IslandHunter – A Java-based GI detection software

Shakuntala Baichoo, Haswanee Goodur, Vyasanand Ramtohul

Over the past decade, researchers have discovered that apart from the essential genes, bacterial genomes also contain a variable amount of accessory genes acquired by horizontal gene transfer (HGT) that are categorized as genomic islands (GIs). GIs encode adaptive traits, which might be beneficial for the species under certain growth or environmental conditions. It has always been a challenge for biologists to identify GIs within a bacterial genome as they evolve very rapidly. This paper proposes a standalone software, IslanHunter, that has been developed using Java and BioJava and can extract GI regions using GC content, codon usage bias, dinucleotide frequency bias, tetranucleotide frequency bias, k-mer signature analysis (2-mer, 3-mer, 4-mer, 5-mer, and 6-mer) and presence of mobility genes. IslandHunter provides a simple graphical user interface where disclosed GIs are displayed in a tree-view and a circular graph. Users are presented with options to save the GI regions as blocks of DNA sequences in FASTA format. They can later use these predicted GI regions for further analysis. IslandHunter can take as input, files in GenBank, EMBL or FASTA formats. IslandHunter provides flexible display options and save options. The software has been evaluated against exiting tools with good performance. It is available for evaluation at https://github.com/ShakunBaichoo/IslandHunter .

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.716v1 | CC-BY 4.0 Open Access | rec: 18 Dec 2014, publ: 18 Dec 2014

IslandHunter – A Java-based GI Detection Software

Authors

Shakuntala Baichoo
shakunb@uom.ac.mu
University of Mauritius

Haswanee Goodur
hashgood1307@gmail.com
University of Mauritius

Vyasanand Ramtohul
vyas.ramtohul@gmail.com
University of Mauritius

Corresponding author: Shakuntala Baichoo
University of Mauritius
shakunb@uom.ac.mu

Abstract

Background: Over the past decade, researchers have discovered that apart from the essential genes, bacterial genomes also contain a variable amount of accessory genes acquired by horizontal gene transfer (HGT) that are categorized as genomic islands (GIs). GIs encode adaptive traits, which might be beneficial for the species under certain growth or environmental conditions. It has always been a challenge for biologists to identify GIs within a bacterial genome as they evolve very rapidly.
Results: This paper proposes a standalone software, IslanHunter, that has been developed using Java and BioJava and can extract GI regions using GC content, codon usage bias, dinucleotide frequency bias, tetranucleotide frequency bias, k-mer signature analysis (2-mer, 3-mer, 4-mer, 5-mer, and 6-mer) and presence of mobility genes. IslandHunter provides a simple graphical user interface where disclosed GIs are displayed in a tree-view and a circular graph. Users are presented with options to save the GI regions as blocks of DNA sequences in FASTA format. They can later use these predicted GI regions for further analysis.
Conclusion: IslandHunter can take as input, files in GenBank, Embl or FASTA formats. IslandHunter provides flexible display options and save options. The software has been evaluated against exiting tools with good performance. It is available for evaluation at https://github.com/ShakunBaichoo/IslandHunter.

Keywords - *bioinformatics; genomic island; comparative genomics; prokaryotic organism; GI detection; horizontal gene transfer*

1.0 Introduction

Over the past decade, researchers have discovered that bacterial genomes contain a number of essential genes as well as a variable amount of accessory genes acquired by HGTs that encode adaptive traits, which might be beneficial for the species under certain growth or environmental conditions (Juhas et al., 2009). This has led to new challenges in the medical as well as the agricultural sector and thus the analysis of bacterial genomes and HGTs has become a major research area in the bioinformatics field.

Identifying horizontally-transferred genes remains a challenging task despite a number of works done in this area in the last decade, mainly because of the large spectrum of variability found in the compositional properties of both native and acquired genes (Azad et al., 2011).

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.716v1 | CC-BY 4.0 Open Access | rec: 18 Dec 2014, publ: 18 Dec 2014

One of the emerging ideas is that GIs cover an overarching family of elements, including mobile genetic elements (MGEs) such as integrative and conjugative elements (ICEs), conjugative transposons and some prophages.

GIs have many specific features. They are often inserted at tRNA genes and are flanked by 16-20 bp perfect direct repeats (DR). They contain mobility genes such as integrases and transposases and unusual guanine and Cytosine (% G+C) content. They are normally large (10-200kb) with small genomic islets (<10kb). Moreover, GIs may be predicted by nucleotide statistics that generally differ from the rest of the genome.

Using these specific features, GI regions can be predicted effectively. The most common GI identification methods are the discrepancies in composition of sequences between the GI and the host DNA, including codon usage, Guanine-Cytosine (GC) content, k-mer signature analysis and the frequency of specific di-nucleotides and tetra-nucleotides.

In this paper, we describe IslandHunter, an application to extract GIs using a combination of methods, and developed using Java and BioJava.

2.0 Background

In order to extract probable genomic islands, a number of genic and sequence-based methods have been implemented in IslandHunter and are described in the sections below:

2.1 Guanine-Cytosine Content Variation

The guanine-cytosine content (GC-content) is the percentage of guanine or cytosine nitrogenous bases in DNA or RNA molecule. GC pairs are bound by three hydrogen bonds while AT pairs are bound by two only. For that reason, DNA with high GC-content is more stable.
GC-content is usually referred to as a percentage value but can sometime be represented as a ratio (G+C ratio). The formula for calculating GC-content are given below (Hurst et al., 2001):

$$\%GC = \frac{G + C}{A + T + G + C} * 100$$

Percentage GC-Content

$$GC - ratio = \frac{A + T}{C + G}$$

Ratio of GC-Content

Bacterial genomes vary largely in their GC content. However, the GC content is normally fairly balanced across a given genome and this is an important method for characterizing species. In addition, GC-content is highly affected by the environment; for example, they vary significantly, for example 34% in sea to 61% in soil (Hildebrand et al., 2010), thus dissimilar genomes will have different GC-contents.
The GC-content in genomic islands often diverges from that of the rest of the genome, thus indicating a potential horizontal transfer of sequence from another organism.

2.2 Codon Usage Bias

A codon is a triplet of nucleotides that encodes for an amino acid. There are four nucleotides namely Adenine (A), Cytosine (C), Guanine (G), and Thymine (T)/Uracil (U), which give a total of 64 possible codon combinations. Each codon codes for one specific amino acid, but there can be several codons coding for the identic amino acid, for instance, GCT/U, GCC, GCA and GCG would all code for Alanine, as shown in figure 1.

The codon usage or codon preference of an organism is a statistical property of DNA sequences such that each individual genome uses a preferred set of codons. Hence, using this codon preference idea, genes that are foreign (GI) to an organism, can be identified through their diversion in codon usage from the whole genome.

