What do we gain when tolerating loss? The information bottleneck, lossy compression, and detecting horizontal gene transfer

Apurva Narechania[1]*, Rob DeSalle[1], Barun Mathema[2], Barry Kreiswirth, and Paul J. Planet[1,4,5]*

[1]Institute for Comparative Genomics, American Museum of Natural History, New York, NY
[2]Department of Epidemiology, Mailman School of Public Health, Columbia University, New York, NY
[3]Center for Discovery and Innovation, Hackensack Meridien Health, Nutley, NJ
[4] Division of Infectious Diseases, Children's Hospital of Philadelphia, Philadelphia, PA
[5] Perelman School of Medicine, University of Pennsylvania, Philadelphia, Pennsylvania, United States of America

*Correspondence to: Apurva Narechania (anarechania@amnh.org) & Paul J. Planet (planetp@email.chop.edu)

Word Count
Abstract: 263
Body: 4824

Running Title: Information bottleneck for detecting recombination

Abstract

Most microbes have the capacity to acquire genetic material from their environment. Recombination of foreign DNA yields genomes that are, at least in part, incongruent with the vertical history of their species. Dominant approaches for detecting such horizontal gene transfer (HGT) and recombination are phylogenetic, requiring a painstaking series of analyses including sequence-based clustering, alignment, and phylogenetic tree reconstruction. Given the breakneck pace of genome sequencing, these traditional pan-genomic methods do not scale. Here we propose an alignment-free and tree-free technique based on the sequential information bottleneck (SIB), an optimization procedure designed to extract some portion of relevant information from one random variable conditioned on another. In our case, this joint probability distribution tabulates occurrence counts of k-mers with respect to their genomes of origin (the relevance information) with the expectation that HGT and recombination will create a strong signal that distinguishes certain sets of co-occuring k-mers. The technique is conceptualized as a rate-distortion problem. We measure distortion in the relevance information as k-mers are compressed into clusters based on their co-occurrence in the source genomes. This approach is similar to topic mining in the Natural Language Processing (NLP) literature. The result is model-free, unsupervised compression of k-mers into genomic topics that trace tracts of shared genome sequence whether vertically or horizontally acquired. We examine the performance of SIB on simulated data and on the known large-scale recombination event that formed the *Staphylococcus aureus* ST239 clade. We use this technique to detect recombined regions and recover the vertically inherited core genome with a fraction of the computing power required of current phylogenetic methods.

Introduction

Whole microbial genomes are being sequenced at an unprecedented rate.[1] Focused sequencing of key organisms and broad sequencing of microbial environments have expanded our knowledge of evolution and the microbiosphere[234]. However, the production of data is outstripping our ability to analyze it[5]. Most work in molecular evolution is grounded in sequence alignment and phylogenetic tree reconstruction. However, whole genome alignment breaks down with increasing diversity, and tree-based techniques suffer from an exponential increase in compute time with broader taxon sampling. The evolution of microbes is particularly challenging because horizontally transferred elements contribute historical signal that is unrelated to vertical descent. Most dominant techniques for capturing horizontal gene transfer (HGT) and recombination require either alignment of reads across a reference genome (eg., single nucleotide polymorphism (SNP) based analysis or whole genome alignment[67]. Where global alignment is impossible, phylogenomic tools require all-against-all analyses designed to fix genes into aligned orthologous groups[8910]. All of these approaches require careful curation, tree-building, HGT/Recombination detection analysis, and deliberate sampling to limit data to reasonable scales. For larger, unbiased datasets that include as much natural variation as possible, these approaches are not sustainable. To handle the onslaught of genomes, we need tools that can tolerate information loss without sacrificing knowledge of key evolutionary events.

Lossy compression, where an individual or algorithm makes decisions about which data are important (or relevant) from a large body of information[11], may offer a solution. To do this in a principled way, the relevance of a given dataset can be measured as information retained

about some other correlated variable. For example, in unsupervised natural language processing (NLP) large corpora of texts are distilled to a few topics that reflect overall themes by comparing patterns of co-occurring words in the source texts. In topic modeling of this sort, the texts themselves are the relevance variable. The goal is to cluster the overall word distribution with respect to the documents from which they arise. If X is the original data distribution, T its compressed representation, and Y the relevance variable, the challenge is to pack X into as few clusters, T, as possible without sacrificing too much information, Y. This idea was first described by Tishby, Pereira and Bialek as the information bottleneck (IB)[12]. It was premised on rate distortion, Shannon's original theory of lossy compression which yoked signal distortion to the rate at which that signal can be encoded[13]. Distortion is severe if the signal is forced through a small communication channel and gets cleaner as the channel widens. The IB's primary innovation was the use of a relevance variable to quantify this distortion. Topic modeling was one of this technique's first applications.

Topic modeling has become an important part of the NLP literature with a number of wider applications to unsupervised machine learning. The dominant technique in the field is Latent Dirchilet Allocation (LDA)[14], a probabilistic method, that like the IB, considers each document as a mixture of topics. Some groups have applied this idea to whole genomes[151617], and since the publication of STRUCTURE, LDA has become foundational in the genetics literature where populations are inferred by the distribution of alleles at measured loci[18]. Despite LDA's popularity and success, a number of authors have shown that unbalanced sampling can lead to erroneous or missed population assignments[19]. LDA also makes a number of statistical assumptions including the assignment of hyperparameters and a Dirchilet prior[20].

In contrast, the IB is model free and less likely to suffer from size sample bias. The distortion measure emerges from the analysis of the relevance variable, revealing underlying topics without having to set any distributional parameters other than the number of clusters expected.

Because it is model free, the IB is a powerful approach for microbial genomics where very little is known about the diversity of the organisms in nature or their distribution. Genomes are living documents that can be sliced into words of arbitrary size. This metaphor is straightforward and has been explored with respect to other NLP techniques elsewhere[212223]. In a genomic context, where words are k-mers (X) and documents (Y) are their genomes of origin we hypothesized that IB derived topics (T) may represent co-occurring groups of k-mers that highlight shared ancestry. These topics might include k-mers arranged in co-linear blocks corresponding to a single element, or k-mers distributed across the genome that were inherited in concert. In either case, compression of these k-mers into topics is guided by how often they co-occur with respect to their genomes of origin. This mechanism will tend to group adjacent k-mers in a recombined region because the recombination event is likely restricted to just a subset of taxa. Additionally, shared tracts of co-occuring k-mers common to all genomes, offer a simple, operational definition of a genomic "core".[76] For microbial genomes where HGT is rampant[2425] we can therefore use the technique to learn which portions of the genome form the vertically inherited core, and which portions have been recombined, or inherited horizontally. In the NLP topic modeling analogy, the core genome of a species could be considered the set of meaningful words across every book in a specialized library, while recombined regions are like themes or ideas restricted to only certain shelves.

Here we apply the IB to microbial genomes. Remarkably, our approach identifies recombination tracts without making any attempt to model evolution, annotate genes, reconstruct trees, or build alignments. In addition, the IB treats genic and intergenic portions of the genome equally, obviating the need for gene-based pangenomic analysis[26]. Applying the information bottleneck to a k-mer occurrence matrix identifies genome segments with shared vertical or horizontal evolutionary history in a fraction of the time used by other approaches.

Theory and Implementation

Consider a set of genomes each of which is chopped into overlapping k-mers. One way to measure the overall relatedness of two of these genomes is to compare their k-mer conditional distributions. To do this we can define

$$p(x|y) = \frac{n(x|y)}{\sum_y n(x|y)}$$

where X is the set of all k-mers, Y the set of all genomes, and $n(x|y)$ is the occurrence count of the k-mer, x, in genome y. The exercise would then be to group genomes with similar k-mer distributions across all k-mers. In the natural language processing literature, this idea was formalized as distributional clustering[27].

However, finding the right distance or distortion measure between these distributions is non-trivial. It is especially difficult when the important features of the signal are unknown. Imagine compressing music into MP3s without data on which frequencies are most important

for human perception, or determining themes from a body of literature if words were decoupled from their books. Even when important components of the signals are known, most clustering algorithms will resort to domain specific, pairwise distances or quantization to find a compressed set of classes with either high levels of internal connectivity or low levels of internal distortion. However, domain specific distortions reduce the usefulness of these clustering techniques. For example, in bioinformatics, clustering based on sequence alignment is subject to all the vagaries of the alignment procedure and parameters therein.

An antidote to these narrow clustering applications is to operate in an information theoretic space where the primary measurement is relevant quantization[12]. The IB extends Shannon's rate distortion theory by guiding it with an additional, orienting variable. Tishby et al[12] enriched a theory about transmission efficiency with the concept of relevance (Y), or the value of the information transmitted. The choice of Y defines relevant features in the signal. If X and Y are tabulated as a joint probability distribution, the information that X provides about Y is squeezed through a simpler representation, T. For the technique to work, the two variables in our joint distribution *p(x,y)* must be non-independent, or more precisely, must have positive mutual information, *I(X,Y)*:

$$I(X,Y) = \sum_x \sum_y p(x)p(y|x) \log \frac{p(y|x)}{p(y)}$$

T is now a meaningful compression of the data, maximizing the mutual information between the clusters and documents, I(T;Y), while minimizing the mutual information between the words and the clusters, I(T;X). The IB is a classic optimization problem.

