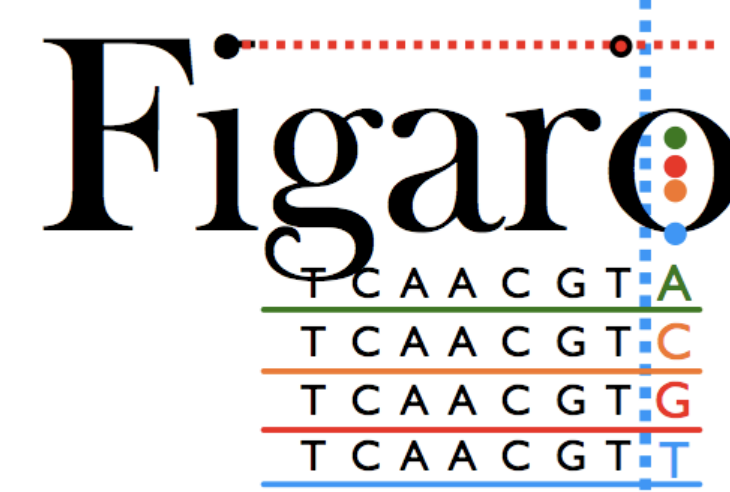


## Vector sequences can ruin analyses!

Be safe.



### Abstract

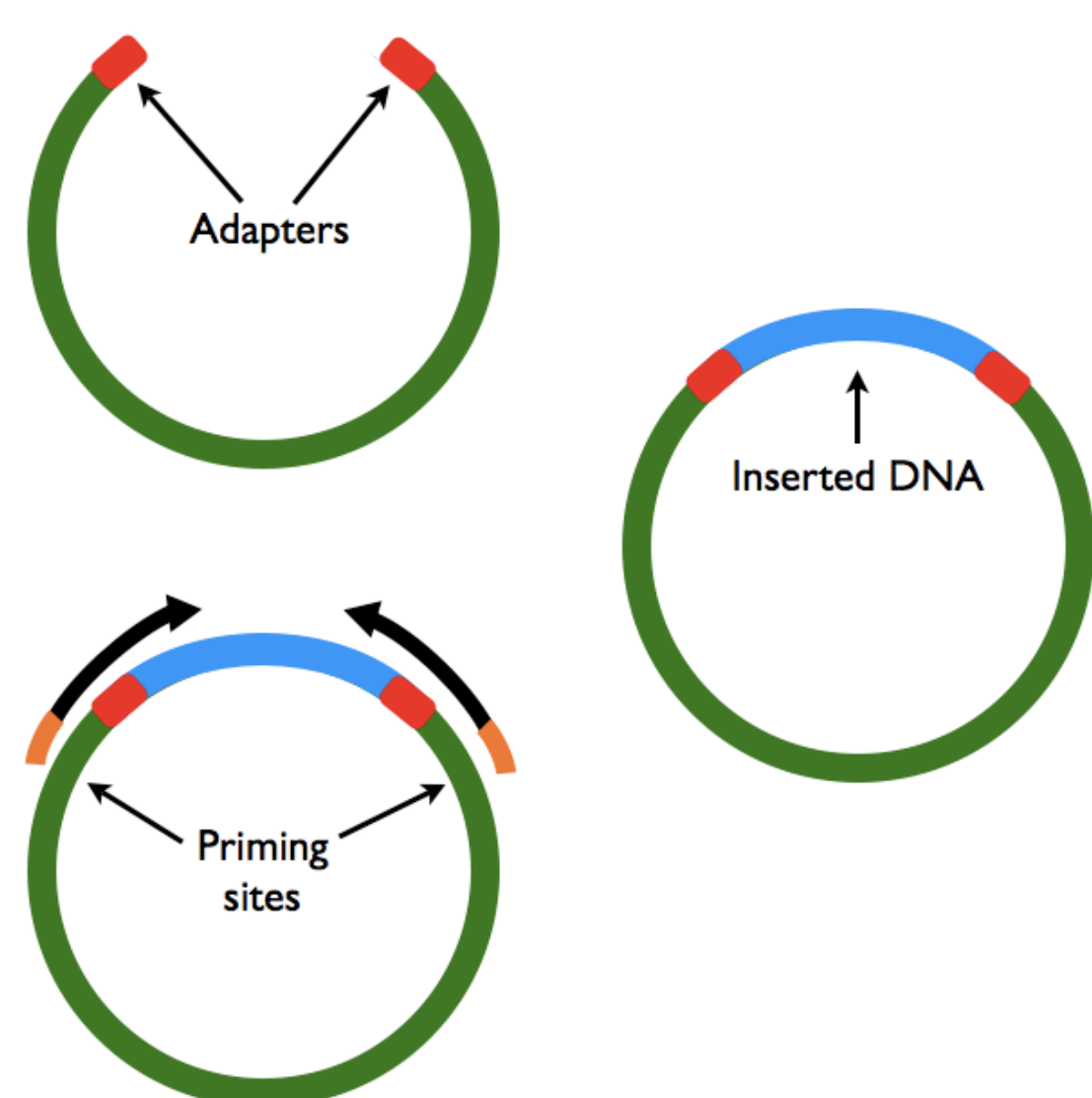
Sequences produced by automated Sanger sequencing machines frequently contain fragments of the cloning vector on their ends. Software tools currently available for identifying and removing the vector sequence require knowledge of the vector sequence, specific splice sites, and any adapter sequences used in the experiment. Such information is often unavailable in public databases, and many sequences deposited in the NCBI Trace Archive contain incorrect or missing vector clipping information, thereby complicating further analyses of the data. We present here Figaro, a novel software tool for identifying and removing the vector from raw sequence data without prior knowledge of the vector sequence. The vector sequence is automatically inferred by analyzing the frequency of occurrence of short oligo-nucleotides using Poisson statistics. We show that Figaro achieves 99.98% sensitivity when tested on ~1.5 million shotgun reads from *Drosophila pseudoobscura*. We further explore the impact of accurate vector trimming on the quality of whole-genome assemblies by re-assembling two bacterial genomes from shotgun sequences deposited in the Trace Archive. Designed as a module in large computational pipelines, Figaro is fast, lightweight, and flexible, and is released under an open-source license through the AMOS package (<http://amos.sourceforge.net>).

