Neptune: A Tool for Rapid Genomic Signature Discovery

Eric Marinier[1], Chrystal Berry[1], Kelly Weedmark[1], Michael Domaratzki[2], Phillip Mabon[1], Natalie Knox[1], Aleisha Reimer[1], Morag Graham[1,3], The Canadian Listeria Detection and Surveillance using Next Generation Genomics (LiDS-NG) Consortium[†] and Gary Van Domselaar [*1,3]

[1] *National Microbiology Laboratory, Public Health Agency of Canada, Winnipeg, MB, Canada*
[2] *Department of Computer Science, University of Manitoba, Winnipeg, MB, Canada*
[3] *Department of Medical Microbiology and Infectious Diseases, University of Manitoba, Winnipeg, MB, Canada*

Abstract

Neptune locates genomic signatures using an exact k-mer matching strategy while accommodating k-mer mismatches. The software identifies sequences that are sufficiently represented within inclusion targets and sufficiently absent from exclusion targets. The signature discovery process is accomplished using probabilistic models instead of heuristic strategies. We have evaluated Neptune on *Listeria monocytogenes* and *Escherichia coli* data sets and found that signatures identified from these experiments are highly sensitive and specific to their respective data sets. Neptune has broad implications in bacterial characterization for public health applications due to its efficient signature discovery based upon differential genomics. In addition, the identified loci may also provide a source material for research leading to investigations of group-specific traits.

1 Introduction

The ability to identify and respond to emergent infectious agents in a time sensitive manner is critical for ensuring public health safety [20]. The advancement of high-throughput next generation sequencing (NGS) has allowed the possibilities of using computational approaches for effective, real-time, comprehensive outbreak investigation and response. An important component of public health response is the characterization of infectious agents. This characterization involves discovering discriminatory signature sequences which aim to uniquely identify a group of organisms of interest from a background group.

This work defines a signature as a string of characters, representing nucleotide bases, capable of discriminating targets of interest from a background

[*]Corresponding author: gary.vandomselaar@phac-aspc.gc.ca

group. These signatures are sufficiently unique to a set of targets and sufficiently dissimilar from any sequence within a set of related non-targets. We define the intended group of interest as the "inclusion group," the background as the "exclusion group," and a reference as any inclusion target from which to extract signatures. Targets will typically comprise of fully-assembled or draft genomes. Signature discovery aims to locate unique and conserved regions within the inclusion group that are not present within an exclusion group background. Signatures will apply within the context of the user-defined groups; however, their sensitivity and specificity may not hold when applied in a broader context.

A naive approach to signature discovery involves exhaustively comparing all sequences using alignments to locate signature regions. However, such approaches do not scale effectively. An approximation to exhaustive comparisons is sequence clustering, but clustering without optimization may remain too slow. An effective algorithm is both sensitive and specific, while remaining computationally tractable. There are two common approaches towards ensuring sensitivity, which trade speed and sensitivity. The first approach requires inclusion sequence to match exactly [17]. This approach is extremely fast, but will be confounded by regions that are not highly conserved. The second approach involves grouping similar sequences together using multiple sequence alignments [20], seeding techniques [18], or leveraging clustering information [2]. While these approaches are more sensitive, they are necessarily slower than exact matching techniques. TOFI avoids this problem by only locating signatures for a single target and not a group. The specificity of signatures is verified using computationally expensive alignments of signature candidates [18–20] against the background, which typically involves using BLAST [1] alignments. However, verification is performed after significant data reduction, making this possible. KPATH [20] performs verification by comparing a consensus sequence produced from inclusion targets to a large non-target database. KPATH achieves acceptable speeds by leveraging suffix trees to find matches.

A significant data reduction is required to perform signature discovery in a reasonable time [18–20]. This involves identifying and removing sequences that are "definitely not unique" [20] in a computationally inexpensive manner. Insignia [17], TOFI [19], and TOPSI [18] use MUMmer [10] to precompute exact matches within inclusion targets and an exclusion background. However, depending on the size of the background database, this may remain a computationally expensive operation. CaSSiS [2] approaches the problem of signature discovery more thoroughly than other signature discovery pipelines. The software produces signatures simultaneously for all locations in a hierarchically clustered data set, such as a phylogentic tree, thereby producing candidate signatures for all possible subgroups. However, this process requires the input data to be provided in a hierarchically clustered format.

Neptune leverages existing strategies for signature detection by using an exact-matching k-mer strategy for speed, while making allowances for inexact matches to enhance sensitivity. However, unlike other existing exact matching approaches [17], Neptune performs signature discovery without precomputation or restriction on targets. Furthermore, Neptune locates signatures that are not perfectly conserved. Lee and Sheu [11] remark that existing signature discovery approaches are not readily parallelizable. With this in mind, Neptune is designed to operate on a high performance computing cluster. Neptune extracts signatures from one or more targets, in a highly parallelizable manner, and is

independent of multiple sequence alignments. Finally, Neptune's signature discovery pipeline is guided with probabilistic models, rather than heuristics, and therefore makes decisions with a degree of certainty.

2 Methods

Neptune uses the distinct k-mers found in each inclusion and exclusion target to identify sequences that are conserved within the inclusion group and absent from the exclusion group. Neptune evaluates all sequence and may therefore produce signatures that correspond to intergenic regions or contain multiple genes. The k-mer generation step produces distinct k-mers from all targets and aggregates this information, reporting the number of inclusion and exclusion targets that contain each k-mer. The signature extraction step identifies candidate signatures from one or more references which are assumed to additionally be inclusion targets. Candidate signatures are filtered by performing an analysis of signature specificity using pairwise sequence alignments. The remaining signatures are ranked by their Neptune-defined sensitivity and specificity scores.