The Relative Synonymous Codon Usage (RSCU) values are the number of times a particular codon is observed, relative to the number of times that the codon would be observed in the absence of any codon usage bias. The RSCU value for each amino acid is used to observe the affinity for a definite codon since distinct organisms have unusual affinity to different tRNA. The 'relative adaptedness' value, ***W*** W_i, of a specific codon is calculated as:

$$W_i = \frac{RSCU_i}{RSCU_{MAX}}$$

where $\boldsymbol{RSCU_i}$ refers to the frequency of a codon ***i*** in the subset of highly expressed genes and $\boldsymbol{RSCU_{MAX}}$ represents the frequency of the codon that is most often used to code for the relevant amino acid in the subset of highly expressed genes.

The Codon Affinity Index (CAI) for a gene (Carbone et al., 2003) is then defined as the geometric mean of $\boldsymbol{W_i}$ values for codons in that gene. Genes with low ***CAI*** value can probably be a GI gene where ***CAI*** is computed as:

$$CAI = \left(\prod_{i=1}^{L} W_i\right)^{1/L} \quad o\square \;\; exp\left(\frac{1}{L}\sum_{i=1}^{L} ln(W_i)\right)$$

where ***L*** represents the number of codons in the gene excluding the start codon (methionine), tryptophan and the stop codons (Moriyama et al., 2006).

2.3 Dinucleotide Frequency Bias

Dinucleotide bias (Karlin et al., 1995) can be used to describe a genome's signature such that if the percentage of a dinucleotide ***XY*** in an entire genome is ***Z***, then a subset of this genome should also have around the same percentage composition of this dinucleotide except for a gene that has come from another genome, will have dinucleotide composition similar to its source genome rather than the one it is currently in

Dinucleotide bias (Karlin's dinucleotide) is assessed through an odds ratio comparing the frequencies of each gene's dinucleotides to the genomic averages (Karlin et al., 1995). If the deviation exceeds an established threshold, the gene is considered to be amply atypical and thus classified as alien. Using this information it is possible to detect genome segments, which are foreign (GI).

Dinucleotide bias which refers to the dinucleotide relative abundance values of a DNA sequence are measured through the odds ratios

$$\rho_{xy} = \frac{f_{XY}}{f_X f_Y}$$

where f_{XY} is the frequency of a dinucleotide in a region and f_X and f_Y are the frequencies of the mononucleotides in the dimer.

For double-stranded DNA sequences, the frequencies of inverted complementary strands of the DNA sequence region are also calculated as ρ^*_{XY} in order to compensate for any asymmetry.

In 2001, Karlin (Karlin et al., 2001) also reported that a helpful way of calculating the differences between the relative abundance value (σ(f,g)) for a given region and the value of the whole genome is through:

$$\sigma(f,g) = \frac{1}{16}\sum_{XY} |P_{XY}(f) - P_{XY}(g)|$$

where ***f*** would be the query region, ***g*** would be the whole genome sequence and the sum extends over all dinucleotides.

In order to identify GIs, the dinucleotide frequency bias of the specific specific regions must show a clear departure from the dinucleotide frequency bias of whole genome.

2.4 Tetranucleotide Frequency Bias

In the same way as dinucleotides, tetranucleotide (pride et al 2003) can also be used to identify horizontally transferred genes. The tetranucleotide usage deviation (TUD) of a the word F(W) for each tetranucleotide combination is calculated as the ratio of the observed frequency O(W) to the expected frequency E(W) in a given window of length N, and is calculated as:

$$F(W_i) = \frac{O(W_i)}{E(W_i)}$$

where ***O(W$_i$)*** is the observed occurrence value, and ***E(W$_i$)*** is the expected occurrence value of a tetranucleotide ***W$_i$*.**

whereby the ***E(W$_i$)*** value is calculated by:

$$E(W = w_1w_2w_3w_4) = f(w_1)f(w_2)f(w_3)|S|$$

where ***W$_i$*** is the ***ith*** nucleotide (of the set A,C,G,T) of ***W***; ***f(A), f(C), f(G),*** and ***f(T)*** represent each nucleotide frequency for the sequence ***S*** and ***/S/*** is the length of the sequence.

Or it can also be calculated as:

$$E(W = w_1w_2w_3w_4) = \frac{O(w_1w_2w_3) * O(w_2w_3w_4)}{O(w_2w_3)}$$

In order to identify GIs, the divergence between the observed and expected tetranucleotide frequency is calculated using the z-score[1] approximation, as follows:

$$Z(W = w_1w_2w_3w_4) = \frac{O(w_1w_2w_3w_4) - E(w_1w_2w_3w_4)}{\sqrt{varO(w_1w_2w_3w_4)}}$$

where the ***varO(W)*** can be approximated as follows:

$$varO(W) = E(W)\frac{|O(w_2w_3) - O(w_1w_2w_3)||O(w_2w_3) - O(w_2w_3\square_4)|}{O(w_2w_3)^2}$$

The Pearson correlation coefficient (r) (Rodgers et al., 1988) for the z-scores is used to determine whether two genomic sequences exhibit a similar pattern for over- or under-represented tetranucleotides. It is defined as follows:

$$r = \frac{\sum Z_xZ_y}{N}$$

Genomic fragments with similar patterns are determined by a high correlation coefficient while distinct patterns are the one with low correlation coefficients (Bolshoy et al., 2010). Therefore, it is obvious that the dissimilar patterns are alien to the genome being analyzed, thus could be probably GIs.

2.5 Presence of Mobility Genes

During HGT, MGEs (mobile genetic elements) such as integrase and transposase genes are acquired (Langille, 2009) along with some virulence factor genes (Ho Sui et al., 2009). These cluster of genes are probably of horizontal transfer origin and may be identified using Annotation in EMBL and GenBank annotation records, thus, helping in disclosing possible GIs.

2.6 K-mer Signature Analysis

K-mer mostly refers to a specific n-tuple or n-gram for nucleic acid or amino acid sequences, which are used to identify certain regions within biomolecules such as DNA or proteins respectively. K-mer analysis is commonly used to predict biological meaningful clusters of DNA words (k-mers) and genomic entities. “Genome entities as diverse as genes, CpG dinucleotides, transcription factor binding sites (TFBSs) or ultra-conserved non-coding regions usually form clusters along the chromosome sequence" (Hackenberg et al., 2011).

K-mer analysis algorithm detects the distance between a cluster of words in DNA sequence and neighbouring DNA sequences. Benjamin (Benjamin et al., 2009) stated that k-mer frequency analysis has been used to identify lateral gene transfer and since k-mer frequency signatures are generally distinct across distinctive species, the frequency signatures of segments of a sequence can be compared with the signature of whole genome of the organism. If these are significantly different, they may be probable GIs.

