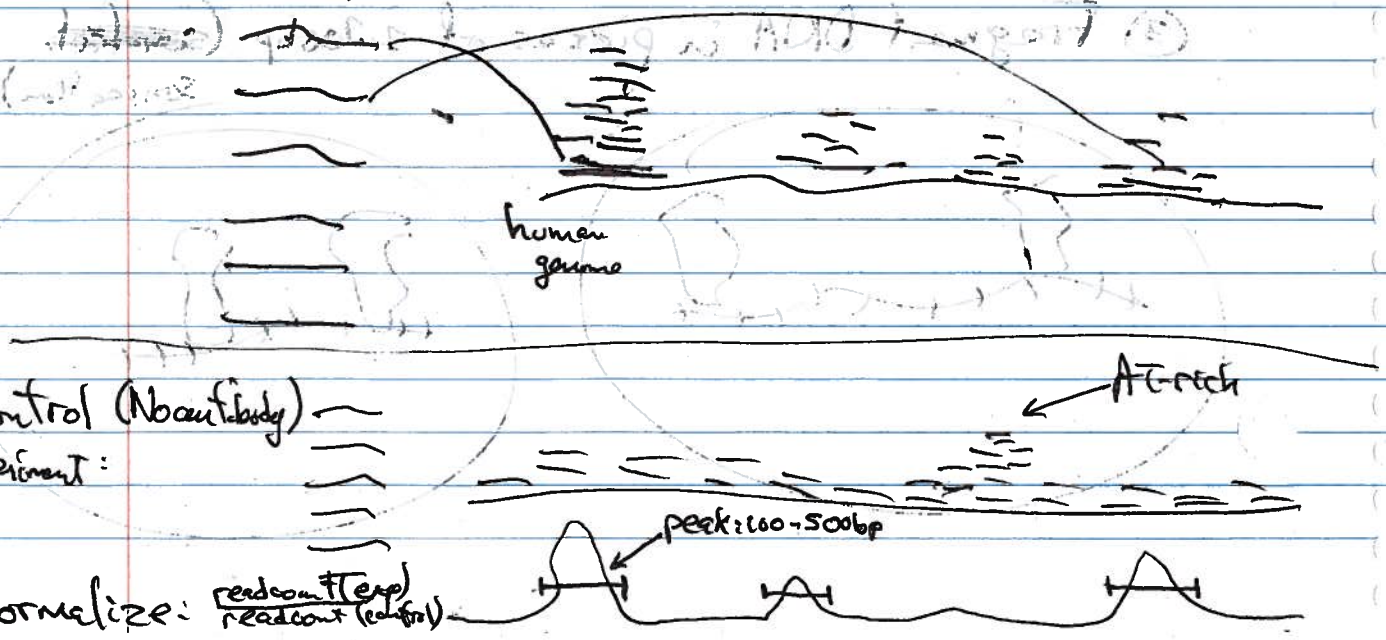
③ Fragment DNA in pieces of ~200bp (~~sonicate~~ sonication)



⑤: Reverse cross-link: Remove the TF from DNA

⑥: Sequence the Bound DNA → read1: ATTAGGTC (200bp)
read2: TGTGCTA...
⋮
read 10^7: TAATGAT...

⑦ ~~Map~~ Read Mapping
Local alignment btw reads and the human genome



100 - 100,000

ChIP-seq outcome: List of genomic regions (100-500bp that are believed to be bound by TF

Problem: Doesn't tell us which 6-15bp region is bound

⇒ Can't build consensus or PWM for TF

Motif Discovery Problem

Given: Set of sequences $S_1, S_2, \ldots S_n$ (of length 100-500) each believed to contain binding site for TF

Find: Consensus sequence for TF

OR

PWM for TF

Consensus sequence: Regular expression

$w = \{A\}\{C\}\{A\}\{T\}\{A\}$
$\quad\quad\{T\}\quad\{C\}$
$\quad\quad\{G\}$

How should we score a candidate cons. seq $w = w_1 \ldots w_k$

Criteria:

- w should occur in each of $S_1 \ldots S_n$
  - → too strict: Some S_i might be false-positives
  - → $\{A\}\{A\}\ldots\{A\}$ matches everywhere

Motif enrichment approach → consensus sequence

Let N_w = # of matches for w in $S_1, S_2, \ldots S_n$

E_w = Expected # of matches of w in a set of random sequences $R_1, R_2, \ldots R_n$, where R_i has same length S_i

→ each nucleotide is chosen independently with $P_A: 0.3$, $P_C: 0.2$, $P_G: 0.2$, $P_T: 0.3$

How to compute E_w? $w = w_1 w_2 \ldots w_k$ where $w_i \in \{A, C, G, T\}$

Ex: $w = \{A\}\{C\}\{A, T\}\{G\}\{T\}$

$\Pr[w$ has a match starting at position p in random seq$] = ?$

R: . . . . . p ... p+k-1

$$= \prod_{i=1}^{k} \Pr[w_i \text{ has match at position } p+i-1]$$

$$P_{match}(w) = \prod_{i=1}^{k} \left(\sum_{a \in w_i} P_a \right)$$

$$E_w = (\#\text{positions eligible for match}) \cdot P_{match}(w)$$

$$= \sum_{i=1}^{n} (\text{length of } S_i - k + 1) \cdot P_{match}(w)$$

Finally, from N_w, E_w

We want to find w where N_w and E_w are the most different, i.e. $N_w >> E_w$

Z-score approach

$$Z_w = \frac{N_w - E_w}{\sqrt{E_w}}$$

Complete algo.

For each possible consensus seq. w

Calculate N_w, E_w, Z_w

Report word w with highest z_w

DNA sequencing + Genome sequencing

Goal :

→ File : >seq1
ACTAGCTA] read1
>seq
TGACTAG..] read2
⋮

Illumina Sequencing : See video.

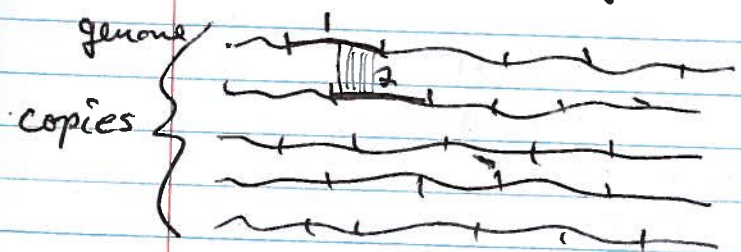
Limitation : - Length of a read is limited ($\lesssim$ 300bp) (Illumina)

Genome sequencing + Assembly

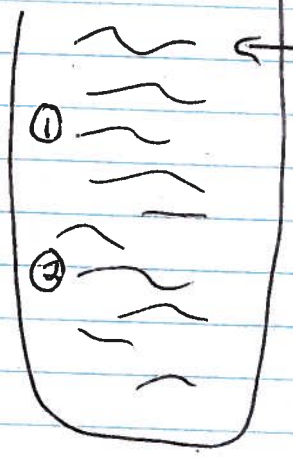
Goal: Get entire DNA seq. of a long genome