### Background

- high-throughput Sanger sequencing begins by cloning a DNA fragment into a vector (usually a plasmid).
- short adapter sequences are often attached to improve efficiency.
- sequencing reactions performed using universal sequencing primers nearby the splice site (fig. 1).
- each sequence contains a small section of the vector, as well as the adapters used during cloning, in addition to the original DNA fragment (fig. 2).

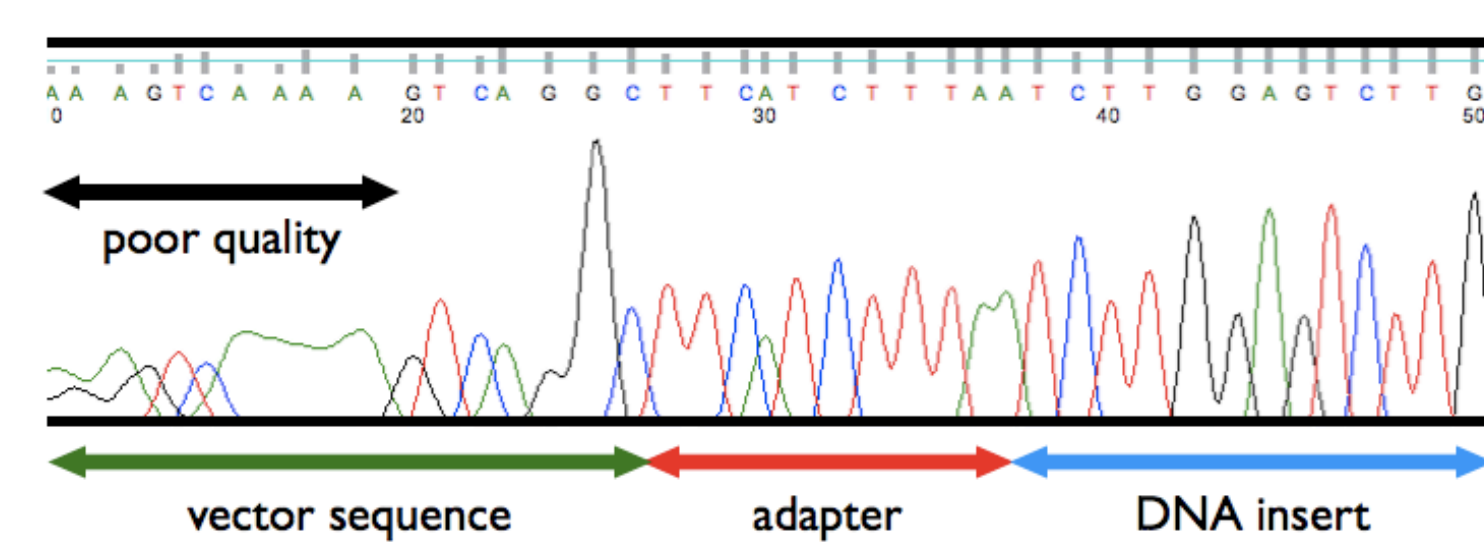
#### Figure 1

DNA from a sample (blue) is cloned into a small circular piece of DNA called a vector (green). Short adapters (red) are used to improve efficiency of cloning the sample DNA. The molecule is then transfected into *E. coli*, amplified, and then sequenced from both ends starting from priming sites (orange) inside the vector.



#### Figure 2

Raw output from sequencing machines contains poor quality sequence on the ends (black) as well as vector (green) and adapter sequence (red), in addition to the DNA being sequenced (blue).



#### Software tools available for vector removal

- Lucy (Chou and Holmes 2001)
- Crossmatch ([www.phrap.org/phredphrapconsed.html](http://www.phrap.org/phredphrapconsed.html))
- VecScreen ([www.ncbi.nlm.nih.gov/VecScreen](http://www.ncbi.nlm.nih.gov/VecScreen))

These programs requires three sets of information:

- the sequence of the cloning vector;
- the splice site used for sequencing; and
- the sequence of the cloning adapters (information that is often lost when the sequences are deposited in public databases).

Note that the NCBI Trace Archive provides a mechanism for recording the location within the read where the vector ends (*vector clip point*), however this information is often missing or incorrect.

**The good news is you do not need any of the above information to accurately trim vector sequence from high-throughput read sets.**

### Results

#### Simulated vector data

- trimmed first 300 bases of 19,633 high quality *Chlamydomonas reinhardtii* reads.
- attached vector sequence of random length (10 to 50 bp).
- used Smal cloning site of the pUC18 vector.
- no vector sequence was attached to about 20% of the reads.
- introduced different error rates in the vector sequence for each trial (0-5%).
- given parameter  $m$ :
  - $(TP_m)$  whenever the trimpoint is within  $m$  bases of the true trimpoint.
  - $(FP_m)$  overtrimming by more than  $m$  bases.
  - $(FN_m)$  undertrimming by more than  $m$  bases.
- Sensitivity and specificity are defined as follows:

$$SN_m = \frac{TP_m}{TP_m + FN_m}$$

$$SP_m = \frac{TP_m}{TP_m + FP_m}$$

Error rate	$SN_0$	$SP_0$	$SN_3$	$SP_3$	$SN_5$	$SP_5$	$SN_{10}$	$SP_{10}$
0%	100%	99.5%	100%	99.7%	100%	99.7%	100%	99.7%
1%	99.6%	99.3%	99.9%	99.7%	99.9%	99.7%	99.9%	99.8%
3%	98.0%	98.9%	99.0%	99.7%	99.1%	99.7%	99.3%	99.8%
5%	96.5%	98.0%	98.3%	99.6%	98.6%	99.6%	98.9%	99.7%

**Table 1**

Sensitivity and specificity results of Figaro on simulated vector contaminant sequence with different error rates. Introducing higher error rates reduces the programs ability to detect the vector sequence boundary, but even with an error rate of 5%, Figaro performs well, effectively removing nearly all of the vector sequence without significantly overtrimming reads.