[1] A z score is defined as the deviate score (the observed score minus the mean) divided by the standard deviation

To calculate the distance between a portion of the genome and the whole genome sequence, the Euclidean distance algorithm is used. The formula is given as:

$$d(p,q) = d(q,p)$$

such that

$$d(q,p) = \sqrt{((q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2)}$$

such that

$$\sqrt{((q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2)} = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

where ***p*** is the array of k-mer frequency signatures and ***q*** is the k-mer frequency signature of the whole genome.

3.0 Implementation

An Embl, GenBank file or FASTA file is loaded and parsed using BioJava. The GI detection algorithms discussed above are implemented as explained below.

3.1. GC Content Variation

The GC content method to identify GIs is implemented using the steps described in flowchart, figure 1.

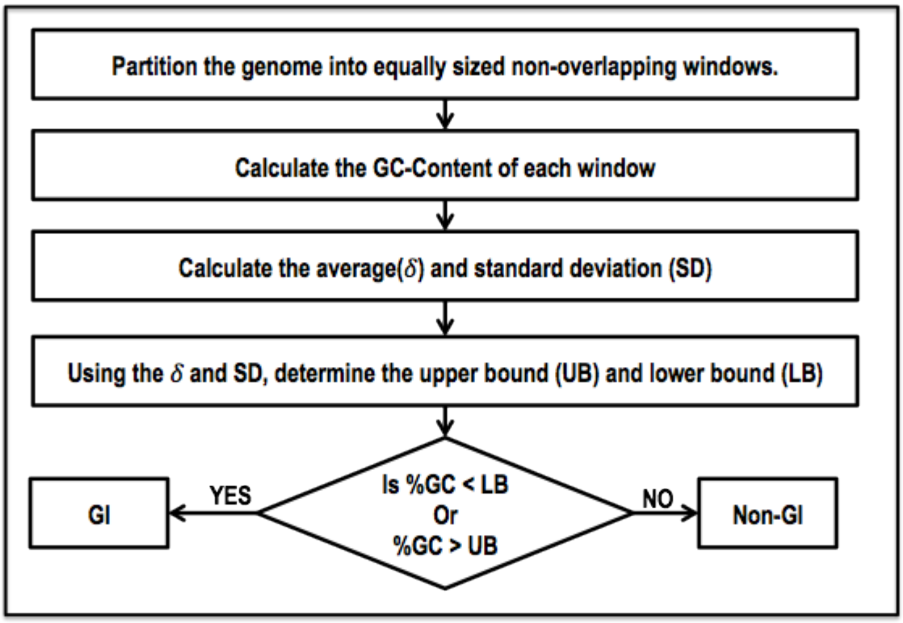

Partition the genome into equally sized non-overlapping windows.
Calculate the GC-Content of each window
Calculate the average(δ) and standard deviation (SD)
Using the δ and SD, determine the upper bound (UB) and lower bound (LB)
Is %GC < LB Or %GC > UB
YES
NO
GI
Non-GI

Figure 1 - Flowchart illustration of %GC-Content

Note: Lower bound (LB) = %GC of whole genome – (SD*n)
Upper bound (UB) = %GC of whole genome + (SD*n)

Where n is a sensitivity factor (typical value is 1.5)

3.2. Codon Usage Bias

The codon usage bias of the coding sequences (based on the annotations) is determined as per the flowchart in figure 2.

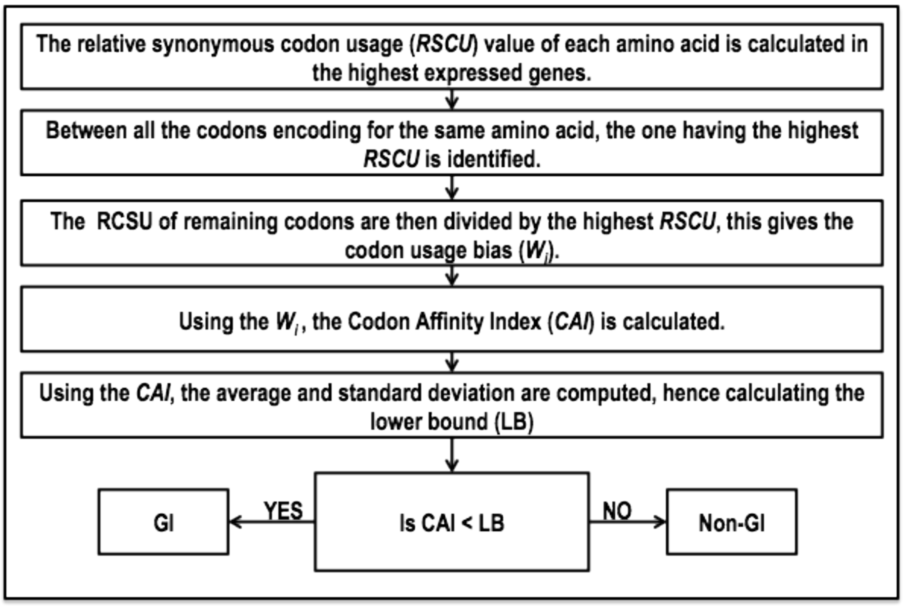

The relative synonymous codon usage (*RSCU*) value of each amino acid is calculated in the highest expressed genes.
Between all the codons encoding for the same amino acid, the one having the highest *RSCU* is identified.
The RCSU of remaining codons are then divided by the highest *RSCU*, this gives the codon usage bias (*W_i*).
Using the *W_i*, the Codon Affinity Index (*CAI*) is calculated.
Using the *CAI*, the average and standard deviation are computed, hence calculating the lower bound (LB)
GI
YES
Is CAI < LB
NO
Non-GI

Figure 2 - Flowchart illustration of the Codon Usage Bias module

Note: Lower bound = Average CAI – (SD x n)

Where n is a sensitivity factor (e.g. 1.5)

For example:	**Codons**	**RSCU$_i$**
All codons coding for alanine	GCU	6035
	GCC	113980
	GCA	9757
	GCG	86501

RSCU$_{max}$= 113980 (GCC)

The codon usage bias is as follows:

W_{GCU} = 6035/113980 = 0.0529...

W_{GCC} = 113980/113980 = 1.0 (Highest preference)

W_{GCA} = 9757/113980 = 0.085...

W_{GCG} = 86501/113980 = 0.758...

The ***W_i*** is then used to calculate the Codon Affinity Index (***CAI***) (Carbone et al., 2003) of each gene.