We provide descriptions of the different stages of signature discovery below and an overview of the signature discovery process is found in Figure 1. The majority of parameters are automatically calculated by Neptune for every reference. However, the user may specify any of these parameters. A full description of the mathematics used in the software is available in supplementary materials. We assume that the probability of observing a nucleotide base in a sequence is independent from all other positions and the probability of all single nucleotide variant (SNV) events (e.g., mutations, sequencing errors) occurring is independent of all other SNV events.

2.1 *k*-mer Generation

Neptune produces the distinct set of k-mers from every inclusion and exclusion target and aggregates these k-mers together. The software is concerned only with the existence of a k-mer within each target and not with the number of times a k-mer is repeated within a target. Neptune converts all k-mers to the lexicographically smaller of either the forward k-mer or its reverse complement. This avoids maintaining both the forward and reverse complement sequence [14]. The number of possible k-mers is bound by the total length of all targets. The k-mers of each target are determined independently and, when possible, in parallel. In order to facilitate parallelizable k-mer aggregation, the k-mers for each target may be organized into several output files. The k-mers in each file are unique to one target and all share the same initial sequence index. This degree of organization may be specified by the user.

The k-mer length is automatically calculated unless provided by the user. A summary of recommended k-mer sizes for various genomes can be found in supplementary material. We suggest a size of k such that we do not expect to see two arbitrary k-mers within the same target match exactly. This suggestion is motivated by wanting to generate distinct k-mer information, thereby having matching k-mers most often be a consequence of homology. Let λ be the most extreme GC-content of all targets and ω be the size of the largest target in

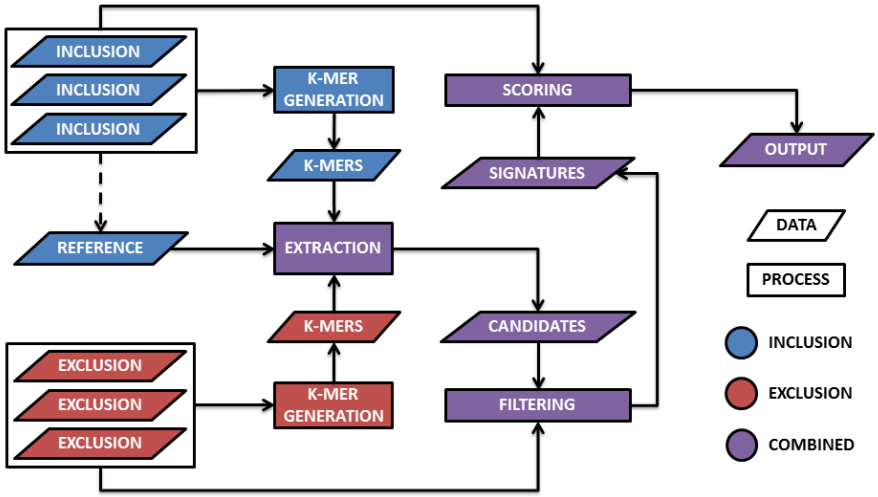

Figure 1: An overview of Neptune's signature discovery process. The first step involves generating k-mers from all inclusion and exclusion targets. These k-mers are aggregated and provided as input to signature extraction. Signature extraction produces candidate signatures, which are filtered and then sorted by their sensitivity and specificity scores.

bases. The probability of any two arbitrary k-mers, k_X and k_Y, matching exactly, $P(k_X = k_Y)_A$, where $x \neq y$, is defined as follows:

$$P(k_X = k_Y)_A = \left(2\left(\frac{1-\lambda}{2}\right)^2 + 2\left(\frac{\lambda}{2}\right)^2\right)^k \tag{1}$$

We use the probability of arbitrary k-mers matching, $P(k_X = k_Y)_A$, to approximate the probability of k-mers matching within a target, $P(k_X = k_Y)$. This is an approximation because the probability of $P(k_{X+1} = k_{Y+1})$ is not independent of $P(k_X = k_Y)$. However, this approximation approaches equality as $P(k_X = k_Y)_A$ decreases, which is accomplished by selecting a sufficiently large k, such that we do not expect to see any arbitrary k-mer matches. We suggest using a large enough k such that the expected number of intra-target k-mer matches is as follows:

$$\sum_{x<y} P(k_X = k_Y) \approx \binom{\omega - k + 1}{2} \cdot P(k_X = k_Y)_A < 0.05 \tag{2}$$

$$\frac{(\omega - k + 1)(\omega - k)}{2} \cdot \left(2\left(\frac{1-\lambda}{2}\right)^2 + 2\left(\frac{\lambda}{2}\right)^2\right)^k < 0.05 \tag{3}$$

The distinct sets of k-mers from all targets are aggregated into a single file which is used to inform signature extraction. This process may be performed