With the distribution in hand and implemented as a k-mer occurrence matrix, we can quantize the set of all k-mers directly by minimizing information lost about their source genomes. If X is compressed into T then we can find the optimal assignments for X by minimizing the following Lagranian with respect to Y:

$$\mathcal{L}[p(t|x)] = I(X;T) - \beta I(X;Y)$$

This formulation balances the compactness of X, with the erosion of information about Y. β is a multiplier that slides through the optimization landscape. As beta approaches 0, k-mers are clumped into fewer and fewer clusters, emphasizing compression. As beta approaches infinity, every k-mer is its own cluster, preserving all relevant information. Of course, collapsing all k-mers into one cluster is overly reductive, and assigning each k-mer to its own cluster is meaningless. The IB negotiates these two extremes (Figure 1). In NLP, the result is a set of clusters that coalesce into topics over a body of literature[28]. In genomics, these same clusters might yield co-occurring and/or spatially co-located k-mers with distinct biological and/or evolutionary meaning.

Remarkably, minimizing the Lagranian above has an exact, optimal solution[12]. The most surprising outcome of this solution is that the relative entropy, or Kullback Liebler divergence[29], emerges as the distortion measure for the information bottleneck. The relative entropy is a fundamental quantity in information theory, and in the IB context, it measures the distortion between the points, x (k-mers), as they are quantized into their clusters, t, with respect to the relevance variable, y (genomes):

178

179 $$D_{KL} = \sum_{y} p(y|x) \log \frac{P(y|x)}{P(y|t)}$$

180

181 Calculation of the optimal solution requires soft clustering, that is, any given k-mer can exist in
182 more than one cluster. But soft clustering can be slow and difficult to devise. Early
183 implementations of the information bottleneck therefore settled on hard clustering
184 approximations. In hard or deterministic clustering, each k-mer is assigned to only one cluster,
185 an assumption that eases computational burden but does not generally arrive at globally
186 optimal solutions.

187 The most obvious hard clustering algorithm is agglomerative, or bottom-up[30]. Consider
188 again the set of all genomes, X, and their compressed representation, T. If we start with a
189 scenario where every k-mer in X occupies its own singleton cluster, we can systematically
190 reduce the dimensionality by merging clusters that minimize some distortion score. This greedy
191 merging procedure produces a tree. But agglomerative clustering does not yield stable cluster
192 membership. The tree varies every time the process is reinitialized. Worse, its computation is
193 expensive, requiring cubic time complexity and quadratic memory complexity. In a genomic
194 context where we routinely deal with billions of k-mers, this approach is a nonstarter.

195 Instead, we implemented a sequential clustering procedure where the number of
196 clusters is defined at the outset and remains consistent throughout the calculation. From an
197 initial random distribution of all k-mers across this set of clusters, we draw one k-mer out, and
198 represent it as a singleton. Now using greedy optimization, we merge this singleton into one of

199 the existing bulk clusters. Slonim's sequential information bottleneck (SIB)[31] employs the
200 Jensen-Shannon divergence[3221] in the cost of merging a k-mer, x, into a cluster, t:
201
202 $$d(x,t) = (p(x) + p(t)) * D_{JS}(p(y|x), p(y|t))$$
203
204 A k-mer will join a new cluster only if its new address reduces the total distortion. Otherwise it
205 will remain in its existing cluster. With respect to our initial random conditions, this algorithm is
206 guaranteed to converge to a local optimum. We mitigate the risk of getting trapped in local
207 optima by testing several random initializations.
208 Once the clusters stabilize, we quantify the information captured by calculating the
209 normalized mutual information, *NMI = I(T;X) / I(X;Y).* Trivially, NMI = 1 when each k-mer
210 occupies its own cluster. The curve traced between T = 1 (NMI = 0) and T = *x* is called the
211 relevance compression curve[33]. This is analogous to the optimization of β in the Lagranian
212 above, but for the deterministic case involving hard clustering. As with β, the shape of this
213 curve describes the compressibility of the data.
214 The most important aspect of the SIB, and the reason we chose it for this work, is that it
215 makes the concept of the information bottleneck accessible to modern genomics. The time
216 complexity is linear in the number of k-mers and the number of clusters. This improvement
217 makes information theoretic NLP a useful tool to discover genomic topics encoded as clusters
218 of co-occurring k-mers.
219
220 **Results and Discussion**

The bottleneck in test: one large, simulated HGT event

The simple example in Figure 2 illustrates how the bottleneck works in practice. In SimBac[34], we simulated four 1 megabase genomes with a single 200 kilobase recombination event. The event is common to genomes 0, 2 and 3, but is not found in strain 1. We initialized the simulation with a random distribution of 19-mers across five clusters. To learn the true distribution, we leveraged information in our relevance variable, the source genomes. The inset table shows how this distribution evolves as we iterate through the sequential information bottleneck (SIB). Since the relevance variable is expected to drive the unsupervised compression of these k-mers, we also included the genomes in this table. Counts across each row therefore reflect how many times a k-mer in that cluster is found in a particular genome.

The SIB starts by randomly distributing the k-mers, destroying all information available in the original occurrence matrix. At the outset, the normalized mutual information is therefore zero. With each SIB loop, we attempt to reclaim as much of this information as possible given the number of clusters we choose to model. Because the technique is inherently lossy, the SIB will never recover all of the information originally encoded, but aims to extract the most salient themes, or topics.

In the example shown here, after the first loop, cluster 3 (the cluster designations are arbitrary) has attracted the most k-mers in roughly even proportion across the genomes. The normalized mutual information has also jumped to 0.69, indicating that just one pass of sorting k-mers into five bins effectively captures 70% of the information available in the original occurrence matrix. The second and third loops refine the other clusters into mutually exclusive

sets and add to cluster 3, which strengthens into a genomic "core" defined here as the cluster of k-mers with the highest average representation across all genomes and the lowest index of dispersion.

By the third pass through the k-mers, the SIB reaches a plateau in the normalized mutual information, and the counts of k-mers across clusters and genomes have stabilized. For this particular set of starting conditions, the SIB reclaims nearly 91% of the information in the original matrix. To put this in perspective, we have effectively reduced the outsized, uninterpretable dimensions of our original data – 1.25 million unique k-mers – into the 5 clusters we set out to model, while sacrificing only 9% of the original information present in the relevance variable.

In a genomic context, we hypothesized that the spatial organization of k-mer clusters would correspond to areas of common ancestry. In Figure 2, we mapped k-mers from various clusters to the genome backbones of strain 1 and strain 2. Cluster 3 occupies the outer tracks of both strains. This cluster emerges as a dense block of shared genome sequence and corresponds to our definition of a bottleneck-defined core. But the block is interrupted by our simulated recombination event. Since this event is restricted to only genomes 0, 2 and 3, the region is absent from the core. Its k-mers are instead captured by cluster 4 while cluster 1 serves as a counterpoint, containing the ancestral state prior to the simulated event.

Several smaller, simulated HGT events

Though large hybridization events like the one we simulated here do occur (see our analysis of ST239 *S. aureus* below), smaller and more abundant events typify most microbial

265 evolution[35]. To see how the bottleneck performs in this more challenging case, we simulated
266 ten 1 megabase genomes with a background mutation rate of 0.01 and a recombination rate of
267 0.0001, resulting in 57 discrete events averaging 500 basepairs in size (from 6 to 2884 bases). In
268 **Figure 3**, the innermost track marks the locations of these events.

269 The ability to detect horizontally transferred sequence is strongly dependent on its
270 evolutionary distance from the genome background[7]. To visualize this dependence, we
271 modulated the divergence of our 57 recombination events (an arbitrary number derived from
272 the first simulation) and measured the effect on the core cluster, one of 60 modeled for this
273 simulation. The innermost histogram in Figure 3 shows the core pattern with an external
274 (between species) divergence rate of 0.1, an order of magnitude higher than the background.
275 We observe clear "valleys" in the k-mer distribution of the core that are coincident with the
276 positions of our 57 events. But this pattern steadily disappears as we sweep through lower
277 rates of divergence (0.05, 0.03, and 0.01). The outermost track models the same mutation rate
278 as the background, resulting in dulled or partially filled valleys in the core genome. Plots of core
279 k-mers function almost as a photographic negative, highlighting blank spaces as regions of
280 potential evolutionary interest.

281 The k-mers that would otherwise occupy these gaps, are sorted into other clusters
282 because they are unique to only a subset of the genomes, and carry the recombination signal.
283 As we have shown in our first simulation, k-mers corresponding to the ancestral state should
284 fall into a different cluster. Note that this does not necessarily mean that each side (donor and
285 recipient) of an HGT event has its *own* cluster. Recall that compression is driven by genome
286 origin. If a single common ancestor sustains multiple transfer events, all k-mers from those

287 events will merge into a single cluster because they are shared by the same subset of
288 descendants.

289 The accounting becomes increasingly complicated when events overlap. Overlapping
290 events might mix across clusters depending on their arrangement and how frequently they
291 have been overwritten. When detection becomes difficult, we instead rely on an evolutionary
292 event's imprint on the core cluster. This approach exploits the idea of the core as a
293 photographic negative or a clonal frame. The pattern of HGT events in this negative is evident
294 by eye, but if the number of input genomes and the number of modeled clusters is large, visual
295 inspection is a burden, and subject to error in interpretation. Instead we introduce a method
296 based in change point detection to automatically detect changes in k-mer frequency[36]. We
297 specifically employ Bayesian change point detection[37] to model probabilities of change in the k-
298 mer frequency stream. As shown in Figure 4 change point probabilities spike at the start and
299 end of HGT events.