Problem: Seq. machines produce reads that short

Shotgun sequencing



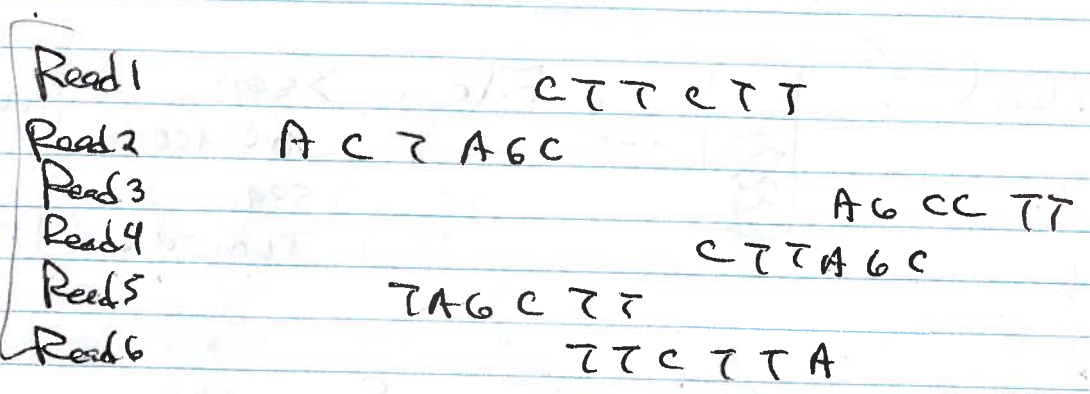
① Generate many copies of genome
② Cut seq. into small pieces
→ sonication, ~~restriction enzyme~~
randomly



③ Sequence DNA fragments
ACTAGC
TAGCC...
⋮

④ Assemble reads into genome

Truegenome : A C T A G C T T C T T A G C C T T
(unknown)



Problem: Shortest Superstring Problem

Given: Set of reads $R_1 \ldots R_n$, of length L = six

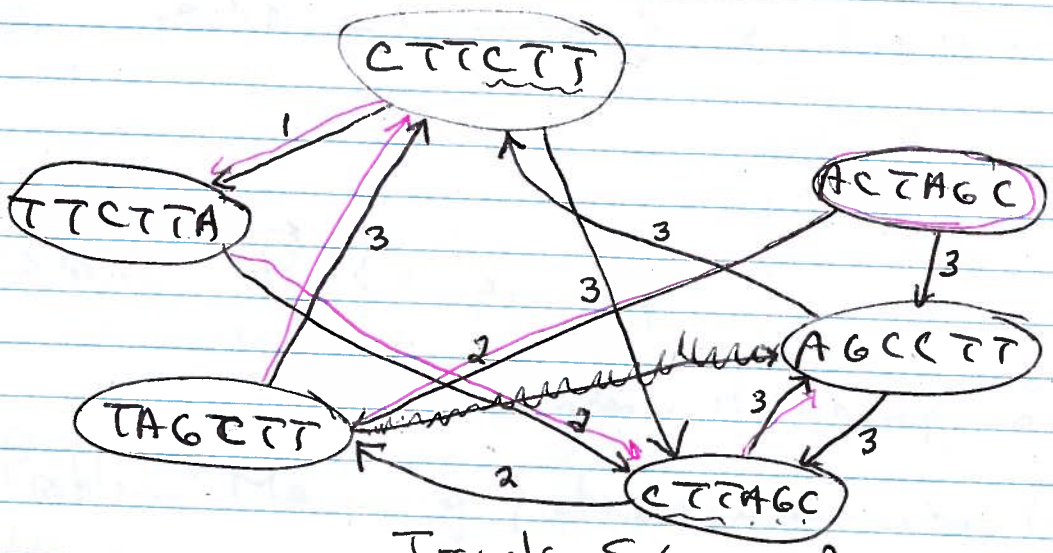
Find: Sequence G such that

1) Each R_i is a substring of G
2) G is a short as possible

Assumption: No sequencing error.

Overlap-Layout-Consensus Approach

Build Graph: V: set of reads
E: overlaps btw reads of at least $k = 3$ bases



Traveling Salesperson Problem → NP-Complete Problems

Goal: Find shortest Hamiltonian Path in G

Smallest total weight

Path that visits each vertex exactly once

ACTAGC
TAGCTT
CTTCTT
TTCTTA
CTTAGC
AGCCTT

weight: 2+3+1+2+3
= 11

Predicted Genome ACTAGCTTCTTAGCCTT

Gene Expression + Class comparison

Final Exam: Dec 14th (6pm) : open book, covers all topics

Goal: Capture and compare "state" of different cells

- cells with positive outcome upon treatment
- negative outcome

→ state $\vec{P} = (p_1, p_2 \ldots, p_{20000})$

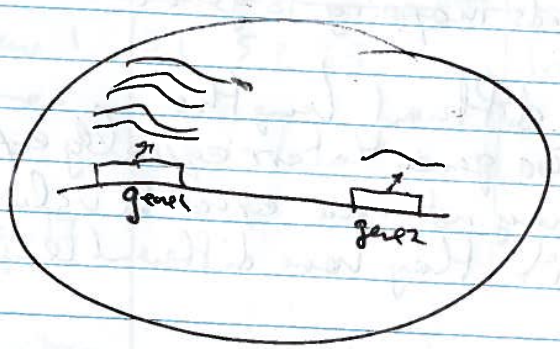
where p_i = abundance of protein p_i in cells.

Problem: Measuring protein abundance is hard (mass spectrometry)

Alternative: Measure mRNA abundance

$\vec{G} = (g_1, g_2 \ldots, g_{20000})$

where g_i = abundance of mRNA from gene i in cells



→ $g_1 = 5$
$g_2 = 1$

Note: $g_i \neq p_i$ because: mRNA degration
translation is regulated

RNA-sequencing (RNA-seq)

Goal: Measure gene expression levels $\vec{G} = (g_1 \cdots g_{20000})$

① Extract RNA from cells
② Reverse transcribed RNA to cDNA (↑ complementary)
③ Fragment cDNA in ~200bp pieces
④ Sequence each cDNA fragment

Read1 : ACTAG...
Read2 : ...
⋮
Read 10^7 : TACTA

Human Genes	R_i
gene1	1
gene2	3
gene3	1
⋮	
gene 20000	0

⑤ Map (↑ align) each read to its gene
Find the gene to which the read best aligns
⑥ Count R_i = #of reads mapping to gene i
Problem: Genes have different lengths
⇒ Two genes that are equally expressed may not have equal R values if they have different length
⑦ Normalize:

$$FPKM(g_i) = \frac{R_i}{(\text{length of } g_i \text{ (in kb)}) \cdot (\text{Total \# reads in Millions})}$$