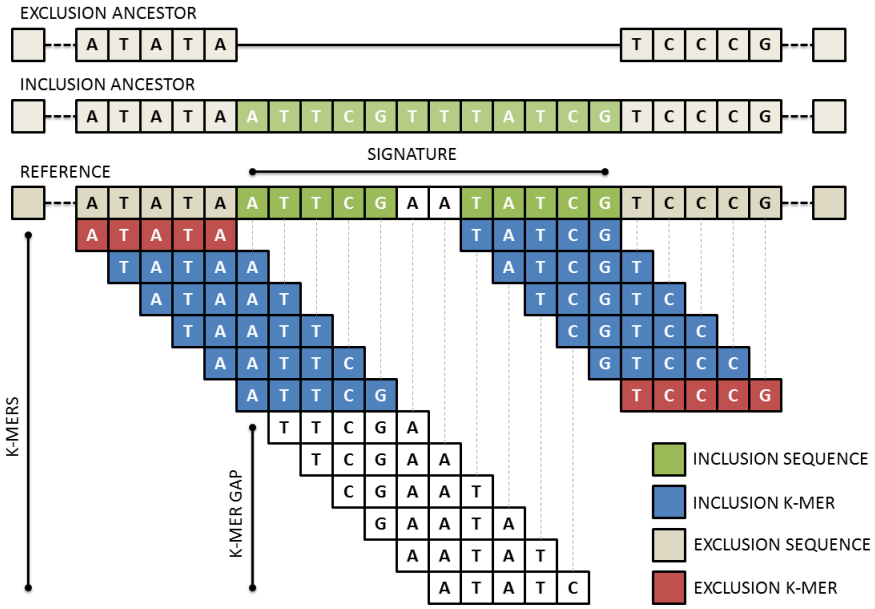

Figure 2: An overview of Neptune's signature extraction process. The reference is decomposed into its composite k-mers. These k-mers may be classified as either inclusion or exclusion and are used to infer inclusion and exclusion sequence in the reference. A signature is constructed from inclusion k-mers containing sufficiently small k-mer gaps and no exclusion k-mers.

in parallel by aggregating k-mers sharing the same initial sequence index and concatenating aggregated files. Aggregation produces a list of k-mers and two values corresponding to the number of inclusion and exclusion targets containing the k-mer. This information is used in the signature extraction step to categorize some k-mers as inclusion or exclusion k-mers.

2.2 Extraction

Signatures are extracted from one or more references, which are drawn from all inclusion targets, unless specified otherwise. However, our probabilistic model assumes all references are included as inclusion targets. In order to identify candidate signatures, Neptune reduces the effective search space of signatures by leveraging the spatial sequencing information inherent within the references. Neptune evaluates all k-mers in each reference, which may be classified as inclusion or exclusion k-mers. An inclusion k-mer is observed in a sufficient number of inclusion targets and not observed in a sufficient number of exclusion targets. The sufficiency requirement is described below. Inclusion and exclusion k-mers are used to infer inclusion and exclusion sequence, with signatures containing primarily inclusion sequence. An inclusion k-mer may contain both inclusion and exclusion sequence because, while they may contain exclusion sequence, these k-mers will be unique to the inclusion group. An exclusion k-mer is, by

default, any k-mer which has been observed at least once in any exclusion target. However, in some applications it may be desirable to relax this stringency. For example, leniency may be appropriate when the inclusion and exclusion groups are not well understood. An exclusion k-mer should not contain any inclusion sequence. A candidate signature begins with the last base position of the first inclusion k-mer, contains allowable k-mer gaps and no exclusion k-mers, and ends with the first base position of the last inclusion k-mer (Figure 2). This process is conceptually similar to taking the intersection of inclusion k-mers and allowable k-mer gaps. Furthermore, it avoids generating a candidate containing exclusion sequence found in inclusion k-mers which overlap inclusion and exclusion sequence regions.

An inclusion k-mer is considered sufficiently represented when it is observed in a number of targets exceeding a minimum threshold. We assume that if there is a signature present in all inclusion targets, then the signature will correspond to homologous sequences in all these targets and these sequences will produce exact matching k-mers with some probability. We start with the probability that two of these homologous bases, X and Y, match is:

$$P(X = Y)_H = (1 - \varepsilon)^2 + (\varepsilon)^2 \cdot P(X_M = Y_M)_H \tag{4}$$

where ε is the probability that two homologous bases do not match exactly, and $P(X_M = Y_M)_H$ is the probability that two homologous bases both mutate to the same base. The default probability of ε is 0.01. We assume that when the homologous bases do not match, the observed base is dependent on the GC-content of the environment. Let λ be the GC-content of the environment. The probability of $P(X_M = Y_M)_H$ is defined as follows:

$$\begin{aligned} P(X_M = Y_M)_H = &\left(2\left(\frac{\lambda}{\lambda+1}\right)^2 + \left(\frac{1-\lambda}{\lambda+1}\right)^2\right)(1-\lambda) \\ &+ \left(2\left(\frac{1-\lambda}{2-\lambda}\right)^2 + \left(\frac{\lambda}{2-\lambda}\right)^2\right)(\lambda) \end{aligned} \tag{5}$$

This probability depends significantly on GC-content of the environment. We assume that the probability of each base matching is independent. Therefore, the probability that two homologous k-mers, k_X and k_Y, match:

$$P(k_X = k_Y)_H = (Pr(X = Y)_H)^k \tag{6}$$

We model the process of homologous k-mer matches with a binomial distribution. If we are observing a true signature region in a reference, we expect that corresponding homologous k-mers exist in all inclusion targets and infer this homology from aggregated k-mer information. An observed reference k-mer will exactly match a corresponding homologous k-mer in another inclusion target with a probability of $p = P(k_X = k_Y)_H$ and not match with a probability of $q = 1 - p$. The expected number of exact k-mer matches with a reference k-mer will be $\mu = (n-1) \cdot p$ and the variance will be $\sigma^2 = (n-1) \cdot p \cdot q$, where n is the number of inclusion targets. We require $n - 1$ because the reference is an inclusion target and its k-mers will exactly match themselves. However, we compensate for this match in our expectation calculation. We assume the