300 In addition to change point detection, we note that if counts of k-mers in an HGT region
301 are significantly lower than the rest of the core's background (Wilcoxon, $p < 0.05$), these
302 depletions can qualify as a simple signal marking some combination of HGT events. With these
303 criteria, at a divergence rate of 0.1, the bottleneck captures 56 of the 57 simulated events,
304 missing only the smallest.

305

306 *The k-mer skim*

307 Accounting for every overlapping k-mer in each strain is an unnecessarily close reading
308 of our genomic text. We can save on both memory and computation by selecting fewer k-mers

(skimming) from our source genomes with some set space between each sample. In Figure 5 we show that even when sampling every 25th 19-mer in our ten 1 Mbase simulated genomes, we still detect 55 of our 57 recombination events. Because the bottleneck relies on the signal inherent in k-mer co-occurrence, as we reduce the density of our k-mer sampling, we lose detection of the smallest events first. However, the compute time savings more than compensate for this loss in sensitivity. While analyzing every 19-mer requires nearly 12 minutes, skimming every 25th reduces the runtime to 30 seconds. This compares favorably with the efficiency of both ClonalFrameML[7] and Gubbins[6], the two dominant HGT detection methods in the literature. ClonalFrameML requires 110 seconds and captures only 47 of our 57 events. Gubbins finds 54 in 21 seconds. However, both ClonalFrameML and Gubbins require alignment and phylogenetic tree reconstruction, which both add massive prior computational cost and time.

Because the IB is alignment-free and tree-free, it is theoretically capable of handling larger datasets than any existing technology in reasonable amounts of time. To test this, we simulated 1000 1 Mb genomes with the same parameters as the smaller dataset shown in Figure 3. The simulation generated 620 unique recombination events. ClonalFrameML detected 564 (91%). Including time required to build a guide tree, this calculation consumed 32.5 CPU hours. Gubbins was slightly more accurate and significantly faster: 583 (95%) events over 16.3 CPU hours. Using Figure 5 as a guide, we ran the 1000 genome dataset through the SIB using a 25 base-pair skim. We detected an HGT imprint at 92% of sites in 1.5 CPU hours.

How well does the IB hold up under extreme evolutionary pressure?

To evaluate the performance of our technique with respect to recombination size and divergence rate, we simulated sets of ten 1 megabase (Mb) genomes for each variable. We set default parameters to 0.01 for background rate, 0.001 for recombination rate, 0.1 for HGT divergence rate, and 500 base pairs for average recombination tract size. We performed 100 replicates at each size and rate, and measured the imprint of the simulated events on the core cluster without the skim feature. Figure 6A shows this sweep for recombination tract length, and Figure 6B, for recombination tract divergence. In both cases, we observe saturating behavior. We see recombination imprints at 90% accuracy when events are larger than 100 base pairs with divergence rates of at least 0.02. Notably, our procedure can detect HGT in at least half of events that diverge at the very low rate of 0.005, well below the background. And only the very smallest recombination events (less than 7 basepairs) elude our technique completely.

Recombination tract length and divergence have direct and measureable effects on the efficacy of detection. As long as the total length of all recombination events is less than half the size of the genome, the core remains intact, and we can easily isolate HGT events of sufficient size and divergence. But recombination and background mutation rates are problematic because they redefine the core. For example, at high rates of recombination, every base of a 1 Mb genome is likely scrambled. Under such flux, some sites recombine several times. A high background mutation rate also disrupts stretches of common sequence that mark the core. As these rates increase, the core genome itself erodes. To measure this phenomenon, we again simulated 100 sets of ten 1 Mb genomes across a variety of recombination and background mutation rates. All three curves in Figure 7 show a steep decline in the size of the core with

353 increasing recombination rate. At rates of 0.01 and 0.1, we see no shared core at all. Each
354 genome has essentially rewritten itself into something distinct from all others. Core genome
355 signal grows stronger with lower background mutation, but even with background mutation set
356 to essentially zero, a high recombination rate destroys the core.
357
358 *The bottleneck in action: one large, real world hybridization event*
359 We used genomes from ST239 *Staphylococcus aureus* to illustrate that our method can
360 corroborate known, large scale recombination events found in nature. The ST239 strain is a
361 hybrid: a segment from a CC30 (clonal complex 30) donor replaced nearly 20% of the
362 homologous region in a CC8 strain[38]. The evolutionary histories of genes across these segments
363 are incongruent. Previous studies compared the histories of thousands of genes to reach this
364 conclusion[39]. Here, we attempt to localize this same phenomenon using the co-occurrence
365 pattern of k-mers alone. We chose 10 genomes (GCA_000146385.1, GCA_000012045.1,
366 GCA_000011505.1, GCA_000011265.1, GCA_000013425.1, GCA_000204665.1,
367 GCA_000159535.2, GCA_000027045.1, GCA_000017085.1, and SA21300), sampled from both
368 the donor clade (CC30), the recipient clade (CC8), and genomes outside of the evolutionary
369 event. When cut into overlapping 19-mers (no skim), these 10 genomes dissolve into 28.8
370 million k-mers, 4.72 million of which are unique.
371 Figure 8 highlights two of these 10 genomes, and three of the 60 clusters we modeled
372 for this analysis. Both *S. aureus* COL (CC8) and *S. aureus* T0131 (ST239) share a large, congruent
373 core. The gap in this core characterizes the dimensions of the recombination event, whose k-
374 mers are split into two other clusters, shown here as the second and third tracks. Like subtopics

in a vast library, the bottleneck learns the complete structural evolution of the clade as tracts, or topics, of co-occurring sequence. The clusters themselves comprise an evolutionary model for the structural event and the core genome. This evolutionary model is derived not from traditional character-based phylogenetic analysis, but from the presence/absence pattern of k-mers squeezed into a predefined number of groups. Genome origin guides the k-mer sort by forming the basis of the distortion measure. We lose information in a controlled and quantitative way, and we short circuit the long and arduous tasks phylogenomic analyses require[39] with an information theoretic procedure that runs for 2 hours on 1 CPU.

By definition, this sort of lossy compression is not perfect. In Figure 8, seemingly unrelated contaminants pollute the recombined region's clusters. This is equivalent to channel noise. It recalls Shannon's original formulation of the rate distortion problem[13]. When we force all the signal in our k-mer occurrence matrix through a narrow five cluster channel, portions of the original message emerge garbled. In this case, modeling more clusters increases the rate of transmission, and reduces the distortion of the message received.

With respect to the information bottleneck, we can quantify this effect using a relevance-compression curve[28]. Figure 9 shows curves for the ST239 genomes alongside 10 genomes of *Mycobacterium tuberculosis* and *Helicobacter pylori.* In all three cases, as the number of clusters modeled increases, we capture more normalized mutual information. The theoretical extremes for this curve are intuitive. At the origin, all the relevant information is destroyed. At the other end, we retain too much relevant information to interpret. The curve traced between these two extremes is a fingerprint of the data. A convex shape suggests natural structure easily modeled with just a few clusters. We see this in *M. tuberculosis*, a

397 species thought to be largely clonal with little recombination. On the other hand, data that
398 resists compression flattens this curve. Highly recombinogenic species like *H. pylori* suffer this
399 sort of steep information loss. Theoretically, the space above the curve for each species is
400 unachievable by any process, forming an upper bound. The relevance-compression curve
401 therefore defines absolute limits on the quantity and quality of information communicated as
402 we sweep through a dilating channel. This approach introduces a new type of comparative
403 genomics based not on alignments and trees, but on compression. We interpret the shape of
404 the relevance compression curve as a proxy for evolutionary mode. A convex curve implies
405 fewer recombination events and more vertical signal, whereas a flattened curve may signal a
406 species with a more open pangenome.

407 In the case of ST239, asking for just two clusters – a very narrow channel – captures
408 more than 40% of the relevant information. Remarkably, these two clusters separate the core
409 from the recombined region. Even the simplest model learns the most prominent evolutionary
410 process. Further along the curve, fifteen clusters capture almost all of the information. Beyond
411 fifteen, the curve elbows, and modeling gains are slight. In this way, the relevance-compression
412 curve defines the optimal number of clusters.[3340] But in the light of evolution this bend may
413 have a deeper meaning. Fifteen clusters are enough to adequately capture the complete set of
414 k-mer aggregation patterns across our chosen genomes. This point of diminishing returns may
415 signify an opportunity for interpretive balance: not so many clusters that we drown dominant
416 evolutionary events, and not so few that we neglect to model subtle k-mer co-occurrence
417 patterns. This particular use of the well-known elbow method in our information theoretic

context puts a crude limit on the dominant evolutionary paths taken by the genomic elements that comprise our species.

Conclusion (words=149)

The information bottleneck, a lossy compression technique borrowed from the information theoretic and Natural Langauge Processing literature, is well suited to detecting evolutionary patterns in sets of co-occuring k-mers. Here we have shown that we can detect simulated and real recombination events while highlighting a core set of k-mers that comprise the vertically inherited portion of any set of genomes. Moreover, the compressibility of any given set of genomes, as embodied in their relevance compression curves, offers a new way to compare the pangenomes of very different clades in the microbial tree of life. In our application, the bottleneck is informed by genome origin, our relevance variable. But the technique is general. The information bottleneck can be used for any biological contingency matrix where the goal is to cluster a variable into interpretable groups by preserving as much information as possible in the variable to which it is linked.