Example: To calculate the ***CAI*** for this sequence:

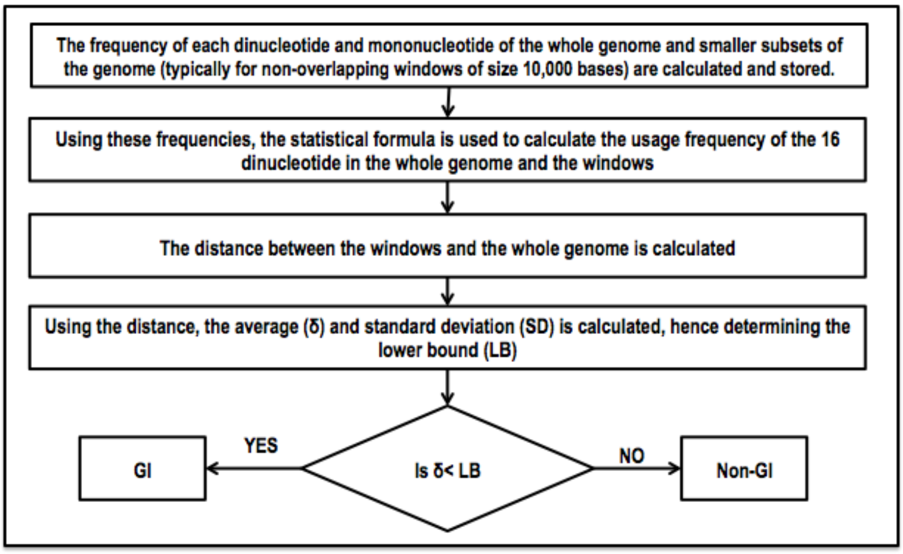
	AUG	UUG	GCC	GAC	UAG
W_i	1.000	0.641	0.213	1.000	0.930

In the above sequence AUG is the start codon and UAG is the stop codon. They are therefore not used when calculating the ***CAI***

$$CAI = (0.641 \times 0.213 \times 1.0)^{1/3} = 0.5149...$$

3.3. DinucleotideFrequency Bias

The dinucleotide frequency bias is implemented based on Karlin's method discussed in section 2.3 and using the steps shown in figure 3.



The frequency of each dinucleotide and mononucleotide of the whole genome and smaller subsets of the genome (typically for non-overlapping windows of size 10,000 bases) are calculated and stored.
Using these frequencies, the statistical formula is used to calculate the usage frequency of the 16 dinucleotide in the whole genome and the windows
The distance between the windows and the whole genome is calculated
Using the distance, the average (δ) and standard deviation (SD) is calculated, hence determining the lower bound (LB)
GI
YES
Is δ< LB
NO
Non-GI

Figure 3 - Flowchart illustration of Dinucleotide frequency bias

Note: Lower bound = Average (δ) - (SD x n)
Where n is a sensitivity factor (typically 2.0)

3.4. Tetranucleotide Frequency Bias

To calculate the tetranucleotide bias, the method of Pride et al. discussed in section 2.4, through the steps shown in figure 4.

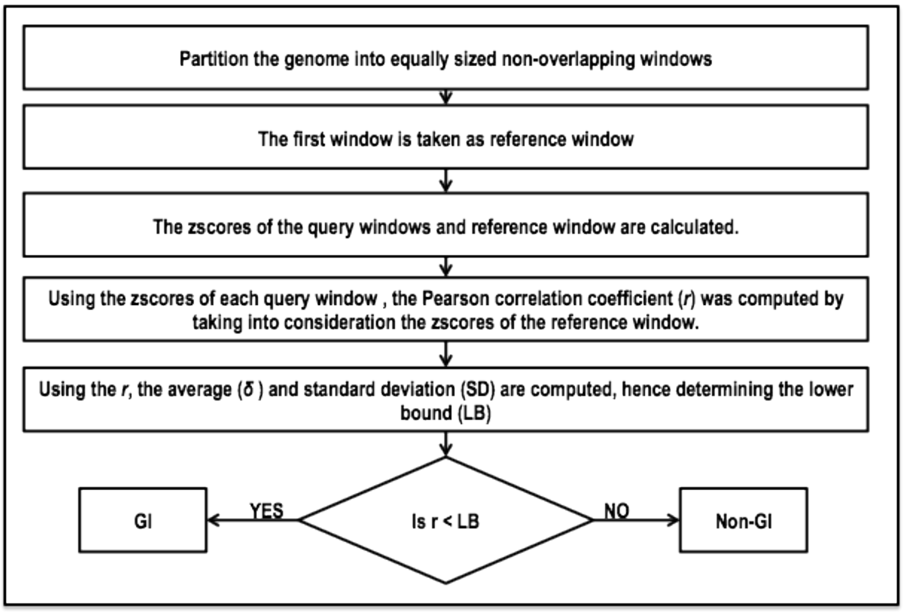

Partition the genome into equally sized non-overlapping windows
The first window is taken as reference window
The zscores of the query windows and reference window are calculated.
Using the zscores of each query window , the Pearson correlation coefficient (r) was computed by taking into consideration the zscores of the reference window.
Using the r, the average (δ) and standard deviation (SD) are computed, hence determining the lower bound (LB)
Is r < LB
YES
GI
NO
Non-GI

Figure 4 - Flowchart illustration of tetranucleotide frequency bias

Note: Lower bound = Average (δ) - (SD x n)

Where n is a sensitivity factor (typically 1.5)

3.5. Presence of Mobility Genes

The mobility genes are identified by reading the annotations, more specifically the product of those annotations which are compared to a list of predefined mobile genes (e.g. integrase, transposase etc.).

For example:

```
FT          CDS complement(704071..705234)
FT          /codon_start=1
FT          /transl_table=11
FT          /locus_tag="P9303_07651"
FT          /product="Phage integrase family protein"
FT          /db_xref="EnsemblGenomes-Gn:P9303_07651"
FT          /db_xref="EnsemblGenomes-Tr:ABM77516"
FT          /db_xref="GOA:A2C7Q6"
FT          /db_xref="InterPro:IPR002104"
FT          /db_xref="InterPro:IPR011010"
FT          /db_xref="InterPro:IPR013762"
FT          /db_xref="UniProtKB/TrEMBL:A2C7Q6"
FT          /protein_id="ABM77516.1"
FT          /translation="MELSNELININRALADSGINLRIEQRGQWLNLRGALPCRNGTGLI
FT          ..."
```

3.6. K-mer Signature Analysis

The whole genome sequence is partitioned using a given window-size and step-size into non-overlapping bins. All the possible k-mers are identified.

For 2-mers, there are 16 possible combinations. The complements of all the 16 possible combinations are identified and removed.