probability of each k-mer match is independent and that k-mer matches are a consequence of homology. When the number of inclusion targets and the probability of homologous k-mers matching are together sufficiently large, the binomial distribution is approximately normal. Let α be our statistical confidence and $\Phi^{-1}(\alpha)$ be the probit function. The minimum number of inclusion targets containing a k-mer, $\wedge_{in}$, required for a reference k-mer to be considered an inclusion k-mer is defined as follows:

$$\wedge_{in} = 1 + \mu - \Phi^{-1}(\alpha)\sigma \tag{7}$$

The $\wedge_{in}$ parameter is automatically calculated unless provided by the user and will inform candidate signature extraction. However, there may be mismatches in the reference which exclude it from the homologous k-mer matching group. We accommodate for this possibility by allowing k-mer gaps in our extraction process. We model the problem of maximum k-mer gap size between exact matching inclusion k-mers as recurrence times of success runs in Bernoulli trials. The mean and variance of the distribution of the recurrence times of k successes in Bernoulli trials is described in Feller 1960 [8]:

$$\mu = \frac{1 - p^k}{q \cdot p^k} \tag{8}$$

$$\sigma^2 = \frac{1}{(q \cdot p^k)^2} - \frac{2k + 1}{q \cdot p^k} - \frac{p}{q^2} \tag{9}$$

This distribution captures how many bases we expect to observe before we see another homologous k-mer match. The probability of a success is defined at the base level as $p = P(X = Y)_H$ and the probability of failure as $q = (1-p)$. This distribution may not be normal for a small number of observations. However, we can use Chebyshev's Inequality to make lower-bound claims about the distribution:

$$P(|X - \mu| \geq \delta\sigma) \leq \frac{1}{\delta^2} \tag{10}$$

where δ is the number of standard deviations, σ, from the mean, μ. Let $P(|X - \mu| \geq \delta\sigma)$ be our stastical confidence, α. The maximum allowable k-mer gap size, $\vee_{gap}$, is calculated as follows:

$$\vee_{gap} = \mu + \sqrt{\frac{1}{1 - \alpha}} \cdot \sigma \tag{11}$$

The $\vee_{gap}$ parameter is automatically calculated unless specified. Candidate signatures are terminated when either no additional inclusion k-mers are located within the maximum gap size, $\vee_{gap}$, or an exclusion k-mer is located. In both cases, the candidate signature ends with the last inclusion k-mer match. The consequence of terminating a signature early is that one true signature may be reported as multiple smaller signatures. We require the minimum signature size, by default, to be four times the size of k. However, for some applications, such as designing assay targets, it may be desirable to use a smaller or larger minimum signature size. Signatures cannot be shorter than k bases. We found that smaller signatures were more likely to overfit the data than larger signatures (data not shown). There is no maximum signature size. As a consequence of Neptune's signature extraction process, signatures may never overlap each other.

2.3 Filtering

The candidate signatures produced will be relatively sensitive, but not necessarily specific, because signature extraction is done using exact k-mer matches. The candidate signatures are guaranteed to contain no more exact matches with any exclusion k-mer than specified by the user. However, there may be inexact matches with exclusion targets. Neptune uses BLAST [1] to locate signatures that align with any exclusion target and, by default, removes any signature that shares 50% identity with any exclusion target aligning to at least 50% of the signature. The remaining signatures are considered filtered signatures and are believed to be sensitive and specific, within the bounds of the relative uniqueness of the inclusion and exclusion groups, and the parameters supplied for target identification.

2.4 Scoring

Signatures are assigned a score corresponding to their highest-scoring BLAST [1] alignments with all inclusion and exclusion targets. This score is the sum of a positive inclusion component and a negative exclusion component, which are analogous to sensitivity and specificity, respectively. Let $|A(S, I_i)|$ be the length of the highest-scoring aligned region between a signature, S, and an inclusion target, I_i. Let $|S|$ be the length of signature S, $PI(S, I_i)$ the percent identity (identities divided by the alignment length) between the aligned region of S and I_i, and $|I|$ be the number inclusion targets. The negative exclusion component is similarly defined. The signature score, $score(S)$, is calculated as follows:

$$\begin{aligned} score(S) = &\sum_{i=0}^{|I|} \frac{|A(S, I_i)| \cdot PI(S, I_i)}{|S||I|} \\ &- \sum_{i=0}^{|E|} \frac{|A(S, E_i)| \cdot PI(S, E_i)}{|S||E|} \end{aligned} \tag{12}$$

This score is maximized when all inclusion targets contain a region exactly matching the entire signature and there exists no exclusion targets that match the signature. Signatures are sorted based on their scores and the best ranking signatures appear first in the output.

2.5 Output

Neptune produces a list of candidate, filtered, and sorted signatures for all references. The candidate signatures are guaranteed to contain, by default, no exact matches with any exclusion k-mer. However, there may still remain potential inexact matches with exclusion targets. The filtered signatures contain no signatures with significant sequence similarity to any exclusion target. Sorted signatures are filtered signatures appearing in descending order of their signature scores.

ID	Length	Summary
1	23,338	O-antigen transport
2	50,038	toxin pilus
3	12,259	phage replication
4	9,652	phage integrase
5	4,282	N-acetylneuraminate lyase
6	10,155	neuraminidase

Table 1: Genomic islands naturally found within *Vibrio cholerae* (NC_012578.1) chromosome I. These islands were used as *in silico* signatures and artificially inserted within a *Bacillus anthracis* genome. These islands were identified with IslandViewer 3 [7].