Software implementation: NECK (https://github.com/narechan/neck)

Figure Legends

Figure 1. The information bottleneck. In the information bottleneck a distribution, *X*, is compressed into *T* while retaining as much information as possible about a correlated relevance variable, *Y*. The joint distribution, *p(x,y)*, has positive mutual information and the goal of the information bottleneck is to capture as much of that information as possible at interpretive

443 scale. The technique is a classic optimization problem wherein the mutual information between
444 *T* and *X* is minimized, while the mutual information between *T* and *Y* is maximized. At
445 optimality, *T* is presumed to be a lossy but adequate model of *X*.
446
447 Figure 2. One simulated HGT event. A simple set of four simulated genomes with a single large
448 transfer event is shown. The transfer occurs in the common ancestor to genomes 0, 2, and 3.
449 The inset chart clearly shows that the k-mers corresponding to this event are captured by
450 cluster 4, while the ancestral state is captured by cluster 1. K-mers from these clusters map to
451 the location of the simulated event in genomes 0, 2 and 3 and genomes 1, respectively. Cluster
452 3 is the core and contains only one gap corresponding to the HGT region.
453
454 Figure 3. Several simulated HGT events. The innermost ring of this circos plot shows the
455 locations of 57 simulated HGT events across 10 1 Mbase genomes. The remaining concentric
456 tracks plot the core set of k-mers as calculated by the information bottleneck. In the outermost
457 frequency plot, the 57 HGT events diverge at the same mutation rate as the background, 0.01.
458 Going in towards the center, we increase the HGT divergence rate of the events to 0.03, 0.05,
459 and 0.1. Gaps in the core correspond with the simulated HGT events whose k-mers are sorted
460 into other clusters.
461
462 Figure 4. Bayesian change point detection. The two innermost rings mirror those in Figure 3.
463 The outermost ring plots the posterior probabilities of change in the k-mer frequencies.
464

Figure 5. The k-mer skim. Here we show the decrease in HGT detection sensitivity as a function of the density of k-mers sampled. The higher the k-mer skim factor (defined as the number of positions skipped before the next k-mer is sampled), the lower the density of k-mers subject to the information bottleneck. The inset shows the plateau behavior near the origin for k-mer skim factors of 1, 5, 10, 25, and 50.

Figure 6. Varying HGT length and divergence. HGT detection rates are shown with respect to increasing HGT length and divergence.

Figure 7. Varying recombination and background mutation rates. We measure the fraction of unique k-mers in each simulation captured by the core genome cluster as a function of recombination rate and background mutation rate. The core genome signal is strongest at low rates of recombination and background mutation. At higher recombination rates, there is no evidence for a core genome of any kind regardless of the background mutation rate.

Figure 8. Modelling ST239's hybridization event. We selected 10 *S. aureus* genomes to track the ST239 hybridization event with the information bottleneck. COL was chosen to represent the CC30 donor strain, and T0131 the CC8 acceptor. Of the 60 clusters we calculated, we show the three that capture the hybridization event. The innermost track is a frequency plot of k-mers that define the core. The second and third tracks are flipsides of the HGT event that created ST239.

Figure 9. Relevance compression curves. In an information bottleneck experiment, the relevance compression curve traces the increase in normalized mutual information with the number of clusters modeled. The curves quantify the amount of information lost at a given modeling threshold. We show how this type of relationship can function as a marker for evolutionary strategy by calculating curves for three very different groups of microbes: *M. tuberculosis*, a species thought to demonstrate little if any HGT; *S. aureus*, a species considered largely clonal with occasional HGT; and H. pylori, a species known to employ HGT as an engine for diversity.

References

1. GenBank and WGS Statistics. https://www.ncbi.nlm.nih.gov/genbank/statistics/.

2. Hug, L. A. *et al.* A new view of the tree of life. *Nat. Microbiol.* **1**, (2016).

3. Rinke, C. *et al.* Insights into the phylogeny and coding potential of microbial dark matter. *Nature* **499**, 431–437 (2013).

4. Sogin, M. L. *et al.* Microbial diversity in the deep sea and the underexplored "rare biosphere". *Proc. Natl. Acad. Sci.* **103**, 12115–12120 (2006).

5. Stephens, Z. D. *et al.* Big Data: Astronomical or Genomical? *PLOS Biol.* **13**, e1002195 (2015).

6. Croucher, N. J. *et al.* Rapid phylogenetic analysis of large samples of recombinant bacterial whole genome sequences using Gubbins. *Nucleic Acids Res.* **43**, e15–e15 (2015).

7. Didelot, X. & Wilson, D. J. ClonalFrameML: Efficient Inference of Recombination in Whole Bacterial Genomes. *PLOS Comput. Biol.* **11**, e1004041 (2015).

8. Chiu, J. C. *et al.* OrthologID: automation of genome-scale ortholog identification within a parsimony framework. *Bioinformatics* **22**, 699–707 (2006).

9. Page, A. J. *et al.* Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics* **31**, 3691–3693 (2015).

10. Zhao, Y. *et al.* PGAP: pan-genomes analysis pipeline. *Bioinformatics* **28**, 416–418 (2012).

11. Marzen Sarah E. & DeDeo Simon. The evolution of lossy compression. *J. R. Soc. Interface* **14**, 20170166 (2017).

12. Tishby, N., Pereira, F. C. & Bialek, W. The information bottleneck method. *arXiv:physics/0004057* (2000).

13. Shannon, C. E. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* **27**, 379–423 (1948).

14. Blei, D. M., Ng, A. Y. & Jordan, M. I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003).

15. Liu, L., Tang, L., Dong, W., Yao, S. & Zhou, W. An overview of topic modeling and its current applications in bioinformatics. *SpringerPlus* **5**, 1608 (2016).

16. La Rosa, M., Fiannaca, A., Rizzo, R. & Urso, A. Probabilistic topic modeling for the analysis and classification of genomic sequences. *BMC Bioinformatics* **16**, S2 (2015).

17. Chen, X., Hu, X., Shen, X. & Rosen, G. Probabilistic topic modeling for genomic data interpretation. in *2010 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* 149–152 (2010). doi:10.1109/BIBM.2010.5706554.

18. Pritchard, J. K., Stephens, M. & Donnelly, P. Inference of Population Structure Using Multilocus Genotype Data. *Genetics* **155**, 945–959 (2000).

19. Wang, J. The computer program structure for assigning individuals to populations: easy to use but easier to misuse. *Mol. Ecol. Resour.* **17**, 981–990 (2017).

20. Wallach, H. M., Mimno, D. M. & McCallum, A. Rethinking LDA: Why Priors Matter. in *Advances in Neural Information Processing Systems 22* (eds. Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I. & Culotta, A.) 1973–1981 (Curran Associates, Inc., 2009).

21. Sims, G. E., Jun, S.-R., Wu, G. A. & Kim, S.-H. Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions. *Proc. Natl. Acad. Sci.* **106**, 2677–2682 (2009).

22. Cong, Y., Chan, Y. & Ragan, M. A. A novel alignment-free method for detection of lateral genetic transfer based on TF-IDF. *Sci. Rep.* **6**, 30308 (2016).

544 23. Cong, Y., Chan, Y. & Ragan, M. A. Exploring lateral genetic transfer among microbial
545 genomes using TF-IDF. *Sci. Rep.* **6**, 29319 (2016).
546 24. Polz, M. F., Alm, E. J. & Hanage, W. P. Horizontal gene transfer and the evolution of
547 bacterial and archaeal population structure. *Trends Genet.* **29**, 170–175 (2013).
548 25. Planet, P. J. Reexamining microbial evolution through the lens of horizontal transfer. *EXS*
549 247–303 (2002).
550 26. Tettelin, H. *et al.* Genome analysis of multiple pathogenic isolates of Streptococcus
551 agalactiae: Implications for the microbial "pan-genome". *Proc. Natl. Acad. Sci.* **102**, 13950–
552 13955 (2005).
553 27. Pereira, F., Tishby, N. & Lee, L. Distributional Clustering of English Words. in
554 *Proceedings of the 31st Annual Meeting on Association for Computational Linguistics* 183–
555 190 (Association for Computational Linguistics, 1993). doi:10.3115/981574.981598.
556 28. Slonim, N. The Information Bottleneck: Theory and Applications. *Dr. Diss. Hebr. Univ.*
557 *Jerus. Isr. 2003* 157.
558 29. Kullback, S. & Leibler, R. A. On Information and Sufficiency. *Ann. Math. Stat.* **22**, 79–
559 86 (1951).
560 30. Slonim, N. & Tishby, N. Agglomerative Information Bottleneck. in *Proceedings of the*
561 *12th International Conference on Neural Information Processing Systems* 617–623 (MIT
562 Press, 1999).
563 31. Slonim, N., Friedman, N. & Tishby, N. Unsupervised Document Classification Using
564 Sequential Information Maximization. in *Proceedings of the 25th Annual International ACM*
565 *SIGIR Conference on Research and Development in Information Retrieval* 129–136 (ACM,
566 2002). doi:10.1145/564376.564401.

32. Lin, J. Divergence measures based on the Shannon entropy. *IEEE Trans. Inf. Theory* **37**, 145–151 (1991).

33. Still, S. & Bialek, W. How Many Clusters? An Information-Theoretic Perspective. *Neural Comput* **16**, 2483–2506 (2004).

