

# 1 hafeZ: Active prophage identification through 2 read mapping

3 Christopher J. R. Turkington<sup>†\*1</sup>, Neda Nezam Abadi<sup>\*2</sup>, Robert A. Edwards<sup>3</sup>, and  
4 Juris A. Grasis<sup>1</sup>

5 *<sup>1</sup>School of Natural Sciences, University of California, Merced, CA 95343, USA*

6 *<sup>2</sup>APC Microbiome Ireland & School of Microbiology, University College Cork, Cork, Ireland*

7 *<sup>3</sup>College of Science and Engineering, Flinders University, Bedford Park, SA 5042, Australia*

8 <sup>†</sup> corresponding author, <sup>\*</sup> these authors contributed equally

## 10 **Abstract**

---

11 **Summary:** Bacteriophages that have integrated their genomes into bacterial chromosomes,  
12 termed prophages, are widespread across bacteria. Prophages are key components of bacte-  
13 rial genomes, with their integration often contributing novel, beneficial, characteristics to the  
14 infected host. Likewise, their induction—through the production and release of progeny virions  
15 into the surrounding environment—can have considerable ramifications on bacterial communi-  
16 ties. Yet, not all prophages can excise following integration, due to genetic degradation by their  
17 host bacterium. Here, we present hafeZ, a tool able to identify ‘active’ prophages (i.e. those un-  
18 dergoing induction) within bacterial genomes through genomic read mapping. We demonstrate  
19 its use by applying hafeZ to publicly available sequencing data from bacterial genomes known to  
20 contain active prophages and show that hafeZ can accurately identify their presence and location  
21 in the host chromosomes.

22 **Availability and Implementation:** hafeZ is implemented in Python 3.7 and freely available  
23 under an open-source GPL-3.0 license from <https://github.com/Chrisjrt/hafeZ>. Bugs and issues  
24 may be reported by submitting them via the hafeZ github issues page.

25 **Contact:** [cturkington@ucmerced.edu](mailto:cturkington@ucmerced.edu) or [chrisjrt1@gmail.com](mailto:chrisjrt1@gmail.com)

---

## 32 **1 Introduction**

33 Bacteriophages, viruses that infect bacteria, are heavily involved in many aspects of bacterial  
34 physiology and evolution. Temperate bacteriophages in particular, are tightly interwoven into  
35 bacterial biology through their ability to integrate their DNA into their host’s chromosome via  
36 the lysogenic cycle of bacteriophage replication (Howard-Varona et al., 2017). When integrated,  
37 bacteriophages are called prophages and can contribute fitness changes to their hosts through  
38 expanding the hosts’ genetic repertoire (e.g. by providing toxin or antibiotic resistance genes); or  
39 by altering the expression of existing bacterial genes (e.g. by integrating directly into a bacterial  
40 gene and causing its disruption) (Ofir and Sorek, 2018).

41 However, prophages not only influence bacterial biology when integrated into the genome,  
42 they also influence bacterial behaviour through their induction. Here, prophages excise from  
43 the chromosome and begin virion production via the lytic cycle of bacteriophage replication—a  
44 process that is often fatal for the host bacterium. Through induction to the lytic cycle, temperate  
45 bacteriophages have been associated with major changes in the densities of both lysogenised

46 and non-lysogenised bacterial populations, alterations in bacterial biofilm formation levels, and  
47 changes in bacteria-host interactions in host-associated systems (Keen and Dantas 2018).

48 Given their importance, numerous computational tools have been developed in recent years  
49 to identify integrated bacteriophages within bacterial genomes. To date though, most prophage  
50 identification tools are unable to determine whether an identified prophage can be induced and  
51 no tool utilises prophage induction as a parameter for their detection. This is particularly im-  
52 portant because prophages are subject to decay in their host (usually by the accumulation of  
53 transposases), wherein some prophages will lose their ability to excise and produce viral parti-  
54 cles, leaving them as dormant regions of the bacterial genome (Bobay et al. 2014). Here we  
55 present hafeZ a tool that identifies active prophages (i.e. those undergoing induction) in bacte-  
56 rial genomes by examining the mapping of genome sequencing data to an assembly for signs of  
57 prophage induction. The workflow of hafeZ is described below and a summarised overview can  
58 be found in figure S1.