FPKM: Fragment per kilobase per million reads

Class Comparison Problem

Given: Normalized gene expression data from two set of samples

A: (control) with $N_A = 20$ samples

$\vec{A_1} = (A_1(1), A_1(2) \ldots A_1(20000))$

$A_1(i)$ ↖ FPKM of gene i in sample 1

$\vec{A_2} = ($ $)$

$\vec{A_{N_A}} = ($ $)$

B = (treatment) with $N_B = 25$ samples

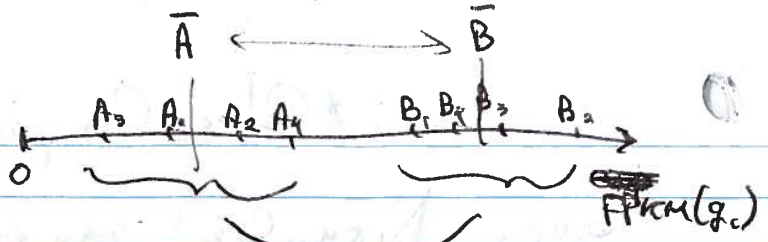
$\vec{B_1} = ($ $)$

$\vec{B_{N_B}} = ($ $)$

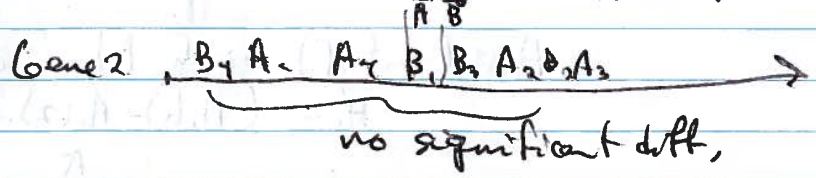
	20			25				t-stat	p-value
	A_1	A_2 ...	A_{N_A}	B_1			B_{N_B}		
Gene 1	5.1	5.7	3.7	1.7	1.3	1.9	2.1	7.1	0.001
Gene 2								1.3	0.25
⋮									⋮
Gene 17								best genes	0.0001 = 1/10^4
Gene 20000									0.35

Goal: Find genes that are "differentially expressed" b/w A, B

Example

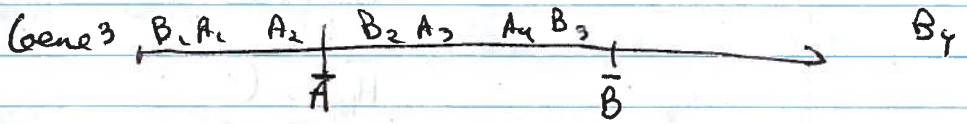


Significant diff



no significant diff.

$d(g_i)$ = diff. of means: $\bar{A}(g_i) - \bar{B}(g_i)$



Student t-test (performed separately for each gene)

H_0: Expression values from samples A and B come from the same normal distribution → Assumption

$\mu_A = \mu_B$

$\sigma_A = \sigma_B$

H_1: $\mu_A \neq \mu_B$

① Calculate $t(g_i) = \dfrac{\bar{A}(g_i) - \bar{B}(g_i)}{\sqrt{\dfrac{S_A^2}{N_A} + \dfrac{S_B^2}{N_B}}}$

S_A^2 = Variance in A

② Calculate [p-value for $t(g_i)$] = Prob that two random samples drawn from same distrib. would have t-statistics $t(g_i)$

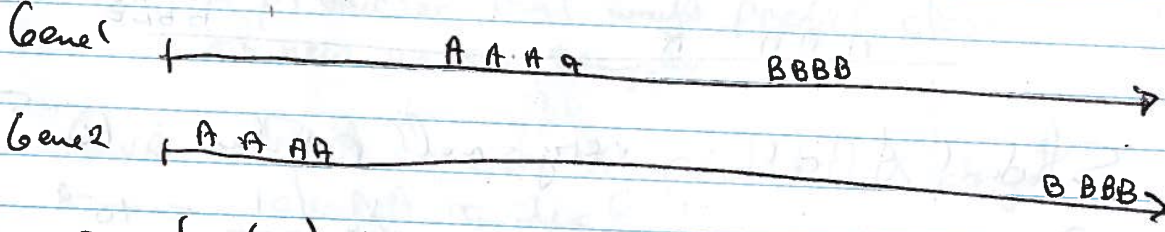
Under H_0: t follow a Student(m) ← degrees of freedom

$$m = \frac{\left(\frac{S_A^2}{N_A} + \frac{S_B^2}{N_B}\right)^2}{\frac{\left(\frac{S_A^2}{N_A}\right)^2}{N_A - 1} + \frac{\left(\frac{S_B^2}{N_B}\right)^2}{N_B - 1}}$$

If p-value(g_i) ≤ 0.05, then all g_i "diff. expressed"

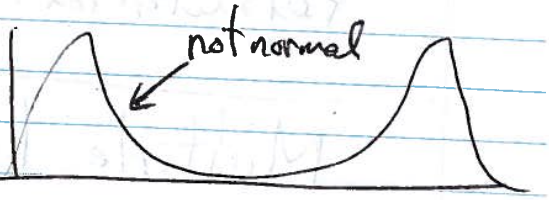
> 0.05, then g_i is not diff. exp.

Issue: Violation of assumptions: - Normality of data



p-value(g_2) < p-value(g_1)

Suppose distribution under H_0



Permutation test: Estimate p-value without assumption about underlying distrib. of data

Real $A_1\ A_2\ A_4\ A_5$

Shuffled $B_4\ A_1\ A_2\ B_3$

For each gene i

① Calculate $t(g_i)$

② Repeat $K=1000$ times

2.1 Randomly reshuffle class labels to obtain $\tilde{A}, \tilde{B}$

2.2 Calculate $\tilde{t}(g_i)$ from

2.3 If $\tilde{t}(g_i) \geq t(g_i)$ the

③ Report p-value$(g_i) = \frac{\text{success}}{K}$

A A A A B B B B

Student t-test: very small p-value: 10^{-}

Permutation: p-value: $\frac{1}{\binom{8}{4}} = 10^{-}$

Multiple hypothesis testing

We've done 20,000 tests

If all genes come from H_0, then best value we would expect to observe would be: $1/20{,}000$