34. Brown, T., Didelot, X., Wilson, D. J. & Maio, N. D. SimBac: simulation of whole bacterial genomes with homologous recombination. *Microb. Genomics* **2**, (2016).

35. Didelot, X. & Maiden, M. C. J. Impact of recombination on bacterial evolution. *Trends Microbiol.* **18**, 315–322 (2010).

36. Truong, C., Oudre, L. & Vayatis, N. Selective review of offline change point detection methods. *Signal Process.* **167**, 107299 (2020).

37. Barry, D. & Hartigan, J. A. A Bayesian Analysis for Change Point Problems. *J. Am. Stat. Assoc.* **88**, 309 (1993).

38. Robinson, D. A. & Enright, M. C. Evolution of Staphylococcus aureus by Large Chromosomal Replacements. *J. Bacteriol.* **186**, 1060–1064 (2004).

39. Narechania, A. *et al.* Clusterflock: a flocking algorithm for isolating congruent phylogenomic datasets. *GigaScience* **5**, (2016).

40. Slonim, N., Atwal, G. S., Tkačik, G. & Bialek, W. Information-based clustering. *Proc. Natl. Acad. Sci.* **102**, 18297–18302 (2005).

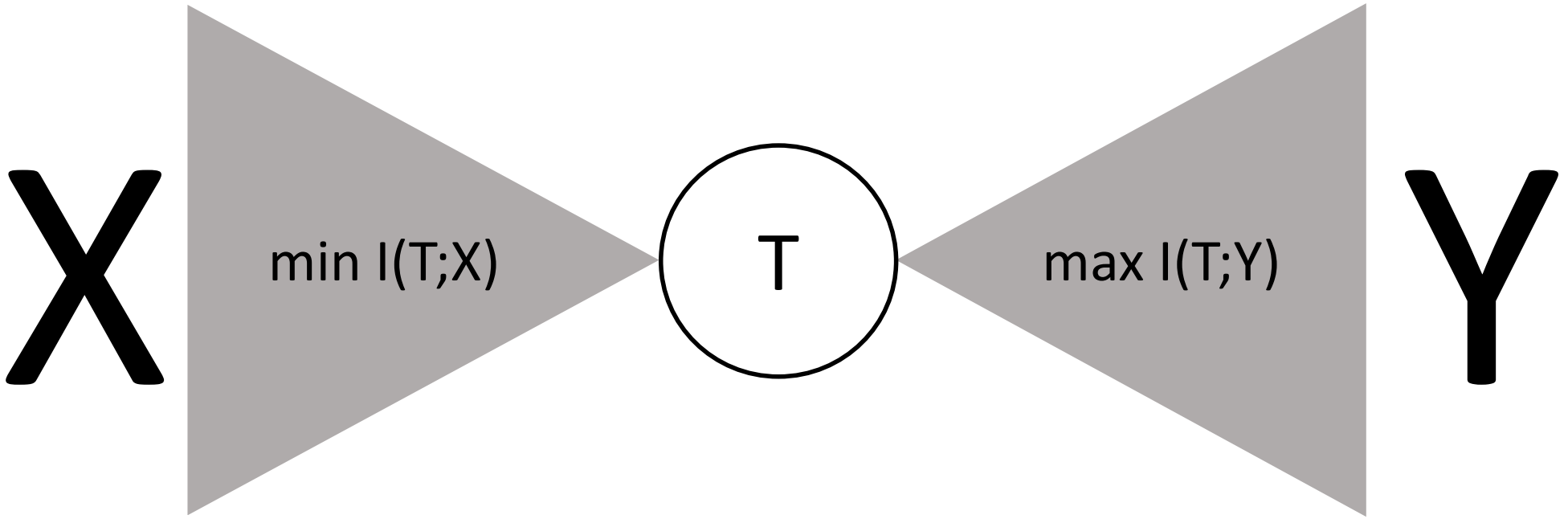
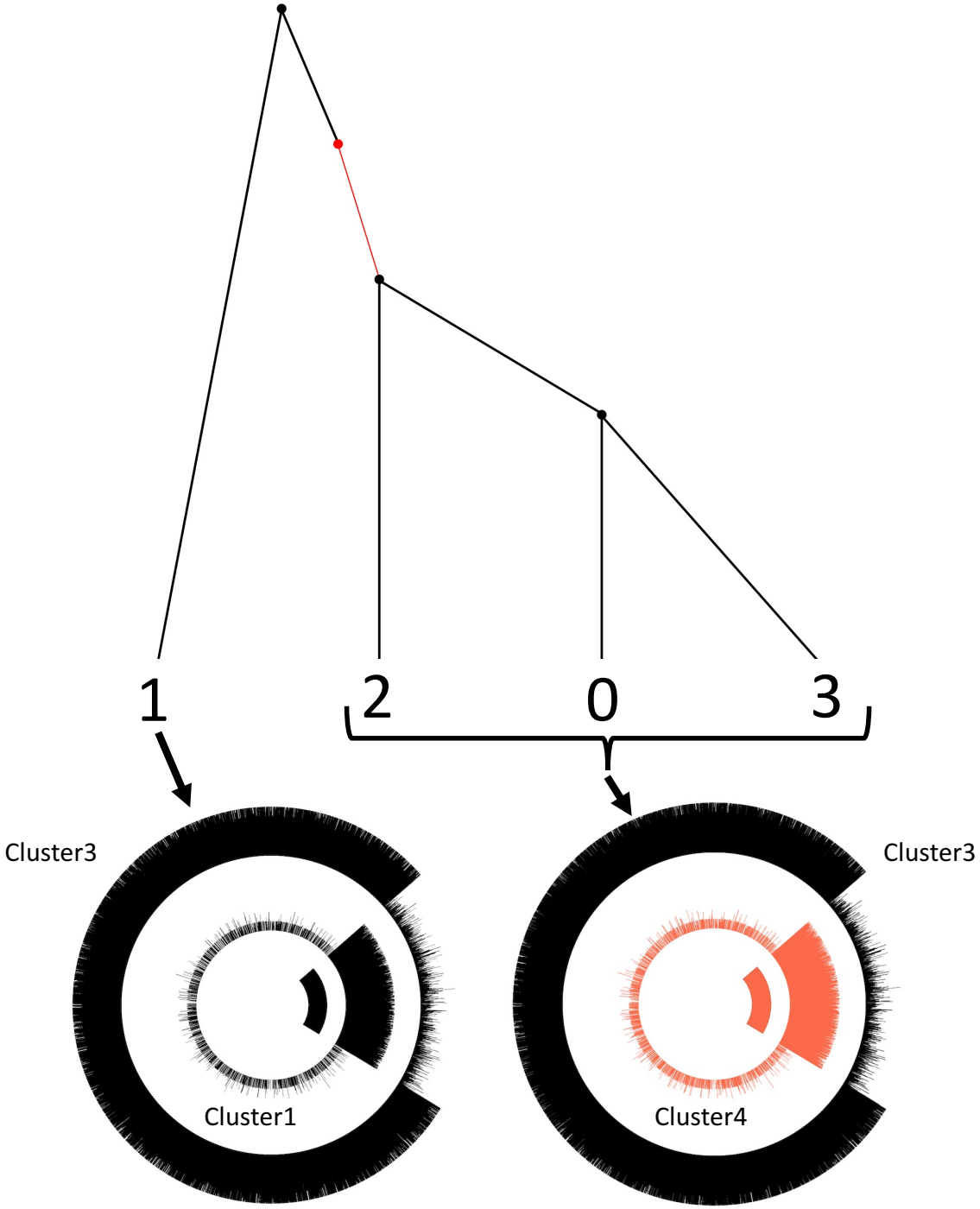


Figure 1



Initialize	genome0	genome1	genome2	genome3
CLUST0	199366	199350	199437	199282
CLUST1	200184	200439	200134	200194
CLUST2	199693	199696	199591	199808
CLUST3	200765	200718	200857	200785
CLUST4	199974	199779	199963	199913
NMI = 0				
loop 1				
CLUST0	3270	3954	4079	0
CLUST1	15808	237766	19369	15327
CLUST2	9348	10535	1175	10552
CLUST3	746003	747727	747728	747728
CLUST4	225553	0	227631	226375
NMI = 0.69				
loop 2				
CLUST0	7093	19812	50361	0
CLUST1	0	206335	0	0
CLUST2	28277	21421	0	37257
CLUST3	746103	752414	752413	752314
CLUST4	218509	0	197208	210411
NMI = 0.89				
loop 3				
CLUST0	0	12719	43268	0
CLUST1	0	206335	0	0
CLUST2	48715	21421	0	51754
CLUST3	753196	759507	759506	752314
CLUST4	198071	0	197208	195914
NMI = 0.91				