## 59 **2 Description**

### 60 **2.1 Inputs**

61 hafeZ requires three main inputs from the user: (1) a complete/contiguous genome assembly  
62 in FASTA format; (2) a set of reads for the given genome in FASTQ format; and (3) the path to  
63 the folder containing the Prokaryotic Virus Orthologous Groups (pVOGs; Grazziotin et al., 2017)  
64 database that is downloaded during the initial setup of hafeZ.

### 65 **2.2 Read mapping and region of interest identification**

66 hafeZ begins by examining the number and size of contigs in an assembly and removing small  
67 contigs (default = < 10,000 bp). Contigs passing this filter are then indexed and the reads are  
68 mapped to the contigs using minimap2 (Li, 2018), with mapping results converted to coverage  
69 depths using samtools (Li et al., 2009) and mosdepth (Pedersen and Quinlan, 2018). Coverage  
70 values are then smoothed using a Savitzky-Golay filter via the 'savgol\_filter' function of SciPy  
71 (Virtanen et al., 2020) to reduce instances of short lapses in coverage depth in otherwise heavily  
72 covered regions.

73 Once coverage values have been collected and smoothed, the modified Z-score of coverage  
74 per base across the length of the genome are then calculated using the equation of Iglewicz and  
75 Hoaglin, 1993:

$$76 M_i = \frac{0.675(x_i - \tilde{x})}{MAD} \quad (1)$$

77 Where  $M_i$  is the modified Z-score of a base's coverage,  $x_i$  is the coverage of that base,  $\tilde{x}$  is the  
78 median of all coverage values, and  $MAD$  is the median absolute deviation.  $MAD$  is the median of  
absolute deviations about the median for all per-base coverage values and is calculated as:

$$79 MAD = median_i\{|x_i - \tilde{x}|\} \quad (2)$$

80 The modified Z-score values are then used to identify regions within the genome with higher  
81 than expected coverage via numpy (Harris et al., 2020). Regions are called if they pass thresholds  
82 for minimum modified Z-score (default = 3.5) and region width (default = 4,000 bp). Regions  
83 passing these filters within a close vicinity of each other are then merged to create a 'region of  
84 interest' (ROI; figure S2).

85 To identify a potential deletion event, the reads mapping within each ROI, and those map-  
86 ping within 15,000 bp either side of it, are extracted using pysam (<https://github.com/pysam-developers/pysam>). For ROIs located centrally in a contig (defined as any ROI > 15,000 bp from  
87 either end of a contig) reads are examined for the presence of at least one read pair where read  
88 partners map a distance of roughly the length of the ROI from each other. This would indicate  
89 that their sequenced fragment includes a case where the prophage has excised, as the read pair  
90 exists closer than would be expected for the assembly (figure S3-A). If no distant reads are found,

91 the ROI is dropped from further analysis. For ROIs passing this threshold and ROIs located near  
92 the end of a contig, the reads are then searched for any soft clipped reads (reads where only part  
93 of a read maps to a given location; figure S3-B). For each ROI, assembly positions where at least  
94 10 reads have been soft clipped are collected. Then, all combinations of these locations that  
95 occur on opposite sides of the ROI center are collated. The collated combinations then serve as  
96 the basis for putative prophage start and end locations for each ROI to then be further filtered.

## 97 **2.3 Region of interest filtering**

98 Each combination of start-end locations are then filtered by examining the modified Z-scores in  
99 and around those locations. If a prophage induction event had occurred, it would be expected  
100 that the region between the start and end positions would have a median modified Z-score value  
101 greater than the Z-score threshold, while in the regions preceding the start and proceeding the  
102 end locations the median modified Z-scores should be below the Z-score threshold as these  
103 should be bacterial chromosome regions. Therefore, hafeZ first removes any ROI start and end  
104 location combinations where this is not the case. For ROIs located at the start/end of a contig, only  
105 the region between the ROI start/end and the sides not near contig ends are used in filtering. The  
106 best combinations of each soft clipping location passing the Z-score filter are retained (default  
107 = best 50), using the sum of start/end soft clipped read count as the scoring metric (higher =  
108 better).

109 As plasmids would pass all filters to this point, hafeZ then examines any ROI start/end com-  
110 binations near the ends of a contig for indications that the contig is circular (i.e. reads mapping  
111 from the end of a contig to the start and vice versa). If true, these ROIs are then flagged as  
112 circular but are carried forward as they may also be extra-chromosomal prophages.