For example:

AA	AC	AT	AG	CA	CC	CG	CT

GA	GC	GG	GT	TA	TC	TG	TT

But, k-mers whose complements are their mirrors (for example, AT is the mirror of TA), are not removed. Therefore, for a 2-mer, 10 combinations are chosen out of the 16. This is done by comparing the k-mer with its reverse complement, and if it is true, the reverse complement is discarded.

For each of the bins and the whole genome, the frequencies of the k-mers are calculated and these frequencies are added together and each of the frequencies is divided by the overall frequency to get the k-mer signature.

For example,

Gene sequence = ACGTGGCAGCAATCGACGGT
Frequency = AA-1, AC-4, AT-1, AG-1, CA-3, CC-2, CG-3, GA-2, GC-2, TA-0
Length = 20
Possible 2-mers= 19
2-mer signature =
AA – 0.0526, AC – 0.2105, AT – 0.0526, AG – 0.0526, CA – 0.1579
CC – 0.1053, CG – 0.1579, GA – 0.1053, GC – 0.1053, TA – 0.0000

Finally, the average and standard deviation are calculated, thus, computing the lower and upper bound. Windows whose k-mer signature is less than the lower bound or greater than the upper bound are identified as probable GIs. The steps are summarized in figure 5.

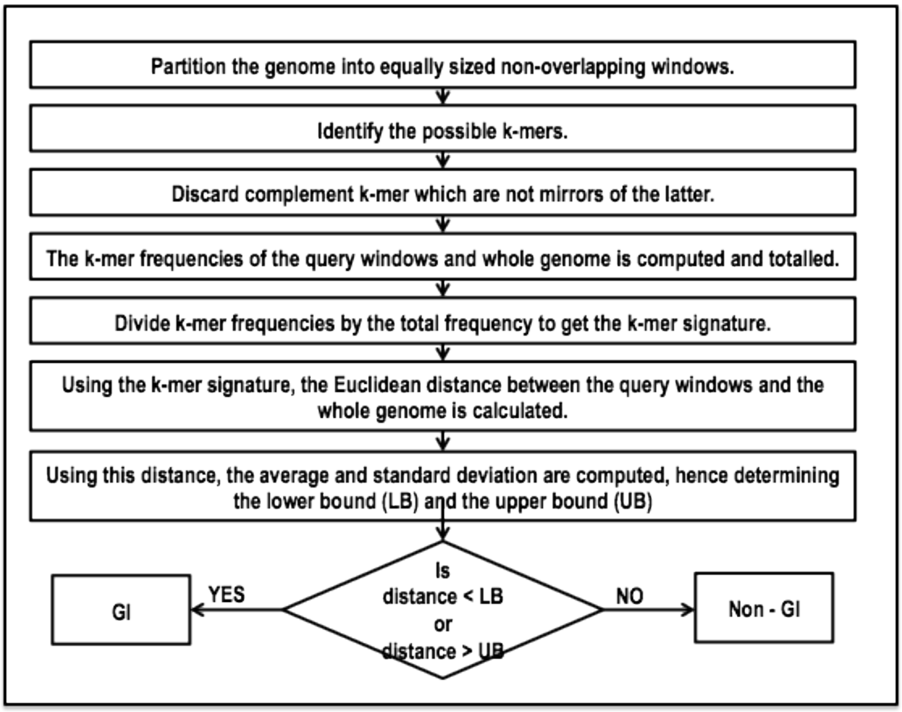

Partition the genome into equally sized non-overlapping windows.
Identify the possible k-mers.
Discard complement k-mer which are not mirrors of the latter.
The k-mer frequencies of the query windows and whole genome is computed and totalled.
Divide k-mer frequencies by the total frequency to get the k-mer signature.
Using the k-mer signature, the Euclidean distance between the query windows and the whole genome is calculated.
Using this distance, the average and standard deviation are computed, hence determining the lower bound (LB) and the upper bound (UB)
GI
YES
Is distance < LB or distance > UB
NO
Non - GI

Figure 5 - Flowchart illustration of k-mer signature analysis.

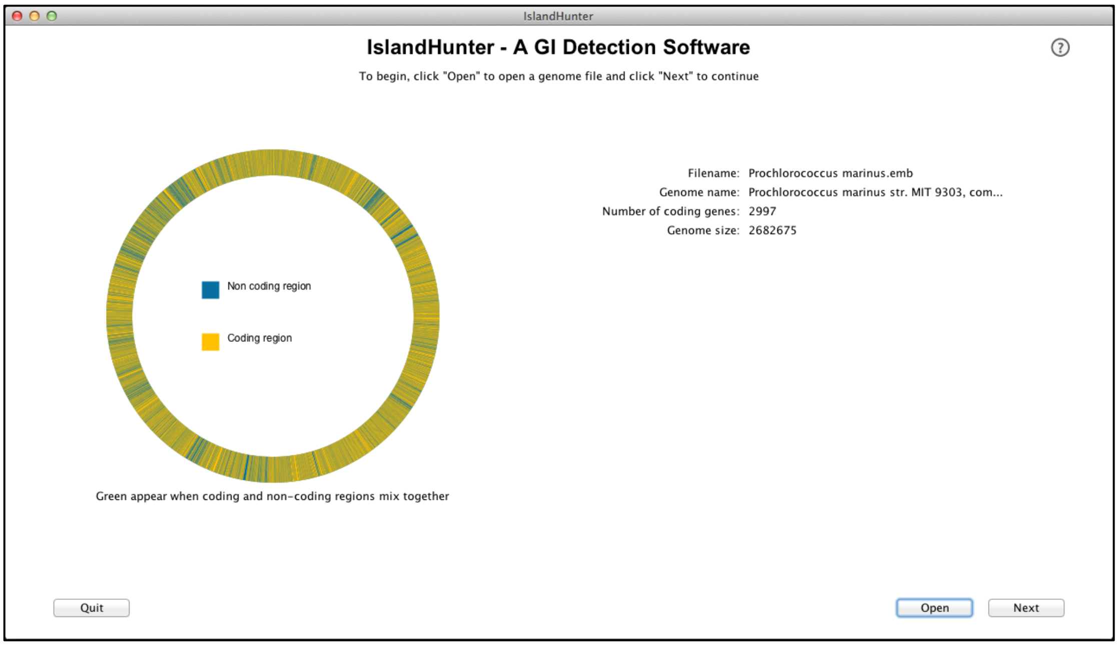
Note: For lower bound = %GC of whole genome – (SD*n)
For upper bound = %GC of whole genome + (SD*n)
Where n is a sensitivity factor (e.g. 1.5)

3.7. Combined GI-detection metods

IslandHunter provides a user with the option to choose any number of the methods discussed above and the software displays the list of GI regions identified by each method individually and the list of GI regions identified by more than one method. The list of GI regions identified by more than one method would be more appropriate because individual methods may give more GI regions than the actual ones e.g. GC content analysis identifies more regions.