Figure 2

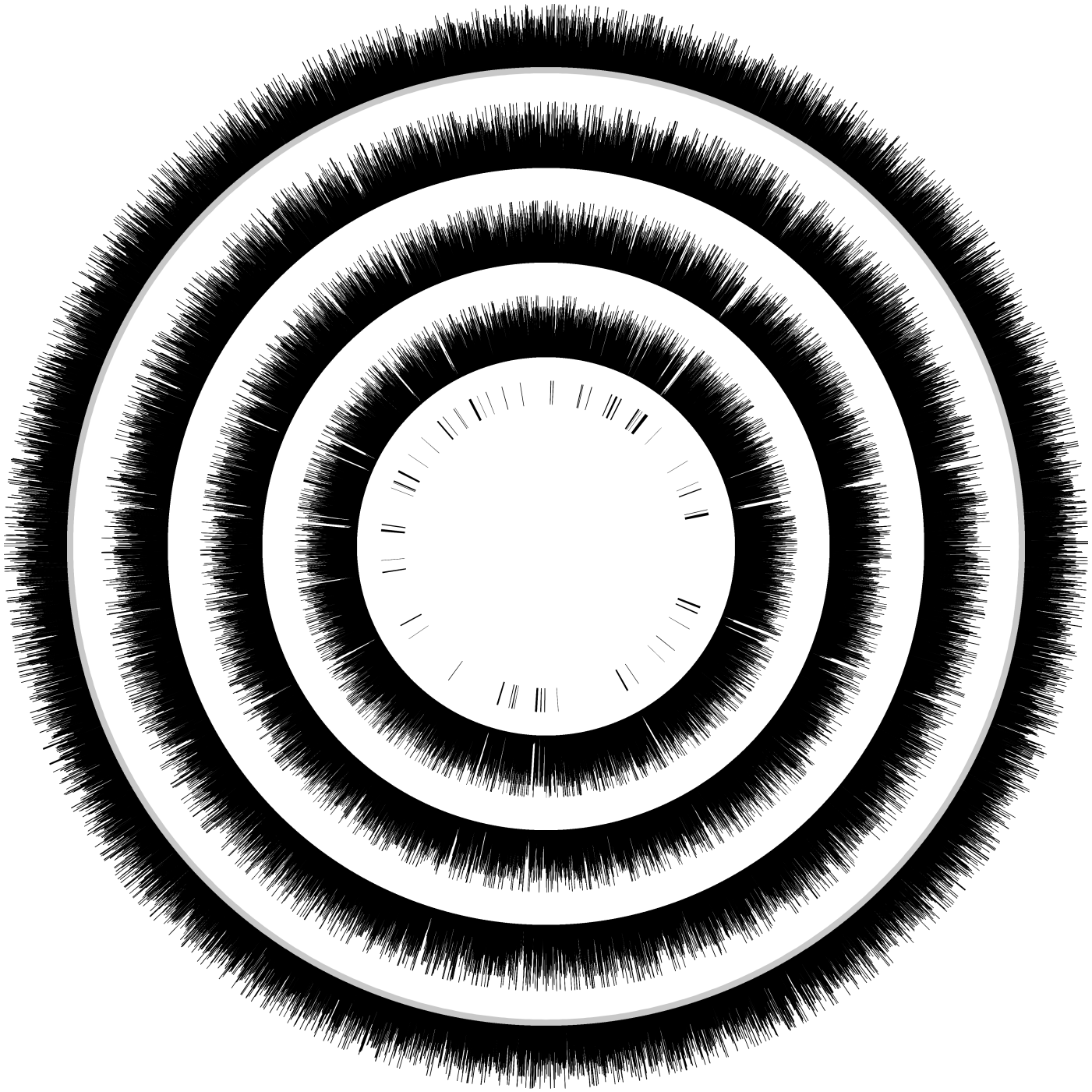


Figure 3

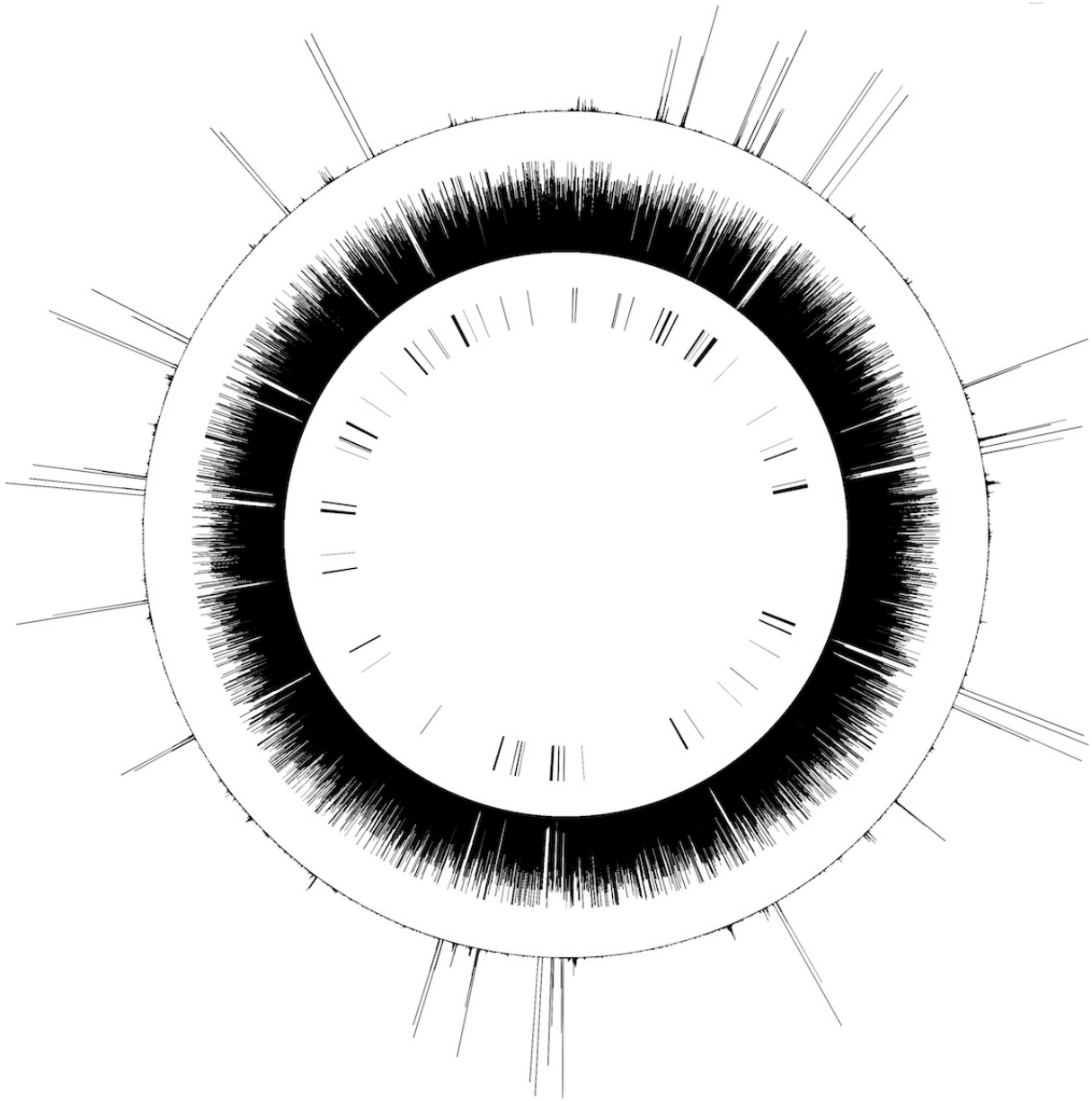


Figure 4

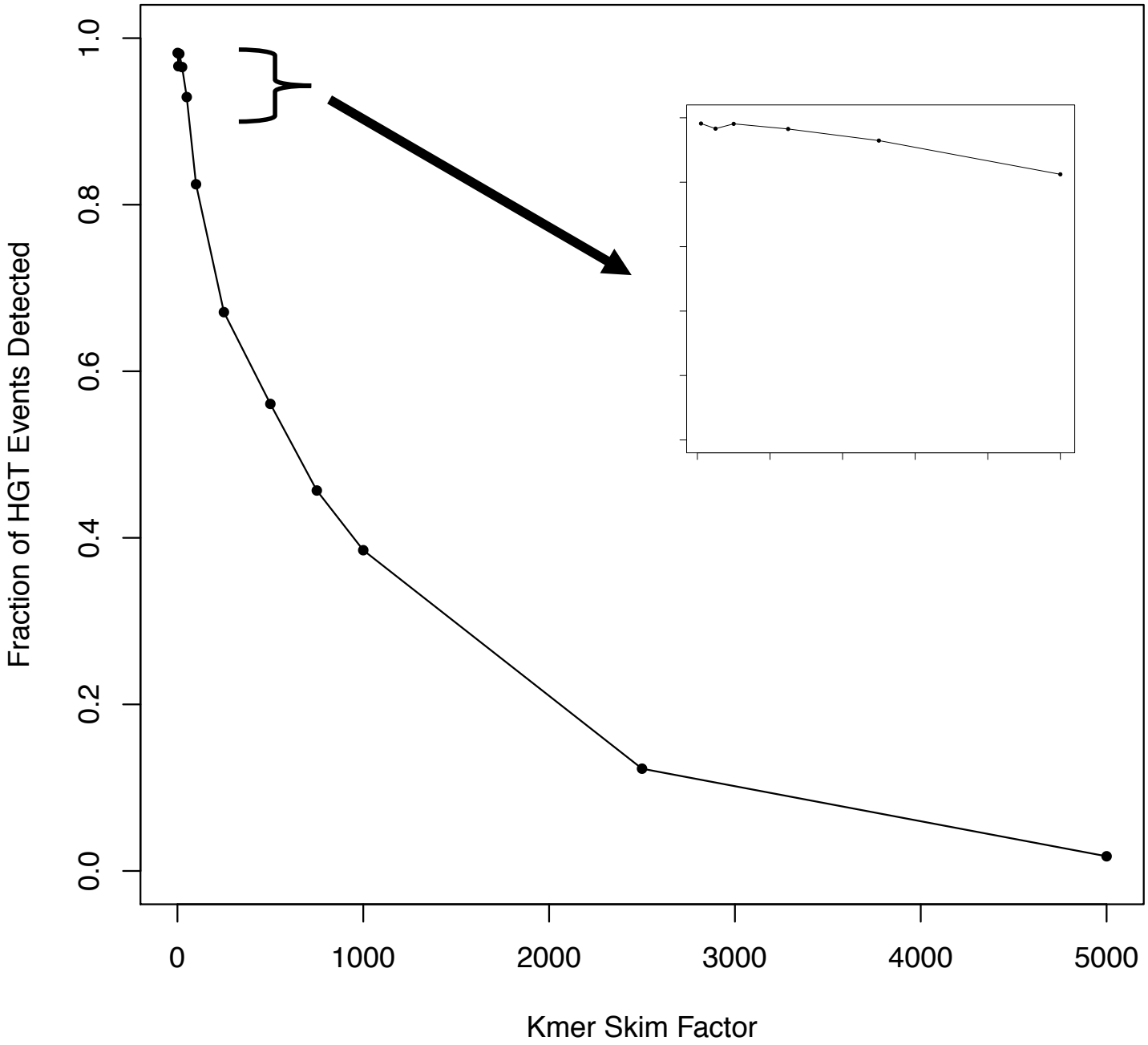

Fraction of HGT Events Detected
1.0
0.8
0.6
0.4
0.2
0.0