Bonferroni correction: corrected Pvalue(g_i) = p-value(g_i) · N=

Class Prediction Problem

	A_1 A_2 ... A_{N_A}	B_1 ... B_{N_B}	X
Gene 1			
2	RNA-seq data		
3			
Gene 20000			

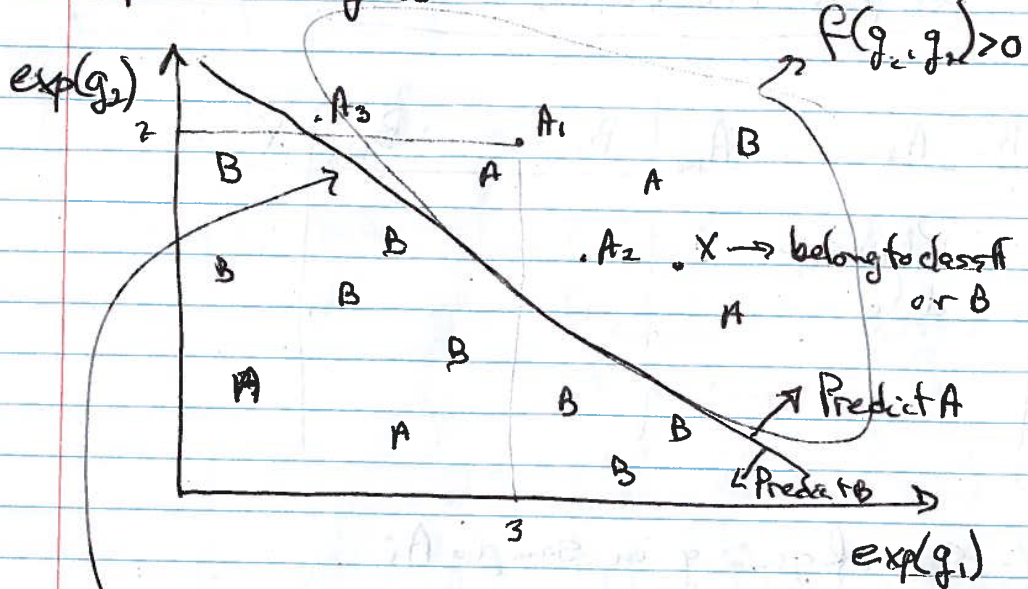
$A_i[g]$ = exp. of gene g in sample A_i

$B_j[g]$ = ...

Goal: From RNA-seq data from A, B, train a predictor that would predict class of new, unseen samples

Given: RNA-seq data X, predict if X belongs to class A or class B.

Assume # genes = 2



Linear classifier: linear function of $exp(g_1)$, $exp(g_2)$

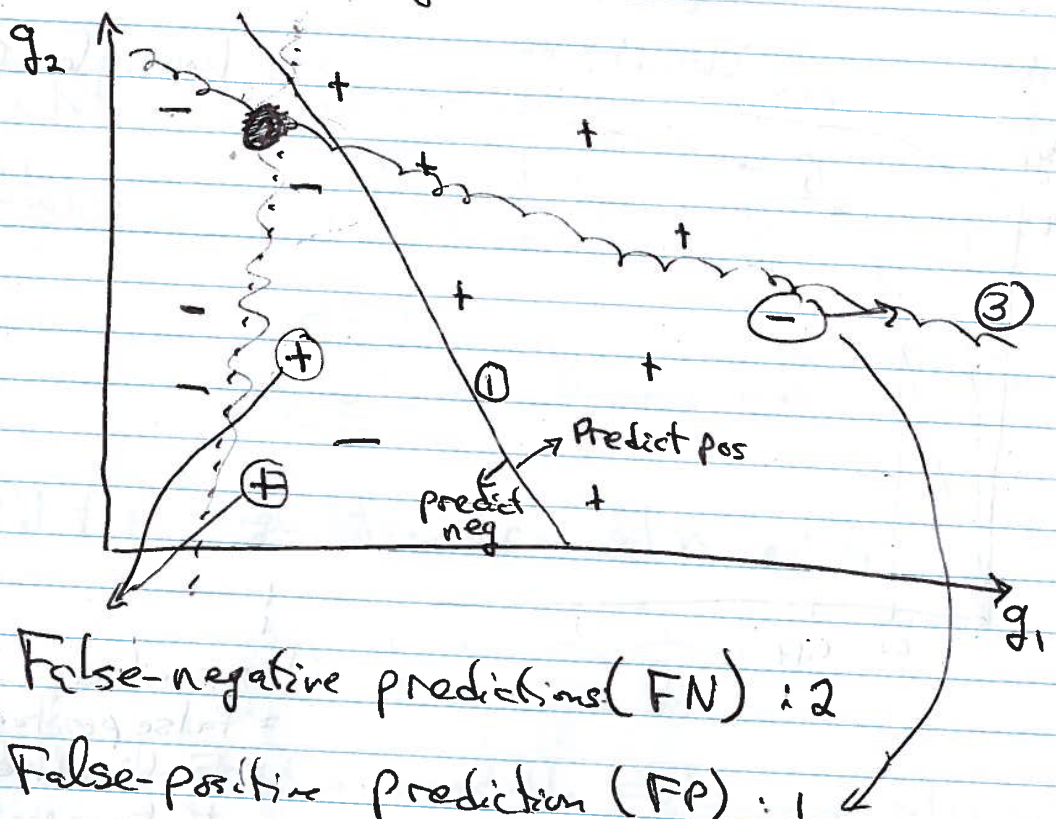
$g_2 = 10 - 2g_1$

$f(g_1, g_2) > 0 \rightarrow$ predict A

$< 0 \rightarrow$ predict B

$f(g_1, g_2) = 2g_1 + g_2 - 10$

Assessing a classifier's accuracy



False-negative predictions (FN) : 2

False-positive prediction (FP) : 1

True-positive (TP) : 7

True-negative (TN) : 5

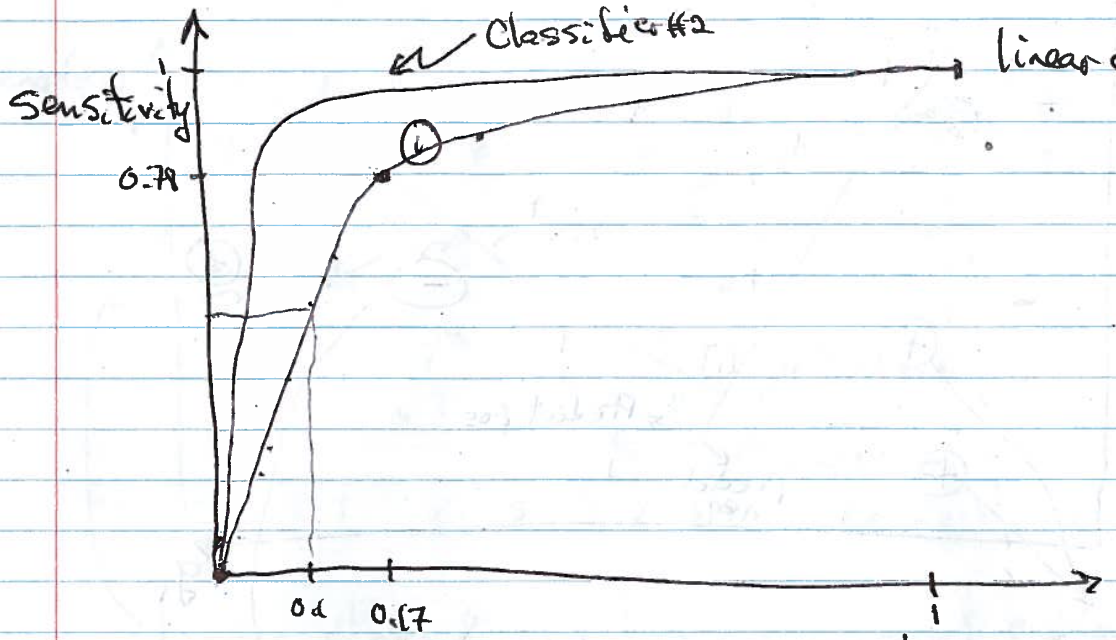
① Sensitivity : $\frac{TP}{TP+FN} = \frac{7}{7+2} = \frac{7}{9} \approx 78\%$

Specificity : $\frac{TN}{TN+FP} = \frac{5}{5+1} = \frac{5}{6} = 83\%$

② Sensitivity : 100%
Specificity : 50%

③ Sensitivity = 4/9 ≈ 55%
Specificity : 100%

Receiving-Operating Curve



1-specificity
= false positive rate
= Fraction neg. example that are predicted pos.