3 Results

We employ Neptune to identify signatures for several distinct bacterial genomes of differing phyla. In order to validate our method and highlight mathematical considerations, we use Neptune to locate signatures within an artificial data set. Furthermore, we use Neptune to identify signatures within a clinically-relevant *Listeria monocytogenes* data set to demonstrate Neptune's behaviour when operated on clonal isolate populations. Lastly, we employed a clinically-relevant *Escherichia coli* data set to demonstrate Neptune's capacity to locate signatures for a diverse data set.

3.1 Artificial *in silico* Data Set

In order to show that Neptune identifies signatures as expected, the software was run with an artificially created data set. We created an initial inclusion genome by inserting non-overlapping, virulence- and pathogen-associated genes from *Vibrio cholerae* (NC_012578.1) into a *Bacillus anthracis* genome (NC_007530) (Table 1). We selected 6 signature regions varying from 4 to 50-kb in size and spaced these signatures evenly throughout the *B. anthracis* genome. The initial exclusion genome represented a copy of the original (naturally found) *B. anthracis* genome lacking modification. We produced a set of 20 inclusion genomes and 20 exclusion genomes by generating copies of the respective initial genomes in each grouping. These copies each had a nucleotide mutation rate of 1% with all mutations being equally probable.

Neptune was used to identify inserted pathogenic and virulence regions in our artificial *B. anthracis* data set. We specified a k-mer size of 27 and used Neptune's default SNV rate of 1%. The k-mer size was derived from Equation 3, given a genome size of 5337-kb and a GC-content of 0.36. Neptune produced signatures from all 20 inclusion targets (supplementary material). We aligned these signatures to the initial inclusion genome and used GView Server [16] to visualize the identified signatures (Figure 3). Neptune identified 6 complete signatures, corresponding to the expected signature regions, from 11/20 (55%) inclusion targets, 7 signatures from 7/20 (35%) targets, and 8 signatures from 2/20 (10%) targets. Acknowledging the low success rate, the additional signatures corresponded to the pathogenic and virulence regions misreported as two or more adjacent, but smaller, signatures, which is a consequence of mismatches

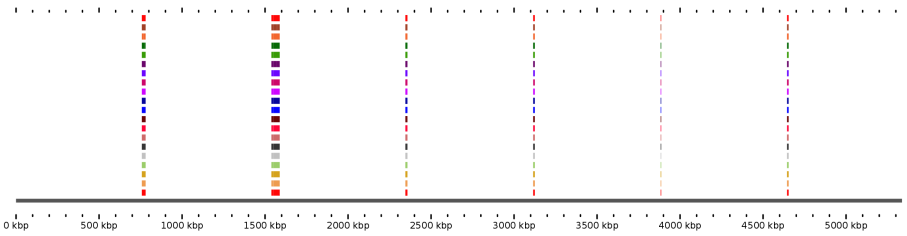

Figure 3: An array of *V. cholerae*-based *in silico* signatures produced using Neptune. All of the artificially inserted *V. cholerae* pathogenic regions were extracted consistently from several artificial *B. anthracis*-*V. cholerae* inclusion group targets against an endogenous *B. anthracis* exclusion group.

introduced into the inclusion sequences. As expected, the break locations for misreported signatures are varied for every reference and, because of its size, the largest (50-kb) *V. cholerae* region contains the majority (7/11) of these locations. However, by Equation 10, we expect to see erroneous breaks with a frequency inversely proportional to our confidence level (95%) in extending signatures over k-mer gaps. This is not a serious issue because these events are relatively rare and all but one of these broken signatures are several thousand nucleotide bases in length. Furthermore, we observed that all identified signatures corresponded to the artificially inserted *V. cholerae* regions and were consistent for all references. Neptune reported all of the *in silico* signatures and reported no false positives. Hence, we conclude that Neptune is able to locate all *in silico* signature regions, though some of these regions are reported as two adjacent signatures.

3.2 *Listeria monocytogenes*

Neptune was next used to locate signature regions within *Listeria monocytogenes* serotypes. *L. monocytogenes* is an opportunistic environmental pathogen that causes listeriosis, a serious and life-threatening disease in humans and animals [15]. Consumption of listeria-contaminated products have caused nationwide outbreaks in the United States and Canada and are a significant concern to the food industry and to public health [6, 12, 13]. *L. monocytogenes* is a clonal organism and recent *L. monocytogenes* evolution has been characterized by deletion events of horizontally acquired bacteriophage and genomic islands. We therefore expect to find signatures corresponding to these events.

We employ a draft genome data set produced by and analyzed for the Canadian Listeria Detection and Surveillance using Next-Generation Genomics (LiDS-NG) project (PRJNA301341). Listeria isolates were serotyped using standard laboratory serotyping procedures [9]. Serotypes 1/2a and 4b were selected for evaluation as they represent distinct bacterial lineages and are clinically relevant [15]. Of the 13 *L. monocytogenes* serotypes, serotypes 1/2a, 1/2b, and 4b are most commonly associated with human illness [15]. *L. monocytogenes* serotype 4b is found within lineage I and is characterized by low diversity and low recombination, whereas serotype 1/2a is found within lineage II and is characterized by high levels of genomic diversity, due to recombination and horizontal gene transfer [15]. In total, 112 serotype 1/2a (inclusion) and 40

ID	Score	Length	Summary
1	0.99	5336	PTS system, L-ascorbate (L-Asc) family
2	0.99	4059	bvrABC locus, β-glucoside-specific sensory system
3	0.99	4830	peptidoglycan-bound protein
4	0.99	5455	PTS system, glucose–glucoside (Glc) family
5	0.98	1943	hypothetical
6	0.98	2839	internalin
7	0.98	4468	two-component response and ABC transport systems
8	0.98	1673	glycosyl-transferase
9	0.97	2567	lineage II specific heat-shock system
10	0.96	968	hypothetical
11	0.95	548	hypothetical

Table 2: A summary of *L. monocytogenes* serotype 1/2a signatures generated by Neptune relative to background serotype 4b genomes. The signatures are ordered by their signature score, which is comprised on a positive inclusion component and a negative exclusion component. We show all signatures with a score ≥ 0.95. As some signatures contain multiple genes, the summary column contains a highlight of the region.

serotype 4b (exclusion) genomes were available. These genomes were randomly divided into two groups of equal size: a training data set and a validation data set.