#### *Drosophila pseudoobscura* data

- sequencing adapters used in the project are known.
- searching for the two adapter sequences (16 bp each) using NUCMER (Delcher et al. 2002; Kurtz et al. 2004).
- collected 1,506,679 reads that matched at least 8 bp of an adapter with at least 90% identity.
- 3' end of the vector was required to match within the first 50 bp.

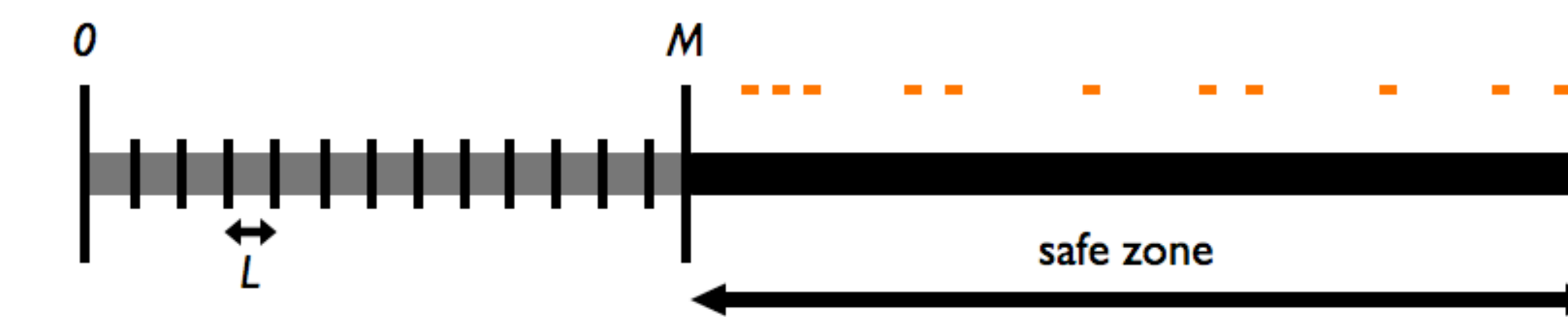
Max distance $m$	$SN_m$	$SP_m$	$TP_m$	$FN_m$	$FP_m$
0	99.98%	99.15%	1,493,582	316	12,781
3	99.99%	99.29%	1,500,662	186	5,831
5	~100%	99.72%	1,502,428	67	4,184
10	~100%	99.79%	1,503,481	54	3,144

**Table 2**

Sensitivity and specificity results of Figaro on *Drosophila pseudoobscura* shotgun reads. Using a threshold of 30, Figaro is able to remove virtually all vector sequence and only overtrims a small proportion of reads by more than 3 bp. Note false positives and false negatives are computed only if they occur in the high-quality region of a read.

