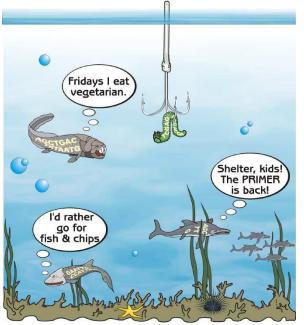
BCB 716 - Sequence Analysis





"The Art of Gene Fishing"

Final exam:

- Take home available to download from the website after midday on 12/1 (Friday)
- Read a paper
- ~5 Short answers that integrate concepts from class

Reference Free DNA Sequence Analysis

Reference Sequences Introduce bias

- Relying on a reference sequence makes many assumptions
 - All samples are assumed to be similarly organized
 - The "reference" allele is assumed to be common when sometimes it isn't
 - The reference is missing things
- How do you overcome this problem?
 - Consensus? Depends on population
 - Many references?





Comp 716 - Fall 2021

3

Rather than aligning reads to a reference, why not align them to each other. Not

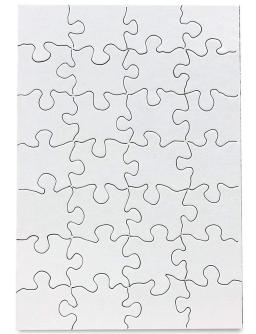
quite an assembly, but just enough to search for all and any sequence substrings.

Like assembling a jigsaw puzzle without an image of the assembled version. Or with no image at all!

Let's align every read with all other reads.

Rather than index a reference genome for searching, we build a BWT index of every read.

Novel idea





MSBWT



A BWT of a string collection instead of just a single string

- Earliest: Mantaci et al. (2005), used concatenation approach
- Bauer et al. (2011) proposed version we will discuss today

Analogy:

- Instead of searching for a substring within a single book, search every book of a library
 - Each book has its own text, suffix array, and end-of-text delimiter
 - Searching allows us to find how many times a substring appears and in which texts

Bioinformatics?

- Search all genomes? You could, but that's not the main application.
- Search multiple chromosomes of an organism? You should, but even that is not the killer app

Naive Construction

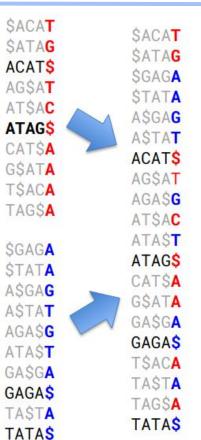


- Create all rotations for all strings in the collection
- Sort all rotations together (Suffix Array)
- Store the predecessor of each suffix
- Strings are "cyclic"
- The predecessor is always from the same string
- Impossible to "jump" from one string to another
- Strings can have different lengths

String1 <mark>ACCA\$</mark>	Sorted <mark>\$ACCA</mark>	MSBWT
CCA\$A	\$CAAA	Â
CA\$AC	A\$ACC	С
A\$ACC	A\$CAA	Α
\$ACCA	AA\$CA	A
	AAA\$C	С
String2	ACCA\$	\$
CAAA\$	CA\$AC	С
AAA\$C	CAAA\$	\$
AA\$CA	CCA\$A	Α
A\$CAA		
\$CAAA		
(Unsorted suffixes)	(Merged and sorted)	(multi-string BWT Note the 2 '\$'s)

Merging msBWTs

- BETTER YET! Rather than inserting new strings, build a BWT of the new strings and merge the new and old BWTs
- Suffixes of BTWs are already sorted
- BTWs are interleaved
- In the worse case (ties) the entire suffix must be considered, but generally the longest common prefix of suffixes is smaller
- Minimal overhead
- Well suited for divide an conquer approaches (like merge sort)
- Easy to merge multiple data sets!
- Compression improves!





K-mer Search & Read Extraction



Basic Use:

CC MRCAs: CC010_MRCA_2020 730,665,362 strings with 110,330,469,662 bases and index size of 14,976,208,773 bytes (1.09 bits per base) Target: CTTGTCCTTGGAAGGAAGATATATG

Found 27 times (15 forward, 12 reverse-complemented)