113 To further filter ROI start/end combinations, hafeZ then maps the clipped portion of soft clipped  
114 reads for all non-circular ROIs and examines their mapping location. Here, hafeZ examines if the  
115 clipped portion of the read maps near the position of where the reads on the opposing side of the  
116 ROI were clipped (figure S3-C). This step adds additional stringency to hafeZ but can be disabled  
117 by using the '-N/- -no\_extra' option.

118 The single best start/end combination for each ROI is then collected, with the best start/end  
119 combination being that with the highest sum of soft clipped reads at the start/end plus the num-  
120 ber of clipped sections from these mapping near the opposing position.

## 121 **2.4 Sequence analysis**

122 After ROI coordinates have been determined, genes are then predicted using Pyrodigal  
123 (<https://github.com/althonos/pyrodigal>), a cythonised version of Prodigal (Hyatt et al., 2010), and  
124 the sequences of the predicted open reading frames (ORFs) are then extracted. ROIs contain-  
125 ing less than a minimum number of ORFs (default = 6) are then removed, with the peptide  
126 sequences encoded by each gene of the surviving ROIs then compared to the pVOGs database  
127 using hmmsearch (HMMER v3.3.1; <http://hmmer.org>). ROIs are retained if the proportion of ORFs in  
128 the ROI showing similarity to a pVOGs hidden Markov model (HMM) profile passes a user-defined  
129 threshold (default = 0.1 i.e. 10% of ORFs). The att sites for each of ROI are then determined by  
130 extracting the region  $\pm 100$  bp either side of the ROI start/end locations and examining them for  
131 homology using BLASTn (Camacho et al. 2009) with the settings '-evalue 10000 -task blastn-  
132 short'. The putative att sites with the lowest e-values and a length  $> 11$  bp are then output as  
133 the potential att sites for each ROI.

## 134 **2.5 Output**

135 hafeZ generates all outputs in the path provided by the user. If an ROI passes all filters, six main  
136 outputs are produced: (1) a multi-FASTA file containing the DNA sequence of each ROI identified;  
137 (2) a fasta file for each ROI containing the DNA sequences of all ORFs; (3) a fasta file for each ROI  
138 containing the amino acid sequences of all ORFs; (4) a tab-separated file containing details of  
139 ORFs hit by the pVOGs comparisons; (5) a tab-separated summary file containing key information  
140 on all identified putative prophage regions; and (6) a figure showing Z-score distributions for each

141 contig found to contain a ROI with the location of the ROI highlighted (e.g. figure S4). If no ROIs  
142 are found, or no ROIs pass the filtering process, only an empty tab-separated summary file will  
143 be output.

## 144 **2.6 Example usage**

145 In their work, Zünd et al., 2021 used a combination of comparing the coverage mapping between  
146 induced vs non-induced samples and read examination on 14 bacterial isolates to identify 10  
147 active prophages. To illustrate the ability of hafeZ to identify active prophages from bacterial  
148 sequencing data, we applied hafeZ to this sequencing data-set (European Nucleotide Archive  
149 project No. PRJEB39818) using default settings and the corresponding publicly available refer-  
150 ence assemblies as mapping targets for each read-set (table S1). We found that overall hafeZ  
151 was able to identify all 10 prophages in the data-set with no deviation in predicted start/end lo-  
152 cations compared to those of Zünd et al., 2021. However, as hafeZ analyses individual samples  
153 for the presence of prophages and the Zünd data-set contains triplicate induced and non-induced  
154 samples that were compared to identify their prophages in their work, the presence or absence of  
155 each of these prophages was dependant on the sample. hafeZ did identify one element not men-  
156 tioned by Zünd, corresponding to a plasmid in *Salmonella enterica* serovar Typhimurium LT2<sup>p22</sup>.  
157 This plasmid was flagged as circular in all cases though, and thus can be easily identified as  
158 such. A table summarising the presence/absence and positional differences for each expected  
159 prophage can be found in table S2, while the combined outputs of hafeZ for all samples can be  
160 found in table S3.