4.0 The Interface of IslandHunter

IslandHunter provides an easy-to-use interface where users are first prompted to load a prokaryotic genome file and thereafter requested to choose one/more methods to identify GIs (Fig. 6). A help button (**?**) is also provided for the convenience of users, such that clicking the question mark icon, the user manual is displayed in PDF.



IslandHunter - A GI Detection Software

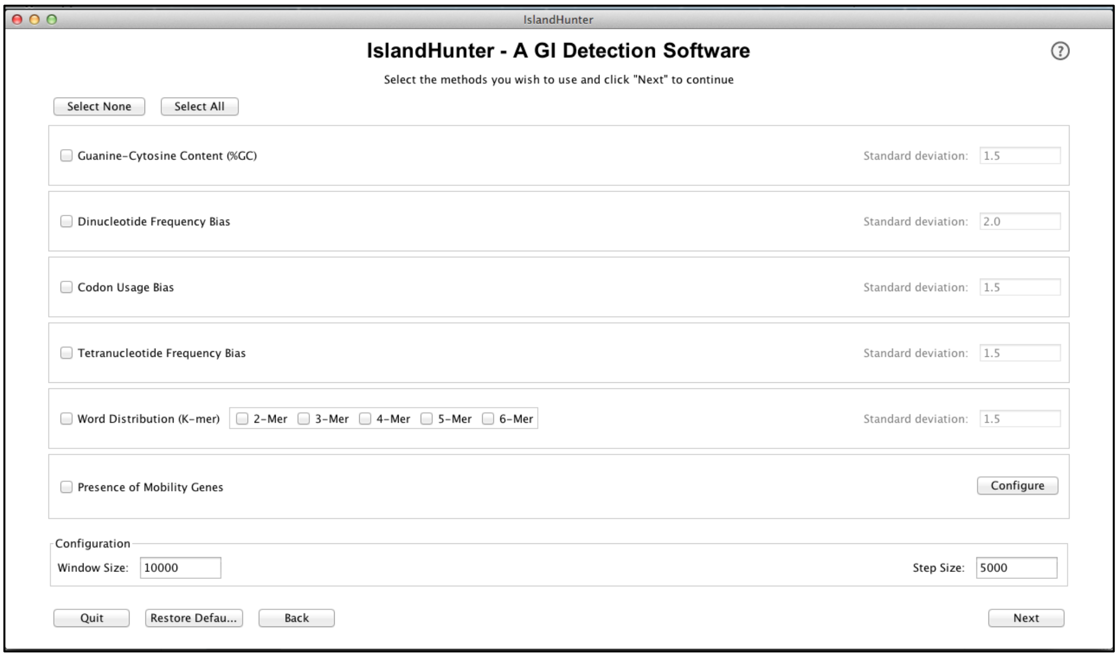
To begin, click "Open" to open a genome file and click "Next" to continue

Filename: Prochlorococcus marinus.emb
Genome name: Prochlorococcus marinus str. MIT 9303, com...
Number of coding genes: 2997
Genome size: 2682675

Green appear when coding and non-coding regions mix together

Figure 6 – IslandHunter LoadFile Interface

Users are then required to go to the next window to choose one or more methods for the detection of probable GIs as shown in figure 7.



IslandHunter

IslandHunter - A GI Detection Software

Select the methods you wish to use and click "Next" to continue

□ Guanine-Cytosine Content (%GC) Standard deviation: 1.5

□ Dinucleotide Frequency Bias Standard deviation: 2.0

□ Codon Usage Bias Standard deviation: 1.5

□ Tetranucleotide Frequency Bias Standard deviation: 1.5

□ Word Distribution (K-mer) □ 2-Mer □ 3-Mer □ 4-Mer □ 5-Mer □ 6-Mer Standard deviation: 1.5

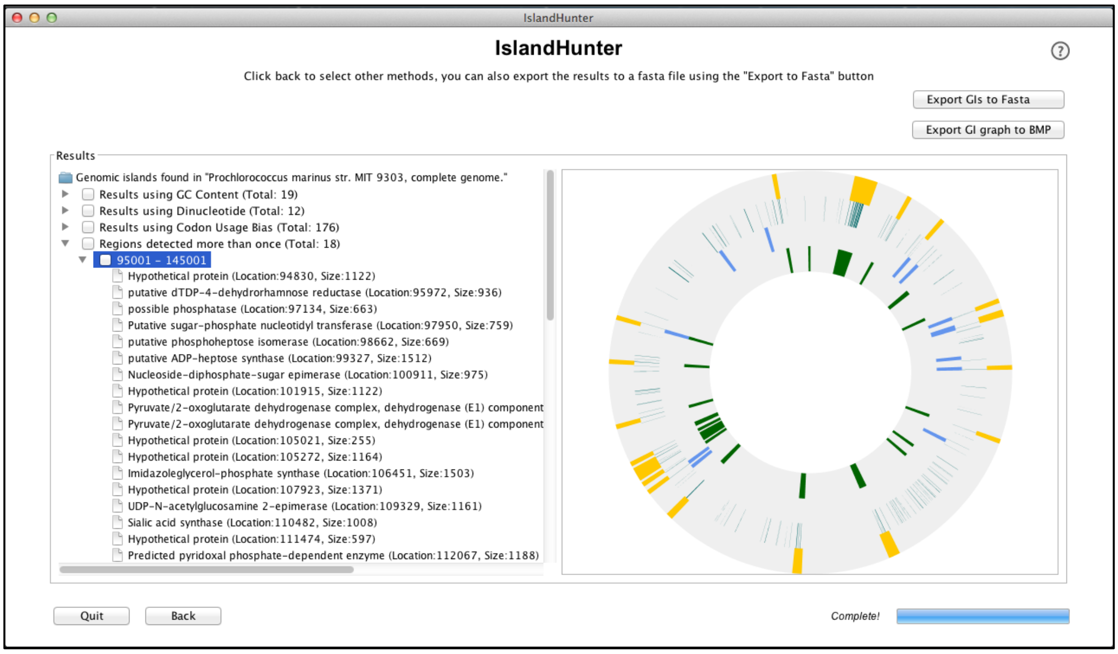
□ Presence of Mobility Genes

Configuration

Window Size: 10000 Step Size: 5000

Figure 7 – Selecting methods to find GIs

After choosing the method/s, IslandHunter will display the list of probable GIs as per each chosen method and the list of probable GIs identified by more than one method. For each probable GI region, the list of coding sequences found in each GI is also displayed in a tree view (Fig. 8). Users can view the contents of each region and if required can choose and export a specific region to FASTA, containing the GI sequences for further processing e.g. BLAST.



