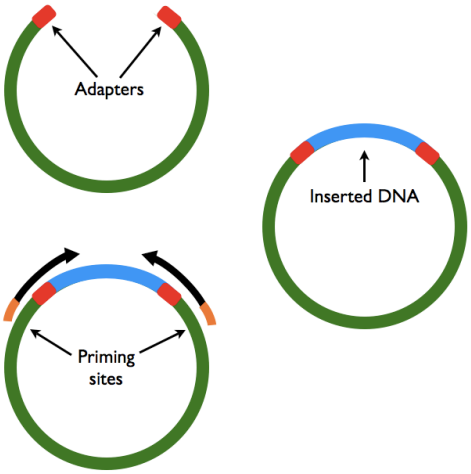
Figaro: a novel vector trimmer

james robert white
whitej@umd.edu
Center for Bioinformatics and Computational Biology
University of Maryland - College Park

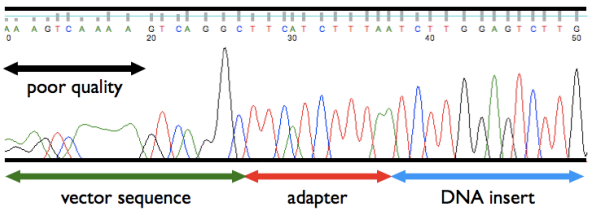
Background

- high-throughput shotgun sequencing.
- cloning pieces of DNA from some sample into a vector (plasmid).
- DNA is read by amplifying the fragment using priming sites in the vector.



Background

- target DNA is read using automated sequencing machines.
- poor quality sequence at the beginning of read.
- parts of vector and adapter sequences are read before the true DNA sequence.
- vector and poor quality must be removed prior to analyses.



Background

- current software for vector removal: Lucy (Chou and Holmes), Crossmatch (Green), VecScreen (NCBI).
- all require prior knowledge of the vector sequence, splice site locations, and any adapter sequences used.
- NCBI Trace Archive frequently has missing or incorrect vector clipping coordinates.

Figaro

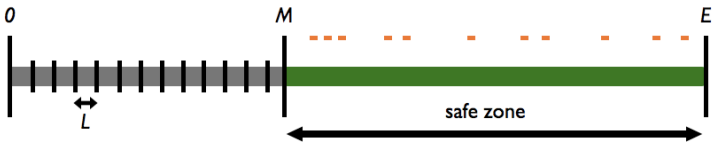
- vector trimmer that requires no prior knowledge of the vector sequence.
- statistically determines kmers most likely part of vector sequence.
- open source software available through the AMOS project (sourceforge).

Algorithms

- Figaro has two major phases:
  1. detection of *vectormers* - kmers likely to represent vector DNA.
  2. estimation of vector clip points.

Detection of vectormers

Step 1: Count kmers.



ACGTGGTA 9 8 13 12 ..... 6 5 9 384*
CCGACGTA 30 25 27 ..... 14 12 1,714*

kmer: K_i, if s_i is the number of occurrences of K_i in the safe zone across all reads, then we define its arrival rate a_i to be:

$$a_i = s_i / (E-M)$$

Detection of vectormers

- Given the arrival rate of K_i, a_i, we model occurrences of K_i as a Poisson process.
- We look at each K_i frequency count in our bins and calculate the probability of seeing this count in a window of length L, given a_i.

ACGTGGTA 9 8 13 12 ..... 6 5 9 384* => a = 384/500 = .768

f1 f2 f3 f4 fn

$$P(X \geq f_j) = 1 - P(X < f_j) = 1 - \sum_{y=0}^{f_j - 1} \frac{e^{-\lambda} \lambda^y}{y!}$$

if P(X >= f_j) < 0.001, we declare ACGTGGTA to be a vectormer.