## 161 **3 Conclusion**

162 We show that hafeZ is highly accurate at detecting active prophages in bacterial genomes and  
163 that it is able to identify prophages in a diverse range of samples and organisms. We believe  
164 hafeZ is an ideal tool to be used in addition to current integration-focused prophage identification  
165 tools to give highly accurate prophage positions, describe the activity of prophages identified by  
166 other tools, and, through its novel use of induction as an identification metric, could potentially  
167 identify novel viruses that would be missed by existing prophage detection algorithms.

## 168 **Funding**

169 This work was supported by startup research funds provided by the University of California  
170 Merced School of Natural Sciences.

## 171 **References**

172 Bobay, L.-M., Touchon, M., & Rocha, E. P. C. (2014). Pervasive domestication of defective prophages  
173 by bacteria. *Proceedings of the National Academy of Sciences*, 111(33), 12127–12132.

174 Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., & Madden, T. L.  
175 (2009). BLAST+: architecture and applications. *BMC bioinformatics*, 10, 421.

176 Grazziotin, A. L., Koonin, E. V., & Kristensen, D. M. (2017). Prokaryotic Virus Orthologous Groups  
177 (pVOGs): A resource for comparative genomics and protein family annotation. *Nucleic  
178 Acids Research*, 45(D1), D491–D498.

179 Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser,  
180 E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett,  
181 M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., ... Oliphant, T. E. (2020). Array  
182 programming with NumPy. *Nature*, 585(7825), 357–362.

183 Howard-Varona, C., Hargreaves, K. R., Abedon, S. T., & Sullivan, M. B. (2017). Lysogeny in nature:  
184 mechanisms, impact and ecology of temperate phages. *The ISME Journal*, 11(7), 1511–  
185 1520.

186 Hyatt, D., Chen, G.-L., LoCascio, P. F., Land, M. L., Larimer, F. W., & Hauser, L. J. (2010). Prodigal:  
187 prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics*, 11(1), 119.  
188

189 Iglewicz, B., & Hoaglin, D. C. (1993). *How to detect and handle outliers* (Vol. 16). Asq Press.  
190 Keen, E. C., & Dantas, G. (2018). Close Encounters of Three Kinds: Bacteriophages, Commensal  
191 Bacteria, and Host Immunity. *Trends in microbiology*, 26(11), 943–954.  
192 Li, H. (2018). Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics (Oxford, England)*, 34(18), 3094–3100.  
193

194 Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin,  
195 R., & 1000 Genome Project Data Processing Subgroup. (2009). The Sequence Alignment/Map  
196 format and SAMtools. *Bioinformatics (Oxford, England)*, 25(16), 2078–9.  
197 Ofir, G., & Sorek, R. (2018). Contemporary Phage Biology: From Classic Models to New Insights.  
198 *Cell*, 172(6), 1260–1270.  
199 Pedersen, B. S., & Quinlan, A. R. (2018). Mosdepth: quick coverage calculation for genomes and  
200 exomes. *Bioinformatics (Oxford, England)*, 34(5), 867–868.  
201 Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E.,  
202 Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman,  
203 K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., ... SciPy 1.0 Contribu-  
204 tors. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature  
methods*, 17(3), 261–272.  
205

206 Zünd, M., Ruscheweyh, H. J., Field, C. M., Meyer, N., Cuenca, M., Hoces, D., Hardt, W. D., & Suna-  
207 gawa, S. (2021). High throughput sequencing provides exact genomic locations of in-  
208 ducible prophages and accurate phage-to-host ratios in gut microbial strains. *Microbiome*,  
209 9(1), 1–18.