Inconsistent read size: reported = 151, found = 151

gatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaagactacaatccggaattggaagggctcaccagtggacccttatctgggagtcttttcggaatggaacgactacaatccggaattggaaggcttaccagtggaacgaatggaaggcttaccagtggaacgaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggctgaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggcttaccagtggaatggaaggctgaccagtggaatggaaggctgaccagtggaatggaaggctgaccagtggaatggaaggctgaccagtggaatggaaggctgacgatggaaggctgacgatggaaggctgacggaatggaaggctgacgatggaaggctgacgatggaaggctgacgatggaaggctgacgatggaaggctgacggaatggaaggctgacgatggaagggctgacgatggaagggctgacggaatggaagggctgacgagggaagggctgacggaatggaagggctgacggaatggaagggctgacggaagggctgacggaagggctgacggaatggaagggcgaagggctgacggaagg
gatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggtgtcaacttgaaattggaaggctaccagtggaaggctaccagtgagegcttgtgaaggttgtcaactggaaggctaccagtggaggtttgtgaaggaggctaccagtggaggttgtgaggtcttggaggttgtcaactggaaggctaggaggttggaggttgtggaggttgtggaggttgtggaggttgtggaggttgggaggttgggaggttgggaggttgggaggttgggaggttgggaggttgggaggttgggaggttgggaggtgggggg
tctgatCTTGTCCTTGGAMGGAMGATATATGgttattgtagtggctattcctggttgtcaacttggaaagactacaatccggaattggaaggctcaccagtggacccttatctggagggtcttatctggaatg
ctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaagactacaatccggaattggaaggctcaccagtggacccttatctggagggtcttatctggaggtctatcaattggaaggcttggaaggcttaccagtggaacgaatggaaggcttaccagtggaacgaatggaaggctgacccttatctggaggtcgaacgaa
.cagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcc <mark>g</mark> ggttgtcaacttggaaataatttggaaatgaactacaatc <mark>a</mark> ggaattggaaggctcaccagtgacccttatctggaggctgtcaacttgscaatatttggaatggaa
<pre>\$ccagctgtctgatCTTGfCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttgacaattattggaaggctaccagtggacctttatcctggaggcttgtcaacttgacaatttggaaggctaccagtggacctttat.</pre>
tgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttgacaattattggaaggctaccagtggacctttggaggcttggaggcttggaggttggaggctaccagtggaccttaggaggcttggaggttgg
<pre>\$ttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaataatttggaatggaactacaatccggaattggaaggctcaccagtggcccttatctggaggct</pre>
<pre>\$tcttgtcccagctgtctgatcTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaataatttggaatggaactacaatccggaattggaaggctcaccagtggcccttatctgggagg.</pre>
tcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaataatttggaaatgaactacaatccggaattggaaggctcaccagtggacccttatctgggagg.
tatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaataatttggaatgaactacaatccggaattggaaggctcaccagtggacctattctggagggctt3
\$ttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaagtgaactacaatccggaattggaaggctcaccagtgacccttatctggag.
SgtggtctgtgtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaatgaactacaatccggaattggaaggctc.
.gtggtctgtgtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaatgaactacaatccggaattggaaaggctc\$.
StagtggtctgtgtgatgaaccagttccttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaatgaactacaatccggaattggaaggc.
.ctagtggtctgtgtgatgaaccagttccttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaatgaactacaatccggaattggaagg\$
taatggaactagtggtctgtgtgatgaaccagttccttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaaatgaactacaatccggaa\$
.ttaatggaactagtggtctgtgtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttggaagtagaaccacaatcctgga\$
\$tttaatggaactagtggtctgtgtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttgtgacaatatttggaagtagaaccagtcgg.
.atcatgtgacaactttaatggaactagtggtctgtggtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggtaggaggtcgtgtggagas.
.\$ttttctctacatgtgacaactttaatgggactagtggtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgttattgtagtggctattcctggttgtcaacttgacaata.
\$cttatcctgctgcattttcctcatcatgtgacaactttaatgggacacagtggtctgtgtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAAGGAAGATATATGgttattgtagtggctatt.
\$gttcccccttctttcttctctgctgcattttcacattttcttctcatcatgtgacaactttaatgggacactgtggtgatgaaccagttcctatcctcttgtcccagctgtctgatCTTGTCCTTGGAAGGAAGATATATGgtt.
cgttccccccttccttctttatcctgctgcattttcacattttctctcatcatgtgacaactttaatggpactagtggtctgtgtgatgaaccagttcctatcctcttgtctgatCTTGTCCTTGGAAGGAAGATATATGgt\$
StctptctgatCTTGTCCTTGGAAGGAAGATATATGgttattptcaacttgatggaaggctcaccagtggaccttatctggaaggcttaggaggcttaggggttggaagtctattctggttggaaggctattctggttggaaggctattctggttggaaggctattctggttggaaggctattctggttggaaggctattctggatgga