IslandHunter

IslandHunter

Click back to select other methods, you can also export the results to a fasta file using the "Export to Fasta" button

Results

Genomic islands found in "Prochlorococcus marinus str. MIT 9303, complete genome."
▶ □ Results using GC Content (Total: 19)
▶ □ Results using Dinucleotide (Total: 12)
▶ □ Results using Codon Usage Bias (Total: 176)
▼ □ Regions detected more than once (Total: 18)
  ▼ □ 95001 - 145001
    Hypothetical protein (Location:94830, Size:1122)
    putative dTDP-4-dehydrorhamnose reductase (Location:95972, Size:936)
    possible phosphatase (Location:97134, Size:663)
    Putative sugar-phosphate nucleotidyl transferase (Location:97950, Size:759)
    putative phosphoheptose isomerase (Location:98662, Size:669)
    putative ADP-heptose synthase (Location:99327, Size:1512)
    Nucleoside-diphosphate-sugar epimerase (Location:100911, Size:975)
    Hypothetical protein (Location:101915, Size:1122)
    Pyruvate/2-oxoglutarate dehydrogenase complex, dehydrogenase (E1) component
    Pyruvate/2-oxoglutarate dehydrogenase complex, dehydrogenase (E1) component
    Hypothetical protein (Location:105021, Size:255)
    Hypothetical protein (Location:105272, Size:1164)
    Imidazoleglycerol-phosphate synthase (Location:106451, Size:1503)
    Hypothetical protein (Location:107923, Size:1371)
    UDP-N-acetylglucosamine 2-epimerase (Location:109329, Size:1161)
    Sialic acid synthase (Location:110482, Size:1008)
    Hypothetical protein (Location:111474, Size:597)
    Predicted pyridoxal phosphate-dependent enzyme (Location:112067, Size:1188)

Complete!

Figure 8 – IslandHunter Output for Prochlorococcus marinus str. MIT 9303, complete genome

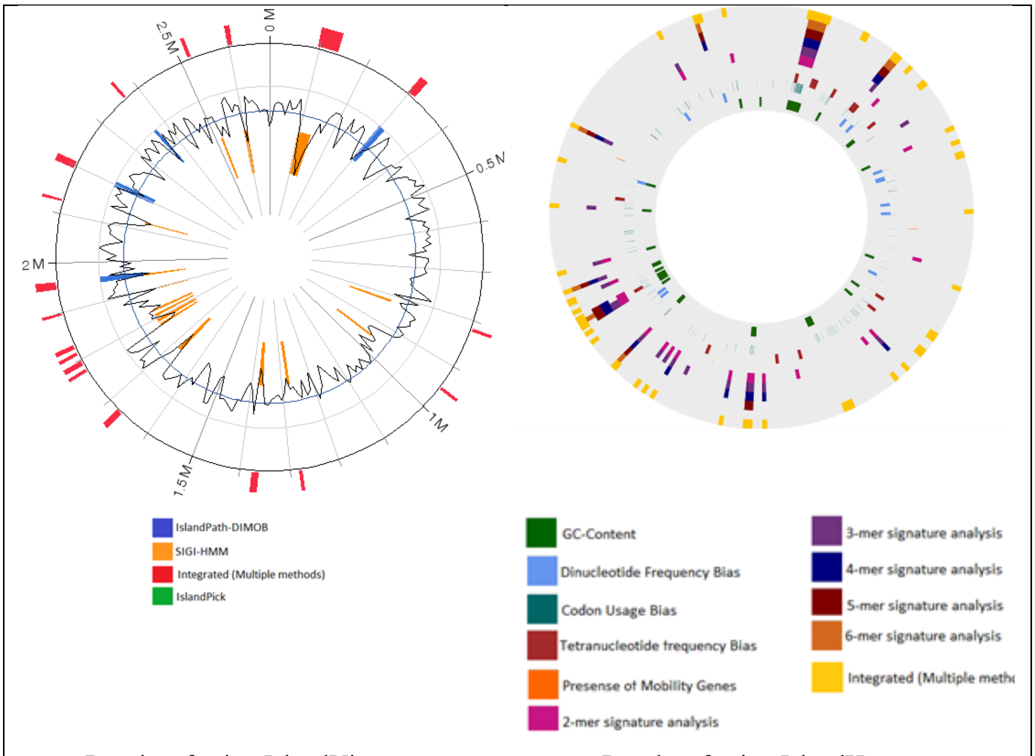
If the user chooses mobility genes as one of the methods to identify GIs, a predefined list of mobility genes is provided. It must be noted that for the mobility genes a number of synonyms may exist and thus users of the system may configure the provided list of mobility genes (i.e. add more).

IslandHunter's interface is better represented than the existing software as it has an animated circular graph which gives the methods used as well as the GI's position. Besides, the tree-view graph allots detailed information about the GI's position by stating its genetic content. Figure 9 gives the result of IslandHunter using Prochlorococcus marinus str. MIT 9303, complete genome in Embl format.

5.0 Results and discussion

The reliability of IslandHunter is validated by comparing its outputs with that of IslandViewer (Langille et al., 2009). A sample output of running the software for *Prochlorococcus marinus str. MIT 9303* is shown in Fig 9.

IslandViewer is a web-based application developed for researchers to view and download GIs. The facility of uploading any unpublished and yet unknown genome is provided. The latter comprises of many refined practices such as IslandPick (Pride et al., 2003), IslandPath-DIMOB (Hsiao et al., 2003) and SIGI-HMM (Waack et al., 2006; Almagor et al., 1983) which are very resource intensive. Moreover, IslandViewer depends upon Internet connection while IslandHunter is a standalone application.



Results of using IslandViewer Results of using IslandHunter

Figure 9 – Predicted GIs in *Prochlorococcus marinus str. MIT 9303 chromosome*

It is clearly shown that the results of IslandViewer are present in IslandHunter. But, IslandHunter identifies more GI regions as its resulting GI regions are predicted by combining the outcomes of two or more algorithms. IslandHunter is a standalone software; it does not need any database unlike the other two. Its output can be exported to a .fasta file which can later used to BLAST to find the origin of the GI segment.

These detailed data may help a researcher to know the purpose for which the foreign segments were inserted into the DNA sequence of the host prokaryote. One last point is that, IslandViewer takes GenBank and Embl files as input, while IslandHunter accepts GenBank, Embl and Fasta as input.