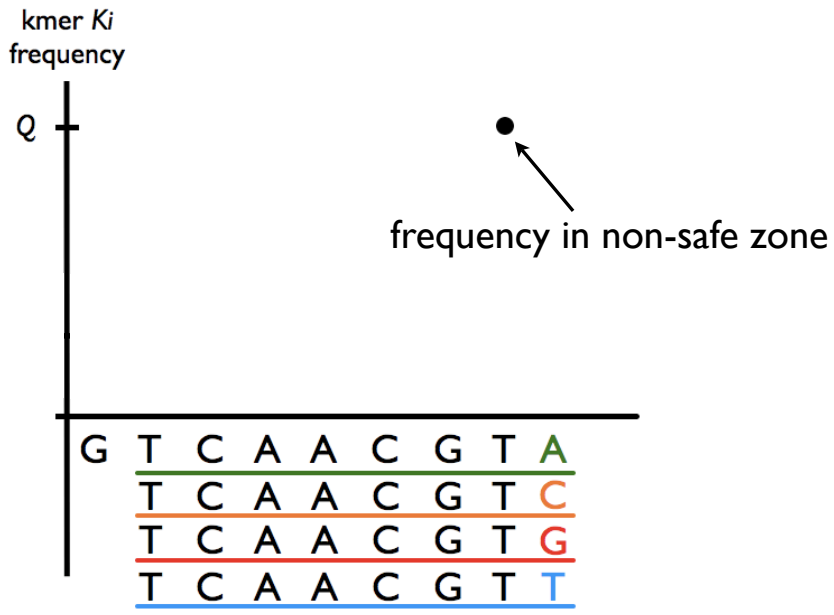
Detection of vectormers

Vectormers:
ACGTGTCA, CCCAAGTA, GTCATGCT, ...

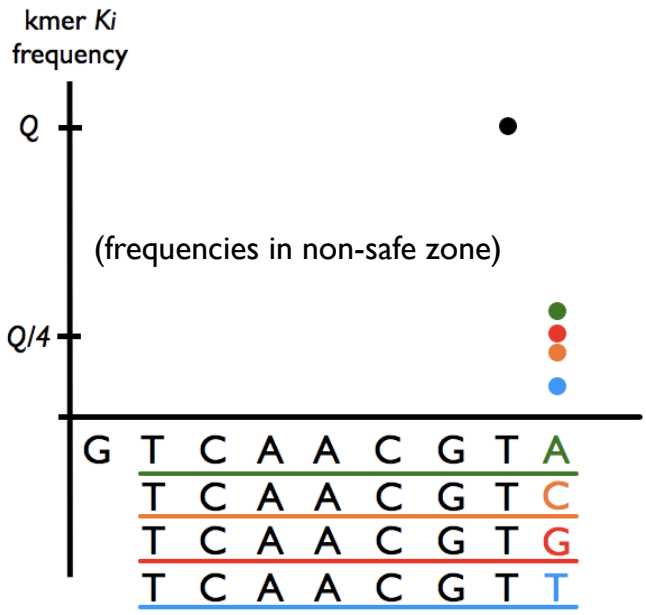
Which ones are most likely to represent the ends of the vector sequence? i.e which vectormers are ***endmers*.**

ATGTCACGTACAGTCACCCAAGTA.....

Detection of endmers

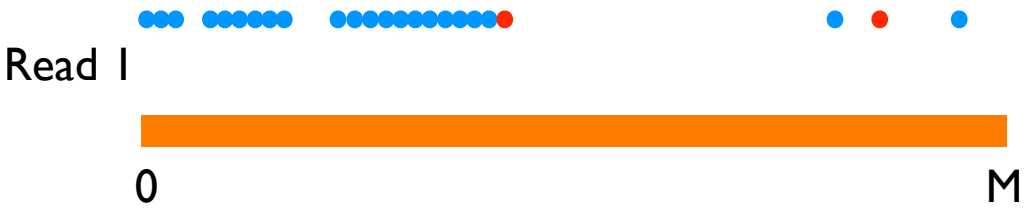


Detection of endmers



Vector clip estimation

- Now we know vectormers and endmers, so we go through each read again looking for them.



- Scanning window searches for a concentration of vectormers ending in an endmer.

D. pseudoobscura test

- sequencing adapters used in the project are known.
- searching for the two adapter sequences (16 bp each) using NUCMER.
- collected 1,506,679 reads that matched at least 8 bp of an adapter with at least 90% identity.

D. pseudoobscura test

e 3 Sensitivity and specificity results of Figaro on *Drosophila pseudoobscura* shotgun reads. Using
ld of 30, Figaro is able to remove virtually all vector sequence and only overtrims a small propor
y more than 3 bp. Note false positives and false negatives are computed only if they occur in the
egion of a read.

Max distance m	SN_m	SP_m	TP_m	FN_m	FP_m
0	99.98%	99.15%	1,493,582	316	12,781
3	99.99%	99.29%	1,500,662	186	5,831
5	~100%	99.72%	1,502,428	67	4,184
10	~100%	99.79%	1,503,481	54	3,144

Figaro usage

.USAGE.
figaro -F <reads file (fasta format)> -P <prefix> [options]

.OPTIONS.
-F reads file (fasta format)
-P output prefix
-T trimming threshold (optional, default is automated threshold estimation)
-M max cut length allowed (default 100)
-E end of safe zone (default 500)
-V verbose output (t or f) (default f)

run_figaro_lucy usage

.USAGE.
run_figaro_lucy -o <prefix> fasta1 ... fastan

.DESCRIPTION.
Outputs a set of clear ranges for the reads which includes vector trimming and quality trimming. The output is a clear range file: <prefix>.clr

Edit Makefile to include correct path to Lucy.

http://amos.sourceforge.net/Figaro