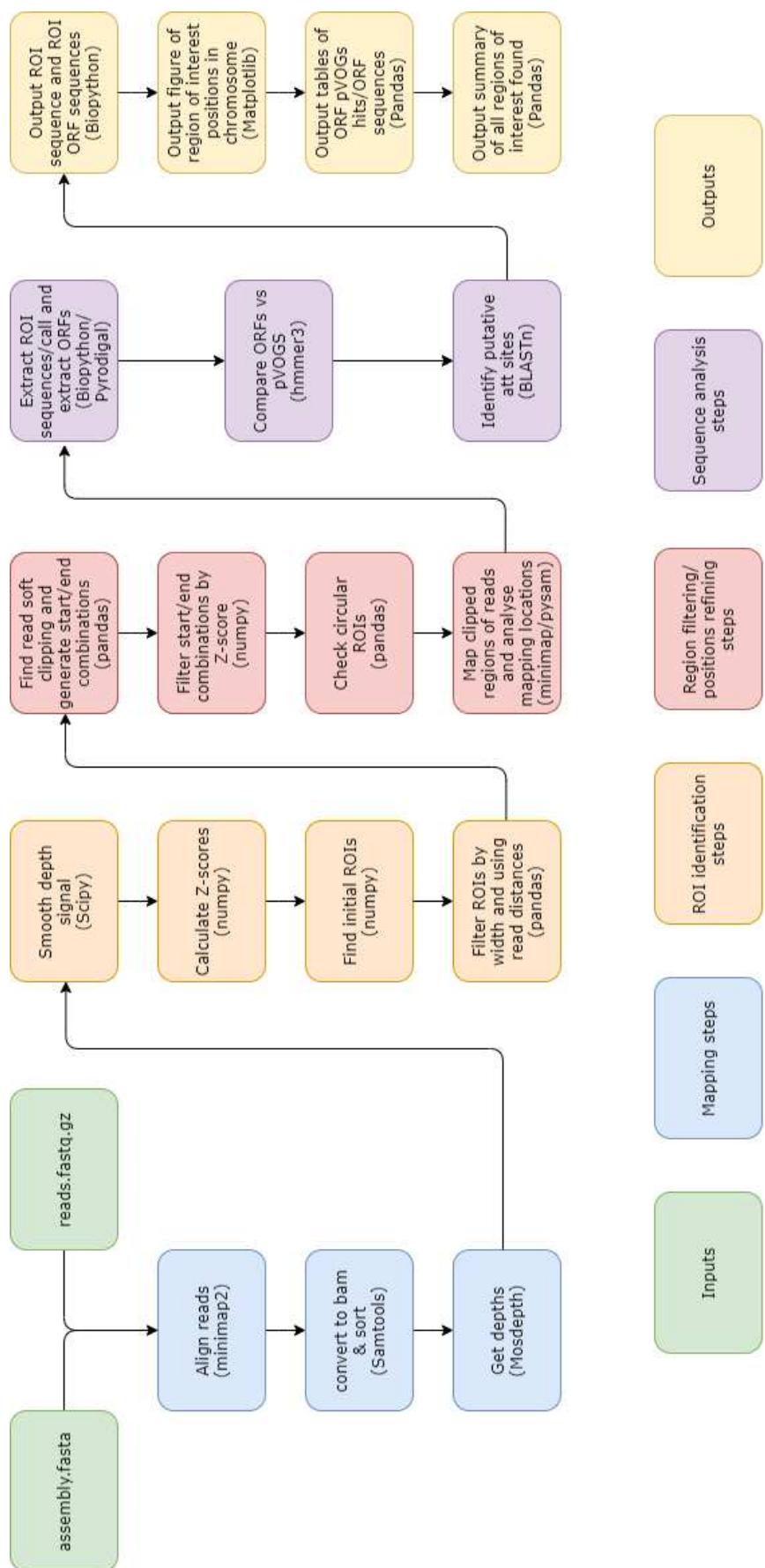


Figure S1: Overview of steps involved in the hafeZ pipeline.