0
1000
2000
3000
4000
5000
Kmer Skim Factor

Figure 5

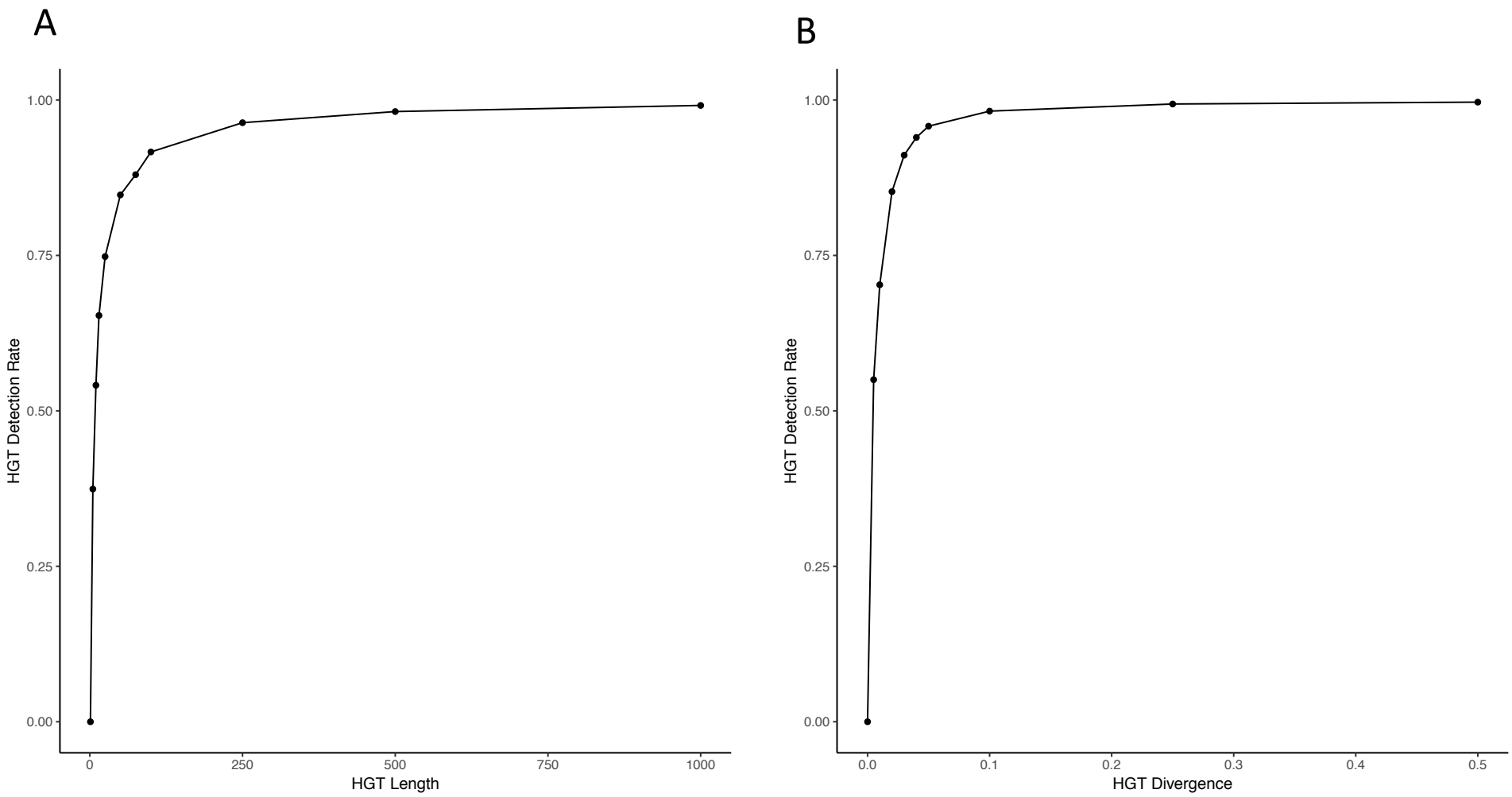

A
B
HGT Detection Rate
1.00
0.75
0.50
0.25
0.00
0
250
500
750
1000
HGT Length
0.0
0.1
0.2
0.3
0.4
0.5
HGT Divergence