RNA secondary structure prediction

DNA

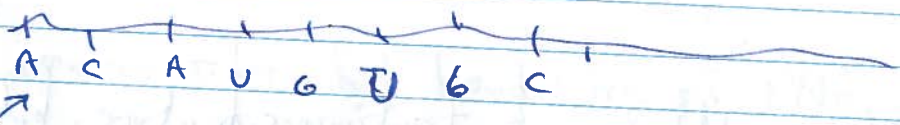
Stable:

A = T

C ≡ G

hydrogen bonds

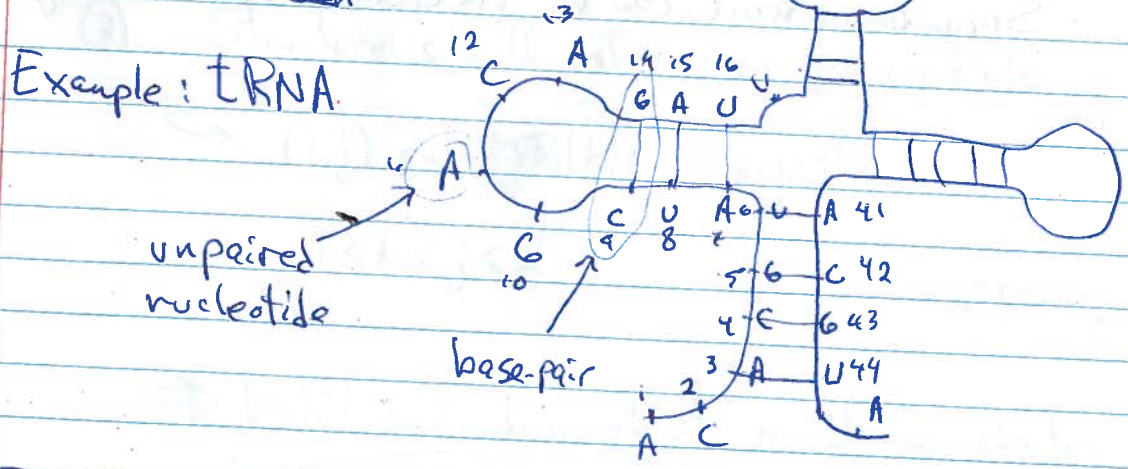
RNA : single-stranded chain of nucleotide



Unfolded structure is unstable ⇒ Sequence folds to maximize its stability

Folded ~~sequence~~:

Example: tRNA



Function of RNA molecules depends on their structure

↓

depends on sequence

Key ideas: A sequence typically folds in its most stable structure

Secondary structure of sequence $S = S_1 S_2 \ldots S_L$ is defined as list of pairs of positions that form basepai

Sec. struct. $\{(3,44), (4,43), (5,42), (6,41), (7,16), (8,15) \ldots\}$

Tertiary structure ≡ 3D structure: (x, y, z) coordinates of each atom in the sequence

Idea: · We want tertiary structure, but it is hard to predict

- Secondary struct. is sufficient to lets us predict function, and also useful for predicting tertiary struct
- Secondary struct. can be predicted computationally

Secondary Structure prediction problem

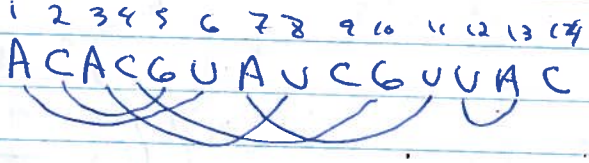
Version 1: Given: Sequence $S_1 \ldots S_L$

Find: Secondary structure for $S_1 \ldots S_L$ that is the most stable

↓

# of base pairings present in the sec. struct.

Example:



$\{(1,6), (2,5), (3,8) \ldots\}$

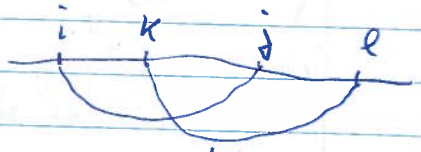
Problem: Ignores rules about bendability of RNA

Rules

① if $(i,j) \in$ Struct, then $|i-j| \geq 3$

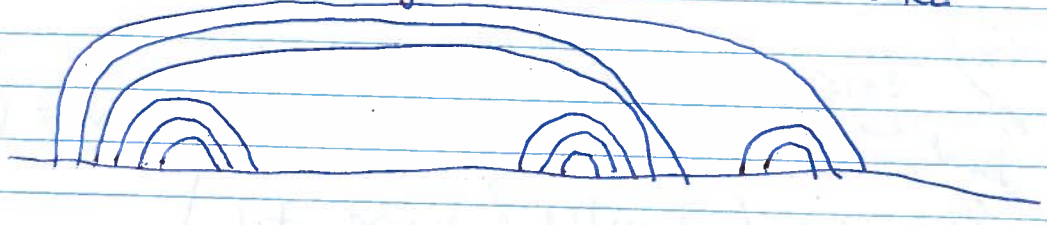
② Structure should not contain pseudoknots

↳ (i,j) and (k,ℓ) such that $i < k < j < \ell$



crossing base pairs

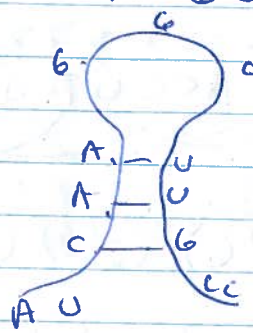
⇒ Valid secondary struct must be nested



Version 2: Given: RNA seq $S_1 \ldots S_L$

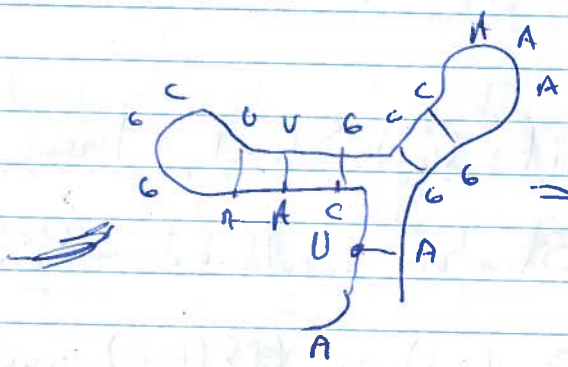
Find: Sec. struct. that maximizes total # of base pairs, subject to rules ① and ②

Example: A U C A A G G C U U G C C A A A G G A



Solution: (short) ⇒ 3 base pairs

longer:

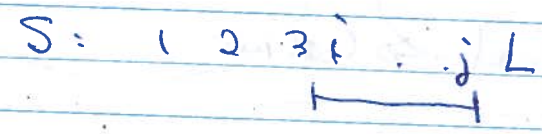


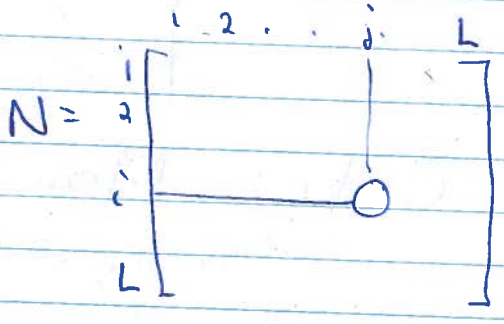
⇒ 6 base pairs

Nussinov Algorithm: Dynamic prog. algo.