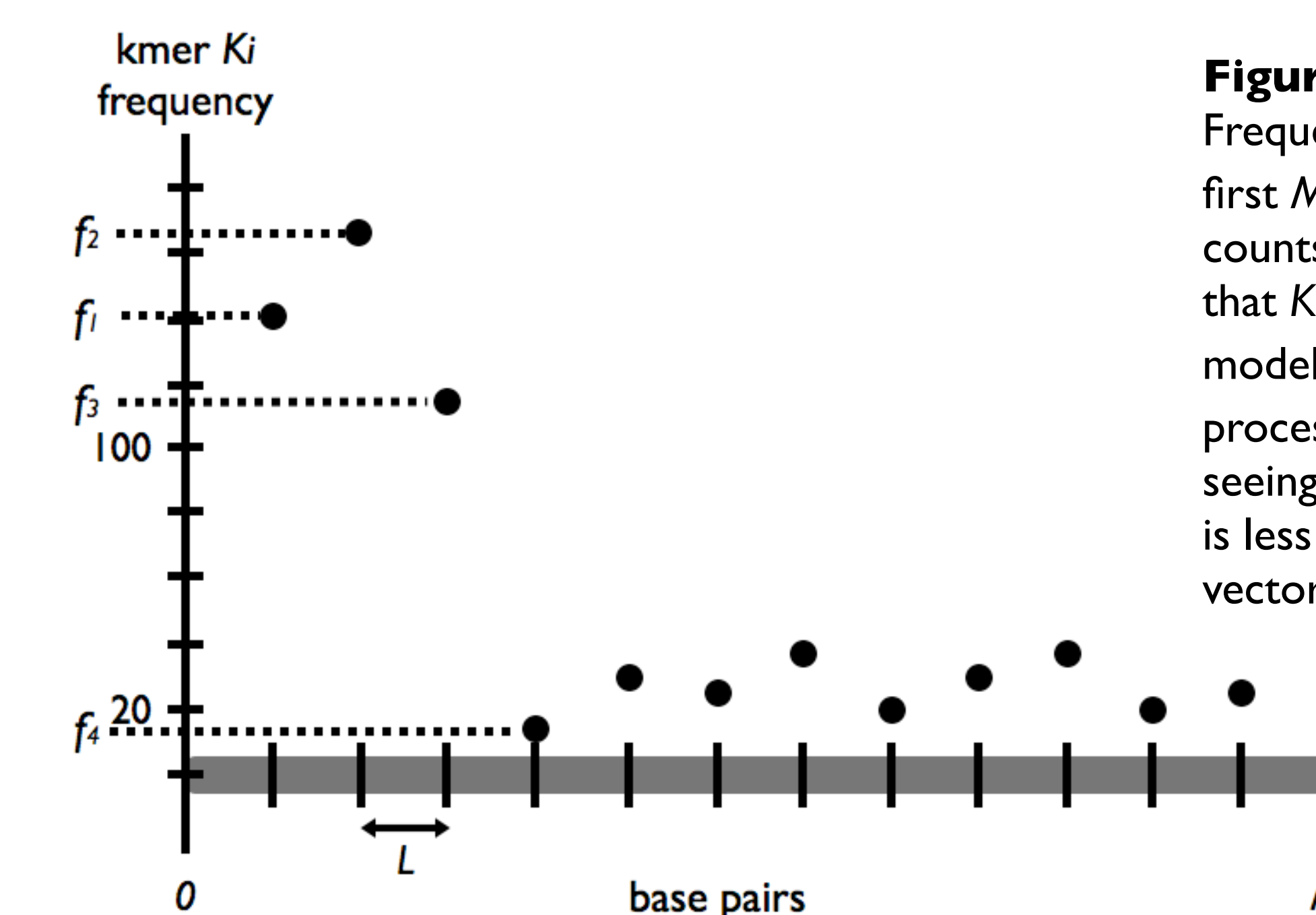
### Algorithms

#### Computation of vectormers (kmers likely to represent vector sequence)



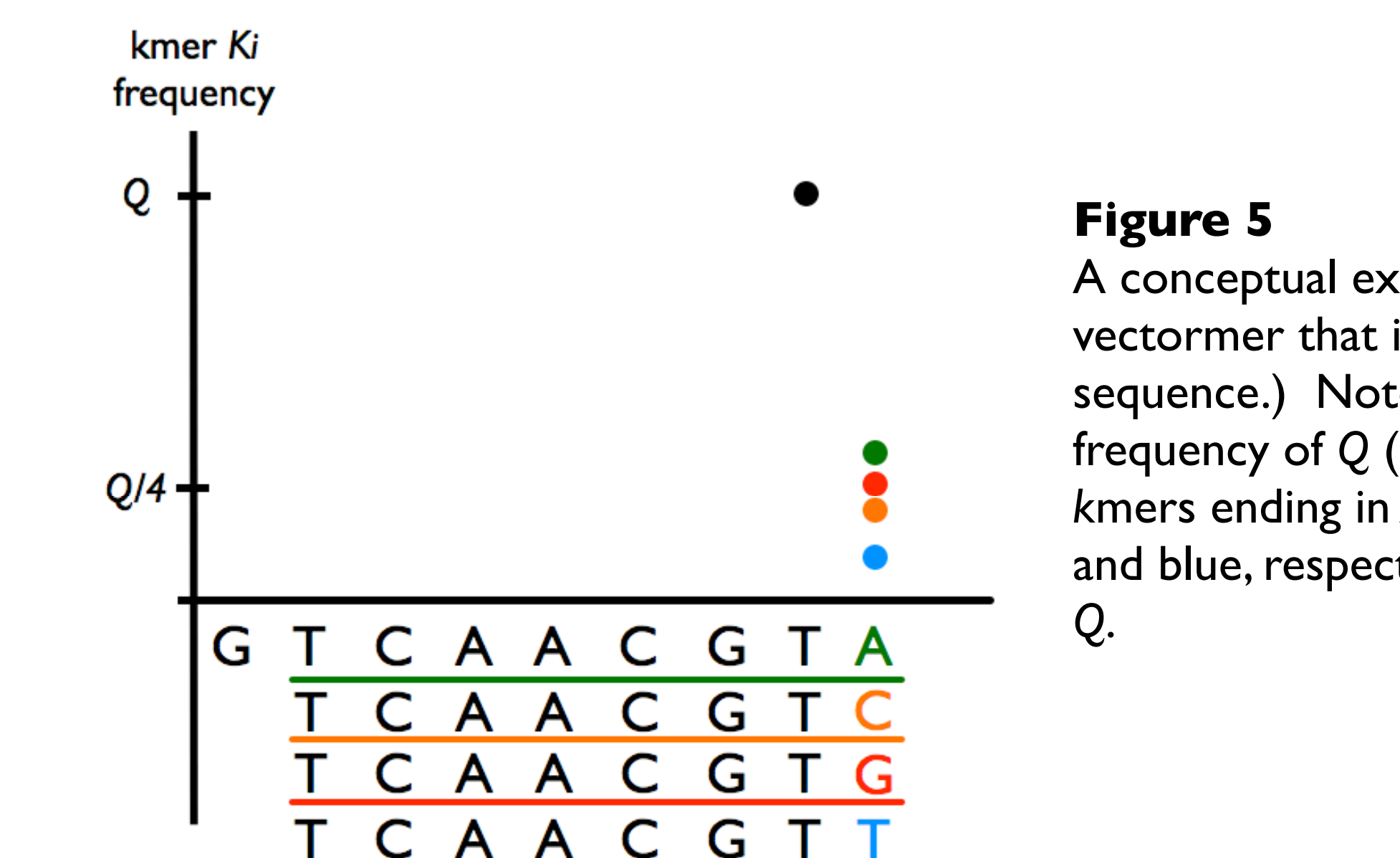
**Figure 3**

Within the safe zone of all reads, we consider the number of occurrences of each kmer  $K_i$  (orange), and calculate its average arrival rate. The beginning of the read is separated into bins of length  $L$  and the frequency of each kmer within each bin is recorded.



**Figure 4**

Frequency distribution for kmer  $K_i$  across first  $M$  bases of all reads. High frequency counts at the beginning of reads indicate that  $K_i$  is a likely vectormer. Statistically modeling  $K_i$  occurrences as Poisson process, we calculate the probability,  $P$ , of seeing the frequency within each bin. If  $P$  is less than 0.001 we declare it to be a vectormer.



**Figure 5**

A conceptual example of identifying endmers (i.e. a vectormer that is likely to be the end of the vector sequence.) Note that the kmer GTCAAGCT has a frequency of  $Q$  (black dot). Frequencies of adjacent kmers ending in A, C, G, and T (green, orange, red, and blue, respectively) are significantly lower than  $Q$ .

- once vectormers and endmers have been computed, a scanning window searches every read for high concentrations of vectormers.
- the first  $M$  base pairs of each sequence are examined right to left, using a 10 kmer (17 bp) moving window.
- if we encounter a window containing 7 or more vectormers that ends in an endmer, we set the vector trim point according to where the endmer's location.
- secondary searches are used if an endmer cannot be found.

### Bibliography

- Chou, H.H. and M.H. Holmes. 2001. DNA sequence quality trimming and vector removal. *Bioinformatics* **17**: 1093-1104.
- Delcher, A.L., A. Phillippy, J. Carlton, and S.L. Salzberg. 2002. Fast algorithms for large-scale genome alignment and comparison. *Nucleic Acids Res* **30**: 2478-2483.
- Kurtz, S., A. Phillippy, A.L. Delcher, M. Smoot, M. Shumway, C. Antonescu, and S.L. Salzberg. 2004. Versatile and open software for comparing large genomes. *Genome Biol* **5**: R12.