Figure 6

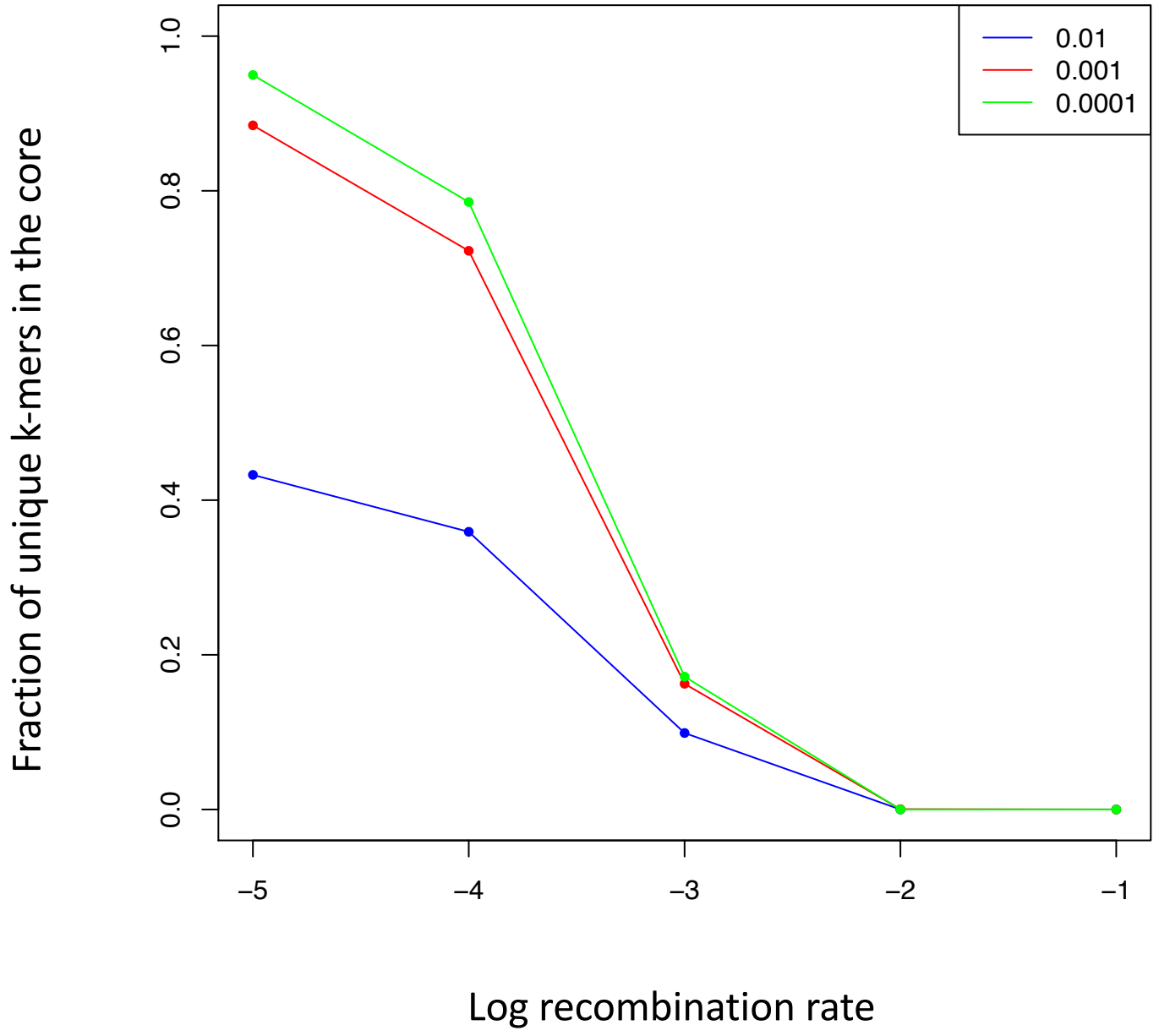
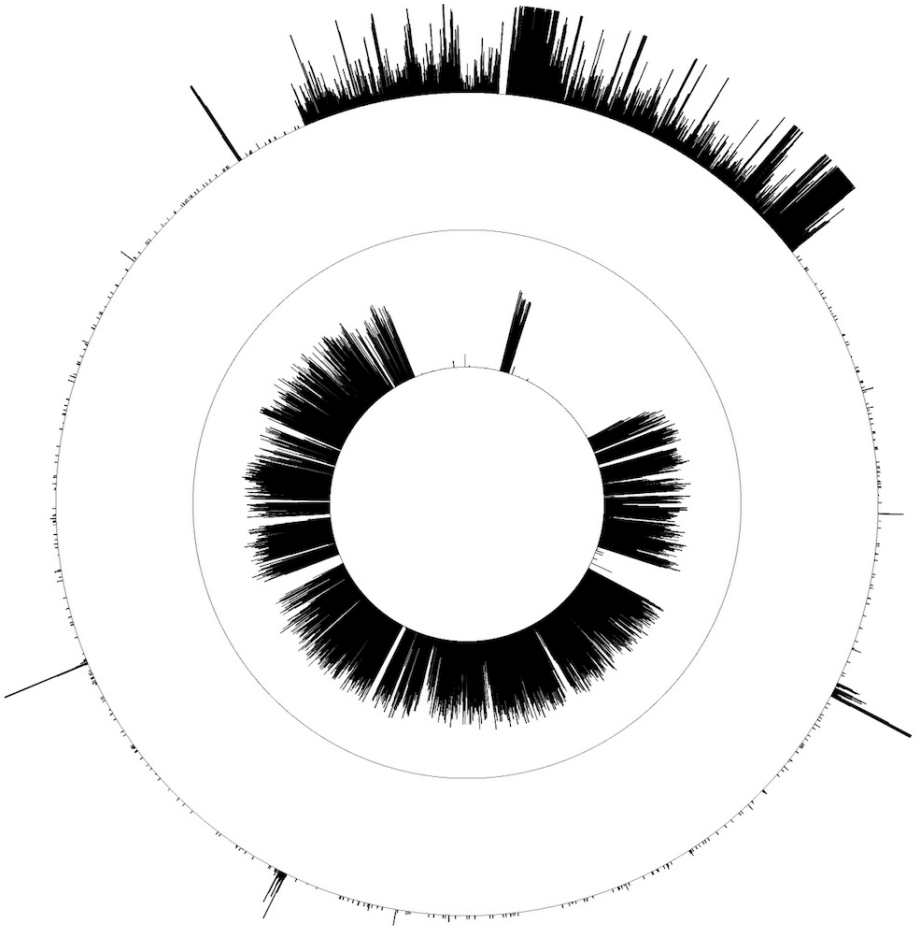
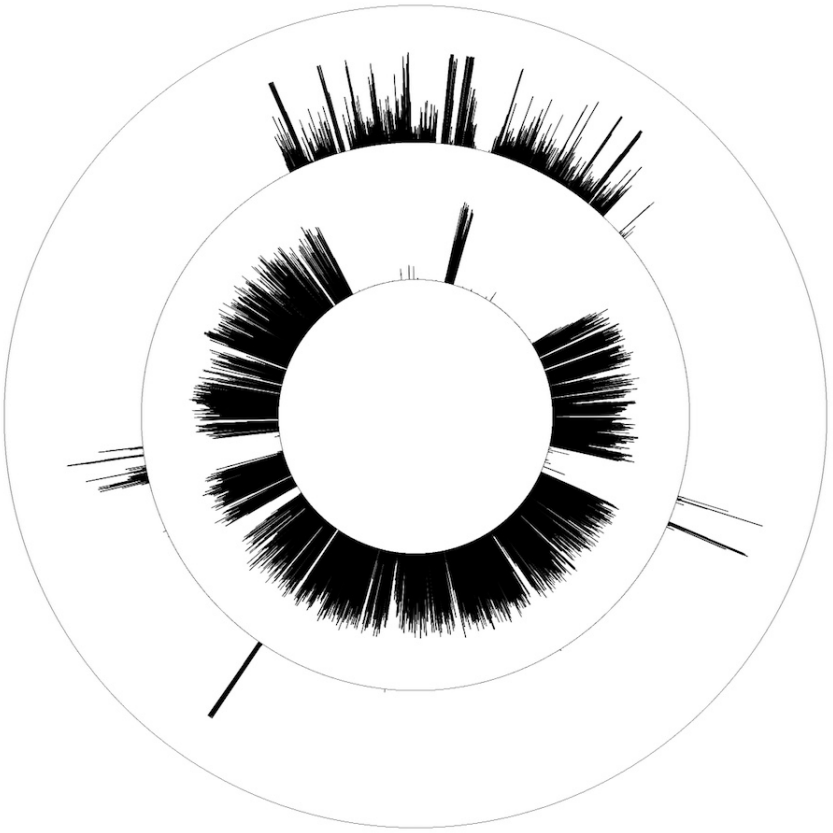

Fraction of unique k-mers in the core
0.0
0.2
0.4
0.6
0.8
1.0
−5
−4
−3
−2
−1
Log recombination rate
0.01
0.001
0.0001

Figure 7



SaCOL



SaT0131

Figure 8

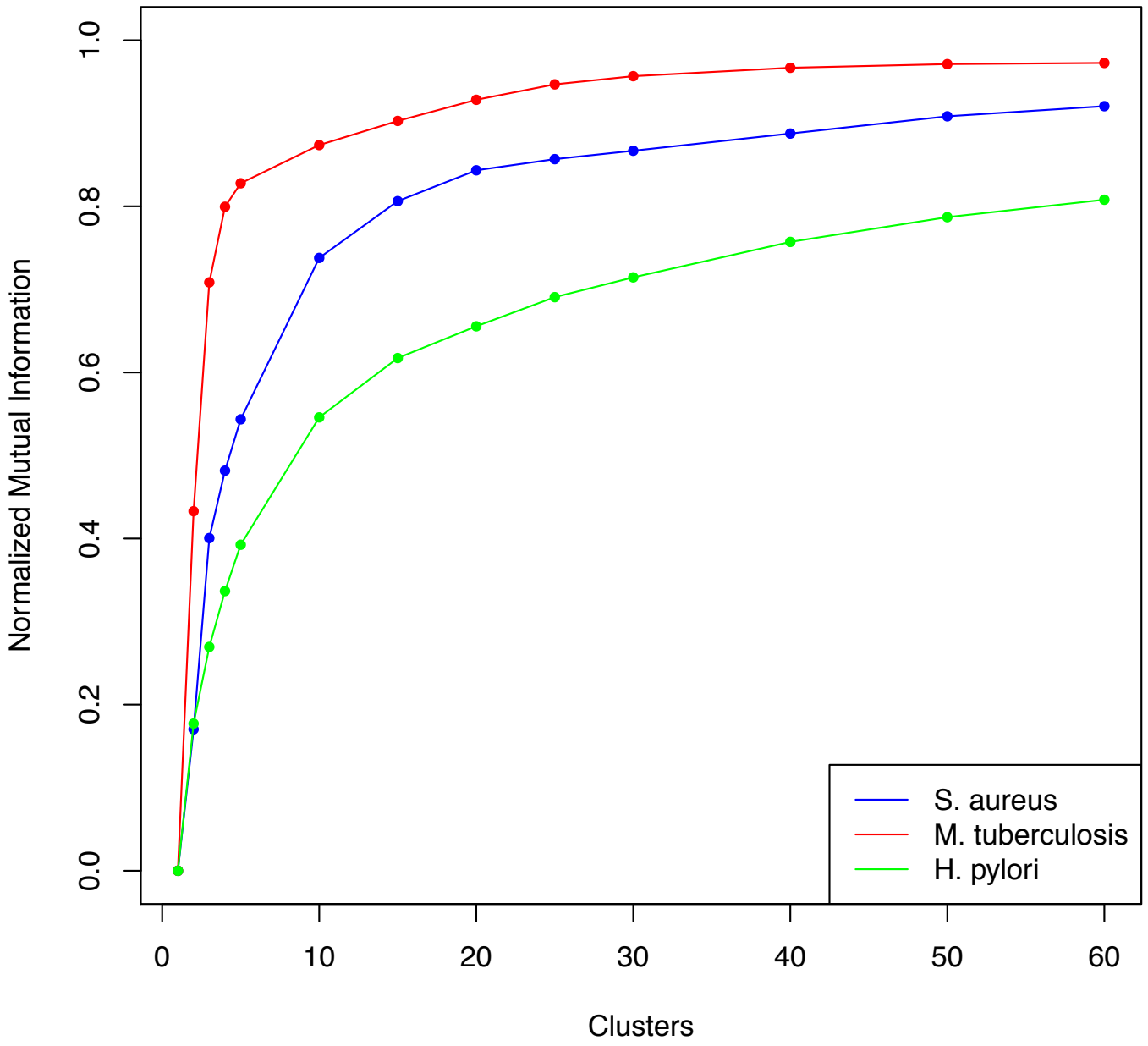

Normalized Mutual Information
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50
60
Clusters
S. aureus
M. tuberculosis
H. pylori

Figure 9