Define $N(i,j)$ = Maximum # of base pairs that can be formed for $S_i \ldots S_j$

We want $N(1,L)$

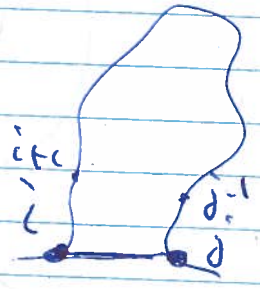




How to calculate $N(i,j)$?

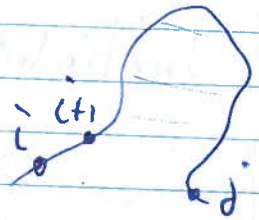
Case 1: S_i is paired with S_j

$$1 + N(i+1, j-1)$$



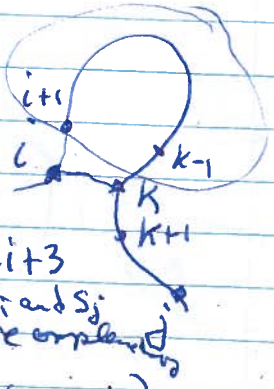
Case 2: S_i is not paired with anything

$$0 + N(i+1, j)$$



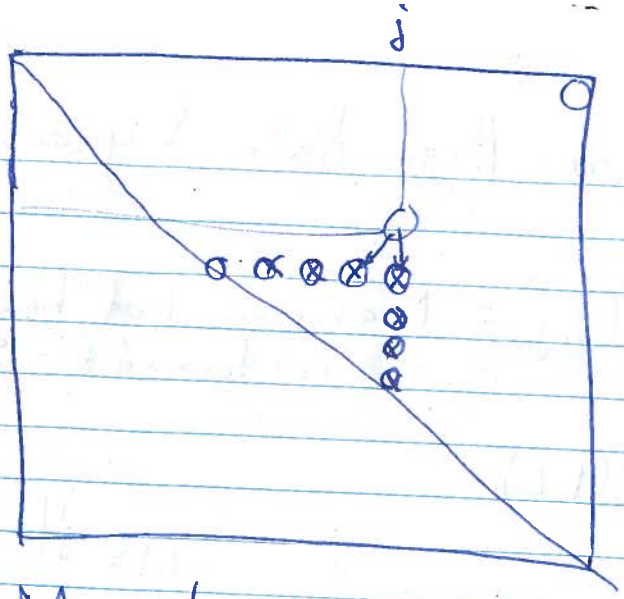
Case 3: S_i is paired with some nucleotide S_k, where $k < j$

$$1 + N(i+1, k-1) + N(k+1, j)$$

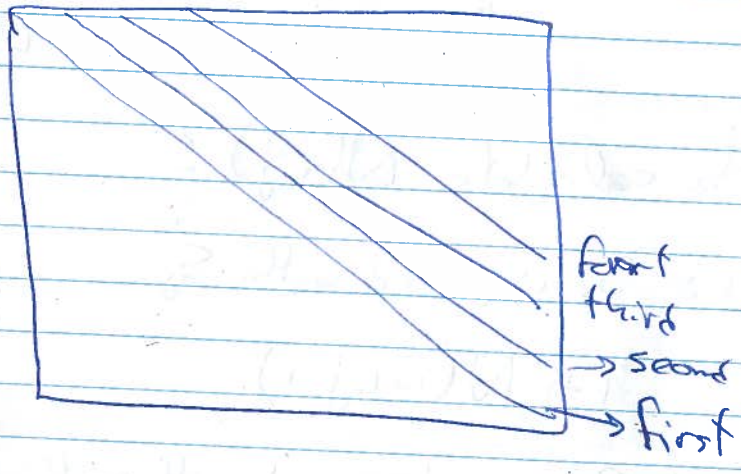


$$N(i,j) = \max \begin{cases} 1 + N(i+1, j-1) & \leftarrow \text{if } j \geq i+3 \text{ and } S_i \text{ and } S_j \text{ are complementary} \\ 0 + N(i+1, j) \\ 1 + \max_{i+3 \leq k < j, \text{ and } S_i \text{ is complementary to } S_k} \{ N(i+1, k-1) + N(k+1, j) \} \end{cases}$$

N:



Order: Main diagonal → Corner



Initialization: $N(i,i) = 0 \quad \forall 1 \le i \le L$

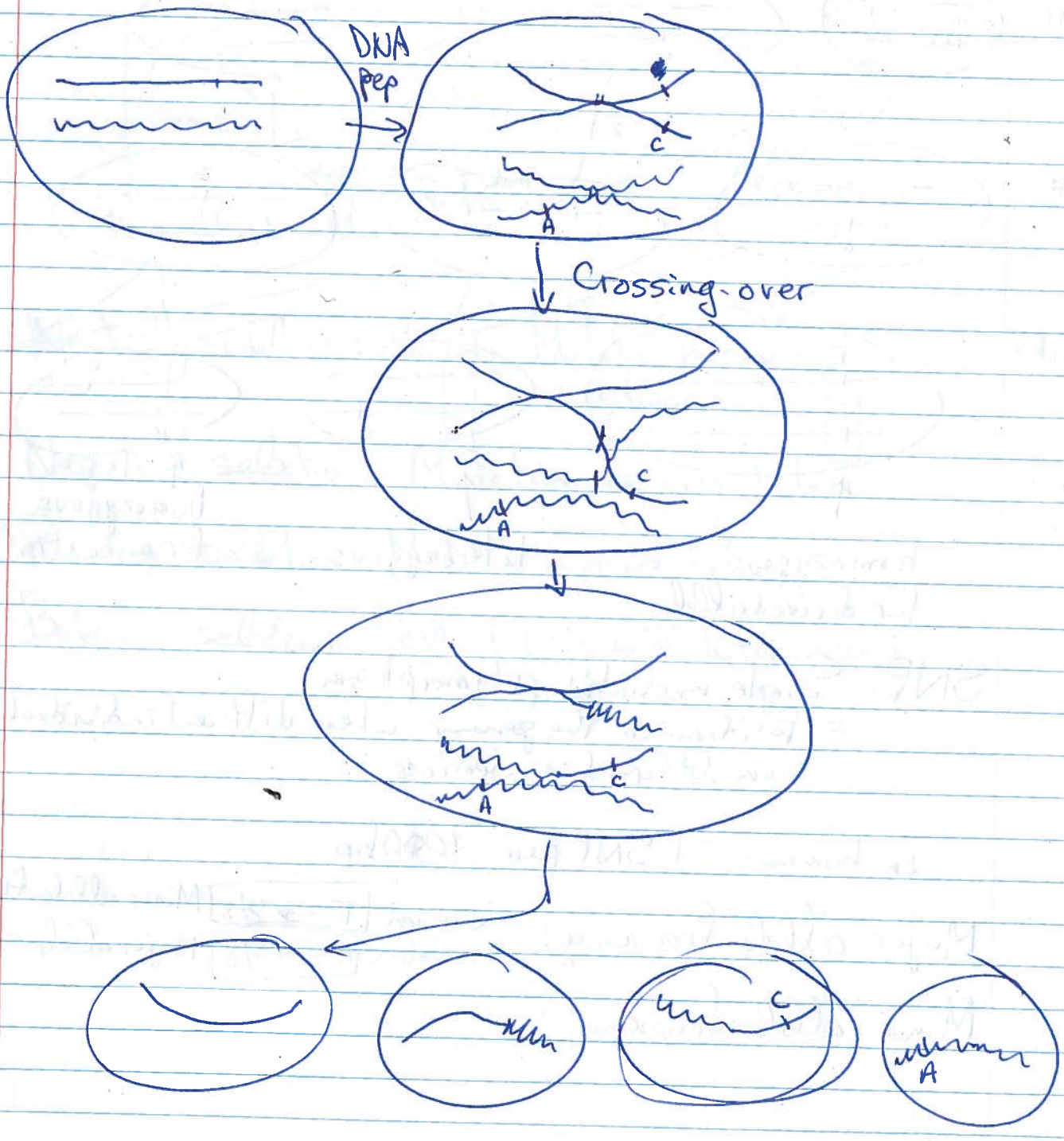
$N(i,i+1) = 0$

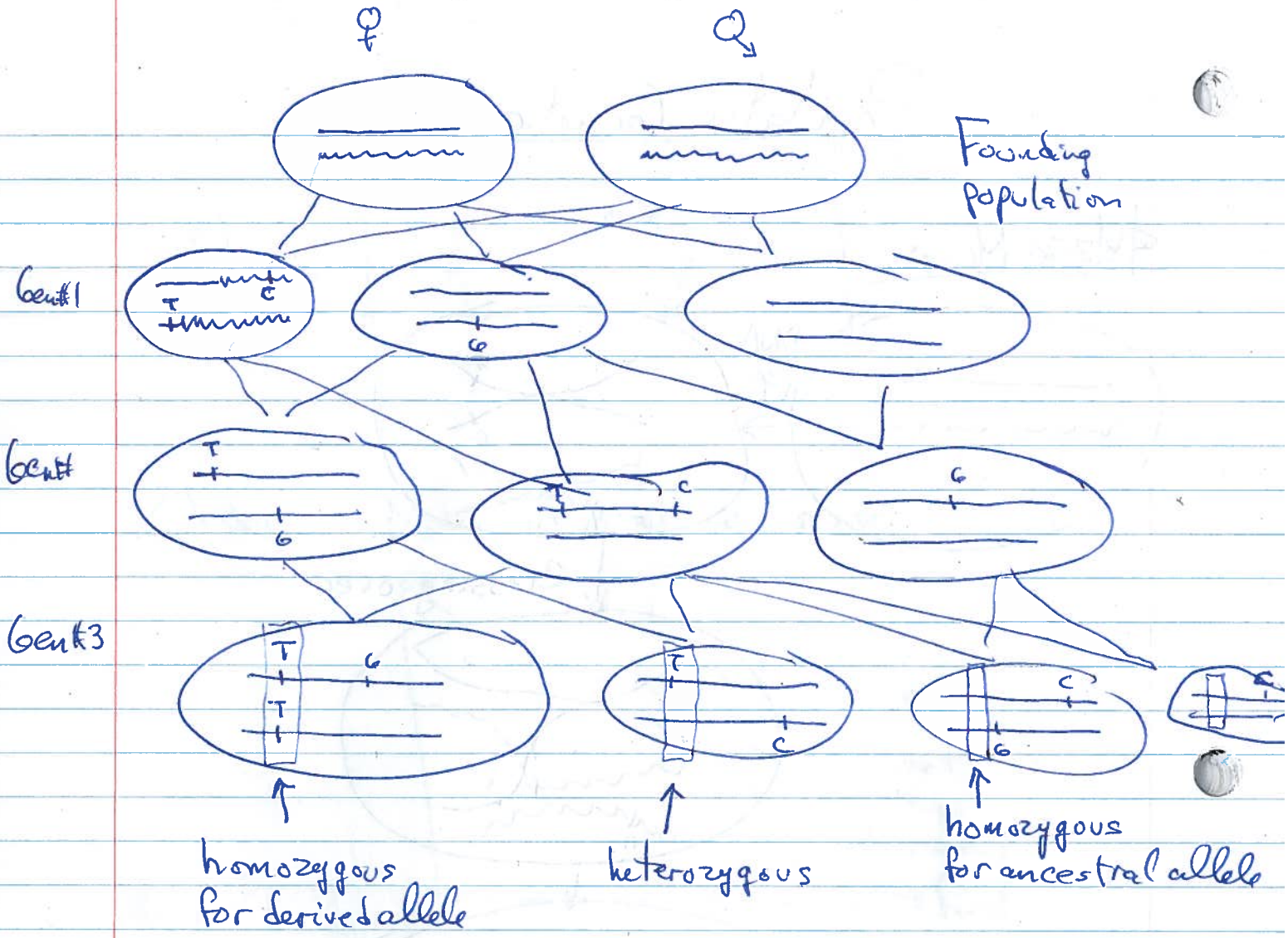
$N(i,i+2) = 0$

From there, use recurrence

Population Genetics

~~Meio~~ Meiosis





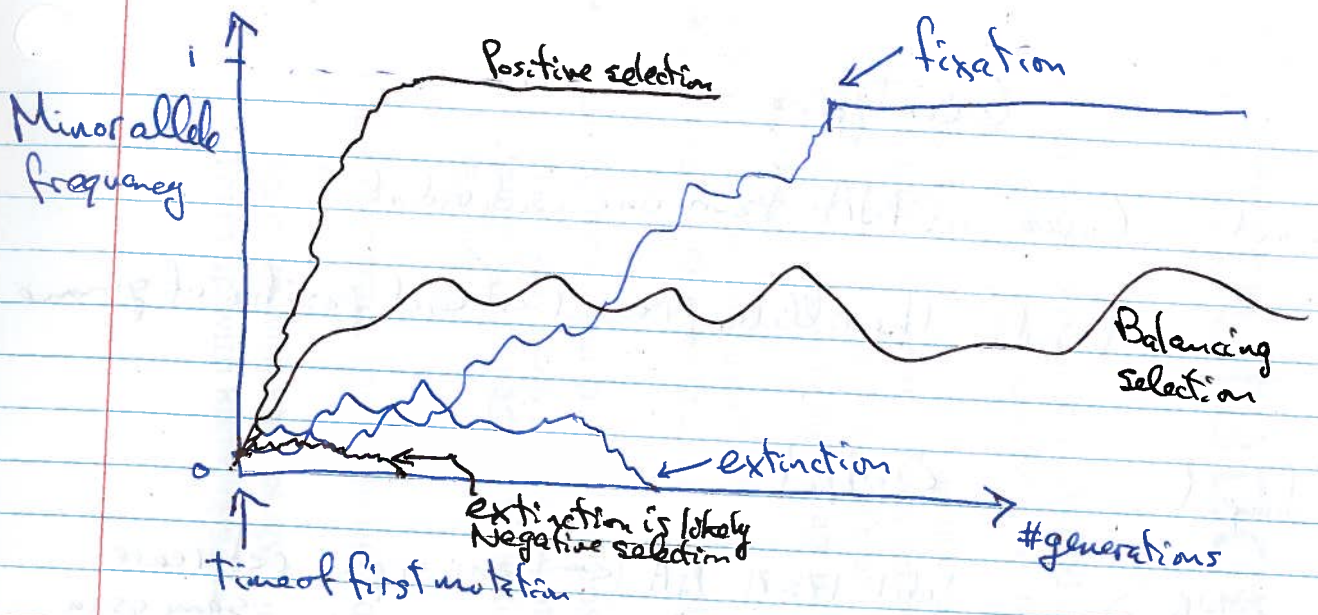
SNP: Single nucleotide polymorphism
= Position in the genome where different individuals have different nucleotides

In human: ~1 SNP per 1000bp

Major allele frequency:

Minor allele frequency:

derived: T → 3/8 Minor allele freq: 0.375
ancestral: A → 5/8 Major allele: 0.625



— Neutral model of selection: Mutation has no consequence on fitness

— Negative selection: Mutation reduces fitness

Positive selection: Mutation improves fitness