Green: query k-mer. Red: forward reads. Blue: reverse-complement reads. Yellow: sequencing errors

- Search for all reads with a given k-mer
- Extract all reads with that k-mer and its reverse-complement
- Build a consensus

Moving over a little bit



CC MRCAs: CC010_MRCA_2020 730,665,362 strings with 110,330,469,662 bases and index size of 14,976,208,773 bytes (1.09 bits per base) Target: GGTTATTGTAGTGGCTATTCCTGGTTGTCAACTT

Found 52 times (27 forward, 25 reverse-complemented)

Inconsistent read size: reported = 151, found = 151

NUMERAL PROVINCE AND	
Satcher	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattacaatccggaattggaaggctcaccagtgacccttatctggaggtccttggatctttggatcttggatcttggatcttggagtcttggaggtcttggaggtcaccagtgacccttatctggaggtcttggaggtcttggaggtcttggaggtcttggaggtcgacaggggtggaggtggaggtggaggtggagggtggagggtggaggggtggaggggtggaggggtggaggggtggagggggg
	a GGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattacaatccggaattggaaggctcacagtgaccttatctggaggcttggaagtcttatctggattttggatttgaagatctta
	aradotTATTGTAGTGGCTATTCCTGGTGTGCACTTgacaatatttggaatgaattaaatcggaattggaagctcaacggaacttgatgccttattgggacttggatcttatttggatctggtttgaaga
	a doith Thigh a for the the second
caagatagcaaacattatctccg	
	ataGGTIATIGIAGIGGCTATICCIGGTIGICAACTIgacaatatttggaatgaatgaatggaatggaaggctcaccagtgacccttatctggaggctcggagatccttatctggagats
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaa <mark>t</mark> tacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatccttatctgg
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaa <mark>t</mark> tacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatccttatctg\$
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaa <mark>t</mark> tacaatccggaattggaaggctcaccagtgacccttatctggaggctt
\$tggggaccttgtacacaagatagcaaaacattatctccg	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaatgaatgaaatggaatggaaggctcaccagtgacccttatctggaggcttgggagtc
	<mark>ata</mark> GGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaa <mark>tagatcggaa</mark> ga <mark>gcgtcgtgtagggaa</mark> ag <mark>agtgttaacgtc</mark> gg <mark>tgtagatctcg</mark> \$
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaatgaatgaaatggaatggaaggctcaccagtgacccttatctggaggcttgg
	<pre>ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaatgaatgaaatggaatggaaggctcaccagtgacccttatctggaggct\$</pre>
	<pre>ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattggaattggaaggctcaccagtgacccttatctg\$</pre>
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattggaattggaaggctcaccagtgacccttatct\$
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattggaattggaatggaaggctcacc
agaaaattcaatagttctaatagttctaactctgacaactatggggggccttgtacacaagatagcaaacattatctccg	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattgaatcggaattggaaggctca\$
\$ctcagagaaaattcaatagttctactctgacaactatgggggaccttgtacacaagagatagcaaacattatctccg	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattacaatccggaattggaag
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaatgaatgaatgaaatggaa
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaattacaatcc\$
\$catgacccaaggaaaactagacgtgcatcccaacactctcagagaaaattcaatagttctactctgacaactatggggaccttgtacacaagatagcaaacattatctccg	
	ataGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaat
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggagcttggagatccttatctggatcttggtttgaagatcttgagcagatcs
	tatGoTTATTGTAFGGCTATTCTGGTGTGACTTgacaatatttggaatgaactacaatccggaattggaatgaaccttatctggaagctatctggagatcttatctggagatttgagaattgaactatagatggaattgaagctatatctggaagtgaaccttatctggagattggaatggaattgagattgagattgaagctatatgggagtgaacttatctggagtgaccttatctggagtgaccttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtgggacttatctggagtggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtgggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggagtggacttatctggaggattggagtggacttatctggagtggacttatctggagtggacttatctggaggattggaggtggacttatctggaggattggaggtggacttatctggaggtggacttatctggaggattggaggtggacttatctggaggtggacttatctggaggattggaggtggacttatctggaggtggacttatctggaggtggacttatctggaggtggacttatctggaggtggacttatctggaggtggacttatctggaggtggactggattggaggtggactggattggaggtggactggattggaggtggactggattggaggtggactggattggaggtggggtggggtggggtggggtggggtggggtggggtgggg
	tatGotTATTGTAGGGATTCCGGTGTGCACTggatatttggaagattggatggattggaaggttggaggttggacggttggaggttggagttggattgg
gara Tagaagaa	
	tateGTIATIGTAGTGGCTATICTGGTGTGACTTgacatattggaatgaattatattggaatggaatggaaggttatagtgaaggttatattggaggttggaggtttattg
	tatos na forma de la ficto de la forma da la titiga a la atriciga esta a actega atga agos catega se contra tega esta de la garage contra de la garacter
	tatoon na no na
	tatGGTTATTGTAGGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggctctggaggtcttatctggatcttggtttg
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatcttatctggatcttg\$
	tat <mark>GGTTATTGTAGTGGCTATTCCTGGTTGTCAACTT</mark> gacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatccttatctggatcttg \$
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggctctggaggatccttatctggatc\$
ctgtcttgatcttgtccttggaaggaagat	
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatccttat
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatcc\$
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatc
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggaga
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggaga\$
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggctt\$
\$ttcctatgtcccagctgtctgatcttgtccttggaaggaa	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggag
\$gtggtctgtgatgaaccagttcctatgtccctagtcccagctgtcgatctgatcttgtcccag	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctc
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctc\$
\$tagtggtctgtggtgatgaaccagttccttgtccctagtcccagtccttgtcccag	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggc
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaagg\$
taatggaactagtggtctgtgtgaaccagttcctatccttgtcccagctgtctgt	
ttaatggaactagtggtctgtgtgatgaaccagttcctatcctcttgtcccagctgtctgatcttgtccttggaaggaa	
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccgg
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaata
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatccggaattggaaggctcaccagtgacccttatctggaggcttggagatccttatctg
	tatGGTTATTGTAGTGGCTATTCCTGGTTGTCAACTTgacaatatttggaatgaactacaatcggaattggaaggctcacagtgaccttatctggaggcttggaagtccttatctggatcttggtttgaagatcttggttgaag
	the dot in the top the test of the second seco
. CATSACTITTGTCCATTATATGGAATTITAATGGAACTATGGGTCTGTGTGGATGAACCAGTTCCTCTCTCT	