Neptune was executed on the *L. monocytogenes* training data in order to produce signatures for validation. We specified a k-mer size of 25, derived given a genome size of 3048 kb, the length of the largest isolate in nucleotides, and a GC-content of 0.38, the most extreme GC-content of all our isolates (Equation 3). Neptune produced an average of 1972 (min 1853, max 2056) candidate signatures and 76 (min 56, max 92) filtered signatures from inclusion targets. We further evaluated the top-scoring (≥ 0.95) signatures generated from the inclusion target that generated the greatest number of filtered signatures. The signatures produced from this target were aligned against a *L. monocytogenes* 1/2a strain 08-5578 genome (PRJNA43671). The top-scoring signatures (≥ 0.95) identified for *L. monocytogenes* serotype 1/2a are listed in Table 2. These signatures included phosphoenolpyruvate (PEP)-dependent phosphotransferase systems (PTS) belonging to L-ascorbate (PTSAsc) and glucose–glucoside (PTS-Glc) families [21], and a 4468 bp locus containing a two-component response regulation system and an ABC transport system [5]. The presence of a variety of PTS systems and transport systems provides *L. monocytogenes* serotype 1/2a with a competitive advantage to survive under different environmental conditions due to its ability to utilize a variety of compounds. A bvrABC locus was found among these signatures which is known to be involved in environmental regulation of virulence genes [4]. An internalin protein was also found, which is known to be a critical factor for pathogenesis [3]. Also, a lineage II-specific heat-shock system [22] constituting an operon with 3 genes, RNA polymerase factor sigma C, lstR thermal regulator, and a cell division related protein, was present among those high scoring signatures. Other signatures included sequences coding for peptidoglycan-bound protein, glycosyl-transferase, and hypothetical proteins.

These training-generated signatures were compared against the validation

data set to evaluate their *in silico* sensitivity and specificity. We used BLAST [1] to align the signatures against our validation data set. The complete alignment output can be found in supplementary material. With a percent identity threshold of 95% and a minimum alignment length of 95% the size of the signature length, 614 out of 616 (99.7%) signature-validation alignments met our sensitivity criteria. The 2 alignments that did not meet this criteria corresponded to signature ID #3 (Table 2) producing broken alignments against distinct validation targets. However, these alignments were all greater than 1 kb in length and over 99% sequence identity. Similarly, with a percent identity threshold of 50% and a minimum alignment length of 50% the size of the signature length, we found no significant hits against any of the serotype 4b validation targets. This suggests that our top-scoring Neptune-generated *L. monocytogenes* serotype 1/2a signatures are highly sensitive and specific to 1/2a against a *L. monocytogenes* serotype 4b background.

3.3 *Escherichia coli*

In an attempt to model a real application of signature discovery, we employ Neptune to locate signatures corresponding to shiga-toxin producing *Escherichia coli* (STEC). The shiga toxin requires both the *stx1a* and *stx1b* subunits to be functional. Therefore, we expect to locate these subunits using Neptune. As *E. coli* exhibits increased genomic diversity over *L. monocytogenes*, it makes differentiating lineages within the species a more challenging signature discovery problem.

The inclusion and exclusion data sets comprised of 6 STEC (stx1) and 11 non-STEC draft assemblies, respectively. Neptune was run with a k-mer size of 25 (Equation 3; see supplementary materials), and produced an average of 558 (min 429, max 703) candidate signatures and 202 (min 177, max 245) filtered signatures from inclusion targets. The top-scoring signature produced from each target had 100% sensitivity and at least 98% specificity. We further evaluated the top-scoring ($\geq$ 0.95) signatures generated from the target that generated the greatest number of filtered signatures (245) (Table 3). We aligned these signatures against an *E. coli* O157:H7 str. Sakai reference (NC_002695.1) to infer sequence annotations. This alignment included the chromosome and both plasmids. The *E. coli* O157:H7 str. Sakai reference was selected because it contains a copy of the Shiga toxin and is well characterized. A summary of the Neptune-identified, stx1-containing *E. coli* signatures regions is located in Table 3.

Neptune identified top-scoring signature regions corresponding to known *E. coli* virulence and pathogenic elements. These included shiga toxin I subunits *stx1a* and *stx1b* (1), an integrase (4), and hemolysin (5). Additionally, Neptune identified signatures corresponding to pathogenic elements, including a tail protein (3) and a colonization factor (6). Furthermore, using BLAST [1], we found that many of the top-scoring signatures aligned to known *E. coli* O157:H7 O-Islands. This included signatures 1, 3, 4, 6, 7, 8, 10; notably shiga toxin I, a tail protein, and an integrase. The hemolysin-predicted signature was the only top-scoring signature located on one of the *E. coli* O157:H7 str. Sakai plasmids. We located this region on the pO157 plasmid. We conclude that Neptune is effective at locating known pathogenic regions within STEC with high sensitivity and high specificity.