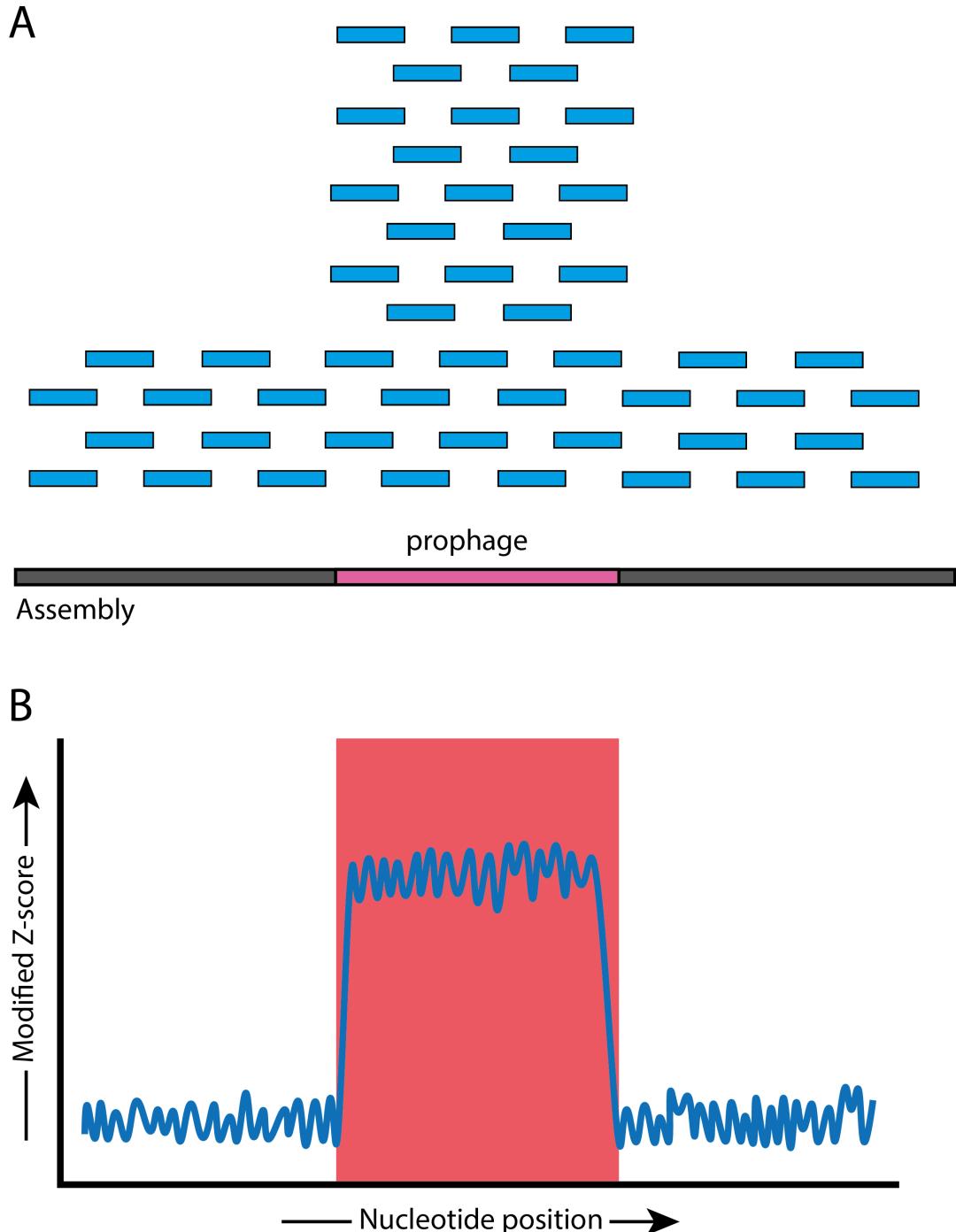


Figure S2: Illustration of how active prophages can be detected from read mapping coverages. A. When reads (blue bars) are mapped to an assembly (grey bar), if a prophage (pink region) exists within the genome that has been induced, a higher frequency of reads mapping to the region of the genome containing the prophage would be expected as copies of the viral DNA should be being produced. B. By converting the per base coverage of the read mapping results to modified Z-score values we can detect regions with higher levels of coverage than the surrounding chromosome (red highlighted region) that may indicate prophage induction.