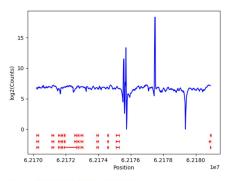
Comp 716 - Fall 2021

Moving over 24 bases shows a different story

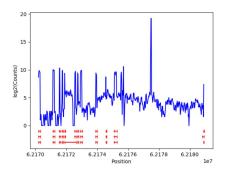
Reference-based Searches

- Rather than align reads to a reference search for reference k-mers in the read dataset.
- Given a reference genome and region
- Split reference into overlapping k-mers
- Count the abundance of each k-mer and plot
- Fast O(k) time per k-mer
- With preprocessing (sorting k-mers in suffix order) it can be even faster approaching O(1) for many k-mers
- Similar to an alignment pileup

Gene: Alad (ENSMUSG0000028393) Cc010_INCA_2020: 730,665,382 strings with 110,330,469,662 bases and index size of 14,976,208,773 bytes (1.09 bits per base) Chromosome 4: 62,170,203 - 62,180,652



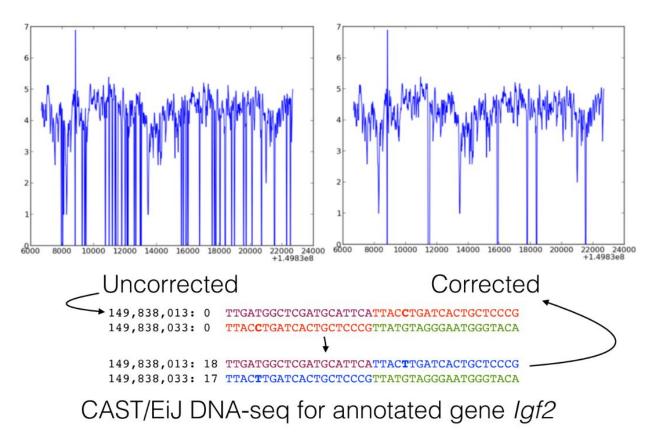






Iterative Reference Correction









Reference-free RNAseq Analysis Approaches