ID	Score	Inclusion	Exclusion	Length	Summary
1	1.00	1.00	0.00	1375	shiga toxin I
2	0.98	1.00	0.01	438	intimin regulator
3	0.98	1.00	0.02	1223	bacteriophage element
4	0.98	0.99	0.02	3293	bacteriophage integrase
5	0.97	0.99	0.02	7778	hemolysin
6	0.96	1.00	0.03	1260	colonization factor
7	0.96	1.00	0.04	474	hypothetical
8	0.96	0.98	0.02	1161	membrane protein
9	0.96	0.99	0.03	193	hypothetical
10	0.96	1.00	0.04	956	bacteriophage elements

Table 3: A summary of stx1-containing *E. coli* signatures generated by Neptune relative to background non-toxin *E. coli*. The signatures are ordered by their signature score, which is comprised on a positive inclusion component and a negative exclusion component. We show all signatures with a score ≥ 0.95. As some signatures contain multiple genes, the summary column contains a highlight of the region.

4 Discussion

4.1 Parameters

While many of Neptune's parameters are automatically calculated, there are a few parameters that deserve special mention. We recommend odd-sized k-mers to avoid the possibility of a k-mer being the reverse complement of itself. The minimum number of inclusion hits and maximum gap size are sensitive to the SNV rate and the size of k. When estimating these parameters, a slightly higher than expected SNV rate is recommended. This overestimation will avoid false negatives at the expense of false positives. However, many of these false positives will be removed during the filtering stage.

4.2 Memory and Computation Time

Neptune is highly parallelizable and performs well on high-performance computing clusters. When identifying signatures within a data set of 122 *L. monocytogenes* inclusion genomes of approximately 3,000 kb in length and 40 related *L. monocytogenes* exclusion genomes, Neptune required 27 minutes on a 40-node computing cluster. The memory requirements of all processes never exceeded more than 10G. Nepune benefits significantly from parallelization and will run much slower in a single-CPU environment.

4.3 Limitations

Neptune's signature extraction step avoids false negatives at the expense of false positives. The software attempts to locate signatures that may not contain an abundance of exact matches. This approach produces some false positives. However, false positives are removed during signature filtering and requires increased computational time. As signatures are extracted from a reference, repeated regions do not confound signature discovery. However, if a repeated region is a

true signature, then Neptune will report each region as a separate signature. In this circumstance, user curation may be required.

Neptune cannot locate isolated SNVs and small mutations. Any region with a high degree of similarity to the exclusion group will either not produce candidate signatures or be removed during filtering. Neptune is designed to locate general-purpose signatures of arbitrary size and does not consider application-specific physical and chemical properties of signatures. Furthermore, Neptune is not capable of selecting the best substring within a signature region. This operation would have the effect of optimizing signature efficacy for applications where smaller signature lengths are desirable. While Neptune is capable of producing signatures as small as the k-mer size, we observed that very short signatures (approximately < 100 bases) tend to overfit the targets from which the signatures are derived. We do not recommend identifying signatures of this size unless application-specific.

Finally, Neptune makes assumptions about the probabilistic independence of bases and SNV events; while these events do not occur indpendently in nature, they allow for significant mathematical simplification. Nonetheless, Neptune is capable of producing highly sensitive and specific signatures using these assumptions.

5 Conclusion

We show that Neptune is capable of locating signatures in an artificial data set. While some signatures are reported as two smaller, adjacent signatures, Neptune reports all the expected signature regions. We apply Neptune to a *L. monocytogenes* data set and show that top-scoring Neptune-identified signatures have high *in silico* sensitivity and specificity to a wet-lab verified validation data set. Finally, we employ Neptune to locate pathogen-associated signatures related to STEC. Neptune locates many expected signature regions with high confidence. As expected, no top-scoring signatures corresponded to rDNA or housekeeping genes. The signatures found in groups of pathogenic bacteria can also provide an array of gene candidates to further investigate their possible role in their pathogenesis. We conclude that Neptune is a powerful and flexible tool for locating signature regions with minimal prior knowledge.

6 Availability

The data used in the manuscript is stored under the PRJNA301341 NCBI accession. Neptune is developed in Python using DRMAA, NumPy, SciPy, and Biopython libraries. The software requires a standard 64-bit Linux environment. The software is available at: *http://github.com/phac-nml/neptune*

7 Acknowledgements

The authors would like to thank Franklin Bristow and Eric Enns of the NML-PHAC for their feedback on various aspects of the software design and implementation. The authors also thank Rahat Zaheer of the NML-PHAC for providing valuable insights into the significance of identified signatures.

Contrib.	Credit	Contributions
EM	Lead	manuscript; math; software; experiments
MG	Co	manuscript; exp, software design; resources
CB	Co	list exp design, analysis; list, ecoli data
KW	Co	manuscript; experiments design
MD	Co	math
LC	Co	listeria data
PM	Co	background work
NK	Co	background work
AR	Co	listeria experiment design
GVD*	Anchor	manuscript; exp, software design; resources
RZ	Ack	manuscript, listeria analysis
FB	Ack	software discussions
EE	Ack	software discussions

Table 4: Author contributions. * = Corresponding

References

[1] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

[2] Kai Christian Bader, Christian Grothoff, and Harald Meier. Comprehensive and relaxed search for oligonucleotide signatures in hierarchically clustered sequence datasets. *Bioinformatics*, 27(11):1546–1554, 2011.

[3] H Bierne, C Sabet, N Personnic, and P Cossart. Internalins: a complex family of leucine-rich repeat-containing proteins in listeria monocytogenes. *Microbes and Infection*, 9(10):1156–1166, 2007.