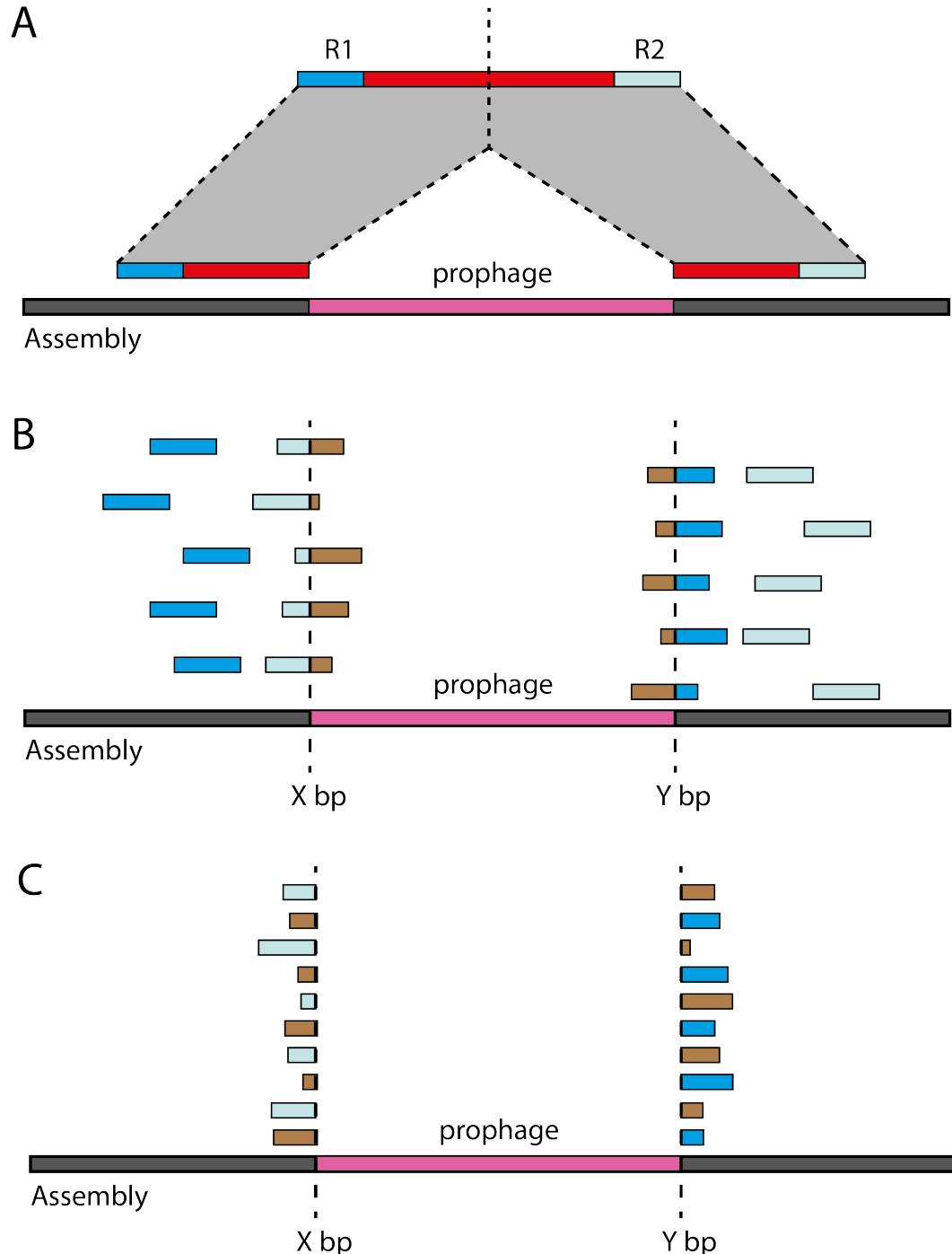


Figure S3: Illustration of how read mapping is used to identify putative active prophages by hafeZ following ROI identification. (A) hafeZ examines reads mapping within an ROI and those mapping within 15,000 bp either side of it for distantly mapping reads. Such reads would indicate that the DNA fragment (red bar) sequenced by a pair of reads (blue ends) contains a deletion compared to the DNA sequence of the genome assembly (dark grey bar) being mapped to. (B) hafeZ inspects the reads mapping within an ROI for the presence of soft-clipped reads, reads where only part of a read maps to the assembly at a given location (blue regions at dotted line = mapped region, brown = clipped region). If a sufficient number of these occur at a consistent bp position (dotted line) this position is then taken as either the start/end of the ROI. (C) By default hafeZ then re-maps the regions soft clipped by the initial read mapping step to determine if these reads map at the opposing end of the ROI. This would indicate that the read has sequenced the deleted region where the prophage has excised from.