6.0 Conclusion

Horizontal gene transfer plays an important role in bacterial evolution. Its effect on evolution is dependent on the number of genes that have been transferred to and successfully maintained in microbial genomes (Boto et al., 2009). In this paper, the features of a standalone GI identification tool, IslandHunter, have been highlighted. IslandHunter has been developed using nucleotide-based statistics, for instance, GC content, codon usage bias, genome signature (dinucleotide frequency bias), tetranucleotide frequency bias, and k-mer signature analysis along with the presence of mobility genes. The results of IslandHunter are very similar to that of IslandViewer. Moreover, IslandHunter gives an integrated result, which is accurate, and it runs locally as compared to IslandViewer.

Despite using multiple methods, categorizing GIs are not that easy as the foreign DNA sequences get adapted to the new host and evolve to incorporate the genome, making it difficult to identify. Amelioration (Lawrence et al., 1997) (the process whereby the sequence of the island becomes similar to that of the host in GC content and codon usage due to mutational biases of the host) may occur and obscure the GI, and for this reason it is less likely to be identified as an island.

Additionally, methionine (AUG) is the start codon for genes, but, in prokaryotes, there are two alternate start codons namely GUG and UUG, which are basically Valine and Leucine respectively ((Elzanowski et al., 2010), (Marri et al., 2008)). This complicates the identification of GI when using genic methods. These problems may be solved using a whole genome comparison method.

Acknowledgment

The authors wish to thank Prof. Christos A. Ouzounis of CERTH (Tessaloniki, Greece) for his invaluable advice on this paper. They would also like to extend their appreciation to the BioJava team for creating a powerful and yet easy-to-use library.

Software Availability

IslandHunter can be obtained from:

https://github.com/ShakunBaichoo/IslandHunter

Sample Genome file used for testing from EBI

Prochlorococcus marinus str. MIT 9303, complete genome in EMBL format, also available along with the software as one of the sample genome files.

References

Almagor HA: **Markov analysis of DNA sequences**. Journal of Theoretical Biology 1983, 104: 633–645.

Azad RK and Lawrence JG: **Towards more robust methods of alien gene detection.** Nucleic Acids Research 2011, Vol. 39(9), doi:10.1093/nar/gkr059

Benjamin A, Genetic Elements of Microbes: **A Comprehensive and Integrated Genomic Database Application**. Thesis, Rochester Institute of Technology 2009, accessed from http://scholarworks.rit.edu/cgi/viewcontent.cgi?article=5077&context=theses , [11 July 2014]

Bolshoy A, Volkovich ZV, Kirzhner V and Barzily Z: **Genome Clustering: from Linguistic Models to Classification of Genetic Texts.** Studies in Computational Intelligence 2010, ISBN 978-3-642-12952-0

Boto L, **Horizontal gene transfer in evolution: facts and challenges**. Proc. Royal Society of Biological Sciences 2009, 277, 819–827 doi:10.1098/rspb.2009.1679

Carbone A, Zinovyev A and Képès F: **Codon adaptation index as a measure of dominating codon bias.** Bioinformatics 2003, 19 (16) 2005–2015

Elzanowski A and Ostell J: **The Genetic Codes 2010**: Accessed from http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi [11 July 2014]

Hackenberg M, Carpena P, Bernaola-Galvan P, Barturen G, Alganza AM and Oliver JL, WordCluster: detecting clusters of DNA words and genomic elements, Algorithms in Molecular Biology 2011, 6(2):, doi:10.1186/1748-7188-6-2

Hildebrand F, Meyer A, Eyre-Walker A: **Evidence of Selection upon Genomic GC-Content in Bacteria.** PLoS Genetics 2010, 6(9):e1001107, DOI: 10.1371/journal.pgen.1001107

Ho Sui SJ , Fedynak A, Hsiao WWL, Langille MGI, Brinkman FSL: **The Association of Virulence Factors with Genomic Island.**, PLoS ONE 2009, 4(12): e8094. doi:10.1371/journal.pone.0008094

Hsiao W, Wan I, Jones SJ, and Brinkman FSL: **IslandPath: aiding detection of genomic islands in prokaryotes.** Bioinformatics 2003, vol.19(3):418-420

Hurst LD and Merchant AR: **High guanine-cytosine content is not an adaptation to high temperature: a comparative analysis amongst prokaryotes**. Proc. Royal Society of Biological Sciences 2001 268(1466): 493–497, 10.1098/rspb.2000.1397

Juhas M, Van de Meer JR, Gaillard M, Harding RM, Hood DW, and Crook DW: **Genomic islands: tools of bacterial horizontal gene transfer and evolution**. FEMS Microbiol Rev 2009, no. 33(2):376-393

Karlin S: **Global dinucleotide signatures and analysis of genomic heterogeneit.**, Current Opininion Microbiology 1998, 1, 598–610.

Karlin S: **Detecting anomalous gene clusters and pathogenicity islands in diverse bacterial genome.**, TRENDS in Microbiology 2001, vol. 9(7):335-343

Karlin S and Burge C: **Dinucleotide relative abundance extremes: a genomic signature.** TRENDS in Microbiology 1995, 11(7):283-290

Langille MGI: **Computational Prediction and characterisation of Genomic Islands: Insights into Bacterial Pathogenicity**. PhD Thesis 2009, Simon Fraser University, accessed from http://bioinformatics.bcgsc.ca/graduates/documents/Langille_PhD_Thesis_Final.pdf [11 July 2014]

Langille MGI and Brinkman FSL: **IslandViewer: an integrated interface for computational identification and visualization of genomic islands**. Bioinformatics 2009, 25(5):664-665, doi: 10.1093/bioinformatics/btp030

Lawrence JG and Ochman H, **Amelioration of bacterial genomes: rates of change and exchange**. Journal of Molecular Evolution 1997, 44(4):383-397

Marri PR, Golding GB, **Gene amelioration demonstrated: The journey of nascent genes in bacteria**. Genome 2008, vol 51(2):164-168

Moriyama EN: **Codon Usage**. University of Nebraska, Lincoln, Nebraska, USA, Published online (Wiley) 2006, DOI: 10.1038/npg.els.0005102, In book: eLS

Pride DT, Meinersmann RJ, Wassenaar TM, and Blaser MJ: **Evolutionary Implications of Microbial Genome Tetranucleotide Frequency Biases**. Genome Research 2003, 13(2):145-58

Rodgers JL and Nicewander WA: **Thirteen ways to look at the correlation coefficient.** The American Statistician 1988, 42(1):59–66

Waack S, Keller O, Asper R, Brodag T, Damm C, Fricke WF, Surovcik K, Meinicke P, and Merkl R: **Score-based prediction of genomic islands in prokaryotic genomes using hidden Markov models**. BMC Bioinformatics 2006, vol. 7:142