[4] Klaus Brehm, María-Teresa Ripio, Jürgen Kreft, and José-Antonio Vázquez-Boland. The bvr locus of listeria monocytogenes mediates virulence gene repression by β-glucosides. *Journal of bacteriology*, 181(16):5024–5032, 1999.

[5] Ron Caspi, Tomer Altman, Richard Billington, Kate Dreher, Hartmut Foerster, Carol A Fulcher, Timothy A Holland, Ingrid M Keseler, Anamika Kothari, Aya Kubo, et al. The metacyc database of metabolic pathways and enzymes and the biocyc collection of pathway/genome databases. *Nucleic acids research*, 42(D1):D459–D471, 2014.

[6] Andrea Currie, Jeffrey M Farber, Céline Nadon, Davendra Sharma, Yvonne Whitfield, Colette Gaulin, Eleni Galanis, Sadjia Bekal, James Flint, Lorelee Tschetter, et al. Multi-province listeriosis outbreak linked to contaminated deli meat consumed primarily in institutional settings, canada, 2008. *Foodborne pathogens and disease*, 12(8):645–652, 2015.

[7] Bhavjinder K Dhillon, Matthew R Laird, Julie A Shay, Geoffrey L Winsor, Raymond Lo, Fazmin Nizam, Sheldon K Pereira, Nicholas Waglechner, Andrew G McArthur, Morgan GI Langille, et al. Islandviewer 3: more

flexible, interactive genomic island discovery, visualization and analysis. *Nucleic acids research*, page gkv401, 2015.

[8] Vilim Feller. *An Introduction to Probability Theory and Its Applications: Volume 1*. J. Wiley & sons, 1960.

[9] Matthew W Gilmour, Morag Graham, Gary Van Domselaar, Shaun Tyler, Heather Kent, Keri M Trout-Yakel, Oscar Larios, Vanessa Allen, Barbara Lee, and Celine Nadon. High-throughput genome sequencing of two listeria monocytogenes clinical isolates during a large foodborne outbreak. *BMC genomics*, 11(1):120, 2010.

[10] Stefan Kurtz, Adam Phillippy, Arthur L Delcher, Michael Smoot, Martin Shumway, Corina Antonescu, and Steven L Salzberg. Versatile and open software for comparing large genomes. *Genome biology*, 5(2):R12, 2004.

[11] Hsiao Ping Lee and Tzu-Fang Sheu. An algorithm of discovering signatures from dna databases on a computer cluster. *BMC bioinformatics*, 15(1):339, 2014.

[12] Michael J Linnan, Laurene Mascola, Xiao Dong Lou, Veronique Goulet, Susana May, Carol Salminen, David W Hird, M Lynn Yonekura, Peggy Hayes, Robert Weaver, et al. Epidemic listeriosis associated with mexican-style cheese. *New England Journal of Medicine*, 319(13):823–828, 1988.

[13] Jeffrey T McCollum, Alicia B Cronquist, Benjamin J Silk, Kelly A Jackson, Katherine A O'Connor, Shaun Cosgrove, Joe P Gossack, Susan S Parachini, Neena S Jain, Paul Ettestad, et al. Multistate outbreak of listeriosis associated with cantaloupe. *New England Journal of Medicine*, 369(10):944–953, 2013.

[14] Pall Melsted and Jonathan K Pritchard. Efficient counting of k-mers in dna sequences using a bloom filter. *BMC bioinformatics*, 12(1):333, 2011.

[15] Renato H Orsi, Henk C den Bakker, and Martin Wiedmann. Listeria monocytogenes lineages: Genomics, evolution, ecology, and phenotypic characteristics. *International Journal of Medical Microbiology*, 301(2):79–96, 2011.

[16] Aaron Petkau, Matthew Stuart-Edwards, Paul Stothard, and Gary Van Domselaar. Interactive microbial genome visualization with gview. *Bioinformatics*, 26(24):3125–3126, 2010.

[17] Adam M Phillippy, Kunmi Ayanbule, Nathan J Edwards, and Steven L Salzberg. Insignia: a dna signature search web server for diagnostic assay development. *Nucleic acids research*, page gkp286, 2009.

[18] Ravi Vijaya Satya, Kamal Kumar, Nela Zavaljevski, and Jaques Reifman. A high-throughput pipeline for the design of real-time pcr signatures. *BMC bioinformatics*, 11(1):340, 2010.

[19] Ravi Vijaya Satya, Nela Zavaljevski, Kamal Kumar, and Jaques Reifman. A high-throughput pipeline for designing microarray-based pathogen diagnostic assays. *Bmc Bioinformatics*, 9(1):185, 2008.

[20] Tom Slezak, Tom Kuczmarski, Linda Ott, Clinton Torres, Dan Medeiros, Jason Smith, Brian Truitt, Nisha Mulakken, Marisa Lam, Elizabeth Vitalis, et al. Comparative genomics tools applied to bioterrorism defence. *Briefings in Bioinformatics*, 4(2):133–149, 2003.

[21] Regina Stoll and Werner Goebel. The major pep-phosphotransferase systems (ptss) for glucose, mannose and cellobiose of listeria monocytogenes, and their significance for extra-and intracellular growth. *Microbiology*, 156(4):1069–1083, 2010.

[22] Chaomei Zhang, Joe Nietfeldt, Min Zhang, and Andrew K Benson. Functional consequences of genome evolution in listeria monocytogenes: the lmo0423 and lmo0422 genes encode σc and lstr, a lineage ii-specific heat shock system. *Journal of bacteriology*, 187(21):7243–7253, 2005.