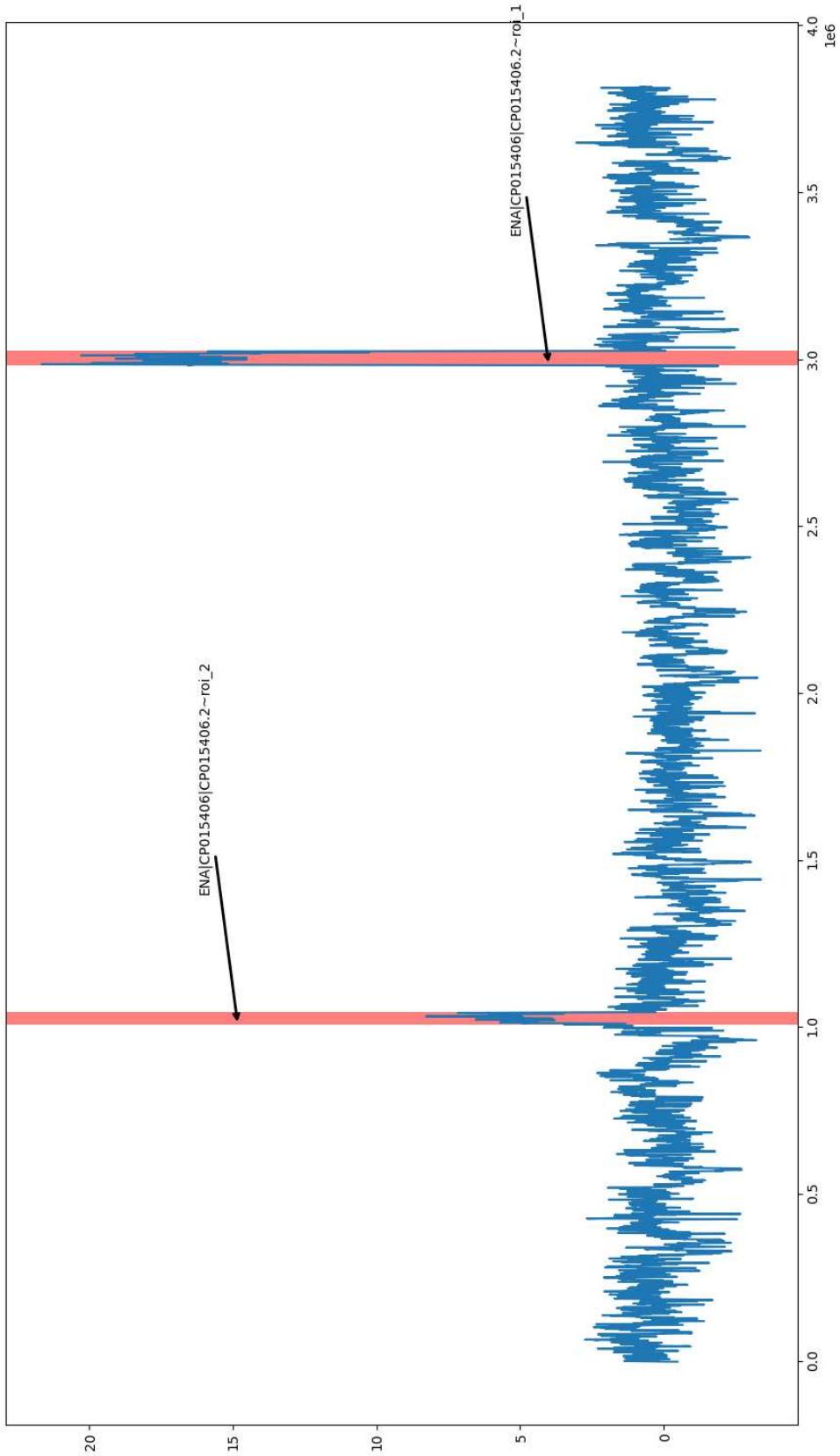


Figure S4: Example of the ROI position figure output by hafeZ for sample YL31\_C1. Figure shows modified Z-score on Y-axis and base pair position on X-axis. Each red area signifies the position of a putative active prophage.

Table S1: List and pairings of genome assemblies and reads examined for each sample in this study.

Table S2: Comparisons of the expected positions for the 10 prophages identified by Zünd et al. 2021 and the hafeZ identifications for each sample. Note - The assembly used by Zünd et al., for YL58 could not be found in public databases. Therefore, although the table shows that a new prophage was identified and the expected prophage in YL58 was not identified, we used BLAST to compare the Zünd prophage sequence to the assembly used here and found that the Zünd prophage mapped to the exact start and end positions identified by hafeZ.

Table S3: Table containing the combined output of all summary tables produced for each sample's hafeZ run.