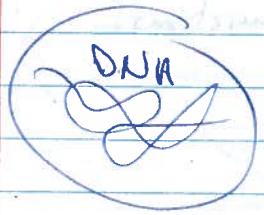
Balancing selection: Best fit is with heterozygous

Genotyping

Goal: Given: DNA from one individual

Find: The alleles present at each position of genome

Input → Output



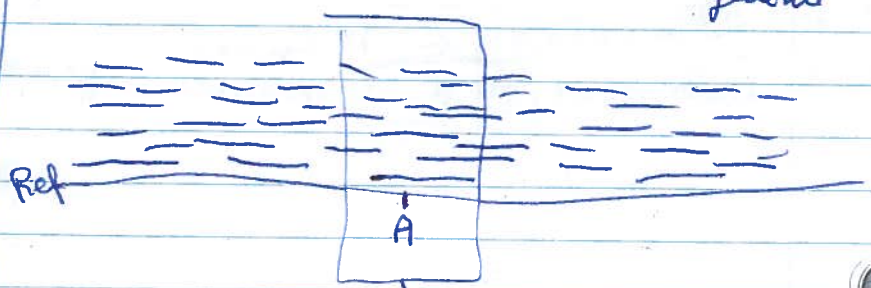
chr1	173271	AA	← homozygous reference = same as in reference human genome
chr2	173471	AC	← heterozygous
chr7	...	TT	← homozygous non-reference

- ~~Gen~~ Genotyping arrays
- Genotyping by sequencing

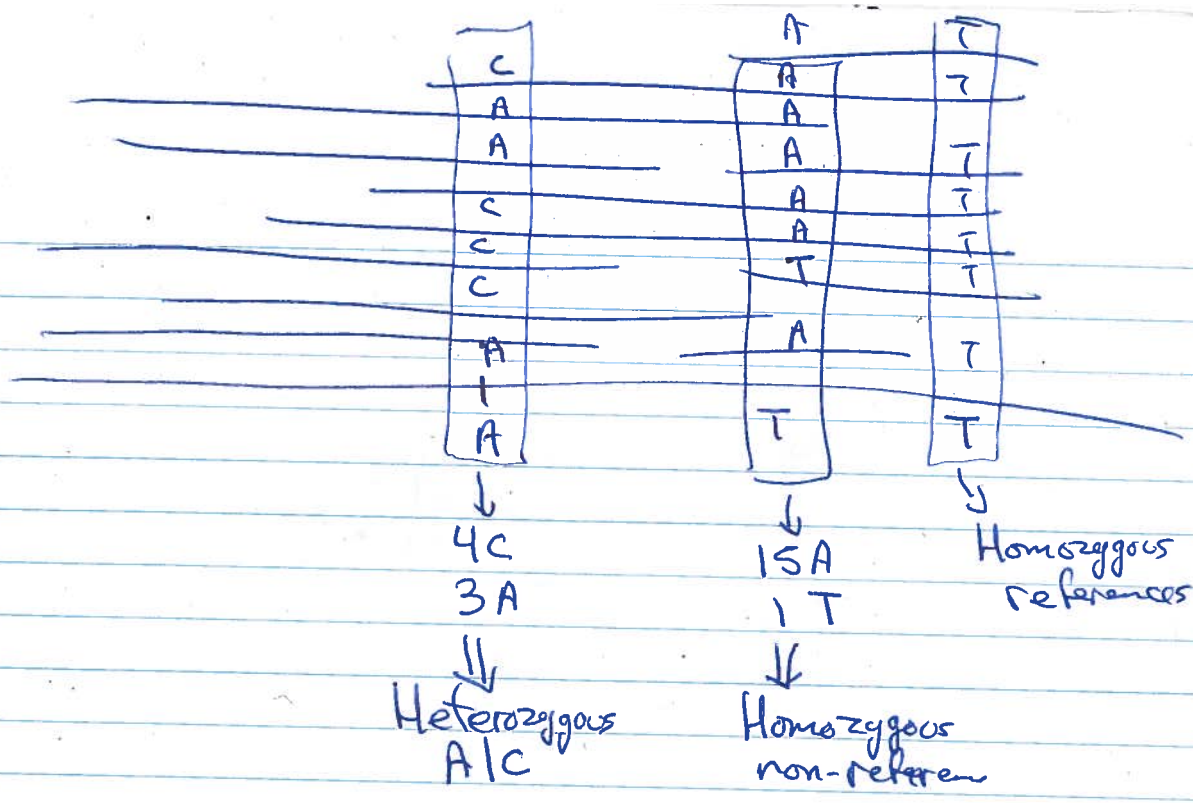
① Extract + fragment + sequence DNA

Fastq file | ACTAAAC..
ATTACACCA

② Align (map) each read to reference human genome



Next page


C
A
A
C
C
C
A
A
A
A
A
A
A
T
T
T
T
T
T
T
T
4C
3A
15A
1T
Homozygous
references
Heterozygous
A/C
Homozygous
non-referen